

CA1-backup

September 29, 2023

0.1 CA 1:

0.1.1 Task 1

Importing the necessary libraries and writing the data:

```
[12]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sksurv.nonparametric import kaplan_meier_estimator

data = {
    "treatment": ["yes", "yes", "yes", "yes", "yes", # no = placebo
                  "yes", "yes", "yes", "yes", "yes",
                  "yes", "yes", "yes", "yes", "yes",
                  "yes", "yes", "yes", "yes", "yes",
                  "no", "no", "no", "no", "no",
                  "no", "no", "no", "no", "no",
                  "no", "no", "no", "no", "no",
                  "no", "no", "no", "no", "no",
                  "no", "no", "no", "no"],
    "survival time (months)": [1, 5, 6, 6, 9,
                               10, 10, 10, 12, 12,
                               12, 12, 12, 13, 15,
                               16, 20, 24, 24, 27,
                               32, 34, 36, 36, 44,
                               3, 6, 6, 6, 6,
                               8, 8, 12, 12, 12,
                               15, 16, 18, 18, 20,
                               22, 24, 28, 28, 28,
                               30, 30, 33, 42],
    "event": [0, 0, 1, 1, 0, # 1 = event happened, 0 = censored
              1, 1, 0, 1, 1,
              1, 1, 0, 0, 0,
              0, 0, 1, 0, 0,
              1, 0, 0, 0, 0]
```

```

        0, 1, 1, 1, 1,
        1, 1, 1, 1, 0,
        0, 0, 0, 0, 1,
        0, 1, 0, 0, 0,
        1, 0, 0, 1]
}

# Create a DataFrame from the data
df = pd.DataFrame(data)

# Display the DataFrame
df.head()

```

```

[12]:  treatment  survival time (months)  event
      0         yes                    1      0
      1         yes                    5      0
      2         yes                    6      1
      3         yes                    6      1
      4         yes                    9      0

```

```

[13]: # Separate data into treatment groups
treatment_yes = df[df["treatment"] == "yes"]
treatment_no = df[df["treatment"] == "no"]

# The groups with only the non-censored data
treatment_yes_1 = df[(df["treatment"] == "yes") & (df["event"] == 1)]
treatment_no_1 = df[(df["treatment"] == "no") & (df["event"] == 1)]

print(f"There are {len(treatment_yes_1)} failures in the treatment group and
↳ {len(treatment_no_1)} failures in the control group") # Double checking that
↳ the number of rows is correct

# Average survival time
avg_survival_time_treatment = sum(df["survival time (months)"][:25]) / 25
avg_survival_time_control = sum(df["survival time (months)"][25:]) / 24

print(f"The average survival time for the treatment group is
↳ {avg_survival_time_treatment} and the average survival time for the control
↳ group is {avg_survival_time_control}")

# Average hazard rate
avg_hazard_treatment = len(treatment_yes_1) / sum(df["survival time (months)"][:
↳ 25])
avg_hazard_control = len(treatment_no_1) / sum(df["survival time (months)"][25:
↳ ])

```

```

print(f"The average hazard rate for the treatment group is_
↳{avg_hazard_treatment} and the average hazard rate for the control group is_
↳{avg_hazard_control}")

# Hazard ratio
hazard_ratio = avg_hazard_control / avg_hazard_treatment
print(f"The hazard ratio is {hazard_ratio}")

```

There are 10 failures in the treatment group and 12 failures in the control group

The average survival time for the treatment group is 17.52 and the average survival time for the control group is 17.958333333333332

The average hazard rate for the treatment group is 0.0228310502283105 and the average hazard rate for the control group is 0.027842227378190254

The hazard ratio is 1.2194895591647332

One can see that the average survival time higher for the control group, but the difference is not significant. Furthermore, the hazard is higher for the control group, as seen above. The ratio is 1.2, which means that the hazard is 20% higher for the control group. One should take this info with a grain of salt, because there are quite many censored patients in the treatment group.

0.1.2 Task 2

Treatment group: $(n = 25)$ $\left\{ \begin{array}{l} 1+, 5+, 6, 6, 9+, 10, 10, 10+, \\ 12, 12, 12, 12, 12+, 13+, 15+, 16+, 20+, \\ 24, 24+, 27+, 32, 34+, 36+, 36+, 44+ \end{array} \right\}$

$$S^{\wedge}(0) = \left(1 - \frac{0}{25}\right) = \underline{1}$$

$$S^{\wedge}(6) = \left(1 - \frac{2}{23}\right) = \frac{21}{23} = \underline{0,913}$$

$$S^{\wedge}(10) = 0,913 \left(1 - \frac{2}{20}\right) = 0,913 \cdot \frac{18}{20} = \underline{0,822}$$

$$S^{\wedge}(12) = 0,822 \left(1 - \frac{4}{17}\right) = 0,822 \cdot \frac{13}{17} = \underline{0,628}$$

$$S^{\wedge}(24) = 0,628 \left(1 - \frac{1}{8}\right) = 0,628 \cdot \frac{7}{8} = \underline{0,550}$$

$$S^{\wedge}(32) = 0,550 \left(1 - \frac{1}{5}\right) = 0,550 \cdot \frac{4}{5} = \underline{0,440}$$

Control group: $\left\{ 3+, 6, 6, 6, 6, 8, 8, 12, 12, 12+, 15+, 16+, 18+, \right.$
 $(n = 24) \quad \left. 18+, 20, 22+, 24, 28+, 28+, 28+, 30, 30+, 33+, 42 \right\}$

$$s^{\wedge}(0) = \left(1 - \frac{0}{24}\right) = \underline{1}$$

$$s^{\wedge}(6) = \left(1 - \frac{4}{23}\right) = \frac{19}{23} = \underline{0,826}$$

$$s^{\wedge}(8) = 0,826 \cdot \left(1 - \frac{2}{19}\right) = 0,826 \cdot \frac{17}{19} = \underline{0,740}$$

$$s^{\wedge}(12) = 0,740 \left(1 - \frac{2}{17}\right) = 0,740 \cdot \frac{15}{17} = \underline{0,652}$$

$$s^{\wedge}(20) = 0,652 \left(1 - \frac{1}{10}\right) = 0,652 \cdot \frac{9}{10} = \underline{0,587}$$

$$s^{\wedge}(24) = 0,587 \left(1 - \frac{1}{8}\right) = 0,587 \cdot \frac{7}{8} = \underline{0,513}$$

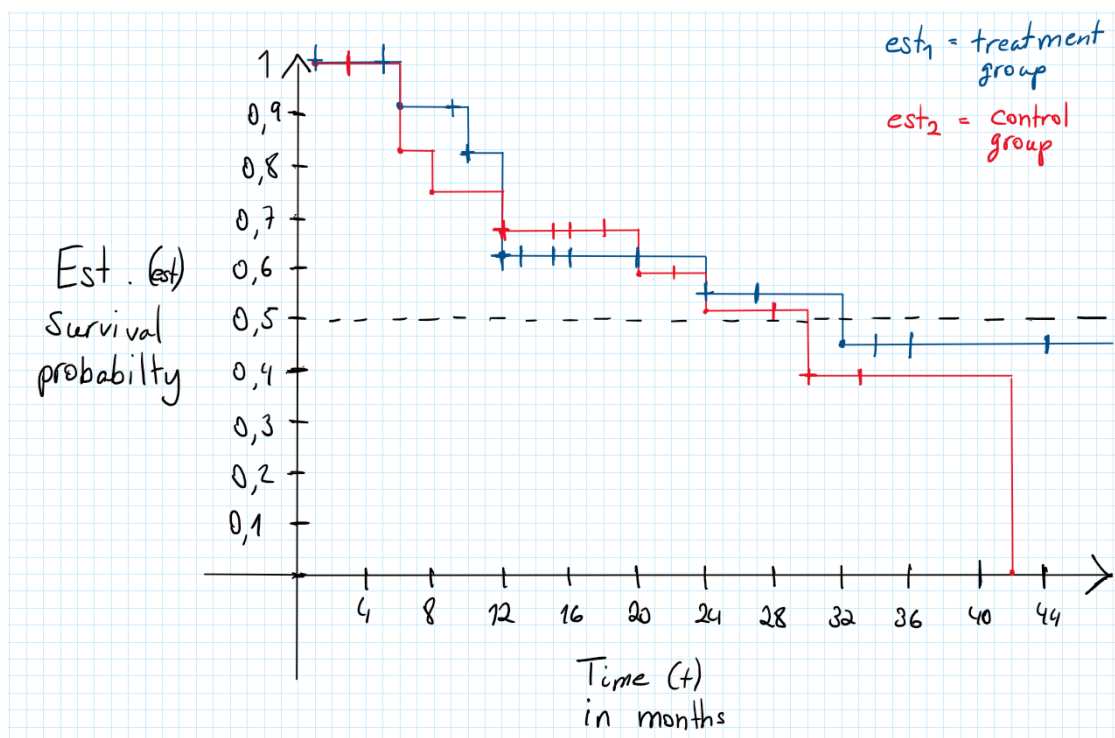
$$s^{\wedge}(30) = 0,513 \left(1 - \frac{1}{4}\right) = 0,513 \cdot \frac{3}{4} = \underline{0,385}$$

$$s^{\wedge}(42) = 0,385 \left(1 - \frac{1}{1}\right) = \underline{0}$$

est_1 is for the treatment group, while est_2 is for the control group.

$t(f)$	$m_1(f)$	$m_2(f)$	$n_1(f)$	$n_2(f)$	est_1	est_2
0	0	0	25	24	1	1
6	2	4	23	23	0,913	0,826
8	0	2	21	19	0,913	0,740
10	2	0	20	17	0,822	0,740
12	4	2	17	17	0,628	0,652
20	0	1	13	10	0,628	0,587
24	1	1	8	8	0,550	0,513
30	0	1	5	4	0,550	0,385
32	1	0	5	2	0,440	0,385
42	0	1	1	1	0,440	0

0.1.3 Task 3



As seen above on the curve, the median survival time for the treatment group is 32, while it is 30 for the control group.

0.1.4 Task 4

Making lists of the data:

```
[33]: # Making lists since they are easier to work with for this instance in sksurv
```

```
treatment = [1, 5, 6, 6, 9,
              10, 10, 10, 12, 12,
              12, 12, 12, 13, 15,
              16, 20, 24, 24, 27,
              32, 34, 36, 36, 44]

control = [3, 6, 6, 6, 6,
           8, 8, 12, 12, 12,
           15, 16, 18, 18, 20,
           22, 24, 28, 28, 28,
           30, 30, 33, 42]

treatment_event = [0, 0, 1, 1, 0, # 1 = event happened, 0 = censored
                   1, 1, 0, 1, 1,
                   1, 1, 0, 0, 0,
                   0, 0, 1, 0, 0,
                   1, 0, 0, 0, 0,]

control_event = [0, 1, 1, 1, 1, # 1 = event happened, 0 = censored
                 1, 1, 1, 1, 0,
                 0, 0, 0, 0, 1,
                 0, 1, 0, 0, 0,
                 1, 0, 0, 1]
```

```
[34]: # Making the event data into boolean (because sksurv requires it)
```

```
treatment_event_bool = np.array(treatment_event, dtype=bool)
control_event_bool = np.array(control_event, dtype=bool)

# Kaplan-Meier estimator

time_treatment, survival_prob_treatment, conf_int = □
↳ kaplan_meier_estimator(treatment_event_bool, treatment, conf_type="log-log")
time_control, survival_prob_control, conf_int = □
↳ kaplan_meier_estimator(control_event_bool, control, conf_type="log-log")

# Plotting the Kaplan-Meier curves

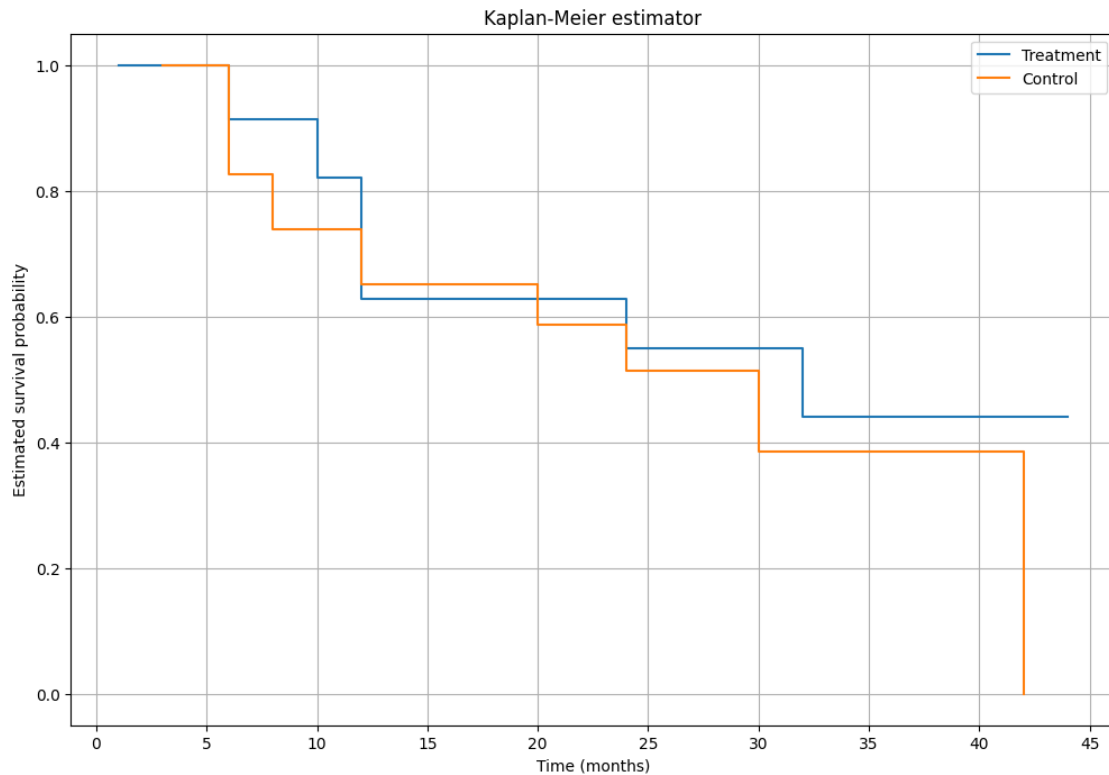
plt.figure(figsize=(12, 8))
```

```

plt.step(time_treatment, survival_prob_treatment, where="post", label="Treatment")
plt.step(time_control, survival_prob_control, where="post", label="Control")
plt.xlabel("Time (months)")
plt.ylabel("Estimated survival probability")
plt.title("Kaplan-Meier estimator")

plt.xticks(np.arange(0, 50, 5))
plt.legend()
plt.grid(True)
plt.show()

```



As seen above, the graph I made by hand is approximately the same as the one made by the Kaplan-Meier estimator.