Cox PH: Diagnostics, Interactions & Model Building

A Comprehensive Guide to Building, Diagnosing, and Interpreting Survival Models

Eric Delmelle

2025-10-16

Link to qmd (quarto markdown)

Learning Objectives

By the end of this session, you will be able to:

Core Competencies

Modeling Skills

- Fit and interpret Cox proportional hazards models with multiple covariates
- Incorporate interaction terms and explain effect modification
- Handle non-linear relationships using transformations and splines

Diagnostic Skills

- Test and address violations of the proportional hazards assumption
- Interpret Schoenfeld, Martingale, Deviance, and DFBeta residuals
- Identify influential observations and outliers

Model Selection

- Apply the Events-Per-Variable (EPV) rule to avoid overfitting
- Compare competing models using AIC and likelihood ratio tests
- Evaluate predictive performance with concordance (C-index)

Success Criteria

You'll know you've mastered these concepts when you can:

- Explain why a hazard ratio changed after adding an interaction
- Decide whether to stratify or transform based on diagnostic plots
- Defend your model choices to a clinical collaborator

---

1 Part 1: Foundation

1.1 The Cox Model: Intuition

Think of the Cox model as answering: **"How do patient characteristics modify their baseline risk?"**

Mathematical Form

$$h(t \mid X) = h_0(t) \exp(\beta_1 X_1 + \cdots + \beta_p X_p)$$

where:

- $h(t|X)$ = hazard at time t for covariate pattern X
- $h_0(t)$ = baseline hazard (everyone with $X = 0$)
- $\exp(\beta_j)$ = hazard ratio (HR) for X_j

Intuitive Interpretation

If $\beta_{\text{age}} = 0.03$: - HR $= e^{0.03} = 1.030$ - Each year older $\rightarrow$ **3% higher risk** - 10 years older $\rightarrow 1.03^{10} = 1.34$ $\rightarrow$ **34% higher risk**

Key insight: Effects are **multiplicative** on the hazard scale.

> **!** The Proportional Hazards Assumption
>
> The "proportional" means the hazard ratio is **constant over time**. We'll test this assumption extensively in Part 4.

---

2 Setup & Data Preparation

2.1 Load Packages

```
# Core survival analysis
library(survival)
library(survminer)

# Model selection & penalization
library(MASS)     # stepAIC
library(glmnet)   # LASSO (optional)

# Data manipulation & visualization
library(dplyr)
library(ggplot2)
library(broom)    # Tidy model outputs
library(knitr)    # Tables
```

Package Roles

- **survival**: Core Cox modeling (`coxph`, `cox.zph`)
- **survminer**: Beautiful survival plots (`ggsurvplot`, `ggcoxdiagnostics`)
- **broom**: Convert model output to tidy data frames

2.2 The Primary Biliary Cirrhosis (PBC) Dataset

```
# Load and preprocess PBC data
data(pbc, package = "survival")

pbc <- pbc %>%
  mutate(
    # Define death as event (status==2); transplant censored for this tutorial
    event = as.integer(status == 2),

    # Convert time from days to years for interpretability
    time_years = time / 365.25,

    # Create interpretable factor labels
    trt = factor(trt, labels = c("D-penicillamine", "Placebo")),
    sex = factor(sex, labels = c("Male", "Female")),
    edema = factor(
      edema,
      levels = c(0, 0.5, 1),
      labels = c("None", "Controlled", "Persistent")
    )
  )

# Quick overview
glimpse(pbc)
```

```
Rows: 418
Columns: 22
$ id         <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, ~
$ time       <int> 400, 4500, 1012, 1925, 1504, 2503, 1832, 2466, 2400, 51, 37~
$ status     <int> 2, 0, 2, 2, 1, 2, 0, 2, 2, 2, 2, 2, 0, 2, 2, 0, 2, 2, 0, 2,~
$ trt        <fct> D-penicillamine, D-penicillamine, D-penicillamine, D-penici~
$ age        <dbl> 58.76523, 56.44627, 70.07255, 54.74059, 38.10541, 66.25873,~
$ sex        <fct> Female, Female, Male, Female, Female, Female, Female, Femal~
$ ascites    <int> 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0,~
$ hepato     <int> 1, 1, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 1, 1, 1,~
$ spiders    <int> 1, 1, 0, 1, 1, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0,~
$ edema      <fct> Persistent, None, Controlled, Controlled, None, None, None,~
$ bili       <dbl> 14.5, 1.1, 1.4, 1.8, 3.4, 0.8, 1.0, 0.3, 3.2, 12.6, 1.4, 3.~
$ chol       <int> 261, 302, 176, 244, 279, 248, 322, 280, 562, 200, 259, 236,~
$ albumin    <dbl> 2.60, 4.14, 3.48, 2.54, 3.53, 3.98, 4.09, 4.00, 3.08, 2.74,~
$ copper     <int> 156, 54, 210, 64, 143, 50, 52, 52, 79, 140, 46, 94, 40, 43,~
$ alk.phos   <dbl> 1718.0, 7394.8, 516.0, 6121.8, 671.0, 944.0, 824.0, 4651.2,~
$ ast        <dbl> 137.95, 113.52, 96.10, 60.63, 113.15, 93.00, 60.45, 28.38, ~
$ trig       <int> 172, 88, 55, 92, 72, 63, 213, 189, 88, 143, 79, 95, 130, NA~
```

```
$ platelet   <int> 190, 221, 151, 183, 136, NA, 204, 373, 251, 302, 258, 71, 2~
$ protime    <dbl> 12.2, 10.6, 12.0, 10.3, 10.9, 11.0, 9.7, 11.0, 11.0, 11.5, ~
$ stage      <int> 4, 3, 4, 4, 3, 3, 3, 3, 2, 4, 4, 4, 3, 4, 3, 3, 4, 4, 3, 4,~
$ event      <int> 1, 0, 1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1,~
$ time_years <dbl> 1.0951403, 12.3203285, 2.7707050, 5.2703628, 4.1177276, 6.8~
```

2.2.1 Dataset Summary

```
cat("Total observations:", nrow(pbc), "\n")
```

```
Total observations: 418
```

```
cat("Deaths (events):", sum(pbc$event, na.rm = TRUE), "\n")
```

```
Deaths (events): 161
```

```
cat("Median follow-up:", round(median(pbc$time_years, na.rm = TRUE), 1), "years\n")
```

```
Median follow-up: 4.7 years
```

```
cat("Missing data patterns:\n")
```

```
Missing data patterns:
```

```
pbc %>%
  summarise(across(everything(), ~sum(is.na(.)))) %>%
  select(where(~. > 0)) %>%
  kable()
```

trt	ascites	hepato	spiders	chol	copper	alk.phos	ast	trig	platelet	protime	stage
106	106	106	106	134	108	106	106	136	11	2	6

> **About PBC**
>
> - Primary Biliary Cirrhosis is a chronic liver disease.
> - Key prognostic factors include:
> - **Bilirubin** (liver function marker)
> - **Albumin** (nutritional status)
> - **Edema** (fluid retention severity)
> - **Prothrombin time** (blood clotting)

> **Competing Risks Note**
>
> Some patients received liver transplants (status==1). For this tutorial, we treat transplant as censoring to focus on Cox PH concepts. A complete analysis would use Fine-Gray competing risks models.

---

3 Part 2: Exploratory Analysis

3.1 Kaplan–Meier: The Foundation

Before fitting Cox models, **always** visualize survival curves.

3.1.1 Overall Survival

```
fit_km <- survfit(Surv(time_years, event) ~ 1, data = pbc)

ggsurvplot(
  fit_km,
  conf.int = TRUE,
  risk.table = TRUE,
  risk.table.height = 0.25,
  xlab = "Time (years)",
  ylab = "Survival Probability",
  title = "Overall Survival: Primary Biliary Cirrhosis",
  ggtheme = theme_minimal()
)
```

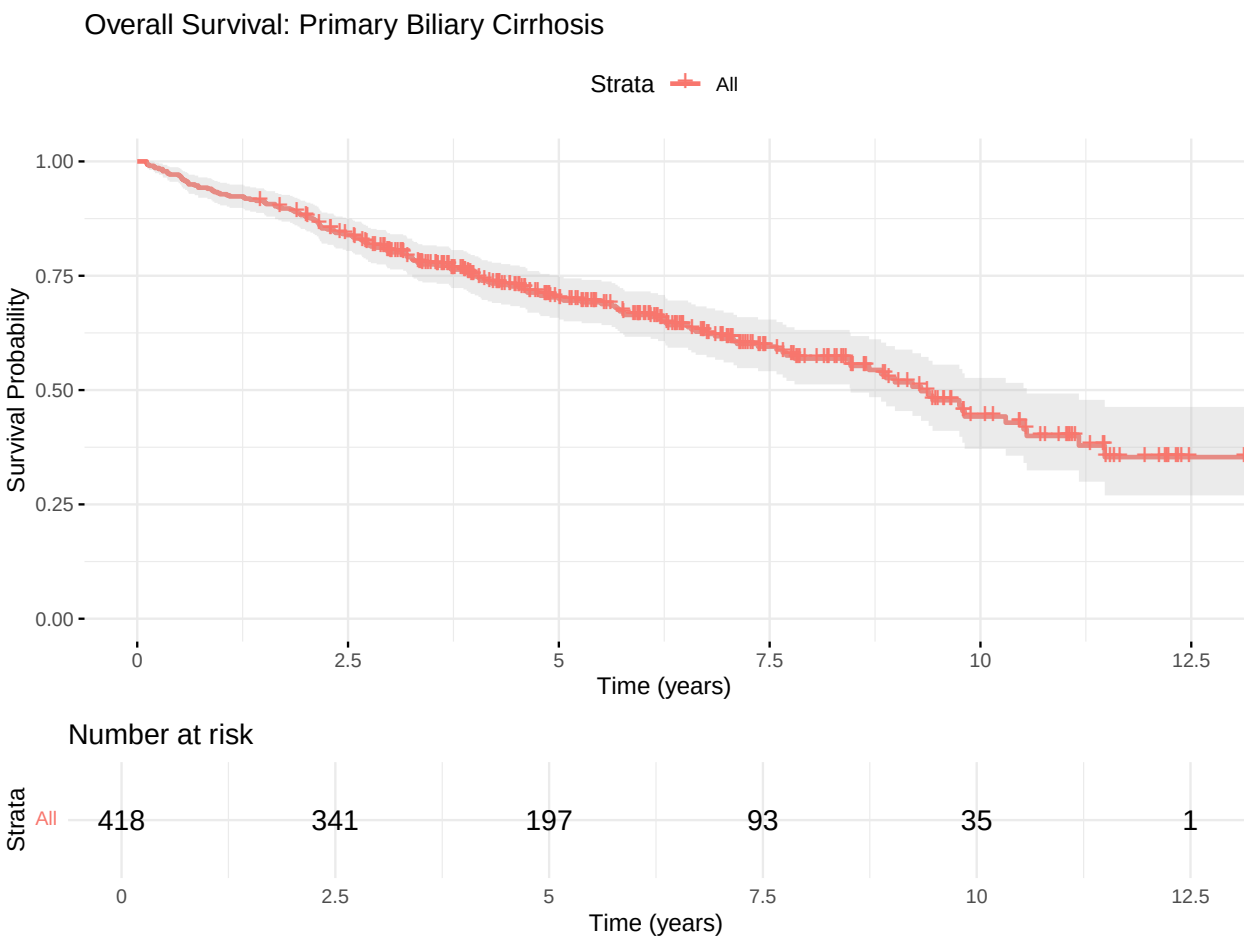


Figure 1: Overall survival in the PBC cohort

> **What to Look For**
>
> - **Median survival**: Where curve crosses 50%
> - **Confidence intervals**: Width indicates uncertainty
> - **Shape**: Steep drops suggest high early mortality

3.1.2 Survival by Treatment Group

```
fit_km_trt <- survfit(Surv(time_years, event) ~ trt, data = pbc)

ggsurvplot(
  fit_km_trt,
  conf.int = TRUE,
  pval = TRUE,
  pval.method = TRUE,
  risk.table = TRUE,
  risk.table.height = 0.25,
  legend.title = "Treatment",
  legend.labs = c("D-penicillamine", "Placebo"),
```

```
  palette = c("#E7B800", "#2E9FDF"),
  xlab = "Time (years)",
  title = "Survival by Treatment Arm"
)
```

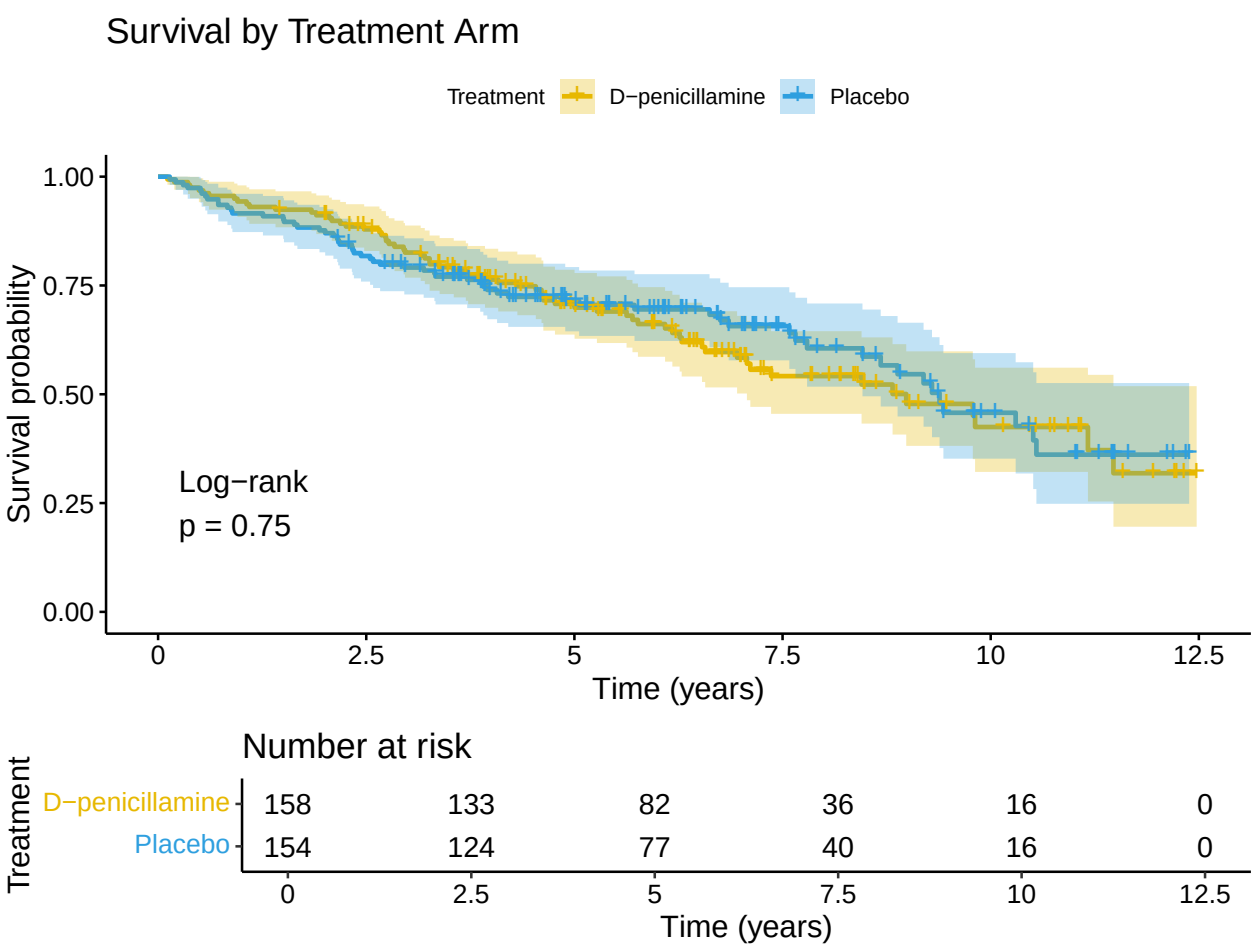


Figure 2: Comparing survival between treatment arms

Interpreting the Log-Rank Test

- **Null hypothesis**: No difference in survival between groups
- **p-value < 0.05**: Evidence of survival difference
- **Visual check**: Do confidence intervals overlap substantially?

3.1.3 Survival by Edema Status

```
fit_km_edema <- survfit(Surv(time_years, event) ~ edema, data = pbc)

ggsurvplot(
  fit_km_edema,
  conf.int = TRUE,
```

```
  pval = TRUE,
  risk.table = TRUE,
  risk.table.height = 0.3,
  legend.title = "Edema Status",
  palette = "jco",
  xlab = "Time (years)",
  title = "Survival by Edema Severity"
)
```

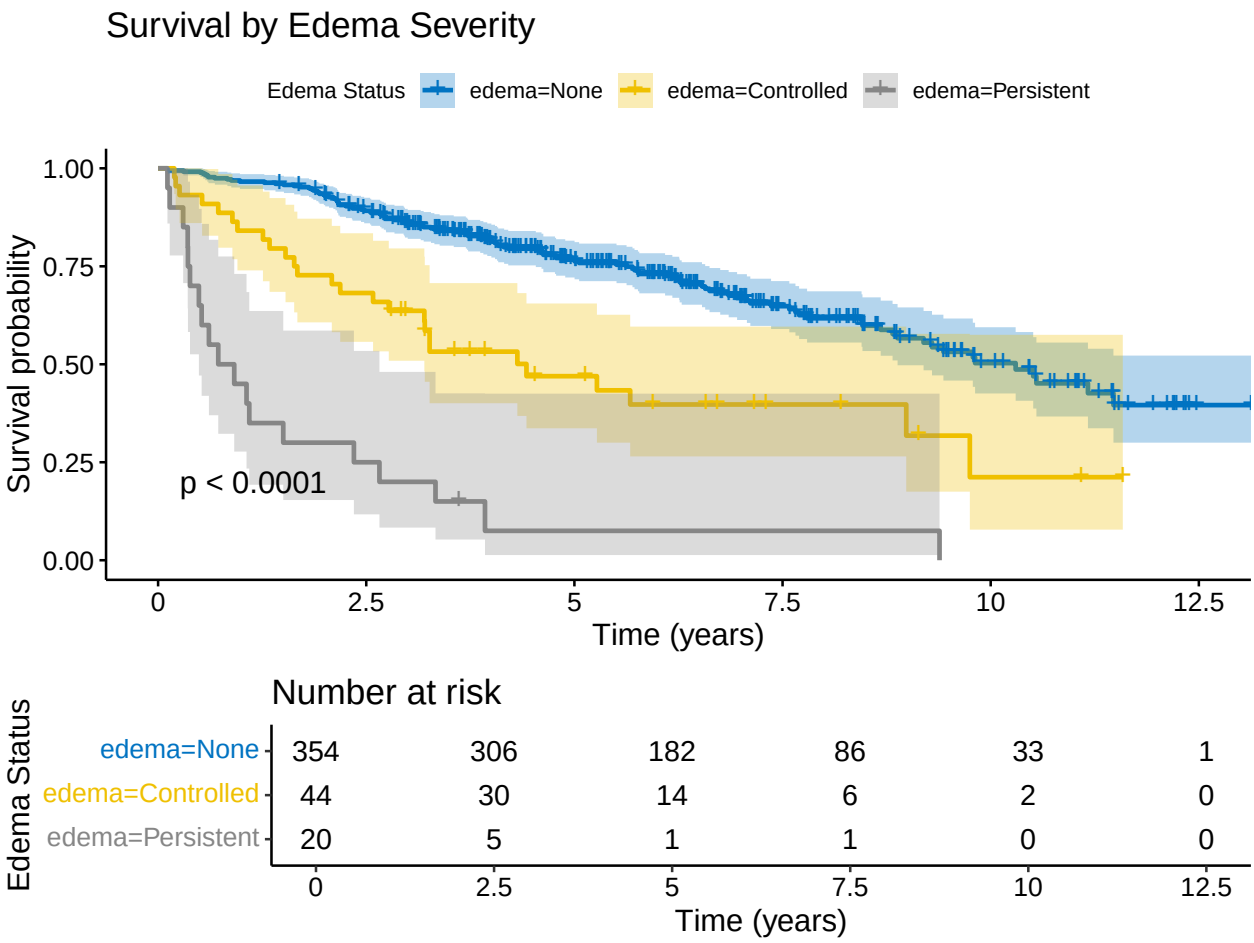


Figure 3: Survival stratified by edema severity

> **!** Clinical Insight
>
> **Persistent edema** (despite diuretics) is a strong prognostic marker. Notice the dramatic separation in survival curves—this will be important when we model interactions.

Quick Check: Based on these KM curves, which factors do you expect to be significant in a Cox model?

---

4 Part 3: Baseline Cox Model

4.1 Building the Core Model

Let's start with a model including clinical variables commonly used in PBC prognosis.

```
# Select core variables and create complete-case dataset
vars_core <- c("time_years", "event", "age", "sex", "trt", "edema",
               "bili", "albumin", "protime")
dat_core <- pbc %>%
  select(all_of(vars_core)) %>%
  na.omit()

cat("Complete cases:", nrow(dat_core), "\n")
```

```
Complete cases: 312
```

```
cat("Events in analysis:", sum(dat_core$event), "\n")
```

```
Events in analysis: 125
```

```
# Fit Cox model
cox_core <- coxph(
  Surv(time_years, event) ~ age + sex + trt + edema +
    log(bili) + albumin + protime,
  data = dat_core
)

# Display results
summary(cox_core)
```

```
Call:
coxph(formula = Surv(time_years, event) ~ age + sex + trt + edema +
    log(bili) + albumin + protime, data = dat_core)

  n= 312, number of events= 125

                     coef exp(coef)  se(coef)      z Pr(>|z|)
age               0.031413  1.031912  0.009229  3.404 0.000664 ***
sexFemale        -0.352636  0.702833  0.254770 -1.384 0.166317
trtPlacebo        0.064516  1.066643  0.193508  0.333 0.738832
edemaControlled   0.197602  1.218478  0.284942  0.693 0.488007
edemaPersistent   0.948578  2.582034  0.309298  3.067 0.002163 **
log(bili)         0.881180  2.413747  0.099447  8.861  < 2e-16 ***
albumin          -0.988534  0.372122  0.238781 -4.140 3.47e-05 ***
protime           0.249756  1.283712  0.086123  2.900 0.003732 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
                exp(coef) exp(-coef) lower .95 upper .95
age                1.0319     0.9691    1.0134    1.0507
sexFemale          0.7028     1.4228    0.4266    1.1580
trtPlacebo         1.0666     0.9375    0.7300    1.5586
edemaControlled    1.2185     0.8207    0.6971    2.1299
edemaPersistent    2.5820     0.3873    1.4083    4.7341
log(bili)          2.4137     0.4143    1.9863    2.9332
albumin            0.3721     2.6873    0.2330    0.5942
protime            1.2837     0.7790    1.0843    1.5198

Concordance= 0.846  (se = 0.02 )
Likelihood ratio test= 201.6  on 8 df,   p=<2e-16
Wald test            = 200.2  on 8 df,   p=<2e-16
Score (logrank) test = 290.8  on 8 df,   p=<2e-16
```

i Why log(bilirubin)?

- Bilirubin is right-skewed and its effect is likely non-linear.
- The log transformation:
- 1. Reduces skewness
- 2. Makes the HR easier to interpret (per doubling)
- 3. Often improves model fit

4.1.1 Visualizing Hazard Ratios

```r
ggforest(cox_core, data = dat_core, main = "Hazard Ratios: Baseline Model")
```

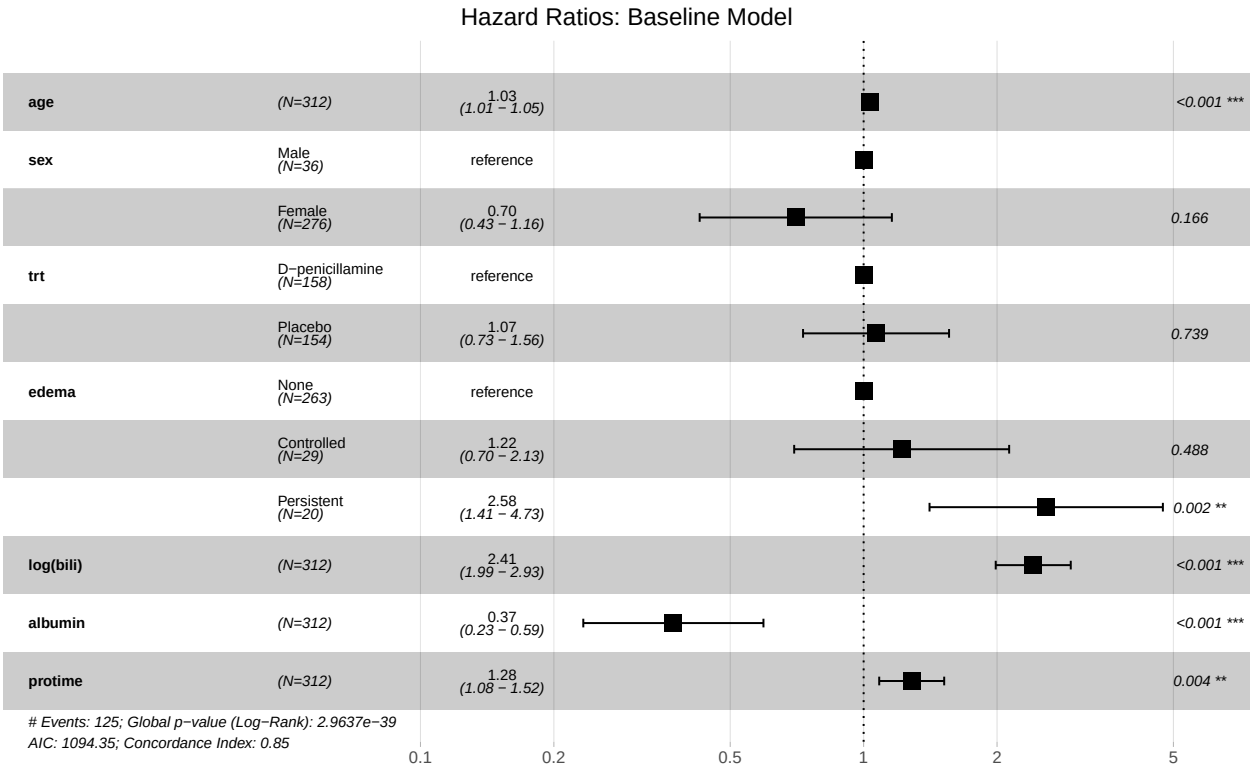


Figure 4: Forest plot of hazard ratios from baseline model

4.1.2 Interpretation Guide

```
tidy(cox_core, exponentiate = TRUE, conf.int = TRUE) %>%
  mutate(
    term = case_when(
      term == "age" ~ "Age (per year)",
      term == "sexFemale" ~ "Female vs Male",
      term == "trtPlacebo" ~ "Placebo vs D-penicillamine",
      term == "edemaControlled" ~ "Controlled edema vs None",
      term == "edemaPersistent" ~ "Persistent edema vs None",
      term == "log(bili)" ~ "log(Bilirubin) (per unit)",
      term == "albumin" ~ "Albumin (per g/dL)",
      term == "protime" ~ "Prothrombin time (per sec)",
      TRUE ~ term
    ),
    interpretation = case_when(
      estimate > 1 ~ paste0(round((estimate - 1) * 100, 1), "% higher risk"),
      estimate < 1 ~ paste0(round((1 - estimate) * 100, 1), "% lower risk"),
      TRUE ~ "No effect"
    )
  ) %>%
  select(Predictor = term, HR = estimate, `95% CI Low` = conf.low,
         `95% CI High` = conf.high, `P-value` = p.value, interpretation) %>%
  kable(digits = c(0, 2, 2, 2, 4, 0), caption = "Hazard Ratios with Clinical Interpretation")
```

Table 2: Hazard Ratios with Clinical Interpretation

Predictor	HR	95% CI Low	95% CI High	P-value	interpretation
Age (per year)	1.03	1.01	1.05	0.0007	3.2% higher risk
Female vs Male	0.70	0.43	1.16	0.1663	29.7% lower risk
Placebo vs D-penicillamine	1.07	0.73	1.56	0.7388	6.7% higher risk
Controlled edema vs None	1.22	0.70	2.13	0.4880	21.8% higher risk
Persistent edema vs None	2.58	1.41	4.73	0.0022	158.2% higher risk
log(Bilirubin) (per unit)	2.41	1.99	2.93	0.0000	141.4% higher risk
Albumin (per g/dL)	0.37	0.23	0.59	0.0000	62.8% lower risk
Prothrombin time (per sec)	1.28	1.08	1.52	0.0037	28.4% higher risk

Reading Hazard Ratios

For continuous variables: - HR = 1.03 for age → 3% higher hazard per year older - HR = 2.50 for log(bili) → 150% higher hazard per unit increase in log(bili) - Equivalently: doubling bili increases hazard by ~250%
For categorical variables: - HR = 2.50 for persistent edema → 150% higher hazard vs no edema - Same as saying "2.5 times the risk"

---

5 Part 4: Interactions

5.1 Understanding Effect Modification

An **interaction** means the effect of one variable depends on the level of another.

Research Question

Does treatment effectiveness differ by edema severity?

Clinical Rationale

Perhaps treatment works well for mild disease (no edema) but is less effective with severe disease (persistent edema).

Statistical Form

$$h(t) = h_0(t) \exp(\beta_1 \text{trt} + \beta_2 \text{edema} + \beta_3 \text{trt} \times \text{edema} + \cdots)$$

The β_3 coefficient tells us how the treatment effect **changes** across edema levels.

5.2 Fitting the Interaction Model

```
cox_int <- coxph(
  Surv(time_years, event) ~ age + sex + trt * edema +
    log(bili) + albumin + protime,
  data = dat_core
)
```

```r
summary(cox_int)
```

```
Call:
coxph(formula = Surv(time_years, event) ~ age + sex + trt * edema +
    log(bili) + albumin + protime, data = dat_core)

  n= 312, number of events= 125

                               coef exp(coef)  se(coef)      z Pr(>|z|)
age                         0.030259  1.030721  0.009269  3.264 0.001097 **
sexFemale                  -0.368232  0.691956  0.255499 -1.441 0.149520
trtPlacebo                  0.074710  1.077572  0.219964  0.340 0.734122
edemaControlled            -0.066842  0.935343  0.354391 -0.189 0.850398
edemaPersistent             1.553823  4.729515  0.442406  3.512 0.000444 ***
log(bili)                   0.880960  2.413214  0.098938  8.904  < 2e-16 ***
albumin                    -0.997501  0.368800  0.245013 -4.071 4.68e-05 ***
protime                     0.274010  1.315228  0.084584  3.239 0.001197 **
trtPlacebo:edemaControlled  1.128856  3.092116  0.580656  1.944 0.051883 .
trtPlacebo:edemaPersistent -0.933610  0.393132  0.540053 -1.729 0.083856 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                           exp(coef) exp(-coef) lower .95 upper .95
age                           1.0307     0.9702    1.0122    1.0496
sexFemale                     0.6920     1.4452    0.4194    1.1417
trtPlacebo                    1.0776     0.9280    0.7002    1.6584
edemaControlled               0.9353     1.0691    0.4670    1.8734
edemaPersistent               4.7295     0.2114    1.9872   11.2562
log(bili)                     2.4132     0.4144    1.9878    2.9296
albumin                       0.3688     2.7115    0.2282    0.5961
protime                       1.3152     0.7603    1.1143    1.5524
trtPlacebo:edemaControlled    3.0921     0.3234    0.9908    9.6496
trtPlacebo:edemaPersistent    0.3931     2.5437    0.1364    1.1330

Concordance= 0.847  (se = 0.02 )
Likelihood ratio test= 209.1  on 10 df,   p=<2e-16
Wald test            = 208.1  on 10 df,   p=<2e-16
Score (logrank) test = 322.4  on 10 df,   p=<2e-16
```

5.2.1 Testing the Interaction

```r
# Likelihood ratio test: Does adding interaction improve fit?
lrt <- anova(cox_core, cox_int, test = "Chisq")
print(lrt)
```

```
Analysis of Deviance Table
 Cox model: response is  Surv(time_years, event)
 Model 1: ~ age + sex + trt + edema + log(bili) + albumin + protime
 Model 2: ~ age + sex + trt * edema + log(bili) + albumin + protime
   loglik  Chisq Df Pr(>|Chi|)
```

```
1 -539.17
2 -535.43 7.4897  2    0.02364 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# Extract p-value
int_pval <- lrt$`Pr(>|Chi|)`[2]
cat("\nInteraction p-value:", round(int_pval, 4), "\n")
```

```
Interaction p-value: 0.0236
```

```r
if(int_pval < 0.05) {
  cat("Interpretation: Treatment effect DOES vary by edema status (p < 0.05)\n")
} else {
  cat("Interpretation: No strong evidence of effect modification (p >= 0.05)\n")
}
```

```
Interpretation: Treatment effect DOES vary by edema status (p < 0.05)
```

> **!** Interpreting Interactions
>
> If the interaction is significant: - **Don't interpret main effects alone**—they're conditional - Calculate treatment HR **at each level** of the modifier - Use adjusted survival curves for visualization. - Report stratum-specific effects in results

5.2.2 Visualizing the Interaction

```r
# Adjusted curves (holding other variables at typical values)
ggadjustedcurves(
  cox_int,
  data = dat_core,
  variable = "edema",
  palette = "jco",
  legend.title = "Edema Status",
  xlab = "Time (years)",
  ylab = "Adjusted Survival Probability",
  ggtheme = theme_minimal()
)
```

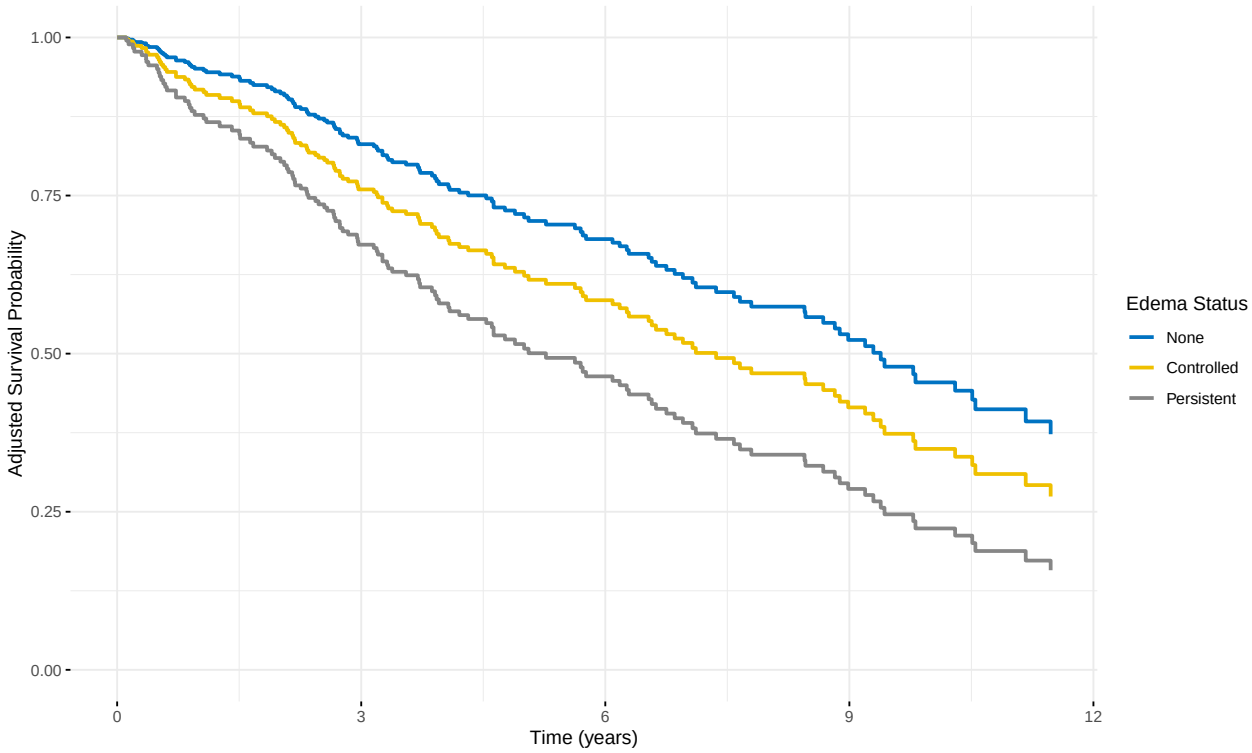


Figure 5: Adjusted survival curves showing treatment × edema interaction

When to Include Interactions

Include if: - Strong prior hypothesis (e.g., treatment × severity) - Exploratory analysis suggests effect modification - Clinical importance justifies complexity
Don't include: - "Fishing" for significance - Sample size is small (reduces power) - You can't interpret the results clearly

---

6 Part 5: Non-Linearity

6.1 Why Linearity Matters

The Cox model assumes each continuous predictor has a **linear** effect on the log-hazard. This may not be true!

Example: Age might have an accelerating effect (quadratic) or a threshold effect (spline).

6.2 Checking Functional Form: Martingale Residuals

```
# Check linearity for age and log(bili)
ggcoxfunctional(
  Surv(time_years, event) ~ age + log(bili),
```

```
  data = dat_core,
  point.col = "blue",
  point.alpha = 0.5
) +
  ggtitle("Functional Form Assessment: Red line should be roughly straight")
```

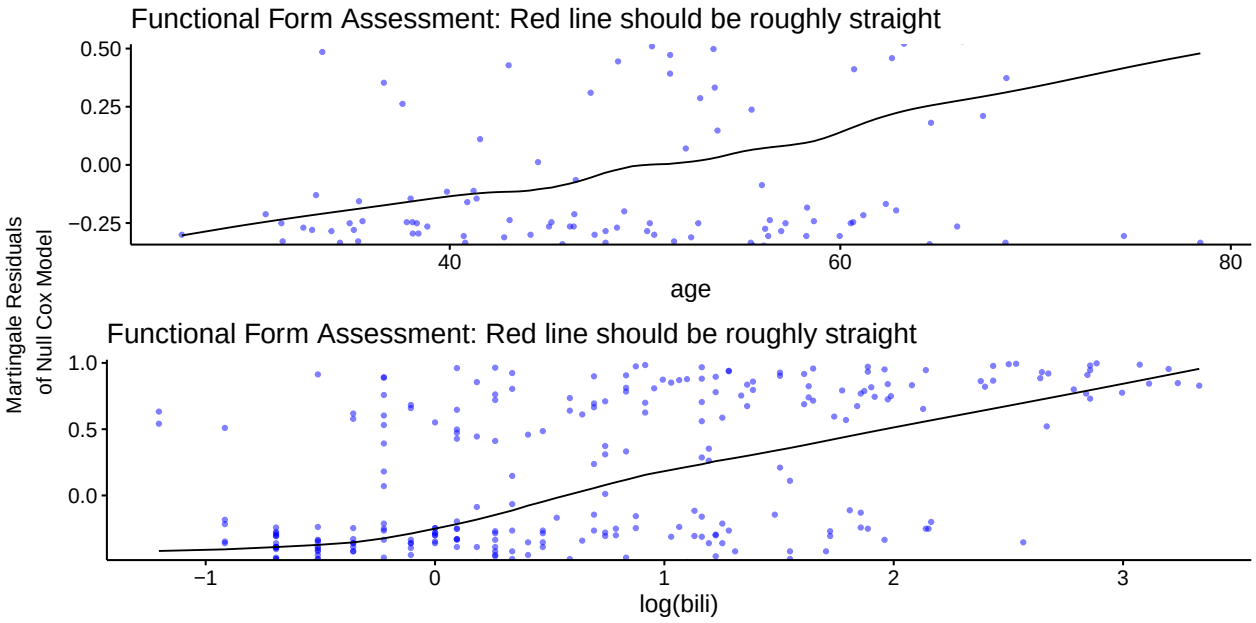


Figure 6: Martingale residual plots for linearity assessment

> **Reading the Plot**
>
> - **Straight red line** → linear assumption is reasonable
> - **Curved red line** → consider transformation or polynomial
> - **U-shape** → quadratic term may help
> - **Complex shape** → spline might be needed

6.3 Option 1: Polynomial Terms

```
cox_quad <- coxph(
  Surv(time_years, event) ~ poly(age, 2, raw = TRUE) + sex + trt + edema +
    log(bili) + albumin + protime,
  data = dat_core
)

# Test improvement
anova(cox_core, cox_quad)
```

```
Analysis of Deviance Table
 Cox model: response is  Surv(time_years, event)
 Model 1: ~ age + sex + trt + edema + log(bili) + albumin + protime
```

```
 Model 2: ~ poly(age, 2, raw = TRUE) + sex + trt + edema + log(bili) + albumin + protime
   loglik  Chisq Df Pr(>|Chi|)
1 -539.17
2 -538.49 1.3789  1     0.2403
```

```
cat("\nQuadratic model summary:\n")
```

```
Quadratic model summary:
```

```
summary(cox_quad)
```

```
Call:
coxph(formula = Surv(time_years, event) ~ poly(age, 2, raw = TRUE) +
    sex + trt + edema + log(bili) + albumin + protime, data = dat_core)

  n= 312, number of events= 125

                               coef  exp(coef)   se(coef)      z Pr(>|z|)
poly(age, 2, raw = TRUE)1  0.1159680  1.1229600  0.0747107  1.552  0.12061
poly(age, 2, raw = TRUE)2 -0.0007798  0.9992205  0.0006831 -1.142  0.25365
sexFemale                 -0.4112696  0.6628082  0.2551358 -1.612  0.10697
trtPlacebo                 0.0490128  1.0502337  0.1932266  0.254  0.79976
edemaControlled            0.1746141  1.1907866  0.2863767  0.610  0.54204
edemaPersistent            0.9357082  2.5490181  0.3114295  3.005  0.00266 **
log(bili)                  0.8922733  2.4406718  0.0997279  8.947  < 2e-16 ***
albumin                   -0.9800468  0.3752935  0.2373883 -4.128 3.65e-05 ***
protime                    0.2440046  1.2763502  0.0861301  2.833  0.00461 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                           exp(coef) exp(-coef) lower .95 upper .95
poly(age, 2, raw = TRUE)1    1.1230     0.8905    0.9700    1.3000
poly(age, 2, raw = TRUE)2    0.9992     1.0008    0.9979    1.0006
sexFemale                    0.6628     1.5087    0.4020    1.0928
trtPlacebo                   1.0502     0.9522    0.7191    1.5338
edemaControlled              1.1908     0.8398    0.6793    2.0874
edemaPersistent              2.5490     0.3923    1.3845    4.6931
log(bili)                    2.4407     0.4097    2.0073    2.9675
albumin                      0.3753     2.6646    0.2357    0.5976
protime                      1.2764     0.7835    1.0781    1.5111

Concordance= 0.846  (se = 0.02 )
Likelihood ratio test= 203  on 9 df,   p=<2e-16
Wald test            = 198.1  on 9 df,   p=<2e-16
Score (logrank) test = 290.9  on 9 df,   p=<2e-16
```

> ⚠ Interpreting Polynomials
>
> With `age + age²`: - Main effect (age) = slope at age = 0 (not meaningful!) - Interpret via **marginal effects** or predicted curves - Or center age: use `poly(age - 50, 2)` for interpretation around age 50

6.4 Option 2: Flexible Splines

```
cox_spline <- coxph(
  Surv(time_years, event) ~ pspline(age, df = 4) + sex + trt + edema +
    log(bili) + albumin + protime,
  data = dat_core
)

# Test improvement
anova(cox_core, cox_spline)
```

```
Analysis of Deviance Table
 Cox model: response is  Surv(time_years, event)
 Model 1: ~ age + sex + trt + edema + log(bili) + albumin + protime
 Model 2: ~ pspline(age, df = 4) + sex + trt + edema + log(bili) + albumin + protime
   loglik  Chisq     Df Pr(>|Chi|)
1 -539.17
2 -533.88 10.589 2.9792    0.01391 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Note: Individual coefficients for splines are not directly interpretable
cat("\nSpline test for non-linearity:\n")
```

```
Spline test for non-linearity:
```

```
print(anova(cox_core, cox_spline))
```

```
Analysis of Deviance Table
 Cox model: response is  Surv(time_years, event)
 Model 1: ~ age + sex + trt + edema + log(bili) + albumin + protime
 Model 2: ~ pspline(age, df = 4) + sex + trt + edema + log(bili) + albumin + protime
   loglik  Chisq     Df Pr(>|Chi|)
1 -539.17
2 -533.88 10.589 2.9792    0.01391 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

6.4.1 Visualizing the Age Effect

```
# Extract predicted log-hazard for age range
age_seq <- seq(min(dat_core$age), max(dat_core$age), length.out = 100)
pred_data <- data.frame(
  age = age_seq,
  sex = "Female",  # Hold others constant
  trt = "Placebo",
  edema = "None",
  bili = median(dat_core$bili),
```

```
  albumin = median(dat_core$albumin),
  protime = median(dat_core$protime)
)

# Get predictions (requires some manipulation for splines)
# For demonstration, show linear vs quadratic comparison instead
pred_linear <- predict(cox_core, newdata = pred_data, type = "risk", reference = "sample")
pred_quad <- predict(cox_quad, newdata = pred_data, type = "risk", reference = "sample")

plot_data <- data.frame(
  age = age_seq,
  linear = pred_linear,
  quadratic = pred_quad
) %>%
  tidyr::pivot_longer(-age, names_to = "model", values_to = "relative_hazard")

ggplot(plot_data, aes(x = age, y = relative_hazard, color = model)) +
  geom_line(size = 1.2) +
  scale_color_manual(values = c("linear" = "blue", "quadratic" = "red")) +
  labs(
    title = "Relative Hazard by Age",
    subtitle = "Comparing linear vs quadratic functional forms",
    x = "Age (years)",
    y = "Relative Hazard (vs median age)",
    color = "Model"
  ) +
  theme_minimal() +
  theme(legend.position = "top")
```

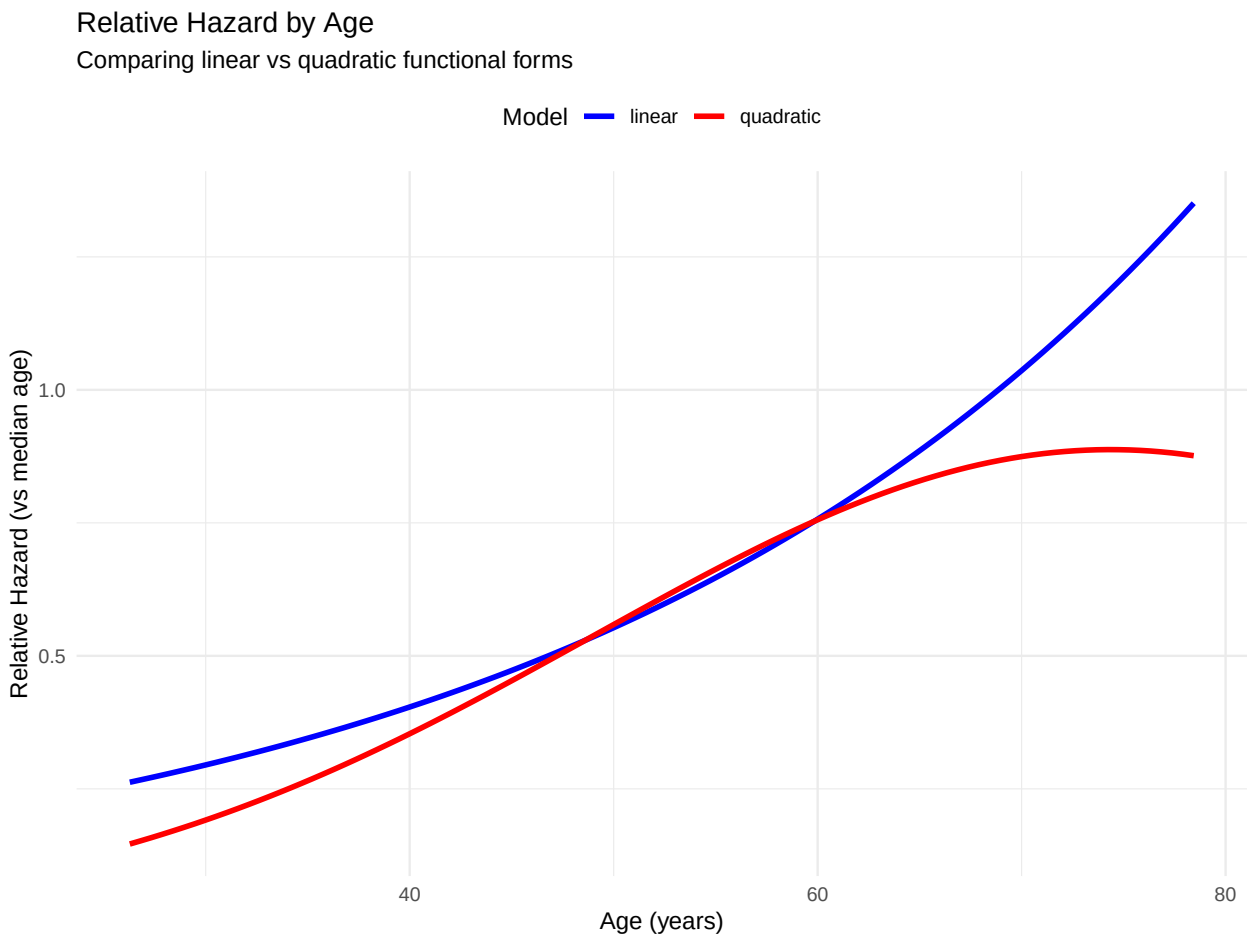


Figure 7: Estimated hazard ratio as a function of age (spline model)

Choosing Between Options

Use polynomial when: - You expect a simple curve (U-shape, inverted U) - You want to explain results in words - Sample size is moderate
Use spline when: - Relationship is complex - Prediction accuracy is paramount - You have large sample size
Use neither when: - Linear assumption holds (check first!) - Sample size is small (conserve df)

---

7 Part 6: Diagnostics

7.1 The Proportional Hazards Assumption

Key Concept: Cox PH assumes hazard ratios are **constant over time**.

If violated, interpretation becomes complex—a HR of 2.0 at 1 year might be 1.5 at 5 years.

7.1.1 Testing PH: Schoenfeld Residuals

```
ph_test <- cox.zph(cox_int)
print(ph_test)
```

```
             chisq df     p
age        0.00586  1 0.939
sex        0.00145  1 0.970
trt        2.40817  1 0.121
edema      2.60199  2 0.272
log(bili)  1.98789  1 0.159
albumin    0.02229  1 0.881
protime    4.49805  1 0.034
trt:edema  2.62474  2 0.269
GLOBAL    15.04839 10 0.130
```

Interpreting the Table

- **Global test**: Overall PH assumption (bottom row)
- **Individual tests**: Each covariate separately
- **$p < 0.05$**: Evidence of violation → investigate
- **$p \geq 0.05$**: Assumption appears reasonable

7.1.2 Visualizing PH Violations

```
ggcoxzph(ph_test,
         point.col = "blue",
         point.alpha = 0.5,
         ggtheme = theme_minimal())
```

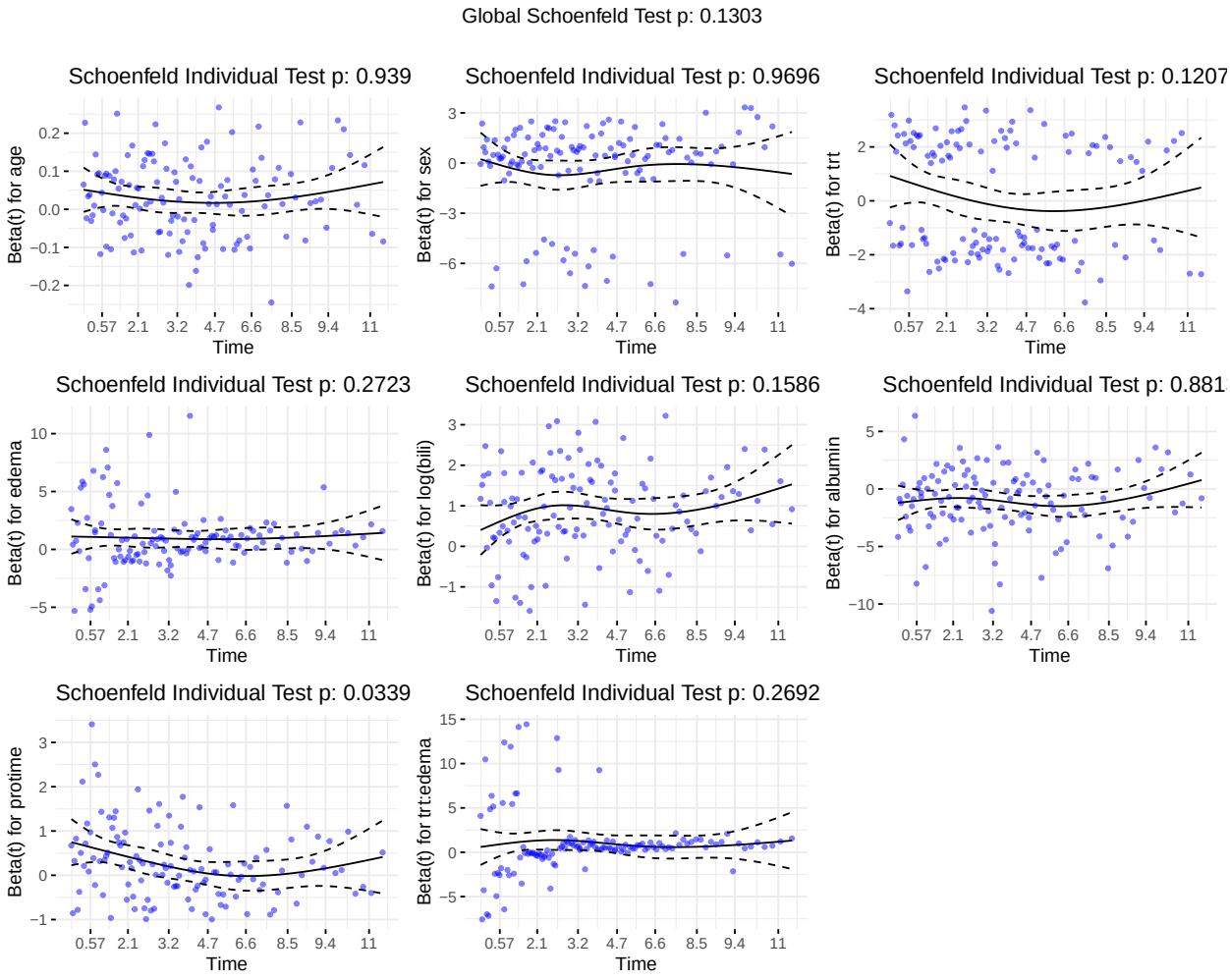


Figure 8: Schoenfeld residual plots (trend indicates PH violation)

! What to Look For

- **Horizontal line (= constant)**: PH assumption met
- **Trending line**: PH violation—effect changes over time
- **Focus on p-value AND visual**: Both matter!

7.2 Addressing PH Violations

7.2.1 Option: Stratification

When a **categorical** variable violates PH, stratify on it:

```
# Example: If edema violates PH, we stratify
cox_strata <- coxph(
  Surv(time_years, event) ~ age + sex + trt +
    log(bili) + albumin + protime + strata(edema),
  data = dat_core
)
```

```r
summary(cox_strata)
```

```
Call:
coxph(formula = Surv(time_years, event) ~ age + sex + trt + log(bili) +
    albumin + protime + strata(edema), data = dat_core)

  n= 312, number of events= 125

                coef exp(coef)  se(coef)      z Pr(>|z|)
age         0.031802  1.032313  0.009428  3.373 0.000743 ***
sexFemale  -0.338551  0.712803  0.253933 -1.333 0.182456
trtPlacebo  0.081623  1.085047  0.194100  0.421 0.674105
log(bili)   0.855776  2.353199  0.099922  8.564  < 2e-16 ***
albumin    -0.934359  0.392837  0.238580 -3.916 8.99e-05 ***
protime     0.256927  1.292950  0.085031  3.022 0.002515 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

           exp(coef) exp(-coef) lower .95 upper .95
age           1.0323     0.9687    1.0134     1.052
sexFemale     0.7128     1.4029    0.4333     1.173
trtPlacebo    1.0850     0.9216    0.7417     1.587
log(bili)     2.3532     0.4250    1.9347     2.862
albumin       0.3928     2.5456    0.2461     0.627
protime       1.2930     0.7734    1.0945     1.527

Concordance= 0.806  (se = 0.027 )
Likelihood ratio test= 132.3  on 6 df,   p=<2e-16
Wald test            = 133  on 6 df,   p=<2e-16
Score (logrank) test = 143.6  on 6 df,   p=<2e-16
```

Trade-offs of Stratification

Advantages: - Allows different baseline hazards by stratum - Retains all other covariate effects
Disadvantages: - **Cannot estimate** HR for the stratified variable - Reduces power slightly - Only works for categorical variables

Not Covered Today: Time-Varying Coefficients

For advanced analyses, you can model $\beta(t)$ directly using `tt()` functions or time-dependent covariates. This is beyond our scope but important for specialized applications.

7.3 Residual Diagnostics

7.3.1 1. Deviance Residuals: Overall Fit

```
ggcoxdiagnostics(
  cox_int,
  type = "deviance",
  linear.predictions = TRUE,
  ggtheme = theme_minimal()
) +
  ggtitle("Deviance Residuals: Should be symmetrically scattered around 0") +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red")
```

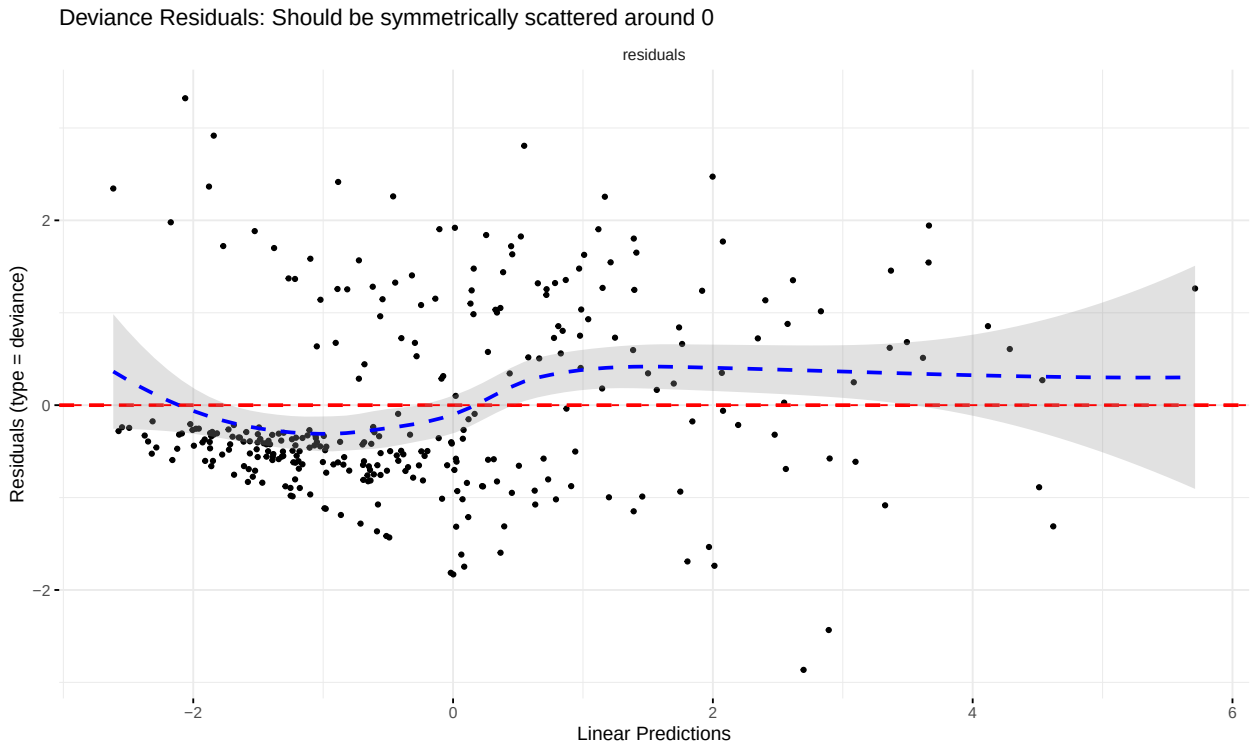


Figure 9: Deviance residuals vs linear predictor

> **What to Look For**
>
> - **Random scatter around 0**: Good fit
> - **Patterns (curves, fans)**: Functional form issues
> - **Outliers**: Investigate individual cases

7.3.2 2. Martingale Residuals: Non-Linearity

```
ggcoxdiagnostics(
  cox_int,
  type = "martingale",
  linear.predictions = FALSE,
  ggtheme = theme_minimal()
) +
  ggtitle("Martingale Residuals: Large negative values indicate poor predictions")
```

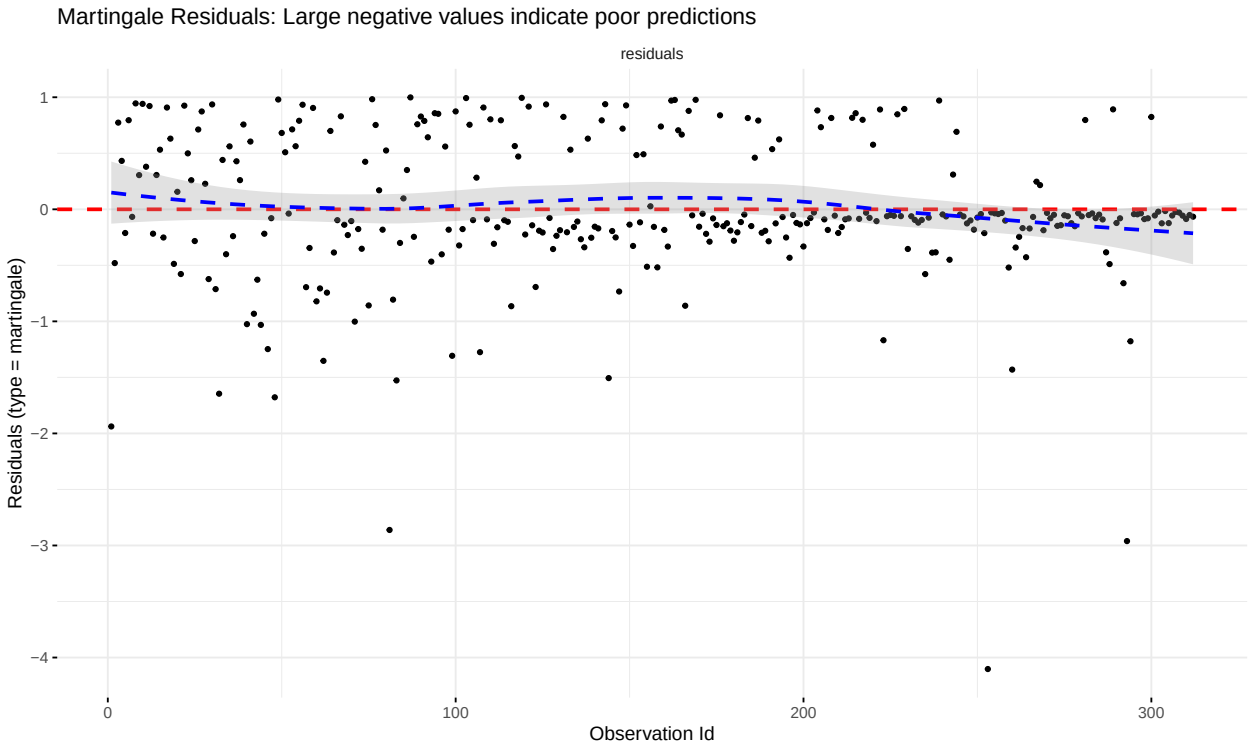


Figure 10: Martingale residuals: Detecting systematic misfits

7.3.3 3. DFBeta: Influential Observations

```
ggcoxdiagnostics(
  cox_int,
  type = "dfbeta",
  linear.predictions = FALSE,
  ggtheme = theme_minimal()
) +
  ggtitle("DFBeta: Large values indicate observations influencing specific coefficients")
```

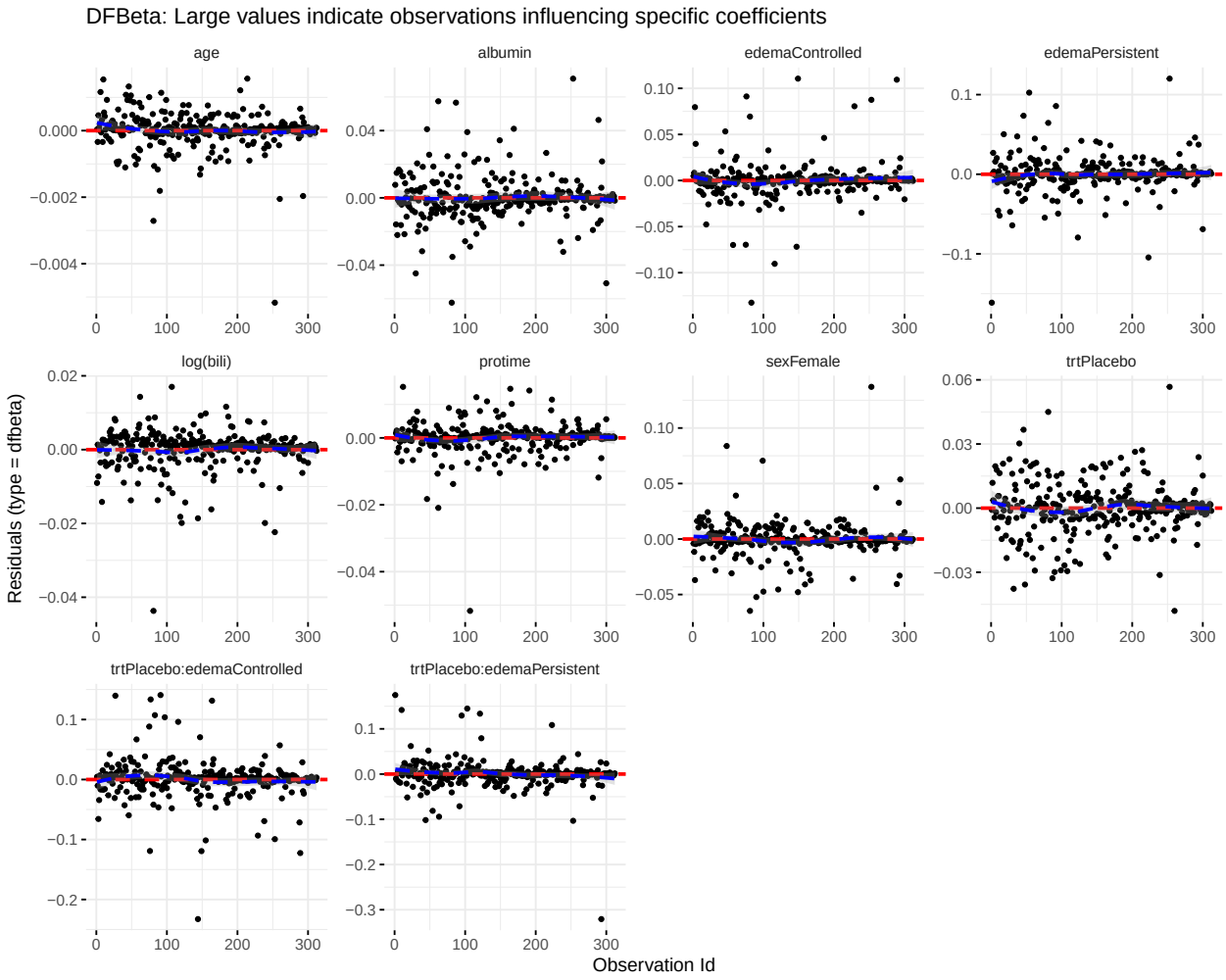


Figure 11: DFBeta residuals: Influence on coefficients

Dealing with Influential Points

Investigate but don't automatically remove: - Check for data entry errors - Understand clinical context (unusual but valid?) - Fit model with/without outliers—does conclusion change? - Report sensitivity analysis if influential
Never remove data just to improve fit without justification!

---

8 Part 7: Model Performance

8.1 Concordance (C-index)

The **C-index** measures discriminative ability: Can the model rank patients by risk?

```r
conc <- summary(cox_int)$concordance
cat("C-index:", round(conc[1], 3), "\n")
```

```
C-index: 0.847
```

```r
cat("SE:", round(conc[2], 3), "\n")
```

```
SE: 0.02
```

C-Index Interpretation

C-Index	Performance	Interpretation
0.50	Random	Model has no discriminative ability
0.60	Poor	Weak discrimination
0.70	Acceptable	Moderate discrimination
0.80	Good	Strong discrimination
0.90	Excellent	Very strong (check for overfitting!)

Rule of thumb: Clinical models often achieve 0.65–0.75 for most endpoints.

8.2 Comparing Models

8.2.1 Likelihood Ratio Tests (Nested Models)

```r
# Compare core vs interaction model
cat("Core vs Interaction Model:\n")
```

```
Core vs Interaction Model:
```

```r
anova(cox_core, cox_int, test = "Chisq")
```

```
Analysis of Deviance Table
 Cox model: response is  Surv(time_years, event)
 Model 1: ~ age + sex + trt + edema + log(bili) + albumin + protime
 Model 2: ~ age + sex + trt * edema + log(bili) + albumin + protime
   loglik  Chisq Df Pr(>|Chi|)
1 -539.17
2 -535.43 7.4897  2    0.02364 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
cat("\n\nCore vs Quadratic Age Model:\n")
```

```


Core vs Quadratic Age Model:
```

```r
anova(cox_core, cox_quad, test = "Chisq")
```

```
Analysis of Deviance Table
 Cox model: response is  Surv(time_years, event)
 Model 1: ~ age + sex + trt + edema + log(bili) + albumin + protime
 Model 2: ~ poly(age, 2, raw = TRUE) + sex + trt + edema + log(bili) + albumin + protime
   loglik  Chisq Df Pr(>|Chi|)
1 -539.17
2 -538.49 1.3789  1     0.2403
```

```r
cat("\n\nCore vs Spline Age Model:\n")
```

```
Core vs Spline Age Model:
```

```r
anova(cox_core, cox_spline, test = "Chisq")
```

```
Analysis of Deviance Table
 Cox model: response is  Surv(time_years, event)
 Model 1: ~ age + sex + trt + edema + log(bili) + albumin + protime
 Model 2: ~ pspline(age, df = 4) + sex + trt + edema + log(bili) + albumin + protime
   loglik  Chisq     Df Pr(>|Chi|)
1 -539.17
2 -533.88 10.589 2.9792    0.01391 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

8.2.2 AIC Comparison (Any Models)

```r
# AIC: Lower is better
aic_table <- AIC(cox_core, cox_int, cox_quad, cox_spline) %>%
  mutate(
    Model = c("Core (main effects only)",
              "Interaction (trt × edema)",
              "Quadratic age",
              "Spline age"),
    Delta_AIC = AIC - min(AIC)
  ) %>%
  arrange(AIC) %>%
  select(Model, df, AIC, Delta_AIC)

kable(aic_table, digits = 1,
      caption = "Model Comparison via AIC (lower is better)")
```

Table 4: Model Comparison via AIC (lower is better)

	Model	df	AIC	Delta_AIC
cox_spline	Spline age	11	1089.7	0.0
cox_int	Interaction (trt × edema)	10	1090.9	1.1
cox_core	Core (main effects only)	8	1094.3	4.6
cox_quad	Quadratic age	9	1095.0	5.3

Interpreting ΔAIC

- **ΔAIC < 2**: Models essentially equivalent
- **ΔAIC 2–7**: Some evidence for better model
- **ΔAIC > 10**: Strong evidence for better model

Prefer simpler model if ΔAIC < 2!

8.2.3 Comprehensive Model Summary

```
models <- list(
  Core = cox_core,
  Interaction = cox_int,
  Quadratic = cox_quad,
  Spline = cox_spline
)

comparison <- data.frame(
  Model = names(models),
  Parameters = sapply(models, \(m) length(coef(m))),
  LogLik = sapply(models, \(m) round(logLik(m)[1], 1)),
  AIC = sapply(models, AIC) %>% round(1),
  C_index = sapply(models, \(m) round(summary(m)$concordance[1], 3))
) %>%
  mutate(
    Delta_AIC = AIC - min(AIC),
    Rank = rank(AIC)
  ) %>%
  arrange(Rank)

kable(comparison,
      caption = "Comprehensive Model Performance Summary",
      digits = c(0, 0, 1, 1, 3, 1, 0))
```

Table 5: Comprehensive Model Performance Summary

	Model	Parameters	LogLik	AIC	C_index	Delta_AIC	Rank
Spline	Spline	19	-533.9	1089.7	0.848	0.0	1
Interaction	Interaction	10	-535.4	1090.9	0.847	1.2	2
Core	Core	8	-539.2	1094.3	0.846	4.6	3
Quadratic	Quadratic	9	-538.5	1095.0	0.846	5.3	4

9 Part 8: Variable Selection

9.1 The Events-Per-Variable (EPV) Rule

Classic guideline: Need 10 events per parameter to avoid overfitting.

```
n_events <- sum(dat_core$event)
n_params_core <- length(coef(cox_core))

cat("Events:", n_events, "\n")
```

```
Events: 125
```

```
cat("Parameters in core model:", n_params_core, "\n")
```

```
Parameters in core model: 8
```

```
cat("EPV ratio:", round(n_events / n_params_core, 1), "\n\n")
```

```
EPV ratio: 15.6
```

```
max_safe_params <- floor(n_events / 10)
cat("Safe parameter budget ( 10 EPV):", max_safe_params, "\n")
```

```
Safe parameter budget ( 10 EPV): 12
```

⚠ EPV Violations: Risks

Too many parameters (low EPV) leads to: - Overfit model (great fit on training data, poor on new data) - Unstable coefficient estimates - Unreliable confidence intervals - Biased p-values
Solutions: - Remove less important predictors - Use domain knowledge for selection - Consider penalization (LASSO) for many weak predictors

9.2 Automated Selection: Stepwise (Use with Caution!)

```
# Expand to include more potential predictors
vars_all <- c("time_years", "event", "age", "sex", "trt", "edema",
              "bili", "albumin", "protime", "chol", "copper")
dat_all <- pbc %>%
  select(all_of(vars_all)) %>%
  na.omit()
```

```r
cat("Complete cases with extended variables:", nrow(dat_all), "\n")
```

```
Complete cases with extended variables: 282
```

```r
cat("Events:", sum(dat_all$event), "\n\n")
```

```
Events: 113
```

```r
# Fit full model
fit_full <- coxph(
  Surv(time_years, event) ~ age + sex + trt + edema +
    log(bili) + albumin + protime + log(chol) + log(copper),
  data = dat_all
)

# Stepwise selection
fit_step <- stepAIC(fit_full, direction = "both", trace = 0)

cat("Stepwise-selected model:\n")
```

```
Stepwise-selected model:
```

```r
print(formula(fit_step))
```

```
Surv(time_years, event) ~ age + edema + log(bili) + albumin +
    protime + log(copper)
<environment: 0x7fac94941968>
```

```r
cat("\n")
```

```r
summary(fit_step)
```

```
Call:
coxph(formula = Surv(time_years, event) ~ age + edema + log(bili) +
    albumin + protime + log(copper), data = dat_all)

  n= 282, number of events= 113

                    coef exp(coef)  se(coef)      z Pr(>|z|)
age              0.033261  1.033820  0.009135  3.641 0.000271 ***
edemaControlled  0.187339  1.206036  0.289005  0.648 0.516841
edemaPersistent  0.917835  2.503863  0.336303  2.729 0.006349 **
log(bili)        0.739865  2.095652  0.120992  6.115 9.66e-10 ***
albumin         -0.787279  0.455081  0.256427 -3.070 0.002139 **
protime          0.230749  1.259543  0.097866  2.358 0.018383 *
log(copper)      0.393806  1.482612  0.145676  2.703 0.006866 **
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                exp(coef) exp(-coef) lower .95 upper .95
age                1.0338     0.9673    1.0155    1.0525
edemaControlled    1.2060     0.8292    0.6845    2.1250
edemaPersistent    2.5039     0.3994    1.2952    4.8403
log(bili)          2.0957     0.4772    1.6532    2.6565
albumin            0.4551     2.1974    0.2753    0.7522
protime            1.2595     0.7939    1.0397    1.5259
log(copper)        1.4826     0.6745    1.1144    1.9725

Concordance= 0.85  (se = 0.02 )
Likelihood ratio test= 177.8  on 7 df,   p=<2e-16
Wald test            = 177.4  on 7 df,   p=<2e-16
Score (logrank) test = 255.8  on 7 df,   p=<2e-16
```

⚠ Stepwise Selection: Caveats

Problems: - Inflates Type I error (false positives) - P-values post-selection are too optimistic - Unstable—small data changes → big model changes - Ignores clinical importance
Better approaches: - **A priori model** based on literature/domain knowledge - **LASSO** for high-dimensional data with cross-validation - **Staged approach**: core clinical model → add interactions/non-linearity if justified

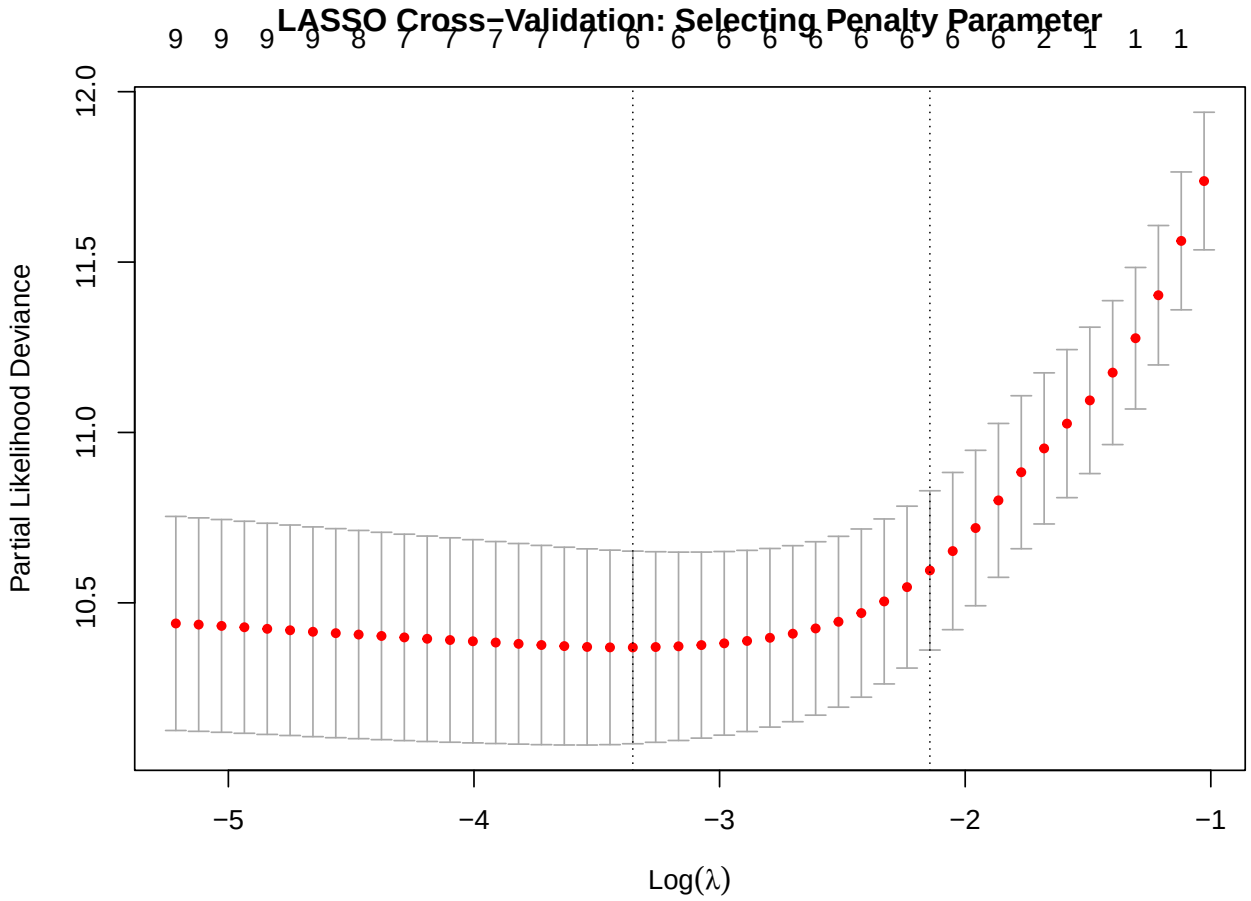
9.3 Advanced: LASSO Penalization

```r
# Prepare design matrix
X <- model.matrix(
  Surv(time_years, event) ~ age + sex + trt + edema +
    log(bili) + albumin + protime + log(chol) + log(copper),
  data = dat_all
)[, -1]  # Remove intercept

Y <- Surv(dat_all$time_years, dat_all$event)

# Fit LASSO with cross-validation
set.seed(123)
fit_lasso <- cv.glmnet(X, Y, family = "cox", alpha = 1, nfolds = 10)

# Plot cross-validation curve
plot(fit_lasso, main = "LASSO Cross-Validation: Selecting Penalty Parameter")
```



9.3.1 LASSO Selected Coefficients

```
# Coefficients at lambda.min (minimum CV error)
lasso_coefs <- coef(fit_lasso, s = "lambda.min")
cat("LASSO coefficients at lambda.min:\n")
```

```
LASSO coefficients at lambda.min:
```

```
print(lasso_coefs)
```

```
10 x 1 sparse Matrix of class "dgCMatrix"
                           1
age                0.02655208
sexFemale          .
trtPlacebo         .
edemaControlled    .
edemaPersistent    0.78286520
log(bili)          0.69812591
albumin           -0.69232388
protime            0.19519962
log(chol)          .
log(copper)        0.31412376
```

```r
cat("\n\nNon-zero coefficients:", sum(lasso_coefs != 0), "\n")
```

```
Non-zero coefficients: 6
```

> **i** When to Use LASSO
>
> **Good for:** - Many potential predictors (p > 20) - Weak effects scattered across many variables - Prediction-focused analyses
> **Not ideal for:** - Small sample sizes (n < 100) - Strong prior knowledge of important variables - Inference/interpretation is primary goal

---

10 Part 9: Team Exercise

10.1 Hands-On Practice (25 minutes)

> **!** Your Challenge
>
> Working in teams of 2–3, build and defend your "best" Cox model for PBC survival.
> **Requirements:**
>
> 1. Include **one interaction term** of your choice
> 2. Include **one non-linear term** (polynomial or spline)
> 3. Check and address **PH violations** (stratify if needed)
> 4. Run complete diagnostics (residuals, influential points)
> 5. Report final model performance (C-index, AIC)
> 6. Write a 2-3 sentence clinical interpretation

10.1.1 Exercise Template

```r
# STEP 1: Choose your interaction
# Ideas: trt*edema, trt*sex, age*sex, bili*albumin
my_interaction <- "trt * edema"  # Replace with your choice

# STEP 2: Choose non-linearity approach
# Ideas: poly(age, 2), pspline(age), poly(bili, 2), log transformations
my_nonlinear <- "poly(age, 2, raw = TRUE)"  # Replace with your choice

# STEP 3: Fit your model
my_model <- coxph(
  Surv(time_years, event) ~ "_____" + "_____" + "_____" + "_____", # Add your terms
  data = dat_core
)

# STEP 4: Run diagnostics
```

```r
## PH assumption
ph_check <- cox.zph(my_model)
print(ph_check)
ggcoxzph(ph_check)

## Functional form
ggcoxfunctional(Surv(time_years, event) ~ age + log(bili), data = dat_core)

## Residuals
ggcoxdiagnostics(my_model, type = "deviance")
ggcoxdiagnostics(my_model, type = "dfbeta")

# STEP 5: Address violations (if any)
## If PH violated, try stratification:
# my_model_v2 <- coxph(
#   Surv(time_years, event) ~ ... + strata(violating_variable),
#   data = dat_core
# )

# STEP 6: Final model summary
summary(my_model)

# STEP 7: Performance metrics
cat("C-index:", summary(my_model)$concordance[1], "\n")
cat("AIC:", AIC(my_model), "\n")

# STEP 8: Tidy results for interpretation
tidy(my_model, exponentiate = TRUE, conf.int = TRUE) %>%
  arrange(p.value) %>%
  kable(digits = 3)
```

10.1.2 Deliverable: Complete This Table

Component	Your Result
Interaction term used	
Non-linear term used	
Significant predictors (p<0.05)	
PH violations detected?	Yes / No
If yes, how addressed?	
Final C-index	
AIC	
Clinical interpretation (2-3 sentences)	

10.1.3 Guiding Questions

Discussion Points

1. **Why did you choose this interaction?** (Clinical rationale)
2. **Did non-linearity improve fit?** (Compare AIC before/after)
3. **Were there influential observations?** (How many? What to do?)
4. **How does your model compare to the baseline?** (Better discrimination?)
5. **What's ONE clinical recommendation** from your model?

---

11 Part 10: Debrief & Synthesis

11.1 Team Presentations (10 minutes)

Each team shares (2 min each):

1. **Model specification**: What you included and why
2. **Key finding**: Most important clinical insight
3. **Challenges**: What was difficult? How did you resolve it?
4. **Model quality**: C-index, any concerns about diagnostics

11.2 Key Takeaways

Core Principles for Cox Modeling

Model Building - Start simple, add complexity only when justified - Use clinical knowledge to guide variable selection - Respect the EPV rule (avoid overfitting)
Diagnostics Are Not Optional - Always check PH assumption (stratify if violated) - Inspect residuals for patterns - Investigate (but don't blindly remove) influential points
Interpretation Over Prediction - Can you explain your model to a clinician? - Are interactions clinically sensible? - Report uncertainty (confidence intervals, not just p-values)
Model Comparison - Use AIC + LRT for nested models - Prefer parsimony when models are equivalent (ΔAIC < 2) - C-index reflects discrimination, not calibration

11.3 Common Mistakes to Avoid

Pitfalls

1. **Ignoring PH violations** → Biased HRs, incorrect inference
2. **Too many variables** → Overfitting, unstable estimates
3. **Stepwise without validation** → Spurious associations
4. **Interpreting main effects when interactions present** → Wrong conclusion
5. **Removing "outliers" without justification** → Cherry-picking
6. **Forgetting clinical context** → Statistically significant clinically important

---

Appendix: Quick Reference

11.4 Diagnostic Checklist

Pre-Modeling

- ☐ Explore data (missingness, outliers, distributions)
- ☐ Kaplan-Meier curves by key groups
- ☐ Choose transformations (e.g., log for skewed variables)

11.5 Model Fitting

- ☐ Check sample size vs parameters (EPV 10)
- ☐ Fit baseline model (main effects only)
- ☐ Add interactions if justified
- ☐ Test non-linearity if suspected

11.6 Diagnostics

- ☐ Test PH assumption (`cox.zph`)
  - ☐ Stratify or transform if violated
- ☐ Check functional form (martingale residuals)
- ☐ Inspect deviance residuals (overall fit)
- ☐ Identify influential points (DFBeta)

11.7 Performance & Comparison

- ☐ Calculate C-index
- ☐ Compare models (AIC, LRT)
- ☐ Validate if sample size permits (bootstrap, cross-validation)

11.8 Reporting

- ☐ HR table with 95% CIs and p-values
- ☐ Forest plot or adjusted survival curves
- ☐ State sample size, events, follow-up time
- ☐ Interpret in clinical terms (not just statistics!)

11.9 Essential R Commands

```r
## Model Fitting
coxph(Surv(time, status) ~ x1 + x2, data = df)

## Interactions
coxph(Surv(time, status) ~ x1 * x2, data = df)  # x1 + x2 + x1:x2

## Non-linearity
coxph(Surv(time, status) ~ poly(x, 2, raw=TRUE), data = df)  # Quadratic
coxph(Surv(time, status) ~ pspline(x, df=4), data = df)      # Spline
```

```r
## Stratification (for PH violations)
coxph(Surv(time, status) ~ x1 + strata(x2), data = df)

## Diagnostics
cox.zph(fit)                                    # PH test
ggcoxzph(cox.zph(fit))                          # PH plots
ggcoxfunctional(Surv(time, status) ~ x, data=df) # Linearity check
ggcoxdiagnostics(fit, type="deviance")          # Deviance residuals
ggcoxdiagnostics(fit, type="dfbeta")            # Influential points

## Model Comparison
anova(fit1, fit2)                               # LRT (nested models)
AIC(fit1, fit2)                                 # AIC (any models)

## Performance
summary(fit)$concordance                        # C-index

## Visualization
ggforest(fit, data=df)                          # Forest plot
ggsurvplot(fit, ...)                            # Survival curves
ggadjustedcurves(fit, variable="x", data=df)   # Adjusted curves
```

11.10 Statistical Tests Summary

Test	Null Hypothesis	Use Case	R Function
Wald	$\beta_j = 0$	Individual coefficient significance	`summary(fit)`
Score	All $\beta = 0$ (global null)	Quick omnibus test	`summary(fit)`
LRT	Nested model fits equally well	Comparing nested models	`anova(fit1, fit2)`
Schoenfeld	PH assumption holds for covariate	Testing proportional hazards	`cox.zph(fit)`

11.11 C-Index Benchmarks

Table 8: Concordance Index Interpretation Guide

C-Index Range	Performance	Clinical Context
0.50	No discrimination	Model adds no value
0.51–0.60	Very weak	Minimal clinical utility
0.61–0.70	Weak	Some prognostic value
0.71–0.80	Acceptable	Useful for risk stratification
0.81–0.90	Good to Excellent	Strong clinical tool
0.91–1.00	Nearly perfect	Check for overfitting!

11.12 Recommended Readings

Essential References

1. **Therneau, T. M. & Grambsch, P. M.** (2000). *Modeling Survival Data: Extending the Cox Model.* Springer.
   - Definitive technical reference
2. **Hosmer, D. W., Lemeshow, S., & May, S.** (2008). *Applied Survival Analysis* (2nd ed.). Wiley.
   - Accessible applied approach
3. **Harrell, F. E.** (2015). *Regression Modeling Strategies* (2nd ed.). Springer.
   - Advanced modeling philosophy; excellent on validation
4. **Schoenfeld, D.** (1982). Partial residuals for the proportional hazards regression model. *Biometrika*, 69(1), 239–241.
   - Original PH diagnostic paper
5. **Peduzzi, P., et al.** (1995). Importance of events per independent variable in proportional hazards regression analysis II. Accuracy and precision of regression estimates. *J Clin Epidemiol*, 49(12), 1373–1379.
   - EPV rule derivation
6. **Grambsch, P. M. & Therneau, T. M.** (1994). Proportional hazards tests and diagnostics based on weighted residuals. *Biometrika*, 81(3), 515–526.
   - Extended diagnostics
7. **Royston, P. & Altman, D. G.** (2013). External validation of a Cox prognostic model: principles and methods. *BMC Med Res Methodol*, 13(1), 33.
   - Validation strategies

Practice Problems

Work through these on your own to reinforce concepts.

11.13 Problem 1: Building & Interpreting

Using the `pbc` data:

1. Fit a Cox model with `age`, `sex`, `log(bili)`, and `albumin`
2. Interpret the HR for albumin in clinical terms
3. Calculate the HR for someone 10 years older (holding other variables constant)
4. Test whether adding `protime` significantly improves the model

Hints

- Use `tidy(fit, exponentiate=TRUE, conf.int=TRUE)` for neat output
- For age effect: HR for 10 years = $(e^{\beta_{\text{age}}})^{10}$
- Compare models with `anova(fit1, fit2)`

11.14 Problem 2: Interactions

1. Fit a model with `trt`, `sex`, and their interaction
2. Calculate the treatment HR separately for males and females
3. Test whether the interaction is significant
4. Create adjusted survival curves for males vs females

11.15 Problem 3: Diagnostics

Using your model from Problem 1:

1. Test the PH assumption for each covariate
2. Check for non-linearity in age and bilirubin
3. Identify the 5 most influential observations (DFBeta)
4. If PH is violated for any variable, fit a stratified model and compare results

11.16 Problem 4: Model Selection

1. Fit three models with different combinations of predictors
2. Compare them using AIC and LRT
3. Calculate C-index for each
4. Which model would you choose? Justify based on:
    - Statistical criteria
    - Clinical interpretability
    - EPV considerations

Model Ideas

- **Model A:** Age + sex + bili (simple)
- **Model B:** Age + sex + bili + albumin + protime (clinical standard)
- **Model C:** Model B + age×sex interaction (complex)

11.17 Problem 5: Capstone Analysis

Scenario: You're consulting on a clinical trial for PBC treatment.

1. Build a **prognostic model** (ignore treatment) using available baseline variables
2. Check all assumptions and diagnostics
3. Report C-index and interpret
4. Add `trt` to your prognostic model—does treatment improve outcomes after adjusting for prognosis?
5. Write a short paragraph (5-7 sentences) explaining your findings to a non-statistician collaborator

---

❗ Next Steps

To deepen your skills: - Apply these methods to your own data - Try bootstrap validation of your models - Explore competing risks (Fine-Gray models) - Learn time-dependent covariates for PH violations - Study calibration plots (predicted vs observed survival)
Resources: - `survival` package vignettes: `vignette(package="survival")` - Harrell's RMS course: https://hbiostat.org/rms/ - DataCamp: "Survival Analysis in R"

Thank you! Questions?

← Return to Course Materials

Link to qmd (quarto markdown)