AMORTIZED SURVIVAL DISTRIBUTION ESTIMATION VIA IN-CONTEXT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Survival analysis models time-to-event (TTE) outcomes in the presence of *censoring*. Most survival models require dataset-specific fitting, whereas prior-fitted networks (PFNs) enable in-context prediction on new datasets without gradient-based adaptation. However, existing survival PFNs provide limited theoretical guidance for the design of their pretraining objective, *context-size* distribution, and synthetic *prior*. We introduce TTE-PFN, an amortized in-context estimator that maps a labeled *right-censored context* and unlabeled *query* covariates to predictive *event-* and *censoring-time distributions* in a single forward pass. TTE-PFN combines *oracle-distillation* supervision from *analytic* conditional distributions, randomized *context sizes*, and a diverse *prior* over survival data-generating processes. We first show that the *oracle-distillation* and matched *sampled-label* objectives share the same *posterior*-predictive population minimizer, but the latter incurs additional positive-semidefinite gradient covariance, which *oracle distillation* removes to facilitate faster and more stable optimization. We further establish *posterior*-predictive optimality across all represented *context sizes* and derive an excess-risk bound decomposing predictive error into *amortization*, *prior coverage*, and *posterior concentration*. Across 115 survival datasets and seven metrics that span *predictive accuracy*, *concordance*, and *calibration*, TTE-PFN achieves the best aggregate rank against 28 competing methods and ranks first on the *predictive accuracy* metrics. TTE-PFN also has the lowest runtime in our evaluation protocol, requiring only 0.052 seconds to predict $1{,}000$ individuals.

1 INTRODUCTION

Survival analysis is an important statistical framework for modeling time-to-event (TTE) distributions, with broad applications in medicine, reliability engineering, and economics (Curtis et al., 2012; Emmerson & Brown, 2021). Most existing survival models can be grouped into several major modeling families. *Classical survival models*, such as Cox proportional-hazards and accelerated failure-time models, impose structured assumptions on the hazard or event-time distribution (Cox, 1972; Norman et al., 2024). *Tree-based methods*, including gradient-boosted survival models and random survival forests, capture nonlinear effects and covariate interactions through recursive partitioning and ensemble learning (Friedman, 2001; Ishwaran et al., 2008). *Neural discrete-time models* represent the event-time distribution over predefined intervals (Lee et al., 2018; Pearce et al., 2022), whereas *neural continuous-time models* directly model continuous event-time distributions, often through flexible parameterizations of an underlying distributional family (Hoseini et al., 2017; Sheng & Henao, 2025). Despite their different assumptions and parameterizations, these approaches generally require dataset-specific preprocessing, optimization, and hyperparameter selection. In contrast, tabular foundation models offer a different paradigm by allowing a single pretrained model to make predictions on new datasets without gradient-based adaptation (Hollmann et al., 2025). Prior-fitted networks (PFNs) achieve this by pretraining on synthetic tasks drawn from a *prior* and amortizing Bayesian inference through in-context learning (Müller et al., 2022). More recently, SurvivalPFN (Qi et al., 2026) demonstrated the feasibility of this paradigm for *right-censored* survival data.

However, the design of SurvivalPFN remains largely empirical, with limited theoretical guidance on what *posterior* quantity the network learns, how its pretraining objective affects optimization, how training at a fixed *context size* transfers to cohorts of different sizes, or how the synthetic *prior* controls performance on real datasets. Our analysis reveals substantial room for improvement along

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.

each of these dimensions. First, *sampled-label* supervision introduces avoidable gradient variance despite access to the *analytic* conditional distributions of the synthetic data-generating processes (DGPs). Second, training at a fixed *context size* imposes no population-risk guarantee at other *context sizes*. Third, limited *prior* coverage creates a persistent mismatch between synthetic training tasks and real survival cohorts. Motivated by these observations, we propose the **time-to-event PFN (TTE-PFN)**, a theoretically grounded PFN for amortized survival-distribution estimation. TTE-PFN combines *oracle-distillation* supervision, randomized *context sizes*, and a diverse mixture of survival DGPs. We further establish an excess-risk decomposition that connects these design choices to three distinct sources of predictive error: *amortization error*, *prior-coverage error*, and residual uncertainty from finite-context *posterior concentration*. Our contributions are summarized below.

- We formulate in-context survival prediction as amortized Bayesian inference and introduce TTE-PFN, which maps a labeled *right-censored context* and unlabeled *query* covariates to predictive *event-* and *censoring-time distributions* in a single forward pass.
- We develop a theoretically guided pretraining framework combining *oracle distillation*, randomized *context sizes*, and a diverse *prior* over *analytic* survival DGPs. We further show that *oracle distillation* preserves the *posterior*-predictive population minimizer while removing *label-sampling* gradient variance, establish optimality across all *context sizes* represented during training, and derive an excess-risk bound decomposing predictive error into *amortization*, *prior coverage*, and *posterior concentration*.
- We evaluate TTE-PFN against 28 competing methods on 115 survival datasets using seven metrics that span *predictive accuracy*, *concordance*, and *calibration*. TTE-PFN achieves the best aggregate rank, ranks first on the *predictive accuracy* metrics, outperforms SurvivalPFN across all seven metrics, and is the fastest method under our evaluation protocol, requiring only 0.052 seconds per $1{,}000$ individuals and running approximately $4.5\times$ faster than SurvivalPFN.

2 Background

Survival Analysis. Survival analysis models time-to-event (TTE) outcomes from partially observed data. Each subject has an event time T and a censoring time C, of which *right censoring* reveals only the earlier, $Y = \min(T, C)$, together with an event indicator $\delta = \mathbb{I}\{T \le C\}$ that records whether Y is an observed event ($\delta = 1$) or not. A survival dataset $\mathcal{D}$ is therefore a finite set of triplets $(\mathbf{x}_i, y_i, \delta_i)$ with covariates $\mathbf{x}_i$, and the target is the conditional event-time distribution $p_T(\cdot|\mathbf{x})$, or accordingly, the survival function $S_T(\cdot|\mathbf{x})$. Here, we focus on *right censoring*, the dominant regime in clinical and reliability data, while the less common *left-* and *interval-censored* cases can be handled by the same machinery (Klein & Moeschberger, 2006). Finally, as is standard in survival analysis, we assume *conditionally independent censoring*, $T \perp C \mid X$, so that once the covariates are known the censoring mechanism carries no further information about the event time.

Survival Data-Generating Processes (Survival DGPs). A survival data-generating process (DGP), indexed by $\theta \in \Theta$, is a triple of laws $(P_X^\theta, P_T^\theta, P_C^\theta)$, including a covariate law, a conditional event law, and a conditional censoring law. Together, they define a joint law P^θ over (X, T, C) that factorizes as $X \sim P_X^\theta$, $T \mid X \sim P_T^\theta$, $C \mid X \sim P_C^\theta$, and a survival dataset $\mathcal{D}$ is then drawn by *i.i.d.* sampling $(X, T, C) \sim P^\theta$. A single θ thus describes one cohort, and one draw from P^θ yields one survival dataset. In practice, we will observe diverse cohorts rather than a single one, each governed by its own θ. Thus, we model this by treating θ itself as random, drawn from a *prior* π (*i.e.*, a distribution over Θ) over DGPs. In the end, sampling $\theta \sim \pi$ and then $\mathcal{D} \sim P^\theta$ therefore generates a whole population of synthetic survival datasets. Details are given in Section 3.2 and Appendix C.3.

Bayesian Survival Inference. Consider a new dataset, or cohort, split into a labeled *context set* $\mathcal{D}_{\text{ctx}} = \{(\mathbf{x}_i, y_i, \delta_i)\}_{i=1}^n$ and a *query set* $\mathcal{D}_{\text{query}} = \{\mathbf{x}_j^*\}_{j=1}^m$ containing the covariate points at which predictions are required. Given a prior π over DGPs, the Bayesian predictive distribution of the *event time* at a query point $\mathbf{x}^* \in \mathcal{D}_{\text{query}}$ is the *posterior predictive distribution* (PPD),

$$q_{\text{PPD},T}(\cdot|\mathbf{x}^*, \mathcal{D}_{\text{ctx}}) = \mathbb{E}_{\theta|\mathcal{D}_{\text{ctx}},\mathbf{x}^*}\left[P_T^\theta(\cdot|\mathbf{x}^*)\right] = \int_\Theta P_T^\theta(\cdot|\mathbf{x}^*)\,\Pi(\mathrm{d}\theta|\mathcal{D}_{\text{ctx}}, \mathbf{x}^*)\,, \tag{1}$$

where $\Pi(\mathrm{d}\theta|\mathcal{D}_{\text{ctx}}, \mathbf{x}^*)$ is the *posterior distribution* over possible DGPs after observing the labeled *context* and the current unlabeled *query* covariate, which satisfies $\Pi(\mathrm{d}\theta|\mathcal{D}_{\text{ctx}}, \mathbf{x}^*) \propto p_X^\theta(\mathbf{x}^*)\, P_{\text{obs}}^\theta(\mathcal{D}_{\text{ctx}})\, \pi(\mathrm{d}\theta)$, where P_{obs}^θ is the law induced by P^θ on the observable triple (X, Y, δ)

under *right censoring*. Intuitively, the PPD averages the *conditional event-time distributions* associated with all possible DGPs, weighting each one by its *posterior* plausibility after observing $(\mathcal{D}_{\text{ctx}}, \mathbf{x}^*)$. Because this *posterior average* is generally intractable (Müller et al., 2022), we build on the prior-fitted network (Hollmann et al., 2025) and propose TTE-PFN to amortize its computation for *right-censored* survival data. This is done by learning a categorical approximation to the *event-time* PPD in (1) over context-dependent time bins by training on *synthetic* tasks drawn from a *prior* over survival DGPs. In the next section, **Lemma 1** identifies this PPD as the population minimizer of our objective, while **Theorem 1** bounds the predictive error on a new cohort.

3 Time to Event Prior Fitted Networks (TTE-PFN)

TTE-PFN is an amortized, in-context estimator for *right-censored* survival distributions. It ingests a *context set* together with *query* covariates and emits, in a single forward pass, discretized predictive distributions for both *event* and *censoring* times. Beyond the estimator itself, we prove that the predictive error of TTE-PFN decomposes into three terms (**Theorem 1**): an *amortization* **term (I)**, measuring the gap between the network and the *posterior* predictive; a *prior-coverage* **term (II)**, measuring the mismatch between the evaluation DGP and the *posterior's* pseudo-true DGP inside the *synthetic prior*; and a *posterior-concentration* **term (III)**, measuring the residual uncertainty left by a finite *context*. The architecture and the *oracle-distillation* loss (Section 3.1) control **(I)**, the synthetic *prior* (Section 3.2) shrinks **(II)**, and **(III)** vanishes as the *context* grows (Section 3.3).

3.1 Learning with TTE-PFN

TTE-PFN amortizes the posterior predictive distribution in (1) with a single pretrained network.

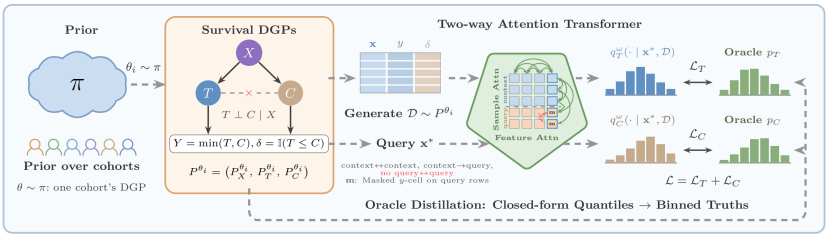


Figure 1: Overview of TTE-PFN. A *two-way attention* transformer learns *event-* and *censoring-time distributions* from *analytic* synthetic targets.

During pretraining, we repeatedly draw a survival DGP $\theta \sim \pi$, sample a labeled *context*, and train on held-out *query* covariates using the DGP's *analytic* conditional distributions as supervision (Figure 1). At inference, the frozen network reads the labeled split of a new cohort as *context* and predicts at unlabeled *query* points without gradient-based adaptation.

Network Architecture. TTE-PFN adopts the *two-way* (*feature-* and *sample-axis*) *attention* transformer of TabPFN (Hollmann et al., 2025), which is permutation-equivariant across rows and handles variable-width tables. It tokenizes the table cell by cell, with *feature-axis attention* capturing cross-covariate interactions and *sample-axis attention* carrying label information from *context* rows to the *query*. We adapt this backbone to survival settings with two changes, both visible in Figure 1. First, each row carries a target token encoding its censored label (y_i, δ_i) for the *context* rows and a learned mask for the *query* row, preventing label leakage. Second, the *query* token feeds two softmax heads, one for the *event distribution* and one auxiliary head for the *censoring distribution*.

Training Objective. We supervise the above network on the *analytic conditional distributions* of each DGP rather than on sampled outcomes. Since both heads predict over bins, we use a *context-dependent discretization* $\kappa_{\mathcal{D}_{\text{ctx}}} : \mathbb{R}_+ \to \{1, \dots, L\}$, fitted from $\mathcal{D}_{\text{ctx}}$ alone, to map times into L bins (see Appendix C.3 for details). Suppressing the arguments $(\mathbf{x}^*, \theta; \mathcal{D}_{\text{ctx}})$, let $p_T, p_C \in \Delta_{L-1}$ be the resulting *binned conditional distributions* for event and censoring times at query $\mathbf{x}^*$ under DGP θ:

$$[p_T]_\ell = \Pr_{T \sim P_T^\theta(\cdot|\mathbf{x}^*)} \left[\kappa_{\mathcal{D}_{\text{ctx}}}(T) = \ell\right], \qquad [p_C]_\ell = \Pr_{C \sim P_C^\theta(\cdot|\mathbf{x}^*)} \left[\kappa_{\mathcal{D}_{\text{ctx}}}(C) = \ell\right]. \tag{2}$$

for $\ell = 1, \dots, L$, both probabilities are available in closed form from the *analytic* conditional CDFs in (5) below. On the network side, the prediction is a pair of softmax outputs over the same L bins, q_T^ω and q_C^ω, where ω denotes the network weights. In the end, we train the network by *cross-entropy* from each soft target to its prediction, averaged over the *prior*, the sampled *context*, and the *query*,

$$\mathcal{L}_{\text{OD},T}(\omega) = \mathbb{E}_{\theta \sim \pi}\, \mathbb{E}_{N \sim P_N}\, \mathbb{E}_{\mathcal{D}_{\text{ctx}} \sim (P^\theta_{\text{obs}})^{\otimes N}}\, \mathbb{E}_{\mathbf{x}^* \sim P_X^\theta} \left[\text{CE}(p_T \,\|\, q_T^\omega)\right], \tag{3}$$

where P_N is the log-uniform distribution over *context* sizes on $[n_{\min}, n_{\max}]$ (Appendix C.3), $(P^{\theta}_{\text{obs}})^{\otimes N}$ denotes N *i.i.d.* draws from P^{θ}_{obs}, $\text{CE}(p_T \,\|\, q^{\omega}_T) = -\sum_{\ell=1}^{L} [p_T]_\ell \log [q^{\omega}_T]_\ell$ with $[\cdot]_\ell$ denoting the bin-ℓ probability mass as in (2), and $\mathcal{L}_{\text{OD},C}$ is defined similarly against p_C. Combining the event and censoring channels gives the *oracle-distillation* objective,

$$\mathcal{L}_{\text{OD}}(\omega) = \mathcal{L}_{\text{OD},T}(\omega) + \lambda_c \, \mathcal{L}_{\text{OD},C}(\omega), \tag{4}$$

where λ_c weights the auxiliary censoring head. In practice, we set $\lambda_c{=}1$ throughout (Appendix D.2). Compared with SurvivalPFN (Qi et al., 2026), the closest prior work, we train on full *analytic* conditionals rather than *sampled labels* and randomize *context sizes* rather than fixing them. The former changes the stochastic gradient estimator, while the latter broadens cohort-size coverage. Their effects are analyzed below and evaluated in Table 2 and Appendix Figure 7.

To isolate the supervision change, let $\mathcal{L}^{\text{match}}_{\text{SL},T}$ denote the *sampled-label* objective obtained from (3) by replacing p_T with the one-hot vector $\mathbf{e}_{\kappa_{\mathcal{D}_{\text{ctx}}}(\tilde{t})}$ for $\tilde{t} \sim P^{\theta}_T(\cdot|\mathbf{x}^*)$, while keeping the same *prior*, *context-size*, *discretization*, and *query*. The censoring counterpart is defined analogously and combined as in (4). The fixed-context *sampled-label* training further replaces P_N by a point mass at $N = n_{\text{fix}}$, where n_{fix} denotes the fixed *context size* used during pretraining. For simplicity, we state the following argument for the event head and drop the channel subscript. Since the one-hot target is unbiased for the *binned conditional distribution* p and *cross-entropy* is linear in its target (Gneiting & Raftery, 2007), for every fixed $(\theta, \mathcal{D}_{\text{ctx}}, \mathbf{x}^*)$, we have $\mathbb{E}_{\tilde{t}}\Big[\text{CE}\Big(\mathbf{e}_{\kappa_{\mathcal{D}_{\text{ctx}}}(\tilde{t})} \| q^{\omega}\Big)\Big] = \text{CE}(p \| q^{\omega})$.

Consequently, $\mathcal{L}^{\text{match}}_{\text{SL}}$ and $\mathcal{L}_{\text{OD}}$ share the same population risk, differing only in gradient variance because the one-hot target $\mathbf{e}_{\kappa_{\mathcal{D}_{\text{ctx}}}(\tilde{t})}$ is a one-sample Monte Carlo draw from p. The following results formalize these claims: **Lemma 1** identifies the common Bayes predictor, **Corollary 1** establishes *context-size* coverage, and **Lemma 2** quantifies the excess *label-sampling* covariance.

Lemma 1 (Population Minimizer, adapted from Müller et al. 2022)**.** *For either channel, the unique minimizer of $\mathcal{L}_{\text{OD}}$ over all measurable predictors $(\mathcal{D}_{\text{ctx}}, \mathbf{x}^*) \mapsto \Delta_{L-1}$ is the binned posterior predictive $q_{\text{PPD}} = \mathbb{E}_{\theta|\mathcal{D}_{\text{ctx}},\mathbf{x}^*}[p_\theta]$, and $\mathcal{L}^{\text{match}}_{\text{SL}}$ has the same population minimizer.*

Corollary 1 (Coverage Across Context Sizes)**.** *Let $\mathcal{N} = \{n : P_N(n) > 0\}$. Any population minimizer of $\mathcal{L}_{\text{OD}}$ equals $q_{\text{PPD}} = \mathbb{E}_{\theta|\mathcal{D}_{\text{ctx}},\mathbf{x}^*}[p_\theta]$ almost surely conditional on $N = n$, for every $n \in \mathcal{N}$. In contrast, training only at $N = n_{\text{fix}}$ imposes no population-risk constraint at $n \neq n_{\text{fix}}$.*

Lemma 2 (Label-Sampling Variance Gap)**.** *Let p denote the binned conditional target distribution for the current DGP, context, and query, and let g_{OD} and g_{SL} denote the corresponding logit gradients. Then g_{SL} is conditionally unbiased for g_{OD}, and $\text{Cov}(g_{\text{SL}}) = \text{Cov}(g_{\text{OD}}) + \mathbb{E}\big[\text{diag}(p) - pp^{\top}\big]$. The excess covariance is positive semidefinite and vanishes exactly when p is a point mass.*

Intuitively, both objectives share the same population target, while *sampled-label* supervision adds zero-mean gradient noise with covariance $\text{diag}(p) - pp^{\top}$, which is positive semidefinite and vanishes when p is a point mass. *Oracle distillation* removes this noise, facilitating faster and more stable optimization, while randomized *context sizes* extend the population guarantee across the support of P_N. Figure 3 examines optimization empirically and proofs are in Appendices A.2, A.3, and A.4.

3.2 Synthetic Survival DGPs

Oracle distillation requires synthetic DGPs whose conditional distributions are analytically available, while **term (II)** of our excess-risk decomposition motivates a *prior* that broadly covers real cohorts. We therefore design the prior along three complementary axes: *structure*, *discrimination*, and *breadth*, and instantiate them with five complementary generators. Specifically, we construct a five-family mixture $\pi = \sum_k w_k \pi_k$ over π_1: `Tree`, π_2: `DeepMLP`, π_3: `Narrow`, π_4: `Parametric`, and π_5: `Bernstein`, each admitting analytic conditional CDFs. The complete constructions are provided in Appendix C.3, and the corresponding leave-one-family-out ablation is in Appendix D.2.

DGP Pipeline. A DGP θ first draws a covariate law P^{θ}_X, varying dimension, marginal shape, scale, and correlation, and then independently draws event- and censoring-time conditionals of the form

$$P^{\theta}_T(\cdot|\mathbf{x}) = \mathcal{F}_T\big(\cdot; \psi_T(\mathbf{x})\big), \qquad P^{\theta}_C(\cdot|\mathbf{x}) = \mathcal{F}_C\big(\cdot; \psi_C(\mathbf{x})\big), \tag{5}$$

where each *risk surface* ψ maps covariates to parameters of a conditional family $\mathcal{F}$ with closed-form CDF. Thus, sampling $\theta \sim \pi$ amounts to sampling *surface–family* pairs, making the targets p_T and

p_C in (2) analytically available. We then design π to span three complementary axes: *interaction*, the functional complexity of ψ; *discrimination*, the strength of covariate-dependent variation in ψ; and *breadth*, the range of shapes induced by $\mathcal{F}$. The surface generators vary ψ: `Tree` introduces axis-aligned interactions, `DeepMLP` smooth nonlinear structure, and `Narrow` high-discrimination risk. The family generators instead vary $\mathcal{F}$: `Parametric` covers Gaussian, LogNormal, Weibull, and Exponential distributions, while `Bernstein` provides flexible distributional shapes. Because no single family spans all three axes, the mixture combines their complementary supports.

Censoring Settings. The independently drawn *event-* and *censoring-time conditional distributions* in (5) can yield nearly all-observed or all-censored cohorts. For each DGP, we therefore draw a target censoring rate $\rho_\theta \sim \mathcal{U}(\rho_{\min}, \rho_{\max})$ and calibrate a multiplicative censoring-time scale on a reference sample. This preserves $T \perp C \mid X$ while making the *prior* span light to heavy censoring.

Compared with SurvivalPFN (Qi et al., 2026), our *prior* plays a different role and has broader structural support. SurvivalPFN samples outcomes from an analytically specified generator, whereas TTE-PFN distills the corresponding conditional distribution directly. More importantly, our mixture extends smooth MLP risk surfaces with complementary interaction structures and conditional families. Section 4 ablates the loss and *prior* separately, with additional results in Appendix D.2.

3.3 Excess-risk Decomposition

We now bound how far the trained network deviates from the truth on a new cohort generated by an unknown DGP $\theta^\star$, where $\theta^\star$ may lie outside $\mathrm{supp}(\pi)$, and decompose the gap along the design choices of Sections 3.1 and 3.2. For any $\theta \in \mathrm{supp}(\pi) \cup \{\theta^\star\}$, let p_θ be the *binned conditional event-time distribution* defined in (2), and measure error by total variation, $\mathrm{TV}(p, q) = \frac{1}{2}\sum_{\ell=1}^{L} |[p]_\ell - [q]_\ell|$ (Gibbs & Su, 2002). For fixed *context size* n, let $\mathbb{M}_n$ be the *prior*-predictive law of $Z = (\mathcal{D}_{\mathrm{ctx}}, \mathbf{x}^*)$, and let $\mathbb{Q}_n^\star = (P_{\mathrm{obs}}^{\theta^\star})^{\otimes n} \otimes P_X^{\theta^\star}$ be the corresponding test-task law. We require four conditions.

Assumption 1 (**A1**, Identifiability)**.** *For an evaluation horizon $[0, t_{\max}]$, every $\theta \in \mathrm{supp}(\pi) \cup \{\theta^\star\}$ satisfies conditional independence, $T \perp C \mid X$, and censoring positivity $P^\theta(C \geq t|X = \mathbf{x}) > 0$ $\forall t \in [0, t_{\max}]$ and P_X^θ-almost every $\mathbf{x}$. Moreover, for any $\theta_1, \theta_2 \in \mathrm{supp}(\pi) \cup \{\theta^\star\}$, $P_{\mathrm{obs}}^{\theta_1} = P_{\mathrm{obs}}^{\theta_2} \Rightarrow$ $F_T^{\theta_1}(t|\mathbf{x}) = F_T^{\theta_2}(t|\mathbf{x}), \forall\, t \in [0, t_{\max}]$, for almost every $\mathbf{x}$ under their common covariate marginal.*

Assumption 2 (**A2**, Lipschitz Conditional Event Law)**.** *There exists a constant $\mathcal{L}_{\mathrm{lip}} < \infty$ such that, $\forall \theta, \theta' \in \mathrm{supp}(\pi) \cup \{\theta^\star\}$, $\mathrm{TV}\Big(P_T^\theta(\cdot|\mathbf{x}), P_T^{\theta'}(\cdot|\mathbf{x})\Big) \leq \mathcal{L}_{\mathrm{lip}}\, \mathrm{dist}_{\mathrm{obs}}(\theta, \theta')$, for $P_X^{\theta^\star}$-almost every $\mathbf{x}$, where $\mathrm{dist}_{\mathrm{obs}}$ denotes the Hellinger distance between the corresponding observable laws.*

Assumption 3 (**A3**, Posterior Concentration)**.** *There exists a pseudo-true DGP $\theta^\dagger \in \mathrm{supp}(\pi)$ whose observable law minimizes the Kullback–Leibler divergence from that of the evaluation DGP: $\theta^\dagger \in \arg\min_{\theta \in \mathrm{supp}(\pi)} D_{\mathrm{KL}}\big(P_{\mathrm{obs}}^{\theta^\star} \,\big\|\, P_{\mathrm{obs}}^{\theta}\big)$. Moreover, there exist a constant $C_{\mathrm{post}} < \infty$ and a sequence $\varepsilon_n \to 0$ such that $\mathbb{E}_{(\mathcal{D}_{\mathrm{ctx}}, \mathbf{x}^*) \sim \mathbb{Q}_n^\star}\left[\int_\Theta \mathrm{dist}_{\mathrm{obs}}(\theta, \theta^\dagger)\, \Pi(\mathrm{d}\theta \mid \mathcal{D}_{\mathrm{ctx}}, \mathbf{x}^*)\right] \leq C_{\mathrm{post}} \varepsilon_n$.*

Assumption 4 (**A4**, Train–Test Task Transfer)**.** *The test-task law is absolutely continuous with respect to the fixed-n prior-predictive law and has an essentially bounded density ratio: $\mathbb{Q}_n^\star \ll \mathbb{M}_n$, $\left\| \frac{\mathrm{d}\mathbb{Q}_n^\star}{\mathrm{d}\mathbb{M}_n} \right\|_\infty \leq B_n < \infty$.*

These assumptions impose standard identification and regularity conditions together with a train–test overlap condition. Intuitively, **A1** makes the event-time law identifiable from *right-censored* observations, while **A2** imposes a continuity condition that converts *prior coverage* in observable-law space into predictive coverage. **A3** is a standard posterior-concentration condition and allows the evaluation DGP to be misspecified relative to the training prior. **A4** requires the evaluation-task law to remain covered by the prior-predictive training law, but does not require $\theta^\star \in \mathrm{supp}(\pi)$. The degree of train–test mismatch is explicitly reflected by B_n, so weaker overlap yields a correspondingly looser bound. These conditions connect, respectively, to the censoring construction and broad *prior* of Section 3.2 and the in-context training of Section 3.1.

Under these conditions, the predictive error decomposes along $q_T^\omega \to q_{\mathrm{PPD},T} \to p_{\theta^\dagger,T} \to p_{\theta^\star,T}$, corresponding respectively to *amortization*, *posterior concentration*, and *prior coverage*. Now, for a fixed *context size* n, let $\mathcal{L}_{\mathrm{OD},n}(\omega)$ denote the fixed-n counterpart of the *oracle-distillation* objective in (4), so that $\mathcal{L}_{\mathrm{OD}}(\omega) = \mathbb{E}_{N \sim P_N}\left[\mathcal{L}_{\mathrm{OD},N}(\omega)\right]$. By **Lemma 1**, the Bayes-risk floor of the fixed-n objective is $\mathcal{L}_{\mathrm{OD},n}^\star := \mathbb{E}_{\mathbb{M}_n}\left[\mathbb{H}(q_{\mathrm{PPD},T}) + \lambda_c \mathbb{H}(q_{\mathrm{PPD},C})\right]$, where $q_{\mathrm{PPD},T}$ and $q_{\mathrm{PPD},C}$ denote the *binned posterior predictive distributions* for the event and censoring channels, respectively.

Term (I): Amortization. **Lemma 1** and the *cross-entropy* decomposition imply $\mathcal{L}_{\mathrm{OD},n}(\omega) - \mathcal{L}^{\star}_{\mathrm{OD},n} \geq \mathbb{E}_{\mathbb{M}_n}\left[D_{\mathrm{KL}}(q_{\mathrm{PPD},T} \,\|\, q^{\omega}_T)\right]$. Applying Pinsker's inequality (Tsybakov, 2008) and Jensen's inequality (Villani et al., 2009), followed by **A4**, gives

$$\mathbb{E}_{\mathbb{Q}^{\star}_n}\left[\mathrm{TV}(q^{\omega}_T, q_{\mathrm{PPD},T})\right] \leq \sqrt{\frac{B_n}{2}\left(\mathcal{L}_{\mathrm{OD},n}(\omega) - \mathcal{L}^{\star}_{\mathrm{OD},n}\right)}. \tag{6}$$

This term captures the approximation and optimization gap between the *amortized predictor* and the *posterior predictive*, together with the train–test transfer factor B_n. Recalling **Lemma 2**, *oracle-distillation* objective removes the additional gradient variance caused by *sampling labels*, thereby eliminating one source of optimization noise that can hinder the reduction of this term.

Terms (II) and (III): Prior Coverage and Posterior Concentration. Inserting $p_{\theta^{\dagger},T}$ between the *posterior predictive* $q_{\mathrm{PPD},T}$ and the *truth* $p_{\theta^{\star},T}$, and applying the triangle inequality, **A2** and **A3**, and contraction of total variation under discretization (Gibbs & Su, 2002), yields

$$\mathbb{E}_{\mathbb{Q}^{\star}_n}\left[\mathrm{TV}(q_{\mathrm{PPD},T}, p_{\theta^{\star},T})\right] \leq \mathcal{L}_{\mathrm{lip}}\,\mathrm{dist}_{\mathrm{obs}}(\theta^{\star}, \theta^{\dagger}) + \mathcal{L}_{\mathrm{lip}}\,C_{\mathrm{post}}\varepsilon_n. \tag{7}$$

Term (II) measures the mismatch between the evaluation DGP $\theta^{\star}$ and its pseudo-true $\theta^{\dagger}$ approximation in the *prior* support, whereas **Term (III)** measures the residual *posterior* uncertainty due to a finite *context*. Finally, combining (6) and (7) gives the main result.

Theorem 1 (Excess-Risk Decomposition)**.** *Under* **A1**–**A4***, for any network weights ω and fixed context size n,*

$$\mathbb{E}_{\mathbb{Q}^{\star}_n}\left[\mathrm{TV}(q^{\omega}_T, p_{\theta^{\star},T})\right] \leq \underbrace{\sqrt{\frac{B_n}{2}\left(\mathcal{L}_{\mathrm{OD},n}(\omega) - \mathcal{L}^{\star}_{\mathrm{OD},n}\right)}}_{\text{(I) amortization}} + \underbrace{\mathcal{L}_{\mathrm{lip}}\,\mathrm{dist}_{\mathrm{obs}}(\theta^{\star}, \theta^{\dagger})}_{\text{(II) prior coverage}} + \underbrace{\mathcal{L}_{\mathrm{lip}}\,C_{\mathrm{post}}\varepsilon_n}_{\text{(III) posterior concentration}}.$$

Theorem 1 separates the main sources of predictive error, thereby clarifying the role of each design choice in TTE-PFN. **Term (I)** captures architecture and optimization, reduced by the *two-way attention* backbone and the removal of *label-sampling* gradient noise (Section 3.1), **Term (II)** reflects *prior* coverage (Section 3.2), and **Term (III)** is the finite-*context* penalty, which vanishes as cohorts grow. Consequently, the fixed-n result holds for every *context size* n in the support of P_N. The proof of **Theorem 1**, together with its extension to mixed-context-size training, is in Appendix A.5.

4 Experiments

Datasets. We evaluate TTE-PFN on a diverse benchmark of 115 datasets, *none* of which is observed during training or used for gradient-based adaptation. The benchmark contains 101 *real-world cohorts* and 14 *synthetic datasets*, spanning *clinical*, *genomic*, *reliability*, *economic*, and *social-science* applications. The *real-world* portion combines all 77 datasets from SurvSet (Drysdale, 2022) with 24 additional datasets collected from recent studies, textbooks, and public software packages. Together, these cohorts cover a broad range of statistical regimes, with *sample sizes* from 26 to 52,422, encoded *covariate dimensions* from 3 to 15,541, and *censoring proportions* from 0 to 0.94. They include both *large-sample*, *low-dimensional clinical cohorts* and *high-dimensional genomic datasets*, as well as cohorts with *light*, *moderate*, and *heavy censoring*. The remaining 14 datasets are drawn from the CQRNN benchmark (Pearce et al., 2022) and provide controlled Gaussian, LogNormal, Weibull, and Exponential event-time distributions under different censoring mechanisms and levels.

Following standard practice, we evaluate each method over 5 random $70/30$ train–test splits. For zero-shot methods that perform no dataset-specific fitting—TTE-PFN, SurvivalPFN, and the *tabular foundation models*—the 70% partition is provided as the *context set* and predictions are evaluated on the remaining 30% *query set*. For all other baselines, the same partitions serve as training and test sets. Dataset details and code are in Appendix C.1 and the Supplementary Material, respectively.

Metrics. We evaluate each model using three categories of metrics: *Predictive Accuracy*, *Concordance*, and *Calibration*. For *predictive accuracy*, we report the Mean Absolute Error based on Pseudo-Observations (MAE-PO, Qi et al. 2023a) and the Integrated Brier Score (IBS, Graf et al. 1999), which quantify how accurately the model predicts survival times over the observation horizon. For *concordance*, we use Harrell's C-index (Harrell et al., 1982), Uno's C-index (Uno et al., 2011), and Log-Rank (Qi et al., 2023b) to assess the model's ability to correctly rank survival times while accounting for censored observations. For *calibration*, we measure the reliability of the predicted survival probabilities using D-Calibration (D-Cal, Haider et al. 2020) and Kaplan-Meier Calibration (KM-Cal, Chapfuwa et al. 2020). Details are provided in Appendix C.2.

Baselines. We compare against 28 baselines spanning five families: *Tabular foundation models*: SurvivalPFN (Qi et al., 2026), TabPFN (Hollmann et al., 2025), StaticSurvivalTFM (Kim et al., 2026), TabICL (Qu et al., 2026), MITRA (Zhang et al., 2026) and TabDPT (Ma et al., 2026); *Classical survival models:* CoxPH (Cox, 1972), CoxNet (Simon et al., 2011) and cSVR (Pölsterl et al., 2015); *Tree-based models:* GB (Friedman, 2001), CWGB (Pölsterl, 2020) and RSF (Ishwaran et al., 2008); *Neural discrete-time models:* DeepHit (Lee et al., 2018), DeepSurv (Katzman et al., 2018), MTLR (Yu et al., 2011), Nnet-survival (Gensheimer & Narasimhan, 2019), CoxTime (Kvamme et al., 2019), CQRNN (Pearce et al., 2022) and BNN-MTLR (Qi et al., 2023a); *Neural continuous-time models:* DSM-Weibull/LogNormal (Nagpal et al., 2021), SuMo-net (Rindt et al., 2022), SurvivalMDN (Han et al., 2022), DeepAFT-Weibull/Loglogistic (Norman et al., 2024), ALD (Sheng & Henao, 2025), ICALD (Sheng & Henao, 2026) and LogNorm (Hoseini et al., 2017). Together, these baselines span diverse modeling strategies, providing a comprehensive benchmark for evaluation. Details for each baseline are provided in Appendix C.3 and *Related Work* in Appendix B.

Overall Performance. Across the benchmark, TTE-PFN achieved the best aggregate performance among all 29 evaluated methods, with an *Overall* average rank of 8.09 (Figure 2 and Table 1). Its advantage was particularly pronounced in predictive *accuracy*: TTE-PFN ranked first on both MAE-PO and IBS, with mean ranks of 5.09 and 4.78, respectively. These results indicate that TTE-PFN improves the estimation of the survival distribution rather than optimizing only a single aspect of predictive performance.

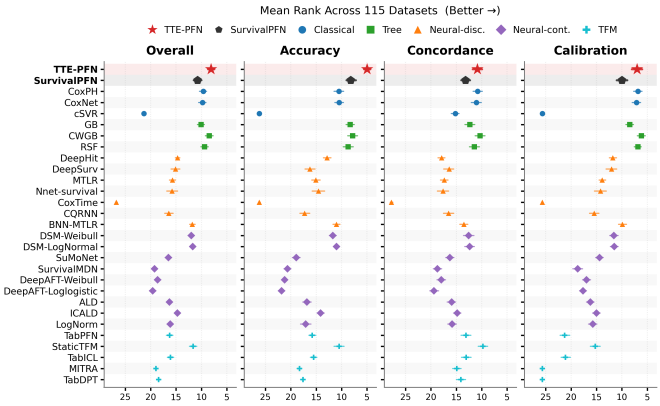


Figure 2: Aggregate benchmark ranks across 115 survival datasets (also see Table 5). Lower ranks indicate better performance. *Overall* gives equal weight to each of the seven evaluation metrics, while *Accuracy*, *Concordance*, and *Calibration* average the metric ranks within their respective categories. Points denote mean dataset-level ranks, and horizontal bars denote 95% bootstrap confidence intervals.

TTE-PFN also maintained strong performance across *concordance* and *calibration* metrics. Although specialized baselines ranked best on individual metrics, for example, StaticTFM on Harrell's and Uno's C-indices, RSF on D-Cal and Log-Rank, and CWGB on KM-Cal, their performance was less consistent across categories. In contrast, TTE-PFN remained near the leading methods throughout, yielding the strongest overall balance among predictive *accuracy*, *concordance*, and *calibration*.

Table 1: Average ranks across 115 datasets (lower is better). The best rank among the displayed methods in each column is **highlighted**. Complete results for all 29 methods are reported in Table 6.

Method	Overall	MAE-PO	IBS	Harrell's CI	Uno's CI	Log-Rank	D-Cal	KM-Cal
TTE-PFN	**8.09**	**5.09**	**4.78**	9.70	10.96	11.93	7.52	6.50
CWGB	8.47	8.36	7.46	10.17	9.59	11.25	6.23	**6.09**
RSF	9.37	10.53	6.79	12.04	11.38	**10.94**	**5.97**	7.70
CoxPH	9.62	13.68	7.30	9.06	8.80	14.53	6.59	7.11
CoxNet	9.79	13.30	7.63	9.33	9.10	14.67	7.13	7.13
GB	10.08	8.08	8.50	11.69	12.13	13.20	8.23	8.62
SurvivalPFN	10.76	7.27	9.09	11.85	11.22	16.45	8.72	11.18
StaticTFM	11.62	8.47	12.52	**8.03**	**8.31**	12.95	12.85	17.59
DSM-LogNormal	11.73	11.04	11.04	12.43	12.93	11.77	9.54	13.46
BNN-MTLR	11.81	11.75	10.36	12.23	11.70	16.63	9.11	10.65

The improvement over SurvivalPFN was consistent across all seven metrics. TTE-PFN reduced the Overall mean rank from 10.76 to 8.09, with especially large gains on IBS (4.78 ***v.s.*** 9.09), Log-Rank (11.93 ***v.s.*** 16.45) and KM-Cal (6.50 ***v.s.*** 11.18). Thus, the improvement is not driven by a single metric or evaluation category, but reflects a broadly stronger survival-distribution estimator.

TTE-PFN *vs.* SurvivalPFN. Having established the overall advantage of TTE-PFN, we next examine which design choices account for the improvement over SurvivalPFN. The two methods differ along three axes: (A) *oracle-distillation* supervision, (B) *context-size* randomization during pretraining, and (C) the proposed diverse *prior*. We evaluate the complete 2^3 factorial design while holding

the backbone, optimization budget, discretization, and random seeds fixed. Table 2 shows that all three components contribute to TTE-PFN's aggregate performance. Among single-component ablations, the Bernstein-only *prior* causes the largest degradation, increasing the *Overall* mean rank by 1.90, compared with 1.01 for fixed-*context-size* pretraining and 0.31 for *sampled-label* supervision. This suggests the ordering diverse *prior* $>$ *context-size* randomization $>$ *oracle-distillation* supervision in aggregate contribution. Removing all three components recovers SurvivalPFN, which attains the worst *Overall* rank of 5.60, a degradation of 2.43 relative to TTE-PFN. Given this, Figure 8 further decomposes the contribution of the diverse *prior* and shows that the five generator families provide complementary coverage, rather than the improvement being driven by any single family.

Table 2: Factorial ablation of TTE-PFN across 115 datasets (lower is better): (A) *Oracle-distillation* loss, (B) *context-size* randomization during pretraining, and (C) the proposed diverse *prior*. The best rank in each column is **highlighted** and Δ is the increase in *Overall* rank relative to TTE-PFN.

Settings	A	B	C	Overall	Δ	MAE-PO	IBS	Harrell's CI	Uno's CI	Log-Rank	D-Cal	KM-Cal
TTE-PFN	✓	✓	✓	**3.17**	–	3.21	**2.25**	2.90	3.16	5.66	**3.04**	**2.69**
w/o A	✗	✓	✓	3.48	+0.31	**2.15**	4.14	**2.68**	**3.05**	3.56	4.28	4.12
w/o A, B	✗	✗	✓	4.15	+0.98	3.14	4.22	3.48	3.64	4.98	4.93	4.55
w/o B	✓	✗	✓	4.18	+1.01	4.26	3.38	3.52	3.73	5.82	4.58	4.14
w/o B, C	✓	✗	✗	4.91	+1.74	6.07	4.25	5.91	5.71	4.70	3.98	4.26
w/o C	✓	✓	✗	5.07	+1.90	6.57	4.49	5.98	5.72	4.64	3.93	4.56
w/o A, C	✗	✓	✗	5.31	+2.14	4.74	6.74	5.59	5.31	**2.84**	5.54	5.65
SurvivalPFN	✗	✗	✗	5.60	+2.43	5.65	6.29	5.94	5.69	3.59	5.72	5.80

These results align with the design guidance provided by **Theorem 1** and **Corollary 1**. The diverse *prior* is designed to reduce the *prior*-coverage **term (II)** by placing synthetic DGPs closer to real cohorts in observable-law space. *Context-size* randomization enforces the *posterior*-predictive target across every *context size* in the support of P_N, rather than constraining the model only at a single training size. Finally, *oracle distillation* facilitates the reduction of the *amortization* **term (I)** by removing an avoidable source of stochastic-gradient noise. Also, the smaller aggregate gain from *oracle-distillation* supervision is expected from our theoretical analysis in Section 3. By **Lemma 1**, the matched *oracle-distillation* and *sampled-label* objectives (*i.e.*, $\mathcal{L}_{\mathrm{OD}}$ and $\mathcal{L}_{\mathrm{SL}}^{\mathrm{match}}$) share the same population minimizer. $\mathcal{L}_{\mathrm{OD}}$ therefore does not change the asymptotic prediction target. Instead, **Lemma 2** shows that $\mathcal{L}_{\mathrm{OD}}$ removes the additional *label-sampling* covariance from the stochastic gradient. Its primary advantage should consequently appear in optimization efficiency and stability rather than as a large shift in the final population optimum. Figure 3 further confirms this optimization advantage. $\mathcal{L}_{\mathrm{OD}}$ and $\mathcal{L}_{\mathrm{SL}}^{\mathrm{match}}$ converge to similar held-out *cross-entropy* values, 3.371 and 3.416, respectively. However, $\mathcal{L}_{\mathrm{OD}}$ follows a faster and smoother optimization trajectory, satisfying the convergence criterion at epoch 809 rather than epoch 976.

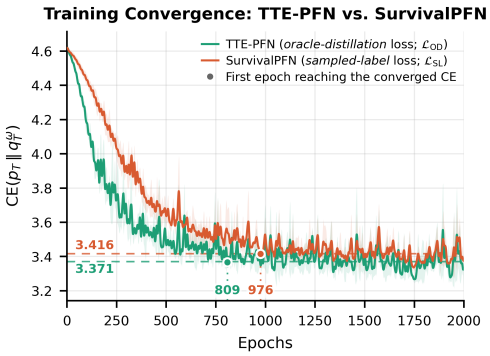


Figure 3: Held-out *cross-entropy* against the true *conditional event-time distribution* during training. Dashed lines show the average loss over the final 500 epochs, and markers indicate the first epoch satisfying the convergence criterion.

Robustness Across Context Sizes. Figure 4 shows that TTE-PFN achieves the best mean *overall* rank at every evaluated context fraction and remains relatively stable as the available *context* varies. Its advantage is already evident in the smallest-*context* regime and persists as more labeled observations become available. In contrast, SurvivalPFN consistently ranks behind TTE-PFN and does not reliably benefit from additional *context*. Among the non-PFN methods, CWGB is the strongest competitor overall, whereas CoxPH becomes progressively more competitive with more training observations but never overtakes TTE-PFN. These results align with the two *context-size* implications of our theoretical analysis. The weaker performance

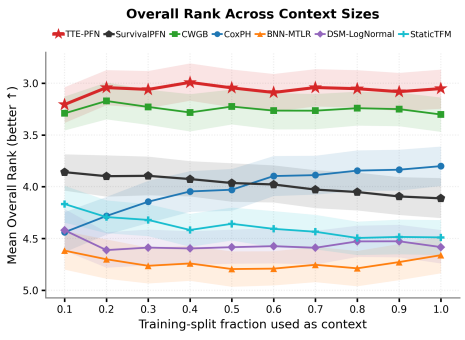


Figure 4: Mean *overall* rank as the context varies from 10% to 100% of the training split. Lower is better, and shaded regions denote 95% CIs.

at small context sizes is consistent with **term (III)** in **Theorem 1**: with fewer labeled observations, the *posterior* is less concentrated around the closest *prior*-supported DGP, leaving greater predictive uncertainty. **Corollary 1**, in turn, explains why this degradation remains mild. By randomizing *context size* during pretraining, TTE-PFN learns the corresponding *posterior predictive* across the support of P_N, rather than only at a single fixed training size. TTE-PFN therefore benefits from larger *contexts* while remaining robust across the entire evaluated range. In the end, a detailed case study in Appendix D.2 further reports the complete metric-wise rank profiles in Figure 10 and a dataset-level analysis on `epileptic` in Figure 11.

Computational Efficiency. We report the benchmark-wide average computational cost per individual, scaled to $1{,}000$ held-out individuals, under a common allocation of one NVIDIA H200 GPU and one CPU. For pretrained models, the reported cost consists only of a forward pass, whereas for fitted baselines it includes preprocessing, model fitting, and prediction. As shown in Figure 5, TTE-PFN achieves the best mean *overall* rank while also attaining the lowest normalized computational cost under our evaluation protocol. It requires only 0.052 seconds per $1{,}000$ held-out individuals on average, approximately $4.5\times$ faster than SurvivalPFN, while improving the mean rank from 10.76 to 8.09. CoxPH and CoxNet also incur relatively low computational costs but have substantially weaker aggregate performance, whereas most neural survival and tabular foundation models are considerably more expensive. TTE-PFN therefore combines strong predictive performance with efficient single-pass inference. Complete computational-cost results and evaluation details are provided in Appendix D.2.

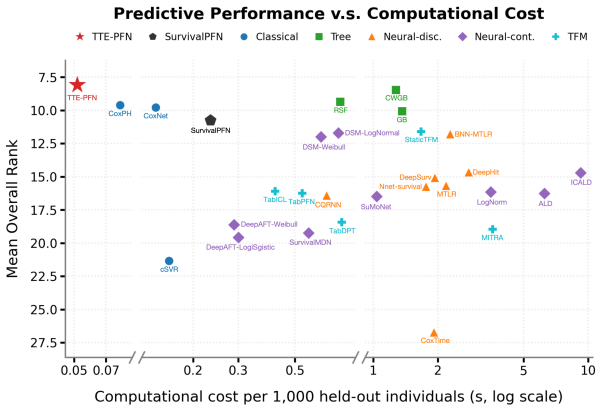


Figure 5: Predictive performance *v.s.* computational cost across the benchmark. The axes report computational cost normalized to $1{,}000$ individuals and mean *overall* rank, with lower values better on both.

5 Conclusions

We introduced TTE-PFN, an amortized in-context estimator that produces predictive *event-* and *censoring-time distributions* from a labeled *right-censored context* and unlabeled *query* covariates in a single forward pass, without dataset-specific fitting. TTE-PFN combines *oracle distillation*, randomized *context sizes*, and a diverse *prior* over survival data-generating processes. Our analysis identifies the *posterior predictive* as the population target, shows that *oracle distillation* removes the additional gradient covariance of matched *sampled-label* supervision, establishes optimality across all represented *context sizes*, and decomposes predictive error into *amortization*, *prior coverage*, and *posterior concentration*. Across 115 datasets, 28 competing methods, and seven metrics, TTE-PFN achieves the best *overall* rank and ranks first on both MAE-PO and IBS, while attaining the lowest normalized computational cost under our evaluation protocol. Factorial ablations further confirm that all three components contribute, with the diverse *prior* providing the largest aggregate gain.

Limitations. TTE-PFN assumes *conditionally independent right censoring*, as is standard in survival analysis. Under informative censoring or insufficient censoring support over the evaluation horizon, the *event-time distribution* may not be identifiable without additional assumptions on the censoring mechanism. This is a general limitation of survival analysis rather than one specific to TTE-PFN (Overgaard & Hansen, 2021). TTE-PFN also inherits the fixed-input limitations of *two-way attention* and may exhibit cohort-level *miscalibration* (Hollmann et al., 2025). Performance may degrade on *high-dimensional* datasets whose feature dimension substantially exceeds that seen during pretraining, such as genomic datasets with up to $15{,}541$ covariates. These limitations can be partially mitigated using the *context* set: *context*-based feature selection or projection (Jolliffe & Cadima, 2016) can reduce input dimensionality, while *post-hoc calibration* (Kuleshov et al., 2018) can adapt the predicted survival distribution to the target cohort. Nevertheless, more efficient and principled approaches to *high-dimensional* inference and cohort-specific calibration remain important directions for future work. Detailed analyses are provided in Appendix D.2.

AI USE STATEMENT

We used generative AI tools to assist with language editing and software development. For language editing, these tools were used to improve clarity and fluency and to identify grammatical errors and typographical mistakes. For software development, they were used to assist with code drafting, debugging, refactoring, and code organization.

All AI-assisted text and code were carefully reviewed, revised, and, where applicable, tested by the authors. The research questions, methodology, experimental design, analysis, and scientific conclusions were developed and validated by the authors. We take full responsibility for the final content of this work, including all text, code, claims, results, and other research artifacts.

REFERENCES

Samuel Böhm, Lennart Purucker, Frank Hutter, and Pascal Schlosser. Survpfn: Towards foundation models for survival predictions. *arXiv preprint arXiv:2606.04564*, 2026.

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.

Imre Csiszár. On information-type measure of difference of probability distributions and indirect observations. *Studia Sci. Math. Hungar.*, 2:299–318, 1967.

Christina Curtis, Sohrab P Shah, Suet-Feung Chin, Gulisa Turashvili, Oscar M Rueda, Mark J Dunning, Doug Speed, Andy G Lynch, Shamith Samarajiwa, Yinyin Yuan, et al. The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups. *Nature*, 486(7403): 346–352, 2012.

Erik Drysdale. SurvSet: An open-source time-to-event dataset repository. *arXiv preprint arXiv:2203.03094*, 2022.

J Emmerson and JM Brown. Understanding survival analysis in clinical trials. *Clinical Oncology*, 33(1):12–14, 2021.

Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pp. 1189–1232, 2001.

Michael F Gensheimer and Balasubramanian Narasimhan. A scalable discrete-time survival model for neural networks. *PeerJ*, 7:e6257, 2019.

Alison L Gibbs and Francis Edward Su. On choosing and bounding probability metrics. *International statistical review*, 70(3):419–435, 2002.

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378, 2007.

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.

Humza Haider, Bret Hoehn, Sarah Davis, and Russell Greiner. Effective ways to build and evaluate individual survival distributions. *Journal of Machine Learning Research*, 21(85):1–63, 2020.

Xintian Han, Mark Goldstein, and Rajesh Ranganath. Survival mixture density networks. In *Machine Learning for Healthcare Conference*, pp. 224–248. PMLR, 2022.

Frank E Harrell, Robert M Califf, David B Pryor, Kerry L Lee, and Robert A Rosati. Evaluating the yield of medical tests. *Jama*, 247(18):2543–2546, 1982.

Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637(8045):319–326, 2025.

Mina Hoseini, Abbas Bahrampour, and Moghaddameh Mirzaee. Comparison of weibull and lognormal cure models with cox in the survival analysis of breast cancer patients in rafsanjan. *Journal of research in health sciences*, 17(1):369, 2017.

Hemant Ishwaran, Udaya B Kogalur, Eugene H Blackstone, and Michael S Lauer. Random survival forests. *The Annals of applied statistics*, 2(3):841–860, 2008.

Ian T Jolliffe and Jorge Cadima. Principal component analysis: a review and recent developments. *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences*, 374(2065):20150202, 2016.

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*, 18:1–12, 2018.

Da In Kim, Wei Siang Lai, and Kelly W Zhang. Tabular foundation models can do survival analysis. *arXiv preprint arXiv:2601.22259*, 2026.

John P Klein and Melvin L Moeschberger. *Survival analysis: techniques for censored and truncated data*. Springer Science & Business Media, 2006.

Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. Accurate uncertainties for deep learning using calibrated regression. In *International conference on machine learning*, pp. 2796–2804. PMLR, 2018.

Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

Håvard Kvamme, Ørnulf Borgan, and Ida Scheel. Time-to-event prediction with neural networks and cox regression. *Journal of machine learning research*, 20(129):1–30, 2019.

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, 2018.

Junwei Ma, Valentin Thomas, Rasa Hosseinzadeh, Alex Labach, Jesse Cresswell, Keyvan Golestan, Guangwei Yu, Anthony L Caterini, and Maks Volkovs. Tabdpt: Scaling tabular foundation models on real data. *Advances in Neural Information Processing Systems*, 38:172692–172722, 2026.

Samuel Müller, Noah Hollmann, Sebastian Pineda Arango, Josif Grabocka, and Frank Hutter. Transformers can do bayesian inference. In *International Conference on Learning Representations*, 2022.

Chirag Nagpal, Xinyu Li, and Artur Dubrawski. Deep survival machines: Fully parametric survival regression and representation learning for censored data with competing risks. *IEEE Journal of Biomedical and Health Informatics*, 25(8):3163–3175, 2021.

Patrick A Norman, Wanlu Li, Wenyu Jiang, and Bingshu E Chen. deepaft: A nonlinear accelerated failure time model with artificial neural network. *Statistics in Medicine*, 43(19):3689–3701, 2024.

Morten Overgaard and Stefan Nygaard Hansen. On the assumption of independent right censoring. *Scandinavian Journal of Statistics*, 48(4):1234–1255, 2021.

Tim Pearce, Jong-Hyeon Jeong, Jun Zhu, et al. Censored quantile regression neural networks for distribution-free survival analysis. *Advances in Neural Information Processing Systems*, 35:7450–7461, 2022.

Sebastian Pölsterl. scikit-survival: A library for time-to-event analysis built on top of scikit-learn. *Journal of Machine Learning Research*, 21(212):1–6, 2020.

Sebastian Pölsterl, Nassir Navab, and Amin Katouzian. Fast training of support vector machines for survival analysis. In *Joint European conference on machine learning and knowledge discovery in databases*, pp. 243–259. Springer, 2015.

Shi-Ang Qi, Neeraj Kumar, Mahtab Farrokh, Weijie Sun, Li-Hao Kuan, Rajesh Ranganath, Ricardo Henao, and Russell Greiner. An effective meaningful way to evaluate survival models. In *International Conference on Machine Learning*, pp. 28244–28276. PMLR, 2023a.

Shi-ang Qi, Weijie Sun, and Russell Greiner. Survivaleval: A comprehensive open-source python package for evaluating individual survival distributions. In *Proceedings of the AAAI Symposium Series*, volume 2, pp. 453–457, 2023b.

Shi-ang Qi, Vahid Balazadeh, Michael Cooper, Russell Greiner, and Rahul G Krishnan. Survivalpfn: Amortizing survival prediction via in-context bayesian inference. *arXiv preprint arXiv:2605.15488*, 2026.

Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICLv2: A better, faster, scalable, and open tabular foundation model. *arXiv preprint arXiv:2602.11139*, 2026.

David Rindt, Robert Hu, David Steinsaltz, and Dino Sejdinovic. Survival regression with proper scoring rules and monotonic neural networks. In *International conference on artificial intelligence and statistics*, pp. 1190–1205. PMLR, 2022.

Dmitrii Seletkov, Paul Hager, Rickmer Braren, Daniel Rueckert, and Raphael Rehms. Survival in-context: Prior-fitted in-context learning tabular foundation model for survival analysis. *arXiv e-prints*, pp. arXiv–2603, 2026.

Deming Sheng and Ricardo Henao. Learning survival distributions with the asymmetric laplace distribution. In *International Conference on Machine Learning*, pp. 54772–54809. PMLR, 2025.

Deming Sheng and Ricardo Henao. Learning survival distributions with individually calibrated asymmetric laplace distribution. In *The Fourteenth International Conference on Learning Representations*, 2026.

Noah Simon, Jerome H Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for cox's proportional hazards model via coordinate descent. *Journal of statistical software*, 39:1–13, 2011.

Alexandre B Tsybakov. Nonparametric estimators. In *Introduction to Nonparametric Estimation*, pp. 1–76. Springer, 2008.

Hajime Uno, Tianxi Cai, Michael J Pencina, Ralph B D'Agostino, and Lee-Jen Wei. On the c-statistics for evaluating overall adequacy of risk prediction procedures with censored survival data. *Statistics in medicine*, 30(10):1105–1117, 2011.

Cédric Villani et al. *Optimal transport: old and new*, volume 338. Springer, 2009.

Frank Wilcoxon. Individual comparisons by ranking methods. *Biometrics bulletin*, 1(6):80–83, 1945.

Chun-Nam Yu, Russell Greiner, Hsiu-Chin Lin, and Vickie Baracos. Learning patient-specific cancer survival distributions as a sequence of dependent regressors. In *Proceedings of the 25th International Conference on Neural Information Processing Systems*, pp. 1845–1853, 2011.

Xiyuan Zhang, Danielle Maddix Robinson, Junming Yin, Nick Erickson, Abdul Fatir Ansari, Boran Han, Shuai Zhang, Leman Akoglu, Christos Faloutsos, Michael Mahoney, et al. Mitra: Mixed synthetic priors for enhancing tabular foundation models. *Advances in neural information processing systems*, 38:15795–15840, 2026.

A Analytical Results

This section provides the detailed arguments supporting the theoretical results stated in Section 3.1 and Section 3.3. Appendix A.1 collects the notation, probability laws, and risk functionals used throughout the appendix. Appendix A.2 proves that the *binned posterior predictive* is the population minimizer of both the *oracle-distillation* objective and its matched *sampled-label* counterpart. Appendix A.3 shows that mixed-size training imposes this Bayes-optimality condition separately at every *context size* in the support of P_N. Appendix A.4 derives the additional gradient covariance introduced by *sampled-label* supervision. Finally, Appendix A.5 proves the excess-risk decomposition of **Theorem 1** and derives its mixed-*context-size* consequence.

A.1 Notation and probability laws

This subsection fixes the notation used in the subsequent proofs and makes explicit the probability laws with respect to which expectations are taken. We first define the probability simplex, the *binned conditional distributions* for the event and censoring channels under each DGP, and their corresponding posterior predictives. We then distinguish fixed-size and mixed-size task laws and define the corresponding channel-specific and two-head risks together with their Bayes-risk floors. These definitions introduce no additional modeling assumptions. They only specialize the quantities from Sections 3.1 and 3.3 to the fixed- and mixed-*context-size* settings used below. Throughout this section, we use the notation introduced in Sections 3.1 and 3.3. Let $Z = (\mathcal{D}_{\mathrm{ctx}}, \mathbf{x}^*)$ denote a *context–query* input, and let $A \in \{T, C\}$ index the event-time or censoring-time channel.

$$\Delta_{L-1} := \left\{ r \in [0,1]^L : \sum_{\ell=1}^{L} [r]_\ell = 1 \right\}. \tag{8}$$

For $A \in \{T, C\}$, let P_A^θ denote P_T^θ when $A = T$ and P_C^θ when $A = C$. For a DGP θ, define

$$[p_{\theta,A}(Z)]_\ell := P_A^\theta \left(\{ u \in \mathbb{R}_+ : \kappa_{\mathcal{D}_{\mathrm{ctx}}}(u) = \ell \} \,|\, \mathbf{x}^* \right), \qquad \ell = 1, \ldots, L. \tag{9}$$

Thus, $p_{\theta,A}(Z) \in \Delta_{L-1}$ is the channel-A binned conditional distribution under DGP θ. Its binned *posterior predictive* is

$$q_{\mathrm{PPD},A}(Z) := \mathbb{E}_{\theta|Z} \left[p_{\theta,A}(Z) \right] = \int_\Theta p_{\theta,A}(Z) \, \Pi(\mathrm{d}\theta|Z). \tag{10}$$

We reserve $N \sim P_N$ for the random *context size* used during pretraining and use n for a fixed realized *context size*. Let $\mathcal{N} := \{n \in \mathbb{N} : P_N(n) > 0\}$ denote the support of the *context-size* distribution. For each fixed $n \in \mathcal{N}$, define the fixed-size *prior*-predictive and test-task laws of Z by

$$\begin{aligned} \mathbb{M}_n(\mathrm{d}Z) &:= \int_\Theta \left[(P_{\mathrm{obs}}^\theta)^{\otimes n} \otimes P_X^\theta \right] (\mathrm{d}Z) \, \pi(\mathrm{d}\theta), \\ \mathbb{Q}_n^\star(\mathrm{d}Z) &:= \left[(P_{\mathrm{obs}}^{\theta^\star})^{\otimes n} \otimes P_X^{\theta^\star} \right] (\mathrm{d}Z). \end{aligned} \tag{11}$$

Equivalently, under $\mathbb{M}_n$, one first draws $\theta \sim \pi$, then samples $\mathcal{D}_{\mathrm{ctx}} \sim (P_{\mathrm{obs}}^\theta)^{\otimes n}$ and $\mathbf{x}^* \sim P_X^\theta$. In contrast, under $\mathbb{Q}_n^\star$, both the *context* and the *query* covariate are generated from the fixed evaluation DGP $\theta^\star$.

Because Z includes $\mathcal{D}_{\mathrm{ctx}}$, its realized *context size* is recoverable from Z. The mixed-size *prior*-predictive task law is therefore the mixture

$$\mathbb{M} := \sum_{n \in \mathcal{N}} P_N(n) \, \mathbb{M}_n. \tag{12}$$

The *posterior* $\Pi(\mathrm{d}\theta|Z)$ in (10) is the conditional law induced by this joint *prior*-predictive construction.

For each channel and fixed size n, define

$$\mathcal{L}_{\mathrm{OD},A,n}(q_A) := \mathbb{E}_{\theta \sim \pi} \mathbb{E}_{\mathcal{D}_{\mathrm{ctx}} \sim (P_{\mathrm{obs}}^\theta)^{\otimes n}} \mathbb{E}_{\mathbf{x}^* \sim P_X^\theta} \left[\mathrm{CE}(p_{\theta,A}(Z) \,\|\, q_A(Z)) \right], \tag{13}$$

where q_A is an arbitrary measurable predictor with values in Δ_{L-1}. When $q_A = q_A^{\omega}$, we write $\mathcal{L}_{\mathrm{OD},A,n}(\omega)$. The corresponding mixed-size channel risk is

$$\mathcal{L}_{\mathrm{OD},A}(q_A) := \mathbb{E}_{N\sim P_N}\left[\mathcal{L}_{\mathrm{OD},A,N}(q_A)\right]. \tag{14}$$

The fixed-size two-head risk is

$$\mathcal{L}_{\mathrm{OD},n}(\omega) := \mathcal{L}_{\mathrm{OD},T,n}(\omega) + \lambda_c \mathcal{L}_{\mathrm{OD},C,n}(\omega). \tag{15}$$

$$\mathcal{L}^{\star}_{\mathrm{OD},n} := \mathbb{E}_{Z\sim\mathbb{M}_n}\left[\mathbb{H}(q_{\mathrm{PPD},T}(Z)) + \lambda_c \mathbb{H}(q_{\mathrm{PPD},C}(Z))\right]. \tag{16}$$

These definitions are the fixed-size counterparts of (3) and (4).

A.2 Proof of the Population Minimizer

*Proof of **Lemma 1**.* Fix a channel $A \in \{T, C\}$. We first prove the claim for the *oracle-distillation* objective. Under the joint pretraining law, conditioning on the task input Z rewrites the population risk as

$$\mathcal{L}_{\mathrm{OD},A}(q_A) = \mathbb{E}_{Z\sim\mathbb{M}}\left[\mathbb{E}_{\theta|Z}\left[\mathrm{CE}(p_{\theta,A}(Z)\,\|\,q_A(Z))\right]\right]. \tag{17}$$

The conditional law of θ in (17) is exactly the *posterior* $\Pi(\mathrm{d}\theta|Z)$ used in (10). Now fix a realization of Z, and abbreviate $q = q_A(Z)$. By the definition of *cross-entropy* and linearity of conditional expectation (Gneiting & Raftery, 2007),

$$\begin{aligned}
&\mathbb{E}_{\theta|Z}\left[\mathrm{CE}(p_{\theta,A}(Z)\,\|\,q)\right] \\
&\quad= \mathbb{E}_{\theta|Z}\left[-\sum_{\ell=1}^{L}[p_{\theta,A}(Z)]_\ell \log[q]_\ell\right] \\
&\quad= -\sum_{\ell=1}^{L}\mathbb{E}_{\theta|Z}\left[[p_{\theta,A}(Z)]_\ell\right]\log[q]_\ell \\
&\quad= -\sum_{\ell=1}^{L}[q_{\mathrm{PPD},A}(Z)]_\ell \log[q]_\ell \\
&\quad= \mathrm{CE}(q_{\mathrm{PPD},A}(Z)\,\|\,q),
\end{aligned} \tag{18}$$

where the third equality follows from the coordinate-wise form of (10). The standard *cross-entropy* decomposition then gives

$$\mathrm{CE}(q_{\mathrm{PPD},A}(Z)\,\|\,q) = \mathbb{H}(q_{\mathrm{PPD},A}(Z)) + D_{\mathrm{KL}}(q_{\mathrm{PPD},A}(Z)\,\|\,q). \tag{19}$$

The entropy term $\mathbb{H}(q_{\mathrm{PPD},A}(Z))$ in (19) does not depend on the candidate prediction q. It therefore remains to minimize the KL-divergence term $D_{\mathrm{KL}}(q_{\mathrm{PPD},A}(Z)\,\|\,q)$. For the softmax predictor considered in TTE-PFN, $[q]_\ell > 0$ for every ℓ. Then, using the elementary inequality $\log u \le u - 1$ for all $u > 0$, we can obtain

$$\begin{aligned}
&D_{\mathrm{KL}}(q_{\mathrm{PPD},A}(Z)\,\|\,q) \\
&\quad= -\sum_{\ell:[q_{\mathrm{PPD},A}(Z)]_\ell>0}[q_{\mathrm{PPD},A}(Z)]_\ell \log\frac{[q]_\ell}{[q_{\mathrm{PPD},A}(Z)]_\ell} \\
&\quad\ge -\sum_{\ell:[q_{\mathrm{PPD},A}(Z)]_\ell>0}[q_{\mathrm{PPD},A}(Z)]_\ell \left(\frac{[q]_\ell}{[q_{\mathrm{PPD},A}(Z)]_\ell} - 1\right) \\
&\quad= 1 - \sum_{\ell:[q_{\mathrm{PPD},A}(Z)]_\ell>0}[q]_\ell \\
&\quad= \sum_{\ell:[q_{\mathrm{PPD},A}(Z)]_\ell=0}[q]_\ell \ge 0.
\end{aligned} \tag{20}$$

For the KL divergence to vanish, both inequalities in (20) must be equalities. Equality in $\log u \le u - 1$ holds only at $u = 1$, which requires

$$[q]_\ell = [q_{\mathrm{PPD},A}(Z)]_\ell \qquad \text{for every } \ell \text{ such that } [q_{\mathrm{PPD},A}(Z)]_\ell > 0. \tag{21}$$

The final equality additionally requires

$$[q]_\ell = 0 \qquad \text{for every } \ell \text{ such that } [q_{\mathrm{PPD},A}(Z)]_\ell = 0. \tag{22}$$

Consequently,

$$D_{\mathrm{KL}}(q_{\mathrm{PPD},A}(Z) \,\|\, q) = 0 \quad \Longleftrightarrow \quad q = q_{\mathrm{PPD},A}(Z). \tag{23}$$

This is Gibbs' inequality, equivalently the non-negativity of the Kullback–Leibler divergence (Kullback & Leibler, 1951). Therefore, among all simplex-valued candidate predictions, the unique minimizer of the conditional risk is $q_{\mathrm{PPD},A}(Z)$. Finally, since the one-hot sampled target is conditionally unbiased for $p_{\theta,A}(Z)$ and *cross-entropy* is linear in its target, averaging over the sampled label recovers exactly the *oracle-distillation* risk. Hence $\mathcal{L}_{\mathrm{SL}}^{\mathrm{match}}$ has the same population minimizer. □

A.3 Proof of Coverage Across Context Sizes

Proof of ***Corollary 1***. Fix a channel $A \in \{T, C\}$. Because the random *context size* N is discrete and has support $\mathcal{N} = \{n \in \mathbb{N} : P_N(n) > 0\}$, the mixed-size population risk can be decomposed as

$$\mathcal{L}_{\mathrm{OD},A}(q_A) = \sum_{n \in \mathcal{N}} P_N(n)\, \mathcal{L}_{\mathrm{OD},A,n}(q_A). \tag{24}$$

Applying (18) and (19) conditional on $N = n$ gives

$$\mathcal{L}_{\mathrm{OD},A,n}(q_A) = \mathbb{E}_{Z\sim\mathbb{M}_n}\left[\mathbb{H}(q_{\mathrm{PPD},A}(Z))\right] + \mathbb{E}_{Z\sim\mathbb{M}_n}\left[D_{\mathrm{KL}}(q_{\mathrm{PPD},A}(Z) \,\|\, q_A(Z))\right]. \tag{25}$$

Define the size-specific Bayes-risk floor by

$$\mathcal{L}^\star_{\mathrm{OD},A,n} := \mathbb{E}_{Z\sim\mathbb{M}_n}\left[\mathbb{H}(q_{\mathrm{PPD},A}(Z))\right]. \tag{26}$$

Subtracting (26) from (25) yields

$$\mathcal{L}_{\mathrm{OD},A,n}(q_A) - \mathcal{L}^\star_{\mathrm{OD},A,n} = \mathbb{E}_{Z\sim\mathbb{M}_n}\left[D_{\mathrm{KL}}(q_{\mathrm{PPD},A}(Z) \,\|\, q_A(Z))\right] \geq 0. \tag{27}$$

Averaging the size-specific identity over $N \sim P_N$ gives

$$\mathcal{L}_{\mathrm{OD},A}(q_A) - \mathcal{L}^\star_{\mathrm{OD},A} = \sum_{n \in \mathcal{N}} P_N(n)\left(\mathcal{L}_{\mathrm{OD},A,n}(q_A) - \mathcal{L}^\star_{\mathrm{OD},A,n}\right), \tag{28}$$

where

$$\mathcal{L}^\star_{\mathrm{OD},A} := \sum_{n \in \mathcal{N}} P_N(n)\, \mathcal{L}^\star_{\mathrm{OD},A,n}. \tag{29}$$

Every summand on the right-hand side of (28) is non-negative by (27). If q_A is a population minimizer of the mixed-size risk, then the left-hand side of (28) equals zero. Hence, for every $n \in \mathcal{N}$,

$$\mathcal{L}_{\mathrm{OD},A,n}(q_A) - \mathcal{L}^\star_{\mathrm{OD},A,n} = 0. \tag{30}$$

Combining (27) and (30) gives

$$\mathbb{E}_{Z\sim\mathbb{M}_n}\left[D_{\mathrm{KL}}(q_{\mathrm{PPD},A}(Z) \,\|\, q_A(Z))\right] = 0. \tag{31}$$

Because the integrand is non-negative, it must vanish $\mathbb{M}_n$-almost surely. Gibbs' inequality (Kullback & Leibler, 1951) therefore implies

$$q_A(Z) = q_{\mathrm{PPD},A}(Z) \qquad \mathbb{M}_n\text{-almost surely} \tag{32}$$

for every $n \in \mathcal{N}$. Since $\lambda_c = 1 > 0$ in the two-head objective of (15), both channel-specific excess risks are non-negative and must vanish separately at any minimizer of the combined risk. To establish the contrast with fixed-size training, suppose that the training law is supported only at $N = n_{\mathrm{fix}}$. Its population risk depends exclusively on predictor values at inputs satisfying $|\mathcal{D}_{\mathrm{ctx}}| = n_{\mathrm{fix}}$. Let q_A be a minimizer and define another predictor $\widetilde{q}_A$ such that

$$\widetilde{q}_A(Z) = q_A(Z) \qquad \text{whenever } |\mathcal{D}_{\mathrm{ctx}}| = n_{\mathrm{fix}}, \tag{33}$$

while allowing $\widetilde{q}_A$ to take arbitrary values whenever $|\mathcal{D}_{\mathrm{ctx}}| \neq n_{\mathrm{fix}}$. The two predictors have identical fixed-n_{fix} population risks. Thus, training only at $N = n_{\mathrm{fix}}$ imposes no population-risk constraint at any context size $n \neq n_{\mathrm{fix}}$. □

A.4 Proof of the Label-Sampling Variance Gap

*Proof of **Lemma 2**.* We prove the result for the event head, as the censoring-head proof is identical. Let

$$\xi := (\theta, N, \mathcal{D}_{\mathrm{ctx}}, \mathbf{x}^*) \tag{34}$$

collect all random variables sampled before drawing the *query* event time. Conditional on ξ, abbreviate

$$p := p_{\theta,T}(Z), \qquad q := q_T^{\omega}(Z), \qquad K := \kappa_{\mathcal{D}_{\mathrm{ctx}}}(\widetilde{T}), \tag{35}$$

where $\widetilde{T} \sim P_T^{\theta}(\cdot|\mathbf{x}^*)$. By the definition of p,

$$\Pr(K = \ell \mid \xi) = [p]_\ell, \qquad \ell = 1, \dots, L. \tag{36}$$

Let $z \in \mathbb{R}^L$ denote the event-head logits, so that $q = \mathrm{softmax}(z)$. For any target $u \in \Delta_{L-1}$, direct differentiation of softmax *cross-entropy* gives

$$\nabla_z \mathrm{CE}(u\|q) = q - u. \tag{37}$$

Consequently, the *oracle-distillation* and *sampled-label* logit gradients are

$$\begin{aligned} g_{\mathrm{OD}} &= q - p, \\ g_{\mathrm{SL}} &= q - \mathbf{e}_K = g_{\mathrm{OD}} + p - \mathbf{e}_K. \end{aligned} \tag{38}$$

The conditional mean of the one-hot target is

$$\mathbb{E}[\mathbf{e}_K \mid \xi] = \sum_{\ell=1}^{L} \Pr(K = \ell \mid \xi)\mathbf{e}_\ell = p. \tag{39}$$

Taking the conditional expectation of (38) and using (39) yields

$$\mathbb{E}[g_{\mathrm{SL}} \mid \xi] = g_{\mathrm{OD}}. \tag{40}$$

This proves conditional unbiasedness.

We next compute the conditional covariance. Since q, p, and g_{OD} are fixed given ξ,

$$\mathrm{Cov}(g_{\mathrm{SL}} \mid \xi) = \mathrm{Cov}(\mathbf{e}_K \mid \xi). \tag{41}$$

The second moment of the one-hot target is

$$\mathbb{E}[\mathbf{e}_K \mathbf{e}_K^\top \mid \xi] = \sum_{\ell=1}^{L} [p]_\ell \mathbf{e}_\ell \mathbf{e}_\ell^\top = \mathrm{diag}(p). \tag{42}$$

Combining (39) and (42) gives

$$\mathrm{Cov}(g_{\mathrm{SL}} \mid \xi) = \mathrm{diag}(p) - pp^\top. \tag{43}$$

The law of total covariance states that

$$\mathrm{Cov}(g_{\mathrm{SL}}) = \mathbb{E}[\mathrm{Cov}(g_{\mathrm{SL}} \mid \xi)] + \mathrm{Cov}(\mathbb{E}[g_{\mathrm{SL}} \mid \xi]). \tag{44}$$

Substituting (40) and (43) into (44) yields

$$\mathrm{Cov}(g_{\mathrm{SL}}) = \mathrm{Cov}(g_{\mathrm{OD}}) + \mathbb{E}\big[\mathrm{diag}(p) - pp^\top\big]. \tag{45}$$

It remains to characterize the excess covariance. For any $v \in \mathbb{R}^L$,

$$\begin{aligned} v^\top \left(\mathrm{diag}(p) - pp^\top\right) v &= \sum_{\ell=1}^{L} [p]_\ell v_\ell^2 - \left(\sum_{\ell=1}^{L} [p]_\ell v_\ell\right)^2 \\ &= \mathrm{Var}_{K \sim \mathrm{Cat}(p)}(v_K) \geq 0. \end{aligned} \tag{46}$$

Hence $\mathrm{diag}(p) - pp^\top$ is positive semidefinite, and so is its expectation. Moreover,

$$\mathrm{tr}\big(\mathrm{diag}(p) - pp^\top\big) = 1 - \|p\|_2^2. \tag{47}$$

For $p \in \Delta_{L-1}$, the right-hand side of (47) is zero if and only if one coordinate of p equals one and every other coordinate equals zero. Therefore, the excess covariance in (45) vanishes if and only if p is a point mass almost surely under the synthetic-task law. This establishes the covariance identity, positive-semidefiniteness, and the stated zero-covariance condition. □

A.5 Proof of the Excess-Risk Decomposition

*Proof of **Theorem 1**.* Fix a context size n, and let

$$Z = (\mathcal{D}_{\mathrm{ctx}}, \mathbf{x}^*) \sim \mathbb{Q}_n^\star. \tag{48}$$

For readability, we suppress the common argument Z from q_T^ω, $q_{\mathrm{PPD},T}$, $p_{\theta^\dagger,T}$, and $p_{\theta^\star,T}$. Assumption 1 ensures that the *conditional event-time distributions* represented by these binned targets are identified from the corresponding observable *right-censored* laws over the evaluation horizon.

The triangle inequality gives, pointwise in Z,

$$\mathrm{TV}(q_T^\omega, p_{\theta^\star,T}) \le \mathrm{TV}(q_T^\omega, q_{\mathrm{PPD},T}) + \mathrm{TV}(q_{\mathrm{PPD},T}, p_{\theta^\star,T})\,. \tag{49}$$

We bound the two terms on the right-hand side separately.

Term (I): Amortization Error. Pinsker's inequality (Tsybakov, 2008) gives, for each Z,

$$\mathrm{TV}(q_T^\omega, q_{\mathrm{PPD},T}) \le \sqrt{\frac{1}{2} D_{\mathrm{KL}}(q_{\mathrm{PPD},T} \,\|\, q_T^\omega)}. \tag{50}$$

Taking expectations under $\mathbb{Q}_n^\star$ and applying Jensen's inequality (Villani et al., 2009) to the concave map $u \mapsto \sqrt{u}$ yields

$$\mathbb{E}_{\mathbb{Q}_n^\star}\left[\mathrm{TV}(q_T^\omega, q_{\mathrm{PPD},T})\right] \quad \le \sqrt{\frac{1}{2} \mathbb{E}_{\mathbb{Q}_n^\star}\left[D_{\mathrm{KL}}(q_{\mathrm{PPD},T} \,\|\, q_T^\omega)\right]}. \tag{51}$$

By **Assumption 4**, $\mathbb{Q}_n^\star \ll \mathbb{M}_n$ and

$$\left\| \frac{\mathrm{d}\mathbb{Q}_n^\star}{\mathrm{d}\mathbb{M}_n} \right\|_\infty \le B_n. \tag{52}$$

Therefore,

$$\begin{aligned} &\mathbb{E}_{\mathbb{Q}_n^\star}\left[D_{\mathrm{KL}}(q_{\mathrm{PPD},T} \,\|\, q_T^\omega)\right] \\ &\quad = \int D_{\mathrm{KL}}(q_{\mathrm{PPD},T}(Z) \,\|\, q_T^\omega(Z)) \, \frac{\mathrm{d}\mathbb{Q}_n^\star}{\mathrm{d}\mathbb{M}_n}(Z) \, \mathbb{M}_n(\mathrm{d}Z) \\ &\quad \le B_n \mathbb{E}_{\mathbb{M}_n}\left[D_{\mathrm{KL}}(q_{\mathrm{PPD},T} \,\|\, q_T^\omega)\right]. \end{aligned} \tag{53}$$

It remains to relate the KL divergence under $\mathbb{M}_n$ to the fixed-size training excess risk. Applying (19) to the event channel and taking expectations under $\mathbb{M}_n$ gives

$$\mathcal{L}_{\mathrm{OD},T,n}(\omega) = \mathbb{E}_{\mathbb{M}_n}\left[\mathbb{H}(q_{\mathrm{PPD},T})\right] + \mathbb{E}_{\mathbb{M}_n}\left[D_{\mathrm{KL}}(q_{\mathrm{PPD},T} \,\|\, q_T^\omega)\right]. \tag{54}$$

The analogous identity for the censoring channel is

$$\mathcal{L}_{\mathrm{OD},C,n}(\omega) = \mathbb{E}_{\mathbb{M}_n}\left[\mathbb{H}(q_{\mathrm{PPD},C})\right] + \mathbb{E}_{\mathbb{M}_n}\left[D_{\mathrm{KL}}(q_{\mathrm{PPD},C} \,\|\, q_C^\omega)\right]. \tag{55}$$

Multiplying (55) by λ_c, adding it to (54), and using the fixed-size two-head Bayes-risk floor in (16) yields the exact identity

$$\mathcal{L}_{\mathrm{OD},n}(\omega) - \mathcal{L}_{\mathrm{OD},n}^\star = \mathbb{E}_{\mathbb{M}_n}\left[D_{\mathrm{KL}}(q_{\mathrm{PPD},T} \,\|\, q_T^\omega)\right] + \lambda_c \mathbb{E}_{\mathbb{M}_n}\left[D_{\mathrm{KL}}(q_{\mathrm{PPD},C} \,\|\, q_C^\omega)\right]. \tag{56}$$

The second term on the right-hand side of (56) is non-negative. Hence,

$$\mathbb{E}_{\mathbb{M}_n}\left[D_{\mathrm{KL}}(q_{\mathrm{PPD},T} \,\|\, q_T^\omega)\right] \le \mathcal{L}_{\mathrm{OD},n}(\omega) - \mathcal{L}_{\mathrm{OD},n}^\star. \tag{57}$$

Combining (51), (53), and (57) gives

$$\mathbb{E}_{\mathbb{Q}_n^\star}\left[\mathrm{TV}(q_T^\omega, q_{\mathrm{PPD},T})\right] \le \sqrt{\frac{B_n}{2}\left(\mathcal{L}_{\mathrm{OD},n}(\omega) - \mathcal{L}_{\mathrm{OD},n}^\star\right)}. \tag{58}$$

This is **term (I)**.

Terms (II) and (III): Prior Coverage and Posterior Concentration. Insert the binned conditional distribution under the pseudo-true DGP $\theta^\dagger$. The triangle inequality gives

$$\mathrm{TV}(q_{\mathrm{PPD},T}, p_{\theta^\star,T}) \le \mathrm{TV}\big(q_{\mathrm{PPD},T}, p_{\theta^\dagger,T}\big) + \mathrm{TV}\big(p_{\theta^\dagger,T}, p_{\theta^\star,T}\big)\,. \tag{59}$$

By (10),

$$q_{\mathrm{PPD},T} = \int_\Theta p_{\theta,T}\,\Pi(\mathrm{d}\theta|Z). \tag{60}$$

Using the coordinate definition of total variation and the triangle inequality for absolute values,

$$\begin{aligned}
&\mathrm{TV}\big(q_{\mathrm{PPD},T}, p_{\theta^\dagger,T}\big) \\
&= \frac{1}{2}\sum_{\ell=1}^{L}\left|\int_\Theta \big([p_{\theta,T}]_\ell - [p_{\theta^\dagger,T}]_\ell\big)\,\Pi(\mathrm{d}\theta|Z)\right| \\
&\le \frac{1}{2}\sum_{\ell=1}^{L}\int_\Theta \big|[p_{\theta,T}]_\ell - [p_{\theta^\dagger,T}]_\ell\big|\,\Pi(\mathrm{d}\theta|Z) \\
&= \int_\Theta \mathrm{TV}\big(p_{\theta,T}, p_{\theta^\dagger,T}\big)\,\Pi(\mathrm{d}\theta|Z).
\end{aligned} \tag{61}$$

For a fixed task input Z, all *binned conditional event-time distributions* use the same measurable discretization $\kappa_{\mathcal{D}_{\mathrm{ctx}}}$. Although this discretization is random before Z is observed, it is a fixed measurable map conditional on Z. To see why total variation cannot increase after discretization, let P and Q be any two probability measures on $\mathbb{R}_+$, and abbreviate $\kappa := \kappa_{\mathcal{D}_{\mathrm{ctx}}}$. The pushforward measure (Csiszár, 1967) is defined by

$$(\kappa_\# P)(B) := P\big(\kappa^{-1}(B)\big)\,, \qquad B \subseteq \{1,\dots,L\}.$$

Using the equivalent setwise representation of total variation (Gibbs & Su, 2002),

$$\begin{aligned}
\mathrm{TV}(\kappa_\# P, \kappa_\# Q) &= \sup_{B\subseteq\{1,\dots,L\}} |(\kappa_\# P)(B) - (\kappa_\# Q)(B)| \\
&= \sup_{B\subseteq\{1,\dots,L\}} \big|P\big(\kappa^{-1}(B)\big) - Q\big(\kappa^{-1}(B)\big)\big| \\
&\le \sup_{A\in\mathcal{B}(\mathbb{R}_+)} |P(A) - Q(A)| \\
&= \mathrm{TV}(P, Q).
\end{aligned} \tag{62}$$

The inequality holds because every set of the form $\kappa^{-1}(B)$ is a measurable subset of $\mathbb{R}_+$, whereas the supremum on the third line is taken over all measurable subsets. Thus, total variation contracts under measurable pushforwards (Csiszár, 1967).

Taking

$$P = P_T^\theta(\cdot|\mathbf{x}^*), \qquad Q = P_T^{\theta'}(\cdot|\mathbf{x}^*),$$

and recalling that $p_{\theta,T}(Z)$ and $p_{\theta',T}(Z)$ are their pushforwards through the same discretization, we obtain

$$\begin{aligned}
\mathrm{TV}(p_{\theta,T}(Z), p_{\theta',T}(Z)) &= \mathrm{TV}\Big((\kappa_{\mathcal{D}_{\mathrm{ctx}}})_\# P_T^\theta(\cdot|\mathbf{x}^*), (\kappa_{\mathcal{D}_{\mathrm{ctx}}})_\# P_T^{\theta'}(\cdot|\mathbf{x}^*)\Big) \\
&\le \mathrm{TV}\Big(P_T^\theta(\cdot|\mathbf{x}^*), P_T^{\theta'}(\cdot|\mathbf{x}^*)\Big)\,.
\end{aligned} \tag{63}$$

Assumption 2 therefore implies, for $P_X^{\theta^\star}$-almost every *query* covariate,

$$\mathrm{TV}(p_{\theta,T}, p_{\theta',T}) \le \mathcal{L}_{\mathrm{lip}}\,\mathrm{dist}_{\mathrm{obs}}(\theta,\theta'). \tag{64}$$

Applying (64) inside (61) gives

$$\mathrm{TV}\big(q_{\mathrm{PPD},T}, p_{\theta^\dagger,T}\big) \le \mathcal{L}_{\mathrm{lip}}\int_\Theta \mathrm{dist}_{\mathrm{obs}}(\theta,\theta^\dagger)\Pi(\mathrm{d}\theta|Z). \tag{65}$$

Applying (64) once more, now to $(\theta,\theta') = (\theta^\star,\theta^\dagger)$, gives

$$\mathrm{TV}\big(p_{\theta^\dagger,T}, p_{\theta^\star,T}\big) \le \mathcal{L}_{\mathrm{lip}}\,\mathrm{dist}_{\mathrm{obs}}(\theta^\star,\theta^\dagger). \tag{66}$$

Substituting (65) and (66) into (59) yields the pointwise bound

$$\mathrm{TV}(q_{\mathrm{PPD},T}, p_{\theta^\star,T}) \le \mathcal{L}_{\mathrm{lip}} \int_\Theta \mathrm{dist}_{\mathrm{obs}}(\theta, \theta^\dagger)\Pi(\mathrm{d}\theta|Z) + \mathcal{L}_{\mathrm{lip}}\,\mathrm{dist}_{\mathrm{obs}}(\theta^\star, \theta^\dagger). \tag{67}$$

Taking expectations under $Z \sim \mathbb{Q}_n^\star$ gives

$$\mathbb{E}_{\mathbb{Q}_n^\star}\left[\mathrm{TV}(q_{\mathrm{PPD},T}, p_{\theta^\star,T})\right] \le \mathcal{L}_{\mathrm{lip}}\mathbb{E}_{\mathbb{Q}_n^\star}\left[\int_\Theta \mathrm{dist}_{\mathrm{obs}}(\theta, \theta^\dagger)\Pi(\mathrm{d}\theta|Z)\right] + \mathcal{L}_{\mathrm{lip}}\,\mathrm{dist}_{\mathrm{obs}}(\theta^\star, \theta^\dagger). \tag{68}$$

Assumption 3 bounds the first expectation on the right-hand side of (68) by $C_{\mathrm{post}}\varepsilon_n$. Therefore,

$$\mathbb{E}_{\mathbb{Q}_n^\star}\left[\mathrm{TV}(q_{\mathrm{PPD},T}, p_{\theta^\star,T})\right] \le \underbrace{\mathcal{L}_{\mathrm{lip}}\,\mathrm{dist}_{\mathrm{obs}}(\theta^\star, \theta^\dagger)}_{\text{(II) prior coverage}} + \underbrace{\mathcal{L}_{\mathrm{lip}}C_{\mathrm{post}}\varepsilon_n}_{\text{(III) posterior concentration}}. \tag{69}$$

Finally, take expectations in (49) and substitute (58) and (69). This gives

$$\begin{aligned}&\mathbb{E}_{\mathbb{Q}_n^\star}\left[\mathrm{TV}(q_T^\omega, p_{\theta^\star,T})\right]\\ &\le \underbrace{\sqrt{\frac{B_n}{2}\left(\mathcal{L}_{\mathrm{OD},n}(\omega) - \mathcal{L}^\star_{\mathrm{OD},n}\right)}}_{\text{(I) amortization}} + \underbrace{\mathcal{L}_{\mathrm{lip}}\,\mathrm{dist}_{\mathrm{obs}}(\theta^\star, \theta^\dagger)}_{\text{(II) prior coverage}} + \underbrace{\mathcal{L}_{\mathrm{lip}}C_{\mathrm{post}}\varepsilon_n}_{\text{(III) posterior concentration}},\end{aligned} \tag{70}$$

which is exactly the claim of **Theorem 1**.

The pseudo-true DGP in **Assumption 3** need not be the unique Kullback–Leibler minimizer. The theorem applies to any selected $\theta^\dagger$ for which the stated *posterior*-concentration condition holds; membership in the argmin alone is not used as a substitute for that condition. □

The final statement in Section 3.3 expresses the fixed-size bound in terms of the mixed-size training objective. Define

$$\mathcal{L}^\star_{\mathrm{OD}} := \mathbb{E}_{N\sim P_N}\left[\mathcal{L}^\star_{\mathrm{OD},N}\right]. \tag{71}$$

By (56), every fixed-size excess risk is non-negative. Hence, for every $n \in \mathcal{N}$,

$$\begin{aligned}&\mathcal{L}_{\mathrm{OD}}(\omega) - \mathcal{L}^\star_{\mathrm{OD}}\\ &\quad= \sum_{m\in\mathcal{N}} P_N(m)\left(\mathcal{L}_{\mathrm{OD},m}(\omega) - \mathcal{L}^\star_{\mathrm{OD},m}\right)\\ &\quad\ge P_N(n)\left(\mathcal{L}_{\mathrm{OD},n}(\omega) - \mathcal{L}^\star_{\mathrm{OD},n}\right).\end{aligned} \tag{72}$$

Since $P_N(n) > 0$, (72) implies

$$\mathcal{L}_{\mathrm{OD},n}(\omega) - \mathcal{L}^\star_{\mathrm{OD},n} \le \frac{\mathcal{L}_{\mathrm{OD}}(\omega) - \mathcal{L}^\star_{\mathrm{OD}}}{P_N(n)}. \tag{73}$$

Substituting (73) into **term (I)** of **Theorem 1** gives

$$\begin{aligned}&\mathbb{E}_{\mathbb{Q}_n^\star}\left[\mathrm{TV}(q_T^\omega, p_{\theta^\star,T})\right]\\ &\le \sqrt{\frac{B_n}{2P_N(n)}\left(\mathcal{L}_{\mathrm{OD}}(\omega) - \mathcal{L}^\star_{\mathrm{OD}}\right)} + \mathcal{L}_{\mathrm{lip}}\,\mathrm{dist}_{\mathrm{obs}}(\theta^\star, \theta^\dagger) + \mathcal{L}_{\mathrm{lip}}C_{\mathrm{post}}\varepsilon_n.\end{aligned} \tag{74}$$

Thus, for every $n \in \mathcal{N}$, the fixed-size predictive-error bound can be controlled by the excess risk of the mixed-context-size training objective. The factor $1/P_N(n)$ quantifies the cost of transferring the aggregate mixed-size guarantee to a particular *context size*: *context sizes* receiving less probability mass during pretraining admit a looser bound. This establishes the extension of **Theorem 1** to mixed-context-size training.

B Related Work

Survival Analysis. *Classical survival methods* impose structured assumptions for modeling censored time-to-event (TTE) outcomes. CoxPH models covariate effects through a linear proportional-hazards predictor (Cox, 1972), CoxNet adds elastic-net regularization for *high-dimensional* settings (Simon et al., 2011), and cSVR learns censored risk rankings using support-vector optimization (Pölsterl et al., 2015). *Tree-based* alternatives capture nonlinearities and interactions through gradient boosting, component-wise gradient boosting, or ensembles of survival trees (Friedman, 2001; Pölsterl, 2020; Ishwaran et al., 2008). *Neural survival models* provide more flexible representations of the hazard or *event-time distribution*. DeepSurv and CoxTime learn nonlinear proportional- and non-proportional-hazards risk functions, respectively (Katzman et al., 2018; Kvamme et al., 2019). DeepHit, MTLR, Nnet-survival, and BNN-MTLR discretize the time axis and predict event probabilities across intervals (Lee et al., 2018; Yu et al., 2011; Gensheimer & Narasimhan, 2019; Qi et al., 2023a), whereas CQRNN directly estimates censored event-time quantiles (Pearce et al., 2022). *Continuous-time distributional models* instead parameterize complete event-time laws using mixtures, parametric families, monotone networks, or asymmetric distributions (Nagpal et al., 2021; Rindt et al., 2022; Han et al., 2022; Norman et al., 2024; Sheng & Henao, 2025; 2026; Hoseini et al., 2017). Despite their different inductive biases, these methods generally require dataset-specific fitting and hyperparameter selection. TTE-PFN instead amortizes this fitting process across tasks and predicts complete *event-time distributions* in context.

Tabular Foundation Models and Prior-Fitted Networks. Prior-data fitted networks (PFNs) pretrain transformers on tasks sampled from a synthetic *prior*, learning to approximate *posterior*-predictive inference from a labeled *context set* in a single forward pass (Müller et al., 2022). TabPFN established this paradigm for general tabular classification and regression (Hollmann et al., 2025). Subsequent tabular foundation models improve different components of the paradigm: TabICL emphasizes scalable in-context learning and diverse task generation (Qu et al., 2026), MITRA studies mixtures of complementary synthetic *priors* (Zhang et al., 2026), and TabDPT combines in-context prediction with self-supervised pretraining and retrieval over real tables (Ma et al., 2026).

Foundation Models for Survival Analysis. Recent work has begun adapting tabular foundation models to right-censored outcomes. StaticSurvivalTFM reformulates survival estimation as a sequence of binary classification problems over discretized time points, allowing an off-the-shelf tabular foundation model to predict survival probabilities (Kim et al., 2026). Survival In-Context pretrains a survival-specific PFN on synthetic time-to-event tasks (Seletkov et al., 2026), while concurrent work applies censored distributional regression to synthetically generated Weibull survival tasks (Böhm et al., 2026). Most closely related, SurvivalPFN amortizes Bayesian inference over identifiable *right-censored* DGPs and produces survival distributions without dataset-specific optimization (Qi et al., 2026). Our work builds on this prior-fitted survival paradigm but focuses on the statistical design of pretraining. In particular, we replace *sampled-label* supervision with *oracle distillation* from analytic conditional laws, randomize the *context size* during pretraining, and construct a mixture *prior* spanning complementary survival DGP families. We further characterize the learned posterior-predictive target, quantify the gradient-variance reduction from oracle supervision, and decompose test error into *amortization*, *prior coverage*, and *posterior concentration*.

C EXPERIMENTAL DETAILS

This section provides additional information about the experimental setup. All experiments were implemented using the PyTorch framework. Detailed descriptions of the datasets, evaluation metrics, our method and baseline models, and implementation specifics are provided in Appendix C.1, Appendix C.2 and Appendix C.3.

Hardware. We pretrain TTE-PFN once on a single NVIDIA H200 GPU in about 2.4 hours over 2,000 epochs, after which it predicts in-context with no per-dataset training. We run zero-shot evaluation of TTE-PFN and the *tabular foundation models* on the same GPU, and fit the *classical*, *tree-based*, and *neural* baselines on CPU.

C.1 DATASETS

Our benchmark spans 115 datasets, none of which is seen during training. We build on the evaluation suite of SurvivalPFN (Qi et al., 2026) and broaden it. Of the 101 *real cohorts*, 77 come from the SurvSet package (Drysdale, 2022). SurvivalPFN discards the SurvSet cohorts that are *longitudinal*, encode *competing risks*, or are *high-dimensional*, whereas we retain the full collection, collapsing each time-varying record to its subject's final follow-up interval and treating any competing event as the event of interest. The other 24 come from textbooks, recent papers, and public software packages. Across these 101 cohorts, *sample sizes* range from 26 to 52,422, *covariate counts* from 3 to 15,541, and *censoring proportions* from 0 to 0.94. The remaining 14 datasets are *synthetic*, drawn from the CQRNN benchmark (Pearce et al., 2022). Table 3 lists all three quantities for every dataset, and the sources cited above document the individual cohorts in full.

Table 3: Summary of the 115 benchmark datasets: encoded covariate dimension (**Feats**), sample size (**Size**), and proportion of right-censored observations (**Cens**). Two datasets are listed per row, ordered within each source group by **Feats**, then **Size**, then **Cens**, and read left to right.

Dataset	Feats	Size	Cens	Dataset	Feats	Size	Cens
SurvSet package (Drysdale, 2022) — 77 datasets							
e1684	3	284	0.31	acath	3	3,504	0.33
ovarian	4	26	0.54	glioma	4	37	0.38
breast	4	100	0.74	heart	4	103	0.27
diabetes	4	394	0.61	follic	4	541	0.36
dataDIVAT2	4	1,837	0.68	divorce	4	3,371	0.69
Melanoma	5	205	0.72	aids	5	467	0.60
epileptic	5	605	0.64	FRTCS	5	697	0.90
grace	5	1,000	0.68	pbc	6	312	0.60
csl	6	446	0.40	rdata	6	1,040	0.47
TRACE	6	1,878	0.49	UnempDur	6	3,241	0.39
prostateSurvival	6	14,294	0.94	Unemployment	7	452	0.43
veteran	8	137	0.07	scania	8	1,931	0.44
GBSG2	9	686	0.56	nwtco	9	4,028	0.86
Bergamaschi	10	82	0.66	ova	11	358	0.26
retinopathy	11	394	0.61	mgus	12	241	0.07
colon	12	929	0.51	Aids2	12	2,839	0.38
rott2	12	2,982	0.57	burn	13	154	0.69
cost	13	518	0.22	oldmort	13	4,603	0.57
uis	14	628	0.19	pharmacoSmoking	16	125	0.29
dataDIVAT3	16	4,267	0.94	dataDIVAT1	16	5,943	0.84
whas500	17	461	0.62	actg	17	1,151	0.92
Framingham	17	4,699	0.69	stagec	18	146	0.63
Pbc3	19	349	0.82	zinc	20	431	0.81
phpl04K8a	21	442	0.47	Z243	23	100	0.00
cgd	23	128	0.66	heartvalve	23	256	0.79
d.oropha.rec	24	192	0.28	prostate	25	502	0.29
Rossi	27	432	0.74	cancer	28	228	0.28
LeukSurv	29	1,043	0.16	wpbc	32	198	0.76
flchain	32	7,874	0.72	smarto	34	3,873	0.88

continued on next page

Dataset	Feats	Size	Cens	Dataset	Feats	Size	Cens
hdfail	36	52,422	0.94	vlbw	41	617	0.83
hepatoCellular	43	227	0.57	support2	65	9,105	0.32
Dialysis	72	6,805	0.76	rhc	73	5,735	0.35
nki70	76	144	0.67	micro.censure	81	117	0.78
dataOvarian1	162	912	0.40	NSBCD	549	115	0.67
MCLcleaned	574	92	0.30	chop	3,833	414	0.60
vdv	4,705	78	0.56	DBCD	4,919	295	0.73
AML_Bull	6,283	116	0.42	DLBCL	7,399	240	0.42
gse3143	8,663	158	0.68	gse4335	12,815	115	0.67
gse1992	15,541	124	0.72				
Additional real-world datasets (Qi et al., 2026) — 24 datasets							
anderson	3	42	0.29	larynx	3	90	0.44
kidtran	3	863	0.84	TMBImmuno	3	1,660	0.50
LGGGBM	5	638	0.62	BreastMSK	5	1,834	0.78
lupus	6	95	0.70	WHAS	6	1,638	0.58
NCCTG	7	228	0.28	GBSG	7	2,232	0.43
COVID	9	1,422	0.91	METABRIC	9	1,904	0.42
PDM	10	1,000	0.60	HFCR	11	299	0.68
SUPPORT	14	8,873	0.32	PBC	18	418	0.61
employee	19	14,999	0.76	ACTG175	21	2,139	0.76
BCCardiotox	25	531	0.90	churn	30	1,958	0.52
BMT	39	187	0.55	NACD	51	2,395	0.36
credit	57	1,000	0.30	MSKCC	210	7,562	0.71
Synthetic (CQRNN benchmark) (Pearce et al., 2022) — 14 datasets							
Weibull	1	2,800	0.21	Gaussian_linear	1	2,800	0.22
LogNorm	1	2,800	0.22	Gaussian_nonlinear	1	2,800	0.26
Exponential	1	2,800	0.34	Gaussian_uniform	1	2,800	0.67
Norm_light	4	2,800	0.23	Norm_med	4	2,800	0.47
Norm_same	4	2,800	0.51	Norm_heavy	4	2,800	0.77
LogNorm_light	8	2,800	0.24	LogNorm_same	8	2,800	0.52
LogNorm_med	8	2,800	0.54	LogNorm_heavy	8	2,800	0.75

Clinical Cohorts. Most *real cohorts* are clinical, with death, relapse, or disease progression as the event. Oncology is the largest domain and spans breast (GBSG, GBSG2, METABRIC, nki70, wpbc), colorectal (colon), prostate (prostate, prostateSurvival), lung (veteran, NCCTG), and lymphoma or leukemia (follic, LeukSurv). Cardiology contributes both long-run and acute outcomes, through the Framingham cohort (Framingham), the Worcester Heart Attack Study (WHAS, whas500), and heart-failure records (HFCR). Infectious disease appears as HIV trials (actg, ACTG175, Aids2) and COVID-19 outcomes (COVID), and transplantation as graft and patient survival (Dialysis, dataDIVAT1, kidtran, BMT). These cohorts are *low-dimensional*, so they vary the event law and the censoring rate while holding covariate width roughly fixed.

Genomic Cohorts. Ten cohorts are gene-expression studies in which the encoded feature count exceeds the number of patients. They include microarray series from the Gene Expression Omnibus (gse1992, gse3143, gse4335), lymphoma and leukemia expression panels (DLBCL, AML_Bull), and breast-cancer signatures (vdv, NSBCD, DBCD). Feature counts reach 15,541 against as few as 78 patients, the most adverse *feature-to-sample regime* in the benchmark.

Non-clinical Cohorts. A minority of cohorts lie outside medicine and extend the benchmark beyond clinical time-to-event laws. Reliability enters as hard-drive failure logs (hdfail), the largest cohort in the suite and among the most heavily censored. Economics and social science contribute unemployment duration (UnempDur, Unemployment), marital dissolution (divorce), criminal recidivism (Rossi), and historical mortality (oldmort, scania). Business applications supply customer and employee attrition (churn, employee) and credit default (credit).

Synthetic Cohorts. The 14 CQRNN cohorts (Pearce et al., 2022) carry a known ground truth. Each draws event times from a Gaussian, LogNormal, Weibull, or Exponential law and censoring times from an independent law calibrated to a target rate, with 2,800 samples per cohort. One family of variants sweeps the *censoring proportion* (Norm_light through Norm_heavy), and a second alters the covariate-to-event mapping and the form of the censoring law (Gaussian_linear,

Gaussian_nonlinear, Gaussian_uniform). Because the conditional law is known, calibration on these cohorts is measured against the truth rather than against an empirical proxy.

Preprocessing. We one-hot encode categorical covariates before any model sees the data, so the feature counts of Table 3 are encoded input dimensions rather than raw column counts. Several cohorts are wider than the input width some methods accept. Appendix C.3 states how each such method reduces them.

C.2 METRICS

All methods are evaluated with the same SurvivalEVAL implementation (Qi et al., 2023b). To remain consistent with Section 2, write the test set as

$$\mathcal{D}_{\text{test}} = \{(\mathbf{x}_i, y_i, \delta_i)\}_{i=1}^{n_{\text{te}}},$$

where $y_i = \min(T_i, C_i)$ is the observed time and $\delta_i = \mathbb{I}\{T_i \le C_i\}$ is the event indicator. For a model that returns a survival distribution, write its conditional survival function as $\widehat{S}_i(t) := \widehat{S}(t|\mathbf{x}_i)$ and define its predicted median event time by

$$\widehat{m}_i := \inf\{t \ge 0 : \widehat{S}_i(t) \le 1/2\}.$$

For a point-prediction-only baseline, $\widehat{m}_i$ denotes its predicted event time, and metrics requiring a full survival curve are left undefined. In either case, define the scalar risk score by $r_i := -\widehat{m}_i$, so larger values indicate earlier predicted failure. Let $\widehat{G}_{\text{tr}}$ be the Kaplan–Meier estimate of the censoring survival function computed on the training/context split, and let $\widehat{S}_{\text{KM}}$ be the Kaplan–Meier estimate of event survival on the test split. All time-integrated metrics are evaluated over a common horizon t_{eval} selected from the training-side event-time support, where $\widehat{G}_{\text{tr}}$ remains positive. We report seven metrics in three groups; $\uparrow$ and $\downarrow$ indicate whether larger or smaller values are preferred.

Predictive Accuracy. The pseudo-observation mean absolute error (MAE-PO, $\downarrow$; Qi et al. 2023a) evaluates point predictions while retaining censored subjects. Let $\widetilde{y}_i^{\text{PO}}$ denote the pseudo event-time target and let $\gamma_i^{\text{PO}} \ge 0$ denote its confidence weight, both computed by SurvivalEVAL. For an uncensored subject, $\widetilde{y}_i^{\text{PO}} = y_i$ and $\gamma_i^{\text{PO}} = 1$; for a censored subject, the pseudo target and weight are obtained from the marginal Kaplan–Meier estimate. Then

$$\text{MAE-PO} = \frac{\sum_{i=1}^{n_{\text{te}}} \gamma_i^{\text{PO}} \left|\widehat{m}_i - \widetilde{y}_i^{\text{PO}}\right|}{\sum_{i=1}^{n_{\text{te}}} \gamma_i^{\text{PO}}}. \tag{75}$$

The integrated Brier score (IBS, $\downarrow$; Graf et al. 1999) averages an inverse-probability-of-censoring-weighted squared error over time. At time $\tau \in [0, t_{\text{eval}}]$, define

$$\text{BS}(\tau) := \frac{1}{n_{\text{te}}} \sum_{i=1}^{n_{\text{te}}} \left[\frac{\widehat{S}_i(\tau)^2 \, \mathbb{I}\{y_i \le \tau,\, \delta_i = 1\}}{\widehat{G}_{\text{tr}}(y_i^-)} + \frac{\left(1 - \widehat{S}_i(\tau)\right)^2 \mathbb{I}\{y_i > \tau\}}{\widehat{G}_{\text{tr}}(\tau)} \right]. \tag{76}$$

The integrated score is

$$\text{IBS} := \frac{1}{t_{\text{eval}}} \int_0^{t_{\text{eval}}} \text{BS}(\tau)\, \mathrm{d}\tau. \tag{77}$$

Concordance. For pairwise concordance, define the tie-aware comparison operator

$$\text{cmp}(r_i, r_j) := \mathbb{I}\{r_i > r_j\} + \frac{1}{2}\mathbb{I}\{r_i = r_j\}.$$

Harrell's C-index (Harrell's CI, $\uparrow$; Harrell et al. 1982) is the fraction of comparable ordered pairs whose risk scores are correctly ranked:

$$\text{Harrell's CI} := \frac{\sum_{i=1}^{n_{\text{te}}} \sum_{\substack{j=1 \\ j \ne i}}^{n_{\text{te}}} \delta_i \, \mathbb{I}\{y_i < y_j\} \, \text{cmp}(r_i, r_j)}{\sum_{i=1}^{n_{\text{te}}} \sum_{\substack{j=1 \\ j \ne i}}^{n_{\text{te}}} \delta_i \, \mathbb{I}\{y_i < y_j\}}. \tag{78}$$

Uno's C-index (Uno's CI, ↑; Uno et al. 2011) applies inverse-probability-of-censoring weights to the same comparable pairs:

$$\text{Uno}'\text{s CI} := \frac{\sum_{i=1}^{n_{\text{te}}} \sum_{\substack{j=1 \\ j\neq i}}^{n_{\text{te}}} \frac{\delta_i \, \mathbb{I}\{y_i < y_j\} \, \text{cmp}(r_i, r_j)}{\widehat{G}_{\text{tr}}(y_i^-)^2}}{\sum_{i=1}^{n_{\text{te}}} \sum_{\substack{j=1 \\ j\neq i}}^{n_{\text{te}}} \frac{\delta_i \, \mathbb{I}\{y_i < y_j\}}{\widehat{G}_{\text{tr}}(y_i^-)^2}}. \tag{79}$$

The log-rank discrepancy (Log-Rank, ↓; Qi et al. 2023b) compares the observed survival sample $\{(y_i, \delta_i)\}$ with the predicted median times $\{\widehat{m}_i\}$, treating the latter as fully observed. At each distinct pooled failure time indexed by k, let $\mathcal{R}_k$ be the pooled risk set, $\mathcal{R}_k^{\text{obs}}$ its observed-sample subset, d_k the total number of failures, and d_k^{obs} the number of failures in the observed sample. Define

$$e_k^{\text{obs}} := d_k \frac{|\mathcal{R}_k^{\text{obs}}|}{|\mathcal{R}_k|},$$

$$v_k := \frac{d_k\big(|\mathcal{R}_k| - d_k\big)|\mathcal{R}_k^{\text{obs}}|\big(|\mathcal{R}_k| - |\mathcal{R}_k^{\text{obs}}|\big)}{|\mathcal{R}_k|^2\big(|\mathcal{R}_k| - 1\big)}.$$

The reported statistic is

$$\text{LR} := \frac{\big[\sum_k \big(d_k^{\text{obs}} - e_k^{\text{obs}}\big)\big]^2}{\sum_k v_k}. \tag{80}$$

A smaller value indicates closer agreement between the distributions of predicted and observed survival times.

Calibration. D-calibration (D-Cal, ↓; Haider et al. 2020) assesses whether the probability-integral-transform values $u_i := \widehat{S}_i(y_i)$ are uniformly distributed after accounting for censoring. Partition $[0, 1]$ into $K_{\text{cal}} = 10$ equal intervals. An uncensored subject contributes one unit of mass to the interval containing u_i, whereas a censored subject distributes its unit mass over the admissible interval $[0, u_i]$ according to the D-calibration rule. Let c_k be the resulting mass in calibration interval k and let $\bar{c} := n_{\text{te}}/K_{\text{cal}}$. The chi-square discrepancy is

$$\text{D-Cal} := \sum_{k=1}^{K_{\text{cal}}} \frac{(c_k - \bar{c})^2}{\bar{c}}. \tag{81}$$

Kaplan–Meier calibration (KM-Cal, ↓; Chapfuwa et al. 2020) compares the cohort-average predicted survival curve

$$\overline{S}_{\text{pred}}(t) := \frac{1}{n_{\text{te}}} \sum_{i=1}^{n_{\text{te}}} \widehat{S}_i(t)$$

with the test-set Kaplan–Meier curve:

$$\text{KM-Cal} := \frac{1}{t_{\text{eval}}} \int_0^{t_{\text{eval}}} \Big(\overline{S}_{\text{pred}}(t) - \widehat{S}_{\text{KM}}(t)\Big)^2 \, \mathrm{d}t. \tag{82}$$

The score is non-negative and equals zero under perfect population-level agreement over the evaluation horizon.

C.3 TTE-PFN and Baselines

TTE-PFN. We provide the implementation details of TTE-PFN, including its input representation, *two-way attention* backbone, *context-dependent* discretization, *oracle-distillation* targets, pretraining procedure, and inference protocol.

Inputs and Tokenization. A forward pass takes a single table consisting of n labeled *context* rows followed by m unlabeled *query* rows. Covariates occupy a fixed budget of $d_{\text{in}} = 16$ feature slots. Cohorts with fewer encoded covariates are zero-padded, whereas wider cohorts are restricted to 16 covariates as described in Appendix C.1. At inference, missing covariate values are imputed using the corresponding *context* means, and all covariates are then standardized using means and standard

deviations computed from the *context* rows only, with standardized values clipped to $[-10, 10]$. Thus, neither *query* outcomes nor statistics computed from *query* covariates enter preprocessing. The resulting table is tokenized *cell by cell*. Each covariate value is mapped to a $d_{\mathrm{emb}} = 192$ dimensional representation using a shared affine map together with a learned feature-slot embedding, yielding d_{in} covariate tokens per row. Each row additionally contains a *target token*: for a *context* row, it is obtained from an affine embedding of (y_i, δ_i), whereas a *query* row receives a learned mask embedding. Consequently, each row contains $d_{\mathrm{in}} + 1$ tokens, with the masked target token preventing label leakage into the *query* rows.

Two-way Attention **Backbone.** The backbone contains 12 identical transformer layers, each consisting of *feature-axis attention*, *sample-axis attention*, and a gated position-wise feed-forward block, with RMSNorm applied before each residual sublayer. *Feature-axis attention* operates over the $d_{\mathrm{in}} + 1$ tokens within each row and captures interactions among covariates and the target token. *Sample-axis attention* operates across rows within each token slot. Queries are formed for all rows, whereas keys and values are restricted to the labeled *context* rows, allowing information to flow from the *context* to the *query* without communication among *query* rows. The feed-forward block has width 384. Attention uses 8 heads with query–key normalization, and sample-axis queries are scaled by $\log_2 n/10$ to accommodate variation in *context size*. For each *query* row, the output corresponding to its target token is passed to two separate three-layer MLP heads with hidden width 256 and dropout 0.1. The two heads produce event- and censoring-time logits over the same L time bins. The resulting model contains 6.97M parameters.

Context-dependent **Discretization.** Both output heads operate on the *context-dependent* discretization $\kappa_{\mathcal{D}_{\mathrm{ctx}}}$ introduced in Section 3.1. We construct this transformation from the empirical distribution of the observed follow-up times in the *context*, using linear interpolation between the ordered observations and anchoring the transformation at the origin. The transformed time axis is then partitioned into $L = 1{,}024$ equal-width bins on $[0, 1]$. Because $\kappa_{\mathcal{D}_{\mathrm{ctx}}}$ is estimated exclusively from the labeled *context*, it adapts the discretization to the cohort-specific time scale without using *query* outcomes. Both censored and uncensored follow-up times contribute to this transformation, and the same discretization is shared by the event and censoring heads.

Oracle-distillation **Targets.** For a synthetic DGP θ and *query* covariate $\mathbf{x}^*$, the targets p_T and p_C in (2) are constructed directly from the corresponding *analytic* conditional distributions. Specifically, we evaluate each conditional quantile function at 500 equally spaced probability levels on $[0.001, 0.999]$, transform the resulting times through $\kappa_{\mathcal{D}_{\mathrm{ctx}}}$, and aggregate them over the L bins. The normalized bin masses provide deterministic approximations to the binned *event-* and *censoring-time conditional distributions*. Thus, target construction does not require sampling an event or censoring outcome, in contrast to the matched *sampled-label* objective analyzed in **Lemma 2**.

Pretraining. TTE-PFN is pretrained for 2,000 epochs, with 16 independently sampled DGPs per epoch, corresponding to 32,000 synthetic cohorts in total. Gradients are accumulated over 4 DGPs, resulting in 4 optimizer updates per epoch and 8,000 updates overall. For each task, the *context size* is drawn log-uniformly from $[200, 2{,}000]$, and the *oracle-distillation* loss is evaluated on 256 *query* rows from the same DGP. Additional held-out query rows are used only for monitoring. Optimization uses AdamW with learning rate 10^{-3} and weight decay 10^{-2}, with linear warmup over the first 15% of optimization steps followed by cosine decay to 5% of the peak learning rate. We use dropout 0.1 and set $\lambda_c = 1$ in (4). Pretraining takes approximately 2.4 hours on a single NVIDIA H200 GPU, the same allocation under which every method is timed in Table 7.

Checkpoint Selection and Inference. Models are checkpointed every 10 epochs and monitored on a fixed bank of 8 synthetic DGPs held out from pretraining at context sizes $\{256, 512, 1{,}024, 2{,}048\}$. No real evaluation cohort or *query* outcome is used for checkpoint monitoring or selection. We report the final checkpoint; the minimum validation loss occurs at epoch 1,900, within the final 5% of pretraining. For each benchmark split, the 70% training partition forms the labeled *context* and the remaining 30% forms the *query* set. When the training partition contains more than 2,000 observations, we randomly subsample 2,000 rows for the *context*; smaller cohorts are used in full. Inference requires a single forward pass and no gradient-based adaptation. The event head is decoded into 50 predicted quantiles at probability levels from 0.02 to 0.98 by linearly inverting the predicted binned CDF and subsequently applying $\kappa^{-1}_{\mathcal{D}_{\mathrm{ctx}}}$. For IBS and KM-Cal, we additionally apply the *context*-based recalibration procedure studied in Appendix D.2.

Synthetic Survival DGPs. The pretraining prior follows the five-family construction introduced in Section 3.2; Table 4 summarizes the mixture weights and family-specific settings. Each generator specifies a covariate law P_X^θ, independently sampled event- and censoring-risk surfaces ψ_T and ψ_C, and conditional families admitting analytic distribution and quantile functions.

Common Construction. The event and censoring branches are independently generated and the censoring branch is subsequently rescaled only by a covariate-independent constant, preserving $T \perp C \mid X$. Each DGP draws an effective covariate dimension d_x uniformly from $\{2, 4, 8, 12\}$ and zero-pads the representation to the network's $d_{\text{in}} = 16$ feature slots. For four of the five generator families, covariates are obtained by transforming an isotropic Gaussian sample with a random rotation, coordinate-wise scaling and shifting, and the bounded transformation $\mathbf{x} = 1 + \tanh(0.6z)$. The `Bernstein` generator instead uses independent $\mathcal{U}(0, 2)$ covariates, so the mixture spans both correlated and uncorrelated covariate laws.

Censoring Calibration. Each DGP draws a target censoring proportion ρ_θ from the family-specific range in Table 4. We calibrate a multiplicative censoring-time scale s on a reference covariate sample so that $\Pr(sC < T)$ approximately matches ρ_θ. Because s is independent of $\mathbf{x}$, this transformation preserves conditional independence and retains an *analytic* censoring quantile function.

Table 4: The five generator families of the pretraining prior $\pi = \sum_k w_k \pi_k$. Each row states the mixture weight, the map from covariates to distribution parameters (*risk surface*), the conditional family with its closed-form quantile, and the range of the per-DGP target censoring proportion.

Generator	w_k	Risk surface ψ	Family $\mathcal{F}$	ρ_θ
`Tree`	0.30	random axis-aligned tree, depth $\mathcal{U}\{2, \ldots, 5\}$, piecewise constant	LogNormal	$\mathcal{U}(0.15, 0.60)$
`DeepMLP`	0.30	random MLP, 2–4 layers, mixed tanh/sin/leaky-ReLU	LogNormal	$\mathcal{U}(0.15, 0.60)$
`Narrow`	0.15	one-hidden-layer tanh MLP with large gain	Normal, small scale	$\mathcal{U}(0.15, 0.55)$
`Parametric`	0.15	one-hidden-layer tanh MLP on location and scale	Normal / LogNormal / Weibull / Exponential	$\mathcal{U}(0.15, 0.65)$
`Bernstein`	0.10	one-hidden-layer tanh MLP on monotone control points	Bernstein transform of $\mathcal{U}(0, 1)$	$\mathcal{U}(0.15, 0.65)$

Tree. The `Tree` generator uses a LogNormal conditional law whose location and scale parameters are piecewise constant over a random axis-aligned binary tree of depth 2–5. Internal nodes randomly select covariates and split thresholds, whereas each leaf independently determines the LogNormal parameters. The independently generated censoring branch follows the same construction. This component contributes axis-aligned interaction structure to the prior.

DeepMLP. The `DeepMLP` generator also uses a LogNormal conditional family, with $(\mu, \log \sigma)$ parameterized by a random MLP containing 2–4 hidden layers. Layer widths are drawn from $\{8, 16, 24, 32\}$ and activations from $\tanh, \sin,$ leaky-ReLU. This component contributes smooth nonlinear interaction structure complementary to `Tree`.

Narrow. The `Narrow` generator uses a Normal conditional law with a high-gain one-hidden-layer MLP controlling its location and a comparatively small conditional scale. It therefore produces concentrated conditional distributions with strong covariate-dependent separation and provides the high-discrimination component of the prior.

Parametric. The `Parametric` generator selects among Normal, LogNormal, Weibull, and Exponential conditional families, with a one-hidden-layer tanh MLP controlling the corresponding distribution parameters. This component broadens support across standard event-time distributions and includes the four families used by the synthetic benchmark datasets of Appendix C.1.

Bernstein. The `Bernstein` generator follows the monotone Bernstein-polynomial construction of SurvivalPFN (Qi et al., 2026). Its covariate-dependent control points define a monotone conditional quantile function, permitting flexible distributional shapes while retaining analytic oracle targets.

***Prior* Composition.** The five generators provide complementary coverage of the three axes introduced in Section 3.2: `Tree` and `DeepMLP` contribute interaction structure, `Narrow` emphasizes

discrimination, `Parametric` broadens standard distribution-family coverage, and `Bernstein` provides flexible conditional shapes. The factorial ablation in Section 4 evaluates the contribution of the complete prior, while Appendix D.2 reports the leave-one-family-out analysis.

Baselines. All baselines use the same five $70/30$ train–test partitions. Unless otherwise stated, covariates are mean-imputed and standardized using statistics computed on the training partition only, and cohorts with more than 100 encoded covariates are reduced to 100 principal components fitted on the training partition (Jolliffe & Cadima, 2016). All metrics are computed using the common evaluation implementation described in Appendix C.2. For publicly released baselines, we use the authors' implementations and recommended settings unless otherwise stated below; complete configurations are provided with the Supplementary Material.

Tabular foundation models. SurvivalPFN (Qi et al., 2026) is evaluated using the authors' released checkpoint.[1] StaticTFM (Kim et al., 2026) reformulates survival prediction as binary classification over multiple time cut points using TabPFN as its underlying classifier. TabPFN (Hollmann et al., 2025)[2] and TabICL (Qu et al., 2026)[3] are applied as in-context regressors to uncensored support observations. MITRA (Zhang et al., 2026) is evaluated through its AutoGluon implementation, [4] whereas TabDPT (Ma et al., 2026) uses the authors' released implementation [5]. MITRA and TabDPT return a single event-time prediction and therefore do not provide distributional outputs. For computational tractability, the in-context support is capped at $1{,}000$ observations for TabPFN, TabICL, TabDPT, and each StaticTFM classifier, and at 500 observations for MITRA; StaticTFM additionally applies PCA above 40 encoded covariates.

Classical models. CoxPH (Cox, 1972), CoxNet (Simon et al., 2011), and cSVR (Pölsterl et al., 2015) are implemented with scikit-survival (Pölsterl, 2020).[6] CoxPH and CoxNet provide fitted survival functions, whereas cSVR produces a risk score only.

Tree-based models. RSF (Ishwaran et al., 2008), GB (Friedman, 2001), and CWGB (Pölsterl, 2020) are also implemented with scikit-survival. For cohorts with more than $2{,}500$ training observations, these models are fitted on a random subsample of $2{,}500$ observations to bound computational cost.

Neural discrete-time models. DeepHit (Lee et al., 2018), MTLR (Yu et al., 2011), Nnet-survival (Gensheimer & Narasimhan, 2019), DeepSurv (Katzman et al., 2018), and CoxTime (Kvamme et al., 2019) are implemented with pycox.[7] We use a common two-layer MLP backbone for these models and retain the corresponding model-specific objectives and output parameterizations. CQRNN (Pearce et al., 2022) is implemented from the authors' released code,[8] whereas BNN-MTLR (Qi et al., 2023a) uses its corresponding reference implementation.[9]

Neural continuous-time models. DSM-Weibull and DSM-LogNormal (Nagpal et al., 2021) are implemented with auton-survival.[10] SurvivalMDN (Han et al., 2022) uses the authors' released mixture-density implementation,[11] whereas SuMo-net (Rindt et al., 2022) uses the released monotone-survival implementation.[12] DeepAFT-Weibull and DeepAFT-Loglogistic are our implementations of neural accelerated-failure-time models trained by censored maximum likelihood. ALD (Sheng & Henao, 2025)[13] and ICALD (Sheng & Henao, 2026)[14] are reimplemented from their corresponding reference code, and LogNorm uses the same training pipeline with a LogNormal likelihood. Complete implementation settings and model configurations are included in the Supplementary Material.

[1] `https://github.com/rgklab/SurvivalPFN`
[2] `https://github.com/PriorLabs/TabPFN`
[3] `https://github.com/soda-inria/tabicl`
[4] `https://github.com/autogluon/autogluon`
[5] `https://github.com/layer6ai-labs/TabDPT`
[6] `https://github.com/sebp/scikit-survival`
[7] `https://github.com/havakv/pycox`
[8] `https://github.com/TeaPearce/Censored_Quantile_Regression_NN`
[9] `https://github.com/shi-ang/BNN-ISD`
[10] `https://github.com/autonlab/auton-survival`
[11] `https://github.com/XintianHan/Survival-MDN`
[12] `https://github.com/MrHuff/Sumo-Net`
[13] `https://github.com/demingsheng/ALD`
[14] `https://github.com/demingsheng/ICALD`

D Additional Results

D.1 Overall Results

Figure 6 expands the aggregate comparison in Figure 2 to all 29 methods and all seven evaluation metrics. Ranks are first computed within each dataset and then averaged over the datasets on which the corresponding metric is defined. This dataset-level ranking prevents large cohorts from dominating the comparison and places methods with different output parameterizations on a common scale.

At the category level, TTE-PFN obtains the best *overall* rank of 8.09 and the best *accuracy* rank of 4.96 (Table 5). Its advantage is therefore concentrated on the primary goal of this work: estimating the complete conditional survival distribution accurately. CWGB is the closest method overall and gives the best aggregate *calibration*, whereas StaticTFM gives the best aggregate *concordance*. Neither specialized advantage transfers uniformly across categories: CWGB is weaker on *concordance*, while StaticTFM falls to a *calibration* rank of 15.22. TTE-PFN instead remains competitive in both categories, producing the best balance across the three objectives.

Table 5: Numerical results underlying the category-level rank comparison in Figure 2 and mean rank over the benchmark datasets (lower is better) for the category-level panels of Figure 6. *Accuracy* is the equally weighted mean of the dataset-level ranks for MAE-PO and IBS; *Concordance* is the equally weighted mean of the dataset-level ranks for Harrell's C-index, Uno's C-index, and Log-Rank; *Calibration* is the equally weighted mean of the dataset-level ranks for D-Calibration and KM-Calibration; and *Overall* is the equally weighted mean of the dataset-level ranks across all seven metrics. Ranks are first computed separately for each metric within each dataset and are then averaged over metrics and datasets. The best rank in each column is **highlighted**, and the per-metric breakdown is reported in Table 6.

Method	Overall	Accuracy	Concordance	Calibration
TTE-PFN	**8.09**	**4.96**	10.87	7.01
CWGB	8.47	7.83	10.33	**6.16**
RSF	9.37	8.69	11.45	6.83
CoxPH	9.62	10.52	10.80	6.85
CoxNet	9.79	10.48	11.03	7.13
GB	10.08	8.30	12.34	8.42
SurvivalPFN	10.76	8.17	13.17	9.95
StaticTFM	11.62	10.48	**9.77**	15.22
DSM-LogNormal	11.73	11.01	12.38	11.50
BNN-MTLR	11.81	11.02	13.52	9.88
DSM-Weibull	12.02	11.72	12.56	11.58
DeepHit	14.68	12.83	17.90	11.77
ICALD	14.72	14.14	14.82	15.02
DeepSurv	15.11	16.25	16.41	12.01
MTLR	15.69	15.07	17.34	13.86
Nnet-survival	15.76	14.55	17.59	14.18
TabICL	16.10	15.49	13.05	21.12
LogNorm	16.16	17.05	15.84	15.74
TabPFN	16.25	15.77	13.10	21.27
ALD	16.28	16.82	15.92	16.22
CQRNN	16.43	17.26	16.53	15.44
SuMoNet	16.49	18.92	16.27	14.39
TabDPT	18.42	17.59	14.11	25.65
DeepAFT-Weibull	18.63	21.20	17.96	16.96
MITRA	18.96	18.27	14.92	25.65
SurvivalMDN	19.25	20.64	18.71	18.71
DeepAFT-Loglogistic	19.60	21.81	19.39	17.67
cSVR	21.34	26.18	15.21	25.65
CoxTime	26.74	26.18	27.77	25.65

Table 6 provides the corresponding metric-level results. TTE-PFN ranks first on both distributional-accuracy metrics, with mean ranks of 5.09 on MAE-PO and 4.78 on IBS. The remaining col-

umn winners are specialized across different model families: StaticTFM on Harrell's and Uno's C-indices, TabDPT on Log-Rank, RSF on D-Calibration, and CWGB on KM-Calibration. This fragmentation illustrates why no individual metric is sufficient to characterize a survival-distribution estimator. In contrast, TTE-PFN has no severe failure mode across the seven metrics: its mean ranks range from 4.78 to 11.93, and this consistency yields the best aggregate rank.

The comparison with SurvivalPFN is especially informative because the two methods share the same in-context survival-prediction setting. TTE-PFN improves on SurvivalPFN for every metric, reducing the mean rank by 2.18 on MAE-PO, 4.31 on IBS, 2.15 on Harrell's C-index, 0.26 on Uno's C-index, 4.52 on Log-Rank, 1.20 on D-Calibration, and 4.68 on KM-Calibration. The largest gains occur on IBS, Log-Rank, and KM-Calibration, but the direction is consistent across all seven columns. The overall improvement from 10.76 to 8.09 is therefore not explained by a single favorable metric.

Table 6: Mean rank over the benchmark for every individual metric (lower is better), providing the per-metric breakdown of Table 5. The best rank in each column is **highlighted**. Methods without a full survival distribution (cSVR, MITRA, and TabDPT) are assigned a rank one below the worst completed method on the metrics they cannot produce (Appendix C.2), rather than being dropped from those comparisons.

Method	Overall	MAE-PO	IBS	Harrell's CI	Uno's CI	Log-Rank	D-Cal	KM-Cal
TTE-PFN	**8.09**	**5.09**	**4.78**	9.70	10.96	11.93	7.52	6.50
CWGB	8.47	8.36	7.46	10.17	9.59	11.25	6.23	**6.09**
RSF	9.37	10.53	6.79	12.04	11.38	10.94	**5.97**	7.70
CoxPH	9.62	13.68	7.30	9.06	8.80	14.53	6.59	7.11
CoxNet	9.79	13.30	7.63	9.33	9.10	14.67	7.13	7.13
GB	10.08	8.08	8.50	11.69	12.13	13.20	8.23	8.62
SurvivalPFN	10.76	7.27	9.09	11.85	11.22	16.45	8.72	11.18
StaticTFM	11.62	8.47	12.52	**8.03**	**8.31**	12.95	12.85	17.59
DSM-LogNormal	11.73	11.04	11.04	12.43	12.93	11.77	9.54	13.46
BNN-MTLR	11.81	11.75	10.36	12.23	11.70	16.63	9.11	10.65
DSM-Weibull	12.02	14.12	9.30	11.66	11.81	14.22	9.08	14.07
DeepHit	14.68	11.11	14.61	20.53	19.06	14.10	11.62	11.92
ICALD	14.72	16.21	12.00	13.18	12.74	18.54	15.29	14.75
DeepSurv	15.11	19.32	13.23	15.09	14.37	19.79	13.30	10.73
MTLR	15.69	16.18	14.05	17.58	16.32	18.13	15.10	12.61
Nnet-survival	15.76	15.17	13.89	17.70	16.73	18.34	14.35	14.02
TabICL	16.10	10.22	20.64	16.45	16.74	5.95	20.40	21.83
LogNorm	16.16	18.22	15.92	14.58	14.80	18.15	18.30	13.17
TabPFN	16.25	10.57	21.04	16.43	16.89	5.98	20.39	22.16
ALD	16.28	18.68	14.95	14.92	13.77	19.08	18.09	14.35
CQRNN	16.43	20.42	14.23	17.59	18.37	13.65	13.85	17.03
SuMoNet	16.49	20.95	16.85	15.24	14.84	18.73	17.21	11.58
TabDPT	18.42	9.42	25.64	19.01	17.97	**5.34**	25.65	25.65
DeepAFT-Weibull	18.63	23.69	18.74	18.43	17.76	17.69	17.78	16.13
MITRA	18.96	10.78	25.64	19.67	19.53	5.55	25.65	25.65
SurvivalMDN	19.25	20.68	20.54	22.76	20.87	12.52	18.90	18.53
DeepAFT-Loglogistic	19.60	24.17	19.45	19.70	20.67	17.81	19.35	15.98
cSVR	21.34	26.70	25.64	8.85	9.12	27.65	25.65	25.65
CoxTime	26.74	26.70	25.64	28.62	27.04	27.65	25.65	25.65

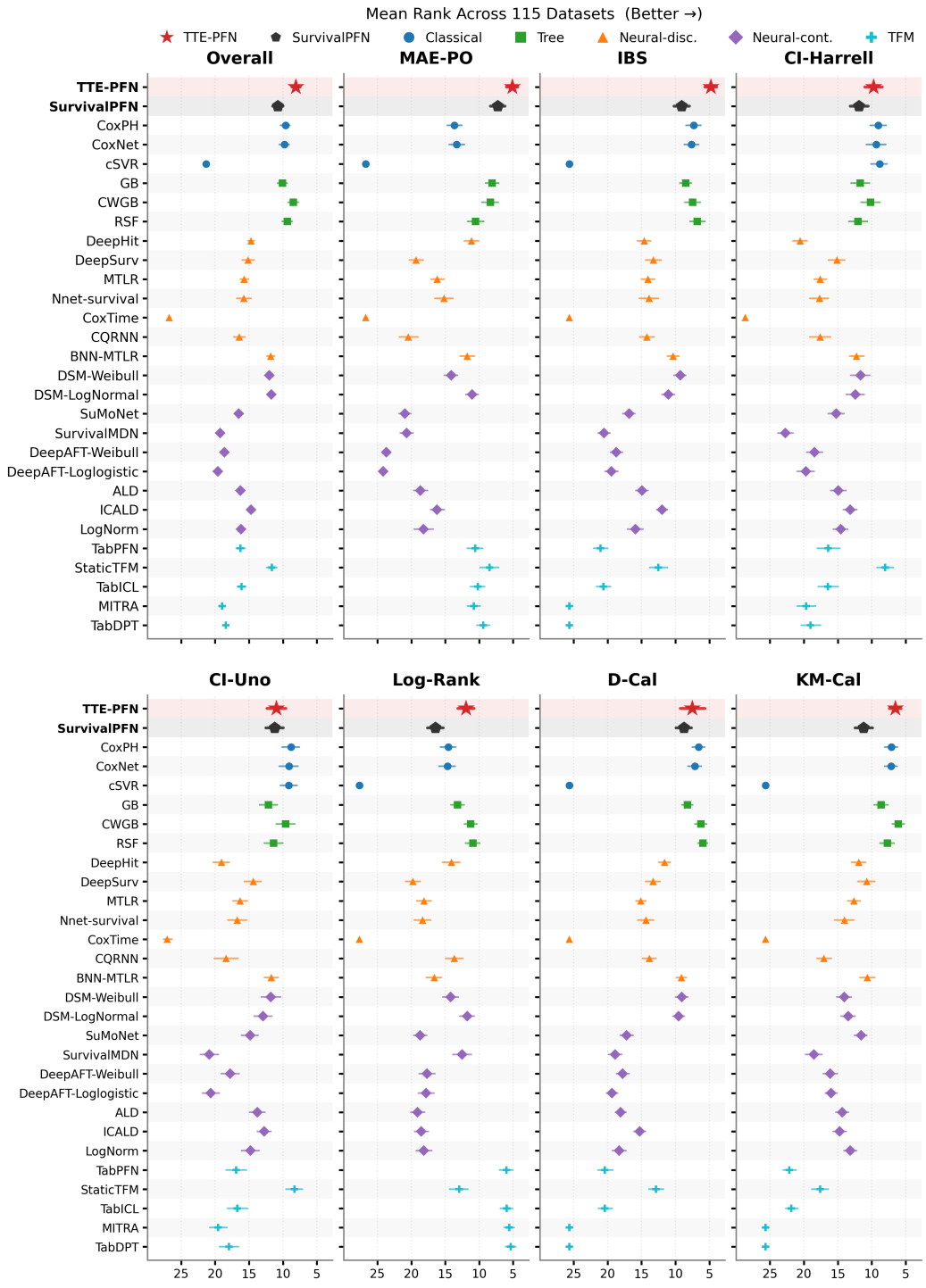


Figure 6: Complete benchmark ranking across all 29 methods. Lower ranks indicate better performance. The *Overall* panel gives equal weight to each of the seven evaluation metrics, while the remaining panels report the seven individual metrics. Points denote mean dataset-level ranks and horizontal bars denote 95% bootstrap confidence intervals. Colors and marker shapes identify method families. TTE-PFN and SurvivalPFN are highlighted separately for direct comparison.

D.2 Case Studies

Case Study I: Factorial Ablation of TTE-PFN's Design Components

The main-text ablation in Table 2 summarizes how the three proposed components affect aggregate performance: (A) *oracle-distillation* supervision, (B) *context-size* randomization, and (C) the diverse synthetic *prior*. Figure 7 shows the same 2^3 factorial design separately for each metric. All ranks in this case study are computed among the eight ablation settings. They should therefore be interpreted comparatively within the factorial design, rather than compared numerically with the 29-method ranks in Table 6.

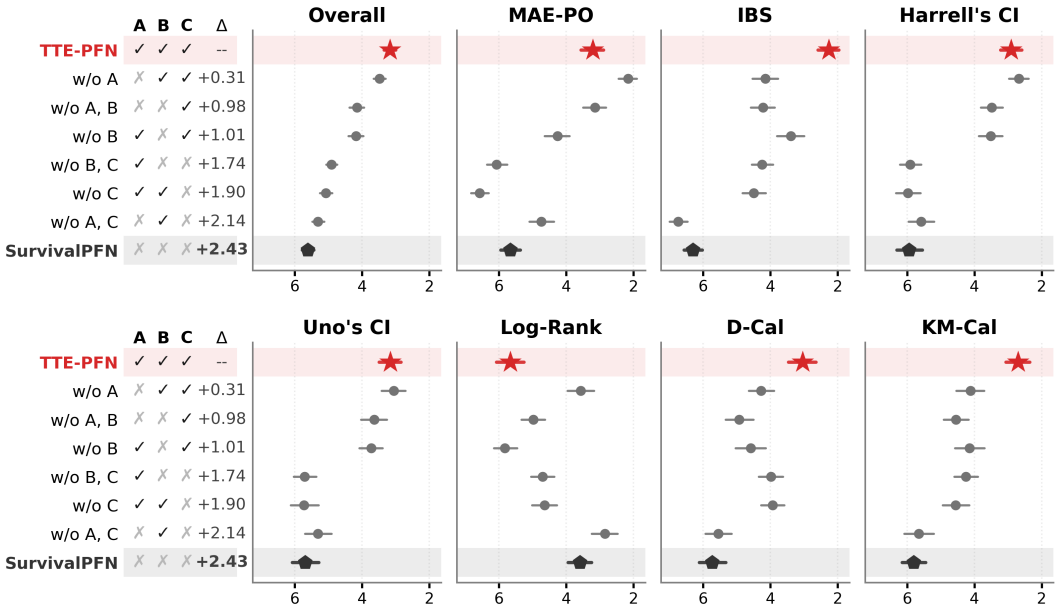


Figure 7: Metric-wise factorial ablation across the eight combinations of the three TTE-PFN components: (A) *oracle-distillation* supervision, (B) *context-size* randomization during pretraining, and (C) the proposed diverse *prior*. Checkmarks indicate retained components and crosses indicate removed components. The leftmost annotation reports the increase Δ in *Overall* mean rank relative to the complete TTE-PFN. Points denote mean dataset-level ranks and horizontal bars denote 95% bootstrap confidence intervals. Lower is better. The complete model and SurvivalPFN are highlighted in red and black, respectively.

The complete TTE-PFN configuration ranks first overall among the eight variants, with an *overall* mean rank of 3.17. Removing any single component worsens this aggregate rank. The largest single-component degradation comes from replacing the diverse *prior* with the Bernstein *prior* ($\Delta = +1.90$), followed by fixed-context pretraining ($\Delta = +1.01$), and then *sampled-label* supervision ($\Delta = +0.31$). Removing all three components recovers SurvivalPFN, which has the worst *overall* rank of 5.60 ($\Delta = +2.43$). Thus, the ordering inferred from the aggregate ablation—diverse *prior*, *context-size* randomization, then *oracle distillation*—is also visible in the complete rank profiles.

The metric-wise view clarifies why the gain from *oracle distillation* is modest in the aggregate but still meaningful. The *sampled-label* variant (w/o A) performs well on MAE-PO and the concordance metrics, but deteriorates on IBS, D-Calibration, and KM-Calibration. This pattern is consistent with the theory in Section 3.1: *oracle distillation* does not change the matched population minimizer, but removes label-sampling variance and provides direct supervision for the full conditional distribution. Its clearest empirical benefit therefore appears on distribution-sensitive *accuracy* and *calibration* metrics rather than uniformly on every ranking metric.

Context-size randomization has the broadest metric-wise effect. Relative to the complete model, the fixed-*context* variant (w/o B) is worse on all seven metrics, which agrees with Corollary 1: mixed-size training imposes the posterior-predictive target throughout the support of the training *context-size* distribution, whereas fixed-size training constrains only one context size.

The diverse prior produces the largest *overall* contribution. Removing it (w/o C) sharply worsens MAE-PO and both C-indices and also degrades both *calibration* metrics, indicating that broader coverage of interaction structures, *discrimination* levels, and distributional shapes improves transfer across heterogeneous real cohorts.

Finally, the combined ablations show why aggregate evaluation is necessary. For example, the w/o A,C variant achieves the best Log-Rank rank among the eight settings, yet performs poorly on IBS and both *calibration* metrics and has an overall degradation of $+2.14$. Similarly, SurvivalPFN remains competitive on Log-Rank but is near the bottom on distributional *accuracy* and *calibration*. An isolated gain on a *concordance* statistic can therefore coexist with a substantially worse survival-distribution estimator. Taken together, the full benchmark and the factorial case study empirically support the design guidance provided by **Theorem 1** and **Corollary 1**: *oracle distillation* facilitates the reduction of the *amortization* **term (I)** by removing *label-sampling* noise, the diverse *prior* reduces the *prior*-coverage **term (II)**, and *context-size* randomization enforces the *posterior*-predictive target across cohort sizes, extending the fixed-N guarantee throughout the support of the training distribution.

Case Study II: Contribution of Individual *Prior* Components

The factorial ablation in **Case Study I** shows that the diverse synthetic *prior* provides the largest aggregate contribution among the three components of TTE-PFN. We now examine which generator families in the five-family mixture described in Section 3.2 are responsible for this gain. For each leave-one-family-out variant, we remove one component π_k while retaining the remaining four generators and keeping the architecture, training objective, and context-size distribution fixed. We additionally include the Bernstein-only configuration, corresponding to the restricted generator family underlying the prior used by SurvivalPFN. All ranks in this analysis are computed among the seven configurations shown in Figure 8. They should therefore be interpreted comparatively within this prior ablation rather than compared numerically with the ranks in Table 6.

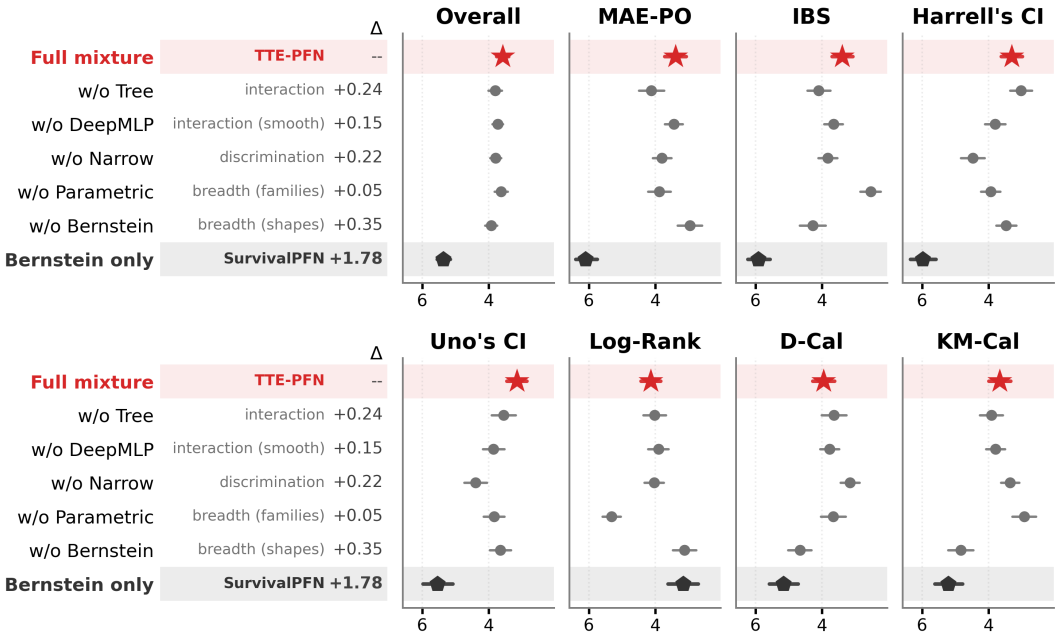


Figure 8: Leave-one-family-out ablation of the synthetic *prior*. The complete mixture contains the five generator families described in Section 3.2. Each *w/o* configuration removes the indicated family while retaining all remaining generators. The `Bernstein`-only configuration represents the restricted *prior* associated with SurvivalPFN. The leftmost annotation reports the increase Δ in *Overall* mean rank relative to the complete mixture. Points denote mean dataset-level ranks and horizontal bars denote 95% bootstrap confidence intervals. Lower is better. The complete mixture and the `Bernstein`-only reference are highlighted in red and black, respectively.

The complete five-family mixture achieves the best *Overall* rank among the seven configurations. Removing any individual generator worsens aggregate performance: the increase is $+0.35$ without

Bernstein, +0.24 without Tree, +0.22 without Narrow, +0.15 without DeepMLP, and +0.05 without Parametric. Thus, every family contributes positively, although the magnitude of its aggregate contribution differs.

The largest leave-one-family-out degradation occurs when Bernstein is removed, indicating that its flexible distributional shapes provide important breadth beyond the conventional parametric families. However, breadth alone is insufficient: restricting the prior to the Bernstein family increases the *Overall* rank by +1.78. The interaction structures supplied by Tree and DeepMLP, together with the high-discrimination regimes supplied by Narrow, are therefore complementary to the distributional flexibility of Bernstein.

The metric-wise profiles further show that no individual generator is uniformly necessary for every metric. Removing a family can occasionally improve an isolated metric while simultaneously worsening others, so the relatively small aggregate change from removing Parametric, for example, should not be interpreted as complete redundancy. Instead, the full mixture provides the most consistent balance across predictive *accuracy*, *concordance*, and *calibration*. These results support the three-axis construction in Section 3.2: combining complementary interaction structures, discrimination levels, and conditional-distribution shapes broadens the support of π and empirically reduces the *prior*-coverage **term (II)** in **Theorem 1**.

Case Study III: Sensitivity to the Censoring Weight λ_c.

The oracle-distillation objective in (4) combines the primary event-time loss with an auxiliary censoring-time loss weighted by λ_c. Although we use λ_c=1 throughout the main experiments, we assess whether performance is sensitive to this choice by evaluating $\lambda_c \in \{0, 0.25, 0.5, 1, 2, 4\}$ while holding all other training and evaluation settings fixed. The setting λ_c=0 removes direct supervision of the censoring head, whereas values larger than one place progressively greater weight on the auxiliary censoring objective. All ranks are computed among the six values of λ_c shown in Figure 9.

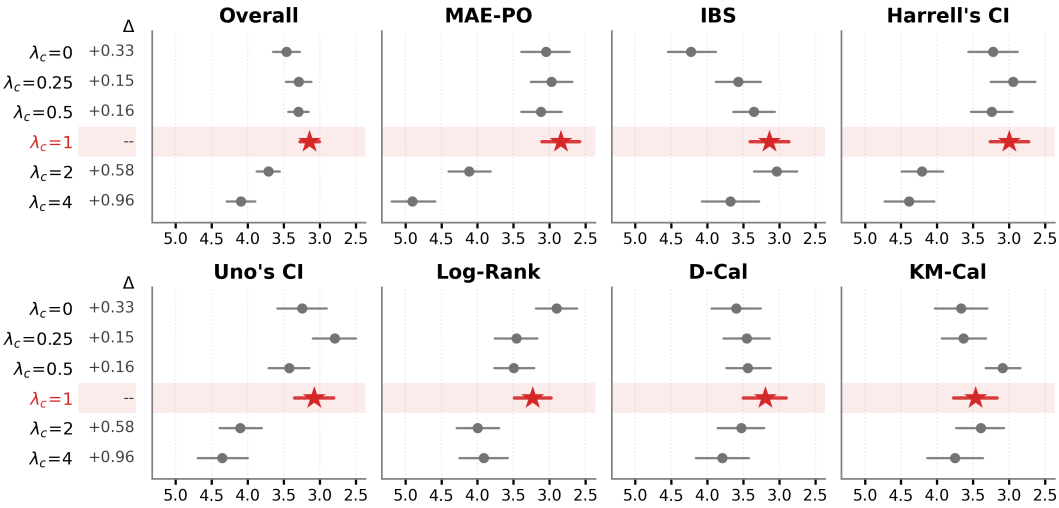


Figure 9: Sensitivity to the censoring-loss weight λ_c in (4). The leftmost annotation reports the increase Δ in *Overall* mean rank relative to the default λ_c=1. Points denote mean dataset-level ranks and horizontal bars denote 95% bootstrap confidence intervals. Lower is better. The default setting λ_c=1 is highlighted in red.

Equal weighting of the event and censoring channels, λ_c=1, achieves the best *Overall* rank. Performance remains relatively stable under moderate reductions of the auxiliary weight: λ_c=0.25 and λ_c=0.5 increase the *Overall* rank by only +0.15 and +0.16, respectively. This indicates that TTE-PFN is not narrowly tuned to a single precise value of λ_c within the moderate-weight regime.

Removing censoring supervision entirely is less effective: λ_c=0 increases the *Overall* rank by +0.33. This result suggests that the auxiliary censoring head provides useful training signal to the shared representation, even though the *event-time distribution* remains the primary prediction target. On the other hand, over-weighting the auxiliary objective causes a more substantial deterioration. The increases reach +0.58 at λ_c=2 and +0.96 at λ_c=4, with particularly visible losses on

MAE-PO and the *concordance* metrics. Excessive emphasis on censoring prediction can therefore divert model capacity and optimization progress away from the primary event-time objective.

The metric-wise results also show that no single value of λ_c dominates every individual metric. Some metrics mildly favor a neighboring value, but these localized gains do not translate into a better aggregate estimator. The default $\lambda_c{=}1$ instead provides the strongest overall balance across predictive *accuracy*, *concordance*, and *calibration*. Together, the results justify equal weighting as a robust default: positive censoring supervision is beneficial, while allowing the auxiliary channel to dominate the objective is detrimental.

Case Study IV: Robustness Across *Context Sizes*

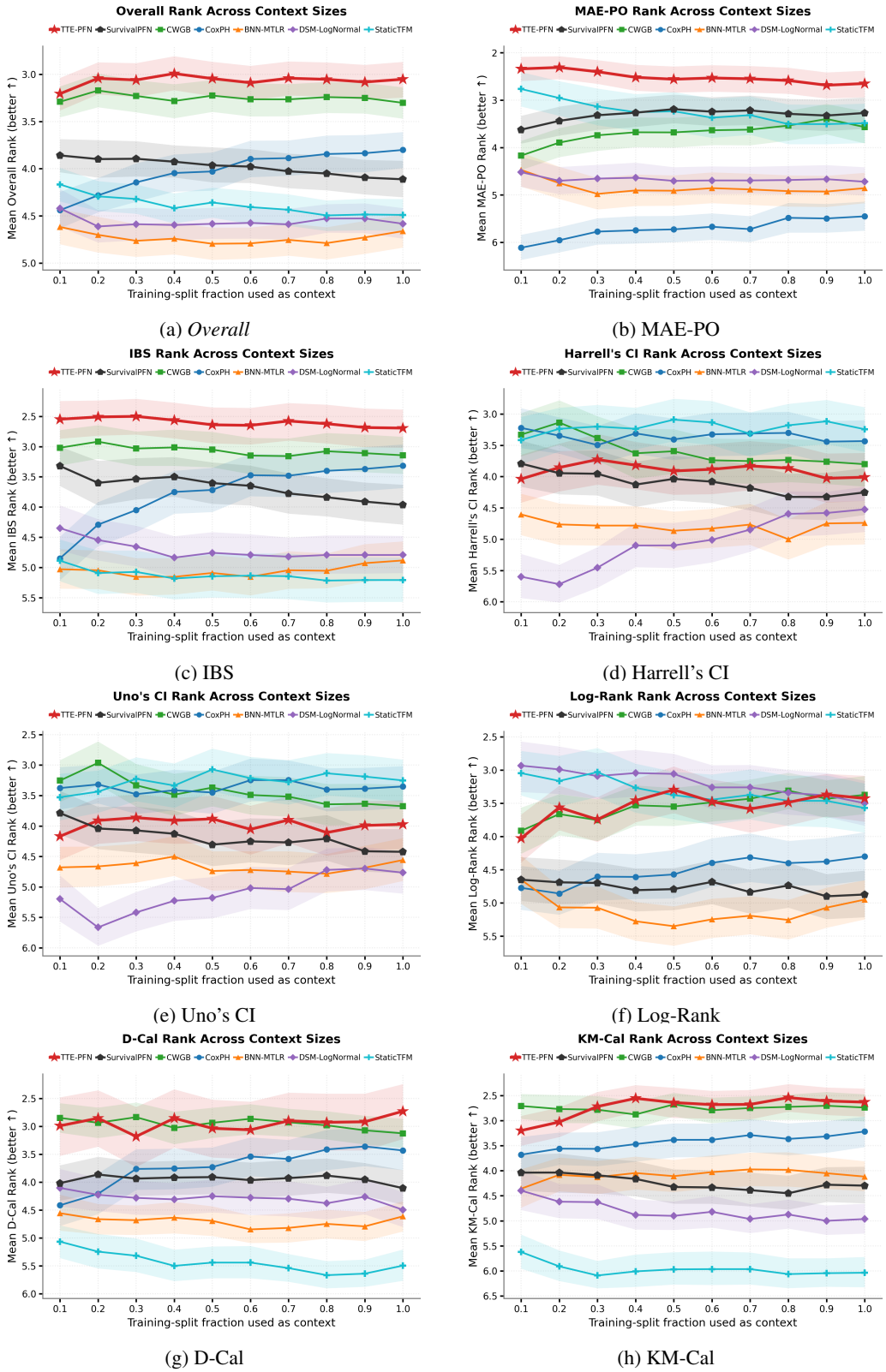
The main text summarizes robustness across context sizes using the mean *Overall* rank. Here, we provide the corresponding rank profiles for all seven evaluation metrics and examine the raw metric values on an individual dataset. At each context fraction, ranks are computed separately within each dataset among the same seven methods considered in the main-text analysis. They should therefore be interpreted comparatively within this subset and are not directly comparable to the 29-method ranks in Table 6.

Figure 10 shows that the aggregate robustness of TTE-PFN is supported primarily by predictive *accuracy* and *calibration*. TTE-PFN achieves the best MAE-PO and IBS ranks at every evaluated context fraction and remains among the leading methods on D-Cal and KM-Cal. Its *concordance* ranks are less dominant, with StaticTFM, CoxPH, CWGB, and DSM-LogNormal remaining competitive on individual *concordance* metrics. These specialized gains, however, do not extend consistently across *accuracy* and *calibration*, leaving TTE-PFN with the best *Overall* rank throughout. SurvivalPFN ranks behind TTE-PFN at every context fraction, with particularly clear gaps in predictive *accuracy* and *calibration*.

To complement these aggregate ranks, Figure 11 reports the raw metric values on the `epileptic` dataset under the same *context-size* intervention. This example illustrates how the cross-dataset rank patterns translate into metric-level behavior on a single cohort.

On `epileptic`, TTE-PFN achieves the lowest IBS throughout the evaluated range and remains best or near-best on MAE-PO. Its D-Cal, KM-Cal, and Log-Rank errors also remain close to zero across *context* fractions, indicating that its distributional predictions remain well *calibrated* even with limited *context*. For Harrell's and Uno's C-indices, TTE-PFN becomes more competitive as the *context* grows, although StaticTFM attains higher *concordance* at several fractions. The dataset-level results therefore mirror the aggregate analysis: specialized methods may lead on individual concordance criteria, whereas TTE-PFN provides the most consistent balance across *predictive accuracy*, *concordance*, and *calibration*.

Together, Figures 10 and 11 illustrate the two effects identified by the theory. Additional *context* reduces the *posterior* uncertainty represented by **term (III)** in **Theorem 1**, while mixed-size pretraining prevents an additional *context-size* mismatch, as characterized by **Corollary 1**.



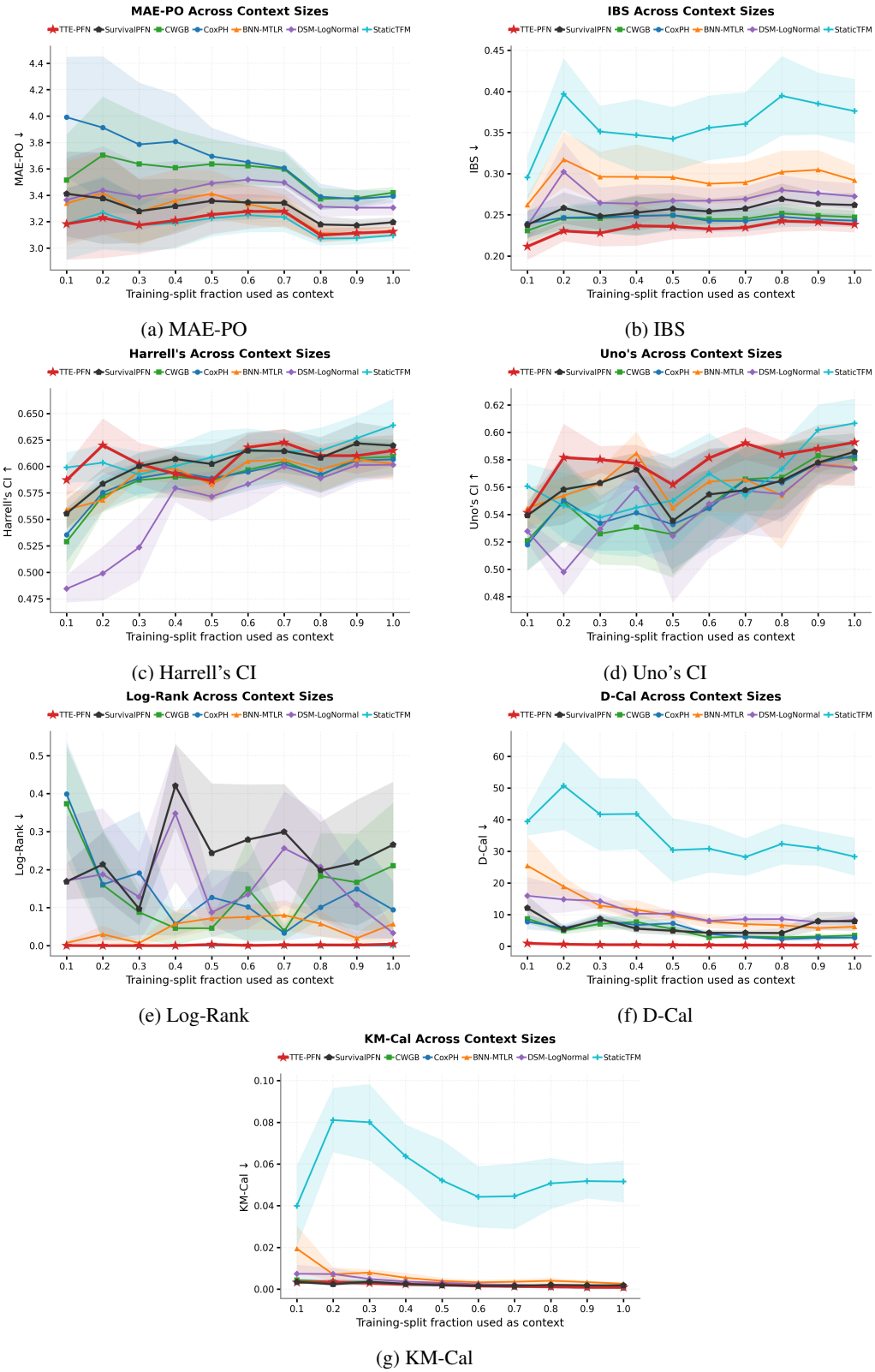
(a) *Overall* (b) MAE-PO

(c) IBS (d) Harrell's CI

(e) Uno's CI (f) Log-Rank

(g) D-Cal (h) KM-Cal

Figure 10: Metric-wise rank profiles as the available labeled data varies from 10% to 100% of the original training split. At each fraction, ranks are computed within each dataset among TTE-PFN, SurvivalPFN, and the five representative baselines, and then averaged across datasets. Panel (a) reports the *Overall* rank, while panels (b)–(h) report the seven constituent metrics. Lower is better, and shaded regions denote 95% confidence intervals.



(a) MAE-PO (b) IBS

(c) Harrell's CI (d) Uno's CI

(e) Log-Rank (f) D-Cal

(g) KM-Cal

Figure 11: Metric-level performance on the `epileptic` dataset as the available labeled data varies from 10% to 100% of the original training split. The subsampled observations are used as context for context-based models and as training data for methods requiring dataset-specific fitting. Higher values are better for Harrell's and Uno's C-indices, whereas lower values are better for MAE-PO, IBS, Log-Rank, D-Cal, and KM-Cal. Lines and shaded regions summarize performance across the five data splits.

Case Study V: Complete Computational-Efficiency Results.

Figure 5 summarizes the performance–efficiency trade-off, while Table 7 reports the complete results for all 29 methods. Because the held-out set size varies across datasets, we divide the total computational time measured for each dataset and split by the number of held-out individuals and scale the resulting per-individual cost to $1{,}000$ individuals. We then report the mean and standard deviation of these standardized computational costs across all 115 datasets and five repeated splits. The reported values therefore represent benchmark-wide computational costs per $1{,}000$ held-out individuals rather than runtimes measured on test sets containing exactly $1{,}000$ individuals. All methods are evaluated under the same allocation of one NVIDIA H200 GPU and one CPU. For pretrained methods, the timed region consists only of the forward pass because no dataset-specific fitting is required. For fitted methods, it includes preprocessing, model fitting, and prediction. The standard deviation captures variation across both datasets and repeated splits, including differences in sample size, dimensionality, and the dataset-dependent cost of model fitting.

Table 7: **Complete computational-efficiency results underlying Figure 5.** Time denotes the mean $\pm$ standard deviation of the computational cost per $1{,}000$ held-out individuals over 115 datasets and five repeated splits. The mean *Overall* rank is taken from Table 5. *Cost* indicates whether the timed region contains inference alone or preprocessing, fitting, and inference. Lower values are better in both numerical columns, and the best value in each is **highlighted**.

Method	Time (s)	Overall rank	Cost
TTE-PFN	**0.052** $\pm$ 0.032	**8.09**	Inference
CoxPH	0.081 $\pm$ 0.059	9.62	Fit + Inference
CoxNet	0.143 $\pm$ 0.121	9.79	Fit + Inference
cSVR	0.161 $\pm$ 0.184	21.34	Fit + Inference
SurvivalPFN	0.234 $\pm$ 0.004	10.76	Inference
DeepAFT-Weibull	0.290 $\pm$ 0.062	18.63	Fit + Inference
DeepAFT-Loglogistic	0.301 $\pm$ 0.065	19.60	Fit + Inference
TabICL	0.418 $\pm$ 0.053	16.10	Inference
TabPFN	0.534 $\pm$ 0.100	16.25	Inference
SurvivalMDN	0.565 $\pm$ 0.860	19.25	Fit + Inference
DSM-Weibull	0.632 $\pm$ 0.425	12.02	Fit + Inference
CQRNN	0.665 $\pm$ 0.228	16.43	Fit + Inference
DSM-LogNormal	0.740 $\pm$ 0.654	11.73	Fit + Inference
RSF	0.752 $\pm$ 1.268	9.37	Fit + Inference
TabDPT	0.760 $\pm$ 0.240	18.42	Inference
SuMoNet	1.040 $\pm$ 0.207	16.49	Fit + Inference
CWGB	1.276 $\pm$ 1.942	8.47	Fit + Inference
GB	1.362 $\pm$ 2.015	10.08	Fit + Inference
StaticTFM	1.670 $\pm$ 0.506	11.62	Inference
Nnet-survival	1.764 $\pm$ 2.432	15.76	Fit + Inference
CoxTime	1.917 $\pm$ 2.542	26.74	Fit + Inference
DeepSurv	1.939 $\pm$ 2.636	15.11	Fit + Inference
MTLR	2.183 $\pm$ 2.968	15.69	Fit + Inference
BNN-MTLR	2.282 $\pm$ 0.431	11.81	Fit + Inference
DeepHit	2.787 $\pm$ 3.771	14.68	Fit + Inference
LogNorm	3.524 $\pm$ 3.466	16.16	Fit + Inference
MITRA	3.598 $\pm$ 0.193	18.96	Inference
ALD	6.269 $\pm$ 6.365	16.28	Fit + Inference
ICALD	9.229 $\pm$ 8.864	14.72	Fit + Inference

The complete results reveal several patterns beyond the aggregate comparison in Figure 5. TTE-PFN is the only method that achieves the best value in both normalized computational cost and mean *overall* rank. Its average cost of 0.052 seconds is approximately $4.5\times$ lower than that of SurvivalPFN. Other pretrained models are also free from dataset-specific fitting, but remain substantially slower. For example, TabICL, TabPFN, TabDPT, StaticTFM, and MITRA require approximately $8.0\times$, $10.3\times$, $14.6\times$, $32.1\times$, and $69.2\times$ the cost of TTE-PFN, respectively.

Among fitted baselines, CoxPH and CoxNet attain low computational costs but weaker mean *Overall* ranks. In contrast, CWGB approaches TTE-PFN in aggregate predictive performance but requires approximately $24.5\times$ more computation. Several fitted neural and ensemble methods also

exhibit large standard deviations, indicating that their computational cost varies substantially across datasets and splits. Overall, the complete results show that TTE-PFN combines strong predictive performance with both low and comparatively stable computational cost across the benchmark.

Case Study VI: High-dimensional Inference Under a Sixteen-feature Budget.

This case study examines the *high-dimensional* limitation of TTE-PFN and evaluates whether an inference-time feature-reduction strategy can mitigate it without modifying or retraining the pre-trained model. TTE-PFN accepts at most sixteen covariates, whereas 37 of the 77 SurvSet cohorts exceed this width. Ten of these cohorts contain at least 500 covariates, all of which are gene-expression datasets, with dimensions ranging from 549 to 15,541. We therefore compare three ways of mapping a wider cohort into the sixteen feature slots available to TTE-PFN.

The supervised variant computes a univariate Harrell's concordance score C_j for each covariate, using the covariate itself as the risk score on the labeled training split. For cohorts with more than 2,000 training observations, the score is computed on a fixed subsample of 2,000 rows. The covariates are ranked by

$$\left| C_j - \frac{1}{2} \right|,$$

which measures departure from chance concordance without favoring either direction of association. The sixteen highest-scoring covariates are retained in their original column order. No *query* outcome is used during this selection. We compare this training-supervised screen with two unsupervised alternatives: projection onto the first sixteen principal components (Jolliffe & Cadima, 2016), with PCA fitted separately on each training split, and positional truncation to the first sixteen columns. For cohorts with at most sixteen covariates, no reduction is applied and all three TTE-PFN variants receive identical inputs.

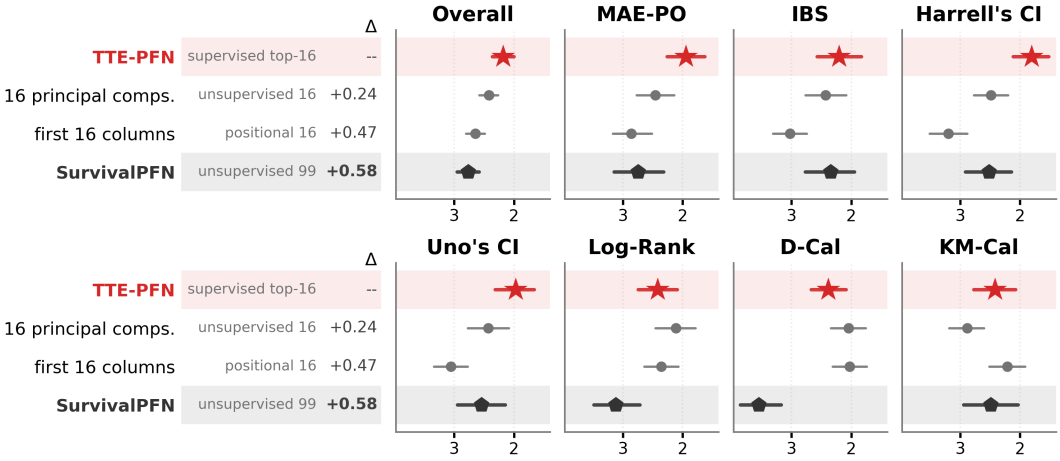


Figure 12: Feature reduction for high-dimensional survival cohorts. The first three rows evaluate the same TTE-PFN checkpoint using supervised selection of sixteen covariates, sixteen principal components, or positional truncation to the first sixteen columns. SurvivalPFN is evaluated using the authors' official implementation, which supports up to 100 input slots and reduces wider cohorts to at most 99 covariate dimensions; when the labeled training set is too small for a 99-component truncated SVD, the implementation retains the first $n_{\text{train}} - 1$ columns instead. Points report mean dataset-level ranks over the 35 wide cohorts for which every evaluation metric is defined, after averaging five random 70/30 splits. Horizontal bars denote 95% bootstrap confidence intervals over datasets. The leftmost annotation reports the increase Δ in *Overall* mean rank relative to supervised feature selection. Lower is better. TTE-PFN and SurvivalPFN are highlighted in red and black, respectively.

Figure 12 shows that supervised feature selection achieves the best aggregate performance among the four configurations. Its *Overall* mean rank is 2.18, compared with 2.42 for PCA-16, 2.65 for positional truncation, and 2.76 for SurvivalPFN. The corresponding rank advantages are 0.24, 0.47, and 0.58, respectively. These ranks are computed only within this four-configuration comparison and are therefore not directly comparable with the 29-method ranks in Table 6.

The same pattern appears in the raw Harrell's C-index values across all 37 wide cohorts. Relative to positional truncation, supervised selection improves Harrell's C-index by 0.038, with a 95% dataset-level bootstrap confidence interval of $[0.024, 0.054]$; a two-sided paired Wilcoxon signed-rank test (Wilcoxon, 1945) gives $p = 1.56 \times 10^{-5}$. The corresponding improvement over PCA-16 is 0.012, with interval $[0.002, 0.023]$ and $p = 0.0116$. Compared with the official SurvivalPFN pipeline, the improvement is 0.028, with interval $[0.009, 0.048]$ and $p = 0.0133$. For each comparison, the five splits are first averaged within a dataset, after which paired differences are formed and datasets are resampled for the confidence interval.

The gain is larger on the ten gene-expression cohorts, where the width constraint is most severe. Supervised selection improves Harrell's C-index over positional truncation by 0.063, with a 95% confidence interval of $[0.029, 0.097]$ and $p = 0.0137$, raising the mean C-index from 0.606 to 0.669. By contrast, the three TTE-PFN input strategies are exactly identical on the 40 cohorts with at most sixteen covariates because no reduction is activated. The improvement is therefore confined by construction to cohorts whose feature dimension exceeds the model's input width.

These results separate the effect of feature allocation from that of model width. At a fixed sixteen-feature budget and with identical pretrained weights, selecting covariates using survival information improves Harrell's C-index by 0.038 relative to selecting columns solely by their original position. Thus, part of the high-dimensional performance gap arises from which features occupy the available input slots, rather than from the number of slots alone. The reduction strategy can be fitted at inference time and does not require any update to the TTE-PFN parameters.

One limitation is that the screening criterion is univariate. Although it improves nine of the ten gene-expression cohorts, `gse1992` is an exception: its mean Harrell's C-index decreases from 0.675 under positional truncation to 0.635 under supervised selection. A possible explanation is that ranking covariates individually can miss features that are weak in isolation but informative through multivariate interactions. However, this interpretation remains conjectural because no joint feature-selection method is evaluated here. Overall, supervised feature selection narrows the high-dimensional gap, but does not eliminate the underlying sixteen-feature constraint. Developing inference-time projections that preserve multivariate interactions remains an important direction for future work.

Case Study VII: Correcting Cohort-level Miscalibration From the *Context*.

Although TTE-PFN performs strongly across the benchmark, its frozen predictions can remain *miscalibrated* on some cohorts. In particular, the predicted survival distributions may preserve subject-level discrimination while systematically placing event times too early or too late for the target cohort. This behavior is natural for a prior-fitted model: TTE-PFN adapts to a new dataset through its labeled *context*, but performs no dataset-specific parameter fitting and does not explicitly estimate a cohort-specific baseline survival distribution. Importantly, however, the same labeled *context* also provides enough information to detect and correct this cohort-level mismatch. We therefore consider a simple post-hoc recalibration procedure that adjusts the predicted time scale without retraining TTE-PFN or using any *query* outcomes.

The key idea is to align the marginal survival distribution predicted on the *context* with the empirical survival distribution observed in that same *context*. Let

$$\overline{S}_{\mathrm{ctx}}(t) = \frac{1}{N_{\mathrm{ctx}}} \sum_{i=1}^{N_{\mathrm{ctx}}} \widehat{S}_i(t)$$

denote the mean predicted survival curve for the labeled *context*, and let $\widehat{S}_{\mathrm{KM,ctx}}$ denote its Kaplan–Meier estimate (Kaplan & Meier, 1958). We construct a monotone time transformation

$$\phi(t) = \widehat{S}^{-1}_{\mathrm{KM,ctx}}\big(\overline{S}_{\mathrm{ctx}}(t)\big)\,, \tag{83}$$

which maps each model time to the corresponding time on the empirical Kaplan–Meier scale. For every *query* subject, the predicted event-time quantiles are then transformed as

$$\widehat{Q}_i^{\,\mathrm{recal}}(\alpha) = \phi\Big(\widehat{Q}_i(\alpha)\Big)\,.$$

The same transformation is applied to all *query* subjects. Thus, recalibration changes the cohort-level time scale while preserving the probability levels produced by the model. The map is estimated entirely from the labeled *context*, so no *query* outcome or gradient-based adaptation is required.

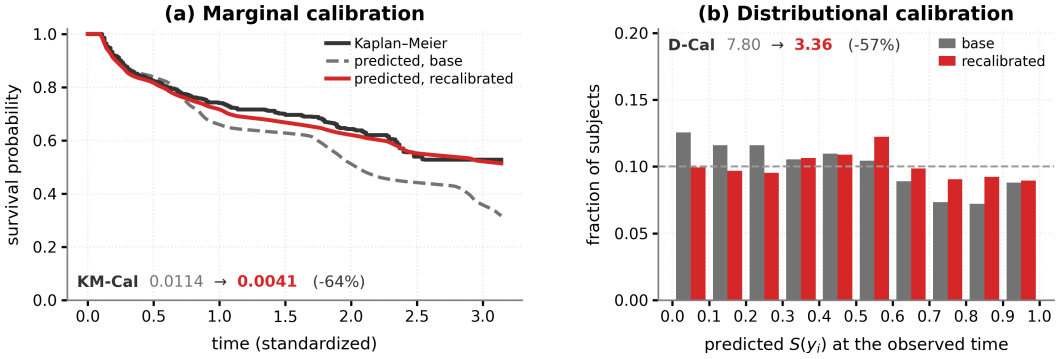


Figure 13: Correcting cohort-level miscalibration on `whas500`. **(a)** Marginal calibration before and after *context*-based recalibration. The black curve is the Kaplan–Meier estimate of the cohort, while the dashed gray and solid red curves are the mean predicted survival curves before and after recalibration. **(b)** Censoring-adjusted D-calibration before and after recalibration. Bars show the fraction of probability mass assigned to ten equal intervals of the predicted survival probability at the observed time; the dashed horizontal line denotes the uniform target. Recalibration reduces KM-Cal from 0.0114 to 0.0041 and D-Cal from 7.80 to 3.36.

The `whas500` cohort provides a representative example. TTE-PFN achieves a Harrell's C-index of approximately 0.774 on this dataset, indicating that it still discriminates well between subjects. Its marginal survival distribution, however, is noticeably miscalibrated. As shown in Figure 13(a), the mean predicted survival curve initially tracks the Kaplan–Meier estimate but subsequently falls below it, indicating that TTE-PFN systematically predicts event times that are too early for this cohort. The context-based time warp substantially reduces this discrepancy: KM-Cal decreases from 0.0114 to 0.0041, corresponding to a 64% reduction.

The correction also improves a distinct notion of calibration. Figure 13(b) shows the censoring-adjusted D-calibration distribution. Before recalibration, the probability mass deviates substantially

from the uniform target, whereas the recalibrated predictions are considerably closer to uniform. Accordingly, D-Cal decreases from 7.80 to 3.36, a 57% reduction, even though D-Cal is not used to fit the transformation. At the same time, Harrell's C-index remains essentially unchanged (0.7739 versus 0.7740), while IBS improves from 0.1636 to 0.1559. MAE-PO worsens by 0.017, indicating that correcting the cohort-level distribution can introduce a modest trade-off in point-prediction error. Overall, this example shows that TTE-PFN's discrimination and calibration can fail separately: a cohort may be ranked well but still lie on the wrong event-time scale, and the latter can be corrected directly from the labeled *context*.

The two calibration diagnostics capture complementary aspects of this mismatch. KM-Cal compares the cohort-average predicted survival curve with the Kaplan–Meier estimate over time, whereas D-Cal assesses the distribution of individual predicted survival probabilities at observed event times while accounting for censoring. The effect of context-based recalibration is not specific to `whas500`. Across the 115 evaluated cohorts, recalibration improves KM-Cal in 77 cohorts, reducing its median from 0.0054 to 0.0024. Importantly, the gains are not confined to the optimized calibration criterion: D-Cal improves in 84 cohorts despite not entering the recalibration objective, while IBS improves in 89 cohorts. At the same time, the correction is not uniformly beneficial across metrics: MAE-PO worsens on 77 of the 115 cohorts for which it is defined, with a median increase of 0.028, illustrating that improved distributional calibration need not translate into better point prediction. Together, these results show that a simple transformation estimated solely from the labeled *context* can correct a substantial fraction of cohort-specific calibration error without retraining the pretrained network.

Overall, this case study identifies cohort-level miscalibration as a distinct failure mode of frozen in-context survival prediction, while showing that it can often be mitigated using information already contained in the labeled *context*. Developing smoother or strictly monotone context-based transformations may retain these calibration gains while reducing the degeneracy observed under extreme censoring.