DIFFSURV: DIFFERENTIABLE SORTING FOR CENSORED TIME-TO-EVENT DATA.

Andre Vauvelle[1], Benjamin Wild[2], Aylin Cakiroglu[3], Roland Eils[2], Spiros Denaxas[1]
University College London[1], Berlin Institute of Health[2], BenevolentAI[3],
{rmhivau,s.denaxas}@ucl.ac.uk, aylin.cakiroglu@benevolent.ai,
{benjamin.wild, roland.eils}@bih-charite.de

ABSTRACT

Survival analysis is a crucial semi-supervised task in machine learning with numerous real-world applications, particularly in healthcare. Currently, the most common approach to survival analysis is based on Cox's partial likelihood, which can be interpreted as a ranking model optimized on a lower bound of the concordance index. This relation between ranking models and Cox's partial likelihood considers only pairwise comparisons. Recent work has developed differentiable sorting methods which relax this pairwise independence assumption, enabling the ranking of sets of samples. However, current differentiable sorting methods can not account for censoring, a key factor in many real-world datasets. To address this limitation, we propose a novel method called *Diffsurv*. We extend differentiable sorting methods to handle censored survival analysis tasks by predicting matrices of possible permutations that take into account the uncertainty introduced by censored samples. We contrast this approach with methods derived from partial likelihood and ranking losses. Our experiments show that Diffsurv outperforms established baselines in various simulated and real-world risk prediction scenarios. Additionally, we demonstrate the benefits of the algorithmic supervision enabled by Diffsurv by presenting a novel method for top-k risk prediction that outperforms current methods.

1 INTRODUCTION AND BACKGROUND

Survival analysis is an important task in numerous machine learning applications, particularly in the healthcare domain. The goal of survival analysis is to predict the time until the occurrence of an event of interest, such as death or disease onset, based on a set of covariates. In clinical studies, these covariates typically include demographic variables such as sex and age, but may also encompass more complex data modalities such as temporal streams or medical images.

However, event times may not be observed due to censoring, especially in observational datasets where many patients may not have experienced the event at the time of data collection. Ignoring censoring can lead to biased predictions towards the censoring event instead of the event of interest. For example, if the end of the study can be determined from the observed covariates, especially if age is included, the predicted event times will be skewed towards the censoring event time instead of the actual event of interest Kvamme & Borgan (2019).

The Cox Proportional Hazards (PH) model is widely used for handling censored data in survival analysis (Cox, 1972). The model optimizes a partial likelihood function over ranked data, considering only the order of events, not their exact time of occurrence. As such, Cox's partial likelihood serves as a ranking loss, learning from the order of patients based on their hazard of experiencing an event, not their exact survival time. Raykar et al. (2007) showed that Cox PH and ranking models can be directly equated, with both providing lower bounds to the concordance index, the primary evaluation metric used in survival analysis. A key step in relating the two models assumes only pairwise comparisons or risk sets of size 2. Goldstein & Langholz (1992) show that sub-sampling risk sets produce consistent parameter estimators but that greater risk sets provide more efficient estimators. Cox's partial likelihood and ranking losses underpin current survival analysis methods in deep learning, including DeepSurv Katzman et al. (2018) and DeepHit Lee et al. (2018).

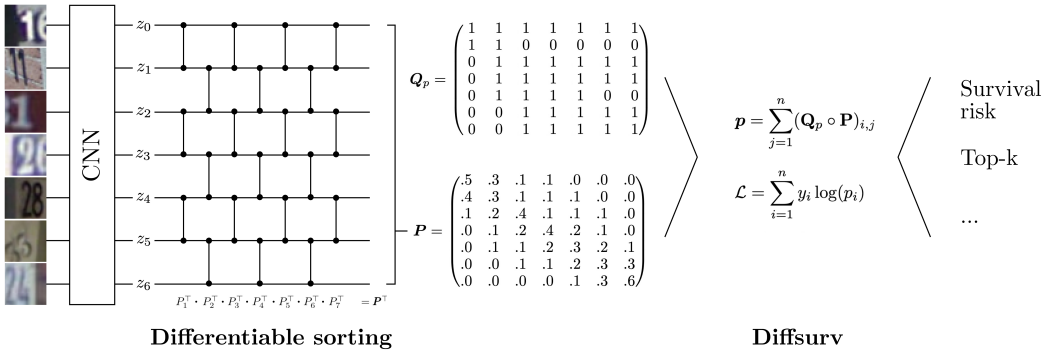


Figure 1: Differentiable Sorting for Censored Time-to-Event Data. Inputs, in this case, SVHN images, are transformed into scalar values through a neural network. A differentiable permutation matrix, $\boldsymbol{P}$, is computed using sorting networks. The model can be optimized for downstream tasks, such as risk stratification and top-k highest risk prediction, by using the matrix $\boldsymbol{Q}_p$ of possible permutations based on the observed events and censoring.

We present an alternative method that leverages recent advancements in continuous relaxations of sorting operations, enabling end-to-end training of neural networks with ordering supervision. (Grover et al., 2019; Blondel et al., 2020; Petersen et al., 2021). This involves incorporating a sorting algorithm into the network architecture, where the order of the samples is known, but their exact values are unsupervised. Here, we introduce *Diffsurv*, an extension of differentiable sorting methods that enables end-to-end training of survival models with censored data.

Briefly, our contributions are summarised:

- Our primary contribution is the extension differentiable sorting methods to account for censoring by introducing the concept of possible permutation matrices.
- We empirically demonstrate that our new differentiable sorting method improve risk ranking performance across multiple simulated and real-worlds censored datasets.
- We demonstrate that differentiable sorting of censored data enables the development of new methods with practical applications, using the example of end-to-end learning for top-k risk stratification.

2 Methods

A dataset with censored event times is summarized as $\mathcal{D} = \{t_i, \boldsymbol{x}_i, \delta_i\}_{i=1}^N$. For a patient i, t_i is the observed minimum of the unobserved true survival time t_i^* and the censoring time c_i^*, δ_i is the event indicator that is 1 if an event is observed ($t_i^* \leq c_i^*$) or 0 if the data is censored ($t_i^* > c_i^*$). Covariates are $\boldsymbol{x}_i \in \mathbb{R}^d$ representing a 1-dimensional vector of size d or larger dimensional tensors such as image data. N is the total number of patients. As previously mentioned, it is common to subsample total possible risk set, we use n to represent the subsampled risk set size.

In order to train models based on ordering information using differentiable sorting algorithms Petersen (2022), we can minimize the cross-entropy between the ground truth orders represented by true permutation matrix $\boldsymbol{Q}$ and a doubly-stochastic predicted permutation matrix $\boldsymbol{P}$. This makes it possible to interpret each element P_{ij} of the predicted permutation matrix as the predicted probability of permuting from a randomly assigned rank i to a true rank j.

There are multiple methods of relaxing sorting algorithms to produce $\boldsymbol{P}$, we will follow Petersen et al. (2021) by using differentiable sorting networks. Sorting networks are a family of sorting algorithms that consist of two basic components: wires and conditional swaps. Wires carry values to be compared at conditional swaps, if one value is bigger than the other, the values carried forward are swapped around. For a random sample of patients to be ordered, each layer of the sorting network can be considered an independent permutation matrix $\boldsymbol{P}_l$ with elements given by

$$P_{l,ii} = P_{l,jj} = \sigma(z_j - z_i) \text{ and } P_{l,ij} = P_{l,ji} = 1 - \sigma(z_j - z_i). \tag{1}$$

These elements represent conditional swaps between two patient risk values (z_i, z_j) and use a differentiable relaxation of the step function such as the logistic-sigmoid, where $\sigma : x \rightarrow \frac{1}{1+e^{-\beta x}}$. The inverse temperature parameter $\beta > 0$ is introduced so when $\beta \rightarrow \infty$ the functions tend to the exact $\min$ and $\max$ functions. The indices being compared are determined by the sorting network and the final predicted probability matrix is the product of each layer of sorting operations, $\boldsymbol{P} = (\prod_{l=1}^{n} \boldsymbol{P}_l^{\intercal})^{\intercal}$. For the base case $n = 2$, Diffsurv is equivalent to the pairwise ranking loss and Cox partial likelihood. Further details on the relations between Diffsurv and baselines is in Appendix A.2.1.

The introduction of censored patients means we no longer have access to a ground truth permutation matrix $\mathbf{Q}$. We cannot determine the exact rank of patients who are censored before another who experienced an event. To address this challenge, we propose a novel extension of differentiable sorting to censored data. Our approach considers the set of *possible permutations* for each patient, taking into account uncertainty about the true ranking. In Figure 3, we show an example of observed and censored events and the resulting set of possible permutations that can be represented as a permutation matrix $\boldsymbol{Q}_p$.

For a right-censored sample i, we only know that the rank must be lower than the rank of all other samples with an event time lower than the censoring time of t_i, i.e. they must be ranked after prior events. For another sample j with an event at t_j, we know that the rank must be lower than the rank of other samples with an event time lower than t_j, and higher than the rank of other samples either with an event time higher than t_j or with a censoring time higher than t_j. We do not know how the rank of j compares to samples with censoring time lower than t_j. If it is possible for patient i to permute to rank j, then $Q_{pij} = 1$.

Given the possible permutation matrix $\mathbf{Q}_p$ and the predicted permutation matrix $\mathbf{P}$, the vector of probabilities $\boldsymbol{p}$ of a patient being ranked within the set of possible permutations can be computed. Although the ground truth probabilities are unknown, the range of possible ranks is known, and the model can be optimized to maximize the sum of the predicted probabilities of all possible ranks for each sample. Noted here as the column-sum of the element-wise product $\circ$, between $\mathbf{Q}_p$ and $\mathbf{P}$.

$$\boldsymbol{p} = \sum_{j=1}^{n} (\mathbf{Q}_p \circ \mathbf{P})_{i,j}. \tag{2}$$

The cross-entropy loss can then be easily applied

$$\mathcal{L} = \sum_{i=1}^{n} y_i \log(p_i) \tag{3}$$

where y_i is the true label of the set of possible ranks.

Finally, we demonstrate how the algorithmic supervision of sorting algorithms enables the development of novel methods in survival analysis, using the example of top-k risk prediction. In practical settings, it is often not necessary to rank all samples correctly. Rather, it is essential to identify the samples with the highest risk, such as by a healthcare provider, to prioritize care and interventions. With Diffsurv, top-k risk prediction is straightforward to implement by optimizing possible permutations within the top-k ranks, whereby $\boldsymbol{Q}_p$ is adjusted such that only the top-k patient's possible permutations are set to 1.

3 Experiments

We evaluate the performance of Diffsurv on censored survival data across semi-synthetic and real-world datasets. In each experiment, we train a neural network using Diffsurv and the partial likelihood loss, then compare their respective results. The partial likelihood and closely related ranking loss are used in popular baselines; Deepsurv (Katzman et al., 2018), Cox-MLP (Kvamme et al., 2019) and DeepHit Lee et al. (2018).

We present a new semi-synthetic dataset, *survSVHN*, to evaluate survival models. Based on the Street View House Numbers (SVHN) dataset Petersen et al. (2021), we simulate survival times akin to survMNIST Pölsterl (2019). The increased complexity of SVHN offers a testbed which is better able to discern the performance differences between methods. Each house number parameterizes an

Table 1: Results for semi-synthetic and real-world datasets. Bold indicates significantly higher performance (t-test with a significance level of 0.01).

(a) Semi-synthetic survSVHN Dataset Results. Mean (and standard deviation) over 5 trails with different seeds. Metric is C-index. † When $n = 2$ both methods are equivalent to the ranking loss.

Method	n=2†	n=4	n=8	n=16	n=32
Diffsurv	.918 (.003)	**.934** (.002)	**.940** (.001)	**.943** (.002)	**.941** (.002)
Cox Partial Likelihood	.913 (.002)	.925 (.002)	.931 (.002)	.933 (.002)	.930 (.003)

(b) Real-world datasets results. Mean (and standard deviation) over 5-folds measured in C-Index for the risk stratification task and proportion correctly predicted for the Top-k task.

	FLCHAIN	NWTCO	SUPPORT	METABRIC
Size	6,524	4,028	8,873	1,904
Censored Proportion	69.9%	85.8%	32.0%	42.1%
Risk Stratification				
Diffsurv	.787 (.012)	.691 (.018)	**.599** (.004)	**.623** (.012)
Cox Partial Likelihood	.787 (.016)	.690 (.014)	.584 (.004)	.615 (.018)
Top 10% prediction				
Diffsurv (Top-k)	**.952** (.014)	.926 (.007)	**.571** (.012)	**.771** (.059)
Cox Partial Likelihood	.937 (.014)	.919 (.026)	.478 (.032)	.561 (.039)

exponential time function for survival times. Risks are calculated as the logarithm of house numbers, standardized and scaled for a mean survival time of 30. We introduce censoring by randomly selecting 30% of house numbers and replacing true times with values sampled uniformly between $(0, t_i]$ (See Figure 5). A convolutional neural network with the same hyperparameters as Petersen et al. (2021) is used, with $z_i = f_{\text{CONV}}(\boldsymbol{x}_i)$. We also evaluate on four real-world healthcare datasets from Kvamme et al. (2019). Each dataset has a fairly small number of patients ($N \leq 8,873$) and a flat vector of covariates as input. Further details in Appendix A.6. For these datasets, a fully connected neural network is used to find the risk, $z_i = f_{\text{MLP}}(\boldsymbol{x}_i)$. Further details on the training and evaluation procedures can be found in Appendix A.4.

The results presented in Table 1 demonstrates that Diffsurv achieves equal to or better performance on all datasets analyzed. Additionally, when Diffsurv is optimized for predicting the top 10% of highest risk individuals, it outperforms Cox's partial likelihood on all four datasets. There is a significant improvement in the top 10% highest-risk prediction when comparing models based on Cox's Partial Likelihood ($\mu = .825$, $\sigma = .005$) and Diffsurv optimized for Top-k prediction ($\mu = .944$, $\sigma = .008$) on the survSVHN dataset.

4 Conclusion

Diffsurv represents a significant step in the field of survival analysis with censored data. Our experiments demonstrate the effectiveness of differentiable sorting methods in improving survival analysis predictions, particularly in censored datasets with Diffsurv matching or improving perfomance against Cox PL on all datasets. Additionally, Diffsurv has the potential to drive the development of new methods, such as the top-k risk stratification method presented in this work. It is noteworthy that while our method has shown promising results, further investigation is necessary to fully understand its potential and limitations. For instance, it would be valuable to examine the scalability of the method with larger real-world datasets and its capability to handle more complex censored scenarios. Further research could also investigate the integration of Diffsurv into clustering models. With its ability to handle censored data and its end-to-end training capability, Diffsurv presents a promising approach to survival analysis and holds great potential for enhancing risk prediction in real-world applications.

REFERENCES

Mathieu Blondel, Olivier Teboul, Quentin Berthet, and Josip Djolonga. Fast Differentiable Sorting and Ranking, June 2020. URL http://arxiv.org/abs/2002.08871. arXiv:2002.08871 [cs, stat].

D. R. Cox. Regression Models and Life-Tables. *Journal of the Royal Statistical Society. Series B (Methodological)*, 34(2):187–220, 1972. ISSN 0035-9246. URL https://www.jstor.org/stable/2985181. Publisher: [Royal Statistical Society, Wiley].

Larry Goldstein and Bryan Langholz. Asymptotic Theory for Nested Case-Control Sampling in the Cox Regression Model. *The Annals of Statistics*, 20(4):1903–1928, December 1992. ISSN 0090-5364, 2168-8966. doi: 10.1214/aos/1176348895. URL https://projecteuclid.org/journals/annals-of-statistics/volume-20/issue-4/Asymptotic-Theory-for-Nested-Case-Control-Sampling-in-the-Cox/10.1214/aos/1176348895.full. Publisher: Institute of Mathematical Statistics.

Aditya Grover, Eric Wang, Aaron Zweig, and Stefano Ermon. Stochastic Optimization of Sorting Networks via Continuous Relaxations. *arXiv:1903.08850 [cs, stat]*, April 2019. URL http://arxiv.org/abs/1903.08850. arXiv: 1903.08850.

Frank E. Harrell, Jr, Robert M. Califf, David B. Pryor, Kerry L. Lee, and Robert A. Rosati. Evaluating the Yield of Medical Tests. *JAMA*, 247(18):2543–2546, May 1982. ISSN 0098-7484. doi: 10.1001/jama.1982.03320430047030. URL https://doi.org/10.1001/jama.1982.03320430047030.

Jared L. Katzman, Uri Shaham, Alexander Cloninger, Jonathan Bates, Tingting Jiang, and Yuval Kluger. DeepSurv: personalized treatment recommender system using a Cox proportional hazards deep neural network. *BMC Medical Research Methodology*, 18(1):24, February 2018. ISSN 1471-2288. doi: 10.1186/s12874-018-0482-1. URL https://doi.org/10.1186/s12874-018-0482-1.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*, January 2017. URL http://arxiv.org/abs/1412.6980. arXiv: 1412.6980.

Håvard Kvamme and Ørnulf Borgan. The Brier Score under Administrative Censoring: Problems and Solutions. Technical Report arXiv:1g912.08581, arXiv, December 2019. URL http://arxiv.org/abs/1912.08581. arXiv:1912.08581 [cs, stat] type: article.

Håvard Kvamme, Ørnulf Borgan, and Ida Scheel. Time-to-Event Prediction with Neural Networks and Cox Regression. *arXiv:1907.00825 [cs, stat]*, September 2019. URL http://arxiv.org/abs/1907.00825. arXiv: 1907.00825.

Changhee Lee, William Zame, Jinsung Yoon, and Mihaela Van Der Schaar. DeepHit: A Deep Learning Approach to Survival Analysis with Competing Risks. *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 8, 2018.

Felix Petersen. Learning with Differentiable Algorithms, September 2022. URL http://arxiv.org/abs/2209.00616. arXiv:2209.00616 [cs].

Felix Petersen, Christian Borgelt, Hilde Kuehne, and Oliver Deussen. Differentiable Sorting Networks for Scalable Sorting and Ranking Supervision. *arXiv:2105.04019 [cs]*, July 2021. URL http://arxiv.org/abs/2105.04019. arXiv: 2105.04019.

Sebastian Pölsterl. Survival Analysis for Deep Learning, July 2019. URL https://k-d-w.org/blog/2019/07/survival-analysis-for-deep-learning/.

Vikas C Raykar, Harald Steck, Balaji Krishnapuram, Cary Dehing-oberije, and Philippe Lambin. On Ranking in Survival Analysis: Bounds on the Concordance Index. In *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007. URL https://papers.nips.cc/paper/2007/hash/33e8075e9970de0cfea955afd4644bb2-Abstract.html.

A APPENDIX

A.1 COX'S PARTIAL LIKELIHOOD AND RANKING LOSSES

Cox (1972) introduced the most popular method for addressing censoring for survival analysis. The original work and many subsequent extensions seek to maximize the partial likelihood, the general form is described as

$$\mathcal{L}(\theta) = \prod_{i:\delta=1} \frac{f_\theta(\boldsymbol{x}_i)}{\sum_{j:T_j>T_i} f_\theta(\boldsymbol{x}_j)}, \tag{4}$$

where f_θ is the hazard function, a real valued score prediction function estimating the probability of an event at particular time, given input features $\boldsymbol{x}_i$. The product is taken over the set of uncensored patients, while the denominator term considers only comparable pairs and includes censored patients with $T_j > T_i$.

The classic Cox proportional hazards model used $f_\theta = \exp(\theta \cdot \boldsymbol{x}_i)$. Multiple extensions of this basic loss relax the linear covariate interaction and proportional hazards assumptions by altering f_θ. For example, (Katzman et al., 2018) parameterized with a neural network h_θ, such that $f_\theta = \exp(h_\theta(\boldsymbol{x}_i))$, to model non-linear interactions between covariates on the hazard. (Kvamme et al., 2019) further show that this can be extended to non-proportional hazards by introducing temporal covariates, $f_\theta = \exp(h_\theta(\boldsymbol{x}_i, T_i))$.

Kvamme et al. (2019) also make a few adjustments to the original partial likelihood loss. First, they consider that the risk set $\mathcal{R} = \{j : T_j > T_i\}$ is intractable for deep learning applications as it considers all comparable patients. Instead, it is possible take a fixed size sample risk set $|\tilde{\mathcal{R}}| = n < N$ and further, it is reasonable to take a constant sample size of 1 and include the individual i in the risk set (such that n=2). This leads to the simplified loss of the form

$$\mathcal{L}(\theta) = \prod_{i:\delta=1} \frac{f_\theta(\boldsymbol{x}_i)}{f_\theta(\boldsymbol{x}_i) + f_\theta(\boldsymbol{x}_j)},\ j \in \mathcal{R} \setminus \{i\}. \tag{5}$$

Further, we can take the mean log partial likelihood to be

$$\text{loss} = \frac{1}{n_e} \sum_{i:\delta=1} \log(1 + \exp[h_\theta(\boldsymbol{x}_j) - h_\theta(\boldsymbol{x}_i)]),\ j \in \mathcal{R} \setminus \{i\}, \tag{6}$$

where n_e is the number of non-censored events. From this simplified form, it can be seen that the partial likelihood only considers the relative ordering or ranking of survival times.

The concordance index or c-index Harrell et al. (1982) is a commonly used as an evaluation for survival analysis methods and is a generalization of the Area Under the Receiver Operating Characteristic Curve (AUROC) that handles right-censored data.

$$\text{c-index} = \frac{1}{n} \sum_{i:\delta=1} \mathbb{1}(f(\boldsymbol{x}_i) < f(\boldsymbol{x}_j)), j \in \mathcal{R} \setminus \{i\}. \tag{7}$$

Raykar et al. (2007) showed that the Cox's partial likelihood is approximately equivalent to maximizing the concordance index or C-index and that closer bounds can be found by maximizing the general ranking loss

$$\text{ranking-loss} = \frac{1}{|\mathcal{A}|} \sum_{(\boldsymbol{x}_i, \boldsymbol{x}_j) \in \mathcal{A}} \phi(f_\theta(\boldsymbol{x}_i) - f_\theta(\boldsymbol{x}_j)), \tag{8}$$

where ϕ is a function that relaxes the non-differentiable $\mathbb{1}$ of the C-index. We have also introduced $\mathcal{A}$ as the graph of acceptable pairs, where each node is a patient that can only be linked to another with an edge if we are sure that the first event occurs before the second. This is another way of describing the risk sets introduced earlier. From Equation 6 is can be seen that $\phi : x \to -\log(1 + \exp(-x)) =$

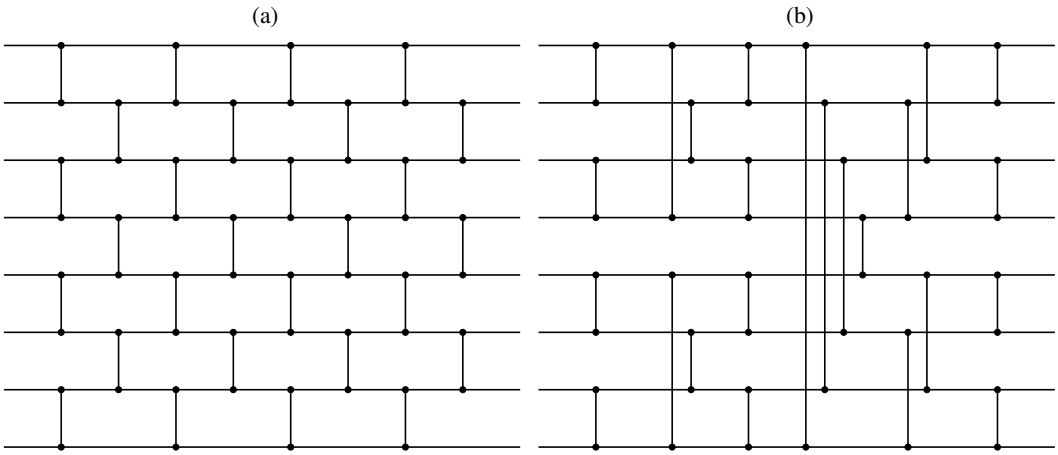


Figure 2: Example sorting networks of size 8; (a) Odd-Even, (b) Bitonic.

$\log(\sigma(x))$. Here, we have shown that the simplifications to the partial likelihood made by Kvamme et al. (2019) are equivalent to using the log-sigmoid ranking loss.

The key difference between ranking and partial likelihood losses comes when considering the assumption that it is reasonable to take a constant sample size of 1 (one pair in the risk set) in the partial likelihood. This effectively introduces the assumption that each pair (i, j) is independent of any other pair. However, this assumption seems puzzling given the inherent transitivity of ranking (if $i > j$ and $j > k$ then $i > k$).

A.2 Differentiable Sorting

Differentiable sorting takes a different approach to the previously discussed partial likelihood loss and will need modification to account for censoring. In order train models based on ordering information, differences between predicted and true orderings are backpropagated through a relaxed sorting algorithm. The idea was first introduced by (Grover et al., 2019), the key motivation being that sorting is a key step in many classical algorithms and machine learning methods (K-nearest neighbours), yet is non-differentiable. This means that direct supervision cannot be done on the outputs of algorithms that rely on sorting (Petersen, 2022).

Sorting algorithms require use of non-differentiable $\max$ and $\min$ operators. These are analogous to the non-differentiable indicator function that was discussed earlier in the c-index equation Equation(7). Differentiable sorting methods similarly rely on approximating these operators with smooth alternatives.

Petersen et al. (2021) propose combining traditional sorting networks and differentiable sorting functions. Sorting networks are a family of sorting algorithms that consist of two basic components: wires and conditional swaps. Wires carry values to be compared at conditional swaps, if one value is bigger than the other then the values carried forward are swapped around. This allows construction of provably guaranteed sorting networks. Conditional swaps are exactly the min and max operators that ensure that with inputs $\{a, b\}$ and outputs $a^* \leq b^*$, $a^* = \min(a, b)$ and $b^* = \max(a, b)$. Note that as a approaches b, the point at which it becomes larger is discontinuous and hence non-differentiable. Just as previously shown in the ranking loss, such operations can be made differentiable using the logistic relaxation

$$\min_\sigma(a, b) = a \cdot \sigma(b - a) + b \cdot \sigma(a - b) \text{ and } \max_\sigma(a, b) = a \cdot \sigma(a - b) + b \cdot \sigma(b - a). \quad (9)$$

Note that if an inverse temperature parameter $\beta > 0$ is introduced such that $\sigma : x \to \frac{1}{1+e^{-\beta x}}$, then as $\beta \to \infty$ the functions tend to the exact $\min$ and $\max$ functions. Other relaxation of the step function can also be considered, Petersen et al. (2021) show that the Cauchy distribution preserves monotonicity which is desirable for optimization. Given this, we use the Cauchy distribution as our relaxation for all experiments, where $\sigma : x \to \frac{1}{\pi}\text{arctan}(\beta x) + \frac{1}{2}$.

There are multiple different types of sorting networks each with varying space complexity. The ability to implement networks with the divide-and-conquer paradigm allows for sorting networks that scale more efficiently than previous differentiable sorting methods. In particular, Petersen et al. (2021) uses the odd-even and bitonic sorting networks. The latter allows construction of networks with size complexity $\mathcal{O}(nlog^2n)$ verses the $\mathcal{O}(n^2)$ in previous differentiable sorting methods. Examples for Odd-Even and Bitonic sorting networks with $n = 8$ are shown in Figure 2.

A.2.1 Relation to ranking and partial likelihood

Differentiable sorting has so far only been applied in the context of uncensored ranking, where we know the true rank for every instance in an input set. It is possible to directly relate Diffsurv with ranking losses. Expanding out the cross entropy loss out we find

$$\mathcal{L} = \sum_{c}^{n} \left(\frac{1}{n} \sum_{i}^{n} q_{ci} \log(p_{ci}) \right), \tag{10}$$

where $q_{ci} = 1$ only when i is the true rank otherwise 0. Each p_{ci} is always a function of the difference in pairs of inputs x_i and x_j. This is complicated by the products of intermediate values a_i introduced by the sorting network but denoted as

$$p_{ci} = \prod_{(a_i,a_j)\in\mathcal{P}_l:l=1}^{n} \sigma(f_\theta(a_i) - f_\theta(a_j)) \tag{11}$$

where $\mathcal{P}_l$ to denotes the set of comparisons to be made at each layer of the sorting network. With risk set of size 2, the loss returns to the same recognisable log-sigmoid ranking loss, and Cox negative log partial likelihood with risk set size 2.

A.3 Non-proportional Hazards

Our current implementation of Diffsurv is limited to proportional hazards, which may not fully capture the complexity of certain survival analysis problems, particularly when non-proportional hazards are present. In this context, we explore alternative approaches that address this limitation.

Previously, we briefly mentioned continuous-time extensions of partial likelihood to enable non-proportional hazards Kvamme et al. (2019). This can be achieved by directly modeling temporal covariates as $f_\theta = \exp(h_\theta(x_i, T_i))$.

Another class of methods focuses on discretizing the time-to-event variable and modeling the probability mass function (PMF) of event times. For instance, the DeepHit model Lee et al. (2018) employs a neural network architecture to learn the relationships between input features and discretized time-to-event outcomes. Time discretization facilitates modeling of non-proportional hazards but introduces two significant challenges: 1) sensitivity to the choice of time intervals, which can affect the model's accuracy and interpretability, and 2) increased computational complexity, as predictions must be made for each time interval. These models can be computationally expensive, especially for deep learning-based models like DeepHit, making them less suitable for high-dimensional and large-scale datasets, such as the imaging dataset used in this study.

Several future work proposals arise from these observations. First, differentiable sorting could explore the approach of directly modeling temporal covariates, resulting in a time-parameterized predicted permutation matrix. Second, extending Diffsurv to discrete time could be achieved by parameterizing a predicted permutation matrix for each time discretization. Finally, in models like DeepHit, the ranking loss term could be replaced with a Diffsurv loss, offering another promising direction for future research in survival analysis.

A.4 Training and evaluation

During training, we use the Adam optimizer Kingma & Ba (2017) with a learning rate of 10^{-3}, early stopping with patience of 10 epochs and a maximum of 10^5 training steps. As in Goldstein & Langholz (1992) and Kvamme et al. (2019), we ensure that each risk set contains a valid risk set by

sampling controls for a given case. Each batch consists of a number of risk sets such that the input data has shape (batch size, risk set size, covariate shape).

For the survSVHN task, the hazard function f_θ for both the Cox Partial Likelihood baseline and Diffsurv a fixed sized convolutional neural network with a batch size of 100. Steepness is determined by risk set size n, $\beta = 2n$ for Odd-even networks and $\beta = \log_2(n)(1 + \log_2(n))$ for Bitonic. The only hyperparameter optimized for Diffsurv is the choice of sorting network. Both sorting network were evaluated on the validation set but only the resulting models with higher validation c-index were used on the test set. This was repeated for each risk set size.

For the real-world datasets, the hazard function f_θ is Multi-layer Perceptron network. Hyperparameters: number of hidden layers, size of hidden layers, dropout rate, batch size and learning rate are determined by a small hyperparameter sweep of 100 trails on random 80:20 splits for each dataset and method. Hyperparameter optimization for Diffsurv also included sorting network and steepness.

Full hyperparameter ranges and resulting best models are provided along with further implementation details at `anon@git.com`.

During evaluation, the sorting network is not used since we only need to evaluate the ranks of the trained risk scores. Similarly, case-control sampling is not used. We measure the ranking performance of the models using the concordance index Harrell et al. (1982).

A.5 Possible Permutation Matrix

In order to account for censoring, we propose utilizing a possible permutation matrix Q_p. We provide a visualization for example risk set size 7 in Figure 3, which corresponds with Equation 12. It is also possible to represent such connections as a graph. Further, this possible permutation graph $\mathcal{G}_p$, is the complement of the order graph $\mathcal{G}_o$ typically used for C-index. Both graphs are visualised in Figure 4

$$\boldsymbol{Q}_p = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \tag{12}$$

A.6 Real World Datasets

- **FLCHAIN dataset:** A dataset containing information on patients with monoclonal gammopathy of undetermined significance (MGUS), focusing on serum free light chain (FLC) levels to study their prognostic significance in predicting disease progression.
- **NWTS dataset:** A dataset from a series of clinical trials on the treatment and outcomes of children with Wilms' tumor, a type of kidney cancer, aiming to improve understanding of tumor biology and optimize treatment strategies.
- **SUPPORT dataset:** A dataset from a multicenter study investigating the prognosis and treatment preferences of seriously ill hospitalized adults, with the goal of improving end-of-life care and informing decision-making processes.
- **METABRIC dataset:** A dataset comprising genomic and clinical data on breast cancer patients, focused on uncovering novel molecular subtypes for more precise prognostication and personalized treatment strategies.

A.7 SurvSVHN Dataset

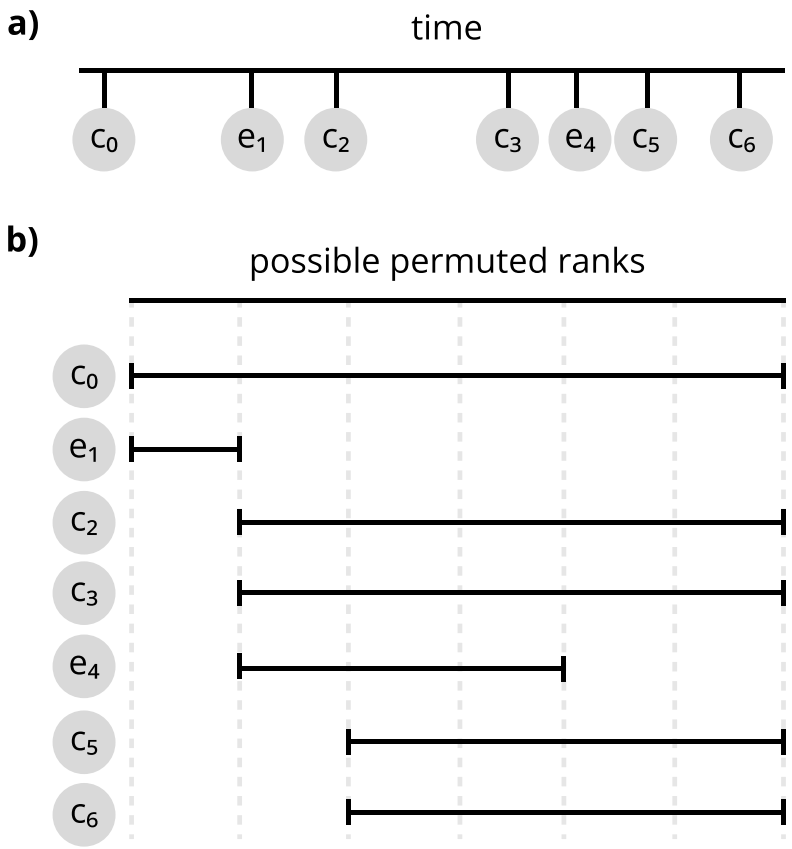
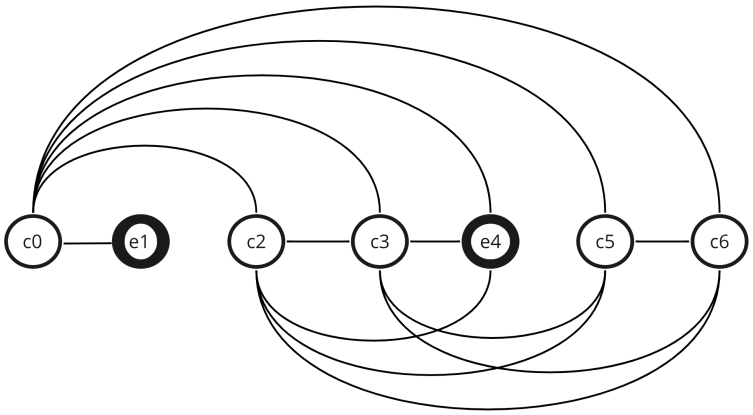


Figure 3: Possible permutations for an example case with two events (e_0 and e_4) and multiple censored samples (c_0, c_2, c_3, c_5, c_6).

(a) Possible Permutation Graph, $\mathcal{G}_p$



(b) Order Graph, $\mathcal{G}_o$

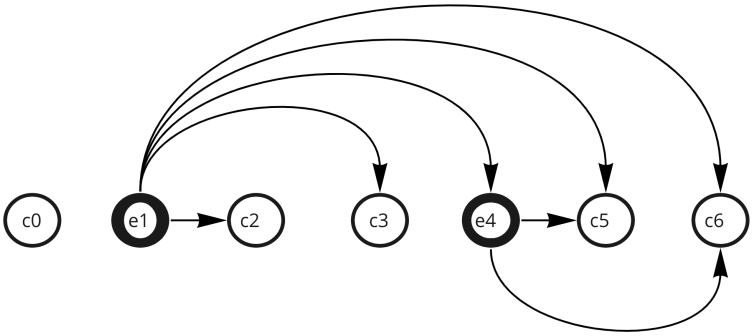


Figure 4: Graphs representation of possible permutations and order.

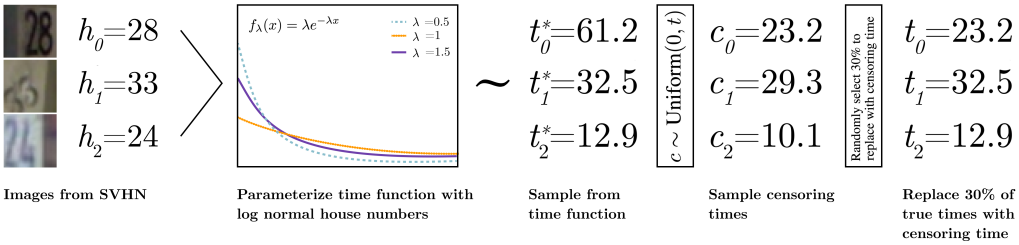


Figure 5: Visual abstract of the survSVHN dataset.