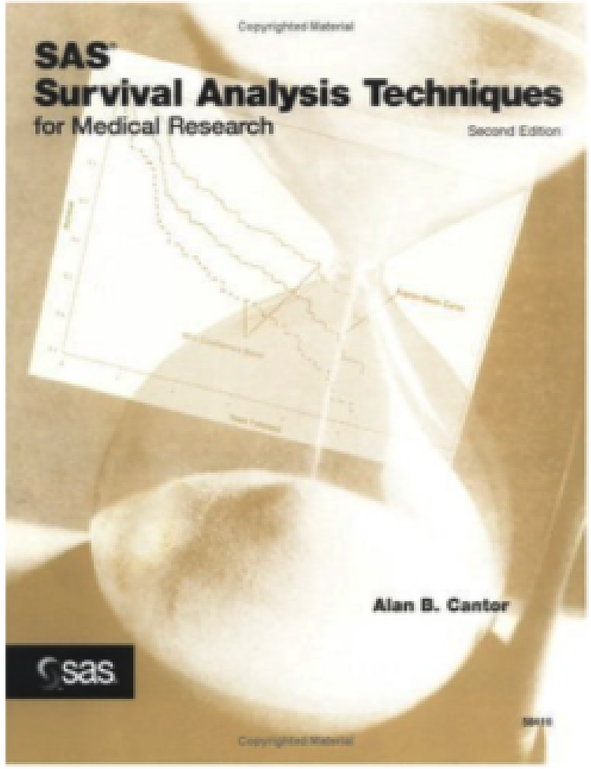
What is survival analysis?

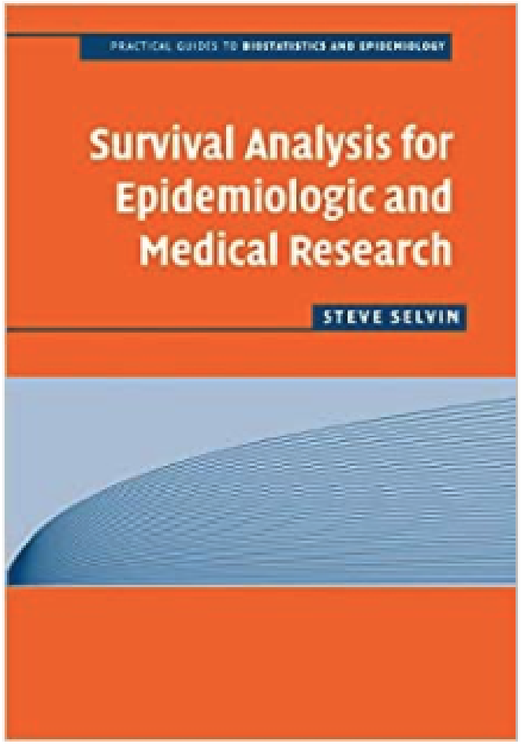
SURVIVAL ANALYSIS IN PYTHON

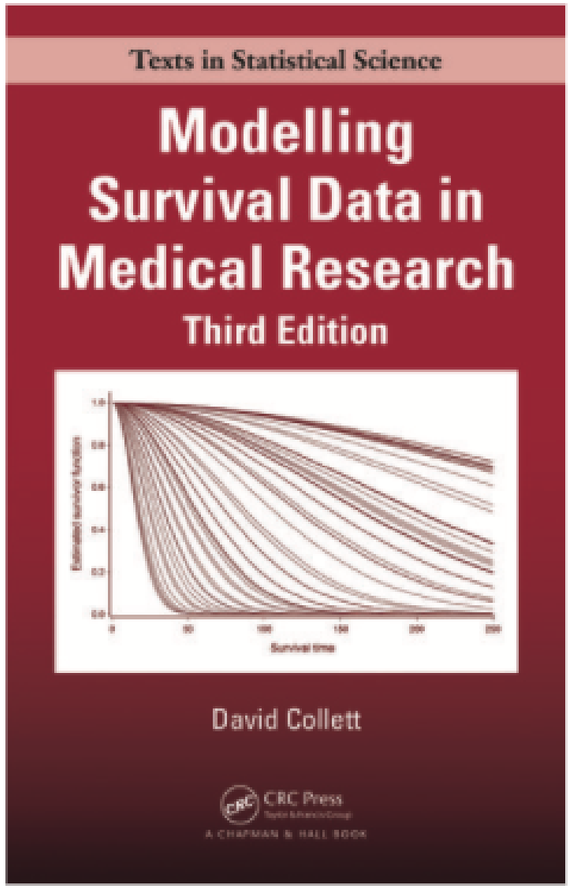
Shae Wang
Senior Data Scientist

datacamp

What is survival analysis?







What is survival analysis?

A branch of statistics focused on analyzing time to an event.

Examples

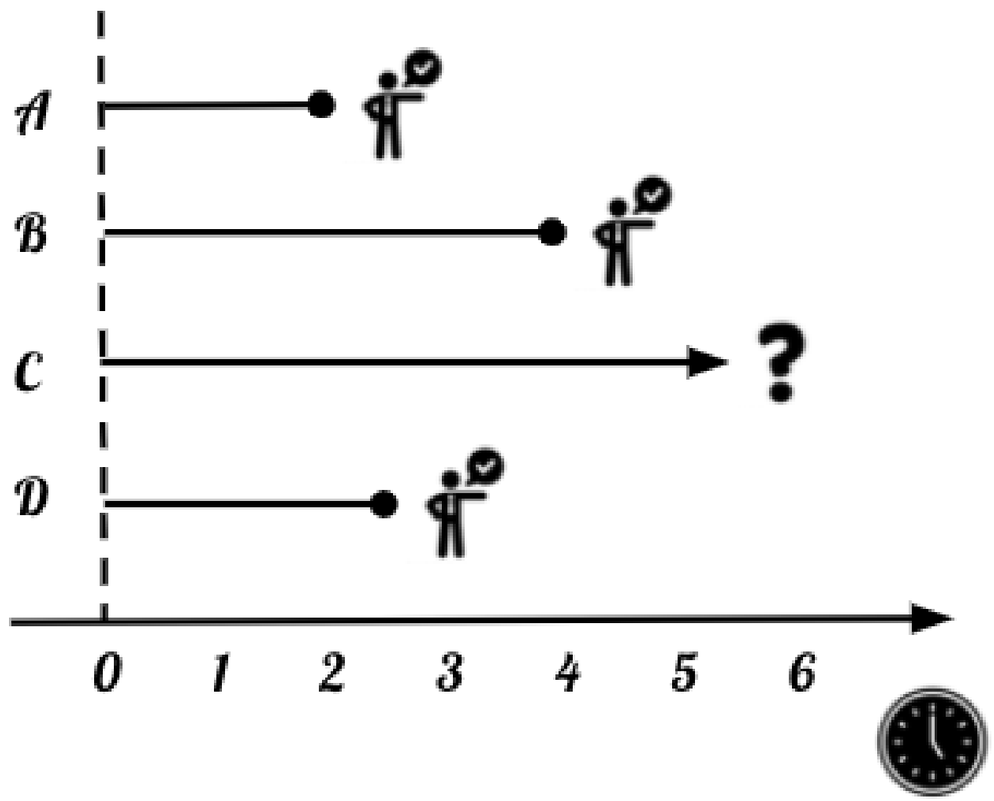
- Time until equipment failure
- Time until loan default
- Time until free-trial users convert to subscribers
- Time until a tree's first fruit

What is survival analysis?

A branch of statistics focused on analyzing time to an event.

- **Event**: Equipment failure, loan default, disease recurrence, user conversion, or other experience of interest
- **Survival**: The event of interest does not occur
- **Survival duration**: Time until the event of interest occurs (or the end of our observations)

Time-to-event data



ID	Duration	Observed
A	2	Yes
B	4	Yes
C	5.2	No
D	2.5	Yes

- **Event:** must be clear and unambiguous, cannot *partially* happen
- There should be abundant data **where the event does occur** for survival analysis to be effective.

The use cases of survival analysis

- Calculate the proportion of subjects that will experience an event
- Estimate the time until event/failure in a population
- Calculate the rate of event in a population
- What factors increase or decrease survival?
- Predict probabilities of event occurrence for subjects

Predict battery failure time

Battery time-to-event data

Battery ID	Duration	Dead	Brand	Truck
1	2.5 yrs	No	Brand A	Long
2	6 yrs	Yes	Brand B	Short
3	5 yrs	No	Brand B	Long
...	...	...	...	...
1000	4.5 yrs	Yes	Brand A	Short

- **Duration**: how long the battery has been in use
- **Dead**: whether the battery has died

Let's practice!

SURVIVAL ANALYSIS IN PYTHON

Why use survival analysis?

SURVIVAL ANALYSIS IN PYTHON

Shae Wang
Senior Data Scientist

datacamp

Average battery life example

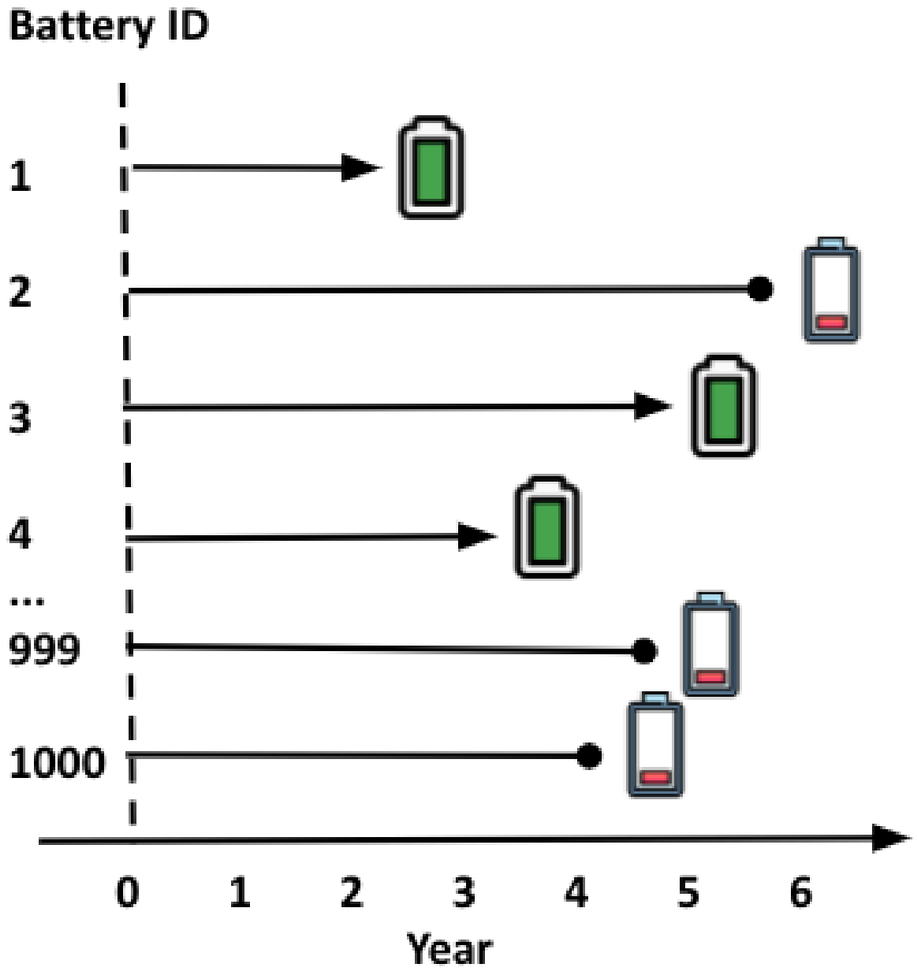
DataFrame name: `battery_df`

Battery ID	Duration	Dead	Brand	Truck
1	2.5 yrs	No	Brand A	Long
2	6 yrs	Yes	Brand B	Short
3	5 yrs	No	Brand B	Long
...	...	...	...	...
1000	4.5 yrs	Yes	Brand A	Short

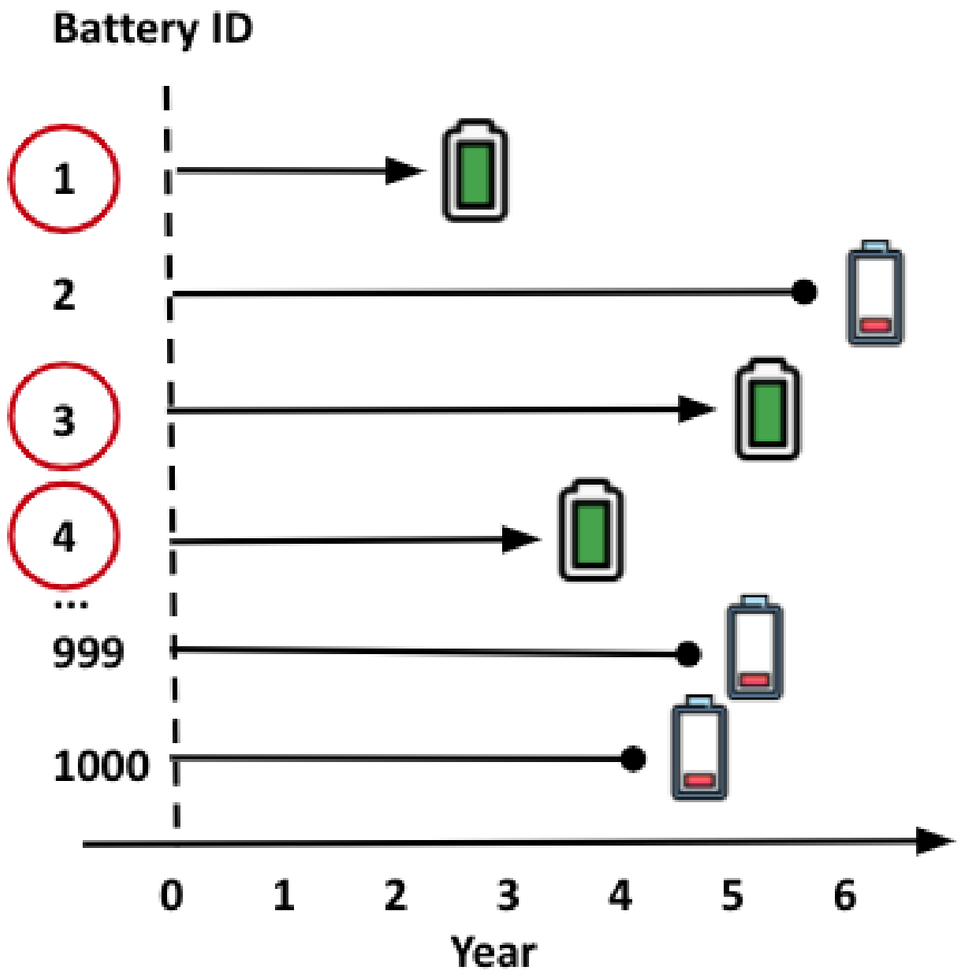
What's the average battery lifetime?

```
np.average(battery_df["Duration"])
```

Average battery life example



Censorship in battery life



- $T_{duration} \neq T_{lifetime}$ for batteries that have not died.
- Batteries 1, 3, 4, and other batteries whose failures haven't been observed are **inappropriately accounted for** in the averaging.

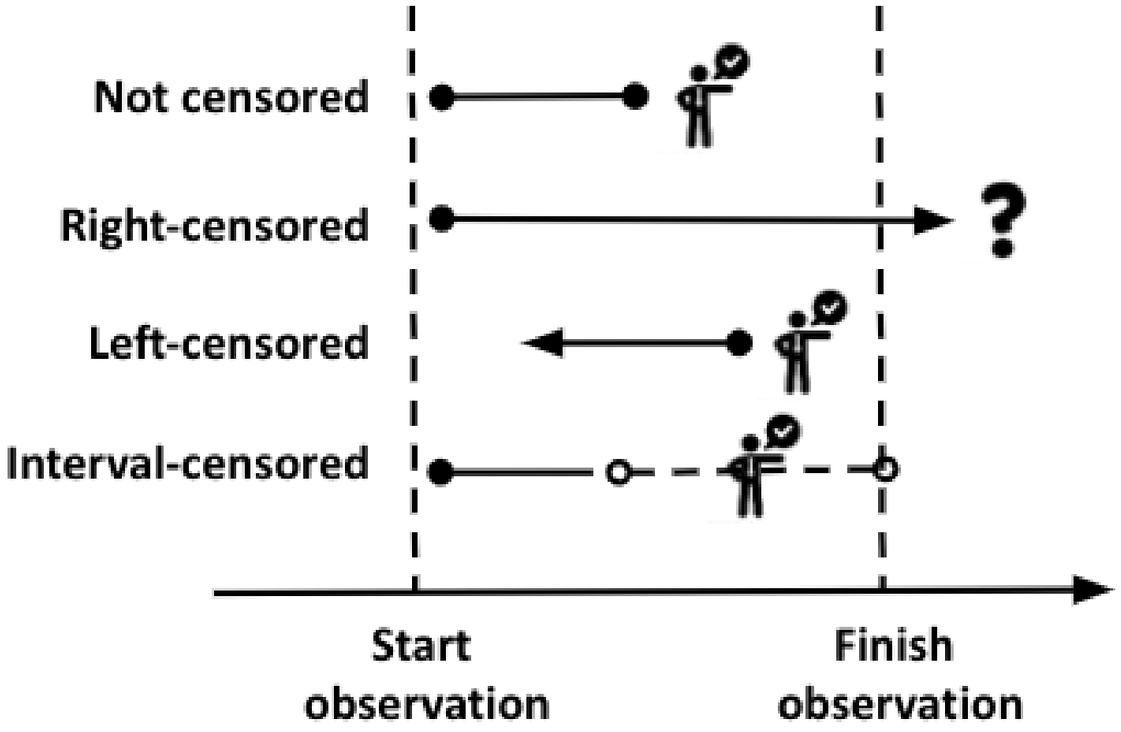
The censorship problem

When the survival time is only partially known.

How does censorship happen?

- The event has **not yet occurred** at the end of the observation.
  - e.g. a free trial user has not converted to a paid user at the end of an experiment.
- The individual's data is missing because of a **dropout** or **loss of contact**.
  - e.g. a free trial user declines to share data for the experiment.

Types of censorship



- **Not censored**: the event occurred and survival duration is known.
- **Right-censored**: the survival duration is greater than the observed duration.
- **Left-censored**: the survival duration is less than observed duration.
- **Interval-censored**: the survival duration is within a certain range but not exactly known.

Why is censorship bad?

Aggregated statistics

- A type of missing data.
- Skew statistics, i.e. `np.average()`, `max()`, `min()`.

Regression

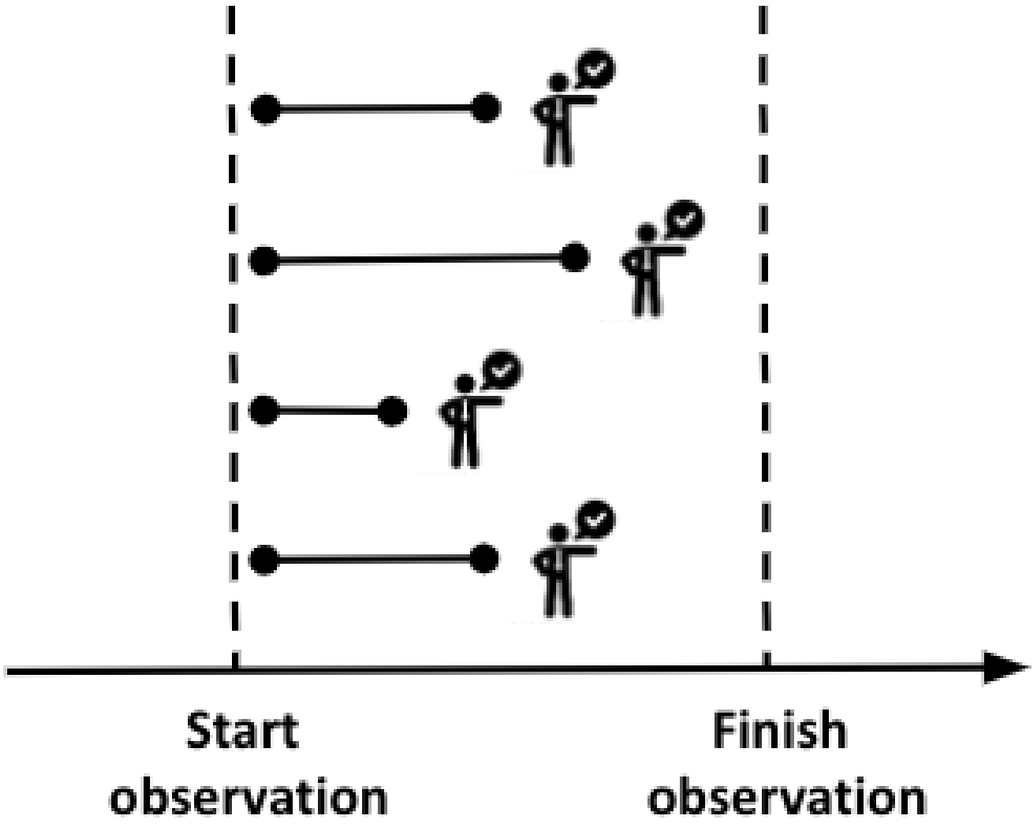
- Linear regression line minimizes the sum of squared errors.
- For censored data, we don't know the error terms.

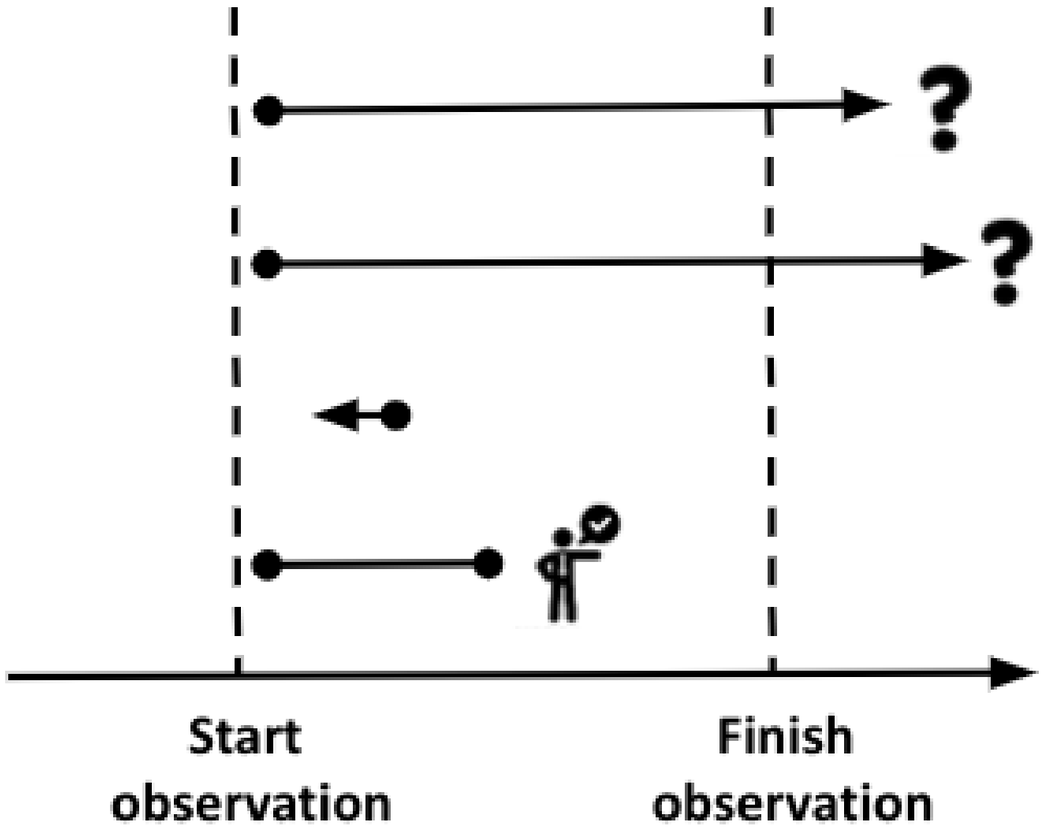
The survival function

- Does not impute censored data.
- Models the probability of a survival duration being larger than a certain value.

$$S(t) = Pr(T > t)$$

Survival analysis versus censorship





Checking data for censorship

Is there a way to identify which data points are censored?

Step 1) Check for censorship columns (often preprocessed).

Is too much data censored?

Step 2) Check the proportion of data points that are censored (a rule of thumb is 50%).

Is the censorship non-informative and random?

Step 3) Investigate the causes of the censorship to ensure that whether a data point is censored has no impact on survival.

Let's practice!

SURVIVAL ANALYSIS IN PYTHON

Your first survival curve!

SURVIVAL ANALYSIS IN PYTHON

Shae Wang
Senior Data Scientist

datacamp

The survival function

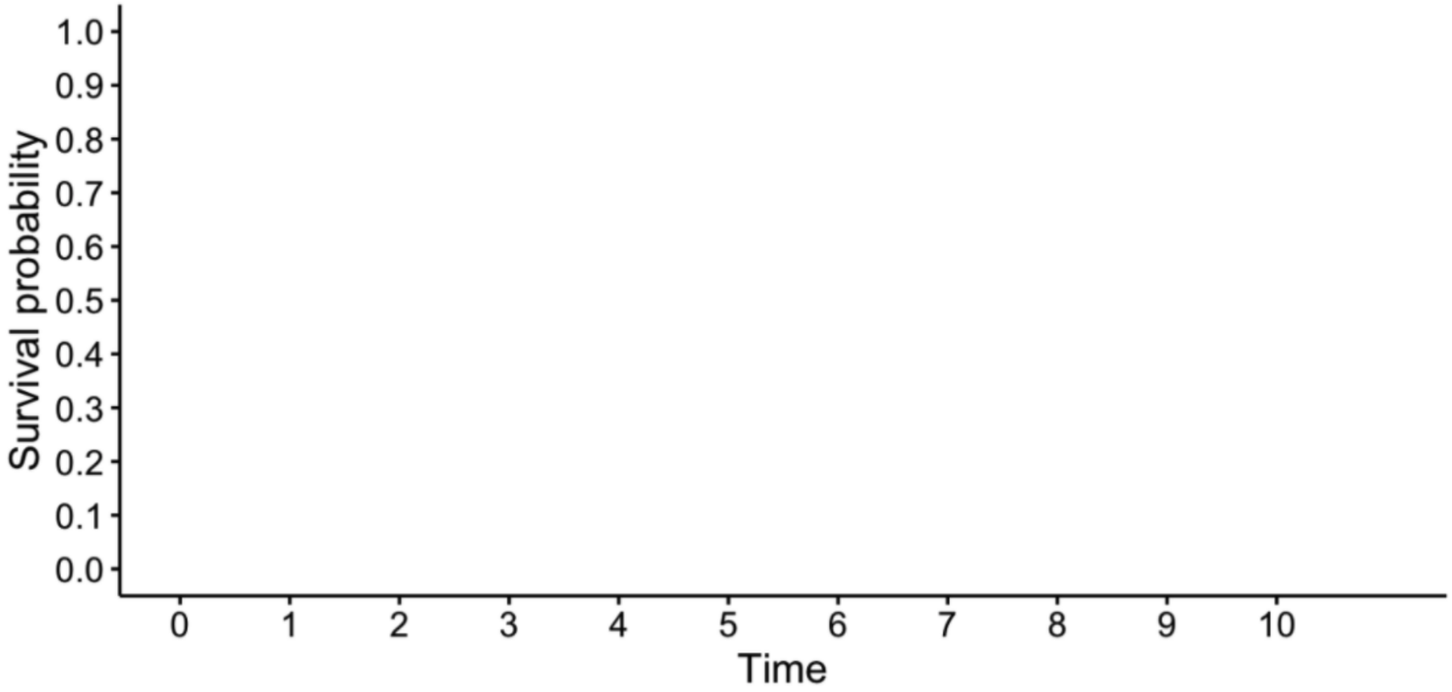
- T: when the event of interest occurs
- t: any point in time during an observation

$$S(t) = Pr(T > t)$$

- $S(t)$: models the probability that the event of interest happens after t
- $Pr(T > t)$: the survival probability

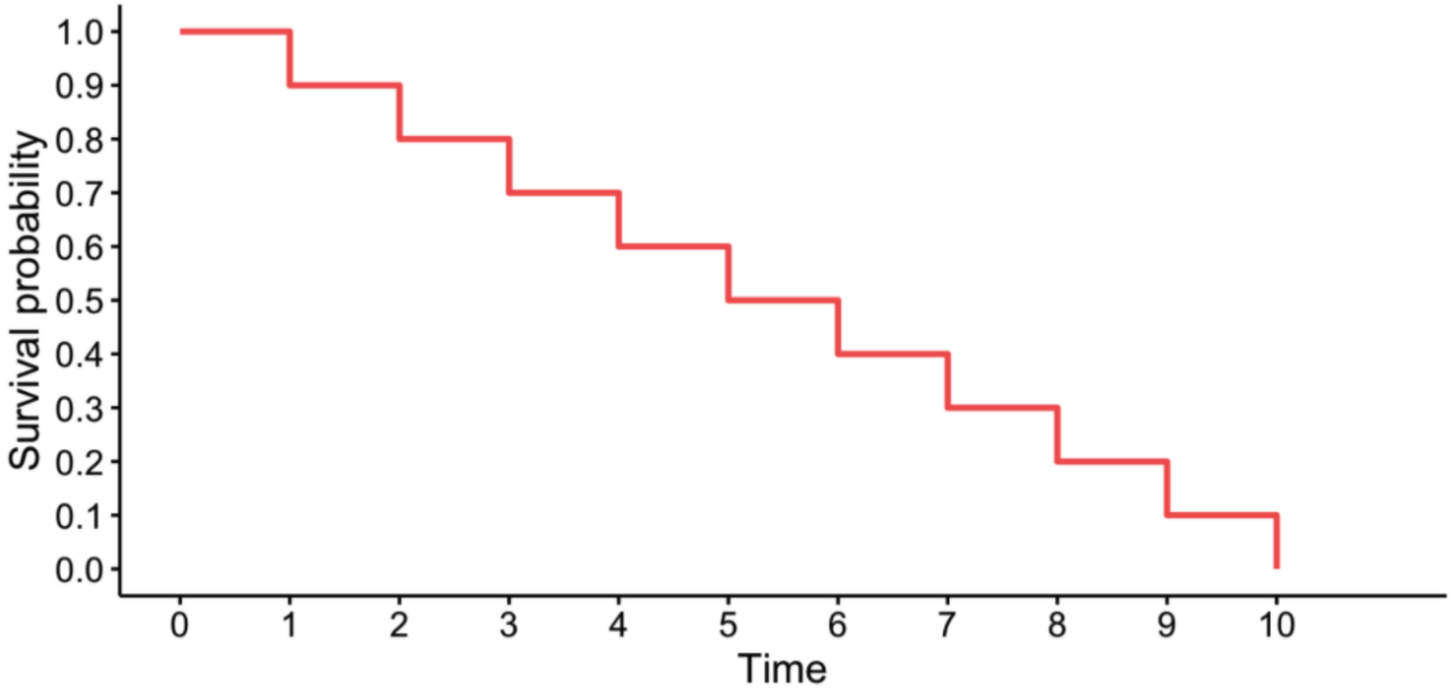
The survival curve

$$S(t) = Pr(T > t)$$



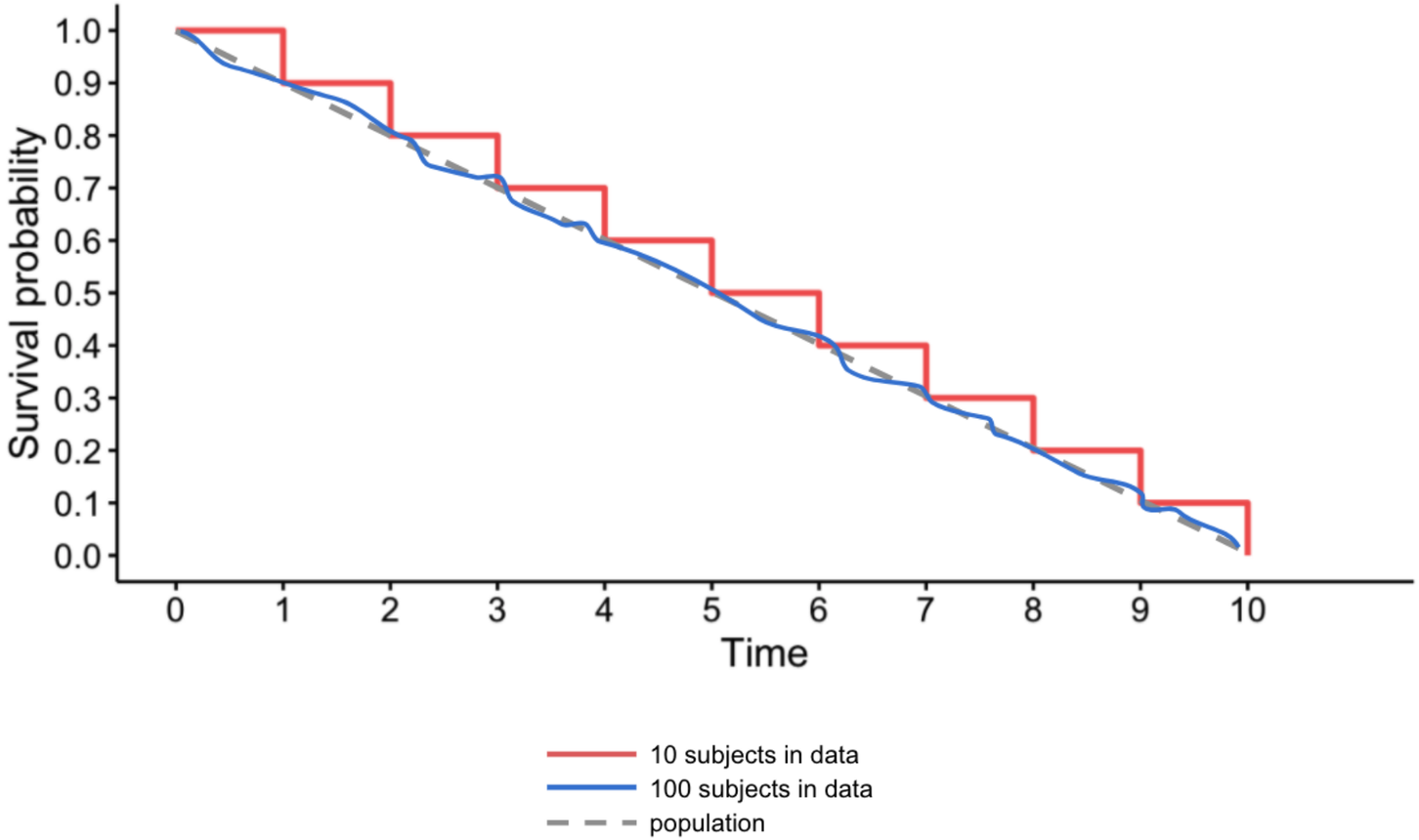
The survival curve

$$S(t) = Pr(T > t)$$



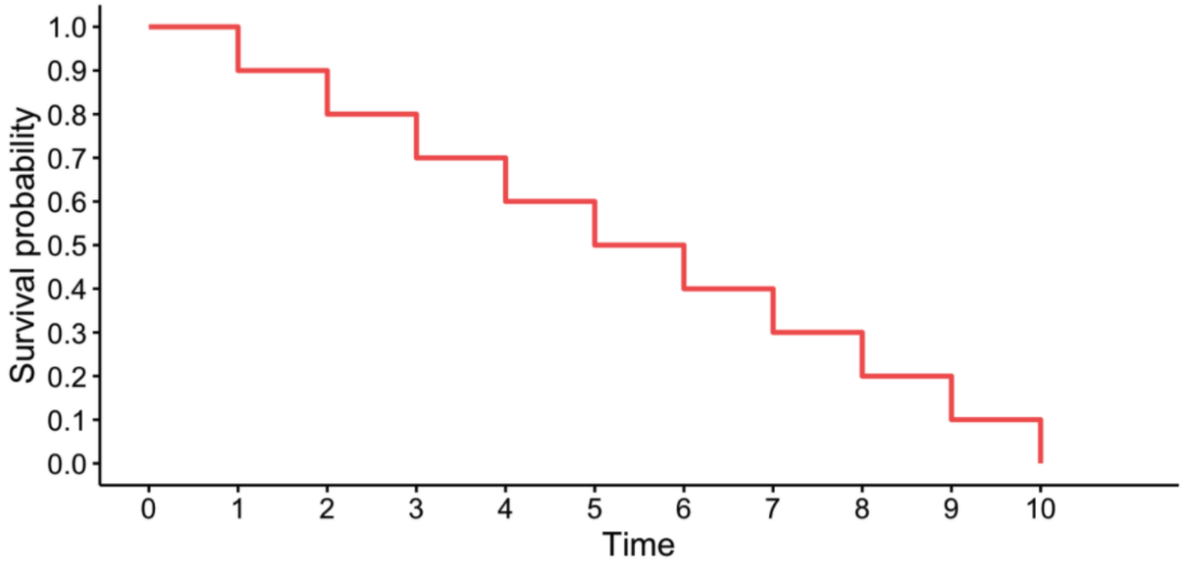
The survival curve

$$S(t) = Pr(T > t)$$



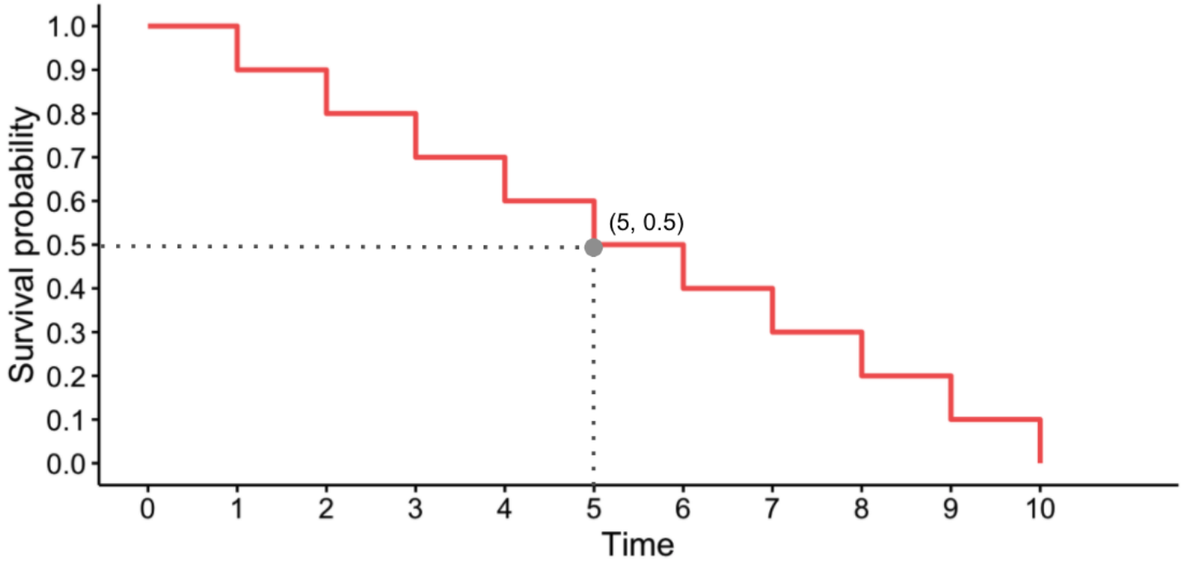
Interpreting a survival curve

- Point(a, b): the probability that an individual survives longer than a is b



Interpreting a survival curve

- Point(a, b): the probability that an individual survives longer than a is b
- Flatter curve: lower rate of event occurrence
- Steeper curve: higher rate of event occurrence



Non-parametric versus parametric models

Non-parametric modeling

- Make no assumptions about the shape of the data

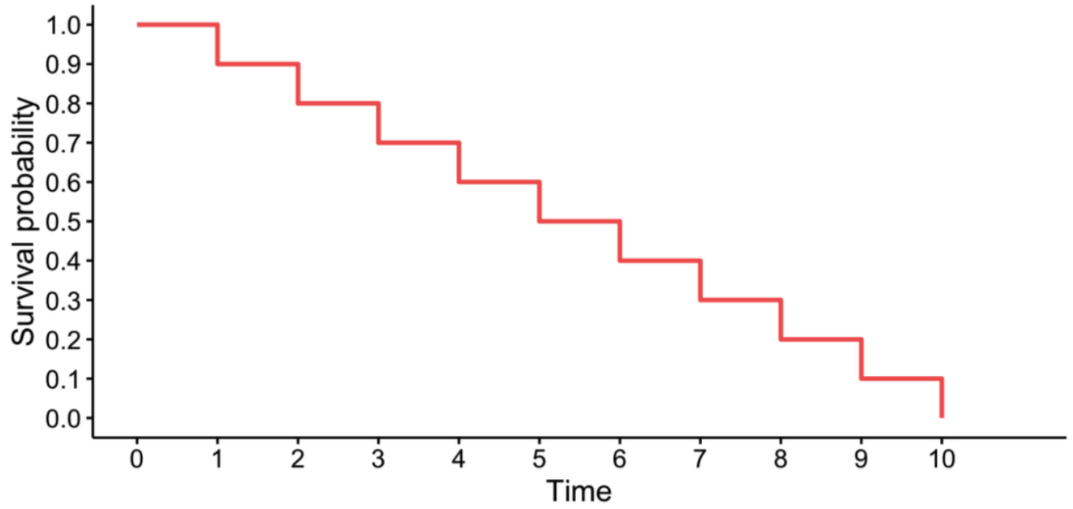
Parametric modeling

- Make some assumptions about the shape of the data
- Described with a limited set of parameters
  - i.e. the survival curve may be assumed to follow an exponential distribution

Non-parametric versus parametric models

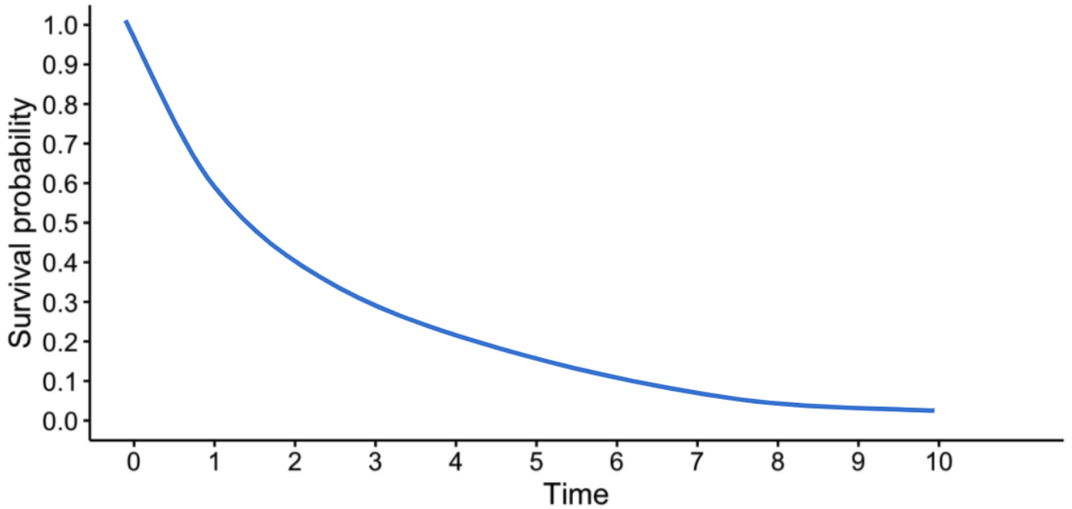
Non-parametric modeling

- Survival curve is **usually NOT smooth**



Parametric modeling

- Survival curve is **usually smooth**



- Relies on the parametric model actually being a good description of the data

Drawing a survival curve

The `lifelines` package is a complete survival analysis library.

- Fit survival functions to data
- Plot survival curves based on the fitted survival functions

```
import lifelines
import matplotlib.pyplot as plt
```

`.fit(durations, event_observed)`

`.plot_survival_function()`

Survival curve example

DataFrame name: `mortgage_df`

id	duration	paid_off
1	25	0
2	17	1
3	5	0
...	...	...
100	30	1

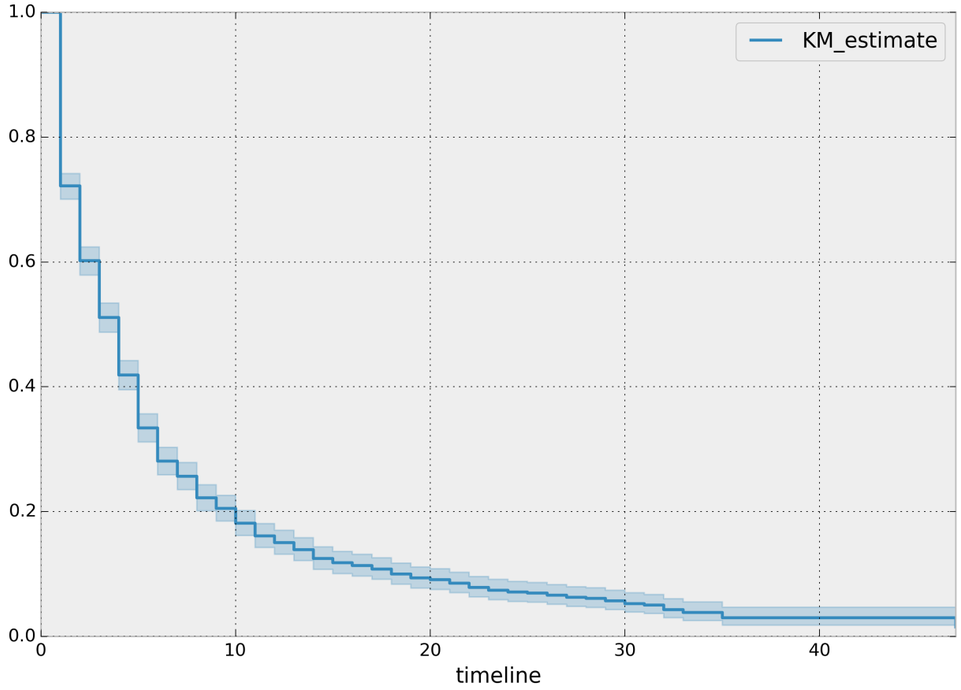
- `id` : the id of a mortgage loan
- `duration` : the number of years the mortgage is not paid off
- `paid_off` : `1` if the mortgage is fully paid off, `0` if not fully paid off

Survival curve example

```
import lifelines
from matplotlib import pyplot as plt
```

```
kmf = lifelines.KaplanMeierFitter()
kmf.fit(duration=mortgage_df["duration"],
        event_observed=mortgage_df["paid_off"])
```

```
kmf.plot_survival_function()
plt.show()
```



Let's practice!

SURVIVAL ANALYSIS IN PYTHON