Loss functions and cross-validation with censored survival data

Anders Munch
joint work with Thomas Gerds

PhD Student, Section of Biostatistics
University of Copenhagen

September 7, 2022 – JICI

Outline

Setting and data structure

Loss functions and hold-out samples for survival data

Double robustness and fluctuation risk

Data structure and target of inference

Survival setting

$O = (\tilde{T}, \Delta, X) \sim P \in \mathcal{P}$ Oberved data with $\mathcal{O} = \mathbb{R}_+ \times \{0, 1\} \times \mathbb{R}^p$.

$Z = (T, X) \sim Q \in \mathcal{Q}$ The distribution Q (or a feature of it) is of interest.

Parameters of interest

- Low-dimensional feature of Q, e.g., the marginal survival probability $Q(T > t)$ for a fixed time horizon $t \in \mathbb{R}_+$.
- The conditional survival probability $S(t \mid x) = Q(T > t \mid X = x)$, for $x \in \mathbb{R}^p$.

The distribution Q is identifiable from the observed data distribution P under coarsening at random. Without further assumptions we would typically need to estimate the conditional survival function S for both problems.

Cross-validation and Super Learning for S

Most machine learning methods depends on one or more hyperparameters which is typically chosen using **cross-validation**.

More generally, to build robust estimators we can use **stacked regression** or **Super Learning** [Breiman, 1996, van der Laan et al., 2007] to select from or combine a collection candidate estimators/algorithms.

A central component for both cross-validation and Super Learning is the partitioning of data into training and test folds. A suitable loss function is then used to evaluate the performance of an estimator in hold-out samples.

Evaluate performance in hold-out samples

$\mathcal{D}$ data set $(O_1, \ldots, O_n)$

$\mathcal{A}$ collection of algorithms for estimating $S \in \mathcal{S}$

$\nu \in \mathcal{A}$ mapping $\mathcal{D} \longmapsto \nu(\mathcal{D}) = \hat{S} \in \mathcal{S}$

L loss function, $L \colon \mathcal{S} \times \mathcal{O} \to \mathbb{R}_+$

To evaluate the performance of $\nu \in \mathcal{A}$ let

$\mathcal{D}_1, \ldots, \mathcal{D}_K$ partition of the data set $\mathcal{D}$

$\mathcal{D}_{-k}$ the k'th training sample, $\mathcal{D}_{-k} = \mathcal{D} \setminus \mathcal{D}_k$, $k = 1, \ldots, K$

For all $k = 1, \ldots, K$,

$$L(\nu(\mathcal{D}_{-k}), O_i), \quad \text{for all} \quad O_i \in \mathcal{D}_k.$$

Averaging these values gives us an estimate of the average loss (risk) of the algorithm $\nu \in \mathcal{A}$, and we can then pick the one with lowest risk. Alternatively, we can use these value to combine all algorithm into a Super Learner.

The partial likelihood and hold-out samples

A popular choice of loss function for training survival models is the negative partial log-likelihood. Under coarsening at random and non-informative censoring the likelihood for the observed data factorizes as

$$\ell(P, O) = \ell_t(S, O) \cdot \ell_c(G, O) \cdot \ell_0(\mu, O),$$

where $G \in \mathcal{G}$ denotes the censoring mechanism and μ the marginal distribution of the baseline covariates. The negative partial log-likelihood for the component S is

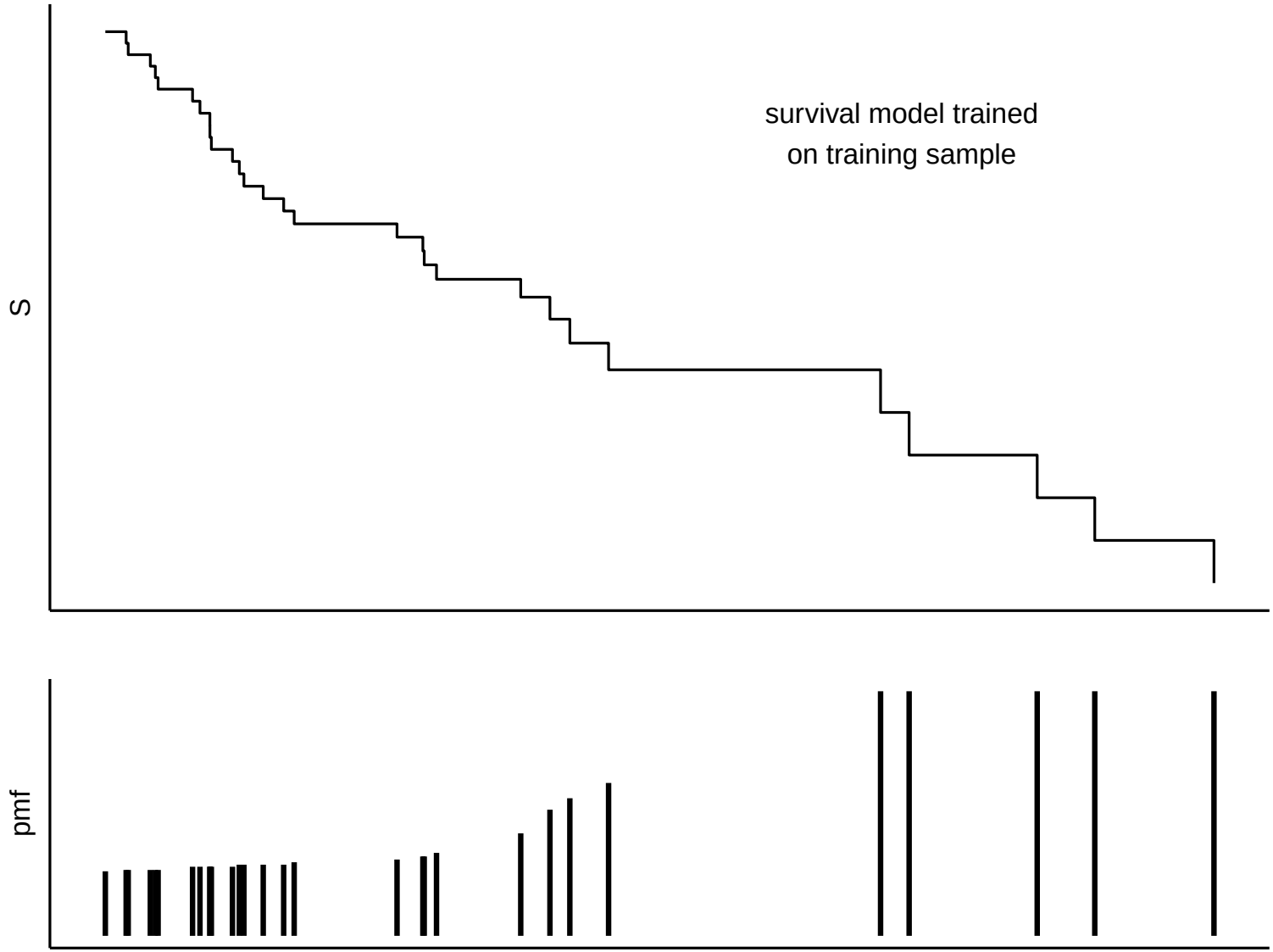
$$-\log \ell_t(S, O) = -\left\{(1 - \Delta) \log S(\tilde{T} \mid X) + \Delta \log f_S(\tilde{T} \mid X)\right\},$$

where f_S is the conditional density or pmf corresponding to S.

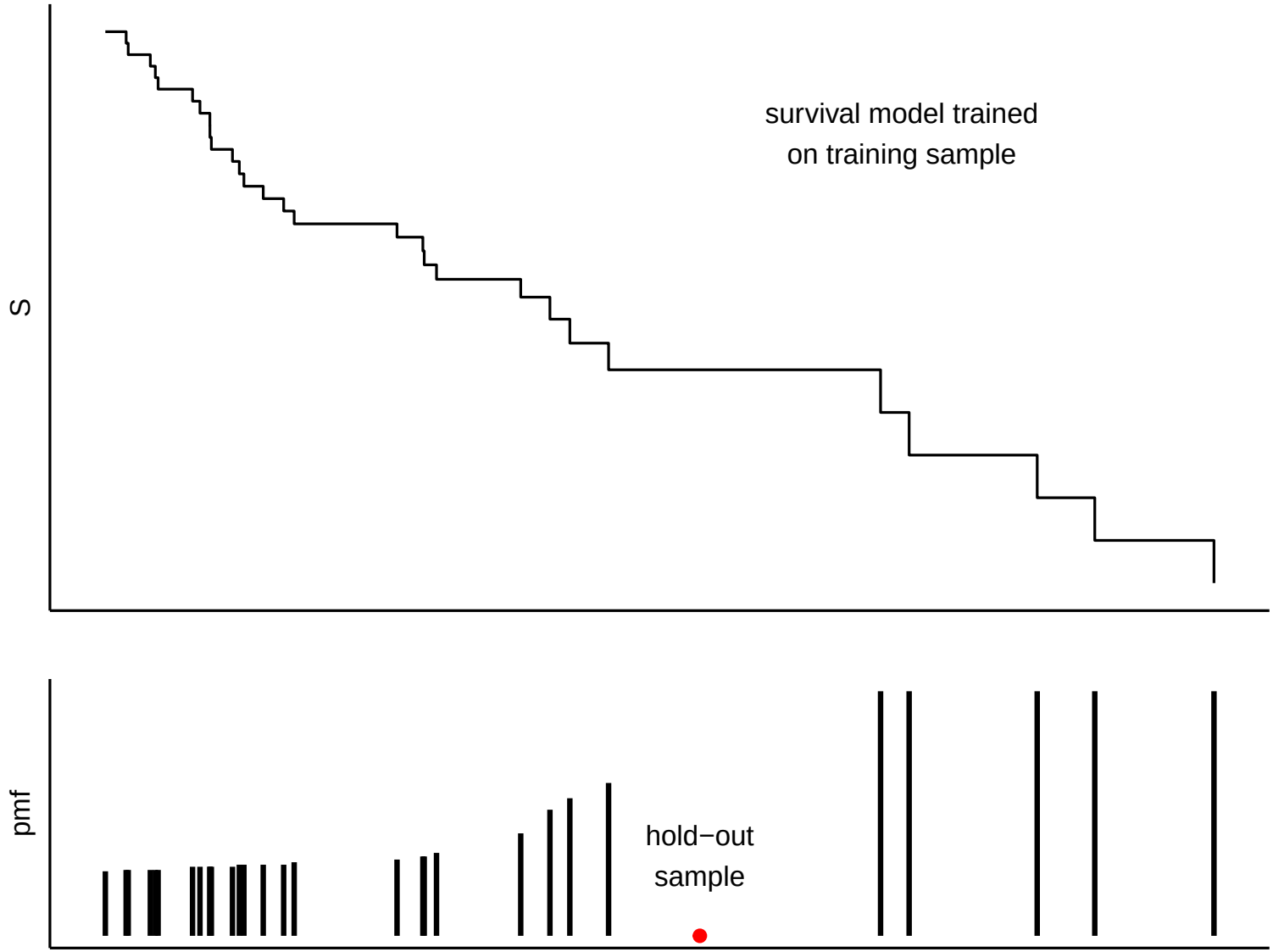
However, for many common survival estimators this loss function is unsuitable for evaluating performance in hold-out samples, because (a.s.)

$$f_{\hat{S}}(\tilde{T}_i \mid X_i) = 0 \quad \text{when} \quad \hat{S} = \nu(\mathcal{D}_{-k}) \quad \text{and} \quad (\tilde{T}_i, \Delta_i, X_i) \in \mathcal{D}_k.$$

Illustration



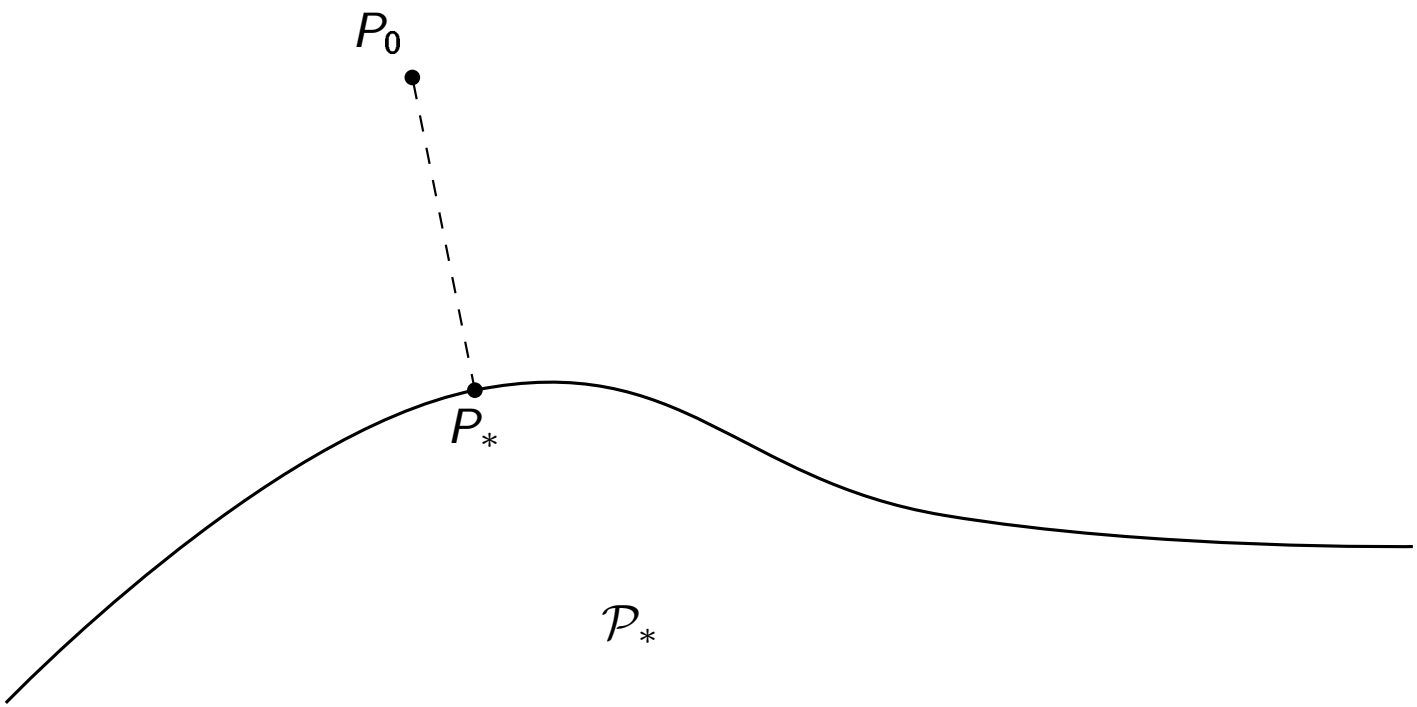
Illustration



Kullback-Leibler divergence and partial likelihoods

Maximum likelihood estimation is connected to minimizing the Kullback-Leibler divergence and gives an interpretation of the MLE under misspecified models.

$$D_{\mathrm{KL}}(P_0 \,||\, P) := P_0\left[\log \frac{p_0}{p}\right], \quad \text{where} \quad P_0 = p_0 \cdot \mu, P = p \cdot \mu.$$



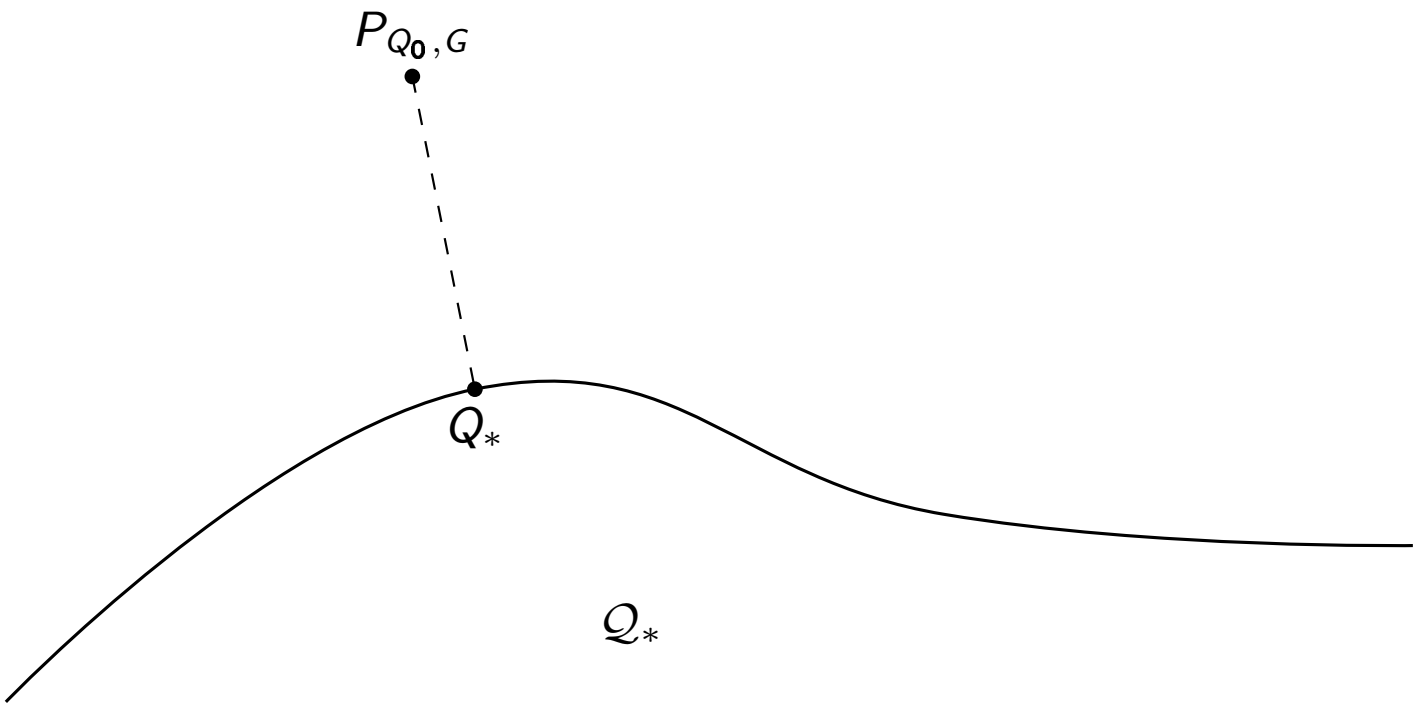
Kullback-Leibler divergence and partial likelihoods

Maximum likelihood estimation is connected to minimizing the Kullback-Leibler divergence and gives an interpretation of the MLE under misspecified models.

$$D_{\mathrm{KL}}(P_0 \,||\, P) := P_0\left[\log \frac{p_0}{p}\right], \quad \text{where} \quad P_0 = p_0 \cdot \mu, P = p \cdot \mu.$$

For a partial likelihood we are minimizing

$$Q \longmapsto D_{\mathrm{KL}}(P_{Q_0,G} \,||\, P_{Q,G}), \quad \text{with} \quad Q \in \mathcal{Q}_*.$$



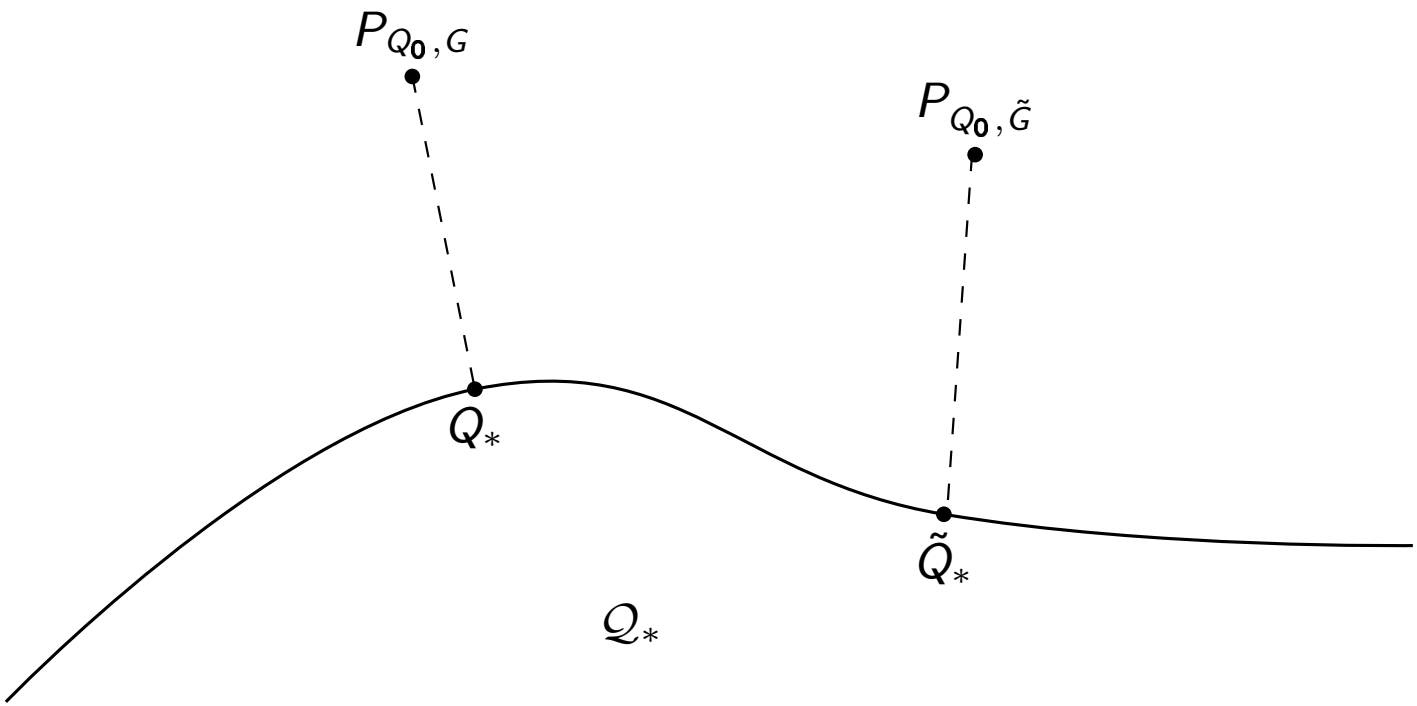
Kullback-Leibler divergence and partial likelihoods

Maximum likelihood estimation is connected to minimizing the Kullback-Leibler divergence and gives an interpretation of the MLE under misspecified models.

$$D_{\mathrm{KL}}(P_0 \,||\, P) := P_0\left[\log \frac{p_0}{p}\right], \quad \text{where} \quad P_0 = p_0 \cdot \mu, P = p \cdot \mu.$$

For a partial likelihood we are minimizing

$$Q \longmapsto D_{\mathrm{KL}}(P_{Q_0,G} \,||\, P_{Q,G}), \quad \text{with} \quad Q \in \mathcal{Q}_*.$$



Inverse probability of censoring weighted loss functions

A conceptually more attractive strategy is to use loss functions that are

(i) suited for evaluating the performance of estimating the *survival function*

(ii) defined in terms of the *distribution Q of interest*

We can do this using inverse probability of censoring weighted (IPCW) loss functions. For instance, with the Brier score

$$L_{\text{Brier}}(S, Z) = \left(S(t \mid X) - \mathbb{1}\{T > t\}\right)^2, \quad Z = (T, X) \sim Q,$$

the risk according to this loss is identifiable through

$$\mathbb{E}_Q\left[L_{\text{Brier}}(S, Z)\right] = \mathbb{E}_P\left[W_G \cdot L_{\text{Brier}}(S, Z)\right],$$

with

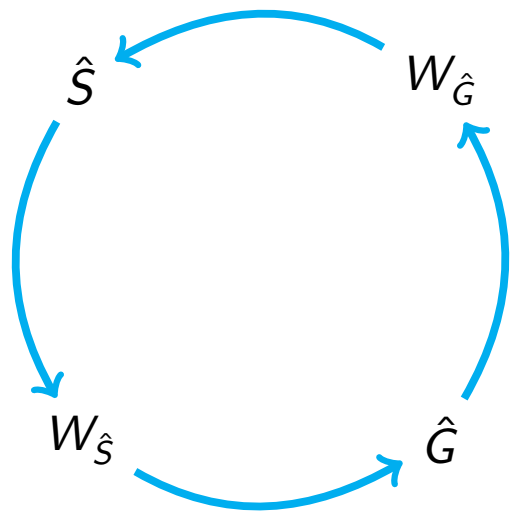
$$W_G = \frac{\mathbb{1}\{\tilde{T} > t\} + \mathbb{1}\{\tilde{T} \leq t\}\Delta}{G(\tilde{T} \wedge t \mid X)},$$

where G is the conditional "survivor" function for the censoring distribution [Graf et al., 1999, Gerds and Schumacher, 2006, van der Laan and Dudoit, 2003].

Estimation of the IPC weights

To use IPCW loss function in practice we need to estimate the G. This can also be seen as a survival problem in the sense that the event time of interest is now observed when $\Delta = 0$ and only partly observed when $\Delta = 1$.

$\implies$ The exact same challenges face us when attacking this problem.



Recently, Han et al. [2021] and Westling et al. [2021] have suggested to iterate between estimation of $\hat{S}$ and $\hat{G}$ until convergence.

The conditional survivor function as nuisance parameter

Consider now the situation where we want to estimate a low dimensional feature of Q; as example we take the marginal survival at a fixed time point, $Q(T > t)$. Under coarsening at random and a positivity assumption we can write

$$Q(T > t) = \Psi(P), \quad \text{where} \quad \Psi(P) = \mathbb{E}_P\left[S_P(t \mid X)\right],$$

where S_P denotes the conditional survival function identifiable from P.

As S is not of interest in itself, we might hope to side-step the issue of finding a suitable loss function for the nuisance parameter S by focusing directly on the target parameter instead.

Cui and Tchetgen [2020], building on ideas from Robins et al. [2007], proposed to exploit double robustness as a model selection criteria.

Double robustness

Many estimators based on the efficient influence function has a double robustness property. For instance, the efficient influence function of Ψ is $\psi(O, P) = \varphi(O, S_P, G_P) - \Psi(P)$, with

$$\varphi(O, S, G) = S(t \mid X) \left(1 - \int_0^t \frac{N(\mathrm{d}u) - \mathbb{1}\{\tilde{T} \geq u\}\Lambda_S(\mathrm{d}u \mid X)}{G(u \mid X)S(u \mid X)}\right),$$

where $N(u) = \mathbb{1}\{\tilde{T} \leq u, \Delta = 1\}$ is the counting process and Λ_S is the conditional cumulative hazard corresponding to S. It holds that

$$\mathbb{E}_P\left[\varphi(O, S_P, G_*)\right] = \mathbb{E}_P\left[\varphi(O, S_*, G_P)\right] = \Psi(P),$$

for any S_* and G_*, where S_P and G_P are the conditional survivor functions of the data generating distribution.

This motivates estimating $\Psi(P)$ with

$$\hat{\Psi} = \frac{1}{n}\sum_{i=1}^{n} \varphi(O_i, \hat{S}, \hat{G}),$$

which is consistent if either $\hat{S}$ or $\hat{G}$ is consistent.

Fluctuation risk – exploiting double robustness

Let $\mathcal{G}$ be a (finite) collection of models for G. The double robustness property implies that $\mathbb{E}_P[\varphi(O, S_P, G)] = \mathbb{E}_P[\varphi(O, S_P, G')]$ for any $G, G' \in \mathcal{G}$. In particular,

$$\max_{G,G' \in \mathcal{G}} \left| \mathbb{E}_P[\varphi(O, S_P, G)] - \mathbb{E}_P\left[\varphi(O, S_P, G')\right] \right| = 0.$$

This motivates the "fluctuation risk",[1]

$$R(S) = \max_{G,G' \in \mathcal{G}} \left| \mathbb{E}_P[\varphi(O, S, G)] - \mathbb{E}_P\left[\varphi(O, S, G')\right] \right|.$$

Let $\mathcal{A}_c$ be a collection of algorithms for estimating G. For any $\nu \in \mathcal{A}$, $\gamma \in \mathcal{A}_c$, and $k = 1, \ldots, K$ define

$$\hat{\Psi}^k_{\nu,\gamma} = \frac{1}{|\mathcal{D}_k|} \sum_{O \in \mathcal{D}_k} \varphi(O, \nu(\mathcal{D}_{-k}), \gamma(\mathcal{D}_{-k})).$$

For any $\nu \in \mathcal{A}$ we approximate the fluctuation risk with

$$\hat{R}(\nu) = \frac{1}{K} \sum_{k=1}^{K} \max_{\gamma,\gamma' \in \mathcal{A}_c} \left| \hat{\Psi}^k_{\nu,\gamma} - \hat{\Psi}^k_{\nu,\gamma'} \right|.$$

[1] or pseudo-risk because it depends $\mathcal{G}$ which is suppressed in the notation.

Illustration of the method

Consider the following simple setting where $X = (A_1, A_2, A_3)^T$ with $A_j \in \{0, 1\}$ for all j, and that we consider using Kaplan-Meier estimators stratified on each of A_j. (In this simulation, only A_1 influences survival and censoring.)

outcome algorithms

```
km1 <- function(d) prodlim(Surv(time, event) ~ A1, data = d)
km2 <- function(d) prodlim(Surv(time, event) ~ A2, data = d)
km3 <- function(d) prodlim(Surv(time, event) ~ A3, data = d)
```

censoring algorithms

```
km1_cens <- function(d) prodlim(Surv(time, !event) ~ A1, data = d)
km2_cens <- function(d) prodlim(Surv(time, !event) ~ A2, data = d)
km3_cens <- function(d) prodlim(Surv(time, !event) ~ A3, data = d)
```

Illustration of the method

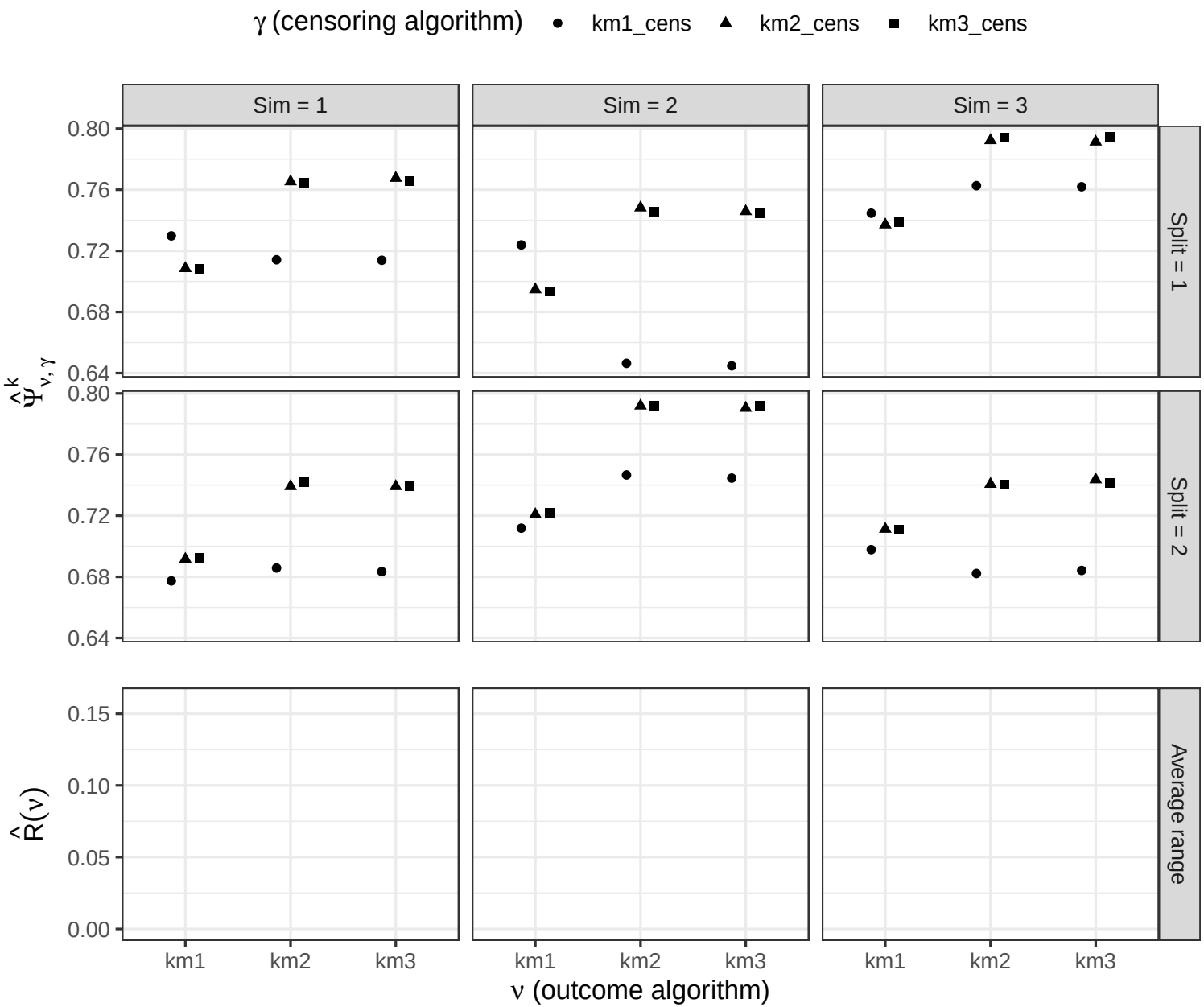


Illustration of the method

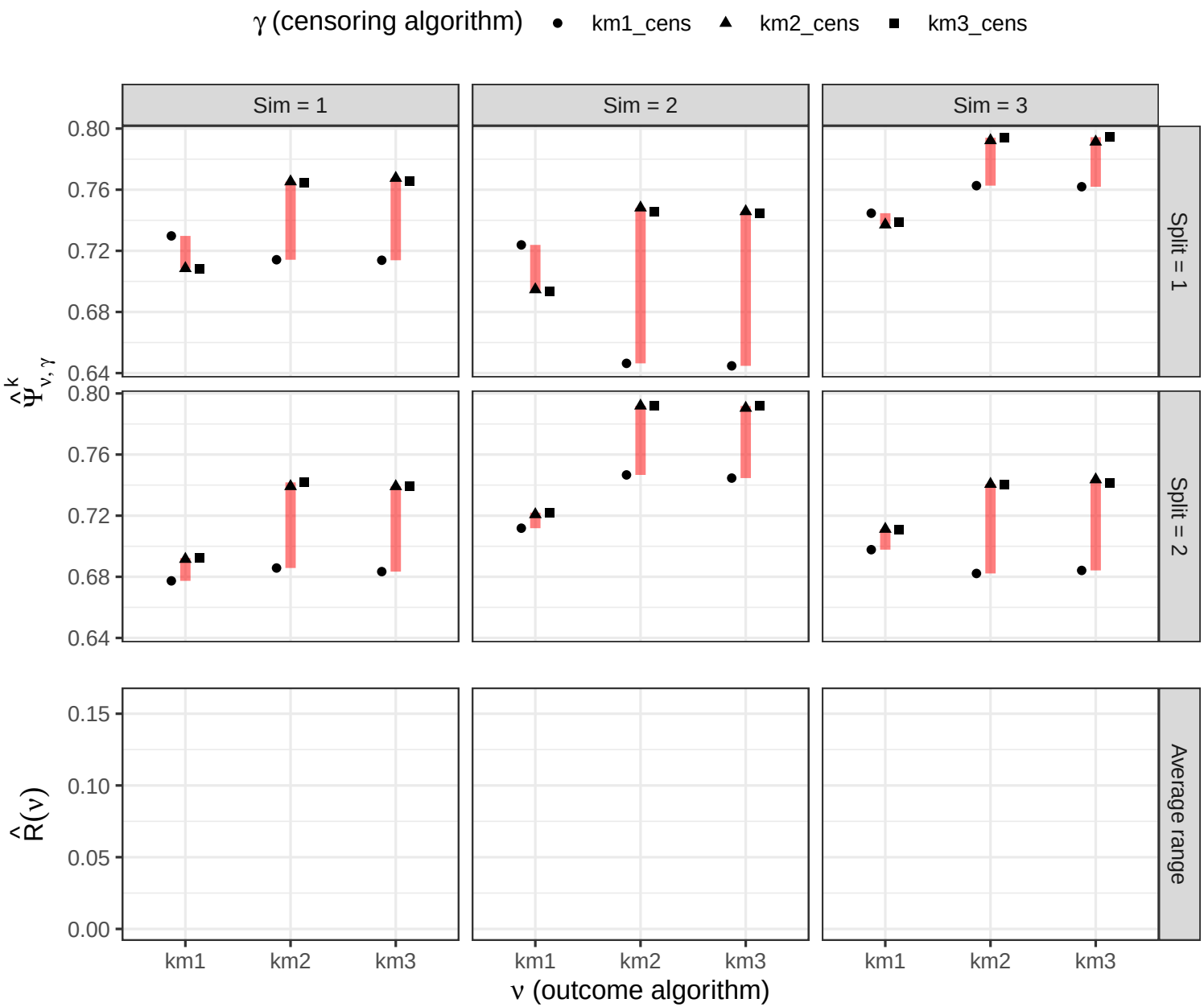
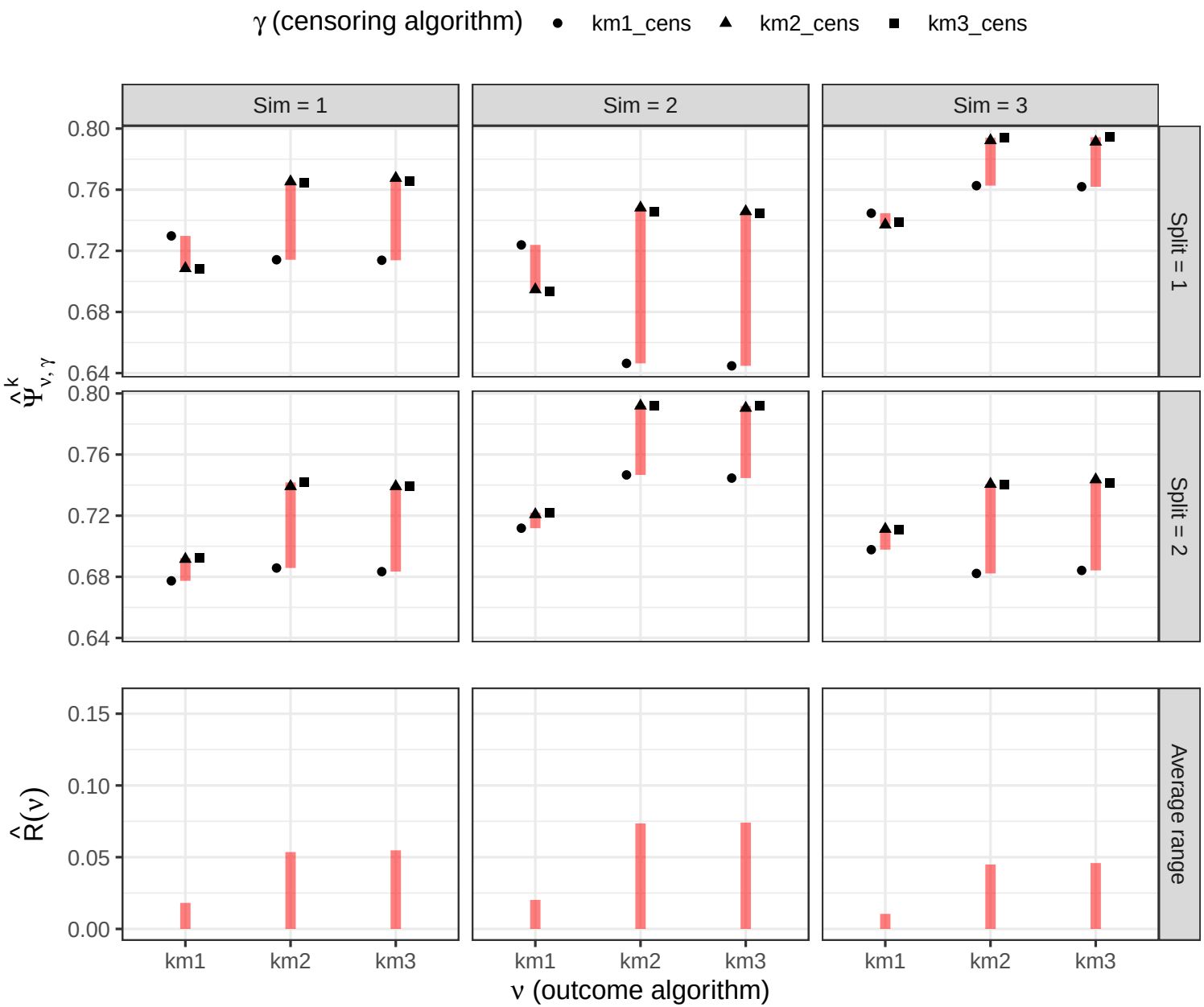
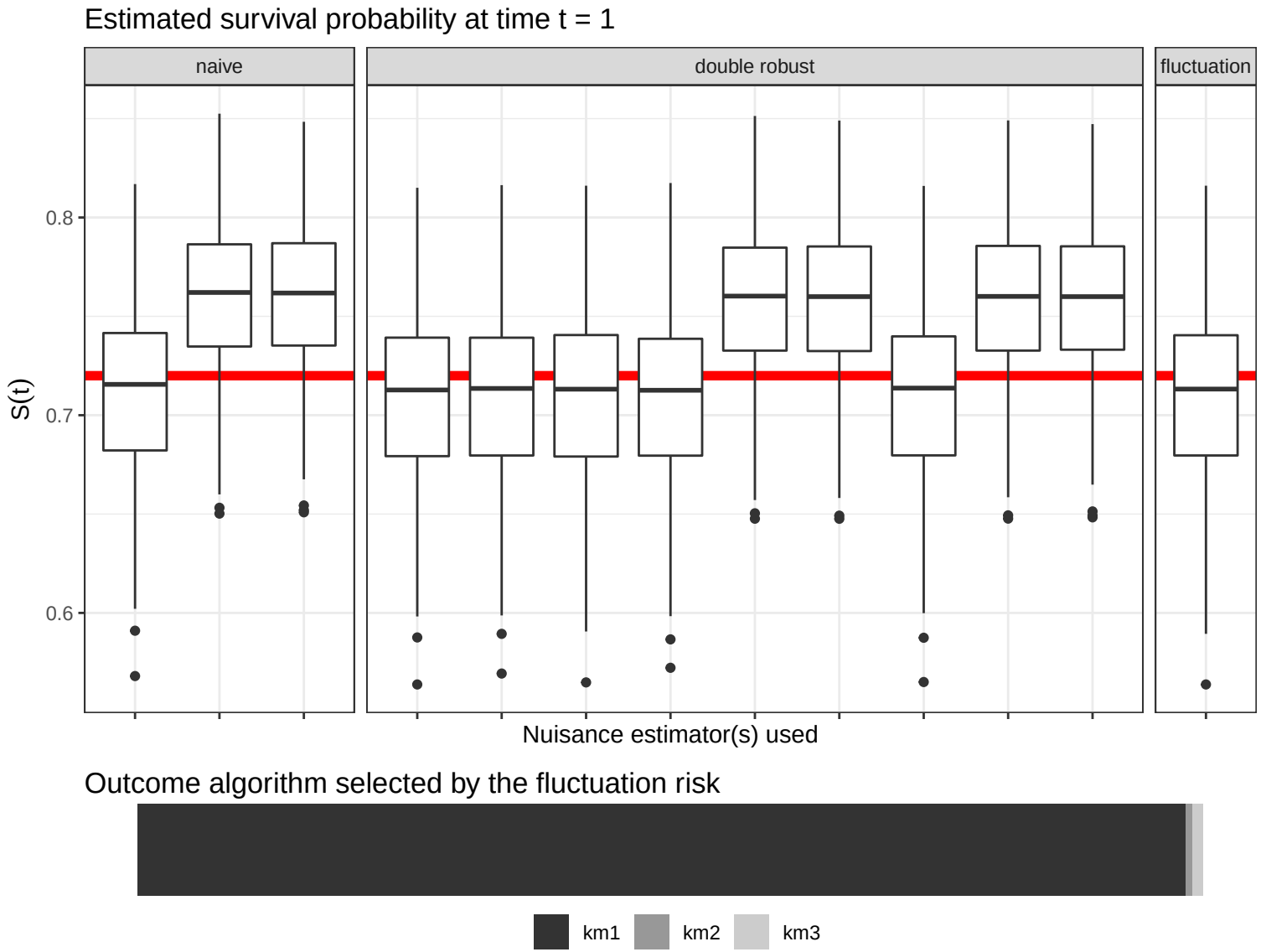


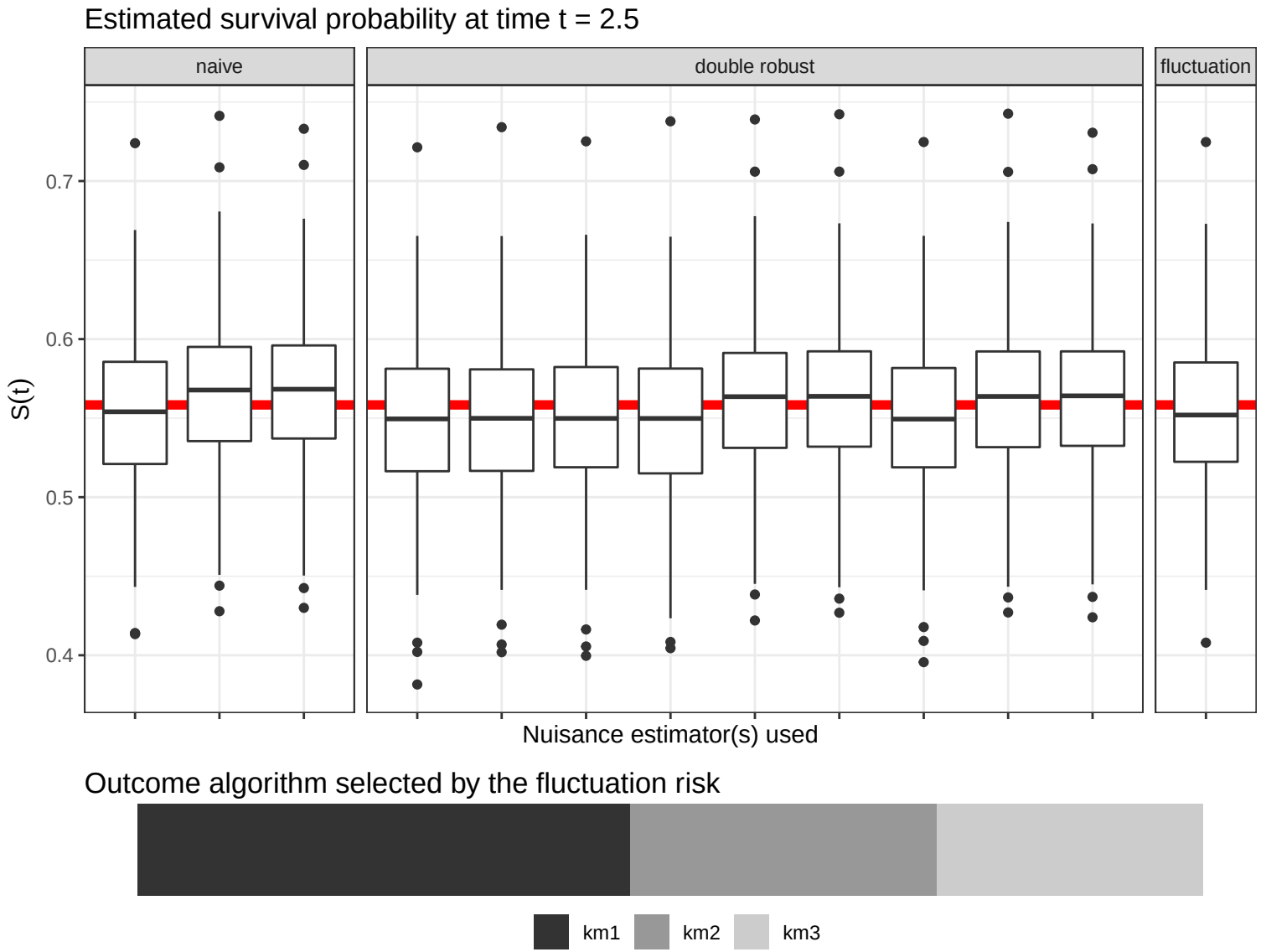
Illustration of the method



Some simulation results



Some simulation results



The conditional survivor function as target parameter

Consider now the situation where the conditional survival function $S(t \mid x)$ is the actual parameter of interest for fixed t. Assume that our goal is to build a prediction model minimizing the average Brier score. Given a model $S \in \mathcal{S}$ we can consider the average Brier score of S as a low dimensional target parameter

$$\Psi_S(P) = \mathbb{E}_P \left[W_G \cdot L_{Brier}(S, Z) \right] \quad \text{with} \quad G = G_P,$$

and proceed as above.

The conditional survivor function as target parameter

Consider now the situation where the conditional survival function $S(t \mid x)$ is the actual parameter of interest for fixed t. Assume that our goal is to build a prediction model minimizing the average Brier score. Given a model $S \in \mathcal{S}$ we can consider the average Brier score of S as a low dimensional target parameter

$$\Psi_S(P) = \mathbb{E}_P \left[W_G \cdot L_{Brier}(S, Z) \right] \quad \text{with} \quad G = G_P,$$

and proceed as above.

- With a finite *collection* of models $\mathcal{S}^* \subset \mathcal{S}$ we get a different target parameter Ψ_S for each $S \in \mathcal{S}^*$.

The conditional survivor function as target parameter

Consider now the situation where the conditional survival function $S(t \mid x)$ is the actual parameter of interest for fixed t. Assume that our goal is to build a prediction model minimizing the average Brier score. Given a model $S \in \mathcal{S}$ we can consider the average Brier score of S as a low dimensional target parameter

$$\Psi_S(P) = \mathbb{E}_P\left[W_G \cdot L_{Brier}(S, Z)\right] \quad \text{with} \quad G = G_P,$$

and proceed as above.

- With a finite *collection* of models $\mathcal{S}^* \subset \mathcal{S}$ we get a different target parameter Ψ_S for each $S \in \mathcal{S}^*$.
- With an infinite collection of models $\mathcal{S}^* \subset \mathcal{S}$ (e.g., indexed by $\beta \in \mathbb{R}^p$) the previous approach is problematic.

The conditional survivor function as target parameter

Consider now the situation where the conditional survival function $S(t \mid x)$ is the actual parameter of interest for fixed t. Assume that our goal is to build a prediction model minimizing the average Brier score. Given a model $S \in \mathcal{S}$ we can consider the average Brier score of S as a low dimensional target parameter

$$\Psi_S(P) = \mathbb{E}_P\left[W_G \cdot L_{Brier}(S, Z)\right] \quad \text{with} \quad G = G_P,$$

and proceed as above.

- With a finite *collection* of models $\mathcal{S}^* \subset \mathcal{S}$ we get a different target parameter Ψ_S for each $S \in \mathcal{S}^*$.
- With an infinite collection of models $\mathcal{S}^* \subset \mathcal{S}$ (e.g., indexed by $\beta \in \mathbb{R}^p$) the previous approach is problematic.

$\Longrightarrow$ It is desirable to fit the weights *once* so that they are "universally" applicable for estimating the performance of all $S \in \mathcal{S}$.

One idea is to use undersmoothed HAL to do this.

Conclusion

- It is not obvious what loss function to use for estimating the conditional survivor function with censored data observed in continuous time.
- If the parameter of interest is a low-dimension feature of the full data distribution we could exploit this and evaluate the performance of the nuisance parameter estimators in terms of their effect on the estimator of the target parameter.
- If the conditional survivor function itself is the parameter of interest this approach has some additional challenges.

References

L. Breiman. Stacked regressions. *Machine learning*, 24(1):49–64, 1996.

Y. Cui and E. J. Tchetgen. Selective machine learning of doubly robust functionals. *arXiv preprint arXiv:2004.03036*, 2020.

T. A. Gerds and M. Schumacher. Consistent estimation of the expected brier score in general survival models with right-censored event times. *Biometrical Journal*, 48(6): 1029–1040, 2006.

E. Graf, C. Schmoor, W. Sauerbrei, and M. Schumacher. Assessment and comparison of prognostic classification schemes for survival data. *Statistics in medicine*, 18 (17-18):2529–2545, 1999.

X. Han, M. Goldstein, A. Puli, T. Wies, A. Perotte, and R. Ranganath. Inverse-weighted survival games. *Advances in Neural Information Processing Systems*, 34, 2021.

J. Robins, M. Sued, Q. Lei-Gomez, and A. Rotnitzky. Comment: Performance of double-robust estimators when" inverse probability" weights are highly variable. *Statistical Science*, 22(4):544–559, 2007.

M. J. van der Laan and S. Dudoit. Unified cross-validation methodology for selection among estimators and a general cross-validated adaptive epsilon-net estimator: Finite sample oracle inequalities and examples. 2003.

M. J. van der Laan, E. C. Polley, and A. E. Hubbard. Super learner. *Statistical applications in genetics and molecular biology*, 6(1), 2007.

T. Westling, A. Luedtke, P. Gilbert, and M. Carone. Inference for treatment-specific survival curves using machine learning. *arXiv preprint arXiv:2106.06602*, 2021.