Biostatistical Methods II: Classical Regression Models (EP03)
Survival Analysis

Dimitris Rizopoulos
Department of Biostatistics, Erasmus University Medical Center

d.rizopoulos@erasmusmc.nl

Biostatistical Methods II: Classical Regression Models

November 21th – December 9th, 2016

Contents

What is this Part About

- We are interested in the Time until a prespecified event of interest occurs
    - ▷ time until a patient dies from a serious disease
    - ▷ time until metastasis
    - ▷ time until a machine breaks down
    - ▷ . . .

- Statistical analysis of time-to-event outcomes (aka Survival Analysis)
    - ▷ Describe the distribution of the survival times
        - * shape of the survival distribution
        - * location measures, e.g., median survival time

What is this Part About (cont'd)

▷ Inference, i.e., understand prognostic factors (strength and shape)

- is the new treatment prolonging survival time of patients?
- are non-smokers surviving longer than smokers?
- is the new machine lasting longer than the old one?
- statistical modelling

Lexical Convention

- Throughout this part we will use several equivalent names for the Time until the event of interest occurs, namely
  - ▷ time-to-event data
  - ▷ event time data
  - ▷ event times
  - ▷ survival times
  - ▷ survival data
  - ▷ failure times
  - ▷ failure time data

Learning Objectives

- We will learn which are the special characteristics of event time data and why they require special treatment (from a statistical point of view)

- From this part it will become clear
  - ▷ which statistical tools are applicable for this kind of data,
  - ▷ which are their advantages and disadvantages, and
  - ▷ which are the optimal inferential strategies

- What is there further in survival analysis than what we will cover in this part

Agenda

- **Part I:** Introduction
  - ▷ Data sets that we will use throughout this part
  - ▷ Features of time-to-event data
  - ▷ Censoring
  - ▷ Truncation

Agenda (cont'd)

- **Part II:** Basic Tools in Survival Analysis
  - ▷ Basic tools in survival analysis
    - * Survival function
    - * Cumulative distribution function
    - * Density function
    - * Hazard function
    - * Cumulative hazard function
  - ▷ Relationships between them

Agenda (cont'd)

- **Part III:** Estimation & Statistical Inference

  ▷ Basic notation for censored event time data

  ▷ Estimating the survival function

  * the Kaplan-Meier estimator

  * the Breslow estimator

  ▷ Comparing survival functions

  * the log-rank test

  * the Peto & Peto modified Gehan-Wilcoxon test

Agenda (cont'd)

- **Part IV:** Regression Models for Time-to-Event Data
  - ▷ Accelerated failure time models
  - ▷ Cox proportional hazards model
  - ▷ Parametric proportional hazards models
  - ▷ For each of the above
    - * Estimation
    - * Interpretation of parameters
    - * Hypothesis testing
    - * Effect plots
    - * Checking the model's assumptions
    - * General statistical modelling strategies

Agenda (cont'd)

- **Part V:** Extensions of the Cox Model
  - ▷ Expected survival
  - ▷ Stratified Cox model
  - ▷ Time-dependent covariates
  - ▷ Clustered Event Time Data
  - ▷ Competing risks
  - ▷ Discrimination

Structure of this Part & Material

- Lectures & Practice Sessions:
  - ▷ theory sessions & software practicals

- Software
  - ▷ all the practice sessions will be in R – we will use both
    - * GUI based (i.e., point-and-click), and
    - * command based

    analysis

Structure of this Part & Material (cont'd)

- Material:
  - ▷ Course Notes
  - ▷ R Code Appendix
  - ▷ R code in soft format

- Within the course notes there are several examples of R code which are denoted by the symbol '`R> `'
  - ▷ more detailed examples in the R Code Appendix & during the practicals

More than what we are going to cover

References

- Standard texts
  - ▷ Kalbfleisch, J. and Prentice, R. (2002). *The Statistical Analysis of Failure Time Data, 2nd Ed.*. New York: Wiley.
  - ▷ Cox, D. and Oakes, D. (1984). *Analysis of Survival Data*. London: Chapman & Hall.
  - ▷ Parmar, M. and Machin, D. (1996). *Survival Analysis: A Practical Approach*. New York: Wiley.
  - ▷ Therneau, T. and Grambsch, P. (2000). *Modeling Survival Data: Extending the Cox Model*. New York: Springer-Verlag.
  - ▷ Harrell, F. (2001). *Regression Modeling Strategies: With Applications to Linear Models, Logistic Regression, and Survival Analysis*. New York: Springer-Verlag.

References (cont'd)

- ▷ Klein, J. and Moeschberger, M. (2003). *Survival Analysis - Techniques for Censored and Truncated Data*. New York: Springer-Verlag.
- ▷ Kleinbaum, D. and Klein, M. (2005). *Survival Analysis - A Self-Learning Text*. New York: Springer-Verlag.

- More theoretical texts

- ▷ Fleming, T. and Harrington, D. (1991). *Counting Processes and Survival Analysis*. New York: Wiley.
- ▷ Andersen, P., Borgan, O., Gill, R. and Keiding, N. (1993). *Statistical Models Based on Counting Processes*. New York: Springer-Verlag.

References (cont'd)

- Some of the books referenced above also contain software examples in R, SAS and other statistical software programs

- Intro to R with some code for survival analysis
  - ▷ Dalgaard, P. (2008) *Introductory Statistics with R, 2nd Ed*. New York: Springer-Verlag.
  - ▷ Venables, W. and Ripley, B. (2002) *Modern Applied Statistics with S*. New York: Springer-Verlag.

Software Resources

- For R
  - ▷ `http://cran.r-project.org/web/views/Survival.html` contains several packages dedicated to several aspects of survival analysis

- For SAS
  - ▷ have a look at the manuals for `Proc LIFETEST`, `Proc LIFEREG` and `Proc PHREG`

Interaction

- Interaction will be important for the comprehension of all the material that we will cover

- Therefore, you are welcome to interrupt and ask questions

Part I

Introduction

1.1 Data Sets – Stanford

- Survival of 184 patients on the waiting list for the Stanford heart transplant program

- Outcomes of interest:
  - ▷ time to death
  - ▷ age
  - ▷ T5 tissue mismatch score

1.1 Data Sets – Lung

- A study of prognostic variables in 228 lung cancer patients conducted by the North Central Cancer Treatment Group

- Outcomes of interest:
  - ▷ time to death
  - ▷ age
  - ▷ sex
  - ▷ ECOG performance score (physician's estimate); values: 0 – 4
  - ▷ Karnofsky performance score (physician's estimate); values: $20, 30, \ldots, 100$

1.1 Data Sets – AIDS

- 467 HIV infected patients who had failed or were intolerant to zidovudine therapy (AZT)

- The aim of this study was to compare the efficacy and safety of two alternative antiretroviral drugs

- Outcomes of interest:
  - time to death
  - randomized treatment: 230 patients didanosine (ddI) and 237 zalcitabine (ddC)
  - gender
  - AZT: failure or intolerance

1.1 Data Sets – AIDS (cont'd)

- Outcomes of interest:
  - ▷ prevOI: previous opportunistic infections
  - ▷ CD4 cell count measurements

1.1 Data Sets – PBC

- Primary Biliary Cirrhosis (PBC):
  - ▷ a chronic, fatal but rare liver disease
  - ▷ characterized by inflammatory destruction of the small bile ducts within the liver

- Data collected by Mayo Clinic from 1974 to 1984 (Murtaugh et al, Hepatology, 1994)

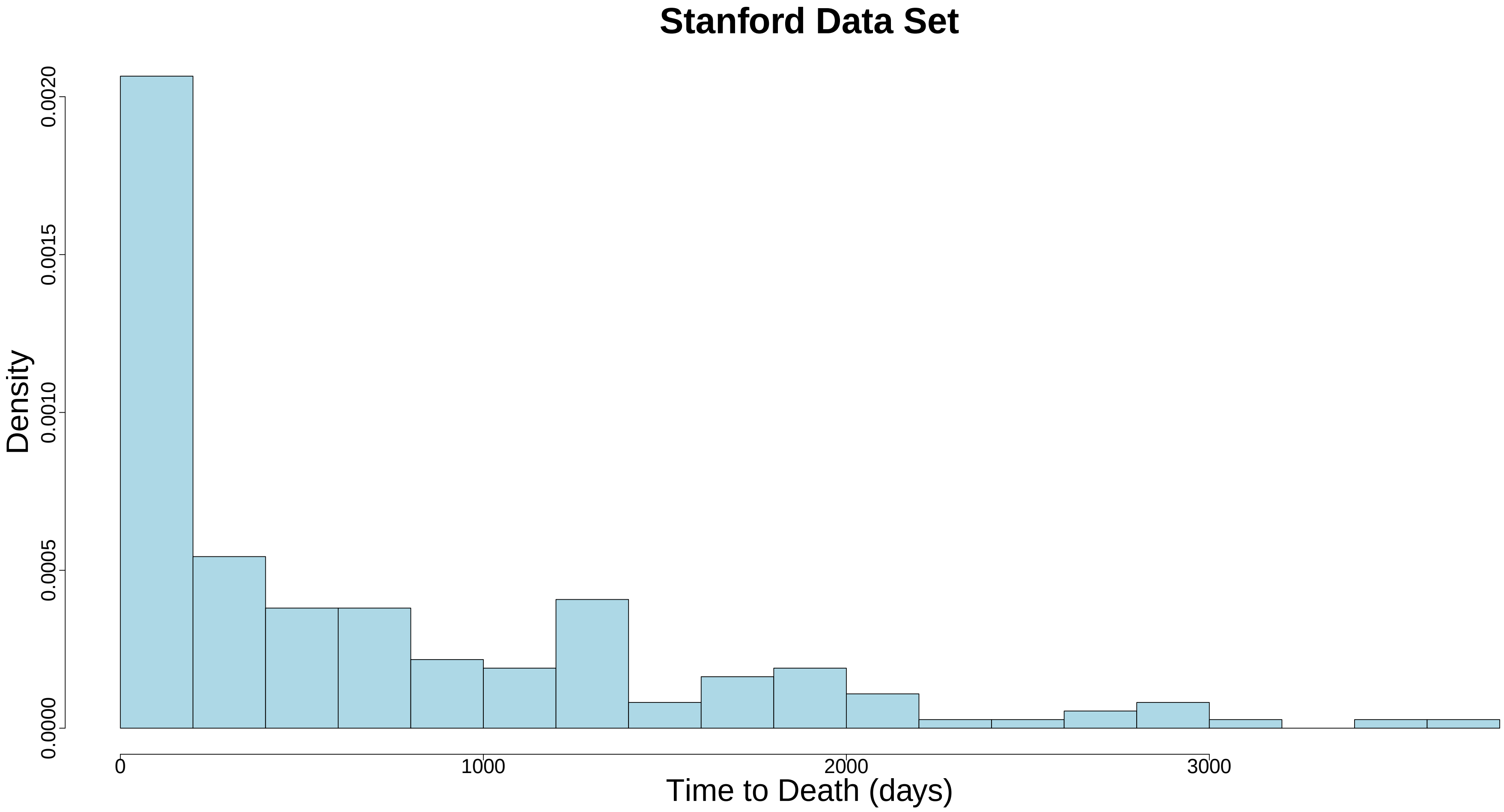
- Outcomes of interest:
  - ▷ time to death and/or time to liver transplantation
  - ▷ randomized treatment: 158 patients received D-penicillamine and 154 placebo
  - ▷ age at baseline
  - ▷ longitudinal bilirubin levels

1.1 Data Sets – Renal Graft Failure

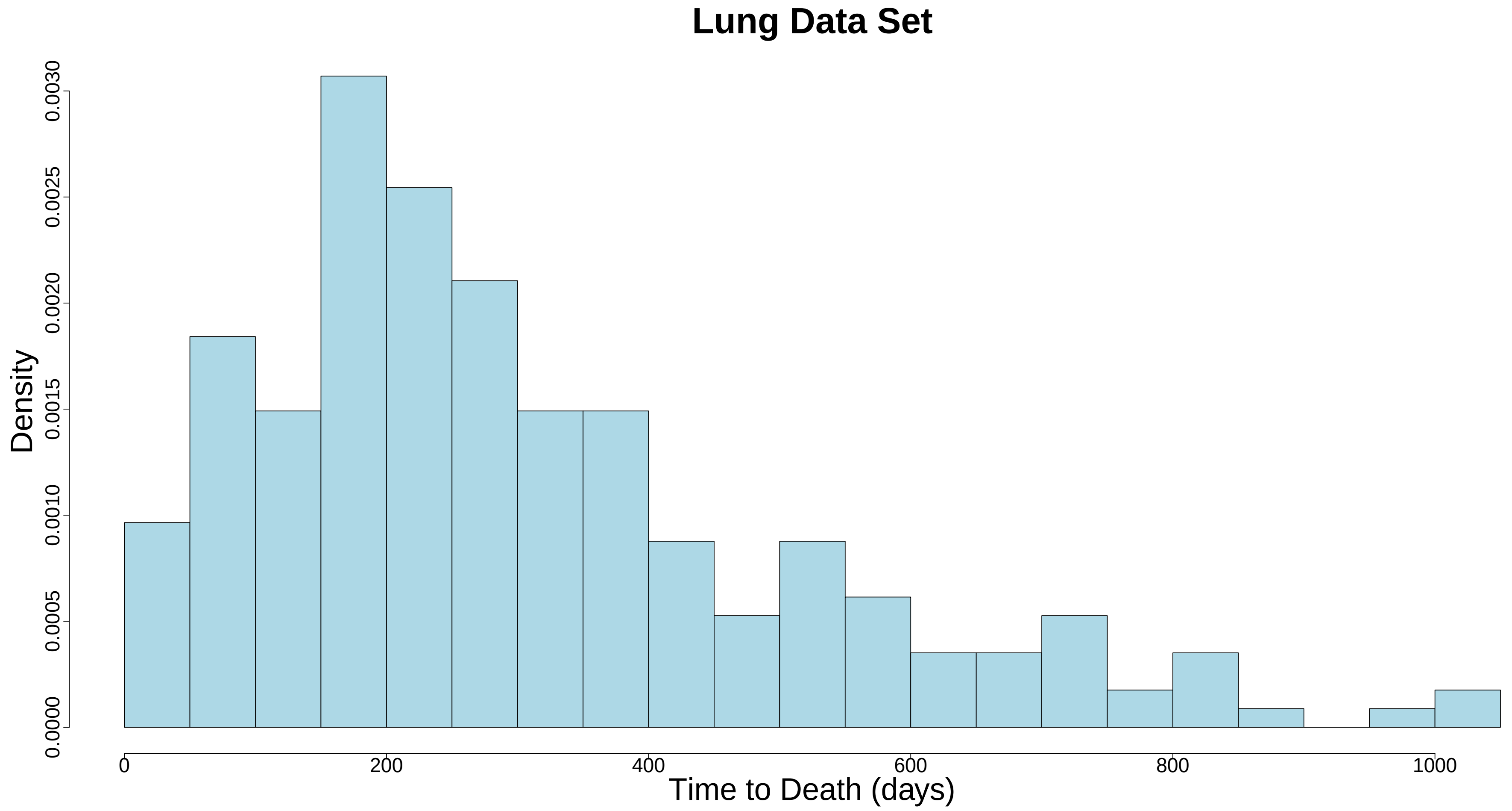
- 407 patients who underwent primary renal transplantation from deceased or leaving donor

- Outcomes of interest:
  - ▷ time to graft failure
  - ▷ smoking status
  - ▷ history of dialysis

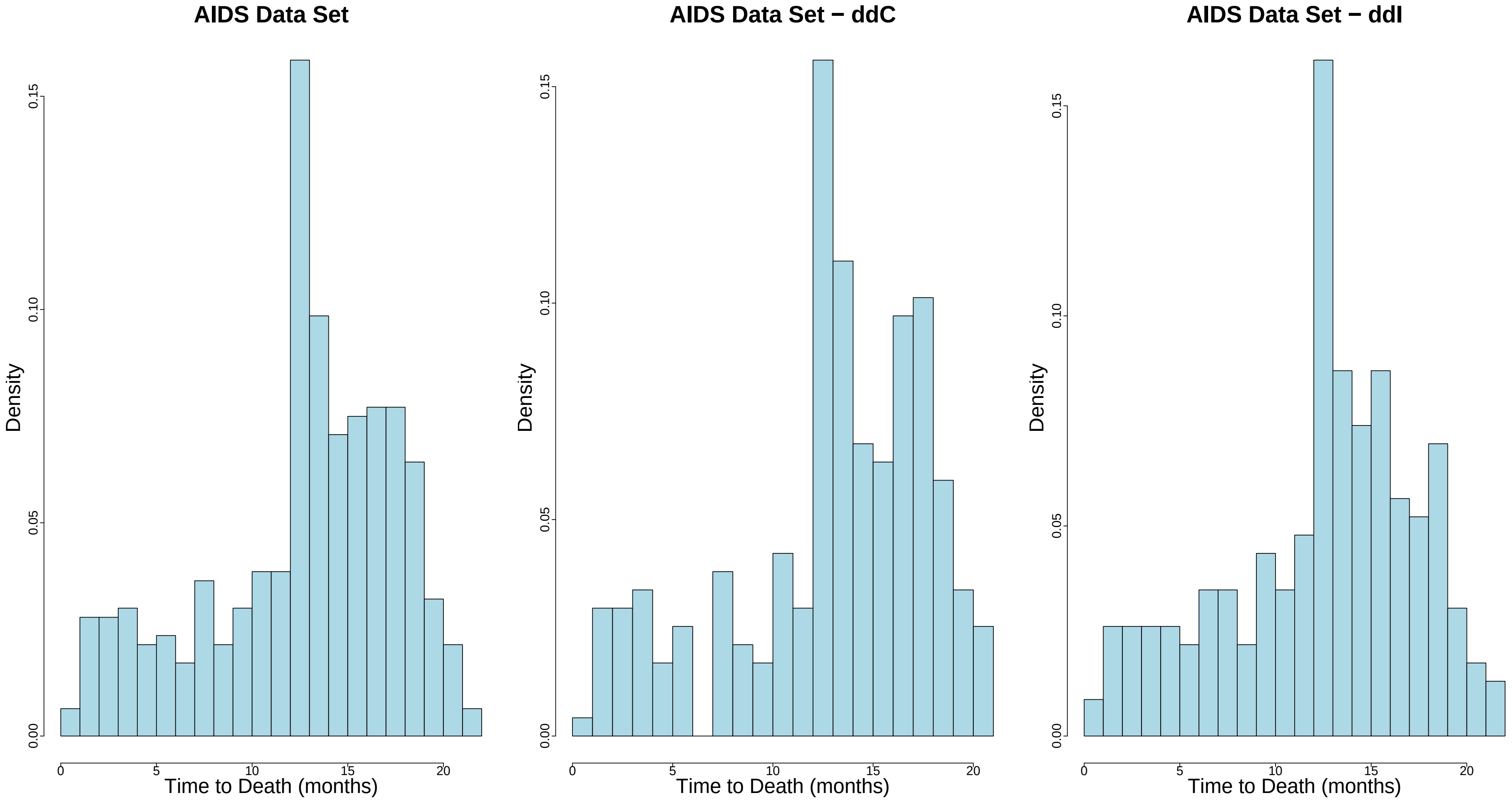
1.2 Features of Time-to-Event Outcomes



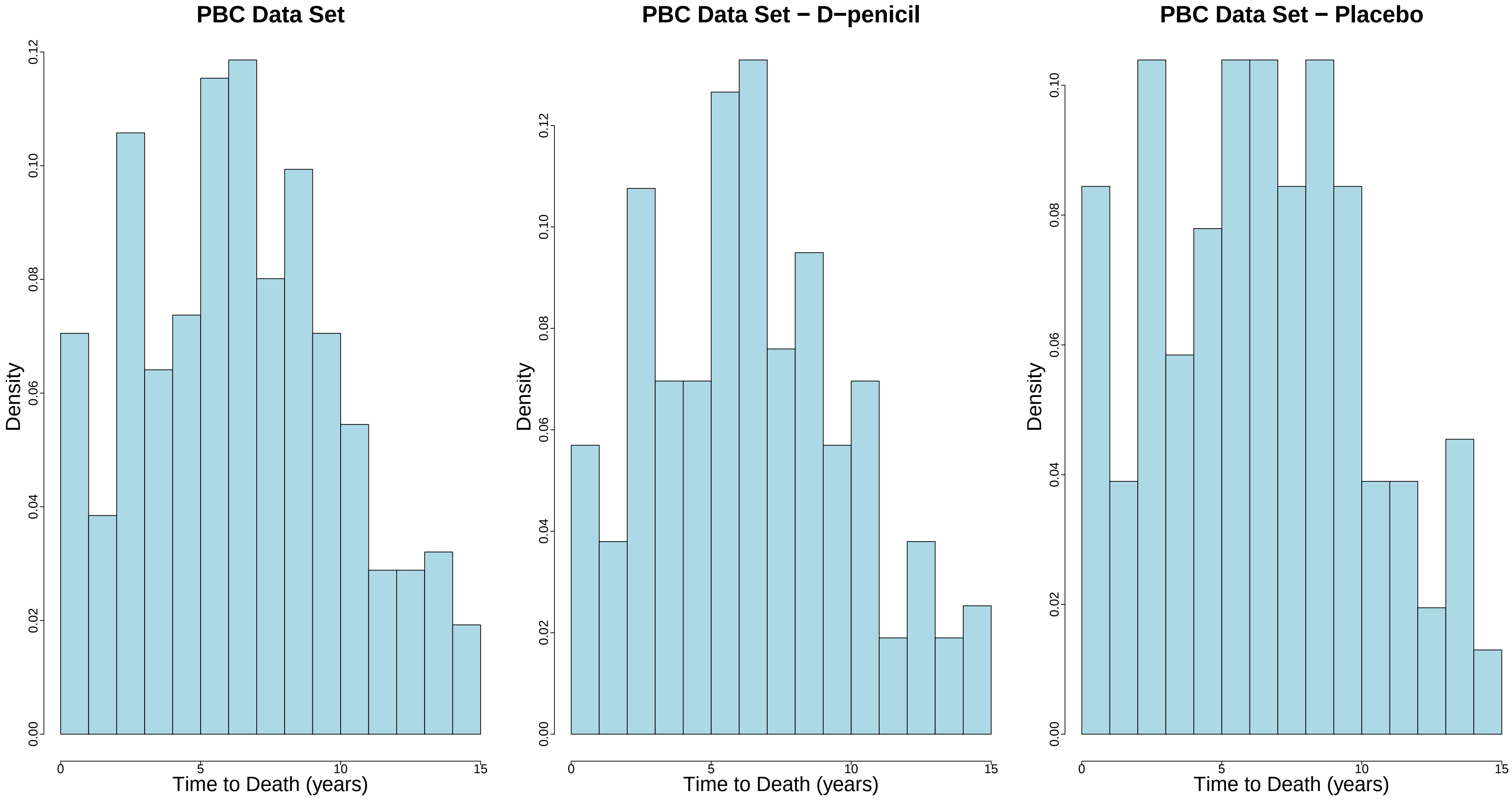
1.2 Features of Time-to-Event Outcomes



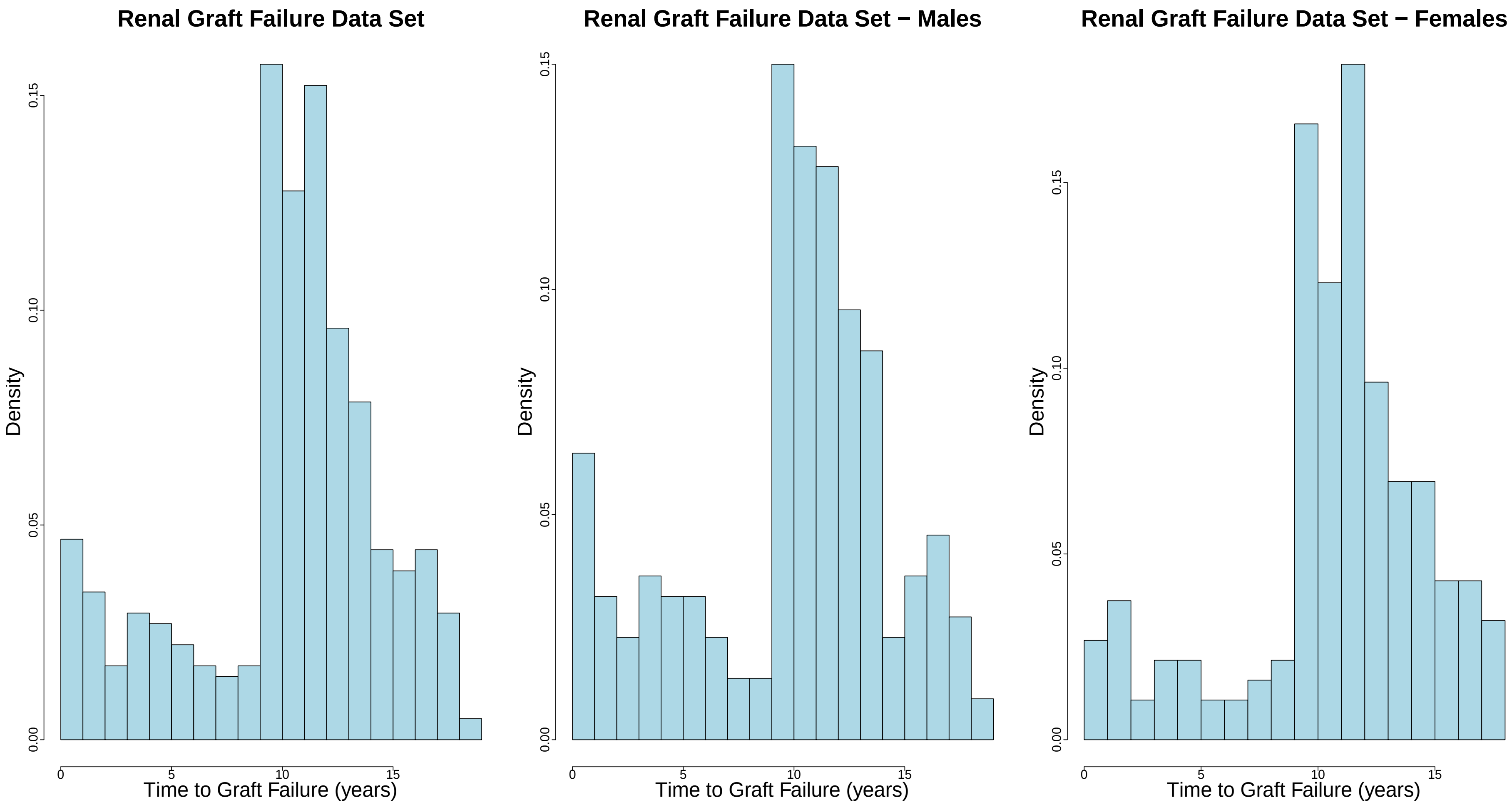
1.2 Features of Time-to-Event Outcomes



1.2 Features of Time-to-Event Outcomes (cont'd)



1.2 Features of Time-to-Event Outcomes (cont'd)



Renal Graft Failure Data Set

Renal Graft Failure Data Set − Males

Renal Graft Failure Data Set − Females

1.2 Features of Time-to-Event Outcomes (cont'd)

- Survival times are non-negative
  - ▷ in many cases the time to failure can have unusual distribution, i.e., does not look like a Normal
  - ▷ skewed to the right or to the left

- Naive analysis of untransformed times may produce invalid results

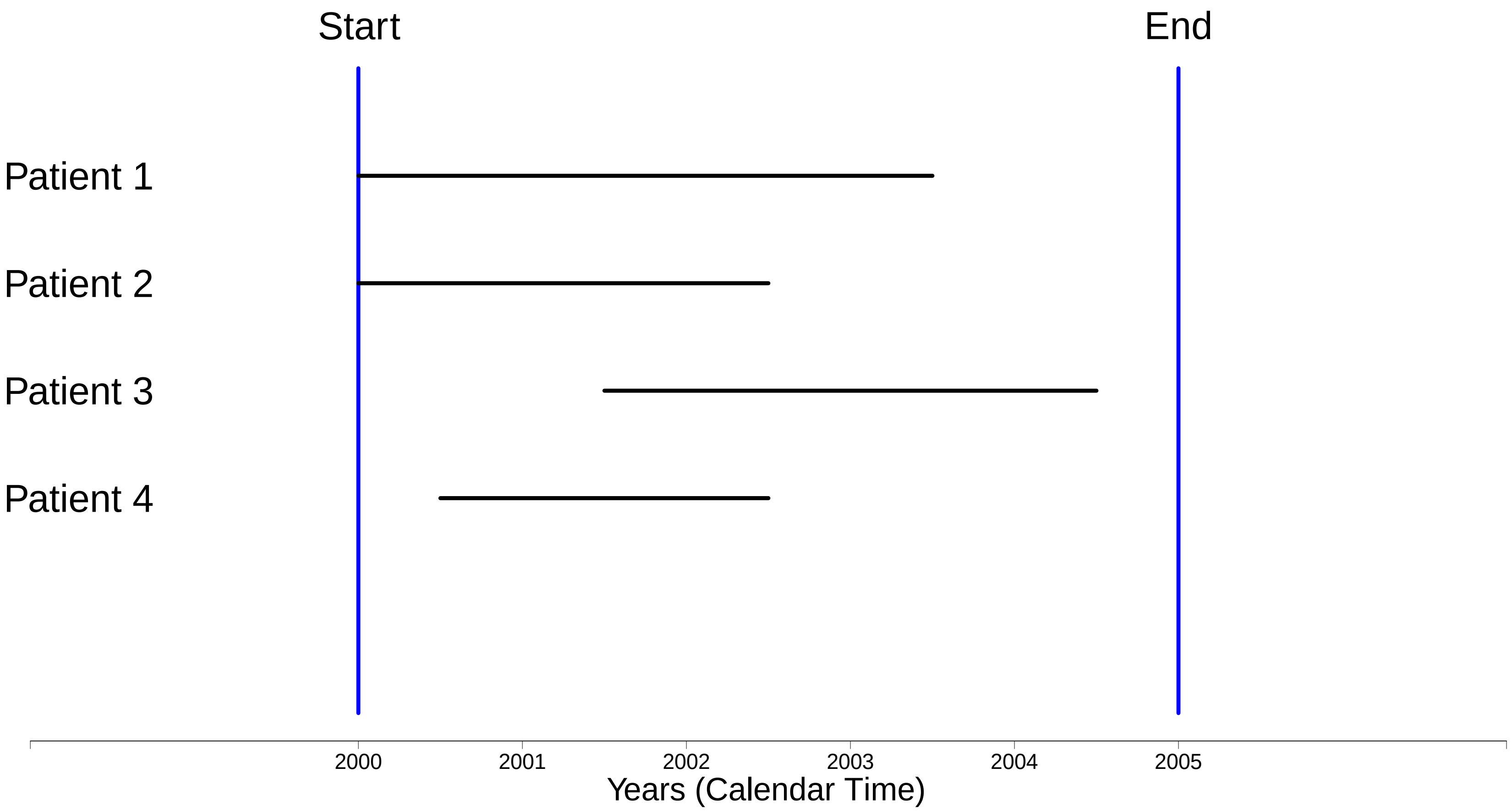
1.3 Calendar Time vs Survival Time

- Some patients enter the study at some point later than its start – that is, at different calendar times

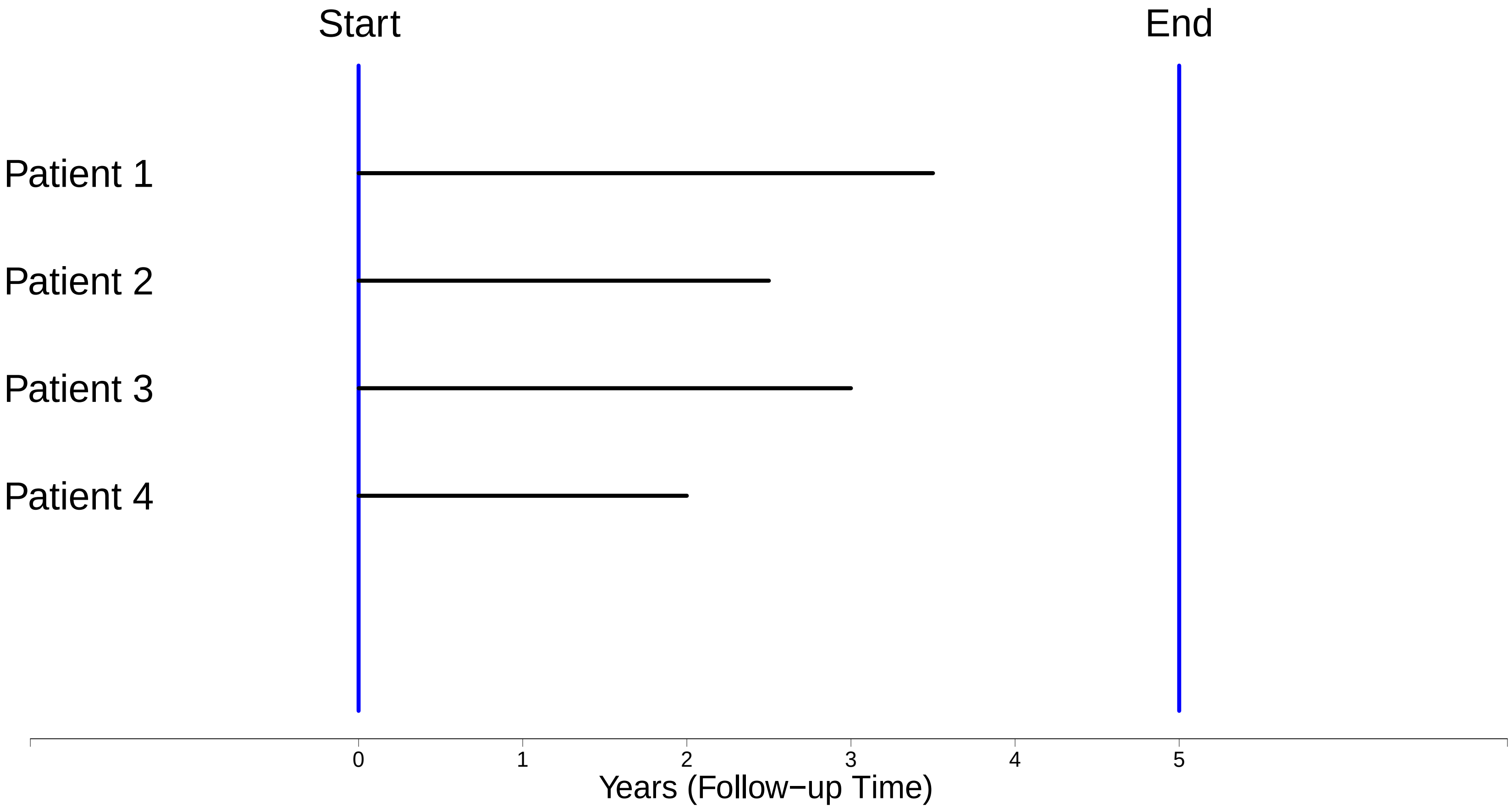
- In the analysis of failure time data we are only interested in the survival time – that is, how long did the patient survive, i.e., how long was she *at risk* for the event

- **Crucial Assumption**: the distribution of survival times of those who enter early is the same as the distribution of the ones who enter late
  - ▷ this is violated if patients who enter later are expected to live longer (or shorter)

1.3 Calendar Time vs Survival Time (cont'd)



1.3 Calendar Time vs Survival Time (cont'd)



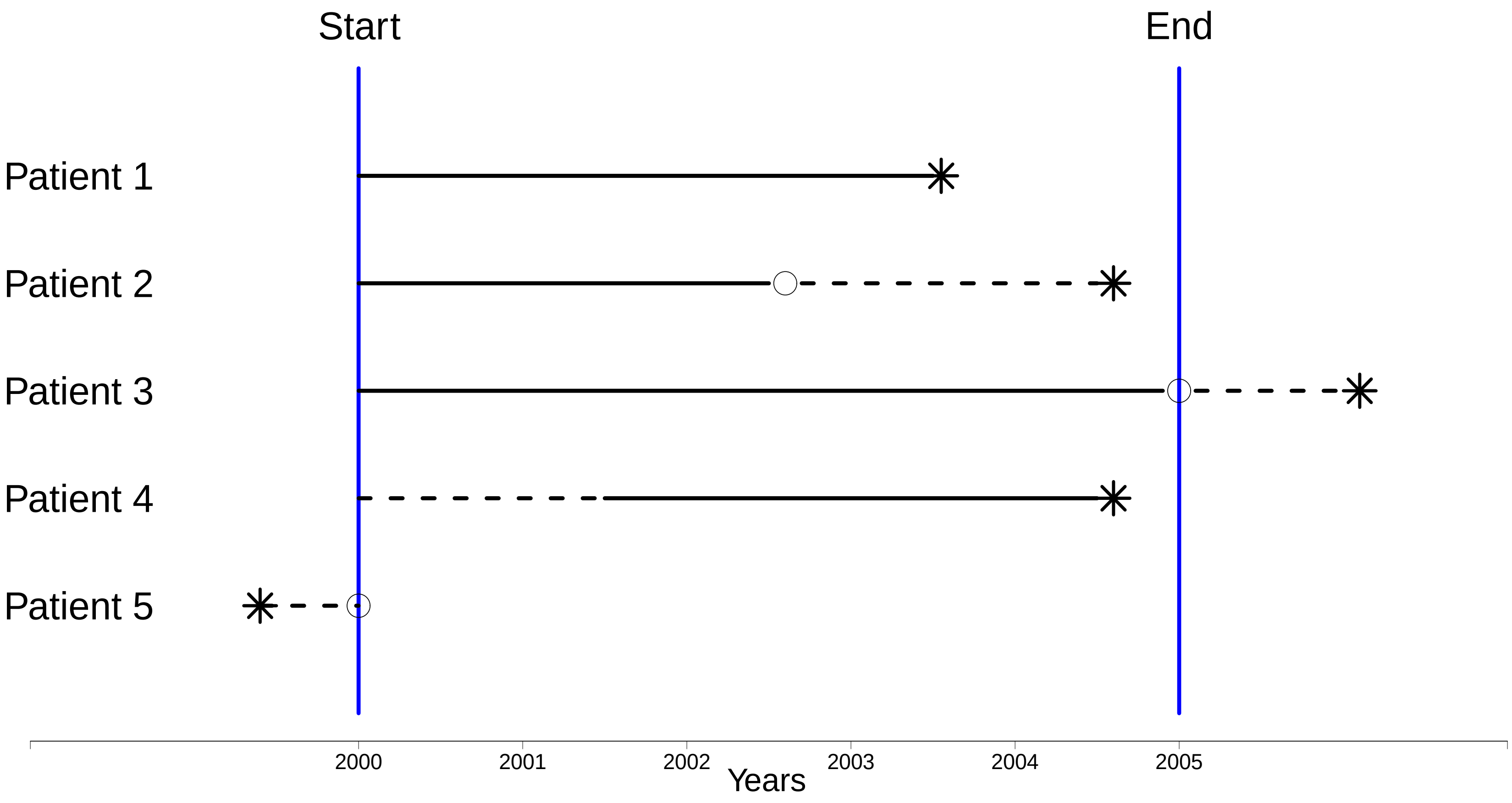
1.4 Censoring

- The time-to-event is only partially known for some patients in the study

- Types of censoring
  - ▷ right censoring
  - ▷ left censoring
  - ▷ interval censoring

- **Caution:** failure to take censoring into account can produce serious bias in estimates of the distribution of event times and related quantities

1.4 Censoring (cont'd)



1.4 Censoring (cont'd)

- Before talking in more detail about censoring ...

- Patients who had the event within the study period
  - ▷ Patient 1 was under observation from the start of the study until 3.5 years when he had the event ⇒ the time-to-event equals 3.5 years
  - ▷ Patient 4 enter the study after 1.5 years from the start (late entry), and she had the event at 4.6 years ⇒ the time-to-event equals $4.6 - 1.5 = 3.1$ years
    - * why can't we treat Patient 4 as observed for the full 5-year period since we know that she has survived 1.5 years?
    - * had this patient died before 1.5 years, she would not have had the opportunity to enroll the study, and the event would have never been observed ⇒ biases survival time upwards

1.4 Censoring (cont'd)

- Right censoring ⇒ the survival time is above a certain value

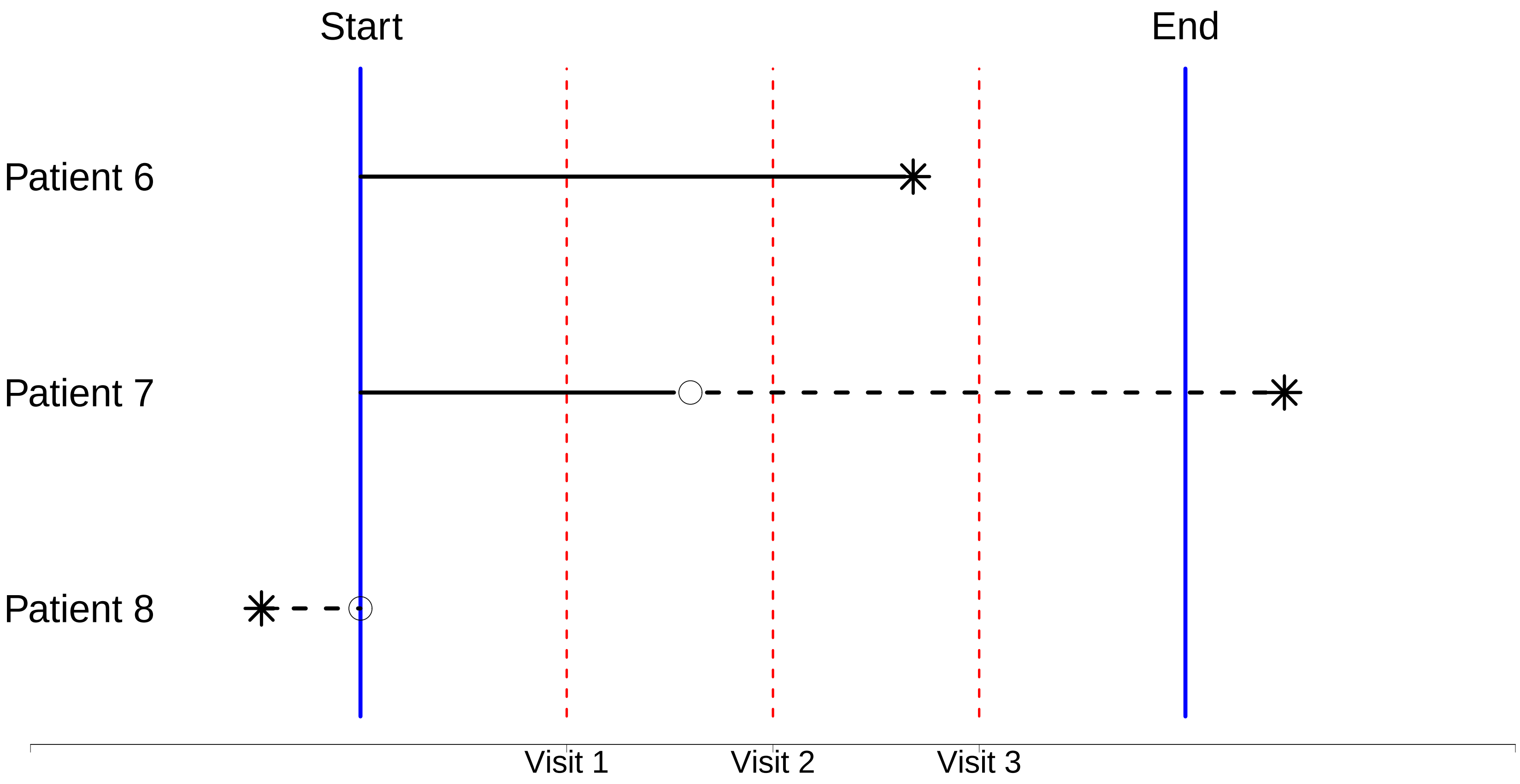
- Types of right censoring – Examples:
  - ▷ Fixed type I: Patient 3 reached the end of the study ⇒ we know this patient had the event after 5 years
  - ▷ Fixed type II: a study ends when there is a prespecified number of events
  - ▷ Random: Patient 2 moved to a new location at 2.6 years ⇒ we know this patient had the event after 2.6 years

1.4 Censoring (cont'd)

- Left censoring ⇒ the survival time is below a certain value

- Example:
  - ▷ Patient 5 had the event before the start of the study

1.4 Censoring (cont'd)



1.4 Censoring (cont'd)

- Interval censoring: ⇒ the survival time is between two values

- Example:
  - ▷ during the study period there are 3 planned visits at which it is checked whether the event has occurred
  - ▷ Patient 6 did not yet have the event at Visit 2 but she had it at Visit 3 ⇒ we know that she had the event in between Visits 2 and 3
  - ▷ Patient 7 did not yet have the event at Visit 1 and she left the study before Visit 2 ⇒ we know that she had the event at some point after Visit 1
  - ▷ Patient 8 had the event before the stat of the study

Interval censoring includes left and right censoring as special cases

1.4 Censoring (cont'd)

- Non-informative versus Informative Censoring
  - ▷ a patient is excluded from the study because he decided to move to a new location from which he cannot easily reach the study center
  - ▷ a patient is excluded from the study because his condition deteriorates (e.g., adverse event) and his physician decides to give him a rescue medication

- What is the substantiative difference in the above two situations?

1.4 Censoring (cont'd)

- Non-informative versus Informative Censoring
  - ▷ a patient is excluded from the study because he decided to move to a new location from which he cannot easily reach the study center
  - ▷ a patient is excluded from the study because his condition deteriorates (e.g., adverse event) and his physician decides to give him a rescue medication

- What is the substantiative difference in the above two situations?
  - ▷ in the second case withdrawal at time c may indicate death is likely to happen sooner than might have been expected otherwise

Informative Censoring: lost to follow-up for reasons related to the event time

1.4 Censoring (cont'd)

- Problems with informative censoring
  - ▷ biased estimates
  - ▷ inaccurate statistical inference

- Note: histograms revisited – interpretation should be done with caution in the presence of censoring

1.5 Truncation

- Truncation has a similar flavor to censoring (both are handled in a a similar manner analytically) but we should distinguish between the two terms

- Censoring period:
  - ▷ during this period the subject is no longer under observation, but she may experience the event of interest

- Truncation period
  - ▷ during this period the subject is no longer under observation, but she cannot experience the event of interest

1.5 Truncation (cont'd)

- Similarly to censoring, there are 2 types of truncation

- Left truncation: a subject enters the population at risk at some stage after the start of the study, and we know that there is no way that the event of interest could have occurred before this date

- Right truncation: a subject leaves the population at risk at some stage after the start of the study, and we know that there is no way that the event of interest could have occurred after this date

1.6 Truncation vs Censoring

- Leukemia patients are given a drug or placebo. Survival time is the duration from remission to relapse. The study ends at 52 weeks with some patients yet to relapse

A left censoring

B right censoring

C left truncation

D right truncation

1.6 Truncation vs Censoring (cont'd)

- College students are asked the age at which they first tried marijuana. Some answer never, and some report using it but forget when

A left censoring

B right censoring

C left truncation

D right truncation

1.6 Truncation vs Censoring (cont'd)

- The age at which children are able to count from 1–10 at school. Some children are already able to count before joining school

  A true event

  B interval censoring

  C left truncation

  D left censoring

1.6 Truncation vs Censoring (cont'd)

- For patients who have been hospitalized for a heart attack, we are interested in testing whether a new treatment that they take after they have been discharged prolongs survival. A patient died in the hospital

A left censoring

B left truncation

C true event

D interval censoring

1.6 Truncation vs Censoring (cont'd)

- We are interested in identifying prognostic factors for the survival of ovarian cancer patients. Only patients who have survived at least 5 years after diagnosis are included in the study

A informative left truncation

B left truncation

C left censoring

D informative left censoring

1.6 Truncation vs Censoring (cont'd)

- For patients who start feeling better, the physicians decide to exclude them from the study

A right truncation

B right censoring

C informative right truncation

D informative right censoring

1.6 Truncation vs Censoring (cont'd)

- We are interested in the years spent in retirement. However, some died before getting retired

A left censoring

B left truncation

C right censoring

D right truncation

1.7 Review of Key Points

- Time-to-event data exhibit special characteristics:
  - ▷ skewed distributions
  - ▷ censoring and/or truncation

- Standard statistical tools do not work optimally for survival data ⇒ specialized statistical techniques are required

Part II

Basic Tools in Survival Analysis

2.1 The Survival Function

- We define T to be a positive random variable denoting the time-to-event

- There are many ways to represent and describe the distribution of T

- The most useful in survival analysis is the *Survival Function*

$$S(t) = \mathsf{Pr}(T > t)$$

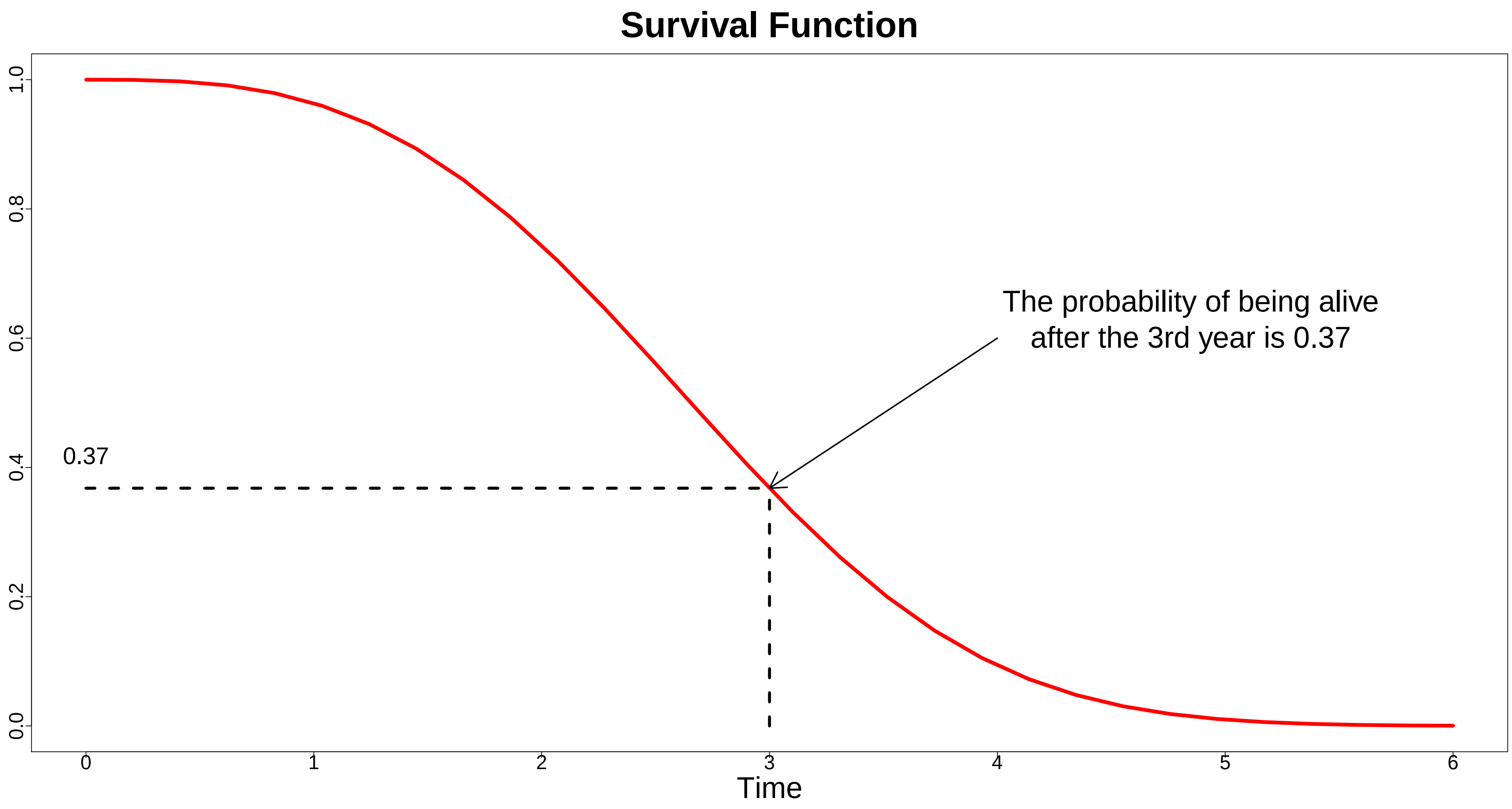
- It denotes the probability of being alive up to time t (i.e., dying after t)

2.1 The Survival Function (cont'd)

- Properties of the survival function
  - ▷ it is constrained between 0 and 1
  - ▷ it is a decreasing function of time, i.e.,
    - * at time $t = 0$ all patients are alive
    - * at time $t = \infty$ all patients have died

- $\boxed{\text{Note:}}$ in some settings patients can be cured and thus, it may not be reasonable to assume that all patients would die from the disease under study
  - ▷ a class of statistical models (aka Cure rate models) has been developed to deal with such phenomena (outside the scope of this course)

2.1 The Survival Function (cont'd)



2.2 The CDF and PDF

- The survival function is related to the cumulative distribution and the probability density function

- The *Cumulative Distribution Function* (CDF)

$$F(t) = \text{Pr}(T \leq t) = 1 - S(t)$$

  denotes the probability of dying until time t

- Properties of the CDF
  - ▷ it is constrained between 0 and 1
  - ▷ it is an increasing function of time

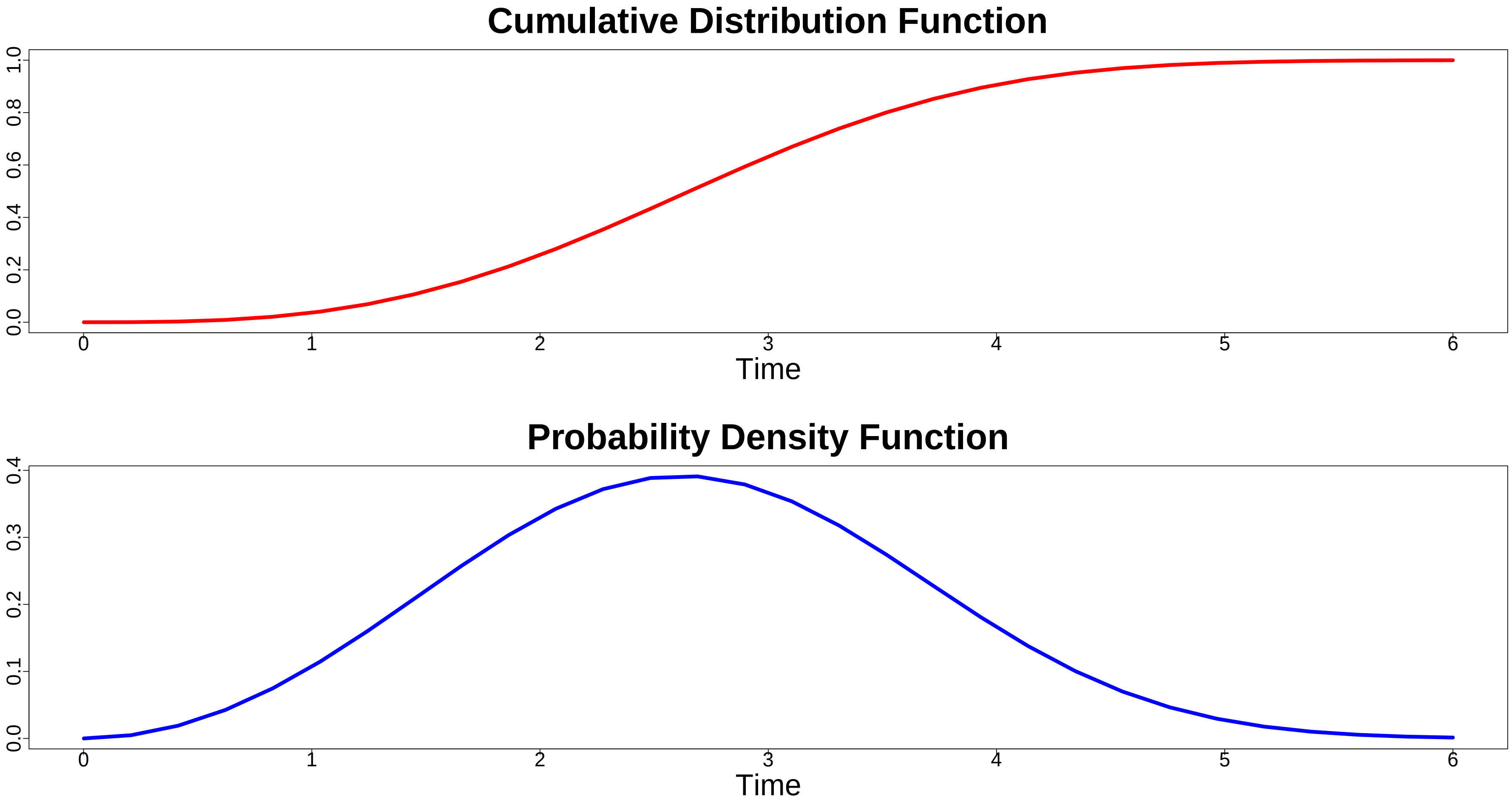
2.2 The CDF and PDF (cont'd)

- The *Probability Density Function* (pdf)

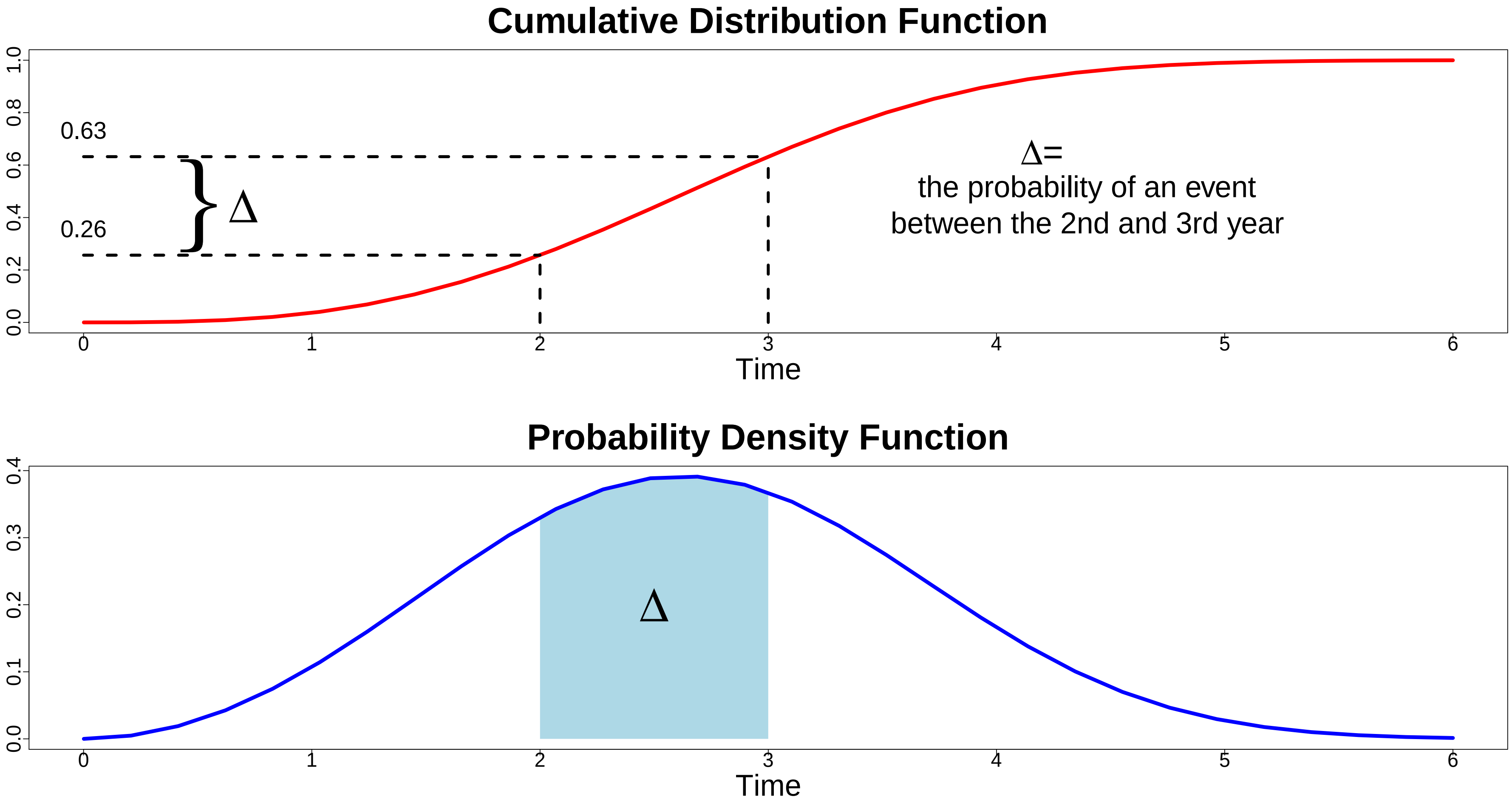
$$f(t) = \frac{dF(t)}{dt} \quad \text{or} \quad F(t) = \int_0^t f(s)\, ds$$

denotes how dense is the probability of dying in a specific time interval

2.2 The CDF and PDF (cont'd)



2.2 The CDF and PDF (cont'd)



2.3 The Hazard Function

- Another useful notion is the risk for an event

- The *Hazard Function*

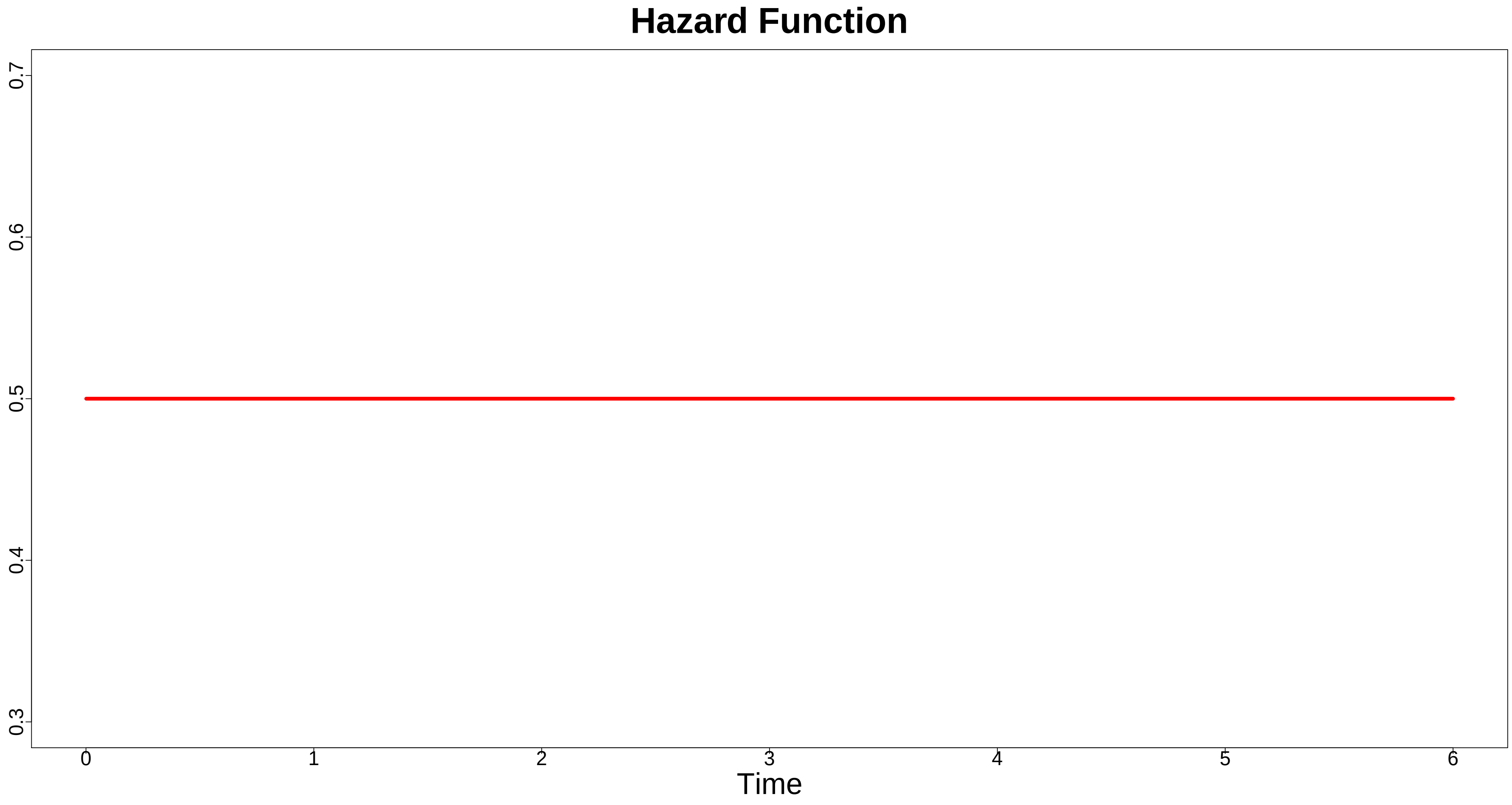
$$h(t) = \lim_{s \to 0} \frac{\mathsf{Pr}(t \leq T < t + s \mid T \geq t)}{s}$$

is the instantaneous risk of an event at time t, given that the event has not occurred until time t

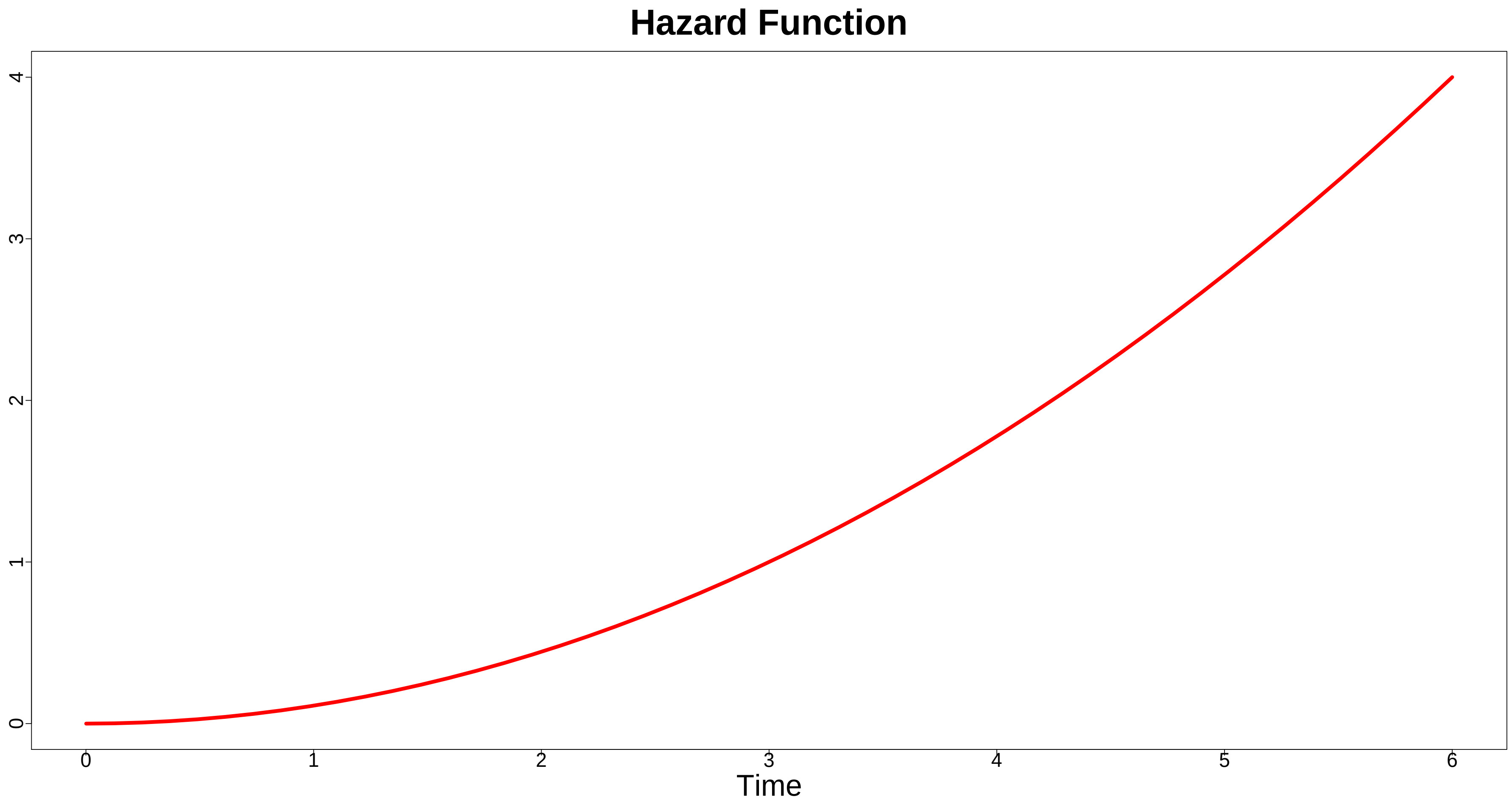
2.3 The Hazard Function (cont'd)

- $\boxed{\text{Note:}}$ the hazard is not a probability ⇒ can be interpretable as the expected number of events per individual per unit of time
  - ▷ it has to be positive
  - ▷ but it can be (much) greater than 1

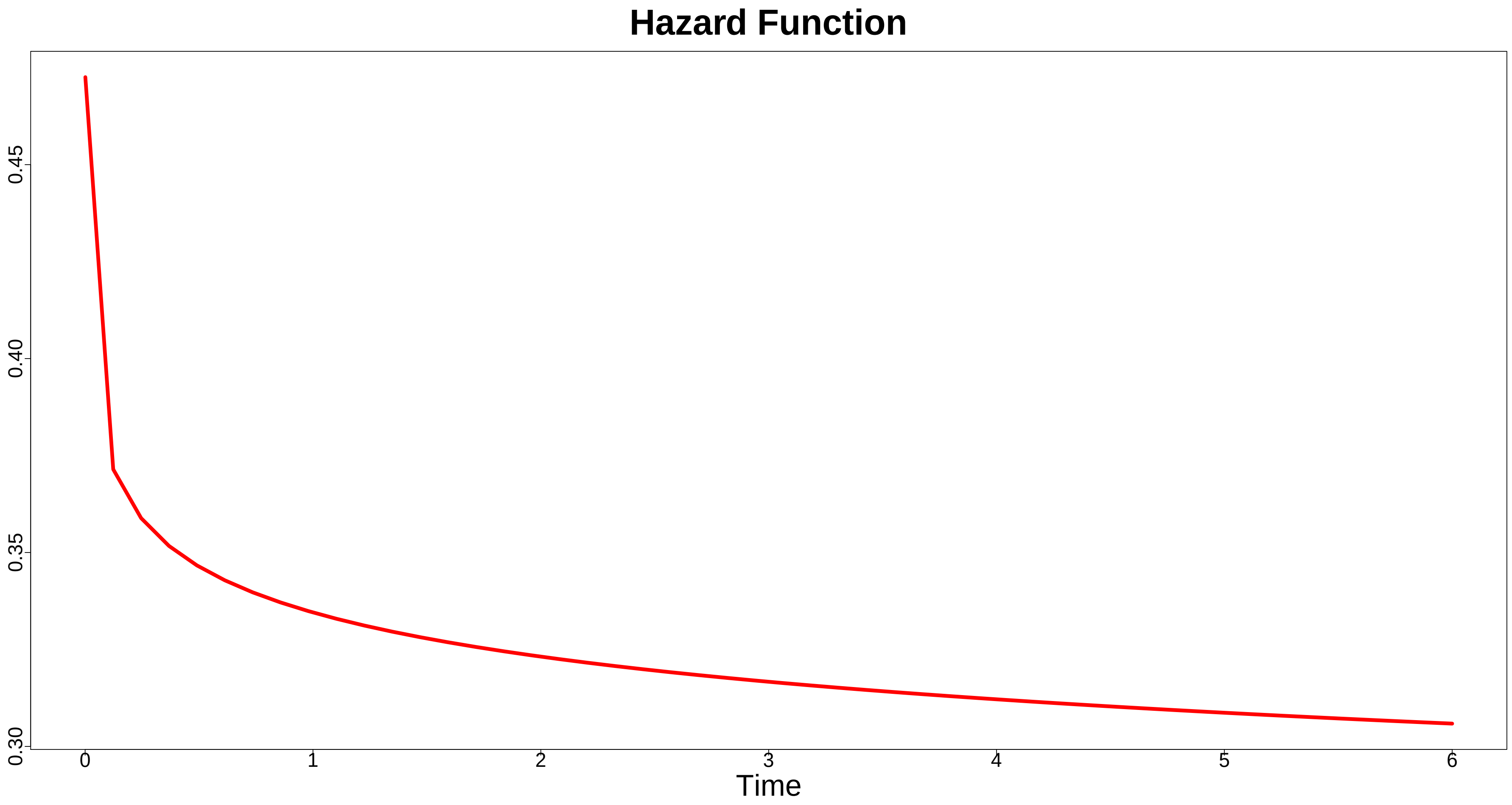
2.3 The Hazard Function (cont'd)



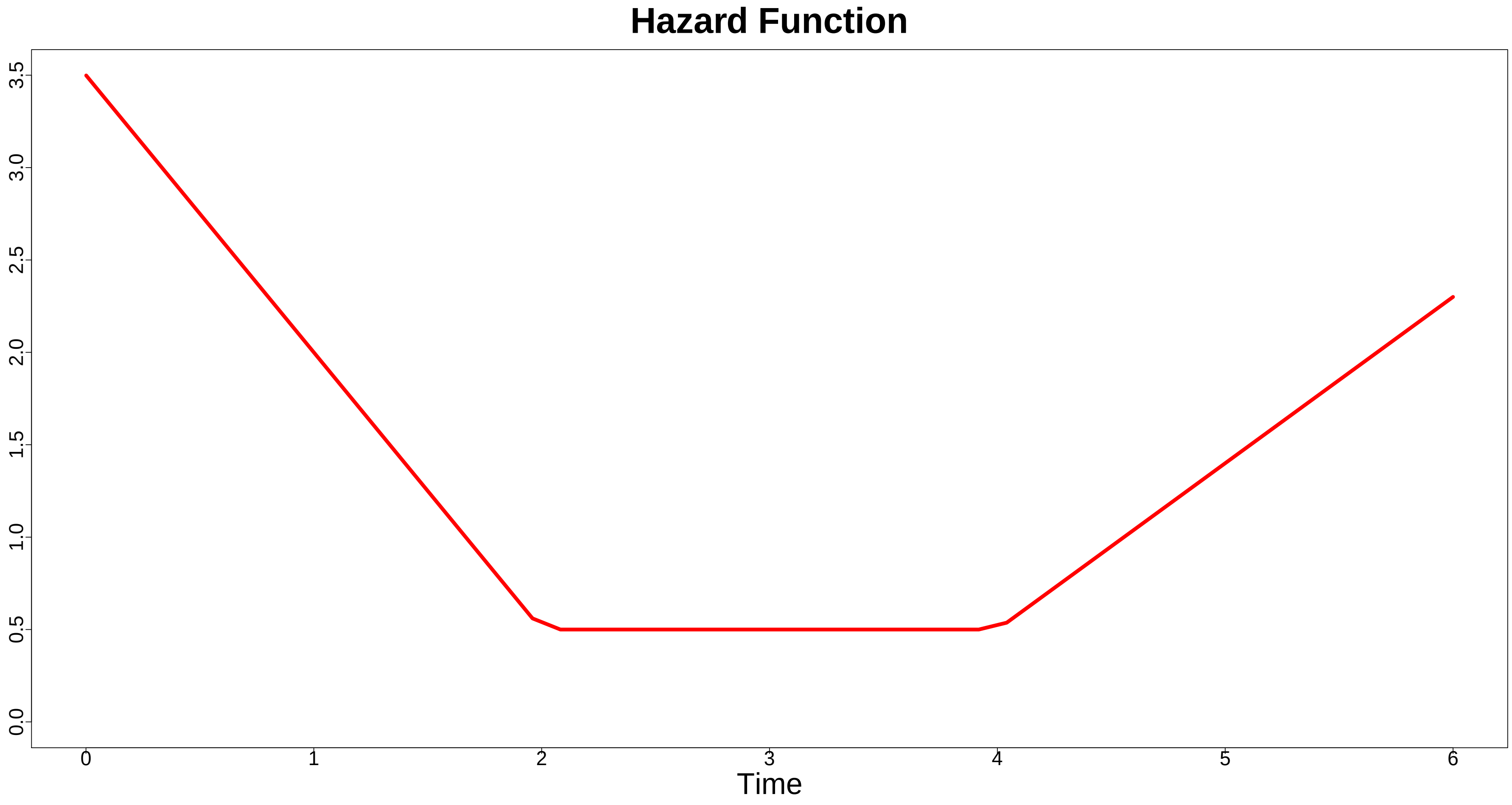
2.3 The Hazard Function (cont'd)



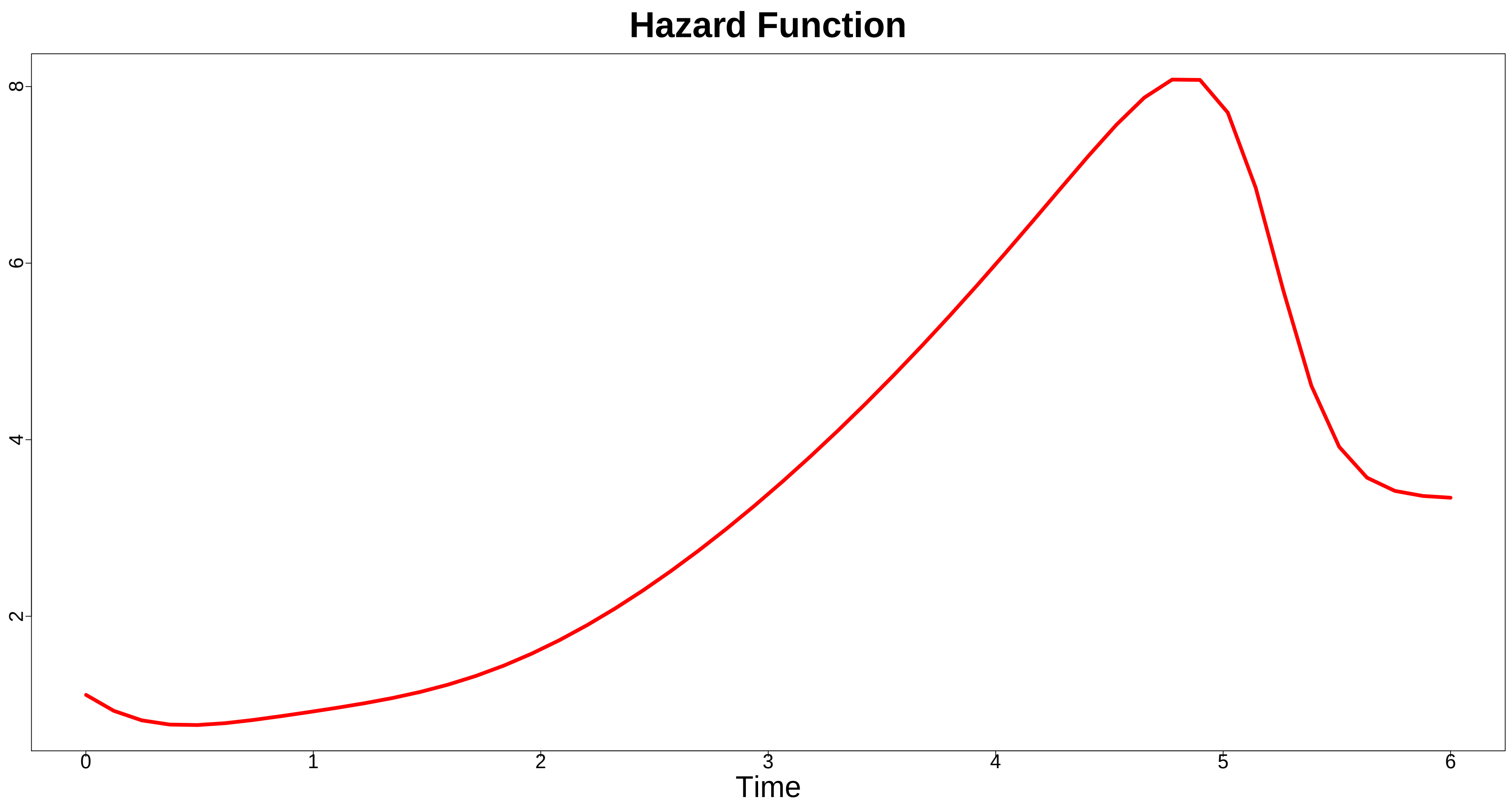
2.3 The Hazard Function (cont'd)



2.3 The Hazard Function (cont'd)



2.3 The Hazard Function (cont'd)



2.4 The Cumulative Hazard Function

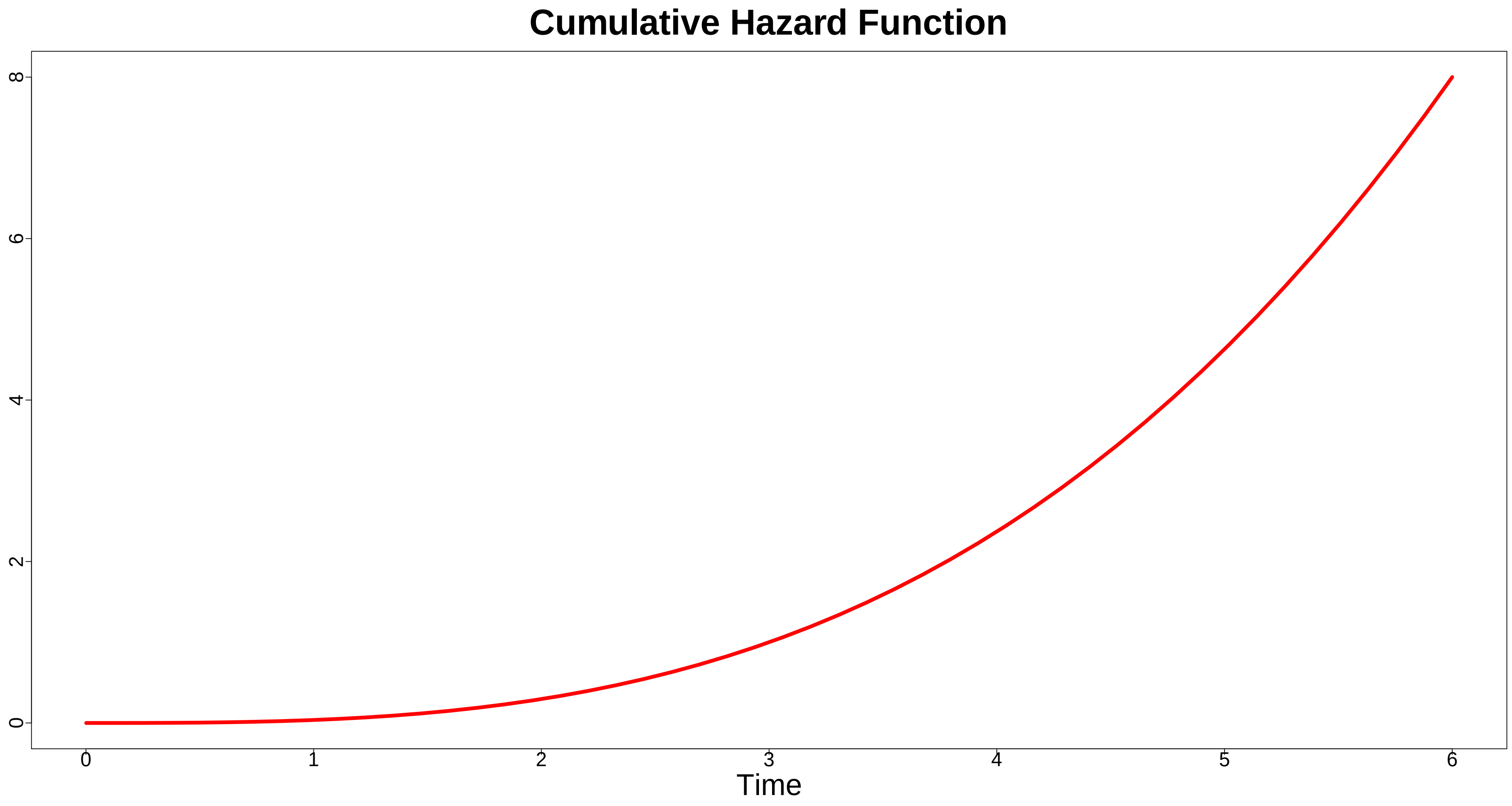
- The *Cumulative Hazard Function* is the integrated hazard function:

$$H(t) = \int_0^t h(s)\, ds$$

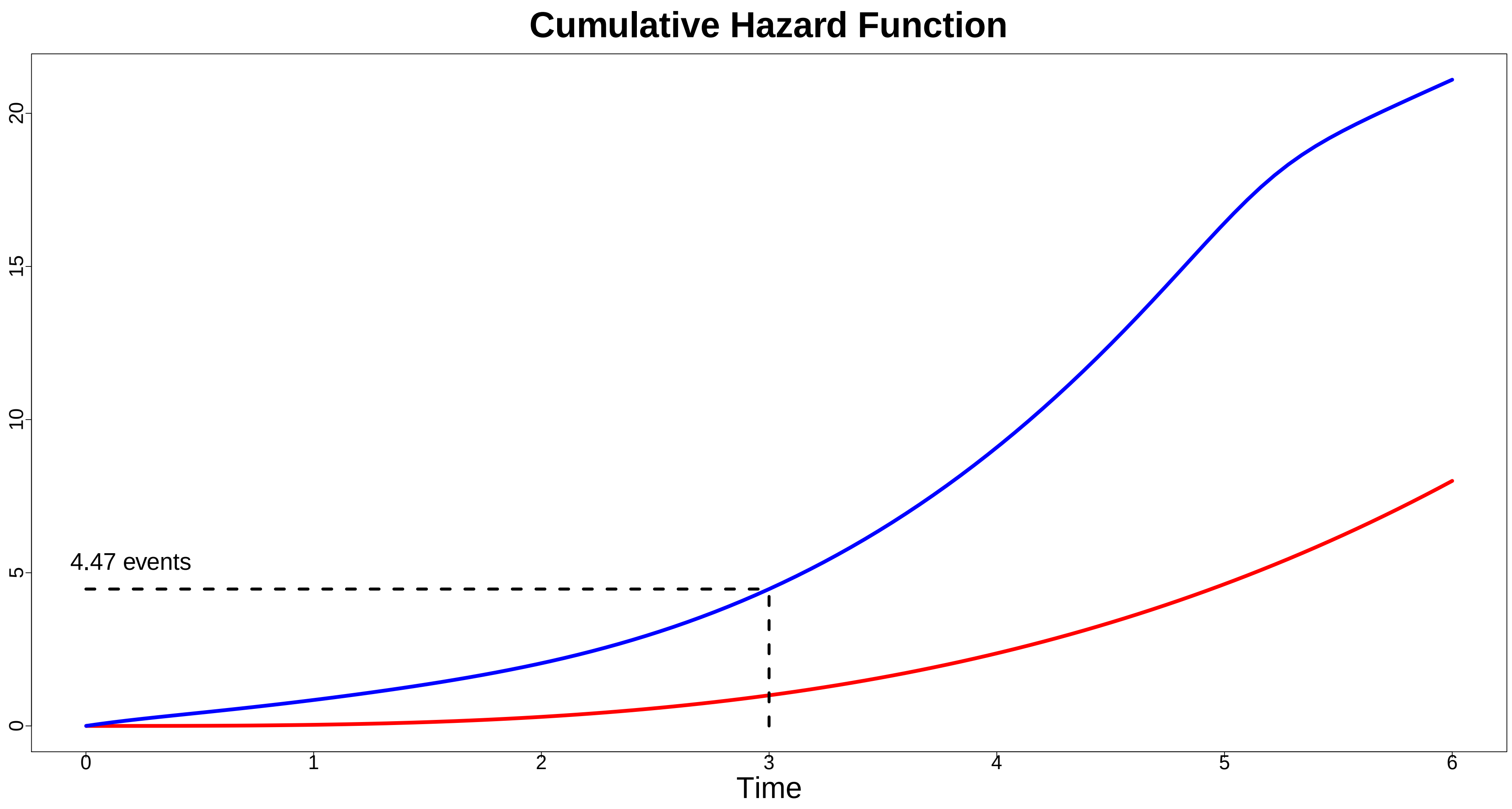
  denotes the cumulative risk up to time t, i.e., the expected number of events that have occurred by time t

- Again this is not a (cumulative) probability
  - ▷ has to be positive
  - ▷ increasing function of t (as the time progresses we expect more events to have occurred)

2.4 The Cumulative Hazard Function (cont'd)



2.4 The Cumulative Hazard Function (cont'd)

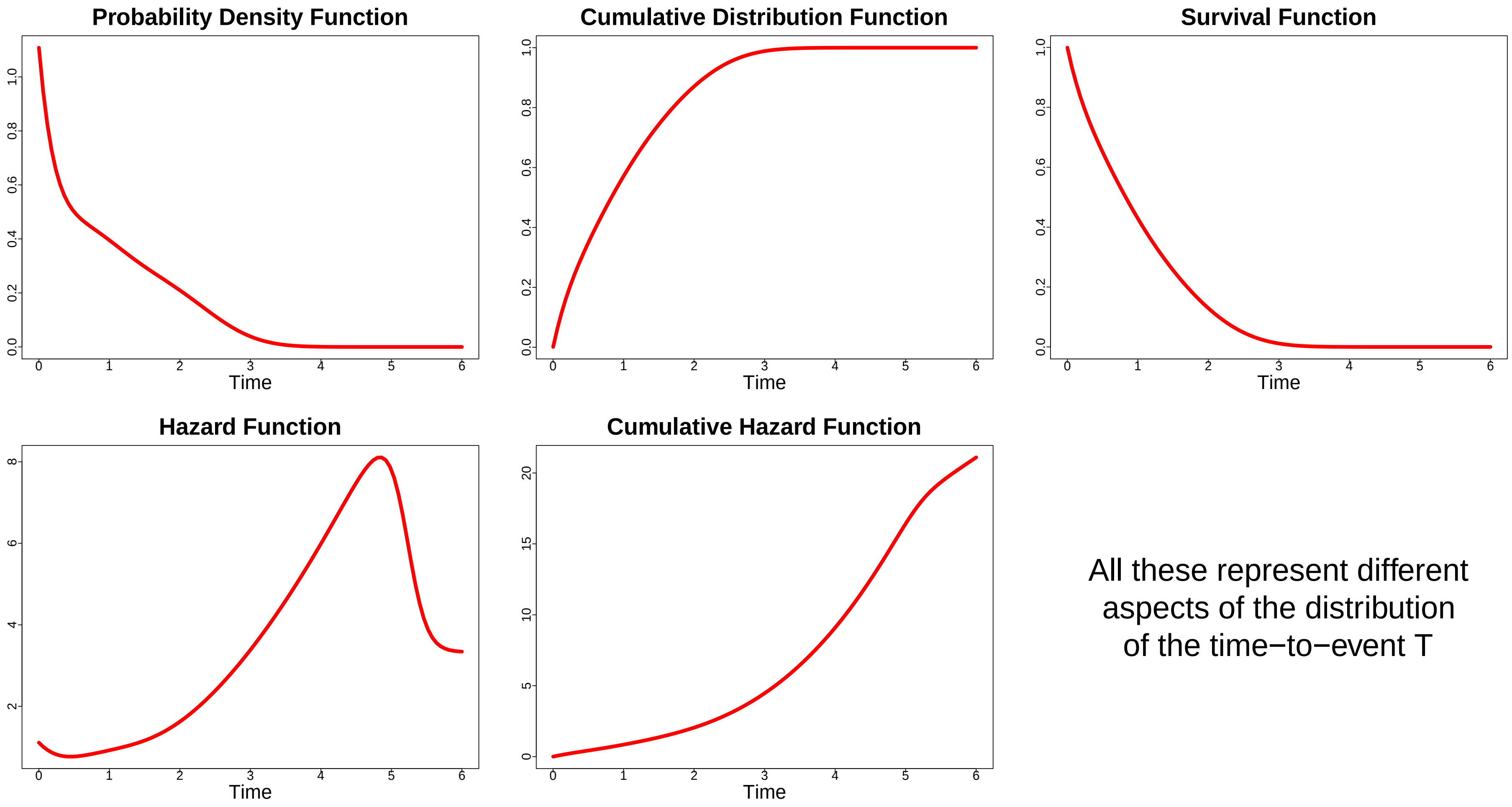


2.5 All are Relatives

- All functions we have seen so far are related
  - ▷ if you know one you know all!

$$
\begin{aligned}
f(t) &= \frac{dF(t)}{dt} \qquad F(t) = 1 - S(t)\\
h(t) &= \frac{f(t)}{S(t)} \qquad H(t) = \int_0^t h(s)\, ds\\
H(t) &= -\log S(t)\\
S(t) &= \exp\{-H(t)\} = \exp\Bigg\{-\int_0^t h(s)\, ds\Bigg\}
\end{aligned}
$$

2.5 All are Relatives (cont'd)



All these represent different aspects of the distribution of the time−to−event T

2.6 Useful Statistical Measures

- *Median Life Length* or *Median Survival* is the time by which half of the subjects will experience the event – it is defined as

$$\begin{aligned} T_{0.5} &= S^{-1}(0.5) \\ &= H^{-1}(\log 2) \end{aligned}$$

where $S^{-1}(\cdot)$ and $H^{-1}(\cdot)$ are the inverse survival and cumulative hazard functions, respectively

2.6 Useful Statistical Measures (cont'd)

- *Mean Survival* or *Average Survival* is the expected failure time – is defined as

$$\mu = \int_0^\infty t f(t)\, dt$$

$$= \int_0^\infty S(t)\, dt$$

- *Expected Future Lifetime* is the expected value of future lifetime given survival up to time point t_0

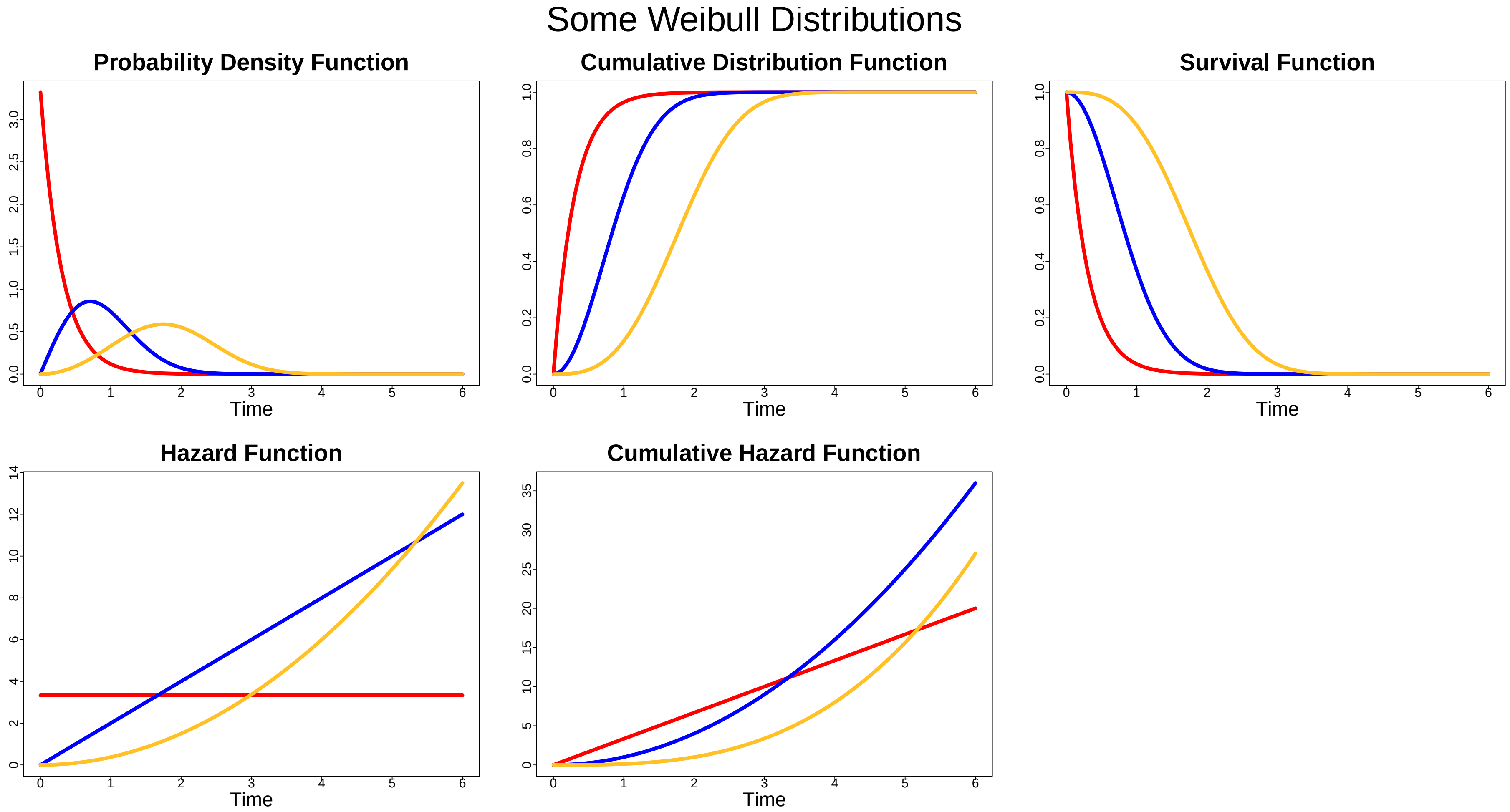
$$\tilde{\mu} = \frac{1}{S(t_0)} \int_0^\infty t f(t + t_0)\, dt$$

2.7 Survival Distributions

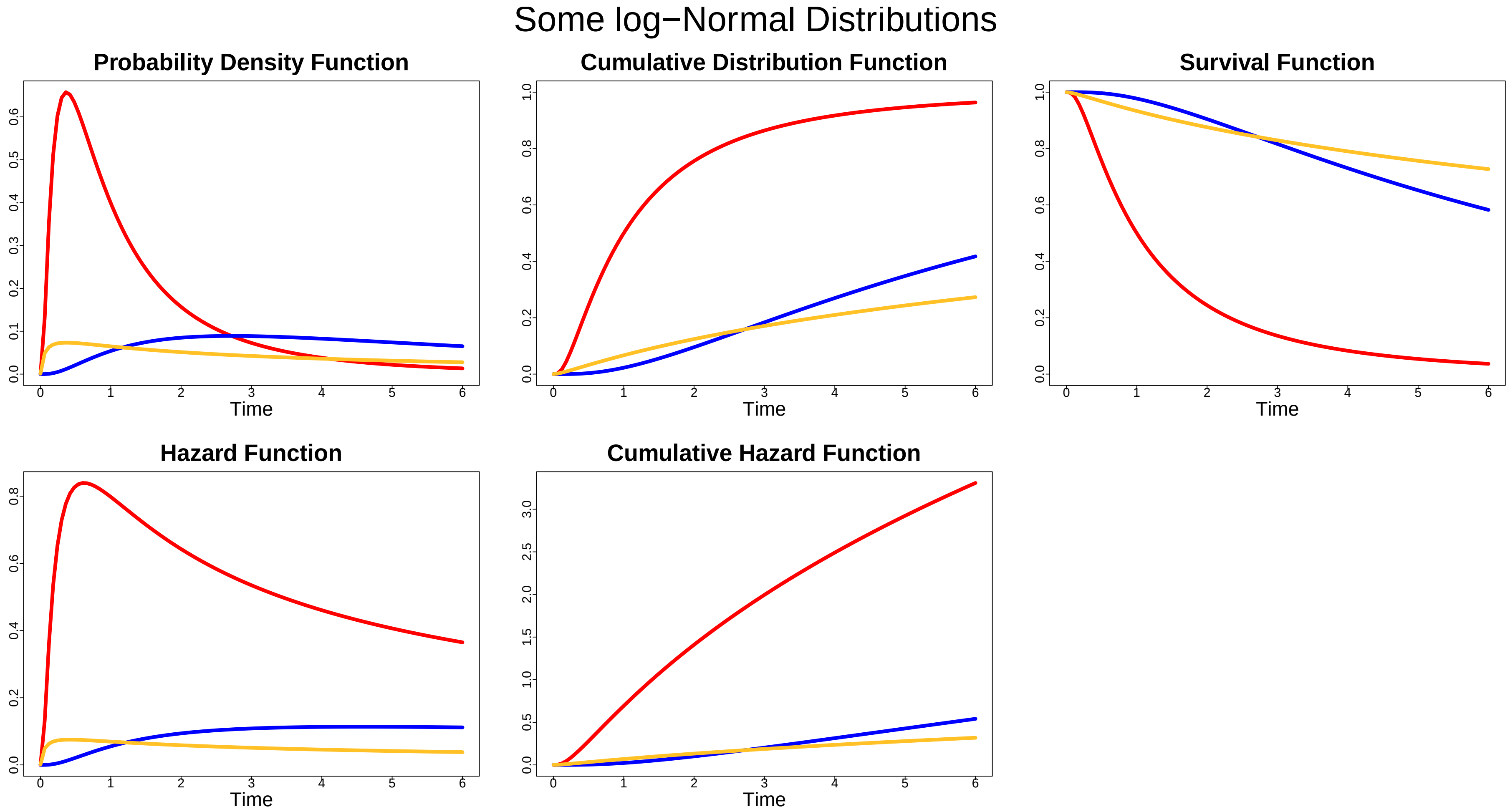
- In the literature there have been proposed many distributions for time-to-event random variables

- Some of the most popular are
  - ▷ Weibull (it has as special case the Exponential)
  - ▷ Gamma (it has as special case the Exponential)
  - ▷ log-Normal
  - ▷ log-Student's-t
  - ▷ log-Logistic

2.7 Survival Distributions (cont'd)



2.7 Survival Distributions (cont'd)



2.8 Review of Key Points

- The basic functions to describe the distribution of time-to-event data
  - ▷ survival function
  - ▷ cumulative distribution and probability density function
  - ▷ hazard and cumulative hazard function

- All these functions are related

2.8 Review of Key Points (cont'd)

- Statistical measures
  - median survival
  - mean survival
  - expected future lifetime

Part III

Estimation & Statistical Inference

3.1 Notation

- We have a sample of failure times ⇒ What is the available information?

- Notation (i denotes the patient)
  - ▷ T_i^* ‘true’ time-to-event
  - ▷ because of censoring we do not always observe T_i^*
  - ▷ C_i the censoring time (e.g., the end of the study or a random censoring time)

- Available data for each patient
  - ▷ observed event time: $T_i = \min(T_i^*, C_i)$
  - ▷ event indicator: $\delta_i = 1$ if event; $\delta_i = 0$ if censored

3.1 Notation (cont'd)

Patient	T_i^*	C_i	T_i	δ_i
1	3.5	—	3.5	1
2	3.4	2.2	2.2	0
3	5.7	5	5	0
⋮	⋮	⋮	⋮	⋮

The end of the study is at 5 years

3.1 Notation (cont'd)

- Based on the available information $\{T_i, \delta_i\}$ we wish to estimate various quantities of interest, e.g.,
  - ▷ the Survival function
  - ▷ mean survival time
  - ▷ median survival time
  - ▷ specific quantiles
    - * by which follow-up time 25% of the patients is still alive
  - ▷ ...

3.2 The Kaplan-Meier Estimator

- Aim: estimate the Survival Function $S(t)$ based on a sample of failure times $T_1, \ldots, T_n$
  - ▷ <u>Remember:</u> $S(t)$ is the probability of being alive at time t (see Section 2.1)

- If there was no censoring, we could simply

$$\hat{S}(t) = \frac{\text{number of patients alive at time } t}{n} = \frac{1}{n} \sum_{i=1}^{n} I(T_i > t)$$

where $I(T_i > t)$ equals 1 if $T_i > t$, and 0 otherwise

- However, we <u>do</u> have censored observations

3.2 The Kaplan-Meier Estimator (cont'd)

- To take into account censoring in the estimation of $S(t)$ we will use *the law of total probability*

- For instance, the probability of surviving 2 years can be computed as:

$$\begin{aligned} S(2) &= \textsf{Pr}(T_i > 2) \\ &= \textsf{Pr}(T_i > 1) \times \textsf{Pr}(T_i > 2 \mid T_i > 1) \end{aligned}$$

- In words, the probability of surviving year 2 is the product of
  - ▷ the probability of surviving year 1 and
  - ▷ the conditional probability of surviving up to year 2 given still being alive at year 1

3.2 The Kaplan-Meier Estimator (cont'd)

- So $S(2)$ can be estimated by

$$\hat{S}(2) \;=\; \frac{\#\text{ patients alive at year 1}}{\#\text{ patients at risk up to year 1}} \times \frac{\#\text{ patients alive at year 2}}{\#\text{ patients at risk up to year 2}}$$

- If we apply this idea repeatedly, we can obtain survival probabilities for every time point t

3.2 The Kaplan-Meier Estimator (cont'd)

- Let t_1, t_2, ..., t_k denote the unique event times in the sample at hand

- We account for censoring by suitably adjusting the risk set ⇒ the *Kaplan-Meier Estimator*

$$\hat{S}_{KM}(t) = \prod_{i: t_i \leq t} \frac{r_i - d_i}{r_i}$$

  where d_i is the number of events at time t_i, and r_i the number of patients still at risk at time t_i

  ▷ still at risk means alive and not censored

3.2 The Kaplan-Meier Estimator (cont'd)

- A small example

$$1 \;\; 5^+ \;\; 6 \;\; 6 \;\; 8 \;\; 8^+ \;\; 9 \;\; 11^+$$

$+$ denotes a censored time

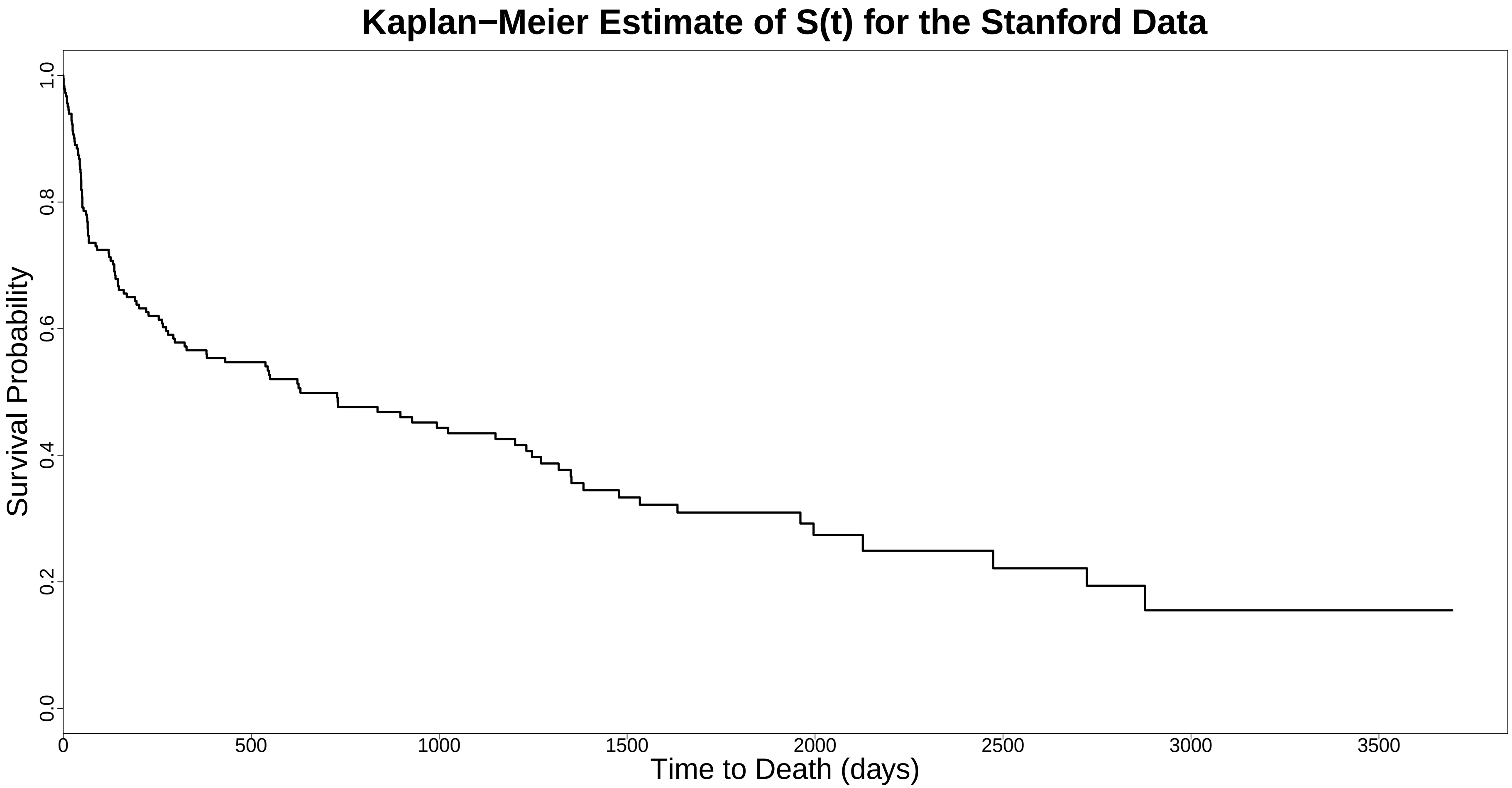
i	t_i	r_i	d_i	$(r_i - d_i)/r_i$
1	1	8	1	7/8
2	6	6	2	4/6
3	8	4	1	3/4
4	9	2	1	1/2

3.2 The Kaplan-Meier Estimator (cont'd)

$$\begin{aligned}\hat{S}_{KM}(t) &= 1, \quad 0 \leq t < 1 \\ &= 7/8 = 0.875, \quad 1 \leq t < 6 \\ &= (7/8)(4/6) = 0.583, \quad 6 \leq t < 8 \\ &= (7/8)(4/6)(3/4) = 0.438, \quad 8 \leq t < 9 \\ &= (7/8)(4/6)(3/4)(1/2) = 0.219, \quad 9 \leq t < 11\end{aligned}$$

Note: the estimate of $S(t)$ is undefined for $t > 11$ since not all subjects have died by $t = 11$

3.2 The Kaplan-Meier Estimator (cont'd)



3.2 The Kaplan-Meier Estimator (cont'd)

- For the following sample of event times (+ denotes a censored time)

$$0.5 \;\; 1^+ \;\; 2 \;\; 2^+ \;\; 5^+ \;\; 8^+ \;\; 9 \;\; 13^+ \;\; 15$$

compute the Kaplan-Meier estimator

Time Interval	r_i	d_i	$\frac{r_i - d_i}{r_i}$	$\hat{S}_{KM}(t)$
$0 \leq t < 0.5$	9	0	9/9	1

3.2 The Kaplan-Meier Estimator (cont'd)

- The variance of $\hat{S}_{KM}(t)$ can be estimated using Greenwood's formula

- Using the formula and asymptotic normality of $\hat{S}_{KM}(t)$, we can derive a 95% confidence interval

- Problem: this can exceed 1 or fall below 0!

- A better asymmetric 95% confidence interval for $\hat{S}_{KM}(t)$ that respects the boundaries is derived from a symmetric 95% confidence interval for either

$$\hat{H}_{KM}(t) = -\log \hat{S}_{KM}(t) \quad \text{or} \quad \log \hat{H}_{KM}(t) = \log\{-\log \hat{S}_{KM}(t)\}$$

3.2 The Kaplan-Meier Estimator (cont'd)

- An estimate for the variance of $\log \hat{H}_{KM}(t)$ is obtained by

$$\hat{\text{var}}\{\log \hat{H}_{KM}(t)\} = \frac{\sum_{i:\, t_i \leq t} d_i / \{r_i(r_i - d_i)\}}{\left[\sum_{i:\, t_i \leq t} \log\{(r_i - d_i)/r_i\}\right]^2}$$

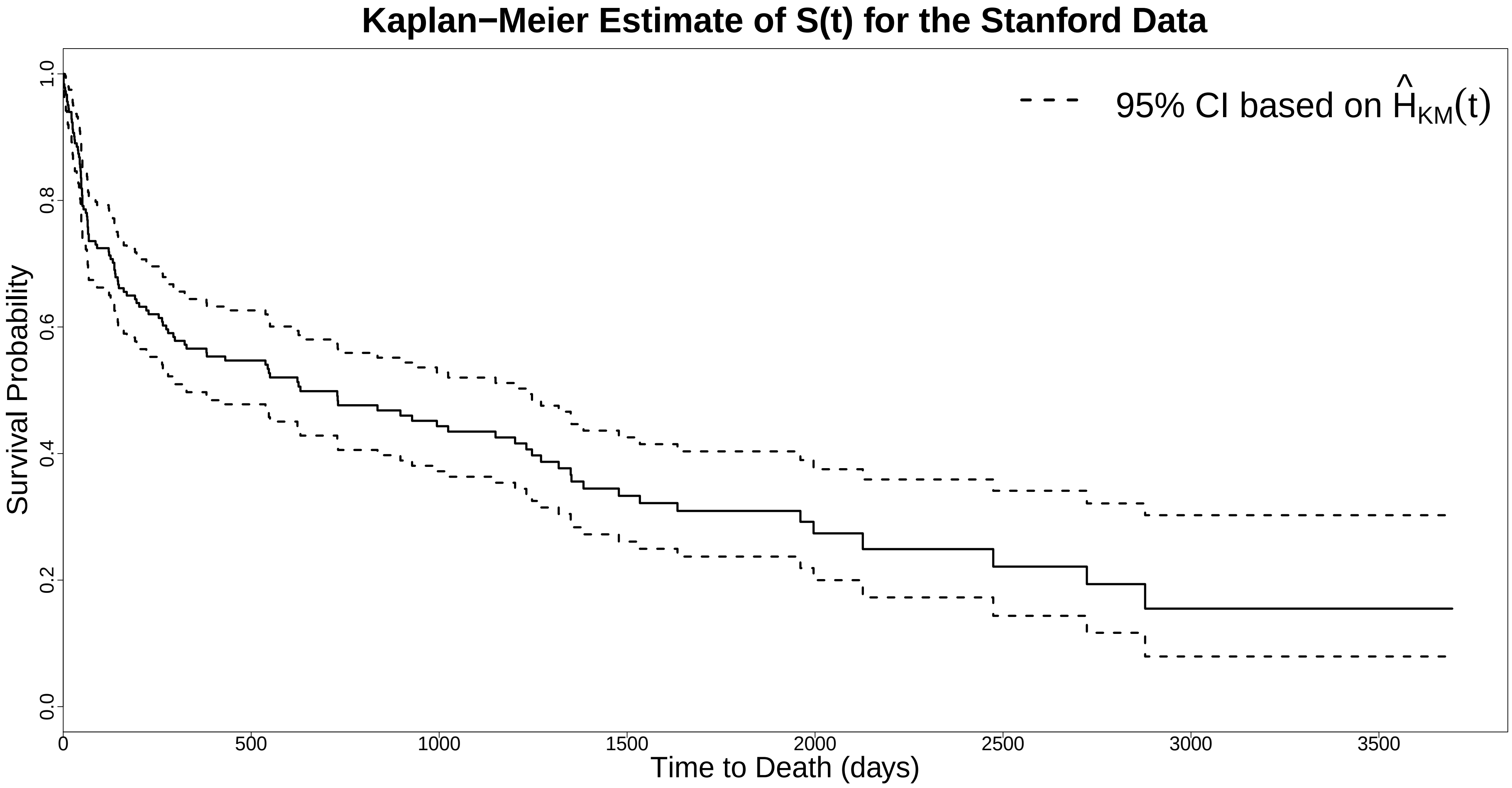
- Based on the estimated variance for $\log \hat{H}_{KM}(t)$ we calculate the confidence interval

$$[a, b] = \log \hat{H}_{KM}(t) \mp 1.96 \times \sqrt{\hat{\text{var}}\big\{\log \hat{H}_{KM}(t)\big\}}$$

the confidence intervals for $\hat{S}_{KM}(t)$ is then obtained as

$$\big[\exp\{-\exp(b)\},\ \exp\{-\exp(a)\}\big]$$

3.2 The Kaplan-Meier Estimator (cont'd)



3.2 The Kaplan-Meier Estimator (cont'd)

R> Survival analysis in R

- ▷ GUI-based ⇒ in practicals
- ▷ command-based ⇒ in slides & in practicals

3.2 The Kaplan-Meier Estimator (cont'd)

R> A key function in R that is used to specify the available event time information in a sample at hand is `Surv()`

R> For right censored failure times (i.e., what we will see in this course) we need to provide the observed event times `time`, and the event indicator `status`, which equals 1 for true failure times and 0 for right censored times

```
Surv(time, status)
```

3.2 The Kaplan-Meier Estimator (cont'd)

R> The function that is used to produce the Kaplan-Meier estimate of a survival function is survfit() – for the Stanford data we have

```
KM <- survfit(Surv(time, status) ~ 1, data = stanford2)
plot(KM)
```

3.2 The Kaplan-Meier Estimator (cont'd)

- The estimated survival function can be used to extract estimates of specific percentiles of interest, such as the median survival time (see Section 2.6)

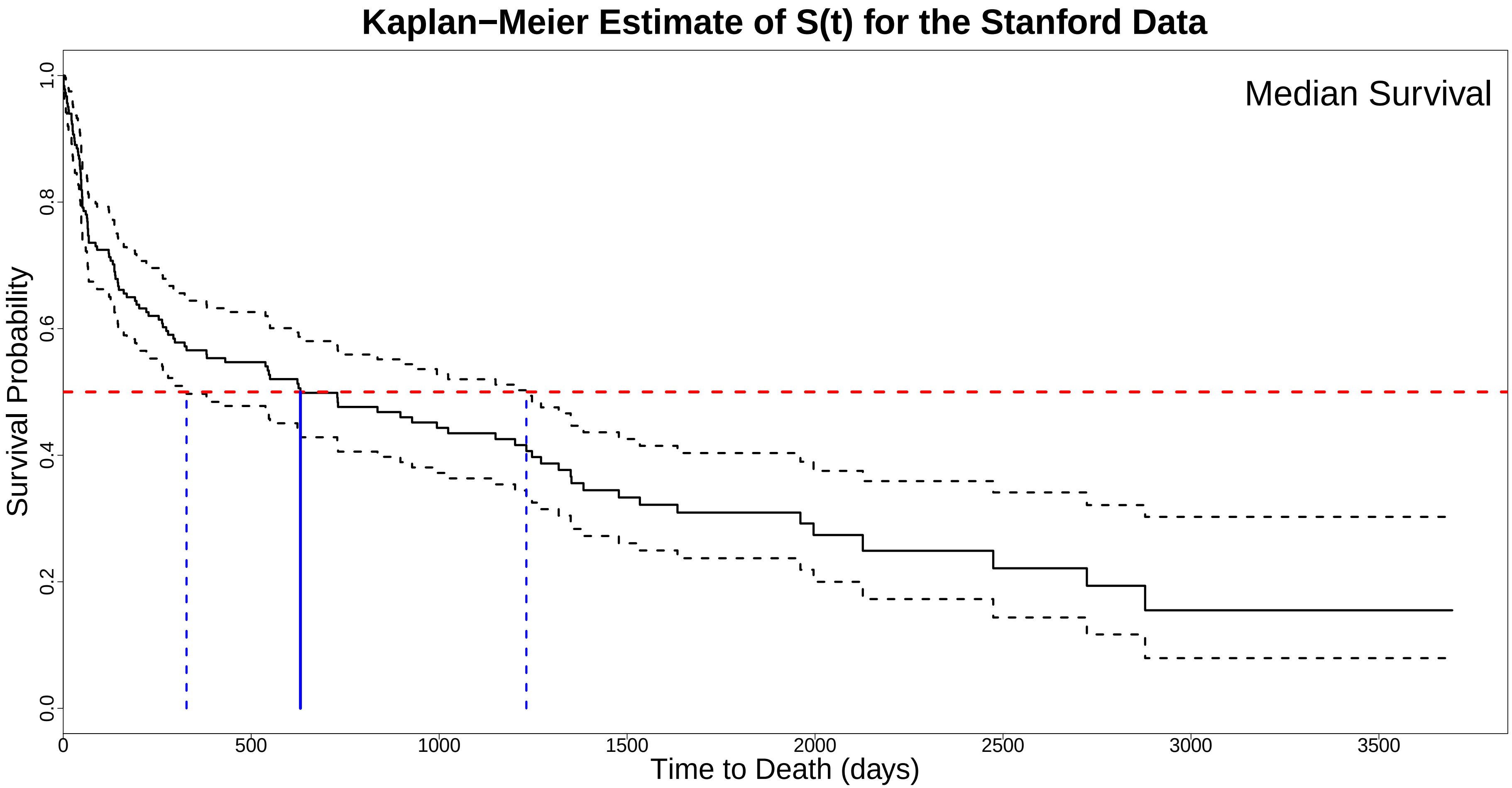
- The following procedure is followed:
  - ▷ draw a horizontal line at the specific probability level of interest (e.g., 0.5 for the median survival time, 0.25 for the 1st quantile, etc.)
  - ▷ the time point at which this horizontal line intersects with the survival curve is the estimated survival time of interest
  - ▷ a 95% confidence interval for this survival time is obtained by extracting the times at which the horizontal line intersects with the 95% confidence interval of the survival function

3.2 The Kaplan-Meier Estimator (cont'd)

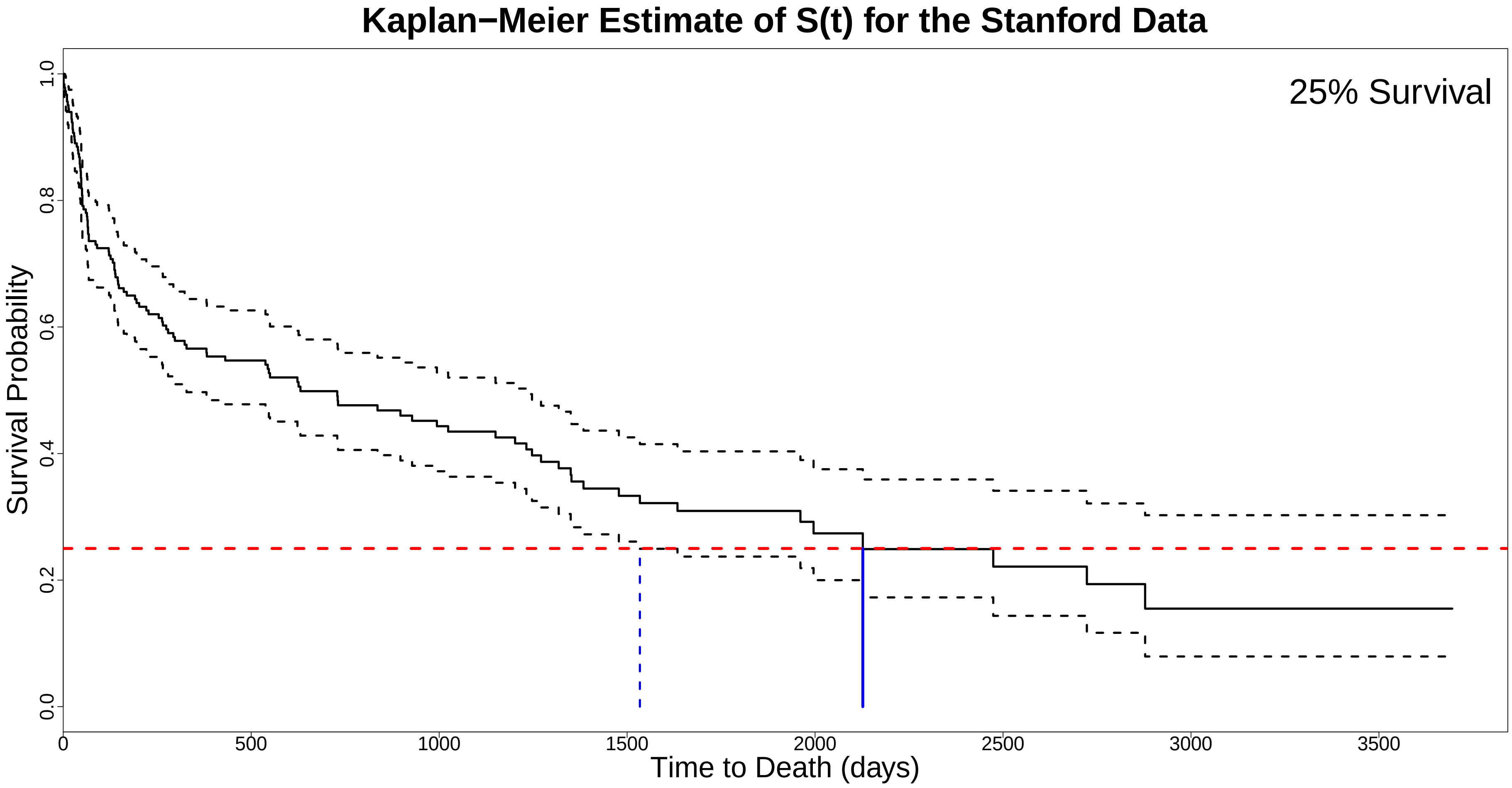
- If the horizontal line does not cross either the survival curve or its confidence interval, then the desired percentile and/or its confidence limits cannot be specified from the nonparametric estimate of $S(t)$

- Example: for the Stanford data and based on the Kaplan-Meier estimate of the survival function we compute
  - ▷ the median survival time (i.e., the time at which 50% of the patients is still alive)
  - ▷ 1st quantile survival time (i.e., the time at which 25% of the patients is still alive)

3.2 The Kaplan-Meier Estimator (cont'd)



3.2 The Kaplan-Meier Estimator (cont'd)



3.2 The Kaplan-Meier Estimator (cont'd)

- We obtain

% Alive	Time	95% Lower	95% Upper
	days	Limit (days)	Limit (days)
50	631	328	1232
25	2127	1534	NA

3.2 The Kaplan-Meier Estimator (cont'd)

R> The following function can be used to easily extract the desired percentiles from the output of `survfit()`

```
quantile.survfit <- function (fit, probs = c(0.25, 0.5, 0.75)) {
    # 'fit' a survfit object
    # 'probs' percentiles of interest
    f <- function (x) ifelse(x == 0, NA, x)
    times <- rev(fit$time)
    ind <- f(findInterval(probs, rev(fit$surv)))
    indL <- f(findInterval(probs, rev(fit$lower)))
    indU <- f(findInterval(probs, rev(fit$upper)))
    cbind(Prob = probs, Time = times[ind],
        "0.95LCL" = times[indL], "0.95UCL" = times[indU])
}
```

3.2 The Kaplan-Meier Estimator (cont'd)

R> Example of `quantile.survfit()`

```
KM <- survfit(Surv(time, status) ~ 1, data = stanford2)

quantile(KM, c(0.25, 0.50))
```

3.3 The Breslow Estimator

- Using a similar approach to the Kaplan-Meier, we can also estimate the cumulative hazard function $H(t)$

- Remember: $H(t)$ denotes the expected number of events up to and including time point t (see Section 2.4)

- So a natural estimator of $H(t)$ is

$$\hat{H}_{NA}(t) = \sum_{i:\, t_i \leq t} \frac{d_i}{r_i}$$

which is called the Nelson-Aalen estimator

3.3 The Breslow Estimator

- Remember: the cumulative hazard function is related to the survival function (see Section 2.5)

- Therefore, an estimator for the survival function based on the Nelson-Aalen estimator is

$$\hat{S}_B(t) = \exp\{-\hat{H}_{NA}(t)\} = \prod_{i:\, t_i \leq t} \exp(-d_i/r_i)$$

  which has been suggested by Breslow and therefore it is known as the *Breslow Estimator*

3.3 The Breslow Estimator (cont'd)

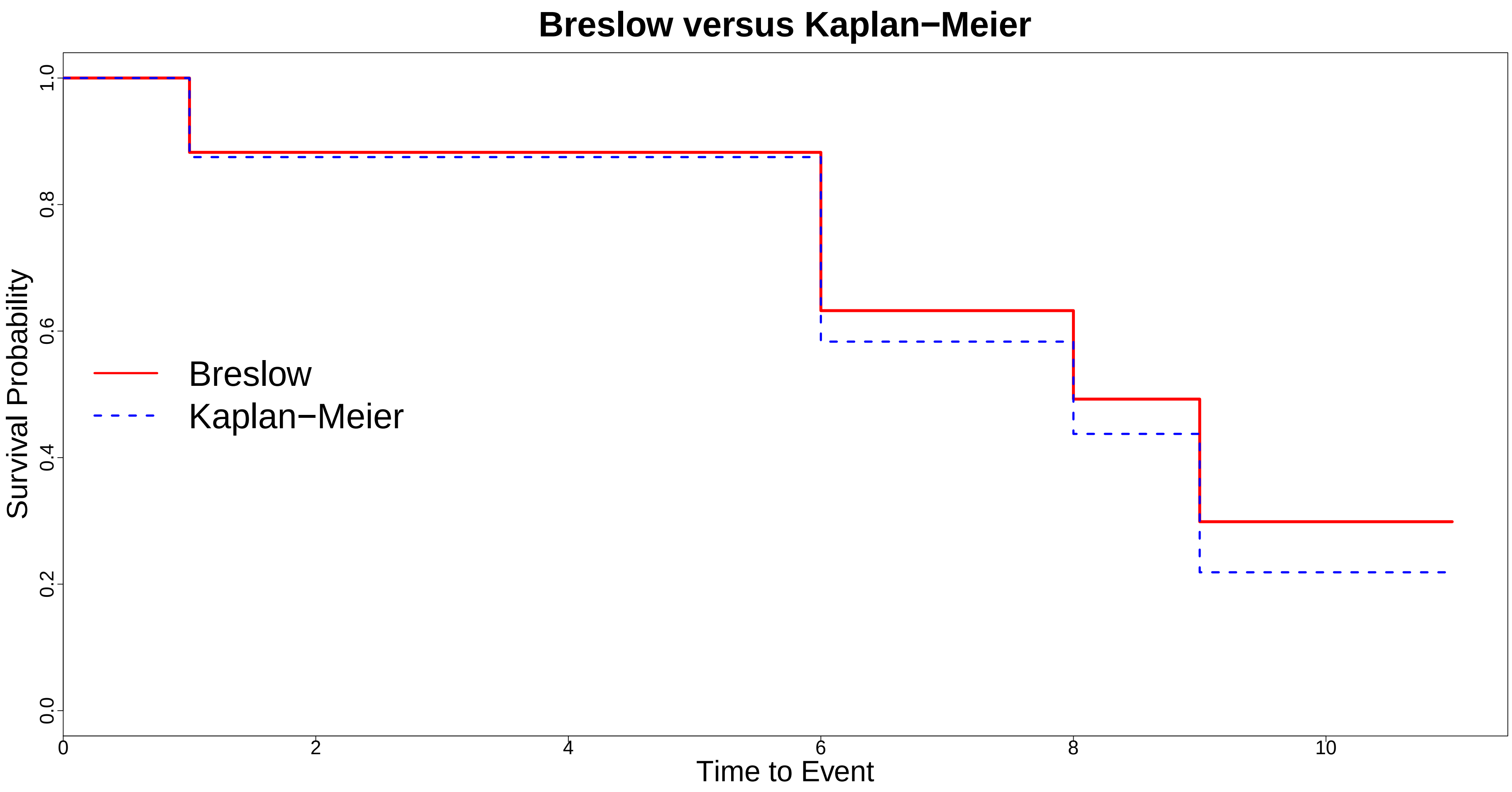
- Using the same example

$$1 \;\; 5^+ \;\; 6 \;\; 6 \;\; 8 \;\; 8^+ \;\; 9 \;\; 11^+$$

$+$ denotes a censored time

i	t_i	r_i	d_i	$\frac{r_i - d_i}{r_i}$	d_i/r_i	$\hat{S}_B(t)$
1	1	8	1	7/8	1/8	$\exp(-1/8) = 0.882$
2	6	6	2	4/6	2/6	$\exp\{-(1/8 + 2/6)\} = 0.632$
3	8	4	1	3/4	1/4	$\exp\{-(1/8 + 2/6 + 1/4)\} = 0.492$
4	9	2	1	1/2	1/2	$\exp\{-(1/8 + 2/6 + 1/4 + 1/2)\} = 0.297$

3.3 The Breslow Estimator (cont'd)



3.3 The Breslow Estimator (cont'd)

- The difference between the Kaplan-Meier and the Breslow estimators is always very small
  - ▷ as $n \to \infty$ the two estimates are equivalent

- The Breslow estimator is biased upwards, especially close to zero, but it has lower variance
  - ▷ if the largest observed time T in the data set is an event, then $\hat{S}_{KM}(t) = 0$ whereas $\hat{S}_B(t)$ is positive

3.3 The Breslow Estimator (cont'd)

R> The Breslow estimator of the survival function is again computed using function `survfit()`, however now the `type` argument needs to be specified – for the Stanford data we have

```
Brs <- survfit(Surv(time, status) ~ 1, data = stanford2,
    type = "fleming-harrington")
plot(Brs)
```

3.3 The Breslow Estimator (cont'd)

- The variance for the Breslow estimator is based on a similar approach as for the Kaplan-Meier estimator

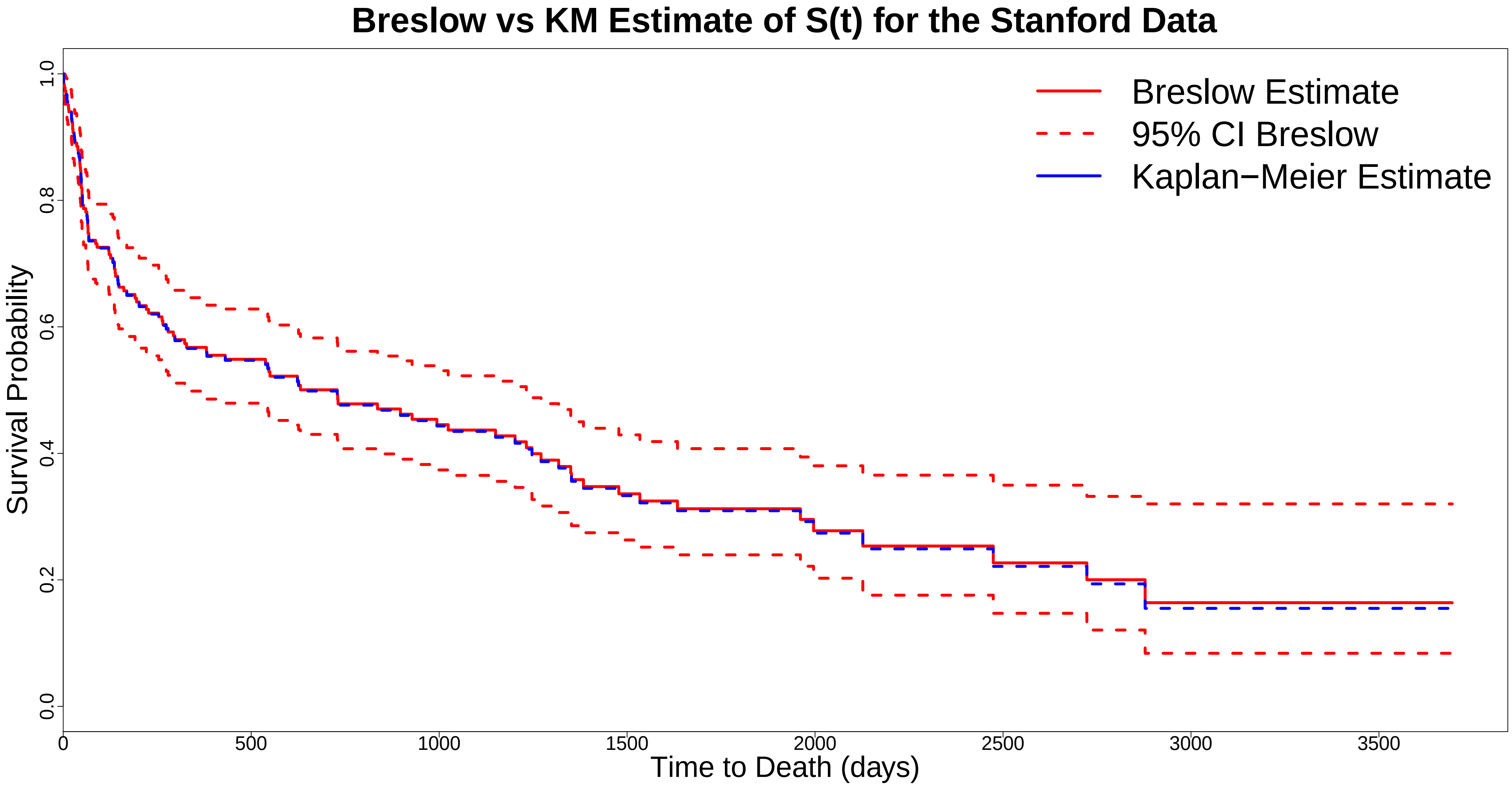
- The same also holds for the calculation of the 95% confidence intervals

  ▷ that is, confidence intervals are computed for $\log H(t)$ and then back-transformed using the relation

$$\exp\Big[- \exp\big\{\text{95\% CI for } \log H(t)\big\}\Big]$$

  to obtain confidence intervals for $S(t)$

3.3 The Breslow Estimator (cont'd)



3.3 The Breslow Estimator (cont'd)

- As before, we observe that the two estimators are indistinguishable

- The Kaplan-Meier is more popular

- However, a lot of theoretical developments in statistics have been based on the Breslow estimator

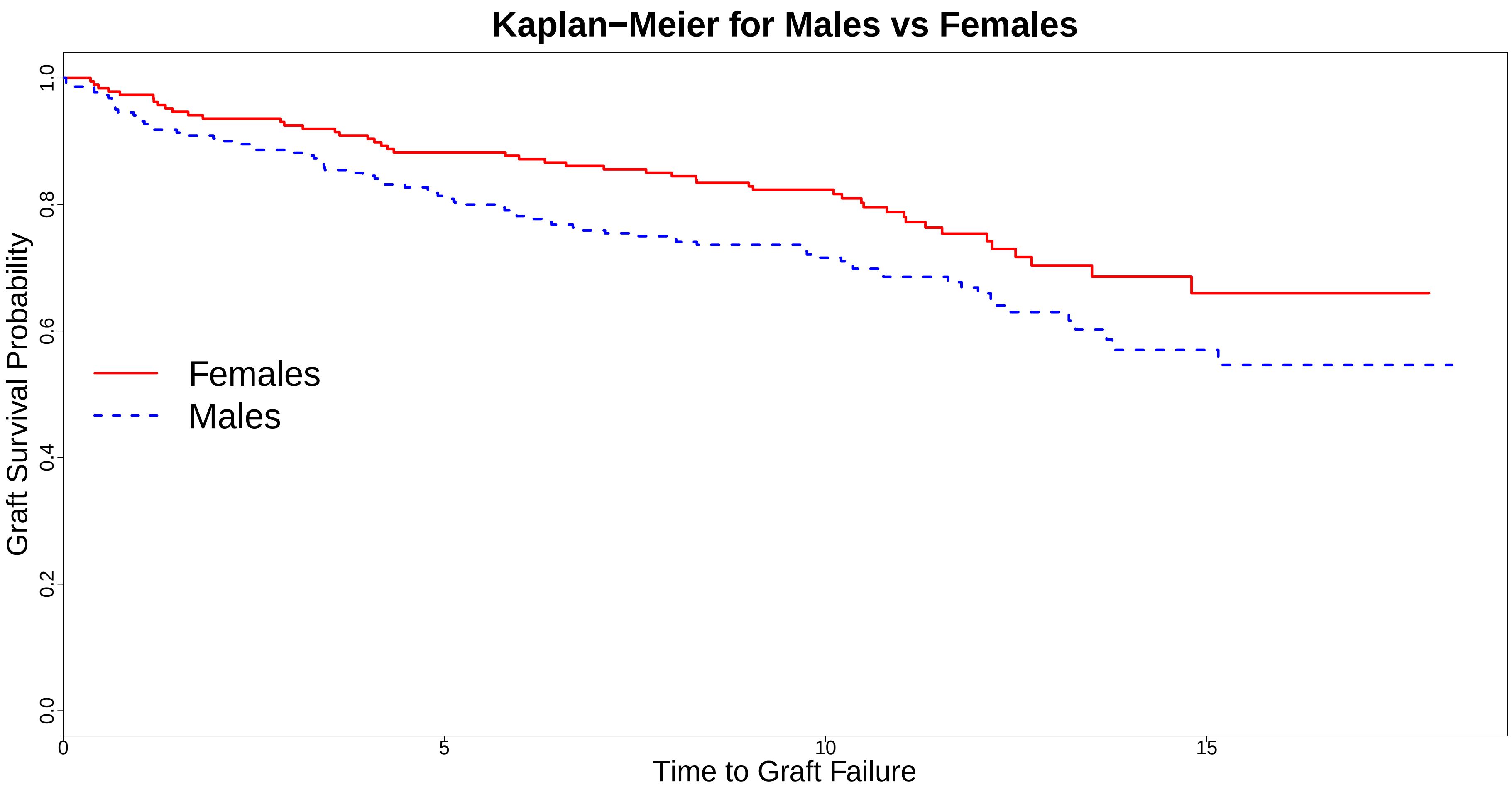
3.4 Comparing Survival Functions

- We have 2 groups of patients
  - ▷ treatment vs placebo
  - ▷ females vs males
  - ▷ history of diabetes, Yes vs No
  - ▷ . . .

- Question of Interest: how can we compare these groups with respect to survival

- We can estimate separate survival curves for the 2 groups,

3.4 Comparing Survival Functions (cont'd)



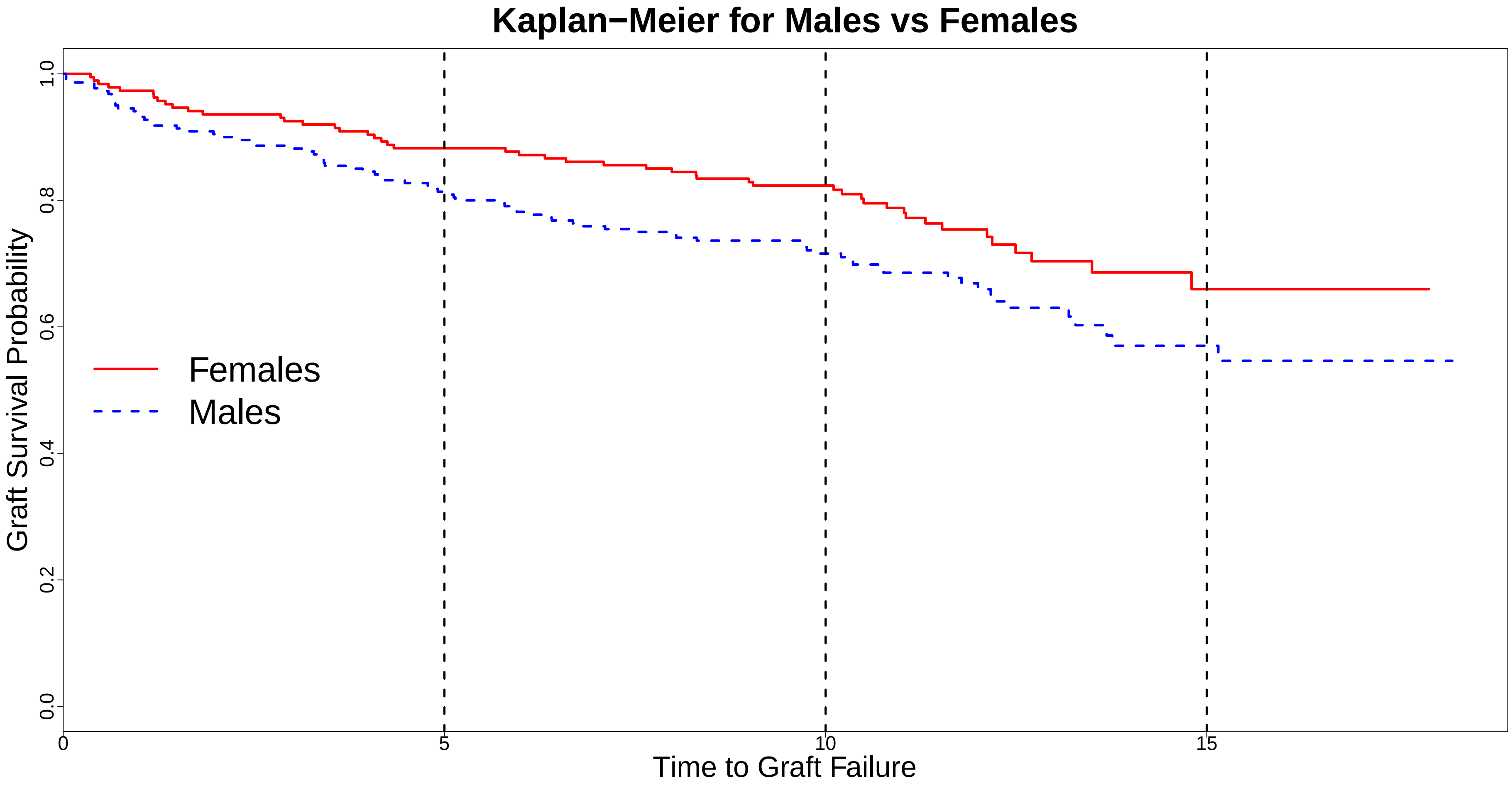
3.4 Comparing Survival Functions (cont'd)

- but how to compare these survival curves?

- We could compare at a specific time point

- At which time point?
  - ▷ start of follow-up
  - ▷ end of follow-up
  - ▷ intermediate points
  - ▷ . . .

3.4 Comparing Survival Functions (cont'd)



3.4 Comparing Survival Functions (cont'd)

- Not very informative because the difference between the survival curves can be greater at some time points than others

- Alternatively, it seems more appropriate to compare the 2 survival curves over the whole follow-up period

- Formally, we are interested in testing the following set of hypotheses

$$\boxed{\begin{array}{lcl} H_0 & : & \text{the distribution of survival times is the same for the 2 groups} \\ H_a & : & \text{it is not the same} \end{array}}$$

3.4 Comparing Survival Functions (cont'd)

- The most famous statistical test to test this hypothesis is the *Mantel-Haenszel Test (aka Log-Rank Test)*

- This is a nonparametric test
  - ▷ no distributional assumption is made for the survival times of the 2 groups

- The philosophy behind it is to construct 2×2 contingency tables for each unique event time, and compare observed with expected numbers of events.

3.4 Comparing Survival Functions (cont'd)

- In particular, let $t_{(i)}$ denote the ith ordered event time in the 2 groups combined

	Group 1	Group 2	Total
Event	d_{1i}	d_{2i}	d_i
No Event	$r_{1i} - d_{1i}$	$r_{2i} - d_{2i}$	$r_i - d_i$
At risk	r_{1i}	r_{2i}	r_i

▷ d_{ji} is the number of subjects experiencing the event at time $t_{(i)}$ in group j

▷ r_{ji} is the number of subjects at risk at time $t_{(i)}$ in group j

▷ d_i is the total number of subjects experiencing the event

▷ r_i is the total number of subjects at risk

3.4 Comparing Survival Functions (cont'd)

- In the case of no ties, one of d_{1i} and d_{2i} will be 1 and the other 0

- Under the null hypothesis (i.e., the population survival curves are the same in the 2 groups), we can estimate the expected number of subjects experiencing the event at time $t_{(i)}$

$$\hat{E}_{ji} = \frac{d_i r_{ji}}{r_i}$$

- The variance of $\hat{E}_{ji}$ can be estimated by

$$\text{var}(\hat{E}_{ji}) = \frac{r_{1i} r_{2i} d_i (r_i - d_i)}{r_i^2 (r_i - 1)}$$

3.4 Comparing Survival Functions (cont'd)

- We construct this 2×2 contingency table for every observed event time $t_{(1)}, \ldots, t_{(m)}$

- Then, the log-rank test has the form of a standard X^2 statistic, i.e.,

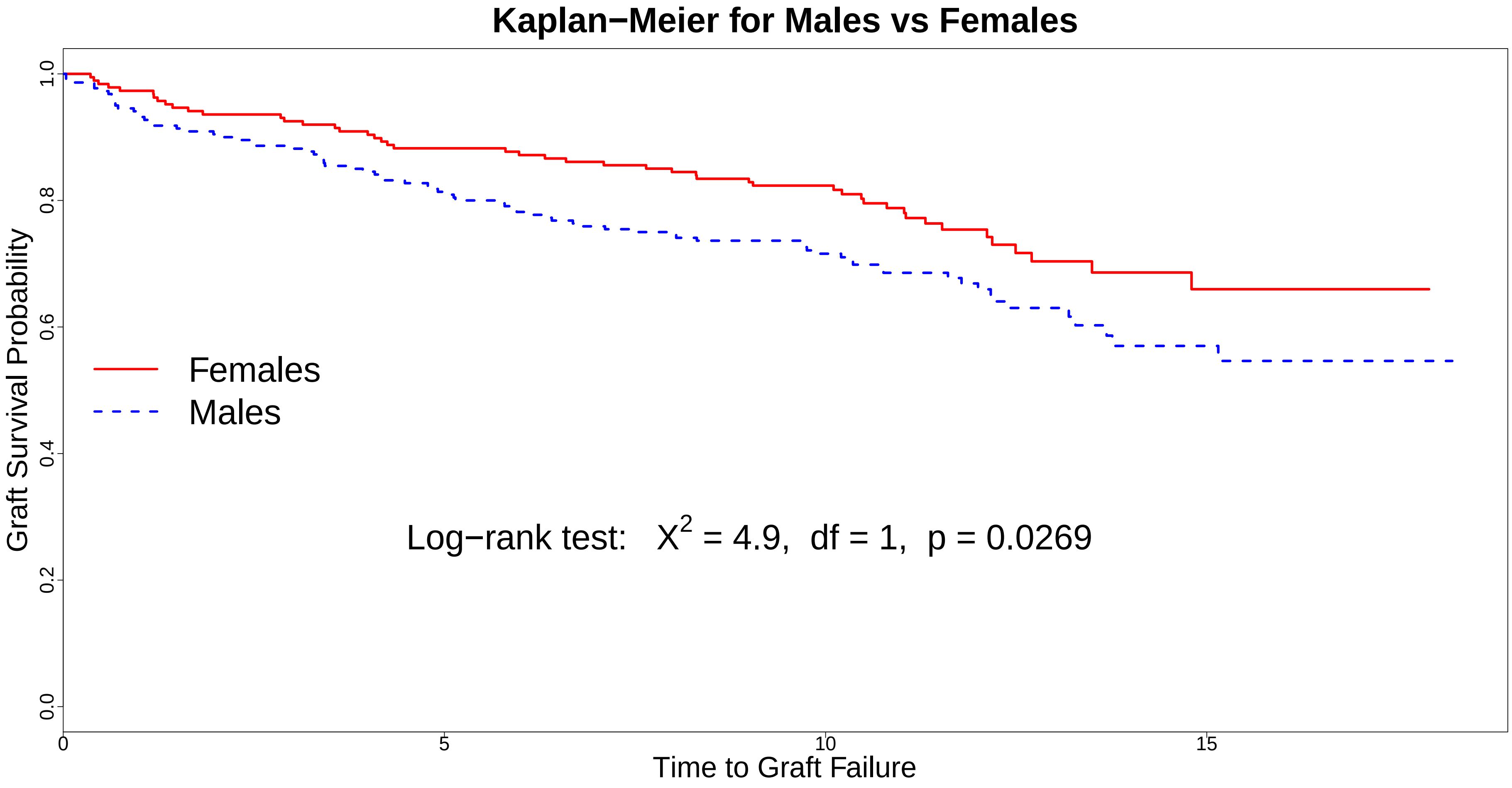
$$X^2 = \frac{\left(\sum_{i=1}^{m} d_{1i} - \hat{E}_{1i} \right)^2}{\sum_{i=1}^{m} \mathsf{var}(\hat{E}_{1i})}$$

- Under the null hypothesis, this statistic is asymptotically distributed as χ^2_1

3.4 Comparing Survival Functions (cont'd)

- $\boxed{\text{Example:}}$ for the Renal Graft failure data we are interested in testing whether the survival curve of males is different from the one of females

3.4 Comparing Survival Functions (cont'd)



3.4 Comparing Survival Functions (cont'd)

R> The Log-Rank test is computed using function `survdiff()` – for the Renal data we used

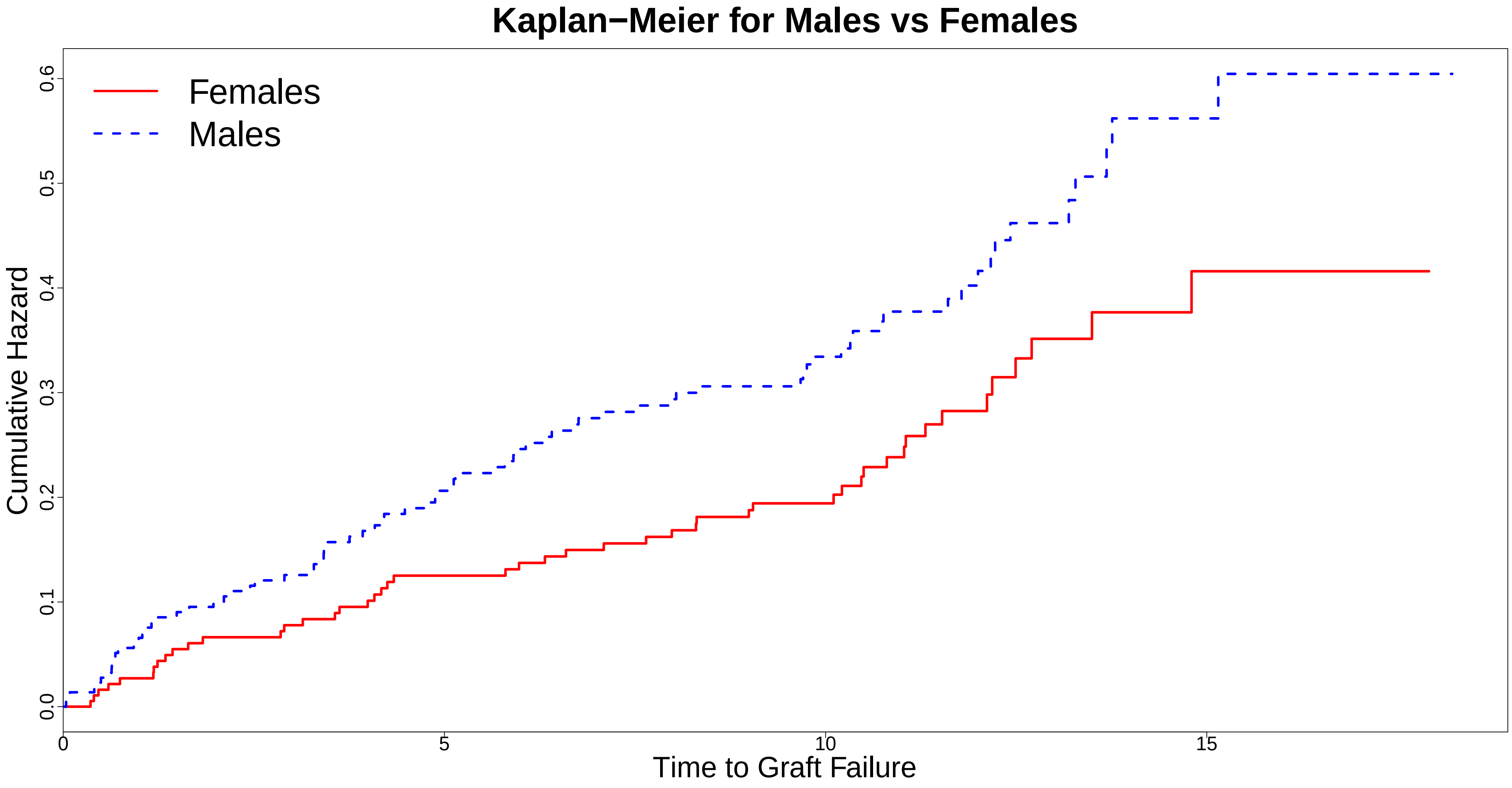
```
survdiff(Surv(Time, failure) ~ gender, data = rgf)
```

3.4 Comparing Survival Functions (cont'd)

- The performance of the Log-Rank Test is compromised when
  - ▷ censoring is informative
  - ▷ the hazard for an event in the one group is not proportional to the hazard for an event in the other group (*proportional hazards assumption*)

- Feature: it places the same weight on all follow-up times

3.4 Comparing Survival Functions (cont'd)



3.4 Comparing Survival Functions (cont'd)

- An alternative test to compare the survival curves, is the modified by *Peto and Peto Gehan-Wilcoxon test*

- Compared to the log-rank test, this test
  - ▷ is more powerful when the hazard functions of the 2 groups are not proportional
  - ▷ puts more emphasis on earlier event times

3.4 Comparing Survival Functions (cont'd)

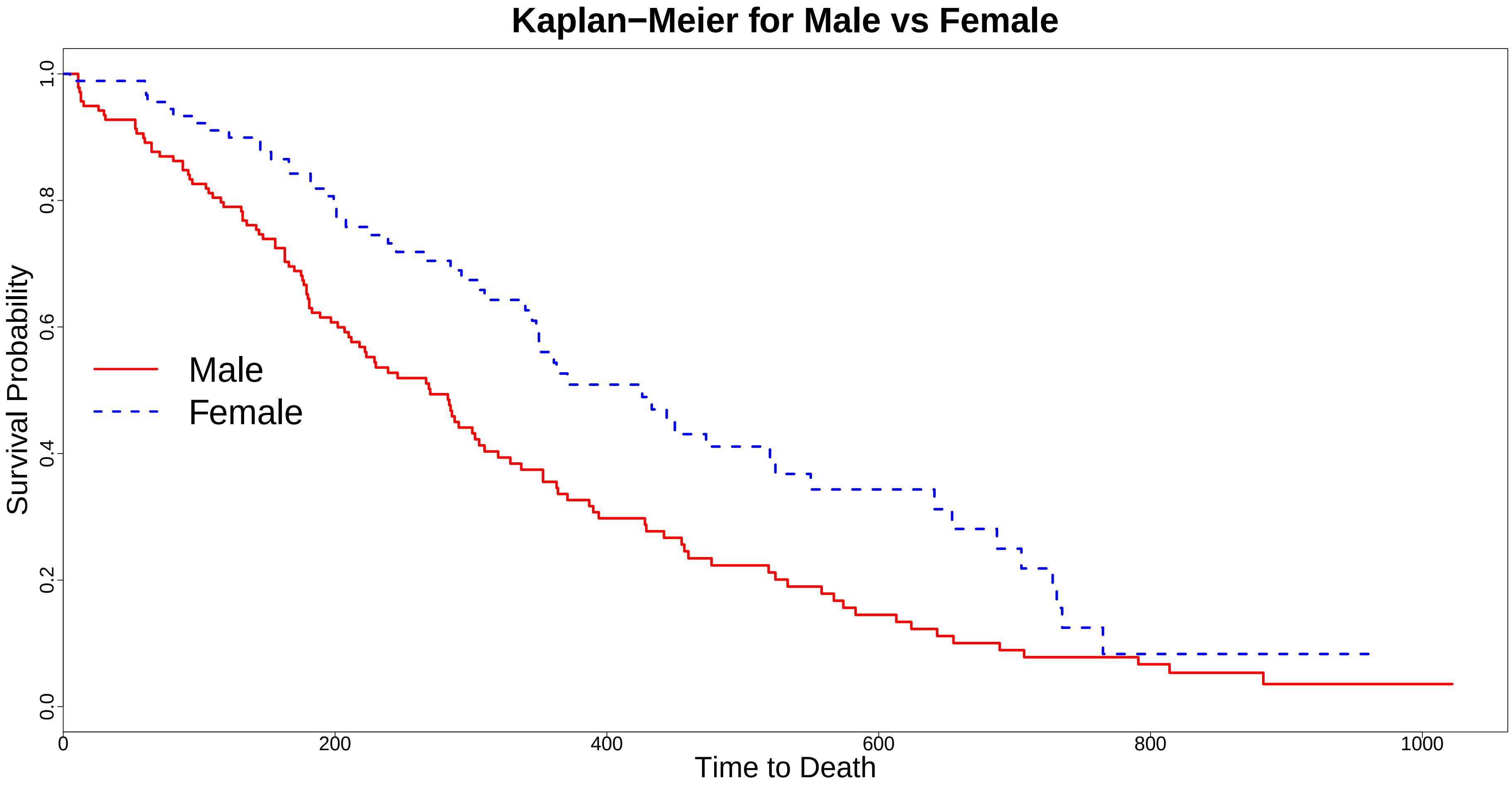
- The Peto-Wilcoxon statistic has a similar form as the X^2 statistic of the log-rank test

$$W = \frac{\left\{ \sum\limits_{i=1}^{m} r_i (d_{1i} - \hat{E}_{1i}) \right\}^2}{\sum\limits_{i=1}^{m} r_i^2 \text{var}(\hat{E}_{1i})}$$

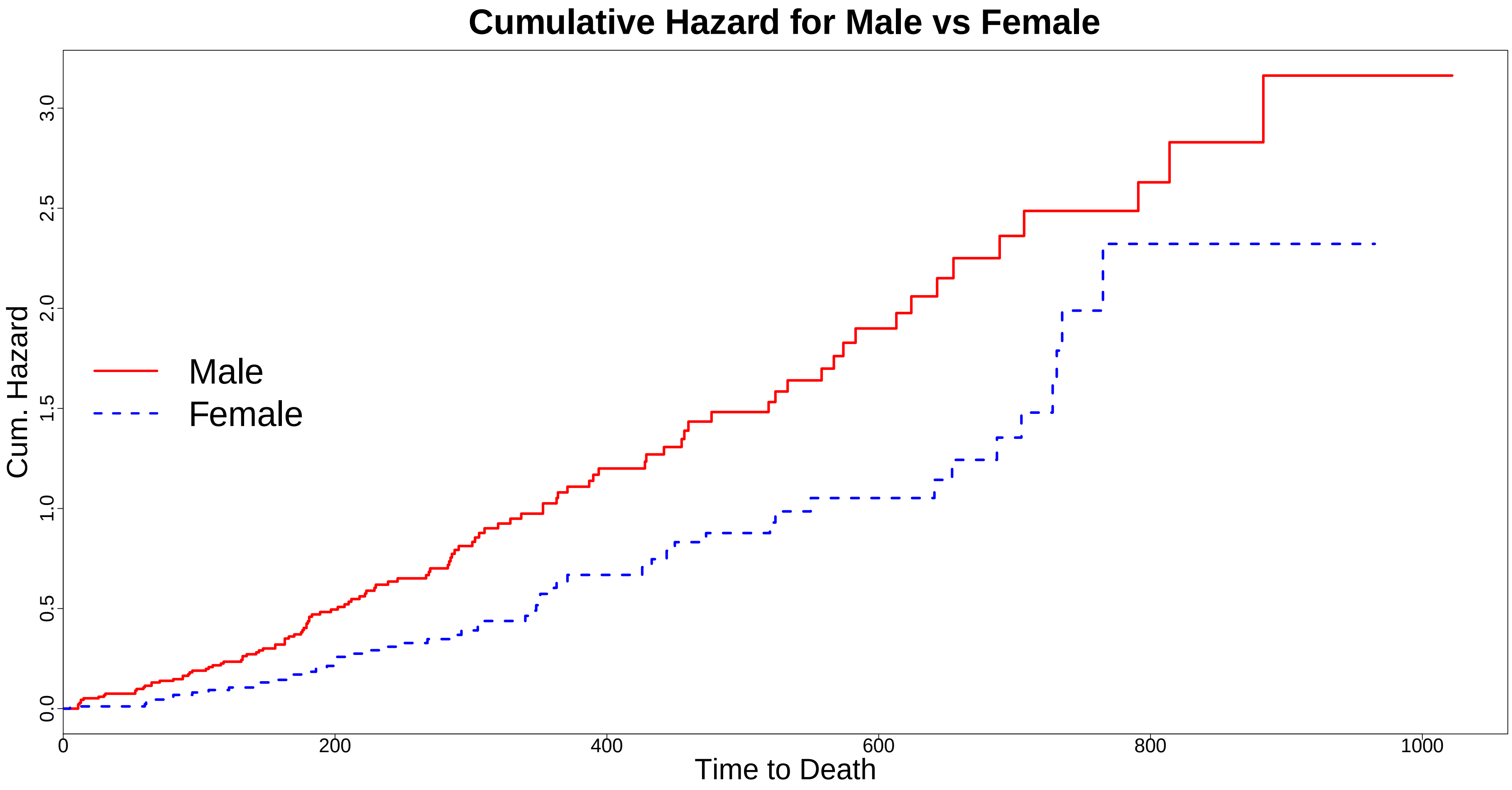
  which weighs the differences between observed and expected deaths in group 1 by the factor r_i – asymptotically (i.e., in large samples) W has a χ^2_1 distribution

- $\boxed{\text{Example:}}$ to illustrate the difference between the two tests, we compare with both the survival curves of males versus females for the Lung data set

3.4 Comparing Survival Functions (cont'd)



3.4 Comparing Survival Functions (cont'd)



3.4 Comparing Survival Functions (cont'd)

- Log-rank test
  - ▷ $X^2 = 10.3$, $df = 1$, $p = 0.00131$

- Peto & Peto modification of the Gehan-Wilcoxon test
  - ▷ $X^2 = 12.7$, $df = 1$, $p = 0.00036$

- In both cases the result is significant, but the p-value from the Log-rank test is almost 4 times the p-value of the Peto & Peto modified Gehan-Wilcoxon test
  - ▷ the survival curves are much closer at the end of the follow-up than in the beginning

3.4 Comparing Survival Functions (cont'd)

R> The Peto & Peto modified Gehan-Wilcoxon test is again computed using function survdiff(); however, we need to set argument rho to 1 – for the Lung data we used

```
survdiff(Surv(time, status == 2) ~ sex, data = lung, rho = 1)
```

3.4 Comparing Survival Functions (cont'd)

- Which of the 2 tests should be preferred?
  - ▷ if the survival curves cross, then both tests are not optimal
  - ▷ check if the proportional hazards assumption is (seriously) violated using the cumulative Hazards plot
  - ▷ the log-rank test will be more powerful if the proportional hazards assumption is valid
  - ▷ otherwise use the Peto-Wilcoxon test

- However, it would not be fair to decide which test to use based on where the survival curves are further apart, i.e.,
  - ▷ differences in earlier vs later survival times

3.5 Review of Key Points

- We need special notation to take into account censored data
  - ▷ T_i observed event time
  - ▷ δ_i equals 1 if T_i is a true event, and 0 if it is a censoring time

- Estimators of the survival function
  - ▷ Kaplan-Meier
  - ▷ Breslow

3.5 Review of Key Points (cont'd)

- Statistical tests to compare survival functions
  - ▷ log-rank test
  - ▷ Gehan-Wilcoxon test

Part IV

Regression Models for Time-to-Event Data

4.1 More Complex Research Questions

- We have seen how we can compare the survival curves of groups of patients
  - ▷ log-rank test
  - ▷ Peto & Peto modified Gehan-Wilcoxon test

- However, in many cases we may have more complex research questions – for example,
  - ▷ what is the effect of weight on survival (continuous covariate which we do not want to categorize)
  - ▷ what is the effect of treatment if we control for other variables (e.g., age at baseline, history of other diseases, etc.)

4.1 More Complex Research Questions (cont'd)

- To handle such type of questions we will use statistical models

- Statistical models are usually developed for one of the following reasons
  - ▷ effect estimation
  - ▷ hypothesis testing
  - ▷ prediction

- Different modelling strategies apply depending on the reason for which we develop a statistical model

4.2 Accelerated Failure Time Models

- In standard statistics we have the *Simple Linear Regression Model*:

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_p X_{ip} + \varepsilon_i$$

where

- ▷ Y_i denotes the value of the response variable for the ith subject
- ▷ $X_{i1}, \ldots, X_{ip}$ denote the value of the p explanatory variables (aka covariates)
- ▷ $\beta_0, \ldots, \beta_p$ regression coefficients
- ▷ ε_i random error terms – usually

$$\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$$

4.2 Accelerated Failure Time Models (cont'd)

- For survival data we have two complications:
  - ▷ the response variable is Time, which is always positive
  - ▷ censoring

- The solution to the first problem is simple, namely

$$\boxed{\text{Use } \log T_i^* \text{ instead of } T_i^* \text{ as a response variable}}$$

- Therefore, we obtain the model

$$\log T_i^* = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_p X_{ip} + \varepsilon_i$$

4.2 Accelerated Failure Time Models (cont'd)

- This model is known as the:

$$\boxed{\text{Accelerated Failure Time Model}}$$

4.2.1 AFT Models – Estimation

- The implications of censoring are twofold
  - ▷ estimation is more difficult from a theoretical point of view (but nowadays straightforward with modern computer software)
  - ▷ sensitivity to distributional assumptions for the error terms
    - * contrary to the linear regression model which is relatively robust against misspecification of the errors' distribution

4.2.1 AFT Models – Estimation (cont'd)

- Therefore, AFT models are not only based on the normal distribution but other distributions as well
  - ▷ Student's-t distribution – heavier tails than the normal
  - ▷ Logistic distribution
  - ▷ Extreme value distribution – this corresponds to the Weibull distribution for T_i, and it also has as a special case the Exponential distribution
  - ▷ . . .

4.2.1 AFT Models – Estimation (cont'd)

- The estimation of the parameters in the AFT model is typically based on the Maximum Likelihood (ML) method

- A (brief) review of ML: we want to find the values of the parameters that are more 'likely' in the light of the data

- As measure of likelihood we use the density function but we treat it as a function of the parameters given the sample at hand

$$L(\theta) = \prod_{i=1}^{n} f(y_i; \theta)$$

  where y_i are the data, and θ the parameters

4.2.1 AFT Models – Estimation (cont'd)

- The most 'likely' parameter values in the light of the data are the values that maximize the likelihood function

- For numerical reasons, it is more convenient to work with the log-likelihood function

$$\ell(\theta) = \sum_{i=1}^{n} \log f(y_i; \theta)$$

- The value of θ that maximizes $L(\theta)$ also maximizes $\ell(\theta) \Rightarrow$ sufficient to maximize $\ell(\theta)$

4.2.1 AFT Models – Estimation (cont'd)

- ML for censored data requires special treatment because not all subject provide the same kind of information

- Remember: the observed event time T_i is the true failure times T_i^* if subject i had the event or the last time point at which we know this subject was still alive (i.e., in this case $T_i^* > T_i$) – (see Section 3.1)

- Therefore, we have two sources of information in the log-likelihood function
  - ▷ subjects who experience the event ⇒ we use $f(T_i; \theta)$, the density function of the assumed distribution for T_i
  - ▷ subjects who did **not** experience the event ⇒ we use Pr$(T_i^* > T_i) = S(T_i; \theta)$, the survival function of the assumed distribution for T_i

4.2.1 AFT Models – Estimation (cont'd)

- Thus, the likelihood and log-likelihood functions take the form

$$L(\theta) \;=\; \prod_{i=1}^{n} f(T_i;\theta)^{\delta_i} \times S(T_i;\theta)^{1-\delta_i} \Rightarrow$$

$$\ell(\theta) \;=\; \sum_{i=1}^{n} \delta_i \log f(T_i;\theta) + (1-\delta_i) \log S(T_i;\theta)$$

where

▷ T_i denotes the observed event times, and δ_i is the event indicator

▷ θ denotes all model parameters, i.e., the β's and the variance parameter of the error terms σ^2

4.2.1 AFT Models – Estimation (cont'd)

- An alternative formulation of the log-likelihood (especially useful for proportional hazards models that we will see later) is in terms of the hazard function

$$\ell(\theta) = \sum_{i=1}^{n} \delta_i \log h(T_i; \theta) - H(T_i; \theta)$$

where we use the relations (see Section 2.5)

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

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

4.2.1 AFT Models – Estimation (cont'd)

- The Maximum Likelihood Estimates (MLEs) cannot be obtained analytically

- Therefore, to find the MLEs we use optimization algorithms that maximize the log-likelihood with respect to θ numerically
  - ▷ Newton-Raphson algorithm
  - ▷ quasi-Newton algorithm

- These algorithms are implemented in standard software, such as R and SAS

4.2.1 AFT Models – Estimation (cont'd)

- The obtained MLEs, which are usually denoted as $\hat{\theta}$, are asymptotically (i.e., when the sample size is large) normally distributed

$$\hat{\theta} \sim \mathcal{N}\big(\theta_0, \{\mathcal{I}(\theta_0)\}^{-1}\big)$$

where

- ▷ θ_0 denotes the true parameter values
- ▷ $\mathcal{I}(\theta_0)$ Fisher Information matrix

4.2.2 AFT Models – Interpretation of Parameters

- The model is

$$\log T_i^* = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_p X_{ip} + \varepsilon_i$$

- One-unit change in variable X_1 corresponds to

$$\begin{aligned} \log T_i^* &= \beta_0 + \beta_1 x + \beta_2 X_{i2} + \ldots + \beta_p X_{ip} \\ \log T_i^* &= \beta_0 + \beta_1 (x + 1) + \beta_2 X_{i2} + \ldots + \beta_p X_{ip} \end{aligned}$$

- Therefore,

$$\beta_1 = \log T_i^* - \log T_i^*$$

4.2.2 AFT Models – Interpretation of Parameters

- In general, one-unit change in variable X_j, $(j = 1, \ldots, p)$ corresponds to
  - ▷ a β_j change in the average $\log T_i^*$
  - ▷ multiplies average T_i^* by a factor of $\exp(\beta_j)$

- Now it is more clear where the name Accelerated Failure Time models stems from
  - ▷ the regression coefficients β directly quantify whether the survival time accelerates or decelerates for a one-unit change in the covariate values

4.2.2 AFT Models – Interpretation of Parameters

- $\boxed{\text{Example:}}$ for the PBC data, we are interested in the treatment effect on survival times after correcting for the effects of Gender and Age at baseline

- To put it in a regression model notation

$$\log T_i^* = \beta_0 + \beta_1 \texttt{Treat}_i + \beta_2 \texttt{Sex}_i + \beta_3 \texttt{Age}_i + \varepsilon_i$$

we are interested in β_1

- We fit the model assuming normal error terms

4.2.2 AFT Models – Interpretation of Parameters

- The results are

	est.	(s.e.)
β_0 – Intercept	4.42	(0.60)
β_1 – D-penicil	0.21	(0.19)
β_2 – Female	0.30	(0.28)
β_3 – Age	-0.05	(0.06)

4.2.2 AFT Models – Interpretation of Parameters

- The coefficient for the active treatment is $\beta_1 = 0.21$

- This means that for patients of the same gender and of the same age at baseline,
  - ▷ the log survival time is 0.21 larger on average for patients receiving D-penicil compared to patients receiving placebo
  - ▷ the average survival time for the D-penicil patients is $\exp(0.21) = 1.23$ times the average survival time of the placebo patients

4.2.2 AFT Models – Interpretation of Parameters

R> AFT models are fitted using function `survreg()`. The `dist` argument specifies the assumed distribution for T_i^* (not the error terms ε_i) – for the PBC data the following code produces the fit under the Weibull and log-normal distributions:

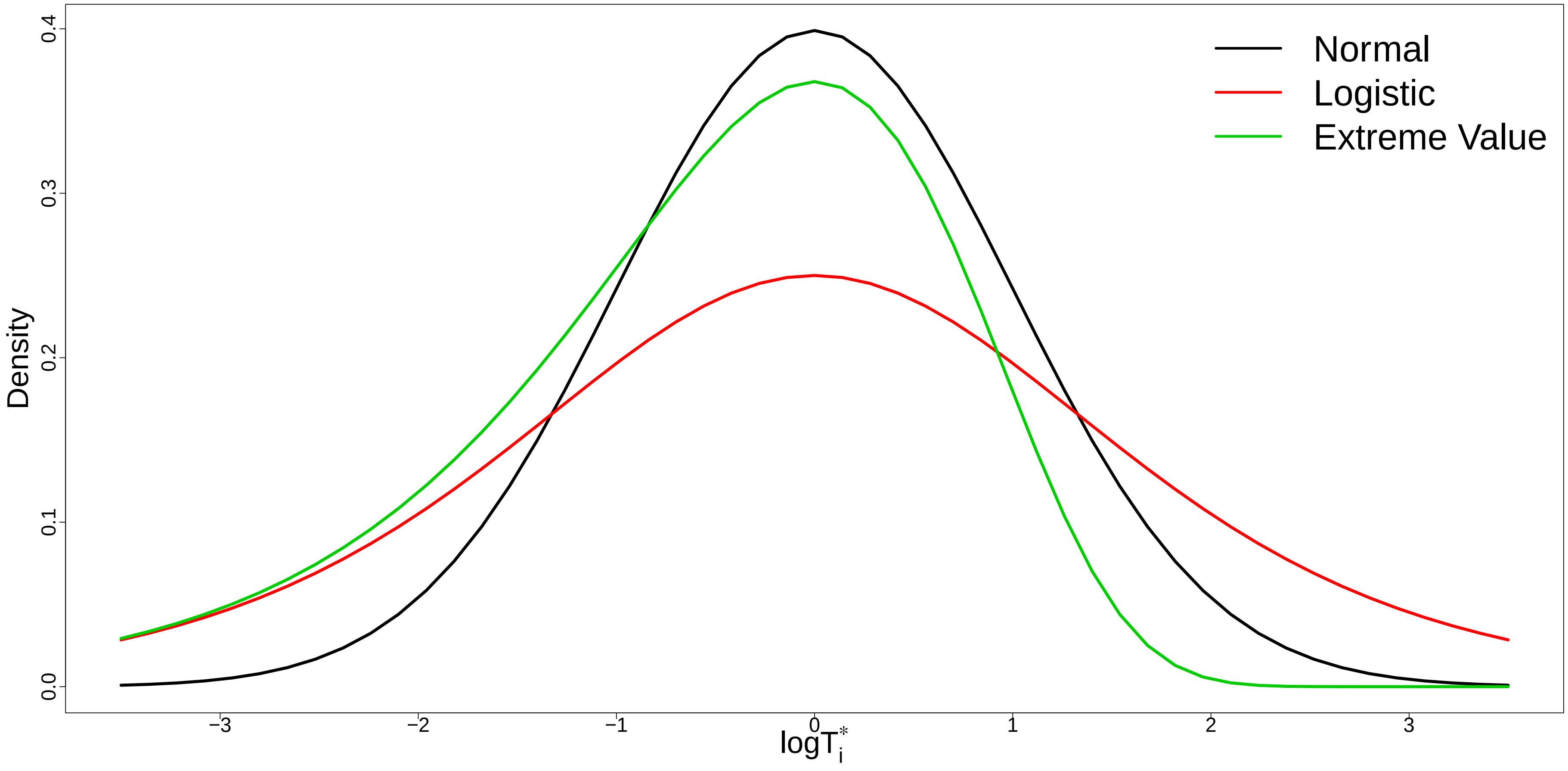
```
survreg(Surv(years, status2) ~ drug + sex + age, data = pbc2.id,
        dist = "weibull")

survreg(Surv(years, status2) ~ drug + sex + age, data = pbc2.id,
        dist = "lognormal")
```

4.2.2 AFT Models – Interpretation of Parameters

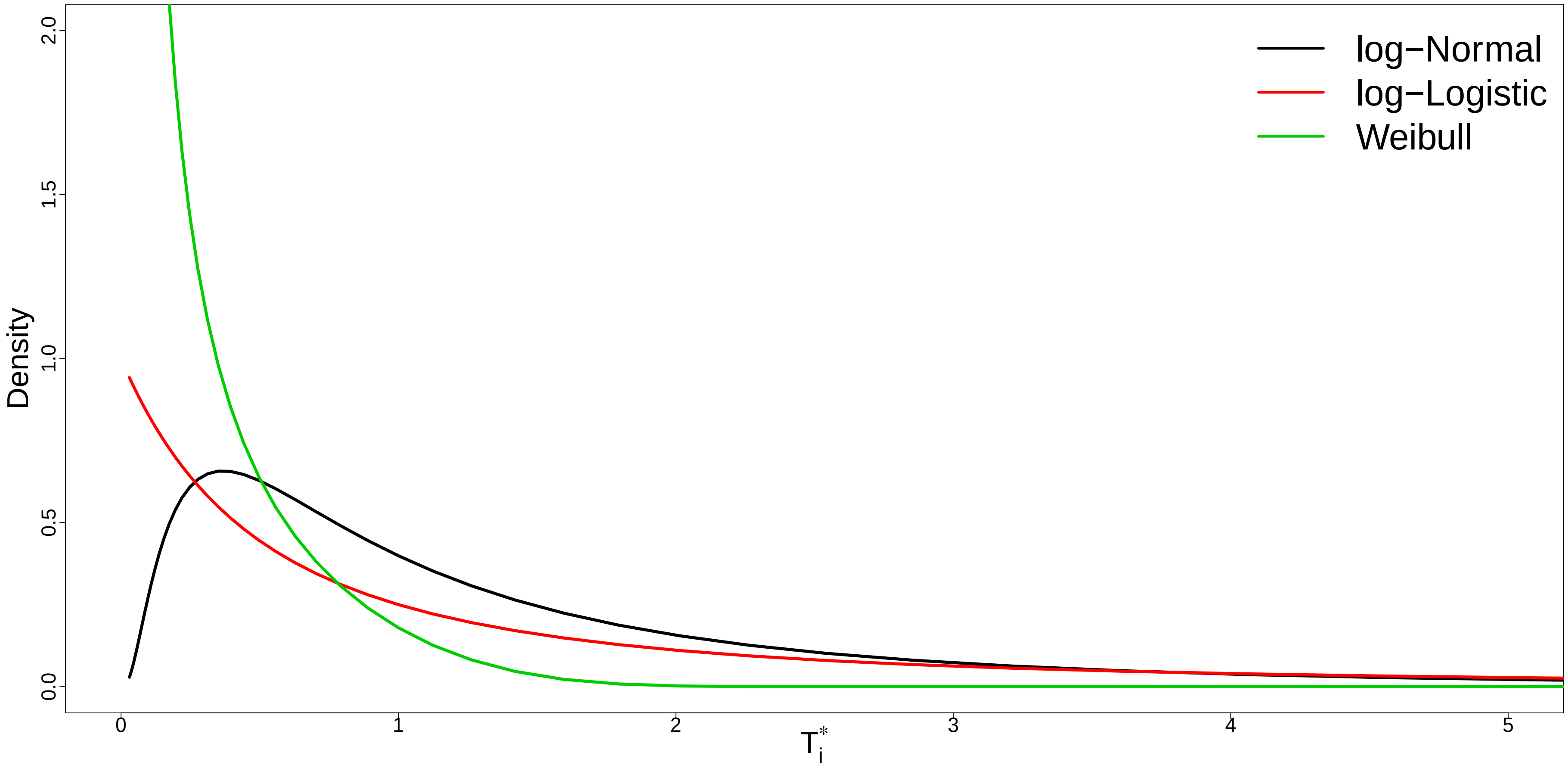
R> The available distributions for T_i^* in R are:

T_i^*	$\log T_i^*$
Weibull / Exponential	Extreme Value
log-Normal	Normal
log-Logistic	Logistic
Normal	—
Logistic	—

4.2.2 AFT Models – Interpretation of Parameters



4.2.2 AFT Models – Interpretation of Parameters



4.2.3 AFT Models – Sensitivity Analysis

- Sensitivity analysis under different error distributions

	Expontl	Weibull	log-Normal	log-Logistic
	est. (s.e.)	est. (s.e.)	est. (s.e.)	est. (s.e.)
Intercept	4.33 (0.52)	4.14 (0.48)	4.42 (0.60)	4.00 (0.55)
D-penicil	0.14 (0.17)	0.13 (0.16)	0.21 (0.19)	0.15 (0.17)
Female	0.48 (0.22)	0.44 (0.20)	0.30 (0.28)	0.38 (0.24)
Age	−0.04 (0.01)	−0.04 (0.07)	−0.05 (0.06)	−0.04 (0.01)

4.2.3 AFT Models – Sensitivity Analysis (cont'd)

- We observe that the distributional assumptions for the error terms have an effect on the derived parameter estimates and standard errors

- For instance, the gender effect ranges
  - ▷ from 0.30 (s.e. $=$ 0.28) for the log-normal model
  - ▷ to 0.48 (s.e. $=$ 0.22) for the exponential model

- We will come back to this later!

4.2.4 AFT Models – Displaying Effects

- Regression models often contain several explanatory variables, possibly interacting with each other ⇒ it is difficult to understand how the relationships between these explanatory variables and the response variable interplay

- Solution: communicate the results of a statistical model using Effect Plots
  - ▷ a picture is worth a 1000 words

- What are Effect Plots
  - ▷ choose a specific effect, i.e., a specific combination of the levels of the covariates
    - * for categorical covariates, specific categories
    - * for continuous covariates, specific quantiles or range of values
  - ▷ calculate the fitted average, including also the standard error of the fit

4.2.4 AFT Models – Displaying Effects (cont'd)

- ▷ display this fitted average with the associated 95% pointwise Confidence Intervals, preferably using trellis plots

- $\boxed{\text{Example:}}$ for the AFT model fitted to the PBC data, we display how the average survival time of female patients changes with respect to the age at baseline, separately for the 2 treatment groups

- In model terms
  - ▷ `Treat`: both groups are considered, i.e., 'placebo' and 'D-penicil'
  - ▷ `Sex`: set to 'female'
  - ▷ `Age`: varies from the minimum observed age to the maximum

4.2.4 AFT Models – Displaying Effects (cont'd)

- Based on the AFT model on p.130 and the results on p.131, we can calculate the average log failure time as estimated by the model

$$\begin{aligned}
\text{Placebo, Age} = 30 &\Rightarrow \log \widehat{T}_i^* = 4.42 + \phantom{0.21 +{}} 0.30 - 0.05 \times 30 = 3.27\\
\text{D-penicil, Age} = 30 &\Rightarrow \log \widehat{T}_i^* = 4.42 + 0.21 + 0.30 - 0.05 \times 30 = 3.48\\[1em]
\text{Placebo, Age} = 40 &\Rightarrow \log \widehat{T}_i^* = 4.42 + \phantom{0.21 +{}} 0.30 - 0.05 \times 40 = 2.78\\
\text{D-penicil, Age} = 40 &\Rightarrow \log \widehat{T}_i^* = 4.42 + 0.21 + 0.30 - 0.05 \times 40 = 2.99\\[1em]
\text{Placebo, Age} = 50 &\Rightarrow \log \widehat{T}_i^* = 4.42 + \phantom{0.21 +{}} 0.30 - 0.05 \times 50 = 2.30\\
\text{D-penicil, Age} = 50 &\Rightarrow \log \widehat{T}_i^* = 4.42 + 0.21 + 0.30 - 0.05 \times 50 = 2.51
\end{aligned}$$

4.2.4 AFT Models – Displaying Effects (cont'd)

- And then we can transform back to the original time scale using exp(), i.e.,

$$\begin{aligned}
\text{Placebo, Age} = 30 &\Rightarrow \widehat{T_i^*} = \exp(4.42 + \phantom{0.21 +{}} 0.30 - 0.05 \times 30) = 26.19\\
\text{D-penicil, Age} = 30 &\Rightarrow \widehat{T_i^*} = \exp(4.42 + 0.21 + 0.30 - 0.05 \times 30) = 32.38\\[1em]
\text{Placebo, Age} = 40 &\Rightarrow \widehat{T_i^*} = \exp(4.42 + \phantom{0.21 +{}} 0.30 - 0.05 \times 40) = 16.14\\
\text{D-penicil, Age} = 40 &\Rightarrow \widehat{T_i^*} = \exp(4.42 + 0.21 + 0.30 - 0.05 \times 40) = 19.95\\[1em]
\text{Placebo, Age} = 50 &\Rightarrow \widehat{T_i^*} = \exp(4.42 + \phantom{0.21 +{}} 0.30 - 0.05 \times 50) = 9.95\\
\text{D-penicil, Age} = 50 &\Rightarrow \widehat{T_i^*} = \exp(4.42 + 0.21 + 0.30 - 0.05 \times 50) = 12.30
\end{aligned}$$

4.2.4 AFT Models – Displaying Effects (cont'd)

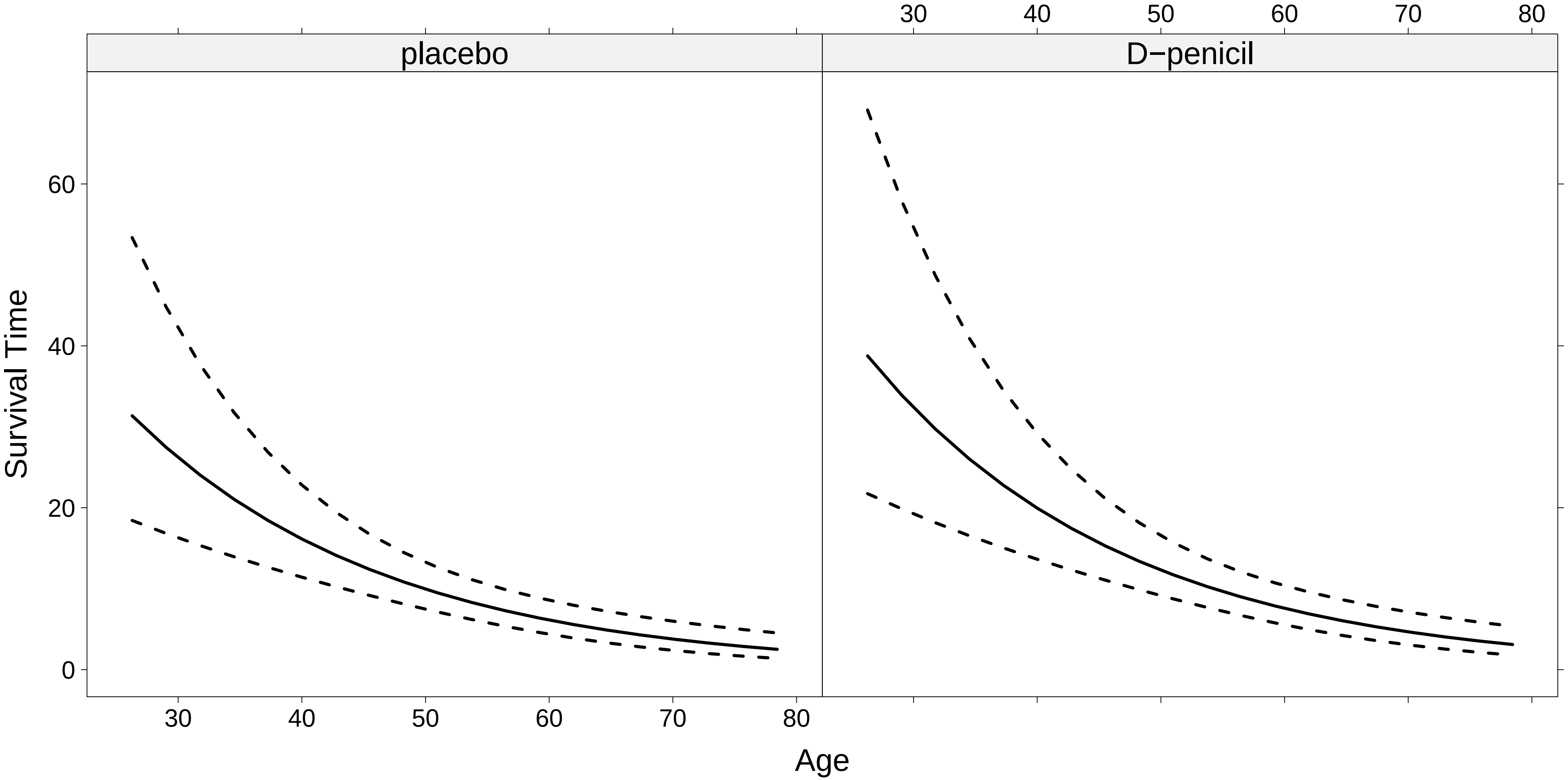
- In addition, for the estimated $\log \widehat{T}_i^*$ we also obtain a standard error – using this standard error we can calculate 95% pointwise confidence intervals

$$\log \widehat{T}_i^* \mp 1.96 \times s.e.(\log \widehat{T}_i^*)$$

which can also be transformed to the original scale

$$\exp\Big\{\log \widehat{T}_i^* \mp 1.96 \times s.e.(\log \widehat{T}_i^*)\Big\}$$

4.2.4 AFT Models – Displaying Effects (cont'd)



4.2.4 AFT Models – Displaying Effects (cont'd)

R> To produce effects, first you fit the desired AFT model using survreg()

```
fit <- survreg(Surv(years, status2) ~ drug + sex + age,
    data = pbc2.id, dist = "lognormal")
```

4.2.4 AFT Models – Displaying Effects (cont'd)

R> At the second stage you need to construct a data frame that contains the combination of covariates for which you would like to compute effects

```
ND <- with(pbc2.id, data.frame(
    age = rep(seq(min(age), max(age), length.out = 20), 2),
    sex = rep("female", 40),
    drug = gl(2, 20, labels = c("placebo", "D-penicil"))
))
```

4.2.4 AFT Models – Displaying Effects (cont'd)

R> This data frame is then used in the predict(), which provides estimates for the desired effects and their standard errors

```
prs <- predict(fit, ND, se.fit = TRUE, type = "lp")
ND$pred <- prs[[1]]
ND$se <- prs[[2]]
ND$lo <- exp(ND$pred - 1.96 * ND$se)
ND$up <- exp(ND$pred + 1.96 * ND$se)
ND$pred <- exp(ND$pred)
```

4.2.4 AFT Models – Displaying Effects (cont'd)

R> Finally we plot the result

```
library(lattice)
xyplot(pred + lo + up ~ age | drug, data = ND, type = "l",
    lty = c(1,2,2), col = "black", lwd = 4, xlab = "Age",
    ylab = "Survival Time")
```

4.2.5 AFT Models – Hypothesis Testing

- As we have seen, one of the purposes of modelling is to test complex hypothesis of scientific interest – in general, we may be interested in

$$\begin{aligned} H_0 : \theta &= \theta_0 \\ H_a : \theta &\neq \theta_0 \end{aligned}$$

- Since we have fitted the AFT model under maximum likelihood, the standard statistical tests are available

- That is we can choose from
  - ▷ Wald test
  - ▷ Likelihood Ratio test
  - ▷ Score test

4.2.5 AFT Models – Hypothesis Testing (cont'd)

- The Wald test is defined as

$$(\hat{\theta}_a - \theta_0)^\top \{\hat{\textsf{var}}(\hat{\theta}_a)\}^{-1} (\hat{\theta}_a - \theta_0) \sim \chi^2_p$$

where

▷ $\hat{\theta}_a$ the maximum likelihood estimate under the alternative hypothesis

▷ $\hat{\textsf{var}}(\hat{\theta}_a) = \bigl\{ -\partial^2 \ell(\theta)/\partial\theta^\top \partial\theta \big|_{\theta=\hat{\theta}_a} \bigr\}^{-1}$ denotes the covariance matrix of the MLEs

▷ p denotes the number of parameters being tested

- The Wald test requires fitting the model only under the alternative hypothesis

4.2.5 AFT Models – Hypothesis Testing (cont'd)

- The Score test is defined as

$$\mathcal{S}(\hat{\theta}_0)^\top \hat{\text{var}}(\hat{\theta}_0)\mathcal{S}(\hat{\theta}_0) \sim \chi^2_p$$

  where

  ▷ $\hat{\theta}_0$ the maximum likelihood estimate under the null hypothesis

  ▷ $\mathcal{S}(\theta) = \partial \ell(\theta) / \partial \theta$ denotes the score vector

  ▷ $\hat{\text{var}}(\hat{\theta}_0)$ denotes the covariance matrix of the MLEs

  ▷ p denotes the number of parameters being tested

- The Score test requires fitting the model only under the null hypothesis

4.2.5 AFT Models – Hypothesis Testing (cont'd)

- The likelihood ratio test (LRT) is defined as

$$-2 \times \{\ell(\hat{\theta}_0) - \ell(\hat{\theta}_a)\} \sim \chi^2_p$$

where

▷ $\ell(\cdot)$ the value of the log-likelihood function

▷ $\hat{\theta}_0$ the maximum likelihood estimate under the null hypothesis

▷ $\hat{\theta}_a$ the maximum likelihood estimate under the alternative hypothesis

▷ p denotes the number of parameters being tested

- The LRT requires fitting the model under both the null & alternative hypotheses

4.2.5 AFT Models – Hypothesis Testing (cont'd)

- Asymptotically (i.e., for large samples) these three tests are equivalent

- **Advice:** prefer to use the likelihood ratio test over the other two

- Why:
  - ▷ it has better theoretical properties
  - ▷ it makes you carefully think about the hypotheses being tested

- Caveat: if you have missing data in the variable(s) being tested then you have to work with complete cases ⇒ decreased efficiency

4.2.5 AFT Models – Hypothesis Testing (cont'd)

- Example: in the AIDS data set we assume that the average failure time may be different
  - ▷ for males and females (main effect of Sex), and
  - ▷ for patients with AZT intolerance (main effect of AZT), but also that
  - ▷ the effect of AZT intolerance on the average failure time is not expected to be the same in males and females (interaction effect)

- We are interested in testing whether AZT intolerance has any effect at all in the average failure time (overall effect of AZT)

4.2.5 AFT Models – Hypothesis Testing (cont'd)

- The full model (i.e., the model under the alternative hypothesis)

$$\log T_i^* = \beta_0 + \beta_1 \texttt{Sex}_i + \beta_2 \texttt{AZT}_i + \beta_3 \texttt{Sex:AZT}_i + \varepsilon_i$$

  we fit the model assuming the extreme value distribution for the error terms (Weibull distribution for T_i^*)

- The reduced model (i.e., the model under the null hypothesis) **must be** a special case of the full model

$$\log T_i^* = \beta_0 + \beta_1 \texttt{Sex}_i + \varepsilon_i$$

4.2.5 AFT Models – Hypothesis Testing (cont'd)

- The models are

$$\begin{aligned} H_0: \ \log T_i^* &= \beta_0 + \beta_1 \texttt{Sex}_i + \varepsilon_i \\ H_a: \ \log T_i^* &= \beta_0 + \beta_1 \texttt{Sex}_i + \beta_2 \texttt{AZT}_i + \beta_3 \texttt{Sex:AZT}_i + \varepsilon_i \end{aligned}$$

- Therefore, the hypothesis that we are interested in is

$$H_0 \ : \ \beta_2 = \beta_3 = 0$$

$$H_a \ : \ \text{at least one of } \beta_2 \text{ and } \beta_3 \text{ is different from 0}$$

4.2.5 AFT Models – Hypothesis Testing (cont'd)

- Understanding the models:
  - ▷ reference levels: for `Sex` is 'female' and for `AZT` is 'failure'

$$
\begin{array}{lllcl}
\text{Female,} & \text{AZT Fail} & \Rightarrow \log T_i^* & = & \beta_0 \qquad\qquad\qquad\qquad\qquad + \varepsilon_i \\
\text{Male,} & \text{AZT Fail} & \Rightarrow \log T_i^* & = & \beta_0 + \beta_1 \qquad\qquad\qquad\quad + \varepsilon_i \\
\text{Female,} & \text{AZT Intol} & \Rightarrow \log T_i^* & = & \beta_0 \qquad\quad + \beta_2 \qquad\quad + \varepsilon_i \\
\text{Male,} & \text{AZT Intol} & \Rightarrow \log T_i^* & = & \beta_0 + \beta_1 + \beta_2 + \beta_3 + \varepsilon_i
\end{array}
$$

4.2.5 AFT Models – Hypothesis Testing (cont'd)

- Understanding the models:
  - ▷ reference level for `Sex` is 'female'

$$\begin{aligned} \text{Female} &\Rightarrow \log T_i^* = \beta_0 \qquad\quad + \varepsilon_i \\ \text{Male} &\Rightarrow \log T_i^* = \beta_0 + \beta_1 + \varepsilon_i \end{aligned}$$

4.2.5 AFT Models – Hypothesis Testing (cont'd)

- Likelihood ratio test
  - ▷ log-likelihood under the reduced model $\ell(\hat{\theta}_0) = -826.18$
  - ▷ log-likelihood under the alternative model $\ell(\hat{\theta}_a) = -813.02$
  - ▷ parameters being tested $p = 2$
  - ▷ LRT $= -2 \times \{-826.18 - (-813.02)\} = 26.32$, $df = 2$, p-value < 0.001

4.2.5 AFT Models – Hypothesis Testing (cont'd)

R> The LRT and the associated p-value is computed by the anova() method. This function calculates the LRT based on two fitted AFT models – the user is responsible to supply nested models for the LRT to be valid

```
fit.null <- survreg(Surv(Time, death) ~ gender,
        data = aids.id)

fit.alt <- survreg(Surv(Time, death) ~ gender * AZT,
        data = aids.id)

anova(fit.null, fit.alt)
```

4.2.6 AFT Models – Checking Assumptions

- All statistical models are based on assumptions

- In order the derived results from a model to be valid, these assumptions need to hold or at least not to be severely violated

- For the AFT model the assumptions are
  - ▷ linearity (to be discussed later – see Section 4.5.1)
  - ▷ additivity (to be discussed later – see Section 4.5.2)
  - ▷ **error terms distribution**

4.2.6 AFT Models – Checking Assumptions (cont'd)

- We rewrite the AFT model

$$\begin{aligned}\log T_i^* &= \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_p X_{ip} + \sigma\varepsilon_i \;\Rightarrow \\ \varepsilon_i &= \{\log T_i^* - (\beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_p X_{ip})\}/\sigma\end{aligned}$$

- Thus, under the AFT model we can check the assumptions for the error terms using the residuals

$$r_i = \{\log T_i - (\hat\beta_0 + \hat\beta_1 X_{i1} + \hat\beta_2 X_{i2} + \ldots + \hat\beta_p X_{ip})\}/\hat\sigma$$

where

▷ $\hat\beta$ denotes the estimated regression coefficients

▷ $\hat\sigma$ the estimated scale parameter of the error terms

4.2.6 AFT Models – Checking Assumptions (cont'd)

- The error terms are defined with respect to the true failure times T_i^*. However, the residuals are defined based on the observed failure times T_i

- Thus, when T_i is censored r_i will be censored as well

- Censoring must be taken into account when using the residuals to check the AFT model assumptions

- For instance, we can use the Kaplan-Meier estimate of r_i and compare it with the assumed survival function for the error terms

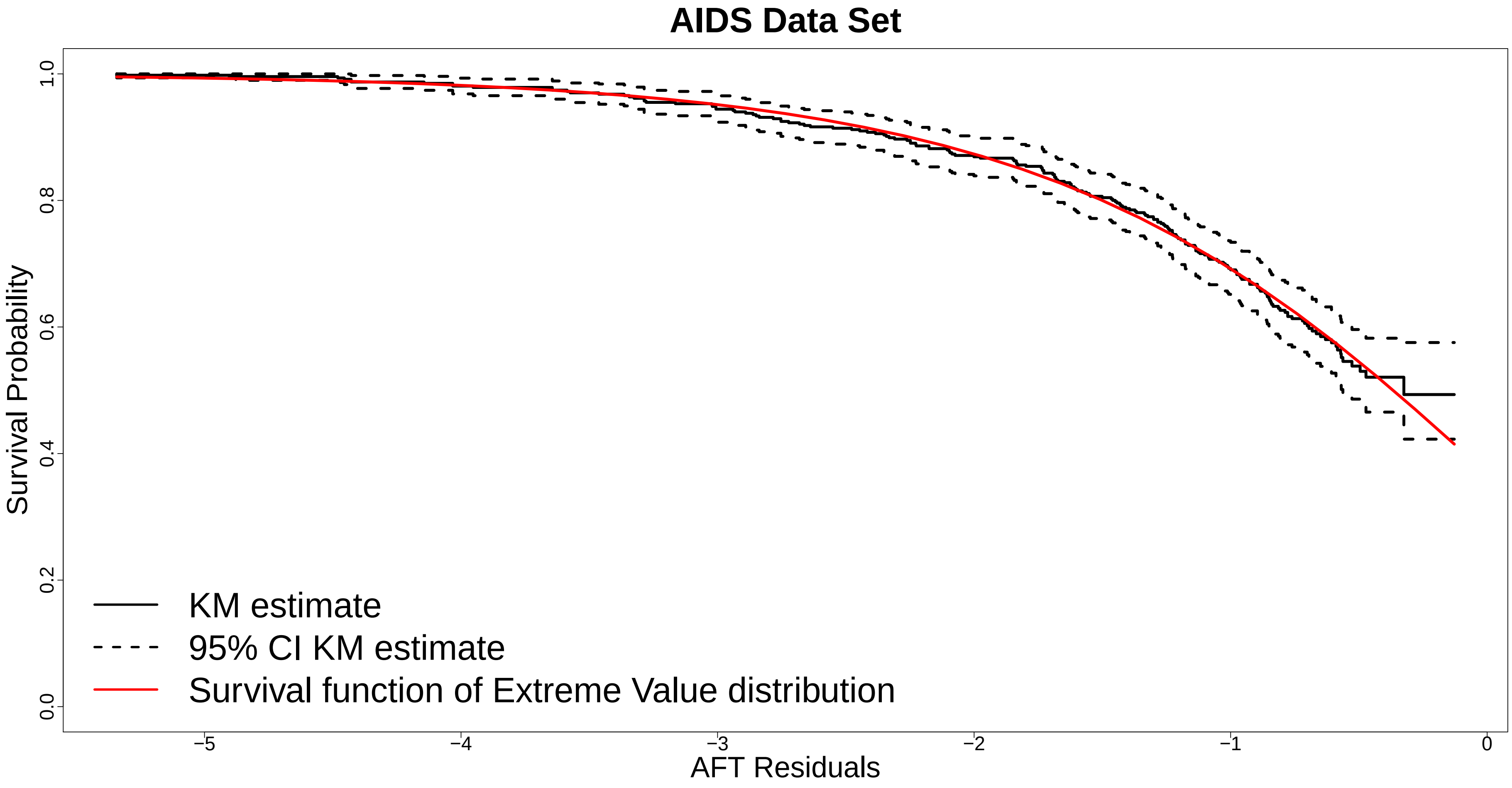
4.2.6 AFT Models – Checking Assumptions (cont'd)

- $\boxed{\text{Example:}}$ we display the AFT residuals for the Weibull model fitted to
  - ▷ the AIDS data set using as only covariate the randomized treatment
  - ▷ the Stanford data set with only an intercept term

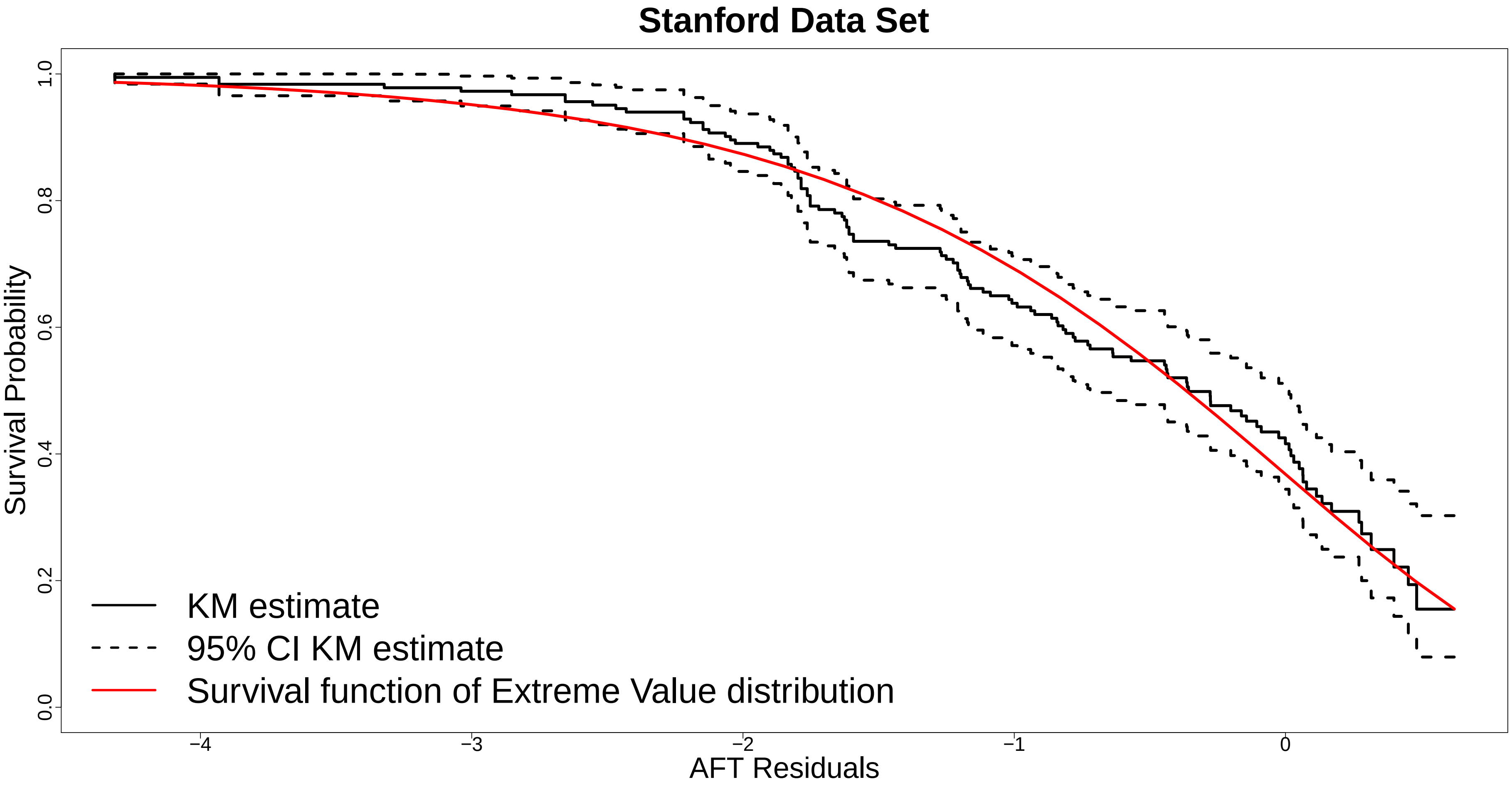
- In particular, we compare
  - ▷ the Kaplan-Meier estimate of the survival function of r_i, with
  - ▷ the survival function of the Extreme Value distribution (this is the distribution which we have assumed for the error terms)

- If the model fits the data well, the two survival functions should be close to each other

4.2.6 AFT Models – Checking Assumptions (cont'd)



4.2.6 AFT Models – Checking Assumptions (cont'd)



4.2.6 AFT Models – Checking Assumptions (cont'd)

- We observe that the Weibull model provides a much better fit to the AIDS data set than to the Stanford data set

- Therefore, conclusions extracted from the fit of the Weibull model to the Stanford data set should be treated with some caution!

4.2.6 AFT Models – Checking Assumptions (cont'd)

R> The AFT residuals can be easily calculated using their definition

```
# fit the model
fit.weib <- survreg(Surv(Time, death) ~ drug, data = aids.id,
        dist = "weibull")

# extract fitted values
fits <- fit.weib$linear.predictors

# compute the AFT residuals
resids <- (fit.weib$y[, 1] - fits) / fit.weib$scale
```

4.2.6 AFT Models – Checking Assumptions (cont'd)

R> Following we need to compute the Kaplan-Meier estimate of the survival function of these residuals and compare it with the survival function of the distribution of the error terms

```
resKM <- survfit(Surv(resids, death) ~ drug, data = aids.id)

# plot the KM estimate
plot(resKM, mark.time = FALSE)

# superimpose the survival function of the assumed
# extreme value distribution
xx <- seq(min(resids), max(resids), length.out = 35)
yy <- exp(- exp(xx))
lines(xx, yy, col = "red", lwd = 2)
```

4.3 Cox Model

- So far we have seen that parametric models (i.e., models which make specific distributional assumptions (e.g., log-Normal, Weibull, log-Logistic, etc.) for the event time data) are sensitive, mainly due to censoring

- That is, changing the distribution of the error terms in the AFT model can have a profound effect on parameter estimates and standard errors, and as a result on inference

- This problem has lead to the development of **Semiparametric Regression Models**
  - ▷ i.e., regression models (parametric component) that make no assumptions for the distribution of the failure times (nonparametric component)

4.3 Cox Model (cont'd)

- The most well known semiparametric regression model for survival data is the

Cox Proportional Hazards Model

proposed by sir D.R. Cox in 1972 (Journal of the Royal Statistical Society, Series B)

4.3.1 Cox Model – Common Ratios in Statistics

- Before going into more details about the Cox model,

- Let's review some common ratios in statistics
  - ▷ odds ratio
  - ▷ risk ratio (aka relative risk)
  - ▷ hazards ratio

4.3.1 Cox Model – Common Ratios in Statistics

- Odds ratio

$$OR = \frac{p_A/(1-p_A)}{p_B/(1-p_B)} = \frac{p_A(1-p_B)}{p_B(1-p_A)}$$

  is the ratio of odds of the event occurring in group A to the odds of it occurring in group B

  ▷ p_A the probability of event in group A

  ▷ p_B the probability of event in group B

4.3.1 Cox Model – Common Ratios in Statistics

- Risk ratio (Relative Risk)

$$RR = \frac{p_A}{p_B}$$

is the ratio of the probability of event in group A to the probability of event in group B

4.3.1 Cox Model – Common Ratios in Statistics

- Hazards ratio

$$HR = \frac{h_A(t)}{h_B(t)}$$

  is the ratio of the hazard for an event at time t for group A to the hazard for an event at t for group B

- The Hazard Ratio is related to the Relative Risk but they are not exactly equivalent

- The difference between the two can be illustrated in the following hypothetical example

4.3.1 Cox Model – Common Ratios in Statistics

- Say that we have 3 patients with increasingly worse prognostic factors
  - ▷ e.g., higher age corresponds to higher risk, and Patient 1 is 20 years old, Patient 2 is 40 years old and Patient 3 is 60 years old

- We assume proportional hazards, that is

$$h_A(t) = 0.5 h_B(t)$$

  where $h_A(t)$ is the hazard of the treated group, and $h_B(t)$ the hazard of the control

- Remember: the survival and hazard functions are related (see Section 2.5)

$$S(t) = \exp\left(-\int_0^t h(s)\,ds\right)$$

4.3.1 Cox Model – Common Ratios in Statistics

- Therefore we have

$$\begin{aligned} S_A(t) &= \exp\Big(- \int_0^t h_A(s)\, ds\Big) \\ &= \exp\Big(- \int_0^t 0.5 h_B(s)\, ds\Big) \\ &= \Big\{ \exp\Big(- \int_0^t h_B(s)\, ds\Big)\Big\}^{0.5} \\ &= \{S_B(t)\}^{0.5} \end{aligned}$$

4.3.1 Cox Model – Common Ratios in Statistics

Patient	5 Y. Survival		Risk Ratio
	B	A	(A/B)
1	0.90	0.95	$0.05/0.10 = 0.50$
2	0.50	0.71	$0.29/0.50 = 0.58$
3	0.20	0.45	$0.55/0.80 = 0.69$

- The risk ratio depends on the survival rate of the B patients

4.3.2 Cox Model – Formulation

- The Cox model assumes that the effect of covariates is multiplicative in the hazard scale, i.e.,

$$\begin{aligned} h_i(t) &= h_0(t)\exp(\beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_p X_{ip}) \;\Rightarrow \\ \log h_i(t) &= \log h_0(t) + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_p X_{ip} \end{aligned}$$

  where

  ▷ $X_{i1}, \ldots, X_{ip}$ denote p explanatory variables (aka covariates) – note: there is no intercept term!

  ▷ $h_i(t)$ denotes the hazard for an event for patient i at time t

  ▷ $h_0(t)$ denotes the baseline hazard

4.3.2 Cox Model – Formulation (cont'd)

- The baseline hazard $h_0(t)$ represents the hazard for an event when all the covariates or all the βs are 0

- That is, $h_0(t)$ represents the instantaneous risk of experiencing the event at time t, without the influence of any covariate

- Therefore
  - ▷ if a covariate has a beneficial effect, it will decrease this baseline risk
  - ▷ if, on the other hand, it has a harmful effect, it will increase $h_0(t)$

4.3.3 Cox Model – Estimation

- The Cox model makes no assumptions for the baseline hazard function (nonparametric component)

- The model parameters $\beta_1, \ldots, \beta_p$ are estimated using a Semi-Parametric Maximum Likelihood (SPML) estimation approach

- In particular, $\beta_1, \ldots, \beta_p$ are obtained as the values that maximize the log partial likelihood function

$$p\ell(\beta) = \sum_{i=1}^{n} \delta_i \Bigl[(X_i^\top \beta) - \log \Bigl\{ \sum_{T_j \geq T_i} \exp(X_j^\top \beta) \Bigr\} \Bigr]$$

where $X_i^\top \beta = \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_p X_{ip}$

4.3.3 Cox Model – Estimation (cont'd)

- The obtained Maximum Partial Likelihood Estimates, which are usually denoted as $\hat{\beta}$, are asymptotically (i.e., when the number of events is large) normally distributed

$$\hat{\beta} \sim \mathcal{N}\bigl(\beta_0, \{\mathcal{I}_p(\beta_0)\}^{-1}\bigr)$$

where

▷ β_0 denotes the true values of parameters β

▷ $\{\mathcal{I}_p(\beta_0)\}$ expected information matrix based on the partial likelihood

4.3.3 Cox Model – Estimation (cont'd)

- Partial likelihood can be considered as measure of how well the model can order the patients with respect to their survival time.

- **Problem:** some times the hazard ratio for one covariate can be 0 – for example, consider the following data set

	Alive	Dead
Treatment	40	0
Control	30	10

Because the treatment group has no deaths its hazard rate is 0

4.3.3 Cox Model – Estimation (cont'd)

- If a Cox model is fitted to such data, then the estimated regression coefficient for treatment is ∞

- Software packages of course cannot detect this problematic case and will usually produce a large in magnitude estimate for β

- Therefore, if in the software output you observe a relatively large value for a β, be alarmed and check your data to see why is this happening (e.g., make tables)

4.3.4 Cox Model – Interpretation of Parameters

- The model is

$$\log h_i(t) \;=\; \log h_0(t) + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_p X_{ip}$$

- One-unit change in variable X_1, $(j = 1, \ldots, p)$ corresponds to

$$\log h_i(t) \;=\; \log h_0(t) + \beta_1 x + \beta_2 X_{i2} + \ldots + \beta_p X_{ip}$$

$$\log h_i(t) \;=\; \log h_0(t) + \beta_1 (x + 1) + \beta_2 X_{i2} + \ldots + \beta_p X_{ip}$$

- Therefore,

$$\beta_1 = \log h_i(t) - \log h_i(t) \quad\Rightarrow\quad \exp(\beta_1) = h_i(t) / h_i(t)$$

4.3.4 Cox Model – Interpretation of Parameters

- In general, one-unit change in variable X_j, $(j = 1, \ldots, p)$ corresponds to
  - ▷ a β_j change of $\log\{h_i(t)/h_0(t)\}$
  - ▷ increases $h_i(t)/h_0(t)$ by a factor of $\exp(\beta_j)$ (if $\beta_j < 0$, then $\exp(\beta_j) < 1$ and therefore the risk is decreased)

- Note: the interpretation of β_j $(j = 1, \ldots, p)$ is different from the one in the AFT model

4.3.4 Cox Model – Interpretation of Parameters

- Example: for the PBC data, we are interested in the treatment effect on the hazard for an event after correcting for the effects of Gender and Age at baseline

- To put in a regression model notation

$$\begin{aligned}\log h_i(t) &= \log h_0(t) + \beta_1 \texttt{Treat}_i + \beta_2 \texttt{Sex}_i + \beta_3 \texttt{Age}_i \\ h_i(t) &= h_0(t) \exp\bigl(\beta_1 \texttt{Treat}_i + \beta_2 \texttt{Sex}_i + \beta_3 \texttt{Age}_i\bigr)\end{aligned}$$

which means that we are interested in β_1

4.3.4 Cox Model – Interpretation of Parameters

- The results are

	est.	exp(est.)	(s.e.)
β_1 – D-penicil	-0.15	0.86	(0.17)
β_2 – Female	-0.47	0.62	(0.22)
β_3 – Age	0.42	1.04	(0.01)

4.3.4 Cox Model – Interpretation of Parameters

- The estimate for the treatment effect is $\beta_1 = -0.15$

- This means that for patients of the same gender and of the same age at baseline,
  - ▷ the log hazard of the D-penicillamine group is at any fixed point in time 0.15 lower than the log hazard of the placebo group
  - ▷ the hazard ratio of the D-penicillamine group to the placebo group is $\exp(-0.15) = 0.86$

4.3.4 Cox Model – Interpretation of Parameters

- For quantitative covariates β quantifies the effect of one-unit change in the hazard

- For instance, for age $\beta_3 = 0.42.$ Therefore, for patients of the same sex and who receive the same treatment,
  - ▷ the log hazard is increased by 0.42 for each one year increase in the baseline age
  - ▷ the hazard of a patient at $d + 1$ years old to the hazard of a patient at d years old is $\exp(0.42) = 1.04$

4.3.4 Cox Model – Interpretation of Parameters

R> Cox models are fitted using function `coxph()`. This has the same syntax as `survreg()` – for instance, for the PBC data the following code fits the Cox model that contains the main effects of 'drug', 'sex' and 'age':

```
coxph(Surv(years, status2) ~ drug + sex + age, data = pbc2.id)
```

4.3.5 Cox Model – Displaying Effects

- As also mentioned in the case of AFT models, Effect Plots are the optimal way for communicating the information in a model

- $\boxed{\text{Example:}}$ we fit a Cox model for the Lung data that contains
  - ▷ main effect of gender
  - ▷ main effect of age at baseline
  - ▷ main effect of Karnofsky performance score
  - ▷ interaction effect of gender with age

4.3.5 Cox Model – Displaying Effects (cont'd)

- In model notation we have

$$h_i(t) \;=\; h_0(t) \exp\bigl(\beta_1 \texttt{Sex}_i + \beta_2 \texttt{Age}_i + \beta_3 \texttt{Karno}_i + \beta_4 \texttt{Sex:Age}_i\bigr)$$

- We are interested in seeing how the risk for death (at any time point) changes with increasing age at baseline and separately for males and females

- Since in the Cox model we also corrected for the effect of the Karnofsky performance score, we need to specify a value for it

  ▷ a reasonable choice is the median Karnofsky score, which in our sample equals 80

4.3.5 Cox Model – Displaying Effects (cont'd)

- The results are

	est.	exp(est.)	(s.e.)
Female	0.26	1.30	(1.22)
Age	0.02	1.02	(0.01)
Karn Score	−0.01	0.99	(0.01)
Age:Female	−0.01	0.99	(0.02)

4.3.5 Cox Model – Displaying Effects (cont'd)

- Based on these results, we can calculate the average log hazard rate as estimated by the model ('M' denotes males and 'F' females)

$$\begin{aligned}
\text{M, Age} = 45 \;&\Rightarrow\; \log\{h_i(t)/h_0(t)\} = \qquad\quad 0.02 \times 45 - 0.01 \times 80 = -0.35\\
\text{F, Age} = 45 \;&\Rightarrow\; \log\{h_i(t)/h_0(t)\} = 0.26 + (0.02 - 0.01) \times 45 - 0.01 \times 80 = -0.63
\end{aligned}$$

$$\begin{aligned}
\text{M, Age} = 55 \;&\Rightarrow\; \log\{h_i(t)/h_0(t)\} = \qquad\quad 0.02 \times 55 - 0.01 \times 80 = -0.18\\
\text{F, Age} = 55 \;&\Rightarrow\; \log\{h_i(t)/h_0(t)\} = 0.26 + (0.02 - 0.01) \times 55 - 0.01 \times 80 = -0.58
\end{aligned}$$

$$\begin{aligned}
\text{M, Age} = 65 \;&\Rightarrow\; \log\{h_i(t)/h_0(t)\} = \qquad\quad 0.02 \times 65 - 0.01 \times 80 = -0.02\\
\text{F, Age} = 65 \;&\Rightarrow\; \log\{h_i(t)/h_0(t)\} = 0.26 + (0.02 - 0.01) \times 65 - 0.01 \times 80 = -0.54
\end{aligned}$$

4.3.5 Cox Model – Displaying Effects (cont'd)

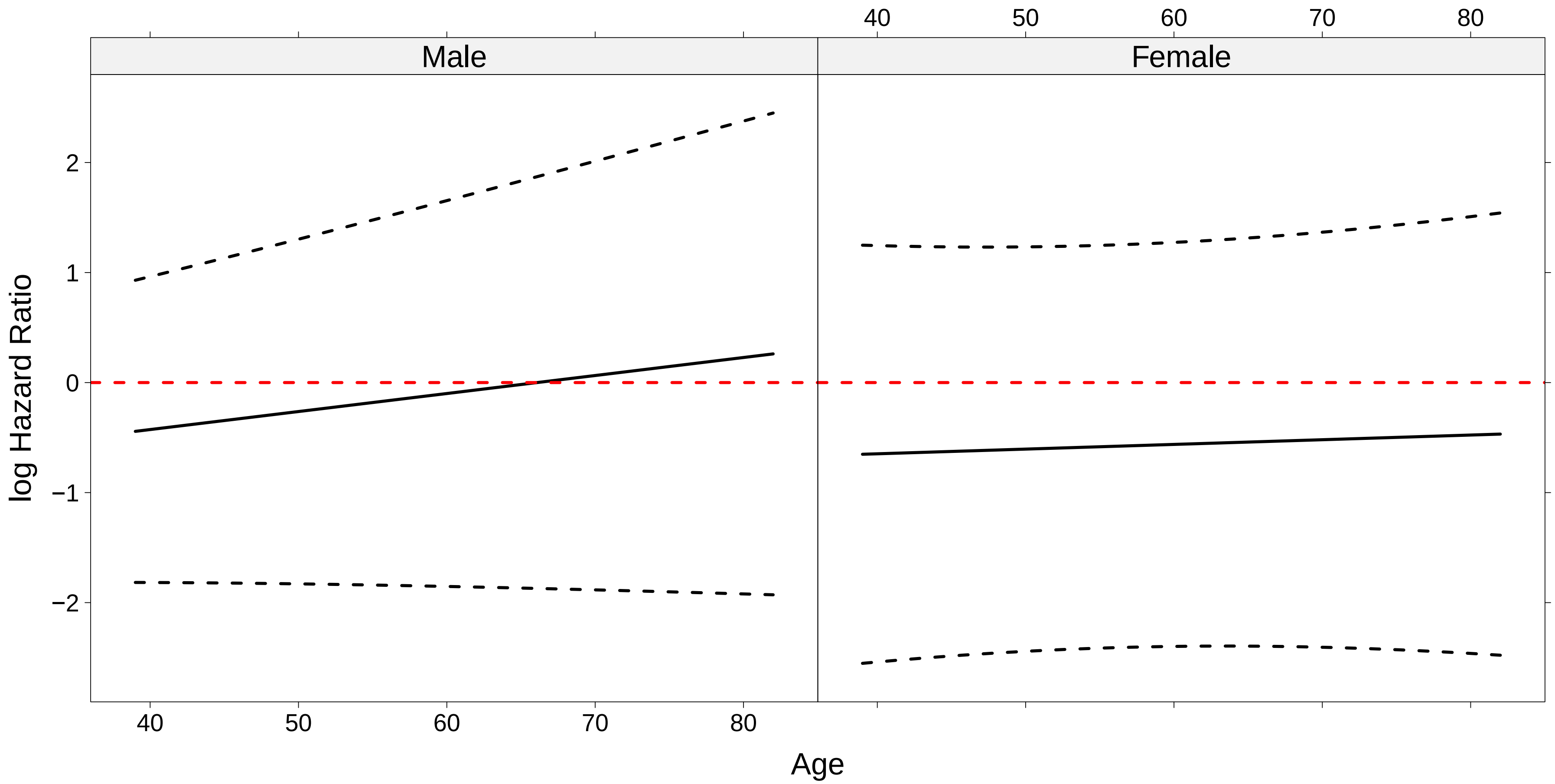
- In addition, for the estimated $\log \widehat{HR} = \log\{h_i(t)/h_0(t)\}$ we also obtain a standard error – using this standard error we can calculate 95% pointwise confidence intervals

$$\log \widehat{HR} \mp 1.96 \times s.e.(\log \widehat{HR})$$

which can also be transformed to the hazard scale by

$$\exp\Big\{\log \widehat{HR} \mp 1.96 \times s.e.(\log \widehat{HR})\Big\}$$

4.3.5 Cox Model – Displaying Effects (cont'd)



4.3.5 Cox Model – Displaying Effects (cont'd)

R> The construction of effect plots for Cox models proceeds in the same manner as for AFT models – we start by fitting the desired Cox model

```
fit <- coxph(Surv(time, status) ~ age*sex + ph.karno, data = lung)
```

4.3.5 Cox Model – Displaying Effects (cont'd)

R> Following we construct a data frame that contains the combination of covariates for which we would like to compute effects

```
ND <- with(lung, data.frame(
    age = rep(seq(min(age, na.rm = TRUE), max(age, na.rm = TRUE),
        length.out = 20), 2),
    sex = gl(2, 20, labels = c("Male", "Female")),
    ph.karno = rep(median(ph.karno, na.rm = TRUE), 40)
))
```

4.3.5 Cox Model – Displaying Effects (cont'd)

R> This data frame is then used in the predict(), which provides estimates for the desired effects and their standard errors

```
prs <- predict(fit, newdata = ND, type = "lp", se.fit = TRUE)
ND$pred <- prs[[1]]
ND$se <- prs[[2]]
ND$lo <- ND$pred - 1.96 * ND$se
ND$up <- ND$pred + 1.96 * ND$se
```

4.3.5 Cox Model – Displaying Effects (cont'd)

R> Finally, we plot the result

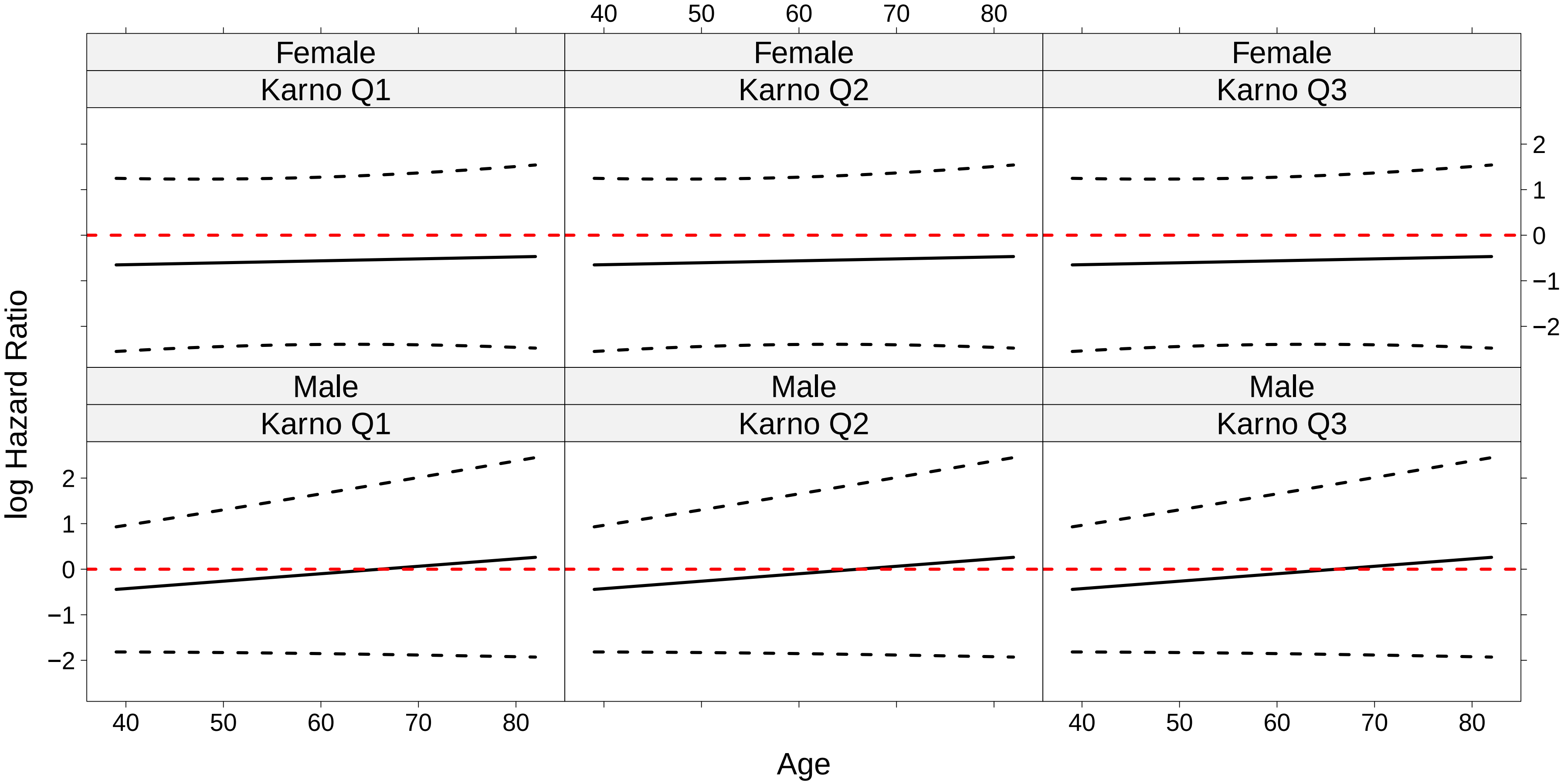
```
library(lattice)
xyplot(pred + lo + up ~ age | sex, data = ND,
    panel = function (...) {
        panel.xyplot(..., lty = c(1,2,2), col = "black",
            lwd = 2, type = "l")
        panel.abline(h = 0, lwd = 2, lty = 2, col = "red")
    },
xlab = "Age", ylab = "log Hazard Ratio")
```

4.3.5 Cox Model – Displaying Effects (cont'd)

- Note that effect plots can be even more flexible

- For example, we may be interested in how the risk for death changes
  - ▷ as baseline age increases
  - ▷ separately for males and females
  - ▷ as the Karnofsky score increases from its 1st quantile value (75), to the median value (80) to the 3rd quantile (90)

4.3.5 Cox Model – Displaying Effects (cont'd)

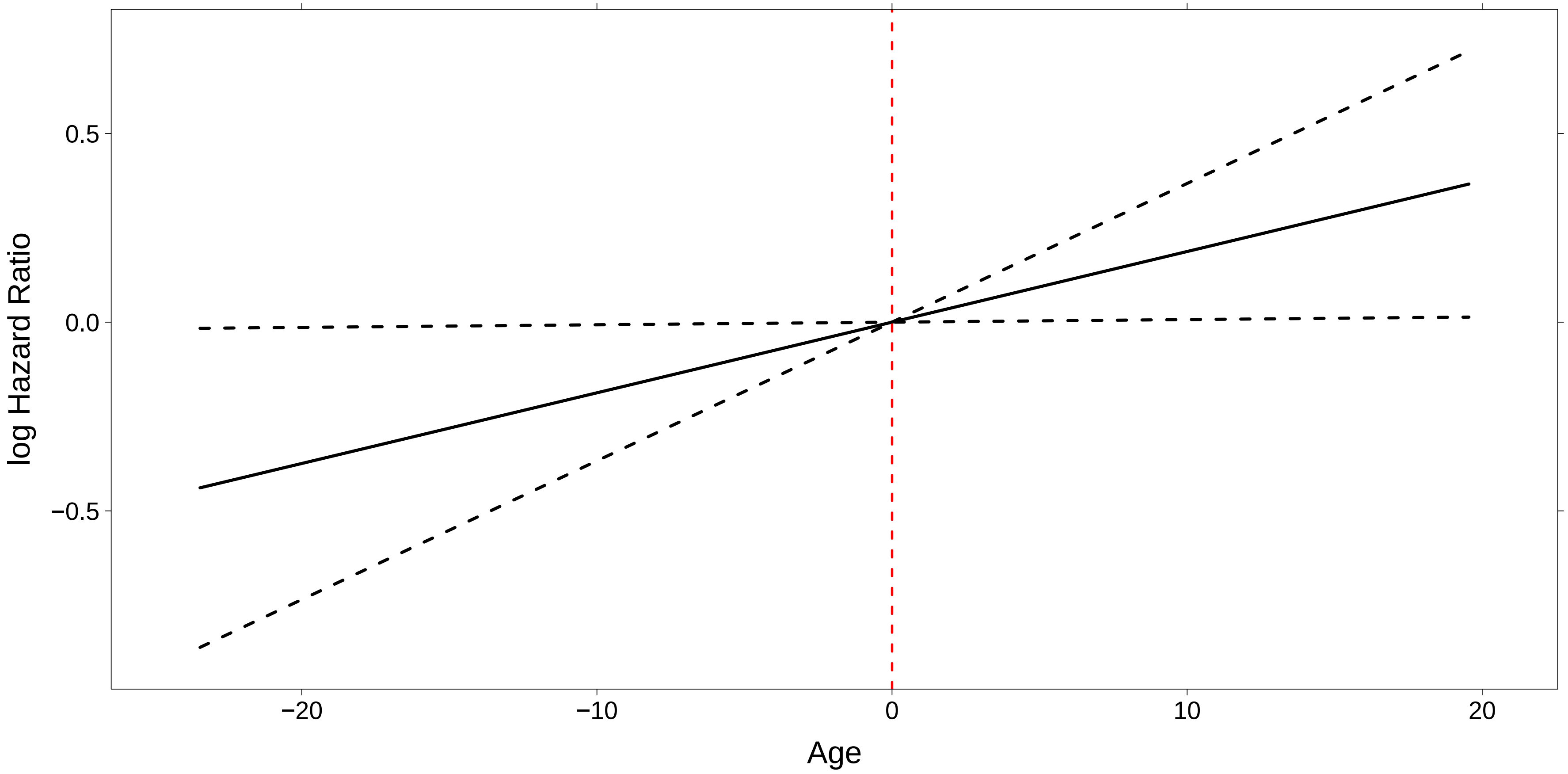


4.3.5 Cox Model – Displaying Effects (cont'd)

- A peculiarity in the Cox model arises when we compute the log hazard ratio for the reference level of continuous covariate
  - ▷ the standard error for the predicted by the Cox model value is effectively 0

- Example: for the Lung data we fit a Cox model in which we correct for centered age (i.e., $\text{age} - mean(\text{age})$), and we produce the corresponding effect plot

4.3.5 Cox Model – Displaying Effects (cont'd)



4.3.6 Cox Model – Hypothesis Testing

- Hypothesis testing for the Cox model follows the same principles as in AFT models

- Namely for testing the general hypothesis,

$$\begin{aligned} H_0 : \beta &= \beta_0 \\ H_a : \beta &\neq \beta_0 \end{aligned}$$

- <u>Remember:</u> β have the interpretation of log hazard ratio (see Section 4.3.4)

4.3.6 Cox Model – Hypothesis Testing (cont'd)

- We can use
  - ▷ Wald test
  - ▷ Likelihood Ratio test (LRT)
  - ▷ Score test

- Important (technical) difference: the Score and LRT statistics are now based on the partial likelihood and not on the full likelihood, as in AFT models

- **Special Case:** when we are interested in testing whether the hazard ratio for a categorical covariate (e.g., treatment) is 1, then
  - ▷ Score test of the Cox model exactly equivalent to the log-rank test

4.3.6 Cox Model – Hypothesis Testing (cont'd)

- Asymptotically (i.e., when the number of events is large) the three test statistics are equivalent and converge to the same value (and therefore p-value)

- In some cases we may observe differences

- Example: in the PBC data set we are interested to know whether gender has a significant impact on survival

4.3.6 Cox Model – Hypothesis Testing (cont'd)

	Statistic	df	p-value
Wald	8.87	1	0.0029
Score (log-rank)	9.18	1	0.0024
LRT	7.73	1	0.0054

- In all cases the result is significant; however, we observe some differences in the value of the statistics and therefore in the level of the p-value

- **Advice:** (as in AFT models) prefer to use the likelihood ratio test over the other two because it has better statistical properties

4.3.6 Cox Model – Hypothesis Testing (cont'd)

- $\boxed{\text{Example:}}$ in the PBC data set we account for
  - ▷ main effect of treatment
  - ▷ main effect of age at baseline
  - ▷ interaction effect between treatment and age

  and we are interested in the overall treatment effect

- In model terms

$$H_0: \quad h_i(t) = h_0(t) \exp\bigl(\beta_2 \texttt{Age}_i\bigr)$$

$$H_a: \quad h_i(t) = h_0(t) \exp\bigl(\beta_1 \texttt{Treat}_i + \beta_2 \texttt{Age}_i + \beta_3 \texttt{Treat:Age}_i\bigr)$$

4.3.6 Cox Model – Hypothesis Testing (cont'd)

- Understanding the models (reference level for `Treat` is 'placebo'):
  - ▷ Under H_0

$$h_i(t) \;=\; h_0(t) \exp\bigl(\beta_2 \texttt{Age}_i\bigr)$$

  - ▷ Under H_a

$$\begin{array}{lrcl} \text{placebo} & \Rightarrow\; h_i(t) & = & h_0(t) \exp\bigl(\beta_2 \texttt{Age}_i\bigr) \\ \text{D-penicillamine} & \Rightarrow\; h_i(t) & = & h_0(t) \exp\bigl({\color{blue}\beta_1} + (\beta_2 + {\color{blue}\beta_3}) \texttt{Age}_i\bigr) \end{array}$$

4.3.6 Cox Model – Hypothesis Testing (cont'd)

- In this example

$$\begin{array}{ll} H_0: & \beta_1 = \beta_3 = 0 \\ H_a: & \text{at least one is different from } 0 \end{array}$$

- Likelihood ratio test
  - ▷ log partial-likelihood under the reduced model $p\ell(\beta_1 = 0, \hat{\beta}_2, \beta_3 = 0) = -712.41$
  - ▷ log partial-likelihood under the alternative model $p\ell(\hat{\beta}_1, \hat{\beta}_2, \hat{\beta}_3) = -711.29$
  - ▷ parameters being tested $p = 2$
  - ▷ LRT $= -2 \times \{-712.41 - (-711.29)\} = 2.24$, $df = 2$, p-value $= 0.326$

4.3.6 Cox Model – Hypothesis Testing (cont'd)

R> As for AFT models, the LRT and the associated p-value for Cox models is computed by the anova() method. Again, this function accepts two arguments, namely the fitted Cox models under the null and alternative hypothesis

```
fit.null <- coxph(Surv(years, status2) ~ age, data = pbc2.id)
fit.alt <- coxph(Surv(years, status2) ~ drug*age, data = pbc2.id)

anova(fit.null, fit.alt)
```

4.3.7 Cox Model – Sample Size

- Usually, before starting a study we need to know how many subjects we should enroll such that we have a high chance to find a statistical significant difference ⇒ Sample size determination

- For instance, say we have two groups A and B and we want to detect a hazard ratio of magnitude $\exp(\beta)$ for a prespecified power level

$$h_A(t) = h_B(t) \exp(\beta)$$

- Remember: power is the probability that we will find a statistically significant difference between the two groups, **given** that this difference truly exists

4.3.7 Cox Model – Sample Size (cont'd)

- Under the Score test for the Cox model (which is equivalent to the log-rank test) we have the formula

$$\#\ \text{events} = \frac{(c_\alpha + z_{power})^2}{p(1-p)\beta^2}$$

  where

  - ▷ β is the log hazard ratio
  - ▷ p is the proportion of patients in the active treatment group
  - ▷ α is the Type I error, and c_α the critical value for the test
  - ▷ z_ν is the upper ν quantile of the standard normal distribution

- Note: the power depends on the number of events and **not** on the sample size

4.3.7 Cox Model – Sample Size (cont'd)

- Number of events required under equal allocation in the two groups, with two-sided $\alpha = 0.05$

	$\exp(\beta)$					
Power	1.10	1.20	1.40	1.60	1.80	2
50%	1692	462	136	70	44	32
70%	2718	743	218	112	71	51
80%	3456	944	277	142	91	65
90%	4627	1264	371	190	190	87

4.3.7 Cox Model – Sample Size (cont’d)

R> The following function computes the number of deaths based on the formula presented in p.215

```
deaths <- function (beta, p, power, alpha = 0.05) {
    # ’beta’ the log hazard ratio
    # ’p’ the proportion of subjects in the treatment group
    # ’power’ the desired level of power
    # ’alpha’ the Type I error
    q <- 1 - p
    z.power <- qnorm(power)
    ca <- abs(qnorm(alpha / 2))
    round((ca + z.power)^2 / (p * q * beta^2))
}
```

4.3.7 Cox Model – Sample Size (cont'd)

R> For instance, to compute the number of events required for a hazard ratio of 1.15, for two groups with the same number of patients on average and power of 90% we use:

```
deaths(log(1.15), 0.5, 0.9)
```

4.3.7 Cox Model – Sample Size (cont'd)

- Challenge: how many patients to enroll in order to obtain the required number of events
  - ▷ enrollment period & enrollment rate
  - ▷ study closer
  - ▷ survival probabilities for the control group

- How can we have a rough idea about the log hazard ratio β, in advance
  - ▷ for rare events we have $\exp(\beta) \approx F_A(t)/F_B(t)$ $\big(F(t)$ denotes the CDF; check Section 2.2$\big)$
  - ▷ that is, a treatment that halves the hazard ratio will approximately halve the proportion of events

4.3.8 Cox Model – Checking Assumptions

- The main motivation to introduce the semiparametric Cox model was to avoid the impact of a possibly wrong assumption for the distribution of the event times

- However, all statistical models make assumptions – in the Cox model we make no assumption for the distribution of T_i^* but we do make other assumptions:
  - ▷ linearity (to be discussed later – see Section 4.5.1)
  - ▷ additivity (to be discussed later – see Section 4.5.2)
  - ▷ **proportional hazards (PH)**

- If one or more of these assumptions are seriously violated, then the results we obtain from the Cox model may not be trustworthy!

4.3.8 Cox Model – Checking Assumptions (cont'd)

- In practice, PH means that the effect of a covariate in the risk for an event is **constant over time**

- Some times the PH assumption may not be reasonable, e.g.,
  - ▷ the new treatment requires a time period to start working ⇒ at the beginning of follow-up the risk for the treatment group is the same as in the control group, however we expect that later the risk for the treatment group will decrease
  - ▷ . . .

4.3.8 Cox Model – Checking Assumptions (cont'd)

- How can we test for PH? We distinguish between
  - ▷ categorical covariates with a small number of levels (e.g., treatment, gender, etc.)
  - ▷ continuous covariates (e.g., age, weight, etc.)

- For categorical covariates we can compare appropriately transformed Kaplan-Meier estimates of the survival functions of the different groups

4.3.8 Cox Model – Checking Assumptions (cont'd)

- In particular, under the PH assumption we have (see pp. 175–177)

$$\begin{aligned}
S_A(t) &= \bigl\{S_B(t)\bigr\}^{\exp(\beta X_i)} \;\Rightarrow \\
\log S_A(t) &= \exp(\beta X_i) \log S_B(t) \;\Rightarrow \\
-\log S_A(t) &= \exp(\beta X_i)\{-\log S_B(t)\} \;\Rightarrow \\
\log\{-\log S_A(t)\} &= \beta X_i + \log\{-\log S_B(t)\}
\end{aligned}$$

- Therefore, if PH holds and we plot the Kaplan-Meier estimates of $\log\{-\log S_A(t)\}$ and $\log\{-\log S_B(t)\}$, we expect two approximately parallel lines

4.3.8 Cox Model – Checking Assumptions (cont'd)

- Example: for the PBC data set we want to test for a impact of cholesterol on the hazard for event – we fit the Cox model

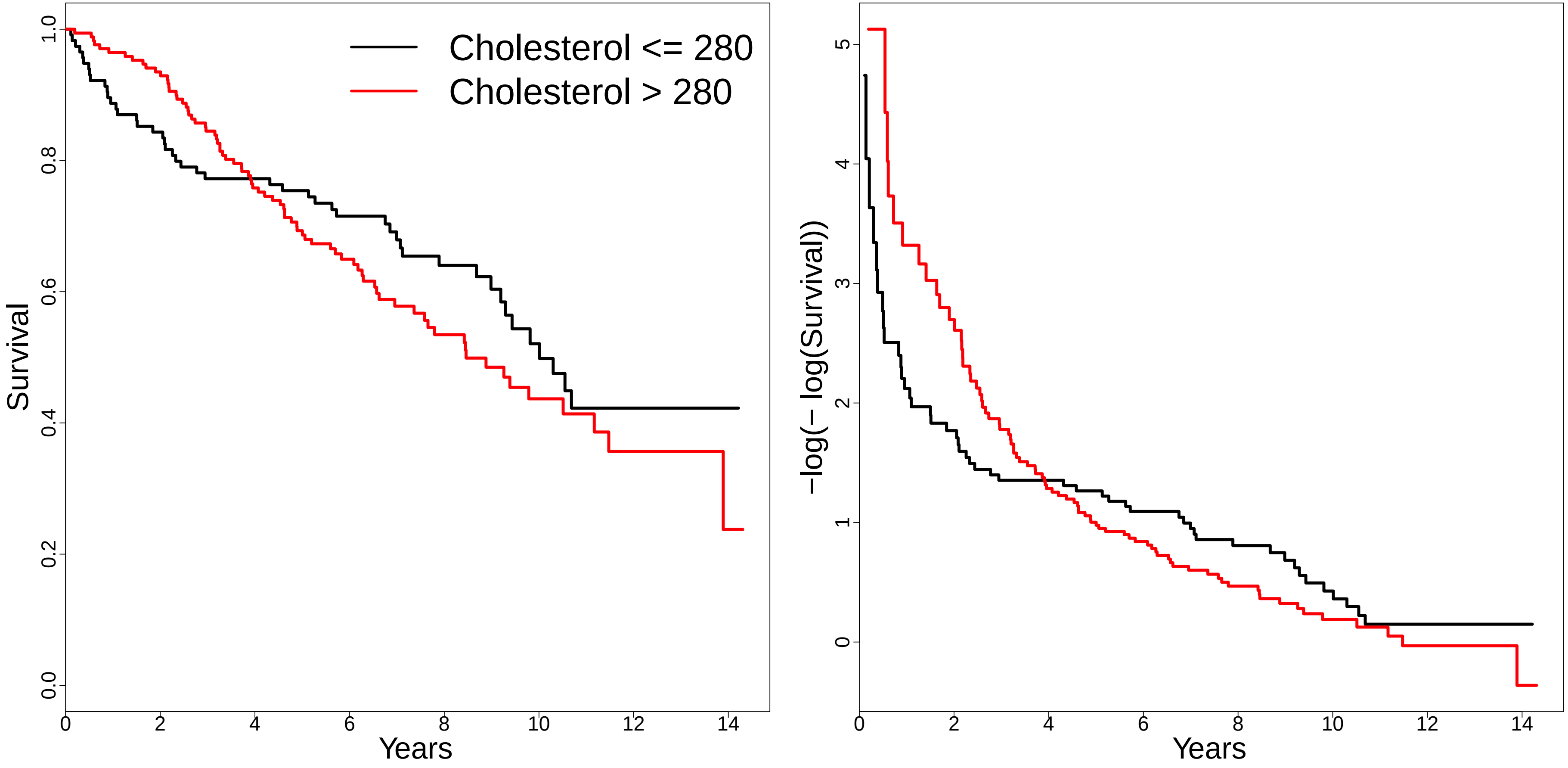
$$h_i(t) \; = \; h_0(t) \exp(\beta X_i)$$

where $X_i = 1$ for patients who had serum cholesterol greater that 280, and $X_i = 0$ otherwise

- We obtain HR $= 1.226$, p-value $= 0.267$

- We test whether the assumption holds graphically by comparing the Kaplan-Meier estimates of $\log\{-\log S(t)\}$ for the two groups

4.3.8 Cox Model – Checking Assumptions (cont'd)



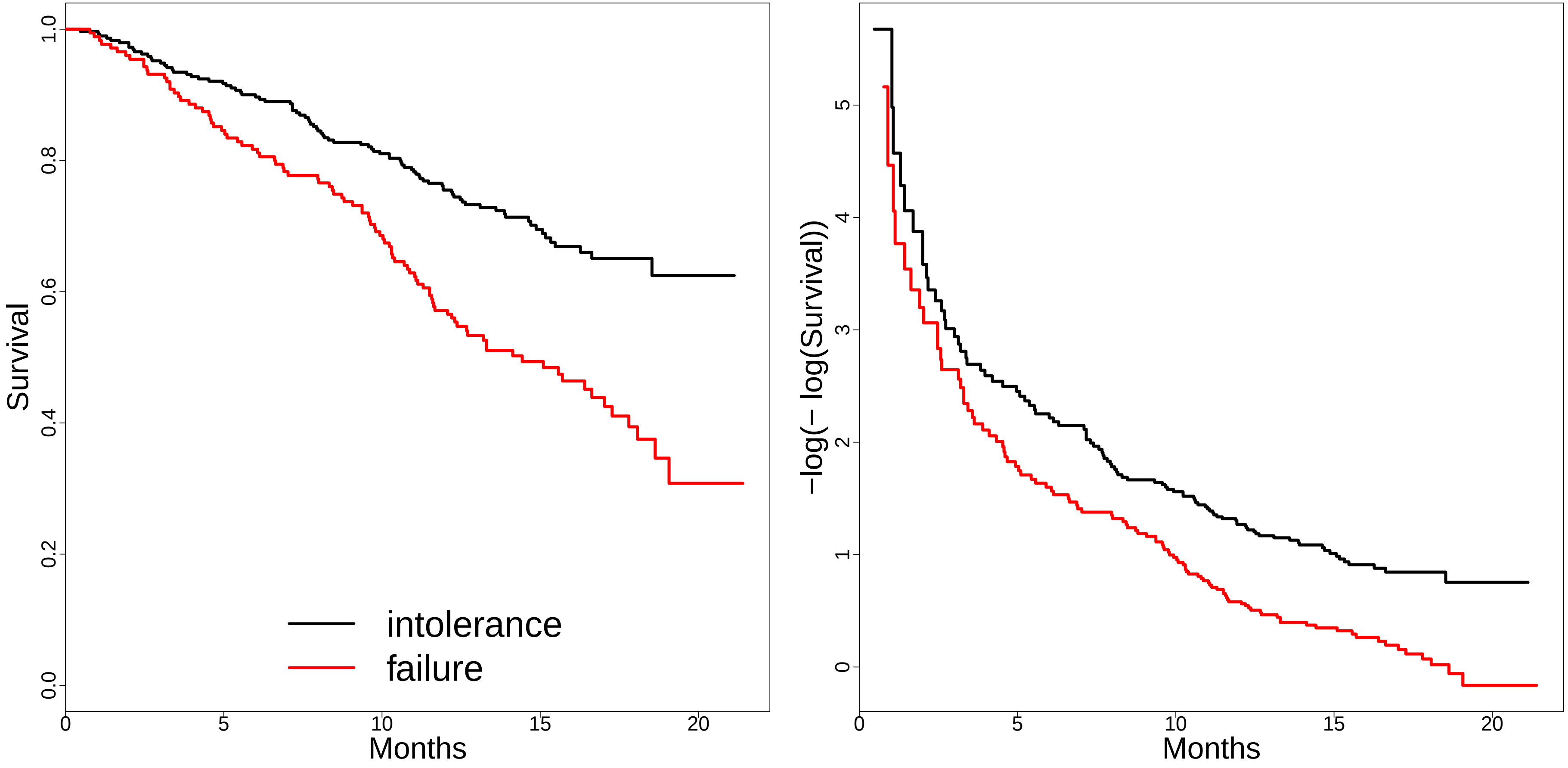
4.3.8 Cox Model – Checking Assumptions (cont'd)

- We observe that the PH assumption seems questionable

- Note: PH may not hold even if the survival curves do not cross over

- Example: for the AIDS data set we want to test for differences in the risk for death between the different AZT groups
  - ▷ AZT intolerance
  - ▷ AZT failure

4.3.8 Cox Model – Checking Assumptions (cont'd)



4.3.8 Cox Model – Checking Assumptions (cont'd)

- The survival curves do not cross but the $\log\{-\log S(t)\}$ lines are not parallel

4.3.8 Cox Model – Checking Assumptions (cont'd)

R> To transform the Kaplan-Meier estimate of $S(t)$ to $-\log\{-\log S(t)\}$ we use the fun argument of the plot() method

```
fit <- survfit(Surv(years, status2) ~ factor(serChol > 280),
    data = pbc2.id)

par(mfrow = c(1, 2))

plot(fit, mark.time = FALSE, xlab = "Years", ylab = "Survival",
    col = 1:2)

plot(fit, mark.time = FALSE, fun = function (s) -log(-log(s)),
    xlab = "Years", ylab = "-log(- log(Survival))", col = 1:2)
```

4.3.8 Cox Model – Checking Assumptions (cont'd)

- When the covariate has a few levels, the Kaplan-Meier can be easily used to test the PH assumption

- However, when the covariate has many levels or is continuous, the Kaplan-Meier plot is not useful for discerning either the fact or the pattern of non-proportional hazards

  ▷ we could collapse some levels of the categorical covariate or discritize the continuous covariate but there is no objective way to do this

- To deal with this we will (hypothetically for now) consider an extension of the Cox model

4.3.8 Cox Model – Checking Assumptions (cont'd)

- Cox model with a *time-dependent coefficient*

$$h_i(t) \;=\; h_0(t)\exp\{X_i\beta(t)\}$$

- The impact of covariate X on the hazard *varies* with time

- Grambsch and Therneau (Biometrika, 1994) have shown that, if $\hat{\beta}$ is the estimated coefficient from the ordinary (time-independent) Cox model, then

$$\beta(t) \approx \hat{\beta} + E\{s^*(t)\}$$

where $s^*(t)$ is the scaled Schoenfeld residual

4.3.8 Cox Model – Checking Assumptions (cont'd)

- The formula and rationale behind the scaled Schoenfeld residuals is rather technical
  - ▷ we will not give them here (see Therneau & Grambsch (2000) for more info)

- Plotting scaled Schoenfeld residuals against time or suitable transformation of time, reveals violations of the PH assumption

- An additional advantage of the scaled Schoenfeld residuals is that they can be used to test PH both graphically and via a formal statistical test

4.3.8 Cox Model – Checking Assumptions (cont'd)

- $\boxed{\text{Example:}}$ for the Lung data we are interested in the relationship between the Karnofsky performance score and the risk for death

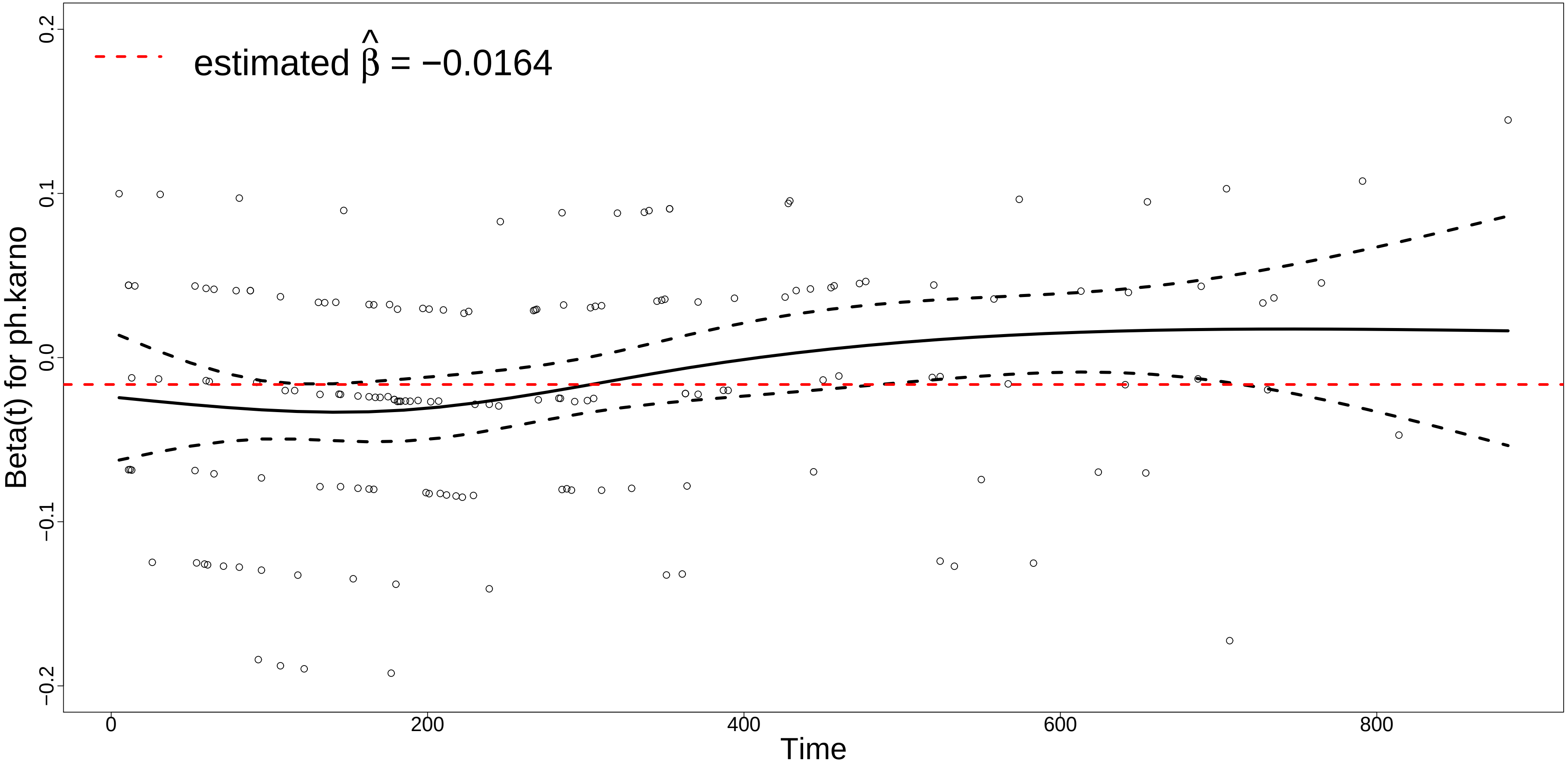
- The ordinary Cox model gives a significant result:

  ▷ HR $= 0.984$, p-value $= 0.005$

- We check the PH assumption using the scaled Schoenfeld residuals

  ▷ we plot the approximated $\beta(t)$ for Karnofsky score against time

4.3.8 Cox Model – Checking Assumptions (cont'd)



4.3.8 Cox Model – Checking Assumptions (cont'd)

- We have some indications that the effect is not constant in time

- We can formally test for non-proportionality

  ▷ correlation between scaled Schoenfeld residuals and time

$$\begin{array}{ll} H_0: & \rho = 0 \text{ (correlation 0 means PH holds)} \\ H_a: & \rho \neq 0 \end{array}$$

  ▷ for the Karnofsky score: $\hat{\rho} = 0.234$, $\chi^2 = 8.08$, $df = 1$, p-value $= 0.0045$

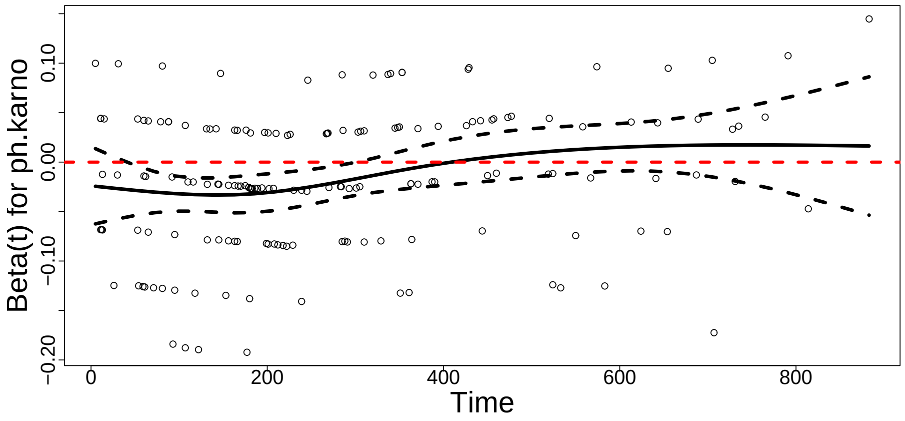
4.3.8 Cox Model – Checking Assumptions (cont'd)

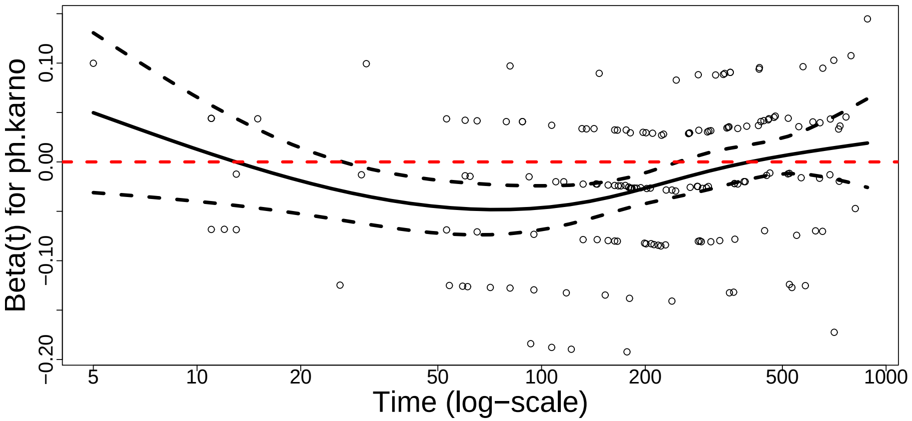
- So far we have used $\beta(t)$ vs t, but we can also try other time transformations
  - ▷ $\beta(t)$ vs $\log(t)$
  - ▷ $\beta(t)$ vs Kaplan-Meier transform of t

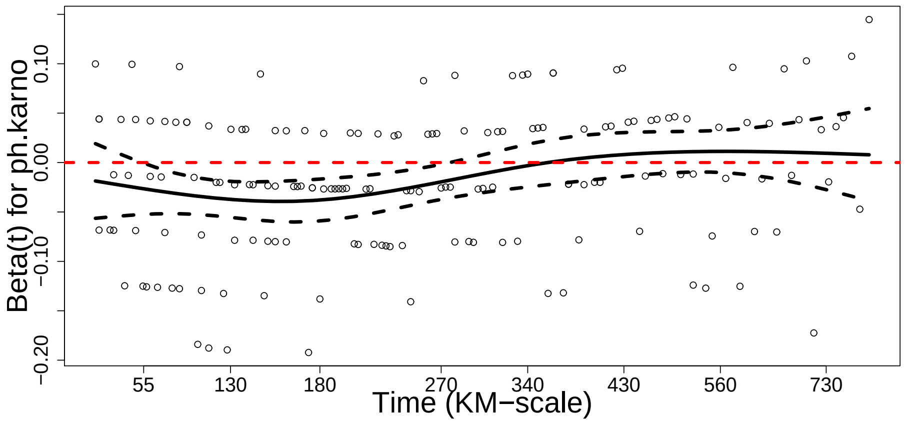
- The last option is more difficult to interpret in practice,
  - ▷ (it is calculated as the inverse CDF, with CDF$(t) = 1 - \hat{S}_{KM}(t)$)
  - ▷ and it aims to spread the residuals evenly in time, avoiding problems with outliers

- We illustrate for the previous example

4.3.8 Cox Model – Checking Assumptions (cont'd)







4.3.8 Cox Model – Checking Assumptions (cont'd)

- The spread of the residuals is not the same for all time scales
  - ▷ for t more residuals to the left – for $\log(t)$ more to the right – for $KM(t)$ more evenly spaced

- This can have a (substantial) effect on the resulting inferences for non-proportionality

	ρ	Statistic	p-value
t	0.234	8.08	0.0045
$\log(t)$	0.139	2.85	0.0913
$KM(t)$	0.232	7.95	0.0048

4.3.8 Cox Model – Checking Assumptions (cont'd)

- The optimal transformation depends on the specific data set at hand

- General guideline: a transformation that spreads the residuals evenly over time such that the plot and the test are not affected by outliers

4.3.8 Cox Model – Checking Assumptions (cont'd)

R> The test for non-proportional hazards based on the scaled Schoenfeld residuals is calculated by function `cox.zph()` – argument `transform` specifies the time scale

```
fit.ph <- coxph(Surv(time, status) ~ ph.karno, data = lung)

zph1 <- cox.zph(fit.ph, transform = "identity")
zph2 <- cox.zph(fit.ph, transform = "log")
zph3 <- cox.zph(fit.ph, transform = "km")

zph1
zph2
zph3
```

4.3.8 Cox Model – Checking Assumptions (cont'd)

R> The `plot()` method is used to produce the plots of $\beta(t)$ versus the selected time scales

```
par(mfrow = c(2, 2))

plot(zph1)
abline(h = 0, lty = 2, lwd = 2, col = "red")

plot(zph2)
abline(h = 0, lty = 2, lwd = 2, col = "red")

plot(zph3)
abline(h = 0, lty = 2, lwd = 2, col = "red")
```

4.3.8 Cox Model – Checking Assumptions (cont'd)

- What if the Proportional Hazards assumption is violated?
  - ▷ does it really matter? How serious is the non-proportionality (see p. 234)

- If proportionality is not seriously affected, then what we will obtain is the average Hazard Ratio, averaged over the event times

4.3.8 Cox Model – Checking Assumptions (cont'd)

- If non-proportionality is large, then
  - ▷ incorporate covariates with nonproportional effects as stratification factors into the model (see Section 5.2)
  - ▷ partition the time axis: the PH assumption may hold over short time periods
    - * caveat: less efficient
  - ▷ model non-proportionality using time-dependent covariates (see Section 5.3)
  - ▷ use nonproportional hazards models, such as AFT models (see Section 4.2)
    - * caveat: if PH does hold for some covariates, the AFT model will assume that it does not

4.4 Parametric PH Models

- The Cox model is a semiparametric proportional hazards model
  - ▷ we have made no assumptions for the baseline hazard

- However, we also have parametric proportional hazard models
  - ▷ we assume that baseline hazard has a specific parametric form

- If we assume that time-to-event T_i^* follows a Weibull distribution then we obtain the model

4.4 Parametric PH Models (cont'd)

$$h_i(t) \; = \; h_0(t) \exp\bigl(\beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_p X_{ip}\bigr)$$

where

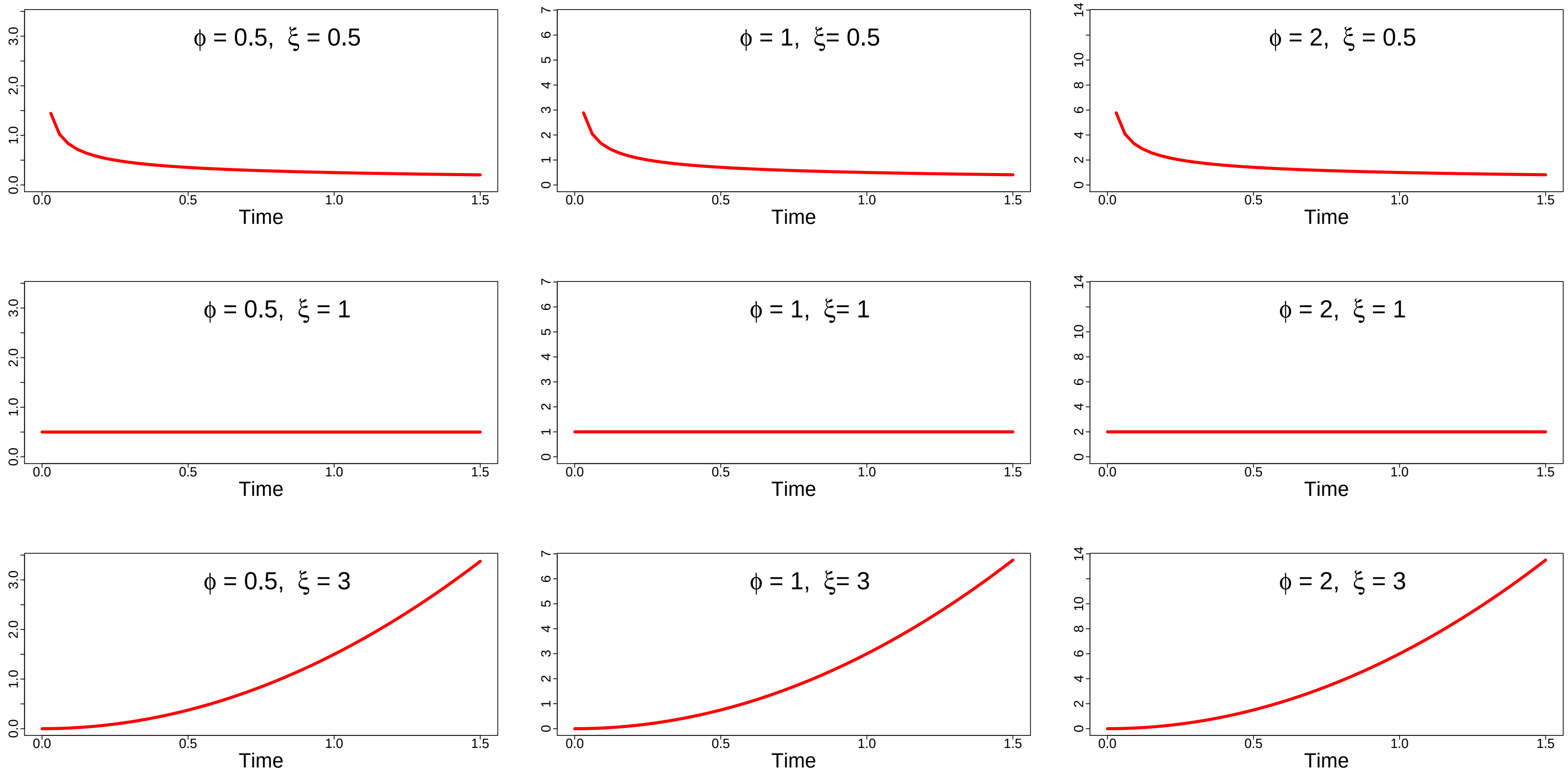
$$h_0(t) \; = \; \phi \xi t^{\xi - 1}$$

- The interpretation of the β parameters is exactly the same as in the Cox model

- The other parameters control the shape and scale of the distribution of T_i^*

4.4 Parametric PH Models (cont’d)

Some Weibull Hazard Functions



4.4 Parametric PH Models (cont'd)

- Example: we compare the Cox and Weibull models for the Stanford data where we control for the effects of age and t5 mismatch score

	Cox	Weibull
	value (std. err.)	value (std. err.)
age	0.0296 (0.0114)	0.0315 (0.0117)
t5	0.1704 (0.1833)	0.1738 (0.1823)

- We observe small differences between the two models
  - ▷ for this data set the Weibull assumption seems reasonable – this however is not always the case

4.4 Parametric PH Models (cont'd)

- The Weibull model (and its special case the Exponential model) is the only model that has both an AFT and PH formulation

$$\begin{aligned} \log T_i^* &= \beta_0 + \beta_1^{aft} X_i + \sigma \varepsilon_i \;\Leftrightarrow \\ h_i(t) &= \phi \xi t^{\xi - 1} \exp(\beta_1^{ph} X_i) \end{aligned}$$

with the following correspondence

$$\phi = \exp(-\beta_0/\sigma), \qquad \beta_1^{ph} = -\beta_1^{aft}/\sigma, \qquad \xi = 1/\sigma$$

4.4 Parametric PH Models (cont'd)

R> Using the previously defined relations we can transform the AFT estimated regression coefficients to PH coefficients for the Weibull model

```
fit.weib <- survreg(Surv(time, status) ~ age + t5,
    data = stanford2)

betasPH <- - coef(fit.weib) / fit.weib$scale
```

4.4 Parametric PH Models (cont’d)

R> For the calculation of standard errors of the PH estimated coefficients, we need to apply the Delta method

```
# we obtain covariances for the logarithm of the scale parameter
v.betas <- vcov(fit.weib)
transf <- list(~ -x1/exp(x4), ~ -x2/exp(x4), ~ -x3/exp(x4))
library(msm) # load package ’msm’
ses <- deltamethod(transf, c(coef(fit.weib), log(fit.weib$scale)),
    v.betas)

# We compare with the Cox model
fit.cph <- coxph(Surv(time, status) ~ age + t5, data = stanford2)
summary(fit.cph)
cbind("coef" = betasPH, "se(coef)" = ses)
```

4.5 Modelling Strategies

- General regression modelling strategies

- How to develop statistical models, with aim to
  - ▷ effect estimation
  - ▷ hypothesis testing
  - ▷ prediction

- Most of these guidelines are not only applicable to regression models for time-to-event data but also to other types of models (e.g., linear regression, logistic regression, etc.)

4.5.1 Modelling Strategies – Linearity

- When modelling continuous covariates it is customary to assume that such covariates affect linearly the log hazard ratio (in PH models) or the average log failure time (in AFT models)

- However, this assumption is very restrictive and in many real applications it may not hold
  - ▷ increasing age from 20y to 25y does not increase the risk in the same amount as increasing age from 60y to 65y

- Wrongly assuming linearity may affect the resulting inference for such covariates as well as the predictive ability of the model

4.5.1 Modelling Strategies – Linearity (cont'd)

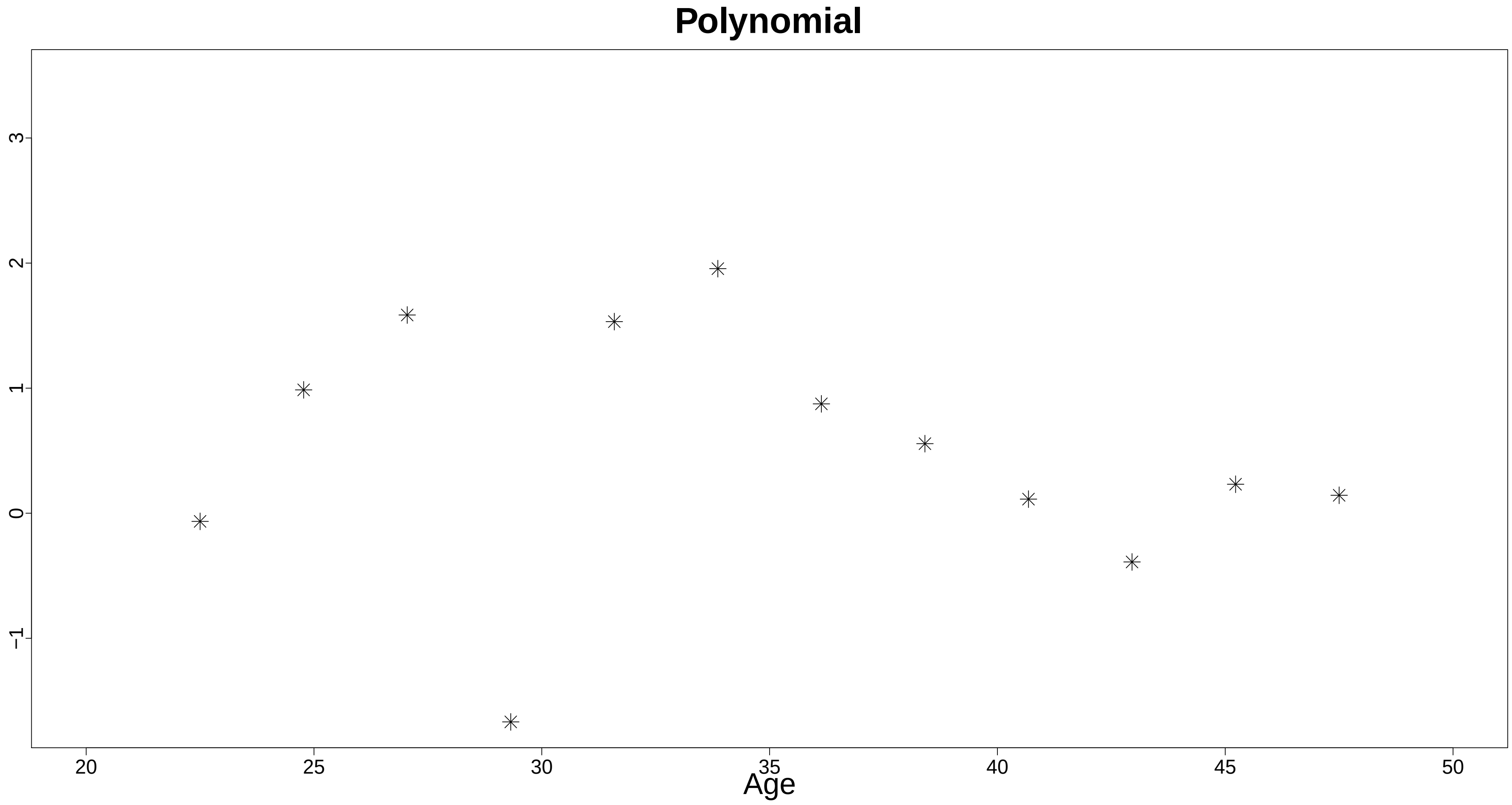
- Therefore, it is highly advisable not to restrict a priori the effects of continuous predictors to be linear and let the data tell you the true story

- The easiest way to relax linearity is to assume polynomial effects

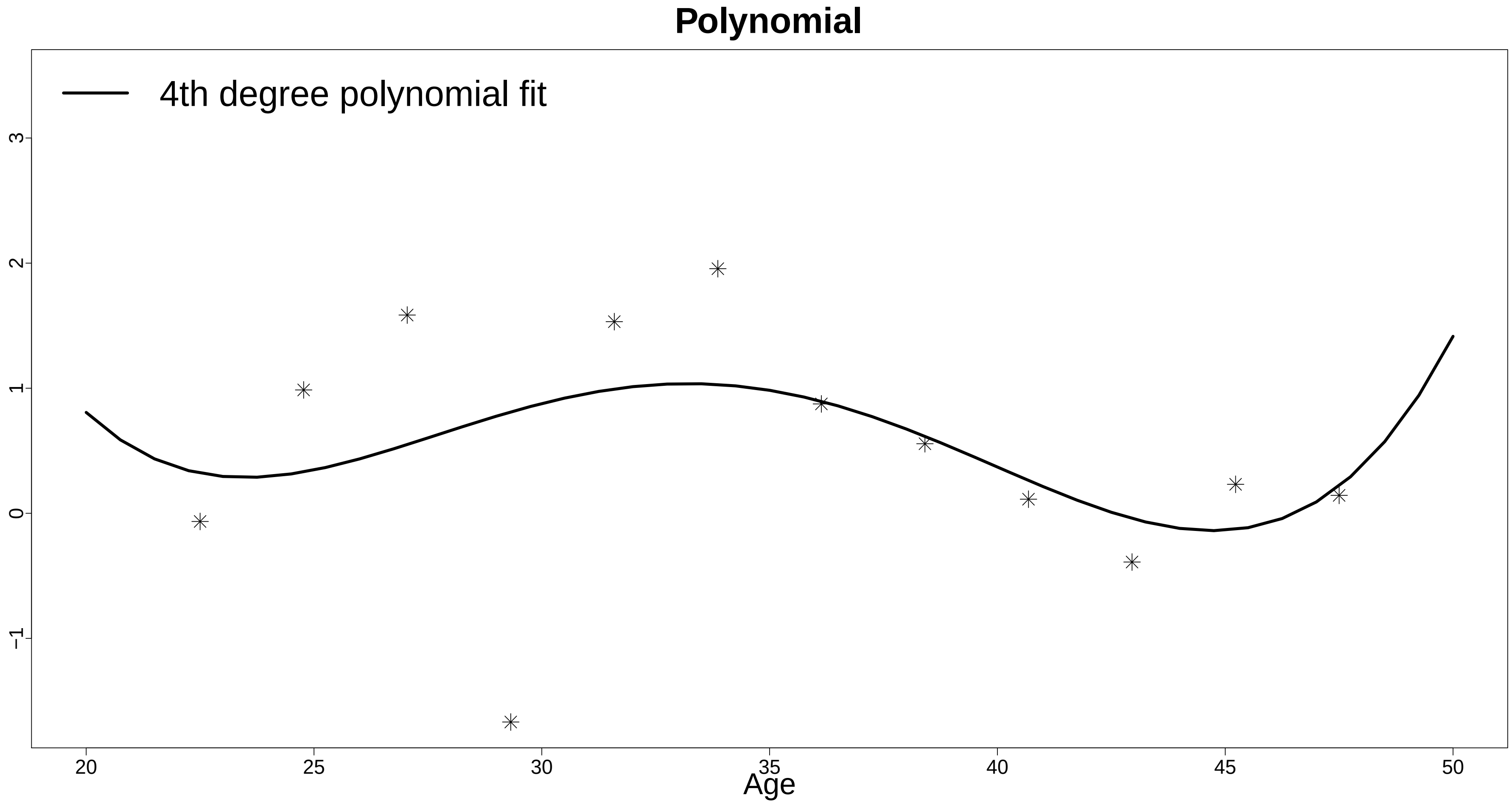
$$\beta_0 + \beta_1 X_i + \beta_2 X_i^2 + \beta_3 X_i^3 + \ldots$$

- However, polynomials have some disadvantages
  - ▷ they are not local ⇒ changing one data point will affect the overall fit
  - ▷ numerically ill-conditioned (however, not too worrisome with modern software)

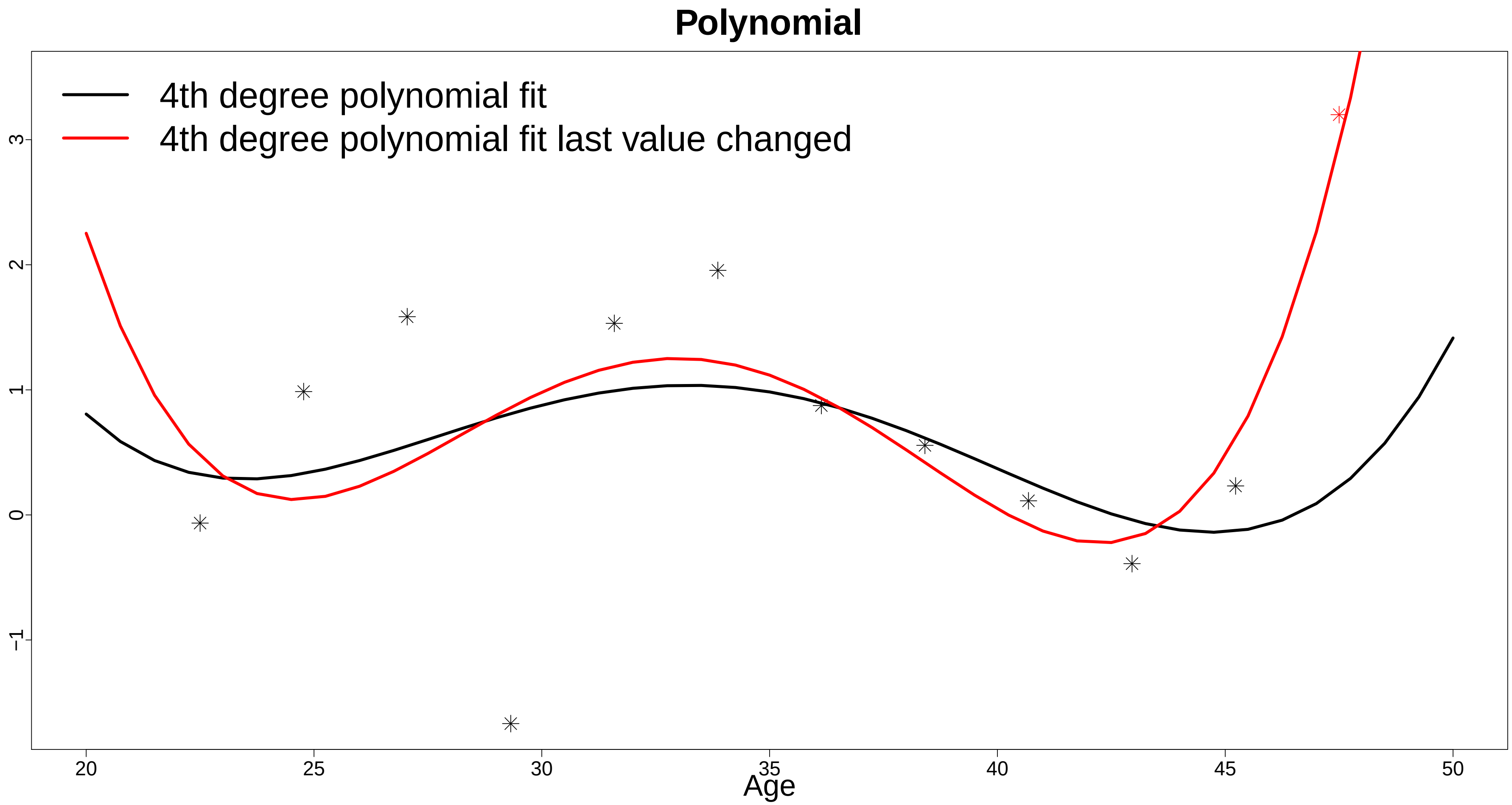
4.5.1 Modelling Strategies – Linearity (cont'd)



4.5.1 Modelling Strategies – Linearity (cont'd)



4.5.1 Modelling Strategies – Linearity (cont'd)

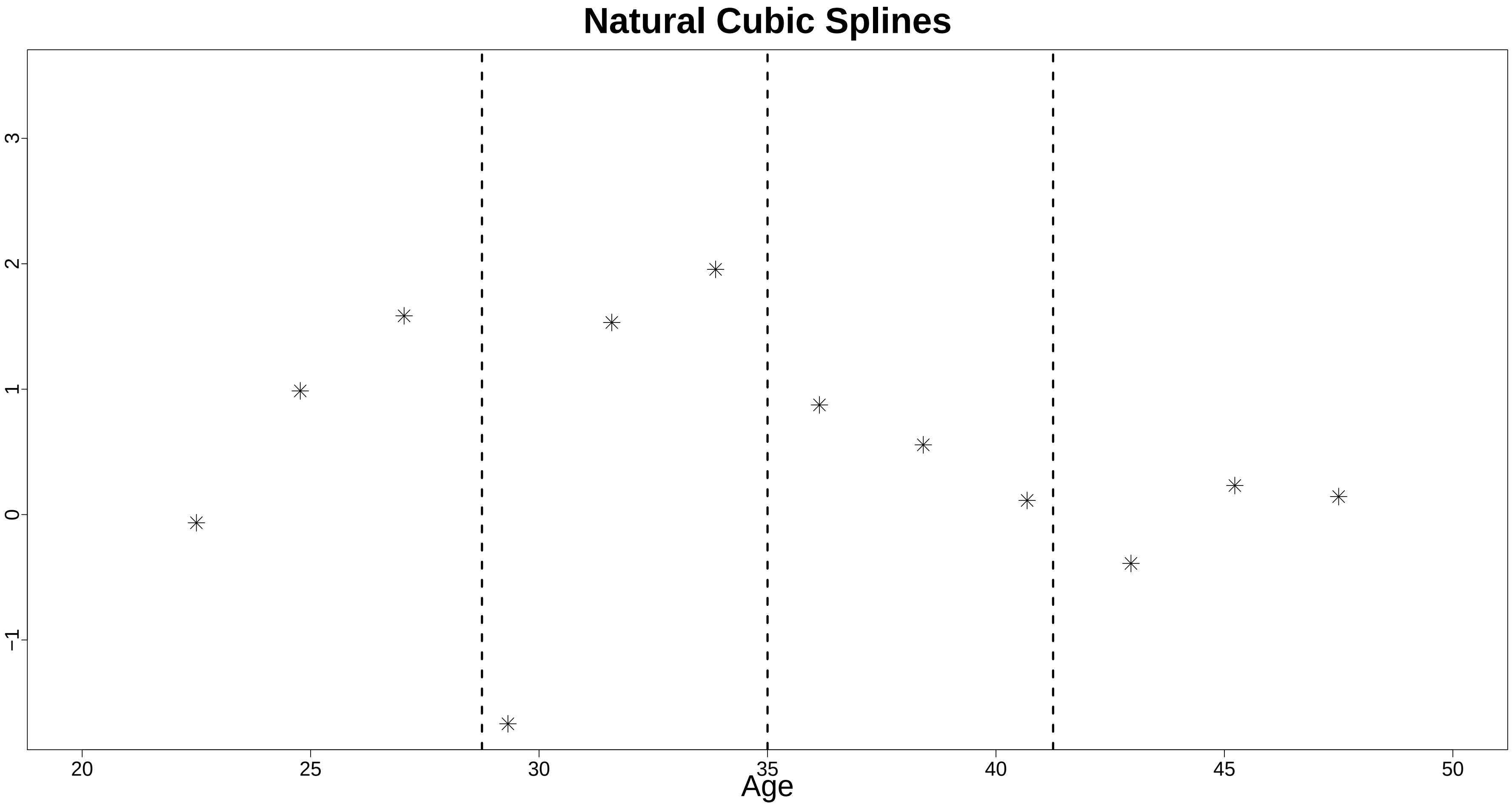


4.5.1 Modelling Strategies – Linearity (cont'd)

- An alternative approach to relax the linearity assumption of continuous predictors is to use regression splines

- Idea behind regression splines: use polynomials but locally
  - ▷ split the range of values of the continuous predictor into subintervals using a series of knots
  - ▷ within each subinterval assume that the effect of the predictor is nonlinear and can be approximated by a cubic polynomial
  - ▷ put extra smoothness assumptions, i.e., the cubic polynomial fits between neighboring subintervals must be connected

4.5.1 Modelling Strategies – Linearity (cont'd)

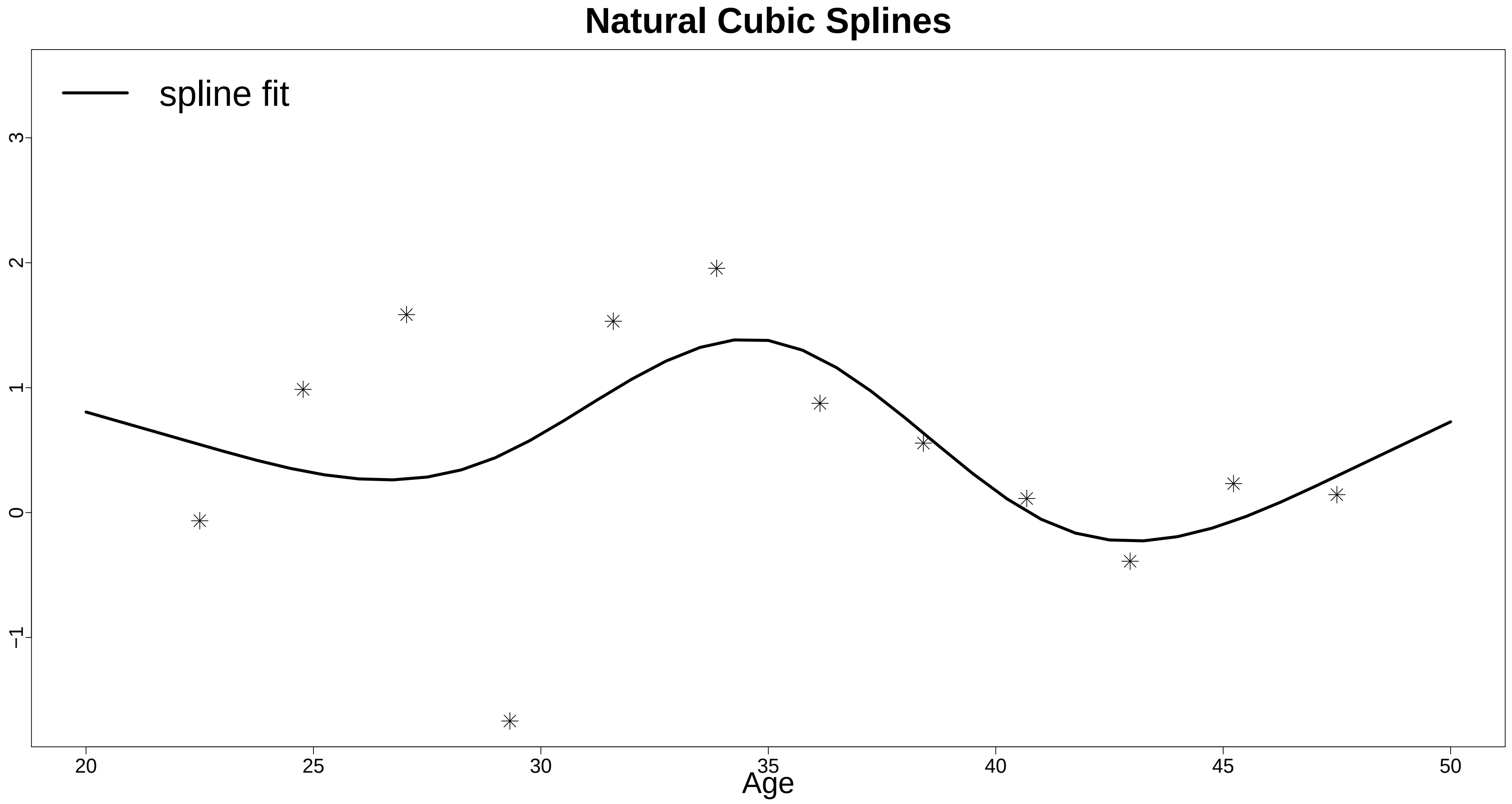


4.5.1 Modelling Strategies – Linearity (cont'd)

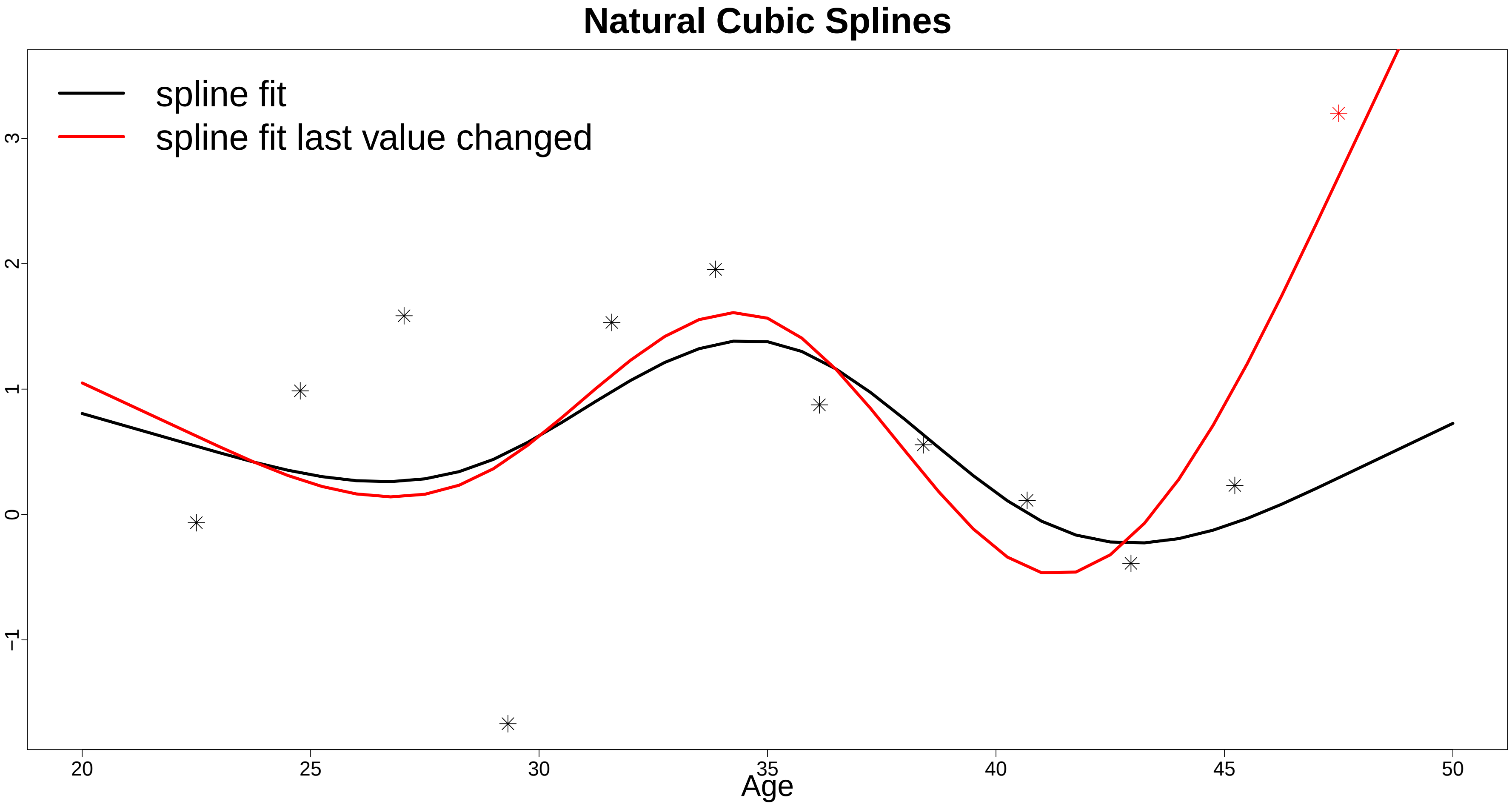
- There are several types of regression splines available
  - ▷ advisable to use natural cubic splines, which assume linearity outside the boundary knots – better statistical properties

- Other approaches (we are not going to discuss them here)
  - ▷ penalized splines
  - ▷ local regression
  - ▷ wavelets
  - ▷ . . .

4.5.1 Modelling Strategies – Linearity (cont'd)



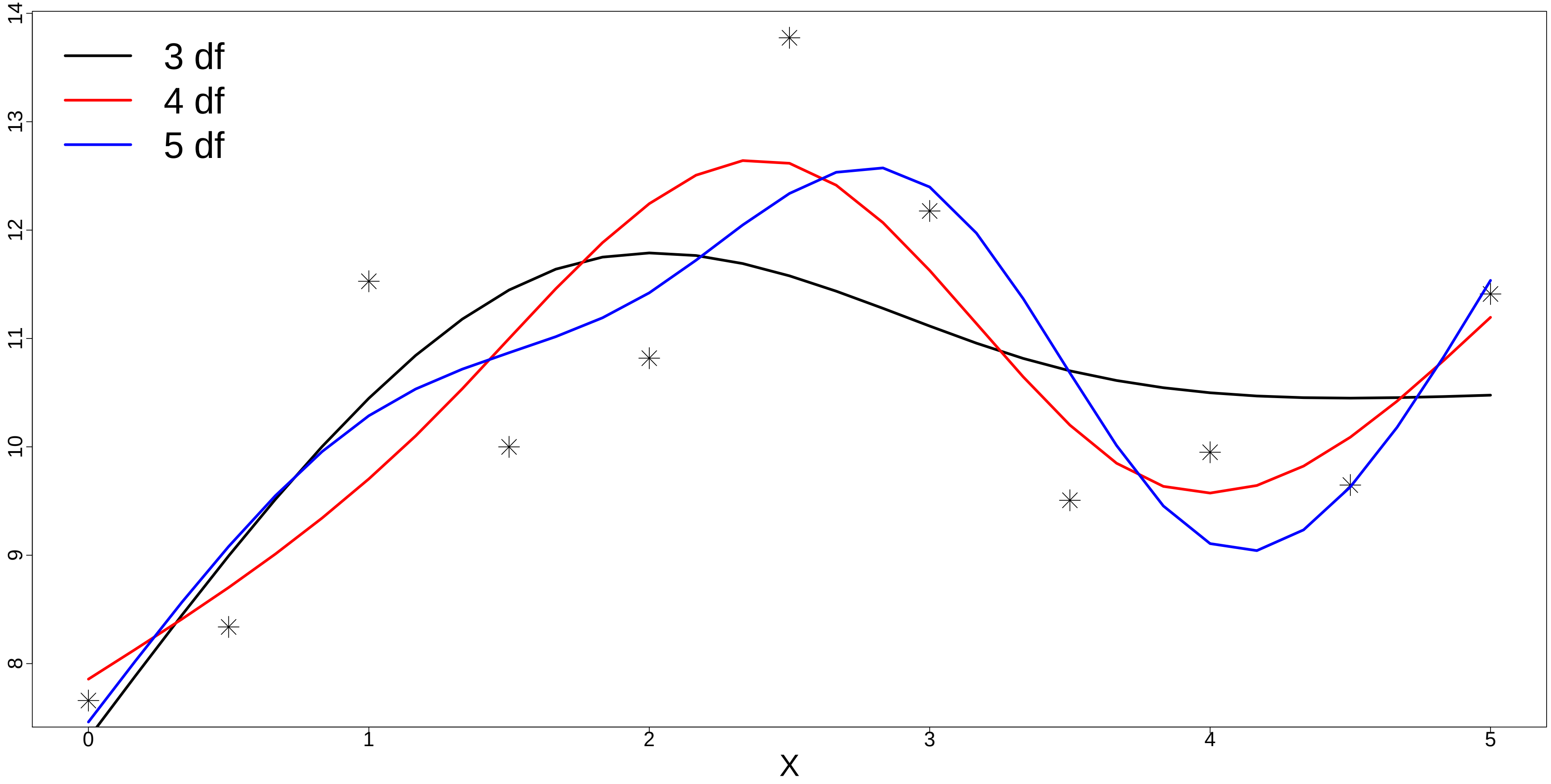
4.5.1 Modelling Strategies – Linearity (cont’d)



4.5.1 Modelling Strategies – Linearity (cont'd)

- As also in the case of the polynomials, we can tune the degree of nonlinearity by specifying the degrees of freedom for the spline
  - ▷ increasing the degrees of freedom results in more flexible modelling
  - ▷ bias-variance tradeoff

4.5.1 Modelling Strategies – Linearity (cont'd)

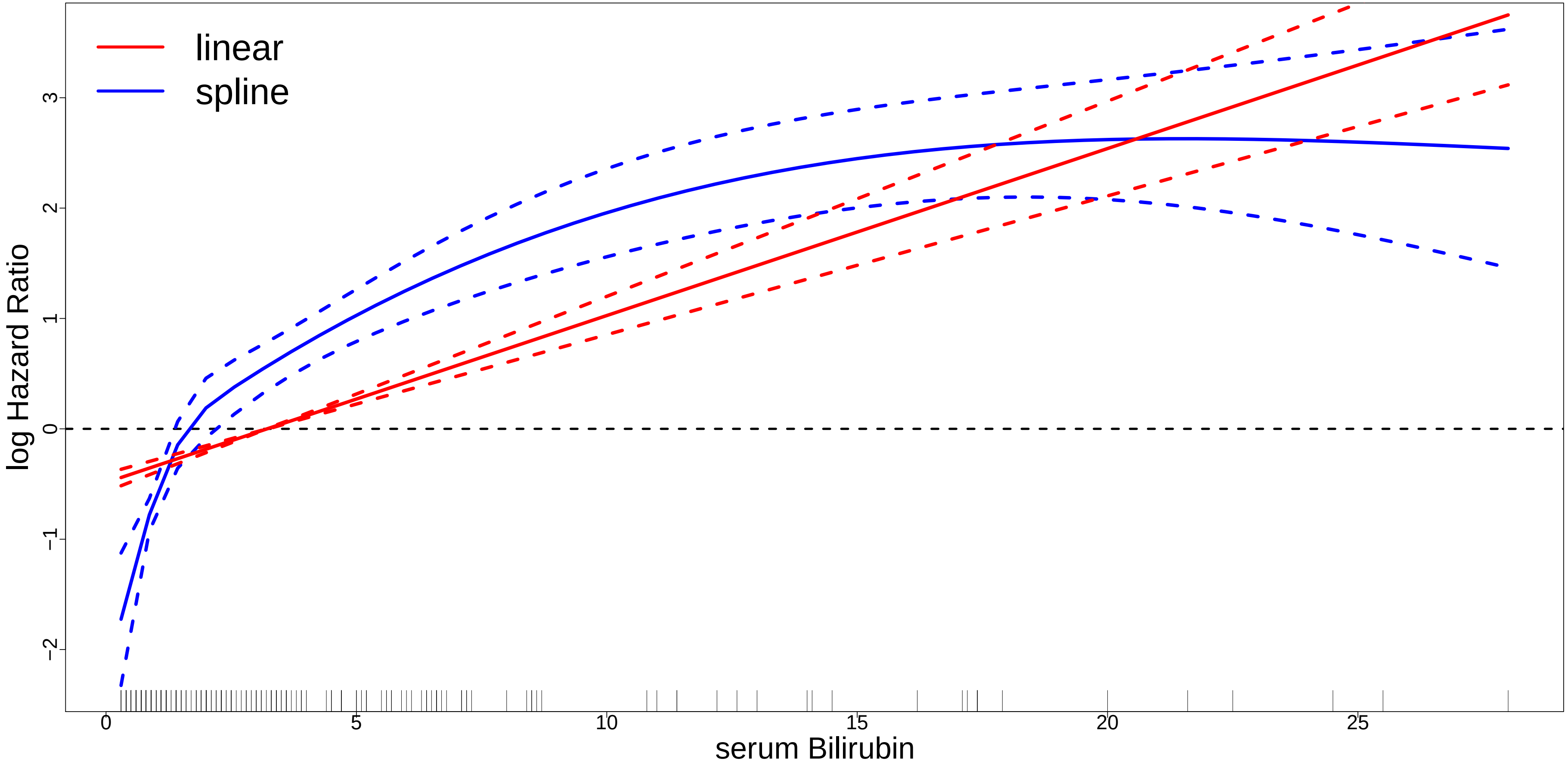


4.5.1 Modelling Strategies – Linearity (cont'd)

- $\boxed{\text{Example:}}$ for the PBC data we are interested in the effect of serum Bilirubin levels in the hazard for death
  - ▷ we will fit two Cox models,
  - ▷ one where the effect of serum Bilirubin is restricted to be linear, and
  - ▷ one where the effect of serum Bilirubin is allowed to be nonlinear

- Note: these are still PH models – relaxing linearity it does not mean that we allow the effect to change in time

4.5.1 Modelling Strategies – Linearity (cont'd)



4.5.1 Modelling Strategies – Linearity (cont’d)

R> Polynomial and spline nonlinear effects can be easily specified in both AFT and Cox models, within the `formula` argument

R> Orthogonal polynomials are defined using function `poly()`

```
# 3rd degree polynomial for serum Bilirubin
coxph(Surv(years, status2) ~ poly(serBilir, 3), data = pbc2.id)
```

(Orthogonal polynomials provide the same fit as the standard polynomials but with better numerical properties)

4.5.1 Modelling Strategies – Linearity (cont'd)

R> Natural cubic splines are defined using function `ns()` of package splines

```
# natural cubic splines for serum Bilirubin with 3 df
coxph(Surv(years, status2) ~ ns(serBilir, 3), data = pbc2.id)
```

4.5.2 Modelling Strategies – Additivity

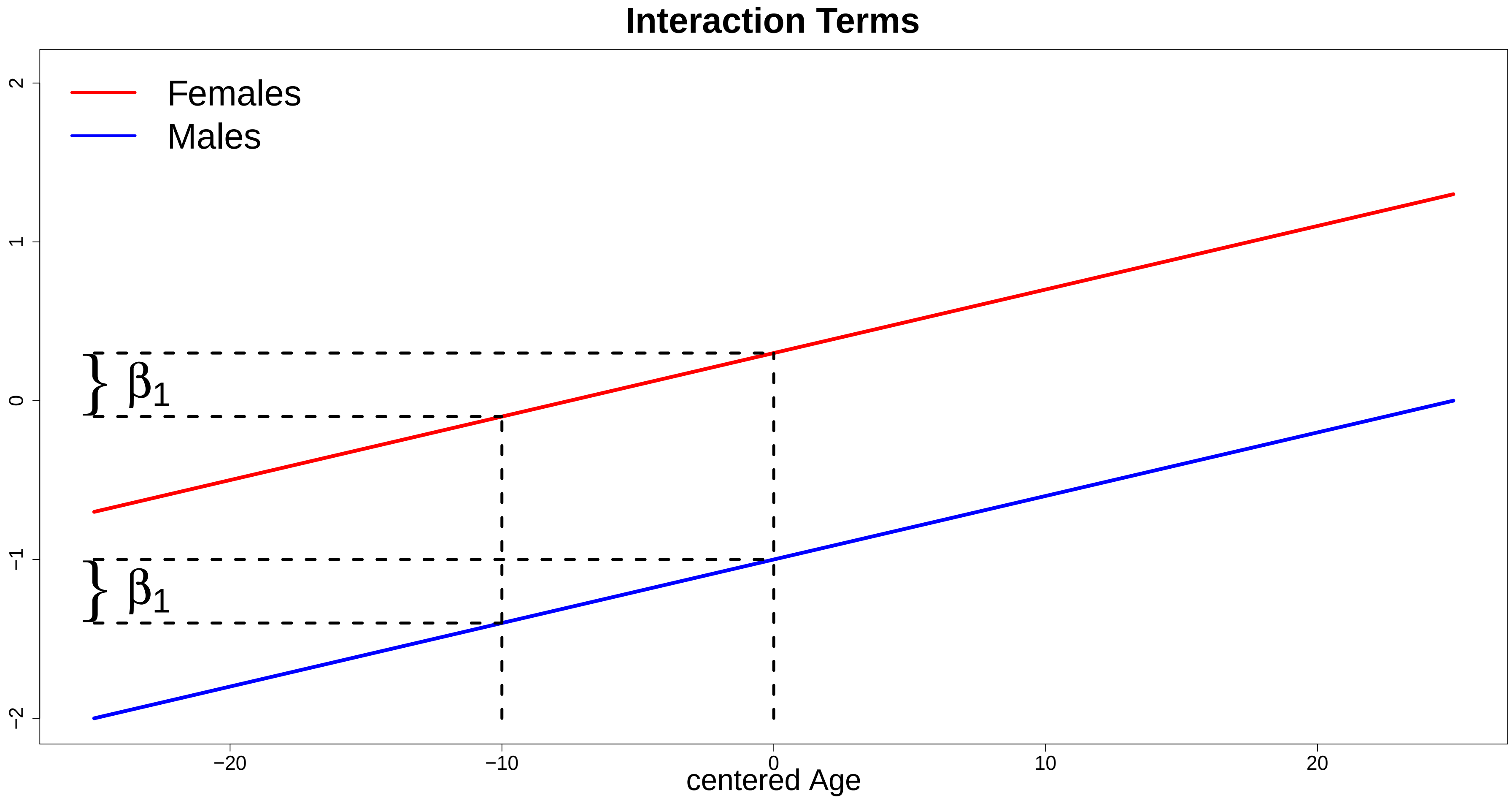
- The additivity assumption can be relaxed by considering meaningful interaction effects

$$\text{Model I} \quad h_i(t) \;=\; h_0(t) \exp\big(\beta_1 \texttt{Age}_i + \beta_2 \texttt{Sex}_i\big)$$

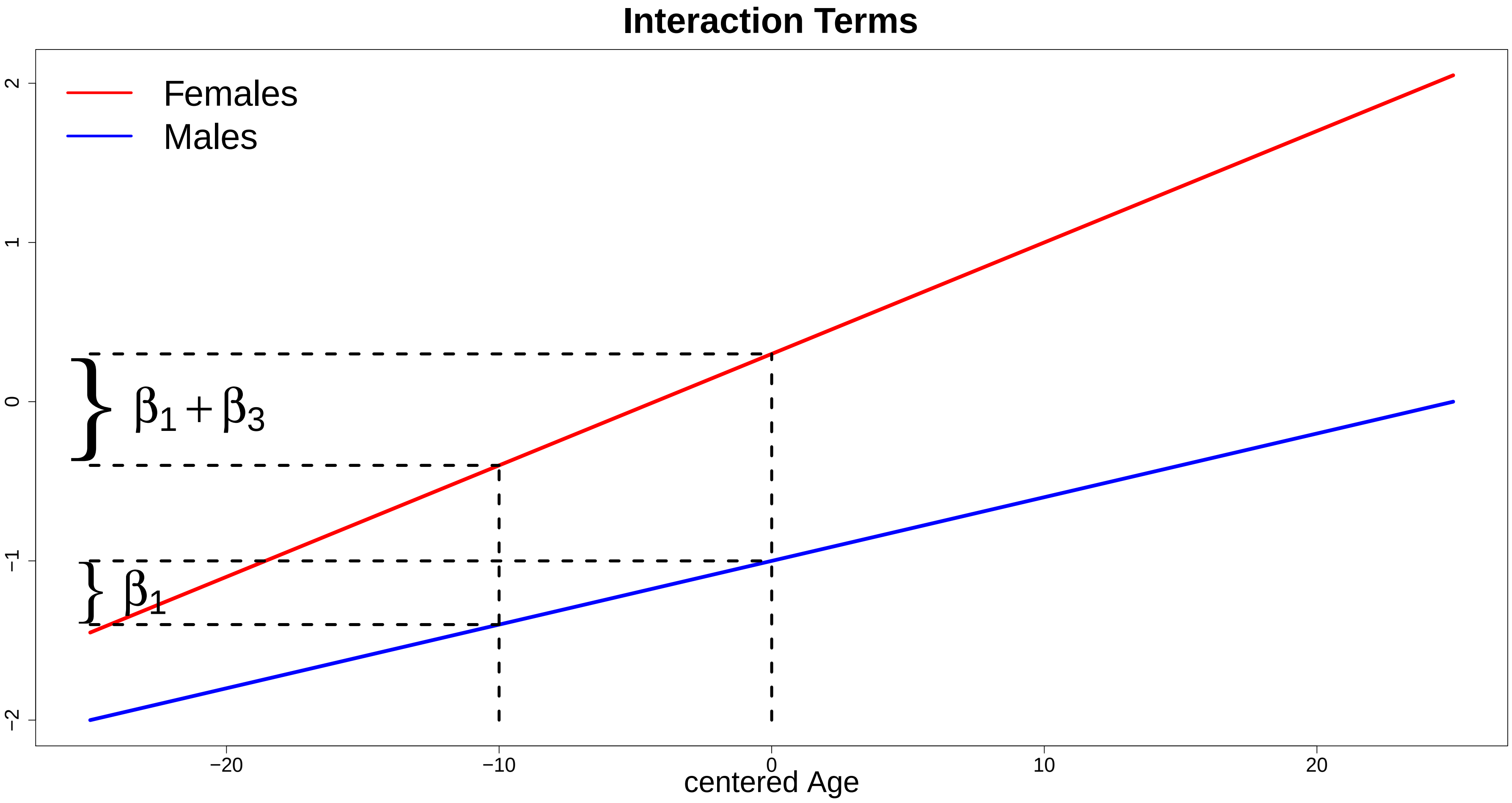
$$\text{Model II} \quad h_i(t) \;=\; h_0(t) \exp\big(\beta_1 \texttt{Age}_i + \beta_2 \texttt{Sex}_i + \beta_3 \texttt{Sex:Age}_i\big)$$

- Model I contains only additive effects whereas Model II contains also the interaction of age with gender
  - ▷ Model I assumes that the age effect is the same for males and females
  - ▷ Model II relaxes this assumption

4.5.2 Modelling Strategies – Additivity (cont'd)



4.5.2 Modelling Strategies – Additivity (cont'd)



4.5.2 Modelling Strategies – Additivity (cont'd)

- More care is required in extracting conclusions from models with interaction terms

- For instance, in Model II let the reference level for gender being 'Male', then
  - ▷ β_1 is the effect of Age for males
  - ▷ β_2 is the effect of females but for 0 years old!
  - ▷ (we could center Age, in which case β_2 is the effect of females for the mean age)

- However, it is clear that it is difficult to interpret β_1 and β_2 in isolation

4.5.2 Modelling Strategies – Additivity (cont'd)

- For Hypothesis testing:

Null or Alternative Hypothesis	In terms of Parameters
Effect of age is independent of sex Age effects are parallel Age and sex are additive	$H_0 : \beta_3 = 0$
Age interacts with sex Sex modifies effect of age Age and sex are nonadditive	$H_a : \beta_3 \neq 0$

4.5.2 Modelling Strategies – Additivity (cont'd)

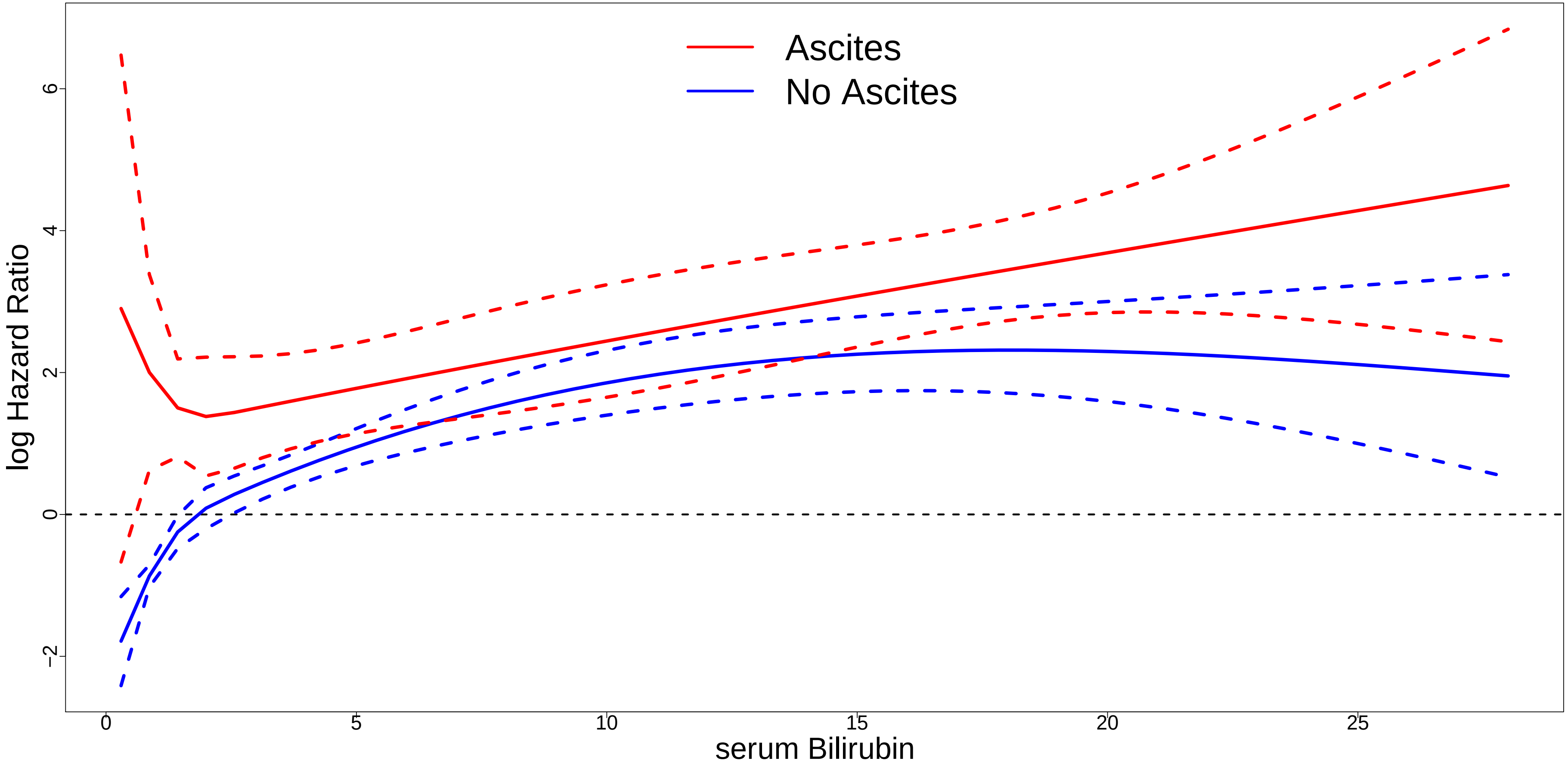
Null or Alternative Hypothesis	In terms of Parameters
Age is not associated with risk for an event	$H_0 : \beta_1 = \beta_3 = 0$
Age is associated with risk for an event	$H_a : \beta_1 \neq 0$ or $\beta_3 \neq 0$
Sex is not associated with risk for an event	$H_0 : \beta_2 = \beta_3 = 0$
Sex is associated with risk for an event	$H_a : \beta_2 \neq 0$ or $\beta_3 \neq 0$
Neither Age nor Sex are associated with risk for an event	$H_0 : \beta_1 = \beta_2 = \beta_3 = 0$
Either Age or Sex are associated with risk for an event	$H_a : \beta_1 \neq 0$ or $\beta_2 \neq 0$ or $\beta_3 \neq 0$

4.5.2 Modelling Strategies – Additivity (cont'd)

- Interactions can be easily combined with splines

- $\boxed{\text{Example:}}$ for the PBC data set we want to investigate the effects of serum Bilirubin and presence of ascites in the risk for death – we relax
  - ▷ the linearity assumption using splines for serum Bilirubin, and
  - ▷ the additivity assumption by taking the interaction of the nonlinear terms with ascites

4.5.2 Modelling Strategies – Additivity (cont'd)



4.5.3 Modelling Strategies – Overfitting

- Fitting too complex models (i.e., models with too many parameters) may result in *Overfitting*

- Overfitting has two important consequences for the fitted models

  ▷ estimated effects have increased variance ⇒ influences confidence intervals and p-values

  ▷ predicted values from the model do not agree with observed values from future data sets (from the same population) ⇒ the model does not validate well

4.5.3 Modelling Strategies – Overfitting (cont'd)

- To avoid overfitting and depending on the amount of information we have available in the data, there is only a limited number of parameters that we can reliably estimate

- Note: the number of parameters is not, in general, equal to the number of covariates
  - ▷ categorical covariates with k levels are represented by $k - 1$ dummy variables ⇒ they require $k - 1$ parameters
  - ▷ allowing for nonlinearity with splines ⇒ number of parameters equals the number of degrees of freedom

4.5.3 Modelling Strategies – Overfitting (cont'd)

- For survival data the number of parameters we can reliably estimate is (rule of thumb)

$$\frac{\#\text{ number of events}}{10} \quad \text{or} \quad \frac{\#\text{ number of events}}{15}$$

- This number can be boosted a bit by censored observations in the case of non-proportional hazards models

4.5.4 Modelling Strategies – General

- How to spend your available number of parameters
  - ▷ decide how important each one of the predictors is
    - * using prior subject-matter expert knowledge
    - * not using plots of each predictor with the outcome (this will lead to overoptimistic p-values)
  - ▷ if continuous predictors are highly skewed consider using a transformation (e.g., log)
  - ▷ if some levels of categorical predictors have very low frequencies consider collapsing these levels with other levels
  - ▷ for important continuous predictors relax the linearity assumption using splines
  - ▷ consider meaningful interactions

4.5.4 Modelling Strategies – General (cont'd)

- ▷ fit the model and check assumptions using residuals plots
- ▷ present the model using effect plots
- ▷ test the hypothesis of interest
- ▷ calculate predictions

- If the model is built to produce predictions, then
  - ▷ try to simplify the model using global tests, i.e.,
  - ▷ test all interaction terms simultaneously; if $p > 0.3$ all interaction terms can be omitted
  - ▷ apply an analogue global test for the nonlinear terms

4.5.4 Modelling Strategies – General (cont'd)

- If the model is built with purpose to estimate a specific effect of interest and to test a specific hypothesis, then
  - ▷ do not conserve degrees of freedom for the predictor(s) of interest
  - ▷ do not try to simplify the model by excluding insignificant predictors – in fact p-values obtained by the full model fit will be more accurate

4.6 Review of Key Points

- When we want to measure the effect of more than one predictors we use statistical models for time-to-event data

- We have two main options
  - ▷ Accelerated Failure Time (AFT) models
  - ▷ Proportional Hazards (PH) models

- AFT models measure the effect of predictors on the average failure time
  - ▷ they are the analogue of linear regression for event time data

4.6 Review of Key Points (cont'd)

- PH models measure the effect of predictors on the risk for an event

- AFT models are mainly parametric
  - ▷ they make specific assumptions for the distribution of the event times
  - ▷ due to censoring they are sensitive to misspecification of this distribution

- This lead to the development of the semiparametric Cox PH model
  - ▷ we make the PH assumption, i.e., the effect of covariate is multiplicative in the hazard scale
  - ▷ no assumption for the distribution of the event times

4.6 Review of Key Points (cont'd)

- For both modelling frameworks we have seen
  - ▷ how to estimate the model parameters
  - ▷ interpretation of model parameters
  - ▷ communicating the results of the model using effect plots
  - ▷ hypothesis testing
  - ▷ how to check the model assumptions

- In addition for the Cox model
  - ▷ sample size calculations based on the score test

4.6 Review of Key Points (cont'd)

- Regression modelling strategies
  - ▷ develop models for effect estimation, hypotheses testing or prediction
  - ▷ relax linearity of continuous predictors using splines
  - ▷ decide how many parameters to include depending on the number of events
  - ▷ decide how many degrees of freedom to spend for each covariate in advance

Part V

Extensions of the Cox Model

5.1 Expected Survival

- The Cox model describes the relationship between the survival curves of the different levels of a covariate using a single number ⇒ the Hazard Ratio

- However, unfortunately, the interpretation of the HR is not that straightforward
  - ▷ in many cases it is desirable to compare differences between groups in a more easily interpretable scale

- What is often medically relevant is to compare survival probabilities
  - ▷ e.g., how much greater is the probability of surviving at least 5 years in the treatment group than the probability of surviving at least 5 years in the placebo group

5.1 Expected Survival (cont'd)

- Our aim is to obtain survival probabilities based on the output of the Cox model

- Remember:
  - ▷ in the Cox model we do not need to estimate the baseline hazard – we leave it completely unspecified; (see Section 4.3.2)
  - ▷ however, the hazard is directly related to the survival function; (see Section 2.5)
  - ▷ therefore in order to estimate survival probabilities we first need an estimator for the baseline hazard function

5.1 Expected Survival (cont'd)

- A semiparametric estimator of the survival function based on the output of the Cox model is given by

$$\hat{S}_B(t) \;=\; \exp\Big\{-\hat{H}_0(t)\exp(\hat{\beta}_1 X_{i1} + \hat{\beta}_2 X_{i2} + \ldots + \hat{\beta}_p X_{ip})\Big\}$$

where the baseline cumulative hazard is estimated by

$$\hat{H}_0(t) \;=\; \sum_{i=1}^{n} \frac{I(T_i \leq t)\delta_i}{\sum_{j \in \mathcal{R}_i} \exp(\hat{\beta}_1 X_{j1} + \hat{\beta}_2 X_{j2} + \ldots + \hat{\beta}_p X_{jp})}$$

with $\mathcal{R}_i = \{j \text{ for which } T_j \geq T_i\}$ denoting the risk set, i.e., the subjects which did not have the event yet and are not censored

5.1 Expected Survival (cont'd)

- This is, in fact, an extension of the Breslow estimator we have used for the estimation of the survival function when we had no covariates (see Section 3.3)

- Its variance can be computed using similar arguments as in the no-covariates case

- $\boxed{\text{Example:}}$ for the renal graft failure data we are interested in the 5 and 10 year survival rates separately for male and female patients, controlling also for age and weight

5.1 Expected Survival (cont'd)

- The Cox model has the form

$$h_i(t) \;=\; h_0(t) \exp\bigl(\beta_1 \texttt{Sex}_i + \beta_2 \texttt{Age}_i + \beta_3 \texttt{Weight}_i\bigr)$$

  ▷ HR `Sex` $= 1.01$, $p = 0.130$

  ▷ HR `Age` $= 0.98$, $p = 0.011$

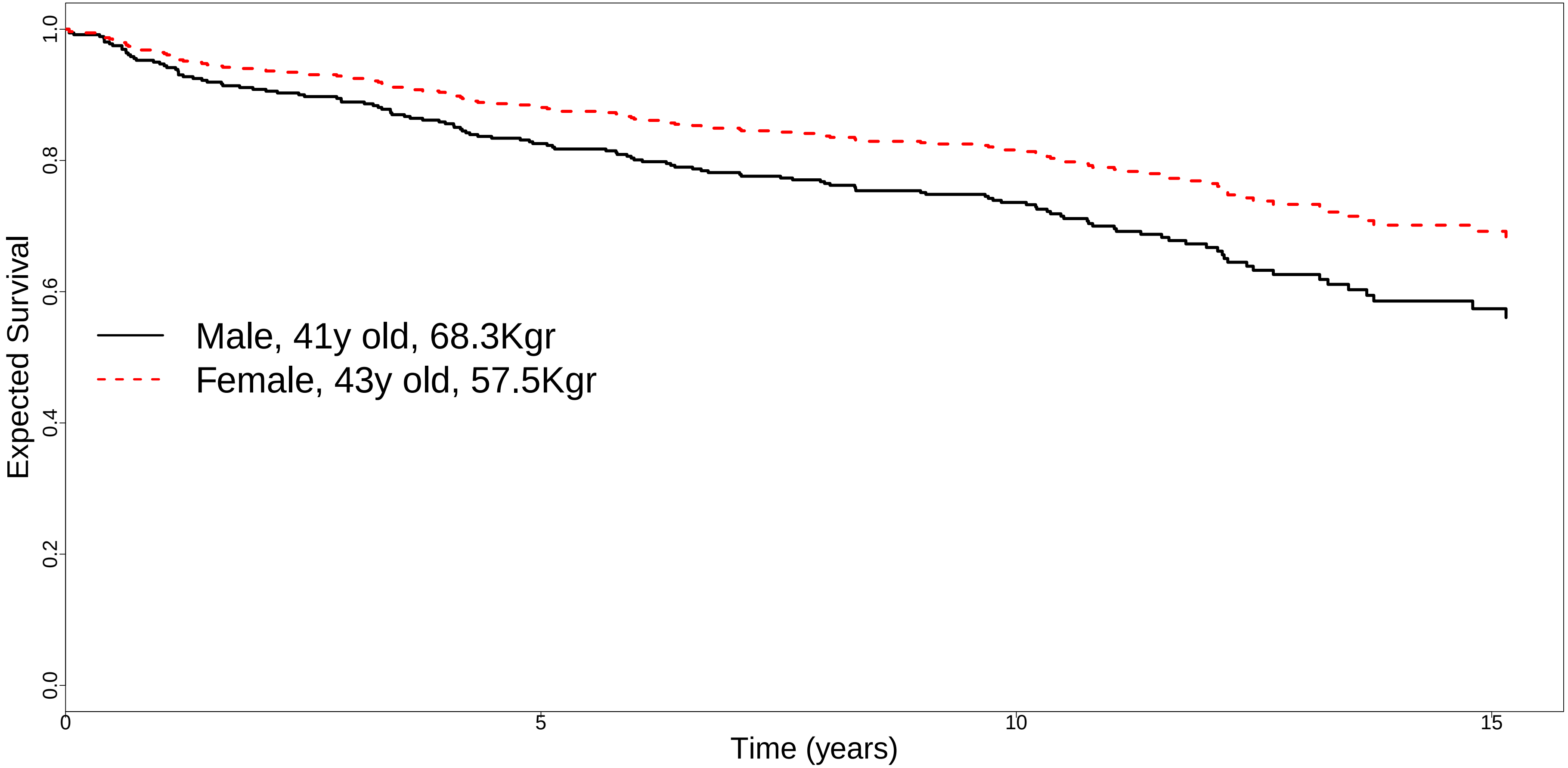
  ▷ HR `Weight` $= 1.25$, $p = 0.290$

- The following figure depicts estimated survival curves separately for

  ▷ the median male (41y old, 68.3Kgr)

  ▷ the median female (43y old, 57.5Kgr)

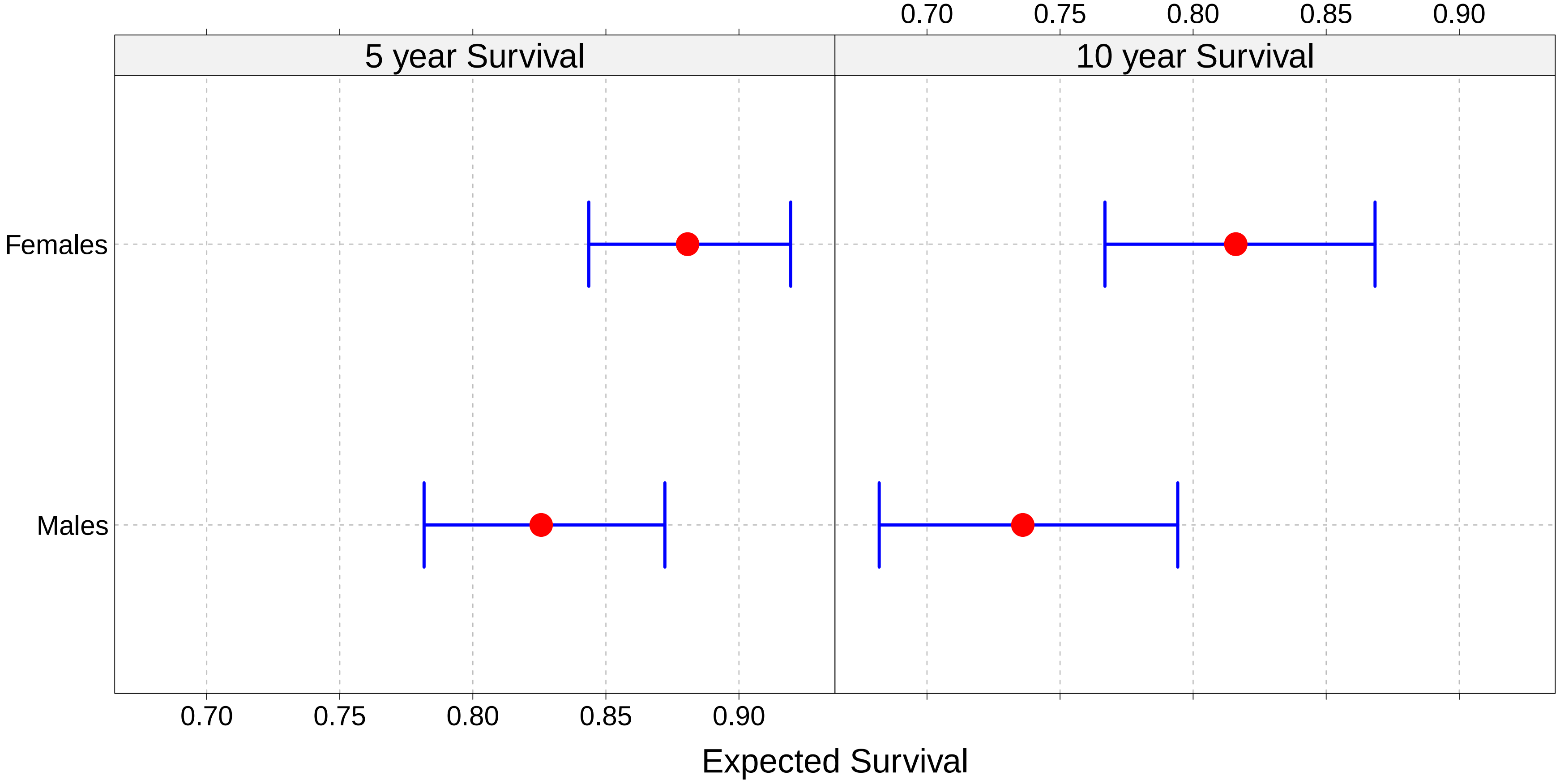
5.1 Expected Survival (cont'd)



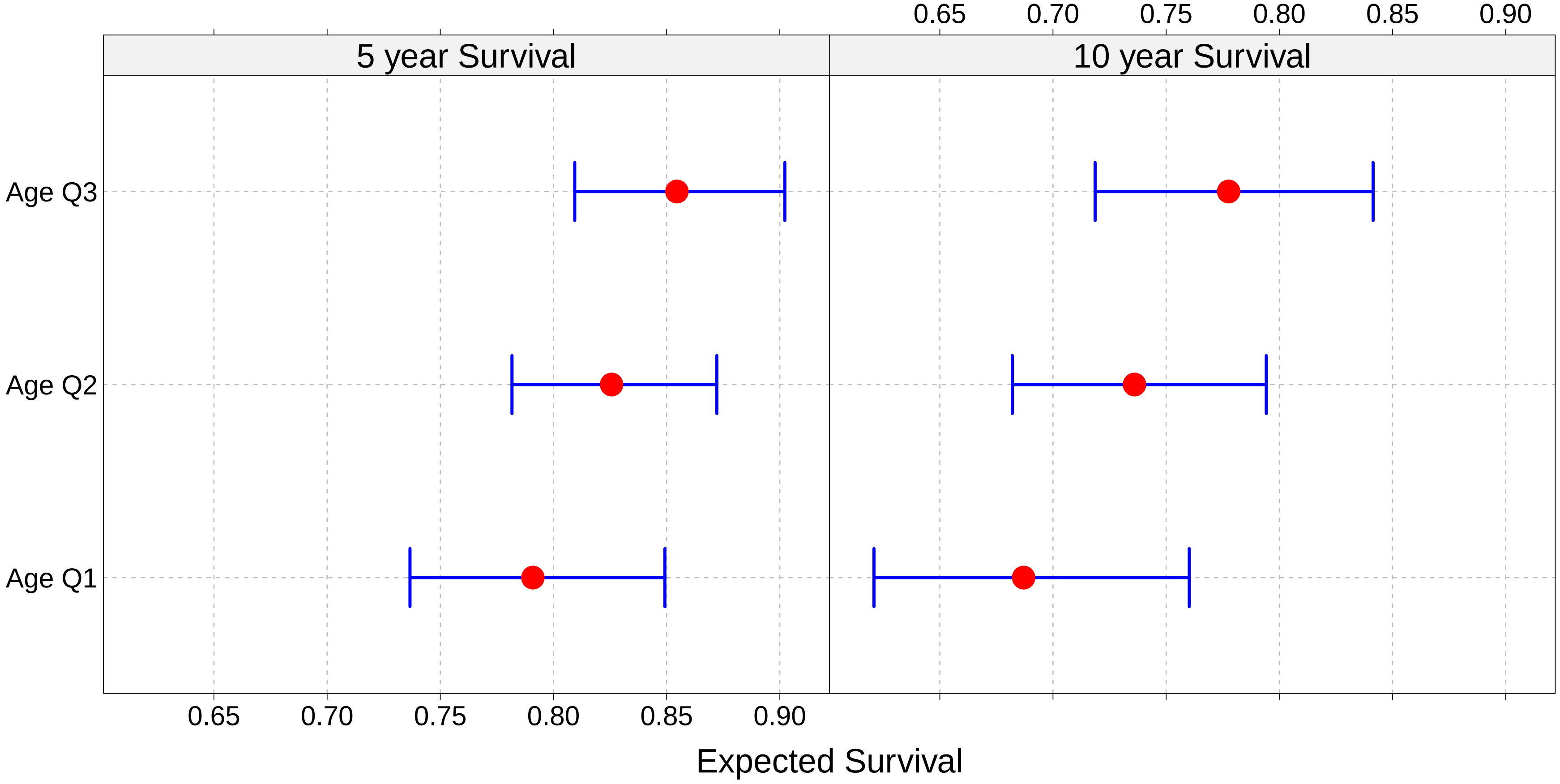
5.1 Expected Survival (cont'd)

- The next two figures illustrate, estimates of
  - ▷ 5y and 10y survival for the median male, and median female
  - ▷ 5y and 10y survival, for males of 68.3Kgr and
    - * 30.75 years old (Q1)
    - * 41 years old (Q2 - median)
    - * 51 years old (Q3)

5.1 Expected Survival (cont'd)



5.1 Expected Survival (cont'd)



5.1 Expected Survival (cont'd)

- A strange result:
  - ▷ the HR for Age is statistically significant different than 1 (HR `Age` $= 0.98$, $p = 0.011$);
  - ▷ however, the confidence intervals for 5y and 10y survival overlap for increasing Age
  - ▷ how is this possible?

- Remember: we compare survival curves over the whole follow-up period (see Section 3.4)
  - ▷ a significant HR does not mean that the survival curves differ everywhere
  - ▷ for instance, the 95% confidence intervals at $t = 0$ will almost certainly always overlap

5.1 Expected Survival (cont'd)

R> Estimates of survival probabilities from Cox models are produced in a similar manner as effect plots – we start with a fitted Cox model and a data frame that contains the specific combinations of covariate values for which we wish to estimate survival

```
fit <- coxph(Surv(Time, failure) ~ weight + age + gender,
        data = rgf)

ND <- with(rgf, data.frame(
    weight = c(median(weight[gender == "male"]),
            median(weight[gender == "female"])),
    age = c(median(age[gender == "male"]),
            median(age[gender == "female"])),
    gender = c("male", "female")
))
```

5.1 Expected Survival (cont'd)

R> The baseline survival function is estimated by survfit() – note that now the first argument is not a formula (as when we used this function compute the Kaplan-Meier estimate) but a fitted Cox model

R> The data frame that contains the specific combinations of covariate values is supplied in argument newdata – the summary() method can be used to provide survival probabilities estimates for specific follow-up times

```
sfit <- survfit(fit, newdata = ND)

sum.sfit <- summary(sfit, times = c(5, 10))
```

5.1 Expected Survival (cont'd)

R> The survival estimates with the associated 95% confidence intervals can be extracted using

```
out <- rbind(ND, ND)
out$times <- gl(2, 2, labels = c("5 year", "10 year"))
out$surv <- c(t(sum.sfit$surv))
out$lower <- c(t(sum.sfit$lower))
out$upper <- c(t(sum.sfit$upper))
out
```

5.2 Stratified Cox Model

- In many cases it may be unreasonable to assume that the baseline hazard is the same for groups of patients
  - ▷ multi-center clinical trials ⇒ varying patient populations are likely to have different baseline survival curves
  - ▷ . . .

- Violation of the proportional hazards assumption
  - ▷ for categorical covariates ⇒ the log hazard functions of the different levels are not parallel

5.2 Stratified Cox Model (cont'd)

- The *Stratified Cox Model* allows for multiple strata that divide subject into disjoint groups

$$h_i(t) = h_{0k}(t) \exp(\beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_p X_{ip})$$

where

▷ each of the $k = 1, \ldots, K$ strata has a distinct baseline hazard $h_{0k}(t)$

▷ the effect of covariates is the same for all strata (i.e., the coefficients β do not depend on k)

5.2 Stratified Cox Model (cont'd)

- $\boxed{\text{Example:}}$ for the PBC data set we are interested in the main effects of age and logarithm of serum Bilirubin – we present three analyses:
  - ▷ with no stratification
  - ▷ stratify for the different edema categories (3 categories)
  - ▷ stratify for the different edema and gender categories ($3 \times 2 = 6$ categories)

5.2 Stratified Cox Model (cont'd)

	No Strata		Edema		Edema × Sex	
	Value	Std. Err	Value	Std. Err	Value	Std. Err
Age	0.045	0.007	0.045	0.008	0.044	0.009
log ser Bilir	1.091	0.092	0.982	0.097	1.066	0.105

- We observe some differences, especially for the effect of log serum Bilirubin

5.2 Stratified Cox Model (cont'd)

R> To fit the stratified Cox model we need to specify the stratification variables – in `coxph()` we use function `strata()` within the `formula` argument

```
coxph(Surv(years, status2) ~ age + log(serBilir) + strata(edema),
      data = pbc2.id)
```

R> More than one stratification variables are specified as multiple arguments in `strata()`

```
coxph(Surv(years, status2) ~ age + log(serBilir) +
      strata(edema, sex), data = pbc2.id)
```

5.2 Stratified Cox Model (cont'd)

- The standard stratified Cox model assumes that the covariate effects are equal across strata
  - ▷ not always a reasonable assumption

- Interactions between strata and covariates can be easily included
  - ▷ if all of the covariate by strata interaction terms are added, then the results are identical to doing separate fits per stratum

- Example: in the stratified Cox model fitted in the PBC data set we are interested in testing whether the effect of serum Bilirubin is equal among the different edema categories

5.2 Stratified Cox Model (cont'd)

	Value	Std. Err	p-value
Age	0.043	0.008	< 0.001
log ser Bilir	1.082	0.118	< 0.001
log ser Bilir – edema no diuretics	-0.022	0.241	0.926
log ser Bilir – edema despite diuretics	-0.753	0.259	0.004

- The omnibus p-value for the interaction parameters is $0.024 \Rightarrow$ the effect of the log serum Bilirubin is not equal among the edema categories

5.2 Stratified Cox Model (cont'd)

R> Covariate by strata interactions are inlcuded in the standard way

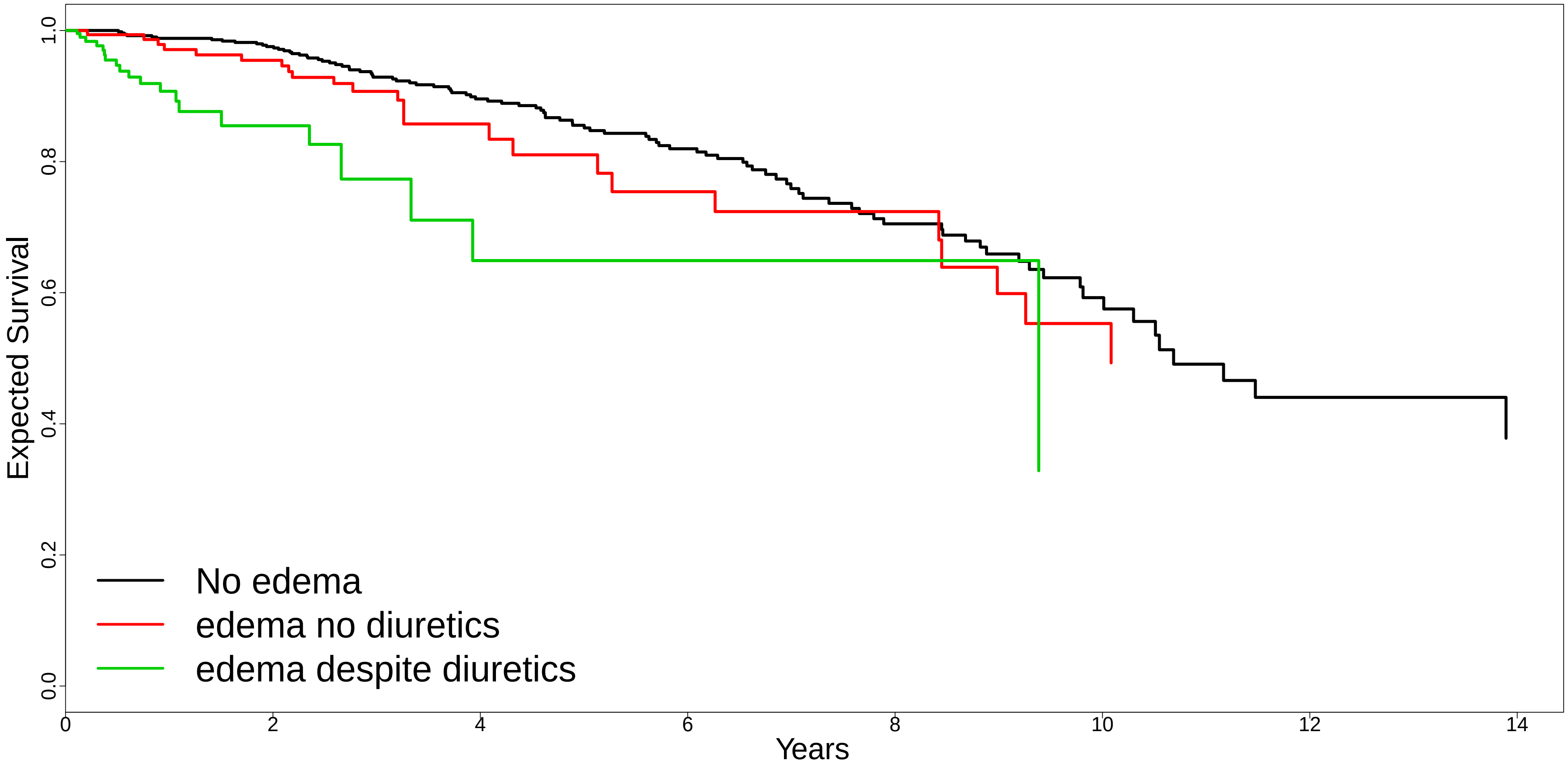
```
coxph(Surv(years, status2) ~ age + log(serBilir) * strata(edema),
        data = pbc2.id)
```

5.2 Stratified Cox Model (cont'd)

- Estimating survival probabilities in stratified Cox models,
  - ▷ a separate baseline hazard for each stratum, thus
  - ▷ we obtain a different survival curve for each stratum

- $\boxed{\text{Example:}}$ for the stratified Cox model fitted in the PBC data set we are interested to estimate the survival function of 40 year old patient with serum Bilirubin equal to 2

5.2 Stratified Cox Model (cont'd)



5.2 Stratified Cox Model (cont'd)

- Advantages of stratification
  - ▷ it gives the more general adjustment for a confounding variable
  - ▷ available in standard software

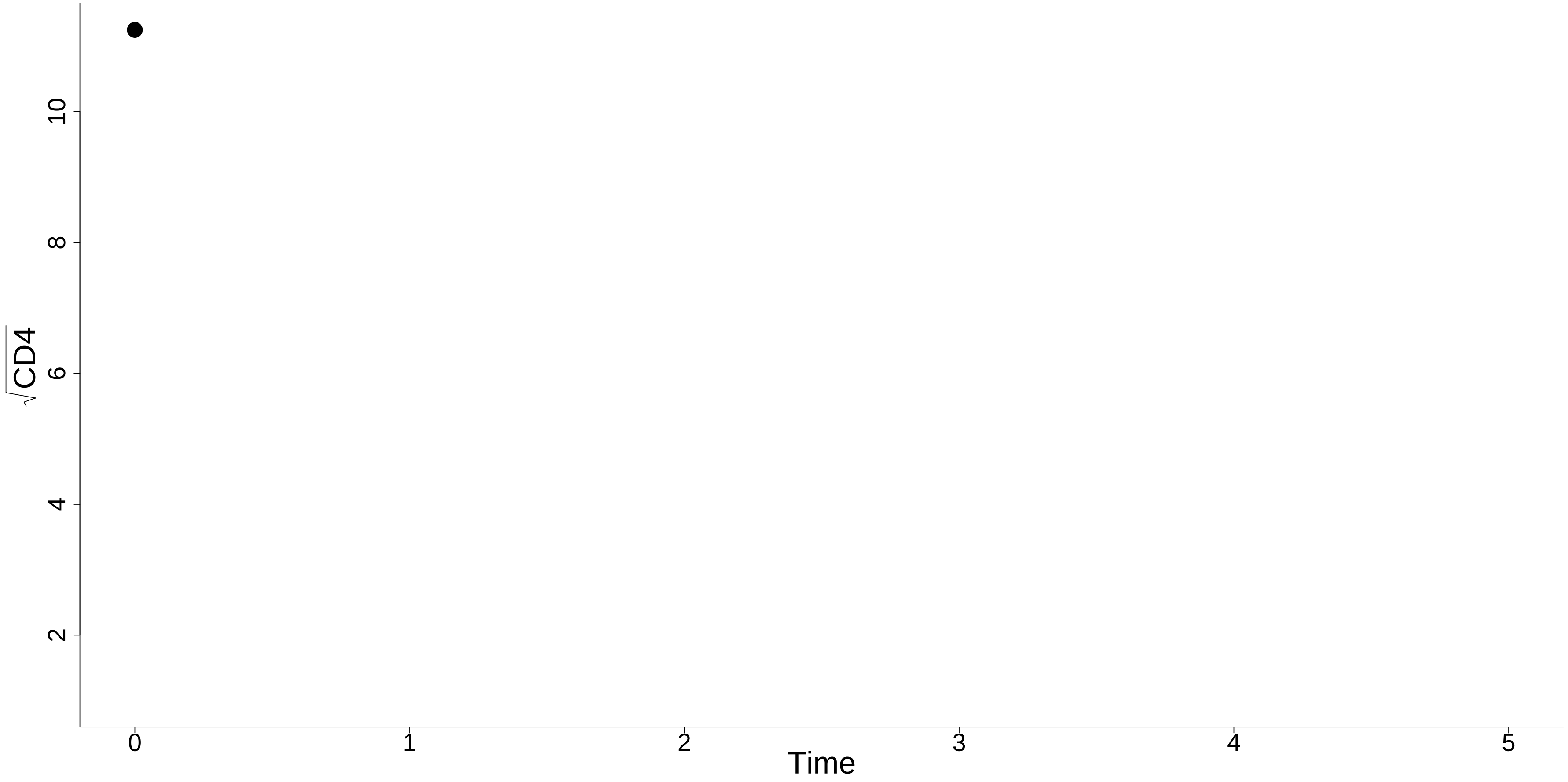
- Disadvantages of stratification
  - ▷ no direct estimate of the importance of the stratification factor is produced (no p-value)
  - ▷ large number of strata may result in decreased efficiency

5.3 Time-Dependent Covariates

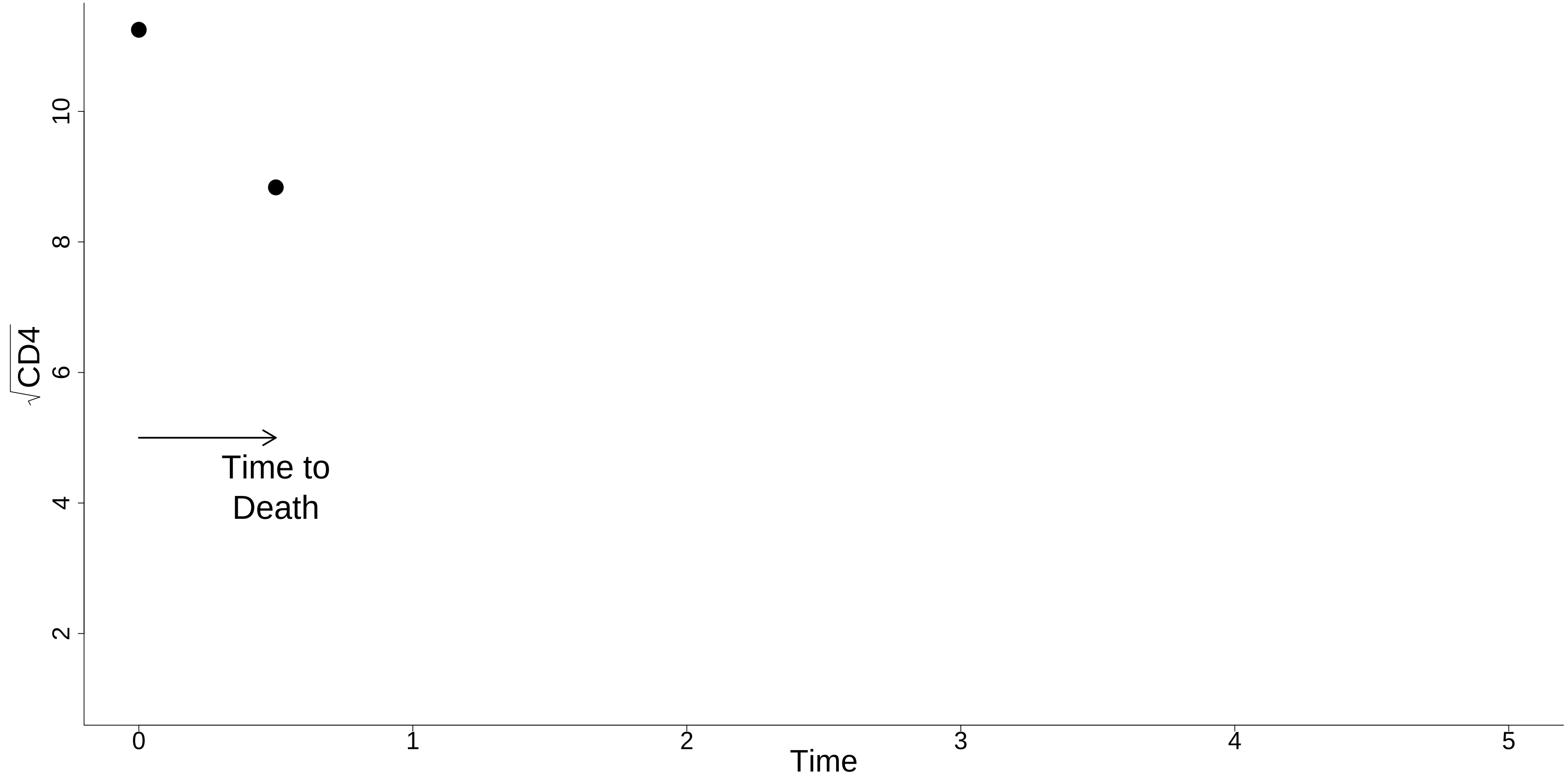
- In many cases, we are interested in the effect of covariates whose value changes in time
  - ▷ treatment (e.g., dose) changes with time
  - ▷ time-dependent exposure (e.g., smoking)
  - ▷ longitudinal measurements on the patient level (e.g., blood values)
  - ▷ . . .

- Example: in the AIDS data set we have repeated measurements of the CD4 cell count, which is a marker for the condition of the immune system

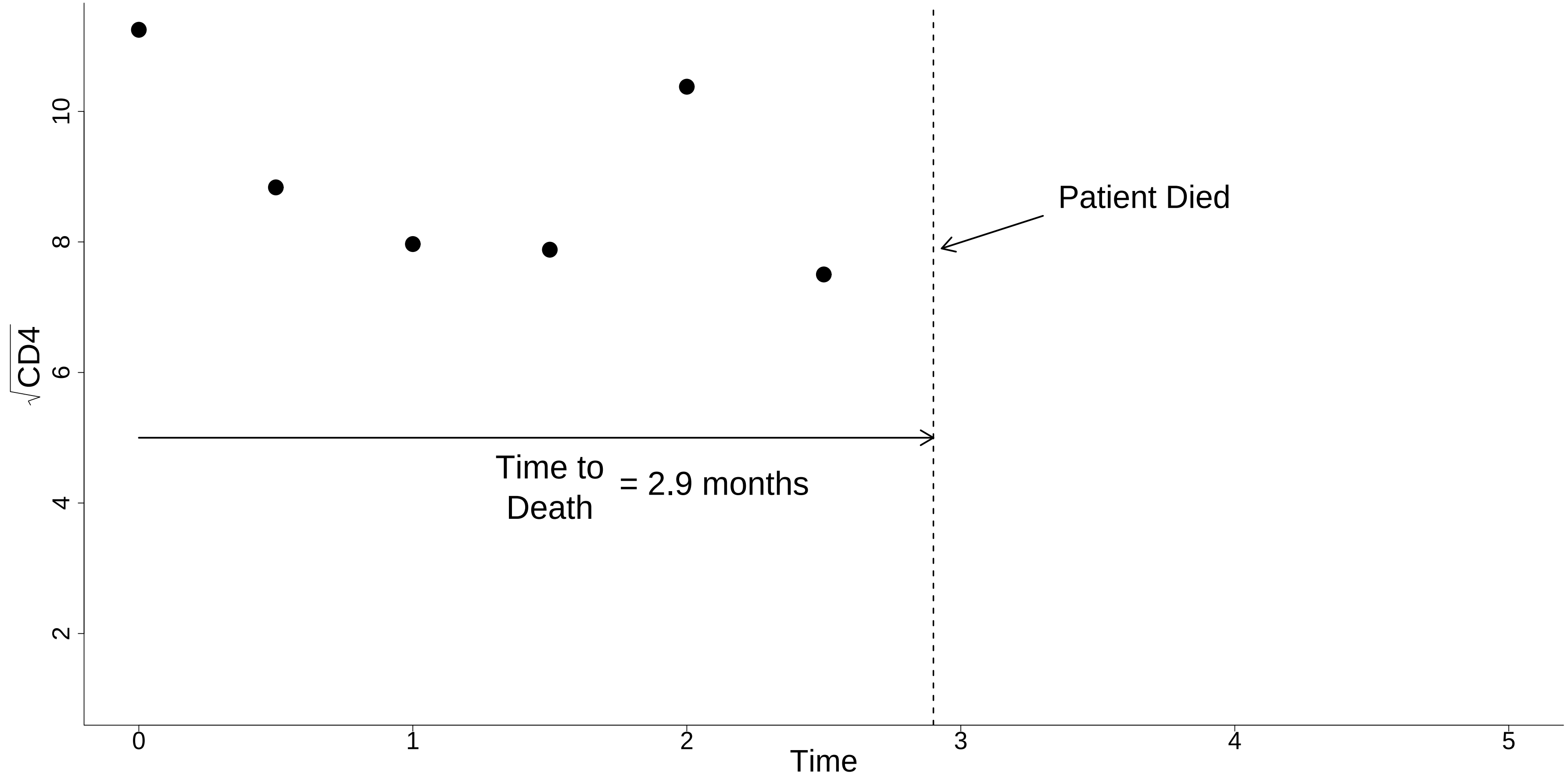
5.3 Time-Dependent Covariates (cont'd)



5.3 Time-Dependent Covariates (cont'd)



5.3 Time-Dependent Covariates (cont'd)



5.3 Time-Dependent Covariates (cont'd)

- There are two types of time-dependent covariates
  (Kalbfleisch & Prentice, *The Stat. Anal. of Failure Time Data*, 2002)
  - ▷ External (aka exogenous): the value of the covariate at time point t is not affected by the occurrence of an event at time point u, with $t > u$
  - ▷ Internal (aka endogenous): not External

- This is a difficult concept and we will try to explain it with an example. . .

5.3 Time-Dependent Covariates (cont'd)

- $\boxed{\text{Example:}}$ Consider a study on asthma, in particular on the time until an asthma attack for a group of patients

- We have two time-varying covariates: Pollution levels & a biomarker for asthma

- Say a patient had an asthma attack at a particular time point u
  - ▷ Pollution levels
    - * will the pollution levels at time $t > u$ be affected by the fact that the patient had an attack at u? $\Rightarrow$ **No**
  - ▷ Biomarker
    - * will the biomarker level at time $t > u$ be affected by the fact that the patient had an attack at u? $\Rightarrow$ **Yes**

5.3 Time-Dependent Covariates (cont'd)

- It is **very important** to distinguish between these two types of time-dependent covariates, because it determines the type of analysis that it should be followed

- If we treat internal covariates as external, we may produce spurious results

5.3 Time-Dependent Covariates (cont'd)

- We are interested in the effect of the type of maschine used to perform the dialysis in the time until kidney failure

A external time-dependent covariate

B internal time-dependent covariate

5.3 Time-Dependent Covariates (cont'd)

- We are interested in the effect of tumor size in survival

  A external time-dependent covariate

  B internal time-dependent covariate

5.3 Time-Dependent Covariates (cont'd)

- We are interested in the effect of cholesterol levels and blood pressure in survival

  A external time-dependent covariates

  B internal time-dependent covariates

5.3 Time-Dependent Covariates (cont'd)

- We are interested in the effect of income in the time to getting married

A external time-dependent covariate

B internal time-dependent covariate

5.3 Time-Dependent Covariates (cont'd)

- The Cox model can be easily extended to handle external time-dependent covariates

$$h_i(t) = h_0(t) \exp\{\beta^\top X_i + \alpha \, m_i(t)\}$$

where

- ▷ $\beta^\top X_i = \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_p X_{ip}$ denotes the baseline covariates as we had so far
- ▷ $m_i(t)$ denotes the value of the time-dependent covariate at time t
- ▷ α quantifies the effect of this covariate at time t to the hazard for an event at the same time point

5.3 Time-Dependent Covariates (cont'd)

- Note: the time-dependent Cox model relaxes the PH assumption
  - ▷ The hazard ratio is

$$\frac{h_i(t)}{h_0(t)} = \exp\{\beta^\top X_i + \alpha \; m_i(t)\}$$

which is not constant in time

- When we want to fit the Cox model taking into account the effect of external time-dependent covariates we need to use the counting process formulation
  - ▷ this is a rather technical subject which we will not describe in detail here

5.3 Time-Dependent Covariates (cont'd)

- To use this formulation the data must be organized in a long format

Patient	Start	Stop	Event	$m_i(t)$	Age
1	0	135	1	5.5	45
2	0	65	0	2.2	38
2	65	120	0	3.1	38
2	120	155	1	4.1	38
3	0	115	0	2.5	29
3	115	202	0	2.9	29
⋮	⋮	⋮	⋮	⋮	

5.3 Time-Dependent Covariates (cont'd)

- $\boxed{\text{Example:}}$ in the Stanford data
  - ▷ we measure the time until patients die
  - ▷ some patients received a heart transplantation
  - ▷ the dichotomous covariate yes/no transplantation can be considered as an external time-dependent covariate
  - ▷ we are interested in testing whether transplantation has a beneficial effect in survival

5.3 Time-Dependent Covariates (cont'd)

- The model that we fit has the form

$$h_i(t) = h_0(t) \exp\{\alpha m_i(t) + \beta Age_i\}$$

  where

  ▷ $m_i(t) = 1$ if patient i had a transplantation at some time $u \leq t$

  ▷ $m_i(t) = 0$ if patient i did not have a transplantation by time t

- We obtain the results

  ▷ $\exp(\hat{\alpha}) = 0.995$, $p = 0.989$

  ▷ transplantation does not improve survival

5.3 Time-Dependent Covariates (cont'd)

R> Cox models with external time-dependent covariates are fitted using the counting process notation – the data need to be arranged in the long format as in p.317

```
coxph(Surv(start, stop, event) ~ transplant, data = heart)
```

5.3 Time-Dependent Covariates (cont'd)

- Note: even though time-dependent covariates may produce better insights for the phenomenon under study, some times you may encounter surprising results

- Example: Cavender et al. (1992, J. Am. Coll. Cardiol.) conducted an analysis to test the effect of cigarette smoking on survival of patients who underwent coronary artery surgery
  - ▷ the estimated effect of current cigarette smoking was positive on survival although not significant (i.e., patient who smoked had higher probability of survival)
  - ▷ most of those who had died were smokers but many stopped smoking at the last follow-up before their death

5.3 Time-Dependent Covariates (cont'd)

- The approaches these authors used to take this feature into account were
  - ▷ instead of the last they used the pre-last assessment as time-dependent covariate (lagged covariates)
  - ▷ to use the percentage of the follow-up period that the patient smoked
  - ▷ both resulted in a statistically significant increased risk from cigarette smoking

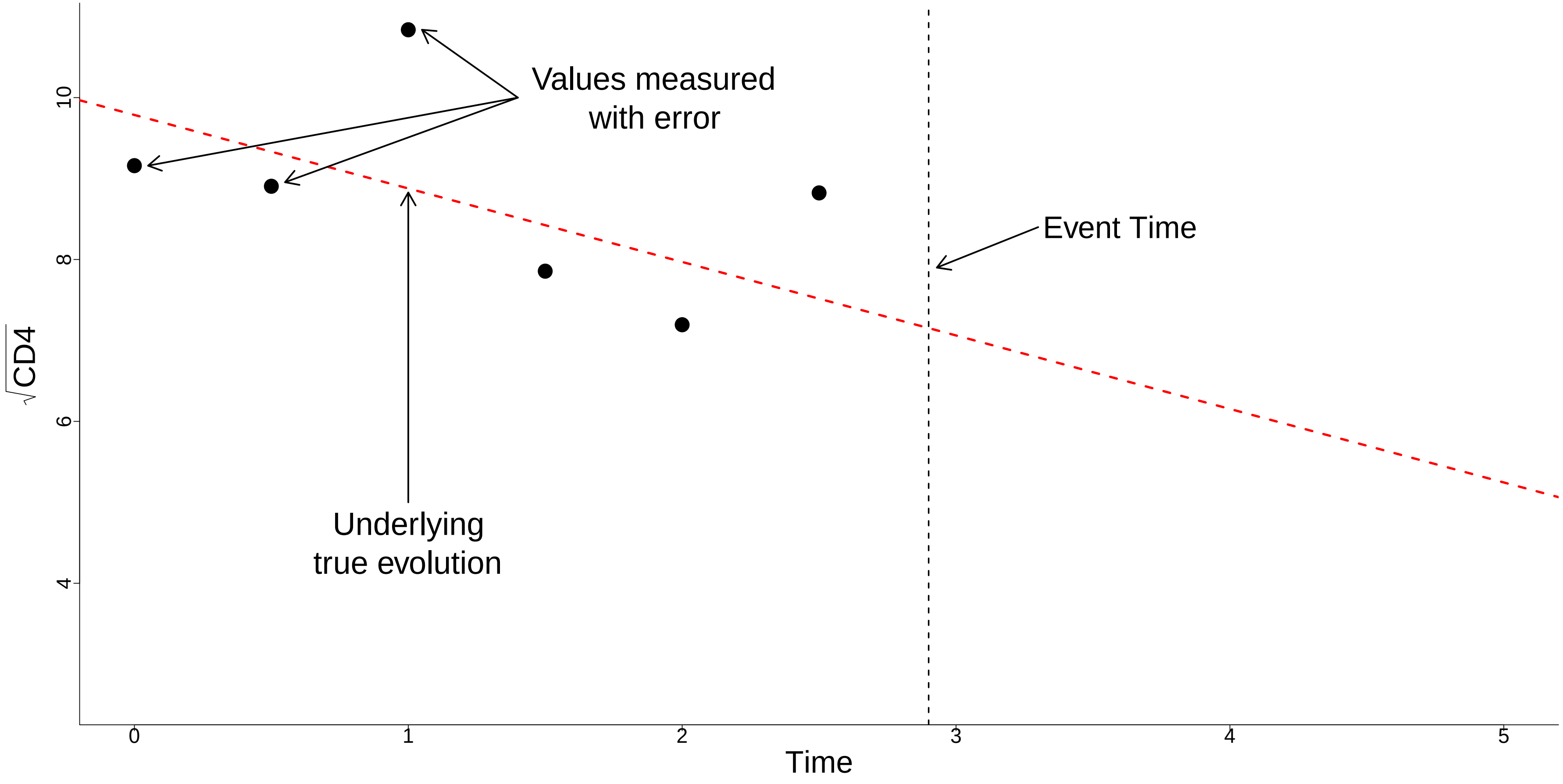
- The choice of the functional form of the time-dependent covariate can have a substantial impact on the derived results

5.3 Time-Dependent Covariates (cont'd)

- Let's turn our attention to internal covariates
  - ▷ what is exactly the problem with such covariates and why we cannot use the time-dependent Cox model

- They have a stochastic nature
  - ▷ they contain measurement error
  - ▷ we do not have the complete history available
    (by 'history' we mean the values at any time point)

5.3 Time-Dependent Covariates (cont'd)



5.3 Time-Dependent Covariates (cont'd)

- To solve this problem there have been developed statistical models that jointly model the time-to-event outcome with the longitudinal responses

- Intuitively, these models reconstruct the history of the time-dependent covariate and then this estimated history is included as a covariate in the survival model

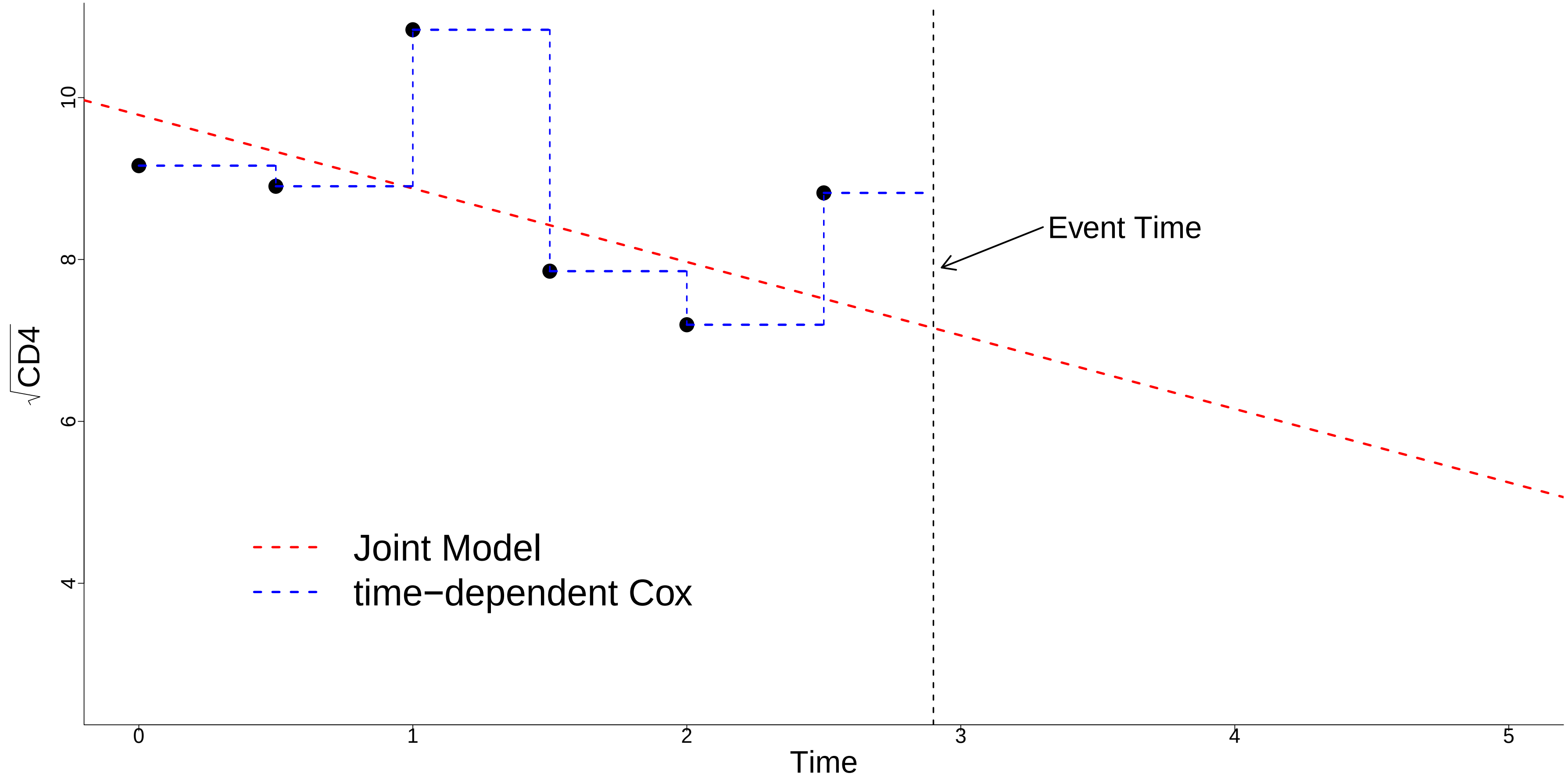
5.3 Time-Dependent Covariates (cont'd)

- To illustrate the virtues of joint modelling, we compare with the standard time-dependent Cox model
  - ▷ i.e., we ignore the measurement error in the CD4 cell count

	Joint Model	Naive TD Cox
	value (std.err)	value (std.err)
Treat	0.35 (0.15)	0.33 (0.15)
CD4	−1.10 (0.12)	−0.72 (0.08)

- Clearly, there is a considerable effect of ignoring the measurement error, especially for the effect of CD4!

5.3 Time-Dependent Covariates (cont'd)



5.3 Time-Dependent Covariates (cont'd)

R> Such joint models are fitted using functions from `JM` package

R> They require fitting separately a linear mixed effects model and a Cox models

R> A detailed example can be found in the **R Code Appendix**

5.3 Time-Dependent Covariates (cont'd)

- Estimating survival probabilities in the presence of time-dependent covariates is a challenging task

- Problem: the notion of the 'mean' or 'median' individual is not that straightforward to define – why is that:
  - ▷ except from meaningful 'median' values for the baseline covariates (i.e., the time-independent covariates),
  - ▷ we also need to specify a path of previous values for the time-dependent, but
  - ▷ how a 'median' path of values can be specified?

5.4 Clustered Event Time Data

- In many case studies sample units are clustered
  - ▷ patients are clustered within hospitals
  - ▷ children are clustered with schools or families
  - ▷ recurrent asthma attacks (cluster is the patient)
  - ▷ . . .

- Subjects from the same cluster are expected to be **(positively) correlated**
  - ▷ if a farmer treats his herd better than an other one, then his cows are expected to live longer

5.4 Clustered Event Time Data (cont'd)

- So far we have assumed that the time-to-event for one subject is completely independent from the time-to-event of another subject

- However, subjects from the same cluster cannot be considered as independent

- Therefore, clustering **must** be taken into account in our analysis

5.4 Clustered Event Time Data (cont'd)

- What are the implications of clustering
  - ▷ it does not affect consistency
  - ▷ it does affect efficiency

- In practice, this implies that
  - ▷ the estimated effects (i.e., parameter estimates) from the Cox model are valid
  - ▷ the estimated standard errors are not $\Rightarrow$ we need to adjust them

5.4 Clustered Event Time Data (cont'd)

- To estimate the standard errors taking into account clustering the **grouped Jackknife** method works satisfactorily

$$\hat{\text{var}}(\hat{\beta}) = \left(\frac{C - p}{C}\right) \sum_{c=1}^{C} (\hat{\beta}_{-c} - \hat{\beta})(\hat{\beta}_{-c} - \hat{\beta})^\top$$

where

▷ C is the number of clusters

▷ p the number of parameters

▷ $\hat{\beta}$ the parameter estimates using all the clusters

▷ $\hat{\beta}_{-c}$ the parameter estimates excluding cluster c

5.4 Clustered Event Time Data (cont'd)

- $\boxed{\text{Example:}}$ the patients in the Lung data set are clustered in institution (we have ignored this feature in the previous analyses of this data set)
  - ▷ we fit a Cox model in which we correct for age, gender, and the Karnofsky performance score

5.4 Clustered Event Time Data (cont'd)

	Value	Std. Err. Naive	Std. Err. Jackknife	Ratio Std. Err. Naive / Jackknife
Age	0.012	0.0094	0.0062	1.52 (52% ↑)
Sex	−0.497	0.1679	0.1252	1.34 (34% ↑)
Karno	−0.013	0.0059	0.0086	0.69 (31% ↓)

- We observe some considerable differences between the naive and jackknife standard errors

5.4 Clustered Event Time Data (cont'd)

R> To fit a marginal Cox model for clustered data we need to specify which observations belong to the same cluster – this is achieved using function `cluster()` within the `formula` argument of `coxph()`

R> By default both the naive and Jackknife (termed 'robust') standard errors are included in the output

```
coxph(Surv(time, status) ~ age + cluster(inst), data = lung)
```

5.4 Clustered Event Time Data (cont'd)

- An alternative approach to handle correlated event time data is to use frailty terms

$$h_{ij}(t) = h_0(t)\, \omega_i \; \exp(\beta_1 X_{ij1} + \beta_2 X_{ij2} + \ldots + \beta_p X_{ijp})$$

where

- ▷ $h_{ij}(t)$ is the hazard for subject j in cluster i
- ▷ ω_i is the frailty term, which is an unobserved random variable that
  - * is shared in all subjects in the same cluster ⇒ induces positive correlation between the subjects of each cluster
  - * explains heterogeneity between clusters

5.4 Clustered Event Time Data (cont'd)

- Basic Assumption: given the frailty term subjects from the same cluster have independent hazard functions – the log-likelihood can be written as (see p.125)

$$\ell(\theta) = \sum_{i=1}^{n} \log \Biggl[\int \prod_{i=1}^{n_i} \Bigl\{ h_0(T_{ij}) \omega_i \exp(\beta^\top X_{ij}) \Bigr\}^{\delta_{ij}} \exp \Bigl\{ -H_0(T_{ij}) \omega_i \exp(\beta^\top X_{ij}) \Bigr\} \\ f(\omega_i; \sigma) \, d\omega_i \Biggr]$$

  where

  ▷ we assume that ω_i follows a distribution (e.g., Gamma, log-Normal, etc.), with density $f(\omega_i; \sigma)$

  ▷ σ is a scale parameter that quantifies correlation within cluster ⇒ heterogeneity between clusters

5.4 Clustered Event Time Data (cont'd)

- ▷ maximizing the log-likelihood is a bit more difficult than for the Cox model, but it is available in current software

- **Important:** the interpretation of parameters is different between
  - ▷ Cox model with grouped jackknife variance estimator (marginal model)
  - ▷ frailty models (conditional model)

5.4 Clustered Event Time Data (cont'd)

- Example: for the Lung data set we fit a marginal Cox model and a frailty Cox model, where in both we correct for the age effect

	Marginal		Frailty	
	Value	Std. Err.	Value	Std. Err.
Age	0.0186	0.0071	0.0194	0.0093
σ	—	—	0.19	

▷ $\exp(0.0194) = 1.020$ is the hazard ratio for 1 year increase in age for patients in the same institution

▷ $\exp(0.0186) = 1.018$ is the hazard ratio for 1 year increase in age independently from the institution (pooled effect)

5.4 Clustered Event Time Data (cont'd)

R> Cox models with frailty terms are fitted similarly to marginal models – now function frailty() is used within the formula argument of coxph() to identify subjects belonging to the same cluster

```
coxph(Surv(time, status) ~ age + frailty(inst, df = 4), data = lung)
```

5.4 Clustered Event Time Data (cont'd)

- Features of frailty models
  - ▷ they directly provide a measure of the correlation between clusters
  - ▷ they can be more efficient
  - ▷ sensitivity in the assumed distribution for the frailty terms
  - ▷ limited availability of model checking tools

- The choice between marginal and frailty models should be dictated by the focus of inference, i.e., conditional versus marginal

5.5 Competing Risks

- Often in clinical studies we are faced with multiple endpoints, e.g.,
  - ▷ cancer studies: recurrence of the disease and death
  - ▷ cancer studies: death from cancer and death due to other causes
  - ▷ cardiovascular studies: nonfatal myocardial infraction and death

- In some cases it makes sense to combine the endpoints
  - ▷ e.g., time until either recurrence of the disease or death (aka *Disease Free Survival*)

- We end up with a single composite event of interest and therefore, all tools that we have seen so far for the analysis of survival times can be used

5.5 Competing Risks (cont'd)

- However, in some cases we may be interested in one of the possible events
  - ▷ e.g., we are interested specifically in the time until recurrence of cancer but not in the time until death
  - ▷ or we are interested separately in the time until recurrence and the time until death

- Problem: occurrence of another event prevents occurrence of the event of main interest
  - ▷ e.g., for patients who died we can never observe the recurrence of cancer

- The simultaneous consideration of more that one events, which are exclusive with each other is known as a **Competing Risks** problem

5.5 Competing Risks (cont'd)

- **Crucial distinction:** are the competing risks independent?
  - ▷ if yes, then treating all events from all other causes (except from the one of interest) as censored will produce valid results
  - ▷ if not, then treating all other events as censored will produce **biased** results

- Examples:
  - ▷ in a study on the survival of cancer patients a patient dies in a car accident ⇒ independent endpoints
  - ▷ in a study on the survival of cancer patients a patient dies from a heart attack ⇒ probably dependent endpoints
  - ▷ in a study on patients with osteoporosis were are interested in the time-to-fracture; however, some patients die ⇒ dependent or independent endpoints?

5.5 Competing Risks (cont'd)

- Caveat: you cannot test for independence
  - ▷ unless unverifiable assumptions are made

- Thus, if you are not sure if the independence assumption is satisfied, you should always do an analysis that allows for dependent competing risks

5.5 Competing Risks (cont'd)

- Notation for competing risks
  - ▷ $T_1^*, T_2^*, \ldots, T_K^*$ time-to-failure from each one of the K causes
  - ▷ C censoring time (censoring independent of all T_k^*, $k = 1, \ldots, K$)

what we observe is

- ▷ $T = \min(T_1^*, T_2^*, \ldots, T_K^*, C)$
- ▷ $D = 0, 1, \ldots, K$ with 0 denoting censored time, and $D = k$ failure from cause k

5.5 Competing Risks (cont'd)

- What is estimable from the available data is the *cause-specific hazard function*

$$h_k(t) = \lim_{s \to 0} \frac{\mathsf{Pr}(t \leq T < t + s, D = k \mid T \geq t)}{s}$$

which is the hazard of failing from cause k at time t

- Anything that can be derived (uniquely) from the cause-specific hazard is also estimable

5.5 Competing Risks (cont'd)

- Remember: based on the hazard function we can derive the survival function using the relation (see Section 2.5)

$$S_k(t) = \exp\Bigl\{ - \int_0^t h_k(u) \, du \Bigr\}$$

which denotes the probability of failure from cause k after time t **if**

$$h_j(t) = 0, \qquad \text{for } j = 1, \ldots, K \quad \text{excluding} \quad k,$$

that is, in the **absence** of competing risks

5.5 Competing Risks (cont'd)

- Therefore, $S_k(t)$ denotes the survival function for cause k in a hypothetical population where failure from other causes has been eliminated
  - ▷ not very relevant for practical use (even though frequently used in medical papers)

- Note: if we use the Kaplan-Meier estimator to estimate the survival function treating events from other causes as censored, then we actually estimate $S_k(t)$
  - ▷ bias $\Rightarrow$ the probability of failing is overestimated

- Need for more easily interpretable functions

5.5 Competing Risks (cont'd)

- The *Overall survival function*

$$S(t) = \exp\left\{-\sum_{k=1}^{K} \int_0^t h_k(u) \, du\right\}$$

describes the probability of not having failed from any cause by time t

- The *Cumulative incidence function* of cause k

$$F_k(t) = \int_0^t h_k(u) S(u) \, du$$

describes the probability of failing from cause k before time t

5.5 Competing Risks (cont'd)

- The cumulative incidence function can be estimated in a similar manner as the Kaplan-Meier estimate of the survival function

- Remember: the Kaplan-Meier estimator was based on the law of total probability (see Section 3.2)

- We follow the notation of Section 3.2
  - ▷ $t_1, t_2, \ldots, t_m$ denote the unique event times in the sample at hand
  - ▷ d_i is the number of events at time t_i, from all causes
  - ▷ d_{ki} is the number of events at time t_i, from cause k
  - ▷ r_i the number of patients still at risk at time t_i

5.5 Competing Risks (cont'd)

- The overall survival function $S(t)$ can be estimated using the Kaplan-Meier estimator, without considering the cause of failure (see Section 3.2)

$$\hat{S}(t) = \prod_{i:t_i \leq t} \frac{r_i - d_i}{r_i}$$

- The cause-specific hazard function can be estimated by

$$\begin{aligned}\hat{h}_k(t) &= \text{Prob failing from cause } k \text{ at time } t_i \text{ given survival up to time } t_{i-1} \\ &= \frac{d_{ki}}{r_i}\end{aligned}$$

5.5 Competing Risks (cont'd)

- Using the definition of the cumulative incidence function

$$F_k(t) = \int_0^t h_k(u) S(u) \, du$$

we obtain the estimator

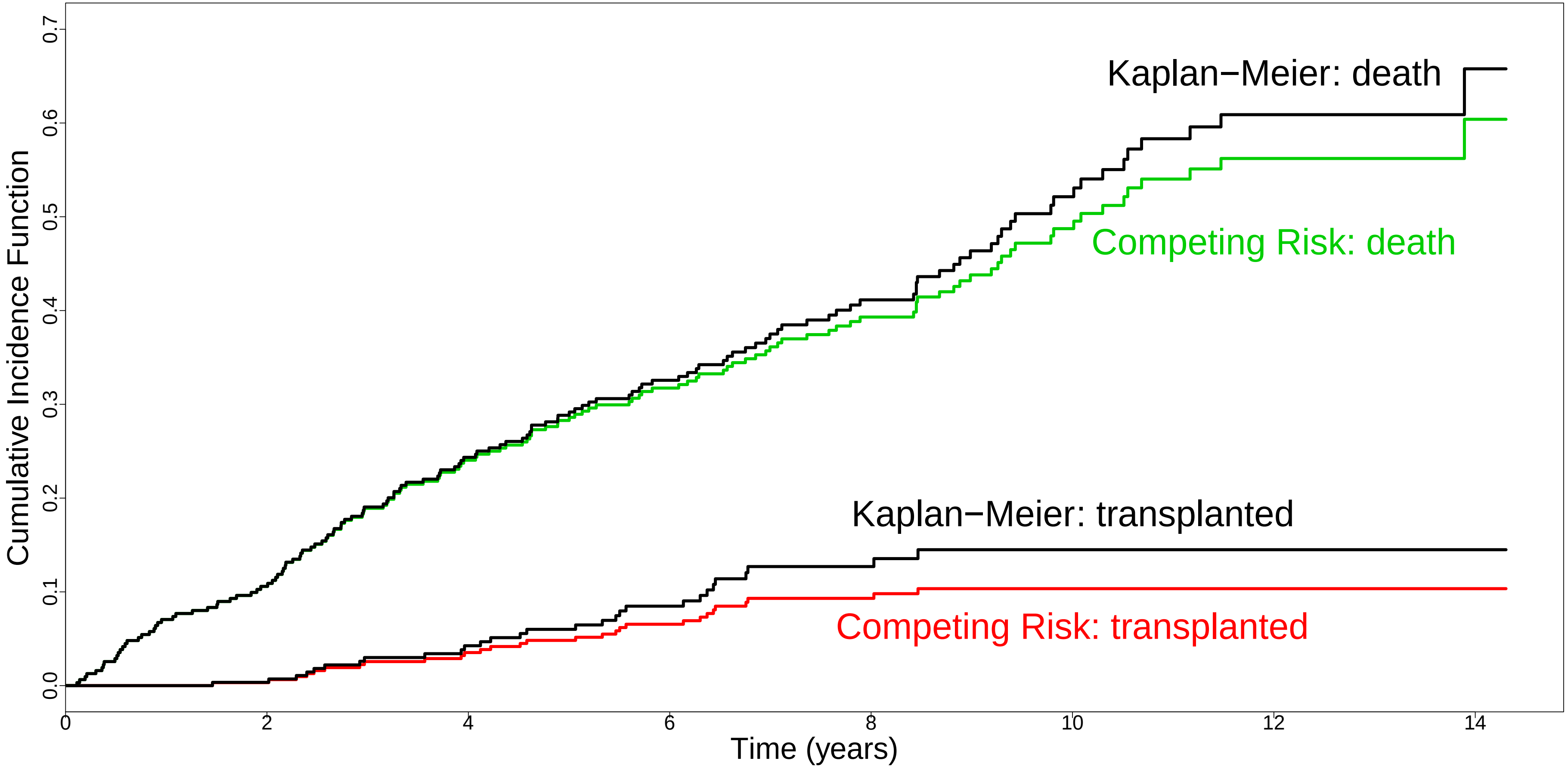
$$\hat{F}_k(t) = \sum_{i:t_i \leq t} \hat{h}_k(t_i) \, \hat{S}(t_{i-1})$$

5.5 Competing Risks (cont'd)

- Example: in the PBC data and when interest is in the time until transplantation, death is a competing risk (i.e., if a patient dies he can never have a transplantation)

- We compare the estimates of the cumulative incidence function for both events (death and transplantation) using
  - ▷ the naive Kaplan-Meier estimator (that treats events from other causes as censored), and
  - ▷ the estimator we derived previously

5.5 Competing Risks (cont'd)



5.5 Competing Risks (cont'd)

- As expected, we observe that the naive Kaplan-Meier overestimates the cumulative incidence function

5.5 Competing Risks (cont'd)

R> The competing risks estimate of the cumulative incidence function is produced by the `survfit()` function

R> Two differences: (i) the event indicator `status3` is 1 whenever either of the two events occurred; (ii) argument `etype` is used to distinguish between the events of interest

```
survfit(Surv(years, status3) ~ 1, data = pbc2.id, etype = status)
```

5.5 Competing Risks (cont'd)

- Inclusion of covariates: two different approaches

  ▷ Cox model for the cause-specific hazards

  * straightforward to implement in standard software (R, SAS)
  * more difficult to get estimates of the cumulative incidence function
  * proportional hazards assumption for the cause-specific hazards

  ▷ Fine & Gray model (JASA, 1999)

  * hazard ratios interpretable on the cumulative incidence scale
  * only available in R (package `cmprsk`)

5.5 Competing Risks (cont'd)

- We will only illustrate the Cox model for the cause-specific hazards

$$h_{ki}(t) \;=\; h_{k0}(t) \exp(\beta_{k1} X_{ki1} + \beta_{k2} X_{ki2} + \ldots + \beta_{kp} X_{kip})$$

  where

  ▷ $h_{ki}(t)$ the hazard for patient i for cause k

  ▷ $h_{k0}(t)$ the baseline hazard for cause k

  ▷ $\beta_{k1}, \ldots, \beta_{kp}$ log hazard ratios for the covariates $X_{k1}, \ldots, X_{kp}$ for cause k

- For its estimation <u>it is correct</u> to treat each failure from other causes as censored

5.5 Competing Risks (cont'd)

- $\boxed{\text{Example:}}$ in the PBC data we are interested in the effects of treatment and age in the hazards for transplantation and death

- We fit the cause-specific hazard models

$$\begin{aligned} h_{1i}(t) &= h_{10}(t) \exp\bigl(\beta_{11} \texttt{Treat}_i + \beta_{12} \texttt{Age}_i\bigr) \\ h_{2i}(t) &= h_{20}(t) \exp\bigl(\beta_{21} \texttt{Treat}_i + \beta_{22} \texttt{Age}_i\bigr) \end{aligned}$$

where

▷ $k = 1$ for transplantation, and

▷ $k = 2$ for death

5.5 Competing Risks (cont'd)

Transplantation	est.	exp(est.)	s.e.	p-value
D-penicil – β_{11}	-0.24	0.79	0.38	0.530
Age – β_{12}	-0.10	0.91	0.02	< 0.001

Death	est.	exp(est.)	s.e.	p-value
D-penicil – β_{21}	-0.16	0.85	0.17	0.347
Age – β_{22}	0.05	1.05	0.01	< 0.001

5.5 Competing Risks (cont'd)

- We observe that after correcting for treatment
  - ▷ younger patients will receive a transplant sooner, whereas
  - ▷ older patient have greater risk of dying

- These risk estimates **cannot** be converted to relative survival probabilities using the known formula (see pp. 175–177)

$$S(t) = \bigl\{S_0(t)\bigr\}^{\exp(\beta^\top X)}$$

due to the competition between the different causes

5.5 Competing Risks (cont'd)

- Moreover, the obtained parameter estimates from the cause-specific hazards models do **not** satisfy the relation

$$F_k(t) = \bigl\{ F_{k0}(t) \bigr\}^{\exp(\beta_k^\top X_k)}$$

that is, the cumulative incidence functions of different groups (e.g., treated vs untreated) are allowed to cross over

▷ this may or may not be reasonable

5.5 Competing Risks (cont'd)

- The Fine & Gray model was developed to provide parameters, which satisfy this relation for the cumulative incidence function, i.e.,

$$F_k(t) \; = \; \bigl\{ F_{k0}(t) \bigr\}^{\exp(\beta_k^\top X_k)}$$

- We will not cover this model further here

  ▷ more information can be found in Putter et al. (Stat. in Med., 2007)

5.5 Competing Risks (cont'd)

R> Cox regression for the cause-specific hazards is straightforward to implement using `coxph()` – for the PBC data set we used

```
coxph(Surv(years, status == "transplanted") ~ drug + age,
        data = pbc2.id)

coxph(Surv(years, status == "dead") ~ drug + age,
        data = pbc2.id)
```

5.6 Discrimination

- Often we are interested in assessing the discriminative capability of a covariate
  - ▷ can we use LDL cholesterol levels to discriminate between patients with low and high risk for heart disease
  - ▷ can we use PSA levels to discriminate between patients with low and high risk for prostate cancer
  - ▷ . . .

- $\boxed{\text{Example:}}$ in the AIDS data set we have seen that the baseline CD4 cell count is highly associated with the risk for the death
  - ▷ but how good is CD4 cell count in discriminating between patients?

5.6 Discrimination (cont'd)

- We denote the marker (e.g., CD4 cell count) by $M_i \Rightarrow$ for any threshold c we can define a prediction rule
  - ▷ if $M_i > c$, we classify patient i as a case (she had the event)
  - ▷ if $M_i \leq c$, we classify patient i as a non-case (she didn't have the event)

- We borrow ideas from standard ROC analysis – let D_i the case indicator, then
  - ▷ Sensitivity (true positive rate): Pr$(M_i > c \mid D_i = 1)$
  - ▷ Specificity (true negative rate): Pr$(M_i \leq c \mid D_i = 0)$

5.6 Discrimination (cont'd)

- To depict the discriminative capability of the marker for all possible threshold values c, we construct the Receiver Operating Characteristic (ROC) curve

$$\begin{aligned} TP(c) &= \mathsf{Pr}(M > c \mid D = 1) \\ FP(c) &= \mathsf{Pr}(M > c \mid D = 0) \end{aligned}$$

$$\text{ROC} \;:\; \{FP(c), TP(c)\}, \quad \text{for every } \; c$$

5.6 Discrimination (cont'd)

- To summarize the discriminative capability of the marker, we use the area under the ROC curve (AUC)

$$\text{AUC} = \int_0^1 \text{ROC}(p)\, dp$$

- Intuitive interpretation:
  - ▷ for a randomly chosen pair of patients $\{i, j\}$ where i is a case and j a control, the AUC is the probability that the marker value for the case is greater than the marker value for the control:

$$\text{AUC} = \text{Pr}(M_i > M_j \mid D_i = 1, D_j = 0)$$

5.6 Discrimination (cont'd)

- In survival analysis we have to account for time
  - ▷ the status of some patients changes at some time point

- Possible solution: consider the status of the patients at the end of the study

- Problems:
  - ▷ random right censoring, e.g., if a patient is lost to follow-up before the end of the study, her status is unknown at the end of the study
  - ▷ we lose the dynamic nature, e.g., considering discrimination at early stages may be more informative than at the end of the study

5.6 Discrimination (cont'd)

- Due to the time dimension, we can have more than one definitions for **Cases** and **Controls**

- For any time t, we can define **Cases** as
  - ▷ event (disease, death) before time t
  - ▷ event (disease, death) at time t

- For any time t, we can define **Controls** as
  - ▷ event-free through time t
  - ▷ event-free through a fixed follow-up time t^*

5.6 Discrimination (cont'd)

- In the following we follow the work of Heagerty, Lumley & Pepe (Biometrics, 2000)

- At time t, we define
  - ▷ **Case**: if a patient had the event at any time before $t \Rightarrow T_i^* \leq t$
  - ▷ **Control**: if a patient did not have the event by time $t \Rightarrow T_i^* > t$

- Features
  - ▷ at any time t, the entire population is classified as either case or a control
  - ▷ a patient plays the role of a control for all $t < T_i^*$, but she then contributes as a case for $t \geq T_i^*$

5.6 Discrimination (cont'd)

- We can now define the True Positive (sensitivity) and False Positive (1 – specificity) rates

$$
\begin{aligned}
TP(c, t) &= \Pr(M_i > c \mid T_i^* \leq t) \\
FP(c, t) &= \Pr(M > c \mid T_i^* > t)
\end{aligned}
$$

- The corresponding ROC and AUC are calculated in exactly the same way as in p.369 and p.370, respectively

- Note: now we have *time-dependent* sensitivity, specificity, ROCs and AUCs

5.6 Discrimination (cont'd)

- We have defined the accuracy measures in the survival context using T_i^*, i.e., the true event time

- However, due to censoring we actually only observe $\{T_i, \delta_i\}$

- Therefore, in estimating $TP(c, t)$ and $FP(c, t)$ we need to account for censoring

5.6 Discrimination (cont'd)

- To estimate both $TP(c, t)$ and $FP(c, t)$ we only require an estimate of the bivariate survival function

$$S(c, t) = \text{Pr}(M > c, T^* > t) = \int_0^\infty S(t \mid M = m) \, dF_M(m)$$

where $F_M(m)$ is the cdf for the marker M

- We will use the Nearest Neighbor Estimation method

$$\hat{S}_\lambda(c, t) = \frac{1}{n} \sum_i \hat{S}_\lambda(t \mid M = m_i) \, I(m_i > c)$$

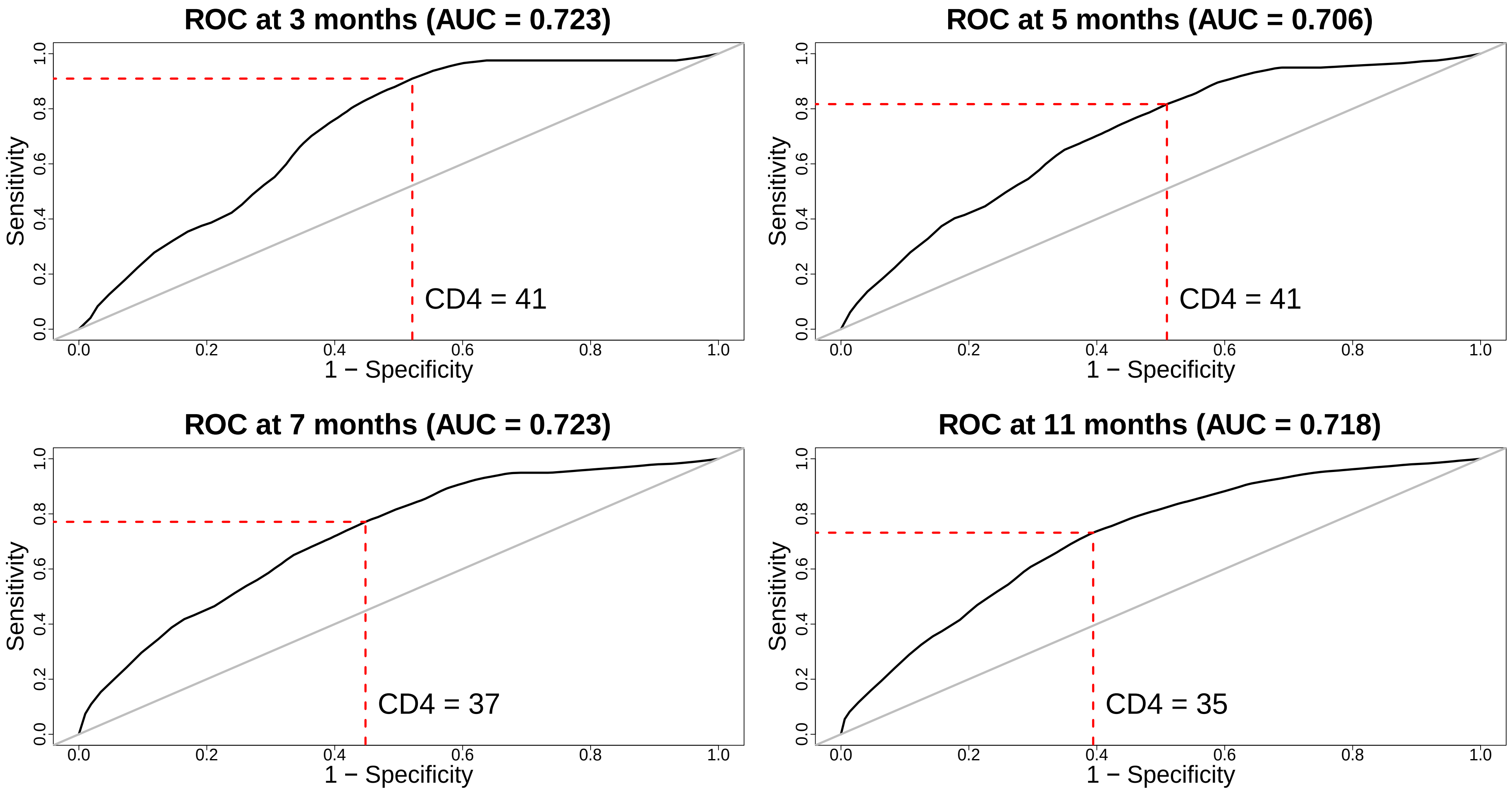
where $\hat{S}_\lambda(t \mid M = m_i)$ is a smooth estimator of the conditional survival function λ controls the degree of smoothness

5.6 Discrimination (cont'd)

- $\boxed{\text{Example:}}$ in the AIDS data set we have seen that the baseline CD4 cell count is highly associated with the risk for the death
  - ▷ but how good is CD4 cell count in discriminating between patients?

- Which follow-up time is most important?
  - ▷ medical importance
  - ▷ we illustrate time-dependent ROCs and AUCs at 3, 5, 7 and 11 months of follow-up
  - ▷ we also include an estimate of the 'optimal' cutoff point using the Youden index (Sens $+$ Spec $- 1$)

5.6 Discrimination (cont'd)


ROC at 3 months (AUC = 0.723)

ROC at 5 months (AUC = 0.706)

ROC at 7 months (AUC = 0.723)

ROC at 11 months (AUC = 0.718)

5.6 Discrimination (cont'd)

R> To calculate time-dependent ROC curves in we need to make use of function `survivalROC()` from package `survivalROC`

```
# ROC curve at 11 months
sroc <- with(aids.id,
    survivalROC(Time, death, -CD4, predict.time = 11,
        span = 0.55*length(Time)^(-0.20))
)

sroc$AUC # area under ROC curve

plot(sroc$FP, sroc$TP, type = "l", lwd = 2, ylab = "Sensitivity",
    xlab = "1 - Specificity")
abline(a = 0, b = 1, col = "grey", lwd = 1.7)
```

5.7 Review of Key Points

- Obtaining survival probabilities from a Cox model
  - ▷ the Cox model makes no assumption for the baseline survival function
  - ▷ in order to estimate survival probabilities this baseline survival function needs to be estimated
  - ▷ Breslow estimator (extension from the univariate case)

- Stratified Cox models
  - ▷ the baseline hazard for an event could be different between strata (e.g., hospitals)
  - ▷ categorical covariate do not satisfy the PH assumption
  - ▷ a simple extension of the Cox model is to consider a different baseline hazard per stratum
  - ▷ disadvantage: we obtain no p-value for the stratification factor

5.7 Review of Key Points (cont'd)

- Time-dependent covariates
  - ▷ in many cases we are interested in the effect of covariates whose values changes with time (e.g., time-dependent treatment dose, blood values, etc.)
  - ▷ important distinction: external vs internal time-dependent covariates
  - ▷ external time-dependent covariates can be easily handled within the framework of the extended Cox model
  - ▷ internal covariates are more difficult and require specialized statistical models → joint modelling of longitudinal and time-to-event data
  - ▷ if internal covariates are treated as external (using the extended Cox model) we may encounter spurious results

5.7 Review of Key Points (cont'd)

- Clustered event data
  - ▷ clustered event times occur frequently (e.g., patients within hospitals)
  - ▷ subjects in the same cluster are expected to be correlated
  - ▷ these correlations must be taken into account in the analysis
    - * marginal models (adjusted variance using jackknife)
    - * frailty models (latent variables)

5.7 Review of Key Points (cont'd)

- Competing risks
  - ▷ often in clinical studies we are interested in multiple endpoints
  - ▷ if the distributions of failure times from different causes are independent, then proceed as usual
  - ▷ if not, then more care is required: work with
    - * cause-specific hazards
    - * cumulative incidence function
    - * Cox model on cause-specific hazards (be careful of the derived interpretations)

5.7 Review of Key Points (cont'd)

- Discrimination
  - ▷ we aim at discriminating between patients of high and low risk of having the event
  - ▷ ROC methodology, estimating the True Positive and False Positive rates
  - ▷ we require special definitions in the survival setting due to censoring and time dimension
  - ▷ in the estimation of the accuracy measures we need to account for censoring

Part VI

Closing: Review of Key Points in Survival Analysis

6.1 Learning Objectives – Revisited

- We will learn which are the special characteristics of event time data and why they require special treatment (from a statistical point of view)

- From the course it will become clear
  - ▷ which statistical tools are applicable for this kind of data
  - ▷ which are their advantages and disadvantages
  - ▷ which are the optimal inferential strategies

- What is there further in survival analysis than what we will cover in this course

6.1 Learning Objectives – Revisited (cont'd)

- Time-to-event data exhibit
  - ▷ skewed distributions
  - ▷ censoring and/or truncation

- Statistical tools applicable to survival data
  - ▷ Kaplan-Meier estimate of the survival function
  - ▷ Log-rank and Peto & Peto modified Gehan-Wilcoxon tests can be used to test whether the survival functions of 2 groups differ statistically significantly
  - ▷ AFT and Cox models can be used to account for effect of more than one explanatory variables in the time-to-event

6.1 Learning Objectives – Revisited (cont'd)

- Modelling strategies
  - ▷ think carefully about the purpose of modelling (i.e., prediction, effect estimation or hypothesis testing)
  - ▷ consider which explanatory variables you want to include in the model
  - ▷ relax the linearity assumption of quantitative predictors (splines)
  - ▷ include meaningful interaction terms
  - ▷ use residuals to check model assumptions
  - ▷ use likelihood ratio tests for hypothesis testing
  - ▷ use effect plots to communicate the results of the model

6.1 Learning Objectives – Revisited (cont'd)

- Extending the Cox model
  - ▷ expected survival
  - ▷ stratification
  - ▷ time-dependent covariates (external & internal)
  - ▷ clustered event times (marginal & frailty models)
  - ▷ competing risks (dependent & independent)

6.1 Learning Objectives – Revisited (cont'd)

- The last three i.e.,

  ▷ time-dependent covariates,

  ▷ clustered event times, and

  ▷ competing risks

  have been briefly covered

- Their full treatment requires more advanced of theoretical statistics and therefore it falls outside the scope of this course

6.2 Closing

- By now you should have a clear view on the different survival analysis approaches, and how they should be used in practice.

- However, as we have seen throughout this course, statistical analysis is based on assumptions – if these assumptions are seriously violated, we may obtain spurious results

- This is especially the case when we deal with more complex models such as the one required for time-dependent covariates, clustered event times and competing risks

6.2 Closing (cont'd)

- Therefore, whenever you do not feel sure about the correct type of analysis for a specific research question at hand, consult a local statistician

The End!

Part VII

Appendix: R Code

R Code Appendix

Dimitris Rizopoulos
Erasmus University Medical Center

1. Introduction

This software appendix is based on the statistical programming language R. R is a free software environment for statistical computing and graphics, which was initiated in 1992 by Ross Ihaka and Robert Gentleman at University of Auckland Australia.

R can be freely downloaded from the CRAN website http://cran.r-project.org/ and installed in a wide variety of platforms including Windows, OS X, and various Unixes (for Windows point your browser at http://cran.r-project.org/bin/windows/base/).

For the examples presented in this document a number of these packages is required. These packages can be easily downloaded and installed by executing the following command within an R console (i.e., open R, type the following command and press 'Enter')[1]:

```
R> install.packages(c("Rcmdr", "RcmdrPlugin.survival", "JM", "msm", "survivalROC"),
+     dependencies = TRUE, repos = c(CRAN = "http://cran.r-project.org"))
```

Following we load all the packages that we will require for this session, and we import data sets that we will use for illustrations

```
R> library(nlme)
R> library(splines)
R> library(survival)
R> library(JM)
R> library(lattice)
R> library(msm)
R> # Note: the Renal Graft Failure data set used in the course
R> # is not publicly available
R> data("aids.id", package = "JM")
R> data("pbc2.id", package = "JM")
R> data("lung", package = "survival")
R> lung$sex <- factor(lung$sex, levels = 1:2, labels = c("Male", "Female"))
```

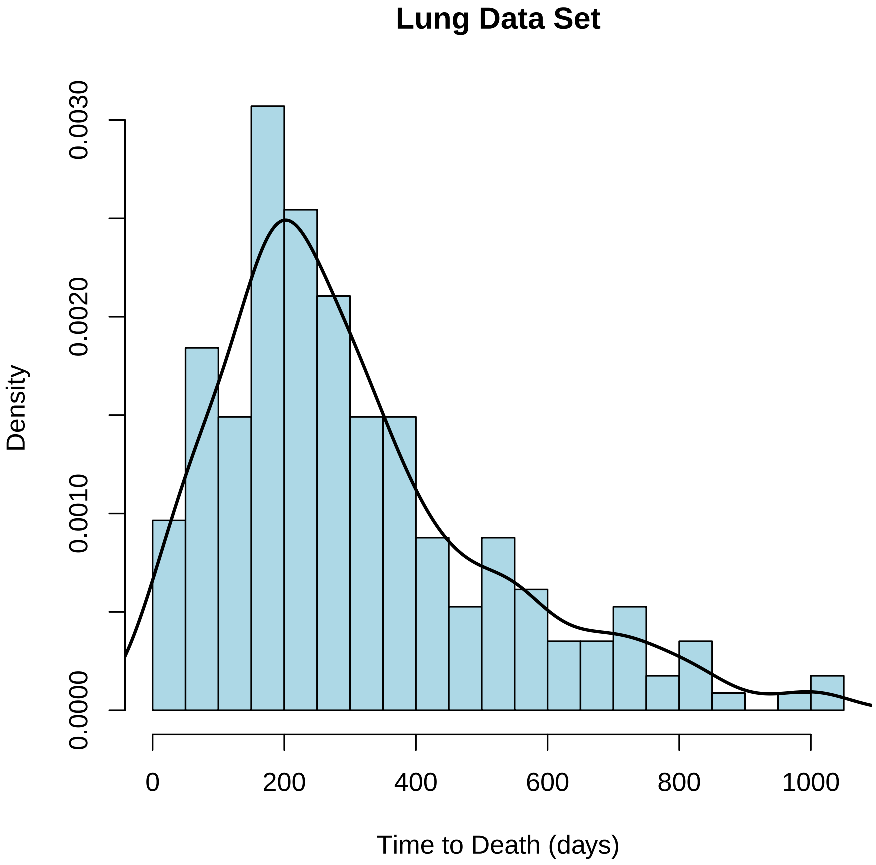
[1] All the code presented in this document starts with the symbol 'R>', which denotes evaluated R code. Moreover, symbol '+' denotes the continuation of the R code from the previous line. Both these symbols have to omitted when you wish to copy and paste this code into an R session. Finally, whatever falls after '#' is a comment and can be included in an R session.

2. Standard Survival Analysis

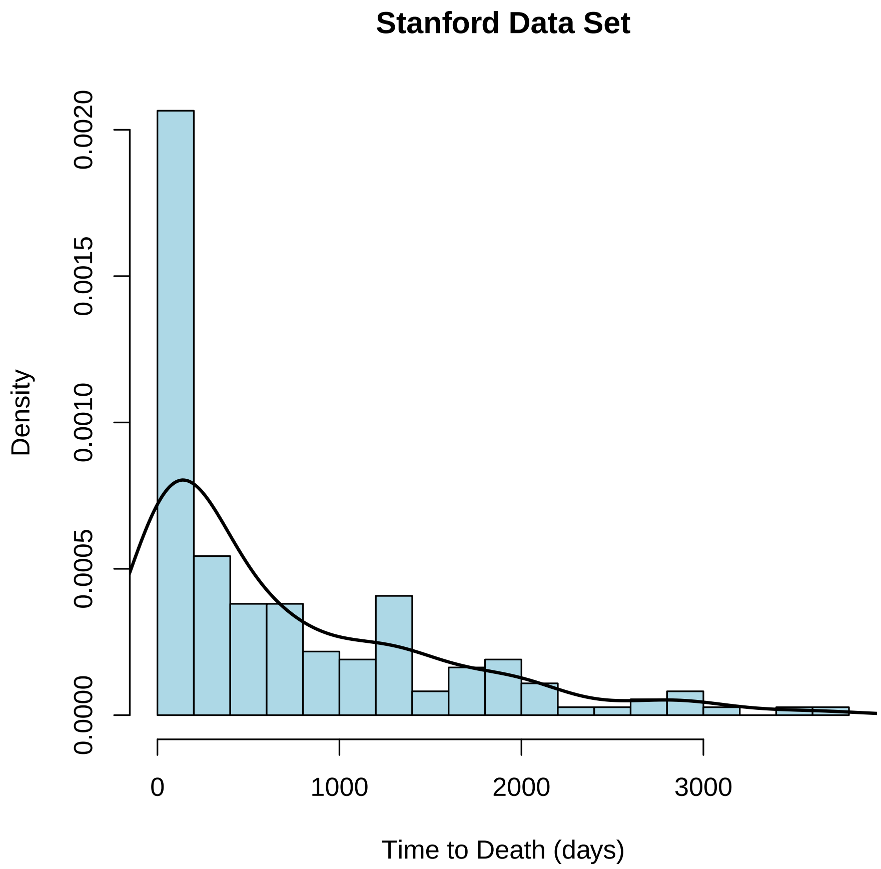
2.1. Histograms of Observed Event Times

Histograms for the observed failure times can be produced using the `hist()` function:

```
R> # Histogram Lung data set
R> hist(lung$time, xlab = "Time to Death (days)", main = "Lung Data Set",
+     breaks = 20, col = "lightblue", freq = FALSE)
R> lines(density(lung$time), lwd = 2)
```



```
R> # Histogram Stanford data set
R> hist(stanford2$time, xlab = "Time to Death (days)",
+     main = "Stanford Data Set", breaks = 20, col = "lightblue",
+     freq = FALSE)
R> lines(density(stanford2$time), lwd = 2)
```



Function `density()` is used to compute the superimposed Kernel density estimate.

2.2. Kaplan-Meier and Breslow Estimators

A key function that will be used in almost all computations that we will illustrate later is function `Surv()`. This function holds the available information for the failure times, that is the observed failure times and their corresponding censoring status (i.e., right, left, interval, counting). Its default use is for right censored data, i.e.,

```
R> # 'times' observed failure times
R> # 'event' indicator: '1' denotes true failure, '0' right censored
R> Surv(times, event)
```

For left censored data we need to specify the `type` argument as follows

```
R> # 'times' observed failure times
R> # 'event' indicator: '1' denotes true failure, '0' left censored
R> Surv(times, event, type = "left")
```

For interval censored we need to specify the left and right limits of the intervals in which an event occurred

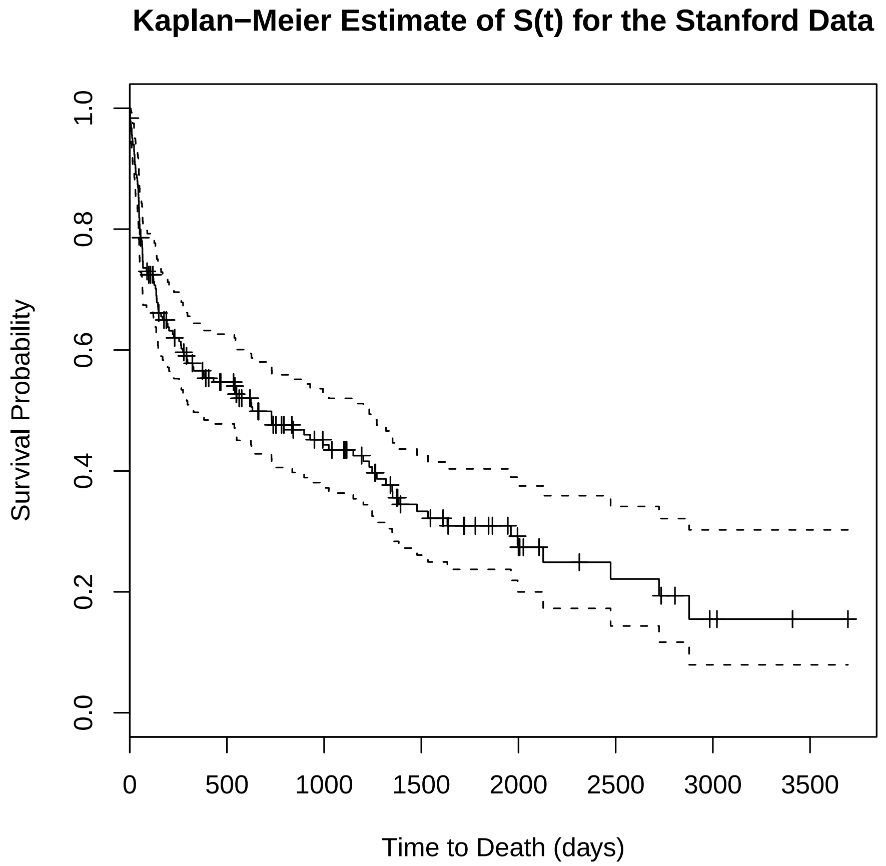
```
R> # 'timesL' left limit; should contain NA for left censored
R> # 'timesR' right limit; should contain NA for right censored
R> # for exact times 'timesL' and 'timesR' should coincide
R> Surv(timesL, timesR, type = "interval2")
```

Finally, for counting process notation we use a syntax similar to the interval censored case but specifying the `event` instead of the `type` argument, i.e.,

```
R> # 'start' left limit
R> # 'stop' right limit
R> # 'event' indicator: '1' if the event occurred at the end of the interval
R> Surv(start, stop, event)
```

The Kaplan-Meier estimator of the survival function is calculated by the `survfit()` function

```
R> fit <- survfit(Surv(time, status) ~ 1, data = stanford2)
R> plot(fit, xlab = "Time to Death (days)", ylab = "Survival Probability",
+     main = "Kaplan-Meier Estimate of S(t) for the Stanford Data")
```



The first argument of `survfit()` is a `formula` in which the right hand side specifies the event time information using `Surv()` and the right hand side could contains a categorical variable when we wish a different Kaplan-Meier estimate per level of the variable – in case we put `1` as above we calculate the Kaplan-Meier estimate based on the whole sample. The second argument `data` denotes the data frame in which these variables are stored. The `plot()` method plots the resulting estimate of the survival function. By default the plot contains the 95% pointwise confidence bands for the Kaplan-Meier estimate as well as small vertical lines denoting the time points at which we have censored observations.

Based on the output of `survfit()` we can also compute quantiles of survival times (e.g., the median survival), and their 95% confidence intervals. This is easily achieved with the following function:

```
R> quantile.survfit <- function (fit, probs = c(0.25, 0.5, 0.75)) {
+     f <- function (x) ifelse(x == 0, NA, x)
+     times <- rev(fit$time)
+     ind <- f(findInterval(probs, rev(fit$surv)))
+     indL <- f(findInterval(probs, rev(fit$lower)))
+     indU <- f(findInterval(probs, rev(fit$upper)))
+     cbind(Prob = probs, Time = times[ind],
+         "0.95LCL" = times[indL], "0.95UCL" = times[indU])
+ }
```

This function accepts as first argument the result of survfit(), i.e., an object of class 'survfit'. The second argument is a numeric vector of of probabilities for which we want to compute quantiles. For the Stanford we have

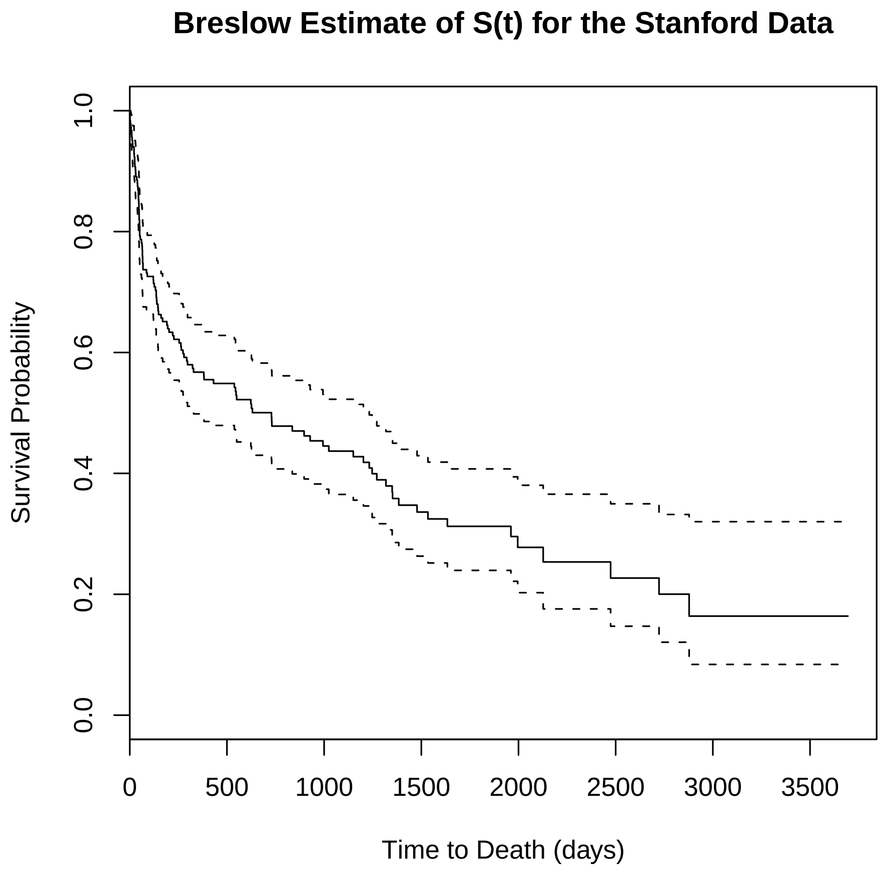
```
R> quantile(fit)
```

```
     Prob Time 0.95LCL 0.95UCL
[1,] 0.25 2127    1534      NA
[2,] 0.50  631     328    1232
[3,] 0.75   66      51     145
```

`NA` is returned whenever the corresponding quantile of its confidence limits are not available, based on the output of `survfit()`.

The Breslow estimator is produced similarly. The only difference in the call to `survfit()` is `type` argument where now we specify 'fleming-harrington'. In the `plot()` method and by setting argument `mark.time` to `FALSE`, we do not include the small vertical lines that denote censored observations.

```
R> fit <- survfit(Surv(time, status) ~ 1, data = stanford2,
+     type = "fleming-harrington")
R> plot(fit, mark.time = FALSE, xlab = "Time to Death (days)",
+     ylab = "Survival Probability",
+     main = "Breslow Estimate of S(t) for the Stanford Data")
```



2.3. Log-Rank and Gehan-Wilcoxon Tests

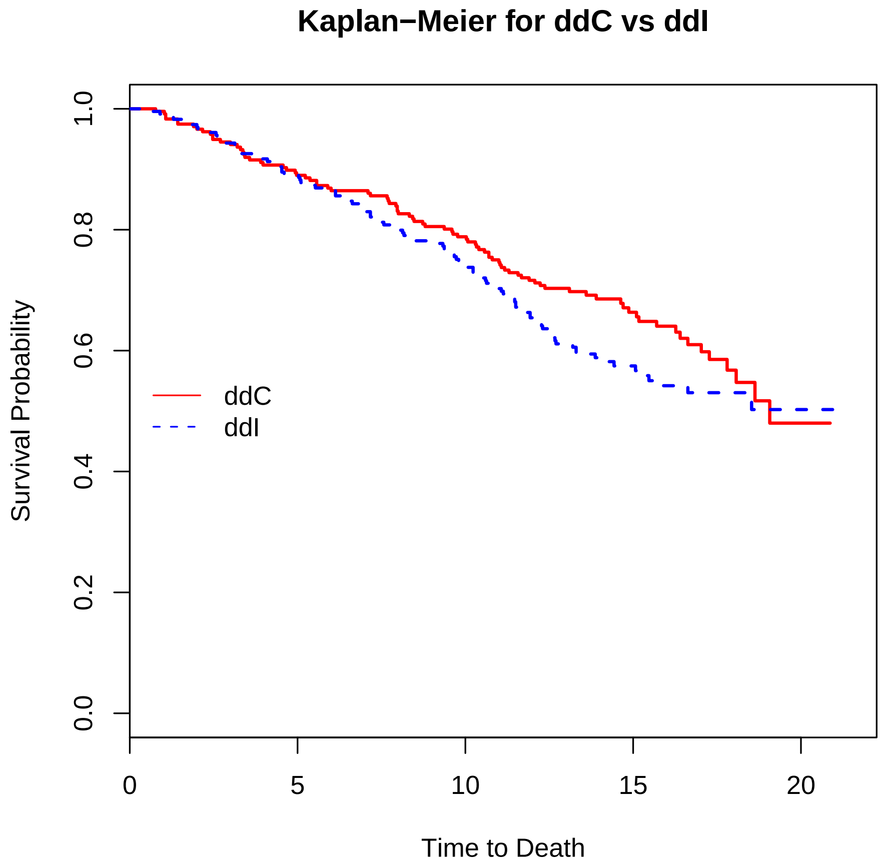
The log-rank test is calculated by function `survdiff()`, which similarly to function `survfit()` accepts a formula and a data frame. In this case the right hand side of the formula argument should contain the categorical variable that specifies t he different groups of patients. For example, for the AIDS data we are testing for differences in the survival curves of the two treatment groups by:

```
R> logrank <- survdiff(Surv(Time, death) ~ drug, data = aids.id)
R> logrank
```

```
Call:
survdiff(formula = Surv(Time, death) ~ drug, data = aids.id)

            N Observed Expected (O-E)^2/E (O-E)^2/V
drug=ddC 237       88     97.8      0.99      2.07
drug=ddI 230      100     90.2      1.08      2.07

 Chisq= 2.1  on 1 degrees of freedom, p= 0.15
```

The corresponding survival curves for the two groups are produced by:

```
R> fitKM <- survfit(Surv(Time, death) ~ drug, data = aids.id)
R> plot(fitKM, conf.int = FALSE, mark.time = FALSE, xlab = "Time to Death",
+      ylab = "Survival Probability", main = "Kaplan-Meier for ddC vs ddI",
+      lty = 1:2, col = c("red", "blue"), lwd = 2)
R> legend("left", c("ddC", "ddI"), lty = 1:2, col = c("red", "blue"),
+      bty = "n")
```



To calculate the Peto & Peto modified Gehan-Wilcoxon test we use again the `survdiff()` function but now we need to set argument `rho` to 1:

```
R> logrank <- survdiff(Surv(time, status == 2) ~ sex, data = lung)
R> peto.peto <- survdiff(Surv(time, status == 2) ~ sex, data = lung, rho = 1)
R> logrank
```

```
Call:
survdiff(formula = Surv(time, status == 2) ~ sex, data = lung)

             N Observed Expected (O-E)^2/E (O-E)^2/V
sex=Male   138      112     91.6      4.55      10.3
sex=Female  90       53     73.4      5.68      10.3

 Chisq= 10.3  on 1 degrees of freedom, p= 0.00131

R> peto.peto

Call:
survdiff(formula = Surv(time, status == 2) ~ sex, data = lung,
    rho = 1)

             N Observed Expected (O-E)^2/E (O-E)^2/V
sex=Male   138     70.4     55.6      3.95      12.7
sex=Female  90     28.7     43.5      5.04      12.7

 Chisq= 12.7  on 1 degrees of freedom, p= 0.000363
```

3. Accelerated Failure Time Models

3.1. Fitting

Accelerated Failure Time (AFT) models are fitted using function `survreg()`. The three main arguments are the `formula` argument that specifies the relationship between the observed failure times and covariates, the `data` argument that denotes the data frame that contains these variables, and the `dist` argument that specifies the distribution of true event times – note that this is different from what we have seen in the course where we specified the distribution of the logarithm of the event times. The default value for `dist` is `"weibull"`, which as the name implies corresponds to the Weibull model for the failure times. As an example, we fit AFT models for the time to death in the PBC data in which we control for the effects of 'drug', 'sex' and 'age' under the Weibull, exponential, log-normal and log-logistic distributions for the failure times, respectively:

```
R> fit.weibull <- survreg(Surv(years, status2) ~ drug + sex + age,
+     data = pbc2.id) # default is dist = "weibull"
R> fit.exp <- survreg(Surv(years, status2) ~ drug + sex + age,
+     data = pbc2.id, dist = "exponential")
R> fit.lnorm <- survreg(Surv(years, status2) ~ drug + sex + age,
+     data = pbc2.id, dist = "lognormal")
R> fit.llogis <- survreg(Surv(years, status2) ~ drug + sex + age,
+     data = pbc2.id, dist = "loglogistic")
R> summary(fit.weibull)
```

```
Call:
survreg(formula = Surv(years, status2) ~ drug + sex + age, data = pbc2.id)
                Value Std. Error      z        p
(Intercept)    4.1386    0.48260  8.576 9.85e-18
drugD-penicil  0.1305    0.15576  0.838 4.02e-01
sexfemale      0.4447    0.20147  2.207 2.73e-02
age           -0.0387    0.00793 -4.886 1.03e-06
Log(scale)    -0.1022    0.07423 -1.377 1.68e-01

Scale= 0.903

Weibull distribution
Loglik(model)= -494.7   Loglik(intercept only)= -511.8
        Chisq= 34.23 on 3 degrees of freedom, p= 1.8e-07
Number of Newton-Raphson Iterations: 5
n= 312

R> summary(fit.exp)

Call:
survreg(formula = Surv(years, status2) ~ drug + sex + age, data = pbc2.id,
    dist = "exponential")
               Value Std. Error      z        p
(Intercept)    4.328     0.5159  8.389 4.89e-17
drugD-penicil  0.144     0.1721  0.835 4.04e-01
sexfemale      0.482     0.2217  2.172 2.98e-02
age           -0.042     0.0084 -5.000 5.73e-07

Scale fixed at 1

Exponential distribution
Loglik(model)= -495.6   Loglik(intercept only)= -512.3
        Chisq= 33.35 on 3 degrees of freedom, p= 2.7e-07
Number of Newton-Raphson Iterations: 4
n= 312

R> summary(fit.lnorm)

Call:
survreg(formula = Surv(years, status2) ~ drug + sex + age, data = pbc2.id,
    dist = "lognormal")
                Value Std. Error     z        p
(Intercept)    4.4153    0.59602  7.41 1.28e-13
drugD-penicil  0.2120    0.18775  1.13 2.59e-01
sexfemale      0.3024    0.27761  1.09 2.76e-01
age           -0.0484    0.00944 -5.13 2.91e-07
Log(scale)     0.3471    0.06473  5.36 8.20e-08
```

```
Scale= 1.41

Log Normal distribution
Loglik(model)= -498.9   Loglik(intercept only)= -515.2
        Chisq= 32.72 on 3 degrees of freedom, p= 3.7e-07
Number of Newton-Raphson Iterations: 3
n= 312

R> summary(fit.llogis)

Call:
survreg(formula = Surv(years, status2) ~ drug + sex + age, data = pbc2.id,
    dist = "loglogistic")
                 Value Std. Error      z        p
(Intercept)    4.0002    0.54926  7.283 3.27e-13
drugD-penicil  0.1453    0.17387  0.836 4.03e-01
sexfemale      0.3845    0.24161  1.591 1.12e-01
age           -0.0415    0.00884 -4.694 2.67e-06
Log(scale)    -0.2613    0.07326 -3.566 3.62e-04

Scale= 0.77

Log logistic distribution
Loglik(model)= -497.4   Loglik(intercept only)= -513
        Chisq= 31.08 on 3 degrees of freedom, p= 8.2e-07
Number of Newton-Raphson Iterations: 4
n= 312
```

3.2. Effect Plots

To create effects plots that depict how the average failure time changes with the values of predictors we need to follow a four-step procedure. We will illustrate this procedure for the PBC data set using the example presented in the slides of the course. In particular, at *Step 1* we fit the AFT model that contains the predictors of main interest, and possibly interaction terms. For our example we include the additive effects of ‘drug’, ‘sex’ and ‘age’:

```
R> fit.lnorm <- survreg(Surv(years, status2) ~ drug + sex + age,
+         data = pbc2.id, dist = "lognormal")
```

At *Step 2* we create a data frame that contains the combination of values of these three predictors for which we would like to compute the average log failure time. In our example we are interested to see how the log failure time changes for female patients, as age increases, and separately for the two treatment groups. Therefore, we create the data frame:

```
R> ND <- with(pbc2.id, data.frame(
+      age  = rep(seq(min(age), max(age), length.out = 20), 2),
+       sex = rep("female", 40),
+       drug = gl(2, 20, labels = c("placebo", "D-penicil"))
+ ))
R> ND
```

```
        age    sex      drug
1  26.27861 female   placebo
2  29.02403 female   placebo
3  31.76945 female   placebo
4  34.51487 female   placebo
5  37.26029 female   placebo
6  40.00571 female   placebo
7  42.75113 female   placebo
8  45.49654 female   placebo
9  48.24196 female   placebo
10 50.98738 female   placebo
11 53.73280 female   placebo
12 56.47822 female   placebo
13 59.22364 female   placebo
14 61.96906 female   placebo
15 64.71448 female   placebo
16 67.45990 female   placebo
17 70.20532 female   placebo
18 72.95073 female   placebo
19 75.69615 female   placebo
20 78.44157 female   placebo
21 26.27861 female D-penicil
22 29.02403 female D-penicil
23 31.76945 female D-penicil
24 34.51487 female D-penicil
25 37.26029 female D-penicil
26 40.00571 female D-penicil
27 42.75113 female D-penicil
28 45.49654 female D-penicil
29 48.24196 female D-penicil
30 50.98738 female D-penicil
31 53.73280 female D-penicil
32 56.47822 female D-penicil
33 59.22364 female D-penicil
34 61.96906 female D-penicil
35 64.71448 female D-penicil
36 67.45990 female D-penicil
37 70.20532 female D-penicil
38 72.95073 female D-penicil
39 75.69615 female D-penicil
40 78.44157 female D-penicil
```

It is important to note that in the definition of factors in this new data frame we need to use the same levels as in the original data frame that was used to fit the model. Thus, in the definition of ‘sex’ in `ND` even though we are only interested in females we specify that in the or

At *Step 3* we compute the fitted log failure times and their standard error using the `predict()` method. In addition, based on these two quantities we can easily construct the asymptotic 95% pointwise confidence intervals for the survival times (by taking the exponent of the 95% CI for the log survival times) and include in the data frame:

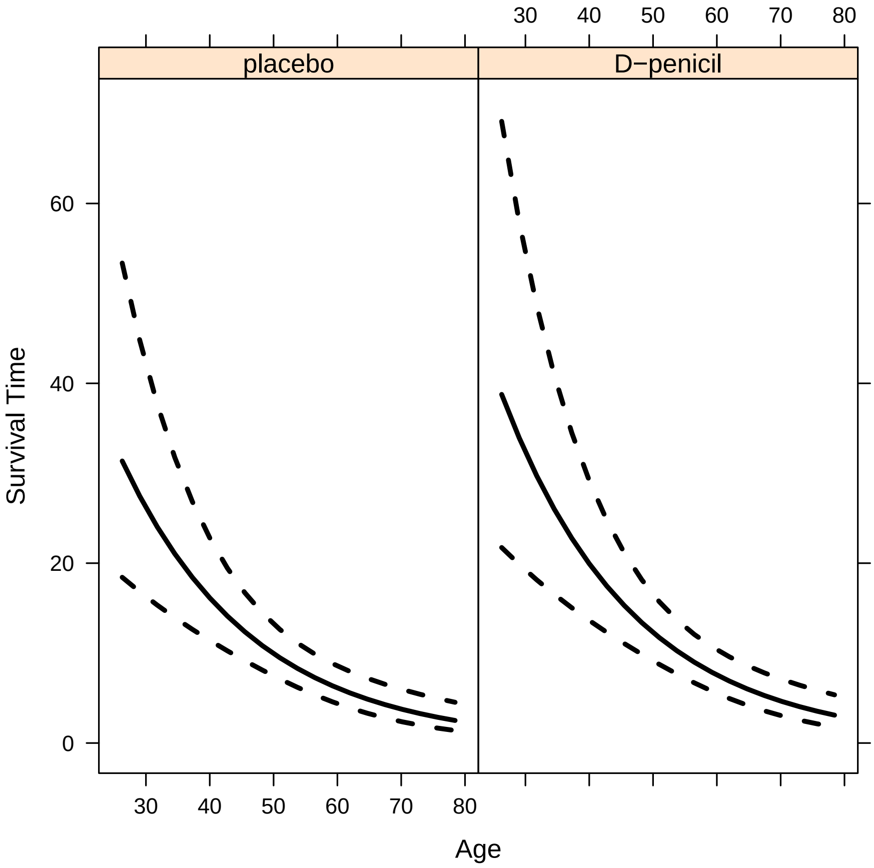
```
R> prs <- predict(fit.lnorm, ND, se.fit = TRUE, type = "lp")
R> ND$pred <- prs[[1]]
R> ND$se <- prs[[2]]
R> ND$lo <- exp(ND$pred - 1.96 * ND$se)
R> ND$up <- exp(ND$pred + 1.96 * ND$se)
R> ND$pred <- exp(ND$pred)
R> ND
```

```
        age    sex       drug      pred        se        lo        up
1  26.27861 female    placebo 31.362630 0.2712671 18.429148 53.372766
2  29.02403 female    placebo 27.459473 0.2498212 16.828294 44.806839
3  31.76945 female    placebo 24.042074 0.2292991 15.338701 37.683850
4  34.51487 female    placebo 21.049978 0.2099719 13.948257 31.767522
5  37.26029 female    placebo 18.430256 0.1922004 12.645241 26.861833
6  40.00571 female    placebo 16.136565 0.1764552 11.418509 22.804092
7  42.75113 female    placebo 14.128329 0.1633235 10.258103 19.458733
8  45.49654 female    placebo 12.370024 0.1534775  9.156465 16.711415
9  48.24196 female    placebo 10.830544 0.1475764  8.110184 14.463381
10 50.98738 female    placebo  9.482656 0.1460990  7.121443 12.626761
11 53.73280 female    placebo  8.302516 0.1491768  6.197662 11.122221
12 56.47822 female    placebo  7.269247 0.1565412  5.348584  9.879615
13 59.22364 female    placebo  6.364571 0.1676284  4.582273  8.840103
14 61.96906 female    placebo  5.572485 0.1817583  3.902412  7.957280
15 64.71448 female    placebo  4.878975 0.1982814  3.307867  7.196298
16 67.45990 female    placebo  4.271774 0.2166510  2.793774  6.531687
17 70.20532 female    placebo  3.740142 0.2364369  2.353037  5.944937
18 72.95073 female    placebo  3.274672 0.2573128  1.977601  5.422466
19 75.69615 female    placebo  2.867131 0.2790341  1.659315  4.954115
20 78.44157 female    placebo  2.510309 0.3014180  1.390449  4.532099
21 26.27861 female D-penicil 38.767156 0.2951195 21.739681 69.131297
22 29.02403 female D-penicil 33.942487 0.2729939 19.877720 57.958982
23 31.76945 female D-penicil 29.718260 0.2515919 18.149473 48.661192
24 34.51487 female D-penicil 26.019749 0.2311147 16.541479 40.929070
25 37.26029 female D-penicil 22.781527 0.2118305 15.040736 34.506155
26 40.00571 female D-penicil 19.946309 0.1940953 13.634692 29.179630
27 42.75113 female D-penicil 17.463942 0.1783717 12.311450 24.772814
28 45.49654 female D-penicil 15.290511 0.1652351 11.060406 21.138440
29 48.24196 female D-penicil 13.387569 0.1553430  9.873498 18.152331
```

```
30 50.98738 female D-penicil 11.721453 0.1493418  8.747001 15.707378
31 53.73280 female D-penicil 10.262689 0.1477062  7.683005 13.708541
32 56.47822 female D-penicil  8.985472 0.1505787  6.689069 12.070245
33 59.22364 female D-penicil  7.867208 0.1577132  5.775272 10.716890
34 61.96906 female D-penicil  6.888114 0.1685693  4.950069  9.584941
35 64.71448 female D-penicil  6.030871 0.1824841  4.217416  8.624097
36 67.45990 female D-penicil  5.280314 0.1988164  3.576217  7.796428
37 70.20532 female D-penicil  4.623166 0.2170210  3.021395  7.074105
38 72.95073 female D-penicil  4.047802 0.2366663  2.545451  6.436854
39 75.69615 female D-penicil  3.544043 0.2574227  2.139815  5.869776
40 78.44157 female D-penicil  3.102978 0.2790423  1.795780  5.361722
```

Finally at *Step 4* we plot the estimated effects using a suitable plotting function – here we produce a trellis plot using function `xyplot()`

```
R> xyplot(pred + lo + up ~ age | drug, data = ND, type = "l",
+     lty = c(1,2,2), col = "black", lwd = 3, xlab = "Age",
+     ylab = "Survival Time")
```



3.3. Hypothesis Testing

The Wald test for individual regression coefficients is included in the output of the `summary()` method:

```
R> fit <- survreg(Surv(Time, death) ~ gender * AZT, data = aids.id)
R> summary(fit)

Call:
survreg(formula = Surv(Time, death) ~ gender * AZT, data = aids.id)
                        Value Std. Error      z        p
(Intercept)            3.3693     0.2229 15.113 1.32e-51
gendermale             0.0866     0.2324  0.372 7.10e-01
AZTfailure            -0.9424     0.3399 -2.772 5.56e-03
gendermale:AZTfailure  0.4501     0.3546  1.269 2.04e-01
Log(scale)            -0.3248     0.0668 -4.861 1.17e-06

Scale= 0.723

Weibull distribution
Loglik(model)= -813   Loglik(intercept only)= -826.5
        Chisq= 26.86 on 3 degrees of freedom, p= 6.3e-06
Number of Newton-Raphson Iterations: 5
n= 467
```

For the likelihood ratio test we need to use function `anova()`. This accepts two arguments, namely the two fitted AFT models under the null and alternative hypotheses. For instance, with the following code we test whether the main effect of AZT and the interaction of AZT with gender are significantly different than zero:

```
R> fit.null <- survreg(Surv(Time, death) ~ gender, data = aids.id)
R> fit.alt <- survreg(Surv(Time, death) ~ gender * AZT, data = aids.id)
R> anova(fit.null, fit.alt)

         Terms Resid. Df    -2*LL            Test Df Deviance    P(>|Chi|)
1       gender       464 1652.360                 NA       NA           NA
2 gender * AZT       462 1626.049 +AZT+gender:AZT  2 26.31116 1.934656e-06
```

Note that the user is responsible to supply appropriately nested AFT models such that the LRT to be valid.

3.4. AFT Residuals

We compute the residuals of an AFT model using directly their definition. In particular, based on a fitted model

```
R> fit.weib <- survreg(Surv(Time, death) ~ drug, data = aids.id,
+     dist = "weibull")
```

we can compute the fitted linear predictor values, i.e., $\hat{\beta}_1 X_1 + \ldots + \hat{\beta}_p X_p$. To achieve this we will use the matrix notation $\hat{\beta}_1 X_1 + \ldots + \hat{\beta}_p X_p = X\hat{\beta}$, where X is the design matrix, which contains as columns variables $X_1, \ldots, X_p$, and $\hat{\beta}$ is the vector of estimated regression coefficients. In R the design matrix can be easily computed using function `model.matrix()` and the estimated coefficients can be extracted using the `coef()` method, i.e.,

```
R> fits <- c(model.matrix(~ drug, data = aids.id) %*% coef(fit.weib))
```

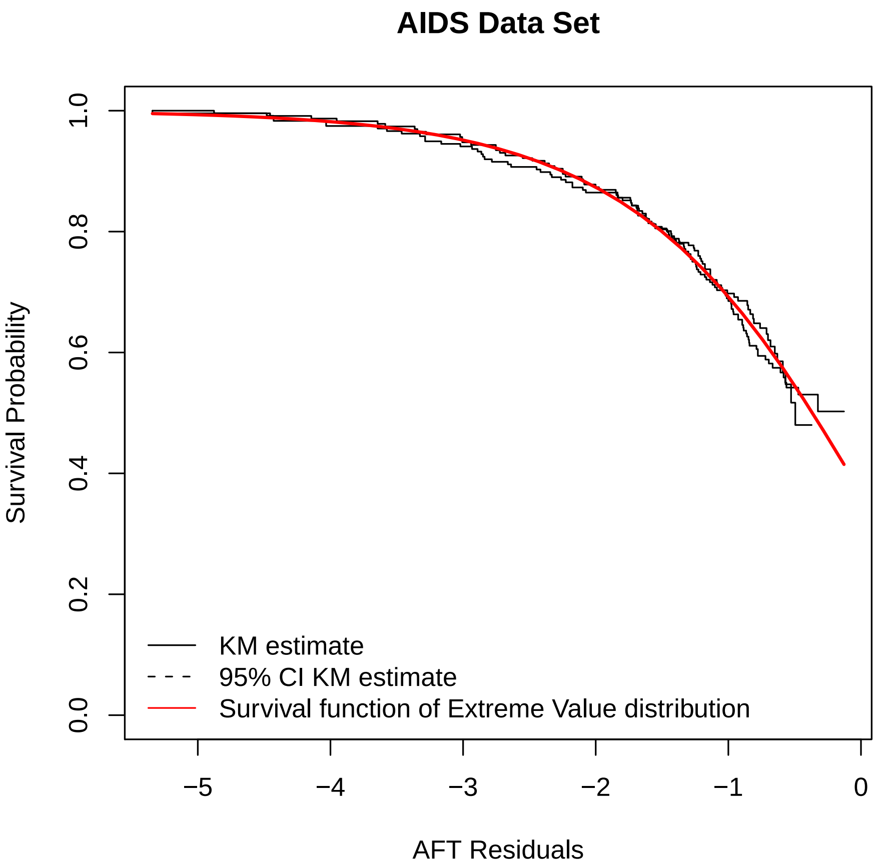
then we can compute the residuals as follows

```
R> resids <- (fit.weib$y[, 1] - fits) / fit.weib$scale
```

where `fit.weib$y[, 1]` extracts $\log T_i$ from the fitted AFT model, and `fit.weib$scale` is the estimate of the scale parameter σ.

Then we compute the Kaplan-Meier estimate of the survival function using `survfit()` as before and plot the result. Furthermore, we superimpose the survival function of the assumed extreme value distribution for the error terms

```
R> resKM <- survfit(Surv(resids, death) ~ drug, data = aids.id)
R> # we plot it, and
R> plot(resKM, mark.time = FALSE, xlab = "AFT Residuals",
+     ylab = "Survival Probability", main = "AIDS Data Set")
R> # we superimpose the survival function of the assumed
R> # extreme value distribution
R> xx <- seq(min(resids), max(resids), length.out = 35)
R> yy <- exp(- exp(xx))
R> lines(xx, yy, col = "red", lwd = 2)
R> # we also add a legend
R> legend("bottomleft", c("KM estimate", "95% CI KM estimate",
+     "Survival function of Extreme Value distribution"),
+     lty = c(1,2,1), col = c(1,1,2), bty = "n")
```



If you have fitted an AFT model assuming the normal distribution for the error terms (i.e., the log-normal distribution for T_i^*) and you are interested in testing this assumption, then above in place of `yy <- exp(- exp(xx))` you should use `yy <- pnorm(xx, lower.tail = FALSE)`. If you have fitted an AFT assuming the logistic distribution for the error terms (i.e., the log-logistic distribution for T_i^*) and you are interested in testing this assumption, then above you should use `yy <- plogis(xx, lower.tail = FALSE)`.

4. Cox Models

4.1. Fitting

The function that is used to fit Cox models is `coxph()`. The syntax is very similar to `survreg()`; however, now there is no `dist` argument since this is a semiparametric model and does not make any assumptions for the distribution of the baseline survival times

```
R> fit <- coxph(Surv(years, status2) ~ drug + sex + age, data = pbc2.id)
R> summary(fit)
```

```
Call:
coxph(formula = Surv(years, status2) ~ drug + sex + age, data = pbc2.id)

  n= 312

                   coef exp(coef)  se(coef)      z Pr(>|z|)
drugD-penicil -0.146013  0.864146  0.172143 -0.848   0.3963
sexfemale     -0.470905  0.624437  0.221785 -2.123   0.0337
age            0.042842  1.043773  0.008505  5.037 4.72e-07

              exp(coef) exp(-coef) lower .95 upper .95
drugD-penicil    0.8641      1.157    0.6167    1.2109
sexfemale        0.6244      1.601    0.4043    0.9644
age              1.0438      0.958    1.0265    1.0613

Rsquare= 0.101   (max possible= 0.991 )
Likelihood ratio test= 33.25  on 3 df,   p=2.854e-07
Wald test            = 34.87  on 3 df,   p=1.301e-07
Score (logrank) test = 35.31  on 3 df,   p=1.048e-07
```

4.2. Effect Plots

Effect plots for Cox models are constructed using exactly the same four-Step procedure as in AFT models. In particular, we start by fitting the Cox model that contains the desired effects of interest:

```
R> fit <- coxph(Surv(time, status) ~ age*sex + ph.karno, data = lung)
R> summary(fit)

Call:
coxph(formula = Surv(time, status) ~ age * sex + ph.karno, data = lung)

  n=227 (1 observation deleted due to missingness)

                   coef exp(coef)  se(coef)      z Pr(>|z|)
age            0.016371  1.016506  0.011429  1.432   0.1520
sexFemale      0.264047  1.302189  1.222922  0.216   0.8291
ph.karno      -0.013523  0.986568  0.005891 -2.296   0.0217
age:sexFemale -0.012109  0.987964  0.019313 -0.627   0.5307

              exp(coef) exp(-coef) lower .95 upper .95
age              1.0165     0.9838    0.9940     1.040
sexFemale        1.3022     0.7679    0.1185    14.310
ph.karno         0.9866     1.0136    0.9752     0.998
age:sexFemale    0.9880     1.0122    0.9513     1.026
```

```
Rsquare= 0.081   (max possible= 0.999 )
Likelihood ratio test= 19.21  on 4 df,   p=0.0007161
Wald test            = 19.69  on 4 df,   p=0.0005742
Score (logrank) test = 20.23  on 4 df,   p=0.00045
```

Following we create the data frame that contains the values of the covariates for which we would like to depict the log hazard ratios

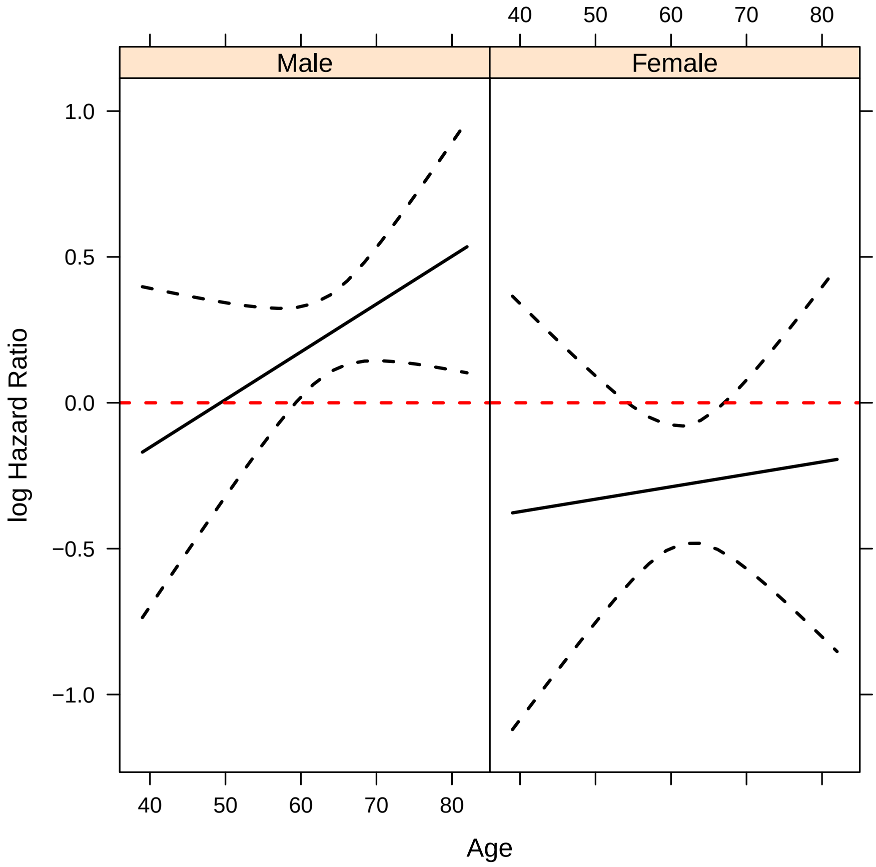
```
R> ND <- with(lung, data.frame(
+     age = rep(seq(min(age, na.rm = TRUE), max(age, na.rm = TRUE),
+         length.out = 20), 2),
+     sex = gl(2, 20, labels = c("Male", "Female")),
+     ph.karno = rep(median(ph.karno, na.rm = TRUE), 40)
+ ))
```

We continue by using function `predict()` to obtain estimates for the log hazard ratios with the associated standard errors. We also compute the asymptotic 95% pointwise confidence intervals and include them in the data frame

```
R> prs <- predict(fit, newdata = ND, type = "lp", se.fit = TRUE)
R> ND$pred <- prs[[1]]
R> ND$se <- prs[[2]]
R> ND$lo <- ND$pred - 1.96 * ND$se
R> ND$up <- ND$pred + 1.96 * ND$se
```

Finally, we create the effect plot using `xyplot()`

```
R> xyplot(pred + lo + up ~ age | sex, data = ND, panel = function (...) {
+     panel.xyplot(..., lty = c(1,2,2), col = "black", lwd = 2, type = "l")
+     panel.abline(h = 0, lwd = 2, lty = 2, col = "red")
+ }, xlab = "Age", ylab = "log Hazard Ratio")
```



4.3. Hypothesis Testing

Hypothesis testing for Cox models is performed with the same functions as for AFT models. In particular, the `summary()` method returns the Likelihood Ratio, Wald and Score tests for the null hypothesis that all regression coefficients are zero versus the alternative that at least one is different than zero. In the case we only include a categorical predictor, then the Score test is equivalent to the Log-Rank test.

```
R> fit <- coxph(Surv(years, status2) ~ sex, data = pbc2.id)
R> summary(fit)
```

```
Call:
coxph(formula = Surv(years, status2) ~ sex, data = pbc2.id)

  n= 312
```

```
             coef exp(coef) se(coef)      z Pr(>|z|)
sexfemale -0.6502    0.5219   0.2183 -2.979  0.00290

          exp(coef) exp(-coef) lower .95 upper .95
sexfemale    0.5219      1.916    0.3403    0.8006

Rsquare= 0.024   (max possible= 0.991 )
Likelihood ratio test= 7.73  on 1 df,   p=0.005445
Wald test            = 8.87  on 1 df,   p=0.002896
Score (logrank) test = 9.18  on 1 df,   p=0.002440
```

For the likelihood ratio test we need to use function `anova()`. This accepts two arguments, namely the two fitted Cox models under the null and alternative hypotheses. For instance, with the following code we test whether the main effect of treatment and the interaction of treatment with age are significantly different than zero:

```
R> fit.null <- coxph(Surv(years, status2) ~ age, data = pbc2.id)
R> fit.alt <- coxph(Surv(years, status2) ~ drug*age, data = pbc2.id)
R> anova(fit.null, fit.alt)
```

```
Analysis of Deviance Table
 Cox model: response is  Surv(years, status2)
 Model 1: ~ age
 Model 2: ~ drug * age
   loglik  Chisq Df P(>|Chi|)
1 -712.41
2 -711.29 2.2259  2    0.3286
```

4.4. Power

The following simple function calculates the number of events required to achieve a specific level of power based on the Log-Rank test

```
R> deaths <- function (beta, p, power, alpha = 0.05) {
+      # 'beta' the log hazard ratio
+      # 'p' the proportion of subjects in the treatment group
+      # 'power' the desired level of power
+      # 'alpha' the Type I error
+      q <- 1 - p
+      z.power <- qnorm(power)
+      ca <- abs(qnorm(alpha / 2))
+      round((ca + z.power)^2 / (p * q * beta^2))
+ }
```

The table presented in the slides was calculated as

```
R> betas <- rep(log(c(1.10, 1.2, 1.4, 1.6, 1.8, 2)), each = 4)
R> pws <- rep(c(0.5, 0.7, 0.8, 0.9), 6)
R> out <- mapply(deaths, beta = betas, power = pws, MoreArgs = list(p = 0.5))
R> dim(out) <- c(4, 6)
R> dimnames(out) <- list(power = c("50%", "70%", "80%", "90%"),
+     "Hazard Ratio" = c(1.10, 1.2, 1.4, 1.6, 1.8, 2))
R> out
```

```
     Hazard Ratio
power  1.1  1.2 1.4 1.6 1.8  2
  50% 1692  462 136  70  44 32
  70% 2718  743 218 112  71 51
  80% 3456  944 277 142  91 65
  90% 4627 1264 371 190 122 87
```

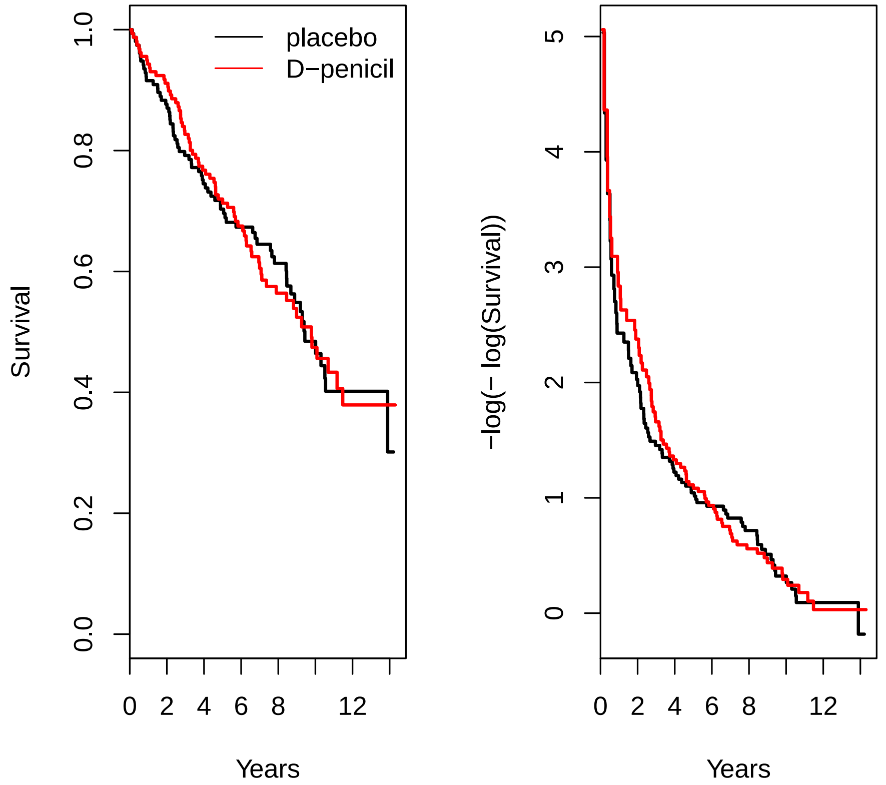
4.5. Testing PH Assumption – Categorical

For categorical covariates with few levels we can check the proportional hazards assumption visually by comparing the Kaplan-Meier estimates based on the subgroups of subjects defined by the different levels of these covariates. As mentioned in the slides, it is easier to spot violations the PH assumption by plotting the Kaplan-Meier estimates of the different groups on the $\log\{-\log \hat{S}(t)\}$ scale, because in this scale and under PH we expect observe approximately parallel lines. As an illustration we will graphically check the PH assumption for the treatment effect in the PBC data: first, we compute the Kaplan-Meier estimates for the two treatment groups using `survreg()`

```
R> fit <- survfit(Surv(years, status2) ~ drug, data = pbc2.id)
```

following we plot the Kaplan-Meier estimates of the two groups, in the original and log-log scale. For the latter we use the `fun` argument which specifies a functional transformation of the survival probabilities:

```
R> par(mfrow = c(1, 2))
R> plot(fit, mark.time = FALSE, xlab = "Years", ylab = "Survival", col = 1:2, lwd = 2)
R> legend("topright", c("placebo", "D-penicil"), lty = 1, col = 1:2, bty = "n")
R> plot(fit, mark.time = FALSE, fun = function (s) -log(-log(s)), xlab = "Years",
+     ylab = "-log(- log(Survival))", col = 1:2, lwd = 2)
```



4.6. Testing PH Assumption – Continuous

The test and the plot for checking the proportional hazards assumption for continuous predictors, based on scaled Schoenfeld residuals, are calculated by function `cox.zph()`. The two main arguments of this function are a fitted Cox model and the transformation of the time scale (default is the Kaplan-Meier transform of the time scale). As an illustration, we will check the PH assumption for Karnofsky score in the Lung data set. We start by fitting the model, and computing the scaled Schoenfeld residuals under three transformation of the time scale. The `print()` method outputs the p-values for the hypothesis of zero correlation between the scaled Schoenfeld residuals and the chosen transformation of the time scale:

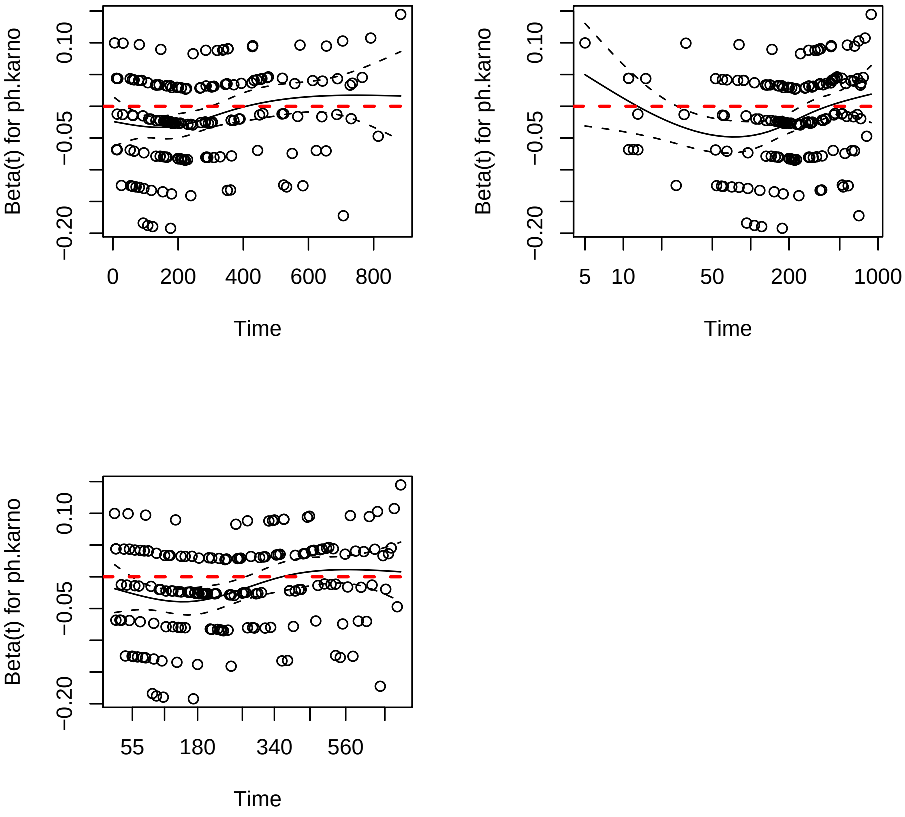
```
R> fit.ph <- coxph(Surv(time, status) ~ ph.karno, data = lung)
R> zph1 <- cox.zph(fit.ph, transform = "identity")
R> zph1
```

```
           rho chisq       p
ph.karno 0.234  8.08 0.00448
```

```
R> zph2 <- cox.zph(fit.ph, transform = "log")
R> zph2

           rho chisq      p
ph.karno 0.139  2.85 0.0913

R> zph3 <- cox.zph(fit.ph, transform = "km")
R> zph3

           rho chisq      p
ph.karno 0.232  7.95 0.0048
```

and the `plot()` method depicts the relation of the time-dependent coefficients $\beta(t)$ versus the chosen transformation of the time scale

```
R> par(mfrow = c(2, 2))
R> plot(zph1)
R> abline(h = 0, lty = 2, lwd = 2, col = "red")
R> plot(zph2)
R> abline(h = 0, lty = 2, lwd = 2, col = "red")
R> plot(zph3)
R> abline(h = 0, lty = 2, lwd = 2, col = "red")
```



4.7. Weibull PH versus Cox

As we have seen in Section 3.1 the Weibull can be fitted using function `survreg()`. However, remember that this function fits the Weibull model under the accelerated failure time formulation. To obtain parameter estimates that have a relative risk interpretation, we need to transform the AFT parameters. As we have seen in the slides, the relative risk parameters are the negative of the AFT parameters divided by the estimated scale parameter. We illustrate how we can apply this transformation for the Stanford data:

```
R> fit.AFT <- survreg(Surv(time, status) ~ age + t5, data = stanford2)
R> betasPH <- - coef(fit.AFT) / fit.AFT$scale
R> betasPH

(Intercept)         age          t5
-5.46735909  0.03147158  0.17375229
```

However, the calculation of standard errors of the transformed relative risk estimated parameter requires a few extra steps. In particular, due to the fact that the relative risk parameter are not a linear transformation of the AFT parameters we need to apply the Delta method.

This method is available in function `deltamethod()` of the **msm** package. For the Stanford data we have:

```
R> # covariance matrix of the AFT parameter estimates -- note that
R> # we obtain the variance and covariances for the logarithm of
R> # the scale parameter
R> Var.betasAFT <- vcov(fit.AFT)
R> # we define the transformation: 'x1', 'x2' and 'x3' denote the
R> # AFT parameters, 'x4' the scale parameter
R> transf <- list(~ - x1 / exp(x4), ~ - x2 / exp(x4), ~ - x3 /
+ exp(x4))
R> # we give these arguments to deltamethod()
R> se.betasPH <- deltamethod(transf, c(coef(fit.AFT),
+ log(fit.AFT$scale)),
+       Var.betasAFT)
```

We compare the derived transformed parameter estimates and standard errors, with the results of the Cox model

```
R> fit.cph <- coxph(Surv(time, status) ~ age + t5, data = stanford2)
R> summary(fit.cph)

Call:
coxph(formula = Surv(time, status) ~ age + t5, data = stanford2)

  n=157 (27 observations deleted due to missingness)

       coef exp(coef) se(coef)     z Pr(>|z|)
age 0.02961   1.03006  0.01136 2.608  0.00911
t5  0.17041   1.18579  0.18326 0.930  0.35243

    exp(coef) exp(-coef) lower .95 upper .95
age     1.030     0.9708     1.007     1.053
t5      1.186     0.8433     0.828     1.698

Rsquare= 0.053   (max possible= 0.997 )
Likelihood ratio test= 8.47  on 2 df,   p=0.01449
Wald test            = 7.81  on 2 df,   p=0.02018
Score (logrank) test = 7.87  on 2 df,   p=0.01950

R> cbind("coef" = betasPH, "se(coef)" = se.betasPH)

                   coef   se(coef)
(Intercept) -5.46735909 0.65558318
age          0.03147158 0.01134276
t5           0.17375229 0.18254109
```

from which observe very small differences.

4.8. Nonlinear Terms

Two possible ways to account for the nonlinear effects of continuous covariate are polynomials and regression splines (there are also other approaches, such as penalized splines, local regression, etc. but we will not discuss them here). Polynomial effects can be directly included using the `I()` function, e.g.,

```
R> coxph(Surv(years, status2) ~ serBilir + I(serBilir^2) + I(serBilir^3),
+      data = pbc2.id)

Call:
coxph(formula = Surv(years, status2) ~ serBilir + I(serBilir^2) +
    I(serBilir^3), data = pbc2.id)


                   coef exp(coef) se(coef)     z       p
serBilir       0.594603     1.812 0.090450  6.57 4.9e-11
I(serBilir^2) -0.031959     0.969 0.009191 -3.48 5.1e-04
I(serBilir^3)  0.000563     1.001 0.000244  2.31 2.1e-02

Likelihood ratio test=127  on 3 df, p=0  n= 312
```

postulates that the log hazard for death is affected by the linear, quadratic and cubic effects of serum Bilirubin. Due to numerical reasons it is advisable instead of standard polynomials to use orthogonal polynomials using function `poly()`. The obtained estimated regression coefficients will be different than using `I()` but the fit of the model is exactly the same. Thus, the equivalent Cox model fit that account for the 3 order degree polynomials effects of serum Bilirubin is

```
R> coxph(Surv(years, status2) ~ poly(serBilir, 3), data = pbc2.id)

Call:
coxph(formula = Surv(years, status2) ~ poly(serBilir, 3), data = pbc2.id)


                    coef exp(coef) se(coef)     z       p
poly(serBilir, 3)1 15.82  7.39e+06     1.38 11.50 0.0e+00
poly(serBilir, 3)2 -7.60  5.03e-04     1.20 -6.32 2.6e-10
poly(serBilir, 3)3  2.52  1.24e+01     1.09  2.31 2.1e-02

Likelihood ratio test=127  on 3 df, p=0  n= 312
```

To model nonlinear effects with regression splines we similarly include a splines basis in the linear predictor of the desired survival model. Continuing the example above, we include a natural cubic splines basis with 3 degrees of freedom for serum Bilirubin using function `ns()` from package **splines**:

```
R> coxph(Surv(years, status2) ~ ns(serBilir, 3), data = pbc2.id)

Call:
coxph(formula = Surv(years, status2) ~ ns(serBilir, 3), data = pbc2.id)


                 coef exp(coef) se(coef)    z       p
ns(serBilir, 3)1 3.93      51.0    0.489 8.03 8.9e-16
ns(serBilir, 3)2 5.87     352.5    0.762 7.70 1.4e-14
ns(serBilir, 3)3 3.53      34.2    0.585 6.03 1.6e-09

Likelihood ratio test=134  on 3 df, p=0  n= 312
```

In order to compare visually the the Cox model that restricts the effect of serum Bilirubin to be linear and the one that allows for possible nonlinearities we will use an Effect Plot. First, we fit the two Cox models

```
R> # linear effect of 'serBilir' on the log hazard ratio
R> fit.lin <- coxph(Surv(years, status2) ~ serBilir, data = pbc2.id)
R> # nonlinear effect of 'serBilir' on the log hazard ratio using
R> # natural cubic splines with 3 degrees of freedom
R> fit.nlin <- coxph(Surv(years, status2) ~ ns(serBilir, ), data = pbc2.id)
```

Following we create a data frame that contains the sequence of values of serum Bilirubin for which we wish to calculate the log hazard ratio:

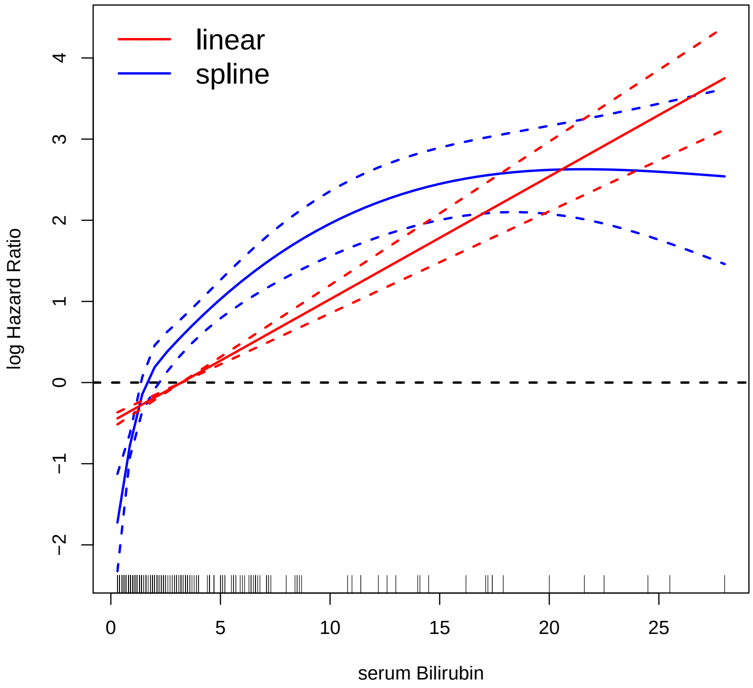
```
R> ND <- with(pbc2.id, data.frame(serBilir = seq(min(serBilir), max(serBilir),
+     length.out = 50)))
```

Then, using the `predict()` method, we calculate the fitted log hazard ratios and their standard errors based on the two Cox models, and construct their asymptotic 95% pointwise confidence intervals:

```
R> # for the linear Cox model
R> prs.l <- predict(fit.lin, ND, se.fit = TRUE, type = "lp")
R> ND$pred.l <- prs.l[[1]]
R> ND$se.l <- prs.l[[2]]
R> ND$lo.l <- ND$pred.l - 1.96 * ND$se.l
R> ND$up.l <- ND$pred.l + 1.96 * ND$se.l
R> # for the nonlinear Cox model
R> prs.nl <- predict(fit.nlin, ND, se.fit = TRUE, type = "lp")
R> ND$pred.nl <- prs.nl[[1]]
R> ND$se.nl <- prs.nl[[2]]
R> ND$lo.nl <- ND$pred.nl - 1.96 * ND$se.nl
R> ND$up.nl <- ND$pred.nl + 1.96 * ND$se.nl
```

Finally, we plot the results by first creating an empty plot with appropriate ranges for the x- and y-axis, and add to it lines that represent the linear and nonlinear fitted log hazard ratios for serum Bilirubin along with their corresponding 95% confidence intervals:

```
R> plot(range(ND$serBilir), range(ND$lo.l, ND$up.l, ND$lo.nl, ND$up.nl),
+       type = "n", xlab = "serum Bilirubin", ylab = "log Hazard Ratio")
R> abline(h = 0, lty = 2.5, lwd = 2)
R> lines(ND$serBilir, ND$pred.nl, col = "blue", lwd = 2)
R> lines(ND$serBilir, ND$lo.nl, lty = 2, col = "blue", lwd = 2)
R> lines(ND$serBilir, ND$up.nl, lty = 2, col = "blue", lwd = 2)
R> lines(ND$serBilir, ND$pred.l, col = "red", lwd = 2)
R> lines(ND$serBilir, ND$lo.l, lty = 2, col = "red", lwd = 2)
R> lines(ND$serBilir, ND$up.l, lty = 2, col = "red", lwd = 2)
R> rug(pbc2.id$serBilir)
R> legend("topleft", c("linear", "spline"), lty = 1, lwd = 2,
+      col = c("red", "blue"), cex = 1.5, bty = "n")
```


linear
spline
log Hazard Ratio
−2
−1
0
1
2
3
4
0
5
10
15
20
25
serum Bilirubin

5. Extensions of the Cox Model

5.1. Expected Survival

The calculation of survival probabilities from a Cox model proceeds in a similar manner to Effect plots. In particular, we start by fitting the desired Cox model – here a Cox model for the AIDS data set, which includes the effects of the square root of the baseline CD4 cell counts measurement, treatment and gender.

```
R> fit <- coxph(Surv(Time, death) ~ sqrt(CD4) + drug + gender, data = aids.id)
```

Following we specify a data frame that contains the specific combination of covariate values for which we want to obtain survival probabilities estimates. For the AIDS data set, we are interested in 5 and 10 month survival of male and female patients, respectively, who received 'ddI', and have the median value for the square root baseline CD4 cell count.

```
R> ND <- with(aids.id, data.frame(
+       CD4 = c(median(CD4[gender == "male"]), median(CD4[gender == "female"])),
+       drug = c("ddI", "ddI"),
+       gender = c("male", "female")
+ ))
R> ND
```

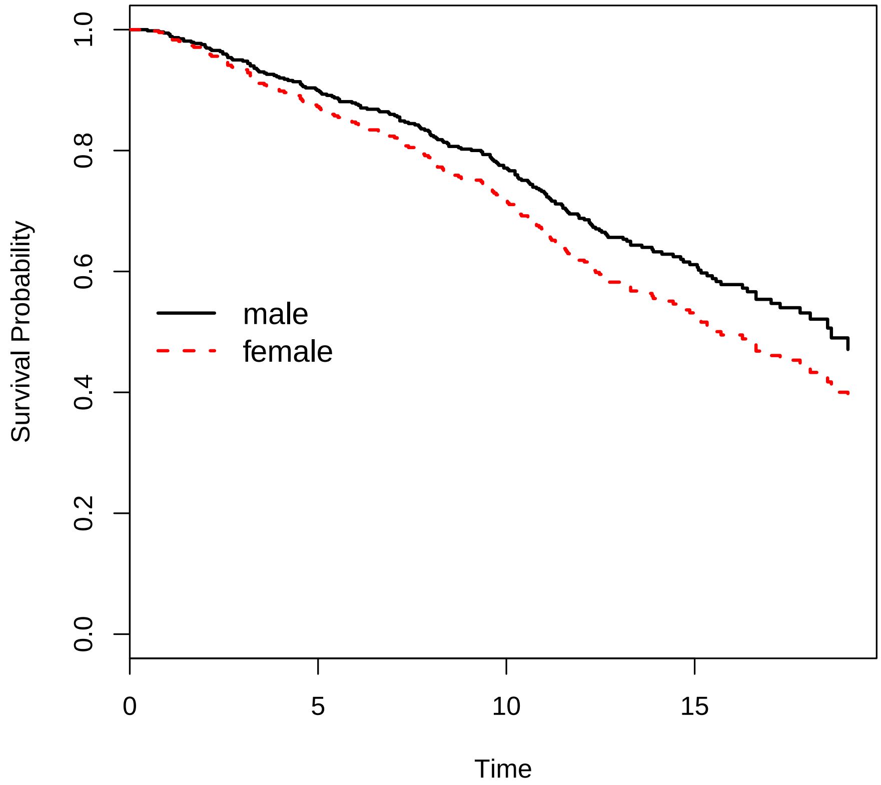
```
       CD4 drug gender
1 6.082763  ddI   male
2 5.744563  ddI female
```

As we have seen in the slides, in order to calculate survival probabilities, we need first to calculate the Breslow estimator of the baseline survival function (due to the fact that the Cox model makes no assumptions about this baseline survival function). This procedure is implemented by function `survfit()`, which contrary to its use to calculate the Kaplan-Meier estimate, accepts now as first argument a fitted Cox model, and as second the data frame of covariate values.

```
R> sfit <- survfit(fit, newdata = ND)
```

The output of `survfit()` is in fact a survival function for each combination of covariate values, i.e., for each row of the data frame `ND`. These survival functions can be depicted using the `plot()` method:

```
R> plot(sfit, col = 1:2, lty = 1:2, lwd = 2, xlab = "Time",
+       ylab = "Survival Probability")
R> legend("left", c("male", "female"), col = 1:2, lty = 1:2, lwd = 2,
+       cex = 1.2, bty = "n")
```



If we wish to extract survival probabilities estimates for specific follow-up times, then we can use the `summary()` method and appropriately specify the `times` argument. We the following code we obtain the 5 and 10 month survival along with the associated 95% pointwise confidence intervals, which we manipulate in order to include in the data frame with the covariate values:

```
R> sum.sfit <- summary(sfit, times = c(5, 10))
R> vals <- sum.sfit$surv
R> low <- sum.sfit$lower
R> up <- sum.sfit$upper
R> out <- rbind(ND, ND)
R> out$times <- gl(2, 2, labels = c("5 months", "10 months"))
R> out$surv <- c(t(vals))
R> out$lower <- c(t(low))
R> out$upper <- c(t(up))
R> out
```

```
       CD4 drug gender     times      surv     lower     upper
1 6.082763  ddI   male  5 months 0.8994630 0.8694965 0.9304622
2 5.744563  ddI female  5 months 0.8727836 0.8120357 0.9380761
```

```
3 6.082763  ddI   male 10 months 0.7710097 0.7232519 0.8219211
4 5.744563  ddI female 10 months 0.7160862 0.6074357 0.8441707
```

5.2. Stratified Cox model

The stratified Cox model is fitted by specifying the stratification variable in the `formula` argument of `coxph()`, which is done using function `strata()`. In cases we are interested in more than one stratification variables, then these are defined as extra arguments in `strata()`. In addition, interactions terms of the stratification variables with the other covariates in the model are included in the standard way. The following examples illustrate these possibilities:

```
R> # no stratification
R> fit1 <- coxph(Surv(years, status2) ~ age + log(serBilir), data = pbc2.id)
R> summary(fit1)

Call:
coxph(formula = Surv(years, status2) ~ age + log(serBilir), data = pbc2.id)

  n= 312

                  coef exp(coef) se(coef)     z Pr(>|z|)
age           0.044988  1.046015 0.007485  6.01 1.85e-09
log(serBilir) 1.091080  2.977487 0.092002 11.86  < 2e-16

              exp(coef) exp(-coef) lower .95 upper .95
age               1.046     0.9560     1.031     1.061
log(serBilir)     2.977     0.3359     2.486     3.566

Rsquare= 0.415   (max possible= 0.991 )
Likelihood ratio test= 167.2  on 2 df,   p=0
Wald test            = 173.9  on 2 df,   p=0
Score (logrank) test = 198.3  on 2 df,   p=0

R> # we stratify be edema
R> fit2 <- coxph(Surv(years, status2) ~ age + log(serBilir) + strata(edema),
+     data = pbc2.id)
R> summary(fit2)

Call:
coxph(formula = Surv(years, status2) ~ age + log(serBilir) +
    strata(edema), data = pbc2.id)

  n= 312

                 coef exp(coef) se(coef)      z Pr(>|z|)
age           0.04453   1.04553  0.00788  5.651 1.60e-08
```

```
log(serBilir) 0.98218   2.67026  0.09749 10.075  < 2e-16

              exp(coef) exp(-coef) lower .95 upper .95
age               1.046     0.9565     1.030     1.062
log(serBilir)     2.670     0.3745     2.206     3.232

Rsquare= 0.329   (max possible= 0.978 )
Likelihood ratio test= 124.5  on 2 df,   p=0
Wald test            = 128.1  on 2 df,   p=0
Score (logrank) test = 135.3  on 2 df,   p=0

R> # we stratify by edema and sex
R> fit3 <- coxph(Surv(years, status2) ~ age + log(serBilir) +
+     strata(edema, sex), data = pbc2.id)
R> summary(fit3)

Call:
coxph(formula = Surv(years, status2) ~ age + log(serBilir) +
    strata(edema, sex), data = pbc2.id)

  n= 312

                  coef exp(coef) se(coef)      z Pr(>|z|)
age           0.044256  1.045249 0.008745  5.061 4.17e-07
log(serBilir) 1.066457  2.905069 0.104954 10.161  < 2e-16

              exp(coef) exp(-coef) lower .95 upper .95
age               1.045     0.9567     1.027     1.063
log(serBilir)     2.905     0.3442     2.365     3.569

Rsquare= 0.322   (max possible= 0.966 )
Likelihood ratio test= 121.2  on 2 df,   p=0
Wald test            = 117.5  on 2 df,   p=0
Score (logrank) test = 129.0  on 2 df,   p=0

R> # interaction of log serum Bilirubin with the stratifying variable
R> # edema
R> fit4 <- coxph(Surv(years, status2) ~ age + log(serBilir) * strata(edema),
+     data = pbc2.id)
R> summary(fit4)

Call:
coxph(formula = Surv(years, status2) ~ age + log(serBilir) *
    strata(edema), data = pbc2.id)

  n= 312
```

```
                                                              coef exp(coef)
age                                                       0.043140  1.044085
log(serBilir)                                             1.082141  2.950990
log(serBilir):strata(edema)edema=edema no diuretics      -0.022451  0.977799
log(serBilir):strata(edema)edema=edema despite diuretics -0.752747  0.471071
                                                          se(coef)      z
age                                                       0.007759  5.560
log(serBilir)                                             0.117971  9.173
log(serBilir):strata(edema)edema=edema no diuretics       0.240803 -0.093
log(serBilir):strata(edema)edema=edema despite diuretics  0.259289 -2.903
                                                         Pr(>|z|)
age                                                       2.7e-08
log(serBilir)                                             < 2e-16
log(serBilir):strata(edema)edema=edema no diuretics       0.92572
log(serBilir):strata(edema)edema=edema despite diuretics  0.00369

                                                         exp(coef) exp(-coef)
age                                                         1.0441     0.9578
log(serBilir)                                               2.9510     0.3389
log(serBilir):strata(edema)edema=edema no diuretics         0.9778     1.0227
log(serBilir):strata(edema)edema=edema despite diuretics    0.4711     2.1228
                                                         lower .95 upper .95
age                                                         1.0283     1.060
log(serBilir)                                               2.3418     3.719
log(serBilir):strata(edema)edema=edema no diuretics         0.6099     1.568
log(serBilir):strata(edema)edema=edema despite diuretics    0.2834     0.783

Rsquare= 0.345   (max possible= 0.978 )
Likelihood ratio test= 131.9  on 4 df,   p=0
Wald test            = 138.6  on 4 df,   p=0
Score (logrank) test = 154.1  on 4 df,   p=0
```

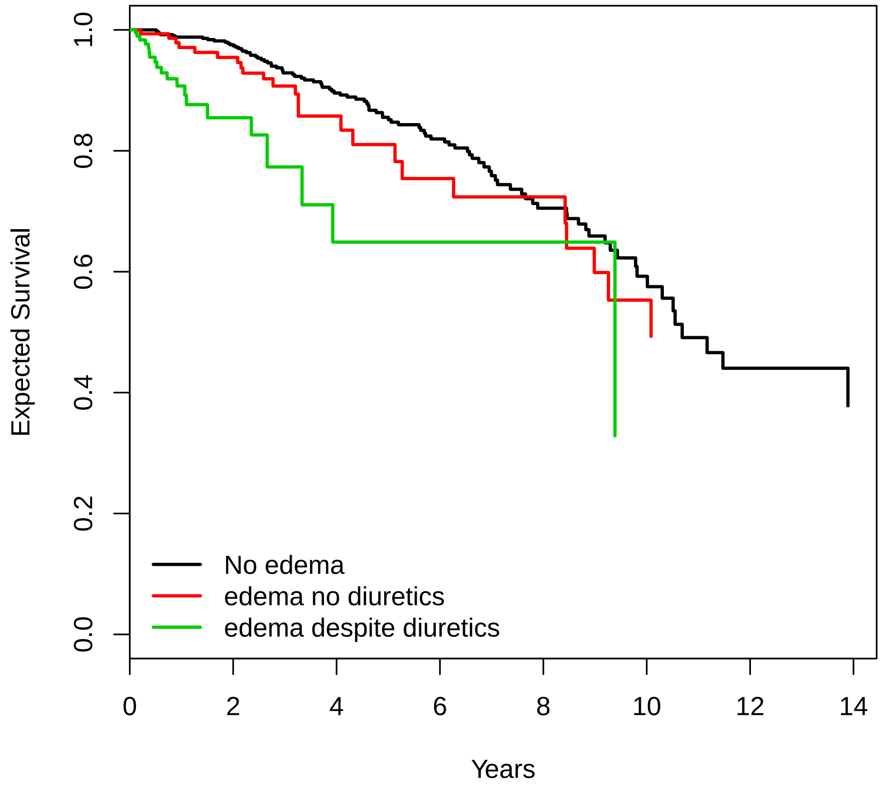
The omnibus *p*-value for the interaction terms in model `fit4` is obtained using the `anova` method

```
R> anova(fit4)
```

```
Analysis of Deviance Table
 Cox model: response is Surv(years, status2)
Terms added sequentially (first to last)

                             loglik    Chisq Df Pr(>|Chi|)
NULL                        -596.31
age                         -585.15  22.3054  1  2.326e-06
log(serBilir)               -534.08 102.1489  1  < 2.2e-16
log(serBilir):strata(edema) -530.34   7.4761  2     0.0238
```

Finally, estimates of survival probabilities in stratified Cox models are obtained using exactly the same syntax as in standard Cox models. The only difference is that we now obtain a separate survival curve per stratum:

```
R> ND <- data.frame(age = 40, serBilir = 2)
R> sfit <- survfit(fit2, newdata = ND)
R> plot(sfit, xlab = "Years", ylab = "Expected Survival", col = 1:3,
+     lwd = 2)
R> legend("bottomleft", levels(pbc2.id$edema), col = 1:3, lty = 1,
+     lwd = 2, bty = "n")
```



5.3. Time-Dependent Covariates

In order to measure the effect of an external time-dependent covariates in a Cox model we need to use the counting process formulation and have the data arrange in the long format (as explained in the course notes). For instance the Stanford data are organized in the long format in the data frame `heart`. The only difference compared to fitting standard Cox models is in the specification of `Surv()` function in the `formula` argument of `coxph()`. In particular, we need to specify the left and right limits of the intervals of observation and the event indicator that equals one if the event occurred at the end of the interval:

```
R> td.Cox <- coxph(Surv(start, stop, event) ~ transplant, data = heart)
R> summary(td.Cox)

Call:
coxph(formula = Surv(start, stop, event) ~ transplant, data = heart)

  n= 172

              coef exp(coef) se(coef)     z Pr(>|z|)
transplant1 0.1271    1.1356   0.3011 0.422    0.673

            exp(coef) exp(-coef) lower .95 upper .95
transplant1     1.136     0.8806    0.6293     2.049

Rsquare= 0.001   (max possible= 0.969 )
Likelihood ratio test= 0.18  on 1 df,   p=0.6715
Wald test            = 0.18  on 1 df,   p=0.6729
Score (logrank) test = 0.18  on 1 df,   p=0.6728
```

To fit Cox model with internal time-dependent covariates it is required to use the specialized functions of **JM** package. You may find more information about the capabilities of this package at: http://rwiki.sciviews.org/doku.php?id=packages:cran:jm. The basic usage requires a linear mixed effects model to describe the patient-specific evolutions of the time-dependent covariate (fitted by function `lme()` of package **nlme**), and a Cox model that contains any further baseline covariate for whose additional effect we wish to account for. These two models are supplied as main arguments in function `jointModel()` that actually fits the joint model. As an illustration, we replicate the analysis for the AIDS data in which we account for the effect of the CD4 cell counts, presented in the course notes

```
R> fitLME <- lme(sqrt(CD4) ~ obstime + obstime:drug, random = ~ obstime | patient,
+     data = aids)
R> fitSURV <- coxph(Surv(Time, death) ~ drug, data = aids.id, x = TRUE)
R> fit.JM <- jointModel(fitLME, fitSURV, timeVar = "obstime",
+        method = "piecewise-PH-GH")
R> summary(fit.JM)

Call:
jointModel(lmeObject = fitLME, survObject = fitSURV, timeVar = "obstime",
    method = "piecewise-PH-GH")

Data Descriptives:
Longitudinal Process                     Event Process
Number of Observations: 1405             Number of Events: 188 (40.3%)
Number of Groups: 467
```

```
Joint Model Summary:
Longitudinal Process: linear mixed effects model
Event Process: Relative risk model with piecewise-constant baseline risk function
                (knots at: 6.2, 11.1, 12.5, 13.9, 16, 17.8)
Parameterization: time-dependent

   log.Lik      AIC      BIC
 -2107.647 4247.295 4313.636

Variance Components:
             StdDev    Corr
(Intercept)  0.8660  (Intr)
obstime      0.0388  0.0680
Residual     0.3754

Coefficients:
Longitudinal Process
                  Value Std.Err z-value p-value
(Intercept)      2.5558  0.0372 68.7961 <0.0001
obstime         -0.0423  0.0046 -9.1931 <0.0001
obstime:drugddI  0.0051  0.0065  0.7821  0.4342

Event Process
            Value Std.Err z-value p-value
drugddI    0.3511  0.1537  2.2839  0.0224
Assoct    -1.1016  0.1180 -9.3388 <0.0001
log(xi.1) -1.6489  0.2498 -6.6000
log(xi.2) -1.3393  0.2394 -5.5940
log(xi.3) -1.0231  0.2861 -3.5758
log(xi.4) -1.5802  0.3736 -4.2299
log(xi.5) -1.4722  0.3500 -4.2069
log(xi.6) -1.4383  0.4283 -3.3584
log(xi.7) -1.4780  0.5455 -2.7094

Integration:
method: Gauss-Hermite
quadrature points: 15

Optimization:
Convergence: 0
```

5.4. Clustered Event Times

As we have seen, there are two possible ways to take into account the correlations of the failure times of subjects in the same cluster. Namely, we can a fit a marginal model and adjust the obtained standard error estimates or we can include an unobserved frailty term that explains heterogeneity between clusters. In `coxph()` both are implemented in the same way. In particular, to produce the grouped Jackknife estimate of the variance of parameter estimates we include in the linear predictor function `cluster()` that in fact is used to distinguish which subjects belong to which cluster. For example, for the Lung data set we obtain the Jackknife estimate by

```
R> marginal.model <- coxph(Surv(time, status) ~ age + cluster(inst), data = lung)
R> summary(marginal.model)

Call:
coxph(formula = Surv(time, status) ~ age + cluster(inst), data = lung)

  n=227 (1 observation deleted due to missingness)

        coef exp(coef) se(coef) robust se     z Pr(>|z|)
age 0.018636  1.018810 0.009212  0.007068 2.636  0.00838

    exp(coef) exp(-coef) lower .95 upper .95
age     1.019     0.9815     1.005     1.033

Rsquare= 0.018   (max possible= 0.999 )
Likelihood ratio test= 4.19  on 1 df,   p=0.04068
Wald test            = 6.95  on 1 df,   p=0.008377
Score (logrank) test = 4.11  on 1 df,   p=0.04275,   Robust = 3.57  p=0.05871

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

In the output both the ‘naive’ and Jackknife estimate of the standard error are included; the latter is labeled ‘robust se’.

To include a frailty term we include in the linear predictor function `frailty()` (instead of `cluster()`). For Cox model, the default is to include a Gamma distributed frailty term:

```
R> frailty.model <- coxph(Surv(time, status) ~ age + frailty(inst, df = 4),
+      data = lung)
R> summary(frailty.model)

Call:
coxph(formula = Surv(time, status) ~ age + frailty(inst, df = 4),
    data = lung)

  n=227 (1 observation deleted due to missingness)
                      coef   se(coef) se2     Chisq DF   p
age                   0.0194 0.00933  0.00925 4.31  1.00 0.038
frailty(inst, df = 4)                         3.33  3.99 0.500
```

```
    exp(coef) exp(-coef) lower .95 upper .95
age      1.02       0.98      1.00      1.04

Iterations: 3 outer, 10 Newton-Raphson
     Variance of random effect= 0.0380   I-likelihood = -743.6
Degrees of freedom for terms= 1 4
Rsquare= 0.043   (max possible= 0.999 )
Likelihood ratio test= 9.96  on 4.97 df,   p=0.075
Wald test            = 4.31  on 4.97 df,   p=0.501
```

We observe small heterogeneity between institutions (i.e., variance of the frailty term), which explains why the obtained results from the frailty model are almost identical with the ones from the marginal model.

5.5. Competing Risks

In competing risks settings the Kaplan-Meier-like estimate of the cumulative incidence function is produced using the `survfit()` function. However, compared to calculating the standard Kaplan-Meier estimate (which as mentioned in the slides is biased in some competing risks settings) a slightly different syntax is required. We will demonstrate using the PBC data set in which we have two competing event namely death and transplantation.

First, we need to construct a new event status indicator that equals one if either of the two events occurred. In the PBC data we construct this variable by

```
R> pbc2.id$status3 <- 1 - as.numeric(pbc2.id$status == "alive")
```

where variable `status` is a factor with levels, 'alive', 'dead', 'transplanted'. The in the call to `Surv()` within the formula argument of `survfit()` we include the new variable `status3` as event variable, and furthermore we need to supply the variable the distinguishes between the different causes in the `etype` argument.

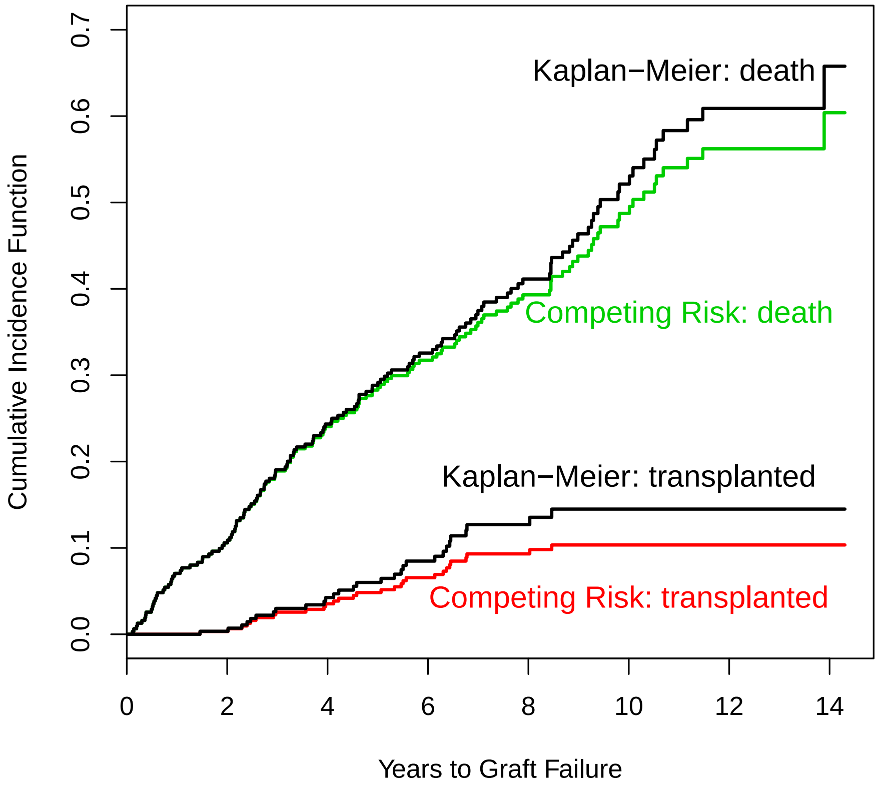
The following code computes: (i) the naive Kaplan-Meier estimates for death and transplantation where events from the other cause are treated as censored, (ii) the correct estimated of the cumulative incidence function

```
R> fit11 <- survfit(Surv(years, status == "dead") ~ 1, data = pbc2.id)
R> fit12 <- survfit(Surv(years, status == "transplanted") ~ 1, data = pbc2.id)
R> fit2 <- survfit(Surv(years, status3) ~ 1, data = pbc2.id, etype = status)
```

We include all estimates of the cumulative incidence function per cause of failure in the same plot:

```
R> plot(fit2, fun = "event", mark.time = FALSE, col = 2:3, lwd = 2,
+       ylim = c(0, 0.7), xlab = "Years to Graft Failure",
+       ylab = "Cumulative Incidence Function")
R> lines(fit11, fun = "event", mark.time = FALSE, conf.int = FALSE, lwd = 2)
R> lines(fit12, fun = "event", mark.time = FALSE, conf.int = FALSE, lwd = 2)
```

```
R> text(11, 0.37, "Competing Risk: death", col = 3, cex = 1.2)
R> text(10.9, 0.65, "Kaplan-Meier: death", cex = 1.2)
R> text(10, 0.04, "Competing Risk: transplanted", col = 2, cex = 1.2)
R> text(10, 0.18, "Kaplan-Meier: transplanted", cex = 1.2)
```



The option `fun = "event"` in the call to the `plot()` and `lines()` method corresponds to the transformation $1 - \hat{S}(t)$.

Finally, Cox regression for the cause-specific hazards proceeds in the same manner as in Section 4.1, since as mentioned in the slides, in this case we can treat events from other causes as censored. For instance, for the PBC data we investigate the effects of treatment and age on the cause-specific hazards by

```
R> fit1 <- coxph(Surv(years, status == "transplanted") ~ drug + age,
+      data = pbc2.id)
R> fit2 <- coxph(Surv(years, status == "dead") ~ drug + age, data = pbc2.id)
R> summary(fit1)
```

```
Call:
coxph(formula = Surv(years, status == "transplanted") ~ drug +
    age, data = pbc2.id)

  n= 312

                  coef exp(coef) se(coef)      z Pr(>|z|)
drugD-penicil -0.23680   0.78915  0.37723 -0.628     0.53
age           -0.09649   0.90802  0.02265 -4.259 2.05e-05

              exp(coef) exp(-coef) lower .95 upper .95
drugD-penicil    0.7891      1.267    0.3768    1.6530
age              0.9080      1.101    0.8686    0.9492

Rsquare= 0.069   (max possible= 0.625 )
Likelihood ratio test= 22.32  on 2 df,   p=1.421e-05
Wald test            = 18.54  on 2 df,   p=9.431e-05
Score (logrank) test = 20.1  on 2 df,   p=4.317e-05

R> summary(fit2)

Call:
coxph(formula = Surv(years, status == "dead") ~ drug + age, data = pbc2.id)

  n= 312

                   coef exp(coef)  se(coef)      z Pr(>|z|)
drugD-penicil -0.162071  0.850380  0.172501 -0.940    0.347
age            0.045718  1.046780  0.008487  5.387 7.16e-08

              exp(coef) exp(-coef) lower .95 upper .95
drugD-penicil    0.8504     1.1759    0.6064     1.192
age              1.0468     0.9553    1.0295     1.064

Rsquare= 0.089   (max possible= 0.991 )
Likelihood ratio test= 29.15  on 2 df,   p=4.687e-07
Wald test            = 29.04  on 2 df,   p=4.951e-07
Score (logrank) test = 29.47  on 2 df,   p=3.978e-07
```

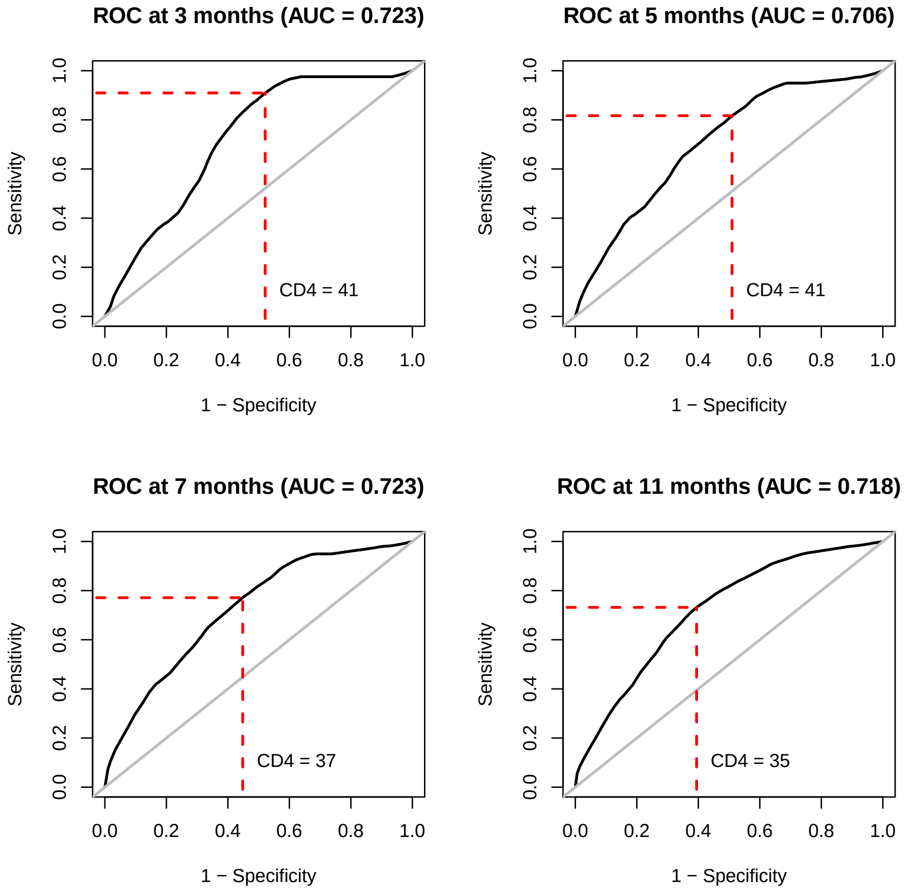
5.6. Discrimination

To calculate time-dependent ROC curves and the corresponding AUCs, we use function `survivalROC()` from package **survivalROC**. As before, in order to access this function we have first to load the package using `library(survivalROC)`.

The basic arguments of function `survivalROC()` are the observed survival times, the event indicator (’1‘ for event, and ‘0’ for censored cases), the numeric vector with the values of the marker, and the span for the nearest neighbor estimation method that controls the degree of smoothing. For the last argument, a default choice (which you need to specify as argument `span` in `survivalROC()` though) is $0.25 \times n^{(} - 0.20)$, with n denoting the number of observations. However, it is advisable to tune it in order to achieve an appropriate degree of smoothness that is not too wiggly but does not also lose important features of the resulting curve.

The following piece of code illustrates the calculation of time-dependent ROC curves for the AIDS data set, using as marker the baseline CD4 cell count measurement. The function assumes that high level for the marker are associated with higher risk for event. However, because for CD4 cell counts the opposite hold, we need to specify −CD4 as our marker. The code also includes in the plot the 'optimal' cutoff value for the CD4 cell count, using the Youden index (i.e., sensitivity + specificity - 1)

```
R> par(mfrow = c(2, 2))
R> for (tt in c(3, 5, 7, 11)) {
+     sroc <- with(aids.id, survivalROC(Time, death, -CD4,
+         predict.time = tt, span = 0.55*length(Time)^(-0.20)))
+     plot(sroc$FP, sroc$TP, type = "l", lwd = 2,
+         ylab = "Sensitivity",
+         xlab = "1 - Specificity",
+         main = paste("ROC at ", tt,  " months (AUC = ",
+           round(sroc$AUC, 3), ")", sep = ""))
+     abline(a = 0, b = 1, col = "grey", lwd = 1.7)
+     ii.ind <- which.max(sroc$TP + (1 - sroc$FP) - 1)
+     segments(sroc$FP[ii.ind], -0.1, sroc$FP[ii.ind], sroc$TP[ii.ind], lty = 2,
+         col = "red", lwd = 2)
+     segments(-0.1, sroc$TP[ii.ind], sroc$FP[ii.ind], sroc$TP[ii.ind], lty = 2,
+         col = "red", lwd = 2)
+     text(sroc$FP[ii.ind] + 0.01, 0.1, pos = 4,
+         paste("CD4 =", round(sroc$cut.value[ii.ind]^2, 2)))
+ }
```


ROC at 3 months (AUC = 0.723)
ROC at 5 months (AUC = 0.706)
ROC at 7 months (AUC = 0.723)
ROC at 11 months (AUC = 0.718)
Sensitivity
1 − Specificity
0.0
0.2
0.4
0.6
0.8
1.0
CD4 = 41
CD4 = 41
CD4 = 37
CD4 = 35