BIOSTAT M215 HW2

Hanbei Xiong

```
library(powertools)
```

```
## Warning: package 'powertools' was built under R version 4.3.3
```

```
library(survival)  # loading survival functions into R
```

```
## Warning: package 'survival' was built under R version 4.3.3
```

```
library(KMsurv)    # datasets from Klein and Moeschberger textbook
```

```
## Warning: package 'KMsurv' was built under R version 4.3.3
```

```
library(SurvRegCensCov)
```

```
## Warning: package 'SurvRegCensCov' was built under R version 4.3.3
```

```
## Registered S3 method overwritten by 'SurvRegCensCov':
##   method    from
##   print.src dplyr
```

Exercise 3.5.

Suppose that the time to death has a log logistic distribution with parameters λ and α. Based on the following left-censored sample, construct the likelihood function. DATA: 0.5, 1, 0.75, 0.25-, 1.25-, where - denotes a left- censored observation.

Answer:

$$L(\lambda,\alpha) = f(0.5)f(1)f(0.75)[1-S(0.25)][1-S(1.25)]$$

$$L(\lambda,\alpha) = \frac{\alpha(0.5)^{\alpha-1}\lambda}{[1+\lambda(0.5)^{\alpha}]^2}\frac{\alpha(1)^{\alpha-1}\lambda}{[1+\lambda(1)^{\alpha}]^2}\frac{\alpha(0.75)^{\alpha-1}\lambda}{[1+\lambda(0.75)^{\alpha}]^2}[1-\frac{1}{1+\lambda(0.25)^{\alpha}}][1-\frac{1}{1+\lambda(1.25)^{\alpha}}]$$

Exercise 3.6

The following data consists of the times to relapse and the times to death following relapse of 10 bone marrow transplant patients. In the sample patients 4 and 6 were alive in relapse at the end of the study and patients 7–10 were alive, free of relapse at the end of the study. Suppose the time to relapse had an exponential distribution with hazard rate λ and the time to death in relapse had a Weibull distribution with parameters θ and α.

Patient	Relapse Time (months)	Death Time (months)
1	5	11
2	8	12
3	12	15
4	24	33+
5	32	45
6	17	28+

Patient	Relapse Time (months)	Death Time (months)
7	16+	16+
8	17+	17+
9	19+	19+
10	30+	30+

+ Censored observation

a. Construct the likelihood for the relapse rate λ.
b. Construct a likelihood for the parameters θ and α.
c. Suppose we were only allowed to observe a patients death time if the patient relapsed. Construct the likelihood for θ and α based on this truncated sample, and compare it to the results in (b).

(a) Answer:

$L(\lambda) = e^{-5\lambda}\, e^{-8\lambda}\, e^{-12\lambda}\, e^{-24\lambda}\, e^{-32\lambda}\, e^{-17\lambda}\, \lambda e^{-16\lambda}\, \lambda e^{-17\lambda}\, \lambda e^{-19\lambda}\, \lambda e^{-30\lambda} = \lambda^4 e^{-(5+8+12+24+32+17+16+17+19+30)\lambda} = \lambda^4 e^{-180\lambda}$

(b) Answer:

Uncersored part of Likelihood = $\alpha^4\theta^4(11\times 12\times 15\times 45)^{\alpha-1} e^{-\theta(11^\alpha+12^\alpha+15^\alpha+45^\alpha)} = \alpha^4\theta^4(89100)^{\alpha-1} e^{-\theta(11^\alpha+12^\alpha+15^\alpha+45^\alpha)}$

Censored part of Likelihood = $e^{-\theta(33^\alpha+28^\alpha+16^\alpha+17^\alpha+19^\alpha+30^\alpha)}$

$L(\lambda,\alpha) = \alpha^4\theta^4(89100)^{\alpha-1} e^{-\theta(11^\alpha+12^\alpha+15^\alpha+45^\alpha)}\; e^{-\theta(33^\alpha+28^\alpha+16^\alpha+17^\alpha+19^\alpha+30^\alpha)} = \alpha^4\theta^4(89100)^{\alpha-1} e^{-\theta(11^\alpha+12^\alpha+15^\alpha+45^\alpha+33^\alpha+28^\alpha+16^\alpha+17^\alpha+19^\alpha+30^\alpha)}$

(c) Answer:

We remove patient 7-10 from the likelihood function since they did not relapse.

Uncensored part of Likelihood is unchanged

Censored part of Likelihood = $e^{-\theta(33^\alpha+28^\alpha)}$

$L(\lambda,\alpha) = \alpha^4\theta^4(89100)^{\alpha-1} e^{-\theta(11^\alpha+12^\alpha+15^\alpha+45^\alpha)}\; e^{-\theta(33^\alpha+28^\alpha)} = \alpha^4\theta^4(89100)^{\alpha-1} e^{-\theta(11^\alpha+12^\alpha+15^\alpha+45^\alpha+33^\alpha+28^\alpha)}$

We take the ratio between the likelihoods in (b) and (c) to see the effect of truncation on likelihood.

$ratio = \frac{\alpha^4\theta^4(89100)^{\alpha-1} e^{-\theta(11^\alpha+12^\alpha+15^\alpha+45^\alpha+33^\alpha+28^\alpha+16^\alpha+17^\alpha+19^\alpha+30^\alpha)}}{\alpha^4\theta^4(89100)^{\alpha-1} e^{-\theta(11^\alpha+12^\alpha+15^\alpha+45^\alpha+33^\alpha+28^\alpha)}} = e^{-\theta(16^\alpha+17^\alpha+19^\alpha+30^\alpha)}$

Exercise 4.2

Using the data reported in section 1.3, find the quantities specified below for the AML low risk and AML high risk groups. Note that most of these quantities are worked out in detail in Example 4.2 and its continuations for the ALL group.

a. Estimate the survival functions and their standard errors for the AML low risk and AML high risk groups.
b. Estimate the cumulative hazard rates and their standard errors for the AML low risk and AML high risk groups.
c. Estimate the mean time to death and find 95% confidence intervals for the mean survival time for both the AML low risk and AML high risk groups. (Answers are given in section 4.5.)
d. Work out estimates of the median time to death and find 95% con- fidence intervals for the median survival time for both the AML low risk and AML high risk groups using the linear, log-transformed, and arcsine formulas. (Answers are given in section 4.5.)
e. Find 95% confidence intervals for the survival functions at 300 days post-transplant for both the AML low risk and AML high risk groups using the log- and arcsine-transformed formulas.
f. Find 95% EP confidence bands for the survival functions over the range 100–400 days post-transplant for both the AML low risk and AML high risk groups using the linear, log-transformed, and arcsine- transformed formulas.
g. Find 95% HW confidence bands for the survival functions over the range 100–400 days post-transplant for both the AML low risk and AML high risk groups using the linear, log-transformed, and arcsine- transformed formulas.
h. Based on the results above and those discussed in Example 4.2 and its continuations, how do the survival experiences of the ALL, AML low risk, and AML high risk groups compare?

(a)-(b) Answer:

```
library(KMsurv)

data(bmt)
```

```
low_risk <- bmt[bmt$group==2,]
high_risk <- bmt[bmt$group==3,]
```

```
estimate_survival_hazard <- function(data) {
  # Aggregate total events at each unique time point
  event_summary <- aggregate(d3 ~ t2, data = data, sum)

  # Filter out time points where no events occurred
  event_summary <- event_summary[event_summary$d3 > 0, ]

  # Number of unique event times
  n <- nrow(event_summary)

  # Initialize vectors
  estimate_surv <- rep(1, n)
  Y <- rep(0, n)
  part_in_var <- rep(0, n)
  var <- rep(0, n)
  hazard <- rep(0, n)
  var_hazard <- rep(0, n)

  # Initial at-risk population
  Y[1] <- nrow(data)

  for (i in 1:n) {
    if (i == 1) {
      estimate_surv[i] <- 1 - event_summary$d3[i] / Y[i]
      Y[i + 1] <- Y[i] - event_summary$d3[i]
      part_in_var[i] <- event_summary$d3[i] / (Y[i] * (Y[i] - event_summary$d3[i]))
      var[i] <- estimate_surv[i]^2 * part_in_var[i]
      hazard[i] <- event_summary$d3[i] / Y[i]
      var_hazard[i] <- event_summary$d3[i] / Y[i]^2
    } else {
      estimate_surv[i] <- estimate_surv[i - 1] * (1 - event_summary$d3[i] / Y[i])
      part_in_var[i] <- part_in_var[i - 1] + event_summary$d3[i] / (Y[i] * (Y[i] - event_summary$d3[i]))
      var[i] <- estimate_surv[i]^2 * part_in_var[i]
      hazard[i] <- hazard[i - 1] + event_summary$d3[i] / Y[i]
      var_hazard[i] <- var_hazard[i - 1] + event_summary$d3[i] / Y[i]^2

      if (i < n) {
        Y[i + 1] <- Y[i] - event_summary$d3[i]
      }
    }
  }

  # Create dataframe and round values to 4 decimal places
  result <- data.frame(
    Time_to_Event = event_summary$t2,
    Total_Events = event_summary$d3,
    At_Risk = round(Y[1:n], 4),
    Survival_Estimate = round(estimate_surv, 4),
    Survival_SE = round(sqrt(var), 4),
    Cumulative_Hazard = round(hazard, 4),
    Hazard_SE = round(sqrt(var_hazard), 4)
  )

  return(result)
}
```

```
estimate_survival_hazard(low_risk)
```

Time_to_Event <int>	Total_Events <int>	At_Risk <dbl>	Survival_Estimate <dbl>	Survival_SE <dbl>	Cumulative_Hazard <dbl>	Hazard_SE <dbl>
10	1	54	0.9815	0.0183	0.0185	0.0185
35	1	53	0.9630	0.0257	0.0374	0.0264
48	1	52	0.9444	0.0312	0.0566	0.0327
53	1	51	0.9259	0.0356	0.0762	0.0381
79	1	50	0.9074	0.0394	0.0962	0.0430
80	1	49	0.8889	0.0428	0.1166	0.0476
105	1	48	0.8704	0.0457	0.1375	0.0520
211	1	47	0.8519	0.0483	0.1587	0.0562
219	1	46	0.8333	0.0507	0.1805	0.0602
248	1	45	0.8148	0.0529	0.2027	0.0642

1-10 of 25 rows Previous **1** 2 3 Next

```
estimate_survival_hazard(high_risk)
```

Time_to_Event <int>	Total_Events <int>	At_Risk <dbl>	Survival_Estimate <dbl>	Survival_SE <dbl>	Cumulative_Hazard <dbl>	Hazard_SE <dbl>
2	1	45	0.9778	0.0220	0.0222	0.0222
16	1	44	0.9556	0.0307	0.0449	0.0318
32	1	43	0.9333	0.0372	0.0682	0.0394
47	2	42	0.8889	0.0468	0.1158	0.0518
48	1	40	0.8667	0.0507	0.1408	0.0575
63	1	39	0.8444	0.0540	0.1665	0.0630
64	1	38	0.8222	0.0570	0.1928	0.0683
74	1	37	0.8000	0.0596	0.2198	0.0734
76	1	36	0.7778	0.0620	0.2476	0.0785
80	1	35	0.7556	0.0641	0.2762	0.0835

1-10 of 33 rows Previous **1** 2 3 4 Next

(c) Answer:

```
low_risk <- bmt[bmt$group==2,]
high_risk <- bmt[bmt$group==3,]

max_time_low_risk <- max(low_risk$t2)
max_time_high_risk <- max(high_risk$t2)

low_risk_estimate <- estimate_survival_hazard(low_risk)
high_risk_estimate <- estimate_survival_hazard(high_risk)
```

```
getMean_Var <- function(estimate, maximum) {
  estimate_descend <- estimate[order(-estimate$Time_to_Event), ]
  mean_temp_record <- rep(0, nrow(estimate_descend)+1)
  mean_temp <- estimate_descend$Survival_Estimate[1]*(maximum-estimate_descend$Time_to_Event[1])
  mean_temp_record[1] <- mean_temp
  for (i in 2:nrow(estimate_descend)) {
    mean_temp <- mean_temp + estimate_descend$Survival_Estimate[i]*
      (estimate_descend$Time_to_Event[i-1]-estimate_descend$Time_to_Event[i])
    mean_temp_record[i] <- mean_temp
  }
  mean_temp <- mean_temp + 1*(estimate_descend$Time_to_Event[nrow(estimate_descend)]-0)
  mean_temp_record[nrow(estimate_descend)+1] <- mean_temp

  var_temp <- 0

  for (i in 1:nrow(estimate_descend)) {
    var_temp <- var_temp + estimate_descend$Total_Events[i] * mean_temp_record[nrow(estimate_descend)+1-1]^
2 / (estimate_descend$At_Risk[i] * (estimate_descend$At_Risk[i] - estimate_descend$Total_Events[i]))
  }
  return(list(mean = mean_temp, se = sqrt(var_temp)))
}
```

```
mean_var_low_risk <- getMean_Var(low_risk_estimate, max_time_low_risk)
lb_low_risk <- mean_var_low_risk$mean - 1.96 * mean_var_low_risk$se
ub_low_risk <- mean_var_low_risk$mean + 1.96 * mean_var_low_risk$se

mean_var_high_risk <- getMean_Var(high_risk_estimate, max_time_high_risk)
lb_high_risk <- mean_var_high_risk$mean - 1.96 * mean_var_high_risk$se
ub_high_risk <- mean_var_high_risk$mean + 1.96 * mean_var_high_risk$se

print(paste("Low Risk Group: Mean = ", round(mean_var_low_risk$mean, 4), ", 95% CI = (", round(lb_low_risk,
4), ", ", round(ub_low_risk, 4), ")"))
```

```
## [1] "Low Risk Group: Mean =  1588.2282 , 95% CI = ( 1197.3873 ,  1979.0691 )"
```

```
print(paste("High Risk Group: Mean = ", round(mean_var_high_risk$mean, 4), ", 95% CI = (", round(lb_high_ri
sk, 4), ", ", round(ub_high_risk, 4), ")"))
```

```
## [1] "High Risk Group: Mean =  792.2291 , 95% CI = ( 386.304 ,  1198.1542 )"
```

(d) Answer:

For the AML low risk group, the median is 2204 days.

For the AML high risk group, the median is 183 days

```
# linear
get_CI <- function(estimate, p) {

  S <- estimate$Survival_Estimate
  SE <- estimate$Survival_SE


  linear_CI <- (S-(1-p))/(SE)
  log_CI <- (log(-log(S))-log(-log(1-p)))*S*log(S)/(SE)
  arcsin_CI <- (2*asin(sqrt(S))-(2*asin(sqrt(1-p))))*sqrt((S*(1-S)))/(SE)
  estimate$linear_CI <- linear_CI
  estimate$log_CI <- log_CI
  estimate$arcsin_CI <- arcsin_CI

  return(estimate)
}
```

```
get_CI(high_risk_estimate, 0.5)
```

Time_to_Event <int>	Total_Events <int>	At_Risk <dbl>	Survival_Estimate <dbl>	Survival_SE <dbl>	Cumulative_Hazard <dbl>	Hazard_SE <dbl>
2	1	45	0.9778	0.0220	0.0222	0.0222
16	1	44	0.9556	0.0307	0.0449	0.0318
32	1	43	0.9333	0.0372	0.0682	0.0394
47	2	42	0.8889	0.0468	0.1158	0.0518
48	1	40	0.8667	0.0507	0.1408	0.0575
63	1	39	0.8444	0.0540	0.1665	0.0630
64	1	38	0.8222	0.0570	0.1928	0.0683
74	1	37	0.8000	0.0596	0.2198	0.0734
76	1	36	0.7778	0.0620	0.2476	0.0785
80	1	35	0.7556	0.0641	0.2762	0.0835

1-10 of 33 rows | 1-7 of 10 columns Previous **1** 2 3 4 Next

```
get_CI(low_risk_estimate, 0.5)
```

Time_to_Event <int>	Total_Events <int>	At_Risk <dbl>	Survival_Estimate <dbl>	Survival_SE <dbl>	Cumulative_Hazard <dbl>	Hazard_SE <dbl>
10	1	54	0.9815	0.0183	0.0185	0.0185
35	1	53	0.9630	0.0257	0.0374	0.0264
48	1	52	0.9444	0.0312	0.0566	0.0327
53	1	51	0.9259	0.0356	0.0762	0.0381
79	1	50	0.9074	0.0394	0.0962	0.0430
80	1	49	0.8889	0.0428	0.1166	0.0476
105	1	48	0.8704	0.0457	0.1375	0.0520
211	1	47	0.8519	0.0483	0.1587	0.0562
219	1	46	0.8333	0.0507	0.1805	0.0602

Time_to_Event <int>	Total_Events <int>	At_Risk <dbl>	Survival_Estimate <dbl>	Survival_SE <dbl>	Cumulative_Hazard <dbl>	Hazard_SE <dbl>
248	1	45	0.8148	0.0529	0.2027	0.0642

1-10 of 25 rows | 1-7 of 10 columns Previous **1** 2 3 Next

By observing the table,

For the low risk group, the lower bound of linear CI is 704 days. The lower bound of log CI and arcsin CI are both 641 days. The upper bound is undermined.

For the high risk group, the linear CI and arcsin CI are both (115, 363). The log CI is (113, 363)

(e) Answer:

```
my.surv <- survfit(Surv(bmt$t2[bmt$group==2], bmt$d3[bmt$group==2])~1,
                   conf.type = 'log')

summary_low_log <- summary(my.surv, times = 300)

my.surv <- survfit(Surv(bmt$t2[bmt$group==2], bmt$d3[bmt$group==2])~1,
                   conf.type = 'arcsin')

summary_low_asin <- summary(my.surv, times = 300)

my.surv <- survfit(Surv(bmt$t2[bmt$group==3], bmt$d3[bmt$group==3])~1,
                   conf.type = 'log')

summary_high_log <- summary(my.surv, times = 300)

my.surv <- survfit(Surv(bmt$t2[bmt$group==3], bmt$d3[bmt$group==3])~1,
                   conf.type = 'arcsin')

summary_high_asin <- summary(my.surv, times = 300)

low_log_ci <- c(summary_low_log$lower[1], summary_low_log$upper[1])
low_asin_ci <- c(summary_low_asin$lower[1], summary_low_asin$upper[1])
high_log_ci <- c(summary_high_log$lower[1], summary_high_log$upper[1])
high_asin_ci <- c(summary_high_asin$lower[1], summary_high_asin$upper[1])

cat("95% Confidence Intervals for Survival at 300 Days:\n")
```

```
## 95% Confidence Intervals for Survival at 300 Days:
```

```
cat("\nAML Low-Risk Group:")
```

```
##
## AML Low-Risk Group:
```

```
cat("\n   Log CI:", round(low_log_ci[1], 4), "-", round(low_log_ci[2], 4))
```

```
##
##    Log CI: 0.6744 - 0.897
```

```
cat("\n   Arcsine CI:", round(low_asin_ci[1], 4), "-", round(low_asin_ci[2], 4))
```

```
##
##    Arcsine CI: 0.6584 - 0.8775
```

```
cat("\n\nAML High-Risk Group:")
```

```
##
##
## AML High-Risk Group:
```

```
cat("\n   Log CI:", round(high_log_ci[1], 4), "-", round(high_log_ci[2], 4))
```

```
##
##    Log CI: 0.3 - 0.5943
```

```
cat("\n   Arcsine CI:", round(high_asin_ci[1], 4), "-", round(high_asin_ci[2], 4), "\n")
```

```
##
##    Arcsine CI: 0.2833 - 0.5678
```

(f) Answer:

```
data_directory <- "../functions/"
```

```
library(km.ci)
```

```
## Warning: package 'km.ci' was built under R version 4.3.3
```

```
source(paste0(data_directory, "conf_bands.R"))
```

```
my.surv <- survfit(Surv(bmt$t2[bmt$group==2], bmt$d3[bmt$group==2])~1,
                   conf.type = 'none')
cbands.region(my.surv, 100, 400)
```

```
## Find critical regions in Klein and Moeschberger 2nd ed. (Appendix C.3a - C.4c)
```

```
## $aL
## [1] 0.1
##
## $aU
## [1] 0.3
```

```
#linear confidence band for the low risk group
ep.linear.band <- cbands.interval(my.surv, tL = 100, tU = 400,
                                  crit.val = 2.3874,
                                  type = "linear", method = 'ep')
```

```
## Returning linear-type confidence bands using ep method.
```

```
#log confidence band for the low risk group
ep.log.band <- cbands.interval(my.surv, tL = 100, tU = 400,
                                  crit.val = 2.3874,
                                  type = "log", method = 'ep')
```

```
## Returning log-type confidence bands using ep method.
```

```
#arcsin confidence band for the low risk group
ep.asin.band <- cbands.interval(my.surv, tL = 100, tU = 400,
                                crit.val = 2.3874,
                                type = "asin", method = 'ep')
```

```
## Returning asin-type confidence bands using ep method.
```

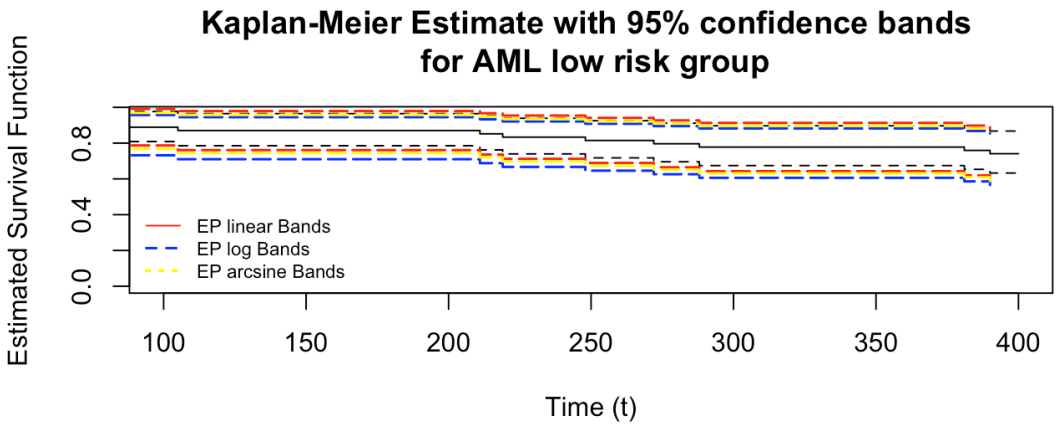
```
plot(my.surv, xlim = c(100, 400), main="Kaplan-Meier Estimate with 95% confidence bands \n for AML low risk
group",
     xlab="Time (t)", ylab="Estimated Survival Function")

lines(ep.linear.band$UL ~ ep.linear.band$t,
      lty = 5, col = "red", lwd = 1.5, type = 's')
lines(ep.linear.band$LL ~ ep.linear.band$t,
      lty = 5, col = "red", lwd = 1.5, type = 's')

lines(ep.log.band$UL ~ ep.log.band$t,
      lty = 5, col = "blue", lwd = 1.5, type = 's')
lines(ep.log.band$LL ~ ep.log.band$t,
      lty = 5, col = "blue", lwd = 1.5, type = 's')

lines(ep.asin.band$UL ~ ep.asin.band$t,
      lty = 5, col = "yellow", lwd = 1.5, type = 's')
lines(ep.asin.band$LL ~ ep.asin.band$t,
      lty = 5, col = "yellow", lwd = 1.5, type = 's')


legend("bottomleft", legend=
         c("EP linear Bands", "EP log Bands", "EP arcsine Bands"), lty=c(1, 2, 3),
       bty = "n", lwd = c(1, 1.5, 2), cex = .7,
       col = c("red", "blue", "yellow"))
```




```
my.surv <- survfit(Surv(bmt$t2[bmt$group==3], bmt$d3[bmt$group==3])~1,
                   conf.type = 'none')
cbands.region(my.surv, 100, 400)
```

```
## Find critical regions in Klein and Moeschberger 2nd ed. (Appendix C.3a - C.4c)
```

```
## $aL
## [1] 0.3
##
## $aU
## [1] 0.6
```

```
#linear confidence band for the high risk group
ep.linear.band <- cbands.interval(my.surv, tL = 100, tU = 400,
                                  crit.val = 2.3668,
                                  type = "linear", method = 'ep')
```

```
## Returning linear-type confidence bands using ep method.
```

```
#log confidence band for the high risk group
ep.log.band <- cbands.interval(my.surv, tL = 100, tU = 400,
                                  crit.val = 2.3668,
                                  type = "log", method = 'ep')
```

```
## Returning log-type confidence bands using ep method.
```

```
#arcsin confidence band for the high risk group
ep.asin.band <- cbands.interval(my.surv, tL = 100, tU = 400,
                                  crit.val = 2.3668,
                                  type = "asin", method = 'ep')
```

```
## Returning asin-type confidence bands using ep method.
```

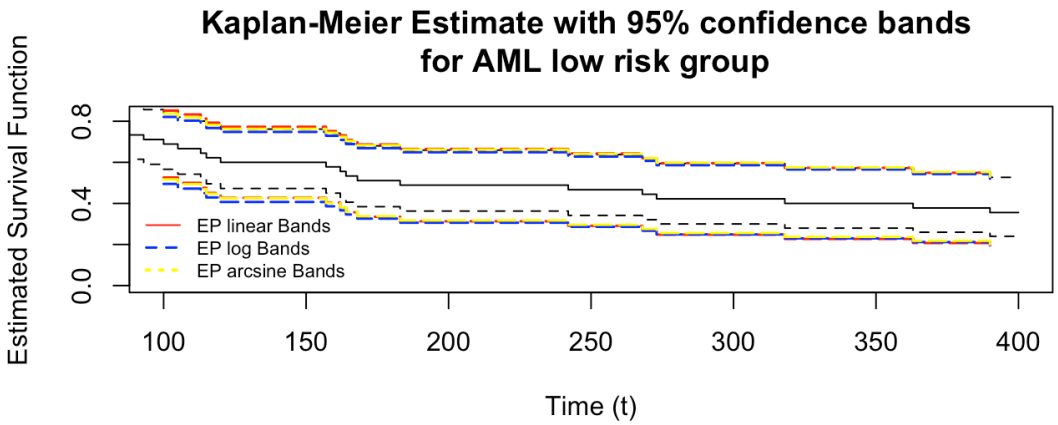
```
plot(my.surv, xlim = c(100, 400), main="Kaplan-Meier Estimate with 95% confidence bands \n for AML low risk
group",
     xlab="Time (t)", ylab="Estimated Survival Function")

lines(ep.linear.band$UL ~ ep.linear.band$t,
      lty = 5, col = "red", lwd = 1.5, type = 's')
lines(ep.linear.band$LL ~ ep.linear.band$t,
      lty = 5, col = "red", lwd = 1.5, type = 's')

lines(ep.log.band$UL ~ ep.log.band$t,
      lty = 5, col = "blue", lwd = 1.5, type = 's')
lines(ep.log.band$LL ~ ep.log.band$t,
      lty = 5, col = "blue", lwd = 1.5, type = 's')

lines(ep.asin.band$UL ~ ep.asin.band$t,
      lty = 5, col = "yellow", lwd = 1.5, type = 's')
lines(ep.asin.band$LL ~ ep.asin.band$t,
      lty = 5, col = "yellow", lwd = 1.5, type = 's')


legend("bottomleft", legend=
         c("EP linear Bands", "EP log Bands", "EP arcsine Bands"), lty=c(1, 2, 3),
       bty = "n", lwd = c(1, 1.5, 2), cex = .7,
       col = c("red", "blue", "yellow"))
```



(g) Answer:

```
my.surv <- survfit(Surv(bmt$t2[bmt$group==2], bmt$d3[bmt$group==2])~1,
                   conf.type = 'none')
cbands.region(my.surv, 100, 400)
```

```
## Find critical regions in Klein and Moeschberger 2nd ed. (Appendix C.3a - C.4c)
```

```
## $aL
## [1] 0.1
##
## $aU
## [1] 0.3
```

```
#linear confidence band for the low risk group
hw.linear.band <- cbands.interval(my.surv, tL = 100, tU = 400,
                                  crit.val = 2.3874,
                                  type = "linear", method = 'hw')
```

```
## Returning linear-type confidence bands using hw method.
```

```
#log confidence band for the low risk group
hw.log.band <- cbands.interval(my.surv, tL = 100, tU = 400,
                                  crit.val = 2.3874,
                                  type = "log", method = 'hw')
```

```
## Returning log-type confidence bands using hw method.
```

```
#arcsin confidence band for the low risk group
hw.asin.band <- cbands.interval(my.surv, tL = 100, tU = 400,
                                  crit.val = 2.3874,
                                  type = "asin", method = 'hw')
```

```
## Returning asin-type confidence bands using hw method.
```

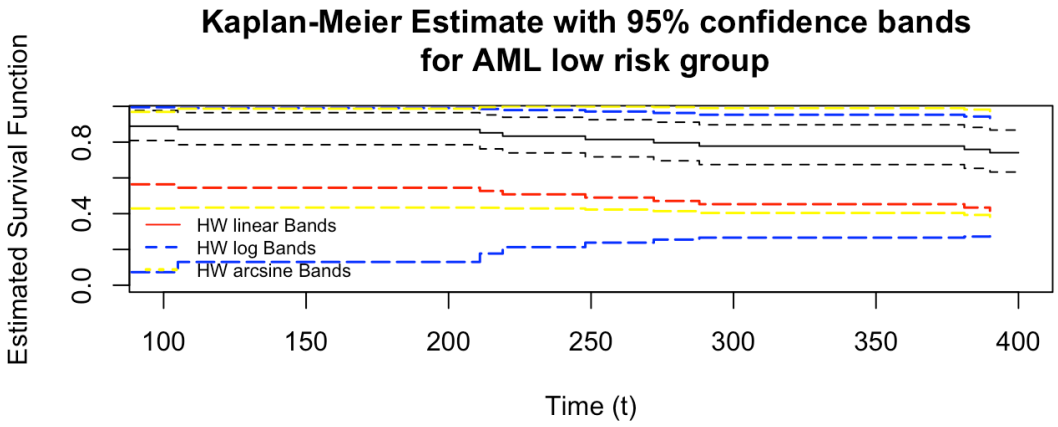
```
plot(my.surv, xlim = c(100, 400), main="Kaplan-Meier Estimate with 95% confidence bands \n for AML low risk
group",
     xlab="Time (t)", ylab="Estimated Survival Function")

lines(hw.linear.band$UL ~ hw.linear.band$t,
      lty = 5, col = "red", lwd = 1.5, type = 's')
lines(hw.linear.band$LL ~ hw.linear.band$t,
      lty = 5, col = "red", lwd = 1.5, type = 's')

lines(hw.log.band$UL ~ hw.log.band$t,
      lty = 5, col = "blue", lwd = 1.5, type = 's')
lines(hw.log.band$LL ~ hw.log.band$t,
      lty = 5, col = "blue", lwd = 1.5, type = 's')

lines(hw.asin.band$UL ~ hw.asin.band$t,
      lty = 5, col = "yellow", lwd = 1.5, type = 's')
lines(hw.asin.band$LL ~ hw.asin.band$t,
      lty = 5, col = "yellow", lwd = 1.5, type = 's')


legend("bottomleft", legend=
         c("HW linear Bands", "HW log Bands", "HW arcsine Bands"), lty=c(1, 2, 3),
       bty = "n", lwd = c(1, 1.5, 2), cex = .7,
       col = c("red", "blue", "yellow"))
```




```
my.surv <- survfit(Surv(bmt$t2[bmt$group==3], bmt$d3[bmt$group==3])~1,
                   conf.type = 'none')
cbands.region(my.surv, 100, 400)
```

```
## Find critical regions in Klein and Moeschberger 2nd ed. (Appendix C.3a - C.4c)
```

```
## $aL
## [1] 0.3
##
## $aU
## [1] 0.6
```

```
#linear confidence band for the high risk group
hw.linear.band <- cbands.interval(my.surv, tL = 100, tU = 400,
                                  crit.val = 2.3668,
                                  type = "linear", method = 'hw')
```

```
## Returning linear-type confidence bands using hw method.
```

```
#log confidence band for the high risk group
hw.log.band <- cbands.interval(my.surv, tL = 100, tU = 400,
                               crit.val = 2.3668,
                               type = "log", method = 'hw')
```

```
## Returning log-type confidence bands using hw method.
```

```
#arcsin confidence band for the high risk group
hw.asin.band <- cbands.interval(my.surv, tL = 100, tU = 400,
                               crit.val = 2.3668,
                               type = "asin", method = 'hw')
```

```
## Returning asin-type confidence bands using hw method.
```

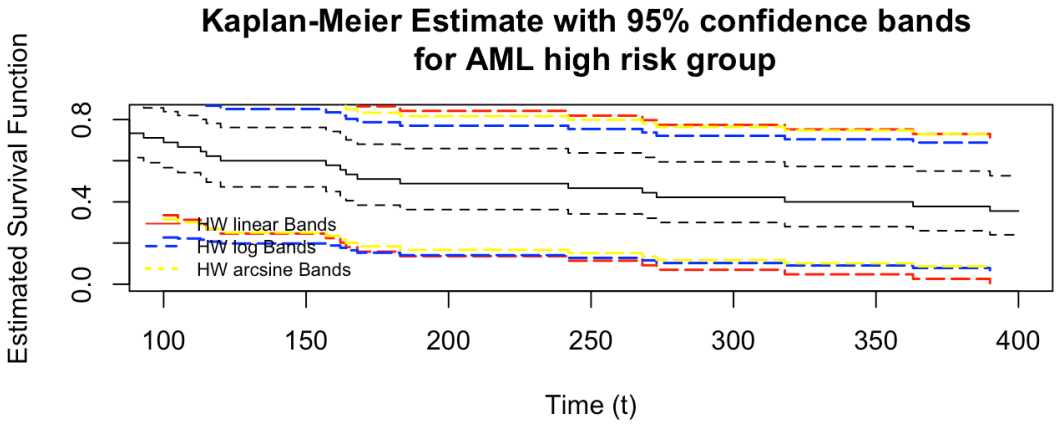
```
plot(my.surv, xlim = c(100, 400), main="Kaplan-Meier Estimate with 95% confidence bands \n for AML high ris
k group",
     xlab="Time (t)", ylab="Estimated Survival Function")

lines(hw.linear.band$UL ~ hw.linear.band$t,
      lty = 5, col = "red", lwd = 1.5, type = 's')
lines(hw.linear.band$LL ~ hw.linear.band$t,
      lty = 5, col = "red", lwd = 1.5, type = 's')

lines(hw.log.band$UL ~ hw.log.band$t,
      lty = 5, col = "blue", lwd = 1.5, type = 's')
lines(hw.log.band$LL ~ hw.log.band$t,
      lty = 5, col = "blue", lwd = 1.5, type = 's')

lines(hw.asin.band$UL ~ hw.asin.band$t,
      lty = 5, col = "yellow", lwd = 1.5, type = 's')
lines(hw.asin.band$LL ~ hw.asin.band$t,
      lty = 5, col = "yellow", lwd = 1.5, type = 's')


legend("bottomleft", legend=
         c("HW linear Bands", "HW log Bands", "HW arcsine Bands"), lty=c(1, 2, 3),
       bty = "n", lwd = c(1, 1.5, 2), cex = .7,
       col = c("red", "blue", "yellow"))
```



(h) Answer:

High risk group has much less mean and median survival days than low risk and all group. The survival function of high risk group is decreasing as time increases. The survival function of low risk group is more stable, and shows sign of going constant as time increases.

Exercise 12.1

In section 1.11, a study of the effects of ploidy on survival for patients with cancer of the tongue was described. In the study patients were classified as having either an aneuploid or diploid DNA profile. The data is presented in Table 1.6.

Table 1.6: Death Times (in Weeks) of Patients with Cancer of the Tongue

Aneuploid Tumors

- **Death Times:**
  1, 3, 3, 4, 10, 13, 13, 16, 16, 24, 26, 27, 28, 30, 30, 32, 41, 51, 65, 67, 70, 72, 73, 77, 91, 93, 96, 100, 104, 157, 167
- **Censored Observations:**
  61, 74, 79, 80, 81, 87, 87, 88, 89, 93, 97, 101, 104, 108, 109, 120, 131, 150, 231, 240, 400

Diploid Tumors

- **Death Times:**
  1, 3, 4, 5, 5, 8, 12, 13, 18, 23, 26, 27, 30, 42, 56, 62, 69, 104, 104, 112, 129, 181
- **Censored Observations:**
  8, 67, 76, 104, 176, 231

---

a. For both the aneuploid and diploid groups fit a Weibull model to the data. Find the maximum likelihood estimates of $lambda$ and α, and their standard errors.
b. For both groups, test the hypothesis that the shape parameter, α, is equal to 1 by both the Wald and likelihood ratio tests.
c. Find the maximum likelihood estimates of the median survival for both groups. Use the delta method to find an estimate of the stan- dard error of your estimates.
d. Fit a Weibull regression model to this data with a single covariate, Z, that is equal to 1 if the patient had an aneuploid DNA profile and 0 otherwise. Test the hypothesis of no effect of ploidy on survival using the likelihood ratio test and the Wald test. Find a point estimate and 95% confidence interval for the relative risk of death for an aneuploid tumor as compared to a diploid tumor. Also find a point estimate and a 95% confidence for the acceleration factor. Provide an interpretation of this factor.

(a) Answer:

```
library(survival)
library(flexsurv)
```

```
## Warning: package 'flexsurv' was built under R version 4.3.3
```

```
aneuploid_deaths <- c(1, 3, 3, 4, 10, 13, 13, 16, 16, 24, 26, 27, 28, 30, 30, 32,
                      41, 51, 65, 67, 70, 72, 73, 77, 91, 93, 96, 100, 104, 157, 167)
aneuploid_censored <- c(61, 74, 79, 80, 81, 87, 87, 88, 89, 93, 97, 101, 104, 108,
                        109, 120, 131, 150, 231, 240, 400)
aneuploid_status <- c(rep(1, length(aneuploid_deaths)), rep(0, length(aneuploid_censored)))
aneuploid_times <- c(aneuploid_deaths, aneuploid_censored)

diploid_deaths <- c(1, 3, 4, 5, 5, 8, 12, 13, 18, 23, 26, 27, 30, 42, 56, 62, 69,
                    104, 104, 112, 129, 181)
diploid_censored <- c(8, 67, 76, 104, 176, 231)
diploid_status <- c(rep(1, length(diploid_deaths)), rep(0, length(diploid_censored)))
diploid_times <- c(diploid_deaths, diploid_censored)

ploidy <- factor(c(rep("Aneuploid", length(aneuploid_times)), rep("Diploid", length(diploid_times))),
                 levels = c("Diploid", "Aneuploid"))  # Reference = Diploid
survival_times <- c(aneuploid_times, diploid_times)
status <- c(aneuploid_status, diploid_status)

surv_data <- data.frame(time = survival_times, status = status, ploidy = ploidy)
```

```
weibull_aneuploid <- survreg(Surv(time, status) ~ 1, data = surv_data, subset = (ploidy == "Aneuploid"), di
st = "weibull")
weibull_diploid <- survreg(Surv(time, status) ~ 1, data = surv_data, subset = (ploidy == "Diploid"), dist =
"weibull")
```

```
lambdahat.wei.aneuploid <- exp(-weibull_aneuploid$coefficients/weibull_aneuploid$scale)
alphahat.wei.aneuploid <- 1/weibull_aneuploid$scale
```

```
lambdahat.wei.diploid <- exp(-weibull_diploid$coefficients/weibull_diploid$scale)
alphahat.wei.diploid <- 1/weibull_diploid$scale
```

```
print(paste("Aneuploid: lambda_hat = ", lambdahat.wei.aneuploid, ", alpha_hat = ", alphahat.wei.aneuploid))
```

```
## [1] "Aneuploid: lambda_hat =  0.0161159376493197 , alpha_hat =  0.832184417623562"
```

```
print(paste("Diploid: lambda_hat = ", lambdahat.wei.diploid, ", alpha_hat = ", alphahat.wei.diploid))
```

```
## [1] "Diploid: lambda_hat =  0.0358623757152242 , alpha_hat =  0.774505796953166"
```

(b) Answer:

```
# Extract scale parameter and its standard error
log_scale_aneuploid <- summary(weibull_aneuploid)$table["Log(scale)", "Value"]
se_log_scale_aneuploid <- summary(weibull_aneuploid)$table["Log(scale)", "Std. Error"]

log_scale_diploid <- summary(weibull_diploid)$table["Log(scale)", "Value"]
se_log_scale_diploid <- summary(weibull_diploid)$table["Log(scale)", "Std. Error"]

# Convert to shape parameter (alpha) since alpha = 1 / scale
alpha_hat_aneuploid <- exp(-log_scale_aneuploid)
alpha_hat_diploid <- exp(-log_scale_diploid)

# Compute SE of alpha using the Delta Method
se_alpha_aneuploid <- exp(-log_scale_aneuploid) * se_log_scale_aneuploid
se_alpha_diploid <- exp(-log_scale_diploid) * se_log_scale_diploid

# Wald Test: Z = (alpha - 1) / SE(alpha)
wald_stat_aneuploid <- (alpha_hat_aneuploid - 1) / se_alpha_aneuploid
wald_stat_diploid <- (alpha_hat_diploid - 1) / se_alpha_diploid

wald_p_aneuploid <- 2 * (1 - pnorm(abs(wald_stat_aneuploid)))
wald_p_diploid <- 2 * (1 - pnorm(abs(wald_stat_diploid)))

# Likelihood Ratio Test
weibull_exp_aneuploid <- survreg(Surv(time, status) ~ 1, data = surv_data, subset = (ploidy == "Aneuploi
d"), dist = "exponential")
weibull_exp_diploid <- survreg(Surv(time, status) ~ 1, data = surv_data, subset = (ploidy == "Diploid"), di
st = "exponential")

lrt_stat_aneuploid <- 2 * (logLik(weibull_aneuploid) - logLik(weibull_exp_aneuploid))
lrt_stat_diploid <- 2 * (logLik(weibull_diploid) - logLik(weibull_exp_diploid))

# Compute LRT p-values
lrt_p_aneuploid <- pchisq(lrt_stat_aneuploid, df = 1, lower.tail = FALSE)
lrt_p_diploid <- pchisq(lrt_stat_diploid, df = 1, lower.tail = FALSE)

print(paste("Aneuploid: Wald p-value =", round(wald_p_aneuploid, 4), ", LRT p-value =", round(lrt_p_aneuplo
id, 4)))
```

```
## [1] "Aneuploid: Wald p-value = 0.1896 , LRT p-value = 0.2116"
```

```
print(paste("Diploid: Wald p-value =", round(wald_p_diploid, 4), ", LRT p-value =", round(lrt_p_diploid,
4)))
```

```
## [1] "Diploid: Wald p-value = 0.0967 , LRT p-value = 0.1228"
```

Since the p values are all greater then 0.05, we fail to reject the null hypothesis.

(c) Answer:

```
# Function to compute median survival and SE using Delta Method
median_survival <- function(weibull_model) {
  # Extract lambda and alpha
  lambda_hat <- exp(-weibull_model$coefficients / weibull_model$scale)
  alpha_hat <- 1 / weibull_model$scale

  # Compute median survival
  median_t0.5 <- (log(2) / lambda_hat)^(1 / alpha_hat)

  # Extract standard errors
  se_log_scale <- summary(weibull_model)$table["Log(scale)", "Std. Error"]  # SE of log(scale)
  se_lambda <- lambda_hat * se_log_scale  # SE of lambda
  se_alpha <- alpha_hat * se_log_scale    # SE of alpha

  # Compute partial derivatives
  d_med_lambda <- -median_t0.5 / (lambda_hat * alpha_hat)
  d_med_alpha <- -median_t0.5 * log(log(2) / lambda_hat) / (alpha_hat^2)

  # Compute standard error using the Delta Method
  se_median <- sqrt((d_med_lambda^2 * se_lambda^2) + (d_med_alpha^2 * se_alpha^2))

  return(list(median = median_t0.5, se = se_median))
}

# Compute median survival and SE for each group
median_aneuploid <- median_survival(weibull_aneuploid)
median_diploid <- median_survival(weibull_diploid)

print(paste("Aneuploid: Median Survival =", round(median_aneuploid$median, 2), "weeks, SE =", round(median_
aneuploid$se, 2)))
```

```
## [1] "Aneuploid: Median Survival = 91.83 weeks, SE = 66.02"
```

```
print(paste("Diploid: Median Survival =", round(median_diploid$median, 2), "weeks, SE =", round(median_dipl
oid$se, 2)))
```

```
## [1] "Diploid: Median Survival = 45.78 weeks, SE = 32.38"
```

(d) Answer:

```
# Fit Weibull regression with ploidy as a covariate
weibull_model <- survreg(Surv(time, status) ~ ploidy, data = surv_data, dist = "weibull")

# Extract coefficients
beta_hat <- coef(weibull_model)["ploidyAneuploid"]
se_beta <- summary(weibull_model)$table["ploidyAneuploid", "Std. Error"]

# Compute Hazard Ratio and 95% CI
HR <- exp(beta_hat)
HR_CI_lower <- exp(beta_hat - 1.96 * se_beta)
HR_CI_upper <- exp(beta_hat + 1.96 * se_beta)

# Compute AF and 95% CI
alpha_hat <- 1 / weibull_model$scale
AF <- exp(-beta_hat / alpha_hat)
AF_CI_lower <- exp(-(beta_hat + 1.96 * se_beta) / alpha_hat)
AF_CI_upper <- exp(-(beta_hat - 1.96 * se_beta) / alpha_hat)

# Wald Test
wald_stat <- beta_hat / se_beta
wald_p_value <- 2 * (1 - pnorm(abs(wald_stat)))

# Likelihood Ratio Test
weibull_null <- survreg(Surv(time, status) ~ 1, data = surv_data, dist = "weibull")
lrt_stat <- 2 * (logLik(weibull_model) - logLik(weibull_null))
lrt_p_value <- pchisq(lrt_stat, df = 1, lower.tail = FALSE)

print(paste("Hazard Ratio (Aneuploid vs. Diploid):", round(HR, 2), "95% CI:", round(HR_CI_lower, 2), "-", r
ound(HR_CI_upper, 2)))
```

```
## [1] "Hazard Ratio (Aneuploid vs. Diploid): 1.95 95% CI: 0.98 - 3.88"
```

```
print(paste("Acceleration Factor (Aneuploid vs. Diploid):", round(AF, 2), "95% CI:", round(AF_CI_lower, 2),
"-", round(AF_CI_upper, 2)))
```

```
## [1] "Acceleration Factor (Aneuploid vs. Diploid): 0.44 95% CI: 0.19 - 1.02"
```

```
print(paste("Wald Test p-value:", round(wald_p_value, 4)))
```

```
## [1] "Wald Test p-value: 0.0566"
```

```
print(paste("Likelihood Ratio Test p-value:", round(lrt_p_value, 4)))
```

```
## [1] "Likelihood Ratio Test p-value: 0.0585"
```

Since AF is 0.44 < 1, survival time is shortened (event occurs faster). The survival time for patients with aneuploid tumors is 44% of the survival time of diploid tumors.