

レポート提出票

科目名: 応用情報工学演習

実験課題名: 寒水研究室

実施日: 2025 年 10 月 12 日

学籍番号: 4623044

氏名: 呉 子吟

共同実験者:

_____	_____
_____	_____
_____	_____
_____	_____

演習課題

演習 1：観察終了による打ち切りの影響

- 時点 30 で観察を終了した場合の生存関数を推定してください
- その結果をこれまでのプロットに重ねる
- 報告事項
 - － プロット
 - － 最大観測時間をもつ被験者がイベントを経験せず打ち切られた場合の、生存関数の推定における問題点の簡単な考察
- ヒント：時点 30 以降の生存関数のプロットはどうなっているか？

1 結果・考察

付録の Listing1 の R コードを作成して実行し、以下のような結果が得られた。

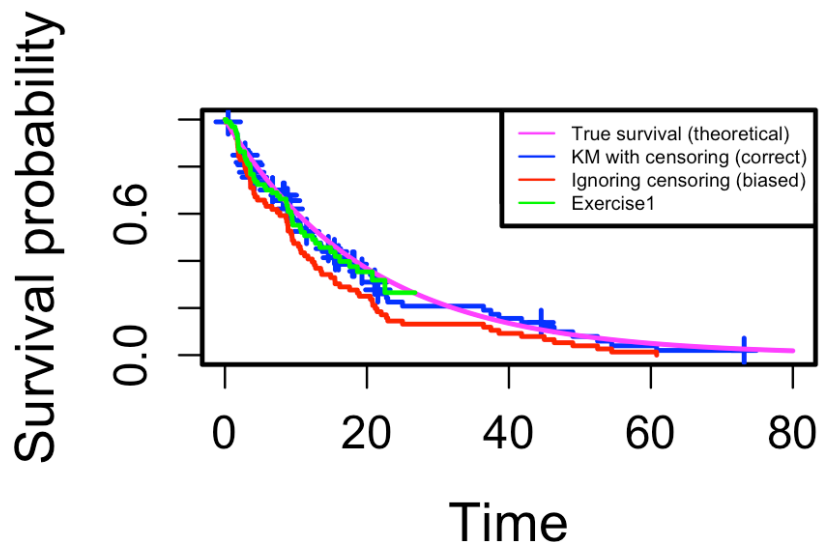


図 1: 演習 1 の結果

観察時点を 30 で終了した場合の生存関数を推定した結果、図 1 の緑線のようなプロットが得られた。図 1 によると、この線は大体青線（打ち切らない正しく KM 推定）と一致しているが、時点 30 以降では切断されており、もし線が続いて描くと、水平方向に伸びていき、生存確率がある程度そのまま変化しなくなってしまう（しかし 30 時点からの生存率という意味ではない）。

この現象は、観察期間の終了によって、それ以降のイベント情報が得られないためである。即ち、30 以降の時点は、全ての被験者が打ち切られたため、新しいイベントが存在しなく、KM 推定法におけるリスク集合の更新が止まった。

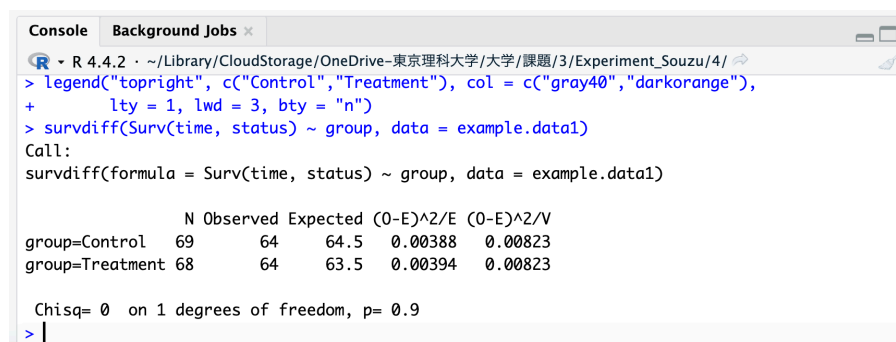
したがって、最大観測時間をもつ被験者がイベントを経験せずに打ち切られた場合、その後の区間では生存確率の推定が行えないため、生存曲線の形状を誤評価するリスクがある。全体的な精度が悪くなり、生存関数の推定が不確実になることが分かる。また、観察終了時点以降の推定値を実際の生存率として解釈できなく、それ以前の観測情報に基づく推定であることのみ示している。

演習課題

- survival パッケージに含まれる、以下の 4 つのデータセットから veteran のデータを選んでログランク検定を実行する
 - veteran
- 報告事項
 - KM プロット
 - ログランク検定を行った結果のスクショ
 - 該当部分の R コード

2 結果・考察

この課題においては、肺がんに対する 2 つの治療のランダム化比較試験のデータ veteran を用いて、付録の Listin2 の R コードを作成して実行した。以下の図 2 と図 3 ような結果が得られた。



```
R - R 4.4.2 - ~/Library/CloudStorage/OneDrive-東京理科大学/大学/課題/3/Experiment_Souzu/4/
> legend("topright", c("Control", "Treatment"), col = c("gray40", "darkorange"),
+       lty = 1, lwd = 3, bty = "n")
> survdiff(Surv(time, status) ~ group, data = example.data1)
Call:
survdiff(formula = Surv(time, status) ~ group, data = example.data1)

              N Observed Expected (O-E)^2/E (O-E)^2/V
group=Control  69         64    64.5   0.00388   0.00823
group=Treatment 68         64    63.5   0.00394   0.00823

      Chisq= 0  on 1 degrees of freedom, p= 0.9
> |
```

図 2: 演習 2 の出力結果

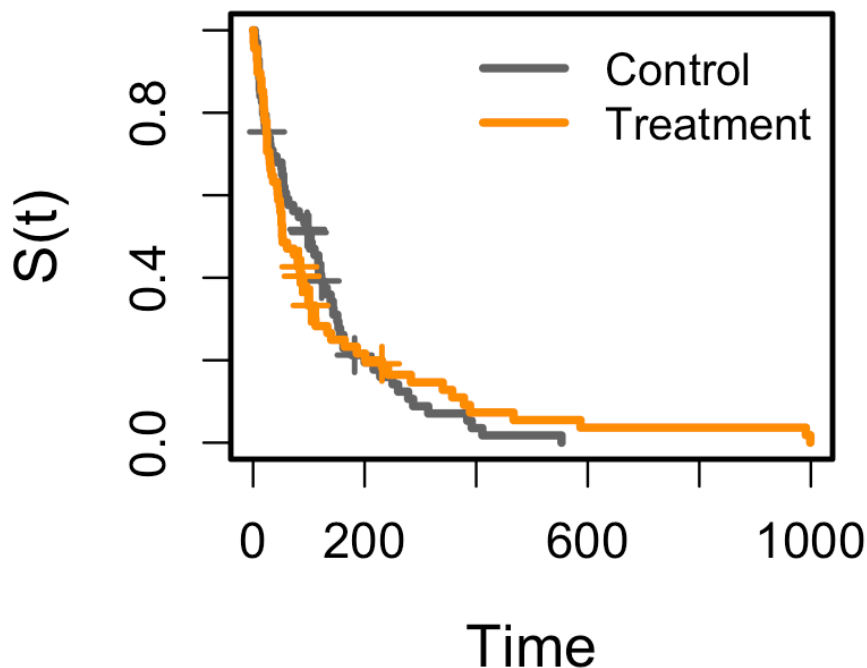


図 3: 演習 2 のプロット

137名の被験者のデータから変数time (生存時間), status (イベント発生指標), および trt (治療群) のみ抽出し, Log-rank 検定の解析対象として使われた. また, trt は因子化され, 1 を「Control」, 2 を「Treatment」とした.

まず, 図 2 の Log-rank 検定の結果から, カイ二乗統計量は 0, 自由度 1, p 値は 0.9 である. ここで, p 値が 0.05 より大きく上回っていることから, 有意水準 $\alpha = 0.05$ のもとで, 帰無仮説「両群の生存関数は等しい」を棄却できない. したがって, Control 群と Treatment 群の間に統計的に生存率の差は認められなかった.

Kaplan-Meier 法により両群の生存関数を推定し, 結果を図 3 に示す. 図中の灰色の線は Control 群, 線は Treatment 群を表す. また, それぞれの階段状の変化点はイベント発生を表しており, 右端の「+」マークは右打ち切りを示している. 灰色と橙色の両曲線は左端から中央の部分まで, 大体的に重なっており, 生存確率の推移に顕著な差は見られなかった. Treatment 群の一部の生存時間が長く伸ばしていき, 約 Time=1000 までイベントが発生されたことがわかる. この区間まで生存していた被験者数が少なく, 出力結果により, 偶然誤差と見なされ, 統計的に有意とは言えない. さらに, 図 2 の解析結果と図 3 の視覚的傾向が一致していることが確認できた.

以上より, 本データにおける治療法の違いは生存時間に有意な影響を与えなかったと考えられる.

付録

```
1 #-----
2 # Kaplan-Meier 法による生存関数の計算
3 #-----
4
5 # (生存時間
6 survtime <- c(2, 4, 4, 5, 7) # 各被験者の生存時間
7 event <- c(0, 0, 1, 1, 1) # イベント発生発生 (1=, 打ち切り0=)
8 library(survival)
9 km <- survfit(Surv(survtime, event) ~ 1)
10 summary(km)
11
12 par(lwd = 2, cex.lab = 1.4, cex.axis = 1.2)
13 plot(km, xlab = "Month", ylab = "S(t)", mark.t = TRUE,
14      conf.int = FALSE, lwd = 2, cex = 2)
15
16
17 #
18 n <- 100 # サンプルサイズ
19 lambda <- 0.05 # 指数分布のパラメータ
20 set.seed(2025)
21 true_time <- rexp(n, rate = 0.05)
22 true_S <- function(t) {exp(-lambda * t)}
23
24 censor_time <- runif(n, min = 0, max = max(true_time))
25 obs_time <- pmin(true_time, censor_time)
26 status <- as.numeric(true_time <= censor_time) # イベント1=, 打ち切り0=
27 dat_censored <- data.frame(time = obs_time, status = status)
28 dat_naive <- subset(dat_censored, status == 1)
29 fit_cens <- survfit(Surv(time, status) ~ 1, data = dat_censored)
30 fit_naive <- survfit(Surv(time, status) ~ 1, data = dat_naive)
31
32 plot(fit_cens,
33      col = "blue", lwd = 2, # 色, 線の太さ collwd
34      xlab = "Time", ylab = "Survival probability", # 軸ラベル
35      xlim = c(0, 80), ylim = c(0, 1), # プロットする範囲の制限
36      conf.int = FALSE, # 信頼区間を非表示
37      mark.t = TRUE # 打ち切りのマークを表示
38 )
39 lines(fit_naive, col = "red", lwd = 2, conf.int = FALSE)
40 curve(true_S, from = 0, to = 80, col = "magenta", lwd = 2, add = TRUE)
41
42 #practice1
43 set.seed(2025)
44 censor_time1 <- pmin(runif(n, min = 0, max = max(true_time)), 30)
45 obs_time1 <- pmin(true_time, censor_time1)
46 status1 <- as.numeric(true_time <= censor_time1) # イベント1=, 打ち切り0=
47 dat_censored1 <- data.frame(time = obs_time1, status = status1)
48 #dat_naive1 <- subset(dat_censored1, status == 1)
49 fit_cens1 <- survfit(Surv(time, status) ~ 1, data = dat_censored1)
50 #fit_naive1 <- survfit(Surv(time, status) ~ 1, data = dat_naive)
51
52 lines(fit_cens1, col = "green", lwd = 2, conf.int = FALSE)
53 legend("topright",
54      legend = c("True survival (theoretical)",
55                 "KM with censoring (correct)",
56                 "Ignoring censoring (biased)",
```

```

57         "Exercise1"),
58     col = c("magenta", "blue", "red", "green"), lty = c(1,1,1,1), lwd = 1,
        cex = 0.5)

```

Listing 1: 演習1のコード

```

1
2 #2
3 (library(survival))
4 str(veteran)
5 example.data1 <- veteran[, c("time", "status", "trt")]
6 example.data1$group <- factor(example.data1$trt,
7                               labels = c("Control", "Treatment"))
8
9 fit.km <- survfit(Surv(time, status) ~ group, data = example.data1)
10 par(lwd = 2, cex.lab = 1.3, cex.axis = 1.1, mar = c(4.5, 4.5, 2, 1))
11 plot(fit.km, col = c("gray40", "darkorange"), lwd = 3, xlab = "Time", ylab
12       = "S(t)", mark.time = TRUE, cex = 1.2)
13 legend("topright", c("Control", "Treatment"), col = c("gray40", "darkorange"),
14       lty = 1, lwd = 3, bty = "n")
15 survdiff(Surv(time, status) ~ group, data = example.data1)

```

Listing 2: 演習2のコード