# REWARD-GUIDED SURVIVAL DATA GENERATION

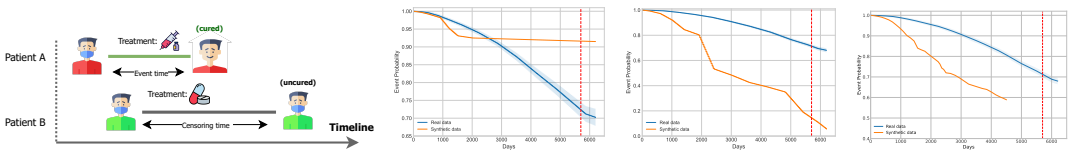
**Anonymous authors**
Paper under double-blind review

## ABSTRACT

The sharing of tabular survival data is restricted by privacy regulations. While synthetic data generation offers a promising alternative, existing methods frequently fail to simultaneously preserve long-term survival dynamics and patient privacy. We introduce **SurvTabGen**, a novel generative framework tailored specifically for survival data synthesis. Unlike prior multi-stage approaches, SurvTabGen jointly models covariates, event indicators, and survival time within a unified structurally aware generator. We first extract causal relationships among covariates via causal discovery. The generation then follows a dependency-aware sequence: event outcomes are conditioned on covariates, and survival times are conditioned on both covariates and events. Crucially, the generator is trained with a reinforcement learning-based survival-aware objective. By using the Integrated Absolute Difference (IAD) between real and synthetic survival curves as a reward signal, the model directly optimizes for macroscopic temporal fidelity. Extensive evaluations against 10 SOTA baselines across seven diverse real-world datasets demonstrate the superiority of SurvTabGen. Our results demonstrate that SurvTabGen provides a highly effective solution for generating high-fidelity, privacy-preserving synthetic cohorts that excel in survival preservation and data quality.

## 1 INTRODUCTION

*Survival data* capture the time until a predefined event occurs, frequently accompanied by *censoring* due to incomplete observations (Hahn et al., 2025). This data consists of descriptive features (e.g., demographic, treatments, lab results), collectively termed *covariates* $X$, alongside two critical features: the *event indicator* $E$ and the *time* $T$. When an event observed (i.e., $E = 1$, such as a patient recovering from a disease), $T = t$ denotes the exact event time. Conversely, when the event is not observed within the study period (i.e., $E = 0$), $T = t$ becomes censoring time, indicating that the patient survived without experiencing the event up to time $t$, as illustrated in Figure 1(a). Survival data are foundational to research in healthcare, epidemiology, and reliability engineering. However, these datasets are costly and time-consuming to collect. More critically, they often contain highly sensitive personal information, limiting their availability and restricting open data sharing (Vallevik et al., 2024). To overcome these barriers, synthetic data generation has emerged as a solution, enabling the creation of realistic, privacy-preserving survival data that can serve as viable proxies for restricted real-world datasets (Norcliffe et al., 2023).




(a) Event and censoring (b) Over-optimistic (c) Over-pessimistic (d) Short-sightedness

Figure 1: **(a)** Illustration of the competing event and censoring mechanisms in survival data. **(b-d)** Common failure modes of temporal bias in synthetic survival data.

A primary application of such data is *survival analysis*. Given a survival dataset, this analysis seeks to answer fundamental prognostic questions, such as: "*What is the probability a patient will survive past time $t$?*". This insight is critical for drug development, treatment efficacy estimation, and risk



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.

assessment. Consequently, any synthetic survival data must accurately preserve these macroscopic survival dynamics. However, generating high-fidelity survival data remains a profound challenge due to the presence of censoring. Standard tabular generative models such as CTGAN (Xu et al., 2019), TabDiff (Shi et al., 2025), and GReaT (Borisov et al., 2023) often fail in this domain because they treat every column as a definitive, fully observed ground truth. Furthermore, real-world survival data exhibits a severe structural imbalance between observed events and censored samples (Eckardt et al., 2024). If a generative model fails to capture this imbalance, it introduces severe temporal bias into the synthetic data, as evidenced by deviations in the Kaplan-Meier survival curves between real and synthetic cohorts (Figure 1(b-d)). This failure typically manifests in three distinct ways: **(1) Over-optimism**: Synthetic events occur later than in reality, inflating the predicted expected lifetime, **(2) Over-pessimism:** Synthetic events occur earlier than expected, deflating the survival probability, and **(3) Short-sightedness:** The model fails to generate long-term horizons, prematurely collapsing the survival curve. These temporal distortions complicate the learning of accurate time-to-event distributions and mislead clinical applications.

To the best of our knowledge, only three models SurvivalGAN (Norcliffe et al., 2023), the Ashhad framework (Ashhad and Henao, 2025), and SurvDiff (Brockschmidt et al., 2026) have been proposed for survival data generation. To mitigate temporal distortions, both SurvivalGAN and Ashhad rely on empirical pre-sampling. By drawing the event indicator $E$ (and the time $T$ in Ashhad's case) directly from the real data's empirical distribution, these models force the synthetic dataset to inherit the real-world censoring ratio. However, this approach introduces two critical vulnerabilities. First, anchoring the generation process to the empirical distribution increases the risk of data memorization, compromising the privacy-preserving mandate of synthetic data generation. Second, this forces an unnatural generative sequence. Sampling outcomes $E$ first and generating the covariates $X$ conditioned on those outcomes requires the model to learn the *reverse-conditional distribution* $p(X \mid E)$. This violates the natural chronological progression of clinical reality (where baseline health dictates future outcomes) and is structurally more difficult to optimize than the forward prospective distribution $p(E \mid X)$. Alternatively, SurvDiff samples from the joint distribution $p(X, E, T)$ via a diffusion process. However, it treats all features as structurally independent and lacks a mechanism to explicitly minimize macroscopic survival curve deviations, leaving it vulnerable to the temporal biases illustrated in Figure 1(b-d).

To resolve these limitations, we propose Survival Tabular Data Generator (**SurvTabGen**), an end-to-end framework that models survival data within a unified generative process. Unlike prior approaches that rely on externally sampled event indicators or reverse-conditional generation, our method enforces the causal dependency structure of clinical data. Specifically, covariates are generated first via structured sub-generators informed by a causal graph (DAG). Event outcomes are then generated conditionally on these covariates, and survival times are generated conditionally on both the covariates and the event outcomes. This sequential DAG architecture ensures that structural dependencies are respected. Furthermore, we introduce a survival-aware training objective that explicitly aligns population-level survival behavior of the real and synthetic cohorts by minimizing the Integrated Absolute Difference (IAD) between their Kaplan-Meier curves. Because this objective is non-differentiable, we optimize it via policy gradients, treating survival curve discrepancy as a global Reinforcement Learning (RL) reward. This formulation enables the model to directly optimize macroscopic temporal fidelity rather than relying solely on marginal distribution matching.

We summarize **our contributions** as follows:

1. **Architectural innovation:** We propose SurvTabGen, an end-to-end generative framework that explicitly extracts and preserves structural causal dependencies for survival data generation.

2. **Reward-guided optimization:** We develop a novel survival-aware training objective that aligns survival behavior between real and synthetic cohorts using non-differentiable metrics (IAD), optimized via policy gradients.

3. **Empirical validation:** We conduct extensive experiments across seven diverse real-world datasets, demonstrating that SurvTabGen is better than existing methods in survival fidelity.

4. **Novelty:** Compared to SurvivalGAN and Ashhad, SurvTabGen trains a single, unified model to learn the joint distribution $p(X, E, T)$, avoiding overly complex multi-stage or reverse-conditional frameworks. Compared to SurvDiff, SurvTabGen respects the causal relationship among features

and explicitly minimize macroscopic curve deviations to neutralize the temporal biases of over-optimism, over-pessimism, and short-sightedness.

## 2 Related Works

**Tabular data generative models** Generative models for tabular data synthesis include GAN/VAE architectures: CTGAN (Xu et al., 2019), CA-GAN (Nguyen et al., 2026), and TVAE (Xu et al., 2019), diffusion frameworks: TabDiff (Shi et al., 2025), and LLMs: GReaT (Borisov et al., 2023), TapTap (Zhang et al., 2023), and Pred-LLM (Nguyen et al., 2024). While these general-purpose models can reproduce marginal distributions (Eckardt et al., 2024), they struggle with long-term censoring dynamics and complex time-to-event trajectories. Because they rely on generic reconstruction rather than explicitly modeling the joint structure of covariates, event indicators, and survival times (Hahn et al., 2025; Ahmed et al., 2026), they frequently violate temporal constraints and fail to achieve true survival consistency (Vallevik et al., 2024).

**Survival data generative models** To address censoring, specialized frameworks like Survival-GAN (Norcliffe et al., 2023) and the Ashhad method (Ashhad and Henao, 2025) use multi-stage pipelines that model covariates and temporal outcomes as disjoint components. Both rely on the empirical pre-sampling of event indicators, which propagates compounding errors across disconnected sub-models and elevates privacy leakage risks by anchoring synthetic data to the original cohort. Alternatively, SurvDiff (Brockschmidt et al., 2026) employs end-to-end diffusion to jointly generate all variables from noise. However, SurvDiff treats all features as structurally independent, ignoring causal dependencies, and lacks a mechanism to explicitly enforce macroscopic temporal consistency. Consequently, current SOTA methods remain largely distribution-driven; they fail to simultaneously enforce the structural causal relationships and the population-level temporal alignment necessary for high-fidelity survival data generation.

## 3 Framework

### 3.1 Problem Formulation

Given a tabular survival dataset $\mathcal{D} = \{u_i\}_{i=1}^N$, where each instance $u_i = (x_i, t_i, e_i)$ is drawn from an unknown data distribution $p_{\text{data}}$, the primary objective is to learn a generative model that produces synthetic samples $\hat{\mathcal{D}} = \{\hat{u}_i\}_{i=1}^N$ following the same joint distribution. Unlike prior works (Norcliffe et al., 2023; Ashhad and Henao, 2025; Brockschmidt et al., 2026), we assume the underlying data is governed by a causal structure over $(X, T, E)$, inducing dependencies that extend beyond simple statistical correlations. Accordingly, our goal is to learn a generator $G_\theta$ that preserves not only marginal distributions but also directional dependencies linking covariates with event occurrence and event time. Crucially, this requires capturing the complex survival dynamics where $X$, $E$, and $T$ jointly determine the survival behavior. This entails aligning macroscopic survival patterns of the synthetic data with the real data, moving beyond conventional tabular distribution matching.

### 3.2 Proposed Method: Survival Tabular Data Generator (SurvTabGen)

We propose **SurvTabGen**, an end-to-end generative framework that integrates structured dependency modeling with a survival-aware optimization loop. Given input noise, the model jointly generates covariates, event indicators, and survival time through a unified, causally consistent generative process. To enforce realistic survival behavior, SurvTabGen leverages a reinforcement learning strategy, utilizing survival curve discrepancies as reward signals to guide training. The overview of the framework is illustrated in Figure 2.

#### 3.2.1 Structural dependency modeling

A foundational component of SurvTabGen is the dependency structure governing how variables are generated. While one could infer a single dependency graph over all variables, applying standard causal discovery algorithms (e.g., the Peter-Clark (PC) algorithm (Spirtes et al., 2000)) directly to the joint space of $(X, E, T)$ may be problematic. In survival data, $T$ and $E$ are coupled by the censoring mechanism, meaning the observed $T$ represents a mixture of true event times and censoring

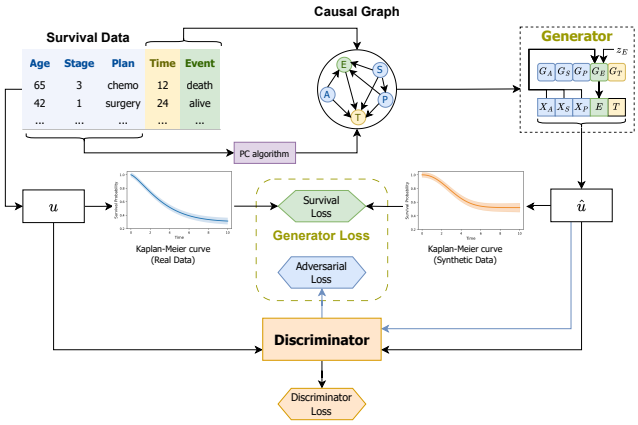



Figure 2: Overview of SurvTabGen. **(1) Causal Discovery:** A directed acyclic graph (DAG) is extracted from real survival covariates to map the underlying structural dependencies. **(2) Structured Generation:** A Wasserstein GAN produces covariates, event indicators, and survival times autoregressively along the DAG while a discriminator enforces distributional realism. **(3) Survival Alignment:** A reinforcement learning-based survival loss translates Kaplan-Meier curve discrepancies into batch-level rewards.

times. This deterministic masking violates the standard conditional independence assumptions (e.g., faithfulness) required by PC, frequently resulting in spurious edges or inverted causal directions between baseline features and clinical outcomes. Furthermore, the chronological flow from covariates ($X$) to temporal outcomes ($E$ and $T$) is an established clinical reality. Rather than forcing an algorithm to blindly rediscover this chronology from noisy, censored data, it is more robust to enforce it as a structural inductive bias. Therefore, we restrict the data-driven causal discovery phase entirely to the covariate space. Let $X = (X_1, \ldots, X_M)$ denote the feature variables. We apply PC to $X$ to infer a directed acyclic graph (DAG) $\mathcal{G}_X = (\mathcal{V}_X, \mathcal{E}_X)$. From this, we obtain the parent sets $\mathrm{PA}_i$ for each $X_i$ and establish a valid topological ordering to guide the generation phase.

While $\mathcal{G}_X$ captures dependencies among covariates, it does not specify the generation sequence for survival variables. Instead of leaving these relations implicit, we encode them directly using a survival-oriented inductive bias: the event indicator depends on the covariates, and the time-to-event depends on both the covariates and the event indicator. Formally, we augment $\mathcal{G}_X$ with the edges $X \to E$ and $(X, E) \to T$, yielding the final graph $\mathcal{G}$ with $\mathrm{PA}_E = X$ and $\mathrm{PA}_T = X \cup \{E\}$. This provides a logical dependency backbone for the generation phase.

### 3.2.2 Wasserstein GAN (WGAN) architecture

SurvTabGen adopts a modular adversarial architecture inspired by CA-GAN (Nguyen et al., 2026). It consists of a discriminator $D$ alongside feature-specific sub-generators $\{G_i\}_{i=1}^{M}$ for covariates, and two terminal sub-generators $G_E$ and $G_T$ for the event indicator and time-to-event, respectively. Let $\pi$ denote a valid topological ordering of the causal graph $\mathcal{G}$. For each feature $X_i$, the corresponding sub-generator $G_i$ models the conditional generation of $X_i$ given $\mathrm{PA}_i$, utilizing the generated values of its parent variables and an independent noise vector $z_i \sim \mathcal{N}(0, I)$. The feature and survival variables are generated as follows:

$$\hat{X}_i = G_i(\mathrm{PA}_i, z_i),\ \hat{E} = G_E(\hat{X}_1, \ldots, \hat{X}_M, z_E),\ \hat{T} = G_T(\hat{X}_1, \ldots, \hat{X}_M, \hat{E}, z_T), \tag{1}$$

where $z_i, z_E, z_T \sim \mathcal{N}(0, I)$. The full synthetic sample $\hat{u} = (\hat{X}_1, \ldots, \hat{X}_M, \hat{E}, \hat{T})$ is constructed autoregressively along $\pi$, ensuring structural consistency with the dependencies encoded in $\mathcal{G}$.

### 3.2.3 Training the WGAN

**Training discriminator** We train $D$ using the Wasserstein GAN with Gradient Penalty (WGAN-GP) objective (Gulrajani et al., 2017). The discriminator is optimized by evaluating real and generated samples via:

$$\mathcal{L}_D = -\mathbb{E}_{u \sim p_{\mathrm{data}}}[D(u)] + \mathbb{E}_{\hat{u} \sim p_\theta}[D(\hat{u})] + \mathcal{L}_{\mathrm{gp}}, \tag{2}$$

where $\mathcal{L}_{\text{gp}}$ denotes the standard gradient penalty defined on interpolated samples. This objective provides a stable adversarial signal for training the structured generator.

**Training generator** Synthetic samples $\hat{u} = (\hat{X}, \hat{E}, \hat{T})$ are generated following the topological ordering $\pi$ of $\mathcal{G}$. These samples update the generator through the adversarial objective, which encourages higher critic scores to reduce the discrepancy between the model distribution $p_\theta$ and the real data distribution $p_{\text{data}}$:

$$\mathcal{L}_{\text{adv}} = -\mathbb{E}_{\hat{u}\sim p_\theta}[D(\hat{u})], \tag{3}$$

Following the WGAN-GP training scheme, the discriminator is updated multiple times per generator step to ensure stable gradients.

### 3.2.4 Survival-preserving objective for the generator

Standard Wasserstein matching is often insufficient to capture macroscopic temporal dynamics under censoring. To mitigate this, we extend the generator objective with a survival-aware regularization term:

$$\mathcal{L}_G = \mathcal{L}_{\text{adv}} + \lambda \mathcal{L}_{\text{surv}}, \tag{4}$$

where $\mathcal{L}_{\text{surv}}$ quantifies the discrepancy in survival behavior between the real and generated cohorts. The coefficient $\lambda$ determines the strength of this constraint, allowing the model to balance feature-level distributional realism with macroscopic temporal accuracy.

Previous works modeled individual-level conditional survival functions using separate components such as DeepHit (Norcliffe et al., 2023) or one-dimensional generators (Ashhad and Henao, 2025). These multi-stage designs prevent survival behavior from being directly optimized within the generator, risking misalignment with the generated features.

In contrast, we leverage the population-level marginal survival function $S(t) = P(T > t)$ as the direct alignment target. Although defined marginally, this function reflects the complex temporal behavior induced by the underlying joint distribution over $(X, E, T)$ because both the event occurrence and the time-to-event are generated conditionally on the covariates through the estimated DAG structure. Aligning $S(t)$ therefore provides a stable and low-variance signal under censoring while encouraging consistency with the temporal behavior induced by the joint distribution.

We estimate $S(t)$ with the Kaplan-Meier estimator, using observed time-event pairs $\{(t_i, e_i)\}_{i=1}^N$:

$$S(t) = \prod_{t_i \le t} \left(1 - \frac{d_i}{n_i}\right), \tag{5}$$

where $d_i$ denotes the number of events at time $t_i$ and $n_i$ is the number at risk just prior to $t_i$. The estimator is evaluated on a discrete grid $\mathcal{T} = \{t_1, \ldots, t_K\}$, constructed from either sorted event times or a uniform partition of the observed range to ensure consistent comparison between datasets.

The discrepancy between survival curves is quantified using *integrated absolute difference* (IAD):

$$\text{IAD} = \int |S_{\text{real}}(t) - S_{\text{syn}}(t)|\, dt \tag{6}$$

which is approximated numerically on the discrete grid $\mathcal{T}$ using the trapezoidal rule. This formulation measures global deviations in temporal dynamics while remaining sensitive to differences in tail behavior and censoring patterns.

To theoretically justify the use of IAD as a training signal, we establish a formal connection between this metric and the optimal transport framework of our underlying WGAN.

**Proposition 1.** *Let $P_T$ and $Q_T$ denote the true and synthetic marginal distributions of the event time $T$ (both with finite first moments), with respective survival functions $S_{\text{real}}(t)$ and $S_{\text{syn}}(t)$. The continuous form of the IAD coincides with the 1-Wasserstein distance between the corresponding one-dimensional time-to-event distributions:* $\text{IAD} = \mathcal{W}_1(P_T, Q_T)$.

*In practice, when survival times are subject to censoring, we approximate $S_{\text{real}}(t)$ and $S_{\text{syn}}(t)$ using Kaplan-Meier estimators, allowing the empirical IAD to serve as a proxy for the Wasserstein distance between the underlying (latent) time-to-event distributions.*

Proposition 1 (*proof in Appendix A*) reveals a theoretical insight: while the discriminator minimizes the *joint* high-dimensional distance, the curse of dimensionality often prevents WGANs from perfectly matching specific margins under censoring. By utilizing this Wasserstein-aligned temporal metric (IAD), we construct a targeted reward signal that directly optimizes this temporal margin.

Because the Kaplan-Meier estimator requires a population to compute, we define a shared **batch-level reward** for a generated batch $\hat{\mathcal{B}} = \{\hat{u}_1, \ldots, \hat{u}_B\}$ as: $R(\hat{\mathcal{B}}) = -\text{IAD}$. This shared reward is uniformly assigned to all samples within the batch. Our **survival loss** is therefore defined over the batch as: $\mathcal{L}_{\text{surv}} = -\mathbb{E}_{\hat{\mathcal{B}} \sim p_\theta}\left[R(\hat{\mathcal{B}})\right]$.

However, the survival loss cannot be optimized via standard backpropagation because the reward depends on the *non-differentiable* Kaplan-Meier procedure and the IAD metric. To optimize $G$, we cast the generation process into a reinforcement learning (RL) paradigm. Because standard WGAN generators are implicit models, the true generative likelihood $p_\theta(\hat{u})$ required for standard policy gradients is intractable. We resolve this by interpreting the terminal layers of the generator as a stochastic policy, thereby constructing a tractable *surrogate likelihood*. Specifically, for **categorical variables**, we employ the Gumbel-Softmax trick (Jang et al., 2016). For **continuous variables**, the generator outputs parameterize a stochastic distribution (e.g., Gaussian). While this formulation acts as an approximation rather than the true model likelihood, it provides a valid surrogate log-probability for a generated sample $\hat{u}$. This enables optimization via the score-function estimator:

$$\log p_\theta(\hat{u}) = \sum_{i=1}^{M} \log p_i(\hat{X}_i \mid \text{PA}_i) + \log p_E(\hat{E} \mid \text{PA}_E) + \log p_T(\hat{T} \mid \text{PA}_T) \tag{7}$$

To stabilize the policy gradient, we augment the reward signal with an auxiliary penalty term $\mathcal{P}(\hat{\mathcal{B}})$ that jointly regularizes discrepancies in marginal event rates and conditional time distributions. Formal definition of $\mathcal{P}(\hat{\mathcal{B}})$ is provided in *Appendix* B. We optimize the generator using the REINFORCE policy gradient algorithm (Williams, 1992):

$$\nabla_\theta \mathcal{L}_{\text{surv}} = -\mathbb{E}_{\hat{\mathcal{B}} \sim p_\theta}\left[(R(\hat{\mathcal{B}}) - \mathcal{P}(\hat{\mathcal{B}}))\frac{1}{B}\sum_{b=1}^{B} \nabla_\theta \log p_\theta(\hat{u}_b)\right] \tag{8}$$

Overall, SurvTabGen integrates structural generation with a survival-aware RL optimization loop. By combining adversarial training with a reward-driven objective, it captures both underlying distribution and macroscopic temporal dynamics. Consequently, by minimizing the discrepancy between the survival curves, it helps mitigate the three failure modes in synthetic survival data generation: *over-optimism*, *over-pessimism*, and *short-sightedness*, as illustrated previously in Figure 1(b-d).

# 4 Experiments

## 4.1 Experiment Settings

**Datasets** We evaluate SurvTabGen on seven datasets, including *AIDS* (Hosmer et al., 2008), Chronic Graft-versus-Host Disease (*Cgvhd*) (Pintilie, 2006), Serum Free Light Chain (*Flchain*) (Dispenzieri et al., 2012), German Breast Cancer Study Group 2 (*Gbsg2*) (Schumacher et al., 1994), Veterans' Lung Cancer (*Veterans*) (Kalbfleisch and Prentice, 2002), Worcester Heart Attack Study (*Whas500*) (Hosmer et al., 2008), and Additively Manufactured (*AM*) (Zhang and Xu, 2023). All datasets contain mixed-type features (continuous and categorical) and are often used in prior synthetic survival literature (Norcliffe et al., 2023; Ashhad and Henao, 2025; Brockschmidt et al., 2026). Their full descriptions are provided in *Appendix* D. Notably, *while prior methods were tested on only 3-5 datasets strictly within the healthcare domain, SurvTabGen is rigorously evaluated across 7 datasets spanning both healthcare and engineering applications*.

**Evaluation metrics** Following previous works (Norcliffe et al., 2023; Ashhad and Henao, 2025; Brockschmidt et al., 2026), we evaluate the generated data across three aspects. **Survival fidelity:** Assessed using Kaplan-Meier based metrics (Norcliffe et al., 2023) such as *KM Divergence*, *Optimism*, and *Short-sightedness*, where values closer to zero indicate superior survival preservation.

**Data quality:** Evaluated in terms of distributional fidelity using *Jensen-Shannon* (JS) and *Wasserstein* distances, where lower values are optimal. **Privacy:** Assessed using the *Distance to Closest Record* (DCR) (Yang et al., 2025) and *Re-identification risk* (Yoon et al., 2020). While excessively low DCR values may indicate potential privacy leakage, excessively high values suggest poor realism. Re-identification risk evaluates whether synthetic samples can be linked back to real individuals. Detailed formulations for these metrics are provided in *Appendix* C.

**Baselines** We benchmark SurvTabGen against 10 recent SOTA baselines spanning four generative paradigms: **(1) VAE and GAN-based models:** TVAE (Xu et al., 2019), CTGAN (Xu et al., 2019), and CA-GAN (Nguyen et al., 2026), **(2) LLM-based approaches:** GReaT (Borisov et al., 2023), TapTap (Zhang et al., 2023), and Pred-LLM (Nguyen et al., 2024), **(3) Diffusion-based methods:** TabDiff (Shi et al., 2025), and **(4) Survival-aware methods:** SurvivalGAN (Norcliffe et al., 2023), the Ashhad framework (Ashhad and Henao, 2025), and SurvDiff (Brockschmidt et al., 2026). SurvivalGAN serves as our main competitor as it also uses a GAN backbone.

**Implementation details** The discriminator is a 3-layer MLP with two hidden layers of 256 units while each sub-generator is a 4-layer MLP. For output activations, we apply *tanh* for continuous variables and *Gumbel-Softmax* trick for categorical variables. The model is trained for 500 epochs with a noise dimension of 16 and a batch size of 500. The survival coefficient is fixed at $\lambda = 0.1$ (sensitivity analysis is provided in *Appendix* E.3). We run PC with $\alpha = 0.05$ to extract the estimated DAG from the covariates (alternative causal discovery methods are investigated in *Appendix* E.4). We repeat each experiment 5 times and report the average score along with standard deviation.

### 4.2 Results and Discussions

**Survival preservation** As shown in Table 1, SurvTabGen achieves the lowest average KM Divergence score, indicating better alignment with real-world survival distributions. The Ashhad method performs well, securing the second-best average. Because Ashhad samples $E$ and $T$ from empirical distributions, it may copy the time-event distribution of the real dataset. Therefore, it introduces a critical weakness: the resulting synthetic datasets map too closely to the real datasets, increasing information leakage. Our subsequent privacy evaluation highlights this vulnerability.

Overall, SurvTabGen outperforms SurvivalGAN by a 3% margin. This advantage is demonstrated in Figure 3. While SurvTabGen accurately tracks the real Kaplan-Meier curves across diverse datasets, capturing both early-onset dynamics and long-term tail behavior, SurvivalGAN exhibits substantial deviations, particularly in later stages. These results suggest that incorporating causal dependencies and a survival-aware objective yields temporal fidelity superior to standard distribution matching.

Table 1: KM Divergence scores (↓) showing the survival preservation performance of each method. **Bold** and underline indicate the best and second-best methods.

| | AIDS | Cgvhd | Flchain | Gbsg2 | Veterans | Whas500 | AM | AVG |
|---|---|---|---|---|---|---|---|---|
| CTGAN | 0.046 (0.01) | 0.044 (0.02) | 0.079 (0.02) | 0.087 (0.02) | 0.091 (0.07) | 0.133 (0.04) | 0.104 (0.01) | 0.084 (0.03) |
| CA-GAN | 0.148 (0.01) | 0.273 (0.18) | 0.099 (0.01) | 0.185 (0.06) | 0.307 (0.07) | 0.104 (0.07) | 0.096 (0.01) | 0.173 (0.06) |
| TVAE | 0.055 (0.01) | 0.053 (0.00) | 0.057 (0.01) | 0.075 (0.01) | 0.078 (0.00) | 0.040 (0.01) | 0.061 (0.01) | 0.060 (0.01) |
| TabDiff | 0.159 (0.06) | 0.140 (0.11) | 0.086 (0.01) | 0.231 (0.08) | 0.265 (0.10) | 0.147 (0.04) | 0.283 (0.06) | 0.187 (0.07) |
| GReaT | 0.059 (0.00) | 0.034 (0.00) | 0.026 (0.00) | 0.109 (0.00) | 0.043 (0.00) | 0.102 (0.00) | **0.012** (0.00) | 0.055 (0.00) |
| TapTap | 0.065 (0.00) | 0.142 (0.00) | 0.174 (0.00) | 0.257 (0.00) | 0.100 (0.00) | 0.199 (0.00) | 0.132 (0.00) | 0.153 (0.00) |
| Pred-LLM | 0.041 (0.00) | 0.032 (0.00) | 0.061 (0.00) | 0.046 (0.00) | 0.043 (0.00) | 0.091 (0.00) | 0.038 (0.00) | 0.050 (0.00) |
| SurvivalGAN | 0.014 (0.00) | 0.034 (0.01) | 0.040 (0.00) | 0.063 (0.03) | 0.055 (0.00) | 0.092 (0.01) | 0.089 (0.02) | 0.055 (0.01) |
| Ashhad | **0.013** (0.00) | **0.016** (0.00) | 0.025 (0.00) | **0.012** (0.00) | **0.020** (0.00) | 0.158 (0.00) | 0.021 (0.00) | 0.038 (0.00) |
| SurvDiff | 0.088 (0.01) | 0.037 (0.00) | 0.151 (0.01) | 0.123 (0.02) | 0.041 (0.01) | 0.179 (0.04) | 0.024 (0.00) | 0.092 (0.01) |
| SurvTabGen | 0.026 (0.01) | 0.023 (0.00) | **0.012** (0.00) | 0.026 (0.03) | **0.020** (0.01) | **0.035** (0.02) | 0.040 (0.01) | **0.026** (0.01) |

**Data quality** As shown in Figure 4, SurvTabGen outperforms all non-LLM methods. Two LLM-based methods GReaT and Pred-LLM achieve slightly lower distributional distances. This is expected since LLMs excel at capturing fine-grained tabular nuances, achieving better distributional alignment at the cost of high computational overhead. SurvTabGen prioritizes the preservation of multivariate survival dynamics. While this introduces a minor trade-off in marginal matching, the performance gap remains minimal ($\leq 0.002$), and is achieved at a fraction of the computational cost.

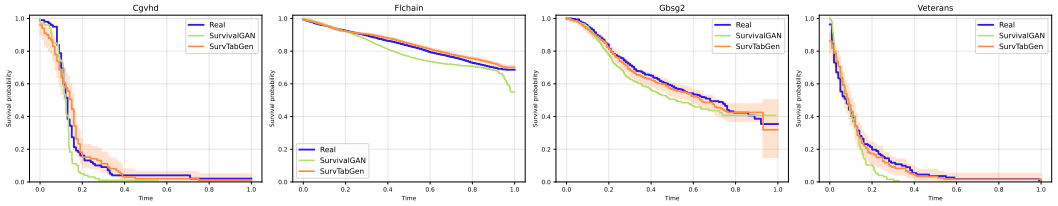



Figure 3: Kaplan-Meier curves across 4 datasets Cgvhd, Flchain, Gbsg2, and Veterans. SurvTabGen closely preserves the real survival trajectories, capturing both early and late-stage dynamics whereas SurvivalGAN exhibits significant deviations. Time is scaled to $[0, 1]$ for consistent representation.

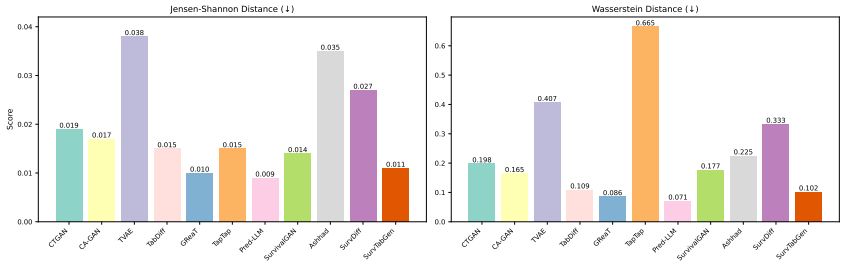



Figure 4: Synthetic data quality measured by the average Jensen-Shannon and Wasserstein distances across all datasets. *Lower values indicate higher distributional fidelity*.

**Privacy preservation** Table 2 reports the average DCR and Re-identification risk over all datasets. Compared to both SurvivalGAN and Ashhad, SurvTabGen achieves a higher 5th percentile DCR and a lower Re-identification risk, indicating a stronger defense against privacy leakage. While SurvDiff registers the lowest Re-identification risk and the highest DCR values, this must be interpreted cautiously. Its substantially inflated DCR suggests that its synthetic samples have drifted far from the real distribution, a conclusion supported by its poor data quality metrics shown in Figure 4. Overall, SurvTabGen achieves the most effective and meaningful balance between privacy preservation and data fidelity among survival-oriented methods.

Table 2: Privacy evaluation using DCR and Re-identification risk across all datasets.

| | CTGAN | CA-GAN | TVAE | TabDiff | GReaT | TapTap | Pred-LLM | SurvivalGAN | Ashhad | SurvDiff | SurvTabGen |
|---|---|---|---|---|---|---|---|---|---|---|---|
| DCR (↑) | 0.105 | 0.134 | 0.052 | 0.110 | 0.059 | 0.088 | 0.051 | 0.056 | 0.067 | 0.215 | 0.128 |
| Risk (↓) | 0.198 | 0.141 | 0.251 | 0.148 | 0.359 | 0.158 | 0.409 | 0.324 | 0.331 | 0.091 | 0.307 |

### 4.3 Ablation Studies

**Component analysis** We conduct an ablation to isolate the contribution of each component. As shown in Figure 5, removing causal discovery module (**no causal structure**) degrades the survival fidelity, highlighting the necessity of learning the underlying feature relationships. Similarly, disabling the survival-aware objective (**no survival loss** i.e., $\lambda = 0$) leads to performance drops, confirming the critical role of survival supervision in guiding temporal alignment. The **full SurvTabGen model** achieves optimal performance (the lowest KM Divergence), suggesting that both components are mutually beneficial. *(Note: Sensitivity analyses regarding the coefficient $\lambda$ and alternative causal discovery algorithms are detailed in Appendices E.3 and E.4).*

**Visualization** Figure 6 presents a t-SNE visualization (Van der Maaten and Hinton, 2008) of the synthetic samples generated by each method on the *Flchain* dataset. GReaT, Pred-LLM, SurvivalGAN, and SurvTabGen successfully capture the major clusters observed in the real data (*first plot*), indicating a reasonable preservation of dominant data structures. However, SurvivalGAN primarily reproduces only the dense clustered regions, exhibiting weak coverage of boundary areas and scattered, non-clustered samples. In contrast, SurvTabGen preserves the primary clusters while consistently capturing the peripheral and dispersed data points. This visual evidence further supports the

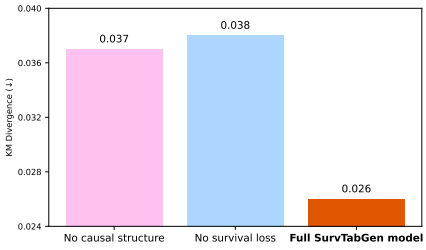



Figure 5: Impact of components on survival fidelity. KM Divergence is averaged across all datasets.

conclusion that SurvTabGen captures a more complete and representative joint survival distribution than SurvivalGAN.

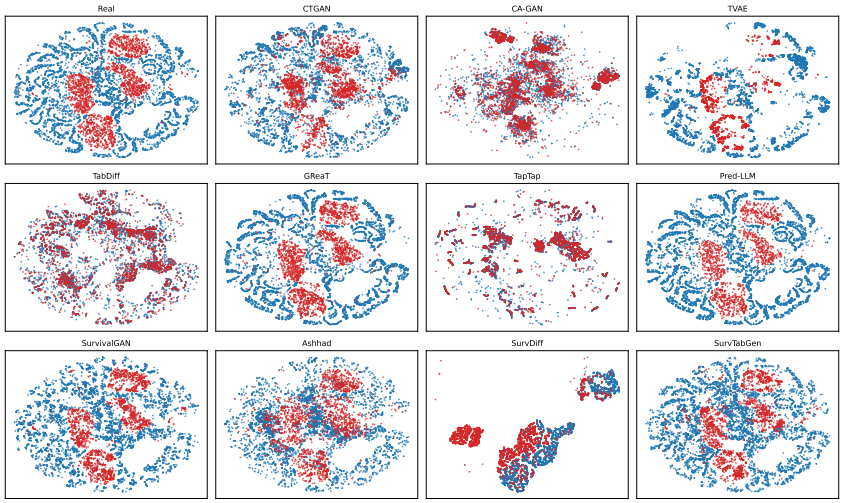



Figure 6: t-SNE visualization of real and synthetic samples on the *Flchain* dataset. Red indicates an observed event ($E = 1$) while blue indicates censoring ($E = 0$). SurvTabGen demonstrates superior coverage of both dense clusters and peripheral data points.

### 4.4 Additional Experiments

**Other metrics:** *Appendix* E.1 reports *Optimism* and *Short-sightedness*. **Subgroup-specific survival preservation:** *Appendix* E.2 reports performance at subpopulation level. **Survival estimators:** *Appendix* E.5 investigates the impact of alternative survival estimators. **Robustness to causal graph $\mathcal{G}$:** *Appendix* E.6 examines the sensitivity to potential errors in $\mathcal{G}$. **Downstream utility:** *Appendix* E.7 reports downstream performance. **Computational efficiency:** *Appendix* E.8 compares the runtime of each method.

## 5 Conclusion

SurvTabGen provides an effective, end-to-end framework for synthetic survival data generation. By unifying the generation of covariates, event indicators, and survival times into a single joint process, it overcomes the disjointed architectures of prior methods. SurvTabGen embeds structural clinical reality via causal discovery while simultaneously enforcing macroscopic temporal alignment through a reinforcement learning-driven survival objective, minimizing Kaplan-Meier curve discrepancies via policy gradients. Extensive evaluations against 10 SOTA baselines across seven diverse real-world datasets show that SurvTabGen demonstrates strong and consistent performance across diverse survival generation benchmarks. It consistently achieves superior survival preservation while maintaining competitive distributional quality and robust privacy protection.

## AI USE STATEMENT

Generative AI tools were used to support language refinement and improve the clarity and readability of the paper. All AI-assisted revisions were reviewed by the authors, who remain fully responsible for the final content.

## REPRODUCIBILITY STATEMENT

The main text and appendix provide detailed information on the proposed method, model design, training configuration, data preprocessing, and evaluation procedure to support reproducibility. The complete source code will be released publicly upon acceptance.

## REFERENCES

Odd Aalen, Ornulf Borgan, and Hakon Gjessing. *Survival and event history analysis: a process point of view*. 2008.

Awais Ahmed, Xiaoyang Zeng, Rui Xi, Mengshu Hou, Muhammad Afzal, and Syed Attique Shah. Identifying pertinent cohorts and addressing imbalance for robust intensive care survival analysis. *Engineering Applications of Artificial Intelligence*, 2026.

Bryan Andrews, Joseph Ramsey, Ruben Sanchez Romero, Jazmin Camchong, and Erich Kummerfeld. Fast scalable and accurate discovery of DAGs using the best order score search and grow shrink trees. In *NeurIPS*, 2023.

Mohd Ashhad and Ricardo Henao. Generating accurate synthetic survival data by conditioning on outcomes. In *Machine Learning for Healthcare Conference*, 2025.

Avinash Barnwal, Hyunsu Cho, and Toby Hocking. Survival regression with accelerated failure time model in XGBoost. *Journal of Computational and Graphical Statistics*, 2022.

Vadim Borisov, Kathrin Seßler, Tobias Leemann, Martin Pawelczyk, and Gjergji Kasneci. Language models are realistic tabular data generators. In *ICLR*, 2023.

Marie Brockschmidt, Maresa Schroder, and Stefan Feuerriegel. SurvDiff: A Diffusion Model for Generating Synthetic Data in Survival Analysis. In *ICML*, 2026.

Maxwell Chickering. Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 2002.

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

Angela Dispenzieri, Jerry Katzmann, Robert Kyle, Dirk Larson, Terry Therneau, Colin Colby, Raynell Clark, Graham Mead, Shaji Kumar, Joseph Melton, et al. Use of nonclonal serum immunoglobulin free light chains to predict overall survival in the general population. In *Mayo Clinic Proceedings*, 2012.

Jan-Niklas Eckardt, Waldemar Hahn, Christoph Rollig, Sebastian Stasik, Uwe Platzbecker, Carsten Muller-Tidow, Hubert Serve, Claudia Baldus, Christoph Schliemann, Kerstin Schafer-Eckart, et al. Mimicking clinical trials with synthetic acute myeloid leukemia patients using generative artificial intelligence. *NPJ Digital Medicine*, 2024.

Thomas Fleming and David Harrington. *Counting processes and survival analysis*. 2013.

Brier Glenn. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 1950.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of Wasserstein GANs. *NeurIPS*, 2017.

Waldemar Hahn, Jan-Niklas Eckardt, Christoph Rollig, Martin Sedlmayr, Jan Moritz Middeke, and Markus Wolfien. Generating reliable synthetic clinical trial data: The role of hyperparameter optimization and domain constraints. *Information Sciences*, 2025.

Frank Harrell, Robert Califf, David Pryor, Kerry Lee, and Robert Rosati. Evaluating the yield of medical tests. *JAMA*, 1982.

David Hosmer, Stanley Lemeshow, and Susanne May. *Applied survival analysis: regression modeling of time-to-event data*. 2008.

Hemant Ishwaran, Udaya Kogalur, Eugene Blackstone, and Michael Lauer. Random survival forests. 2008.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

John Kalbfleisch and Ross Prentice. *The statistical analysis of failure time data*. 2002.

Dang Nguyen, Sunil Gupta, Kien Do, Thin Nguyen, and Svetha Venkatesh. Generating realistic tabular data with large language models. In *ICDM*, 2024.

Tu Anh Hoang Nguyen, Dang Nguyen, Tri-Nhan Vo, Thuc Duy Le, Trung Le, and Sunil Gupta. Causal-aware tabular gans with reinforcement learning. In *ACML*, 2026.

Alexander Norcliffe, Bogdan Cebere, Fergus Imrie, Pietro Lio, and Mihaela van der Schaar. SurvivalGAN: Generating time-to-event data for survival analysis. In *AISTAT*, 2023.

Melania Pintilie. *Competing risks: a practical perspective*. 2006.

Martin Schumacher, Guenter Bastert, Henryk Bojar, Karl Hubner, Manfred Olschewski, Willi Sauerbrei, Cornelia Schmoor, Christa Beyerle, Rolf Neumann, and Heinz Rauschecker. Randomized 2 x 2 trial evaluating hormonal treatment and the duration of chemotherapy in node-positive breast cancer patients. *Journal of Clinical Oncology*, 1994.

Juntong Shi, Minkai Xu, Harper Hua, Hengrui Zhang, Stefano Ermon, and Jure Leskovec. TabDiff: a unified diffusion model for multi-modal tabular data generation. In *ICLR*, 2025.

Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, prediction, and search*. MIT Press, 2000.

Peter Spirtes, Christopher Meek, and Thomas Richardson. Causal inference in the presence of latent variables and selection bias. *arXiv preprint arXiv:1302.4983*, 2013.

Binz Vallevik, Aleksandar Babic, Serena Marshall, Severin Elvatun, Helga Brogger, Sharmini Alagaratnam, Bjorn Edwin, Narasimha Veeraragavan, Kjersti Befring, and Jan Nygaard. Can I trust my fake data–a comprehensive quality assessment framework for synthetic tabular data in healthcare. *International Journal of Medical Informatics*, 2024.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 2008.

Ronald Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 1992.

Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. Modeling tabular data using Conditional GAN. In *NeurIPS*, 2019.

Shuo Yang, Zheyu Zhang, Bardh Prenkaj, and Gjergji Kasneci. Doubling Your Data in Minutes: Ultra-fast Tabular Data Generation via LLM-Induced Dependency Graphs. In *EMNLP*, 2025.

Jinsung Yoon, Lydia Drumright, and Mihaela Van Der Schaar. Anonymization through data synthesis using generative adversarial networks (ADS-GAN). *IEEE Journal of Biomedical and Health Informatics*, 2020.

Tianping Zhang, Shaowen Wang, Shuicheng Yan, Jian Li, and Qian Liu. Generative table pre-training empowers models for tabular prediction. In *EMNLP*, 2023.

Zian Zhang and Zhiping Xu. Fatigue database of additively manufactured alloys. *Scientific Data*, 2023.

## A Proof for Proposition 1

For one-dimensional probability measures on $\mathbb{R}$ with finite first moments, the 1-Wasserstein distance is the integral of the absolute difference between their cumulative distribution functions (CDFs). Let $F_{\text{real}}(t) = 1 - S_{\text{real}}(t)$ and $F_{\text{syn}}(t) = 1 - S_{\text{syn}}(t)$ be the respective CDFs. Thus:

$$\mathcal{W}_1(P_T, Q_T) = \int_{-\infty}^{\infty} |F_{\text{real}}(t) - F_{\text{syn}}(t)|\, dt$$

As $T \geq 0$, the lower bound of integration becomes zero:

$$\begin{aligned}
\mathcal{W}_1(P_T, Q_T) &= \int_{0}^{\infty} |F_{\text{real}}(t) - F_{\text{syn}}(t)|\, dt \\
&= \int_{0}^{\infty} |(1 - S_{\text{real}}(t)) - (1 - S_{\text{syn}}(t))|\, dt \\
&= \int_{0}^{\infty} |S_{\text{syn}}(t) - S_{\text{real}}(t)|\, dt \\
&= \text{IAD}
\end{aligned}$$

In practical survival settings, the survival functions $S_{\text{real}}(t)$ and $S_{\text{syn}}(t)$ are not observable and are instead estimated via Kaplan-Meier estimators. Therefore, the resulting empirical IAD provides a nonparametric approximation of the above quantity under standard assumptions (e.g., independent censoring).

## B Formulation of the Auxiliary Penalty $\mathcal{P}(\hat{\mathcal{B}})$

To stabilize the policy gradient, we define an auxiliary penalty $\mathcal{P}(\hat{\mathcal{B}})$. This joint penalty minimizes discrepancies in the marginal event rates and conditional time distributions between the generated batch $\hat{\mathcal{B}}$ and the real dataset $\mathcal{D}$. We set the generated batch size equal to the real dataset size (i.e., $B = N$) to ensure stable, full-population metric evaluations.

The penalty consists of two components. The **event penalty** $\mathcal{P}_E$ measures the absolute difference in expected event rates:

$$\mathcal{P}_E(\hat{\mathcal{B}}) = \left| \frac{1}{N} \sum_{i=1}^{N} e_i - \frac{1}{N} \sum_{j=1}^{N} \hat{e}_j \right| \tag{9}$$

The **time penalty** $\mathcal{P}_T$ aligns the temporal distributions by minimizing the absolute difference across a set of $Q$ quantiles of the survival times:

$$\mathcal{P}_T(\hat{\mathcal{B}}) = \sum_{q \in Q} \left| \text{Quantile}_q(T) - \text{Quantile}_q(\hat{T}) \right| \tag{10}$$

The final joint auxiliary penalty is the sum of these components:

$$\mathcal{P}(\hat{\mathcal{B}}) = \mathcal{P}_E(\hat{\mathcal{B}}) + \mathcal{P}_T(\hat{\mathcal{B}}) \tag{11}$$

## C Metrics

We summarize survival evaluation metrics used in this work. They are standard metrics in survival analysis and synthetic survival data evaluation (Norcliffe et al., 2023; Ashhad and Henao, 2025; Brockschmidt et al., 2026).

### C.1 KAPLAN-MEIER (KM) DIVERGENCE

KM divergence measures the discrepancy between survival curves estimated from real and synthetic data:

$$\text{KM Divergence} = \frac{1}{T}\int_0^T |S_{\text{syn}}(t) - S_{\text{real}}(t)|\, dt, \tag{12}$$

where $S_{\text{real}}(t)$ and $S_{\text{syn}}(t)$ are Kaplan-Meier survival functions, and $T$ is the maximum observed time horizon. *Lower values indicate better alignment of survival dynamics*.

### C.2 OPTIMISM

Optimism captures bias in survival estimation:

$$\text{Optimism} = \frac{1}{T}\int_0^T (S_{\text{syn}}(t) - S_{\text{real}}(t))\, dt \tag{13}$$

Positive values indicate *over-optimism* (synthetic survival is higher) while negative values indicate *over-pessimism*. Values closer to 0 are preferred.

### C.3 SHORT-SIGHTEDNESS

Short-Sightedness measures how well the synthetic data covers the temporal range of real data:

$$\text{Short-Sightedness} = \frac{T_{\text{real}} - T_{\text{syn}}}{T_{\text{real}}}, \tag{14}$$

where $T_{\text{real}}$ is the maximum observed time in the real dataset and $T_{\text{syn}}$ is the corresponding maximum time in the synthetic dataset. Values close to 0 indicate good coverage while larger values suggest early truncation of events.

## D DATA DESCRIPTION

We use seven real-world survival datasets with diverse characteristics in terms of size, censoring ratio, and feature composition, covering both clinical and industrial domains. They are summarized in Table 3.

Overall, the datasets vary significantly in scale (from 100 to 7,874 instances), degree of censoring (from low to highly imbalanced), and feature types (mixed continuous-categorical or fully continuous). This diversity ensures a comprehensive evaluation setting, allowing us to assess the robustness and generalization ability of the proposed method across different survival data scenarios.

Table 3: Summary statistics of the datasets used in the study.

| Dataset | No. instances | No. censored instances | No. features | No. continuous features | No. categorical features |
|---|---|---|---|---|---|
| AIDS | 1,151 | 1,055 | 11 | 3 | 8 |
| Cgvhd | 100 | 4 | 4 | 1 | 3 |
| Flchain | 7,874 | 5,705 | 9 | 4 | 5 |
| Gbsg2 | 686 | 387 | 8 | 5 | 3 |
| Veterans | 137 | 9 | 6 | 3 | 3 |
| Whas500 | 500 | 285 | 14 | 6 | 8 |
| AM | 3,245 | 2,780 | 10 | 10 | 0 |

## E ADDITIONAL EXPERIMENTS

### E.1 SURVIVAL PRESERVATION

In addition to the KM Divergence metric reported in the main paper, we further evaluate survival preservation using Optimism and Short-sightedness metrics (described in *Appendix* C), where values closer to zero indicate better alignment with real survival behavior.

As shown in Table 4, SurvTabGen achieves the average values closest to zero on both metrics, indicating balanced and unbiased survival modeling. Other methods (especially SurvivalGAN) exhibit larger deviations, reflecting either over-optimistic or over-pessimistic behavior. These results further support the effectiveness of SurvTabGen in preserving realistic survival dynamics.

Table 4: Optimism and Short-Sightedness metrics showing the survival preservation performance of each method across datasets.

| Metrics | Methods | AIDS | Cgvhd | Flchain | Gbsg2 | Veterans | Whas500 | AM | AVG |
|---|---|---|---|---|---|---|---|---|---|
| Optimism ($\rightarrow 0$) | CTGAN | -0.019 (0.02) | 0.020 (0.07) | -0.043 (0.02) | -0.040 (0.07) | 0.064 (0.10) | -0.094 (0.06) | -0.104 (0.01) | -0.031 (0.05) |
| | CA-GAN | -0.129 (0.00) | 0.263 (0.19) | -0.076 (0.01) | -0.089 (0.14) | 0.300 (0.08) | -0.006 (0.11) | -0.062 (0.01) | 0.029 (0.01) |
| | TVAE | 0.055 (0.01) | -0.051 (0.00) | -0.056 (0.01) | 0.054 (0.02) | 0.076 (0.00) | 0.032 (0.02) | 0.053 (0.02) | 0.023 (0.01) |
| | TabDiff | -0.146 (0.07) | 0.114 (0.13) | **0.008** (0.04) | 0.085 (0.23) | 0.262 (0.10) | 0.044 (0.14) | -0.283 (0.06) | **0.012** (0.11) |
| | GReat | 0.022 (0.00) | -0.033 (0.00) | 0.022 (0.00) | -0.073 (0.00) | -0.043 (0.00) | -0.090 (0.00) | **-0.008** (0.00) | -0.029 (0.00) |
| | TapTap | -0.065 (0.00) | -0.142 (0.00) | -0.174 (0.00) | -0.257 (0.00) | -0.100 (0.00) | -0.199 (0.00) | 0.033 (0.00) | -0.129 (0.00) |
| | Pred-LLM | 0.040 (0.00) | 0.600 (0.00) | 0.020 (0.00) | 0.080 (0.00) | 0.590 (0.00) | 0.070 (0.00) | -0.034 (0.00) | 0.195 (0.00) |
| | SurvivalGAN | **-0.009** (0.01) | -0.025 (0.01) | -0.035 (0.00) | -0.037 (0.04) | -0.045 (0.01) | -0.031 (0.02) | -0.089 (0.02) | -0.039 (0.02) |
| | Ashhad | **-0.009** (0.00) | -0.012 (0.00) | -0.012 (0.00) | **0.008** (0.00) | **-0.004** (0.00) | 0.158 (0.00) | **-0.008** (0.00) | 0.017 (0.00) |
| | SurvDiff | -0.086 (0.01) | -0.032 (0.01) | -0.145 (0.01) | -0.123 (0.02) | -0.035 (0.01) | -0.009 (0.07) | -0.016 (0.00) | -0.064 (0.02) |
| | SurvTabGen | -0.019 (0.01) | **-0.009** (0.01) | -0.010 (0.01) | -0.017 (0.02) | 0.006 (0.01) | **0.002** (0.03) | -0.037 (0.02) | **-0.012** (0.01) |
| Short-Sightedness ($\rightarrow 0$) | CTGAN | **0.000** (0.00) | 0.238 (0.12) | 0.002 (0.00) | **0.000** (0.00) | 0.156 (0.22) | **0.002** (0.00) | 0.002 (0.00) | 0.057 (0.05) |
| | CA-GAN | 0.142 (0.01) | 0.364 (0.09) | 0.038 (0.00) | 0.294 (0.02) | 0.310 (0.12) | 0.110 (0.03) | 0.072 (0.03) | 0.190 (0.04) |
| | TVAE | **0.000** (0.00) | 0.808 (0.01) | 0.006 (0.01) | **0.000** (0.00) | 0.820 (0.03) | 0.024 (0.02) | **0.000** (0.00) | 0.237 (0.01) |
| | TabDiff | 0.002 (0.00) | 0.140 (0.17) | **0.000** (0.00) | 0.004 (0.01) | 0.148 (0.15) | **0.002** (0.00) | 0.016 (0.02) | 0.045 (0.05) |
| | GReat | 0.040 (0.00) | 0.330 (0.00) | 0.010 (0.00) | 0.110 (0.00) | 0.300 (0.00) | 0.080 (0.00) | 0.020 (0.00) | 0.127 (0.00) |
| | TapTap | 0.030 (0.00) | 0.850 (0.00) | 0.010 (0.00) | 0.180 (0.00) | 0.750 (0.00) | 0.060 (0.00) | **0.000** (0.00) | 0.269 (0.00) |
| | Pred-LLM | 0.041 (0.00) | **-0.024** (0.00) | 0.061 (0.00) | -0.045 (0.00) | **-0.043** (0.00) | 0.016 (0.00) | 0.020 (0.00) | **0.004** (0.00) |
| | SurvivalGAN | 0.006 (0.00) | 0.521 (0.10) | 0.001 (0.00) | 0.052 (0.02) | 0.550 (0.10) | 0.024 (0.02) | 0.012 (0.01) | 0.167 (0.04) |
| | Ashhad | -0.109 (0.00) | 0.295 (0.00) | -0.041 (0.00) | -0.114 (0.00) | 0.051 (0.00) | -0.218 (0.00) | -0.038 (0.00) | -0.025 (0.00) |
| | SurvDiff | 0.153 (0.02) | 0.514 (0.10) | 0.335 (0.01) | 0.080 (0.05) | 0.409 (0.26) | -0.310 (0.30) | -0.010 (0.00) | 0.167 (0.11) |
| | SurvTabGen | 0.008 (0.03) | 0.042 (0.23) | 0.004 (0.01) | 0.007 (0.05) | -0.075 (0.16) | -0.033 (0.05) | 0.016 (0.02) | **-0.004** (0.08) |

### E.2 Survival preservation on subgroups

Since preserving survival patterns for subpopulations is important in some applications, we demonstrate SurvTabGen's performance at subpopulation level.

On the Whas500 dataset, we evaluate three age cohorts: *Young* ($< 65$), *Elderly* (65–74), and *Very Elderly* ($\geq 75$). As shown in Table 5, SurvTabGen accurately preserves subgroup-specific survival distributions, achieving the lowest KM Divergence on the overall population, as well as on the Elderly and Very Elderly subgroups. These results suggest that optimizing the global survival objective also helps preserve subgroup-specific survival patterns.

Table 5: Subgroup-specific KM Divergence (↓) on the Whas500 dataset across age cohorts.

| | Ashhad | SurvivalGAN | SurvDiff | SurvTabGen |
|---|---|---|---|---|
| All Ages | 0.158 (0.00) | 0.092 (0.01) | 0.179 (0.04) | **0.035 (0.02)** |
| Young ($< 65$) | 0.118 (0.01) | **0.063 (0.02)** | 0.283 (0.07) | 0.131 (0.02) |
| Elderly (65–74) | 0.077 (0.00) | 0.121 (0.04) | 0.180 (0.05) | **0.076 (0.04)** |
| Very Elderly ($\geq 75$) | 0.239 (0.00) | 0.131 (0.04) | 0.121 (0.02) | **0.079 (0.01)** |

### E.3 Different values for $\lambda$

We conduct an ablation study on the survival loss weight $\lambda$, which balances the survival loss with the adversarial loss. As shown in Figure 7, increasing $\lambda$ initially improves performance, with KM divergence decreasing from 0.038 to 0.026 as $\lambda$ increases to 0.1, indicating better alignment of temporal dynamics. Importantly, even at $\lambda = 0$, SurvTabGen already outperforms SurvivalGAN, suggesting that generation based on a structured causal process is more effective than multi-stage conditional generation. However, further increasing $\lambda$ degrades performance as the model overemphasizes the survival objective. Overall, SurvTabGen remains stably in a large range of $\lambda \in [0, 0.1]$, where it is always better than SurvivalGAN.

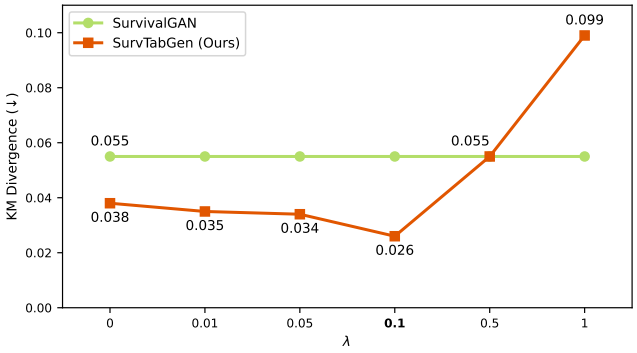



Figure 7: Influence of $\lambda$ on SurvTabGen's performance. The average KM Divergence is computed across all datasets.

### E.4 Different causal discovery methods

Besides the PC algorithm (Spirtes et al., 2000) used in our framework, we explore alternative causal discovery methods such as Greedy Equivalence Search (GES) (Andrews et al., 2023), Fast Causal Inference (FCI) (Spirtes et al., 2013), and Best Order Score Search (BOSS) (Chickering, 2002). As shown in Table 6, PC achieves the lowest average KM divergence but other methods also perform well. Overall, SurvTabGen is robust to different causal discovery methods.

Table 6: Impact of causal discovery method on SurvTabGen's performance. The average KM Divergence is computed across all datasets.

| Metric | Method | AIDS | Cgvhd | Flchain | Gbsg2 | Veterans | Whas500 | AM | AVG |
|---|---|---|---|---|---|---|---|---|---|
| KM Divergence (↓) | BOSS | 0.024 | 0.030 | 0.043 | 0.031 | 0.027 | 0.041 | 0.039 | 0.033 |
| | FCI | **0.023** | 0.031 | 0.075 | 0.030 | 0.034 | 0.048 | **0.031** | 0.039 |
| | GES | 0.029 | 0.028 | 0.031 | 0.028 | 0.027 | **0.038** | 0.041 | 0.032 |
| | PC | 0.026 | **0.023** | **0.016** | **0.026** | **0.020** | 0.043 | 0.040 | **0.026** |

### E.5 Different survival estimators

We replace the default Kaplan-Meier estimator with alternative survival estimators, including Breslow (Fleming and Harrington, 2013), Nelson-Aalen (Aalen et al., 2008), and Weibull (Kalbfleisch and Prentice, 2002) estimators. Any of these estimators can be used to compute the survival function, which defines the reward. As shown in Figure 8, the Kaplan-Meier estimator reaches the lowest KM Divergence while other estimators lead to degraded performance. This highlights the advantage of the Kaplan-Meier estimator as a non-parametric estimator that does not assume a fixed distribution and can accurately handle censored data.

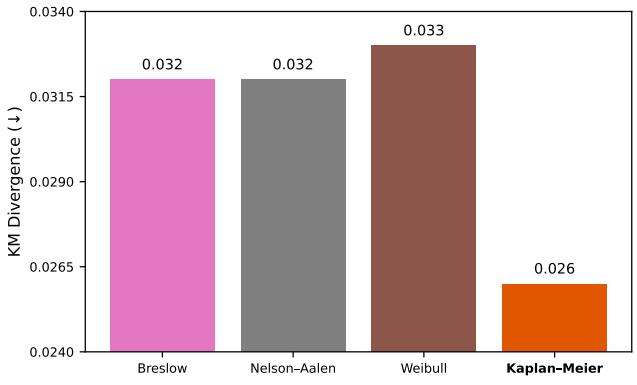


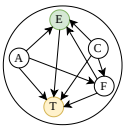
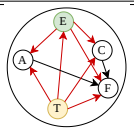
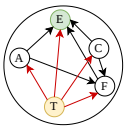
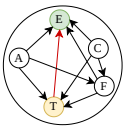
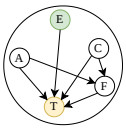
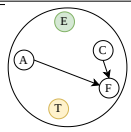

Figure 8: Influence of survival estimator choice on SurvTabGen's performance. The average KM Divergence is computed across all datasets.

### E.6 Robustness to causal graph

To evaluate the robustness of SurvTabGen to causal graph quality, we construct a synthetic dataset (10K samples), which consists of a covariate group $X = \{A, C, F\}$, one event indicator $E$, and one survival time variable $T$. The internal relationships among $A$, $C$, and $F$ are kept fixed to preserve the structure of the covariate space. We then perturb the causal dependencies among $X$, $E$, and $T$ while maintaining the ground-truth survival mechanism $X \to E$ and $(X, E) \to T$.

Table 7 shows that multiple graph variants are generated by progressively degrading the causal structure (e.g., reversing directions, removing edges, or introducing incorrect chains) without altering the underlying data generation process within $X$.

Table 7: Graph-quality spectrum for a five-variable causal graph derived from the synthetic dataset. A, C, F, E, and T denotes Age, Comorbidity, Frailty, Event, and Time, respectively.

| **Accuracy level** | **100%** | **80%** | **60%** | **40%** | **20%** | **0%** |
|---|---|---|---|---|---|---|
| Graph Structure |  |  |  |  |  |  |
| Description | Ground truth - true mechanism | Predicting causes from effect | Effect treated as cause | One wrong chain | Missed one parent | No edge found |

We evaluate the impact of these perturbations in Figure 9. It shows that when the causal graph is closer to the ground truth (i.e., *graph-accuracy level approaches 100%*), KM Divergence is significantly reduced. In contrast, distorted graph structures lead to degradation in KM Divergence. However, SurvTabGen's performance remains highly robust across a wide spectrum of graph-accuracy levels, even when the graph accuracy drops to 20%.

These results suggest that accurate modeling of the dependencies between $X$, $E$, and $T$ is critical for preserving survival dynamics. More importantly, it also demonstrates the robustness of SurvTabGen under varying causal graph quality.

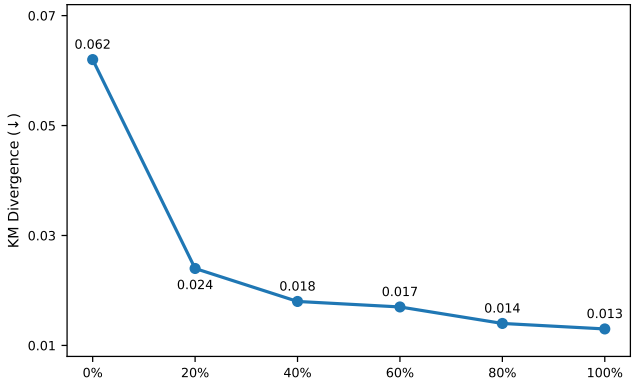



Figure 9: SurvTabGen's performance on a synthetic dataset under perturbed causal structures.

### E.7 Downstream performance

We train three models Cox Proportional Hazards (CoxPH) (Cox, 1972), SurvivalXGBoost (Barnwal et al., 2022), and Random Survival Forest (Ishwaran et al., 2008) on the synthetic data and evaluate them on real held-out data. To align with SurvivalGAN, Ashhad, and SurvDiff evaluation protocols, we use a 67/33 data split (i.e., 2-fold for training and 1-fold for testing) across all datasets, except for the small dataset Cgvhd, where we use a 75/25 split to ensure sufficient training samples. The hyperparameters for these survival models are set exactly as detailed in SurvivalGAN and Ashhad. Performance is measured using *C-index* (↑) (Harrell et al., 1982) and *Brier Score* (↓) (Glenn, 1950), reporting the best performance achieved across the three models.

As shown in Table 8, SurvTabGen consistently achieves strong performance under both metrics, remaining competitive with SurvDiff in C-index and TapTap in Brier Score.

Table 8: C-index and Brier Score metrics showing downstream utility performance of each method across datasets. "*Real Data*" indicates the survival models are trained directly on real data.

| Metrics | Methods | AIDS | Cgvhd | Flchain | Gbsg2 | Veterans | Whas500 | AM | AVG |
|---|---|---|---|---|---|---|---|---|---|
| C-index (↑) | *Real Data* | *0.738 (0.04)* | *0.500 (0.00)* | *0.934 (0.00)* | *0.672 (0.02)* | *0.700 (0.03)* | *0.769 (0.01)* | *0.794 (0.03)* | *0.729* |
| | CTGAN | 0.567 (0.08) | 0.549 (0.09) | 0.929 (0.00) | 0.498 (0.11) | 0.524 (0.09) | 0.633 (0.06) | 0.590 (0.10) | 0.613 |
| | CA-GAN | 0.550 (0.11) | 0.501 (0.11) | 0.758 (0.01) | 0.516 (0.11) | 0.525 (0.06) | 0.483 (0.11) | 0.635 (0.09) | 0.567 |
| | TVAE | 0.697 (0.03) | 0.525 (0.07) | 0.927 (0.00) | 0.647 (0.02) | 0.613 (0.06) | **0.746** (0.03) | 0.722 (0.02) | 0.697 |
| | TabDiff | 0.479 (0.13) | 0.513 (0.05) | 0.547 (0.20) | 0.572 (0.08) | 0.546 (0.10) | 0.562 (0.07) | 0.484 (0.05) | 0.529 |
| | GReaT | 0.444 (0.07) | 0.500 (0.00) | 0.929 (0.00) | 0.549 (0.05) | 0.581 (0.04) | 0.642 (0.07) | 0.634 (0.06) | 0.611 |
| | TapTap | 0.608 (0.12) | 0.523 (0.05) | 0.930 (0.00) | 0.571 (0.06) | 0.540 (0.05) | 0.642 (0.07) | 0.688 (0.02) | 0.643 |
| | Pred-LLM | 0.674 (0.05) | **0.584** (0.12) | 0.932 (0.00) | 0.580 (0.03) | 0.500 (0.00) | 0.639 (0.03) | **0.747** (0.02) | 0.665 |
| | SurvivalGAN | 0.631 (0.02) | 0.501 (0.09) | 0.928 (0.00) | 0.652 (0.03) | **0.656** (0.06) | 0.734 (0.02) | 0.673 (0.08) | 0.682 |
| | Ashhad | 0.532 (0.09) | 0.500 (0.00) | 0.894 (0.03) | 0.442 (0.03) | 0.541 (0.08) | 0.571 (0.09) | 0.604 (0.17) | 0.583 |
| | SurvDiff | **0.710** (0.03) | 0.500 (0.00) | **0.937** (0.00) | **0.665** (0.02) | 0.655 (0.08) | 0.742 (0.03) | 0.742 (0.03) | **0.711** |
| | SurvTabGen | 0.707 (0.05) | 0.551 (0.09) | 0.929 (0.00) | 0.656 (0.02) | 0.641 (0.03) | 0.728 (0.01) | 0.711 (0.05) | 0.703 |
| Brier Score (↓) | *Real Data* | *0.060 (0.00)* | *0.519 (0.21)* | *0.102 (0.00)* | *0.271 (0.01)* | *0.116 (0.02)* | *0.179 (0.01)* | *0.058 (0.00)* | *0.186* |
| | CTGAN | 0.065 (0.00) | 0.168 (0.03) | 0.054 (0.00) | 0.267 (0.02) | 0.203 (0.08) | 0.244 (0.02) | 0.169 (0.01) | 0.167 |
| | CA-GAN | **0.058** (0.00) | 0.296 (0.07) | 0.124 (0.01) | 0.193 (0.01) | 0.345 (0.12) | 0.265 (0.03) | 0.099 (0.01) | 0.197 |
| | TVAE | 0.067 (0.00) | 0.152 (0.03) | 0.099 (0.00) | 0.183 (0.01) | 0.231 (0.10) | **0.187** (0.01) | 0.068 (0.01) | 0.141 |
| | TabDiff | 0.066 (0.00) | 0.158 (0.06) | 0.142 (0.00) | 0.202 (0.02) | 0.221 (0.12) | 0.287 (0.01) | 0.149 (0.02) | 0.175 |
| | GReaT | 0.067 (0.00) | 0.521 (0.21) | **0.048** (0.00) | 0.199 (0.01) | 0.144 (0.02) | 0.230 (0.02) | 0.080 (0.01) | 0.184 |
| | TapTap | 0.064 (0.00) | 0.122 (0.01) | **0.048** (0.00) | 0.195 (0.01) | 0.151 (0.04) | 0.219 (0.00) | 0.080 (0.00) | **0.126** |
| | Pred-LLM | 0.066 (0.00) | **0.120** (0.01) | 0.100 (0.00) | 0.262 (0.00) | 0.625 (0.03) | 0.217 (0.01) | 0.071 (0.01) | 0.209 |
| | SurvivalGAN | 0.067 (0.01) | 0.160 (0.07) | 0.049 (0.00) | **0.178** (0.00) | 0.191 (0.05) | 0.198 (0.01) | 0.181 (0.02) | 0.146 |
| | Ashhad | 0.066 (0.00) | 0.525 (0.20) | 0.105 (0.01) | 0.197 (0.01) | 0.143 (0.03) | 0.236 (0.00) | 0.087 (0.01) | 0.194 |
| | SurvDiff | 0.066 (0.01) | 0.525 (0.20) | 0.108 (0.00) | 0.270 (0.01) | **0.135** (0.03) | 0.189 (0.01) | **0.063** (0.01) | 0.194 |
| | SurvTabGen | 0.061 (0.00) | 0.129 (0.02) | 0.050 (0.00) | 0.272 (0.01) | 0.144 (0.03) | 0.199 (0.01) | 0.083 (0.01) | 0.134 |

## E.8 COMPUTATIONAL EFFICIENCY

Figure 10 presents the average wall-clock runtime (*in minutes*) of each method across seven datasets. All runtime measurements are conducted on a machine with an NVIDIA Tesla V100 GPU, 8 CPU cores, and 64GB RAM. SurvTabGen is much faster than SurvivalGAN (**1.7× faster**) and SurvDiff (**8.6× faster**).

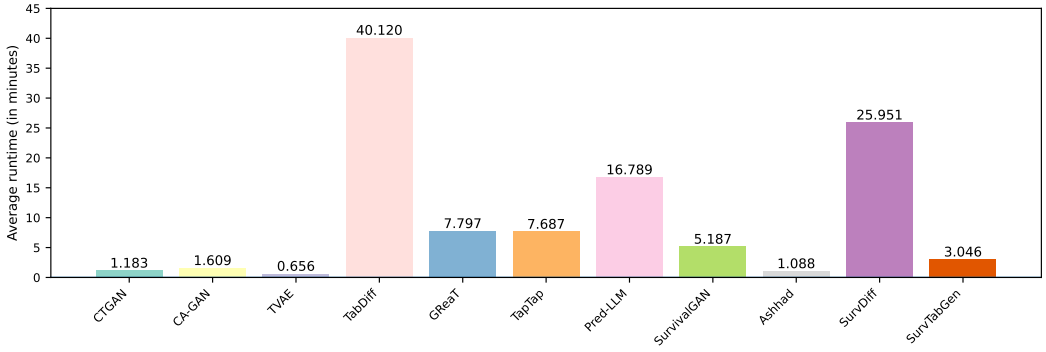



Figure 10: Average runtime (*in minutes*) of each method across all datasets.