POST-HOC SELF-CORRECTION FOR CONDITIONAL CALIBRATION OF SURVIVAL MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose a post-hoc *self-correction* procedure that moves a fitted survival model toward conditional calibration by transforming its estimated conditional cumulative distribution function (CDF) separately for each covariate value. The transformation is a nonparametric calibration map built from the conditional D-calibration criterion in survival analysis. At the population level, a single application recovers the true conditional CDF when there is no censoring. Under right censoring, each application reduces the discrepancy from the true conditional CDF by a factor determined by the censoring mechanism, and repeated applications converge to the true conditional CDF under conditions on the time horizon. We estimate the map using kernel weighting in a reduced covariate space. Experiments on ten datasets and six survival models show that a single application of the estimated map improves conditional calibration and the integrated Brier score in most pairwise comparisons with existing post-hoc methods, while maintaining comparable discrimination.

1 INTRODUCTION

Calibration assesses the reliability of predicted survival probabilities. Decisions involving treatment planning, resource allocation, and preventive maintenance depend on accurate survival probabilities across different time horizons. Poor calibration can systematically misrepresent failure risks. Much research on modern survival models has emphasized discrimination, particularly with the development of deep neural networks capable of capturing complex covariate relationships (Wiegrebe et al., 2024). Performance is commonly assessed using the concordance index (C-index), which measures the ability to rank individuals by risk. However, discrimination alone does not establish the reliability of predicted probabilities. Recent studies have therefore placed increasing emphasis on calibration alongside discrimination (Qi et al., 2024a;b; Lee et al., 2024).

Distributional calibration (D-calibration) evaluates the uniformity of estimated cumulative distribution function (CDF) values at event times while accounting for censoring (Haider et al., 2020). Training-based methods such as X-cal (Goldstein et al., 2020) and S-cal (Park et al., 2025a) incorporate penalties based on this criterion into model fitting. Although these penalties can improve calibration, they may introduce a trade-off with predictive performance. To better balance these objectives, recent work has explored alternative loss designs that aim to improve discrimination while maintaining calibration (Lee et al., 2024).

Post-processing adjusts the predictions of an already fitted survival model. Representative methods targeting marginal calibration include conformalized survival distributions (CSD; Qi et al., 2024a) and KSP (Park et al., 2025b). These approaches separate calibration from model training and enable a fitted predictor to be corrected using additional calibration data. However, marginal calibration can conceal systematic errors through cancellation across subpopulations. Overestimation in one covariate group may offset underestimation in another, producing apparently well-calibrated predictions at the population level. This motivates conditional calibration, which requires calibration given the covariates and can reveal covariate-specific calibration errors that are hidden by marginal calibration. CSD-iPOT (Qi et al., 2024b) is a recent post-processing approach that pursues this stronger objective.

Motivated by the one-shot calibration approach in Roth (2022) and the self-consistency principle for censored data (Efron, 1967), we propose a post-hoc *self-correction* procedure for survival models.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

The procedure turns the conditional D-calibration criterion into a covariate-dependent calibration map and applies it to the estimated conditional CDF, so that corrections adapt to individual covariate profiles without restricting the calibration map to a prespecified parametric family. Since the map depends on the current predictions, it can be recomputed and applied repeatedly. Conditionally D-calibrated predictions are fixed points of the transformation. We call this procedure *conditional self-correction* and denote its practical implementation by CoSC. We study the procedure theoretically in both uncensored and censored settings and evaluate CoSC across diverse datasets and survival models.

Our main contributions are as follows:

1. We introduce conditional self-correction, a nonparametric post-hoc procedure whose calibration map is defined directly from the conditional D-calibration criterion. This allows the correction to adapt to local calibration errors across covariate regions.
2. At the population level, we establish the exact recovery of the true conditional CDF after a single application of the map without censoring and convergence through repeated applications under censoring, subject to appropriate assumptions. We also characterize the error of the kernel-based estimator.
3. We evaluate CoSC against existing post-processing methods across ten datasets and six survival models. The results demonstrate its effectiveness in improving calibration and predictive performance.

Proofs and supplementary experiments are provided in the appendices.

2 Related work

Evaluation Calibration of survival models is often assessed with calibration curves (Alonzo, 2009; Crowson et al., 2016) or at a fixed time horizon via 1-calibration (Yan et al., 2022; Li et al., 2023; Xia et al., 2023). D-calibration evaluates the full estimated distribution while accounting for censoring (Haider et al., 2020), and KS-cal measures the maximum discrepancy for D-calibration over the probability domain without predefined intervals (Park et al., 2025a). KM-calibration compares sample-average predictions with the Kaplan-Meier (KM) estimator (Yanagisawa, 2023; Qi et al., 2024a), and $\mathrm{Cal}_{\mathrm{ws}}$ (Qi et al., 2024b) evaluates conditional calibration.

In-processing Calibration can be incorporated into training (Goldstein et al., 2020; Chapfuwa et al., 2020; Avati et al., 2020; Lee et al., 2024; Park et al., 2025a). X-cal (Goldstein et al., 2020) uses a differentiable surrogate of a bin-based D-calibration penalty and S-cal (Park et al., 2025a) a penalty over random probability intervals, while Chapfuwa et al. (2020) enforce agreement with KM estimates. Lee et al. (2024) propose a survival outcome-aware contrastive objective that improves discrimination while maintaining calibration.

Post-processing Post-processing adjusts an already fitted model using calibration data. Conformal inference has been used to construct prediction intervals or bounds for censored event times (Candès et al., 2023; Gui et al., 2024; Qin et al., 2025; Davidov et al., 2025; Farina et al., 2026). For full survival distributions, CSD (Qi et al., 2024a) applies conformal corrections to predicted quantile times, and CSD-iPOT (Qi et al., 2024b) applies corrections in the probability domain to target conditional calibration; both operate at selected quantile levels and represent censored outcomes through sampling. KSP (Park et al., 2025b) optimizes a parametric CDF transformation using KS-cal, but its parameters are shared across individuals, so a given predicted probability receives the same correction regardless of the covariates. Isotonic distributional regression and its survival extension impose stochastic order restrictions for conditional distribution estimation (Henzi et al., 2021b;a; Bladt et al., 2026).

Individual calibration In uncensored regression, calibration is usually defined by the uniformity of the probability integral transform (PIT) averaged over the covariates (Gneiting et al., 2007), and post-hoc recalibration learns a map from the estimated distribution alone (Kuleshov et al., 2018; Song et al., 2019). Zhao et al. (2020) define *individual calibration*, which requires PIT uniformity at each input, and introduce probably approximately individually calibrated (PAIC) as a relaxation. For deterministic forecasters, individual calibration holds only when the forecast equals the true conditional distribution and cannot be verified from finite data. These limitations motivate the

use of randomized forecasters and a monotone variant of PAIC, termed mPAIC. Sheng & Henao (2026) adopt these definitions for classification and train with a kernel calibration error (Marx et al., 2023). Related notions include multi-calibration (Hébert-Johnson et al., 2018), conditional calibration (Gneiting & Resin, 2023), kernel-based local calibration (Luo et al., 2022), and nonparametric regression recalibration via conditional kernel mean embeddings (Jung et al., 2026).

The conditional D-calibration criterion in Section 4 takes a similar covariate-level view for right-censored data, in which censored observations are handled through the conditional distribution of the PIT. Rather than randomizing the forecaster or modifying training, we turn this criterion into a covariate-dependent calibration map, estimate it post hoc from nearby calibration observations by kernel weighting, and provide guarantees that account for censoring.

3 Background on Survival Analysis

We introduce the notation for right-censored survival data, followed by the discrimination and calibration criteria used in this work.

3.1 Survival Analysis

We consider right-censored survival data, where the event time T is subject to censoring by a random censoring time C. The observed time and event indicator are defined as $Y = \min(T, C)$, $\delta = \mathbb{I}(T \leq C)$, respectively, where $\delta = 1$ indicates an observed event and $\delta = 0$ indicates a censored observation. Each observation is represented by $(Y, \delta, \boldsymbol{Z})$, where $\boldsymbol{Z} \in \mathcal{Z} \subseteq \mathbb{R}^d$, with $d \geq 1$, denotes a vector of covariates. We assume that T and C are conditionally independent given $\boldsymbol{Z}$, that is, $T \perp\!\!\!\perp C \mid \boldsymbol{Z} = \boldsymbol{z}$. We observe an independent and identically distributed sample $\{(y_i, \delta_i, \boldsymbol{z}_i)\}_{i=1}^n$.

Let $F_{\boldsymbol{z}}(t) = \mathbb{P}(T \leq t \mid \boldsymbol{Z} = \boldsymbol{z})$ denote the conditional cumulative distribution function (CDF). The corresponding survival function $S_{\boldsymbol{z}}(t)$ and hazard function $\lambda_{\boldsymbol{z}}(t)$ are defined as

$$\lambda_{\boldsymbol{z}}(t) = \lim_{\Delta \downarrow 0} \frac{\mathbb{P}(t \leq T < t + \Delta \mid T \geq t, \boldsymbol{Z} = \boldsymbol{z})}{\Delta}, \; S_{\boldsymbol{z}}(t) = \mathbb{P}(T > t \mid \boldsymbol{Z} = \boldsymbol{z}) = 1 - F_{\boldsymbol{z}}(t).$$

The cumulative hazard function $\Lambda_{\boldsymbol{z}}(t)$ is defined by $\int_0^t \lambda_{\boldsymbol{z}}(s)\, ds$, and the relationship between the survival function and the cumulative hazard function is derived as $S_{\boldsymbol{z}}(t) = \exp\{-\Lambda_{\boldsymbol{z}}(t)\}$.

3.2 Discrimination

In survival analysis, the concordance index (C-index; Harrell Jr et al., 1996) is widely used to evaluate discriminative performance. The C-index measures the proportion of concordant pairs between observed times and predictors, and is defined as

$$\text{C-index} = \frac{\sum_{i \neq j} \delta_i \mathbb{I}(y_i < y_j) \mathbb{I}(m_i < m_j)}{\sum_{i \neq j} \delta_i \mathbb{I}(y_i < y_j)},$$

where m_i, m_j are predicted quantities, such as the predicted mean or a (negative) risk score. In this work, we use the predicted mean as m_i. The C-index ranges from 0 to 1, with higher values indicating better discrimination. A value of 0.5 is expected to represent a random guess.

3.3 D-calibration

Suppose that the true conditional CDF $F_{\boldsymbol{z}}$ is continuous. Then $F_{\boldsymbol{Z}}(T) \mid \boldsymbol{Z} = \boldsymbol{z} \sim \text{Unif}[0, 1]$. Under conditionally independent censoring, for a censored observation with $Y = y$ and $\boldsymbol{Z} = \boldsymbol{z}$, the conditional distribution of $F_{\boldsymbol{z}}(T)$ is $\text{Unif}[F_{\boldsymbol{z}}(y), 1]$. Combining these properties yields

$$\mathbb{E}\Big[\delta \mathbb{I}(F_{\boldsymbol{Z}}(Y) \in I) + (1 - \delta) \mathbb{P}(F_{\boldsymbol{Z}}(T) \in I \mid Y, \boldsymbol{Z}, \delta = 0)\Big] = |I| \tag{1}$$

where $\mathbb{P}(F_{\boldsymbol{Z}}(T) \in I \mid Y, \boldsymbol{Z}, \delta = 0) = \mathbb{I}(F_{\boldsymbol{Z}}(Y) \in I) \frac{b - F_{\boldsymbol{Z}}(Y)}{1 - F_{\boldsymbol{Z}}(Y)} + \mathbb{I}(F_{\boldsymbol{Z}}(Y) < a) \frac{b - a}{1 - F_{\boldsymbol{Z}}(Y)}$ and $I = [a, b]$ is a subinterval in $[0, 1]$. Because the expectation is taken over the covariate distribution, this criterion characterizes marginal D-calibration. In practice, the true conditional CDF $F_{\boldsymbol{z}}$

is replaced by its estimate $\hat{F}_{\boldsymbol{z}}$, and the expectation is approximated by the corresponding sample average. Conventional D-calibration metrics summarize the resulting deviations from $|I|$ over pre-specified intervals (Haider et al., 2020; Qi et al., 2024a; Park et al., 2025a). The dependence on interval choice motivates an alternative assessment using KS-cal (Park et al., 2025b).

3.4 KS-CAL

Park et al. (2025a) proposed a Kolmogorov-Smirnov metric for measuring D-calibration, referred to as KS-cal. From (1), by setting $a = 0$ and $b = x$ for $x \in [0, 1]$, the definition of D-calibration is simplified to

$$\mathbb{E}\left[\mathbb{I}\left(F_{\boldsymbol{Z}}\left(Y\right) \leq x\right)\left\{\delta + (1-\delta)\frac{x - F_{\boldsymbol{Z}}\left(Y\right)}{1 - F_{\boldsymbol{Z}}\left(Y\right)}\right\}\right] = x \tag{2}$$

We define $\tilde{F}_n\left(x\right) = \frac{1}{n}\sum_{i=1}^{n}\mathbb{I}(\hat{F}_{\boldsymbol{z}_i}\left(y_i\right) \leq x)\left\{\delta_i + (1-\delta_i)\frac{x - \hat{F}_{\boldsymbol{z}_i}(y_i)}{1-\hat{F}_{\boldsymbol{z}_i}(y_i)}\right\}$. Then, KS-cal is defined as $\sup_{x\in[0,1]}|\tilde{F}_n\left(x\right) - x|$. KS-cal measures the largest discrepancy over the probability domain without requiring predefined intervals. In the following section, we use its conditional counterpart to construct covariate-dependent calibration maps.

4 CONDITIONAL CALIBRATION AND SELF-CORRECTION CALIBRATOR

4.1 CONDITIONAL CALIBRATION

In survival analysis, D-calibration is conventionally formulated as a marginal property. As shown in (2), its criterion integrates over the distribution of $\boldsymbol{Z}$. Consequently, calibration errors within a specific covariate region $\mathcal{B}$ may be masked by errors elsewhere: even when marginal D-calibration holds, calibration conditional on $\boldsymbol{Z} \in \mathcal{B}$ may fail and can vary across covariate regions. Motivated by the covariate-level calibration perspective of Jung et al. (2026), we consider conditional[1] D-calibration, which requires the D-calibration criterion to hold conditional on the covariates and therefore prevents cancellation of calibration errors across covariate regions. Formally, an estimated CDF $\hat{F}_{\boldsymbol{z}}$ is conditionally D-calibrated if it satisfies

$$\mathbb{E}\left[\mathbb{I}\left(\hat{F}_{\boldsymbol{Z}}\left(Y\right) \leq x\right)\left\{\delta + (1-\delta)\,\frac{x - \hat{F}_{\boldsymbol{Z}}\left(Y\right)}{1 - \hat{F}_{\boldsymbol{Z}}\left(Y\right)}\right\}\,\middle|\,\boldsymbol{Z} = \boldsymbol{z}\right] = x$$

for every $x \in [0, 1]$ and almost every covariate value $\boldsymbol{z}$ under the covariate distribution.

4.2 SELF-CORRECTION CALIBRATOR

Directly estimating this conditional expectation is challenging because, for continuous or high-dimensional covariates, there are typically few or no observations with covariate values exactly equal to $\boldsymbol{z}$. We first define the population calibration map and then describe its kernel estimator. For a fixed $\boldsymbol{z}$, define

$$H_{\boldsymbol{z}}\left(x\right) = \mathbb{E}\left[\mathbb{I}\left(\hat{F}_{\boldsymbol{Z}}\left(Y\right) \leq x\right)\left\{\delta + (1-\delta)\,\frac{x - \hat{F}_{\boldsymbol{Z}}\left(Y\right)}{1 - \hat{F}_{\boldsymbol{Z}}\left(Y\right)}\right\}\,\middle|\,\boldsymbol{Z} = \boldsymbol{z}\right].$$

To see how $H_{\boldsymbol{z}}$ can be used to correct the estimated CDF, consider the uncensored setting. In this case, $H_{\boldsymbol{z}}$ is the conditional CDF of $\hat{F}_{\boldsymbol{z}}\left(T\right)$ given $\boldsymbol{Z} = \boldsymbol{z}$. If $\hat{F}_{\boldsymbol{z}}$ is strictly increasing, then

$$H_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}\left(t\right)\right) = \mathbb{P}\left(\hat{F}_{\boldsymbol{z}}\left(T\right) \leq \hat{F}_{\boldsymbol{z}}\left(t\right) \mid \boldsymbol{Z} = \boldsymbol{z}\right) = \mathbb{P}\left(T \leq t \mid \boldsymbol{Z} = \boldsymbol{z}\right) = F_{\boldsymbol{z}}\left(t\right).$$

Thus, composing $\hat{F}_{\boldsymbol{z}}$ with $H_{\boldsymbol{z}}$ recovers the true conditional CDF in a single step. This identity motivates the transformation $\hat{F}^{*}_{\boldsymbol{z}}\left(t\right) = H_{\boldsymbol{z}}(\hat{F}_{\boldsymbol{z}}\left(t\right))$ for $t \geq 0$. With censoring, one-step recovery is generally not guaranteed, but the transformation can be applied iteratively by recomputing

[1] Related work sometimes uses the term individual calibration for a similar covariate-conditional notion. We use conditional calibration to emphasize that the criterion is defined conditional on $\boldsymbol{Z} = \boldsymbol{z}$ and is required to hold for almost every $\boldsymbol{z}$ in the support of the covariate distribution.

the calibration map from the updated CDF at each step. This population-level correction follows the classical self-consistency principle for censored data (Efron, 1967). Specifically, starting with $\hat{F}_{\boldsymbol{z}}^{(0)} = \hat{F}_{\boldsymbol{z}}$, we define

$$\hat{F}_{\boldsymbol{z}}^{(k+1)}(t) = H_{\boldsymbol{z}}^{(k)}\left(\hat{F}_{\boldsymbol{z}}^{(k)}(t)\right), \qquad k = 0, 1, \ldots,$$

where $H_{\boldsymbol{z}}^{(k)}$ is obtained by replacing $\hat{F}_{\boldsymbol{Z}}$ with $\hat{F}_{\boldsymbol{Z}}^{(k)}$ in the definition of $H_{\boldsymbol{z}}$. The following propositions characterize the population-level behavior of this transformation. The proofs are provided in Appendices B.1 and B.2.

Proposition 1 (Uncensored case)**.** *Suppose that there is no censoring and that $\hat{F}_{\boldsymbol{z}}$ is strictly increasing. Then, $\hat{F}_{\boldsymbol{z}}^{(1)}(t) = F_{\boldsymbol{z}}(t)$ for every $t \geq 0$.*

This result coincides with the classical recalibration identity of Kuleshov et al. (2018); Roth (2022). In the presence of censoring, exact recovery is generally no longer guaranteed. Nevertheless, under suitable conditions, repeated application of the transformation reduces the estimation error. The following proposition quantifies this reduction through a censoring-weighted L_2 error bound.

Proposition 2 (Censored case)**.** *Suppose that $\hat{F}_{\boldsymbol{z}}$ is strictly increasing. Define the censoring distribution function $G_{\boldsymbol{z}}(t) = \mathbb{P}(C \leq t \mid \boldsymbol{Z} = \boldsymbol{z})$, and let $\tau > 0$ be a finite time horizon satisfying the conditions in Appendix B. Then,*

$$\left(\int_0^\tau \left|\hat{F}_{\boldsymbol{z}}^{(k)}(t) - F_{\boldsymbol{z}}(t)\right|^2 dG_{\boldsymbol{z}}(t)\right)^{1/2} \leq \{G_{\boldsymbol{z}}(\tau)\}^k \left(\int_0^\tau \left|\hat{F}_{\boldsymbol{z}}(t) - F_{\boldsymbol{z}}(t)\right|^2 dG_{\boldsymbol{z}}(t)\right)^{1/2} \text{ or}$$

$$\left\|\hat{F}_{\boldsymbol{z}}^{(k)} - F_{\boldsymbol{z}}\right\|_{L^2([0,\tau],G_{\boldsymbol{z}})} \leq \{G_{\boldsymbol{z}}(\tau)\}^k \left\|\hat{F}_{\boldsymbol{z}} - F_{\boldsymbol{z}}\right\|_{L^2([0,\tau],G_{\boldsymbol{z}})}$$

When $G_{\boldsymbol{z}}(\tau) < 1$, the bound establishes geometric convergence to $F_{\boldsymbol{z}}$ in the censoring-weighted L_2 norm, with contraction factor $G_{\boldsymbol{z}}(\tau)$. Although direct evaluation of this error requires knowledge of both the true conditional CDF and the conditional censoring distribution, the bound characterizes how censoring affects the correction. In particular, a smaller $G_{\boldsymbol{z}}(\tau)$ gives a stronger contraction guarantee over the specified time horizon. This guarantee applies to the population-level conditional calibration maps. The simulation study in Appendix D examines the behavior of repeated transformations across different models, censoring rates, and sample sizes. We next turn to the practical estimation of the calibration map from finite data.

To implement this transformation, we estimate $H_{\boldsymbol{z}}$ by borrowing information from nearby observations through kernel weighting. We refer to the dataset used to post-process the predictions of a fitted survival model as the calibration set, denoted by $\mathcal{D}_{\text{cal}}$. For a query covariate $\boldsymbol{z}$, we estimate $H_{\boldsymbol{z}}$ as

$$\hat{H}_{\boldsymbol{z}}(x; h) = \frac{\sum_{j \in \mathcal{D}_{\text{cal}}} w_j(\boldsymbol{z}; h)\, \mathbb{I}\left(\hat{F}_{\boldsymbol{z}_j}(y_j) \leq x\right) \left\{\delta_j + (1 - \delta_j)\frac{x - \hat{F}_{\boldsymbol{z}_j}(y_j)}{1 - \hat{F}_{\boldsymbol{z}_j}(y_j)}\right\}}{\sum_{j \in \mathcal{D}_{\text{cal}}} w_j(\boldsymbol{z}; h)} \tag{3}$$

where $w_j(\boldsymbol{z}; h) = K(\|\boldsymbol{z} - \boldsymbol{z}_j\|_2 / h)$ is a kernel weight with bandwidth h. This estimator assigns greater weights to observations in $\mathcal{D}_{\text{cal}}$ that are closer to the query point, thereby borrowing information from observations with similar covariate values.

To reduce the dimension used for kernel smoothing, we apply principal component analysis (PCA) before computing the kernel weights and measure distances in the projected space. Let $\boldsymbol{V} \in \mathbb{R}^{d \times p}$ contain the $p\,(< d)$ retained principal loading vectors and let σ_s^2 denote the variance of the sth retained component. Define $\tilde{\boldsymbol{Z}} = \boldsymbol{V}^\top \boldsymbol{Z} / \sqrt{\sum_{s=1}^p \sigma_s^2}$, normalizing the projected covariates to unit total variance. Instead of $H_{\boldsymbol{z}}(x)$, we then estimate $H_{\tilde{\boldsymbol{z}}}(x) = \mathbb{E}[H_{\boldsymbol{Z}}(x) \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}}]$. It coincides with $H_{\boldsymbol{z}}(x)$ whenever $H_{\boldsymbol{z}}$ depends on $\boldsymbol{z}$ only through $\tilde{\boldsymbol{z}}$, and their difference is the dimension-reduction error, which we discuss below. The estimator $\hat{H}_{\tilde{\boldsymbol{z}}}(x; h)$ is obtained from (3) by replacing $w_j(\boldsymbol{z}; h)$ with $w_j(\tilde{\boldsymbol{z}}; h)$ and we use the Laplacian kernel, with weights $w_j(\tilde{\boldsymbol{z}}; h) = \exp(-\|\tilde{\boldsymbol{z}} - \tilde{\boldsymbol{z}}_j\|_2 / h)$. We examine the effect of kernel choice in the ablation study in Section 5. We select the bandwidth h using cross-validation. The detailed bandwidth-selection procedure is provided in Appendix C. Finally, we obtain the post-processed CDF as $\hat{F}_{\boldsymbol{z}}^{(k+1)}(t) = \hat{H}_{\tilde{\boldsymbol{z}}}^{(k)}(\hat{F}_{\boldsymbol{z}}^{(k)}(t); \hat{h}^{(k)})$. We refer to this procedure as conditional self-correction (CoSC). Figure 1 illustrates how CoSC transforms the estimated CDFs.

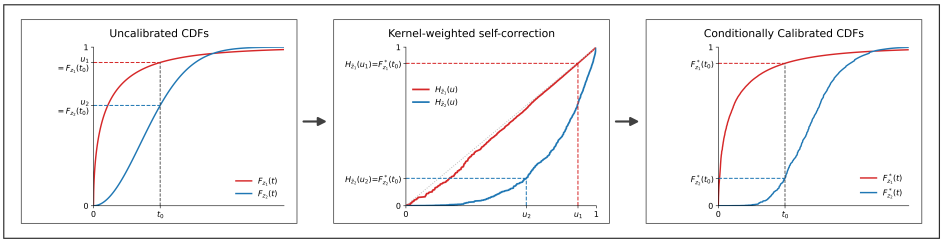

Figure 1: Illustration of the self-correction procedure.

4.3 Theoretical Properties of the Kernel Estimator

Let $\hat{H}_{\tilde{\boldsymbol{z}},n}(x;h)$ denote the kernel estimator $\hat{H}_{\tilde{\boldsymbol{z}}}(x;h)$ constructed from n observations. We assume the regularity conditions stated in Appendix B.3, which require sufficient smoothness of $H_{\tilde{\boldsymbol{z}}}$. Under these conditions, Lemma 2 in Appendix B.4 implies that, with a probability of at least $1-\delta_\epsilon$,

$$\forall x \in [0,1], \quad \left|\hat{H}_{\tilde{\boldsymbol{z}}_i,n}(x;h) - H_{\tilde{\boldsymbol{z}}_i}(x)\right| \le \epsilon$$

with $h \asymp n^{-1/(4+p)}$ and $n = \Theta\left(\left(\frac{W(-2\epsilon/(4+p))}{-2\epsilon/(4+p)}\right)^{(4+p)/2}\right)^2$. Thus, the result characterizes the sample size n required to achieve a uniform error tolerance $\epsilon > 0$ with a probability of at least $1-\delta_\epsilon$. The proof is provided in Appendix B.4.

Implication of uniform convergence To examine local behavior, consider a shrinking covariate region $\mathcal{B}_n$ as n increases. A marginal calibration method shared across covariates cannot explicitly adapt to local regions. If $\mathbb{P}(\tilde{\boldsymbol{Z}} \in \mathcal{B}_n) \asymp n^{-\alpha}$, the effective sample size in $\mathcal{B}_n$ is of order $n^{1-\alpha}$, yielding an estimation error of $O_p(n^{-(1-\alpha)/2})$. By contrast, the kernel estimator borrows information from nearby observations, including those outside $\mathcal{B}_n$, and achieves a rate of $O_p\left(n^{-2/(4+p)}\right)$, which is faster when $\alpha > p/(4+p)$. The gain from kernel weighting therefore arises from its ability to adapt locally, rather than from a uniformly faster convergence rate.

Error analysis By the Minkowski inequality, the total error $\|\hat{F}_{\boldsymbol{z}}^{(k+1)} - F_{\boldsymbol{z}}\|_{L_2([0,\tau],G_{\boldsymbol{z}})}$ is bounded above as follows:

$$\begin{aligned}\left\|\hat{H}_{\tilde{\boldsymbol{z}}}^{(k)}\left(\hat{F}_{\boldsymbol{z}}^{(k)};\hat{h}^{(k)}\right) - F_{\boldsymbol{z}}\right\|_{L_2([0,\tau],G_{\boldsymbol{z}})} \le{}& \left\|\hat{H}_{\tilde{\boldsymbol{z}}}^{(k)}\left(\hat{F}_{\boldsymbol{z}}^{(k)};\hat{h}^{(k)}\right) - H_{\tilde{\boldsymbol{z}}}^{(k)}\left(\hat{F}_{\boldsymbol{z}}^{(k)}\right)\right\|_{L_2([0,\tau],G_{\boldsymbol{z}})} \\ &+ \left\|H_{\tilde{\boldsymbol{z}}}^{(k)}\left(\hat{F}_{\boldsymbol{z}}^{(k)}\right) - H_{\boldsymbol{z}}^{(k)}\left(\hat{F}_{\boldsymbol{z}}^{(k)}\right)\right\|_{L_2([0,\tau],G_{\boldsymbol{z}})} \\ &+ \left\|H_{\boldsymbol{z}}^{(k)}\left(\hat{F}_{\boldsymbol{z}}^{(k)}\right) - F_{\boldsymbol{z}}\right\|_{L_2([0,\tau],G_{\boldsymbol{z}})}\end{aligned}$$

The three terms represent the kernel estimation error, dimension-reduction error, and population-level transformation error, respectively. Proposition 2 bounds the third term, whereas the first two arise from estimating the calibration map using finite data and projected covariates. The kernel estimation rate above depends on the smoothing dimension p. Retaining fewer PCs can improve this rate but may increase the dimension-reduction error, which need not vanish as the sample grows. Appendix B.5 considers additional conditions under which this error becomes negligible relative to the kernel estimation rate. In general, the effect of PCA on the total error depends on the trade-off between these two components. Appendix D examines this tradeoff and the population-level transformation error in finite sample experiments.

5 Experiments

We compare CoSC with the baseline, KSP (Park et al., 2025b), CSD (Qi et al., 2024a), and CSD-iPOT (Qi et al., 2024b).

[2] $W(\cdot)$ is the Lambert W function, defined by $W(s)\exp(W(s)) = s$.

5.1 EXPERIMENTAL SETUP

Datasets We evaluate the methods on ten datasets: WHAS, METABRIC, GBSG, NACD, NB-SEQ, SUPPORT, MIMIC-III, SEER-liver, SEER-stomach, and SEER-bladder. For each repetition, each dataset is randomly partitioned into training, validation, and test sets in a 3:1:1 ratio, with stratification by censoring status. The training set is used to fit the underlying survival model, the validation set to fit the post-processing methods and select their hyperparameters, and the test set for final evaluation. We repeat the entire procedure 30 times using distinct random seeds. Based on the number of observations, we categorize the datasets into three groups: Small (WHAS, METABRIC, GBSG, and NACD), Medium (NB-SEQ, SUPPORT, and MIMIC-III), and Large (SEER-liver, SEER-stomach, and SEER-bladder). Details of dataset construction and preprocessing are provided in Appendix G.

Post-processing setup For CSD and CSD-iPOT, the validation set serves as the conformal set, and we use nine quantile levels, $\rho = \{0.1, \ldots, 0.9\}$, following the recommendation in the original work. For CoSC, we use a single correction step. Although the preceding section establishes convergence to the true conditional CDF under repeated application, each additional iteration requires re-estimating the calibration map and re-selecting the bandwidth, which may propagate estimation error and increase computational cost. Moreover, the simulation study in Appendix D shows that the population-level transformation achieves its largest reduction in error at the first iteration, with subsequent iterations yielding comparatively modest improvements in most settings. Accordingly, throughout the main experiments, CoSC denotes the one-step estimator $\hat{F}^{(1)}_{\boldsymbol{z}}$. For bandwidth selection, we use 2-fold cross-validation with 10 repetitions. We retain enough principal components to explain at least 90% of the total variance and employ the Laplacian kernel. We assess the sensitivity of CoSC to these design choices in the ablation study. All experiments are conducted on a system equipped with an Intel Xeon Gold 6526Y CPU and an NVIDIA A6000 GPU.

Survival models We use six baseline models: DeepSurv (Katzman et al., 2018), Multi-Task Logistic Regression (MTLR; Yu et al., 2011), LognormalNN (Goldstein et al., 2020), Survival Continuous Ranked Probability Score (S-CRPS; Avati et al., 2020), DeepHit (Lee et al., 2018), and the Weibull accelerated failure time (AFT) model (Stute, 1993), following the experimental setup of Park et al. (2025b). Because MTLR and DeepHit produce predictions at discrete time points, we linearly interpolate their estimated CDFs. Detailed descriptions of each model are provided in Appendix F.

Evaluation metrics We evaluate each post-processing method in terms of discrimination, marginal and conditional calibration, and overall predictive accuracy. The C-index and cen-log-simple (Yanagisawa, 2023) (referred to as PSR) are used to assess discriminative performance. KS-cal is used as a marginal calibration metric to assess D-calibration, while KM-cal (Qi et al., 2024a) and IBS (Graf et al., 1999) are used to evaluate 1-calibration and overall predictive accuracy, respectively. In addition, $\mathrm{Cal}_{\mathrm{ws}}$ (Qi et al., 2024b) is used as a conditional D-calibration metric showing the calibration error in the worst-calibrated subregion of the covariate space. The detailed formulations of these metrics are provided in Appendix A.

5.2 RESULTS

Pairwise comparisons We evaluate six survival models across ten datasets, yielding 60 combinations. Table 1 summarizes the direction, frequency, and statistical significance of the pairwise comparisons between CoSC and the other methods. Figure 2 complements these counts by showing the magnitude and distribution of the differences between CoSC and the competing post-processing methods. Positive values favor CoSC, negative values favor the comparator, and the dashed horizontal line indicates equal performance.

CoSC outperforms each comparator in 52-58 of the 60 combinations for IBS and in 49-54 combinations for $\mathrm{Cal}_{\mathrm{ws}}$, including 50-57 and 32-47 statistically significant wins, respectively. The corresponding distributions are largely concentrated above zero. For KS-cal, CoSC outperforms CSD and CSD-iPOT in 52 and 49 combinations, respectively. Its comparison with KSP is evenly split at 30 wins and 30 losses, with differences centered close to zero. The advantage in KM-cal is less uniform, particularly against CSD, as reflected by the 36-24 win-loss count and the wide distribution of differences. This pattern is consistent with CSD's use of the KM estimator during calibration, which makes it competitive under the KM-cal criterion.

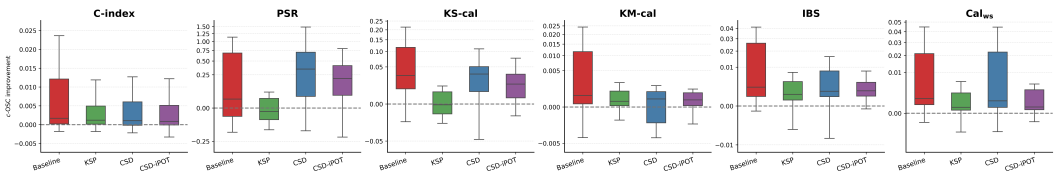

Figure 2: Pairwise performance differences between CoSC and competing post-processing methods across 60 combinations. Positive values favor CoSC, whereas negative values favor the comparator. The dashed line indicates equal performance.

CoSC achieves a higher C-index in 40-48 combinations against each comparator. For PSR, CoSC outperforms CSD and CSD-iPOT in 47 and 49 combinations, respectively, whereas KSP performs better in 33 of the 60 combinations.

In the Large group, CoSC has the best median value for all six metrics. This pattern is consistent with the simulation results in Appendix D, where increasing the calibration sample size reduces the kernel estimation error. Detailed group-wise results are provided in Table 5 in Appendix A.

Table 1: Pairwise comparisons of CoSC with the Baseline and competing post-processing methods across 60 combinations. Entries report wins / losses for CoSC, with the numbers of significant wins and losses in parentheses. Significant wins and losses are determined using one-sided paired t-tests in the corresponding direction at $\alpha = 0.05$.

Comparison	C-index	PSR	KS-cal	KM-cal	IBS	Cal_{ws}
vs Baseline	48(44) / 12(7)	38(37) / 22(18)	52(48) / 8(3)	48(45) / 12(8)	57(57) / 3(0)	54(47) / 6(2)
vs KSP	48(41) / 12(6)	27(23) / 33(30)	30(20) / 30(16)	49(41) / 11(11)	52(50) / 8(6)	49(32) / 11(3)
vs CSD	40(34) / 20(13)	47(45) / 13(12)	52(50) / 8(6)	36(36) / 24(21)	52(50) / 8(6)	51(34) / 9(1)
vs CSD-iPOT	42(35) / 18(10)	49(47) / 11(9)	49(44) / 11(4)	45(39) / 15(11)	58(54) / 2(1)	51(34) / 9(1)

Computational cost We measure the post-processing runtime of each method in seconds. We report the overall median runtime across the 60 combinations, as well as the median runtime within each dataset-size group. Full results are provided in Appendix I. Overall, the post-processing runtime tends to increase with dataset size. Although CoSC has a higher overall median runtime, its group-specific median runtimes range from 5.42 to 6.75 seconds. Moreover, for the Large group, CoSC is faster than both CSD and CSD-iPOT.

Table 2: Median post-processing runtime in seconds across 60 combinations and within each dataset-size group. Lower values indicate greater computational efficiency.

Method	Overall	Small	Medium	Large
KSP	1.6603	1.6196	2.0645	1.7465
CSD	3.2893	1.0994	3.1606	17.7882
CSD-iPOT	0.6906	0.1478	0.9855	8.6981
CoSC	5.5842	5.4176	6.7466	6.2123

Ablation Study We conduct an ablation study to examine four design choices in CoSC: (1) the number of cross-validation folds K, (2) the number of cross-validation repetitions B, (3) the cumulative explained variance threshold used for PCA, and (4) the kernel function. We also assess the contribution of covariate-dependent weighting by comparing CoSC with a marginal variant. Setting $h = \infty$ gives every observation equal weight, yielding a single calibration map based on the marginal D-calibration criterion. Using $K = 2$, $B = 10$, a 90% PCA threshold, and the Laplacian kernel as the default configuration, we vary one design choice at a time. For each configuration, including the marginal variant, we first average each metric over 30 repetitions within each dataset-model combination and then report the median across the 60 combinations.

Table 3 reports the resulting median performance. With $K = 2$ fixed, increasing B from 1 to 10 generally improves performance, particularly in KS-cal and IBS. Further increases yield only small

gains, so we use $B = 10$ in the main experiments. With $B = 10$ fixed, $K = 2$ provides the best C-index, PSR, and KS-cal among the settings considered, while additional folds yield small and inconsistent changes in KM-cal and IBS. The 90% PCA setting yields slightly lower median KS-cal, IBS, and $\mathrm{Cal}_{\mathrm{ws}}$ than using raw covariates, whereas raw covariates yield a slightly higher C-index and lower PSR. Thus, the ablation does not show a consistent benefit from PCA across metrics. The Laplacian kernel performs better than the Gaussian kernel on all metrics except KM-cal, for which the two kernels give identical values at the reported precision. Compared with the marginal variant ($h = \infty$), the default CoSC achieves better median performance across all six metrics. This result suggests that covariate-dependent correction improves overall predictive performance.

Table 3: Median performance of CoSC under alternative design choices and of its marginal variant ($h = \infty$), which assigns equal weight to all calibration observations. K and B denote the numbers of cross-validation folds and repetitions, respectively. Higher values are better for the C-index, whereas lower values are better for all other metrics.

Component	Setting	C-index	PSR	KS-cal	KM-cal	IBS	$\mathrm{Cal}_{\mathrm{ws}}$
Number of repetitions	$K = 2, B = 1$	0.7116	2.1105	0.0346	0.0012	0.1417	0.0013
	$K = 2, B = 2$	0.7117	2.1098	0.0342	0.0011	0.1415	0.0012
	$K = 2, B = 5$	0.7121	2.1074	0.0328	0.0012	0.1394	0.0012
	$K = 2, B = 10$	0.7121	2.1019	0.0324	0.0012	0.1374	0.0012
	$K = 2, B = 20$	0.7121	2.1022	0.0325	0.0012	0.1371	0.0012
	$K = 2, B = 50$	0.7121	2.1001	0.0326	0.0012	0.1371	0.0012
Number of folds	$K = 2, B = 10$	0.7121	2.1019	0.0324	0.0012	0.1374	0.0012
	$K = 3, B = 10$	0.7120	2.1025	0.0326	0.0012	0.1364	0.0012
	$K = 4, B = 10$	0.7120	2.1043	0.0327	0.0011	0.1382	0.0012
	$K = 5, B = 10$	0.7119	2.1068	0.0327	0.0011	0.1397	0.0013
Use of PCA	80%	0.7120	2.1053	0.0330	0.0012	0.1414	0.0013
	90%	0.7121	2.1019	0.0324	0.0012	0.1374	0.0012
	Not used	0.7122	2.0983	0.0331	0.0013	0.1408	0.0013
Kernel	Laplacian	0.7121	2.1019	0.0324	0.0012	0.1374	0.0012
	Gaussian	0.7120	2.1230	0.0347	0.0012	0.1417	0.0013
Marginal	$h = \infty$	0.7116	2.1025	0.0413	0.0013	0.1437	0.0014

Calibration plots Calibration plots complementing the quantitative results are provided in Appendix E.

6 Conclusion

We introduced CoSC, a nonparametric post-hoc procedure that converts the conditional D-calibration criterion into a covariate-dependent recalibration map. At the population level, we established exact one-step recovery of the true conditional CDF in the absence of censoring and convergence under repeated correction with right censoring. Across ten datasets and six survival models, the one-step estimator improved IBS and conditional calibration in most pairwise comparisons while showing a comparable or improved C-index in the majority of comparisons.

Several limitations suggest directions for future work. First, finite-sample error in the kernel estimator can be non-negligible, particularly when the sample size is small. Although our population-level analysis motivates iterative self-correction, each empirical iteration requires re-estimating the calibration map and re-selecting the bandwidth. Estimation errors may therefore propagate across iterations while computational costs increase, and our current implementation consequently uses a single correction step. More stable and sample-efficient estimators, together with principled stopping criteria, could make iterative correction practically viable. Second, PCA can introduce projection error, and our experiments do not show a consistent gain in overall accuracy from dimension reduction. Its Euclidean projection also does not explicitly preserve the structure of one-hot-encoded categorical covariates. Structure-aware dimension reduction or kernels designed for mixed-type covariates are potential alternatives.

AI USE STATEMENT

The research idea originated with the authors. We used ChatGPT to assist with parts of the mathematical proofs, improve the wording of the manuscript, and write code. The authors reviewed and corrected the AI-assisted mathematical arguments and text, and tested the AI-assisted code for correctness. We take responsibility for the final content of this work, including all claims, text, and code produced with AI assistance.

ETHICS STATEMENT

This work develops a post-hoc calibration method and does not involve new data collection, participant recruitment, or intervention with human subjects. The experiments use previously released benchmark datasets and de-identified or access-controlled clinical and cancer-registry data, including MIMIC-III and SEER, which were accessed and analyzed in accordance with their applicable licenses and data-use requirements. We do not redistribute restricted raw data or attempt to re-identify any individual. The source cohorts and baseline prediction models may nevertheless reflect demographic, institutional, and measurement biases, and improved calibration does not by itself guarantee fairness, clinical validity, or generalizability across populations. The proposed method is intended for methodological research and should not be used for clinical decision-making without external validation, subgroup-specific evaluation, and appropriate expert oversight.

REPRODUCIBILITY STATEMENT

We provide detailed information to support the reproducibility of both the theoretical and empirical results. Section 4 describes CoSC, and Appendix C provides the complete bandwidth-selection algorithm. The assumptions and proofs of the theoretical results are given in Appendix B, while the simulation design and additional analyses are reported in Appendix D. Evaluation metrics, baseline models, data sources, preprocessing procedures, complete experimental results, and optimization settings are documented in Appendices A and E-I. Section 5.1 specifies the data-splitting procedure, model configurations, and repeated-seed evaluation protocol. Source code is available at `https://github.com/anon-survival/CoSC`. Restricted raw datasets are not redistributed; Appendix G provides access information and the preprocessing procedures required after authorized access has been obtained.

REFERENCES

Todd A. Alonzo. Clinical prediction models: A practical approach to development, validation, and updating: By ewout w. steyerberg. *American Journal of Epidemiology*, 170(4):528, 2009. doi: 10.1093/aje/kwp129.

Anand Avati, Tony Duan, Sharon Zhou, Kenneth Jung, Nigam H. Shah, and Andrew Y. Ng. Countdown regression: sharp and calibrated survival predictions. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, pp. 145–155. PMLR, 2020.

Martin Bladt, Alexander Henzi, Bram van den Heuvel, and Johanna Ziegel. Survival isotonic distributional regression. arXiv preprint arXiv:2608.02914, 2026. URL `https://arxiv.org/abs/2608.02914`.

Emmanuel Candès, Lihua Lei, and Zhimei Ren. Conformalized survival analysis. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 85(1):24–45, 2023.

Paidamoyo Chapfuwa, Chenyang Tao, Chunyuan Li, Irfan Khan, Karen J. Chandross, Michael J. Pencina, Lawrence Carin, and Ricardo Henao. Calibration and uncertainty in neural time-to-event modeling. *IEEE Transactions on Neural Networks and Learning Systems*, 34(4):1666–1680, 2020.

David R. Cox. Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological)*, 34(2):187–202, 1972.

Cynthia S. Crowson, Elizabeth J. Atkinson, and Terry M. Therneau. Assessing calibration of prognostic risk scores. *Statistical Methods in Medical Research*, 25(4):1692–1706, 2016.

Hen Davidov, Shai Feldman, Gil Shamai, Ron Kimmel, and Yaniv Romano. Conformalized survival analysis for general right-censored data. In *the International Conference on Learning Representations*, 2025. https://openreview.net/forum?id=JQtuCumAFD.

Bradley Efron. The two sample problem with censored data. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 4, pp. 831–853, 1967.

Rebecca Farina, Eric J Tchetgen Tchetgen, and Arun Kumar Kuchibhotla. Doubly robust and efficient calibration of prediction sets for right-censored time-to-event outcomes. *CoRR*, abs/2501.04615, 2026.

Tilmann Gneiting and Johannes Resin. Regression diagnostics meets forecast evaluation: conditional calibration, reliability diagrams, and coefficient of determination. *Electronic Journal of Statistics*, 17(2):3226–3286, 2023.

Tilmann Gneiting, Fadoua Balabdaoui, and Adrian E Raftery. Probabilistic forecasts, calibration and sharpness. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(2): 243–268, 2007.

Mark Goldstein, Xintian Han, Aahlad Puli, Adler Perotte, and Rajesh Ranganath. X-cal: explicit calibration for survival analysis. In *Advances in Neural Information Processing Systems*, volume 33, pp. 18296–18307, 2020.

Erika Graf, Claudia Schmoor, Willi Sauerbrei, and Martin Schumacher. Assessment and comparison of prognostic classification schemes for survival data. *Statistics in Medicine*, 18(17-18):2529–2545, 1999.

Yu Gui, Rohan Hore, Zhimei Ren, and Rina Foygel Barber. Conformalized survival analysis with adaptive cut-offs. *Biometrika*, 111(2):459–477, 2024.

Humza Haider, Bret Hoehn, Sarah Davis, and Russell Greiner. Effective ways to build and evaluate individual survival distributions. *Journal of Machine Learning Research*, 21(1):3289–3351, 2020.

Frank E Harrell Jr, Kerry L Lee, and Daniel B Mark. Multivariable prognostic models: issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. *Statistics in Medicine*, 15(4):361–387, 1996.

Hrayr Harutyunyan, Hrant Khachatrian, David C. Kale, Greg Ver Steeg, and Aram Galstyan. Multitask learning and benchmarking with clinical time series data. *Scientific Data*, 6(1):96, 2019. doi: 10.1038/s41597-019-0103-9.

Úrsula Hébert-Johnson, Michael P Kim, Omer Reingold, and Guy N Rothblum. Multicalibration: Calibration for the (computationally-identifiable) masses. In *Proceedings of the International Conference on Machine Learning*, volume 80, pp. 1939–1948, 2018.

Alexander Henzi, Gian-Reto Kleger, and Johanna F Ziegel. Distributional (single) index models. *Journal of the American Statistical Association*, 118, 2021a.

Alexander Henzi, Johanna F Ziegel, and Tilmann Gneiting. Isotonic distributional regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 83, 2021b.

Ádám Jung, Domokos M Kelen, and András A Benczúr. Nonparametric distribution regression re-calibration. arXiv preprint arXiv:2602.13362, 2026. URL https://arxiv.org/abs/2602.13362.

Edward L Kaplan and Paul Meier. Nonparametric estimation from incomplete observations. *Journal of the American statistical association*, 53(282):457–481, 1958.

Jared L. Katzman, Uri Shaham, Alexander Cloninger, Jonathan Bates, Tingting Jiang, and Yuval Kluger. Deepsurv: Personalized treatment recommender system using a cox proportional hazards deep neural network. *BMC Medical Research Methodology*, 18(1):24, 2018.

Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. Accurate uncertainties for deep learning using calibrated regression. In *Proceedings of the International Conference on Machine Learning*, volume 80, pp. 2796–2804, 2018.

Changhee Lee, William Zame, Jinsung Yoon, and Mihaela Van Der Schaar. Deephit: A deep learning approach to survival analysis with competing risks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, pp. 2314–2321, 2018.

Dongjoon Lee, Hyeryn Park, and Changhee Lee. Toward a well-calibrated discrimination via survival outcome-aware contrastive learning. In *Advances in Neural Information Processing Systems*, volume 37, pp. 30985–31014, 2024.

Ruowen Li, Wenjie Qu, Qingqing Liu, Yilin Tan, Wenjing Zhang, Yiping Hao, Nan Jiang, Zhonghao Mao, Jinwen Ye, Jun Jiao, et al. Development and validation of a deep learning survival model for cervical adenocarcinoma patients. *BMC Bioinformatics*, 24(1):1–15, 2023.

D. Y. Lin. On the breslow estimator. *Lifetime Data Analysis*, 13(4):471–480, 2007. doi: 10.1007/s10985-007-9048-y.

Rachel Luo, Aadyot Bhatnagar, Yu Bai, Shengjia Zhao, Huan Wang, Caiming Xiong, Silvio Savarese, Stefano Ermon, Edward Schmerling, and Marco Pavone. Local calibration: metrics and recalibration. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, volume 180, pp. 1286–1295, 2022.

Charles Marx, Sofian Zalouk, and Stefano Ermon. Calibration by distribution matching: Trainable kernel calibration metrics, 2023. URL https://arxiv.org/abs/2310.20211.

Jeongho Park, Sangwook Kang, and Gwangsu Kim. Stochastic explicit calibration algorithm for survival models. *IEEE Access*, 13:78679–78706, 2025a.

Jeongho Park, Daheen Kim, Cheoljun Kim, Hyungbin Park, Sangwook Kang, and Gwangsu Kim. Ksp: Kolmogorov-smirnov metric-based post-hoc calibration for survival analysis. In *Advances in Neural Information Processing Systems*, 2025b.

Shi-Ang Qi, Yakun Yu, and Russell Greiner. Conformalized survival distributions: A generic post-process to increase calibration. In *Proceedings of the International Conference on Machine Learning*, volume 235, pp. 41303–41339. PMLR, 2024a.

Shi-Ang Qi, Yakun Yu, and Russell Greiner. Toward conditional distribution calibration in survival prediction. In *Advances in Neural Information Processing Systems*, volume 37, pp. 86180–86225, 2024b.

Jing Qin, Jin Piao, Jing Ning, and Yu Shen. Conformal predictive intervals in survival analysis: a resampling approach. *Biometrics*, 81(2):ujaf063, 2025.

Aaron Roth. Uncertain: Modern Topics in Uncertainty Estimation. Lecture notes, University of Pennsylvania, 2022. URL https://www.cis.upenn.edu/~aaroth/uncertainty-notes.pdf. Working draft.

Deming Sheng and Ricardo Henao. From individual calibration to reliable classifiers: ALD parameterization with mPAIC guarantees. In *Proceedings of the International Conference on Machine Learning*, volume 306, 2026.

Hao Song, Tom Diethe, Meelis Kull, and Peter Flach. Distribution calibration for regression. In *Proceedings of the International Conference on Machine Learning*, volume 97, pp. 5897–5906, 2019.

Winfried Stute. Consistent estimation under random censorship when covariables are present. *Journal of Multivariate Analysis*, 45(1):89–103, 1993.

Simon Wiegrebe, Philipp Kopper, Raphael Sonabend, Bernd Bischl, and Andreas Bender. Deep learning for survival analysis: a review. *Artificial Intelligence Review*, 57(3):65, 2024.

Yi Xia, Baifu Zhang, and Yongliang Zhang. Deep survival analysis using pseudo values and its application to predict the recurrence of stage iv colorectal cancer after tumor resection. *Computer Methods in Biomechanics and Biomedical Engineering*, 27(15):1–10, 2023.

Lizhao Yan, Nan Gao, Fangxing Ai, Yingsong Zhao, Yu Kang, Jianghai Chen, and Yuxiong Weng. Deep learning models for predicting the survival of patients with chondrosarcoma based on a surveillance, epidemiology, and end results analysis. *Frontiers in Oncology*, 12:967758, 2022.

Hiroki Yanagisawa. Proper scoring rules for survival analysis. In *Proceedings of the International Conference on Machine Learning*, volume 202, pp. 39165–39182. PMLR, 2023.

Chun-Nam Yu, Russell Greiner, Hsiu-Chin Lin, and Vickie Baracos. Learning patient-specific cancer survival distributions as a sequence of dependent regressors. In *Advances in Neural Information Processing Systems*, volume 24, pp. 1845–1853, 2011.

Shengjia Zhao, Tengyu Ma, and Stefano Ermon. Individual calibration with randomized forecasting. In *Proceedings of the International Conference on Machine Learning*, volume 119, pp. 11387–11397, 2020.

APPENDIX

A EVALUATION METRICS AND ADDITIONAL EXPERIMENTAL RESULTS

A.1 DEFINITIONS OF EVALUATION METRICS

We provide detailed definitions of the metrics used in our experiments. PSR is a proper scoring rule-based discrimination metric proposed by Yanagisawa (2023). It is defined as

$$-\frac{1}{n}\sum_{i=1}^{n}\sum_{j=0}^{B-1}\mathbb{I}\left(\zeta_j < y_i \le \zeta_{j+1}\right)\left\{\delta_i \log \hat{f}_{ij} + (1-\delta_i)\log\left(1-\hat{F}_{\boldsymbol{z}_i}\left(\zeta_{j+1}\right)\right)\right\}$$

where $\hat{f}_{ij} = \hat{F}_{\boldsymbol{z}_i}\left(\zeta_{j+1}\right) - \hat{F}_{\boldsymbol{z}_i}\left(\zeta_j\right)$ and $\zeta_j = \frac{j}{B}t_{\max}$ with $B = 32$. Here, $t_{\max}$ denotes the maximum observed time in the dataset.

The remaining metrics are defined as follows:

$$\text{KS-cal} = \sup_{x\in[0,1]}\left|\frac{1}{n}\sum_{i=1}^{n}\mathbb{I}\left(\hat{F}_{\boldsymbol{z}_i}\left(y_i\right)\le x\right)\left\{\delta_i + (1-\delta_i)\frac{x-\hat{F}_{\boldsymbol{z}_i}\left(y_i\right)}{1-\hat{F}_{\boldsymbol{z}_i}\left(y_i\right)}\right\} - x\right|$$

$$\text{KM-cal} = \frac{1}{t_{\max}}\int_0^{t_{\max}}\left(S_{\mathrm{KM}}\left(t\right) - \frac{1}{n}\sum_{i=1}^{n}\hat{S}_{\boldsymbol{z}_i}\left(t\right)\right)^2 dt$$

$$\text{IBS} = \frac{1}{t_{\max}}\int_0^{t_{\max}}\mathrm{BS}\left(t\right)dt \quad \text{where}$$

$$\mathrm{BS}\left(t\right) = \frac{1}{n}\sum_{i=1}^{n}\left[\frac{\mathbb{I}\left(y_i\le t,\delta_i=1\right)\left(0-\hat{S}_{\boldsymbol{z}_i}\left(t\right)\right)^2}{\hat{G}\left(y_i\right)} + \frac{\mathbb{I}\left(y_i > t\right)\left(1-\hat{S}_{\boldsymbol{z}_i}\left(t\right)\right)^2}{\hat{G}\left(t\right)}\right]$$

Here $S_{\mathrm{KM}}\left(t\right)$ denotes the Kaplan-Meier estimator (Kaplan & Meier, 1958) of the survival function of T, $\hat{S}_{\boldsymbol{z}} = 1 - \hat{F}_{\boldsymbol{z}}$, and $\hat{G}$ is the Kaplan-Meier estimator of the censoring distribution. For details on approximating the supremum in KS-cal, we refer to Park et al. (2025b).

Following Qi et al. (2024b), we randomly split the test set into an exploration set $\mathcal{D}_{\mathrm{explore}}$ (25%) and an exploitation set $\mathcal{D}_{\mathrm{exploit}}$ (75%). For a nonempty subset $\mathcal{A}$, define

$$\tilde{F}_{\mathcal{A}}\left(u\right) = \frac{1}{|\mathcal{A}|}\sum_{i\in\mathcal{A}}\mathbb{I}\left(\hat{F}_{\boldsymbol{z}_i}\left(y_i\right)\le u\right)\left\{\delta_i + (1-\delta_i)\frac{u-\hat{F}_{\boldsymbol{z}_i}\left(y_i\right)}{1-\hat{F}_{\boldsymbol{z}_i}\left(y_i\right)}\right\},$$

and its calibration error as

$$\mathrm{Cal}(\mathcal{A}) = \frac{1}{|\mathcal{P}|}\sum_{u\in\mathcal{P}}\left\{\tilde{F}_{\mathcal{A}}\left(u\right) - u\right\}^2,$$

where $\mathcal{P} = \{0.1, 0.2, \ldots, 0.9\}$.

A slab is defined by

$$\mathbb{S}_{\boldsymbol{v},a,b} = \left\{\boldsymbol{z}\in\mathbb{R}^d : a \le \boldsymbol{v}^\top\boldsymbol{z} \le b\right\}.$$

Let $\mathcal{D}\left(\mathbb{S}\right)$ denote the observations in $\mathcal{D}$ whose covariates belong to $\mathbb{S}$. We select

$$(\hat{\boldsymbol{v}},\hat{a},\hat{b}) \in \underset{\boldsymbol{v}\in\mathcal{V},\, a<b}{\arg\max}\,\mathrm{Cal}\left(\mathcal{D}_{\mathrm{explore}}\left(\mathbb{S}_{\boldsymbol{v},a,b}\right)\right) \quad \text{where } \left|\mathcal{D}_{\mathrm{explore}}\left(\mathbb{S}_{\boldsymbol{v},a,b}\right)\right| \ge \kappa|\mathcal{D}_{\mathrm{explore}}|$$

with $\kappa = 0.33$, and $\mathcal{V}$ contains 1,000 candidate directions sampled uniformly from the unit sphere. We search for the slab boundaries among the projected covariate values in the exploration set.

The selected slab is then evaluated on the exploitation set:

$$\mathrm{Cal}_{\mathrm{ws}} = \mathrm{Cal}\left(\mathcal{D}_{\mathrm{exploit}}\left(\mathbb{S}_{\hat{\boldsymbol{v}},\hat{a},\hat{b}}\right)\right).$$

Lower values indicate better calibration within the selected subgroup.

A.2 Overall Median Performance

We summarize the performance of each method using the median across the 60 dataset-model combinations, after averaging each metric over 30 repetitions within each combination. As shown in Table 4, CoSC achieves the best overall median performance across all six metrics. However, these aggregate medians do not directly capture paired differences within individual combinations and should be interpreted alongside the pairwise comparisons in the main manuscript.

Table 4: Median performance of each method for 60 combinations.

Method	C-index	PSR	KS-cal	KM-cal	IBS	$\mathbf{Cal_{ws}}$
Baseline	0.7076	2.2113	0.0588	0.002483	0.1523	0.002272
KSP	0.7116	2.1192	0.0379	0.002044	0.1438	0.001641
CSD	0.7114	2.4052	0.0605	0.001192	0.1520	0.002282
CSD-iPOT	0.7116	2.3596	0.0454	0.001669	0.1440	0.001437
CoSC	**0.7121**	**2.1019**	**0.0324**	**0.001158**	**0.1374**	**0.001186**

A.3 Comparison by sample size

In this subsection, we present a comparison of calibration methods across sample sizes.

Table 5 shows that the relative performance of the methods varies across dataset-size groups. In the Small group, CoSC achieves the best median C-index, IBS, and $\mathrm{Cal_{ws}}$, whereas KSP performs best in PSR and KS-cal, and CSD in KM-cal. In the Medium group, KSP leads in four metrics, while CoSC achieves the lowest $\mathrm{Cal_{ws}}$, with only a small difference from KSP.

In the Large group, CoSC has the best median value for all six metrics. Across the three groups it attains the lowest median $\mathrm{Cal_{ws}}$ in every group and the lowest median IBS in the Small and Large groups, ranking second in the Medium group.

Table 5: Median performance of each method across sample sizes.

Group	Method	C-index	PSR	KS-cal	KM-cal	IBS	$\mathbf{Cal_{ws}}$
Small	Baseline	0.6749	2.3463	0.0815	0.005983	0.1718	0.006665
	KSP	0.6743	**2.2003**	**0.0428**	0.003437	0.1621	0.002220
	CSD	0.6747	2.4052	0.0613	**0.001866**	0.1701	0.004251
	CSD-iPOT	0.6748	2.2959	0.0449	0.003003	0.1654	0.002252
	CoSC	**0.7060**	2.2356	0.0445	0.003235	**0.1595**	**0.002056**
Medium	Baseline	0.6609	1.8451	0.0601	0.000730	0.1174	0.002003
	KSP	0.6621	**1.7252**	**0.0288**	**0.000308**	**0.0449**	0.001126
	CSD	0.6639	2.1759	0.0700	0.000716	0.0592	0.001833
	CSD-iPOT	**0.6663**	2.1471	0.0783	0.000948	0.0493	0.001290
	CoSC	0.6650	1.7811	0.0311	0.000490	0.0471	**0.001123**
Large	Baseline	0.7328	2.1790	0.0439	0.001375	0.1291	0.001308
	KSP	0.7336	2.1772	0.0205	0.001375	0.1297	0.001028
	CSD	0.7337	2.7616	0.0538	0.001098	0.1272	0.001114
	CSD-iPOT	0.7337	2.6042	0.0405	0.001132	0.1285	0.000478
	CoSC	**0.7346**	**2.1562**	**0.0167**	**0.000694**	**0.1250**	**0.000254**

A.4 MODEL-WISE COMPARISON

In this subsection, we present a comparison of calibration methods across different models. Table 6 compares the median performance of the calibration methods across ten datasets for each survival model. CoSC achieves the lowest median $\mathrm{Cal}_{\mathrm{ws}}$ for all six models and the lowest median IBS for four models, ranking second for DeepSurv and DeepHit.

For MTLR and S-CRPS, CoSC has the best median value in five of the six metrics. The exceptions are KM-cal for MTLR, where CSD performs best, and PSR for S-CRPS, where KSP performs best. For S-CRPS, which exhibits substantial calibration errors before post-processing, CoSC also improves on the competing methods across all three calibration metrics and IBS.

For DeepSurv and LognormalNN, KSP has the lowest PSR and KS-cal, while CoSC provides the lowest $\mathrm{Cal}_{\mathrm{ws}}$. For DeepHit, CoSC achieves the best PSR, KS-cal, and $\mathrm{Cal}_{\mathrm{ws}}$. CSD has the lowest IBS (0.1354), followed by CoSC (0.1356), although CoSC has a slightly lower C-index than the other methods.

Table 6: Median performance of each method across ten datasets.

Model	Method	C-index	PSR	KS-cal	KM-cal	IBS	$\mathbf{Cal_{ws}}$
DeepSurv	Baseline	0.7238	2.0427	0.0336	**0.000042**	0.1317	0.000995
	KSP	0.7238	**2.0350**	**0.0229**	0.000140	**0.1310**	0.000990
	CSD	0.7238	2.3462	0.0611	0.000575	0.1369	0.001564
	CSD-iPOT	0.7238	2.3345	0.0383	0.000217	0.1316	0.000950
	CoSC	0.7238	2.0522	0.0248	0.000074	0.1312	**0.000655**
MTLR	Baseline	0.7043	2.2104	0.0503	0.002849	0.1439	0.001794
	KSP	0.7043	2.1525	0.0412	0.002487	0.1458	0.001795
	CSD	0.7043	2.2491	0.0413	**0.000809**	0.1385	0.001597
	CSD-iPOT	0.7043	2.3471	0.0405	0.002624	0.1423	0.001420
	CoSC	**0.7080**	**2.1362**	**0.0340**	0.001907	**0.1377**	**0.001413**
LognormalNN	Baseline	0.7194	2.0962	0.0534	0.001870	0.1365	0.002773
	KSP	0.7235	**2.0766**	**0.0293**	**0.000616**	0.1369	0.001242
	CSD	**0.7236**	2.4457	0.0633	0.001222	0.1511	0.003028
	CSD-iPOT	0.7235	2.4125	0.0432	0.000925	0.1371	0.001252
	CoSC	0.7235	2.0823	0.0348	0.000661	**0.1357**	**0.000976**
S-CRPS	Baseline	0.6855	4.9149	0.5298	0.213916	0.3793	0.180290
	KSP	0.7037	**2.2364**	0.0723	0.002938	0.1507	0.011166
	CSD	0.7002	3.2728	0.1235	0.026478	0.1867	0.080548
	CSD-iPOT	0.7027	2.2909	0.0951	0.002834	0.1660	0.011591
	CoSC	**0.7043**	2.2523	**0.0485**	**0.002424**	**0.1430**	**0.003697**
DeepHit	Baseline	**0.7258**	2.1907	0.0527	0.003007	0.1380	0.001723
	KSP	0.7256	2.1247	0.0394	0.002521	0.1381	0.001068
	CSD	**0.7258**	2.2375	0.0432	**0.001000**	**0.1354**	0.001104
	CSD-iPOT	0.7256	2.3012	0.0420	0.002700	0.1373	0.000886
	CoSC	0.7250	**2.1124**	**0.0321**	0.001902	0.1356	**0.000811**
AFT	Baseline	0.5991	2.2729	0.0676	**0.001091**	0.1791	0.010507
	KSP	0.5991	2.2624	**0.0336**	0.003066	0.1473	0.006600
	CSD	0.5991	2.7337	0.0585	0.001098	0.1479	0.008695
	CSD-iPOT	0.5991	2.6067	0.0449	0.002547	0.1476	0.006933
	CoSC	**0.6870**	**2.0896**	0.0355	0.002794	**0.1377**	**0.003759**

B THEORETICAL RESULTS AND PROOFS

Assumptions

A.1. $T \perp\!\!\!\perp C \mid \boldsymbol{Z} = \boldsymbol{z}$.

A.2. $F_{\boldsymbol{z}}$ and $G_{\boldsymbol{z}}$ are continuous and strictly increasing on $[0, \tau]$, with supports $[0, \infty)$.

A.3. $\hat{F}_{\boldsymbol{z}}$ is continuous and strictly increasing on $[0, \tau]$, with $\hat{F}_{\boldsymbol{z}}(0) = 0$.

A.4. $\tau > 0$ is a fixed finite time horizon satisfying the following conditions:

1. $\mathbb{P}(Y > \tau \mid \boldsymbol{Z} = \boldsymbol{z}) > 0$.
2. $\hat{F}_{\boldsymbol{z}}(\tau) < 1$.

Remark. The horizon τ does not represent the study end time. Nevertheless, it can be chosen close to that time, provided that the stated assumptions hold.

B.1 ERROR BOUND FOR A SINGLE CORRECTION

We first consider the case $k = 0$. Let

$$H_{\boldsymbol{z}}(x) = \mathbb{E}\left[\mathbb{I}\left(\hat{F}_{\boldsymbol{Z}}(Y) \le x\right)\left\{\delta + (1-\delta)\frac{x - \hat{F}_{\boldsymbol{Z}}(Y)}{1 - \hat{F}_{\boldsymbol{Z}}(Y)}\right\} \middle| \boldsymbol{Z} = \boldsymbol{z}\right]$$

Then,

$$H_{\boldsymbol{z}}(x) = \mathbb{E}\left[\mathbb{I}\left(\hat{F}_{\boldsymbol{Z}}(Y) \le x\right)\left\{\delta + (1-\delta)\frac{x - \hat{F}_{\boldsymbol{Z}}(Y)}{1 - \hat{F}_{\boldsymbol{Z}}(Y)}\right\}\mathbb{I}(\delta = 1) \middle| \boldsymbol{Z} = \boldsymbol{z}\right] \tag{4}$$

$$+ \mathbb{E}\left[\mathbb{I}\left(\hat{F}_{\boldsymbol{Z}}(Y) \le x\right)\left\{\delta + (1-\delta)\frac{x - \hat{F}_{\boldsymbol{Z}}(Y)}{1 - \hat{F}_{\boldsymbol{Z}}(Y)}\right\}\mathbb{I}(\delta = 0) \middle| \boldsymbol{Z} = \boldsymbol{z}\right] \tag{5}$$

The first term (4) can be written as

$$\begin{aligned}
&\mathbb{E}\left[\mathbb{I}\left(\hat{F}_{\boldsymbol{Z}}(T) \le x\right)\mathbb{I}(T \le C) \middle| \boldsymbol{Z} = \boldsymbol{z}\right] \\
&= \mathbb{E}\left[\mathbb{I}\left(T \le \hat{F}_{\boldsymbol{Z}}^{-1}(x)\right)\mathbb{I}(T \le C) \middle| \boldsymbol{Z} = \boldsymbol{z}\right] \\
&= \int_0^{\hat{F}_{\boldsymbol{z}}^{-1}(x)} [1 - G_{\boldsymbol{z}}(s)]\, dF_{\boldsymbol{z}}(s) \\
&= \int_0^{x} \left[1 - G_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(s)\right)\right] dF_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(s)\right) \\
&= \left[1 - G_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(x)\right)\right] F_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(x)\right) + \int_0^{x} F_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(s)\right) dG_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(s)\right).
\end{aligned}$$

The last equality follows from integration by parts. The second term (5) is

$$\begin{aligned}
&\mathbb{E}\left[\mathbb{I}\left(\hat{F}_{\boldsymbol{Z}}(C) \le x\right)\mathbb{I}(C < T)\frac{x - \hat{F}_{\boldsymbol{Z}}(C)}{1 - \hat{F}_{\boldsymbol{Z}}(C)} \middle| \boldsymbol{Z} = \boldsymbol{z}\right] \\
&= \mathbb{E}\left[\mathbb{I}\left(C \le \hat{F}_{\boldsymbol{Z}}^{-1}(x)\right)\mathbb{I}(C < T)\frac{x - \hat{F}_{\boldsymbol{Z}}(C)}{1 - \hat{F}_{\boldsymbol{Z}}(C)} \middle| \boldsymbol{Z} = \boldsymbol{z}\right] \\
&= \int_0^{x} \left[1 - F_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(s)\right)\right] \frac{x - s}{1 - s} dG_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(s)\right).
\end{aligned}$$

Combining these two results, we get

$$\begin{aligned}
&F_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(x)\right) \\
&+\int_0^x\left[F_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(s)\right)-F_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(x)\right)+\left\{1-F_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(s)\right)\right\}\frac{x-s}{1-s}\right]dG_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(s)\right) \\
&=F_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(x)\right) \\
&-\int_0^x\left\{F_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(x)\right)-x\right\}dG_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(s)\right)+\int_0^x\left\{F_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(s)\right)-s\right\}\frac{1-x}{1-s}dG_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(s)\right)
\end{aligned}$$

Substituting $x=\hat{F}_{\boldsymbol{z}}(t)$ gives

$$\begin{aligned}
&H_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}(t)\right) \\
=\;&F_{\boldsymbol{z}}(t)-\int_0^{\hat{F}_{\boldsymbol{z}}(t)}\left\{F_{\boldsymbol{z}}(t)-\hat{F}_{\boldsymbol{z}}(t)\right\}dG_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(s)\right) \\
&+\int_0^{\hat{F}_{\boldsymbol{z}}(t)}\left\{F_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(s)\right)-s\right\}\frac{1-\hat{F}_{\boldsymbol{z}}(t)}{1-s}dG_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}^{-1}(s)\right) \\
=\;&F_{\boldsymbol{z}}(t)-\int_0^t\left\{F_{\boldsymbol{z}}(t)-\hat{F}_{\boldsymbol{z}}(t)\right\}dG_{\boldsymbol{z}}(s)+\int_0^t\left\{F_{\boldsymbol{z}}(s)-\hat{F}_{\boldsymbol{z}}(s)\right\}\frac{1-\hat{F}_{\boldsymbol{z}}(t)}{1-\hat{F}_{\boldsymbol{z}}(s)}dG_{\boldsymbol{z}}(s)
\end{aligned}$$

In the uncensored setting, we may take $C=\infty$, so that $G_{\boldsymbol{z}}(t)=0$ for $t\in[0,\tau]$. The preceding identity therefore yields

$$H_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}(t)\right)=F_{\boldsymbol{z}}(t),$$

which proves Proposition 1.

In the censored case, we compute the difference between $H_{\boldsymbol{z}}(\hat{F}_{\boldsymbol{z}}(t))$ and $F_{\boldsymbol{z}}(t)$. With censoring, the transformed CDF generally differs from the true conditional CDF.

$$\begin{aligned}
H_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}(t)\right)-F_{\boldsymbol{z}}(t)&=\left\{\hat{F}_{\boldsymbol{z}}(t)-F_{\boldsymbol{z}}(t)\right\}G_{\boldsymbol{z}}(t)-\int_0^t\left\{\hat{F}_{\boldsymbol{z}}(s)-F_{\boldsymbol{z}}(s)\right\}\frac{1-\hat{F}_{\boldsymbol{z}}(t)}{1-\hat{F}_{\boldsymbol{z}}(s)}dG_{\boldsymbol{z}}(s) \\
&=\left\{1-\hat{F}_{\boldsymbol{z}}(t)\right\}\left[\frac{\hat{F}_{\boldsymbol{z}}(t)-F_{\boldsymbol{z}}(t)}{1-\hat{F}_{\boldsymbol{z}}(t)}G_{\boldsymbol{z}}(t)-\int_0^t\frac{\hat{F}_{\boldsymbol{z}}(s)-F_{\boldsymbol{z}}(s)}{1-\hat{F}_{\boldsymbol{z}}(s)}dG_{\boldsymbol{z}}(s)\right]
\end{aligned}$$

Although this difference need not vanish, it admits the following error bound.

Let $\hat{S}_{\boldsymbol{z}}(t)=1-\hat{F}_{\boldsymbol{z}}(t)$ and define

$$r_{\boldsymbol{z}}(t)\triangleq\frac{\hat{F}_{\boldsymbol{z}}(t)-F_{\boldsymbol{z}}(t)}{\hat{S}_{\boldsymbol{z}}(t)}\text{ and }\bar{r}_{\boldsymbol{z}}(t)\triangleq\frac{1}{G_{\boldsymbol{z}}(t)}\int_0^t r_{\boldsymbol{z}}(s)\,dG_{\boldsymbol{z}}(s).$$

Here, we set $\bar{r}_{\boldsymbol{z}}(t)=0$ whenever $G_{\boldsymbol{z}}(t)=0$. Then,

$$H_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}(t)\right)-F_{\boldsymbol{z}}(t)=\hat{S}_{\boldsymbol{z}}(t)\left[r_{\boldsymbol{z}}(t)\,G_{\boldsymbol{z}}(t)-\bar{r}_{\boldsymbol{z}}(t)\,G_{\boldsymbol{z}}(t)\right]=G_{\boldsymbol{z}}(t)\,\hat{S}_{\boldsymbol{z}}(t)\left[r_{\boldsymbol{z}}(t)-\bar{r}_{\boldsymbol{z}}(t)\right].$$

Consequently, for the horizon τ specified in Assumption A.4,

$$\int_0^\tau\left|H_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}(t)\right)-F_{\boldsymbol{z}}(t)\right|^2dG_{\boldsymbol{z}}(t)=\int_0^\tau\left\{G_{\boldsymbol{z}}(t)\right\}^2\left\{\hat{S}_{\boldsymbol{z}}(t)\right\}^2\left|r_{\boldsymbol{z}}(t)-\bar{r}_{\boldsymbol{z}}(t)\right|^2dG_{\boldsymbol{z}}(t)$$

By definition of $\bar{r}_{\boldsymbol{z}}(t)$, we have the following

$$\begin{aligned}
d\left(G_{\boldsymbol{z}}(t)\,\bar{r}_{\boldsymbol{z}}(t)\right)&=r_{\boldsymbol{z}}(t)\,dG_{\boldsymbol{z}}(t) \\
\bar{r}_{\boldsymbol{z}}(t)\,dG_{\boldsymbol{z}}(t)+G_{\boldsymbol{z}}(t)\,d\left(\bar{r}_{\boldsymbol{z}}(t)\right)&=r_{\boldsymbol{z}}(t)\,dG_{\boldsymbol{z}}(t) \\
G_{\boldsymbol{z}}(t)\,d\left(\bar{r}_{\boldsymbol{z}}(t)\right)&=\left[r_{\boldsymbol{z}}(t)-\bar{r}_{\boldsymbol{z}}(t)\right]dG_{\boldsymbol{z}}(t)
\end{aligned}$$

and we have

$$\begin{aligned} d\left(G_{\boldsymbol{z}}(t)\left\{\bar{r}_{\boldsymbol{z}}(t)\right\}^2\right) &= \left\{\bar{r}_{\boldsymbol{z}}(t)\right\}^2 dG_{\boldsymbol{z}}(t) + 2\bar{r}_{\boldsymbol{z}}(t)\, G_{\boldsymbol{z}}(t)\, d\left(\bar{r}_{\boldsymbol{z}}(t)\right) \\ &= \left\{\bar{r}_{\boldsymbol{z}}(t)\right\}^2 dG_{\boldsymbol{z}}(t) + 2\bar{r}_{\boldsymbol{z}}(t)\left[r_{\boldsymbol{z}}(t) - \bar{r}_{\boldsymbol{z}}(t)\right] dG_{\boldsymbol{z}}(t) \\ &= \left[2\bar{r}_{\boldsymbol{z}}(t)\, r_{\boldsymbol{z}}(t) - \left\{\bar{r}_{\boldsymbol{z}}(t)\right\}^2\right] dG_{\boldsymbol{z}}(t) \\ \Rightarrow \left|r_{\boldsymbol{z}}(t) - \bar{r}_{\boldsymbol{z}}(t)\right|^2 dG_{\boldsymbol{z}}(t) &= \left\{r_{\boldsymbol{z}}(t)\right\}^2 dG_{\boldsymbol{z}}(t) - 2\bar{r}_{\boldsymbol{z}}(t)\, r_{\boldsymbol{z}}(t)\, dG_{\boldsymbol{z}}(t) + \left\{\bar{r}_{\boldsymbol{z}}(t)\right\}^2 dG_{\boldsymbol{z}}(t) \\ &= \left\{r_{\boldsymbol{z}}(t)\right\}^2 dG_{\boldsymbol{z}}(t) - d\left(G_{\boldsymbol{z}}(t)\left\{\bar{r}_{\boldsymbol{z}}(t)\right\}^2\right) \end{aligned}$$

By the monotonicity of $G_{\boldsymbol{z}}$,

$$\begin{aligned} &\int_0^\tau \left\{G_{\boldsymbol{z}}(t)\right\}^2 \left\{\hat{S}_{\boldsymbol{z}}(t)\right\}^2 \left|r_{\boldsymbol{z}}(t) - \bar{r}_{\boldsymbol{z}}(t)\right|^2 dG_{\boldsymbol{z}}(t) \\ &\leq \left\{G_{\boldsymbol{z}}(\tau)\right\}^2 \int_0^\tau \left\{\hat{S}_{\boldsymbol{z}}(t)\right\}^2 \left|r_{\boldsymbol{z}}(t) - \bar{r}_{\boldsymbol{z}}(t)\right|^2 dG_{\boldsymbol{z}}(t) \\ &= \left\{G_{\boldsymbol{z}}(\tau)\right\}^2 \int_0^\tau \left\{\hat{S}_{\boldsymbol{z}}(t)\right\}^2 \left[\left\{r_{\boldsymbol{z}}(t)\right\}^2 dG_{\boldsymbol{z}}(t) - d\left(G_{\boldsymbol{z}}(t)\left\{\bar{r}_{\boldsymbol{z}}(t)\right\}^2\right)\right] \\ &= \left\{G_{\boldsymbol{z}}(\tau)\right\}^2 \int_0^\tau \left\{\hat{S}_{\boldsymbol{z}}(t)\right\}^2 \left\{r_{\boldsymbol{z}}(t)\right\}^2 dG_{\boldsymbol{z}}(t) - \left\{G_{\boldsymbol{z}}(\tau)\right\}^2 \int_0^\tau \left\{\hat{S}_{\boldsymbol{z}}(t)\right\}^2 d\left(G_{\boldsymbol{z}}(t)\left\{\bar{r}_{\boldsymbol{z}}(t)\right\}^2\right) \end{aligned}$$

By integration by parts and the fact that $\hat{S}_{\boldsymbol{z}}$ is non-increasing,

$$\begin{aligned} &\int_0^\tau \left\{\hat{S}_{\boldsymbol{z}}(t)\right\}^2 d\left(G_{\boldsymbol{z}}(t)\left\{\bar{r}_{\boldsymbol{z}}(t)\right\}^2\right) \\ &= \left\{\hat{S}_{\boldsymbol{z}}(\tau)\right\}^2 G_{\boldsymbol{z}}(\tau)\left\{\bar{r}_{\boldsymbol{z}}(\tau)\right\}^2 - \int_0^\tau G_{\boldsymbol{z}}(t)\left\{\bar{r}_{\boldsymbol{z}}(t)\right\}^2 d\left(\left\{\hat{S}_{\boldsymbol{z}}(t)\right\}^2\right) \\ &\geq -\int_0^\tau G_{\boldsymbol{z}}(t)\left\{\bar{r}_{\boldsymbol{z}}(t)\right\}^2 d\left(\left\{\hat{S}_{\boldsymbol{z}}(t)\right\}^2\right) \geq 0. \end{aligned}$$

Finally, the following results are obtained.

$$\begin{aligned} \int_0^\tau \left|H_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}(t)\right) - F_{\boldsymbol{z}}(t)\right|^2 dG_{\boldsymbol{z}}(t) &\leq \left\{G_{\boldsymbol{z}}(\tau)\right\}^2 \int_0^\tau \left\{\hat{S}_{\boldsymbol{z}}(t)\right\}^2 \left\{r_{\boldsymbol{z}}(t)\right\}^2 dG_{\boldsymbol{z}}(t) \\ &= \left\{G_{\boldsymbol{z}}(\tau)\right\}^2 \int_0^\tau \left|\hat{F}_{\boldsymbol{z}}(t) - F_{\boldsymbol{z}}(t)\right|^2 dG_{\boldsymbol{z}}(t) \\ \therefore \left(\int_0^\tau \left|H_{\boldsymbol{z}}\left(\hat{F}_{\boldsymbol{z}}(t)\right) - F_{\boldsymbol{z}}(t)\right|^2 dG_{\boldsymbol{z}}(t)\right)^{1/2} &\leq G_{\boldsymbol{z}}(\tau)\left(\int_0^\tau \left|\hat{F}_{\boldsymbol{z}}(t) - F_{\boldsymbol{z}}(t)\right|^2 dG_{\boldsymbol{z}}(t)\right)^{1/2} \end{aligned}$$

B.2 Convergence under repeated correction

Using the results of the previous subsection, we obtain the following recurrence relation. Starting from $\hat{F}_{\boldsymbol{z}}^{(0)} = \hat{F}_{\boldsymbol{z}}$, we define, for $k = 0, 1, \ldots$,

$$H_{\boldsymbol{z}}^{(k)}(x) = \mathbb{E}\left[\mathbb{I}\left(\hat{F}_{\boldsymbol{Z}}^{(k)}(Y) \leq x\right)\left\{\delta + (1-\delta)\frac{x - \hat{F}_{\boldsymbol{Z}}^{(k)}(Y)}{1 - \hat{F}_{\boldsymbol{Z}}^{(k)}(Y)}\right\} \middle| \boldsymbol{Z} = \boldsymbol{z}\right],$$

so that $H_{\boldsymbol{z}}^{(0)} = H_{\boldsymbol{z}}$. The CDF is then updated as

$$\hat{F}_{\boldsymbol{z}}^{(k+1)}(t) = H_{\boldsymbol{z}}^{(k)}\left(\hat{F}_{\boldsymbol{z}}^{(k)}(t)\right).$$

By Lemma 1, $\hat{F}_{\boldsymbol{z}}^{(k)}$ is strictly increasing for every $k \geq 0$. Applying the same argument as in the case $k = 0$, we obtain the following inequality.

$$\begin{aligned}
\left(\int_0^\tau \left|\hat{F}_{\boldsymbol{z}}^{(k+1)}(t) - F_{\boldsymbol{z}}(t)\right|^2 dG_{\boldsymbol{z}}(t)\right)^{1/2} &\leq G_{\boldsymbol{z}}(\tau)\left(\int_0^\tau \left|\hat{F}_{\boldsymbol{z}}^{(k)}(t) - F_{\boldsymbol{z}}(t)\right|^2 dG_{\boldsymbol{z}}(t)\right)^{1/2} \\
&\leq \{G_{\boldsymbol{z}}(\tau)\}^2\left(\int_0^\tau \left|\hat{F}_{\boldsymbol{z}}^{(k-1)}(t) - F_{\boldsymbol{z}}(t)\right|^2 dG_{\boldsymbol{z}}(t)\right)^{1/2} \\
&\vdots \\
&\leq \{G_{\boldsymbol{z}}(\tau)\}^{k+1}\left(\int_0^\tau \left|\hat{F}_{\boldsymbol{z}}(t) - F_{\boldsymbol{z}}(t)\right|^2 dG_{\boldsymbol{z}}(t)\right)^{1/2}
\end{aligned}$$

Since $G_{\boldsymbol{z}}(\tau) < 1$, the geometric contraction bound implies

$$\lim_{k\to\infty} \hat{F}_{\boldsymbol{z}}^{(k)}(t) = F_{\boldsymbol{z}}(t) \qquad \text{for almost every } t \in [0,\tau].$$

Lemma 1. *$\hat{F}_{\boldsymbol{z}}^{(k)}$ is strictly increasing on $[0,\tau]$ for all $k \in \mathbb{N} \cup \{0\}$.*

Proof. We use mathematical induction.

(1) $k = 0$: $\hat{F}_{\boldsymbol{z}}^{(0)} = \hat{F}_{\boldsymbol{z}}$ is strictly increasing on $[0,\tau]$ by the assumption.

(2) $k \geq 0$: We assume $\hat{F}_{\boldsymbol{z}}^{(k)}$ is strictly increasing. Then, for $0 \leq a < b \leq \tau$,

$$\begin{aligned}
&\hat{F}_{\boldsymbol{z}}^{(k+1)}(b) - \hat{F}_{\boldsymbol{z}}^{(k+1)}(a) \\
&= \mathbb{E}\left[\mathbb{I}\left(\hat{F}_{\boldsymbol{Z}}^{(k)}(Y) \leq \hat{F}_{\boldsymbol{z}}^{(k)}(b)\right)\left\{\delta + (1-\delta)\frac{\hat{F}_{\boldsymbol{z}}^{(k)}(b) - \hat{F}_{\boldsymbol{Z}}^{(k)}(Y)}{1 - \hat{F}_{\boldsymbol{Z}}^{(k)}(Y)}\right\}\middle|\, \boldsymbol{Z} = \boldsymbol{z}\right] \\
&\quad - \mathbb{E}\left[\mathbb{I}\left(\hat{F}_{\boldsymbol{Z}}^{(k)}(Y) \leq \hat{F}_{\boldsymbol{z}}^{(k)}(a)\right)\left\{\delta + (1-\delta)\frac{\hat{F}_{\boldsymbol{z}}^{(k)}(a) - \hat{F}_{\boldsymbol{Z}}^{(k)}(Y)}{1 - \hat{F}_{\boldsymbol{Z}}^{(k)}(Y)}\right\}\middle|\, \boldsymbol{Z} = \boldsymbol{z}\right] \\
&= \mathbb{E}\left[\mathbb{I}\left(\hat{F}_{\boldsymbol{z}}^{(k)}(a) < \hat{F}_{\boldsymbol{Z}}^{(k)}(Y) \leq \hat{F}_{\boldsymbol{z}}^{(k)}(b)\right)\left\{\delta + (1-\delta)\frac{\hat{F}_{\boldsymbol{z}}^{(k)}(b) - \hat{F}_{\boldsymbol{Z}}^{(k)}(Y)}{1 - \hat{F}_{\boldsymbol{Z}}^{(k)}(Y)}\right\}\middle|\, \boldsymbol{Z} = \boldsymbol{z}\right] \\
&\quad + \mathbb{E}\left[\mathbb{I}\left(\hat{F}_{\boldsymbol{Z}}^{(k)}(Y) \leq \hat{F}_{\boldsymbol{z}}^{(k)}(a)\right)\left\{\delta + (1-\delta)\frac{\hat{F}_{\boldsymbol{z}}^{(k)}(b) - \hat{F}_{\boldsymbol{Z}}^{(k)}(Y)}{1 - \hat{F}_{\boldsymbol{Z}}^{(k)}(Y)}\right\}\middle|\, \boldsymbol{Z} = \boldsymbol{z}\right] \\
&\quad - \mathbb{E}\left[\mathbb{I}\left(\hat{F}_{\boldsymbol{Z}}^{(k)}(Y) \leq \hat{F}_{\boldsymbol{z}}^{(k)}(a)\right)\left\{\delta + (1-\delta)\frac{\hat{F}_{\boldsymbol{z}}^{(k)}(a) - \hat{F}_{\boldsymbol{Z}}^{(k)}(Y)}{1 - \hat{F}_{\boldsymbol{Z}}^{(k)}(Y)}\right\}\middle|\, \boldsymbol{Z} = \boldsymbol{z}\right] \\
&= \mathbb{E}\left[\mathbb{I}\left(\hat{F}_{\boldsymbol{z}}^{(k)}(a) < \hat{F}_{\boldsymbol{Z}}^{(k)}(Y) \leq \hat{F}_{\boldsymbol{z}}^{(k)}(b)\right)\left\{\delta + (1-\delta)\frac{\hat{F}_{\boldsymbol{z}}^{(k)}(b) - \hat{F}_{\boldsymbol{Z}}^{(k)}(Y)}{1 - \hat{F}_{\boldsymbol{Z}}^{(k)}(Y)}\right\}\middle|\, \boldsymbol{Z} = \boldsymbol{z}\right] \\
&\quad + \mathbb{E}\left[\mathbb{I}\left(\hat{F}_{\boldsymbol{Z}}^{(k)}(Y) \leq \hat{F}_{\boldsymbol{z}}^{(k)}(a)\right)(1-\delta)\frac{\hat{F}_{\boldsymbol{z}}^{(k)}(b) - \hat{F}_{\boldsymbol{z}}^{(k)}(a)}{1 - \hat{F}_{\boldsymbol{Z}}^{(k)}(Y)}\middle|\, \boldsymbol{Z} = \boldsymbol{z}\right] \\
&\geq \mathbb{E}\left[\mathbb{I}\left(\hat{F}_{\boldsymbol{z}}^{(k)}(a) < \hat{F}_{\boldsymbol{Z}}^{(k)}(Y) \leq \hat{F}_{\boldsymbol{z}}^{(k)}(b)\right)\delta\,\middle|\, \boldsymbol{Z} = \boldsymbol{z}\right] \\
&\geq \mathbb{E}\left[\mathbb{I}\left(a < T \leq b, T \leq C\right)\middle|\, \boldsymbol{Z} = \boldsymbol{z}\right] \\
&= \int_a^b \{1 - G_{\boldsymbol{z}}(s)\}\, dF_{\boldsymbol{z}}(s) \\
&\geq \{1 - G_{\boldsymbol{z}}(\tau)\}\{F_{\boldsymbol{z}}(b) - F_{\boldsymbol{z}}(a)\} > 0
\end{aligned}$$

Thus $\hat{F}_{\boldsymbol{z}}^{(k+1)}$ is also strictly increasing on $[0,\tau]$. □

B.3 KERNEL ESTIMATION AND PCA

We first analyze the kernel estimator in the PCA-reduced space. We then examine how PCA affects the target function and establish the consistency of the kernel estimator.

Let $\tilde{\boldsymbol{z}} = \boldsymbol{V}^\top \boldsymbol{z}$, where $\boldsymbol{z} \in \mathbb{R}^d$, $\tilde{\boldsymbol{z}} \in \mathbb{R}^p$, and $\boldsymbol{V} \in \mathbb{R}^{d\times p}$ is the PCA loading matrix. We define

$$q\left(x; U, \delta\right) = \mathbb{I}\left(U \leq x\right)\left\{\delta + (1-\delta)\frac{x-U}{1-U}\right\},$$

where $U = \hat{F}_{\boldsymbol{Z}}\left(Y\right)$. For the jth observation, let $u_j = \hat{F}_{\boldsymbol{z}_j}\left(y_j\right)$ and $q_j\left(x\right) = q\left(x; u_j, \delta_j\right)$.

Notation. Define the kernel weights as

$$w_j\left(\tilde{\boldsymbol{z}}_i\right) = \frac{K\left(\frac{\tilde{\boldsymbol{z}}_j - \tilde{\boldsymbol{z}}_i}{h}\right)}{\sum_{k=1}^n K\left(\frac{\tilde{\boldsymbol{z}}_k - \tilde{\boldsymbol{z}}_i}{h}\right)}.$$

The kernel estimator admits two equivalent representations:

$$\hat{H}_{\tilde{\boldsymbol{z}}_i}\left(x; h\right) = \sum_{j=1}^n w_j\left(\tilde{\boldsymbol{z}}_i\right) q_j\left(x\right) = \frac{\mathrm{Upper}_n\left(x \mid \tilde{\boldsymbol{z}}_i\right)}{\mathrm{Lower}_n\left(\tilde{\boldsymbol{z}}_i\right)},$$

where

$$\mathrm{Upper}_n\left(x \mid \tilde{\boldsymbol{z}}_i\right) = \frac{1}{nh^p}\sum_{j=1}^n K\left(\frac{\tilde{\boldsymbol{z}}_j - \tilde{\boldsymbol{z}}_i}{h}\right) q_j\left(x\right),$$

$$\mathrm{Lower}_n\left(\tilde{\boldsymbol{z}}_i\right) = \frac{1}{nh^p}\sum_{j=1}^n K\left(\frac{\tilde{\boldsymbol{z}}_j - \tilde{\boldsymbol{z}}_i}{h}\right).$$

We use the following notation for the empirical average of these estimates:

$$\mathbb{E}_{\tilde{\boldsymbol{z}}_{1:n}}\mathbb{E}_{K_{1:n}, w(\tilde{\boldsymbol{z}})}\left[q\left(x; U, \delta\right) \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}}\right] \triangleq \frac{1}{n}\sum_{i=1}^n \hat{H}_{\tilde{\boldsymbol{z}}_i}\left(x; h\right).$$

Regularity conditions. Let $q\left(x \mid \tilde{\boldsymbol{z}}\right) = \mathbb{E}[q\left(x; U, \delta\right) \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}}]$. We impose the following assumptions:

1. The density $f\left(\tilde{\boldsymbol{z}}\right)$ is three times differentiable with respect to $\tilde{\boldsymbol{z}}$.
2. There exists a constant $c_0 > 0$ such that $\inf_{\tilde{\boldsymbol{z}}\in\tilde{\mathcal{Z}}} f(\tilde{\boldsymbol{z}}) \geq c_0$, where $\tilde{\mathcal{Z}}$ denotes the evaluation region in the PCA space.
3. Let $f_{\tilde{\boldsymbol{z}}}$ and $g_{\tilde{\boldsymbol{z}}}$ denote the conditional densities of T and C given $\tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}}$, respectively. For each $x \in [0,1]$, the functions $\int_{\mathcal{Q}_x(\tilde{\boldsymbol{z}})} f_{\tilde{\boldsymbol{z}}}(t)\, dt$ and $\int_{\mathcal{Q}_x(\tilde{\boldsymbol{z}})} g_{\tilde{\boldsymbol{z}}}(t)\, dt$ are three times differentiable with respect to $\tilde{\boldsymbol{z}}$, where $\mathcal{Q}_x(\tilde{\boldsymbol{z}}) = \{t : \hat{F}_{\tilde{\boldsymbol{z}}}(t) \leq x\}$.
4. For each fixed t, $\hat{F}_{\tilde{\boldsymbol{z}}}\left(t\right)$ is three times differentiable with respect to $\tilde{\boldsymbol{z}}$.
5. $\mathbb{P}(T \leq C \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}})$ is three times differentiable with respect to $\tilde{\boldsymbol{z}}$.
6. The kernel is symmetric and satisfies:
   (a) $K\left(\boldsymbol{s}\right) \geq 0$ and $\|K\|_\infty < \infty$.
   (b) $\int_{\mathbb{R}^p} K\left(\boldsymbol{s}\right) d\boldsymbol{s} = 1$.
   (c) $\int_{\mathbb{R}^p} \boldsymbol{s} K\left(\boldsymbol{s}\right) d\boldsymbol{s} = \boldsymbol{0}$.
   (d) $\int_{\mathbb{R}^p} \boldsymbol{s}^\top \boldsymbol{s} K\left(\boldsymbol{s}\right) d\boldsymbol{s} < \infty$.
   (e) $\int_{\mathbb{R}^p} K^2\left(\boldsymbol{s}\right) d\boldsymbol{s} = k_2 < \infty$.
7. For each $x \in [0,1]$, the Hessian of $q\left(x \mid \tilde{\boldsymbol{z}}\right) f\left(\tilde{\boldsymbol{z}}\right)$ is uniformly bounded over $x \in [0,1]$ and the covariate region considered in the analysis.

The following results establish the consistency of the kernel estimator for $q(x \mid \tilde{\boldsymbol{z}})$. Although kernel weighting does not improve the convergence rate of marginal estimation, it allows the calibration map to vary with the retained principal components.

Bias and variance of the numerator and denominator.

For $x \in [0,1]$ and an interior point $\tilde{\boldsymbol{z}}_0$, consider

$$\hat{H}_{\tilde{\boldsymbol{z}}_0}(x;h) = \frac{\mathrm{Upper}_n(x \mid \tilde{\boldsymbol{z}}_0)}{\mathrm{Lower}_n(\tilde{\boldsymbol{z}}_0)}.$$

For independent observations and bounded f, the regularity conditions on the kernel and $0 \leq q_j(x) \leq 1$ imply $\mathrm{Var}[\mathrm{Upper}_n(x \mid \tilde{\boldsymbol{z}}_0)] = \mathrm{Var}[\mathrm{Lower}_n(\tilde{\boldsymbol{z}}_0)] = O(1/nh^p)$.

To obtain the expectation of the numerator, we use the Taylor expansion:

$$\begin{aligned}
& \frac{1}{h^p}\int_{\mathbb{R}^p} q(x \mid \tilde{\boldsymbol{z}})\, K\left((\tilde{\boldsymbol{z}} - \tilde{\boldsymbol{z}}_0)/h\right) f(\tilde{\boldsymbol{z}})\, d\tilde{\boldsymbol{z}} \\
= & \int_{\mathbb{R}^p} q(x \mid \tilde{\boldsymbol{z}}_0 + h\boldsymbol{s})\, K(\boldsymbol{s})\, f(\tilde{\boldsymbol{z}}_0 + h\boldsymbol{s})\, d\boldsymbol{s} \\
= & \int_{\mathbb{R}^p} \left\{ q(x \mid \tilde{\boldsymbol{z}}_0) f(\tilde{\boldsymbol{z}}_0) + h\left(\nabla_{\tilde{\boldsymbol{z}}_0} qf\right)^\top \boldsymbol{s} + \frac{1}{2}h^2 \boldsymbol{s}^\top \left(\nabla^2_{\tilde{\boldsymbol{z}}_0} qf\right)\boldsymbol{s} + o\left(h^2 \boldsymbol{s}^\top \boldsymbol{s}\right) \right\} K(\boldsymbol{s})\, d\boldsymbol{s} \\
= & \; q(x \mid \tilde{\boldsymbol{z}}_0) f(\tilde{\boldsymbol{z}}_0) + O\left(h^2\right)
\end{aligned}$$

under the regularity conditions. Similarly, $\mathbb{E}[\mathrm{Lower}_n(\tilde{\boldsymbol{z}}_0)] = f(\tilde{\boldsymbol{z}}_0) + O\left(h^2\right)$. Balancing squared bias and variance gives the optimal bandwidth order $h = O(n^{-1/(4+p)})$ for both Upper_n and Lower_n in terms of MSE.

Remark: differentiability. To justify the preceding Taylor expansion, we examine the differentiability of qf with respect to $\tilde{\boldsymbol{z}}$. Since f is assumed to be three times differentiable, it suffices to establish the same property for q. Let $U_T = \hat{F}_{\tilde{\boldsymbol{Z}}}(T)$. We decompose $q(x \mid \tilde{\boldsymbol{z}})$ as follows:

$$\begin{aligned}
& q(x \mid \tilde{\boldsymbol{z}}) \\
= & \; \mathbb{E}\left[\mathbb{I}(T \leq C,\, U \leq x) \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}}\right] + \mathbb{E}\left[\mathbb{I}(T > C,\, U \leq x)\, \frac{x-U}{1-U} \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}}\right] \\
= & \; \mathbb{E}\left[\mathbb{I}(U \leq x) \mid T \leq C, \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}}\right] \mathbb{P}(T \leq C \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}}) \qquad (6) \\
& + \mathbb{E}\left[\mathbb{I}(U \leq x)\, \frac{x-U}{1-U} \mid T > C, \tilde{\boldsymbol{z}} = \tilde{\boldsymbol{z}}\right] \mathbb{P}(T > C \mid \tilde{\boldsymbol{Z}} = \tilde{\tilde{\boldsymbol{z}}}). \qquad (7)
\end{aligned}$$

With the assumption that $\mathbb{P}(U \leq x \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}})$ is a three times differentiable function w.r.t. $\tilde{\boldsymbol{z}}$, i.e., we have for $\tilde{\boldsymbol{z}} = \tilde{\boldsymbol{z}}_0 + \boldsymbol{a}$ with $Y = T\ (T \leq C)$,

$$\begin{aligned}
& \mathbb{P}(U_T \leq x \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}}) \\
& \approx \mathbb{P}(U_T \leq x \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}}_0) + \boldsymbol{a}^\top \frac{\partial \mathbb{P}(U_T \leq x \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}})}{\partial \tilde{\boldsymbol{z}}}\bigg|_{\tilde{\boldsymbol{z}}=\tilde{\boldsymbol{z}}_0} + \frac{1}{2}\boldsymbol{a}^\top \frac{\partial^2 \mathbb{P}(U_T \leq x \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}})}{\partial \tilde{\boldsymbol{z}} \partial \tilde{\boldsymbol{z}}^\top}\bigg|_{\tilde{\boldsymbol{z}}=\tilde{\boldsymbol{z}}_0} \boldsymbol{a}.
\end{aligned}$$

It implies that (6) is three times differentiable under the regularity conditions, including the assumption that $\mathbb{P}(T \leq C \mid \tilde{\boldsymbol{z}} = \tilde{\boldsymbol{z}})$ is three times differentiable. Furthermore, with $Y = C\ (T > C)$,

$$\mathbb{E}\left[\mathbb{I}(U \leq x)\, \frac{x-U}{1-U} \mid T > C, \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}}\right] \qquad (8)$$

is three times differentiable due to the differentiability of $(x-U)/(1-U)$ with respect to $\tilde{\boldsymbol{z}}$ in the regularity conditions. This implies the differentiability of (7) under the regularity conditions, including $\mathbb{P}(T \leq C \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}})$ is third-order differentiable. Therefore, $q(x \mid \tilde{\boldsymbol{z}})$ is three times differentiable with respect to $\tilde{\boldsymbol{z}}$.

Under the regularity conditions, the derivatives $\partial^k \mathbb{P}(U_T \leq x \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}})/\partial \tilde{\boldsymbol{z}}^k$ exist for $k = 1, 2, 3$. The first derivative can be written as

$$\frac{\partial \int_{\mathcal{Q}_x(\tilde{\boldsymbol{z}})} f_{\tilde{\boldsymbol{z}}}(t)\, dt}{\partial \tilde{\boldsymbol{z}}}$$

For the second and third derivatives, it is required that the following exist:

$$\frac{\partial^2 \int_{\mathcal{Q}_x(\tilde{\boldsymbol{z}})} f_{\tilde{\boldsymbol{z}}}(t)\,dt}{\partial \tilde{\boldsymbol{z}} \tilde{\boldsymbol{z}}^\top}, \frac{\partial^3 \int_{\mathcal{Q}_x(\tilde{\boldsymbol{z}})} f_{\tilde{\boldsymbol{z}}}(t)\,dt}{\partial \tilde{z}_i \tilde{z}_j \tilde{z}_k},\ (i,j,k) \in [p]^3$$

where $[p] = \{1, 2, \ldots, p\}$. The existence of these derivatives is ensured by the regularity conditions. A similar argument applies to $g_{\tilde{\boldsymbol{z}}}(t)$ in (8). □

B.4 Convergence rate of the kernel estimator

Lemma 2. *Assume that the regularity conditions hold and $h = O(n^{-1/(4+p)})$. Then for any $M > 0$,*

$$\sup_{i\in[n]} \sup_{x\in[0,1]} \left| \sum_{j=1}^{n} w_j(\tilde{\boldsymbol{z}}_i)\, q_j(x) - q(x \mid \tilde{\boldsymbol{z}}_i) \right| < M n^{-2/(4+p)} \log n$$

with probability tending to 1 as $n \to \infty$.

Using the mean and variance bounds established in the previous section for $\text{Upper}_n(x \mid \tilde{\boldsymbol{z}}_i)$ and $\text{Lower}_n(\tilde{\boldsymbol{z}}_i)$, we apply Bernstein's inequality. For conditionally independent random variables $X_1, \ldots, X_n$ with conditional mean zero,

$$\mathbb{P}\left(\left| \frac{1}{n} \sum_{j=1}^{n} X_j \right| > \epsilon \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}}_i \right) \le 2 \exp\left(-\frac{n\epsilon^2/2}{\sigma_i^2 + k\epsilon/3} \right) \tag{9}$$

where $|X_j| \le k$ and $\sigma_i^2 = n^{-1} \sum_{j=1}^{n} \mathbb{E}[X_j^2 \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}}_i]$. Define

$$\begin{aligned} X_j \equiv X_j(x, \tilde{\boldsymbol{z}}_i) =& \frac{1}{h^p} K\left(\frac{\tilde{\boldsymbol{Z}}_j - \tilde{\boldsymbol{z}}_i}{h} \right) q(x; U_j, \delta_j) - q(x \mid \tilde{\boldsymbol{z}}_i) f(\tilde{\boldsymbol{z}}_i) \\ &+ q(x \mid \tilde{\boldsymbol{z}}_i) f(\tilde{\boldsymbol{z}}_i) - \mathbb{E}\left[\frac{1}{h^p} K\left(\frac{\tilde{\boldsymbol{Z}}_j - \tilde{\boldsymbol{z}}_i}{h} \right) q(x; U_j, \delta_j) \,\middle|\, \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}}_i \right] \end{aligned}$$

Then $X_1, \ldots, X_n$ are conditionally independent and have a conditional mean of zero.

Since $|q(x; U_j, \delta_j)| \le 1$, the kernel assumption gives $|X_j| \le 2/h^p$. Moreover,

$$\begin{aligned} \sigma_i^2 = \frac{1}{n} \sum_{j=1}^{n} \mathbb{E}\left[X_j^2 \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}}_i \right] &\le \frac{1}{nh^{2p}} \sum_{j=1}^{n} \mathbb{E}\left[K^2\left(\frac{\tilde{\boldsymbol{Z}}_j - \tilde{\boldsymbol{z}}_i}{h} \right) \,\middle|\, \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}}_i \right] \\ &= \frac{1}{h^{2p}} \int_{\mathbb{R}^p} K^2\left(\frac{\tilde{\boldsymbol{z}} - \tilde{\boldsymbol{z}}_i}{h} \right) f(\tilde{\boldsymbol{z}})\, d\tilde{\boldsymbol{z}} \\ &= \frac{1}{h^p} \int_{\mathbb{R}^p} K^2(\boldsymbol{s})\, f(\tilde{\boldsymbol{z}}_i + h\boldsymbol{s})\, d\boldsymbol{s} \\ &\le \frac{k_2 \|f\|_\infty}{h^p} = \frac{c}{h^p} \end{aligned}$$

Set $h = n^{-1/(4+p)}$ and $\epsilon \equiv \epsilon_n = M n^{-2/(4+p)} \log n$ for any fixed $M > 0$. The exponent on the right-hand side of (9) is then bounded above by

$$\begin{aligned} -\frac{\frac{1}{2} n \epsilon_n^2}{c n^{p/(p+4)} + \frac{2}{3} n^{p/(p+4)} \epsilon_n} &= -\frac{n^{1-p/(p+4)} M^2 n^{-4/(p+4)} (\log n)^2}{2c + \frac{4}{3} M n^{-2/(p+4)} \log n} \\ &= -\frac{M^2 (\log n)^2}{2c + \frac{4}{3} M n^{-2/(p+4)} \log n} \\ &\le -CM^2 (\log n)^2 \end{aligned} \tag{10}$$

for some constant $C > 0$ and sufficiently large n. So the right-hand side of (9) converges to zero. Since $\left| q(x \mid \tilde{\boldsymbol{z}}_i) f(\tilde{\boldsymbol{z}}_i) - \mathbb{E}[K((\tilde{\boldsymbol{z}}_j - \tilde{\boldsymbol{z}}_i)/h) q_j(x)/h^p \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}}_i] \right| = O\left(n^{-2/(4+p)} \log n \right)$, we

have

$$\left|\frac{1}{n}\sum_{j=1}^{n}\frac{K\left(\left(\tilde{\boldsymbol{z}}_j-\tilde{\boldsymbol{z}}_i\right)/h\right)}{h^p}q_j\left(x\right)-q\left(x\mid\tilde{\boldsymbol{z}}_i\right)f\left(\tilde{\boldsymbol{z}}_i\right)\right|>\epsilon_n$$

with probability at most $\exp\left(-CM^2\left(\log n\right)^2\right)$. For each fixed x, applying the union bound over the n evaluation points yields

$$\begin{aligned}\mathbb{P}\left(\frac{1}{n}\sup_{i\in[n]}\left|\sum_{j=1}^{n}X_j\left(x,\tilde{\boldsymbol{z}}_i\right)\right|>\epsilon_n\right) &= \mathbb{E}\left[\mathbb{P}\left(\frac{1}{n}\sup_{i\in[n]}\left|\sum_{j=1}^{n}X_j\left(x,\tilde{\boldsymbol{z}}_i\right)\right|>\epsilon_n\,\middle|\,\tilde{\boldsymbol{Z}}=\tilde{\boldsymbol{z}}_i\right)\right]\\ &\leq \mathbb{E}\left[2n\exp\left(-CM^2\left(\log n\right)^2\right)\right]\\ &= 2n\exp\left(-CM^2\left(\log n\right)^2\right)\\ &= 2\exp\left(-CM^2\left(\log n\right)^2+\log n\right)\end{aligned}$$

Thus, choosing $\epsilon_n=Mn^{-2/(4+p)}\log n$ yields convergence uniformly over $i\in[n]$ for each fixed x. To extend the convergence uniformly over $x\in[0,1]$, for each $i\in[n]$, construct a finite set $\mathcal{B}_i\in[0,1]$ satisfying:

1. For every $x\in[0,1]$, there exists $x^*\in\mathcal{B}_i$ such that $\left|q\left(x\mid\tilde{\boldsymbol{z}}_i\right)-q\left(x^*\mid\tilde{\boldsymbol{z}}_i\right)\right|\leq\frac{1}{n}$.
2. $|\mathcal{B}_i|=n$.

Combining the discretization argument with the union bound gives

$$\begin{aligned}\mathbb{P}\left(\frac{1}{n}\sup_{x\in[0,1]}\sup_{i\in[n]}\left|\sum_{j=1}^{n}X_j\left(x,\tilde{\boldsymbol{z}}_i\right)\right|>\epsilon_n\right) &\leq 2n^2\exp\left(-CM^2(\log n)^2\right)\\ &= 2\exp\left(-CM^2(\log n)^2+2\log n\right)\end{aligned}$$

The same argument applies to the denominator. Define

$$X_j\equiv X_j\left(\tilde{\boldsymbol{z}}_i\right)=\frac{1}{h^p}K\left(\frac{\tilde{\boldsymbol{z}}_j-\tilde{\boldsymbol{z}}_i}{h}\right)-f\left(\tilde{\boldsymbol{z}}_i\right)+f\left(\tilde{\boldsymbol{z}}_i\right)-\mathbb{E}\left[\frac{1}{h^p}K\left(\frac{\tilde{\boldsymbol{z}}_j-\tilde{\boldsymbol{z}}_i}{h}\right)\,\middle|\,\tilde{\boldsymbol{Z}}=\tilde{\boldsymbol{z}}_i\right]$$

With $h=n^{-1/(4+p)}$ and $\epsilon_n=Mn^{-2/(4+p)}\log n$, the same concentration argument and a union bound over $i\in[n]$ yield

$$\mathbb{P}\left(\frac{1}{n}\sup_{i\in[n]}\left|\sum_{j=1}^{n}X_j\left(\tilde{\boldsymbol{z}}_i\right)\right|>\epsilon_n\right)\leq 2\exp\left(-CM^2(\log n)^2+2\log n\right).$$

Thus, $\sup_{i\in[n]}|\mathrm{Lower}_n\left(\tilde{\boldsymbol{z}}_i\right)-f\left(\tilde{\boldsymbol{z}}_i\right)|\leq\epsilon_n$ with probability tending to one exponentially as $n\to\infty$. Moreover, by the fact that when $a_n\left(\boldsymbol{z}_i\right)\to a\left(\boldsymbol{z}_i\right)$ uniformly with tending to one exponentially as $n\to\infty$ and also $b_n\left(\boldsymbol{z}_i\right)\to b\left(\boldsymbol{z}_i\right)$, $\forall\boldsymbol{z}_i, b\left(\boldsymbol{z}_i\right)\geq c_0>0, a_n\left(\boldsymbol{z}_i\right)/b_n\left(\boldsymbol{z}_i\right)\to a\left(\boldsymbol{z}_i\right)/b\left(\boldsymbol{z}_i\right)$ uniformly with tending to one exponentially as $n\to\infty$.

By combining two convergence results of $\mathrm{Upper}_n\left(x\mid\tilde{\boldsymbol{z}}_i\right)$ and $\mathrm{Lower}_n\left(\tilde{\boldsymbol{z}}_i\right)$ with the regularity condition 1, we conclude that for any positive M,

$$\sup_{i\in[n]}\sup_{x\in[0,1]}\left|\sum_{j=1}^{n}w_j\left(\tilde{\boldsymbol{z}}_i\right)q_j\left(x\right)-q\left(x\mid\tilde{\boldsymbol{z}}_i\right)\right|<Mn^{-2/(4+p)}\log n$$

with probability tending to 1 as $n\to\infty$.

Remark 1. The concentration bounds in the proof of Lemma 2 also quantify the dependence of the estimation error on the sample size. Setting $M=1$ and $\epsilon_n=n^{-2/(4+p)}\log n$, we obtain, for sufficiently large n,

$$\mathbb{P}\left(\sup_{i\in[n]}\sup_{x\in[0,1]}\left|\hat{H}_{\tilde{\boldsymbol{z}}_i}\left(x;h\right)-q\left(x\mid\tilde{\boldsymbol{z}}_i\right)\right|>\epsilon_n\right)\leq\delta_n:=4\exp\left\{-C(\log n)^2+2\log n\right\},$$

where $C > 0$ is independent of n. For sufficiently small $\epsilon_n > 0$, solving $\epsilon_n = n^{-2/(4+p)} \log n$ gives

$$n = \Theta\left(\left(\frac{W\left(-2\epsilon/(4+p)\right)}{-2\epsilon/(4+p)}\right)^{(4+p)/2}\right).$$

where W denotes the Lambert W function satisfying $W(s)\, e^{W(s)} = s$.

Remark 2. Under additional regularity conditions, $4 + p$ can be replaced by $2k + p$ when the order of differentiability increases to $k + 1$, where $k > 2$. Boundary effects may lead to slower convergence near the boundary and require careful treatment when establishing uniform convergence. These effects can be addressed using kernels adapted to the boundary. Moreover, the proportion of observations near the boundary tends to zero as $n \to \infty$. □

B.5 Target function Approximation in the Algorithm

In practice, we use $H_{\tilde{\boldsymbol{z}}}(x)$ instead of $H_{\boldsymbol{z}}(x)$. This may lead to a mismatch between the domains, motivating the following conditions on the principal components.

For the theoretical analysis, write $\boldsymbol{z}^* = (\tilde{\boldsymbol{z}}, \boldsymbol{z}^*_{d-p})$, where $\boldsymbol{z}^*$ denotes the full principal component representation of $\boldsymbol{z}$, preserving all its information, and $\tilde{\boldsymbol{z}}$ contains the p principal components with the largest variances.

Without loss of generality, assume that $\boldsymbol{z}$ has mean zero. Then, $\hat{F}_{\boldsymbol{z}}(\cdot) = \hat{F}_{\boldsymbol{z}^*}(\cdot)$. We consider the following condition, analogous to a Lipschitz condition:

$$\text{C1}: \quad \text{for } \epsilon > 0,\ \sup_x \left| h(\hat{F}_{\boldsymbol{z}^*})(x) - h(\hat{F}_{\boldsymbol{z}^*+\epsilon \boldsymbol{e}})(x) \right| \le k\epsilon,$$

where $\boldsymbol{e}$ is a unit vector whose first p entries are zero and whose remaining entries are nonzero, and h is a Lipschitz continuous function.

Motivated by the lower variance of the discarded principal components, we also impose the following small-ball probability condition:

$$\text{C2}: \quad \mathbb{P}\left(\|\boldsymbol{z}^*_{d-p}\| \le \epsilon_n\right) \ge 1 - \exp\left(-1/\epsilon_n\right),$$

where $\epsilon_n \to 0$. Let $\epsilon_n = o(n^{-2/(4+p)})$. Combining C1 and C2, we obtain that, for all x, $|H_{\boldsymbol{z}}(x) - H_{\tilde{\boldsymbol{z}}}(x)|$ is bounded by $n^{-2/(4+p)}$ with a probability tending to one exponentially. Thus, the approximation error is no larger in order than the convergence rate established in Lemma 2.

C Algorithm for bandwidth selection in CoSC

Algorithm 1 describes the bandwidth-selection procedure for CoSC.

The iterative procedure repeatedly applies Algorithm 1, using the updated CDF as the input for the next iteration. At each iteration, the bandwidth is reselected and the calibration map is re-estimated using the validation set. The resulting map is applied to the CDFs for both the validation and test sets before the next iteration.

We use KS-cal for bandwidth selection because it is computationally efficient to evaluate and optimize. Although KS-cal measures marginal calibration, the correction remains covariate-dependent through kernel weighting, with smaller bandwidths allowing more localized adjustments. An alternative is to optimize $\mathrm{Cal}_{\mathrm{ws}}$, which more directly assesses conditional calibration. However, its sampling-based computation introduces additional variability and computational cost.

A limitation of using KS-cal is that it may not sufficiently distinguish between large bandwidths that yield nearly uniform kernel weights. Figure 3 illustrates the cross-validation loss as a function of h. Some curves exhibit a well-defined minimum at $h < 1$, whereas others display broad, nearly flat regions at larger bandwidths. To avoid initializing optimization in these flat regions, we restrict the initial bandwidth to $h < 1$. Specifically, we first perform a grid search over $[0.1, 0.5]$ and select the bandwidth with the lowest cross-validation KS-cal. We then initialize gradient descent at this value to further refine h. Future work could investigate bandwidth-selection objectives that better capture conditional calibration while retaining computational efficiency.

Figure 4 shows the selected bandwidths across dataset-model combinations.

Algorithm 1 One-step CoSC

Require: validation set $\mathcal{D}_{\text{val}}$, test set $\mathcal{D}_{\text{test}}$, Estimated CDF $\hat{F}_{\boldsymbol{z}}$, number of folds K, number of repetitions B, bandwidth search space $h \in \mathcal{H} \subseteq (0, \infty)$

Ensure: Post-processed test CDFs $\hat{F}^*_{\boldsymbol{z}}$

for $b = 1, \ldots, B$ **do**
  Randomly partition $\mathcal{D}_{\text{val}}$ into K folds $\{\mathcal{D}_{b,k}\}_{k=1}^{K}$
end for
function $\mathrm{L}(h)$
  for $b = 1, \ldots, B$ **do**
    for $k = 1, \ldots, K$ **do**
      $\mathcal{D}_{\text{cal}} \leftarrow \mathcal{D}_{\text{val}} \setminus \mathcal{D}_{b,k}$
      $\mathcal{D}_{\text{eval}} \leftarrow \mathcal{D}_{b,k}$
      Estimate $\hat{H}^{(b,k)}_{\tilde{\boldsymbol{z}}}(\cdot; h)$ using $\mathcal{D}_{\text{cal}}$
      Transform the CDFs in $\mathcal{D}_{\text{eval}}$: $\hat{F}^{*(b,k)}_{\boldsymbol{z}}(t) \leftarrow \hat{H}^{(b,k)}_{\tilde{\boldsymbol{z}}}(\hat{F}_{\boldsymbol{z}}(t); h)$
      $\ell_{b,k}(h) \leftarrow$ KS-cal of the transformed CDFs on $\mathcal{D}_{\text{eval}}$
    end for
    $\ell_b(h) \leftarrow \dfrac{1}{K}\sum_{k=1}^{K} \ell_{b,k}(h)$
  end for
  return $\dfrac{1}{B}\sum_{b=1}^{B} \ell_b(h)$
end function
$\hat{h} \leftarrow \arg\min_{h \in \mathcal{H}} \mathrm{L}(h)$
Estimate $\hat{H}_{\tilde{\boldsymbol{z}}}(\cdot; \hat{h})$ using the entire validation set $\mathcal{D}_{\text{val}}$
Transform the CDFs in $\mathcal{D}_{\text{test}}$: $\hat{F}^*_{\boldsymbol{z}}(t) \leftarrow \hat{H}_{\tilde{\boldsymbol{z}}}(\hat{F}_{\boldsymbol{z}}(t); \hat{h})$
return Post-processed test CDFs $\hat{F}^*_{\boldsymbol{z}}$

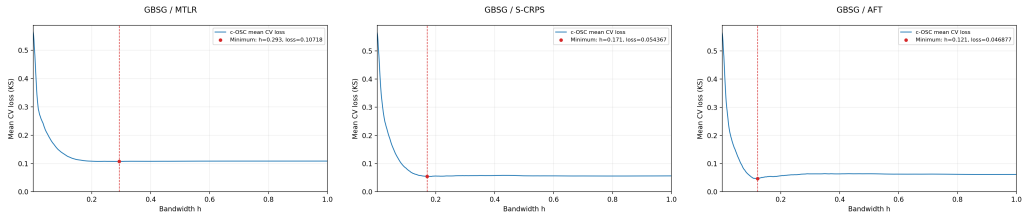

Figure 3: Examples of cross-validation loss as a function of the bandwidth h.

D Simulation Studies

To complement the theoretical error analysis in Section 4, we conduct controlled simulation studies to empirically investigate the three components of the estimation error: the kernel estimation error, the dimension-reduction error induced by PCA, and the population-level transformation error. In all simulation experiments, we run CoSC for 10 iterations.

Our simulation study has three objectives. First, we vary the size of the calibration set to examine how the three error components change with the size of the calibration data. Second, we vary the dimension of the covariates to investigate the trade-off between kernel estimation error and dimension-reduction error. Finally, we apply CoSC on top of other post-processing methods to evaluate its use as a second-stage correction after KSP, CSD, or CSD-iPOT.

D.1 Simulation Setup

We use a common data-generating mechanism across the three simulation studies. Unless otherwise specified, the covariate dimension is $d = 32$, and the covariates are generated as

$$\boldsymbol{Z}_i \sim N_d(\mathbf{0}, \boldsymbol{\Sigma}), \qquad \Sigma_{rs} = 10\rho^{|r-s|},$$

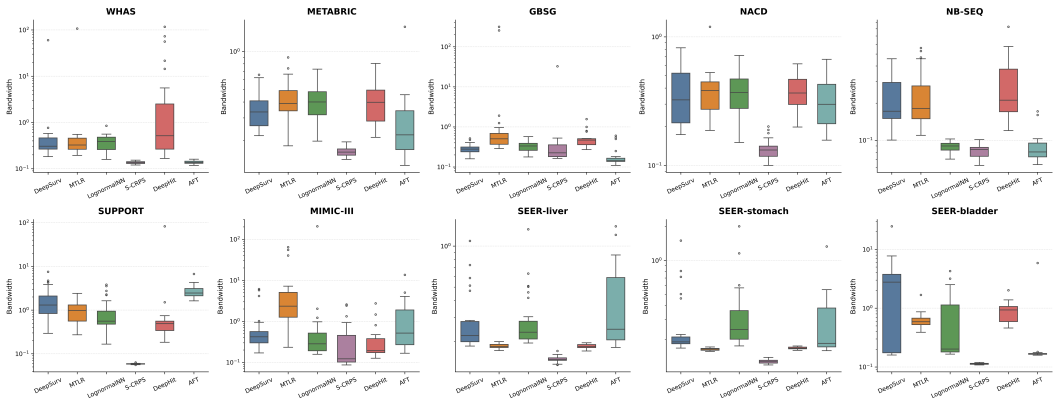

Figure 4: Selected bandwidths for CoSC.

with $\rho = 0.9$. Thus, Σ follows an AR(1) correlation structure with a marginal variance of 10.

For $q \in \{T, C\}$, we define $\boldsymbol{X}_i^{(q)} = \boldsymbol{W}^{(q)}\boldsymbol{Z}_i + \boldsymbol{b}^{(q)} = (X_{i1}^{(q)}, X_{i2}^{(q)})^\top$ where $\boldsymbol{W}^{(q)} \in \mathbb{R}^{2\times d}$ and $\boldsymbol{b}^{(q)} \in \mathbb{R}^2$. Each entry of $\boldsymbol{W}^{(q)}$ and $\boldsymbol{b}^{(q)}$ is independently sampled from Unif $[-0.1, 0.1]$, and the resulting coefficients are held fixed within each simulation repetition. Let

$$s_{(q)}^2 = Var\left(X_{i2}^{(q)}\right) = \left(\boldsymbol{w}_2^{(q)}\right)^\top \boldsymbol{\Sigma}\boldsymbol{w}_2^{(q)},$$

where $\boldsymbol{w}_2^{(q)}$ denotes the second row of $\boldsymbol{W}^{(q)}$. We then define $a_i^{(q)} = \tanh(X_{i2}^{(q)}/s_{(q)}), p_{i2}^{(q)} = 2\exp\left(0.6a_i^{(q)}\right)$. Conditional on $\boldsymbol{Z}_i$, the event and censoring times are independently generated as

$$T_i \mid \boldsymbol{Z}_i \sim \text{Weibull}\left(\exp\left\{X_{i1}^{(T)}\right\}, p_{i2}^{(T)}\right), \quad C_i \mid \boldsymbol{Z}_i \sim \text{Weibull}\left(m_r \exp\left\{X_{i1}^{(C)}\right\}, p_{i2}^{(C)}\right).$$

where the Weibull distribution is parametrized by its scale and shape, respectively. The multiplier m_r is chosen to achieve the censoring rate $\mathbb{P}(C_i < T_i) \approx r$ for $r \in \{0.3, 0.6, 0.9\}$. The observed outcome is defined by $Y_i = \min(T_i, C_i), \delta_i = \mathbb{I}(T_i \le C_i)$.

For each repetition, we independently generate training, validation, calibration, and test sets containing 4,000, 2,000, 2,000, and 2,000 observations, respectively. The training and validation sets are used to fit and tune the baseline survival model, respectively, while the calibration set is used to estimate the calibration map and select its bandwidth. The test set is used only for final evaluation. In the calibration-sample-size experiment, we use calibration subsets of sizes $n_{\text{cal}} \in \{500, 1{,}000, 2{,}000\}$. An additional independent pilot sample of 20,000 observations is used only to determine m_r and the evaluation time horizons τ, such that $\tau_\alpha = F^{-1}(\alpha)$ with $F(\tau_\alpha) = \mathbb{E}_{\boldsymbol{z}}[F_{\boldsymbol{z}}(\tau_\alpha)]$. Each simulation setting is repeated over 30 random seeds. For each error term, we report the sample average.

D.2 Error decomposition

For each of the six survival models, we plot the total error together with the three terms in its upper-bound decomposition: the kernel estimation error, dimension-reduction error, and population-level transformation error.

To evaluate the population-level transformation error, we construct an oracle population update using the true event and censoring distributions. As derived in Appendix B, the update based on the current estimator $\hat{F}_{\boldsymbol{z}}^{(k)}$ satisfies

$$H_{\boldsymbol{z}}^{(k)}\left(\hat{F}_{\boldsymbol{z}}^{(k)}(t)\right) = F_{\boldsymbol{z}}(t) + G_{\boldsymbol{z}}(t)\,\hat{S}_{\boldsymbol{z}}^{(k)}(t)\left[r_{\boldsymbol{z}}^{(k)}(t) - \bar{r}_{\boldsymbol{z}}^{(k)}(t)\right].$$

We approximate the projected population calibration map $H_{\tilde{\boldsymbol{z}}}^{(k)}(x) = \mathbb{E}[H_{\boldsymbol{Z}}^{(k)}(x) \mid \tilde{\boldsymbol{Z}} = \tilde{\boldsymbol{z}}]$ by Monte Carlo integration, using the conditional covariate distribution derived from the known Gaussian data-generating distribution. Figures 5-10 present the total error and its three components for

the six survival models across different calibration set sizes and censoring rates. Overall, increasing the calibration set size tends to reduce the total error. The differences among calibration set sizes are primarily attributable to the kernel estimation error, which generally decreases as n_{cal} increases. In contrast, the population-level transformation and dimension-reduction errors are largely unaffected by n_{cal}, as these terms arise from the population transformation and the information lost through projection, respectively.

The population-level transformation error measures the discrepancy between this oracle update and $F_{\boldsymbol{z}}$. Across most models and censoring rates, the first population update produces the largest reduction in this error, after which the improvement is relatively modest. Under severe censoring, particularly for S-CRPS, further iterations provide more gradual improvements.

The KS-cal values of the kernel-updated CDFs generally improve as the calibration set size increases. At low-to-moderate censoring rates, most models achieve most of their improvement in the first iteration, whereas additional iterations yield only small gains or may slightly degrade calibration. Under 90% censoring, however, several models continue to improve over subsequent iterations. These results motivate the use of one correction step in the main experiments, although it is not optimal in every setting.

For most models, the kernel estimation error reaches its minimum within the first few iterations and subsequently remains stable or increases, even with larger calibration sample sizes. Therefore repeated application does not necessarily reduce the finite-sample estimation error. This pattern is consistent with the propagation of estimation noise when the calibration map and bandwidth are repeatedly re-estimated from the same finite calibration sample. The dimension-reduction error is generally insensitive to n_{cal} and relatively stable across iterations, consistent with its interpretation as projection error that cannot be eliminated merely by increasing the calibration set size. An exception is S-CRPS under heavy censoring, for which the dimension-reduction error also decreases over iterations.

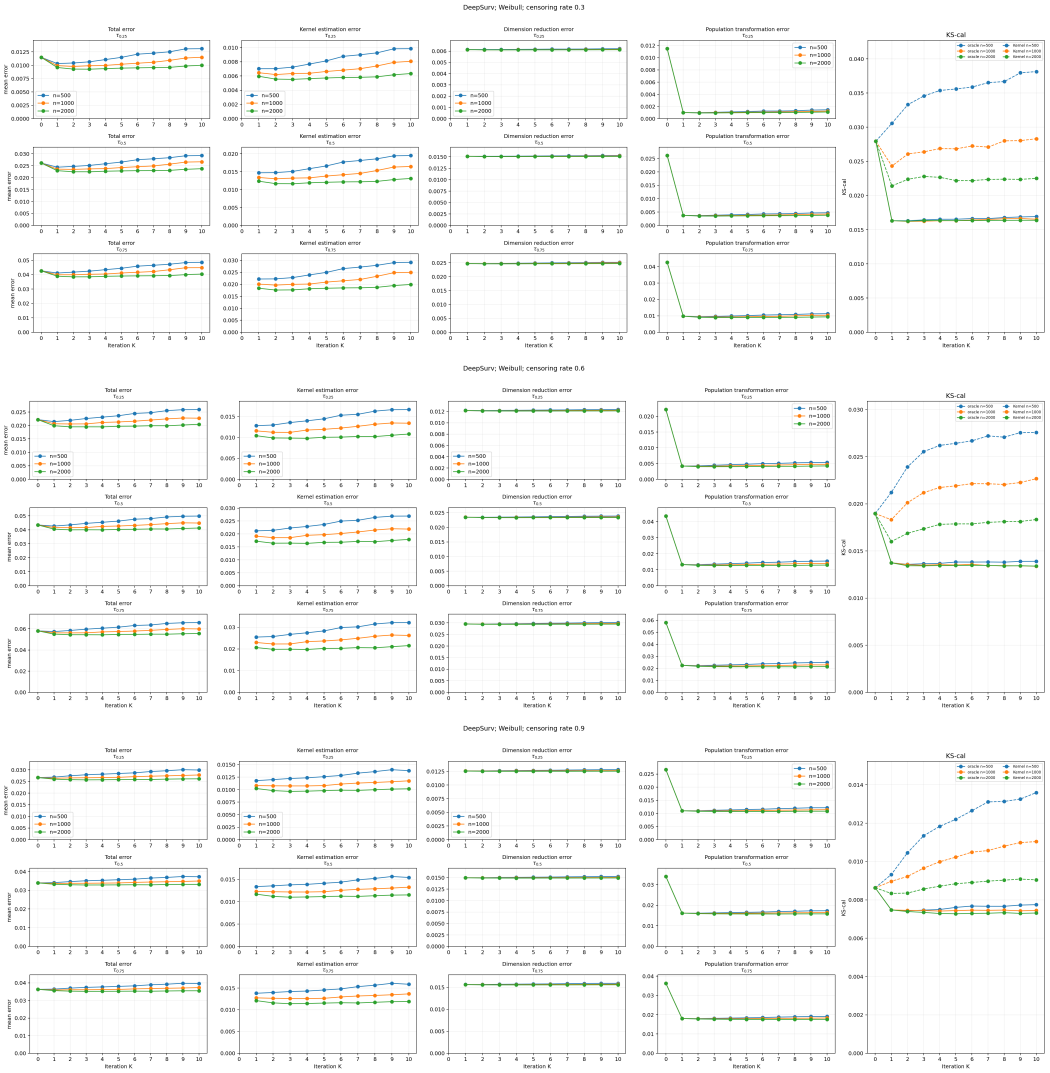

Figure 5: Error decomposition and KS-cal for DeepSurv across varying calibration set sizes and censoring rates. From left to right, the columns show the total error, kernel estimation error, dimension-reduction error, population-level transformation error, and KS-cal. The rows correspond to the evaluation horizons $\tau_{0.25}$, $\tau_{0.50}$, and $\tau_{0.75}$. All error terms are averaged over the observations.



Figure 6: Error decomposition and KS-cal for MTLR across varying calibration set sizes and censoring rates. From left to right, the columns show the total error, kernel estimation error, dimension-reduction error, population-level transformation error, and KS-cal. The rows correspond to the evaluation horizons $\tau_{0.25}$, $\tau_{0.50}$, and $\tau_{0.75}$. All error terms are averaged over the observations.

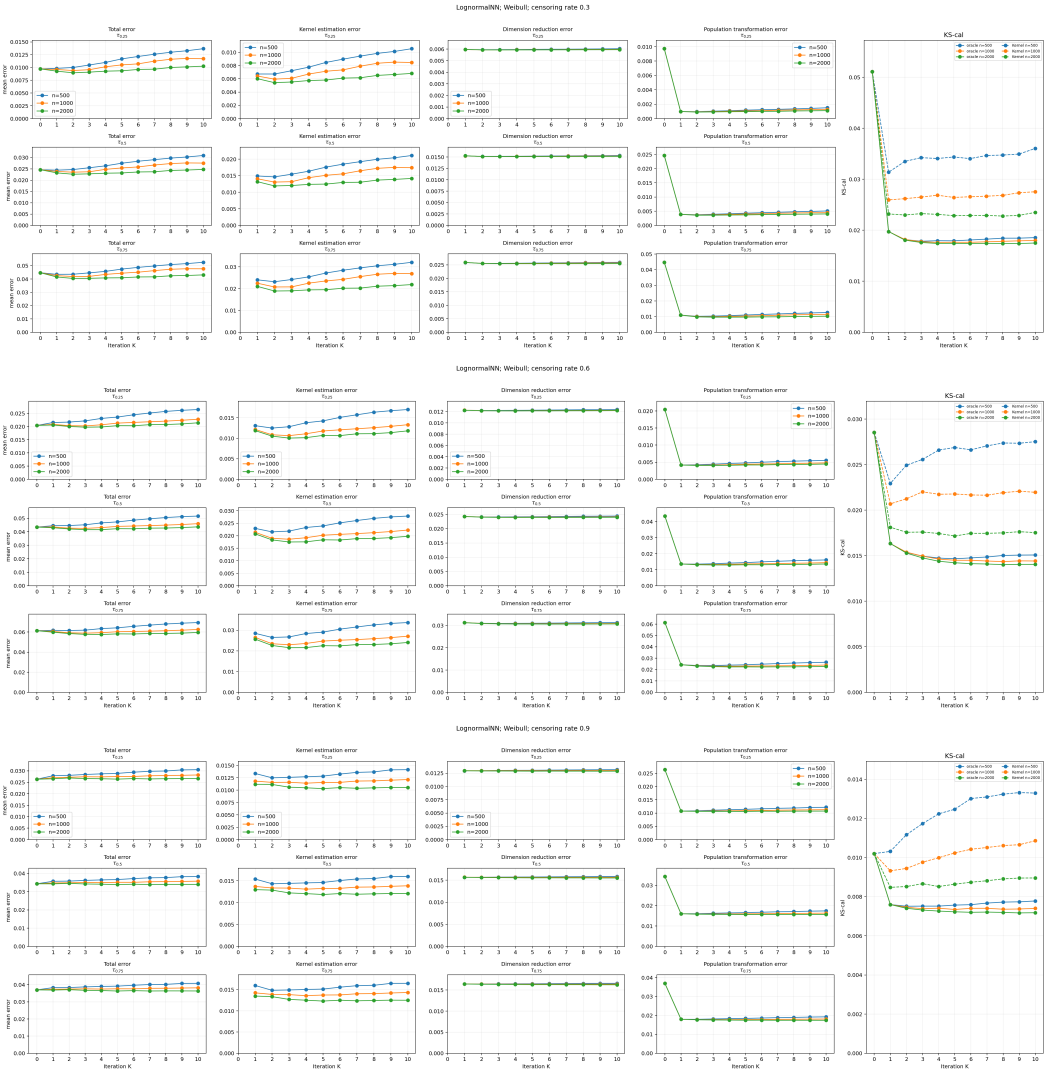

Figure 7: Error decomposition and KS-cal for LognormalNN across varying calibration set sizes and censoring rates. From left to right, the columns show the total error, kernel estimation error, dimension-reduction error, population-level transformation error, and KS-cal. The rows correspond to the evaluation horizons $\tau_{0.25}$, $\tau_{0.50}$, and $\tau_{0.75}$. All error terms are averaged over the observations.

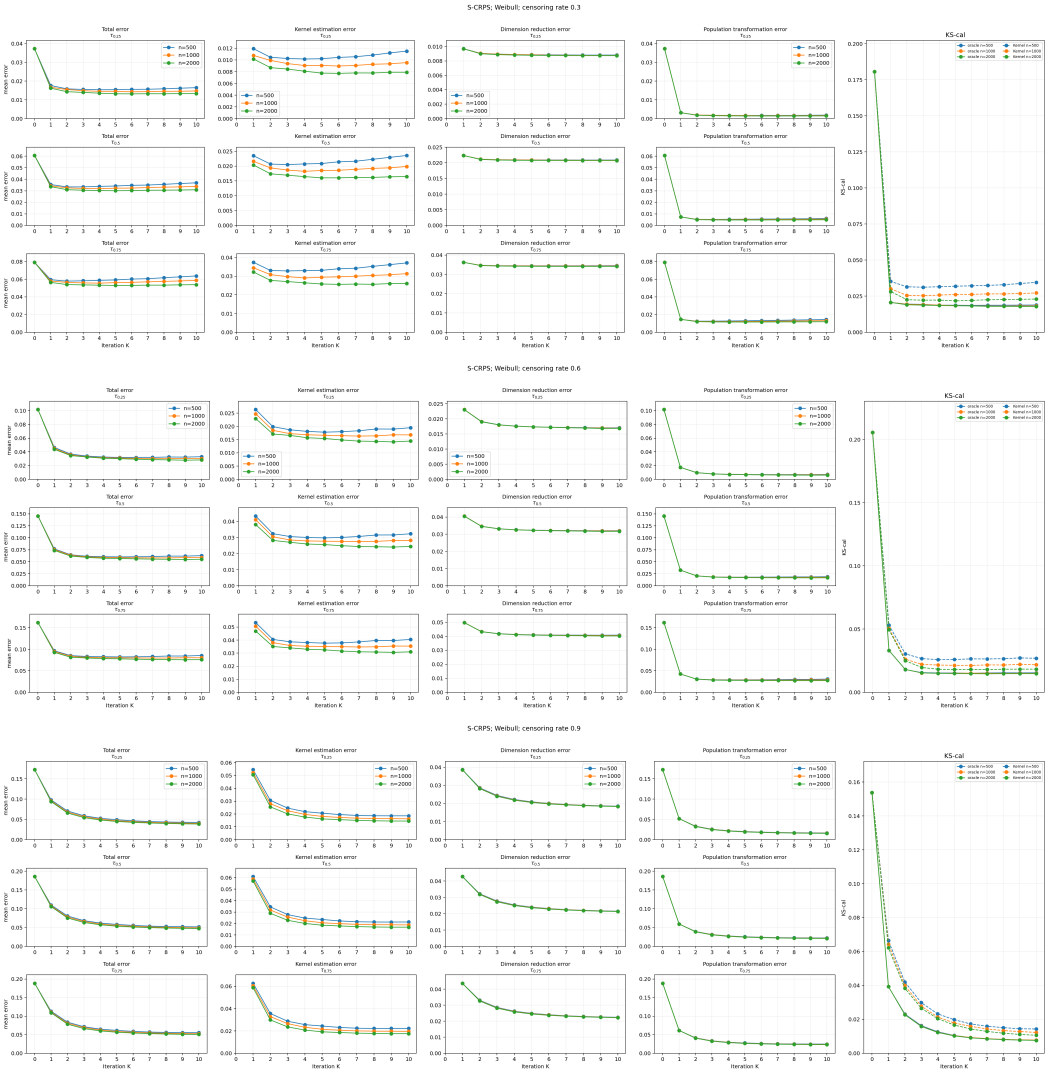

Figure 8: Error decomposition and KS-cal for S-CRPS across varying calibration set sizes and censoring rates. From left to right, the columns show the total error, kernel estimation error, dimension-reduction error, population-level transformation error, and KS-cal. The rows correspond to the evaluation horizons $\tau_{0.25}$, $\tau_{0.50}$, and $\tau_{0.75}$. All error terms are averaged over the observations.

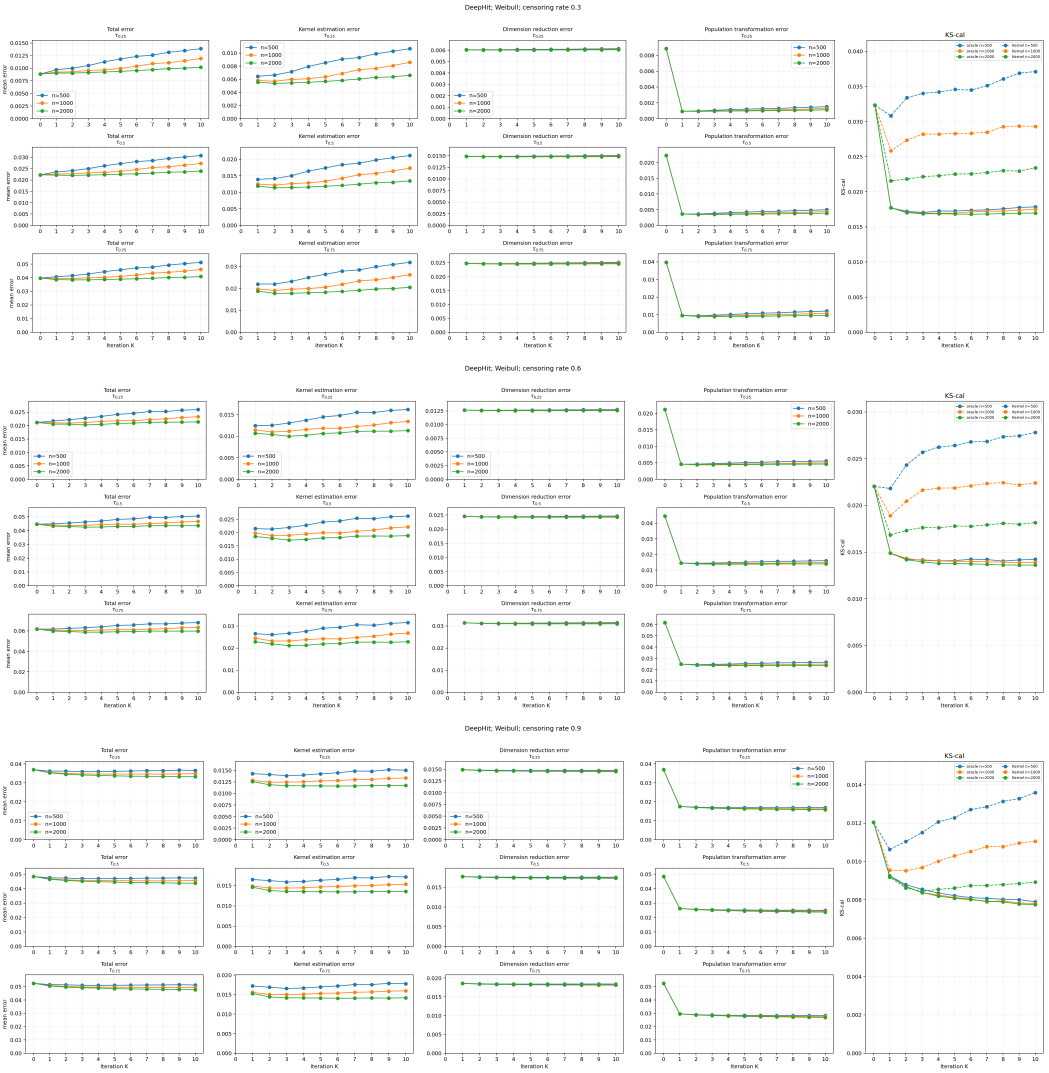

Figure 9: Error decomposition and KS-cal for DeepHit across varying calibration set sizes and censoring rates. From left to right, the columns show the total error, kernel estimation error, dimension-reduction error, population-level transformation error, and KS-cal. The rows correspond to the evaluation horizons $\tau_{0.25}$, $\tau_{0.50}$, and $\tau_{0.75}$. All error terms are averaged over the observations.



Figure 10: Error decomposition and KS-cal for AFT across varying calibration set sizes and censoring rates. From left to right, the columns show the total error, kernel estimation error, dimension-reduction error, population-level transformation error, and KS-cal. The rows correspond to the evaluation horizons $\tau_{0.25}$, $\tau_{0.50}$, and $\tau_{0.75}$. All error terms are averaged over the observations.

D.3 TRADE-OFF BETWEEN KERNEL ESTIMATION AND DIMENSION-REDUCTION

We next examine the effect of reducing the dimension used for kernel weighting. We fix the calibration set size at $n_{\mathrm{cal}} = 2,000$ and vary the original covariate dimension over $d \in \{16, 32, 64\}$. For each d, we compare updates using all covariates with updates using principal components that explain at least 90% of their variance. The corresponding reduced dimensions are approximately $p = 4$, 7, and 14. Comparisons between PCA and raw covariates are made within each original dimension.

Figures 11-16 show that PCA generally lowers the kernel estimation error for its projected target $H_{\tilde{\boldsymbol{z}}}$, relative to estimating $H_{\boldsymbol{z}}$ using all covariates. This reduction comes with a nonzero dimension-reduction error; the corresponding error for the raw covariate estimator is zero by definition. The population-level transformation errors are largely similar between the two approaches. At the first update used in the main experiments, their total errors and KS-cal values are often close, with no consistent advantage for either representation across models, censoring rates, and evaluation horizons. Later iterations show the same broad pattern, although some settings exhibit modest improvements with PCA. Because the three components form an upper bound rather than an additive identity, their plotted values need not numerically account for the difference in total error.

These results show that PCA can make the projected calibration map easier to estimate under the correlated covariates considered here, but a smaller kernel estimation component alone does not imply a smaller error for the final CDF. In the real-data ablation in Section 5.2, the 90% PCA configuration yields small improvements in median KS-cal, IBS, and $\mathrm{Cal}_{\mathrm{ws}}$, while raw covariates yield slightly better C-index and PSR. We therefore treat PCA as a practical choice for kernel weighting rather than a required component of CoSC or a guarantee of improved overall accuracy.

D.4 CORRECTION PROCEDURE FOR OTHER POST-PROCESSING METHODS

The initial estimator $\hat{F}_{\boldsymbol{z}}$ used to construct $H_{\boldsymbol{z}}$ need not be the raw output of a baseline survival model; it may be any estimated conditional CDF, including the output of another post-processing method. CoSC can therefore be used as a second-stage calibration procedure. We evaluate this compositional property by applying CoSC to predictions previously adjusted using KSP, CSD, or CSD-iPOT and examining whether the self-correction procedure provides further improvements. In this experiment, we fix the calibration set size at $n_{\mathrm{cal}} = 2,000$.

As shown in Figures 17-22, across the different initial estimators, CoSC generally produced similar update trajectories, with the largest improvements occurring when the initial error at $k = 0$ was relatively large. These gains were particularly pronounced for some predictions calibrated by CSD under moderate or heavy censoring and for poorly calibrated raw predictions. In contrast, when the initial estimator was already accurate, as was often the case for KSP or CSD-iPOT, the trajectories tended to remain nearly flat, indicating limited room for further correction.

For some CSD and CSD-iPOT initializations, improvements plateau after a few iterations or remain negligible. Across the settings considered, CoSC generally preserves or improves accuracy, with gains varying by the initial method.

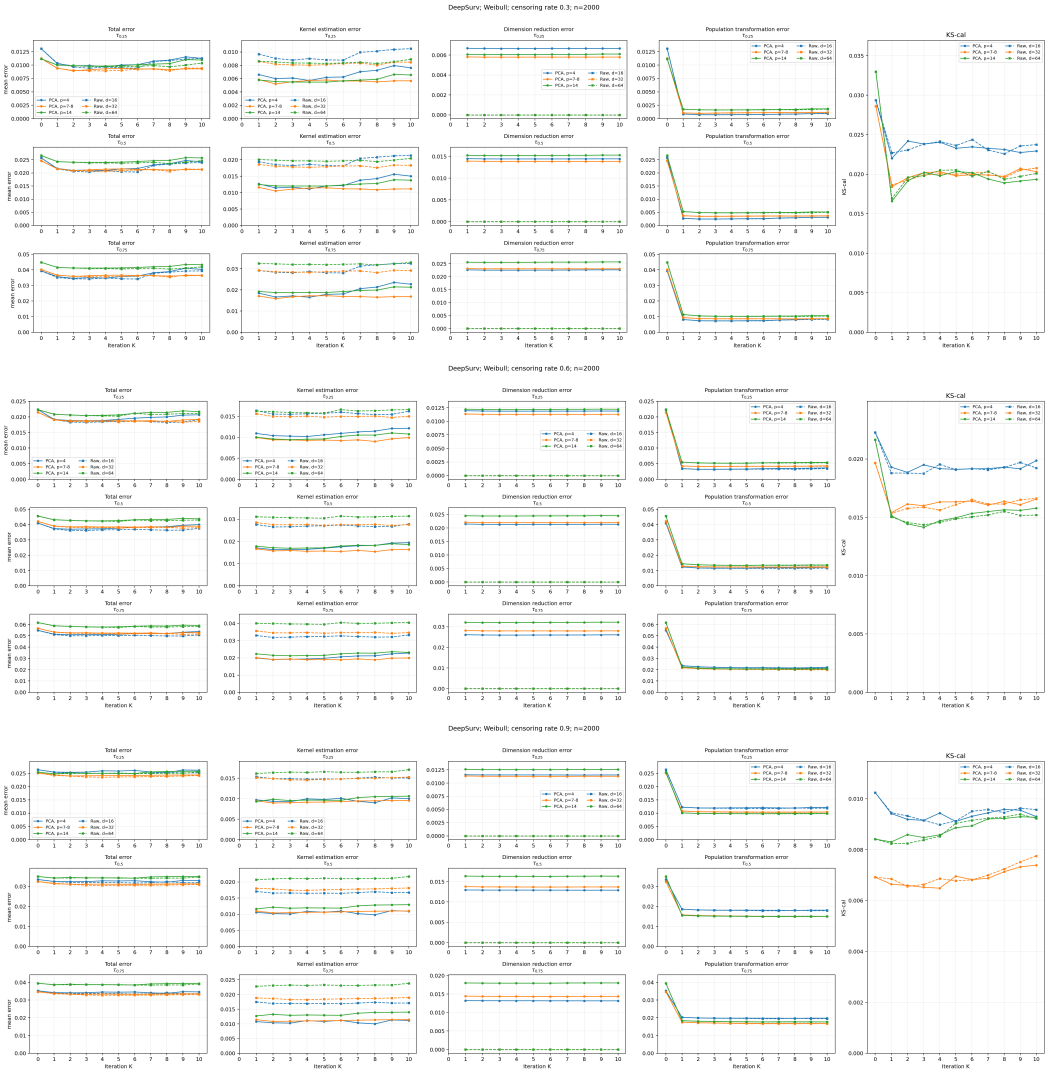

Figure 11: PCA versus raw covariates for DeepSurv with $n_{\text{cal}} = 2,000$. The three plots correspond to censoring rates of 0.3, 0.6, and 0.9. Within each plot, rows show $\tau_{0.25}$, $\tau_{0.50}$, and $\tau_{0.75}$, while columns show the total error, three terms of its upper bound, and KS-cal. Solid and dashed curves indicate PCA and raw covariates, respectively.

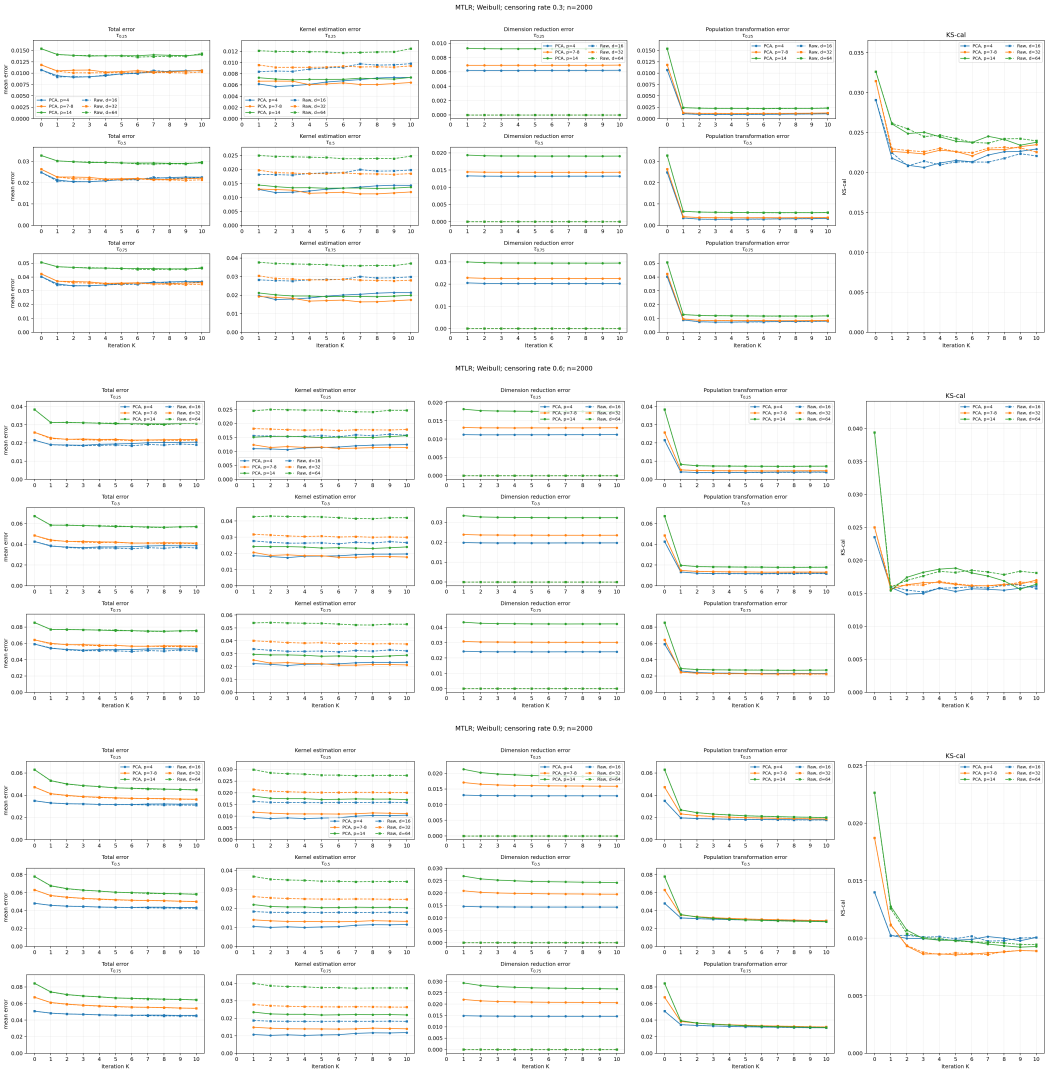

Figure 12: PCA versus raw covariates for MTLR with $n_{\text{cal}} = 2,000$. The three plots correspond to censoring rates of 0.3, 0.6, and 0.9. Within each plot, rows show $\tau_{0.25}$, $\tau_{0.50}$, and $\tau_{0.75}$, while columns show the total error, three terms of its upper bound, and KS-cal. Solid and dashed curves indicate PCA and raw covariates, respectively.



Figure 13: PCA versus raw covariates for LognormalNN with $n_{\text{cal}} = 2,000$. The three plots correspond to censoring rates of 0.3, 0.6, and 0.9. Within each plot, rows show $\tau_{0.25}$, $\tau_{0.50}$, and $\tau_{0.75}$, while columns show the total error, three terms of its upper bound, and KS-cal. Solid and dashed curves indicate PCA and raw covariates, respectively.

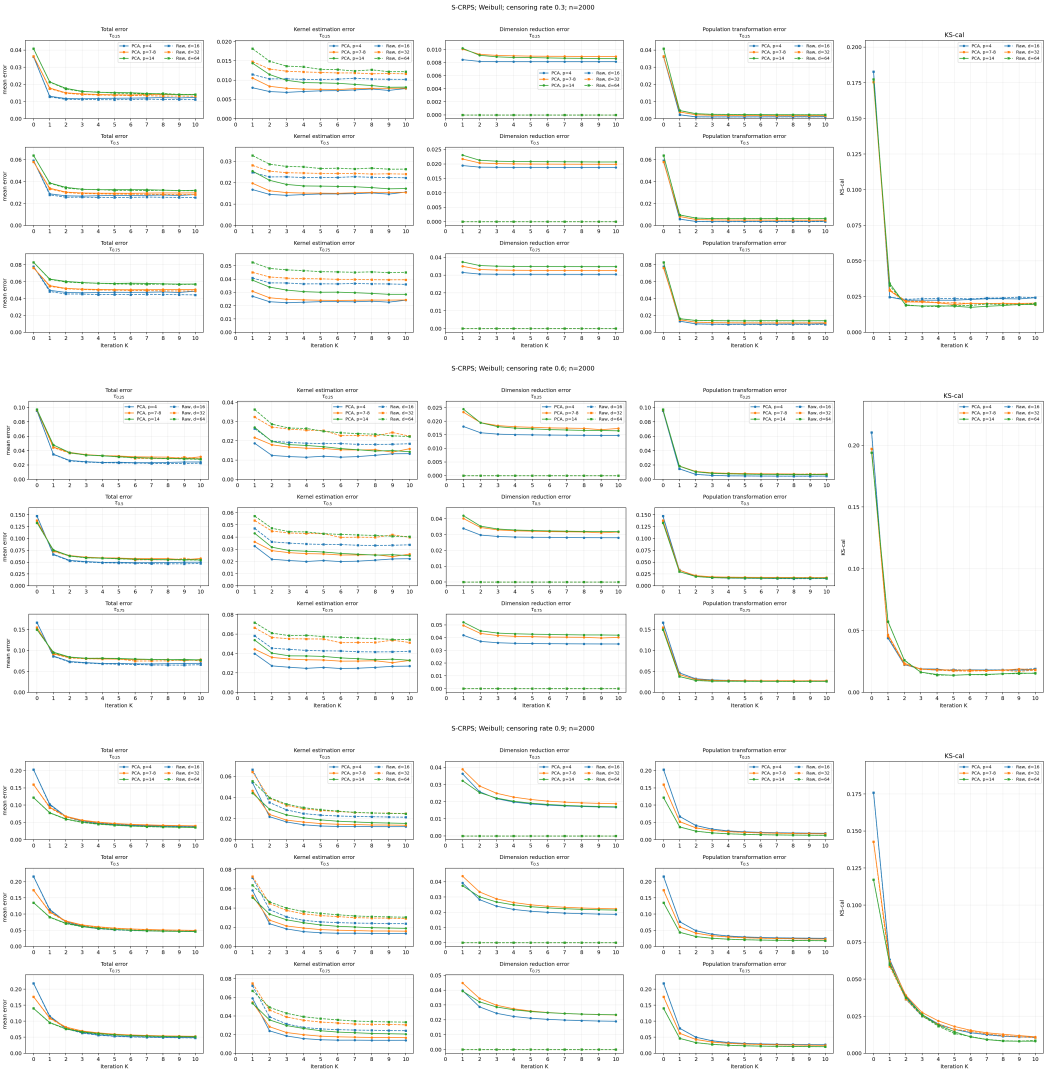

Figure 14: PCA versus raw covariates for S-CRPS with $n_{\text{cal}} = 2,000$. The three plots correspond to censoring rates of 0.3, 0.6, and 0.9. Within each plot, rows show $\tau_{0.25}$, $\tau_{0.50}$, and $\tau_{0.75}$, while columns show the total error, three terms of its upper bound, and KS-cal. Solid and dashed curves indicate PCA and raw covariates, respectively.

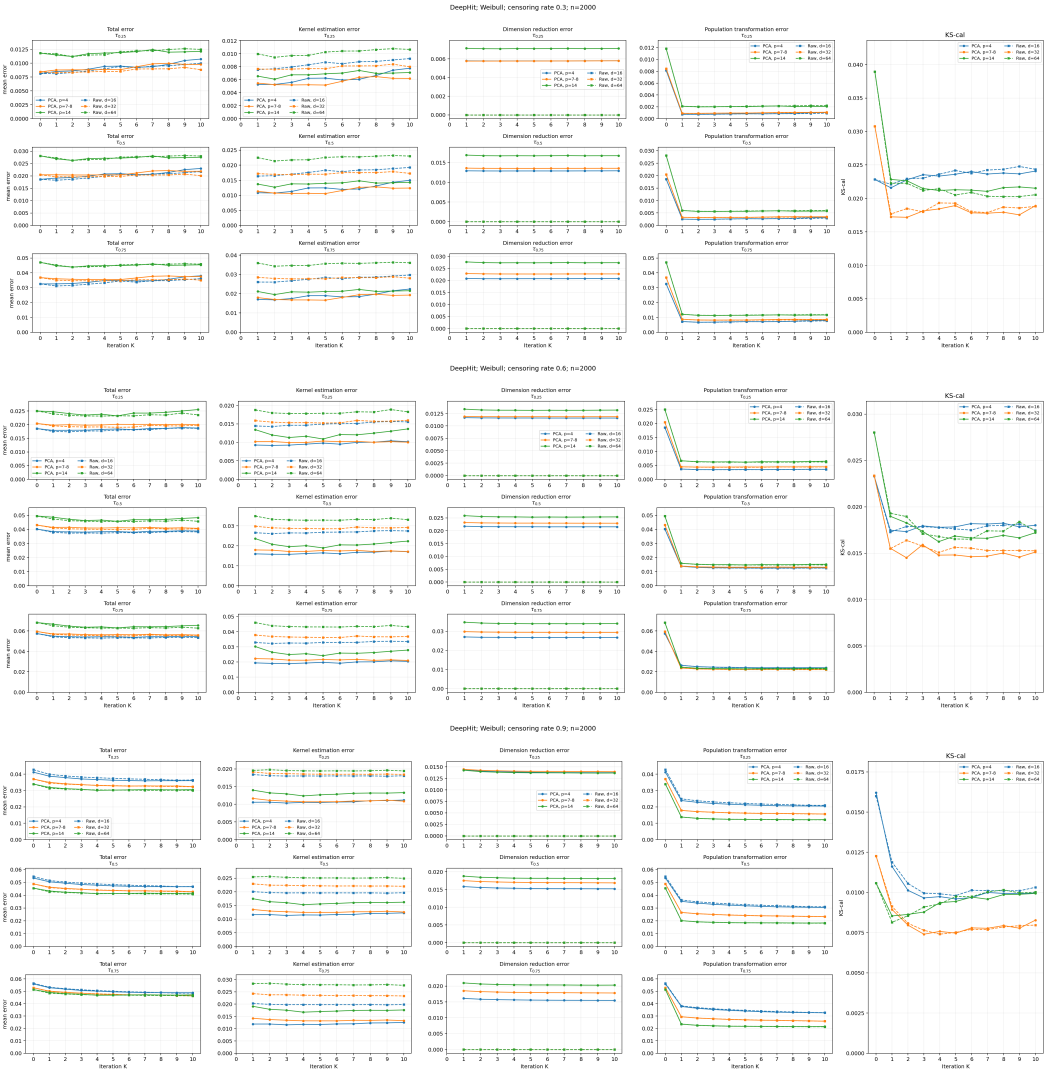

Figure 15: PCA versus raw covariates for DeepHit with $n_{\text{cal}} = 2,000$. The three plots correspond to censoring rates of 0.3, 0.6, and 0.9. Within each plot, rows show $\tau_{0.25}$, $\tau_{0.50}$, and $\tau_{0.75}$, while columns show the total error, three terms of its upper bound, and KS-cal. Solid and dashed curves indicate PCA and raw covariates, respectively.



Figure 16: PCA versus raw covariates for AFT with $n_{\text{cal}} = 2,000$. The three plots correspond to censoring rates of 0.3, 0.6, and 0.9. Within each plot, rows show $\tau_{0.25}$, $\tau_{0.50}$, and $\tau_{0.75}$, while columns show the total error, three terms of its upper bound, and KS-cal. Solid and dashed curves indicate PCA and raw covariates, respectively.

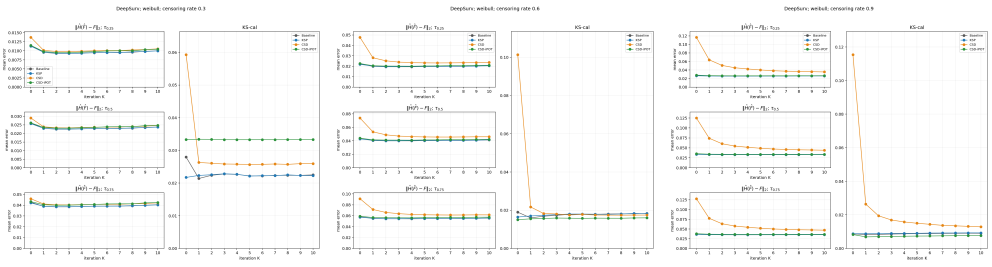

Figure 17: Performance of CoSC for DeepSurv predictions before or after post-processing by KSP, CSD, and CSD-iPOT. The columns correspond to censoring rates of 0.3, 0.6, and 0.9, while the rows show averaged total errors at $\tau_{0.25}$, $\tau_{0.50}$, and $\tau_{0.75}$. KS-cal is additionally reported for each censoring rate.

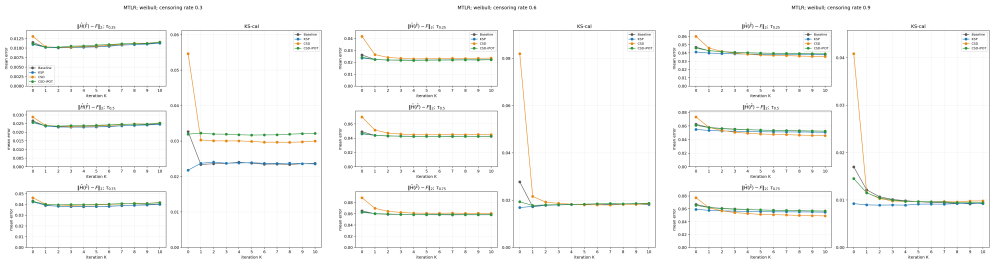

Figure 18: Performance of CoSC for MTLR predictions before or after post-processing by KSP, CSD, and CSD-iPOT. The columns correspond to censoring rates of 0.3, 0.6, and 0.9, while the rows show averaged total errors at $\tau_{0.25}$, $\tau_{0.50}$, and $\tau_{0.75}$. KS-cal is additionally reported for each censoring rate.

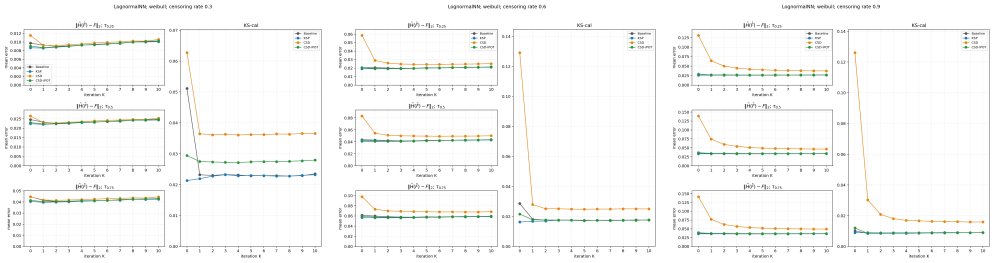

Figure 19: Performance of CoSC for LognormalNN predictions before or after post-processing by KSP, CSD, and CSD-iPOT. The columns correspond to censoring rates of 0.3, 0.6, and 0.9, while the rows show averaged total errors at $\tau_{0.25}$, $\tau_{0.50}$, and $\tau_{0.75}$. KS-cal is additionally reported for each censoring rate.

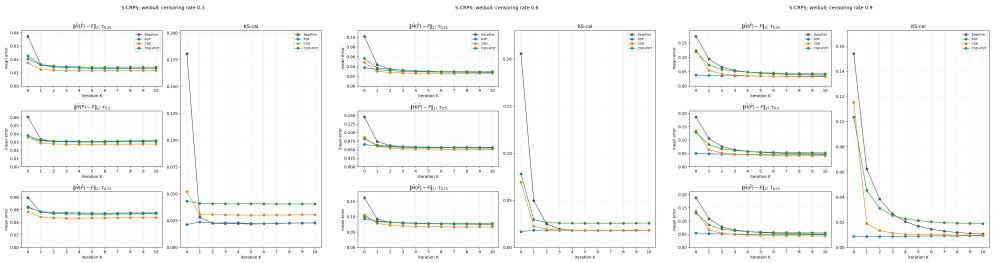

Figure 20: Performance of CoSC for S-CRPS predictions before or after post-processing by KSP, CSD, and CSD-iPOT. The columns correspond to censoring rates of 0.3, 0.6, and 0.9, while the rows show averaged total errors at $\tau_{0.25}$, $\tau_{0.50}$, and $\tau_{0.75}$. KS-cal is additionally reported for each censoring rate.

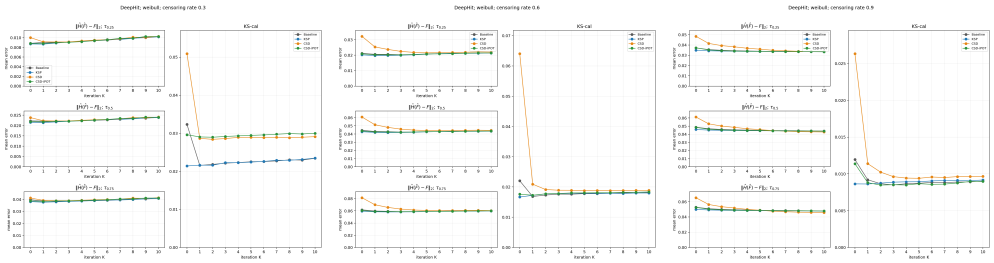

Figure 21: Performance of CoSC for DeepHit predictions before or after post-processing by KSP, CSD, and CSD-iPOT. The columns correspond to censoring rates of 0.3, 0.6, and 0.9, while the rows show averaged total errors at $\tau_{0.25}$, $\tau_{0.50}$, and $\tau_{0.75}$. KS-cal is additionally reported for each censoring rate.

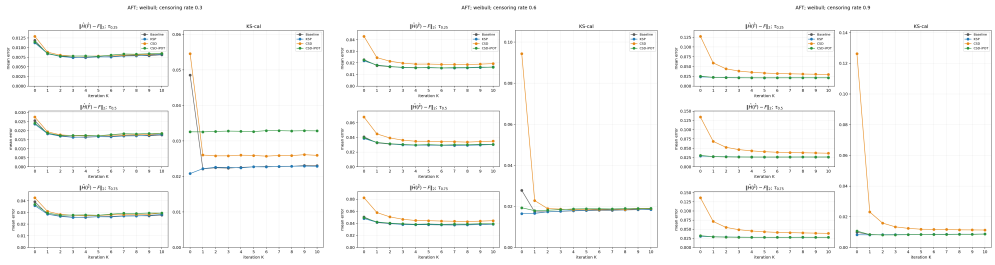

Figure 22: Performance of CoSC for AFT predictions before or after post-processing by KSP, CSD, and CSD-iPOT. The columns correspond to censoring rates of 0.3, 0.6, and 0.9, while the rows show averaged total errors at $\tau_{0.25}$, $\tau_{0.50}$, and $\tau_{0.75}$. KS-cal is additionally reported for each censoring rate.

E CALIBRATION PLOTS

The following figures present calibration plots comparing CoSC with the baseline and alternative calibration methods across datasets and models. Predicted probabilities are plotted against observed probabilities, with the diagonal line indicating perfect calibration. Each subplot compares CoSC with one other method.

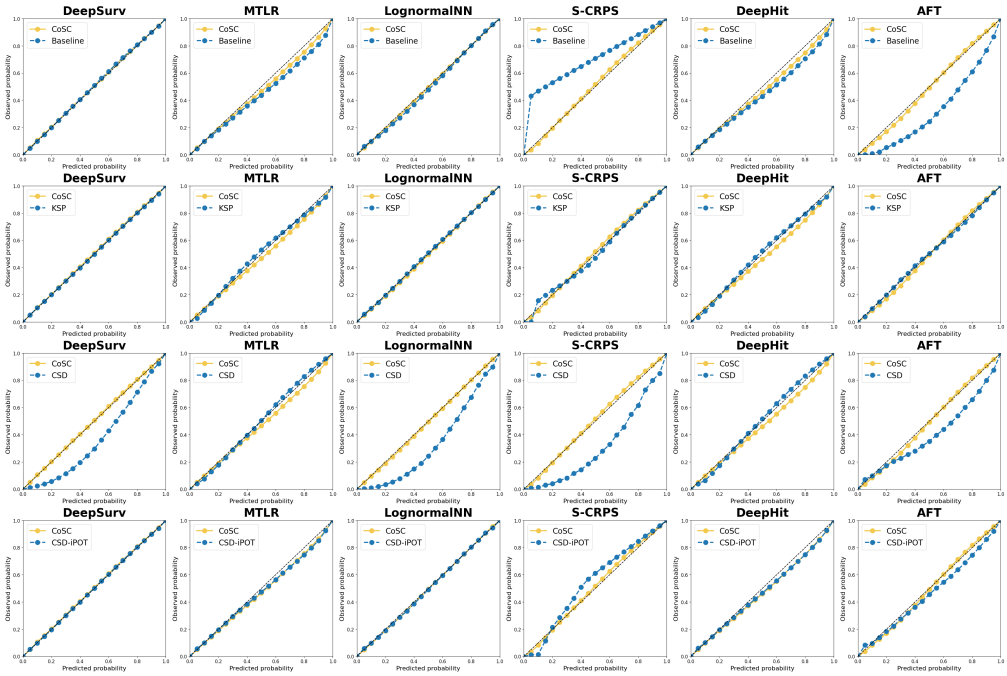

Figure 23: Calibration plots across six models for the WHAS dataset.

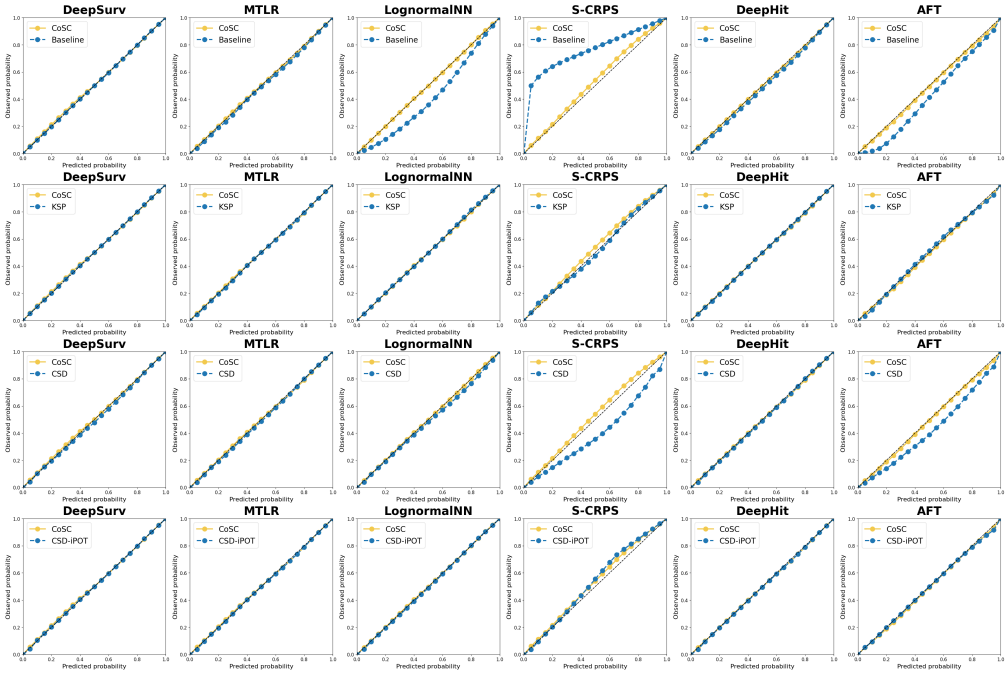

Figure 24: Calibration plots across six models for the METABRIC dataset.

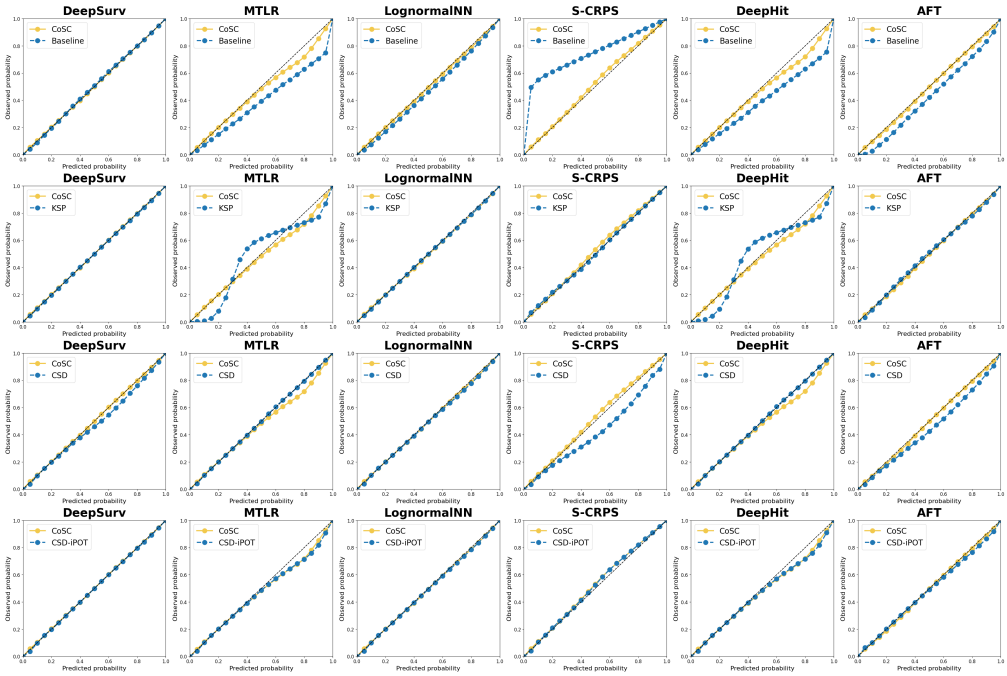

Figure 25: Calibration plots across six models for the GBSG dataset.

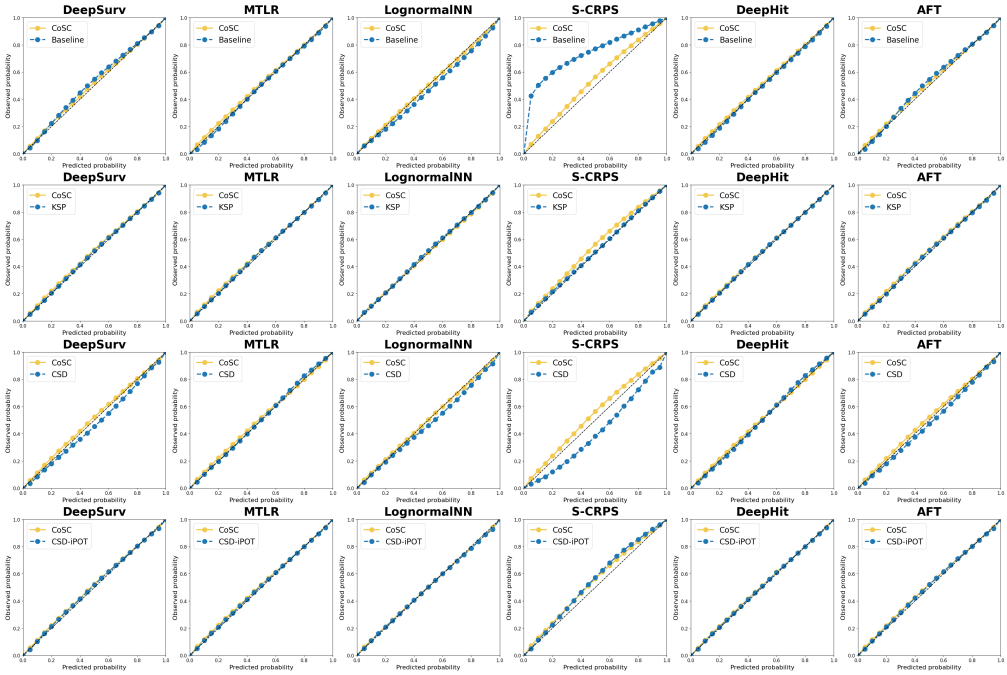

Figure 26: Calibration plots across six models for the NACD dataset.

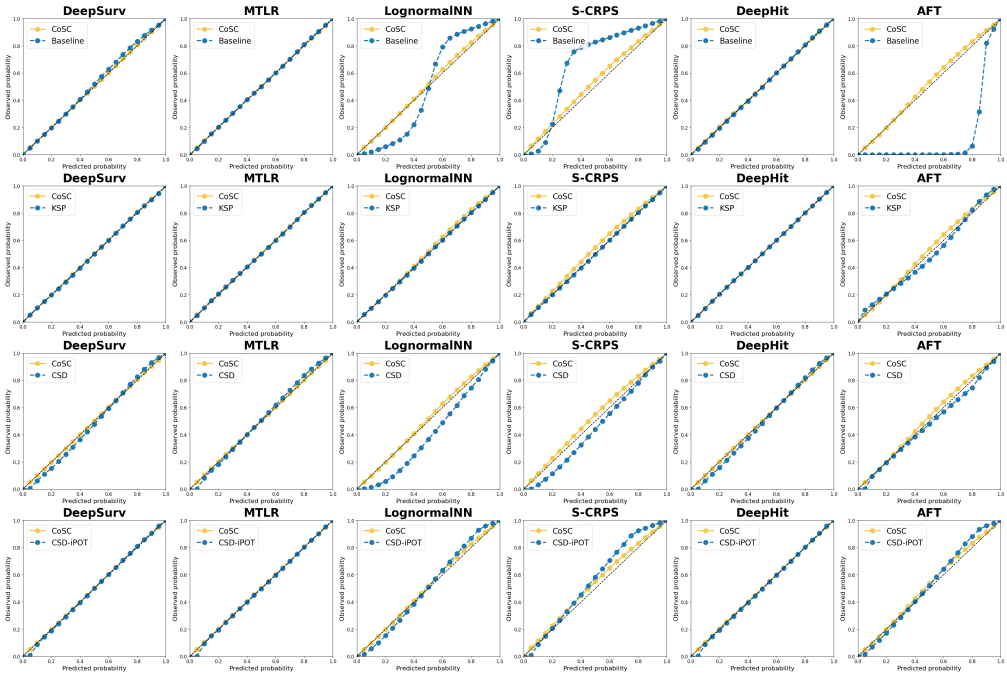

Figure 27: Calibration plots across six models for the NB-SEQ dataset.

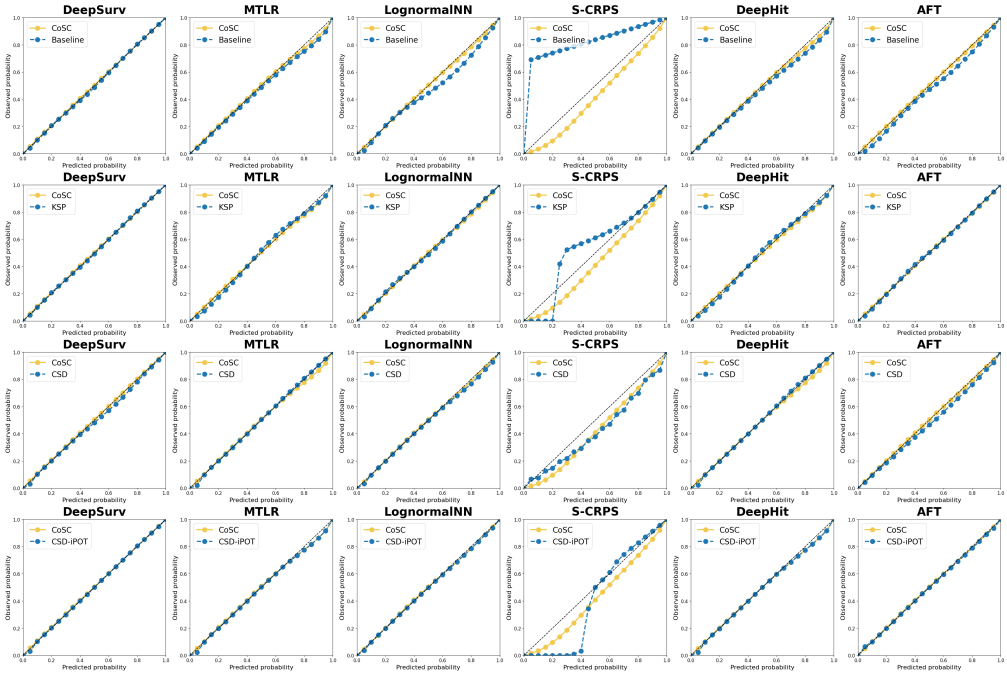

Figure 28: Calibration plots across six models for the SUPPORT dataset.

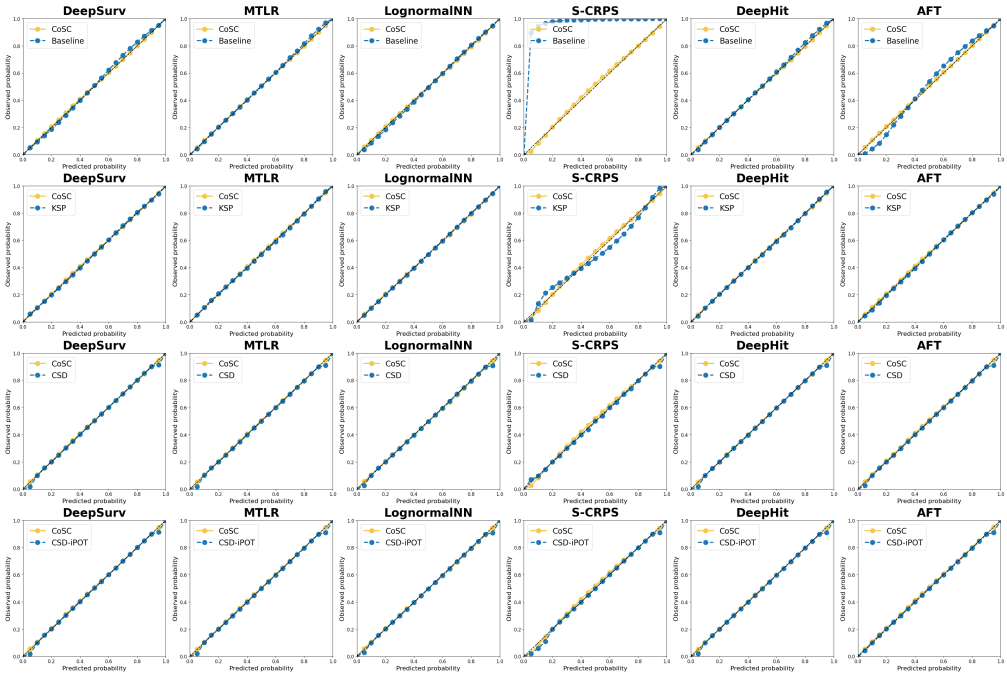

Figure 29: Calibration plots across six models for the MIMIC-III dataset.

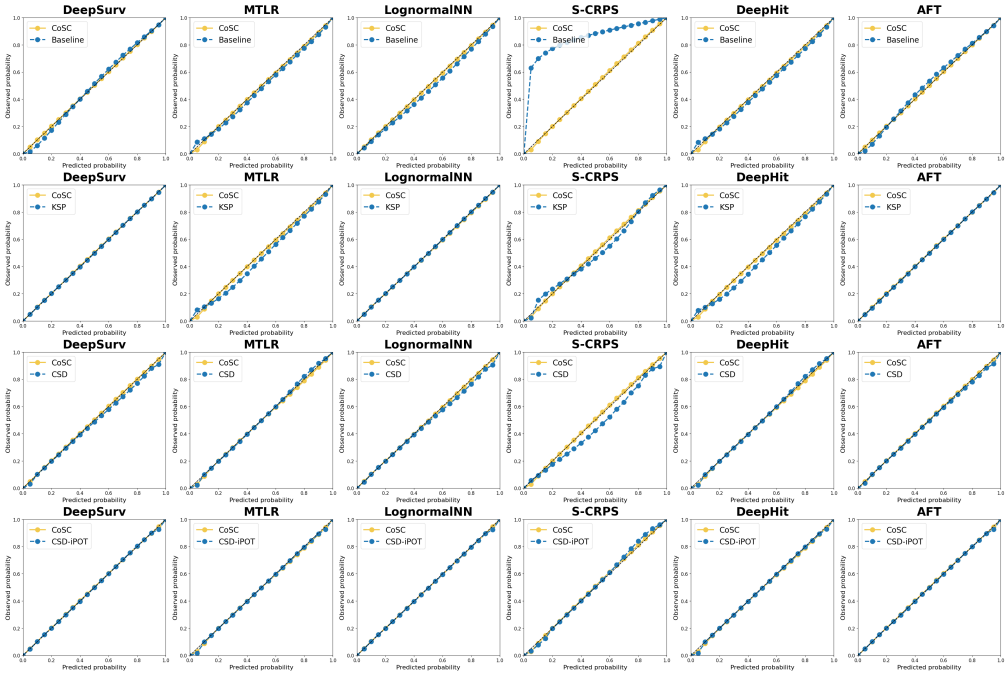

Figure 30: Calibration plots across six models for the SEER-liver dataset.

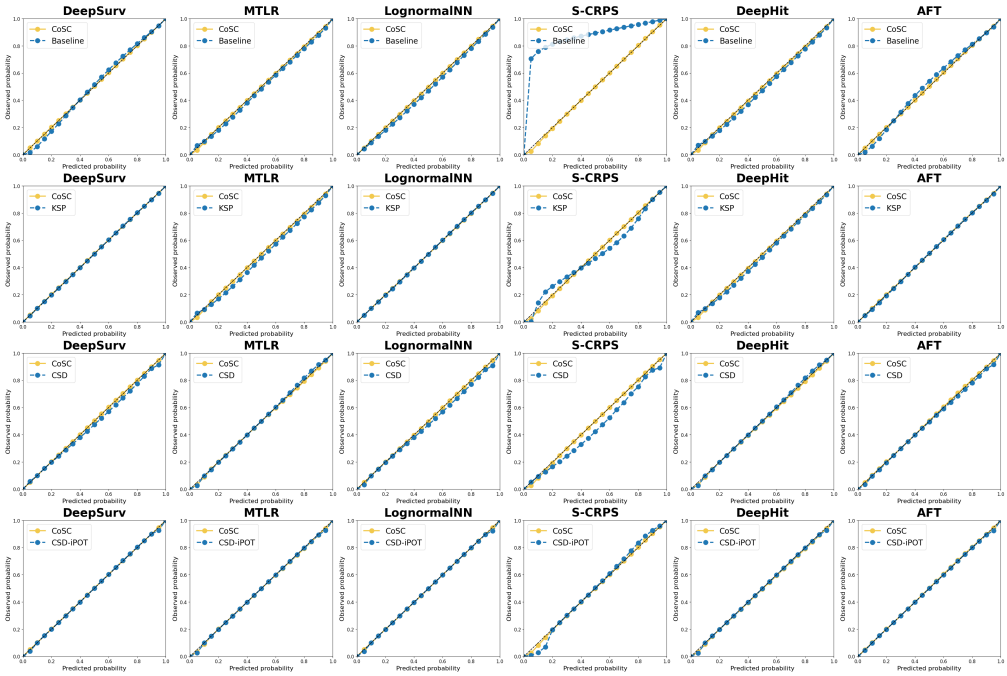

Figure 31: Calibration plots across six models for the SEER-stomach dataset.

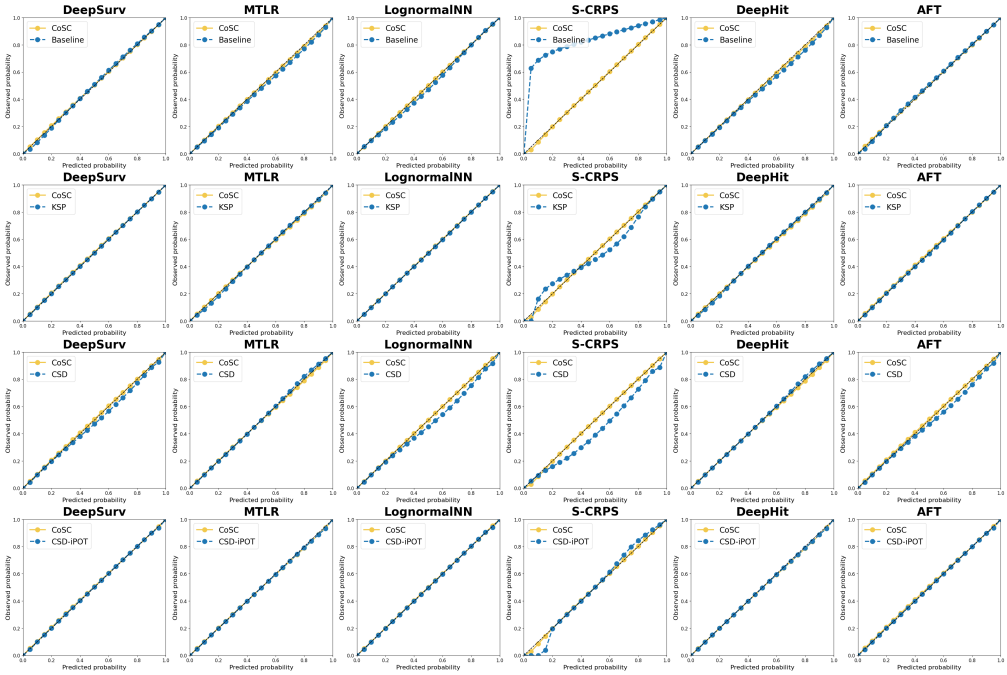

Figure 32: Calibration plots across six models for the SEER-bladder dataset.

F BASELINE MODELS

F.1 DEEPSURV

The classical Cox proportional hazards (PH) model (Cox, 1972), $\lambda_{\boldsymbol{z}}(t) = \lambda_0(t)\exp(\boldsymbol{\beta}^\top \boldsymbol{z})$, assumes a linear effect of the covariates $\boldsymbol{z}$. To relax this linearity assumption, we replace the linear predictor with a deep neural network (DNN). The hazard function is then modeled as $\lambda_{\boldsymbol{z}}(t) = \lambda_0(t)\exp\{g(\boldsymbol{z};\boldsymbol{\theta})\}$ where $g(\boldsymbol{z};\boldsymbol{\theta})$ is a nonlinear function parameterized by a DNN. The original Cox PH model optimizes the logarithm of the partial likelihood which is given by

$$\sum_{i=1}^{n} \delta_i \left[\boldsymbol{\beta}^\top \boldsymbol{z}_i - \log \sum_{j \in R_i} \exp(\boldsymbol{\beta}^\top \boldsymbol{z}_j) \right]$$

where $R_i = \{j : Y_j \geq Y_i\}$ is a risk set and $\boldsymbol{\beta}$ is the vector of regression coefficients. Maximizing this objective, or equivalently minimizing the negative log-partial likelihood, yields the estimator $\hat{\boldsymbol{\beta}}$. When adopting DeepSurv (Katzman et al., 2018), the negative log-partial likelihood becomes $-\sum_{i=1}^{n} \delta_i \left[g(\boldsymbol{z}_i;\boldsymbol{\theta}) - \log \sum_{j \in R_i} \exp\{g(\boldsymbol{z}_j;\boldsymbol{\theta})\} \right]$ which is used as the training loss. In DeepSurv, only the estimated risk scores $g(\boldsymbol{z}_i;\hat{\boldsymbol{\theta}})$ are required to compute discrimination metrics such as the C-index, since the baseline hazard $\lambda_0(t)$ is common across individuals at a fixed time t. However, to obtain the estimated distribution function $\hat{F}_{\boldsymbol{z}}(t)$, the baseline hazard must be specified. Rather than estimating $\lambda_0(t)$ directly, we estimate the cumulative baseline hazard $\Lambda_0(t)$ using the Breslow estimator (Lin, 2007), denoted by $\hat{\Lambda}_0(t)$. Using the relationship between the cumulative hazard and the distribution function, we obtain

$$\hat{F}_{\boldsymbol{z}}(t) = 1 - \exp\left[-\hat{\Lambda}_0(t)\exp\{g(\boldsymbol{z};\hat{\boldsymbol{\theta}})\} \right]$$

F.2 MTLR

Multi-task Logistic Regression (MTLR; Yu et al., 2011) models the probability of failure over discretized time intervals using a sequence of logistic functions. Let $(t_1, t_2, \ldots, t_B)$ denote a predefined set of ordered time points, and let $(s_1, s_2, \ldots, s_N)$ denote the observed survival times. For each subject, the survival status is represented by a binary vector $(y_1, y_2, \ldots, y_B)$ corresponding to these time points.

The probability of failure after the l-th time point is modeled as

$$\mathbb{P}(T > t_l \mid \boldsymbol{Z} = \boldsymbol{z}) = \left(1 + \exp\left(-\boldsymbol{\theta}_l^\top \boldsymbol{z}\right)\right)^{-1},$$

where $\boldsymbol{\theta}_l$ is the parameter vector associated with the l-th time interval.

When the event is observed at time s_i with $\delta_i = 1$, the corresponding survival status vector is encoded as

$$(y_1(s_i), \ldots, y_B(s_i)) = (\underbrace{0, \ldots, 0}_{l \text{ entries}}, 1, \ldots, 1),$$

where $t_l \leq s_i < t_{l+1}$. For censored observations ($\delta_i = 0$), the survival status vector is given by

$$(y_1(s_i), \ldots, y_B(s_i)) = (\underbrace{0, \ldots, 0}_{l \text{ entries}}).$$

For uncensored observations, the probability of observing the survival status vector is

$$f_{\boldsymbol{z}_i}(s_i) = \mathbb{P}_{\boldsymbol{\theta}}(Y = (y_1, \ldots, y_B) \mid \boldsymbol{Z} = \boldsymbol{z}_i) = \frac{\exp\left\{\sum_{k=1}^{B} y_k(s_i)\, \boldsymbol{\theta}_k^\top \boldsymbol{z}_i\right\}}{\sum_{k=0}^{B} \exp\left\{\sum_{j=k+1}^{B} \boldsymbol{\theta}_j^\top \boldsymbol{z}_i\right\}},$$

where $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_B)^\top$.

For censored observations, we only know that the failure time exceeds the observed censoring time s_i. Let t_l be the smallest time point such that $t_l > s_i$. The likelihood contribution of a censored subject is then given by

$$S_{\boldsymbol{z}_i}(s_i) = \mathbb{P}_{\boldsymbol{\theta}}(T_i \geq t_l \mid \boldsymbol{Z} = \boldsymbol{z}_i) = \frac{\sum_{k=l}^{B} \exp\left\{\sum_{j=k+1}^{B} \boldsymbol{\theta}_j^\top \boldsymbol{z}_i\right\}}{\sum_{k=0}^{B} \exp\left\{\sum_{j=k+1}^{B} \boldsymbol{\theta}_j^\top \boldsymbol{z}_i\right\}}.$$

Combining uncensored and censored observations, the negative log-likelihood used for training is defined as

$$\ell(\boldsymbol{\theta}) = -\sum_{i=1}^{n} \left[\delta_i \log f_{\boldsymbol{z}_i}(s_i) + (1-\delta_i) \log S_{\boldsymbol{z}_i}(s_i)\right].$$

Here, $f_{\boldsymbol{z}}$ and $S_{\boldsymbol{z}}$ denote the likelihood contributions defined by the MTLR model, rather than the true density and survival functions of the failure time. We refer readers to Yu et al. (2011) for further details.

For calibration evaluation, the estimated cumulative distribution function is obtained as

$$\hat{F}_{\boldsymbol{z}}(t) = \frac{\sum_{k=0}^{l} \exp\left\{\sum_{j=k+1}^{B} \hat{\boldsymbol{\theta}}_j^\top \boldsymbol{z}\right\}}{\sum_{k=0}^{B} \exp\left\{\sum_{j=k+1}^{B} \hat{\boldsymbol{\theta}}_j^\top \boldsymbol{z}\right\}}, \qquad t \in [t_l, t_{l+1}).$$

We set $B = 20$ for all experiments, except for the SEER datasets, where $B = 10$ is used.

F.3 LognormalNN

We use the LognormalNN proposed by Goldstein et al. (2020). This model uses a DNN to parameterize the location and scale (μ, σ) of a lognormal distribution. Let $f_{\boldsymbol{z}}$ and $S_{\boldsymbol{z}}$ denote the probability density function and survival function of the lognormal distribution, respectively, where the parameters are estimated by the DNN.

The model is trained by minimizing the negative log-likelihood,

$$\ell(\boldsymbol{\theta}) = -\sum_{i=1}^{n} \left[\delta_i \log f_{\boldsymbol{z}_i}(y_i) + (1-\delta_i) \log S_{\boldsymbol{z}_i}(y_i)\right],$$

where $\boldsymbol{\theta} = (\mu, \sigma)^\top$ and y_i denotes the observed time for subject i.

Given the estimated parameters $\hat{\boldsymbol{\theta}}$, the cumulative distribution function is evaluated at time t as

$$\hat{F}_{\boldsymbol{z}}(t) = \Phi\left(\frac{\log t - \hat{\mu}(\boldsymbol{z})}{\hat{\sigma}(\boldsymbol{z})}\right),$$

where $\Phi(\cdot)$ denotes the standard normal cumulative distribution function.

F.4 S-CRPS

The Survival Continuous Ranked Probability Score (S-CRPS; Avati et al., 2020) is defined as

$$\int_0^y \left\{\hat{F}_{\boldsymbol{z}}(t)\right\}^2 dt + \delta \int_y^\infty \left\{1 - \hat{F}_{\boldsymbol{z}}(t)\right\}^2 dt,$$

where y denotes the observed time and $\delta \in \{0, 1\}$ is the censoring indicator. For uncensored observations ($\delta = 1$), the score penalizes both early and late prediction errors, while for censored observations ($\delta = 0$), only the first term contributes.

For the lognormal distribution, a closed-form expression of the S-CRPS is available (see Appendix B in Avati et al., 2020). Therefore, we adopt the lognormal distribution for $\hat{F}_{\boldsymbol{z}}$ when optimizing the S-CRPS loss.

F.5 DeepHit

DeepHit (Lee et al., 2018) is a discrete-time survival model that directly estimates the probability mass function $\mathbb{P}(T = t \mid \boldsymbol{Z} = \boldsymbol{z})$ over a set of discretized time intervals. Let $T_{\max}$ denote the maximum time point. Given covariates $\boldsymbol{z}$, the model outputs a probability vector

$$\hat{\boldsymbol{p}} = (\hat{p}_1, \hat{p}_2, \ldots, \hat{p}_{T_{\max}}), \qquad \sum_{t=1}^{T_{\max}} \hat{p}_t = 1,$$

via a softmax layer.

The model is trained by minimizing the negative log-likelihood. For an uncensored observation with event time t, the loss is given by $-\log \hat{p}_t$. For a censored observation with censoring time t, the loss is $-\log \sum_{s=t+1}^{T_{\max}} \hat{p}_s$, which corresponds to the estimated survival probability $\hat{S}_{\boldsymbol{z}}(t) = \sum_{s=t+1}^{T_{\max}} \hat{p}_s$.

In this work, we adopt DeepHit in the single-event setting and train the model using only the log-likelihood loss, without the ranking loss originally proposed by Lee et al. (2018). Similar to MTLR, DeepHit depends on the choice of time discretization, and we use the same number of time bins as in MTLR.

F.6 Weibull AFT model

The Weibull accelerated failure time (AFT) model (Stute, 1993) assumes the regression form

$$\log T = \boldsymbol{\beta}^\top \boldsymbol{z} + \sigma\epsilon,$$

where ϵ follows a standard Gumbel distribution. Under this formulation, the failure time T follows a Weibull distribution with scale parameter $\exp(\boldsymbol{\beta}^\top \boldsymbol{z})$ and shape parameter $1/\sigma$.

The corresponding survival function is given by

$$S_{\boldsymbol{z}}(t) = \exp\left(-\left\{t \exp\left(-\boldsymbol{\beta}^\top \boldsymbol{z}\right)\right\}^{1/\sigma}\right).$$

The model parameters are estimated by maximizing the likelihood.

G Data preprocessing

We split each dataset into training, validation, and test sets while balancing censoring rates across the splits.

Table 7 presents a summary of each dataset. The following subsections describe the data preprocessing steps.

Table 7: Summary of datasets used in the experiments, including sample size, train/validation/test splits, censoring rate, and number of covariates including one-hot encoded categorical variables.

Group	Dataset	Training/Validation/Test	Censoring rate	# Covariates (after PCA)
Small	WHAS	982/328/328	58%	6 (5)
	METABRIC	1141/381/381	42%	9 (6)
	GBSG	1339/446/447	43%	7 (5)
	NACD	1437/479/480	36%	51 (14)
Medium	NB-SEQ	2976/992/992	31%	21 (5)
	SUPPORT	5323/1775/1775	32%	14 (9)
	MIMIC-III	5346/1782/1783	0%	27 (9)
Large	SEER-liver	10855/3618/3619	18%	59 (19)
	SEER-stomach	20361/6787/6788	18%	59 (23)
	SEER-bladder	33886/11295/11296	27%	59 (23)

G.1 Details on WHAS, METABRIC, GBSG, SUPPORT

The DeepSurv package (Katzman et al., 2018) in Python includes four pre-processed real-world datasets: the Worcester Heart Attack Study (WHAS), Molecular Taxonomy of Breast Cancer International Consortium (METABRIC), German Breast Cancer Study Group (GBSG), and the Study to Understand Prognoses Preferences Outcomes and Risks of Treatment (SUPPORT). To ensure consistency across experiments, we combined the training and test sets provided by the package and re-split the data into training, validation, and test sets in a 3:1:1 ratio. No additional data preprocessing was performed.

G.2 Details on NACD

The Northern Alberta Cancer Dataset (NACD; Qi et al., 2024a) includes patients diagnosed with various cancers, such as lung, colorectal, head and neck, esophageal, stomach, and other types of cancer. The outcome of interest is time to failure. We obtain the dataset from `http://pssp.srv.ualberta.ca` under the "Public Predictors" section. It consists of 2,396 patients with 51 features.

G.3 Details on NB-SEQ

NB-SEQ is a collection of neuroblastoma datasets and other supervised penalty learning benchmarks designed for censored regression problems arising in change-point detection. The dataset provides summary statistics as well as the minimum and maximum values of the penalty parameter λ, where λ controls the strength of the penalty function. The primary response variable, treated as a failure time, is the maximum estimated value of λ. In some instances, this maximum value is not directly observed; instead, it is only known to exceed a given threshold, resulting in right-censored observations.

All datasets are obtained from the public repository `https://github.com/tdhock/neuroblastoma-data` and merged into a single dataset. The final dataset consists of 4,960 sequence-level observations with 21 continuous variables. A detailed description of these variables is provided in Table 8.

Table 8: Variables used in the NB-SEQ dataset.

original scale (8)	**log.1. scale (7)**	**log. scale (3)**	**log.log. scale (3)**
quartile.25.	log.1.quartile.25.	·	·
quartile.50.	log.1.quartile.50.	·	·
quartile.75.	log.1.quartile.75.	·	·
quartile.100.	log.1.quartile.100.	log.quartile.100.	log.log.quartile.100.
mean	log.1.mean	log.mean	·
sd	log.1.sd	log.sd	·
bases	log.1.bases	·	log.log.bases
data	·	·	log.log.data

G.4 Details on MIMIC-III

MIMIC-III is a large, publicly available database containing de-identified health-related data from patients admitted to the Beth Israel Deaconess Medical Center (BIDMC) in Boston, Massachusetts, USA, with a primary focus on intensive care unit (ICU) admissions. We follow the preprocessing pipeline introduced by Harutyunyan et al. (2019), using the implementation provided in the MIMIC-III benchmark repository `https://github.com/YerevaNN/mimic3-benchmarks`.

The original dataset contains multiple length-of-stay records per subject, with repeated measurements collected at different time points. After excluding ICU transfers and patients younger than 18 years, the training and test sets consist of 2,925,434 and 525,912 instances, respectively. Since directly using these instances is inappropriate due to strong within-subject correlations, we apply additional preprocessing steps to construct a subject-level survival dataset.

Specifically, we first remove variables with excessively high missing rates. We then create a single instance per subject by defining the survival time as the duration between ICU admission and discharge, which yields uncensored outcomes. For variables measured repeatedly over time, we aggregate them into a single representative value using the mean for continuous variables and the mode for categorical variables.

We further exclude observations with implausible physiological measurements, retaining only subjects with diastolic blood pressure below 110 mmHg, body temperature greater than or equal to 30°C, and pH less than 10. After this final filtering step, the dataset consists of 8,911 observations with 14 variables, as summarized in Table 9.

Table 9: Variables used in the MIMIC-III dataset.

Continuous	**Range**
Diastolic blood pressure	31 – 102
Glascow coma scale total	3 – 15
Glucose	67 – 498
Heart rate	39 – 144
Mean blood pressure	43 – 128
Oxygen saturation	61 – 100
Respiratory rate	7 – 48
Systolic blood pressure	63 – 182
Temperature	31 – 40
Weight	0 – 280
pH	3 – 9
Categorical	**Category**
Glascow coma scale eye opening	No Response, To pain, To speech, Spontaneously
Glascow coma scale motor response	No Response, Abnorm extensn, Abnorm flexion, Flex-withdraws, Localizes Pain, Obeys Commands
Glascow coma scale verbal response	1.0 ET/Trach, No Response, Incomp sounds, Inapprop words, Confused, Oriented

G.5 Details on SEER-liver, SEER-stomach, SEER-bladder

The Surveillance, Epidemiology, and End Results (SEER) Program dataset, maintained by the National Cancer Institute (NCI), Division of Cancer Control and Population Sciences (DCCPS), Surveillance Research Program, is a large-scale, population-based cancer registry covering approximately 49% of the U.S. population (Qi et al., 2024a). The dataset contains information on cancer incidence, patient demographics, treatment characteristics, and survival outcomes.

We use the *Incidence – SEER Research Data, 17 Registries, Nov 2024 Sub (2000–2022)* release and select 11 variables, which are listed in Table 10. In this study, we construct three disease-specific cohorts from the SEER database: SEER-liver, SEER-stomach, and SEER-bladder, corresponding to patients diagnosed with liver, stomach, and urinary bladder cancers, respectively. Cancer sites are identified using the Site.recode.ICD.O.3.WHO.2008 variable. We restrict the analysis to patients aged 20 to 89 years at diagnosis.

The primary objective is to model the time from cancer diagnosis to death, with survival time measured in months. The dataset can be downloaded from `https://seer.cancer.gov/`.

Table 10: Variables used in the SEER dataset.

Continuous	**Range**
Age.recode.with.single.ages.and.90.	20 – 89
Categorical	**Category**
Sex	Male, Female
Grade.Recode..thru.2017.	Well differentiated; Grade I, Moderately differentiated; Grade II, Poorly differentiated; Grade III, Undifferentiated; anaplastic; Grade IV
Derived.AJCC.Stage.Group..6th.ed..2004.2015.	I, IA, IB, II, III, IIIA, IIIB, IIIC, IV
Derived.AJCC.T..6th.ed..2004.2015.	T1, T2, T2a, T2b, T3, T3a, T3b, T4, T4a, T4b
Derived.AJCC.N..6th.ed..2004.2015.	N0, N1, N2, N3
Derived.AJCC.M..6th.ed..2004.2015.	M0, M1
Race.recode..White..Black..Other.	White, Black, Other (American Indian/AK Native, Asian/Pacific Islander)
Marital.status.at.diagnosis	Divorced, Married (including common law), Separated, Single (never married), Unmarried or Domestic Partner, Widowed
Median.household.income.inflation.adj.to.2023	$< \$40,000$, $\$40,000 - \$44,999$, $\$45,000 - \$49,999$, $\$50,000 - \$54,999$, $\$55,000 - \$59,999$, $\$60,000 - \$64,999$, $\$65,000 - \$69,999$, $\$70,000 - \$74,999$, $\$75,000 - \$79,999$, $\$80,000 - \$84,999$, $\$85,000 - \$89,999$, $\$90,000 - \$94,999$, $\$95,000 - \$99,999$, $\$100,000 - \$109,999$, $\$110,000 - \$119,999$, $\$120,000+$
RX.Summ..Surv.Prim.Site..1998..	Yes, No

H DETAILS ON EXPERIMENTAL RESULTS

In this section, we present the experimental results for each dataset in tables.

Table 11: Performance on the WHAS dataset across survival models and calibration methods, averaged over 30 repetitions. Higher values are better for the C-index, whereas lower values are better for all other metrics.

Model	Method	C-index	PSR	KS-cal	KM-cal	IBS	$\mathbf{Cal_{ws}}$
DeepSurv	Baseline	0.8474	1.4710	0.0295	0.000541	0.1057	0.001264
	KSP	0.8474	1.4743	0.0296	0.000416	0.1057	0.001284
	CSD	0.8474	2.0162	0.2176	0.016417	0.1313	0.045826
	CSD-iPOT	0.8474	1.7401	0.0300	0.000447	0.1063	0.001004
	CoSC	0.8518	1.5061	0.0318	0.000603	0.1015	0.001208
MTLR	Baseline	0.8243	2.6071	0.0994	0.007860	0.1368	0.010256
	KSP	0.8241	2.1384	0.0566	0.008258	0.1408	0.013543
	CSD	0.8244	1.6828	0.0505	0.001880	0.1286	0.003806
	CSD-iPOT	0.8248	1.8450	0.0622	0.005976	0.1342	0.006234
	CoSC	0.8369	1.7422	0.0625	0.006220	0.1243	0.005858
LognormalNN	Baseline	0.8177	1.6626	0.0484	0.003852	0.1137	0.002282
	KSP	0.8463	1.6505	0.0428	0.004316	0.1151	0.002043
	CSD	0.8456	2.4508	0.2691	0.031948	0.1512	0.098808
	CSD-iPOT	0.8463	1.8239	0.0405	0.003794	0.1143	0.001962
	CoSC	0.8507	1.5108	0.0399	0.004113	0.1088	0.001511
S-CRPS	Baseline	0.8065	4.6159	0.3854	0.104501	0.2886	0.182457
	KSP	0.8079	2.1276	0.0851	0.002870	0.1516	0.024469
	CSD	0.8134	3.2582	0.2840	0.046611	0.1862	0.121443
	CSD-iPOT	0.8092	1.9491	0.1237	0.010125	0.1729	0.052954
	CoSC	0.8252	2.0184	0.0473	0.008082	0.1163	0.003920
DeepHit	Baseline	0.8453	2.4451	0.1075	0.007858	0.1161	0.009361
	KSP	0.8444	1.9447	0.0601	0.008404	0.1233	0.010235
	CSD	0.8443	1.6012	0.0578	0.002219	0.1117	0.005967
	CSD-iPOT	0.8444	1.7263	0.0645	0.006057	0.1144	0.005640
	CoSC	0.8493	1.6282	0.0653	0.006147	0.1097	0.004591
AFT	Baseline	0.5018	2.6601	0.2651	0.071081	0.3216	0.169215
	KSP	0.5018	2.4530	0.0369	0.014987	0.2489	0.057756
	CSD	0.5018	3.0456	0.1706	0.033690	0.2747	0.120101
	CSD-iPOT	0.5018	2.6177	0.0705	0.013462	0.2654	0.075999
	CoSC	0.7803	1.9813	0.0505	0.010784	0.1338	0.005690

Table 12: Performance on the METABRIC dataset across survival models and calibration methods, averaged over 30 repetitions. Higher values are better for the C-index, whereas lower values are better for all other metrics.

Model	Method	C-index	PSR	KS-cal	KM-cal	IBS	$\mathbf{Cal_{ws}}$
DeepSurv	Baseline	0.6458	2.1837	0.0259	0.000271	0.1623	0.001000
	KSP	0.6458	2.1854	0.0317	0.000351	0.1624	0.001026
	CSD	0.6458	2.2637	0.0419	0.000670	0.1634	0.001278
	CSD-iPOT	0.6458	2.2562	0.0325	0.000383	0.1627	0.000978
	CoSC	0.6475	2.2093	0.0373	0.000343	0.1614	0.001248
MTLR	Baseline	0.6222	2.2602	0.0455	0.001305	0.1686	0.001874
	KSP	0.6219	2.2572	0.0430	0.001230	0.1682	0.001809
	CSD	0.6217	2.2447	0.0418	0.000955	0.1676	0.001690
	CSD-iPOT	0.6216	2.2611	0.0416	0.001143	0.1681	0.001559
	CoSC	0.6276	2.2854	0.0421	0.001100	0.1665	0.001836
LognormalNN	Baseline	0.6050	2.3075	0.1553	0.011228	0.1802	0.014911
	KSP	0.6142	2.2479	0.0394	0.003931	0.1715	0.002434
	CSD	0.6162	2.3703	0.0523	0.002602	0.1726	0.003514
	CSD-iPOT	0.6148	2.2751	0.0419	0.003040	0.1711	0.002207
	CoSC	0.6235	2.2708	0.0397	0.003549	0.1691	0.001945
S-CRPS	Baseline	0.6211	4.4573	0.4695	0.213129	0.3873	0.129580
	KSP	0.6213	2.4310	0.0560	0.004690	0.1723	0.003350
	CSD	0.6215	3.9587	0.1726	0.026763	0.2227	0.081500
	CSD-iPOT	0.6213	2.3167	0.0881	0.013717	0.1821	0.007082
	CoSC	0.6332	2.5074	0.0621	0.012724	0.1760	0.003474
DeepHit	Baseline	0.6465	2.2188	0.0504	0.001573	0.1627	0.001565
	KSP	0.6458	2.2151	0.0402	0.001163	0.1618	0.001041
	CSD	0.6457	2.2131	0.0437	0.000930	0.1613	0.001040
	CSD-iPOT	0.6452	2.2547	0.0430	0.001277	0.1619	0.001106
	CoSC	0.6468	2.2619	0.0425	0.001125	0.1615	0.000964
AFT	Baseline	0.5317	2.9074	0.1392	0.033319	0.2507	0.041905
	KSP	0.5317	2.8215	0.0489	0.042999	0.2655	0.018482
	CSD	0.5317	3.6431	0.1245	0.024526	0.2545	0.053384
	CSD-iPOT	0.5317	3.4015	0.0512	0.038776	0.2661	0.021268
	CoSC	0.5490	2.9283	0.0445	0.033895	0.2466	0.016862

Table 13: Performance on the GBSG dataset across survival models and calibration methods, averaged over 30 repetitions. Higher values are better for the C-index, whereas lower values are better for all other metrics.

Model	Method	C-index	PSR	KS-cal	KM-cal	IBS	$\mathbf{Cal_{ws}}$
DeepSurv	Baseline	0.6766	2.3148	0.0281	0.000032	0.1750	0.001005
	KSP	0.6766	2.3150	0.0274	0.000204	0.1751	0.001073
	CSD	0.6766	2.4037	0.0648	0.001811	0.1773	0.002138
	CSD-iPOT	0.6766	2.3601	0.0286	0.000150	0.1751	0.001201
	CoSC	0.6766	2.3912	0.0300	0.000072	0.1751	0.001127
MTLR	Baseline	0.6573	4.8747	0.2055	0.027197	0.2120	0.026047
	KSP	0.6556	3.0542	0.1563	0.015556	0.2074	0.015700
	CSD	0.6571	2.4068	0.0410	0.000444	0.1853	0.004641
	CSD-iPOT	0.6576	2.5554	0.1104	0.002913	0.1881	0.009879
	CoSC	0.6698	2.5241	0.0866	0.002944	0.1838	0.005914
LognormalNN	Baseline	0.6494	2.3779	0.0584	0.002375	0.1805	0.003969
	KSP	0.6636	2.3699	0.0375	0.000543	0.1788	0.001970
	CSD	0.6638	2.4406	0.0446	0.001028	0.1795	0.002650
	CSD-iPOT	0.6636	2.4498	0.0417	0.000666	0.1790	0.002297
	CoSC	0.6673	2.4472	0.0400	0.000466	0.1780	0.001929
S-CRPS	Baseline	0.6538	4.3805	0.4566	0.140649	0.3351	0.126826
	KSP	0.6544	2.4894	0.0441	0.000537	0.1912	0.008255
	CSD	0.6544	3.7741	0.1441	0.010247	0.2183	0.080858
	CSD-iPOT	0.6543	2.4128	0.0499	0.000882	0.1918	0.009543
	CoSC	0.6612	2.5592	0.0497	0.001190	0.1837	0.004399
DeepHit	Baseline	0.6732	4.8441	0.2027	0.026166	0.2069	0.023768
	KSP	0.6720	3.0300	0.1581	0.016116	0.2067	0.015548
	CSD	0.6728	2.3887	0.0427	0.000539	0.1819	0.003168
	CSD-iPOT	0.6731	2.5413	0.1098	0.002966	0.1849	0.008297
	CoSC	0.6772	2.5029	0.0841	0.002981	0.1815	0.005095
AFT	Baseline	0.5048	2.9484	0.1014	0.015341	0.2701	0.058896
	KSP	0.5048	2.8750	0.0403	0.012944	0.2641	0.023377
	CSD	0.5048	3.2618	0.0941	0.008652	0.2640	0.050456
	CSD-iPOT	0.5048	3.2709	0.0452	0.010918	0.2698	0.032195
	CoSC	0.5678	3.0812	0.0380	0.007568	0.2288	0.014125

Table 14: Performance on the NACD dataset across survival models and calibration methods, averaged over 30 repetitions. Higher values are better for the C-index, whereas lower values are better for all other metrics.

Model	Method	C-index	PSR	KS-cal	KM-cal	IBS	$\mathbf{Cal_{ws}}$
DeepSurv	Baseline	0.7486	2.0151	0.0636	0.001971	0.1417	0.002824
	KSP	0.7486	2.0062	0.0471	0.001102	0.1406	0.001762
	CSD	0.7486	2.4268	0.0713	0.001147	0.1425	0.005481
	CSD-iPOT	0.7486	2.3591	0.0510	0.001003	0.1413	0.001731
	CoSC	0.7483	2.0386	0.0541	0.001559	0.1411	0.002166
MTLR	Baseline	0.7411	2.1714	0.0379	0.004045	0.1511	0.001714
	KSP	0.7411	2.1540	0.0383	0.003829	0.1507	0.001782
	CSD	0.7409	2.1212	0.0416	0.001235	0.1483	0.001884
	CSD-iPOT	0.7410	2.2653	0.0394	0.004061	0.1504	0.001449
	CoSC	0.7423	2.1473	0.0445	0.003489	0.1494	0.001735
LognormalNN	Baseline	0.7352	2.0971	0.0635	0.001365	0.1502	0.003711
	KSP	0.7357	2.0813	0.0428	0.003044	0.1509	0.001594
	CSD	0.7357	2.6131	0.0657	0.001149	0.1509	0.003861
	CSD-iPOT	0.7357	2.4886	0.0445	0.001726	0.1509	0.001576
	CoSC	0.7369	2.1066	0.0414	0.002101	0.1491	0.001717
S-CRPS	Baseline	0.7225	3.4026	0.4129	0.135928	0.3006	0.097755
	KSP	0.7299	2.1023	0.0428	0.003009	0.1497	0.002326
	CSD	0.7280	3.2874	0.1369	0.026194	0.1873	0.044466
	CSD-iPOT	0.7289	2.1845	0.0923	0.011545	0.1592	0.006444
	CoSC	0.7347	2.2649	0.0770	0.011738	0.1577	0.004431
DeepHit	Baseline	0.7499	2.1204	0.0388	0.004108	0.1468	0.002186
	KSP	0.7498	2.1107	0.0386	0.003945	0.1463	0.002113
	CSD	0.7498	2.0736	0.0475	0.001853	0.1436	0.001631
	CSD-iPOT	0.7497	2.2519	0.0409	0.004390	0.1462	0.001699
	CoSC	0.7503	2.1067	0.0422	0.003637	0.1453	0.001907
AFT	Baseline	0.7509	2.0431	0.0628	0.001154	0.1421	0.002151
	KSP	0.7509	2.0283	0.0409	0.000599	0.1414	0.001581
	CSD	0.7509	2.3847	0.0521	0.000624	0.1423	0.002402
	CSD-iPOT	0.7509	2.3437	0.0447	0.000652	0.1417	0.002104
	CoSC	0.7514	2.0621	0.0472	0.001004	0.1416	0.001828

Table 15: Performance on the NB-SEQ dataset across survival models and calibration methods, averaged over 30 repetitions. Higher values are better for the C-index, whereas lower values are better for all other metrics.

Model	Method	C-index	PSR	KS-cal	KM-cal	IBS	$\mathbf{Cal_{ws}}$
DeepSurv	Baseline	0.7746	1.7398	0.0443	0.000304	0.0408	0.001536
	KSP	0.7746	1.7411	0.0244	0.000349	0.0406	0.000953
	CSD	0.7746	1.7332	0.0604	0.001112	0.0416	0.002700
	CSD-iPOT	0.7746	1.7597	0.0510	0.000712	0.0410	0.000921
	CoSC	0.7756	1.7313	0.0246	0.000277	0.0401	0.000740
MTLR	Baseline	0.6964	1.6956	0.0279	0.000215	0.0454	0.001653
	KSP	0.6966	1.6961	0.0292	0.000252	0.0454	0.002382
	CSD	0.6965	1.8073	0.0586	0.000662	0.0460	0.002430
	CSD-iPOT	0.6966	1.8542	0.0561	0.000678	0.0456	0.002133
	CoSC	0.7039	1.7109	0.0279	0.000217	0.0448	0.001503
LognormalNN	Baseline	0.7393	2.3125	0.2137	0.052813	0.0835	0.044615
	KSP	0.7393	1.7092	0.0279	0.016548	0.0511	0.002714
	CSD	0.7393	2.0799	0.1697	0.018404	0.0590	0.031047
	CSD-iPOT	0.7393	1.7948	0.0835	0.023813	0.0536	0.004859
	CoSC	0.7375	1.7960	0.0387	0.018958	0.0502	0.001164
S-CRPS	Baseline	0.7805	2.4663	0.4091	0.108563	0.1512	0.119097
	KSP	0.7805	1.5727	0.0283	0.002205	0.0401	0.001156
	CSD	0.7805	2.1599	0.0934	0.012667	0.0594	0.013141
	CSD-iPOT	0.7805	1.7467	0.1414	0.019365	0.0529	0.013186
	CoSC	0.7124	1.7662	0.0600	0.016262	0.0494	0.003287
DeepHit	Baseline	0.7709	1.5273	0.0282	0.000285	0.0375	0.000502
	KSP	0.7711	1.5259	0.0262	0.000250	0.0375	0.000594
	CSD	0.7715	1.6431	0.0641	0.000739	0.0381	0.001476
	CSD-iPOT	0.7716	1.6721	0.0580	0.000517	0.0378	0.000614
	CoSC	0.7718	1.5563	0.0272	0.000244	0.0377	0.000661
AFT	Baseline	0.4832	4.2422	0.7483	0.229975	0.2052	0.226137
	KSP	0.4832	2.5089	0.0632	0.042400	0.1145	0.057106
	CSD	0.4832	2.1918	0.0619	0.001927	0.0628	0.027263
	CSD-iPOT	0.4832	2.4238	0.0899	0.057707	0.1169	0.031379
	CoSC	0.5545	2.2919	0.0514	0.025587	0.0787	0.007198

Table 16: Performance on the SUPPORT dataset across survival models and calibration methods, averaged over 30 repetitions. Higher values are better for the C-index, whereas lower values are better for all other metrics.

Model	Method	C-index	PSR	KS-cal	KM-cal	IBS	$\mathbf{Cal_{ws}}$
DeepSurv	Baseline	0.6098	1.8310	0.0180	0.000018	0.1915	0.000439
	KSP	0.6098	1.8311	0.0179	0.000077	0.1915	0.000474
	CSD	0.6098	1.9360	0.0378	0.000397	0.1921	0.000821
	CSD-iPOT	0.6098	1.9021	0.0272	0.000135	0.1916	0.000403
	CoSC	0.6098	1.8299	0.0202	0.000076	0.1913	0.000361
MTLR	Baseline	0.5711	2.2404	0.0609	0.012407	0.2212	0.002124
	KSP	0.5705	2.1025	0.0393	0.007807	0.2165	0.001097
	CSD	0.5708	1.8716	0.0325	0.000380	0.2088	0.000688
	CSD-iPOT	0.5713	2.1652	0.0381	0.007832	0.2164	0.001188
	CoSC	0.5722	1.9625	0.0367	0.007126	0.2157	0.001082
LognormalNN	Baseline	0.5873	1.8592	0.0907	0.007358	0.2012	0.003264
	KSP	0.6025	1.8287	0.0307	0.000227	0.1946	0.000489
	CSD	0.6037	2.1378	0.0357	0.004803	0.1990	0.000762
	CSD-iPOT	0.6025	2.0388	0.0243	0.001184	0.1946	0.000501
	CoSC	0.6027	1.8319	0.0269	0.000702	0.1937	0.000391
S-CRPS	Baseline	0.5615	10.5213	0.6575	0.397427	0.6115	0.178123
	KSP	0.5689	2.9455	0.2471	0.012937	0.2208	0.025281
	CSD	0.5687	3.1422	0.1336	0.023658	0.2596	0.065969
	CSD-iPOT	0.5682	1.9811	0.3810	0.003353	0.2149	0.031921
	CoSC	0.5408	3.2344	0.1203	0.003277	0.2435	0.008718
DeepHit	Baseline	0.6036	2.2039	0.0683	0.013113	0.2116	0.001882
	KSP	0.6025	2.0539	0.0374	0.007600	0.2060	0.000687
	CSD	0.6028	1.8851	0.0322	0.001071	0.1989	0.000343
	CSD-iPOT	0.6026	2.1291	0.0371	0.008456	0.2061	0.000666
	CoSC	0.6040	1.9246	0.0351	0.007240	0.2052	0.000628
AFT	Baseline	0.5402	2.0076	0.0593	0.001028	0.2473	0.018005
	KSP	0.5402	2.0085	0.0282	0.005139	0.2477	0.011620
	CSD	0.5402	2.3882	0.0506	0.000692	0.2479	0.014988
	CSD-iPOT	0.5402	2.3273	0.0262	0.003481	0.2491	0.011762
	CoSC	0.5407	2.0242	0.0234	0.004583	0.2473	0.011127

Table 17: Performance on the MIMIC-III dataset across survival models and calibration methods, averaged over 30 repetitions. Higher values are better for the C-index, whereas lower values are better for all other metrics.

Model	Method	C-index	PSR	KS-cal	KM-cal	IBS	$\mathbf{Cal_{ws}}$
DeepSurv	Baseline	0.7042	1.6182	0.0378	0.000051	0.0322	0.000818
	KSP	0.7042	1.6185	0.0214	0.000051	0.0319	0.000455
	CSD	0.7042	2.2887	0.0759	0.000084	0.0323	0.000536
	CSD-iPOT	0.7042	2.3099	0.0768	0.000109	0.0321	0.000467
	CoSC	0.7045	1.6265	0.0250	0.000047	0.0318	0.000542
MTLR	Baseline	0.6637	1.6874	0.0296	0.000365	0.0346	0.000819
	KSP	0.6653	1.6847	0.0301	0.000256	0.0343	0.001530
	CSD	0.6698	2.4151	0.0800	0.000175	0.0344	0.001504
	CSD-iPOT	0.6709	2.4289	0.0802	0.000197	0.0342	0.001391
	CoSC	0.6676	1.7137	0.0313	0.000183	0.0341	0.001323
LognormalNN	Baseline	0.7057	1.5952	0.0308	0.000076	0.0316	0.000725
	KSP	0.7059	1.5976	0.0273	0.000088	0.0316	0.000889
	CSD	0.7069	2.3669	0.0817	0.000188	0.0319	0.000951
	CSD-iPOT	0.7070	2.3752	0.0817	0.000200	0.0319	0.000928
	CoSC	0.7062	1.6253	0.0310	0.000071	0.0316	0.000788
S-CRPS	Baseline	0.6365	6.9314	0.8577	0.265042	0.3763	0.302535
	KSP	0.6589	1.9976	0.0799	0.003972	0.0444	0.020160
	CSD	0.6434	3.4308	0.0968	0.070494	0.1803	0.075899
	CSD-iPOT	0.6618	2.4655	0.0949	0.002055	0.0440	0.016051
	CoSC	0.6601	1.9469	0.0418	0.001571	0.0408	0.005848
DeepHit	Baseline	0.7031	1.6174	0.0324	0.000431	0.0326	0.000703
	KSP	0.7047	1.6137	0.0277	0.000267	0.0322	0.000703
	CSD	0.7069	2.3100	0.0786	0.000180	0.0325	0.000556
	CSD-iPOT	0.7079	2.3477	0.0799	0.000171	0.0322	0.000655
	CoSC	0.7063	1.6564	0.0290	0.000156	0.0319	0.000605
AFT	Baseline	0.6581	1.7061	0.0725	0.000309	0.0349	0.002872
	KSP	0.6581	1.7084	0.0302	0.000193	0.0348	0.001562
	CSD	0.6581	2.4805	0.0802	0.000204	0.0350	0.002161
	CSD-iPOT	0.6581	2.5067	0.0805	0.000330	0.0352	0.001426
	CoSC	0.6624	1.7061	0.0330	0.000140	0.0345	0.001238

Table 18: Performance on the SEER-liver dataset across survival models and calibration methods, averaged over 30 repetitions. Higher values are better for the C-index, whereas lower values are better for all other metrics.

Model	Method	C-index	PSR	KS-cal	KM-cal	IBS	$\mathbf{Cal_{ws}}$
DeepSurv	Baseline	0.7367	2.0850	0.0432	0.000031	0.1216	0.000620
	KSP	0.7367	2.0817	0.0153	0.000069	0.1213	0.000299
	CSD	0.7367	2.7765	0.0618	0.000236	0.1223	0.000685
	CSD-iPOT	0.7367	2.6051	0.0440	0.000203	0.1219	0.000308
	CoSC	0.7365	2.0917	0.0152	0.000039	0.1213	0.000173
MTLR	Baseline	0.7385	2.1804	0.0692	0.002591	0.1291	0.000946
	KSP	0.7385	2.1785	0.0773	0.002767	0.1296	0.002572
	CSD	0.7384	2.2535	0.0300	0.001493	0.1272	0.000885
	CSD-iPOT	0.7383	2.5358	0.0395	0.002565	0.1285	0.000389
	CoSC	0.7371	2.1580	0.0245	0.001613	0.1260	0.000266
LognormalNN	Baseline	0.7331	2.0953	0.0484	0.000447	0.1228	0.001799
	KSP	0.7347	2.0884	0.0180	0.000212	0.1229	0.000390
	CSD	0.7349	2.8223	0.0632	0.000778	0.1236	0.001036
	CSD-iPOT	0.7348	2.6414	0.0475	0.000126	0.1233	0.000409
	CoSC	0.7350	2.0972	0.0188	0.000098	0.1223	0.000245
S-CRPS	Baseline	0.6906	5.2138	0.6016	0.214703	0.3823	0.209633
	KSP	0.7177	2.2232	0.0647	0.002120	0.1350	0.006478
	CSD	0.7117	3.0866	0.0917	0.032687	0.1822	0.083434
	CSD-iPOT	0.7158	2.2851	0.0586	0.002315	0.1341	0.004743
	CoSC	0.7239	2.2396	0.0270	0.001126	0.1282	0.000439
DeepHit	Baseline	0.7383	2.1776	0.0692	0.002794	0.1291	0.001430
	KSP	0.7383	2.1758	0.0800	0.002940	0.1298	0.003321
	CSD	0.7384	2.2619	0.0301	0.001543	0.1272	0.001168
	CSD-iPOT	0.7383	2.5387	0.0382	0.002688	0.1284	0.000548
	CoSC	0.7367	2.1545	0.0250	0.001618	0.1260	0.000312
AFT	Baseline	0.7376	2.0990	0.0412	0.000370	0.1225	0.001186
	KSP	0.7376	2.0902	0.0167	0.000498	0.1227	0.000370
	CSD	0.7376	2.7310	0.0552	0.000571	0.1231	0.000512
	CSD-iPOT	0.7376	2.6033	0.0426	0.000902	0.1234	0.000361
	CoSC	0.7378	2.0927	0.0176	0.000312	0.1218	0.000300

Table 19: Performance on the SEER-stomach dataset across survival models and calibration methods, averaged over 30 repetitions. Higher values are better for the C-index, whereas lower values are better for all other metrics.

Model	Method	C-index	PSR	KS-cal	KM-cal	IBS	$\mathbf{Cal_{ws}}$
DeepSurv	Baseline	0.7365	2.0703	0.0413	0.000033	0.1162	0.000687
	KSP	0.7365	2.0638	0.0117	0.000066	0.1160	0.000213
	CSD	0.7365	2.7466	0.0581	0.000338	0.1169	0.001059
	CSD-iPOT	0.7365	2.6163	0.0457	0.000230	0.1166	0.000212
	CoSC	0.7364	2.0657	0.0115	0.000040	0.1159	0.000138
MTLR	Baseline	0.7354	2.1514	0.0551	0.001814	0.1233	0.000923
	KSP	0.7354	2.1510	0.0612	0.001998	0.1237	0.001687
	CSD	0.7360	2.3069	0.0259	0.001316	0.1220	0.000545
	CSD-iPOT	0.7357	2.5273	0.0343	0.001891	0.1230	0.000354
	CoSC	0.7355	2.1251	0.0191	0.001080	0.1203	0.000198
LognormalNN	Baseline	0.7347	2.0762	0.0355	0.000381	0.1171	0.000962
	KSP	0.7353	2.0720	0.0144	0.000176	0.1171	0.000211
	CSD	0.7357	2.8287	0.0634	0.000815	0.1182	0.000991
	CSD-iPOT	0.7356	2.6648	0.0492	0.000255	0.1178	0.000210
	CoSC	0.7355	2.0673	0.0133	0.000151	0.1166	0.000163
S-CRPS	Baseline	0.7095	6.0901	0.6646	0.237483	0.4072	0.235902
	KSP	0.7197	2.2496	0.0825	0.002090	0.1356	0.014077
	CSD	0.7218	3.2204	0.0932	0.035836	0.1773	0.086972
	CSD-iPOT	0.7207	2.2966	0.0954	0.001097	0.1321	0.009997
	CoSC	0.7214	2.2388	0.0273	0.000769	0.1240	0.000264
DeepHit	Baseline	0.7379	2.1428	0.0551	0.001886	0.1222	0.000970
	KSP	0.7378	2.1387	0.0571	0.001756	0.1220	0.000962
	CSD	0.7380	2.3161	0.0260	0.001470	0.1209	0.001010
	CSD-iPOT	0.7379	2.5368	0.0341	0.002068	0.1218	0.000323
	CoSC	0.7367	2.1181	0.0203	0.001096	0.1197	0.000241
AFT	Baseline	0.7326	2.1213	0.0447	0.000936	0.1205	0.003009
	KSP	0.7326	2.1041	0.0157	0.000994	0.1207	0.000955
	CSD	0.7326	2.7364	0.0525	0.001275	0.1210	0.001743
	CSD-iPOT	0.7326	2.6101	0.0416	0.001613	0.1214	0.000900
	CoSC	0.7343	2.0866	0.0135	0.000544	0.1182	0.000360

Table 20: Performance on the SEER-bladder dataset across survival models and calibration methods, averaged over 30 repetitions. Higher values are better for the C-index, whereas lower values are better for all other metrics.

Model	Method	C-index	PSR	KS-cal	KM-cal	IBS	$\mathbf{Cal_{ws}}$
DeepSurv	Baseline	0.7111	2.4114	0.0200	0.000018	0.1521	0.000991
	KSP	0.7111	2.4087	0.0083	0.000034	0.1520	0.001099
	CSD	0.7111	2.8181	0.0379	0.000480	0.1527	0.001850
	CSD-iPOT	0.7111	2.7139	0.0232	0.000091	0.1523	0.000980
	CoSC	0.7112	2.4099	0.0124	0.000030	0.1517	0.000570
MTLR	Baseline	0.7121	2.5353	0.0319	0.003107	0.1608	0.002323
	KSP	0.7121	2.5119	0.0235	0.002207	0.1599	0.001307
	CSD	0.7121	2.5045	0.0225	0.000606	0.1579	0.001380
	CSD-iPOT	0.7121	2.6257	0.0171	0.002683	0.1597	0.001198
	CoSC	0.7122	2.4542	0.0140	0.002200	0.1585	0.000878
LognormalNN	Baseline	0.7058	2.4065	0.0309	0.000891	0.1526	0.000841
	KSP	0.7123	2.4028	0.0110	0.000690	0.1524	0.000280
	CSD	0.7123	2.9183	0.0606	0.001295	0.1535	0.003406
	CSD-iPOT	0.7123	2.6879	0.0220	0.000484	0.1526	0.000237
	CoSC	0.7121	2.4067	0.0099	0.000619	0.1522	0.000161
S-CRPS	Baseline	0.6804	6.4378	0.5901	0.243727	0.4291	0.229332
	KSP	0.6898	2.6890	0.0925	0.003007	0.1741	0.018595
	CSD	0.6888	3.6568	0.1134	0.016746	0.1929	0.080237
	CSD-iPOT	0.6896	2.5794	0.1156	0.001167	0.1731	0.013843
	CoSC	0.6963	2.6799	0.0231	0.001475	0.1617	0.000193
DeepHit	Baseline	0.7137	2.5270	0.0400	0.003311	0.1598	0.002261
	KSP	0.7134	2.4987	0.0230	0.002102	0.1585	0.001094
	CSD	0.7136	2.4848	0.0219	0.000705	0.1564	0.000962
	CSD-iPOT	0.7133	2.6019	0.0171	0.002712	0.1583	0.001192
	CoSC	0.7134	2.4481	0.0159	0.002186	0.1578	0.000961
AFT	Baseline	0.7101	2.4245	0.0204	0.000234	0.1530	0.002611
	KSP	0.7101	2.4207	0.0130	0.000324	0.1533	0.000937
	CSD	0.7101	2.8793	0.0466	0.000922	0.1536	0.001460
	CSD-iPOT	0.7101	2.7249	0.0238	0.000437	0.1534	0.000991
	CoSC	0.7116	2.4200	0.0148	0.000156	0.1513	0.000218

I ADDITIONAL EXPERIMENTAL DETAILS

Table 21 reports the post-processing time (in seconds) for each method across all datasets. The hyperparameter settings used for training are summarized in Table 22. All baseline models are optimized using Adam with early stopping (patience $= 200$) for a maximum of 1,000 epochs. KSP and the bandwidth of CoSC are also optimized using Adam with early stopping, with initial learning rates of 0.05 and 0.03 and patience values of 100 and 30, respectively.

Table 21: Average post-processing time (in seconds) for KSP, CSD, CSD-iPOT, and CoSC across datasets.

Dataset	Model	KSP	CSD	CSD-iPOT	CoSC	Dataset	Model	KSP	CSD	CSD-iPOT	CoSC
WHAS	DeepSurv	1.2011	1.7254	0.1252	7.3195	SUPPORT	DeepSurv	1.1126	4.1944	0.9524	9.3217
	MTLR	0.8477	0.5481	0.1180	7.1091		MTLR	2.0651	3.0585	1.0409	7.3132
	LognormalNN	3.8643	0.4678	0.1129	5.3072		LognormalNN	3.3763	3.9206	1.0130	6.9110
	S-CRPS	1.3110	0.4415	0.1056	5.4360		S-CRPS	2.3519	2.9236	1.0170	6.5821
	DeepHit	0.8583	0.4983	0.1179	15.9061		DeepHit	2.0640	3.2628	1.0242	7.2815
	AFT	1.6062	6.4603	0.1235	5.0193		AFT	2.5976	15.9069	1.0089	12.0806
METABRIC	DeepSurv	1.1066	0.7741	0.1423	5.1089	MIMIC-III	DeepSurv	0.7626	2.9585	1.0031	7.5440
	MTLR	1.5039	0.6986	0.1355	4.8017		MTLR	0.8157	2.4249	1.0333	18.5731
	LognormalNN	2.1184	1.0676	0.1314	5.2198		LognormalNN	0.9140	2.2967	0.9929	7.1406
	S-CRPS	1.4400	4.6612	0.1281	5.4302		S-CRPS	2.5289	3.2692	0.9782	5.9641
	DeepHit	1.4005	0.5590	0.1351	5.4909		DeepHit	0.9101	3.3095	1.0158	5.7713
	AFT	3.3409	7.1739	0.1424	4.7558		AFT	2.8691	2.3788	0.9739	8.7695
GBSG	DeepSurv	1.1504	1.3862	0.1651	5.4861	SEER-liver	DeepSurv	1.9927	12.0879	3.0299	5.1945
	MTLR	2.9246	0.6521	0.1616	9.5363		MTLR	0.5109	7.0877	3.2641	5.0920
	LognormalNN	1.5073	0.6281	0.1557	5.2565		LognormalNN	1.1488	8.7236	2.8742	5.1357
	S-CRPS	3.7893	6.6056	0.1533	6.3324		S-CRPS	2.2022	6.2566	2.8380	4.9293
	DeepHit	3.0794	0.6534	0.1616	5.4953		DeepHit	0.5386	7.1931	3.2719	4.7031
	AFT	3.5171	8.4815	0.1655	5.0122		AFT	3.1282	12.1593	2.8781	8.7695
NACD	DeepSurv	2.3581	2.9688	0.1783	5.4050	SEER-stomach	DeepSurv	2.0365	18.5026	8.8857	5.2541
	MTLR	2.0958	1.4573	0.1762	5.8382		MTLR	0.5024	17.8313	9.9224	6.3034
	LognormalNN	1.1654	1.7420	0.1697	5.2742		LognormalNN	0.9937	15.9514	8.5106	6.6146
	S-CRPS	1.6330	0.6578	0.1677	4.7998		S-CRPS	3.6561	15.2560	8.3838	6.1212
	DeepHit	1.6637	1.1311	0.1755	5.6180		DeepHit	0.7262	17.7450	9.9201	6.5467
	AFT	2.7846	2.1216	0.1770	4.6045		AFT	3.5403	33.4430	8.4879	4.7169
NB-SEQ	DeepSurv	0.8647	1.5946	0.4125	4.7538	SEER-bladder	DeepSurv	1.6569	52.3438	22.4989	23.4078
	MTLR	1.2099	1.6137	0.4080	4.9890		MTLR	0.5619	42.7505	26.0342	12.3083
	LognormalNN	2.3688	12.3861	0.3979	5.1710		LognormalNN	1.8362	54.2306	21.8329	13.3554
	S-CRPS	3.5731	6.1111	0.4043	5.2778		S-CRPS	1.9737	69.4247	21.5958	10.1362
	DeepHit	1.0447	1.6114	0.4288	5.0513		DeepHit	0.5610	43.1028	26.1103	15.1663
	AFT	3.8948	11.5441	0.4095	5.8035		AFT	1.9492	47.9892	21.8772	9.7856

Table 22: Hyperparameter settings used in the experiments.

Dataset	Model	Batch size	Learning rate	Dropout rate	Hidden layer sizes	Number of bins
WHAS	DeepSurv	50	1e-4	0.1	24/12/12	-
	MTLR	50	1e-2	-	-	20
	LognormalNN	50	1e-3	0.1	24/12/12	-
	S-CRPS	100	1e-2	0.1	24/12/12	-
	DeepHit	50	1e-3	0.1	24/12/12	20
	AFT	50	1e-2	-	-	-
METABRIC	DeepSurv	64	1e-4	0.3	36/18/18	-
	MTLR	64	1e-2	-	-	20
	LognormalNN	64	1e-3	0.3	36/18/18	-
	S-CRPS	128	1e-2	0.3	36/18/18	-
	DeepHit	64	1e-3	0.3	36/18/18	20
	AFT	64	1e-2	-	-	-
GBSG	DeepSurv	64	1e-4	0.3	28/14/14	-
	MTLR	64	1e-2	-	-	20
	LognormalNN	64	1e-3	0.3	28/14/14	-
	S-CRPS	128	1e-2	0.3	28/14/14	-
	DeepHit	64	1e-3	0.3	28/14/14	20
	AFT	64	1e-2	-	-	-
NACD	DeepSurv	64	1e-4	0.1	204/102/102	-
	MTLR	64	1e-2	-	-	20
	LognormalNN	64	1e-3	0.1	204/102/102	-
	S-CRPS	128	1e-2	0.1	204/102/102	-
	DeepHit	64	1e-3	0.1	204/102/102	20
	AFT	64	1e-2	-	-	-
NB-SEQ	DeepSurv	128	1e-4	0.1	84/42/42	-
	MTLR	128	1e-2	-	-	20
	LognormalNN	128	1e-3	0.1	84/42/42	-
	S-CRPS	256	1e-2	0.1	84/42/42	-
	DeepHit	128	1e-3	0.1	84/42/42	20
	AFT	128	1e-2	-	-	-
SUPPORT	DeepSurv	256	1e-4	0.3	56/28/28	-
	MTLR	256	1e-2	-	-	20
	LognormalNN	256	1e-3	0.3	56/28/28	-
	S-CRPS	512	1e-2	0.3	56/28/28	-
	DeepHit	256	1e-3	0.3	56/28/28	20
	AFT	256	1e-2	-	-	-
MIMIC-III	DeepSurv	256	1e-4	0.1	108/54/54	-
	MTLR	256	1e-2	-	-	20
	LognormalNN	256	1e-3	0.1	108/54/54	-
	S-CRPS	512	1e-2	0.1	108/54/54	-
	DeepHit	256	1e-3	0.1	108/54/54	20
	AFT	256	1e-2	-	-	-
SEER-liver	DeepSurv	512	1e-4	0.1	236/118/118	-
	MTLR	512	1e-2	-	-	10
	LognormalNN	512	1e-3	0.1	236/118/118	-
	S-CRPS	1024	1e-2	0.1	236/118/118	-
	DeepHit	512	1e-3	0.1	236/118/118	10
	AFT	512	1e-2	-	-	-
SEER-stomach	DeepSurv	1024	1e-4	0.1	236/118/118	-
	MTLR	1024	1e-2	-	-	10
	LognormalNN	1024	1e-3	0.1	236/118/118	-
	S-CRPS	2048	1e-2	0.1	236/118/118	-
	DeepHit	1024	1e-3	0.1	236/118/118	10
	AFT	1024	1e-2	-	-	-
SEER-bladder	DeepSurv	2048	1e-4	0.1	236/118/118	-
	MTLR	2048	1e-2	-	-	10
	LognormalNN	2048	1e-3	0.1	236/118/118	-
	S-CRPS	4096	1e-2	0.1	236/118/118	-
	DeepHit	2048	1e-3	0.1	236/118/118	10
	AFT	2048	1e-2	-	-	-