Deep Discrete-Time Survival Analysis with Guaranteed Monotonicity

Anonymous authors
Paper under double-blind review

Abstract

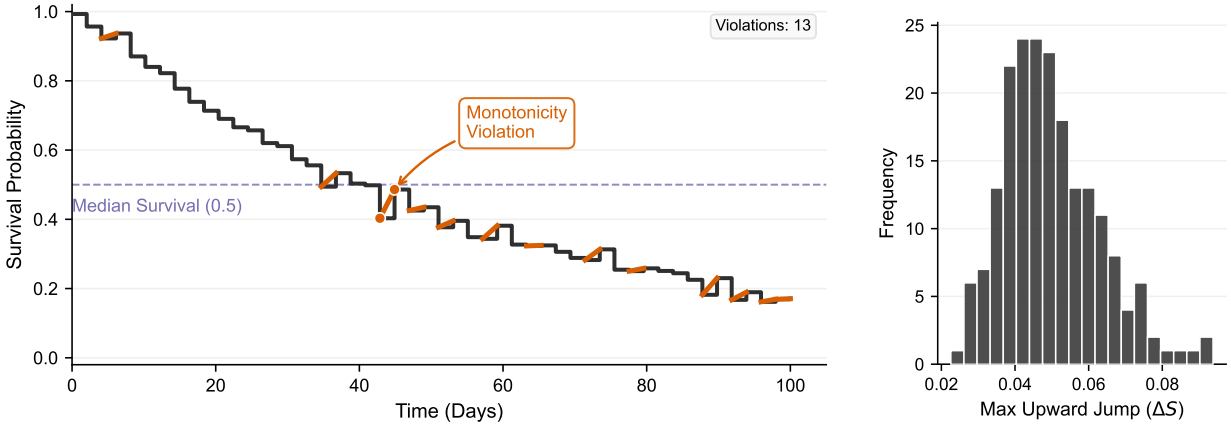
Discrete-time neural survival models trained with binary cross-entropy are attractive due to their simplicity. However, they can produce invalid patient-specific survival curves that increase over time when survival probabilities at different time points are learned without structural constraints. We propose Kaplan–Meier Net (KMNet), a discrete-time neural survival model that predicts interval-wise conditional survival probabilities and constructs the survival curve through a Kaplan–Meier style product, guaranteeing non-increasing survival predictions by design. KMNet is trained with a censoring-aware weighted binary cross-entropy objective and is further augmented with a smooth ranking term that compares individuals using the conditional survival probability at the event interval of the anchor observation, which differs from the global ranking losses used in existing deep survival models. We evaluate KMNet on eight benchmark datasets and compare it with seven strong neural baselines. Across datasets, KMNet achieves the best overall average rank in both time-dependent concordance and integrated brier score, while consistently producing valid survival curves.

1 Introduction

Survival analysis studies when an event happens, such as death, relapse, or re-incarceration, while accounting for the fact that some individuals may leave a study early or the event may not be observed within follow-up (right censoring). In many medical applications, the central goal is patient-specific prognosis. Given a patient's covariates x, we would like to estimate a survival curve $S(t \mid x)$, the probability of remaining event-free beyond time t. Such individualized survival curves support clinical decision-making by summarizing risk over clinically meaningful horizons.

In recent years, deep learning models have been increasingly used for time-to-event prediction when flexible function approximation is desired. A common practical choice is discrete-time modeling, where follow-up is represented on a fixed time grid, and the learning task becomes predicting survival behaviour one interval at a time. This setup integrates naturally with neural networks and has motivated a wide range of deep survival models Katzman et al. (2018); Lee et al. (2018); Gensheimer & Narasimhan (2019); Fotso (2018); Yu et al. (2011). However, a fundamental requirement of any survival curve is that it should never increase over time. As time passes, the probability of having survived beyond that time can only stay the same or decrease. In widely used binary cross-entropy (BCE) based discrete-time models such as BCESurv Kvamme & Borgan (2019), survival probabilities at different grid points are often learned without an explicit cross-time constraint. As a result, the predicted curve can violate monotonicity, producing $\widehat{S}(t + \Delta \mid x) > \widehat{S}(t \mid x)$ for some $\Delta \geq 0$. This behaviour is incompatible with the definition of a survival function and can occur even when ranking-based metrics remain strong.

This behaviour of BCESurv is illustrated in Figure 1. In Subfigure 1a, we show the predicted survival curve for a representative patient from the METABRIC study, which includes genomic and transcriptomic profiles from approximately 2,500 primary breast cancers, revealing 10 molecular subtypes with associated clinical outcomes Curtis et al. (2012). The curve



(a) Example BCESurv prediction for a single patient. The predicted survival curve should be non-increasing, but BCESurv may produce upward steps (highlighted), violating monotonicity and potentially creating ambiguous crossings of the 0.5 level used to define the median survival time.

(b) Distribution of the maximum upward jump $\Delta S = \max_k \max\big(0, \hat{S}(t_{k+1}) - \hat{S}(t_k)\big)$ across patients, quantifying the severity of monotonicity violations.

Figure 1: Motivation for KMNet: BCE-based discrete-time survival models that directly predict survival at each time point can yield non-monotone survival curves, which contradicts the interpretation of survival probability and complicates downstream summaries such as the median survival time.

is not monotone and therefore violates a fundamental requirement of survival functions. Beyond being theoretically invalid, such non-monotonicity has direct practical consequences for interpretation and the generation of downstream clinical summaries. For instance, the median survival time is defined as the earliest time at which the survival probability falls below 0.5. When the predicted curve oscillates, the level 0.5 may be crossed multiple times (as in this example) or not crossed at all, and the resulting estimate becomes ambiguous or highly sensitive to small perturbations, unless additional ad hoc post-processing is applied. Subfigure 1b quantifies the magnitude of these monotonicity violations by reporting the empirical distribution of upward jump heights, highlighting that a non-negligible fraction of predictions exhibit substantial increases over time.

We address these limitations by introducing the Kaplan-Meier Net (KMNet). This discrete-time neural survival model predicts per-interval conditional survival probabilities on a user-defined time grid and constructs individual survival functions via a Kaplan-Meier-style product. This construction enforces the defining property of a survival function, namely monotone non-increasing over time, without sacrificing the representational flexibility of neural networks. To further enhance risk stratification, we incorporate a smooth ranking objective that operates on the exact conditional probabilities used in the product construction, rather than on global risk scores or cumulative quantities as in many existing deep survival models. The contributions of this work are as follows:,

1. We propose KMNet, a discrete-time neural survival model that produces valid patient-specific survival curves by construction through a product of learned conditional survival probabilities across discrete time intervals.
2. We conduct an extensive empirical evaluation of both discrimination and calibration across eight benchmark survival datasets, comparing KMNet against seven established neural baselines. Hyperparameters are selected using Bayesian optimization on a validation split, and performance is reported on a held-out test set. We further assess statistical significance using standard post hoc testing procedures.
3. **We provide an open-source Python implementation of KMNet with a streamlined interface to support reproducibility and facilitate adoption by the research community. To preserve double-blind review, the PyPI link is**

withheld during review; however, the package materials are included in the supplementary material to support reviewers' evaluation.

4. We provide a simulation study that isolates the role of key hyperparameters and illustrates how they affect the calibration–discrimination trade-off and risk stratification behaviour.
5. We include ablation experiments that vary the loss components to quantify the contribution of each design choice in the proposed objective.

The remainder of the paper is organized as follows. Section 2 reviews patient-specific survival curves, the Kaplan–Meier estimator, and discrete-time survival models trained with binary cross-entropy. Section 3 then introduces KMNet and its training objective. Section 4 presents the real-data evaluation, including dataset details, baseline methods, the experimental protocol, and comparative results. Statistical significance analyses are reported in Section 5. Section 6 provides a controlled simulation study to examine the behavior of key hyperparameters. Finally, Section 7 concludes with a discussion and directions for future work.

Statement of Significance. A fundamental challenge in discrete-time neural survival analysis is that widely used binary cross-entropy-based models can produce non-monotone patient-specific survival curves, which are theoretically invalid and can undermine clinical interpretation. Existing deep survival methods often achieve competitive discrimination, but they do not always guarantee survival predictions that remain non-increasing over time. This paper introduces Kaplan–Meier Net (KMNet), a discrete-time neural survival model that predicts interval-wise conditional survival probabilities and constructs survival curves via a Kaplan–Meier-style product, thereby guaranteeing monotonicity by design. Across eight benchmark datasets, KMNet demonstrates strong discrimination and calibration, consistently producing valid survival curves. This work will benefit researchers developing survival models and clinicians who rely on interpretable, patient-specific survival predictions for prognosis and decision-making.

2 Background

In this section, we first introduce patient-specific survival functions and related notation. We then review the classical Kaplan–Meier estimator, which provides a population-level survival curve and does not incorporate patient covariates. Finally, we discuss discrete-time neural survival models trained with binary cross-entropy objectives and highlight the challenges that arise when translating their outputs into valid survival functions.

2.1 Patient-specific survival probability

Let $X \in \mathbb{R}^d$ denote a random covariate vector and let $T \geq 0$ denote the actual event time. In a typical time-to-event In studies, event times may be subject to right censoring. Let $C \geq 0$ denote the censoring time and define the observed time and event indicator as

$$Y = \min(T, C), \qquad \delta = 1\{T \leq C\}.$$

Hence, $\delta = 1$ indicates that the event is observed at time Y, whereas $\delta = 0$ indicates that the observation is censored at time Y and the event time is only known to satisfy $T > Y$. For a covariate value x, the patient-specific survival function[1] is defined as

$$S(t \mid x) = \Pr(T > t \mid X = x), \qquad t \geq 0.$$

[1]Terminology varies in the literature: some authors use the term conditional survival to mean conditioning on covariates, i.e., $S(t \mid x)$. In this work, we reserve conditional survival probability for the discrete-time setting, where survival is conditioned on having survived up to the previous grid point (e.g., $\Pr(T > t_j \mid T > t_{j-1}, X = x)$). Accordingly, we use patient-specific survival probability for covariate-conditioned survival $S(t \mid x)$, patient-specific conditional survival probability when conditioning on both covariates and prior survival, and omit "patient-specific" when covariates are not conditioned on.

The function $S(\cdot \mid x)$ gives the probability that a patient with covariates x survives beyond time t. It satisfies $S(0 \mid x) = 1$, takes values in $[0, 1]$, and is non-increasing in t.

2.2 Kaplan–Meier survival probability

Kaplan-Meier survival probability estimation is a nonparametric method. Consider a cohort of N individuals with observed time and event indicator (Y_i, δ_i) for $i = 1, \ldots, N$. The classical Kaplan–Meier estimator does not use covariates and targets the marginal survival function $S(t) = \Pr(T > t)$ under right censoring. Let $\tau_1 < \tau_2 < \cdots < \tau_K$ denote the distinct observed event times in the sample, i.e., the distinct values among $\{Y_i : \delta_i = 1\}$. For each event time τ_k, define the risk set size and the number of events as

$$n_k = \sum_{i=1}^{N} 1\{Y_i \geq \tau_k\}, \qquad d_k = \sum_{i=1}^{N} 1\{Y_i = \tau_k,\ \delta_i = 1\}.$$

Here, n_k counts the individuals still at risk at time τ_k, and d_k counts the events occurring at τ_k. The The Kaplan–Meier estimator is defined as a product over event times,

$$\widehat{S}_{KM}(t) = \prod_{k:\ \tau_k \leq t} \left(1 - \frac{d_k}{n_k}\right), \qquad t \geq 0.$$

Each factor $\left(1 - \frac{d_k}{n_k}\right)$ lies in $[0, 1]$, so $\widehat{S}_{KM}(t)$ is non-increasing in t. Intuitively, $\frac{d_k}{n_k}$ estimates the conditional probability of experiencing the event at time τ_k among those at risk at τ_k, Moreover, the Kaplan–Meier estimator combines these conditional quantities multiplicatively to obtain a survival curve.

In discrete-time settings, the same idea can be expressed on a fixed grid $0 = t_0 < t_1 < \cdots < t_J$ by defining a conditional survival probability at each grid point and then taking a product across time. This product construction is the key mechanism used later to ensure that predicted survival curves remain non-increasing.

2.3 Binary cross-entropy based discrete-time survival models

A common approach to discrete-time survival prediction converts time-to-event data into a sequence of binary classification problems on a fixed, researcher-specified grid $0 = t_0 < t_1 < \cdots < t_J$ over a time interval. Practitioners often use equally spaced grids, but this is not required and the grid may have non-equal spacing. For each individual i and grid point t_j, a binary label is formed, typically $y_{ij} = 1\{Y_i > t_j\}$. A neural network is trained to output marginal survival probabilities $\widehat{S}(t_j \mid X_i) \in (0, 1)$ by minimizing a weighted binary cross-entropy that accounts for right censoring by including only time points at which the individual is still observed and at risk. This family of models is often referred to as BCE-based survival modeling, and `BCESurv` is a representative example that directly learns $\widehat{S}(t_j \mid x)$ at each grid point using a sigmoid output and binary cross-entropy.

While these methods are simple and effective for learning risk scores, they do not enforce the defining structural constraint of a survival function, namely, monotonicity in time. Because $\widehat{S}(t_j \mid x)$ is learned independently across j, it is possible to obtain non-monotone predictions with $\widehat{S}(t_{j+1} \mid x) > \widehat{S}(t_j \mid x)$ for some j, which is not a valid survival curve. This lack of structure also complicates the extraction of clinically meaningful summaries such as the median survival time. In a valid survival model, the median survival time is defined as $\inf\{t : S(t \mid x) \leq 0.5\}$ and can be read off as the first time the survival curve crosses 0.5. When the predicted curve is non-monotone, the crossing may not exist, may occur multiple times, or may be highly sensitive to small perturbations, making the median survival time ill-defined or unstable without additional post-processing such as isotonic regression or ad hoc smoothing. Consequently, BCE-based models can produce probabilities that are difficult to interpret as proper patient-specific survival functions, even when their ranking performance is competitive.

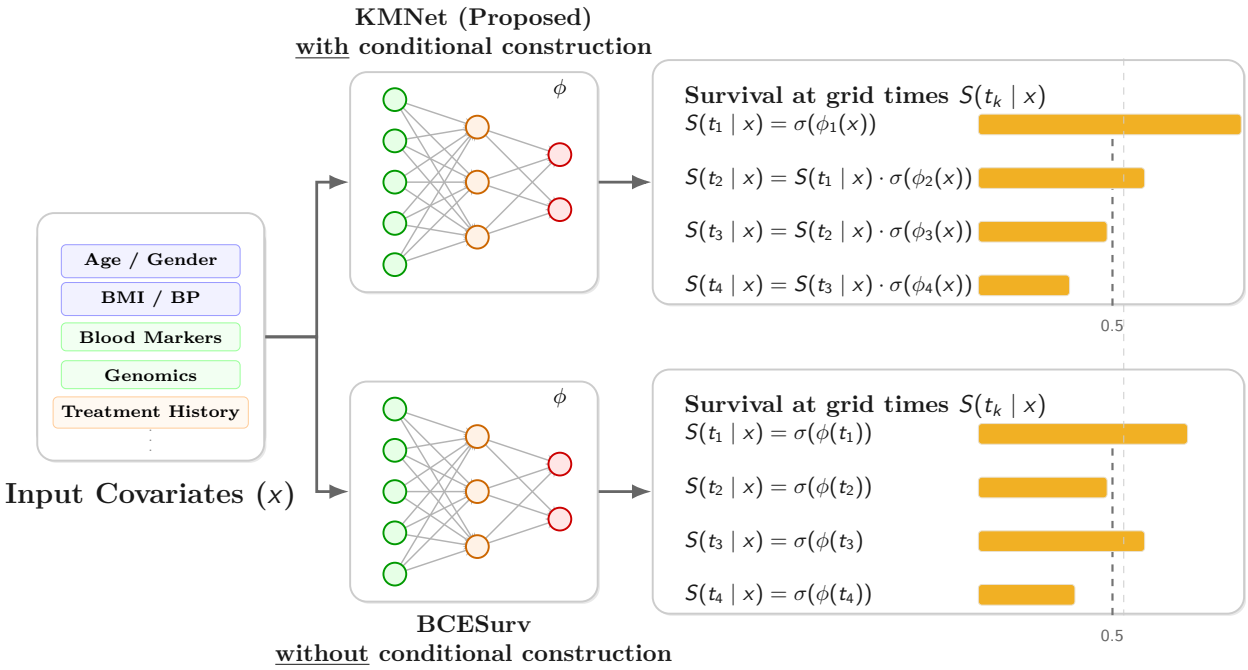


Figure 2: Conceptual comparison between KMNet and BCESurv on a discrete time grid. KMNet predicts interval-wise conditional survival probabilities $\phi_k(x)$ and constructs survival recursively as $S(t_k \mid x) = S(t_{k-1} \mid x) \cdot \sigma(\phi_k(x))$, which guarantees a non-increasing survival curve across grid times. In contrast, BCESurv predicts marginal survival values at each grid time independently as $S(t_k \mid x) = \sigma(\phi(t_k))$, which can lead to inconsistent, non-monotone survival curves.

3 Kaplan–Meier Net

We consider right-censored survival data $\{(X_i, Y_i, \delta_i)\}_{i=1}^N$, where $X_i \in \mathbb{R}^d$ is a covariate vector, Y_i is the observed time, and $\delta_i \in \{0, 1\}$ is the event indicator. We work on a fixed, discrete time grid $0 = t_0 < t_1 < \cdots < t_J$. Intuitively, the grid partitions follow-up into intervals $(t_{j-1}, t_j]$, and the model predicts survival one interval at a time. We aim to estimate a patient-specific survival curve $S(t \mid x)$ on this grid while respecting the non-increasing property of survival in time. Figure 2 demonstrates the conditional construction of KMNet in comparison to BCESurv.

KMNet is a neural network $\phi\colon \mathbb{R}^d \to \mathbb{R}^J$ that outputs logits $\phi(x) = (\phi_1(x), \ldots, \phi_J(x))$, where each component $\phi_j(x)$ corresponds to the interval $(t_{j-1}, t_j]$. The jth output is mapped through a sigmoid function to obtain a patient-specific conditional survival probability

$$p_j(x) = \sigma(\phi_j(x)) \in (0, 1), \qquad j = 1, \ldots, J,$$

which we interpret as the probability of surviving the next interval given survival up to its start,

$$p_j(x) \approx \Pr(T > t_j \mid T > t_{j-1}, X = x).$$

The sigmoid ensures that each $p_j(x)$ is a valid probability in $(0, 1)$.

Given $\{p_j(x)\}_{j=1}^J$, KMNet constructs the survival curve by a Kaplan–Meier style product on the grid,

$$\widehat{S}(t_0 \mid x) = 1, \qquad \widehat{S}(t_j \mid x) = \prod_{k=1}^{j} p_k(x), \quad j = 1, \ldots, J.$$

This construction guarantees a non-increasing survival curve whenever $p_k(x) \in [0, 1]$, which holds by design under the sigmoid output.

To define supervision targets on the grid, each individual is assigned an index $\kappa_i \in \{1, \dots, J\}$ indicating the first grid time greater than or equal to the observed time Y_i,

$$\kappa_i = \min\{j \in \{1, \dots, J\} \mid Y_i \le t_j\}.$$

Thus, κ_i is the discrete-time bin index of subject i. We then define a binary survival indicator at each grid point,

$$s_{ij} = 1\{j < \kappa_i\}, \qquad i = 1, \dots, N, \;\; j = 1, \dots, J,$$

so that $s_{ij} = 1$ indicates subject i is known to have survived beyond t_j, while $s_{ij} = 0$ indicates that t_j lies at or after the subject's observed bin.

Learning must account for the fact that individuals only contribute to the loss while they are at risk and under observation. We introduce an at-risk-at-start indicator

$$r_{ij} = \begin{cases} 1, & j = 1, \\ s_{i,j-1}, & j = 2, \dots, J, \end{cases}$$

so that $r_{ij} = 1$ means subject i is at risk at the start of interval $(t_{j-1}, t_j]$. We further apply an exclusive censoring rule through a counting mask

$$c_{ij} = \delta_i + (1 - \delta_i)\, s_{ij}.$$

For event observations ($\delta_i = 1$), $c_{ij} = 1$ for all j, so the event bin $j = \kappa_i$ is eligible to contribute to the loss (with label $s_{i\kappa_i} = 0$). For censored observations ($\delta_i = 0$), $c_{ij} = s_{ij}$, so all bins $j \ge \kappa_i$ are excluded and the model is not forced to make assertions beyond the censoring bin. The final weight is

$$w_{ij} = r_{ij}\, c_{ij}.$$

Hence, $w_{ij} = 1$ exactly for subject–interval pairs that are both observable and at risk, and $w_{ij} = 0$ otherwise. Note that r_{ij} enforces the sequential at-risk constraint: since $r_{ij} = s_{i,j-1}$ for $j \ge 2$, we have $w_{ij} = 0$ for all $j > \kappa_i$, so no loss is accumulated after the event or censoring bin.

The base loss is a weighted binary cross-entropy applied to conditional survival probabilities. With $p_{ij} = p_j(X_i) = \sigma(\phi_j(X_i))$, we define

$$\mathcal{L}_{\mathrm{BCE}} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{J} w_{ij} \Big[s_{ij} \log(p_{ij}) + (1 - s_{ij}) \log(1 - p_{ij}) \Big].$$

This objective trains the network to estimate per-interval conditional survival while restricting supervision to time points at which an individual remains under observation (and therefore at risk), with censoring handled through the weights w_{ij}. Intuitively, each training sample contributes gradients only for the consecutive intervals that it is known to have survived. Thus, at every discrete time point the loss is effectively computed on the subset of individuals who are still at risk at that time, and no supervision is imposed beyond an individual's event time or censoring time.

To improve risk discrimination, we augment the objective with a smooth ranking term computed at the event interval of each uncensored individual. For each event anchor i with $\delta_i = 1$, define the set of comparable indices

$$\mathcal{C}_i = \Big\{ k \in \{1, \dots, N\} \;\Big|\; \kappa_i < \kappa_k \Big\} \;\cup\; \Big\{ k \in \{1, \dots, N\} \;\Big|\; \kappa_i = \kappa_k,\; \delta_k = 0 \Big\}.$$

The first set contains individuals whose observed bin occurs strictly after the event bin of i, and the second set contains individuals censored in the same bin as i (who are known to have survived up to that bin).

For each event anchor i with $\delta_i = 1$, we compare conditional survival probabilities at the anchor index κ_i via

$$m_{ik} = p_{\kappa_i}(X_k) - p_{\kappa_i}(X_i), \qquad k \in \mathcal{C}_i.$$

The ranking loss uses an exponential penalty with temperature $\tau > 0$,

$$\mathcal{L}_{\mathrm{Rank}} = \frac{1}{N} \sum_{i=1}^{N} 1\{\delta_i = 1\}\, 1\{|\mathcal{C}_i| > 0\}\, \frac{1}{|\mathcal{C}_i|} \sum_{k \in \mathcal{C}_i} \exp\Big(-\frac{m_{ik}}{\tau}\Big).$$

Minimizing $\mathcal{L}_{\mathrm{Rank}}$ encourages $p_{\kappa_i}(X_k)$ to be larger than $p_{\kappa_i}(X_i)$ whenever individual k survives longer than individual i, aligning the ordering of interval-conditional survival with observed event times.

In addition to treating τ as a tunable hyperparameter, we allow an automatic setting that adapts τ to the current scale of the network outputs. For a minibatch of size B, we collect the anchor logits $\{\phi_{\kappa_i}(X_i)\}_{i=1}^{B}$ (where κ_i is the observed-bin index used in the loss) and set

$$\tau = \max\big(10^{-3},\ 0.25 \cdot \mathrm{Std}\big(\{\phi_{\kappa_i}(X_i)\}_{i=1}^{B}\big)\big),$$

computed without gradient tracking. This heuristic scales ranking margins by the empirical variability of the anchor logits, which stabilizes optimization and reduces sensitivity of the ranking term to the absolute logit scale induced by the network architecture and learning dynamics.

This ranking term differs from the ranking losses commonly used in DeepHit and related survival models, which typically rank patients using a global risk score or a cumulative quantity derived from the full survival distribution. In contrast, our ranking is local to the event interval of the anchor subject. For an event observation i, we compare subjects only through the conditional survival probability at κ_i, namely $p_{\kappa_i}(x)$, which corresponds to survival on the interval $(t_{\kappa_i-1}, t_{\kappa_i}]$ given survival up to t_{κ_i-1}. If two subjects have both survived up to t_{κ_i-1}, then their ordering at time κ_i should be determined by how likely they are to survive the next interval, rather than by a global score that aggregates information from other time regions. As a result, the ranking term directly encourages separation in the interval-specific conditional probabilities that define the Kaplan–Meier product, while preserving the interpretation of KMNet outputs as conditional survival probabilities.

The final training objective combines the base loss and the ranking loss using a convex weight and an additional scale factor:

$$\mathcal{L} = \alpha\, \mathcal{L}_{\mathrm{BCE}} + (1 - \alpha)\, \lambda\, \mathcal{L}_{\mathrm{Rank}}, \qquad \alpha \in [0, 1],\ \lambda > 0,$$

where α controls the trade-off between fitting the conditional survival probabilities and improving discrimination, λ scales the ranking term, and τ is the temperature parameter inside the ranking loss. We treat α, τ, and λ as hyperparameters and select them on the validation split. At inference time, KMNet outputs interval-wise conditional survival probabilities $\{p_j(x)\}_{j=1}^{J}$ and constructs the predicted survival curve by the cumulative product

$$\widehat{S}(t_0 \mid x) = 1, \qquad \widehat{S}(t_j \mid x) = \prod_{k=1}^{j} p_k(x), \quad j = 1, \ldots, J.$$

Proposition 1. *Let $\{p_j(x)\}_{j=1}^{J}$ denote the conditional survival probabilities produced by KMNet on the grid $0 = t_0 < t_1 < \cdots < t_J$ for a covariate vector x, with $p_j(x) \in [0, 1]$ for all j. Define $\widehat{S}(t_0 \mid x) = 1$ and*

$$\widehat{S}(t_j \mid x) = \prod_{k=1}^{j} p_k(x), \qquad j = 1, \ldots, J.$$

Then the sequence $\{\widehat{S}(t_j \mid x)\}_{j=0}^{J}$ is non-increasing in j.

Proof. For any $j \in \{0, \ldots, J-1\}$,

$$\widehat{S}(t_{j+1} \mid x) = \widehat{S}(t_j \mid x)\, p_{j+1}(x).$$

Since $0 \leq p_{j+1}(x) \leq 1$, it follows that $\widehat{S}(t_{j+1} \mid x) \leq \widehat{S}(t_j \mid x)$. Therefore $\widehat{S}(t_j \mid x)$ is non-increasing in j. □

Unlike BCE-based models that directly learn marginal survival probabilities at each grid point without enforcing cross-time constraints, KMNet constructs survival curves through a product of interval-wise conditional probabilities, which guarantees valid non-increasing survival predictions by design.

4 Experiments

In this section, we evaluate KMNet against established neural survival baselines on multiple benchmark datasets. We first describe the datasets and preprocessing, then present the experimental protocol and evaluation metrics, and finally report comparative results along with a sensitivity analysis of key hyperparameters. An ablation study is provided in Appendix A

Datasets

We evaluate KMNet on eight widely used right-censored survival benchmarks covering diverse application domains. Support Knaus et al. (1995) is a large clinical cohort of seriously ill hospitalized patients with mortality as the endpoint. Metabric Curtis et al. (2012) and Rotterdam-Gbsg Foekens et al. (2000) are oncology benchmarks that model time-to-event outcomes in breast cancer cohorts. Flchain Dispenzieri et al. (2012) is a clinical cohort related to serum free light chain measurements and models time to death. Nwtco Breslow & Chatterjee (1999) is a pediatric oncology benchmark from the National Wilms Tumor Study and considers time-to-relapse outcomes. We additionally include two datasets from the Northern Alberta Cancer Database: Nacd contains patients across multiple cancer sites, while Nacd-Col is the colorectal cancer subset of the same registry; the two share the same feature definition while differing in disease focus and cohort composition Haider et al. (2020). Finally, Recidivism Rossi et al. (2013) is a non-clinical dataset that models time to re-incarceration after release. Detailed dataset characteristics and summary statistics are reported in Table 1.

Dataset Name	Size (N)	Feature (d)	Censoring (%)	Median/IQR/Range Event Time	Median/IQR/Range Censored Time
Flchain	6524	8	70	4621/734/5165	2084/2343/4998
Metabric	1904	9	42	158/109/337	86/102/355
Nacd	2402	48	36	33/35/84	12/17/81
Nacd-Col	950	48	52	33/34/83	17/21/81
Nwtco	4028	6	86	2323/2580/6204	280/337/4162
Recidivism	1445	14	61	19/64/76	74/6/11
Rotterdam-Gbsg	2232	7	43	75/32/87	24/26/82
Support	8873	14	32	918/917/1685	57/236/1941

Table 1: Details of the datasets used in the experiments. The reported number of features is after preprocessing.

Experimental setup

For each dataset, we created a random split in which 80% of the samples were used for model development and the remaining 20% were held out as an independent test set. From the

development split, we further set aside 10% of the samples as a validation set used only for hyperparameter optimization and model selection. Model performance was evaluated using the time-dependent concordance (Concordance) Heagerty et al. (2000); Antolini et al. (2005) and the integrated brier score (IBS) Graf et al. (1999); Gerds & Schumacher (2006); Kvamme & Borgan (2019), with definitions provided in Appendix B. Hyperparameters were tuned via Bayesian optimization Bergstra et al. (2011), and the full search spaces are reported in Appendix C. After selecting the best hyperparameters, we refit the model using the training portion of the split and report results on the held-out test set. This procedure was repeated across multiple random splits, and we report the mean and standard deviation across runs. For Rotterdam-Gbsg, we follow the standard protocol in which the Rotterdam cohort is used for training and the GBSG cohort is used for testing, and results are reported for this single split.

We compare KMNet with seven established neural survival baselines. BCESurv Kvamme & Borgan (2019) is a discrete-time MLP that learns marginal survival probabilities on a fixed grid using a censoring-aware binary cross-entropy objective. CoxCC Kvamme et al. (2019) is a Cox proportional hazards network trained with a case-control approximation to the partial likelihood, and CoxTime Kvamme et al. (2019) extends this approach by allowing time-dependent effects. DeepHit Lee et al. (2018) is a discrete-time model that learns the event-time distribution and includes a ranking component for discrimination. DeepSurv Katzman et al. (2018) is a neural Cox model that replaces the linear predictor with a deep network while optimizing the Cox partial likelihood. MTLR Yu et al. (2011) models survival through a sequence of logistic regressions across time and outputs a full discrete-time survival distribution. NNet Gensheimer & Narasimhan (2019) models the hazard as piecewise constant over time intervals and is trained through likelihood-based objectives.

4.1 Results

Tables 2 and 3 report the mean and standard deviation of the time-dependent concordance and the integrated Brier score (IBS) across repeated random splits. For each dataset, higher concordance indicates better discrimination, while lower IBS indicates better overall accuracy of the predicted survival probabilities. We also report the average rank across datasets to summarize overall performance.

Discrimination KMNet achieves the best overall discrimination with the lowest average rank of 1.62 in Table 2. It attains the top concordance on Nacd, Nwtco, and Support, and remains competitive on the remaining datasets where the best model varies between DeepSurv, CoxTime, and DeepHit. Compared with BCESurv, KMNet improves concordance on every dataset, with particularly strong gains on Nacd-Col, Metabric, Recidivism, and Support. These results indicate that learning interval-wise conditional survival probabilities together with the proposed conditional ranking term yields consistent improvements in risk stratification across heterogeneous censoring levels and application domains.

Calibration and overall accuracy Table 3 shows that KMNet obtains the best average rank in terms of IBS, with an average rank of 2.38. The strongest competing baseline for IBS is NNet, which achieves the lowest score on several datasets, but KMNet remains consistently close and attains the best IBS on Rotterdam-Gbsg. Relative to BCESurv, KMNet reduces IBS across all datasets, with the largest reduction observed on Nwtco, and smaller but consistent reductions on Flchain, Metabric, Nacd, and Nacd-Col. On Support, the differences in IBS between methods are small, suggesting that the main gains on this dataset arise from improved discrimination rather than large changes in average probability error.

Overall, KMNet provides strong performance across both discrimination and integrated error metrics, while producing valid non-increasing survival curves by construction. The results support the use of interval-wise conditional survival modeling combined with conditional ranking as a practical alternative to BCE-based marginal survival learning.

Dataset	BCESurv	CoxCC	CoxTime	DeepHit	DeepSurv	NNet	MTLR	KMNet
Flchain	0.782 ± 0.009	0.793 ± 0.009	0.793 ± 0.009	0.788 ± 0.009	**0.795 ± 0.007**	0.789 ± 0.015	0.783 ± 0.013	0.793 ± 0.006
Metabric	0.638 ± 0.022	0.650 ± 0.015	**0.665 ± 0.012**	0.660 ± 0.027	0.643 ± 0.012	0.653 ± 0.027	0.661 ± 0.029	0.664 ± 0.007
Nacd	0.745 ± 0.018	0.755 ± 0.021	0.756 ± 0.017	0.753 ± 0.012	0.755 ± 0.015	0.750 ± 0.014	0.759 ± 0.008	**0.760 ± 0.015**
Nacd-Col	0.673 ± 0.035	0.703 ± 0.053	0.692 ± 0.016	**0.722 ± 0.031**	0.697 ± 0.040	0.680 ± 0.056	0.698 ± 0.036	0.704 ± 0.016
Nwtco	0.695 ± 0.043	0.710 ± 0.023	0.709 ± 0.025	0.712 ± 0.018	0.709 ± 0.021	0.704 ± 0.036	0.703 ± 0.023	**0.713 ± 0.024**
Recidivism	0.615 ± 0.031	0.614 ± 0.029	0.623 ± 0.044	0.599 ± 0.024	**0.640 ± 0.021**	0.603 ± 0.025	0.611 ± 0.026	0.637 ± 0.012
Rotterdam-Gbsg	0.677 ± 0.000	0.660 ± 0.000	**0.693 ± 0.000**	0.680 ± 0.000	0.675 ± 0.000	0.652 ± 0.000	0.660 ± 0.000	0.682 ± 0.000
Support	0.619 ± 0.013	0.605 ± 0.007	0.610 ± 0.008	0.636 ± 0.010	0.609 ± 0.009	0.624 ± 0.018	0.623 ± 0.016	**0.637 ± 0.006**
Average Rank	6.62	5.00	3.50	4.00	4.25	6.00	5.00	**1.62**

Table 2: Time-dependent concordance performance across eight benchmark survival datasets. Results are reported as mean ± standard deviation across repeated splits; higher values indicate better discrimination. The best result for each dataset is shown in bold, the second-best is underlined, and the final row reports the average rank across datasets, where lower is better.

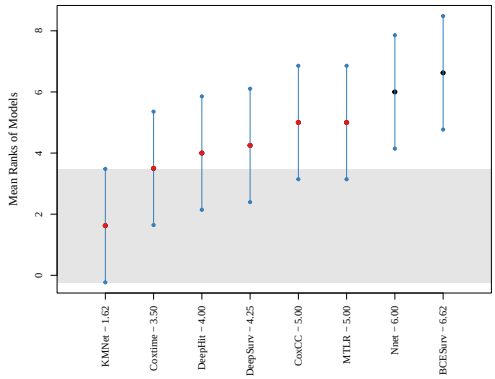
Dataset	BCESurv	CoxCC	CoxTime	DeepHit	DeepSurv	NNet	MTLR	KMNet
Flchain	0.105 ± 0.003	0.101 ± 0.002	0.100 ± 0.003	0.101 ± 0.003	0.101 ± 0.003	**0.098 ± 0.003**	0.100 ± 0.002	0.099 ± 0.002
Metabric	0.176 ± 0.017	0.163 ± 0.007	0.160 ± 0.003	0.161 ± 0.005	0.166 ± 0.015	**0.157 ± 0.001**	0.163 ± 0.004	0.162 ± 0.004
Nacd	0.155 ± 0.006	0.144 ± 0.006	0.144 ± 0.008	0.145 ± 0.006	**0.139 ± 0.004**	0.147 ± 0.003	0.149 ± 0.007	0.142 ± 0.004
Nacd-col	0.187 ± 0.030	0.188 ± 0.013	0.175 ± 0.013	0.173 ± 0.013	0.180 ± 0.009	**0.171 ± 0.012**	0.178 ± 0.024	0.173 ± 0.005
Nwtco	0.126 ± 0.008	0.103 ± 0.007	0.103 ± 0.005	0.102 ± 0.015	0.103 ± 0.005	**0.083 ± 0.005**	0.090 ± 0.005	0.086 ± 0.005
Recidivism	0.183 ± 0.007	0.177 ± 0.009	**0.175 ± 0.004**	0.178 ± 0.008	0.179 ± 0.014	0.179 ± 0.006	0.183 ± 0.008	0.177 ± 0.005
Rotterdam-gbsg	0.179 ± 0.000	0.174 ± 0.000	0.170 ± 0.000	0.175 ± 0.000	0.167 ± 0.000	0.169 ± 0.000	0.167 ± 0.000	**0.165 ± 0.000**
Support	0.192 ± 0.001	0.191 ± 0.001	0.192 ± 0.003	0.191 ± 0.002	0.192 ± 0.003	**0.190 ± 0.001**	0.192 ± 0.001	0.191 ± 0.001
Average Rank	7.75	4.88	4.12	4.50	5.12	2.50	4.75	**2.38**

Table 3: Integrated Brier score performance across eight benchmark survival datasets. Results are reported as mean ± standard deviation across repeated splits; lower values indicate better calibration and prediction accuracy. The best result for each dataset is shown in bold, the second-best is underlined, and the final row reports the average rank across datasets, where lower is better.

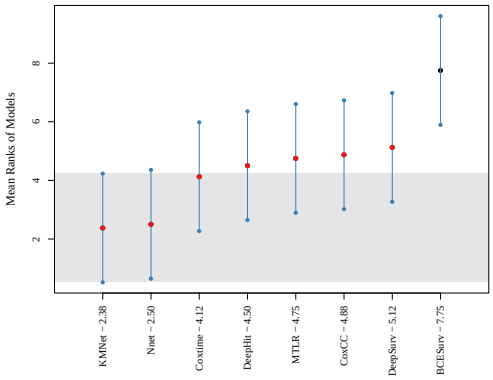
5 Statistical significance

We assess whether the observed differences between methods are statistically significant using the Friedman test followed by a Nemenyi post hoc analysis, visualized through the multiple comparisons with the best (MCB) diagram. In this setting, each algorithm is assigned a rank on every dataset, ranks are averaged across datasets, and the MCB plot reports these mean ranks together with the critical distance (CD) at a prescribed confidence level. At the 5% level, any method whose mean rank differs from the best method by more than the CD is deemed significantly worse than the best under the Nemenyi correction.

Across the eight datasets and eight algorithms, the Friedman omnibus test rejects the null hypothesis of equal performance for both discrimination and calibration, with $p = 0.0021$ for Concordance and $p = 4 \times 10^{-4}$ for IBS. The corresponding Nemenyi critical distance is CD = 3.712. For Concordance, KMNet achieves the best mean rank (1.62); the differences in mean rank between KMNet and BCESurv (6.62), as well as between KMNet and NNet (6.00), exceed the CD, indicating that these two baselines are significantly worse than KMNet in terms of concordance at the 5% level. For IBS, KMNet again attains the best mean rank (2.38); BCESurv has the worst mean rank (7.75) and lies beyond the CD from KMNet, implying significantly worse calibration for BCESurv relative to KMNet. The remaining methods fall within the CD from KMNet and are therefore not significantly different from KMNet under the Nemenyi correction.



(a) MCB plot for Concordance (mean ranks; lower is better).



(b) MCB plot for IBS (mean ranks; lower is better).

Figure 3: Nemenyi post hoc analysis after the Friedman test across all datasets. The horizontal bar indicates the critical distance (CD) at the 5% level; methods whose mean ranks differ by less than the CD are not statistically distinguishable under the Nemenyi correction.

Table 4: Paired Wilcoxon signed-rank tests comparing each baseline to KMNet across datasets. We report two-sided p-values and FDR-adjusted p-values for both discrimination (Concordance) and calibration (IBS).

Method	Concordance		IBS	
	p	p (FDR)	p	p (FDR)
BCESurv	0.0143	0.0250	0.0143	0.0487
CoxCC	0.0143	0.0250	0.0209	0.0487
CoxTime	0.1415	0.1415	0.1415	0.1651
DeepHit	0.1415	0.1415	0.0587	0.0822
DeepSurv	0.0423	0.0592	0.0587	0.0822
MTLR	0.0143	0.0250	0.0143	0.0487
NNet	0.0143	0.0250	1.0000	1.0000

To complement the rank-based Nemenyi analysis, we additionally perform paired Wilcoxon signed-rank tests across datasets, using KMNet as the reference method (Table 4). For Concordance, KMNet shows statistically significant improvements over BCESurv, CoxCC, MTLR, and NNet after FDR correction (all adjusted $p \leq 0.025$), while the differences to CoxTime and DeepHit are not significant (adjusted $p = 0.1415$). The comparison with DeepSurv is significant before adjustment ($p = 0.0423$) but not after FDR correction ($p = 0.0592$), indicating a weaker and less consistent advantage across datasets. For IBS, KMNet significantly outperforms BCESurv, CoxCC, and MTLR after FDR correction (all adjusted $p = 0.0487$), whereas differences to CoxTime, DeepHit, and DeepSurv are not significant at the 5% level. Notably, the IBS comparison against NNet is not significant ($p = 1.0$), suggesting comparable calibration between these two methods under our evaluation protocol.

6 Simulation study

This section presents a controlled simulation study designed to complement the real-data experiments and to provide additional insight into the behaviour of KMNet under known data-generating mechanisms. Across Experiments, we use the same data-generation protocol so that observed differences are attributable to the model configuration or the specific analysis being performed rather than changes in the underlying data.

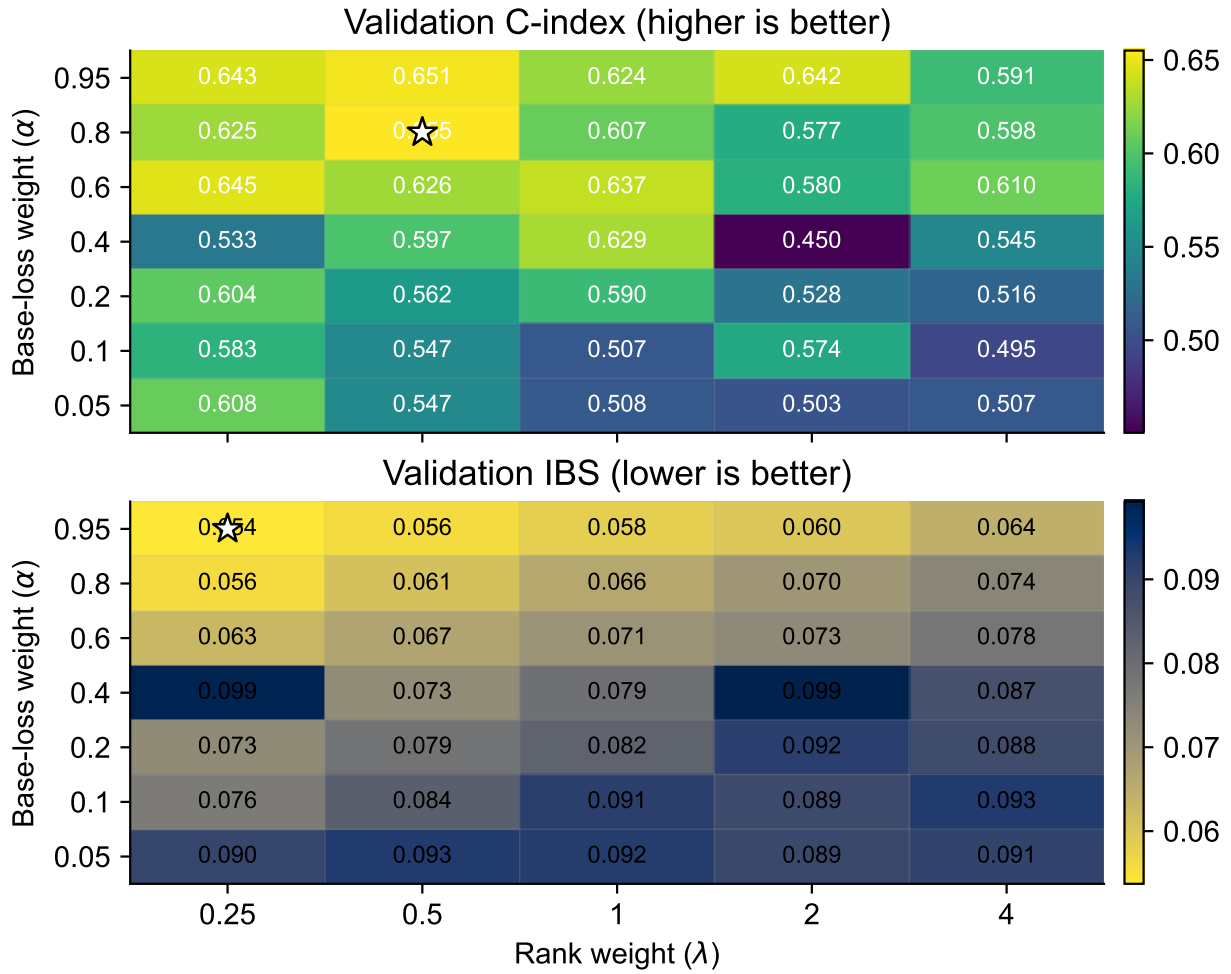


Figure 4: Sensitivity of KMNet to the loss weights on simulated data. Top: validation concordance. Bottom: validation IBS. Stars indicate the best configuration for each metric on this grid.

For each simulated individual $i = 1, \ldots, n$, we sample a feature vector $X_i \in \mathbb{R}^p$ with independent standard normal entries. Event times are generated from a proportional hazards mechanism with a log-linear risk score $\eta_i = X_i^\top \beta$, where β has only a small number of non-zero coefficients. Specifically, the true event time T_i is sampled from an exponential distribution with rate $\lambda_i = \lambda_0 \exp(\eta_i)$, where $\lambda_0 > 0$ is a baseline rate. Censoring times C_i are sampled independently from an exponential distribution, with the rate chosen by bisection to match a prescribed censoring fraction. The observed time and event indicator are then given by

$$Y_i = \min\{T_i, C_i\}, \qquad \delta_i = 1\{T_i \le C_i\}.$$

For each run, we split the simulated data into 80% development and 20% test sets. From the development set, we further hold out 10% as a validation set used for model selection when required. Continuous features are standardized using statistics computed on the training split only. To train discrete-time models, the observed times are discretized into J intervals using the training split, and the same discretization is applied to validation and test data.

Effect of α and λ Figure 4 summarizes a grid-based sensitivity analysis on the synthetic dataset, where we vary the base-loss weight α and the ranking weight λ while keeping the remaining KMNet configuration fixed. For each (α, λ) pair, we train KMNet on the training split and evaluate the time-dependent concordance and the integrated Brier score (IBS)

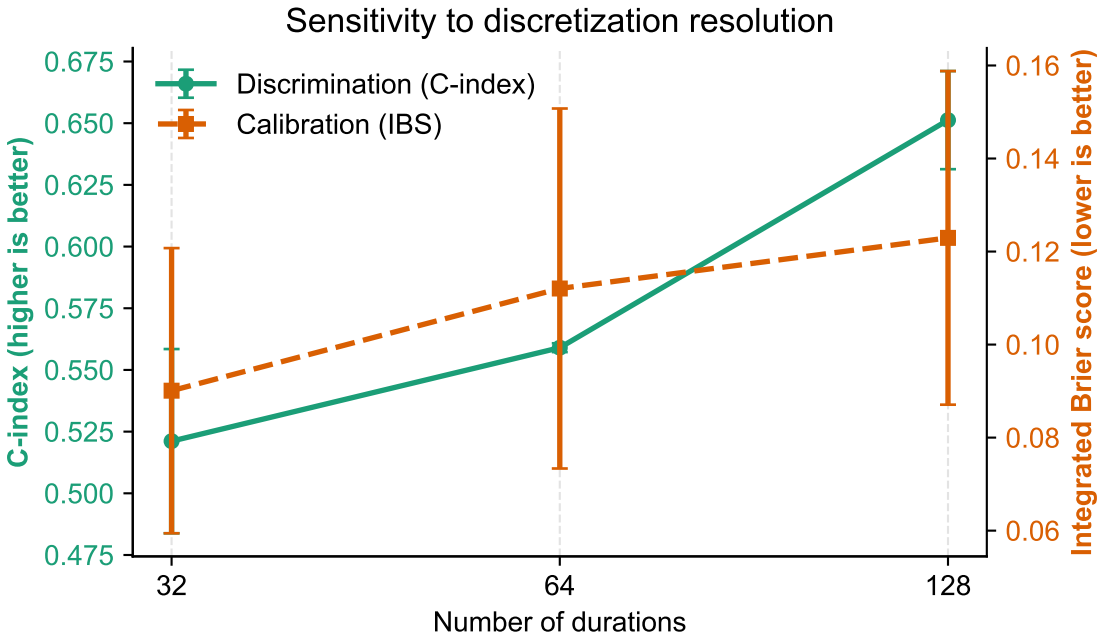


Figure 5: Sensitivity of KMNet to the discretization resolution on simulated data. We vary the number of discrete time intervals $J \in \{32, 64, 128\}$ and report mean $\pm$ standard deviation across random seeds.

on the validation split. The upper heatmap reports the validation concordance (higher is better), and the lower heatmap reports the validation IBS (lower is better).

Two complementary trends emerge. First, the best discrimination is achieved for moderate ranking strength and a non-negligible contribution from the ranking term, with the highest concordance attained at an intermediate λ (here $\lambda = 0.5$) and a large α (here $\alpha = 0.8$). Second, the best calibration is achieved when the objective is dominated by the base BCE term, with the lowest IBS occurring at a large α (here $\alpha = 0.95$) and a smaller ranking weight (here $\lambda = 0.25$). These results illustrate the expected calibration–discrimination trade-off induced by the ranking component. Increasing λ typically improves ordering in settings where individuals are comparable, but can degrade probability calibration when over-emphasized; conversely, larger α stabilizes probability estimation through the base BCE term, improving IBS but potentially limiting gains in concordance. Overall, the heatmaps indicate that KMNet is most effective in the regime of large α with moderate λ, motivating our use of Bayesian optimization to select (α, λ) rather than fixing them a priori.

Sensitivity to the number of intervals J Discrete-time survival models depend on a user-specified discretization of the time axis into J intervals. While a finer discretization can represent more detailed temporal dynamics, it also increases the number of outputs and may amplify optimization noise. In Experiment C, we assess the robustness of KMNet to the discretization resolution by varying the number of intervals $J \in \{32, 64, 128\}$ on the same simulated dataset and under the same training protocol. For each value of J, we discretize observed times using the training split, train KMNet with the proposed configuration, and report the time-dependent concordance and the integrated Brier score (IBS) on the validation split, averaged over multiple random seeds.

Figure 5 shows that KMNet is not overly sensitive to the choice of J in this range. As J increases, concordance tends to improve, while IBS may slightly worsen, indicating a modest discrimination–calibration trade-off. Overall, an intermediate resolution provides a favourable balance, with little additional benefit from increasing J beyond this point, while larger J increases the output dimensionality and training cost. This experiment supports the

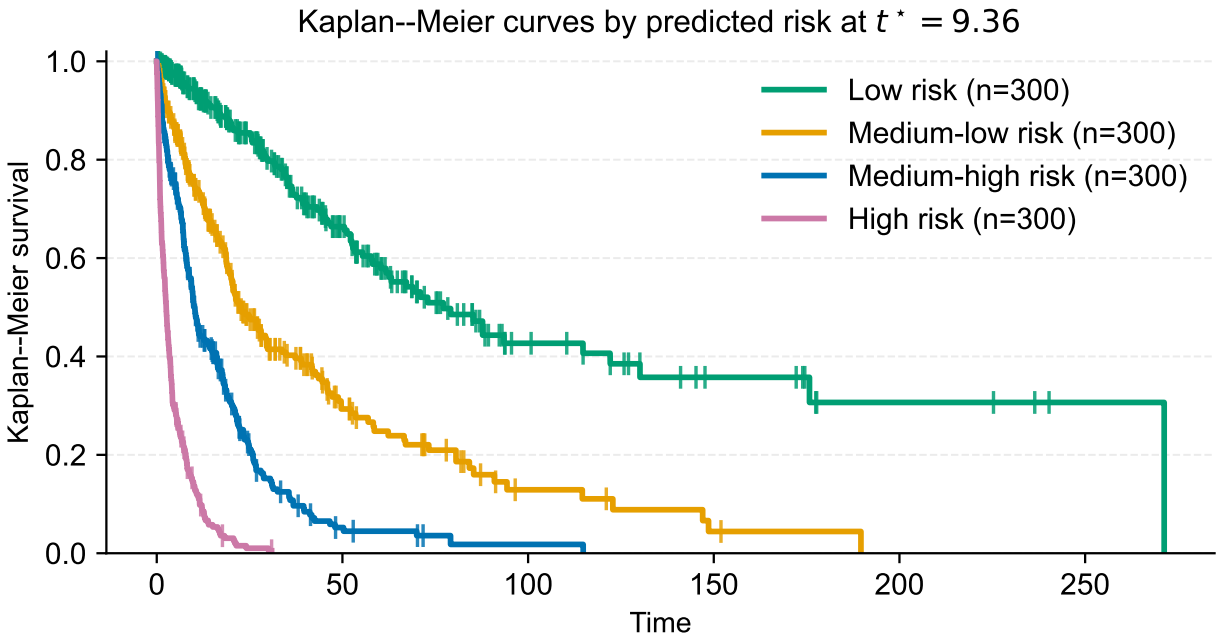


Figure 6: Kaplan–Meier curves for four groups formed by quartiles of the predicted risk $r(x) = -\log \widehat{S}(t^\star \mid x)$ at $t^\star = 9.36$ on simulated data.

use of a mid-range discretization (e.g., $J = 64$) as a robust default that yields competitive performance without requiring careful dataset-specific tuning.

Risk stratification. To assess whether KMNet induces clinically meaningful separation of individuals into risk strata, we summarize each subject by the predicted survival at a reference horizon $t^\star$ and define a scalar risk score $r(x) = -\log \widehat{S}(t^\star \mid x)$. We set $t^\star$ to the empirical median of the observed times, which provides a balanced horizon with a substantial fraction of individuals still at risk. We then partition subjects into four equally sized groups according to risk quartiles and compute the Kaplan–Meier estimate within each group using the observed (Y, δ) pairs. The resulting curves, shown in Figure 6 exhibits clear separation and preserves the expected ordering from low to high risk, with the high-risk group showing the steepest early decline and the low-risk group maintaining the highest survival across follow-up. This stratification plot provides an interpretable complement to concordance and IBS by directly linking model predictions to empirical survival differences between risk groups.

7 Conclusion

We proposed Kaplan–Meier Net, a discrete-time neural survival model that learns interval-wise conditional survival probabilities and constructs patient-specific survival curves through a Kaplan–Meier style product. This design guarantees non-increasing survival predictions by construction while handling right censoring through an at-risk weighted training objective. We further introduced a conditional ranking loss computed at the event interval of the anchor individual, which directly targets discrimination in the conditional probabilities that define the survival curve. Across eight benchmark datasets, KMNet achieved strong performance in both time-dependent concordance and integrated Brier score, with the best overall average rank on both metrics, while always producing valid survival curves.

Several directions remain for future work. First, the current formulation uses a fixed time grid, and performance can depend on the choice of discretization. An adaptive or data-driven time grid, or multi-resolution grids that allocate more bins in high-event-density regions,

may further improve both calibration and discrimination. Second, while our conditional ranking term improves risk stratification, it can be extended to incorporate time-varying comparisons across multiple horizons or to combine local and global ranking signals in a single objective. Third, extending KMNet to competing risks and multi-state settings would broaden its clinical applicability, since many real-world outcomes involve multiple event types. Fourth, further improvements in uncertainty quantification and calibration, such as conformal survival prediction or Bayesian treatments of the conditional probabilities, could provide more reliable decision support. Finally, integrating KMNet with interpretability tools and structured clinical priors, and evaluating it in prospective or external validation studies, are important steps toward deployment in high-stakes medical applications.

References

Laura Antolini, Patrizia Boracchi, and Elia Biganzoli. A time-dependent discrimination index for survival data. Statistics in medicine, 24(24):3927–3944, 2005.

James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. Advances in neural information processing systems, 24, 2011.

Norman E Breslow and Nilanjan Chatterjee. Design and analysis of two-phase studies with binary outcome applied to wilms tumour prognosis. Journal of the Royal Statistical Society: Series C (Applied Statistics), 48(4):457–468, 1999.

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.

Angela Dispenzieri, Jerry A Katzmann, Robert A Kyle, Dirk R Larson, Terry M Therneau, Colin L Colby, Raynell J Clark, Graham P Mead, Shaji Kumar, L Joseph Melton III, et al. Use of nonclonal serum immunoglobulin free light chains to predict overall survival in the general population. In Mayo Clinic Proceedings, volume 87, pp. 517–523. Elsevier, 2012.

John A Foekens, Harry A Peters, Maxime P Look, Henk Portengen, Manfred Schmitt, Michael D Kramer, Nils Brünner, Fritz Jänicke, Marion E Meijer-van Gelder, Sonja C Henzen-Logmans, et al. The urokinase system of plasminogen activation and prognosis in 2780 breast cancer patients. Cancer research, 60(3):636–643, 2000.

Stephane Fotso. Deep neural networks for survival analysis based on a multi-task framework. arXiv preprint arXiv:1801.05512, 2018.

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

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

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.

Patrick J Heagerty, Thomas Lumley, and Margaret S Pepe. Time-dependent roc curves for censored survival data and a diagnostic marker. Biometrics, 56(2):337–344, 2000.

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.

William A Knaus, Frank E Harrell, Joanne Lynn, Lee Goldman, Russell S Phillips, Alfred F Connors, Neal V Dawson, William J Fulkerson, Robert M Califf, Norman Desbiens, et al. The support prognostic model: Objective estimates of survival for seriously ill hospitalized adults. Annals of internal medicine, 122(3):191–203, 1995.

Håvard Kvamme and Ørnulf Borgan. The brier score under administrative censoring: Problems and solutions. arXiv preprint arXiv:1912.08581, 2019.

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.

Peter H Rossi, Richard A Berk, and Kenneth J Lenihan. Money, work, and crime: experimental evidence. Elsevier, 2013.

Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. Advances in neural information processing systems, 25, 2012.

Chun-Nam Yu, Russell Greiner, Hsiu-Chin Lin, and Vickie Baracos. Learning patient-specific cancer survival distributions as a sequence of dependent regressors. Advances in neural information processing systems, 24, 2011.

Appendix

A Ablation Study

In addition to the main configuration of KMNet, we study how individual design choices in the training objective affect discrimination and calibration. Unless stated otherwise, all ablations use the same network architecture, discretization grid, optimizer, early stopping protocol, and evaluation pipeline as in the main experiments. The primary model used in this work sets the base loss to masked BCE on the discrete-time alive indicators, and augments it with a conditional ranking objective computed on per-interval conditional survival probabilities. Concretely, the default configuration is base loss BCE, ranking mode conditional, ranking space probability, and an exponential ranking penalty. We consider the following controlled variations.

1. Conditional ranking compared to global CDF ranking. The conditional variant ranks patients using the per-interval conditional survival probability at the event interval of the anchor individual. The global variant ranks using a cumulative distribution comparison evaluated at the anchor time, which depends on the full product-form survival curve.
2. Exponential compared to softplus ranking penalties. The exponential penalty imposes stronger separation between mis-ordered pairs, while the softplus penalty provides a smoother and numerically stable alternative.
3. Ranking in probability space compared to logit space. In conditional ranking, pairwise margins can be formed either on $\sigma(\phi)$ or directly on ϕ, which changes the scale of the ranking gradients.
4. BCE compared to NLL implementations of the base objective. In our implementation the masked BCE-with-logits formulation corresponds to the same exclusive discrete-time likelihood as the NLL (NNet uses NLL loss) form, but we include both for completeness since they may differ in numerical behavior.

Table 5 summarizes the ablated configurations considered in this study and highlights how each variant deviates from the primary model.

Table 5: Ablations of KMNet under the fixed conditional-ranking formulation. The primary configuration is KMNet-BPE.

Variant	Base loss	Rank space	Penalty
KMNet-NLS	NLL	Logit	Softplus
KMNet-NLE	NLL	Logit	Exp
KMNet-NPS	NLL	Prob	Softplus
KMNet-NPE	NLL	Prob	Exp
KMNet-BLS	BCE	Logit	Softplus
KMNet-BLE	BCE	Logit	Exp
KMNet-BPS	BCE	Prob	Softplus
KMNet-BPE	BCE	Prob	Exp

Table 6: Concordance (Higher is Better) (mean ± std)

Dataset	KMNet-NLS	KMNet-NLE	KMNet-NPS	KMNet-NPE	KMNet-BLS	KMNet-BLE	KMNet-BPS	KMNet-BPE
Flchain	0.791 ± 0.008	0.790 ± 0.009	0.788 ± 0.008	0.793 ± 0.005	0.791 ± 0.008	0.790 ± 0.007	0.791 ± 0.006	**0.793 ± 0.006**
Metabric	0.659 ± 0.016	0.660 ± 0.008	0.661 ± 0.017	0.658 ± 0.017	0.653 ± 0.006	0.652 ± 0.024	0.664 ± 0.016	**0.664 ± 0.007**
Nacd	0.752 ± 0.019	0.757 ± 0.011	0.756 ± 0.014	0.752 ± 0.023	0.758 ± 0.015	0.757 ± 0.013	0.757 ± 0.015	**0.760 ± 0.015**
Nacd-Col	0.700 ± 0.055	**0.727 ± 0.048**	0.711 ± 0.038	0.690 ± 0.025	0.711 ± 0.056	0.696 ± 0.038	0.696 ± 0.036	0.704 ± 0.016
Nwtco	0.714 ± 0.025	0.714 ± 0.014	0.714 ± 0.019	0.706 ± 0.026	0.713 ± 0.026	0.699 ± 0.038	**0.718 ± 0.014**	0.713 ± 0.024
Recidivism	0.631 ± 0.022	0.619 ± 0.024	0.630 ± 0.017	0.609 ± 0.023	0.616 ± 0.023	0.621 ± 0.034	0.612 ± 0.023	**0.637 ± 0.012**
Rotterdam-Gbsg	0.681 ± 0.000	0.673 ± 0.004	0.678 ± 0.003	0.680 ± 0.000	0.669 ± 0.007	0.677 ± 0.000	0.671 ± 0.003	**0.682 ± 0.001**
Support	0.623 ± 0.010	0.625 ± 0.017	0.630 ± 0.002	0.621 ± 0.019	0.621 ± 0.014	0.623 ± 0.015	0.616 ± 0.022	**0.637 ± 0.006**

Table 7: Integrated Brier Score (Lower is Better) (mean ± std)

Dataset	KMNet-NLS	KMNet-NLE	KMNet-NPS	KMNet-NPE	KMNet-BLS	KMNet-BLE	KMNet-BPS	KMNet-BPE
Flchain	0.099 ± 0.002	0.100 ± 0.003	0.099 ± 0.002	0.100 ± 0.002	0.100 ± 0.002	**0.099 ± 0.002**	0.099 ± 0.002	0.099 ± 0.002
Metabric	0.163 ± 0.006	0.164 ± 0.009	**0.160 ± 0.006**	0.163 ± 0.004	0.164 ± 0.010	0.160 ± 0.004	0.161 ± 0.007	0.162 ± 0.004
Nacd	0.146 ± 0.005	0.146 ± 0.005	0.146 ± 0.007	0.143 ± 0.004	0.144 ± 0.002	0.147 ± 0.006	0.147 ± 0.007	**0.142 ± 0.004**
Nacd-Col	0.178 ± 0.017	0.170 ± 0.024	0.184 ± 0.013	0.173 ± 0.017	0.183 ± 0.029	**0.170 ± 0.018**	0.175 ± 0.010	0.173 ± 0.005
Nwtco	0.089 ± 0.002	0.088 ± 0.005	0.092 ± 0.006	0.087 ± 0.004	**0.084 ± 0.006**	0.086 ± 0.001	0.087 ± 0.006	0.086 ± 0.005
Recidivism	0.180 ± 0.017	0.179 ± 0.002	0.177 ± 0.010	**0.173 ± 0.001**	0.173 ± 0.004	0.176 ± 0.007	0.177 ± 0.006	0.177 ± 0.005
Rotterdam-Gbsg	0.178 ± 0.007	0.167 ± 0.001	0.167 ± 0.007	0.170 ± 0.000	0.173 ± 0.002	0.168 ± 0.004	0.171 ± 0.000	**0.165 ± 0.002**
Support	0.191 ± 0.002	0.192 ± 0.001	0.192 ± 0.001	**0.191 ± 0.002**	0.193 ± 0.002	0.191 ± 0.001	0.192 ± 0.003	0.191 ± 0.001

Ablation analysis. Tables 6 and 7 compare eight KMNet variants obtained by changing only the training objective while keeping the same MLP backbone, discretization scheme, and Kaplan–Meier product construction for survival prediction. Overall, the primary configuration KMNet-BPE (BCE base loss with probability-space conditional ranking and exponential penalty) achieves the best average rank for concordance and remains among the top variants for calibration, indicating that the proposed ranking formulation is robust to the choice of the base likelihood surrogate. The gains in concordance are consistent on SUPPORT, RECIDIVISM, METABRIC, and ROTTERDAM-GBSG, suggesting that ranking directly in the probability space aligns well with the patient-specific conditional survival outputs that are multiplied to form $\widehat{S}(t \mid x)$. In contrast, switching to NLL (KMNet-N*) or to logit-space ranking (*L*) typically produces smaller or inconsistent improvements, which is expected because the rank loss is applied at the event interval and interacts with the sigmoid nonlinearity and the multiplicative survival construction. The few exceptions are informative: on NACD-COL, KMNet-NLE attains the best concordance and strong IBS, while on NWTCO the best concordance is reached by KMNet-BPS but the best IBS is achieved by KMNet-BLS, indicating that datasets with different censoring patterns or time-resolution requirements can favor smoother penalties (softplus) or logit comparisons for stability. Importantly, differences in IBS across variants are generally modest (often within the reported standard deviations), whereas concordance is more sensitive to the ranking design, supporting the view that the conditional ranking term primarily affects discrimination while the base loss dominates calibration. This behavior is consistent with the architecture of KMNet: calibration is largely shaped by the per-interval supervision from the base loss, while discrimination is driven by how the rank loss separates conditional survival scores at the event interval.

B Performance Metric

Time Dependent Concordance The concordance index evaluates discrimination by measuring how well a model preserves the ordering of event times. It is particularly natural for proportional hazards models, where the relative risk ordering is consistent over time. For general non-proportional survival models, the choice of evaluation time matters, so we use a time-dependent concordance measure in the spirit of Antolini et al. (2005). In this formulation, a pair of individuals (i, j) is comparable when individual i experiences the event before individual j, that is $T_i < T_j$ with $\delta_i = 1$. The prediction is concordant if the model assigns lower survival to the individual who fails earlier when evaluated at T_i,

$$C = \Pr\Big(\hat{S}(T_i \mid x_i) < \hat{S}(T_i \mid x_j) \,\Big|\, T_i < T_j,\ \delta_i = 1\Big). \tag{1}$$

Integrated Brier Score The Brier score assesses the accuracy of probabilistic predictions through a mean squared error criterion. In binary classification, for labels $y_i \in \{0, 1\}$ with predicted probabilities p_i, it is $BS = \frac{1}{N}\sum_i (y_i - p_i)^2$. In survival analysis, one considers the binary outcome at a fixed time t, indicating whether the event occurs after t, and compares

Model	Hyperparameter	Search Space
Neural Structure (For all models)	Number of Layers	[1, 2, 4]
	Number of Nodes	[16, 32, 64]
	Learning Rate	[0.0001, 0.1]
CoxTime, NNet, BCESurv	J	[32, 64, 128]
DeepSurv	Weight Decay	[0.01, 0.1]
DeepHit	α	[0.1, 0.5, 0.9]
	σ	(0.1, 10)
	J	[32, 64, 128]
	Weight Decay	[0.01, 0.1]
KMNet	α	[0.1, 0.5, 0.9]
	σ	Auto $\cup$ (0.1, 10)
	λ	Auto $\cup$ (0.25, 4)
	J	[32, 64, 128]

Table 8: Hyperparameter search space for different model, Neural architecture search space is same for all the models

this to the predicted survival probability $\hat{S}(t \mid x_i)$. To account for right censoring, we use the inverse probability of censoring weighting approach of Graf et al. (1999), yielding a time-dependent Brier score $BS(t)$. We summarize performance across time by the integrated Brier score over a finite horizon $[0, \tau]$,

$$IBS = \frac{1}{\tau} \int_0^\tau BS(t)\, dt. \tag{2}$$

C Bayesian Optimization

We tune hyperparameters using Bayesian optimization, which treats the validation performance as a black-box function of a hyperparameter vector θ defined over a search space Θ. Instead of exhaustively evaluating configurations on a fixed grid, Bayesian optimization uses the outcomes of previously evaluated trials to guide the selection of the next configuration, thereby concentrating computation on regions of the search space that are most likely to improve performance Snoek et al. (2012). This is particularly well suited to neural survival models, where each evaluation of $f(\theta)$ requires training a model and the cost of naive search quickly becomes prohibitive. In our study, the objective value $f(\theta)$ is computed on the held-out validation split after training the model with hyperparameters θ on the training split, following the data splitting protocol in Section 4. We use the Tree-structured Parzen Estimator Bergstra et al. (2011) (TPE) method, which is a Bayesian optimization strategy designed for mixed discrete and continuous hyperparameter spaces. In our experiments, each trial consists of selecting θ from the search space in Table 8, training the model on the training portion of the development split, and evaluating $f(\theta)$ on the validation portion. After completing the trial budget, the configuration with the best validation objective is selected and used for final model fitting and test evaluation. The complete tuning procedure is summarized in Algorithm 1.

Algorithm 1 TPE-based Bayesian optimization used for hyperparameter tuning

Input
Training data $\mathcal{D}_{tr}$, validation data $\mathcal{D}_{va}$
Search space Θ
Trial budget B
Quantile level $\gamma \in (0, 1)$ used to define good trials

Output
Best hyperparameters $\theta^\star$

Initialize an empty history $\mathcal{H} \leftarrow \emptyset$
for $b = 1$ to B **do**
  if $|\mathcal{H}|$ is small **then**
    Sample θ_b from Θ using the prior distribution
  else
    Let $\{(\theta_k, f_k)\}_{k=1}^{|\mathcal{H}|} = \mathcal{H}$ where f_k is the validation objective
    Set a threshold $f^\star$ as the γ-quantile of $\{f_k\}$
    Fit two TPE density models
    $l(\theta) \approx p(\theta \mid f(\theta) \leq f^\star)$ using trials with $f_k \leq f^\star$
    $g(\theta) \approx p(\theta \mid f(\theta) > f^\star)$ using trials with $f_k > f^\star$
    Draw candidates $\{\tilde{\theta}\}$ from $l(\theta)$ and select
    $\theta_b \in \arg\max_{\tilde{\theta}} \; l(\tilde{\theta})/g(\tilde{\theta})$
  end if
  Train the model with hyperparameters θ_b on $\mathcal{D}_{tr}$
  Compute validation objective $f_b = f(\theta_b)$ on $\mathcal{D}_{va}$
  Update history $\mathcal{H} \leftarrow \mathcal{H} \cup \{(\theta_b, f_b)\}$
end for
Return $\theta^\star \in \arg\min_{(\theta,f)\in\mathcal{H}} f$