

# Estimador de Kaplan-Meier

Gabriel D'assumpção de Carvalho

2025-07-21

## Contents

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Estimador de Kaplan-Meier</b>	<b>1</b>
2.1	Função de Sobrevivência $S(t)$	2
<b>3</b>	<b>Aplicações do estimador de Kaplan-Meier</b>	<b>2</b>
3.1	Dados de Hepatite	2

## 1 Introdução

Neste documento, apresentamos o estimador de Kaplan-Meier, uma técnica estatística não paramétrica amplamente utilizada para estimar a função de sobrevivência com base em dados observacionais ao longo do tempo. Essa abordagem é especialmente útil em estudos de sobrevivência nas áreas médica, epidemiológica e em ciências sociais. Também será demonstrado como aplicar o estimador utilizando o pacote **survival** do R, além de comparações com a estimativa feita manualmente, sem o uso de pacotes.

```
# Carregamento dos pacotes necessários

# Pacote para análise de sobrevivência
# install.packages("survival")
library(survival)

# Pacote para visualização dos dados
# install.packages("ggplot2")
library(ggplot2)
```

## 2 Estimador de Kaplan-Meier

O estimador de Kaplan-Meier é uma técnica estatística não paramétrica usada para estimar a função de sobrevivência a partir de dados observacionais ao longo de um período de tempo. Ele é amplamente utilizado em estudos de sobrevivência, como pesquisa médica, epidemiologia, ciências sociais, também sendo empregado em análise financeira e de risco.

### Exemplo de aplicação:

- Seu estimador pode ser usado para determinar o período de permanência de desemprego de um indivíduo após a sua demissão;

- Pode estimar o tempo de falha de um equipamento após a sua instalação;
- Tempo que uma flor brota após o plantio;
- Tempo de vida de um paciente após o início de um tratamento.

## 2.1 Função de Sobrevivência $S(t)$

A função de sobrevivência,  $S(t)$ , define a probabilidade de um indivíduo sobreviver além de um determinado tempo  $t$ . É expressa como:

$$S(t) = P(T > t) = 1 - F(t)$$

onde  $F(t)$  é a função de distribuição acumulada (FDA) do tempo de sobrevivência  $T$ . Para os casos contínuos, a função de sobrevivência é definida como:

$$S(t) = 1 - \int_0^t f(u) du = \int_t^\infty f(u) du$$

onde  $f(u)$  é a função densidade de probabilidade (FDP) do tempo de sobrevivência. Para os casos discretos, ela é dada como:

$$S(t) = \prod_{i:t_i \leq t} \left(1 - \frac{d_i}{n_i}\right)$$

Sendo:

- $t_i$ : tempo em que ocorreu pelo menos um evento (falha);
- $d_i$ : número de eventos (falhas) observados no tempo  $t_i$ ;
- $n_i$ : número de indivíduos em risco no tempo  $t_i$ .

Nota-se que até o momento não foi introduzido o conceito de censura. A censura ocorre quando o tempo de sobrevivência de um indivíduo não é completamente observada, ou seja, quando o evento de interesse (como falha ou morte) não ocorre durante o período de observação.

## 3 Aplicações do estimador de Kaplan-Meier

Os exemplos utilizados nesse estudo foram retirados do livro: “Análise de Sobrevivência Aplicada – Eurico Antônio Colosimo e Suely Ruiz Giolo, 2006. Ed. Blucher”

### 3.1 Dados de Hepatite

Foram escolhidos 29 pacientes com hepatite viral aguda e separados aleatoriamente em dois grupos. Os pacientes foram acompanhados durante 4 meses ou até a falha ou censura do paciente.

- **Tipo de estudo:** Estudo clínico randomizado;
- **Grupo de tratamento:** Pacientes que receberam terapia com esteroides;
- **Grupo controle:** Pacientes que receberam placebo.

```
# Carregamento dos Dados
```

```
controle <- data.frame(
  tempo = c(
```

```

1, 2, 3, 3, 3, 5, 5, 16, 16, 16, 16,
16, 16, 16, 16
),
evento = c(0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
)

casos <- data.frame(
  tempo = c(
    1, 1, 1, 1, 4, 5, 7, 8, 10, 10, 12, 16, 16,
    16
  ),
  evento = c(1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0)
)

survControle <- survfit(Surv(controle$tempo, controle$evento) ~ 1)
print(summary(survControle))

## Call: survfit(formula = Surv(controle$tempo, controle$evento) ~ 1)
##
##   time n.risk n.event survival std.err lower 95% CI upper 95% CI
##    3      13       2   0.846    0.1      0.671      1
survCasos <- survfit(Surv(casos$tempo, casos$evento) ~ 1)
print(summary(survCasos))

## Call: survfit(formula = Surv(casos$tempo, casos$evento) ~ 1)
##
##   time n.risk n.event survival std.err lower 95% CI upper 95% CI
##    1      14       3   0.786    0.110    0.598    1.000
##    5       9       1   0.698    0.128    0.488    0.999
##    7       8       1   0.611    0.138    0.392    0.952
##    8       7       1   0.524    0.143    0.306    0.896
##   10       6       1   0.437    0.144    0.229    0.832

ekm <- function(time, evento, classEvento = 1) {
  tempo_evento <- sort(unique(time[evento == classEvento]))

  data <- data.frame(
    time = tempo_evento,
    n_risk = NA,
    n_event = NA,
    censored = NA,
    q_i = NA,
    p_i = NA,
    S_t_i = NA
  )

  S_t_anterior <- 1

  for (i in seq_along(data$time)) {
    t <- data$time[i]

    data$n_risk[i] <- sum(time >= t)
    data$n_event[i] <- sum(time == t & evento == classEvento)
    data$censored[i] <- sum(time == t & evento != classEvento)
  }
}

```

```

# Cálculo das probabilidades
if (data$n_risk[i] > 0) {
  data$q_i[i] <- data$n_event[i] / data$n_risk[i]
  data$p_i[i] <- 1 - data$q_i[i]
  data$S_t_i[i] <- S_t_anterior * data$p_i[i]
  S_t_anterior <- data$S_t_i[i] # atualiza
} else {
  data$S_t_i[i] <- S_t_anterior
}
}

return(data)
}

```

```

ekmControle <- ekm(controle$tempo, controle$evento, 1)
print(ekmControle)

```

```

##   time n_risk n_event censored      q_i      p_i      S_t_i
## 1     3     13        2         1 0.1538462 0.8461538 0.8461538

```

```

ekmCasos <- ekm(casos$tempo, casos$evento, 1)
print(ekmCasos)

```

```

##   time n_risk n_event censored      q_i      p_i      S_t_i
## 1     1     14        3         1 0.2142857 0.7857143 0.7857143
## 2     5      9        1         0 0.1111111 0.8888889 0.6984127
## 3     7      8        1         0 0.1250000 0.8750000 0.6111111
## 4     8      7        1         0 0.1428571 0.8571429 0.5238095
## 5    10      6        1         1 0.1666667 0.8333333 0.4365079

```