

# Introduction to Survival Analysis

## Contents

<b>1</b>	<b>Introduction to Survival Analysis</b>	<b>2</b>
	Configuring R . . . . .	2
1.1	Overview . . . . .	3
1.1.1	Time-to-event outcomes . . . . .	3
1.2	Time-to-event outcome distributions . . . . .	3
1.2.1	Distributions of Time-to-Event Data . . . . .	3
1.2.2	Right Censoring . . . . .	3
1.2.3	Left and Interval Censoring . . . . .	3
1.3	Distribution functions for time-to-event variables . . . . .	3
1.3.1	The Probability Density Function (PDF) . . . . .	3
1.3.2	The Cumulative Distribution Function (CDF) . . . . .	4
1.3.3	The Survival Function . . . . .	5
1.3.4	The Hazard Function . . . . .	7
1.3.5	The Cumulative Hazard Function . . . . .	9
1.3.6	Some Key Mathematical Relationships among Survival Concepts . . . . .	10
1.3.7	Likelihood with censoring . . . . .	15
1.4	Parametric Models for Time-to-Event Outcomes . . . . .	17
1.4.1	Exponential Distribution . . . . .	17
1.4.2	Other Parametric Survival Distributions . . . . .	19
1.5	Nonparametric Survival Analysis . . . . .	20
1.5.1	Basic ideas . . . . .	20
1.6	Example: clinical trial for pediatric acute leukemia . . . . .	20
1.6.1	Overview of study . . . . .	20
1.6.2	Study design . . . . .	20
1.6.3	Data documentation for <code>drug6mp</code> . . . . .	20
1.6.4	Descriptive Statistics . . . . .	20
1.6.5	Exponential model . . . . .	22
1.7	The Kaplan-Meier Product Limit Estimator . . . . .	22
1.7.1	Estimating survival in datasets without censoring . . . . .	22
1.7.2	Estimating survival in datasets with censoring . . . . .	22
1.7.3	Kaplan-Meier curve for <code>drug6mp</code> data . . . . .	23
1.7.4	Kaplan-Meier calculations . . . . .	24
1.8	Using the <code>survival</code> package in R . . . . .	26
1.8.1	The <code>Surv</code> function . . . . .	26
1.8.2	The <code>survfit</code> function . . . . .	27
1.8.3	Plotting estimated survival functions . . . . .	28
1.9	The log-rank test . . . . .	29
1.9.1	The <code>survdif</code> function . . . . .	30
1.9.2	Example: <code>survdif()</code> with <code>drug6mp</code> data . . . . .	30
1.10	Example: Bone Marrow Transplant Data . . . . .	34
1.10.1	Study design . . . . .	34
1.10.2	<code>KMsurv::bmt</code> data in R . . . . .	35
1.10.3	Analysis plan . . . . .	35
1.10.4	Survival Function Estimate and Variance . . . . .	35
1.10.5	Understanding Greenwood's formula (optional) . . . . .	36

1.10.6	Test for differences among the disease groups . . . . .	37
1.10.7	Cumulative Hazard . . . . .	37
1.11	Nelson-Aalen Estimates of Cumulative Hazard and Survival . . . . .	39
1.11.1	Application to bmt dataset . . . . .	39

# 1 Introduction to Survival Analysis

---

## Configuring R

Functions from these packages will be used throughout this document:

```
library(conflicted) # check for conflicting function definitions
# library(printr) # inserts help-file output into markdown output
library(rmarkdown) # Convert R Markdown documents into a variety of formats.
library(pander) # format tables for markdown
library(ggplot2) # graphics
library(ggfortify) # help with graphics
library(dplyr) # manipulate data
library(tibble) # `tibble`s extend `data.frame`s
library(magrittr) # `%>%` and other additional piping tools
library(haven) # import Stata files
library(knitr) # format R output for markdown
library(tidyr) # Tools to help to create tidy data
library(plotly) # interactive graphics
library(dobson) # datasets from Dobson and Barnett 2018
library(parameters) # format model output tables for markdown
library(haven) # import Stata files
library(latex2exp) # use LaTeX in R code (for figures and tables)
library(fs) # filesystem path manipulations
library(survival) # survival analysis
library(survminer) # survival analysis graphics
library(KMsurv) # datasets from Klein and Moeschberger
library(parameters) # format model output tables for
library(webshot2) # convert interactive content to static for pdf
library(forcats) # functions for categorical variables ("factors")
library(stringr) # functions for dealing with strings
library(lubridate) # functions for dealing with dates and times
```

Here are some R settings I use in this document:

```
rm(list = ls()) # delete any data that's already loaded into R

conflicts_prefer(dplyr::filter)
ggplot2::theme_set(
  ggplot2::theme_bw() +
    # ggplot2::labs(col = "") +
    ggplot2::theme(
      legend.position = "bottom",
      text = ggplot2::element_text(size = 12, family = "serif"))
)

knitr::opts_chunk$set(message = FALSE)
options('digits' = 6)

panderOptions("big.mark", ",")
pander::panderOptions("table.emphasize.rownames", FALSE)
pander::panderOptions("table.split.table", Inf)
conflicts_prefer(dplyr::filter) # use the `filter()` function from dplyr() by default
```

```
legend_text_size = 9
run_graphs = TRUE
```

## 1.1 Overview

### 1.1.1 Time-to-event outcomes

**Survival analysis** is a framework for modeling *time-to-event* outcomes. It is used in:

- clinical trials, where the event is often death or recurrence of disease.
- engineering reliability analysis, where the event is failure of a device or system.
- insurance, particularly life insurance, where the event is death.

#### Note

The term *survival analysis* is a bit misleading. Survival outcomes can sometimes be analyzed using binomial models (logistic regression). *Time-to-event models* or *survival time analysis* might be a better name.

## 1.2 Time-to-event outcome distributions

### 1.2.1 Distributions of Time-to-Event Data

- The distribution of event times is asymmetric and can be long-tailed, and starts at 0 (that is,  $P(T < 0) = 0$ ).
- The base distribution is not normal, but exponential.
- There are usually **censored** observations, which are ones in which the failure time is not observed.
- Often, these are **right-censored**, meaning that we know that the event occurred after some known time  $t$ , but we don't know the actual event time, as when a patient is still alive at the end of the study.
- Observations can also be **left-censored**, meaning we know the event has already happened at time  $t$ , or **interval-censored**, meaning that we only know that the event happened between times  $t_1$  and  $t_2$ .
- Analysis is difficult if censoring is associated with treatment.

### 1.2.2 Right Censoring

- Patients are in a clinical trial for cancer, some on a new treatment and some on standard of care.
- Some patients in each group have died by the end of the study. We know the survival time (measured for example from time of diagnosis—each person on their own clock).
- Patients still alive at the end of the study are right censored.
- Patients who are lost to follow-up or withdraw from the study may be right-censored.

### 1.2.3 Left and Interval Censoring

- An individual tests positive for HIV.
- If the event is infection with HIV, then we only know that it has occurred before the testing time  $t$ , so this is left censored.
- If an individual has a negative HIV test at time  $t_1$  and a positive HIV test at time  $t_2$ , then the infection event is interval censored.

## 1.3 Distribution functions for time-to-event variables

### 1.3.1 The Probability Density Function (PDF)

For a time-to-event variable  $T$  with a continuous distribution, the **probability density function** is defined as usual (see [?@sec-prob-dens](#)).

In most time-to-event models, this density is assumed to be 0 for all  $t < 0$ ; that is,  $f(t) = 0, \forall t < 0$ . In other words, the support of  $T$  is typically  $[0, \infty)$ .

---

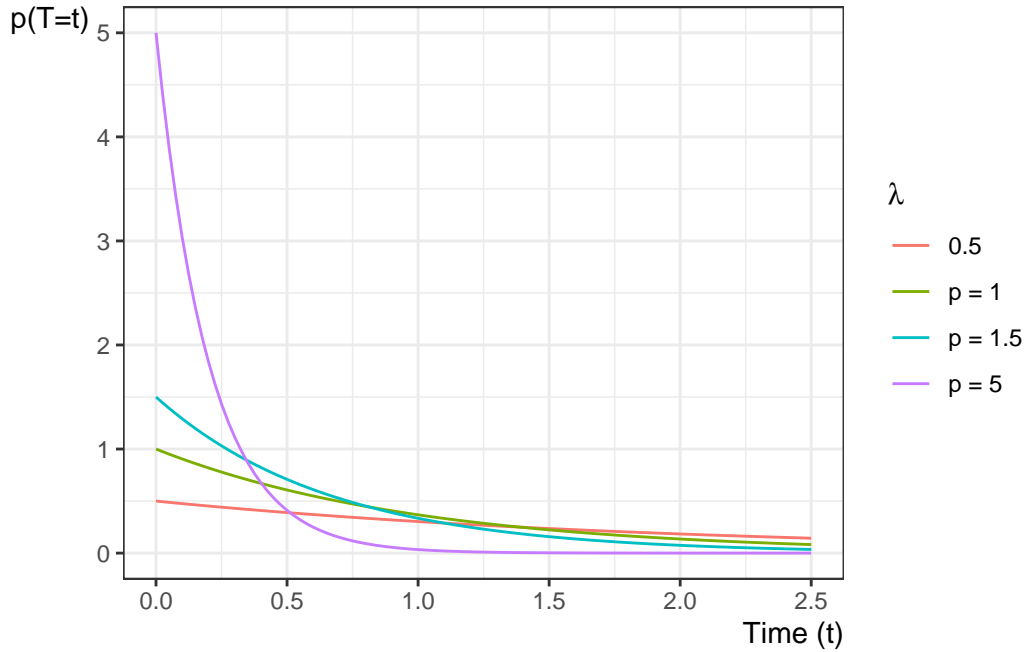
**Example 1.1** (exponential distribution). Recall from Epi 202: the pdf of the exponential distribution family of models is:

$$p(T = t) = 1_{t \geq 0} \cdot \lambda e^{-\lambda t}$$

where  $\lambda > 0$ .

---

Here are some examples of exponential pdfs:



### 1.3.2 The Cumulative Distribution Function (CDF)

The **cumulative distribution function** is defined as:

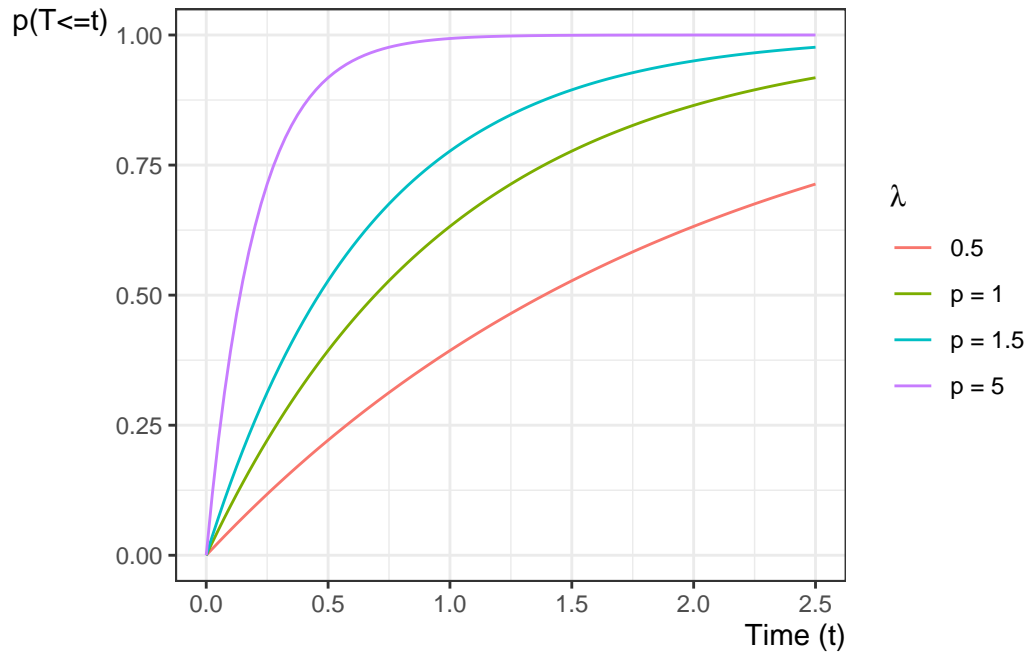
$$\begin{aligned} F(t) &\stackrel{\text{def}}{=} \Pr(T \leq t) \\ &= \int_{u=-\infty}^t f(u) du \end{aligned}$$

**Example 1.2** (exponential distribution). Recall from Epi 202: the cdf of the exponential distribution family of models is:

$$P(T \leq t) = 1_{t \geq 0} \cdot (1 - e^{-\lambda t})$$

where  $\lambda > 0$ .

Here are some examples of exponential cdfs:



### 1.3.3 The Survival Function

For survival data, a more important quantity is the **survival function**:

**Definition 1.1** (Survival function).

Given a random time-to-event variable  $T$ , the **survival function** or **survivor function**, denoted  $S(t)$ , is the probability that the event time is later than  $t$ . If the event in a clinical trial is death, then  $S(t)$  is the expected fraction of the original population at time 0 who have survived up to time  $t$  and are still alive at time  $t$ ; that is:

$$S(t) \stackrel{\text{def}}{=} \Pr(T > t)$$

**Theorem 1.1.**

$$\begin{aligned} S(t) &\stackrel{\text{def}}{=} \Pr(T > t) \\ &= \int_{u=t}^{\infty} p(u) du \\ &= 1 - F(t) \end{aligned}$$

**Example 1.3** (exponential distribution). Since  $S(t) = 1 - F(t)$ , the survival function of the exponential distribution family of models is:

$$P(T > t) = \begin{cases} e^{-\lambda t}, & t \geq 0 \\ 1, & t \leq 0 \end{cases}$$

where  $\lambda > 0$ .

Figure 1 shows some examples of exponential survival functions.

```
library(ggplot2)
ggplot() +
  geom_function(
    aes(col = "0.5"),
    fun = pexp,
```

```

    args = list(lower = FALSE, rate = 0.5)
  ) +
  geom_function(
    aes(col = "p = 1"),
    fun = pexp,
    args = list(lower = FALSE, rate = 1)
  ) +
  geom_function(
    aes(col = "p = 1.5"),
    fun = pexp,
    args = list(lower = FALSE, rate = 1.5)
  ) +
  geom_function(
    aes(col = "p = 5"),
    fun = pexp,
    args = list(lower = FALSE, rate = 5)
  ) +
  theme_bw() +
  ylab("S(t)") +
  guides(col = guide_legend(title = expr(lambda))) +
  xlab("Time (t)") +
  xlim(0, 2.5) +
  theme(
    legend.position = "bottom",
    axis.title.x =
      element_text(
        angle = 0,
        vjust = 1,
        hjust = 1
      ),
    axis.title.y =
      element_text(
        angle = 0,
        vjust = 1,
        hjust = 1
      )
  )
)

```

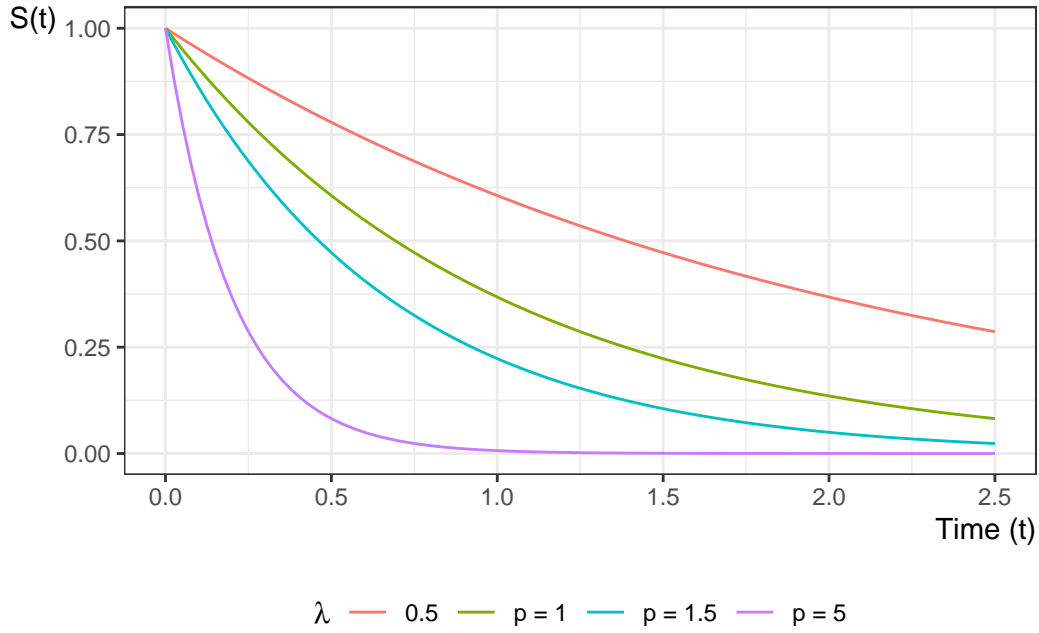


Figure 1: Exponential Survival Functions

---

**Theorem 1.2.** If  $A_t$  represents survival status at time  $t$ , with  $A_t = 1$  denoting alive at time  $t$  and  $A_t = 0$  denoting deceased at time  $t$ , then:

$$S(t) = \Pr(A_t = 1) = E[A_t]$$


---

**Theorem 1.3.** If  $T$  is a nonnegative random variable, then:

$$E[T] = \int_{t=0}^{\infty} S(t) dt$$


---

*Proof.* See <https://statproofbook.github.io/P/mean-nnrvar.html> or □

### 1.3.4 The Hazard Function

Another important quantity is the **hazard function**:

**Definition 1.2** (Hazard function, hazard rate, hazard rate function).

The **hazard function**, **hazard rate**, **hazard rate function**, for a random variable  $T$  at value  $t$ , typically denoted as  $h(t)$ <sup>1</sup> or  $\lambda(t)$ ,<sup>2</sup> is the conditional density<sup>3</sup> of  $T$  at  $t$ , given  $T \geq t$ . That is:

$$\lambda(t) \stackrel{\text{def}}{=} p(T = t | T \geq t)$$

If  $T$  represents the time at which an event occurs, then  $\lambda(t)$  is the probability that the event occurs at time  $t$ , given that it has not occurred prior to time  $t$ .

---

<sup>1</sup>for example in Dobson and Barnett (2018), Vittinghoff et al. (2012), Klein and Moeschberger (2003), and Kleinbaum and Klein (2012)

<sup>2</sup>for example, in Rothman et al. (2021) and Kalbfleisch and Prentice (2011)

<sup>3</sup>[probability.qmd#def-pdf](#)

**Definition 1.3** (Incidence rate). Given a population of  $N$  individuals indexed by  $i$ , each with their own hazard rate  $\lambda_i(t)$ , the **incidence rate** for that population is the mean hazard rate:

$$\bar{\lambda}(t) \stackrel{\text{def}}{=} \frac{1}{N} \sum_{i=1}^N \lambda_i(t)$$


---

**Theorem 1.4** (Incidence rate in a homogenous population). *If a population of individuals indexed by  $i$  all have identical hazard rates  $\lambda_i(t) = \lambda(t)$ , then the **incidence rate** for that population is equal to the hazard rate:*

$$\bar{\lambda}(t) = \lambda(t)$$


---

The hazard function has an important relationship to the density and survival functions, which we can use to derive the hazard function for a given probability distribution (Theorem 1.5).

**Lemma 1.1** (Joint probability of a variable with itself).

$$p(T = t, T \geq t) = p(T = t)$$

*Proof.* Recall from Epi 202: if  $A$  and  $B$  are statistical events and  $A \subseteq B$ , then  $p(A, B) = p(A)$ . In particular,  $\{T = t\} \subseteq \{T \geq t\}$ , so  $p(T = t, T \geq t) = p(T = t)$ .  $\square$

---

**Theorem 1.5** (Hazard equals density over survival).

$$\lambda(t) = \frac{f(t)}{S(t)}$$


---

*Proof.*

$$\begin{aligned} \lambda(t) &= p(T = t | T \geq t) \\ &= \frac{p(T = t, T \geq t)}{p(T \geq t)} \\ &= \frac{p(T = t)}{p(T \geq t)} \\ &= \frac{f(t)}{S(t)} \end{aligned}$$

$\square$

---

**Example 1.4** (exponential distribution). The hazard function of the exponential distribution family of models is:

$$\begin{aligned} P(T = t | T \geq t) &= \frac{f(t)}{S(t)} \\ &= \frac{\mathbb{1}_{t \geq 0} \cdot \lambda e^{-\lambda t}}{e^{-\lambda t}} \\ &= \mathbb{1}_{t \geq 0} \cdot \lambda \end{aligned}$$

Figure 2 shows some examples of exponential hazard functions.

---



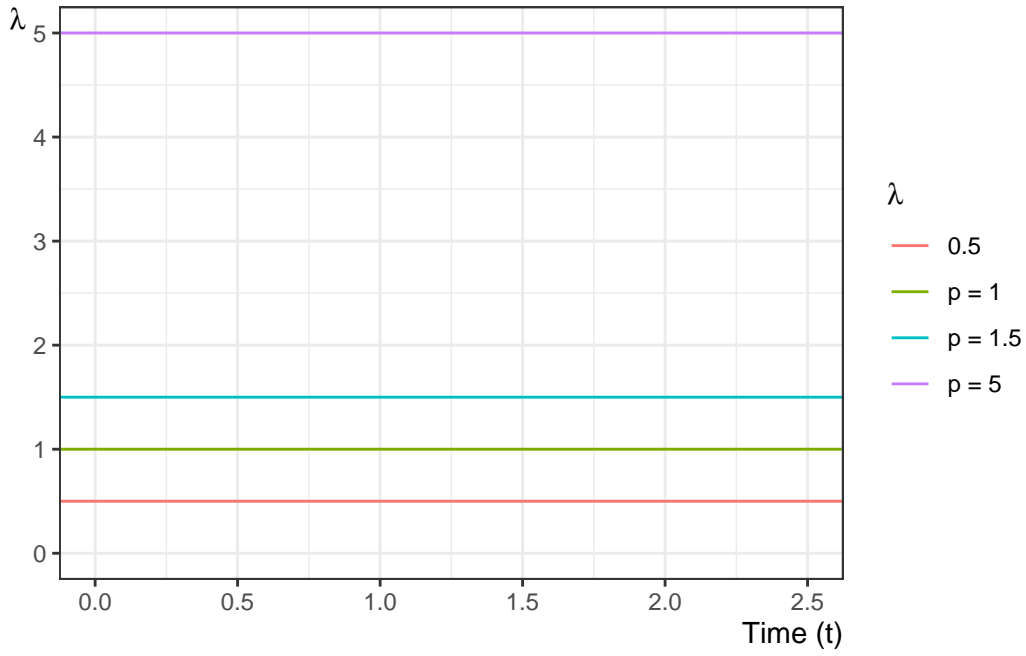


Figure 2: Examples of hazard functions for exponential distributions

---

We can also view the hazard function as the derivative of the negative of the logarithm of the survival function:

**Theorem 1.6** (transform survival to hazard).

$$\lambda(t) = \frac{\partial}{\partial t} \{-\log S(t)\}$$


---

*Proof.*

$$\begin{aligned} \lambda(t) &= \frac{f(t)}{S(t)} \\ &= \frac{-S'(t)}{S(t)} \\ &= -\frac{S'(t)}{S(t)} \\ &= -\frac{\partial}{\partial t} \log S(t) \\ &= \frac{\partial}{\partial t} \{-\log S(t)\} \end{aligned}$$

□

---

**Definition 1.4** (hazard ratio).

$$\theta(t|\tilde{x} : \tilde{x}^*) \stackrel{\text{def}}{=} \frac{\lambda(t|\tilde{x})}{\lambda(t|\tilde{x}^*)}$$

### 1.3.5 The Cumulative Hazard Function

Since  $\lambda(t) = \frac{\partial}{\partial t} \{-\log S(t)\}$  (see Theorem 1.6), we also have:

**Corollary 1.1.**

$$S(t) = \exp\left\{-\int_{u=0}^t \lambda(u)du\right\} \quad (1)$$

The integral in Equation 1 is important enough to have its own name: **cumulative hazard**.

**Definition 1.5** (cumulative hazard). The **cumulative hazard function**, often denoted  $\Lambda(t)$  or  $H(t)$ , is defined as:

$$\Lambda(t) \stackrel{\text{def}}{=} \int_{u=0}^t \lambda(u)du$$

As we will see below,  $\Lambda(t)$  is tractable to estimate, and we can then derive an estimate of the hazard function using an approximate derivative of the estimated cumulative hazard.

**Example 1.5.** The cumulative hazard function for the exponential distribution with rate parameter  $\lambda$  is:

$$\Lambda(t) = \mathbb{1}_{t \geq 0} \cdot \lambda t$$

Figure 3 shows some examples of exponential cumulative hazard functions.

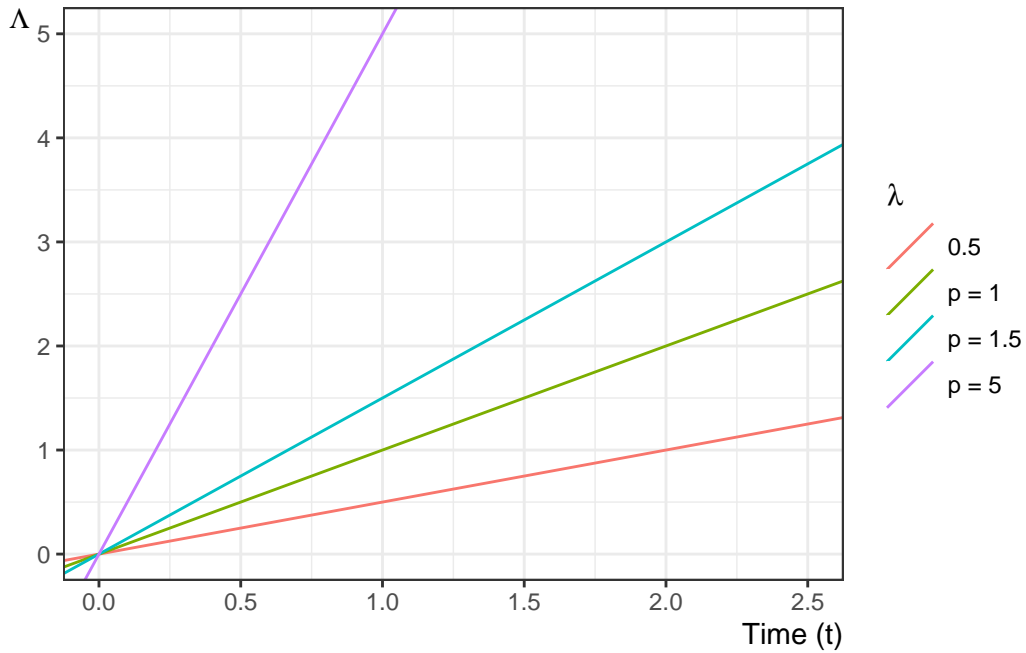


Figure 3: Examples of exponential cumulative hazard functions

### 1.3.6 Some Key Mathematical Relationships among Survival Concepts

**Diagram:**

$$f(t) \xleftarrow[\frac{-S'(t)}{S(t)\lambda(t)}]{} S(t) \xleftarrow[\exp\{-\Lambda(t)\}]{} \Lambda(t) \xleftarrow[\int_{u=0}^t \lambda(u)du]{} \lambda(t) \xleftarrow[\exp\{\eta(t)\}]{} \eta(t)$$

$$f(t) \xrightarrow[\int_{u=t}^{\infty} f(u) du]{f(t)/\lambda(t)} S(t) \xrightarrow[-\log S(t)]{} \Lambda(t) \xrightarrow[\Lambda'(t)]{} \lambda(t) \xrightarrow[\log\{\lambda(t)\}]{} \eta(t)$$


---

**Identities:**

$$\begin{aligned} S(t) &= 1 - F(t) \\ &= \exp\{-\Lambda(t)\} \\ S'(t) &= -f(t) \\ \Lambda(t) &= -\log\{S(t)\} \\ \Lambda'(t) &= \lambda(t) \\ \lambda(t) &= \frac{f(t)}{S(t)} \\ &= -\frac{\partial}{\partial t} \log S(t) \\ f(t) &= \lambda(t) \cdot S(t) \end{aligned}$$


---

Some proofs (others left as exercises):

$$\begin{aligned} S'(t) &= \frac{\partial}{\partial t} (1 - F(t)) \\ &= -F'(t) \\ &= -f(t) \end{aligned}$$


---

$$\begin{aligned} \frac{\partial}{\partial t} \log S(t) &= \frac{S'(t)}{S(t)} \\ &= -\frac{f(t)}{S(t)} \\ &= -\lambda(t) \end{aligned}$$


---

$$\begin{aligned} \Lambda(t) &\stackrel{\text{def}}{=} \int_{u=0}^t h(u) du \\ &= \int_0^t -\frac{\partial}{\partial u} \log\{S(u)\} du \\ &= [-\log\{S(u)\}]_{u=0}^{u=t} \\ &= [\log\{S(u)\}]_{u=t}^{u=0} \\ &= \log\{S(0)\} - \log\{S(t)\} \\ &= \log\{1\} - \log\{S(t)\} \\ &= 0 - \log\{S(t)\} \\ &= -\log\{S(t)\} \end{aligned}$$


---

Corollary:

$$S(t) = \exp\{-\Lambda(t)\}$$


---

## Example: Time to death the US in 2004

The first day is the most dangerous:

```
# download `survexp.rda` from:
# paste0(
# "https://github.com/therneau/survival/raw/",
# "f3ac93704949ff26e07720b56f2b18ffa8066470/",
# "Data/survexp.rda")

# (newer versions of `survival` don't have the first-year breakdown; see:
# https://cran.r-project.org/web/packages/survival/news.html)

fs::path(
  here::here(),
  "Data",
  "survexp.rda"
) |>
  load()
s1 <- survexp.us[, "female", "2004"]
age1 <- c(
  0.5 / 365.25,
  4 / 365.25,
  17.5 / 365.25,
  196.6 / 365.25,
  1:109 + 0.5
)
s2 <- 365.25 * s1[5:113]
s2 <- c(s1[1], 6 * s1[2], 22 * s1[3], 337.25 * s1[4], s2)
cols <- rep(1, 113)
cols[1] <- 2
cols[2] <- 3
cols[3] <- 4

plot(age1, s1, type = "b", lwd = 2, xlab = "Age", ylab = "Daily Hazard Rate", col = cols)

text(10, .003, "First Day", col = 2)
text(18, .00030, "Rest of First Week", col = 3)
text(18, .00015, "Rest of First month", col = 4)
```

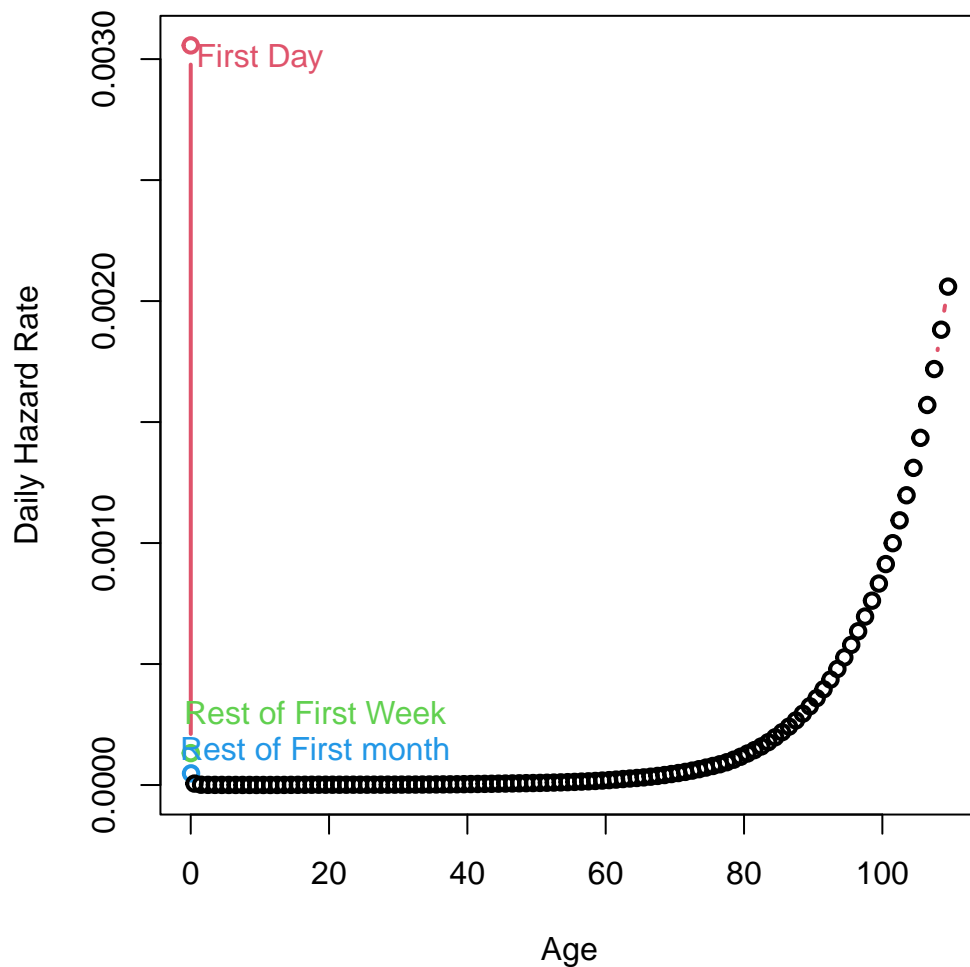


Figure 4: Daily Hazard Rates in 2004 for US Females

**Exercise 1.1.** Hypothesize why the male and female hazard functions in Figure 5 differ where they do?

```

yrs <- 1:40
s1 <- survexp.us[5:113, "male", "2004"]
s2 <- survexp.us[5:113, "female", "2004"]

age1 <- 1:109

plot(age1[yrs], s1[yrs], type = "l", lwd = 2, xlab = "Age", ylab = "Daily Hazard Rate")
lines(age1[yrs], s2[yrs], col = 2, lwd = 2)
legend(5, 5e-6, c("Males", "Females"), col = 1:2, lwd = 2)

```

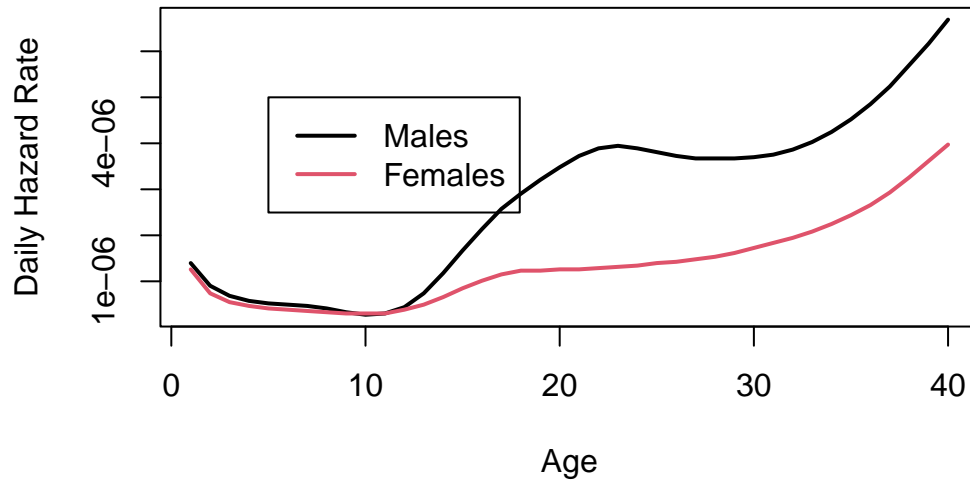


Figure 5: Daily Hazard Rates in 2004 for US Males and Females 1-40

---

**Exercise 1.2.** Compare and contrast Figure 6 with Figure 4.

---

```
s1 <- survexp.us[, "female", "2004"]

s2 <- 365.25 * s1[5:113]
s2 <- c(s1[1], 6 * s1[2], 21 * s1[3], 337.25 * s1[4], s2)
cs2 <- cumsum(s2)
age2 <- c(1 / 365.25, 7 / 365.25, 28 / 365.25, 1:110)
plot(age2, exp(-cs2), type = "l", lwd = 2, xlab = "Age", ylab = "Survival")
```

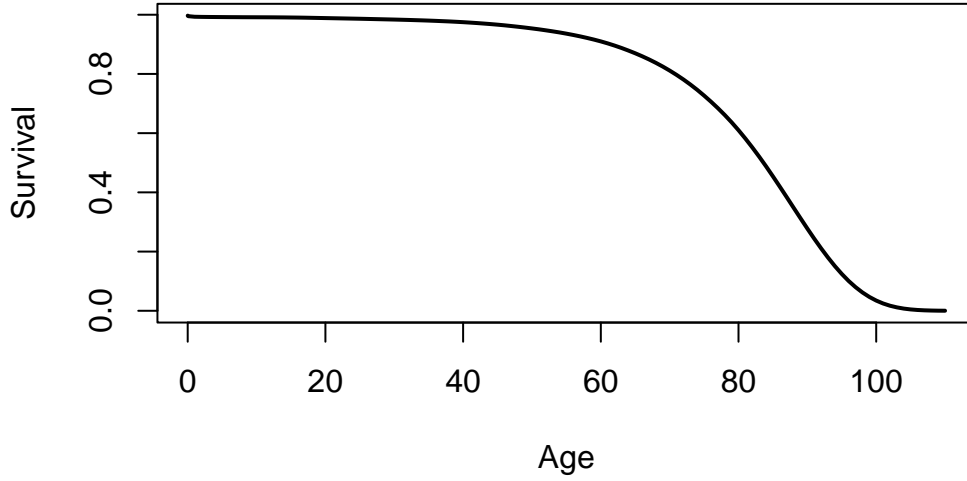


Figure 6: Survival Curve in 2004 for US Females

### 1.3.7 Likelihood with censoring

If an event time  $T$  is observed exactly as  $T = t$ , then the likelihood of that observation is just its probability density function:

$$\begin{aligned}
 \mathcal{L}(t) &= f(T = t) \\
 &\stackrel{\text{def}}{=} f_T(t) \\
 &= \lambda_T(t)S_T(t) \\
 \ell(t) &\stackrel{\text{def}}{=} \log\{\mathcal{L}(t)\} \\
 &= \log\{\lambda_T(t)S_T(t)\} \\
 &= \log\{\lambda_T(t)\} + \log\{S_T(t)\} \\
 &= \log\{\lambda_T(t)\} - \Lambda_T(t)
 \end{aligned}$$


---

If instead the event time  $T$  is censored and only known to be after time  $y$ , then the likelihood of that censored observation is instead the survival function evaluated at the censoring time:

$$\begin{aligned}
 \mathcal{L}(y) &= p_T(T > y) \\
 &\stackrel{\text{def}}{=} S_T(y) \\
 \ell(y) &\stackrel{\text{def}}{=} \log\{\mathcal{L}(y)\} \\
 &= \log\{S(y)\} \\
 &= -\Lambda(y)
 \end{aligned}$$


---

What's written above is incomplete. We also observed whether or not the observation was censored. Let  $C$  denote the time when censoring would occur (if the event did not occur first); let  $f_C(y)$  and  $S_C(y)$  be the corresponding density and survival functions for the censoring event.

Let  $Y$  denote the time when observation ended (either by censoring or by the event of interest occurring), and let  $D$  be an indicator variable for the event occurring at  $Y$  (so  $D = 0$  represents

a censored observation and  $D = 1$  represents an uncensored observation). In other words, let  $Y \stackrel{\text{def}}{=} \min(T, C)$  and  $D \stackrel{\text{def}}{=} \mathbb{1}\{T \leq C\}$ .

Then the complete likelihood of the observed data  $(Y, D)$  is:

$$\begin{aligned}\mathcal{L}(y, d) &= p(Y = y, D = d) \\ &= [p(T = y, C > y)]^d \cdot [p(T > y, C = y)]^{1-d}\end{aligned}$$


---

Typically, survival analyses assume that  $C$  and  $T$  are mutually independent; this assumption is called “non-informative” censoring.

Then the joint likelihood  $p(Y, D)$  factors into the product  $p(Y)p(D)$ , and the likelihood reduces to:

$$\begin{aligned}\mathcal{L}(y, d) &= [p(T = y, C > y)]^d \cdot [p(T > y, C = y)]^{1-d} \\ &= [p(T = y)p(C > y)]^d \cdot [p(T > y)p(C = y)]^{1-d} \\ &= [f_T(y)S_C(y)]^d \cdot [S(y)f_C(y)]^{1-d} \\ &= [f_T(y)^d S_C(y)^d] \cdot [S_T(y)^{1-d} f_C(y)^{1-d}] \\ &= (f_T(y)^d \cdot S_T(y)^{1-d}) \cdot (f_C(y)^{1-d} \cdot S_C(y)^d)\end{aligned}$$


---

The corresponding log-likelihood is:

$$\begin{aligned}\ell(y, d) &= \log\{\mathcal{L}(y, d)\} \\ &= \log\{(f_T(y)^d \cdot S_T(y)^{1-d}) \cdot (f_C(y)^{1-d} \cdot S_C(y)^d)\} \\ &= \log\{f_T(y)^d \cdot S_T(y)^{1-d}\} + \log\{f_C(y)^{1-d} \cdot S_C(y)^d\}\end{aligned}$$

Let

- $\theta_T$  represent the parameters of  $p_T(t)$ ,
  - $\theta_C$  represent the parameters of  $p_C(c)$ ,
  - $\theta = (\theta_T, \theta_C)$  be the combined vector of all parameters.
- 

The corresponding score function is:

$$\begin{aligned}\ell'(y, d) &= \frac{\partial}{\partial \theta} [\log\{f_T(y)^d \cdot S_T(y)^{1-d}\} + \log\{f_C(y)^{1-d} \cdot S_C(y)^d\}] \\ &= \left( \frac{\partial}{\partial \theta} \log\{f_T(y)^d \cdot S_T(y)^{1-d}\} \right) + \left( \frac{\partial}{\partial \theta} \log\{f_C(y)^{1-d} \cdot S_C(y)^d\} \right)\end{aligned}$$


---

As long as  $\theta_C$  and  $\theta_T$  don't share any parameters, then if censoring is non-informative, the partial derivative with respect to  $\theta_T$  is:

$$\begin{aligned}\ell'_{\theta_T}(y, d) &\stackrel{\text{def}}{=} \frac{\partial}{\partial \theta_T} \ell(y, d) \\ &= \left( \frac{\partial}{\partial \theta_T} \log\{f_T(y)^d \cdot S_T(y)^{1-d}\} \right) + \left( \frac{\partial}{\partial \theta_T} \log\{f_C(y)^{1-d} \cdot S_C(y)^d\} \right) \\ &= \left( \frac{\partial}{\partial \theta_T} \log\{f_T(y)^d \cdot S_T(y)^{1-d}\} \right) + 0 \\ &= \frac{\partial}{\partial \theta_T} \log\{f_T(y)^d \cdot S_T(y)^{1-d}\}\end{aligned}$$


---



Thus, the MLE for  $\theta_T$  won't depend on  $\theta_C$ , and we can ignore the distribution of  $C$  when estimating the parameters of  $f_T(t) = p(T = t)$ .

Then:

$$\begin{aligned}\mathcal{L}(y, d) &= f_T(y)^d \cdot S_T(y)^{1-d} \\ &= (h_T(y)^d S_T(y)^d) \cdot S_T(y)^{1-d} \\ &= h_T(y)^d \cdot S_T(y)^d \cdot S_T(y)^{1-d} \\ &= h_T(y)^d \cdot S_T(y) \\ &= S_T(y) \cdot h_T(y)^d\end{aligned}$$

That is, if the event occurred at time  $y$  (i.e., if  $d = 1$ ), then the likelihood of  $(Y, D) = (y, d)$  is equal to the hazard function at  $y$  times the survival function at  $y$ . Otherwise, the likelihood is equal to just the survival function at  $y$ .

---

The corresponding log-likelihood is:

$$\begin{aligned}\ell(y, d) &= \log \{ \mathcal{L}(y, d) \} \\ &= \log \{ S_T(y) \cdot h_T(y)^d \} \\ &= \log \{ S_T(y) \} + \log \{ h_T(y)^d \} \\ &= \log \{ S_T(y) \} + d \cdot \log \{ h_T(y) \} \\ &= -H_T(y) + d \cdot \log \{ h_T(y) \}\end{aligned}$$

In other words, the log-likelihood contribution from a single observation  $(Y, D) = (y, d)$  is equal to the negative cumulative hazard at  $y$ , plus the log of the hazard at  $y$  if the event occurred at time  $y$ .

## 1.4 Parametric Models for Time-to-Event Outcomes

### 1.4.1 Exponential Distribution

- The exponential distribution is the base distribution for survival analysis.
- The distribution has a constant hazard  $\lambda$
- The mean survival time is  $\lambda^{-1}$

---

#### Mathematical details of exponential distribution

$$\begin{aligned}f(t) &= \lambda e^{-\lambda t} \\ F(t) &= 1 - e^{-\lambda t} \\ S(t) &= e^{-\lambda t} \\ \ln(S(t)) &= -\lambda t \\ \lambda(t) &= -\frac{f(t)}{S(t)} = -\frac{\lambda e^{-\lambda t}}{e^{-\lambda t}} = \lambda \\ E(t) &= \lambda^{-1} \\ Var(t) &= \lambda^{-2} \\ \log\{f(t)\} &= \log\{\lambda\} - \lambda t \\ \frac{\partial}{\partial \lambda} \log\{f(t)\} &= \lambda^{-1} - t \\ &= E[t] - t \\ &= -(E[t] - t) \\ &= -\varepsilon\end{aligned}$$


---

## Prediction intervals for time-to-event outcomes

**Exercise 1.3** (Construct a prediction interval). Suppose a cancer patient is predicted to have an expected (mean) lifetime of 7 years after diagnosis, and suppose the distribution is exponential.

Construct a 95% prediction interval for survival.

### Tip

Use the quantiles of the exponential distribution.

---

*Solution 1.1.* If the mean is 7 years until death, then the rate parameter  $\lambda = 1/7$  events (deaths) per year.

The 0.025 quantile of the exponential distribution with  $\lambda = 1/7$  is `qexp(p = 0.025, rate = 1/7)` = 0.177225 and the 0.975 quantile is `qexp(p = 0.975, rate = 1/7)` = 25.822156, so the prediction interval is `qexp(p = c(.025, 0.975), rate = 1/7)` = (0.177225, 25.822156).

---

**Exercise 1.4.** Graph the prediction interval as a function of the mean, for Gaussian ( $\sigma = 1$ ), Binomial, Poisson, and Exponential.

---

*Solution 1.2.* Left to the reader for now.

---

**Exercise 1.5** (Explain the results). Why do time-to-event models have such wide predictive intervals?

### Tip

Consider the relationship between the mean, variance, and standard deviation of the exponential distribution, and contrast that relationship with the Poisson distribution and the Bernoulli distribution.

---

*Solution 1.3.* In the exponential distribution, variance is the square of the mean (hence SD is equal to mean); as opposed to Poisson, where variance was equal to the mean (and SD is the square-root of the mean), or the Bernoulli, where the variance is the mean minus the square of the mean (so the SD is smaller than the square-root of the mean).

## Estimating $\lambda$

- Suppose we have  $m$  exponential survival times of  $t_1, t_2, \dots, t_m$  and  $k$  right-censored values at  $u_1, u_2, \dots, u_k$ .
- A survival time of  $t_i = 10$  means that subject  $i$  died at time 10. A right-censored time  $u_i = 10$  means that at time 10, subject  $i$  was still alive and that we have no further follow-up.
- For the moment we will assume that the survival distribution is exponential and that all the subjects have the same parameter  $\lambda$ .

We have  $m$  exponential survival times of  $t_1, t_2, \dots, t_m$  and  $k$  right-censored values at  $u_1, u_2, \dots, u_k$ . The log-likelihood of an observed survival time  $t_i$  is

$$\log \{ \lambda e^{-\lambda t_i} \} = \log \{ \lambda \} - \lambda t_i$$

and the likelihood of a censored value is the probability of that outcome (survival greater than  $u_j$ ) so the log-likelihood is

$$\begin{aligned}\ell_j(\lambda) &= \log \{\lambda e^{u_j}\} \\ &= -\lambda u_j\end{aligned}$$


---

**Theorem 1.7.** *Let  $T = \sum t_i$  and  $U = \sum u_j$ . Then:*

$$\hat{\lambda}_{ML} = \frac{m}{T + U} \tag{2}$$


---

*Proof.*

$$\begin{aligned}\ell(\lambda) &= \sum_{i=1}^m (\ln \lambda - \lambda t_i) + \sum_{j=1}^k (-\lambda u_j) \\ &= m \ln \lambda - (T + U)\lambda \\ \ell'(\lambda) &= m\lambda^{-1} - (T + U) \\ \hat{\lambda} &= \frac{m}{T + U}\end{aligned}$$

□

---


$$\begin{aligned}\ell'' &= -m/\lambda^2 \\ &< 0 \\ \hat{E}[T] &= \hat{\lambda}^{-1} \\ &= \frac{T + U}{m}\end{aligned}$$


---

## Fisher Information and Standard Error

$$\begin{aligned}E[-\ell''] &= m/\lambda^2 \\ \text{Var}(\hat{\lambda}) &\approx (E[-\ell''])^{-1} \\ &= \lambda^2/m \\ \text{SE}(\hat{\lambda}) &= \sqrt{\text{Var}(\hat{\lambda})} \\ &\approx \lambda/\sqrt{m}\end{aligned}$$

$\hat{\lambda}$  depends on the censoring times of the censored observations, but  $\text{Var}(\hat{\lambda})$  only depends on the number of uncensored observations,  $m$ , and not on the number of censored observations ( $k$ ).

---

### 1.4.2 Other Parametric Survival Distributions

- Any density on  $[0, \infty)$  can be a survival distribution, but the most useful ones are all skew right.
- The most frequently used generalization of the exponential is the Weibull<sup>4</sup>.
- Other common choices are the gamma, log-normal, log-logistic, Gompertz, inverse Gaussian, and Pareto.
- Most of what we do going forward is non-parametric or semi-parametric, but sometimes these parametric distributions provide a useful approach.

---

<sup>4</sup>[probability.qmd#sec-weibull](#)

## 1.5 Nonparametric Survival Analysis

### 1.5.1 Basic ideas

- Mostly, we work without a parametric model.
- The first task is to estimate a survival function from data listing survival times, and censoring times for censored data.
- For example one patient may have relapsed at 10 months. Another might have been followed for 32 months without a relapse having occurred (censored).
- The minimum information we need for each patient is a time and a censoring variable which is 1 if the event occurred at the indicated time and 0 if this is a censoring time.

## 1.6 Example: clinical trial for pediatric acute leukemia

### 1.6.1 Overview of study

This is from a clinical trial in 1963 for 6-MP treatment vs. placebo for Acute Leukemia in 42 children.

- Pairs of children:
- matched by remission status at the time of treatment (**remstat**: 1 = partial, 2 = complete)
- randomized to 6-MP (exit times in **t2**) or placebo (exit times in **t1**)
- Followed until relapse or end of study.
- All of the placebo group relapsed, but some of the 6-MP group were censored (which means they were still in remission); indicated by **relapse** variable (0 = censored, 1 = relapse).
- 6-MP = 6-Mercaptopurine (Purinethol) is an anti-cancer (“antineoplastic” or “cytotoxic”) chemotherapy drug used currently for Acute lymphoblastic leukemia (ALL). It is classified as an antimetabolite.

### 1.6.2 Study design

- Clinical trial in 1963 for 6-MP treatment vs. placebo for Acute Leukemia in 42 children.
- Pairs of children:
- matched by remission status at the time of treatment (**remstat**)
- **remstat** = 1: partial
- **remstat** = 2: complete
- randomized to 6-MP (exit time: **t2**) or placebo (**t1**).
- Followed until relapse or end of study.
- All of the placebo group relapsed,
- Some of the 6-MP group were censored.

---

### 1.6.3 Data documentation for drug6mp

```
# library(printr) # inserts help-file output into markdown output
library(KMsurv)
?drug6mp
```

### 1.6.4 Descriptive Statistics

- The average time in each group is not useful. Some of the 6-MP patients have not relapsed at the time recorded, while all of the placebo patients have relapsed.
- The median time is not really useful either because so many of the 6-MP patients have not relapsed (12/21).
- Both are biased down in the 6-MP group. Remember that lower times are worse since they indicate sooner recurrence.

Table 1: `drug6mp` pediatric acute leukemia data

```
library(KMsurv)
data(drug6mp)
drug6mp <- drug6mp |>
  tibble::as_tibble() |>
  print()
#> # A tibble: 21 x 5
#>   pair remstat   t1   t2 relapse
#>   <int>  <int> <int> <int>   <int>
#> 1     1      1     1     10      1
#> 2     2      2     2     22      7
#> 3     3      2     3     32      0
#> 4     4      2    12     23      1
#> 5     5      2     8     22      1
#> 6     6      1    17      6      1
#> 7     7      2     2     16      1
#> 8     8      2    11     34      0
#> 9     9      2     8     32      0
#> 10    10      2    12     25      0
#> # i 11 more rows
```

Table 2: Summary statistics for `drug6mp` data

```
summary(drug6mp)
#>      pair      remstat      t1      t2      relapse
#> Min.   : 1  Min.   :1.00  Min.   : 1.00  Min.   : 6.0  Min.   :0.000
#> 1st Qu.: 6  1st Qu.:2.00  1st Qu.: 4.00  1st Qu.: 9.0  1st Qu.:0.000
#> Median :11  Median :2.00  Median : 8.00  Median :16.0  Median :0.000
#> Mean   :11  Mean   :1.76  Mean   : 8.67  Mean   :17.1  Mean   :0.429
#> 3rd Qu.:16  3rd Qu.:2.00  3rd Qu.:12.00  3rd Qu.:23.0  3rd Qu.:1.000
#> Max.   :21  Max.   :2.00  Max.   :23.00  Max.   :35.0  Max.   :1.000
```

### 1.6.5 Exponential model

- We *can* compute the hazard rate, assuming an exponential model: number of relapses divided by the sum of the exit times (Equation 2).

$$\hat{\lambda} = \frac{\sum_{i=1}^n D_i}{\sum_{i=1}^n Y_i}$$

- For the placebo, that is just the reciprocal of the mean time:

$$\begin{aligned}\hat{\lambda}_{\text{placebo}} &= \frac{\sum_{i=1}^n D_i}{\sum_{i=1}^n Y_i} \\ &= \frac{\sum_{i=1}^n 1}{\sum_{i=1}^n Y_i} \\ &= \frac{n}{\sum_{i=1}^n Y_i} \\ &= \frac{1}{\bar{Y}} \\ &= \frac{1}{8.666667} \\ &= 0.115385\end{aligned}$$


---

- For the 6-MP group,  $\hat{\lambda} = 9/359 = 0.025$

$$\begin{aligned}\hat{\lambda}_{\text{6-MP}} &= \frac{\sum_{i=1}^n D_i}{\sum_{i=1}^n Y_i} \\ &= \frac{9}{359} \\ &= 0.02507\end{aligned}$$

- The estimated hazard in the placebo group is 4.6 times as large as in the 6-MP group (assuming the hazard is constant over time).

## 1.7 The Kaplan-Meier Product Limit Estimator

### 1.7.1 Estimating survival in datasets without censoring

In the `drug6mp` dataset, the estimated survival function for the placebo patients is easy to compute. For any time  $t$  in months,  $S(t)$  is the fraction of patients with times greater than  $t$ :

### 1.7.2 Estimating survival in datasets with censoring

- For the 6-MP patients, we cannot ignore the censored data because we know that the time to relapse is greater than the censoring time.
- For any time  $t$  in months, we know that 6-MP patients with times greater than  $t$  have not relapsed, and those with relapse time less than  $t$  have relapsed, but we don't know if patients with censored time less than  $t$  have relapsed or not.
- The procedure we usually use is the Kaplan-Meier product-limit estimator of the survival function.
- The Kaplan-Meier estimator is a step function (like the empirical cdf), which changes value only at the event times, not at the censoring times.
- At each event time  $t$ , we compute the at-risk group size  $Y$ , which is all those observations whose event time or censoring time is at least  $t$ .

- If  $d$  of the observations have an event time (not a censoring time) of  $t$ , then the group of survivors immediately following time  $t$  is reduced by the fraction

$$\frac{Y-d}{Y} = 1 - \frac{d}{Y}$$

---

**Definition 1.6** (Kaplan-Meier Product-Limit Estimator of Survival Function). If a time-to-event data set contains  $k$  event times  $t_i$ , ( $i \in 1 : k$ ), where  $n_i$  is the number of individuals at risk at time  $t_i$  and  $d_i$  is the number of events at time  $t_i$ , then the **Kaplan-Meier Product-Limit Estimator** of the survival function is:

$$\hat{\lambda}_i = \frac{d_i}{n_i}$$

$$\hat{S}_{KM}(t) \stackrel{\text{def}}{=} \prod_{\{i: t_i < t\}} [1 - \hat{\lambda}_i]$$


---

**Theorem 1.8** (Kaplan-Meier Estimate with No Censored Observations). *If there are no censored data, and there are  $n$  data points, then just after (say) the third event time*

$$\begin{aligned} \hat{S}(t) &= \prod_{t_i < t} \left[ 1 - \frac{d_i}{Y_i} \right] \\ &= \left[ \frac{n - d_1}{n} \right] \left[ \frac{n - d_1 - d_2}{n - d_1} \right] \left[ \frac{n - d_1 - d_2 - d_3}{n - d_1 - d_2} \right] \\ &= \frac{n - d_1 - d_2 - d_3}{n} \\ &= 1 - \frac{d_1 + d_2 + d_3}{n} \\ &= 1 - \hat{F}(t) \end{aligned}$$

where  $\hat{F}(t)$  is the usual empirical CDF estimate.

### 1.7.3 Kaplan-Meier curve for drug6mp data

Here is the Kaplan-Meier estimated survival curve for the patients who received 6-MP in the **drug6mp** dataset (we will see code to produce figures like this one shortly):

```
# | echo: false

require(KMsurv)
data(drug6mp)
library(dplyr)
library(survival)

drug6mp_km_model1 <-
  drug6mp |>
  mutate(surv = Surv(t2, relapse)) |>
  survfit(formula = surv ~ 1, data = _)

library(ggfortify)
drug6mp_km_model1 |>
  autoplot(
    mark.time = TRUE,
    conf.int = FALSE
  ) +
  expand_limits(y = 0) +
```

```
xlab("Time since diagnosis (months)") +
ylab("KM Survival Curve")
```

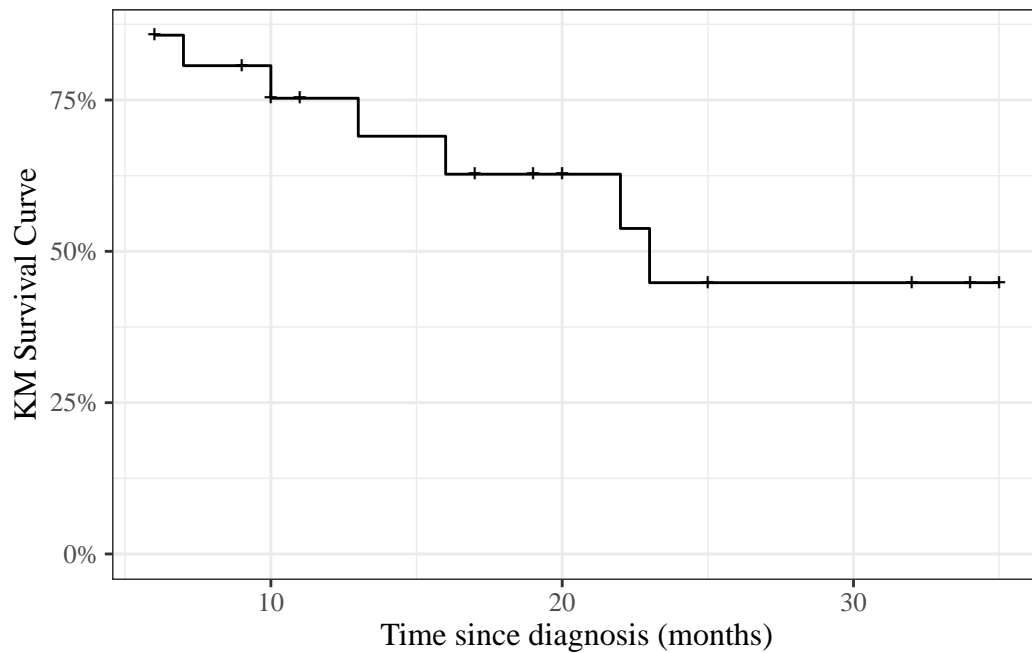


Figure 7: Kaplan-Meier Survival Curve for 6-MP Patients

#### 1.7.4 Kaplan-Meier calculations

Let's compute these estimates and build the chart by hand:

```
library(KMsurv)
library(dplyr)
data(drug6mp)

drug6mp.v2 <-
  drug6mp |>
  as_tibble() |>
  mutate(
    remstat = remstat |>
      case_match(
        1 ~ "partial",
        2 ~ "complete"
      ),
    # renaming to "outcome" while relabeling is just a style choice:
    outcome = relapse |>
      case_match(
        0 ~ "censored",
        1 ~ "relapsed"
      )
  )

km.6mp <-
  drug6mp.v2 |>
  summarize(
    .by = t2,
    Relapses = sum(outcome == "relapsed"),
    Censored = sum(outcome == "censored")
  )
```



```

) |>
# here we add a start time row, so the graph starts at time 0:
bind_rows(
  tibble(
    t2 = 0,
    Relapses = 0,
    Censored = 0
  )
) |>
# sort in time order:
arrange(t2) |>
mutate(
  Exiting = Relapses + Censored,
  `Study Size` = sum(Exiting),
  Exited = cumsum(Exiting) |> dplyr::lag(default = 0),
  `At Risk` = `Study Size` - Exited,
  Hazard = Relapses / `At Risk`,
  `KM Factor` = 1 - Hazard,
  `Cumulative Hazard` = cumsum(`Hazard`),
  `KM Survival Curve` = cumprod(`KM Factor`)
)

library(pander)
pander(km.6mp)

```

t2	Re-lapses	Cen-sored	Exit-ing	Study Size	Ex-ited	At Risk	Haz-ard	KM Factor	Cumula-tive Hazard	KM Survival Curve
0	0	0	0	21	0	21	0	1	0	1
6	3	1	4	21	0	21	0.1429	0.8571	0.1429	0.8571
7	1	0	1	21	4	17	0.05882	0.9412	0.2017	0.8067
9	0	1	1	21	5	16	0	1	0.2017	0.8067
10	1	1	2	21	6	15	0.06667	0.9333	0.2683	0.7529
11	0	1	1	21	8	13	0	1	0.2683	0.7529
13	1	0	1	21	9	12	0.08333	0.9167	0.3517	0.6902
16	1	0	1	21	10	11	0.09091	0.9091	0.4426	0.6275
17	0	1	1	21	11	10	0	1	0.4426	0.6275
19	0	1	1	21	12	9	0	1	0.4426	0.6275
20	0	1	1	21	13	8	0	1	0.4426	0.6275
22	1	0	1	21	14	7	0.1429	0.8571	0.5854	0.5378
23	1	0	1	21	15	6	0.1667	0.8333	0.7521	0.4482
25	0	1	1	21	16	5	0	1	0.7521	0.4482
32	0	2	2	21	17	4	0	1	0.7521	0.4482
34	0	1	1	21	19	2	0	1	0.7521	0.4482
35	0	1	1	21	20	1	0	1	0.7521	0.4482

## Summary

For the 6-MP patients at time 6 months, there are 21 patients at risk. At  $t = 6$  there are 3 relapses and 1 censored observations.

The Kaplan-Meier factor is  $(21 - 3)/21 = 0.857$ . The number at risk for the next time ( $t = 7$ ) is  $21 - 3 - 1 = 17$ .

At time 7 months, there are 17 patients at risk. At  $t = 7$  there is 1 relapse and 0 censored observations. The Kaplan-Meier factor is  $(17 - 1)/17 = 0.941$ . The Kaplan Meier estimate is  $0.857 \times 0.941 = 0.807$ .

The number at risk for the next time ( $t = 9$ ) is  $17 - 1 = 16$ .

Now, let's graph this estimated survival curve using `ggplot()`:

```
library(ggplot2)
conflicts_prefer(dplyr::filter)
km.6mp |>
  ggplot(aes(x = t2, y = `KM Survival Curve`)) +
  geom_step() +
  geom_point(data = km.6mp |> filter(Censored > 0), shape = 3) +
  expand_limits(y = c(0, 1), x = 0) +
  xlab("Time since diagnosis (months)") +
  ylab("KM Survival Curve") +
  scale_y_continuous(labels = scales::percent)
```

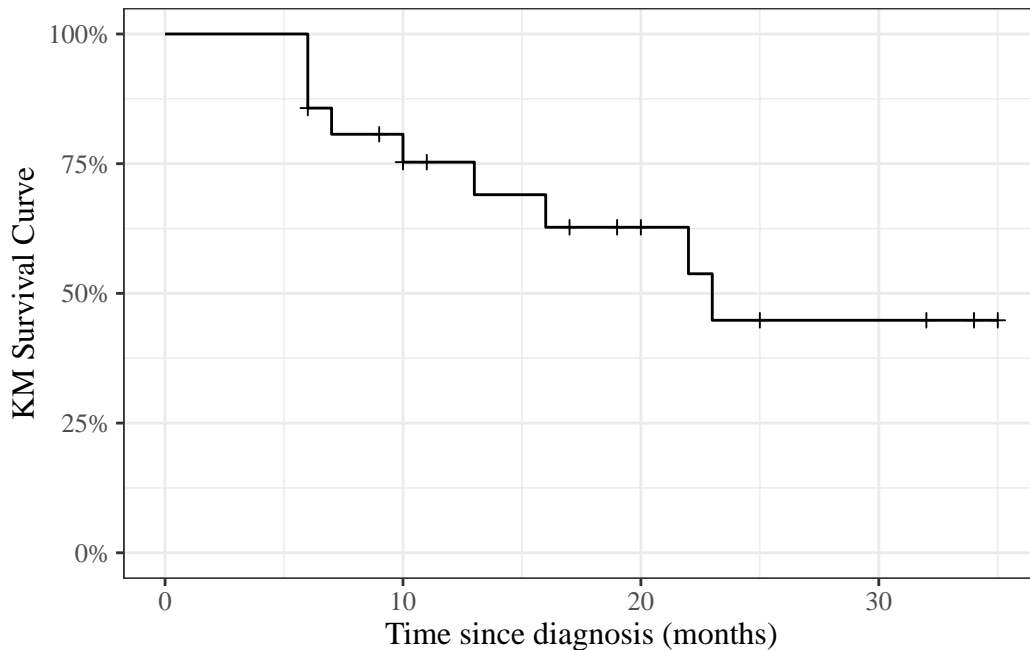


Figure 8: KM curve for 6MP patients, calculated by hand

## 1.8 Using the `survival` package in R

We don't have to do these calculations by hand every time; the `survival` package and several others have functions available to automate many of these tasks (full list: <https://cran.r-project.org/web/views/Survival.html>).

### 1.8.1 The `Surv` function

To use the `survival` package, the first step is telling R how to combine the exit time and exit reason (censoring versus event) columns. The `Surv()` function accomplishes this task.

**Example: `Surv()` with `drug6mp` data**

```
1 library(survival)
2 drug6mp.v3 <-
3   drug6mp.v2 |>
4   mutate(
5     surv2 = Surv(
```

```

6     time = t2,
7     event = (outcome == "relapsed")
8   )
9 )
10
11 print(drug6mp.v3)
12 #> # A tibble: 21 x 7
13 #>   pair remstat    t1    t2 relapse outcome  surv2
14 #>   <int> <chr>   <int> <int>   <int> <chr>   <Surv>
15 #> 1     1 1 partial     1    10     1 relapsed    10
16 #> 2     2 2 complete    22     7     1 relapsed     7
17 #> 3     3 3 complete     3    32     0 censored   32+
18 #> 4     4 4 complete    12    23     1 relapsed    23
19 #> 5     5 5 complete     8    22     1 relapsed    22
20 #> 6     6 6 partial    17     6     1 relapsed     6
21 #> 7     7 7 complete     2    16     1 relapsed    16
22 #> 8     8 8 complete    11    34     0 censored   34+
23 #> 9     9 9 complete     8    32     0 censored   32+
24 #> 10    10 complete    12    25     0 censored   25+
25 #> # i 11 more rows

```

The output of `Surv()` is a vector of objects with class `Surv`. When we print this vector:

- observations where the event was observed are printed as the event time (for example, `surv2 = 10` on line 1)
- observations where the event was right-censored are printed as the censoring time with a plus sign (+; for example, `surv2 = 32+` on line 3).

### 1.8.2 The `survfit` function

Once we have constructed our `Surv` variable, we can calculate the Kaplan-Meier estimate of the survival curve using the `survfit()` function.

#### **i** Note

The documentation for `?survfit` isn't too helpful; the `survfit.formula` documentation is better.

#### Example: `survfit()` with `drug6mp` data

Here we use `survfit()` to create a `survfit` object, which contains the Kaplan-Meier estimate:

```

drug6mp.km_model <- survfit(
  formula = surv2 ~ 1,
  data = drug6mp.v3
)

```

`print.survfit()` just gives some summary statistics:

```

print(drug6mp.km_model)
#> Call: survfit(formula = surv2 ~ 1, data = drug6mp.v3)
#>
#>      n events median 0.95LCL 0.95UCL
#> [1,] 21      9     23      16     NA

```

`summary.survfit()` shows us the underlying Kaplan-Meier table:

```
summary(drug6mp.km_model)
#> Call: survfit(formula = surv2 ~ 1, data = drug6mp.v3)
#>
#>   time n.risk n.event survival std.err lower 95% CI upper 95% CI
#>    6     21      3    0.857  0.0764    0.720    1.000
#>    7     17      1    0.807  0.0869    0.653    0.996
#>   10     15      1    0.753  0.0963    0.586    0.968
#>   13     12      1    0.690  0.1068    0.510    0.935
#>   16     11      1    0.627  0.1141    0.439    0.896
#>   22      7      1    0.538  0.1282    0.337    0.858
#>   23      6      1    0.448  0.1346    0.249    0.807
```

---

We can specify which time points we want using the `times` argument:

```
summary(
  drug6mp.km_model,
  times = c(0, drug6mp.v3$t2)
)
#> Call: survfit(formula = surv2 ~ 1, data = drug6mp.v3)
#>
#>   time n.risk n.event survival std.err lower 95% CI upper 95% CI
#>    0     21      0    1.000  0.0000    1.000    1.000
#>   10     15      1    0.753  0.0963    0.586    0.968
#>    7     17      1    0.807  0.0869    0.653    0.996
#>   32      4      0    0.448  0.1346    0.249    0.807
#>   23      6      1    0.448  0.1346    0.249    0.807
#>   22      7      1    0.538  0.1282    0.337    0.858
#>    6     21      3    0.857  0.0764    0.720    1.000
#>   16     11      1    0.627  0.1141    0.439    0.896
#>   34      2      0    0.448  0.1346    0.249    0.807
#>   32      4      0    0.448  0.1346    0.249    0.807
#>   25      5      0    0.448  0.1346    0.249    0.807
#>   11     13      0    0.753  0.0963    0.586    0.968
#>   20      8      0    0.627  0.1141    0.439    0.896
#>   19      9      0    0.627  0.1141    0.439    0.896
#>    6     21      3    0.857  0.0764    0.720    1.000
#>   17     10      0    0.627  0.1141    0.439    0.896
#>   35      1      0    0.448  0.1346    0.249    0.807
#>    6     21      3    0.857  0.0764    0.720    1.000
#>   13     12      1    0.690  0.1068    0.510    0.935
#>    9     16      0    0.807  0.0869    0.653    0.996
#>    6     21      3    0.857  0.0764    0.720    1.000
#>   10     15      1    0.753  0.0963    0.586    0.968
```

---

```
?summary.survfit
```

### 1.8.3 Plotting estimated survival functions

We can plot `survfit` objects with `plot()`, `autoplot()`, or `ggsurvplot()`:

```
library(ggfortify)
autoplot(drug6mp.km_model)
```

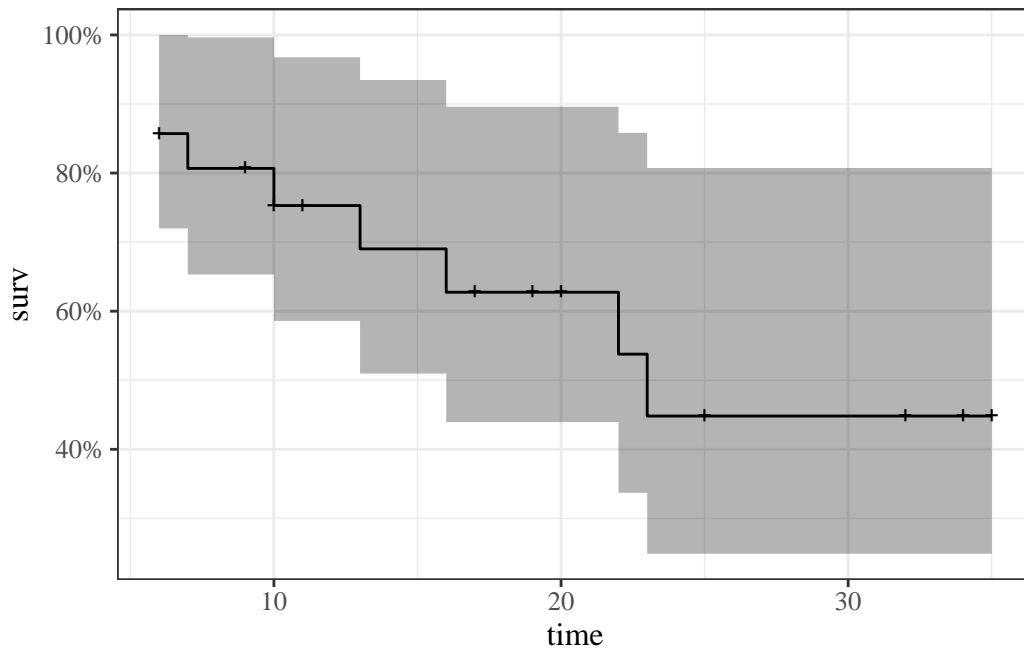


Figure 9: Kaplan-Meier Survival Curve for 6-MP Patients

```
# not shown:
# plot(drug6mp.km_model)

# library(survminer)
# ggsurvplot(drug6mp.km_model)
```

### quantiles of survival curve

We can extract quantiles with `quantile()`:

```
1 drug6mp.km_model |>
2   quantile(p = c(.25, .5)) |>
3   as_tibble() |>
4   mutate(p = c(.25, .5)) |>
5   relocate(p, .before = everything())
6 #> # A tibble: 2 x 4
7 #>       p quantile lower upper
8 #>   <dbl>   <dbl> <dbl> <dbl>
9 #> 1  0.25     13     6    NA
10 #> 2  0.5     23    16    NA
```

## 1.9 The log-rank test

(a.k.a. the Mantel-Cox test)

**Exercise 1.6.** How do we test the null hypothesis that two or more groups have the same time-to-event distribution?

*Solution 1.4.* One option is the log-rank test comparing the Kaplan-Meier estimates of the survival functions of those groups.

---

Adapted from Kleinbaum and Klein (2012) p68:

- The log-rank test is a large-sample chi-square test.
- The log-rank test uses a test statistic that compares KM curves between groups across all survival times.
- Like many other statistics used in other kinds of chi-square tests, the log-rank statistic makes use of observed versus expected cell counts over categories of outcomes.
- The categories for the log-rank statistic are defined by each of the ordered failure times for the entire set of data being analyzed.

For  $t \in t_1, \dots, t_n$ :

$$\hat{\lambda}_t = \frac{\sum_x m_{x,t}}{\sum_x n_{x,t}}$$
$$\hat{E}_{t,x} = \hat{\lambda}_t * n_{x,t}$$

### 1.9.1 The survdiff function

```
?survdiff
```

### 1.9.2 Example: survdiff() with drug6mp data

Now we are going to compare the placebo and 6-MP data. We need to reshape the data to make it usable with the standard survival workflow:

```
library(survival)
library(tidyr)
drug6mp.v4 <-
  drug6mp.v3 |>
  select(pair, remstat, t1, t2, outcome) |>
  # here we are going to change the data from a wide format to long:
  pivot_longer(
    cols = c(t1, t2),
    names_to = "treatment",
    values_to = "exit_time"
  ) |>
  mutate(
    treatment = treatment |>
      case_match(
        "t1" ~ "placebo",
        "t2" ~ "6-MP"
      ),
    outcome = if_else(
      treatment == "placebo",
      "relapsed",
      outcome
    ),
    surv = Surv(
      time = exit_time,
      event = (outcome == "relapsed")
    )
  )
```

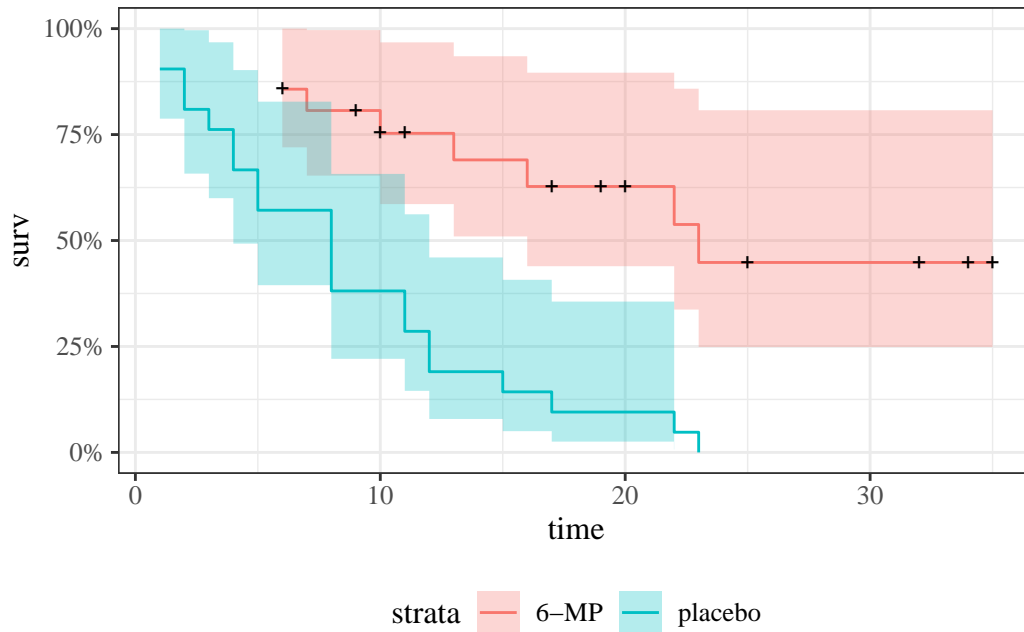
---

Using this long data format, we can fit a Kaplan-Meier curve for each treatment group simultaneously:

```
drug6mp.km_model2 <-
  survfit(
    formula = surv ~ treatment,
    data = drug6mp.v4
  )
```

We can plot the curves in the same graph:

```
drug6mp.km_model2 |> autoplot()
```



We can also perform something like a t-test, where the null hypothesis is that the curves are the same:

```
o_e_summ <- o_e |>
  summarize(
    across(starts_with("expected"), sum),
    across(starts_with("n_events_"), sum)
  )
pander::pander(o_e_summ)
```

Table 5: Observed and expected sums for the 6-MP data, for log-rank test

expected_6mp	expected_plc	n_events_placebo	n_events_6-MP
19.25	10.75	21	9

The exact variance formula for each of two groups is:

$$\text{Var}(O_i - E_i) = \sum_j \frac{n_{1j}n_{2j}(m_j)(n_j - m_j)}{(n_j)^2(n_j - 1)}$$

See Kleinbaum and Klein (2012), Chapter 2 Appendix for the exact variance formula for more than two groups.

Table 4: Observed and expected event counts for the 6-MP data, for log-rank test

```

o_e <- drug6mp.v4 |>
  arrange(exit_time) |>
  mutate(
    .by = treatment,
    n_exited = row_number(),
    n_at_risk = n() - n_exited + 1
  ) |>
  dplyr::summarize(
    .by = all_of(c("exit_time", "treatment")),
    n_at_risk = max(n_at_risk),
    n_events = sum(outcome == "relapsed")
  ) |>
  tidyr::pivot_wider(
    names_from = "treatment",
    values_from = c(n_at_risk, n_events)
  ) |>
  tidyr::fill(
    starts_with("n_at_risk"),
    .direction = "up"
  ) |>
  replace_na(list("n_events_placebo" = 0,
                  "n_events_6-MP" = 0)) |>
  mutate(
    n_at_risk = rowSums(across(starts_with("n_at_risk"))),
    n_events = rowSums(across(starts_with("n_events"))),
    marginal_hazard = n_events / n_at_risk,
    expected_6mp = marginal_hazard * `n_at_risk_6-MP`,
    expected_plc = marginal_hazard * n_at_risk_placebo,
    diff_6mp = `n_events_6-MP` - expected_6mp,
    diff_plc = n_events_placebo - expected_plc
  ) |>
  filter(n_events > 0)

o_e
#> # A tibble: 17 x 12
#>   exit_time n_at_risk_placebo `n_at_risk_6-MP` n_events_placebo `n_events_6-MP`
#>   <int>          <dbl>          <dbl>          <int>          <int>
#> 1         1             21             21             2             0
#> 2         2             19             21             2             0
#> 3         3             17             21             1             0
#> 4         4             16             21             2             0
#> 5         5             14             21             2             0
#> 6         6             12             21             0             3
#> 7         7             12             17             0             1
#> 8         8             12             16             4             0
#> 9        10              8             15             0             1
#> 10        11              8             13             2             0
#> 11        12              6             12             2             0
#> 12        13              4             12             0             1
#> 13        15              4             11             1             0
#> 14        16              3             11             0             1
#> 15        17              3             10             1             0
#> 16        22              2              7             1             1
#> 17        23              1              6             1             1
#> # i 7 more variables: n_at_risk <dbl>, n_events <dbl>, marginal_hazard <dbl>,
#> #   expected_6mp <dbl>, expected_plc <dbl>, diff_6mp <dbl>, diff_plc <dbl>

```



---

Or we can use an approximate statistic:

$$X^2 \approx \sum_{i=1}^p \frac{(O_i - E_i)^2}{E_i}$$

```
with(
  o_e_summ,
  tibble(
    "6mp" = (`n_events_6-MP` - expected_6mp)^2 / expected_6mp,
    "placebo" = (n_events_placebo - expected_plc)^2 / expected_plc,
    sum = `6mp` + placebo
  )
) |>
pander::pander()
```

---

6mp	placebo	sum
5.458	9.775	15.23

---

R gives us both the exact and approximate results:

```
survdiff(
  formula = surv ~ treatment,
  data = drug6mp.v4
)
#> Call:
#> survdiff(formula = surv ~ treatment, data = drug6mp.v4)
#>
#>               N Observed Expected (O-E)^2/E (O-E)^2/V
#> treatment=6-MP  21         9     19.3      5.46     16.8
#> treatment=placebo 21        21     10.7      9.77     16.8
#>
#>  Chisq= 16.8  on 1 degrees of freedom, p= 4e-05
```

---

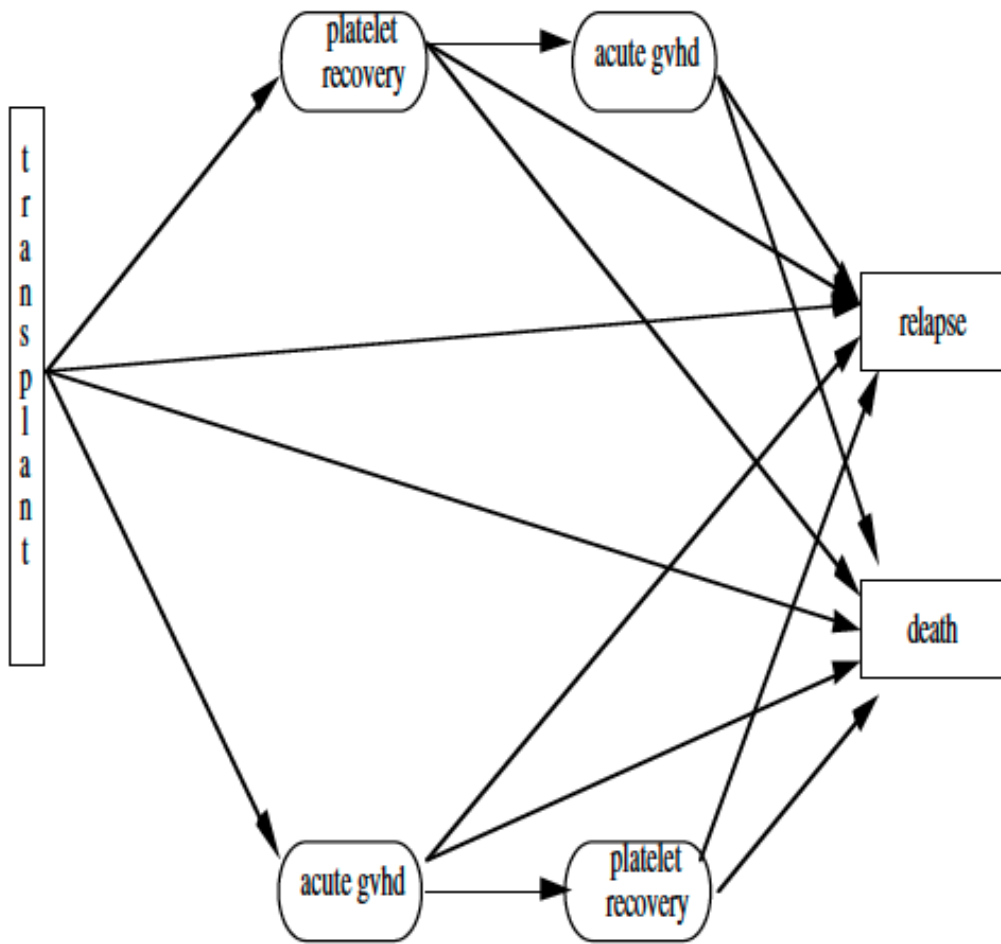
By default, `survdiff()` ignores any pairing, but we can use `strata()` to perform something similar to a paired t-test:

```
lrank_test <- survdiff(
  formula = surv ~ treatment + strata(pair),
  data = drug6mp.v4
)
lrank_test
#> Call:
#> survdiff(formula = surv ~ treatment + strata(pair), data = drug6mp.v4)
#>
#>               N Observed Expected (O-E)^2/E (O-E)^2/V
#> treatment=6-MP  21         9     16.5      3.41     10.7
#> treatment=placebo 21        21     13.5      4.17     10.7
#>
#>  Chisq= 10.7  on 1 degrees of freedom, p= 0.001
```

Interestingly, accounting for pairing reduces the significance of the difference.

## 1.10 Example: Bone Marrow Transplant Data

Data from Copelan et al. (1991)



**Figure 1.1** *Recovery Process from a Bone Marrow Transplant*

Figure 10: Recovery process from a bone marrow transplant (Fig. 1.1 from Klein and Moeschberger (2003))

### 1.10.1 Study design

#### Treatment

- allogeneic (from a donor) bone marrow transplant therapy

#### Inclusion criteria

- acute myeloid leukemia (AML)
- acute lymphoblastic leukemia (ALL).

#### Possible intermediate events

- graft vs. host disease (GVHD): an immunological rejection response to the transplant
- platelet recovery: a return of platelet count to normal levels.

One or the other, both in either order, or neither may occur.

### End point events

- relapse of the disease
- death

Any or all of these events may be censored.

### 1.10.2 KMsurv::bmt data in R

```
library(KMsurv)
?bmt
```

### 1.10.3 Analysis plan

- We concentrate for now on disease-free survival (t2 and d3) for the three risk groups, ALL, AML Low Risk, and AML High Risk.
- We will construct the Kaplan-Meier survival curves, compare them, and test for differences.
- We will construct the cumulative hazard curves and compare them.
- We will estimate the hazard functions, interpret, and compare them.

### 1.10.4 Survival Function Estimate and Variance

$$\hat{S}(t) = \prod_{t_i < t} \left[ 1 - \frac{d_i}{Y_i} \right]$$

where  $Y_i$  is the group at risk at time  $t_i$ .

The estimated variance of  $\hat{S}(t)$  is:

**Theorem 1.9** (Greenwood's estimator for variance of Kaplan-Meier survival estimator).

$$\widehat{Var}(\hat{S}(t)) = \hat{S}(t)^2 \sum_{t_i < t} \frac{d_i}{Y_i(Y_i - d_i)} \quad (3)$$

We can use Equation 3 for confidence intervals for a survival function or a difference of survival functions.

---

### Kaplan-Meier survival curves

```
library(KMsurv)
library(survival)
data(bmt)

bmt <-
  bmt |>
  as_tibble() |>
  mutate(
    group =
      group |>
      factor(
        labels = c("ALL", "Low Risk AML", "High Risk AML")
      ),
    surv = Surv(t2, d3)
  )

km_model1 <- survfit(
  formula = surv ~ group,
```

```

data = bmt
)

library(ggfortify)
autoplot(
  km_model1,
  conf.int = TRUE,
  ylab = "Pr(disease-free survival)",
  xlab = "Time since transplant (days)"
) +
  theme_bw() +
  theme(legend.position = "bottom")

```

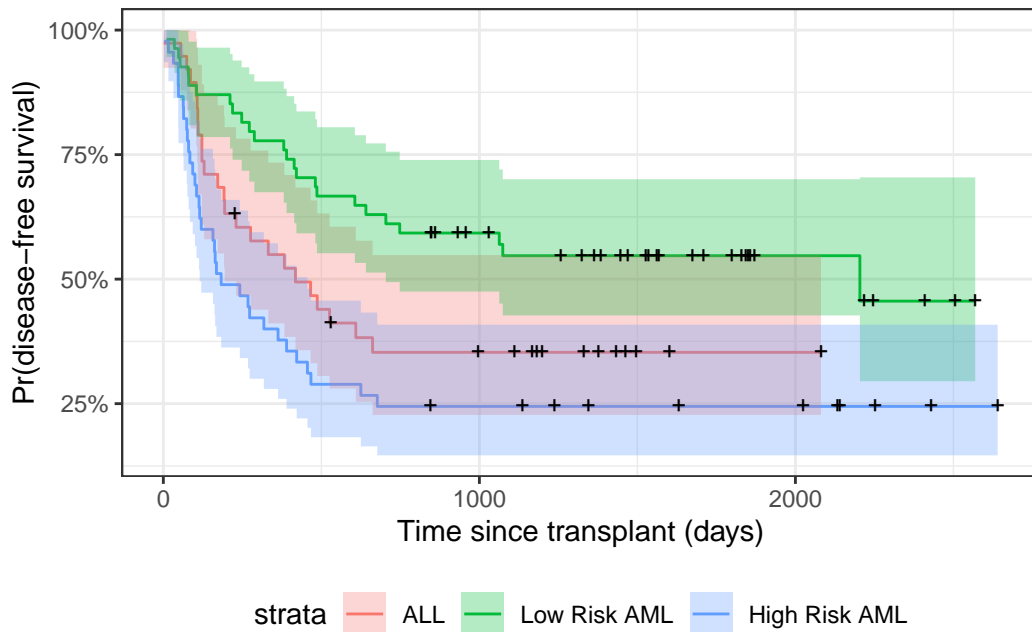


Figure 11: Disease-Free Survival by Disease Group

#### 1.10.5 Understanding Greenwood's formula (optional)

To see where Greenwood's formula comes from, let  $x_i = Y_i - d_i$ . We approximate the solution treating each time as independent, with  $Y_i$  fixed and ignore randomness in times of failure and we treat  $x_i$  as independent binomials  $\text{Bin}(Y_i, p_i)$ . Letting  $S(t)$  be the "true" survival function

$$\hat{S}(t) = \prod_{t_i < t} x_i / Y_i$$

$$S(t) = \prod_{t_i < t} p_i$$

$$\begin{aligned}
\frac{\hat{S}(t)}{S(t)} &= \prod_{t_i < t} \frac{x_i}{p_i} \frac{Y_i}{Y_i} \\
&= \prod_{t_i < t} \frac{\hat{p}_i}{p_i} \\
&= \prod_{t_i < t} \left( 1 + \frac{\hat{p}_i - p_i}{p_i} \right) \\
&\approx 1 + \sum_{t_i < t} \frac{\hat{p}_i - p_i}{p_i}
\end{aligned}$$


---

$$\begin{aligned}
\text{Var} \left( \frac{\hat{S}(t)}{S(t)} \right) &\approx \text{Var} \left( 1 + \sum_{t_i < t} \frac{\hat{p}_i - p_i}{p_i} \right) \\
&= \sum_{t_i < t} \frac{1}{p_i^2} \frac{p_i(1-p_i)}{Y_i} \\
&= \sum_{t_i < t} \frac{(1-p_i)}{p_i Y_i} \\
&\approx \sum_{t_i < t} \frac{(1-x_i/Y_i)}{x_i} \\
&= \sum_{t_i < t} \frac{Y_i - x_i}{x_i Y_i} \\
&= \sum_{t_i < t} \frac{d_i}{Y_i(Y_i - d_i)} \\
\therefore \text{Var}(\hat{S}(t)) &\approx \hat{S}(t)^2 \sum_{t_i < t} \frac{d_i}{Y_i(Y_i - d_i)}
\end{aligned}$$

### 1.10.6 Test for differences among the disease groups

Here we compute a chi-square test for association between disease group (**group**) and disease-free survival:

```

survdifff(surv ~ group, data = bmt)
#> Call:
#> survdifff(formula = surv ~ group, data = bmt)
#>
#>
#>           N Observed Expected (O-E)^2/E (O-E)^2/V
#> group=ALL      38      24    21.9     0.211     0.289
#> group=Low Risk AML 54      25    40.0     5.604    11.012
#> group=High Risk AML 45      34    21.2     7.756    10.529
#>
#> Chisq= 13.8 on 2 degrees of freedom, p= 0.001

```

### 1.10.7 Cumulative Hazard

$$\begin{aligned}
\lambda(t) &\stackrel{\text{def}}{=} p(T = t | T \geq t) \\
&= \frac{p(T = t)}{P(T \geq t)} \\
&= -\frac{\partial}{\partial t} \log\{S(t)\}
\end{aligned}$$

The **cumulative hazard** (or **integrated hazard**) function is

$$\Lambda(t) \stackrel{\text{def}}{=} \int_0^t \lambda(t) dt$$

Since  $\lambda(t) = -\frac{\partial}{\partial t} \log\{S(t)\}$  as shown above, we have:

$$\Lambda(t) = -\log\{S(t)\}$$

---

So we can estimate  $\Lambda(t)$  as:

$$\begin{aligned} \hat{\Lambda}(t) &= -\log\{\hat{S}(t)\} \\ &= -\log\left\{\prod_{t_i < t} \left[1 - \frac{d_i}{Y_i}\right]\right\} \\ &= -\sum_{t_i < t} \log\left\{1 - \frac{d_i}{Y_i}\right\} \end{aligned}$$

This is the **Kaplan-Meier (product-limit) estimate of cumulative hazard**.

---

**Example: Cumulative Hazard Curves for Bone-Marrow Transplant (bmt) data**

```
autoplot(
  fun = "cumhaz",
  km_model1,
  conf.int = FALSE,
  ylab = "Cumulative hazard (disease-free survival)",
  xlab = "Time since transplant (days)"
) +
  theme_bw() +
  theme(legend.position = "bottom")
```

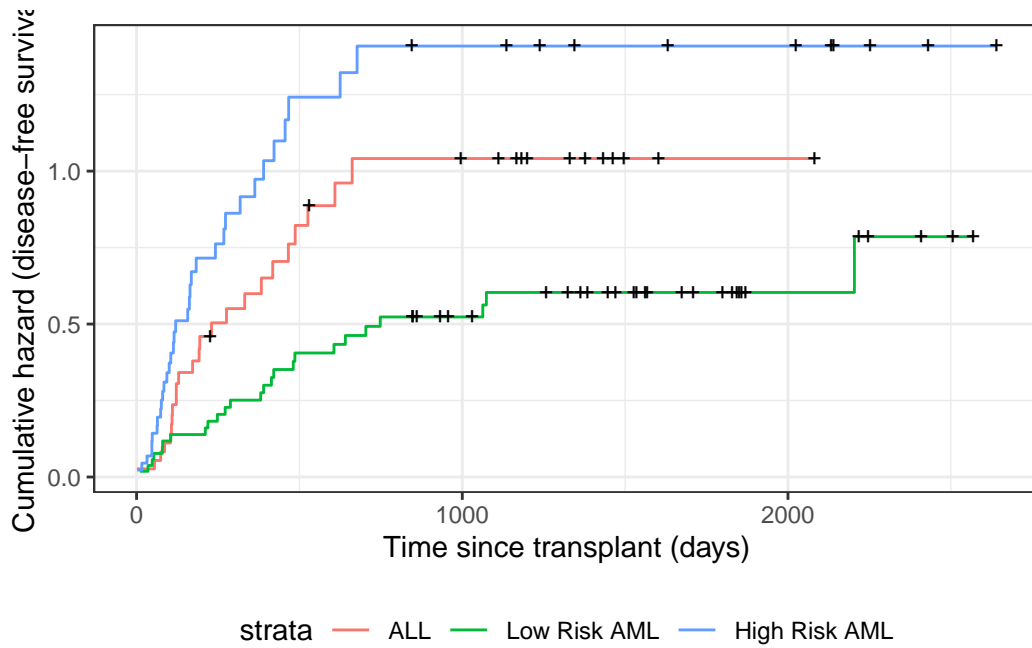


Figure 12: Disease-Free Cumulative Hazard by Disease Group

## 1.11 Nelson-Aalen Estimates of Cumulative Hazard and Survival

---

**Definition 1.7** (Nelson-Aalen Cumulative Hazard Estimator).

The point hazard at time  $t_i$  can be estimated by  $d_i/Y_i$ , which leads to the **Nelson-Aalen estimator of the cumulative hazard**:

$$\hat{\Lambda}_{NA}(t) \stackrel{\text{def}}{=} \sum_{\{i: t_i < t\}} \hat{\lambda}_i$$


---

**Theorem 1.10** (Variance of Nelson-Aalen estimator).

*The variance of this estimator is approximately:*

$$\begin{aligned} \hat{Var}(\hat{H}_{NA}(t)) &= \sum_{t_i < t} \frac{(d_i/Y_i)(1 - d_i/Y_i)}{Y_i} \\ &\approx \sum_{t_i < t} \frac{d_i}{Y_i^2} \end{aligned} \tag{4}$$


---

Since  $S(t) = \exp\{-\Lambda(t)\}$ , the Nelson-Aalen cumulative hazard estimate can be converted into an alternate estimate of the survival function:

$$\begin{aligned} \hat{S}_{NA}(t) &= \exp\{-\hat{H}_{NA}(t)\} \\ &= \exp\left\{-\sum_{t_i < t} \frac{d_i}{Y_i}\right\} \\ &= \prod_{t_i < t} \exp\left\{-\frac{d_i}{Y_i}\right\} \end{aligned}$$


---

Compare these with the corresponding Kaplan-Meier estimates:

$$\begin{aligned} \hat{H}_{KM}(t) &= -\sum_{t_i < t} \log\left\{1 - \frac{d_i}{Y_i}\right\} \\ \hat{S}_{KM}(t) &= \prod_{t_i < t} \left[1 - \frac{d_i}{Y_i}\right] \end{aligned}$$

The product limit estimate and the Nelson-Aalen estimate often do not differ by much. The latter is considered more accurate in small samples and also directly estimates the cumulative hazard. The "fleming-harrington" method for `survfit()` reduces to Nelson-Aalen when the data are unweighted. We can also estimate the cumulative hazard as the negative log of the KM survival function estimate.

### 1.11.1 Application to bmt dataset

```
na_fit <- survfit(
  formula = surv ~ group,
  type = "fleming-harrington",
  data = bmt
)

km_fit <- survfit(
  formula = surv ~ group,
  type = "kaplan-meier",
```

```

data = bmt
)

km_and_na <-
  bind_rows(
    .id = "model",
    "Kaplan-Meier" = km_fit |> fortify(surv.connect = TRUE),
    "Nelson-Aalen" = na_fit |> fortify(surv.connect = TRUE)
  ) |>
  as_tibble()

km_and_na |>
  ggplot(aes(x = time, y = surv, col = model)) +
  geom_step() +
  facet_grid(. ~ strata) +
  theme_bw() +
  ylab("S(t) = P(T>=t)") +
  xlab("Survival time (t, days)") +
  theme(legend.position = "bottom")

```

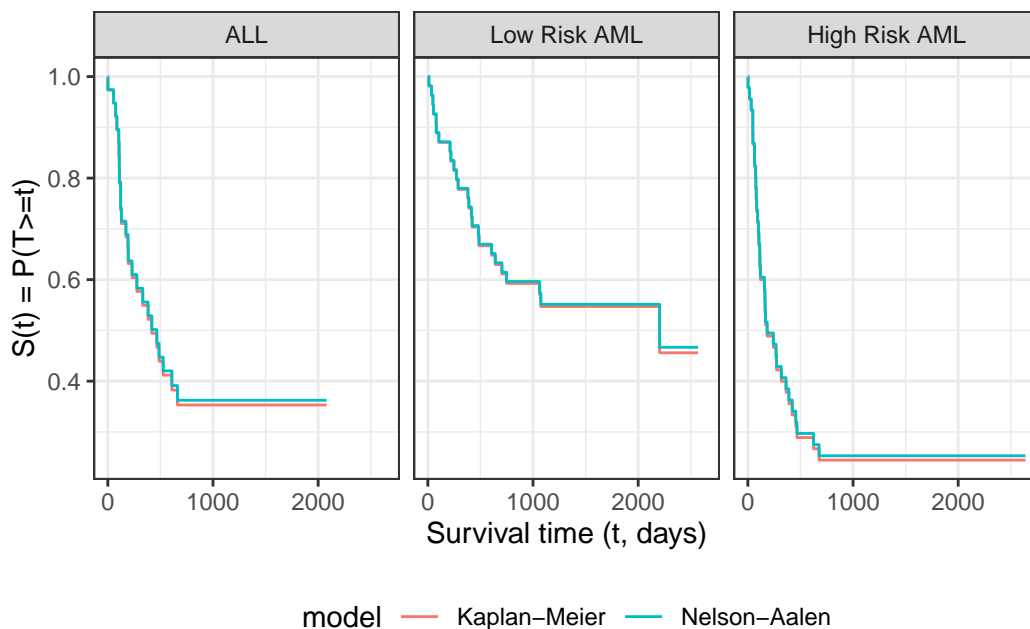


Figure 13: Kaplan-Meier and Nelson-Aalen Survival Function Estimates, stratified by disease group

The Kaplan-Meier and Nelson-Aalen survival estimates are very similar for this dataset.

- Copelan, Edward A, James C Biggs, James M Thompson, Pamela Crilley, Jeff Szer, John P Klein, Neena Kapoor, Belinda R Avalos, Isabel Cunningham, and Kerry Atkinson. 1991. "Treatment for Acute Myelocytic Leukemia with Allogeneic Bone Marrow Transplantation Following Preparation with BuCy2." <https://doi.org/10.1182/blood.V78.3.838.838>.
- Dobson, Annette J, and Adrian G Barnett. 2018. *An Introduction to Generalized Linear Models*. 4th ed. CRC press. <https://doi.org/10.1201/9781315182780>.
- Kalbfleisch, John D, and Ross L Prentice. 2011. *The Statistical Analysis of Failure Time Data*. John Wiley & Sons.
- Klein, John P, and Melvin L Moeschberger. 2003. *Survival Analysis: Techniques for Censored and Truncated Data*. Vol. 1230. Springer. <https://link.springer.com/book/10.1007/b97377>.
- Kleinbaum, David G, and Mitchel Klein. 2012. *Survival Analysis: A Self-Learning Text*. 3rd ed. Springer. <https://link.springer.com/book/10.1007/978-1-4419-6646-9>.



- Rothman, Kenneth J., Timothy L. Lash, Tyler J. VanderWeele, and Sebastien Haneuse. 2021. *Modern Epidemiology*. Fourth edition. Philadelphia: Wolters Kluwer.
- Vittinghoff, Eric, David V Glidden, Stephen C Shiboski, and Charles E McCulloch. 2012. *Regression Methods in Biostatistics: Linear, Logistic, Survival, and Repeated Measures Models*. 2nd ed. Springer. <https://doi.org/10.1007/978-1-4614-1353-0>.