Models for time-to-event data

From Cox's proportional hazards model to deep learning

Sebastian Pölsterl

Artificial Intelligence in Medical Imaging |
Ludwig Maximilian Universität Munich

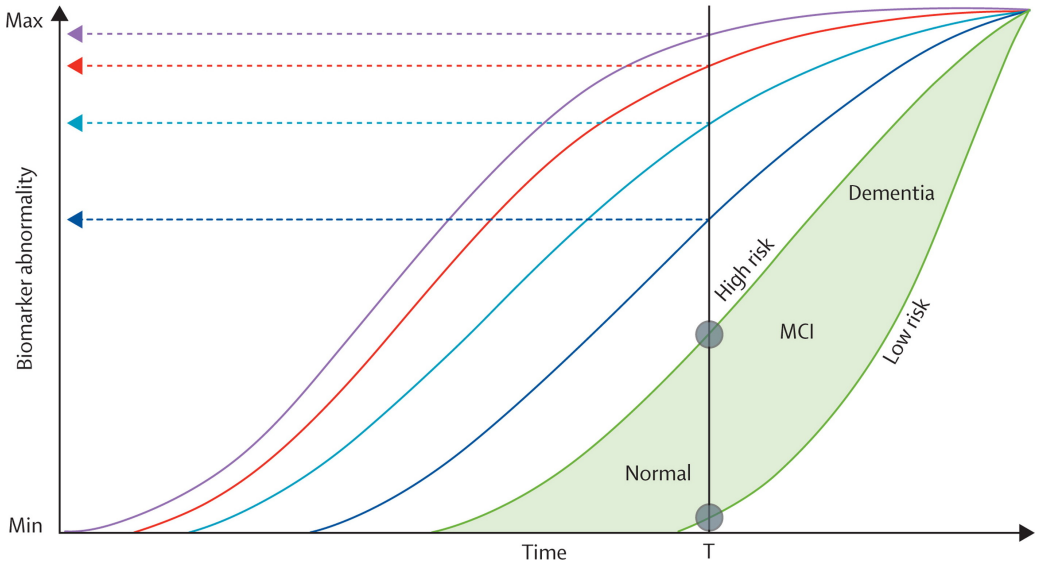
October 2nd 2018

École Centrale de Nantes

Outline

Time-to-event Data in Medical Research

Alzheimer's disease progression

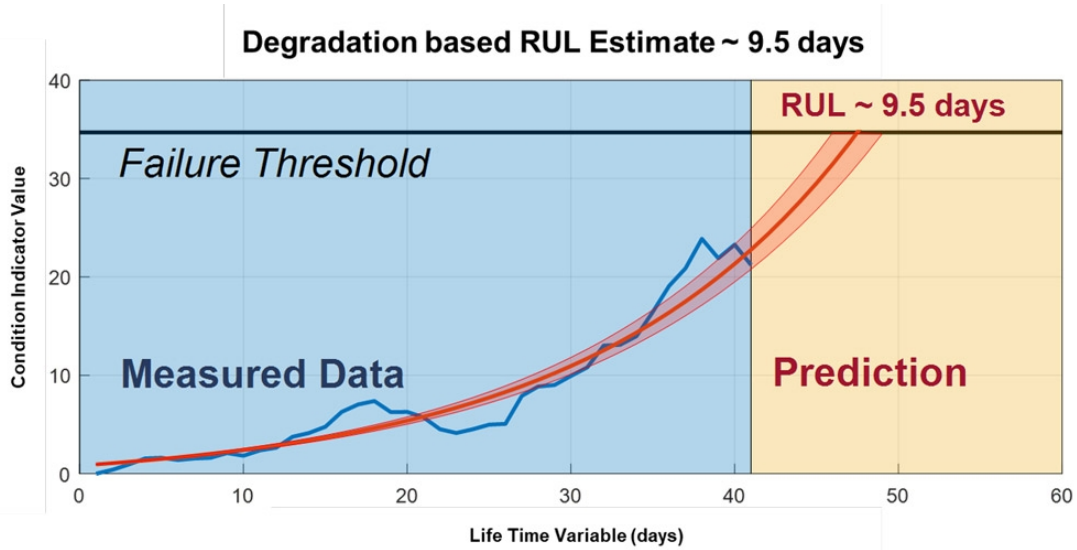


Source: Jack et al. (2013)

- Mild cognitive impairment (MCI) is a common precursor to dementia in Alzheimer's disease and is associated with isolated memory loss.
- Some patients with MCI remain stable, whereas others progress to Alzheimer's disease.
- For an effective therapy, we want to know the probability of conversion at any time point.

Time-to-event Data in Maintenance

Remaining useful life of equipment

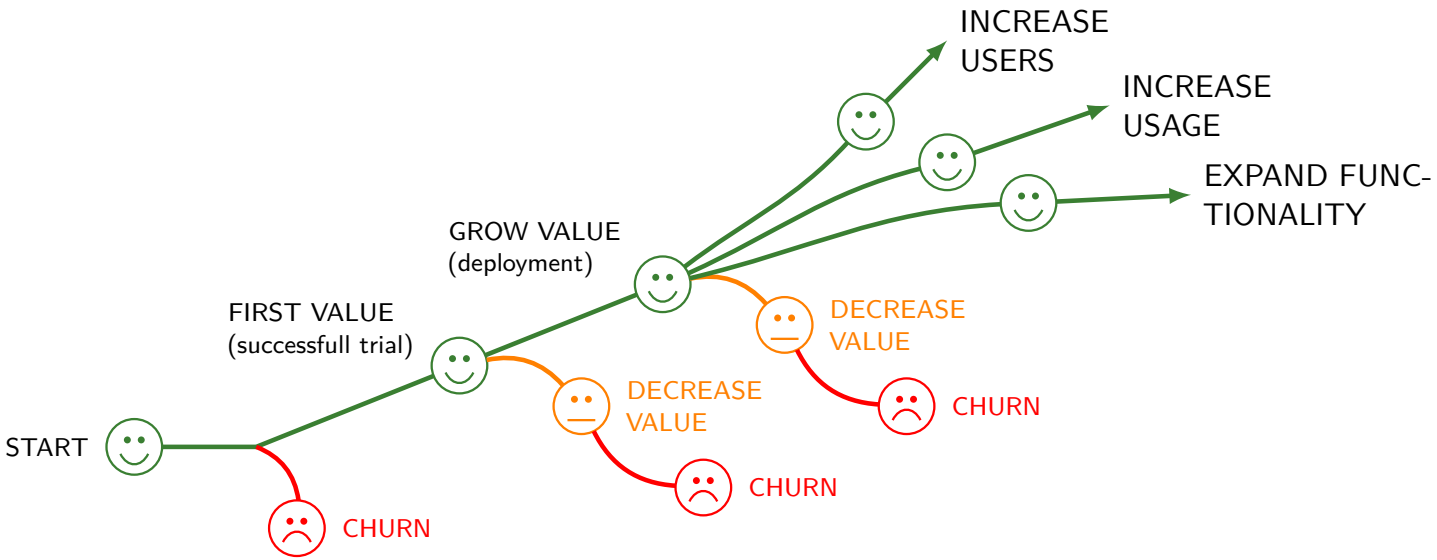


Source: MathWorks

- Most equipment, such as a pump, will experience failure eventually.
- Failure is usually determined by threshold values on various censors: temperature cannot exceed 74°C and pressure must be under 10 bar.
- We want to know the probability of failure at any time point such that replacing the equipment can be scheduled in advance to minimize downtime.

Time-to-event Data in Economics

Customer relationship management

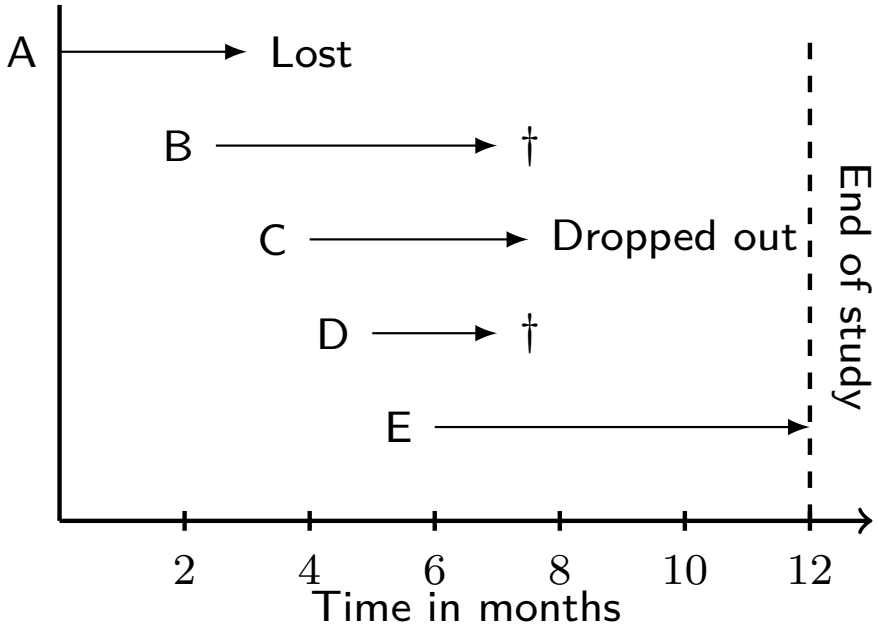


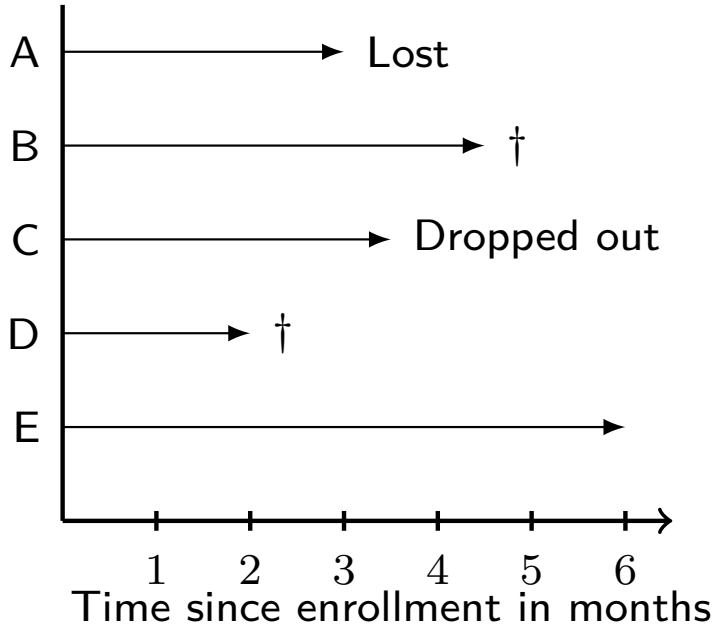
Source: For Entrepreneurs

- All businesses will lose some of its customers (customer churn).
- For each customer, we have a record of purchases and previous interactions with the company.
- We want to know how likely it is for a customer to turn away (churn) at any given time point so we can provide targeted incentives to induce customers to stay.

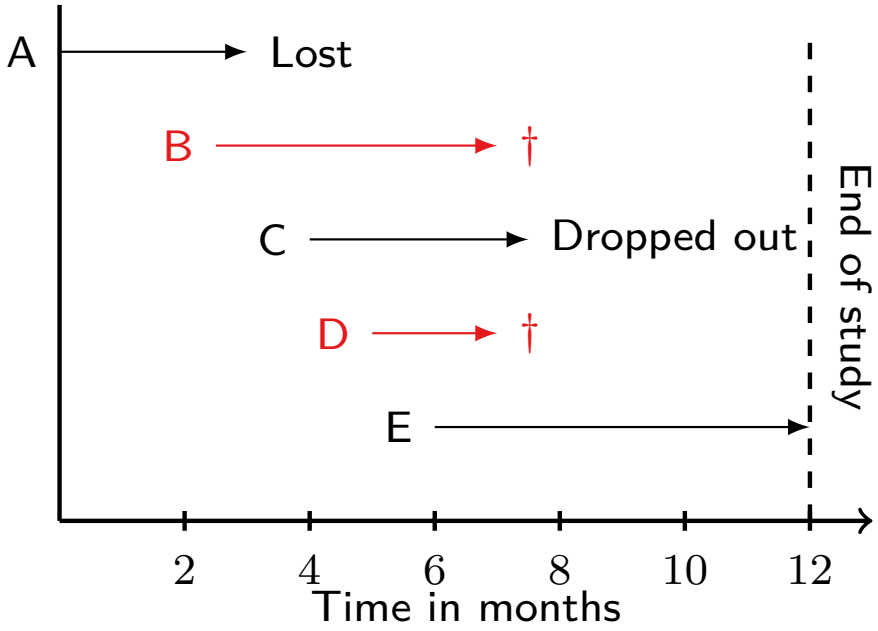
Outline

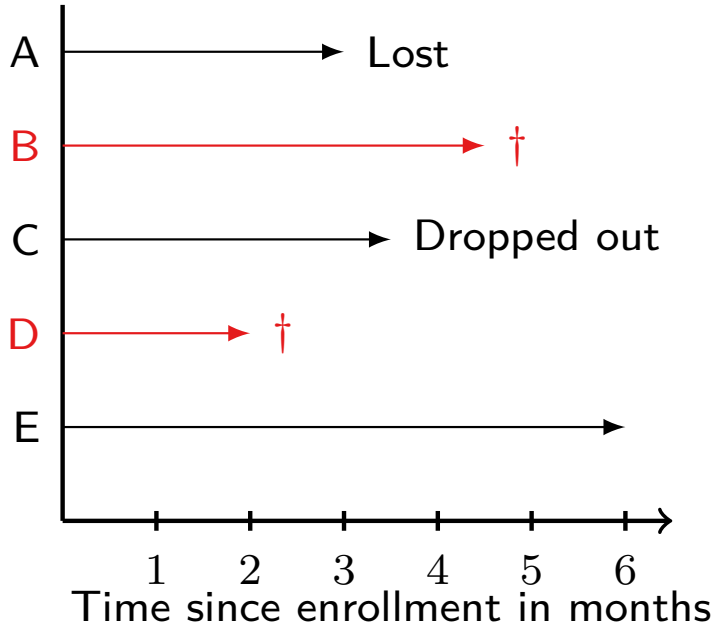
Censoring


A
Lost
B
†
C
Dropped out
D
†
E
End of study
2
4
6
8
10
12
Time in months


A
Lost
B
†
C
Dropped out
D
†
E
1
2
3
4
5
6
Time since enrollment in months

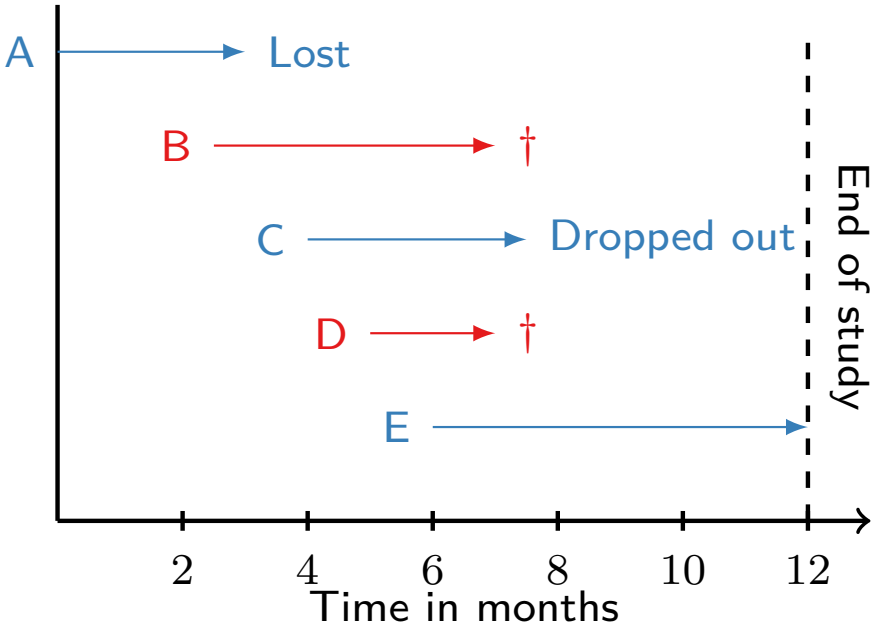
Censoring

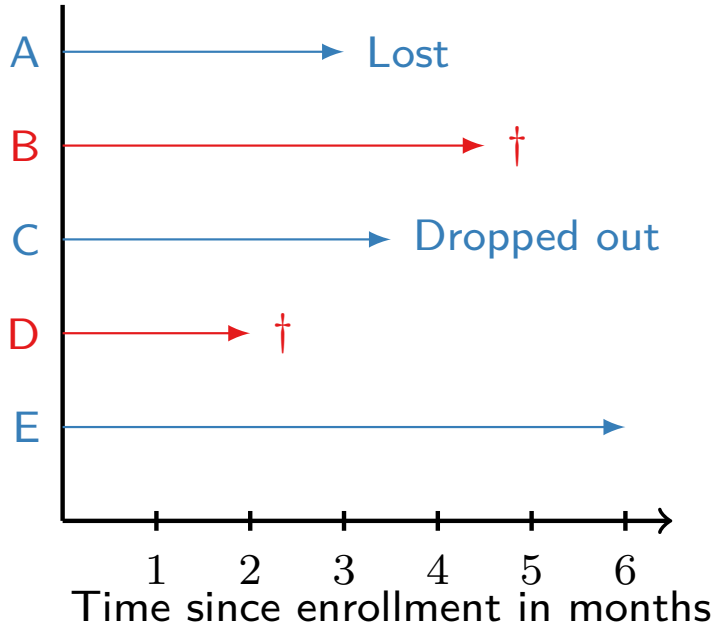




- A record is **uncensored** if an event was observed during the study period: the exact time of the event is known.

Censoring





- A record is **uncensored** if an event was observed during the study period: the exact time of the event is known.
- A record is **right censored** if a patient remained event-free: it is unknown whether an event occurred after the study ended.

Types of Censoring

Let y_i denote the **observable** time, t_i the **actual** time of an event, and c_i the time of **censoring**.

- Right censoring

$$y_i = \min(c_i^{\text{right}}, t_i)$$

Types of Censoring

Let y_i denote the **observable** time, t_i the **actual** time of an event, and c_i the time of **censoring**.

- Right censoring

$$y_i = \min(c_i^{\text{right}}, t_i)$$

- Left censoring

$$y_i = \max(c_i^{\text{left}}, t_i)$$

Types of Censoring

Let y_i denote the **observable** time, t_i the **actual** time of an event, and c_i the time of **censoring**.

- Right censoring

$$y_i = \min(c_i^{\text{right}}, t_i)$$

- Left censoring

$$y_i = \max(c_i^{\text{left}}, t_i)$$

- Interval censoring

$$t_i \in (\tau_i^l; \tau_i^r]$$

Types of Censoring

Let y_i denote the **observable** time, t_i the **actual** time of an event, and c_i the time of **censoring**.

- Right censoring
$$y_i = \min(c_i^{\text{right}}, t_i)$$
- Left censoring
$$y_i = \max(c_i^{\text{left}}, t_i)$$
- Interval censoring
$$t_i \in (\tau_i^l; \tau_i^r]$$
- Any combination of left, right, or interval censoring may occur in a study.

Basic Quantities

Let T denote a **continuous** non-negative random variable corresponding to a patient's survival time with probability density function $f(t)$.

Survival function

$$S(t) = P(T > t) = 1 - P(T \leq t) = 1 - F(t) = \int_t^\infty f(u)du$$

Basic Quantities

Let T denote a **continuous** non-negative random variable corresponding to a patient's survival time with probability density function $f(t)$.

Survival function

$$S(t) = P(T > t) = 1 - P(T \leq t) = 1 - F(t) = \int_t^\infty f(u)du$$

Hazard function

$$h(t) = \lim_{\Delta t \to 0} \frac{P(t \leq T < t + \Delta t \mid T \geq t)}{\Delta t} \geq 0$$

Basic Quantities

Let T denote a **continuous** non-negative random variable corresponding to a patient's survival time with probability density function $f(t)$.

Survival function

$$S(t) = P(T > t) = 1 - P(T \leq t) = 1 - F(t) = \int_t^\infty f(u)du$$

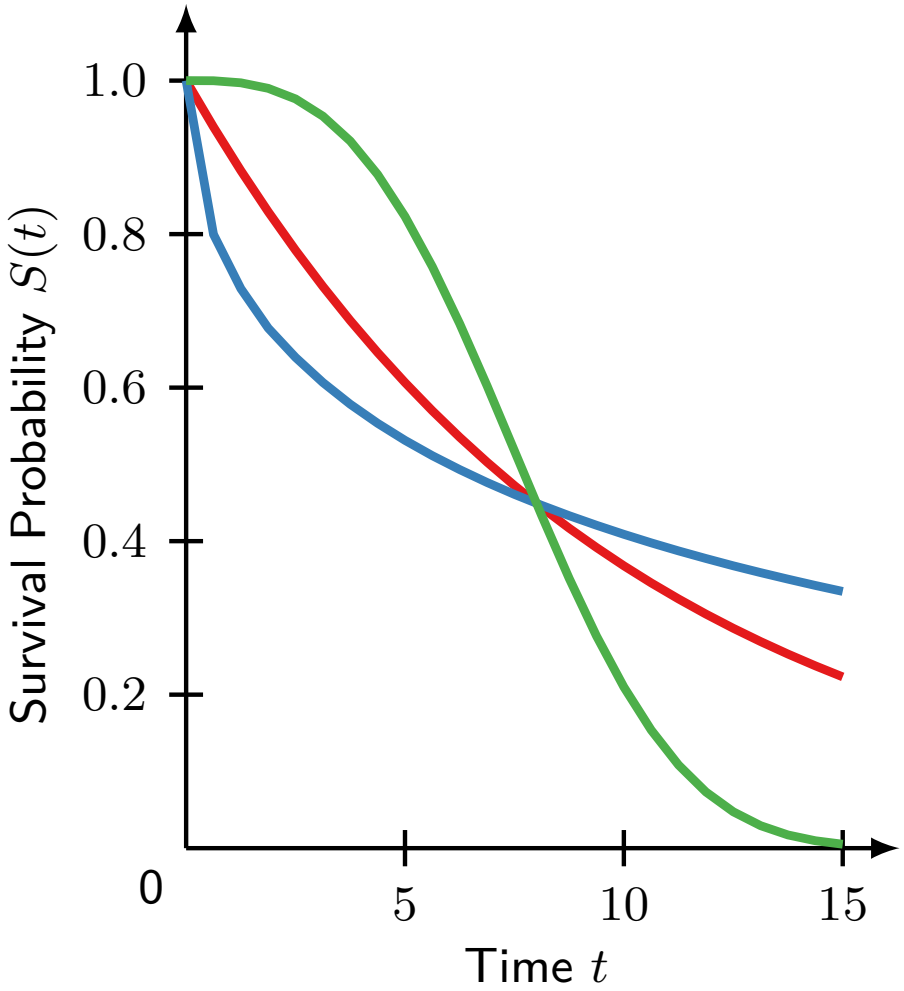
Hazard function

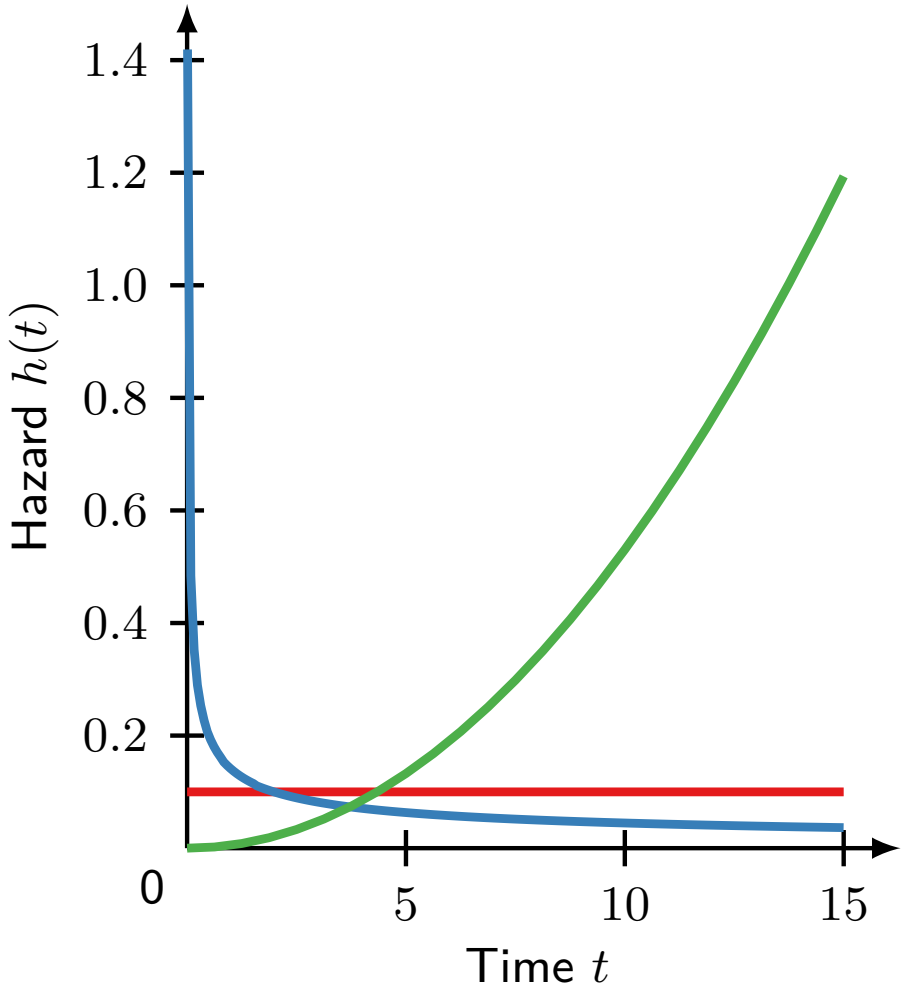
$$h(t) = \lim_{\Delta t \to 0} \frac{P(t \leq T < t + \Delta t \mid T \geq t)}{\Delta t} \geq 0$$

Cumulative hazard function

$$H(t) = \int_0^t h(u)du$$

Survival and Hazard Function





$$h(t) = \frac{f(t)}{S(t)}; \qquad H(t) = -\log S(t)$$

Discrete Survival Times

Let T be a **discrete** random variable, which can take on values t_i $(i \in \mathbb{N})$ with probability mass function $P(T = t_i)$ and $t_i < t_j$ if and only if $i < j$.

Survival function

$$S(t) = \sum_{\{i|t_i > t\}} P(T = t_i) \Leftrightarrow P(T = t_i) = S(t_{i-1}) - S(t_i)$$

Discrete Survival Times

Let T be a **discrete** random variable, which can take on values t_i $(i \in \mathbb{N})$ with probability mass function $P(T = t_i)$ and $t_i < t_j$ if and only if $i < j$.

Survival function

$$S(t) = \sum_{\{i|t_i>t\}} P(T = t_i) \Leftrightarrow P(T = t_i) = S(t_{i-1}) - S(t_i)$$

Hazard function

$$h(t) = P(T = t_i \mid T \geq t_i)$$

Discrete Survival Times

Let T be a **discrete** random variable, which can take on values t_i $(i \in \mathbb{N})$ with probability mass function $P(T = t_i)$ and $t_i < t_j$ if and only if $i < j$.

Survival function

$$S(t) = \sum_{\{i|t_i > t\}} P(T = t_i) \Leftrightarrow P(T = t_i) = S(t_{i-1}) - S(t_i)$$

Hazard function

$$h(t) = P(T = t_i \mid T \geq t_i)$$

Cumulative hazard function

$$H(t) = \sum_{\{i|t_i \leq t\}} h(t_i)$$

Outline

Maximum Likelihood Optimization

- Assume we have a dataset of d covariates for each of n observations:

$$\mathcal{D} = \{(y_i, \boldsymbol{x}_i)\}_{i=1}^{n}$$

- We want to fit a model with parameters $\boldsymbol{\Theta}$ to estimate $S(t)$ – the probability of survival beyond time t – via maximum likelihood optimization.
- Observed times y_i can be
  1. uncensored
  2. right-censored
  3. left-censored
  4. interval-censored
- **We need to consider carefully what information each observation gives us.**

Noninformative Censoring

Definition (Noninformative Censoring)

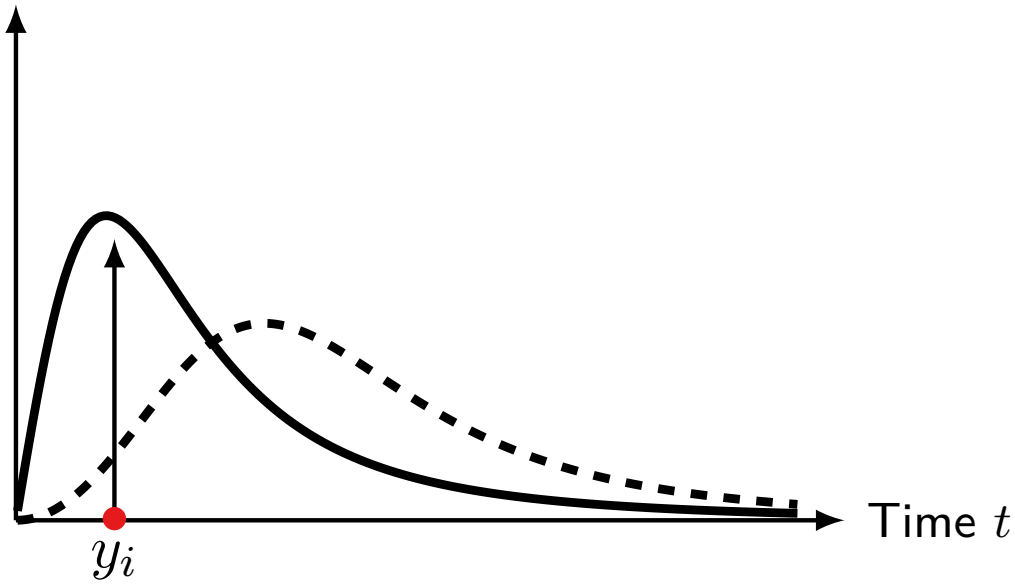
Usually, we assume that the distribution of survival times T is independent of the distribution of censoring times C:

$$T \perp C \,|\, \boldsymbol{x}$$

This assumption would be violated if the prognosis of individuals who get censored is worse compared to those who are not censored.

Constructing the Likelihood Function

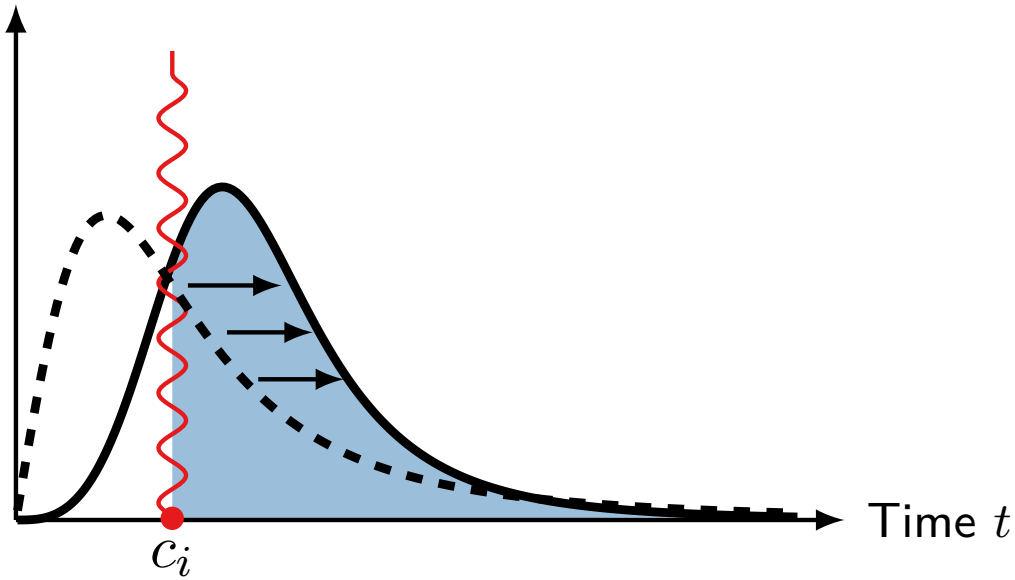
Exact time of event is known



$$\underset{\boldsymbol{\Theta}}{\operatorname{argmax}}\, P(T = y_i; \boldsymbol{\Theta} \mid \boldsymbol{x}_i) = f(y_i; \boldsymbol{\Theta} \mid \boldsymbol{x}_i)$$

Constructing the Likelihood Function

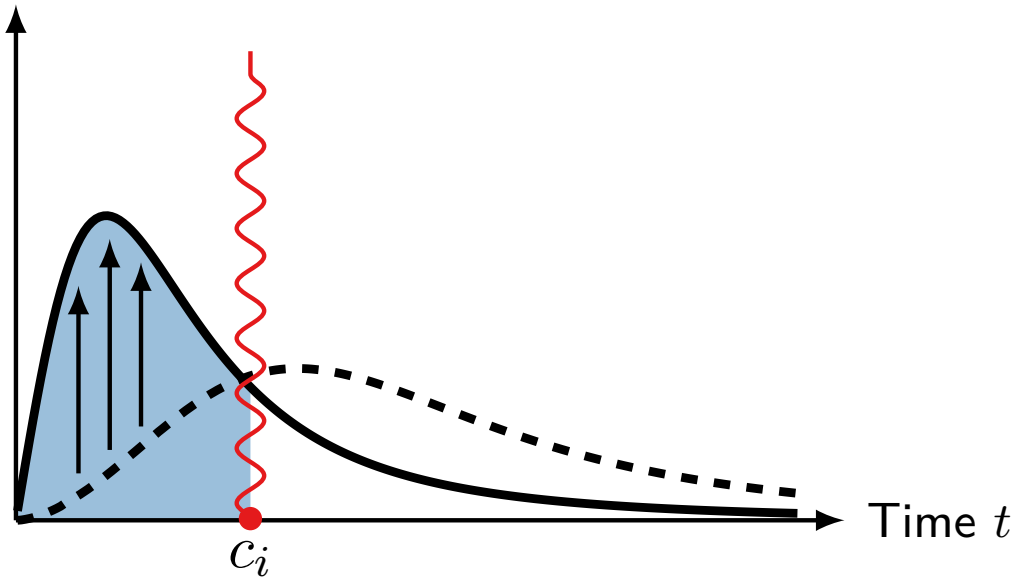
Time of event is right-censored



$$\underset{\boldsymbol{\Theta}}{\operatorname{argmax}}\, P(T > c_i; \boldsymbol{\Theta} \mid \boldsymbol{x}_i) = S(c_i; \boldsymbol{\Theta} \mid \boldsymbol{x}_i)$$

Constructing the Likelihood Function

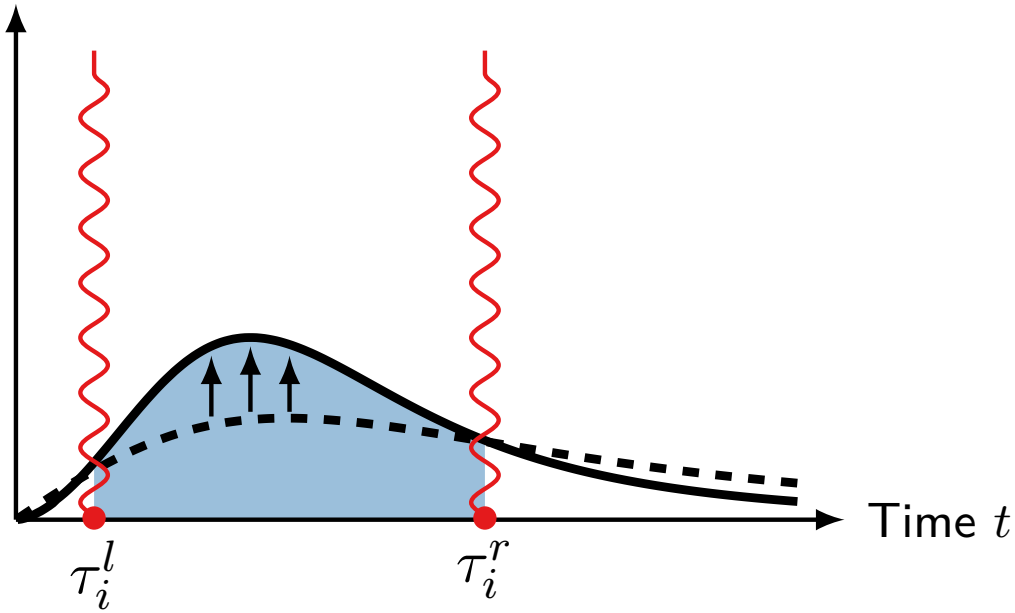
Time of event is left-censored



$$\underset{\boldsymbol{\Theta}}{\operatorname{argmax}}\, P(T \leq c_i; \boldsymbol{\Theta} \,|\, \boldsymbol{x}_i) = 1 - S(c_i; \boldsymbol{\Theta} \,|\, \boldsymbol{x}_i)$$

Constructing the Likelihood Function

Time of event is interval-censored



$$\underset{\boldsymbol{\Theta}}{\operatorname{argmax}}\, P(\tau_i^l < T \leq \tau_i^r; \boldsymbol{\Theta} \,|\, \boldsymbol{x}_i) = \int_{\tau_i^l}^{\tau_i^r} f(u; \boldsymbol{\Theta} \,|\, \boldsymbol{x}_i)\, du$$

$$= S(\tau_i^l; \boldsymbol{\Theta} \,|\, \boldsymbol{x}_i) - S(\tau_i^r; \boldsymbol{\Theta} \,|\, \boldsymbol{x}_i)$$

Constructing the Likelihood Function

Putting it all together

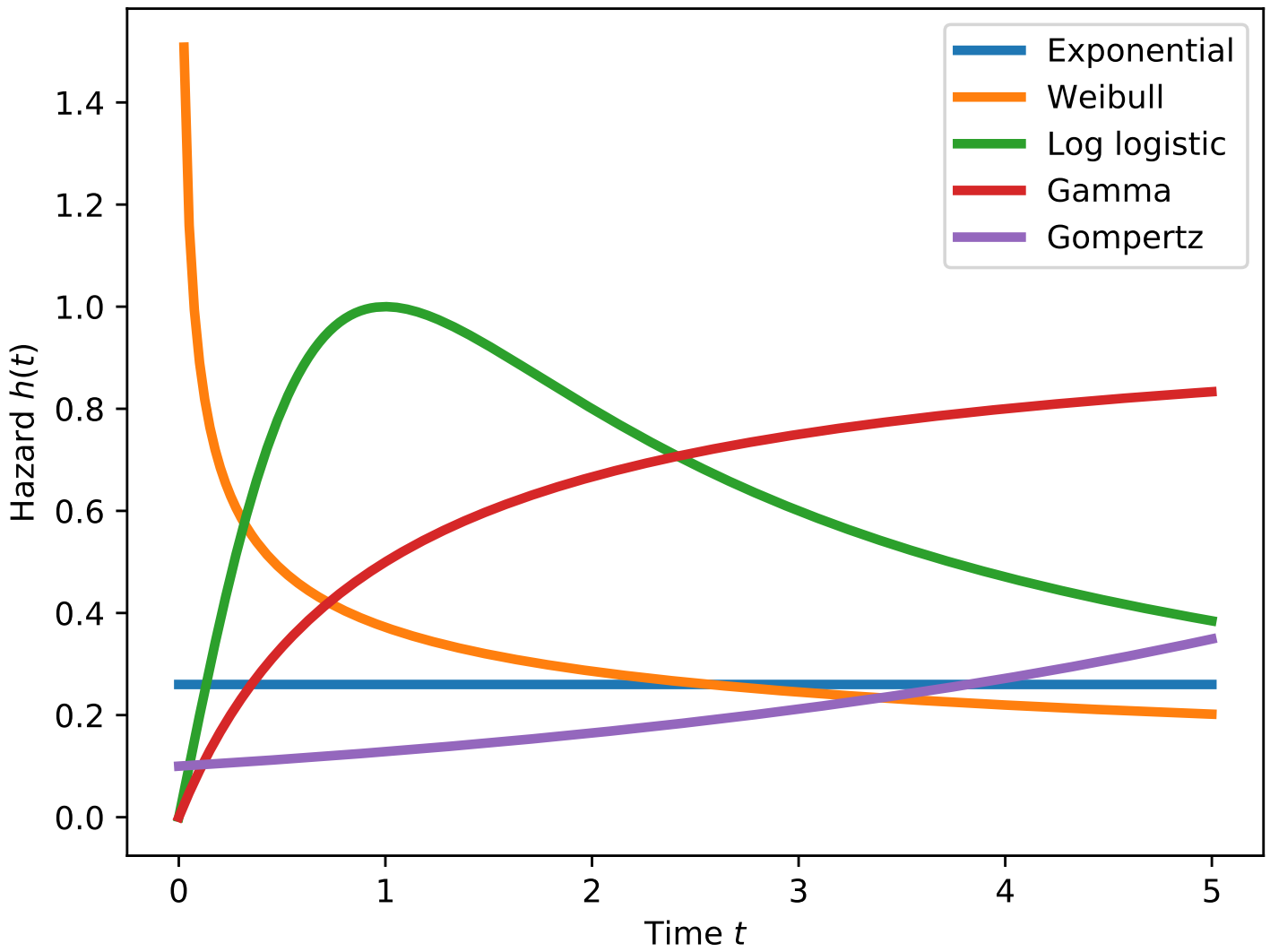
For training, we need to solve the optimization problem

$$\underset{\boldsymbol{\Theta}}{\operatorname{argmax}} \quad LL(\boldsymbol{\Theta})$$

where the likelihood function comprises all of the components

$$\begin{aligned} LL(\boldsymbol{\Theta}) = & \prod_{i \in \text{uncensored}} f(y_i; \boldsymbol{\Theta} \mid \boldsymbol{x}_i) \\ & \prod_{i \in \text{right-censored}} S(y_i; \boldsymbol{\Theta} \mid \boldsymbol{x}_i) \\ & \prod_{i \in \text{left-censored}} (1 - S(y_i; \boldsymbol{\Theta} \mid \boldsymbol{x}_i)) \\ & \prod_{i \in \text{interval-censored}} \Big(S(\tau_i^l; \boldsymbol{\Theta} \mid \boldsymbol{x}_i) - S(\tau_i^r; \boldsymbol{\Theta} \mid \boldsymbol{x}_i) \Big) \end{aligned}$$

Common Parametric Distributions



Outline

Semiparametric Survival Models

Parametric Models

- Distribution's parameters are data-dependent based on covariates.
- Work extremely well when survival times follow the chosen distribution.
- Can easily account for various censoring schemes.
- Inference is easy.

Semiparametric Survival Models

Parametric Models

- Distribution's parameters are data-dependent based on covariates.
- Work extremely well when survival times follow the chosen distribution.
- Can easily account for various censoring schemes.
- Inference is easy.

Semiparametric Models

- Often, we do not know what distribution we should choose.
- Split the model into 2 parts:
  1. part that models influence of covariates.
  2. part that models time.
- Usually only account for right-censoring.

Common Semiparametric Linear Models

- Cox's Proportional Hazards model (Cox PH)

$$h(t \mid \boldsymbol{x}) = h_0(t) \exp\left(\boldsymbol{x}^\top \boldsymbol{\beta}\right) \Leftrightarrow \frac{h(t \mid \boldsymbol{x})}{h_0(t)} = \exp\left(\boldsymbol{x}^\top \boldsymbol{\beta}\right)$$

Common Semiparametric Linear Models

- Cox's Proportional Hazards model (Cox PH)

$$h(t \mid \boldsymbol{x}) = h_0(t) \exp\left(\boldsymbol{x}^\top \boldsymbol{\beta}\right) \Leftrightarrow \frac{h(t \mid \boldsymbol{x})}{h_0(t)} = \exp\left(\boldsymbol{x}^\top \boldsymbol{\beta}\right)$$

- Accelerated Failure Time model (AFT)

$$h(t \mid \boldsymbol{x}) = h_0(t \exp(-\boldsymbol{x}^\top \boldsymbol{\beta})) \exp(-\boldsymbol{x}^\top \boldsymbol{\beta})$$

Common Semiparametric Linear Models

- Cox's Proportional Hazards model (Cox PH)

$$h(t \,|\, \boldsymbol{x}) = h_0(t) \exp\left(\boldsymbol{x}^\top \boldsymbol{\beta}\right) \Leftrightarrow \frac{h(t \,|\, \boldsymbol{x})}{h_0(t)} = \exp\left(\boldsymbol{x}^\top \boldsymbol{\beta}\right)$$

- Accelerated Failure Time model (AFT)

$$h(t \,|\, \boldsymbol{x}) = h_0(t \exp(-\boldsymbol{x}^\top \boldsymbol{\beta})) \exp(-\boldsymbol{x}^\top \boldsymbol{\beta})$$

- Proportional Odds model

$$\frac{P(T > t \,|\, \boldsymbol{x})}{P(T \leq t \,|\, \boldsymbol{x})} = \frac{1 - S(t \,|\, \boldsymbol{x})}{S(t \,|\, \boldsymbol{x})} = \frac{1 - S_0(t)}{S_0(t)} \exp\left(\boldsymbol{x}^\top \boldsymbol{\beta}\right)$$

Common Semiparametric Linear Models

- Cox's Proportional Hazards model (Cox PH)

$$h(t \,|\, \boldsymbol{x}) = h_0(t) \exp\left(\boldsymbol{x}^\top \boldsymbol{\beta}\right) \Leftrightarrow \frac{h(t \,|\, \boldsymbol{x})}{h_0(t)} = \exp\left(\boldsymbol{x}^\top \boldsymbol{\beta}\right)$$

- Accelerated Failure Time model (AFT)

$$h(t \,|\, \boldsymbol{x}) = h_0(t \exp(-\boldsymbol{x}^\top \boldsymbol{\beta})) \exp(-\boldsymbol{x}^\top \boldsymbol{\beta})$$

- Proportional Odds model

$$\frac{P(T > t \,|\, \boldsymbol{x})}{P(T \leq t \,|\, \boldsymbol{x})} = \frac{1 - S(t \,|\, \boldsymbol{x})}{S(t \,|\, \boldsymbol{x})} = \frac{1 - S_0(t)}{S_0(t)} \exp\left(\boldsymbol{x}^\top \boldsymbol{\beta}\right)$$

- **All models are multiplicative**.

Survival Data

Definition (Survival data)

Right-censored survival data consists of n triplets:

$\boldsymbol{x}_i \in \mathbb{R}^d$	a d-dimensional feature vector.
$y_i > 0$	observed time (time of event *or* time of censoring).
$\delta_i \in \{0; 1\}$	a boolean event indicator (right censoring).

Cox's Proportional Hazards model

- Cox PH is by far the most popular survival model.
- Coefficients can be interpreted in terms of *hazard ratio*:

$$\frac{h(t \mid x_1, \ldots, x_j, \ldots, x_p)}{h(t \mid x_1, \ldots, x_j + 1, \ldots, x_p)} = \exp\left(\beta_j\right).$$

- The hazard ratio is a constant *independent of time* (proportional hazards assumption).
- Optimization is easy: baseline hazard function $h_0(t)$ can be ignored until $\boldsymbol{\beta}$ has been estimated (*partial likelihood optimization*):

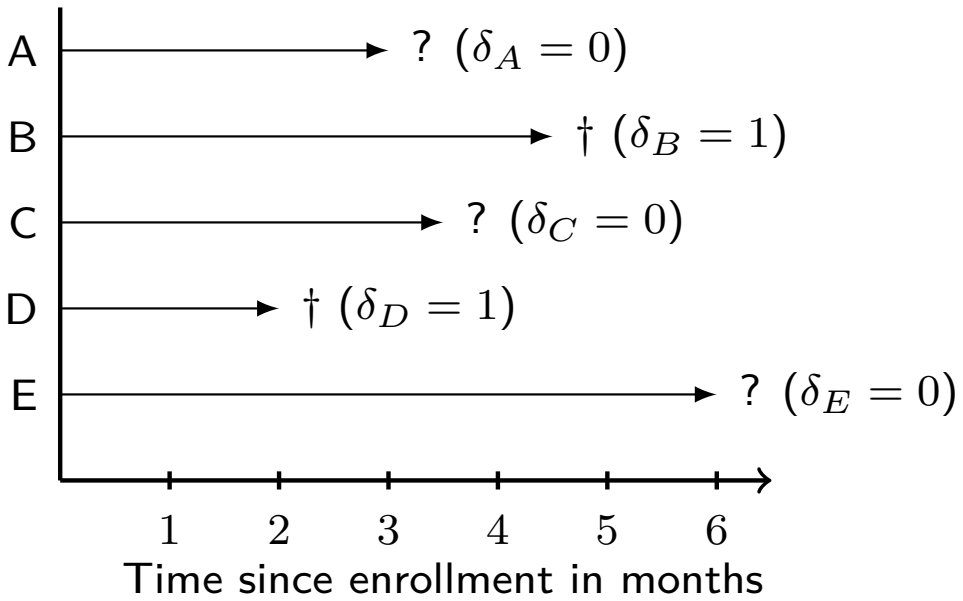
$$\underset{\boldsymbol{\beta}}{\operatorname{argmax}} \quad \sum_{i=1}^{n} \delta_i \left[\boldsymbol{x}_i^\top \boldsymbol{\beta} - \log \left(\sum_{j \in \mathcal{R}_i} \exp(\boldsymbol{x}_j^\top \boldsymbol{\beta}) \right) \right],$$

where $\mathcal{R}_i = \{j \mid y_j \geq t_i\}$ denotes the risk set.

Comparable Pairs

Definition (Set of comparable pairs)

$$\mathcal{P} = \{(i,j) \mid y_i > y_j \wedge \delta_j = 1\}_{i,j=1,\ldots,n}$$

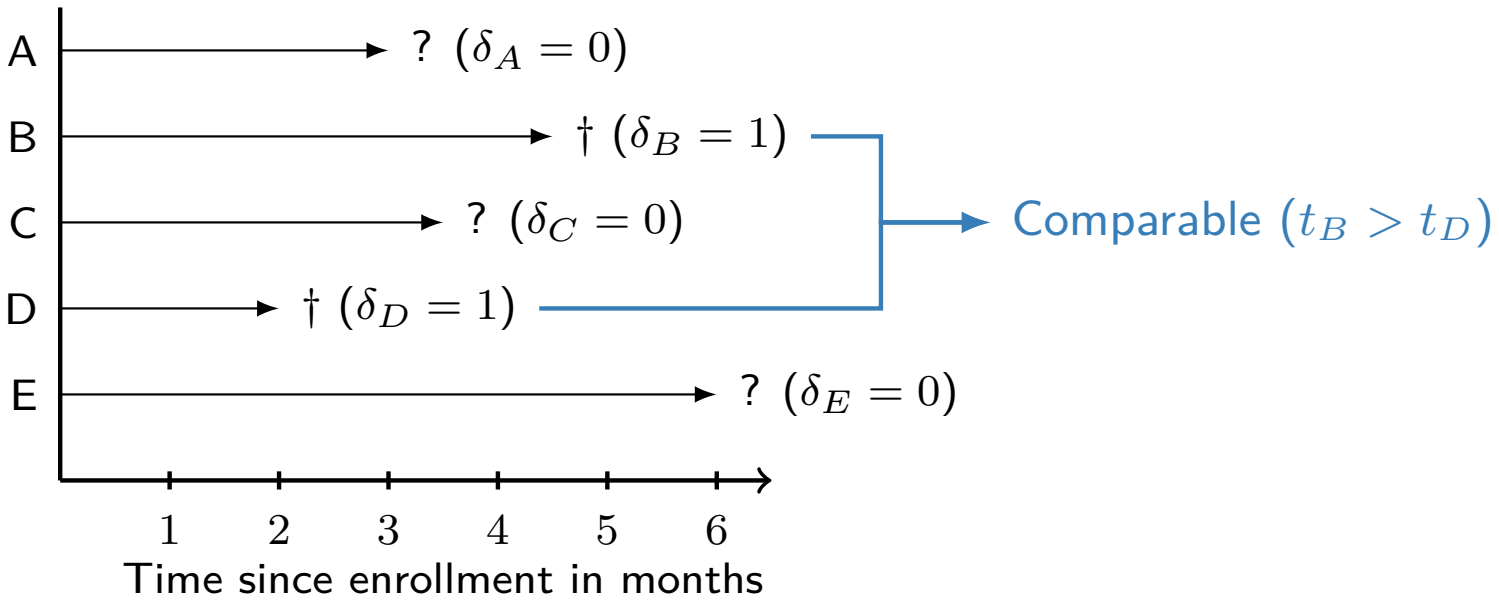


$$\mathcal{P} = \{\}$$

Comparable Pairs

Definition (Set of comparable pairs)

$$\mathcal{P} = \{(i,j) \mid y_i > y_j \wedge \delta_j = 1\}_{i,j=1,\ldots,n}$$

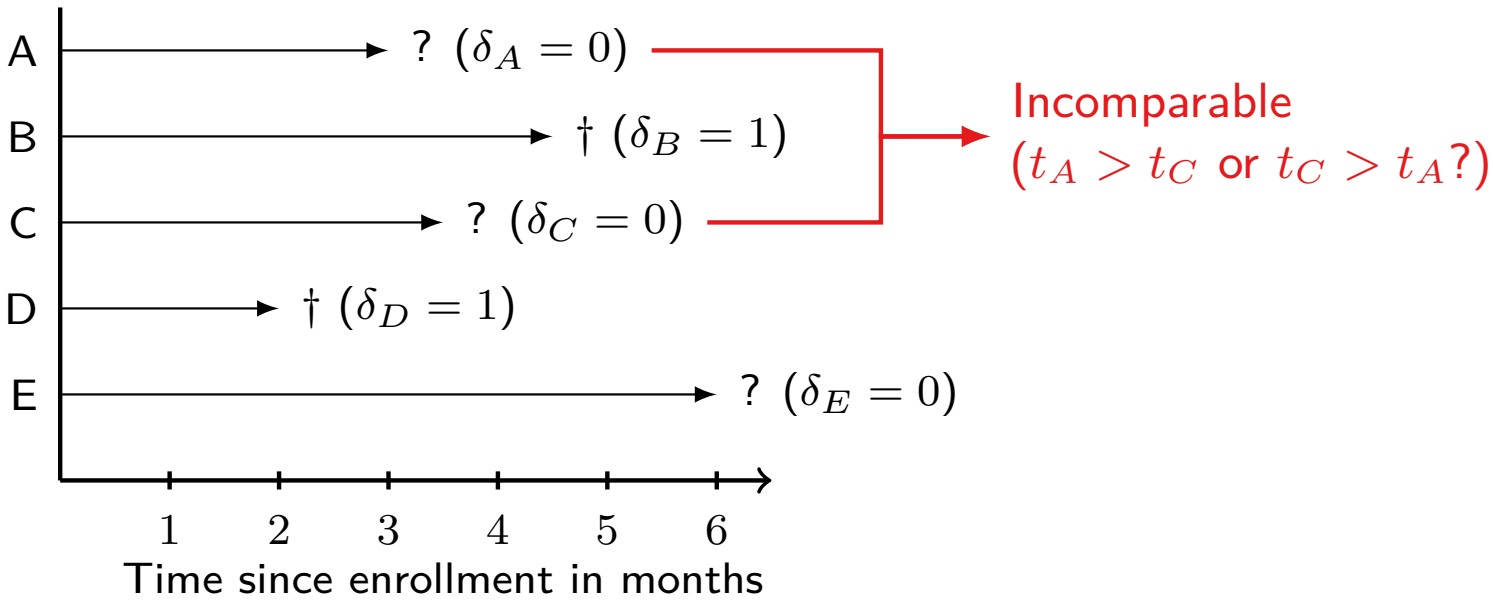


$$\mathcal{P} = \{(\mathbf{B}, \mathbf{D})\}$$

Comparable Pairs

Definition (Set of comparable pairs)

$$\mathcal{P} = \{(i,j) \mid y_i > y_j \wedge \delta_j = 1\}_{i,j=1,\ldots,n}$$

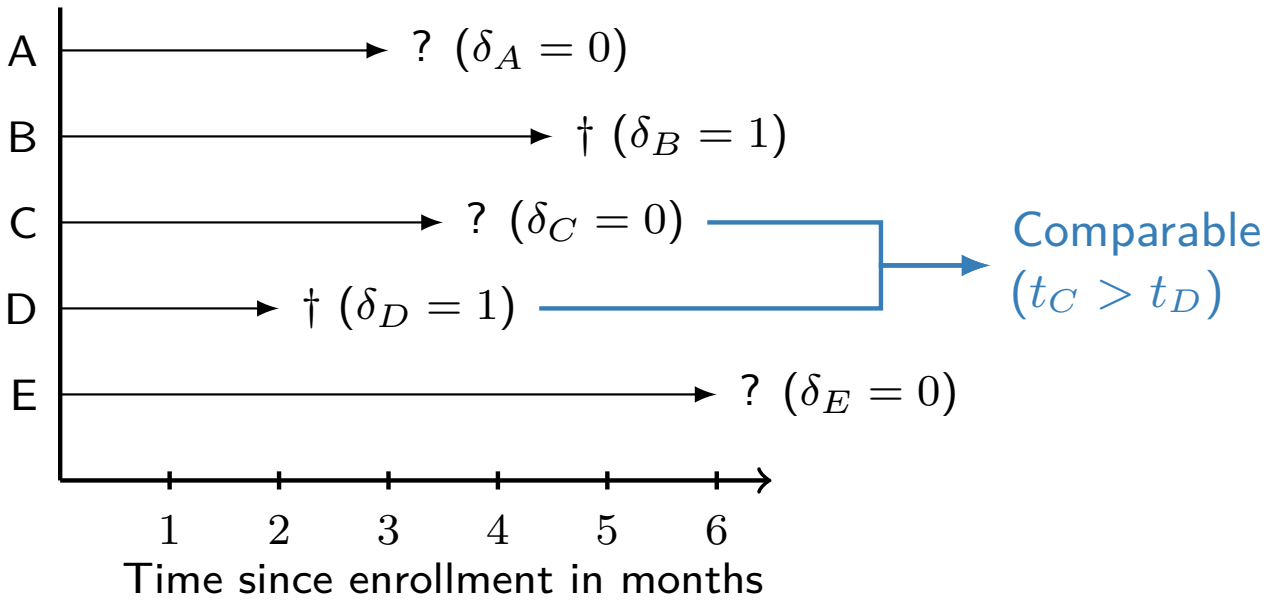


$$\mathcal{P} = \{(B,D)\}$$

Comparable Pairs

Definition (Set of comparable pairs)

$$\mathcal{P} = \{(i,j) \mid y_i > y_j \wedge \delta_j = 1\}_{i,j=1,\ldots,n}$$

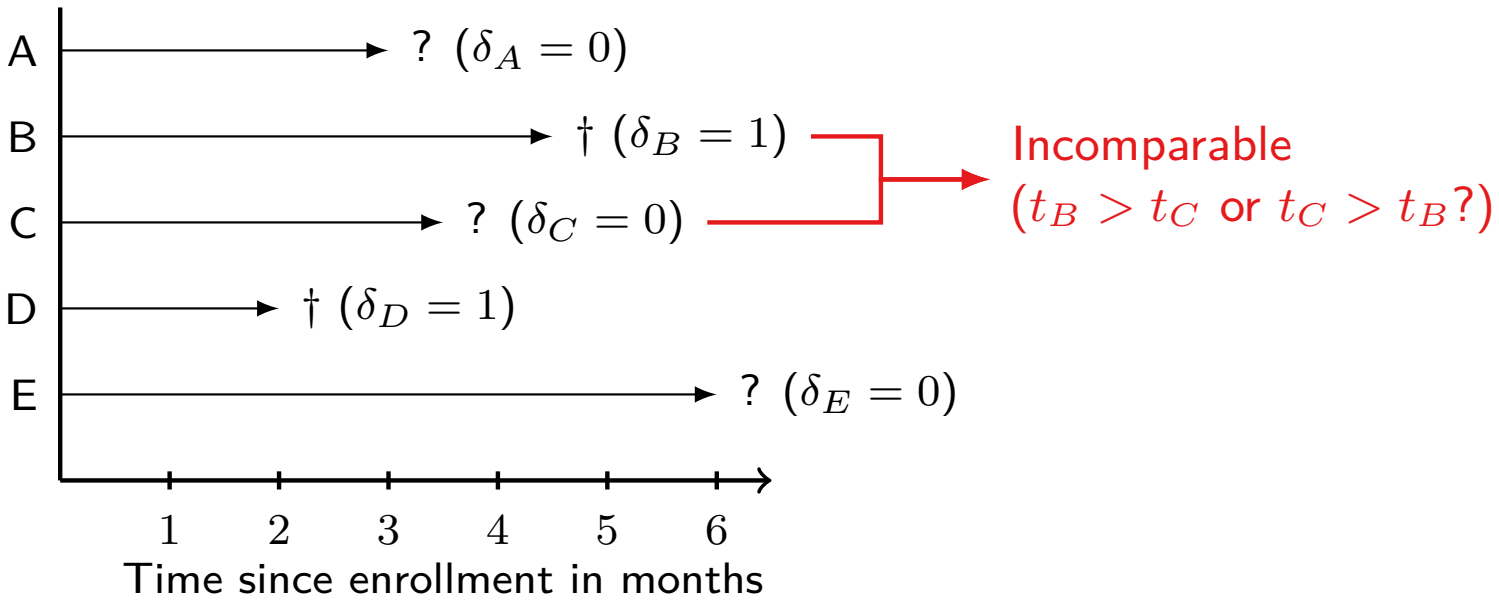


$$\mathcal{P} = \{(B,D), (\mathbf{C},\mathbf{D})\}$$

Comparable Pairs

Definition (Set of comparable pairs)

$$\mathcal{P} = \{(i,j) \mid y_i > y_j \wedge \delta_j = 1\}_{i,j=1,\ldots,n}$$

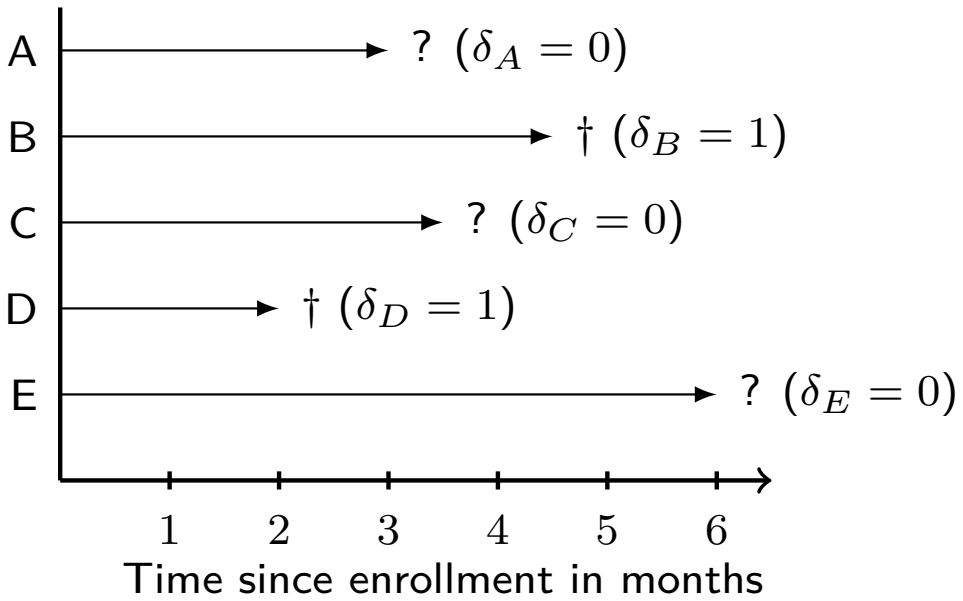


$$\mathcal{P} = \{(B,D),(C,D)\}$$

Comparable Pairs

Definition (Set of comparable pairs)

$$\mathcal{P} = \{(i,j) \mid y_i > y_j \wedge \delta_j = 1\}_{i,j=1,\ldots,n}$$



$$\mathcal{P} = \{(B,D),(C,D),(A,D),(E,D),(E,B)\}$$

Concordance Index

- The concordance index (c index) is a measure of rank correlation between predicted risk scores $\hat{f}(\boldsymbol{x})$ and observed time points y.
- It is the ratio of correctly ordered (concordant) pairs to comparable pairs:

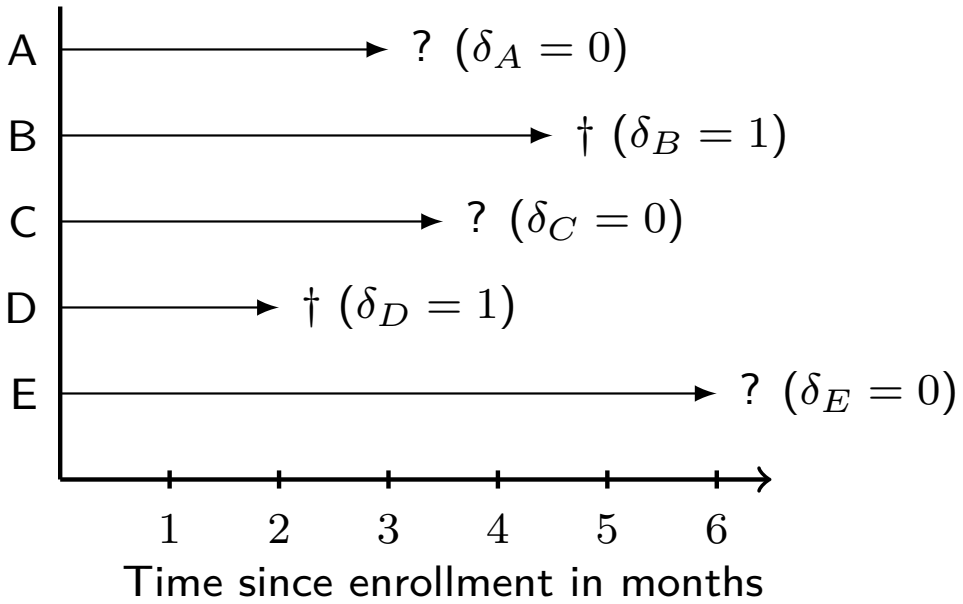
$$\hat{c}_{\text{Harrell}} = \frac{1}{|\mathcal{P}|} \sum_{(i,j) \in \mathcal{P}} I(\hat{f}(\boldsymbol{x}_i) < \hat{f}(\boldsymbol{x}_j)).$$

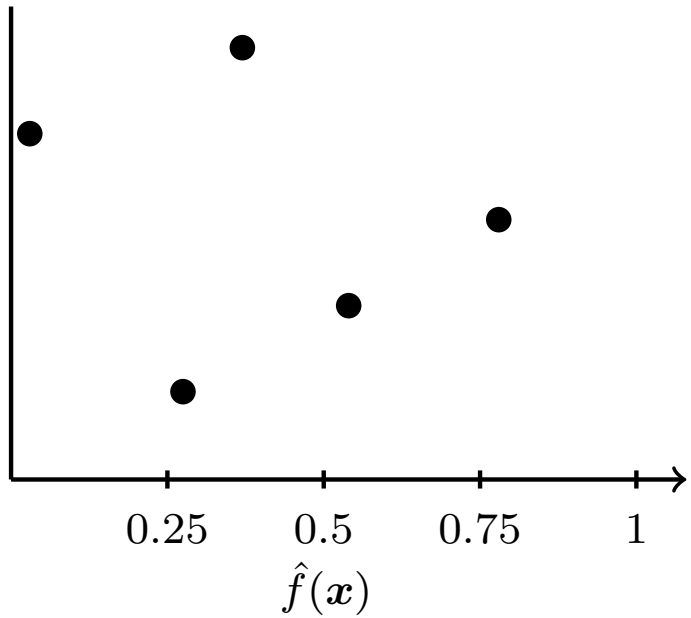
- A random model has c index 0.5, a perfect model 1.0
- Risk scores can be on any scale, only their relative ordering matters.
- c index is independent of time.

Concordance Index
Example

Definition (Concordance index)

$$\frac{1}{|\mathcal{P}|} \sum_{(i,j)\in\mathcal{P}} I(\hat{f}(\boldsymbol{x}_i) < \hat{f}(\boldsymbol{x}_j))$$



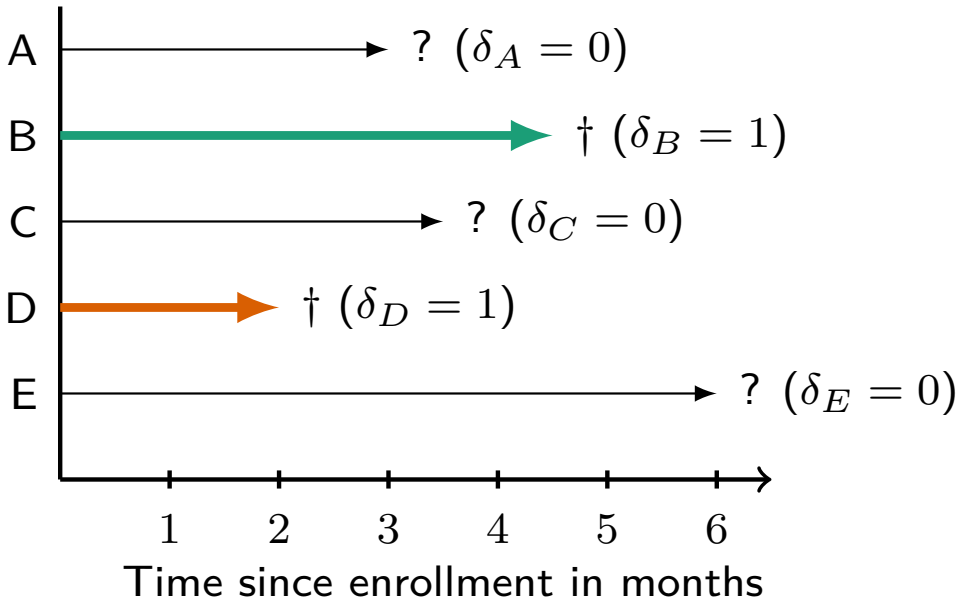


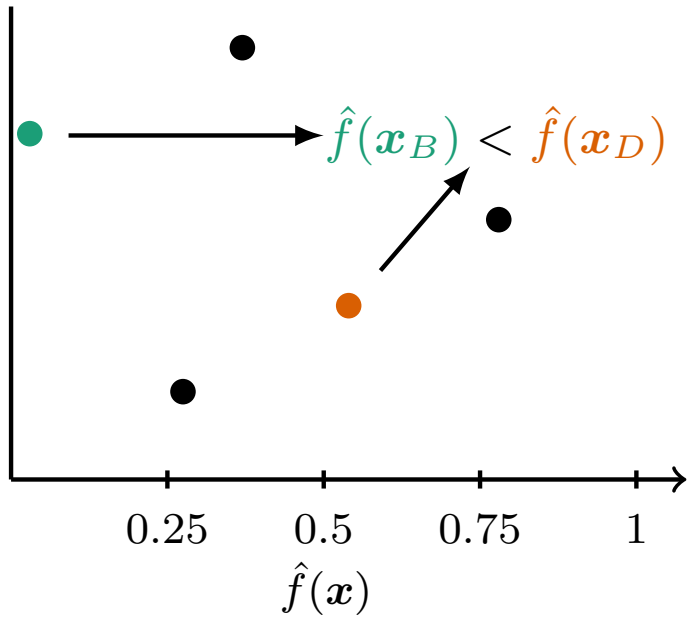
$$\mathcal{P} = \{(B, D), (C, D), (A, D), (E, D), (E, B)\} \Rightarrow \hat{c} = ?$$

Concordance Index
Example

Definition (Concordance index)

$$\frac{1}{|\mathcal{P}|} \sum_{(i,j)\in\mathcal{P}} I(\hat{f}(\boldsymbol{x}_i) < \hat{f}(\boldsymbol{x}_j))$$



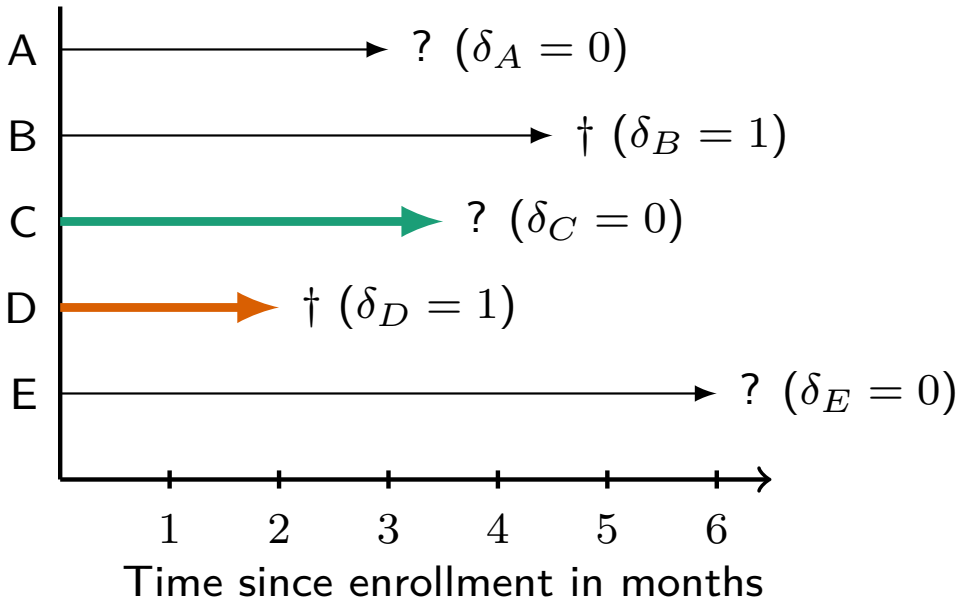


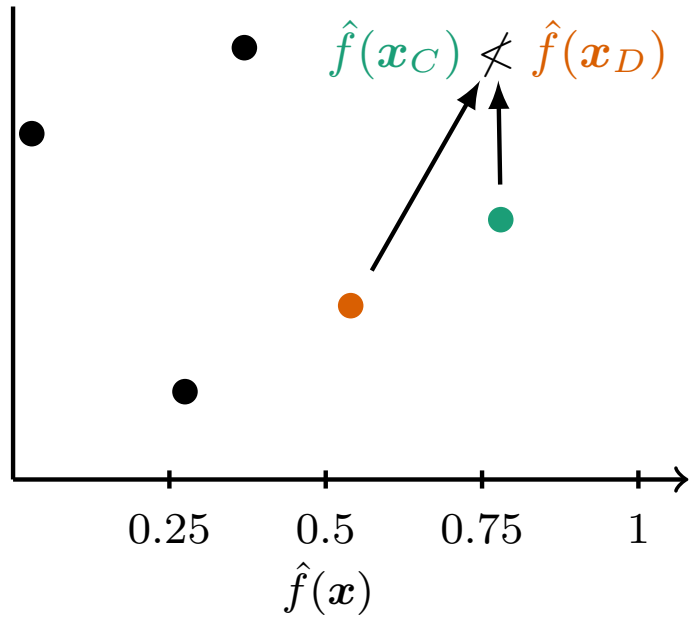
$$\mathcal{P} = \{(\mathbf{B}, \mathbf{D}), (C, D), (A, D), (E, D), (E, B)\} \Rightarrow \hat{c} = ?$$

Concordance Index
Example

Definition (Concordance index)

$$\frac{1}{|\mathcal{P}|} \sum_{(i,j)\in\mathcal{P}} I(\hat{f}(\boldsymbol{x}_i) < \hat{f}(\boldsymbol{x}_j))$$





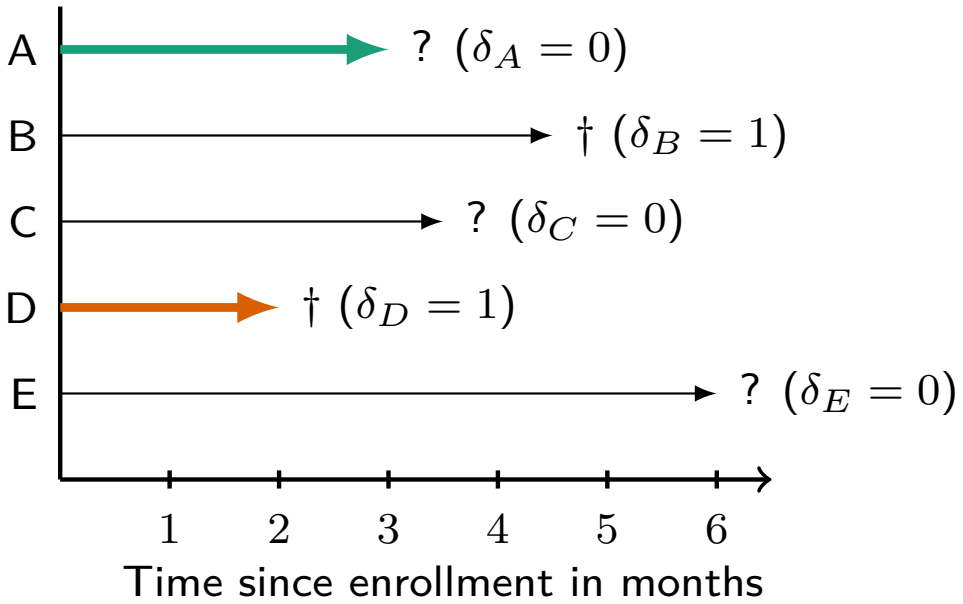
$$\mathcal{P} = \{(B, D), (\mathbf{C}, \mathbf{D}), (A, D), (E, D), (E, B)\} \Rightarrow \hat{c} = ?$$

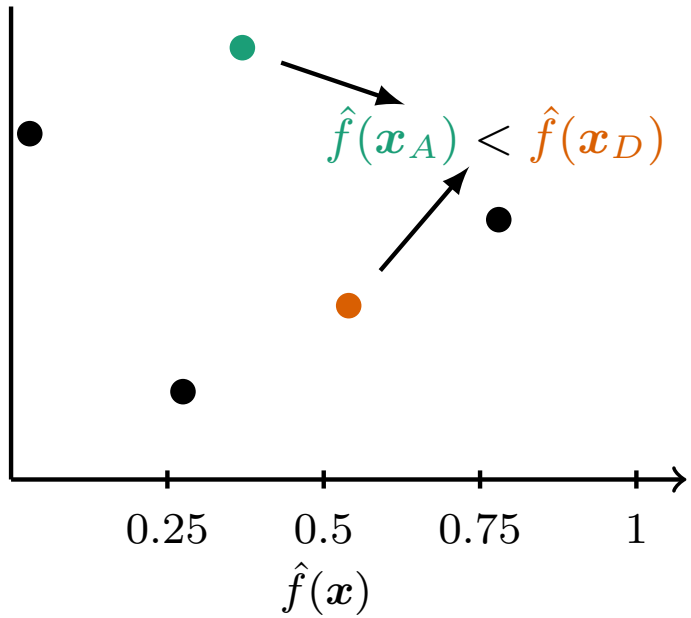
Concordance Index

Example

Definition (Concordance index)

$$\frac{1}{|\mathcal{P}|} \sum_{(i,j)\in\mathcal{P}} I(\hat{f}(\boldsymbol{x}_i) < \hat{f}(\boldsymbol{x}_j))$$





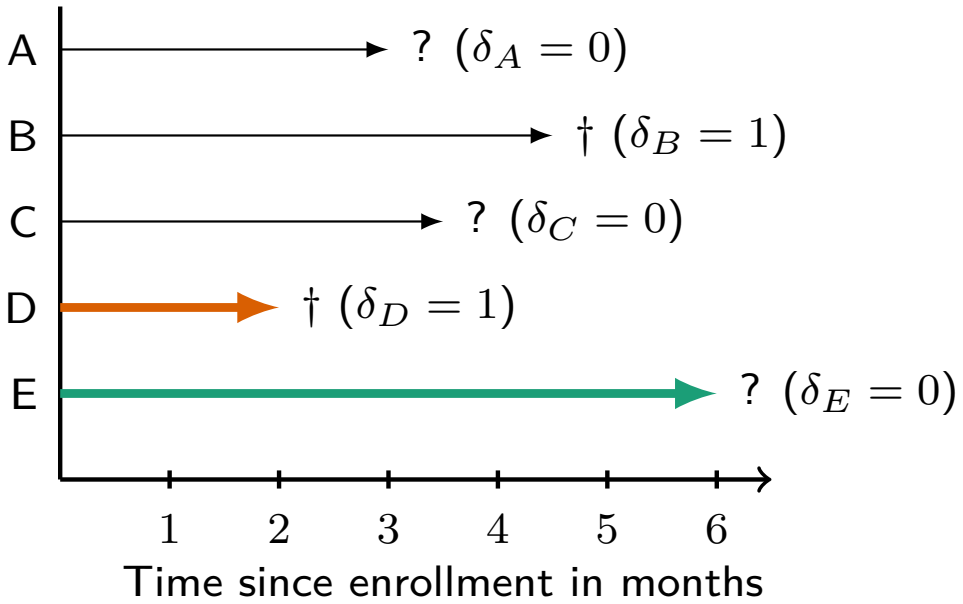
$$\mathcal{P} = \{(B, D), (C, D), (\mathbf{A}, \mathbf{D}), (E, D), (E, B)\} \Rightarrow \hat{c} = ?$$

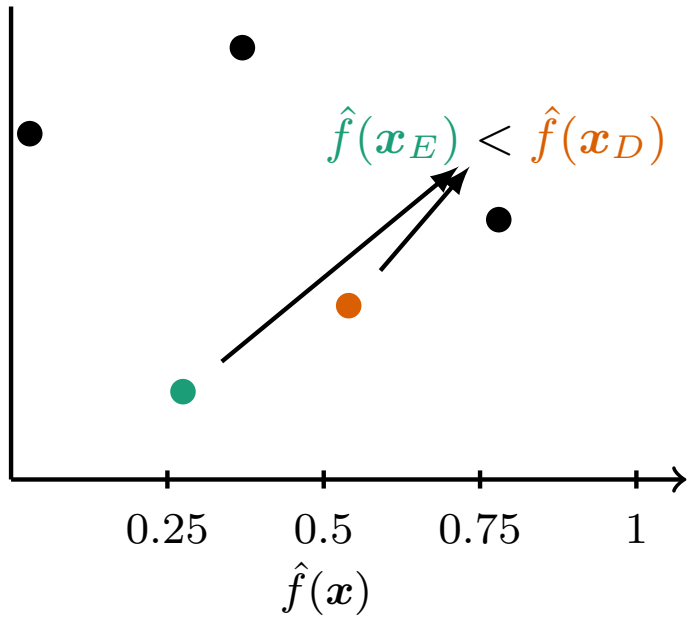
Concordance Index

Example

Definition (Concordance index)

$$\frac{1}{|\mathcal{P}|} \sum_{(i,j)\in\mathcal{P}} I(\hat{f}(\boldsymbol{x}_i) < \hat{f}(\boldsymbol{x}_j))$$





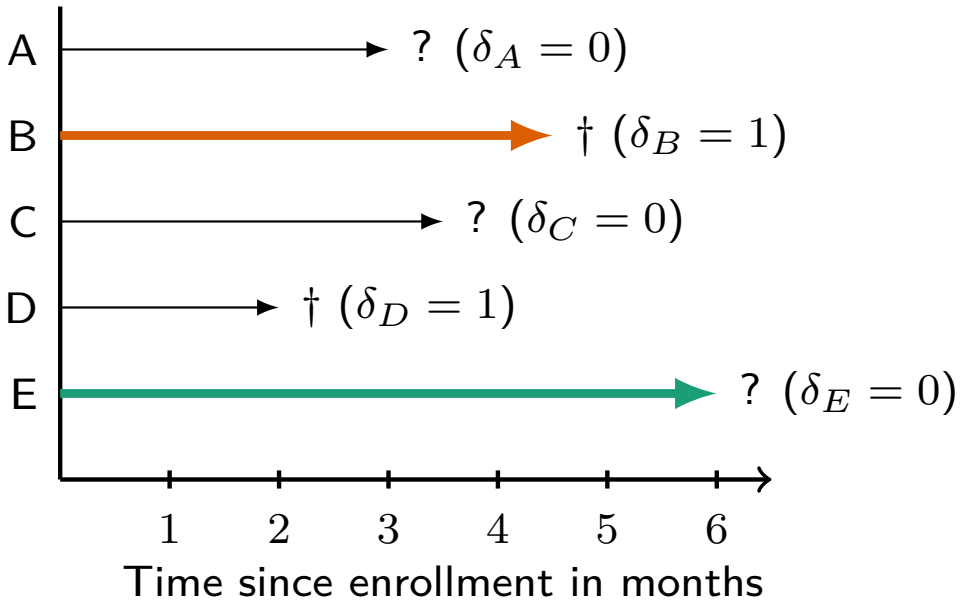
$$\mathcal{P} = \{(B, D), (C, D), (A, D), (\mathbf{E}, \mathbf{D}), (E, B)\} \Rightarrow \hat{c} = ?$$

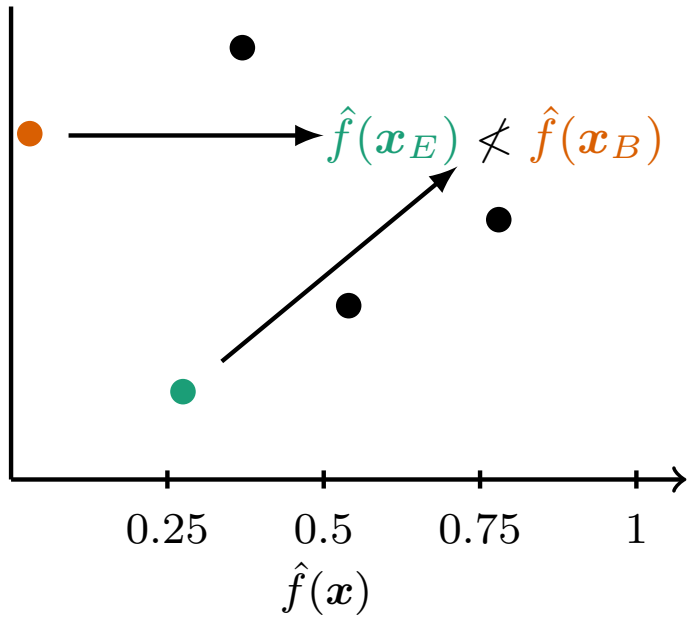
Concordance Index

Example

Definition (Concordance index)

$$\frac{1}{|\mathcal{P}|} \sum_{(i,j)\in\mathcal{P}} I(\hat{f}(\boldsymbol{x}_i) < \hat{f}(\boldsymbol{x}_j))$$





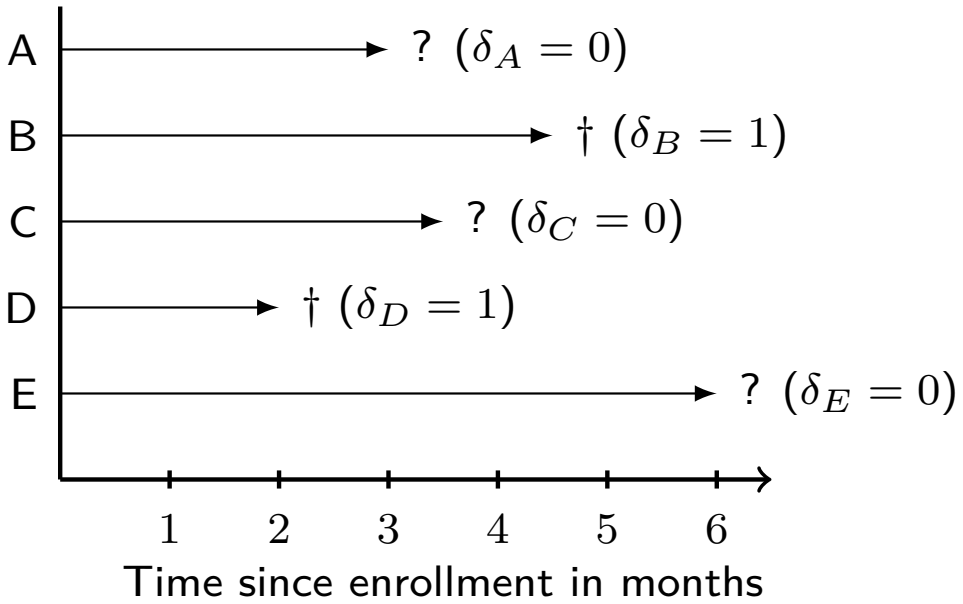
$$\mathcal{P} = \{(B, D), (C, D), (A, D), (E, D), (\mathbf{E}, \mathbf{B})\} \Rightarrow \hat{c} = ?$$

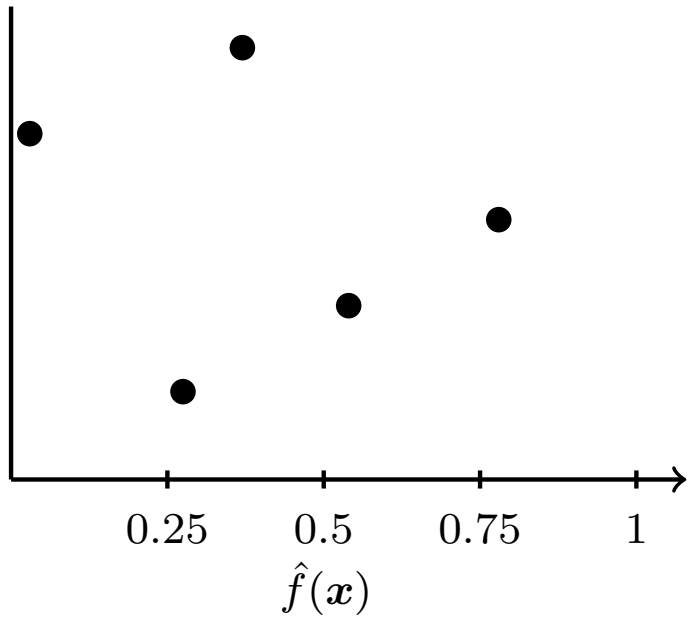
Concordance Index

Example

Definition (Concordance index)

$$\frac{1}{|\mathcal{P}|} \sum_{(i,j)\in\mathcal{P}} I(\hat{f}(\boldsymbol{x}_i) < \hat{f}(\boldsymbol{x}_j))$$





$$\mathcal{P} = \{(B,D),(C,D),(A,D),(E,D),(E,B)\} \Rightarrow \hat{c} = 3/5$$

Outline

Gradient Boosting

- Take a linear model and replace the linear predictor $\boldsymbol{x}_i^\top \boldsymbol{\beta}$ with an unknown, more complex function $f(\boldsymbol{x})$.
- We can model $f(\boldsymbol{x})$ as an additive model by performing gradient descent in function space (gradient boosting).
- Loss function:
  - Cox PH (Binder and Schumacher, 2008; Li and Luan, 2005; Ridgeway, 1999)
  - AFT (Hothorn et al., 2006; Schmid and Hothorn, 2008; Wang and Wang, 2010)
  - c index (Benner, 2002; Mayr and Schmid, 2014)
- Base learner:
  - regression tree (Breiman et al., 1984)
  - componentwise least squares (Bühlmann and Yu, 2003)

Support Vector Machine

- We can treat survival analysis as ranking problem (Van Belle et al., 2008).
- We want to optimize a smooth approximation of the c index:

$$\begin{aligned} \min_{\boldsymbol{w}} \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 + \gamma \sum_{(i,j)\in\mathcal{P}} \xi_{ij} \\ \text{subject to} \quad & \boldsymbol{w}^\top \boldsymbol{x}_i - \boldsymbol{w}^\top \boldsymbol{x}_j \geq 1 - \xi_{ij}, \quad \forall (i,j) \in \mathcal{P}, \\ & \xi_{ij} \geq 0, \qquad\qquad\qquad\qquad \forall (i,j) \in \mathcal{P} \end{aligned}$$

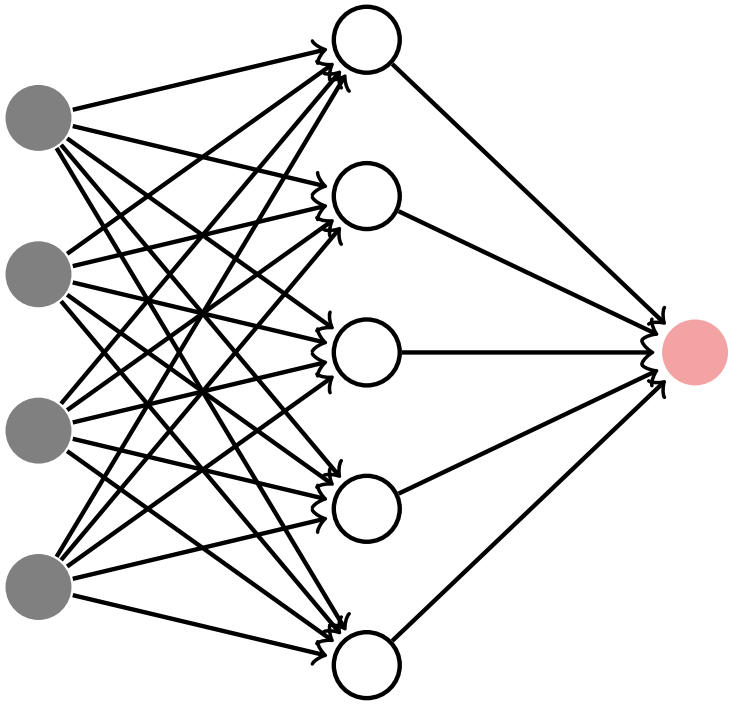
- Optimization algorithm needs to be clever to avoid dependency on kernel matrix of size $O(|\mathcal{P}|^2) = O(n^4)$ (Pölsterl et al., 2015, 2016).
- Alternative models: regression with non-symmetric loss (Khan and Zubek, 2008; Shivaswamy et al., 2007), quantile regression (Eleuteri, 2008; Eleuteri and Taktak, 2012).

Neural Networks

- Faraggi and Simon (1995) proposes a multi-layer perceptron that extends the Cox PH model.
- Biganzoli et al. (1998) and Liestøl et al. (1994) propose the *Partial Logistic Artificial Neural Network* that considers survival times grouped into mutually exclusive intervals and a loss based on a piecewise exponential model.

Loss by Faraggi and Simon

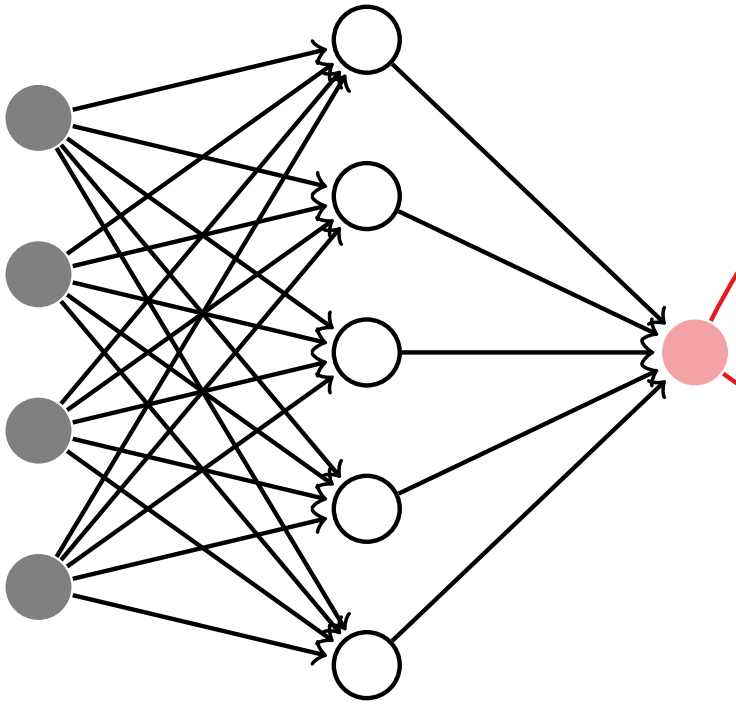
Input layer | Hidden layer | Output layer | Cox PH loss



$$\underset{\boldsymbol{\beta}}{\operatorname{argmax}} \quad \sum_{i=1}^{n} \delta_i \left[\boldsymbol{x}_i^\top \boldsymbol{\beta} - \log \left(\sum_{j \in \mathcal{R}_i} \exp(\boldsymbol{x}_j^\top \boldsymbol{\beta}) \right) \right],$$

Loss by Faraggi and Simon

Input layer | Hidden layer | Output layer | Cox PH loss



$$\underset{\boldsymbol{\Theta}}{\operatorname{argmin}} \sum_{i=1}^{n} \delta_i \left[o(\boldsymbol{x}_i \mid \boldsymbol{\Theta}) - \log \left(\sum_{j \in \mathcal{R}_i} \exp\left(o(\boldsymbol{x}_j \mid \boldsymbol{\Theta}) \right) \right) \right]$$

Loss by Faraggi and Simon

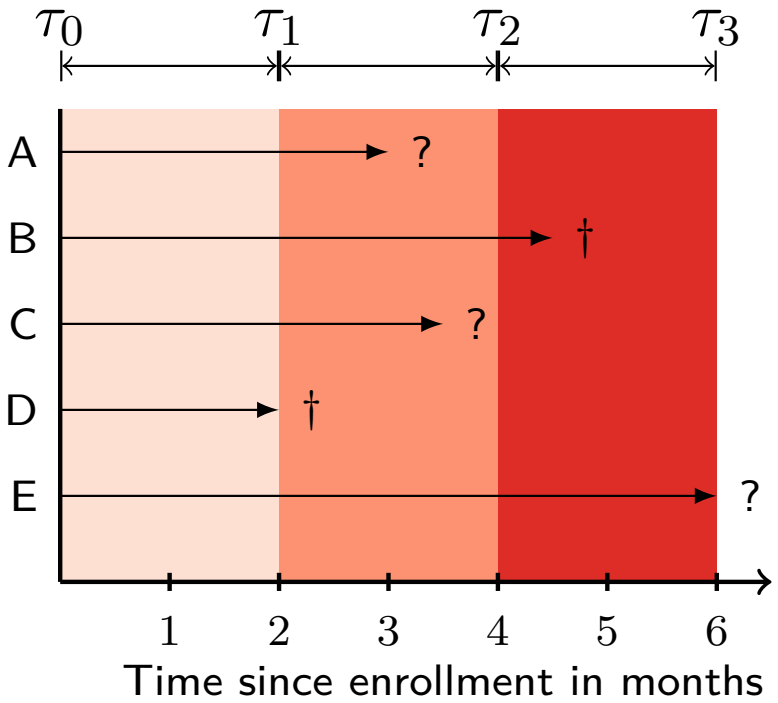
Problems

- Samples need to be sorted by observed time y_i due to sum over $\mathcal{R}_i = \{j \mid y_j \geq t_i\}$.
- Batch size needs to be large, otherwise gradient is very noisy.
- Only considers **time-invariant features** (proportional hazards assumption).

Partial Logistic ANN

Biganzoli et al. (1998) and Liestøl et al. (1994)

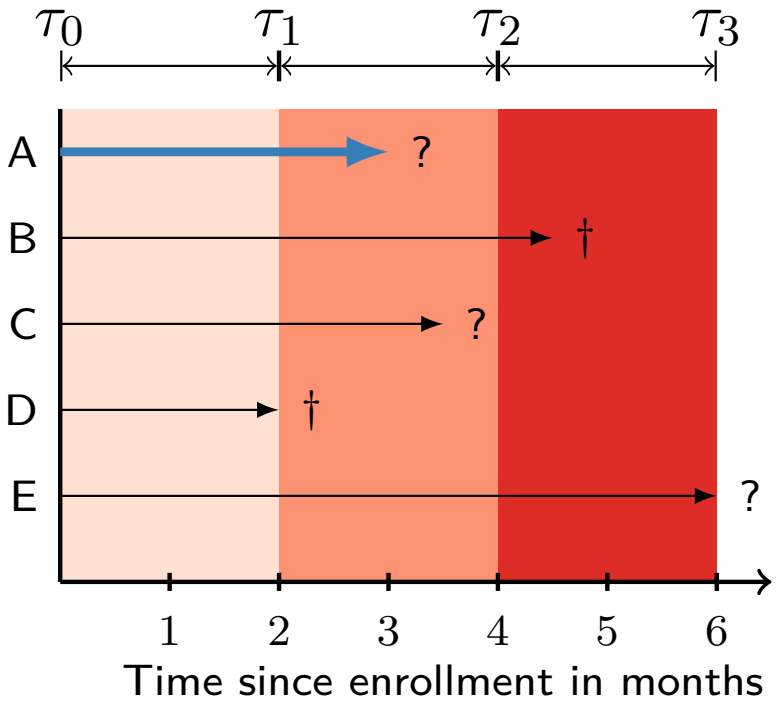
The *Partial Logistic Artificial Neural Network* considers survival times grouped into mutually exclusive intervals.



Partial Logistic ANN

Biganzoli et al. (1998) and Liestøl et al. (1994)

The *Partial Logistic Artificial Neural Network* considers survival times grouped into mutually exclusive intervals.



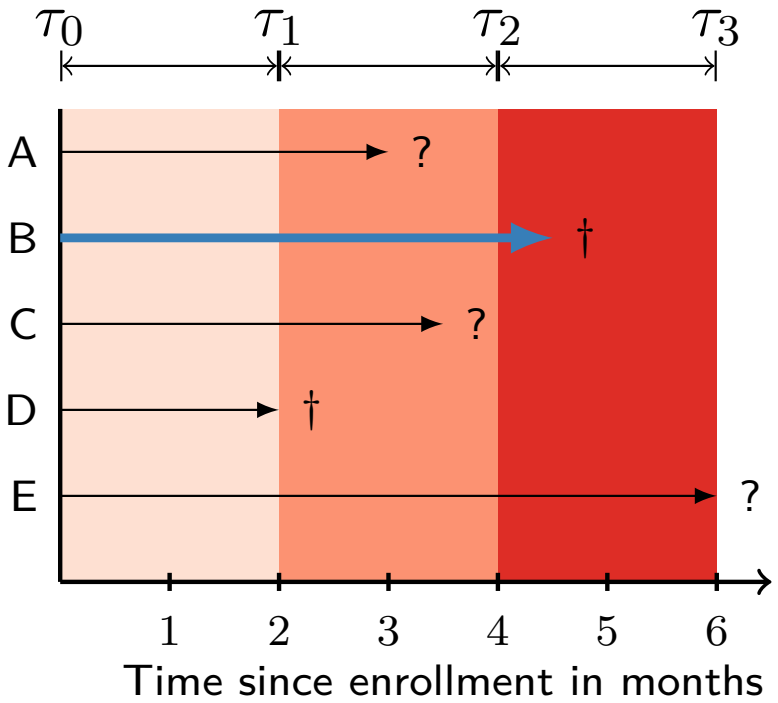
Event in k-th interval?
$\delta_{A1} = 0, \quad \delta_{A2} = 0, \quad \delta_{A3} = 0$

Time spent in k-th interval:
$\tilde{y}_{A1} = 2, \quad \tilde{y}_{A2} = 1, \quad \tilde{y}_{A3} = 0$

Partial Logistic ANN

Biganzoli et al. (1998) and Liestøl et al. (1994)

The *Partial Logistic Artificial Neural Network* considers survival times grouped into mutually exclusive intervals.



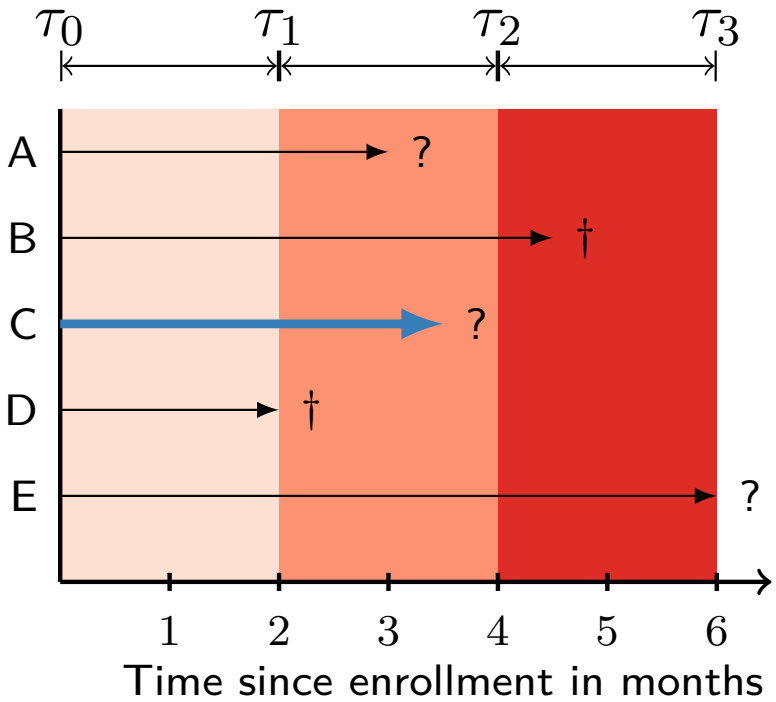
Event in k-th interval?
$\delta_{B1} = 0, \quad \delta_{B2} = 0, \quad \delta_{B3} = 1$

Time spent in k-th interval:
$\tilde{y}_{B1} = 2, \quad \tilde{y}_{B2} = 2, \quad \tilde{y}_{B3} = 0.5$

Partial Logistic ANN

Biganzoli et al. (1998) and Liestøl et al. (1994)

The *Partial Logistic Artificial Neural Network* considers survival times grouped into mutually exclusive intervals.



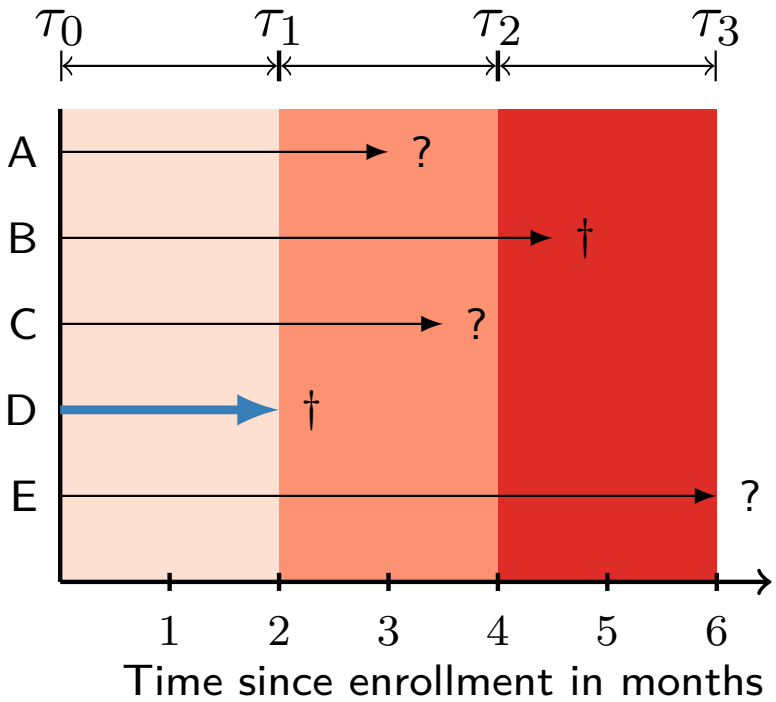
Event in k-th interval?
$\delta_{C1} = 0, \quad \delta_{C2} = 0, \quad \delta_{C3} = 0,$

Time spent in k-th interval:
$\tilde{y}_{C1} = 2, \quad \tilde{y}_{C2} = 1.5, \quad \tilde{y}_{C3} = 0$

Partial Logistic ANN

Biganzoli et al. (1998) and Liestøl et al. (1994)

The *Partial Logistic Artificial Neural Network* considers survival times grouped into mutually exclusive intervals.



Event in k-th interval?
$\delta_{D1} = 1, \quad \delta_{D2} = 0, \quad \delta_{D3} = 0,$

Time spent in k-th interval:
$\tilde{y}_{D1} = 2, \quad \tilde{y}_{D2} = 0, \quad \tilde{y}_{D3} = 0$

Partial Logistic ANN

Biganzoli et al. (1998) and Liestøl et al. (1994)

- A *piecewise exponential model* has a constant hazard rate $\lambda_l > 0$ in the l-th interval and has survival function

$$S(t) = \exp(-\lambda_l(t - \tau_{l-1})) \prod_{k=1}^{l-1} \exp(-\lambda_k(\tau_k - \tau_{k-1}))$$

Partial Logistic ANN

Biganzoli et al. (1998) and Liestøl et al. (1994)

- A *piecewise exponential model* has a constant hazard rate $\lambda_l > 0$ in the l-th interval and has survival function

$$S(t) = \exp(-\lambda_l(t - \tau_{l-1})) \prod_{k=1}^{l-1} \exp(-\lambda_k(\tau_k - \tau_{k-1}))$$

- Substituting the definition into the log-likelihood function of a parametric model, we obtain

$$\underset{\{\lambda_1,\ldots,\lambda_L\}}{\operatorname{argmax}} \quad \sum_{i=1}^{n} \sum_{k=1}^{L} [\delta_{ik} \log(\lambda_k) - \lambda_k \tilde{y}_{ik}]$$

Partial Logistic ANN

Biganzoli et al. (1998) and Liestøl et al. (1994)

- A *piecewise exponential model* has a constant hazard rate $\lambda_l > 0$ in the l-th interval and has survival function

$$S(t) = \exp(-\lambda_l(t - \tau_{l-1})) \prod_{k=1}^{l-1} \exp(-\lambda_k(\tau_k - \tau_{k-1}))$$

- Substituting the definition into the log-likelihood function of a parametric model, we obtain

$$\underset{\{\lambda_1,\ldots,\lambda_L\}}{\operatorname{argmax}} \quad \sum_{i=1}^{n} \sum_{k=1}^{L} \left[\delta_{ik} \log(\lambda_k) - \lambda_k \tilde{y}_{ik}\right]$$

- Finally, the parameters λ_k are modeled by a neural network $o(\boldsymbol{x}_i \,|\, \boldsymbol{\Theta})$ conditional on feature vectors $\boldsymbol{x}_i$ as

$$\lambda_k(\boldsymbol{x}_i) = \exp(\underbrace{\log \lambda_{0k}}_{\substack{\text{baseline} \\ =\text{bias term}}} + \boldsymbol{w}^\top o(\boldsymbol{x}_i \,|\, \boldsymbol{\Theta}))$$

Outline

Literature Survey

- I could find 24 papers using deep learning[1] techniques with a loss accounting for censored event times.
- 10 use the Cox PH loss of Faraggi and Simon (1995).
- 18 have been applied to medical data.
  - 8 to medical images (6 of which are on histopathology images).
  - 4 to genomic data.
  - The remaining use tabular clinical data or EHR.

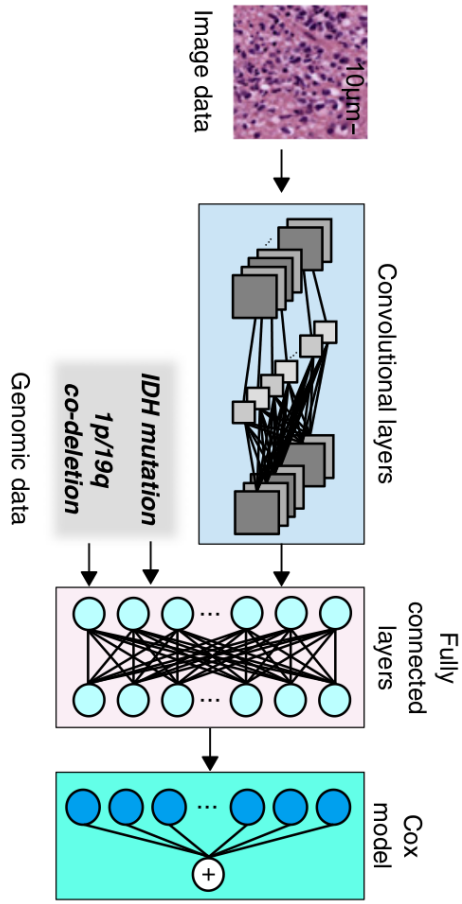
[1] excluding work using Deep Gaussian Processes

Example 1: Histology + Genomics

Mobadersany et al. (2018)

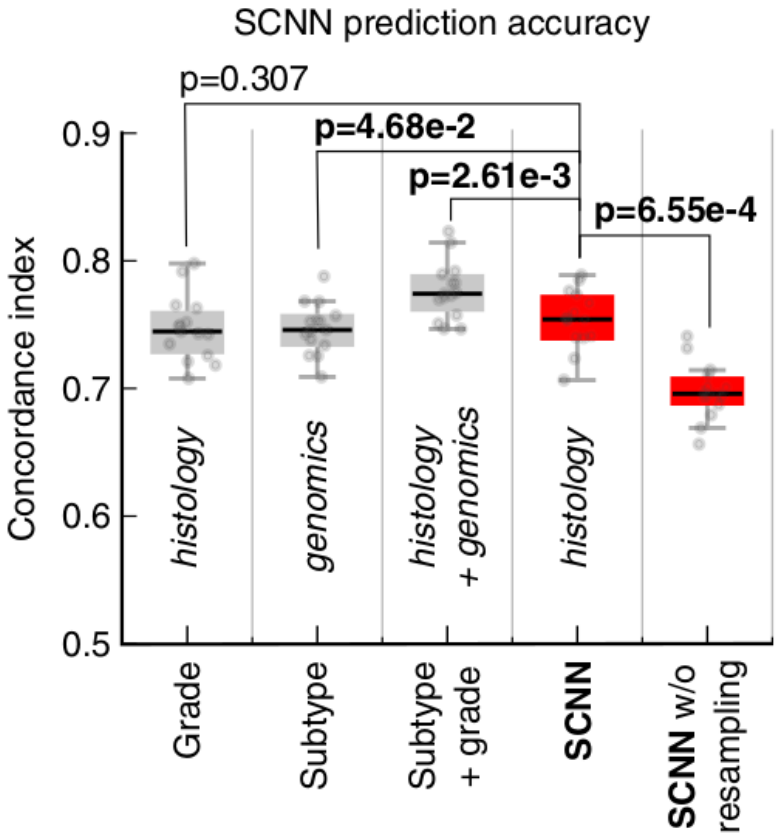
Mobadersany et al. (2018), "Predicting cancer outcomes from histology and genomics using convolutional networks", PNAS.

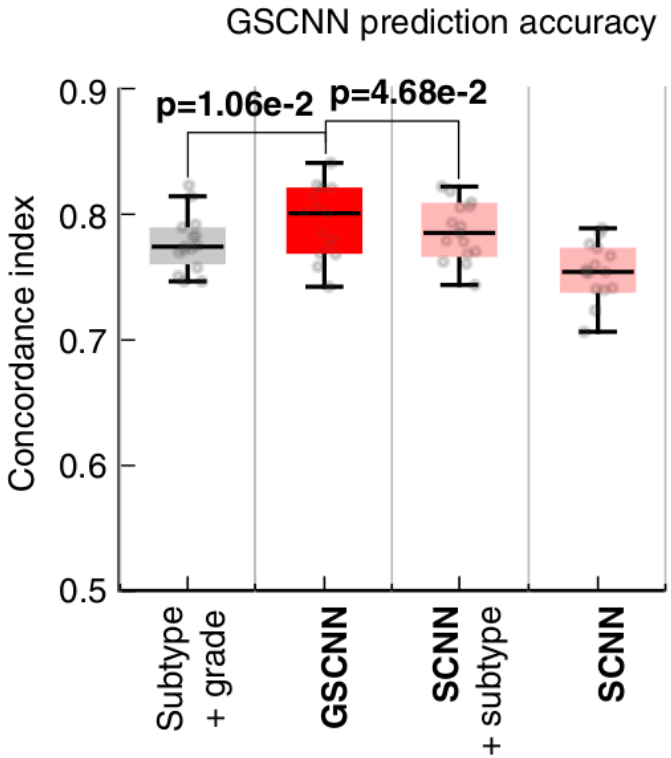
- Objective: Survival prediction of patients with diffuse gliomas.
- Network integrates information from both histology images and genomic biomarkers.
- Uses a modified VGG-19 architecture with loss of Faraggi and Simon.
- Training and testing use random sampling of patches from region of interest.
- Genomic markers (IDH mutation status and 1p/19q co-deletion) are integrated as input to shared FC layer.



Example 1: Histology + Genomics

Mobadersany et al. (2018)





Example 2: Web User Return Time

Grob et al. (2018)

Grob et al. (2018), "A RNN Survival Model: Predicting Web User Return Time", ECML-PKDD.

- Objective: Predict the return times of users to a website.
- Each user has a sequence of previous sessions.
- Each session is has a start time and a set of features.
- Time T is defined as the period between the end of a session and the beginning of the succeeding session.
- The hazard function up to the j-th session $h_j(t)$ is modeled as a recurrent marked temporal point process:

$$h_j(t) = \exp\left(\underbrace{\boldsymbol{v}^{(t)}\boldsymbol{h}_j}_{\text{past}} + \underbrace{w(t-t_j)}_{\text{temporal}} + \underbrace{b^{(t)}}_{\text{bias}}\right)$$

Example 2: Web User Return Time

Grob et al. (2018)

	Baseline	Cox PH	RNN-MSE	RNN-SM
RMSE (days)	43.25	49.99	**28.69**	59.99
Concordance	0.500	**0.816**	0.706	0.739
Non-returning AUC	0.743	0.793	0.763	**0.796**
Non-returning recall	0.000	0.246	0.000	**0.538**

Outline

Conclusion

- Time-to-event analysis is applicable across a wide range of domains.
- It is a well studied topic in statistics.
- Most classical machine learning models have been modified for time-to-event data.
- It is slowly being adapted by the deep learning community, although most of the approaches are rather naive.
- Cox PH model is surprisingly hard to beat.

References I

Benner, A. (2002). "Application of "Aggregated Classifiers" in Survival Time Studies". In: *Proc. in Computational Statistics: COMPSTAT*. Ed. by W. Härdle and B. Rönz, pp. 171–176.

Biganzoli, E., P. Boracchi, L. Mariani, and E. Marubini (May 1998). "Feed forward neural networks for the analysis of censored survival data: a partial logistic regression approach". In: *Stat. Med.* 17.10, pp. 1169–1186. ISSN: 0277-6715.

Binder, H. and M. Schumacher (2008). "Allowing for mandatory covariates in boosting estimation of sparse high-dimensional survival models". In: *BMC Bioinformatics* 9, p. 14.

Breiman, L., J. H. Friedman, C. J. Stone, and R. A. Ohlsen (1984). *Classification and Regression Trees*. Wadsworth International Group.

Bühlmann, P. and B. Yu (2003). "Boosting With the L_2 Loss". In: *J Am Stat Assoc* 98.462, pp. 324–339.

Eleuteri, A. (2008). "Support vector survival regression". In: *4th IET International Conference on Advances in Medical, Signal and Information Processing*, pp. 1–4.

References II

Eleuteri, A. and A. F. Taktak (2012). "Support Vector Machines for Survival Regression". In: *Computational Intelligence Methods for Bioinformatics and Biostatistics*. Ed. by E. Biganzoli, A. Vellido, F. Ambrogi, and R. Tagliaferri. Vol. 7548. LNCS. Springer, pp. 176–189.

Faraggi, D. and R. Simon (Jan. 1995). "A neural network model for survival data". In: *Stat. Med.* 14.1, pp. 73–82. ISSN: 02776715.

Grob, G. L., A. Cardoso, C. H. B. Liu, D. A. Little, and B. P. Chamberlain (2018). "A Recurrent Neural Network Survival Model: Predicting Web User Return Time". In: *Eur. Conf. Mach. Learn. Princ. Pract. Knowl. Discov. Databases*.

Hothorn, T., P. Bühlmann, S. Dudoit, A. Molinaro, and M. J. van der Laan (2006). "Survival ensembles". In: *Biostatistics* 7.3, pp. 355–373.

Jack, C. R., D. S. Knopman, W. J. Jagust, R. C. Petersen, M. W. Weiner, et al. (Feb. 2013). "Tracking pathophysiological processes in Alzheimer's disease: an updated hypothetical model of dynamic biomarkers". In: *The Lancet Neurology* 12.2, pp. 207–216.

References III

Khan, F. M. and V. B. Zubek (2008). "Support Vector Regression for Censored Data (SVRc): A Novel Tool for Survival Analysis". In: *8th IEEE International Conference on Data Mining*, pp. 863–868.

Li, H. and Y. Luan (2005). "Boosting proportional hazards models using smoothing splines, with applications to high-dimensional microarray data". In: *Bioinformatics* 21.10, pp. 2403–2409.

Liestøl, K., P. K. Andersen, and U. Andersen (June 1994). "Survival analysis and neural nets". In: *Stat. Med.* 13.12, pp. 1189–1200. ISSN: 02776715.

Mayr, A. and M. Schmid (2014). "Boosting the concordance index for survival data – a unified framework to derive and evaluate biomarker combinations". In: *PLoS One* 9.1, e84483.

Mobadersany, P., S. Yousefi, M. Amgad, D. A. Gutman, J. S. Barnholtz-Sloan, J. E. Velázquez Vega, D. J. Brat, and L. A. D. Cooper (Mar. 2018). "Predicting cancer outcomes from histology and genomics using convolutional networks". In: *Proc. Natl. Acad. Sci.* 115.13, E2970–E2979. ISSN: 0027-8424.

References IV

Pölsterl, S., N. Navab, and A. Katouzian (2015). "Fast Training of Support Vector Machines for Survival Analysis". In: *Machine Learning and Knowledge Discovery in Databases*. Ed. by A. Appice, P. P. Rodrigues, V. Santos Costa, J. Gama, A. Jorge, and C. Soares. Lecture Notes in Computer Science, pp. 243–259.

– (Sept. 2016). "An Efficient Training Algorithm for Kernel Survival Support Vector Machines". In: *3rd Workshop on Machine Learning in Life Sciences*.

Ridgeway, G. (1999). "The state of boosting". In: *Computing Science and Statistics*, pp. 172–181.

Schmid, M. and T. Hothorn (2008). "Flexible boosting of accelerated failure time models". In: *BMC Bioinformatics* 9, p. 269.

Shivaswamy, P. K., W. Chu, and M. Jansche (2007). "A Support Vector Approach to Censored Targets". In: *7th IEEE International Conference on Data Mining*, pp. 655–660.

Van Belle, V., K. Pelckmans, J. A. K. Suykens, and S. Van Huffel (2008). "Survival SVM: a practical scalable algorithm". In: *ESANN*, pp. 89–94.

References V

Wang, Z. and C. Wang (2010). "Buckley-James Boosting for Survival Analysis with High-Dimensional Biomarker Data". In: *Statistical Applications in Genetics and Molecular Biology* 9.1.