R code for Survival Lecture #2

Spring 2025

Contents

Setup

Load all required packages at once

```
# Load required packages for data import, survival analysis, plotting, and arranging multiple plots
packages_to_load <- c("haven", "survival", "survminer", "gridExtra", "dplyr",
                       "gtsummary","biostat3")

invisible(lapply(packages_to_load, function(pkg) {
  suppressPackageStartupMessages(library(pkg, character.only = TRUE))
}))
```

Load PBC Data

```
# Import the PBC dataset from a Stata (.dta) file into R
PBC <- read_dta("pbc.dta")
head(PBC)
```

```
## # A tibble: 6 x 23
##   number status    rx         sex asictes hepatom spiders edema bilirubin cholest
##    <dbl> <dbl+lbl> <dbl+l> <dbl> <dbl+l> <dbl+l> <dbl+l> <dbl>     <dbl>   <dbl>
## 1      1 1 [Dead]  0 [Pla~     1 1 [Yes] 1 [Yes] 1 [Yes]     1    14.5       261
## 2      2 0 [Censo~ 0 [Pla~     1 0 [No]  1 [Yes] 1 [Yes]     0     1.10      302
## 3      3 1 [Dead]  0 [Pla~     0 0 [No]  0 [No]  0 [No]      1     1.40      176
## 4      4 1 [Dead]  0 [Pla~     1 0 [No]  1 [Yes] 1 [Yes]     1     1.80      244
## 5      5 0 [Censo~ 1 [DPC~     1 0 [No]  1 [Yes] 1 [Yes]     0     3.40      279
## 6      6 1 [Dead]  1 [DPC~     1 0 [No]  1 [Yes] 0 [No]      0     0.800     248
## # i 13 more variables: albumin <dbl>, copper <dbl>, alkphos <dbl>, sgot <dbl>,
## #   trigli <dbl>, platel <dbl>, prothrom <dbl>, histol <dbl>, age <dbl>,
## #   years <dbl>, logbili <dbl>, logalbu <dbl>, logprot <dbl>
```

Cox Regrssion With a Categorical Variable (Histology)

Fit a Cox model with histology as a categorical variable

Note that the default approach for handling ties used by the “coxph” function is Efron’s method, which offers higher accuracy when there is a large number of ties. To obtain results consistent with STATA output, apply the Breslow method by specifying ties=“breslow”; this method may be less accurate but is somewhat faster to compute.

```r
# Convert 'histol' from numeric to a factor with descriptive labels for each histology stage
PBC$histol <- factor(PBC$histol, levels=1:4, labels=c("Stage.I", "Stage.II", "Stage.III", "Stage.IV"))

# Surv(years, status) defines the survival object
cox.histol <- coxph(Surv(years, status) ~ histol, data = PBC)

# Display a detailed summary of the fitted Cox model:
# coef: log hazard ratio for each histology stage, compared to a reference group (here, Stage 1)
# exp(coef): hazard ratio for each histology stage, compared to a reference group (here, Stage 1)
# se(coef) = standard error of the coefficient
# z = Wald test statistic
# Pr(>|z|) = p-value for testing HR = 1

# Additionally, the summary provides 95% confidence intervals for each hazard ratio,
# a concordance statistic reflects the model's predictive accuracy,
# and results from global tests (Likelihood ratio, Wald, and Score tests).


summary(cox.histol)
```

```
## Call:
## coxph(formula = Surv(years, status) ~ histol, data = PBC)
##
##   n= 312, number of events= 125
##
##                    coef exp(coef) se(coef)     z Pr(>|z|)
## histolStage.II   1.607     4.988    1.031 1.559   0.1191
## histolStage.III  2.150     8.581    1.012 2.124   0.0337 *
## histolStage.IV   3.063    21.387    1.009 3.036   0.0024 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                  exp(coef) exp(-coef) lower .95 upper .95
## histolStage.II       4.988    0.20049    0.6611     37.64
## histolStage.III      8.581    0.11654    1.1800     62.39
## histolStage.IV      21.387    0.04676    2.9606    154.50
##
## Concordance= 0.702  (se = 0.022 )
## Likelihood ratio test= 52.74  on 3 df,   p=2e-11
## Wald test            = 43.92  on 3 df,   p=2e-09
## Score (logrank) test = 53.85  on 3 df,   p=1e-11
```

```r
# Present results in a clean table with exponentiated coefficients (HRs) and 95% CIs
# library(gtsummary)  # Uncomment if not already loaded
tbl_regression(cox.histol,
               exponentiate = TRUE,    # show hazard ratios instead of log(HR)
               conf.level = 0.95)      # 95% confidence intervals
```

Characteristic	HR	95% CI	p-value
histol			
Stage.I	—	—	
Stage.II	4.99	0.66, 37.6	0.12
Stage.III	8.58	1.18, 62.4	0.034
Stage.IV	21.4	2.96, 154	0.002

Comparing Stage III vs Stage II

> Use lincom() from *biostat3* package: You can calculate the hazard ratio between any two groups using the lincom() function from the biostat3 package. This function performs contrast tests to compare the model's coefficients directly without needing to re-fit the model.

```r
## Compare Stage III to Stage II in the Cox model using lincom()
# - This computes: log(HR of Stage III vs I) - log(HR of StageII vs I))
# - eform = TRUE returns the hazard ratio (HR) instead of the log-HR

#library(biostat3)  # Uncomment if not already loaded
lincom(cox.histol, "histolStage.III - histolStage.II", eform = TRUE)
```

```
##                                  Estimate     2.5 %   97.5 %    Chisq
## histolStage.III - histolStage.II 1.720279 0.9680236 3.057115 3.419484
##                                  Pr(>Chisq)
## histolStage.III - histolStage.II 0.06443115
```

Comparing Stage IV vs Stage III

```r
lincom(cox.histol, "histolStage.IV - histolStage.III", eform = TRUE)
```

```
##                                  Estimate    2.5 %   97.5 %    Chisq
## histolStage.IV - histolStage.III 2.492472 1.692194 3.671221 21.36575
##                                    Pr(>Chisq)
## histolStage.IV - histolStage.III 3.794904e-06
```

Trend test

The code below fits a Cox proportional hazards model using the ordered stage as a predictor to evaluate the linear relationship between the ordered stage and survival.

```
# Fit a Cox proportional hazards model treating 'histol' as an
# ordered variable
# - This tests for a linear trend in hazard across the stage levels
# - Assumes that higher stages are associated with increasing hazard
# - 'ordered(stage)' automatically treats the factor as numeric ranks (1 < 2 < 3 < 4)

fit.trend <- coxph(Surv(years, status) ~ ordered(histol), data = PBC)

# View model summary
# - Check the coefficient and p-value for the ordered stage term
# - A significant positive coefficient suggests an increasing trend in hazard with stage
summary(fit.trend)
```

```
## Call:
## coxph(formula = Surv(years, status) ~ ordered(histol), data = PBC)
##
##   n= 312, number of events= 125
##
##                      coef exp(coef) se(coef)      z Pr(>|z|)
## ordered(histol).L  2.1759    8.8099   0.6801  3.199  0.00138 **
## ordered(histol).Q -0.3469    0.7069   0.5248 -0.661  0.50867
## ordered(histol).C  0.3209    1.3784   0.2990  1.073  0.28316
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                   exp(coef) exp(-coef) lower .95 upper .95
## ordered(histol).L    8.8099     0.1135    2.3231    33.410
## ordered(histol).Q    0.7069     1.4146    0.2527     1.977
## ordered(histol).C    1.3784     0.7255    0.7671     2.477
##
## Concordance= 0.702  (se = 0.022 )
## Likelihood ratio test= 52.74  on 3 df,   p=2e-11
## Wald test            = 43.92  on 3 df,   p=2e-09
## Score (logrank) test = 53.85  on 3 df,   p=1e-11
```

Cox Regrssion With a Continuous Variable (Age)

Effect of age in years

```
cox.age <- coxph(Surv(years, status) ~ age, data = PBC)
summary(cox.age)
```

```
## Call:
## coxph(formula = Surv(years, status) ~ age, data = PBC)
##
##   n= 312, number of events= 125
##
##         coef exp(coef) se(coef)     z Pr(>|z|)
## age 0.039995  1.040806 0.008811 4.539 5.65e-06 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
##     exp(coef) exp(-coef) lower .95 upper .95
## age     1.041     0.9608     1.023     1.059
##
## Concordance= 0.625  (se = 0.027 )
## Likelihood ratio test= 20.51  on 1 df,   p=6e-06
## Wald test            = 20.6  on 1 df,   p=6e-06
## Score (logrank) test = 20.86  on 1 df,   p=5e-06
```

```
# Present results in a clean table with exponentiated coefficients (HRs) and 95% CIs
# library(gtsummary)  # Uncomment if not already loaded
tbl_regression(cox.age,
               exponentiate = TRUE,    # show hazard ratios instead of log(HR)
               conf.level = 0.95)      # 95% confidence intervals
```

Characteristic	HR	95% CI	p-value
Age (years)	1.04	1.02, 1.06	<0.001

Effect of age in days

```
PBC$age_days <- PBC$age * 365.25
cox.age.days <- coxph(Surv(years, status) ~ age_days, data = PBC)
summary(cox.age.days)
```

```
## Call:
## coxph(formula = Surv(years, status) ~ age_days, data = PBC)
##
##   n= 312, number of events= 125
##
```

```
##                coef exp(coef)  se(coef)     z Pr(>|z|)
## age_days 1.095e-04 1.000e+00 2.412e-05 4.539 5.65e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##          exp(coef) exp(-coef) lower .95 upper .95
## age_days         1     0.9999         1         1
##
## Concordance= 0.625  (se = 0.027 )
## Likelihood ratio test= 20.51  on 1 df,   p=6e-06
## Wald test            = 20.6  on 1 df,   p=6e-06
## Score (logrank) test = 20.86  on 1 df,   p=5e-06
```

```
tbl_regression(cox.age.days,
               exponentiate = TRUE,    # show hazard ratios instead of log(HR)
               conf.level = 0.95)      # 95% confidence intervals
```

Characteristic	HR	95% CI	p-value
Age (years)	1.00	1.00, 1.00	<0.001

Effect of age in decades

```
PBC$age_decades<- PBC$age /10
cox.age.decades <- coxph(Surv(years, status) ~ age_decades, data = PBC)
summary(cox.age.decades)
```

```
## Call:
## coxph(formula = Surv(years, status) ~ age_decades, data = PBC)
##
##   n= 312, number of events= 125
##
##                 coef exp(coef) se(coef)     z Pr(>|z|)
## age_decades 0.39995   1.49175  0.08811 4.539 5.65e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##             exp(coef) exp(-coef) lower .95 upper .95
## age_decades     1.492     0.6704     1.255     1.773
##
## Concordance= 0.625  (se = 0.027 )
## Likelihood ratio test= 20.51  on 1 df,   p=6e-06
## Wald test            = 20.6  on 1 df,   p=6e-06
## Score (logrank) test = 20.86  on 1 df,   p=5e-06
```

```
tbl_regression(cox.age.decades,
               exponentiate = TRUE,    # show hazard ratios instead of log(HR)
               conf.level = 0.95)      # 95% confidence intervals
```

Characteristic	**HR**	**95% CI**	**p-value**
Age (years)	1.49	1.26, 1.77	<0.001

Equivalently, lincom can give the same result:

```r
lincom(cox.age, "10 * age", eform = TRUE)
```

```
##          Estimate    2.5 %   97.5 %    Chisq   Pr(>Chisq)
## 10 * age 1.491752 1.255145 1.772961 20.60277 5.651415e-06
```

Adjusted Survival Curves

Fit a multivariate Cox Model with sex and copper level as covariates

We include sex (0=male, 1=female) and copper level as predictors in the Cox model.

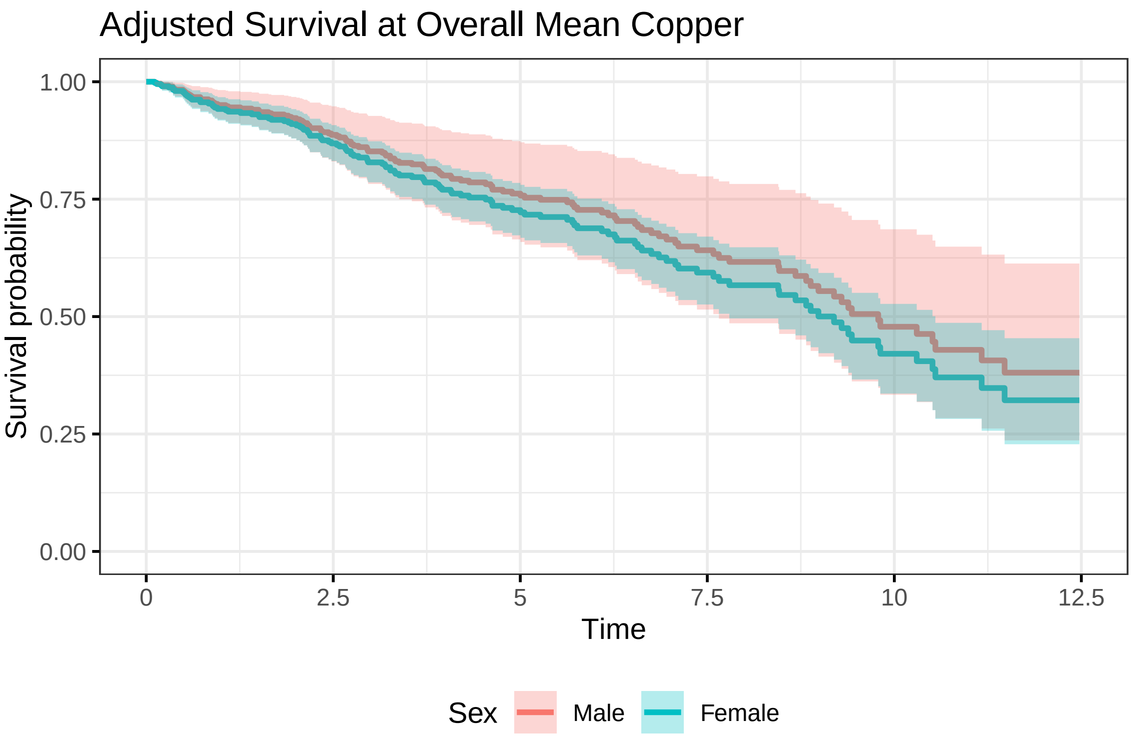
```
# Fit a Cox proportional hazards model with 'sex' and 'copper' as predictors.
# Surv(years, status) defines the survival object

cox.reg <- coxph(Surv(years, status) ~ sex + copper, data = PBC)

# Display a detailed summary of the fitted Cox proportional hazards model
# This single command prints:
#   The original model call and dataset info (number of observations & events)
#   A coefficients table showing for each predictor:
#      - coef: log(hazard ratio)
#      - exp(coef): hazard ratio (HR)
#      - se(coef): standard error of the log-HR
#      - z: Wald test statistic (coef divided by se)
#      - Pr(>|z|): p-value testing whether HR != 1
#   Three global tests of model fit (Likelihood ratio, Wald, Score) with statistics & p-values
#   Concordance statistic (C-index) indicating predictive discrimination

summary(cox.reg)
```

```
## Call:
## coxph(formula = Surv(years, status) ~ sex + copper, data = PBC)
##
##   n= 310, number of events= 124
##    (2 observations deleted due to missingness)
##
##              coef exp(coef)  se(coef)     z Pr(>|z|)
## sex    0.1600974 1.1736252 0.2558346 0.626    0.531
## copper 0.0069205 1.0069445 0.0008279 8.359   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##        exp(coef) exp(-coef) lower .95 upper .95
## sex        1.174     0.8521    0.7108     1.938
## copper     1.007     0.9931    1.0053     1.009
##
## Concordance= 0.736  (se = 0.023 )
## Likelihood ratio test= 55.35  on 2 df,   p=1e-12
## Wald test            = 77.6  on 2 df,   p=<2e-16
## Score (logrank) test = 85.84  on 2 df,   p=<2e-16
```

Adjusted Survival Curves at Overall Mean Copper

Compute the mean copper level (ignoring missing values) and generate adjusted survival curves for males vs. females holding copper fixed at that mean.

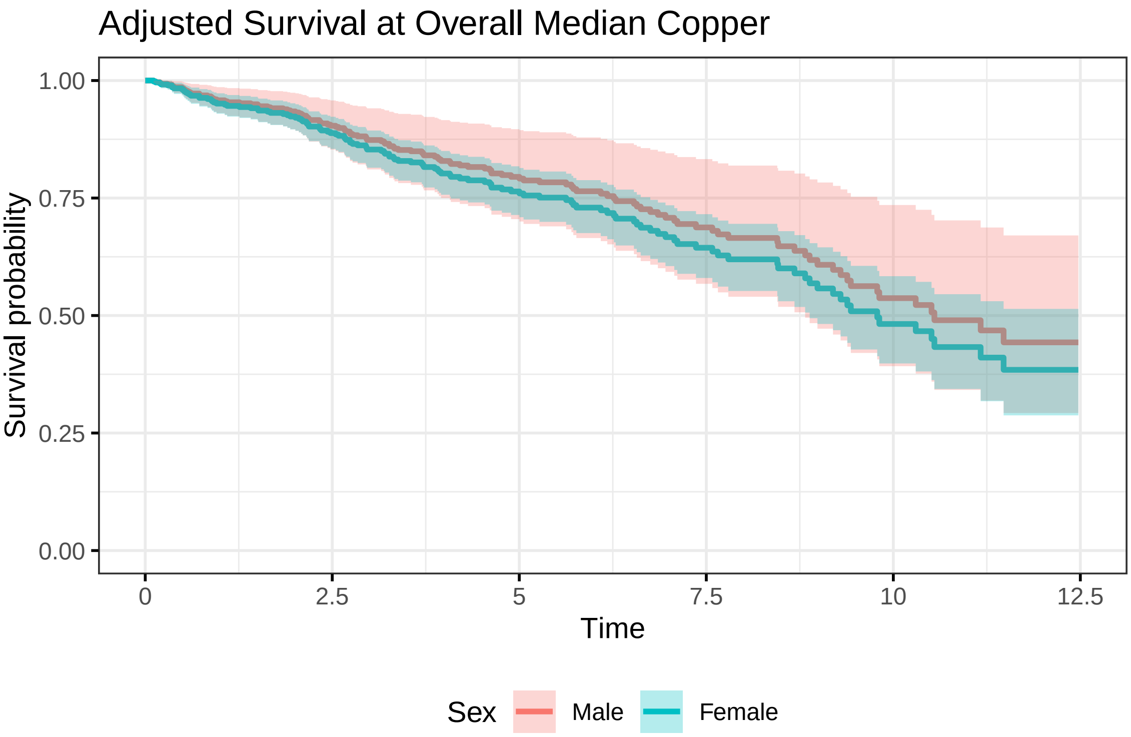
```
# Compute the overall mean copper level (exclude missing values)
mean_copper <- mean(PBC$copper, na.rm = TRUE)

# Create a new data frame for prediction: one row per sex (0=Male, 1=Female),
# with copper being set to the overall mean
new_data_mean <- data.frame(sex = 0:1, copper = mean_copper)

# Generate adjusted survival curves at the overall mean copper level
surv_mean <- survfit(cox.reg, newdata = new_data_mean)

# Plot adjusted survival curves (mean copper) with legend at bottom
ggsurvplot(surv_mean, data = PBC,
           censor = FALSE,
           title = "Adjusted Survival at Overall Mean Copper",
           legend = "bottom", legend.title = "Sex",
           legend.labs = c("Male", "Female"),
           ggtheme = theme_bw())
```



Adjusted Survival Curves at Overall Median Copper

Repeat using the median copper level.

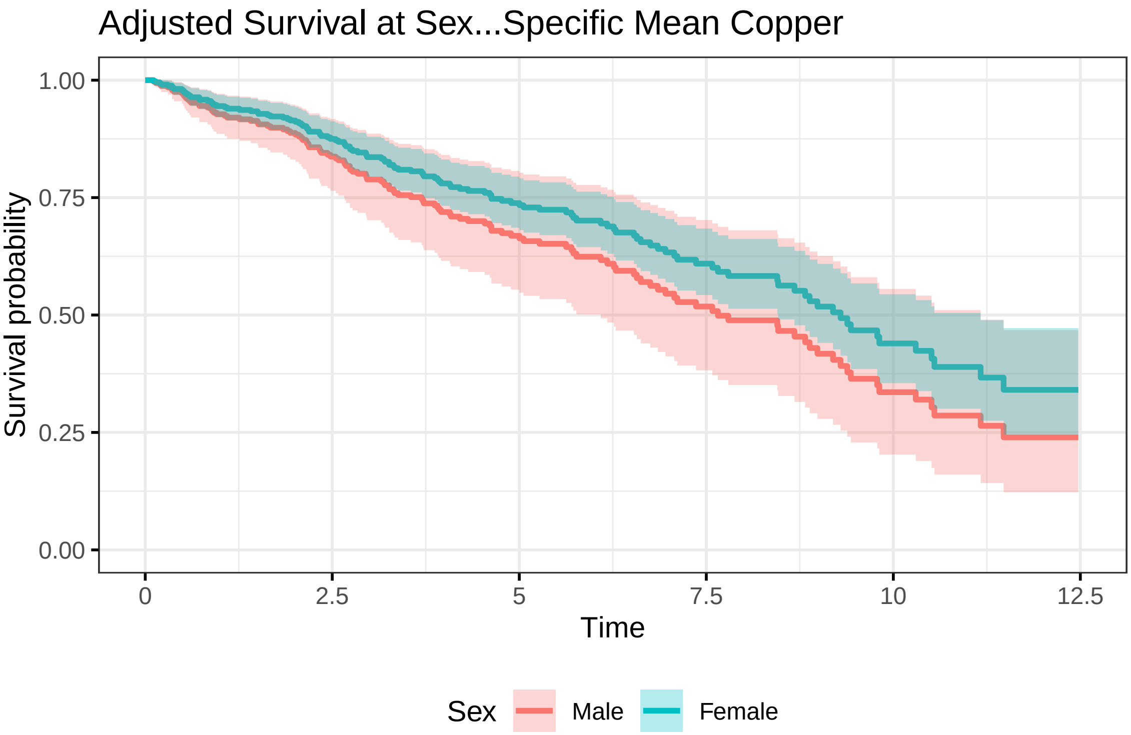
```r
# Compute the overall median copper level (exclude missing values)
median_copper <- median(PBC$copper, na.rm = TRUE)

# New data frame for prediction at the median copper level
new_data_median <- data.frame(sex = 0:1, copper = median_copper)

# Generate adjusted survival curves at the overall median copper level
surv_median <- survfit(cox.reg, newdata = new_data_median)

# Plot adjusted survival curves (median copper)
ggsurvplot(surv_median, data = PBC,
           censor = FALSE,
           title = "Adjusted Survival at Overall Median Copper",
           legend = "bottom", legend.title = "Sex",
           legend.labs = c("Male", "Female"),
           ggtheme = theme_bw())
```



Sex-Specific Adjusted Curves (Mean Copper by Sex)

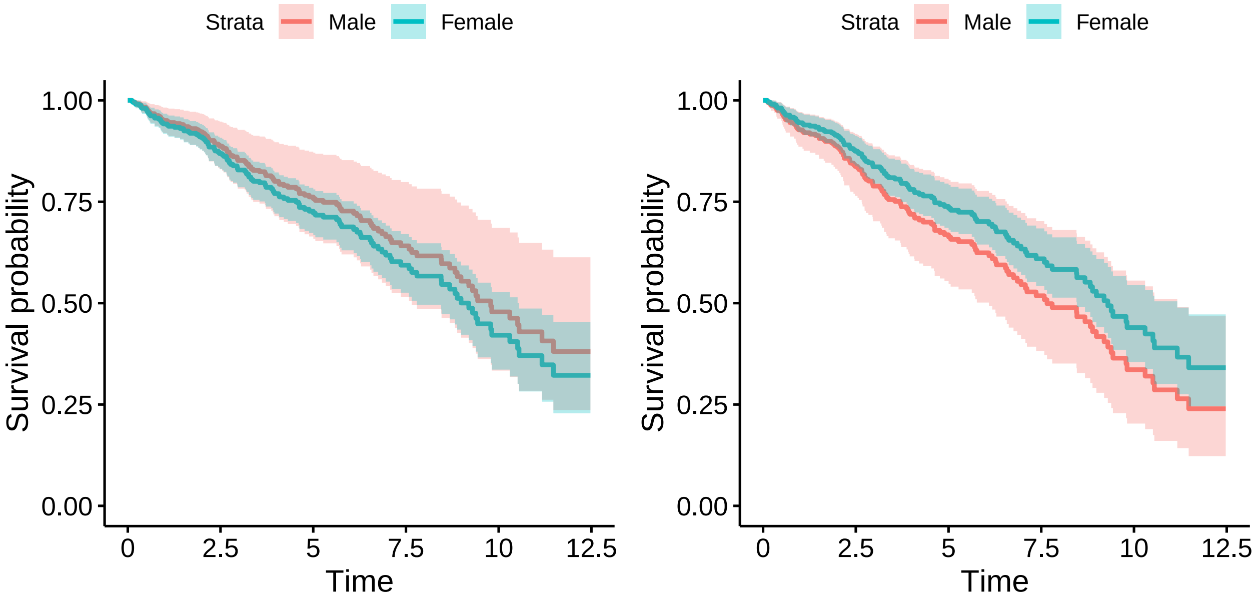
Compute each gender group's mean copper level, then plot survival curves holding copper at each group's own mean.

```
# Calculate mean copper separately by sex for sex-specific adjustment
mean_copper_by_sex <- aggregate(copper ~ sex, data = PBC, FUN = mean, na.rm = TRUE)

# Generate survival curves at each group's own mean copper level
surv_sex <- survfit(cox.reg, newdata = mean_copper_by_sex)

# Plot sex-specific adjusted survival curves
ggsurvplot(surv_sex, data = PBC,
           censor = FALSE,
           title = "Adjusted Survival at Sex-Specific Mean Copper",
           legend = "bottom", legend.title = "Sex",
           legend.labs = c("Male", "Female"),
           ggtheme = theme_bw())
```



Compare Plots Side-by-Side

```r
# Arrange the "mean copper" and "sex-specific copper" plots side by side for comparison
p_mean <- ggsurvplot(surv_mean, data = PBC,
           censor = FALSE,
           legend.labs = c("Male", "Female"))
p_sex  <- ggsurvplot(surv_sex, data = PBC,
           censor = FALSE,
           legend.labs = c("Male", "Female"))

grid.arrange(p_mean$plot, p_sex$plot, ncol = 2)
```


Strata
Male
Female
Survival probability
Time
Strata
Male
Female
Survival probability
Time