Supplementary Materials for FedPseudo: Privacy-Preserving Pseudo Value-Based Deep Learning Models for Federated Survival Analysis

Md Mahmudur Rahman
mrahman6@umbc.edu
University of Maryland, Baltimore County
Baltimore, Maryland, USA

Sanjay Purushotham
psanjay@umbc.edu
University of Maryland, Baltimore County
Baltimore, Maryland, USA

1 DATASET DESCRIPTION

In this section, we describe real-world survival datasets and simulated datasets with different censoring mechanisms.

1.1 Real Survival Datasets:

We evaluate the models on four real-world survival datasets. Table 1 shows the descriptive statistics of the datasets.

METABRIC: The Molecular Taxonomy of Breast Cancer International Consortium (METABRIC) dataset aims to determine new breast cancer subgroups at risk of death using the gene and protein expression profiles and clinical information of patients. The dataset contains 1,904 patients with a median survival time of 115 months, out of which 57.9% experienced death due to breast cancer [2]. The dataset consists of 4 gene indicators (MKI67, EGFR, PGR, and ERBB2) and 5 clinical features (hormone treatment indicator, radiotherapy indicator, chemotherapy indicator, ER-positive indicator, and age at diagnosis).

SUPPORT: Study to Understand Prognoses Preferences Outcomes and Risks of Treatment (SUPPORT) is the study of the survival time of seriously ill hospitalized adults [7]. The dataset consists of 8873 patients with a median survival time of 231 days, out of which 68% experienced death during the study with a median death time of 57 days, and 14 features (age, sex, race, number of comorbidities, presence of diabetes, presence of dementia, presence of cancer, mean arterial blood pressure, heart rate, respiration rate, temperature, white blood cell count, serum's sodium, and serum's creatinine).

GBSG: Rotterdam and German Breast Cancer Study Group (GBSG) contains breast cancer data of 2232 patients from the Rotterdam tumor bank [3] and the German Breast Cancer Study Group (GBSG) [11]. The covariates of the dataset include hormonal therapy, tumor grade, menopausal status, age, number of positive lymph nodes, progesterone receptors, and estrogen receptors.

We use the same train-test split of the METABRIC, SUPPORT, and GBSG datasets provided in [6]. The datasets can be found in https://github.com/jaredleekatzman/DeepSurv.

META-HD: The METABRIC Breast Cancer Omics dataset [2] contains over 25000 gene expression and copy number data from 144 normal breast tissue and 1989 tumor samples. We select 1980 patients [5], who appears in both the METABRIC dataset and the Omics dataset, and combine their features to obtain META-HD dataset. This dataset contains a total of 16377 features from 1980 patients.

Real-world Survival Dataset for FL (TCGA): We download and use datasets from The Cancer Genome Atlas (TCGA) from the GDC data portal. We selected 17 datasets for 17 cancer types based on the lowest missing values and available information for the common covariates. We first downloaded patients' clinical information for those cancer types, combined the data for all cancer types, and created a single dataset. We use the cancer types as one covariate and remove the missing values from the dataset. We perform one-hot encoding for the categorical variables. Using the tissue sort site (TSS) code, we first identify the health center from which data has been collected for a particular patient. Then, we form 7 local datasets by distributing the dataset into 7 regions (South, West, Midwest, Northeast, Europe, Canada, and Other) based on the health centers' location and consider the regions as clients in FL. We only use unrestricted data and do not attempt to identify participants from whom the data were obtained.

1.2 Simulated Datasets with Various Censoring Mechanisms

Censoring is a critical inherent problem in survival analysis, which leads to an overestimation of the survival prediction and results in unintentional biases toward the prediction [9]. Therefore, it is crucial to handle censoring to obtain accurate and approximately unbiased survival predictions efficiently. We adapt pseudo values in our model building, which are proven to efficiently handle censoring [10, 14]. Furthermore, we generate multiple simulated datasets in a federated setup with different censoring scenarios to show how well our models perform in different censoring settings under a federated setup compared to the baselines.

We generate 12 simulated datasets in a federated setup replicating different censoring scenarios to show how well our models perform under different types of censoring in a federated setup compared to the baselines. We set the number of clients to 10 and generated 5000 observations for each client. We first create a complete follow-up dataset (all uncensored) for each of the clients and generate multiple censored datasets from the complete follow-up dataset using different censoring mechanisms, such as (a) time censoring (**TC**), (b) interim censoring (**IC**), and case censoring (**CC**) with respectively 25% (**CC25**), 50% (**CC50**), and 75% (**CC75**) censored observations [1]. The complete follow-up (uncensored) dataset is constructed by generating 12 covariates from a multivariate normal distribution with mean μ and variance σ^2 followed by Weibull distributed survival times through Cox models [8], taking the nonlinear combination of covariates. Finally, we generate IID and non-IID censored data, assuming similar and varying survival distributions

https://portal.gdc.cancer.gov/

Table 1. Descriptive statistics of the real-world distributed TCGA datasets

Region →		South	Northeast	West	Midwest	Europe	Canada	Others
No. of Patients		1326	1210	697	1174	901	461	315
No. of Censoring (%)		926 (69.8)	849 (70.2)	528 (75.8)	791 (67.4)	641 (71.1)	345 (74.8)	150 (47.6)
Event Time	**min**	0.1	0.1	0.1	0.1	0.1	0.1	0.1
	max	6593	6873	3683	3258	10870	2803	10346
	mean	980.2	853.5	756.7	655.1	848.5	443.7	1102.7
Censoring Time	**min**	0.1	0.1	0.1	0.1	0.1	0.1	0.1
	max	8605	6768	6940	5925	7248	4694	7563
	mean	1375.7	1231.1	843.4	996.4	803.3	722.5	1493.6

for all clients. We also generate two simulated datasets (**DSDEC** and **DSDUC**), considering different survival time and censoring time distributions with equal and unequal numbers of censoring across the clients.

Here, we briefly describe the different censoring mechanisms.

Time Censoring: A random time is generated from a Weibull distribution with median and Q99 percentile times and a censoring proportion as a counterpart of each case in the complete follow-up dataset. A particular case is updated if the new random time is shorter than the complete follow-up time. The complete follow-up time is then replaced by the new random time, and the event status is updated from 1 (uncensored) to 0 (censored).

Interim Censoring: A random recruit time is generated as a counterpart of each case in the complete follow-up dataset, and a random interim time is generated from a Weibull distribution with median and Q99 percentile times. Suppose the interval between recruit time and interim time is shorter than the complete follow-up time. In that case, the complete follow-up time is then replaced by the interval, and the event status is updated from 1 (uncensored) to 0 (censored).

Case Censoring: A sample of cases is randomly selected from a complete follow-up dataset with a pre-specified probability of censoring. The corresponding time for the selected cases is shortened by a random amount, and the event status is updated from 1 (uncensored) to 0 (censored). This mechanism is equivalent to real-world censoring due to loss-to-follow-up and withdrawal from the study. We generated 3 datasets by selecting random samples with censoring probabilities 0.25, 0.50, and 0.75, respectively, from a complete follow-up dataset.

Different survival distribution and censoring ratio for the clients: We generate the covariates, survival time, and censoring time for each client, employing different parameters assuming different distributions. We fix the number of uncensored observations by 2500 for each client and consider both equal (2500) and unequal (randomly selected from 500 to 2500) numbers of censored observations for the clients.

2 OUR PROPOSED MODELS FOR FEDPSEUDO FRAMEWORK

Here, we provide the details of our proposed models for the FedPseudo framework. We graphically show our proposed FedPseudo framework-based models' architectures in figure 1. The pseudocode for the federated pseudo values computation and federated training of our pseudo value-based survival models is provided in algorithm 1.

Pseudo Value-Based Deep Survival Models for FSA: Once the pseudo values for the subjects in the clients are calculated, they are used as response variables (ground truth) in our proposed client-specific pseudo value-based deep survival models. Here, we introduce three deep learning-based model architectures- Once the pseudo values for the subjects in the clients are calculated, they are used as response variables (ground truth) in our proposed client-specific pseudo value-based deep survival models. Here, we introduce three deep learning-based model architectures- (1) `FedPDNN`: a simple feed-forward deep neural network, which uses fully connected (FC) layers to learn the time-varying nonlinear covariate effect on survival probability, (2) `FedPLSTM`: consists of LSTM [4] model architecture to efficiently capture the time-dependency in the pseudo values, and (3) `FedPAttn`: consists of a global attention-based Bi-LSTM architecture [12]. We modify the architecture of the AttenSurv [12] using static covariates as input and pseudo values as output. Similar to the AttenSurv, `FedPAttn` consists of the global attention mechanism for identifying the risk factors, a Bi-LSTM module for learning the hidden representations of the trajectory of patients and pseudo values, and an FC output layer followed by a sigmoid activation for survival prediction. The global attention mechanism is designed as follows. The global weight $\alpha_p^i, i = 1, 2, .., N, p = 1, 2, ..., P$ obtained from the subject covariate weight layer is normalized as $\beta_p^i = \frac{exp(\alpha_p^i)}{\sum_{p=1}^{P} exp(\alpha_p^i)}$. The β_p^i are multiplied by covariate vectors X_p^i to produce the output of the global attention module, c_p^i. A residual connection, $c_p^i + = X_p^i$, is used to capture the shared and individual information. The residual connections are fed into a Bi-LSTM module as input. Bi-LSTM learns the hidden representation of the trajectory of subjects and pseudo values from pre-specified time points $\tau_j, j = 1, 2, .., M$, i.e., forward hidden states $\overrightarrow{h}_j^i$ and backward hidden states $\overleftarrow{h}_j^i$. Finally, the forward and backward representations are concatenated as $h_j^i = [\overrightarrow{h}_j^i, \overleftarrow{h}_j^i]$, and the concatenated representation is fed into a fully connected layer followed by a sigmoid activation function

Algorithm 1 FedPseudo Framework

Input: Local dataset $D^k = \{X^k, Y^k, \delta^k\}$, number of clients K, total number of communication rounds V, number of local epochs E, learning rate η, a vector of pre-specified time points $\mathbf{s} = \{s_1 < s_2 < .. < s_M\}$, number of subject n_k in client k, sensitivity parameter S, privacy budget parameter ϵ.
Output: The final model w^V.
Federated Pseudo Value Calculation:
for $k \in K$ `in parallel` **do**
  Create a local dictionary of input data
  $D_k : keys(D_k) = \tau_k$ &
  $values(D_{k,t}) = (d_{k,t}, c_{k,t}); \forall t \in \tau_k$ and calculate the risk set at starting time point $t_0 \in \tau_k$, $R_{k,t_0} = n_k$.
  if DP enforced **then**
    $values(D_{k,t}) = values(D_{k,t}) + Lap(S/\epsilon); \forall t \in \tau_k$ and $R_{k,t_0} = R_{k,t_0} + Lap(S/\epsilon)$
  else
    $values(D_{k,t}) = values(D_{k,t})$ and $R_{k,t_0} = R_{k,t_0}$
  end if
end for
Send D_k and R_{k,t_0} to the global server
Create a global dictionary
$D : values(D) = \cup_{t \in keys(D_k)} values(D_k, t)$ where $\tau = \cup_{k \in K} \tau_k$
Sort the values of the dictionary by its keys.
For every $t \in \tau : d_t \leftarrow \sum_{t \in \tau_k} d_{k,t}$, $c_t \leftarrow \sum_{t \in \tau_k} c_{k,t}$ and $R_{t_0} = \sum_{k \in K} R_{k,t_0}$
Create a global partial matrix M (or M' if DP enforced); $[R_{t_0}, d_t, c_t] \in M \forall t \in \tau$.
Compute: $R_{t_j} = R_{t_{j-1}} - d_{t_{j-1}} - c_{t_{j-1}}; j = 1, 2, .., m$
$\hat{S}_G(t) = \prod_{t_j \in \tau \leq t} \frac{R_{t_j} - d_{t_j}}{R_{t_j}}$
for $k \in K$ `in parallel` **do**
  Send global partial matrix M to client k
  for $i = 1, 2, ..., n_k$ **do**
    $R_{t_0}^{-ik} = R_{t_0} - 1$
    if $T_{ik} = t_j \in \tau$ and $\delta_{ik} = 1$ **then**
      $d_{t_j}^{-ik} \leftarrow d_{t_j} - 1$
    end if
    if $T_{ik} = t_j \in \tau$ and $\delta_{ik} = 0$ **then**
      $c_{t_j}^{-ik} \leftarrow c_{t_j} - 1$
    end if
    for $j = 1, ..., m$ **do**
      $R_{t_j}^{-ik} = R_{t_{j-1}}^{-ik} - d_{t_{j-1}}^{-ik} - c_{t_{j-1}}^{-ik}$
    end for
    $\hat{S}_G^{-ik}(t) = \prod_{t_j \in \tau \leq t} \frac{R_{t_j}^{-ik} - d_{t_j}^{-ik}}{R_{t_j}^{-ik}}$
    $J_{ik}(s) = n\hat{S}_G(s) - (n-1)\hat{S}_G^{-ik}(s)$
  end for
end for
Federated Training:
Server executes:
initialize w^0
for v = 0, 1, ..., V-1 **do**
  Randomly sample a set of clients Q_v
  $n \leftarrow \sum_{k \in Q_v} |n_k|$
  for $k \in Q_v$ `in parallel` **do**
    send the global model w^v to client k
    $\Delta w_k^v \leftarrow$ **LocalTraining**(k, w^v)
  end for
  if DP enforced **then**
    $w^{v+1} \leftarrow w^v - \eta \sum_{k \in Q_v} \frac{|n_k|}{n} \Delta w_k^v$
  else
    $w^{v+1} \leftarrow \hat{w}^v - \eta \sum_{k \in Q_v} \frac{|n_k|}{n} (\Delta w_k^v + \mathbf{G}(0, S_\sigma I))$
  end if
end for
return w^V
Client executes:
$L(w; \mathbf{b}) = \sum_{(x,j) \in \mathbf{b}} l(w; x; j)$
LocalTraining(k, w^v) :
$w_k^v \leftarrow w^v$
for epoch i = 1, 2,..., E **do**
  for each batch $\mathbf{b} = \{\mathbf{X}, \mathbf{J}\}$ of D^k **do**
    $w_k^v \leftarrow w_k^v - \eta(\nabla L(w_k^v; \mathbf{b}))$
  end for
end for
if DP enforced **then**
  $\Delta w_k^v \leftarrow (w^v - w_k^v)/max\left(1, \frac{||(w^v - w_k^v)||_2}{S}\right)$
else
  $\Delta w_k^v \leftarrow w^v - w_k^v$
end if
return w_k^v

to predict the survival function at the pre-specified time points $\tau_j, j = 1, 2, .., M$. All the proposed local models take the client's covariates, $\mathbf{X_k}$, as input and the client's subject-specific pseudo values ($J_{ik}(t)$) as the target variable.

Furthermore, we investigate applying (ϵ, δ) - differential privacy in the `FedPDNN` model to protect it from plausible adversarial attack, we call it `DP-FedPDNN`. In `DP-FedPDNN`, the global server clips the local updates of each client Δw_K^v to keep the L2-norm at most sensitivity S. The global server aggregate the clipped updates $\Delta \hat{w}_K^v$ followed by adding Gaussian noise, and finally performs an average to obtain noisy global update $\hat{w}^{v+1}$. The noisy global update is sent back to clients for local training.

3 ADDITIONAL EXPERIMENTS

In this section, we discuss tests performed for checking assumptions mentioned in the main paper, such as differential privacy for global survival function, linearity, and proportional hazard assumptions for covariates.

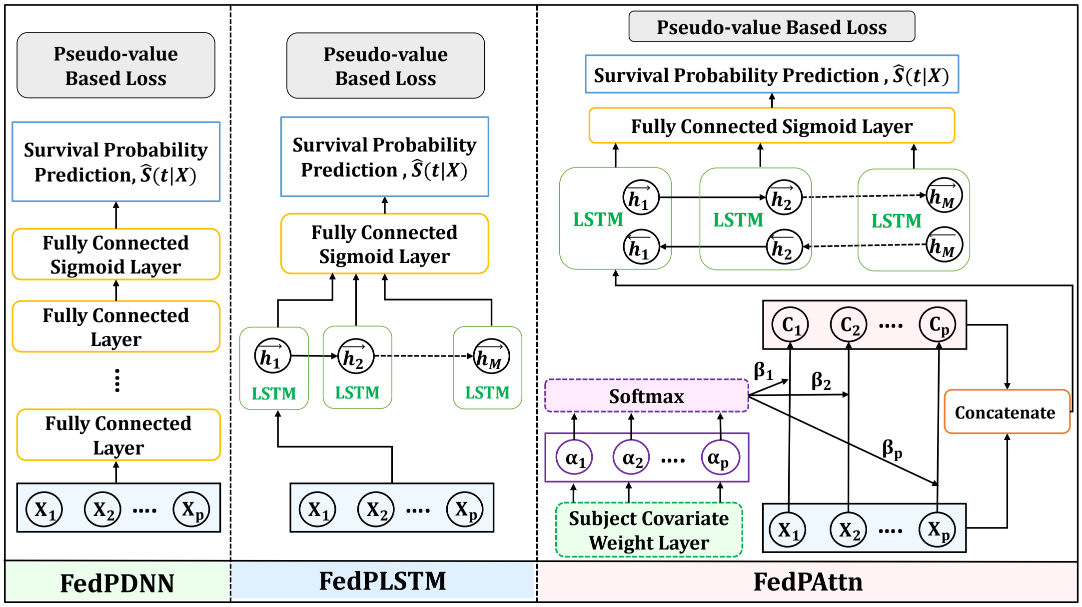

Figure 1. FedPseudo framework-based model architectures

3.1 Log-Rank Test for Differential privacy of Global survival function

In section 4 of the main paper, we discuss how we enforce differential privacy (DP) while computing the global survival function, which results in differentially private federated pseudo values for the survival function following theorem 2. To ensure differential privacy in the global survival function, we must carefully choose the privacy budget parameter ϵ since it significantly impacts the survival function estimate. To choose the optimal value of ϵ, we perform a log-rank test of the overall difference between the survival function estimates in DP and non-DP settings. We select the minimum ϵ for which the test becomes insignificant at a significance level $\alpha = 0.05$, i.e., there is no significant difference in the global survival functions before and applying noise to the quantities of the global partial matrix to compute the global survival function. We show how we choose ϵ to preserve differential privacy on METABRIC data in figure 2. The figure shows that for a value of ϵ less than 4, the log-rank test is significant, which implies a significant difference between global survival probability with and without DP. However, the test becomes insignificant for $\epsilon = 4$, which is considered for computing differentially private global survival function.

3.2 Checking Linearity and Proportional Hazard Assumptions

To answer research question 4 (Q4) in section 5 (Experiments), we first check the linearity and proportional assumptions for the DSDUC dataset.

Checking Linearity Assumption: In traditional CoxPH models, it is often assumed that the continuous covariates have a linear form. However, this assumption often gets violated and, therefore, should be checked. We use the function `ggcoxfunctional` in the R package `survminer` to check the functional form of the continuous covariates, which plots the Martingale residuals against continuous covariates to detect nonlinearity. Figure 3 shows that all the covariates in the DSDUC data violate the linearity assumption, i.e., they have a nonlinear functional form.

Checking PH Assumption: CoxPH models also frequently assume that the covariate's influence relative to the baseline does not change over time, i.e., the proportional hazard (PH) assumption. This assumption also gets violated in real-world scenarios, especially in the federated setup where it is difficult to hold the assumptions for all the clients. We first graphically show the violation of the PH assumption of the covariates for a particular client in figure 4 based on the scaled Schoenfeld residuals on the DSDUC dataset. We use the function `ggcoxzph` in the R package `survminer` to detect the violation of the PH assumption. A non-random pattern against time in the plots indicates the violation of the PH assumption. Figure 4 shows that almost all the covariates in client 1 violate the PH assumption except for the covariate X_5 and X_7. However, the range of the coefficient Beta(t) is narrow for these two covariates. Then, we perform a statistical Chi-square test to check the violation of the PH assumption on the DSDUC dataset. We use the function `cox.zph` in R package `survival` to test for independence between residuals and time for each covariate in each client and show the test statistics and P-value of the tests in table 4. A P-value less than 0.05 indicates the covariate's influence relative to the baseline changes over time (i.e., violation of the PH assumption). In addition, we perform a global test for the model to check whether the overall PH assumption holds for a particular client. We consider Cox-based baselines where the `LinearPH` model makes both linearity and PH assumption, and the `NNph` model makes PH assumption. In contrast, `NNnph` and our proposed models do not make any of the assumptions and thus, result in better performance. Table 4 shows that the PH assumptions do not hold for most of the

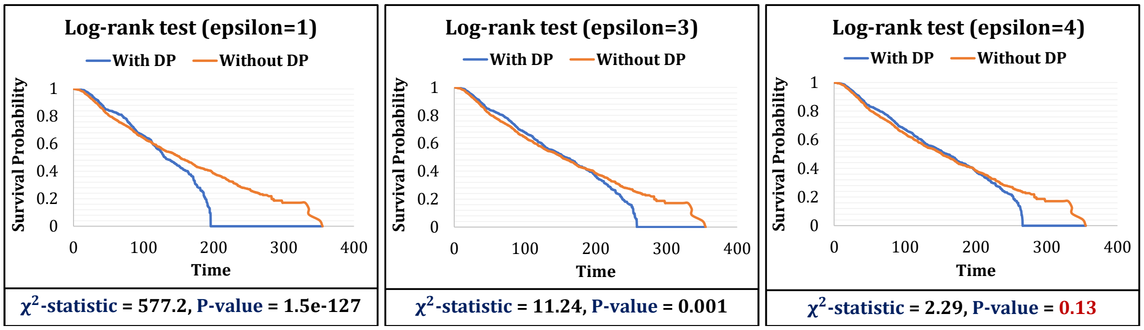

Figure 2. Log-rank test on METABRIC dataset for choosing the value of ϵ

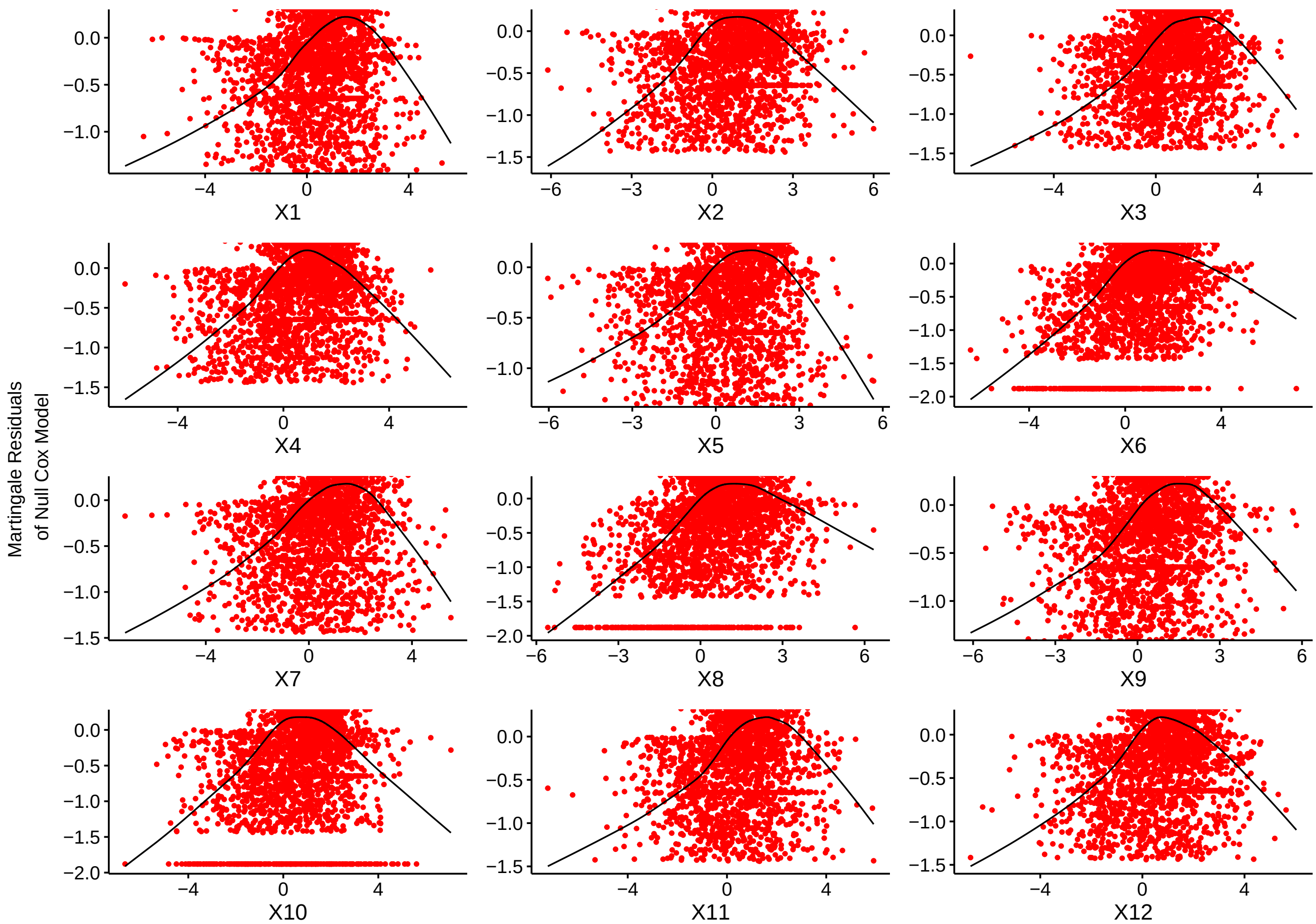

Figure 3. Plots of the Martingale residuals against continuous covariates to detect nonlinearity on the DSDUC dataset.

covariates in all the clients in the DSDUC dataset, and even the global test is significant for all the clients, which indicates a strong violation of the PH assumption.

4 ADDITIONAL RESULTS

Here, we provide the detailed results of Tables 4 and 5 in the main paper and some additional results.

4.1 Comparing performance in different censoring settings

In table 3 (table 4 in main paper), our proposed models show better calibration performance, i.e., better Brier scores, and similar discriminative performance, i.e., similar C-index compared to the baselines in different censoring settings. Our models achieve significant improvement over the `LinearPH` model, which suffers from the presence of nonlinearity in the simulated datasets. It is clearly

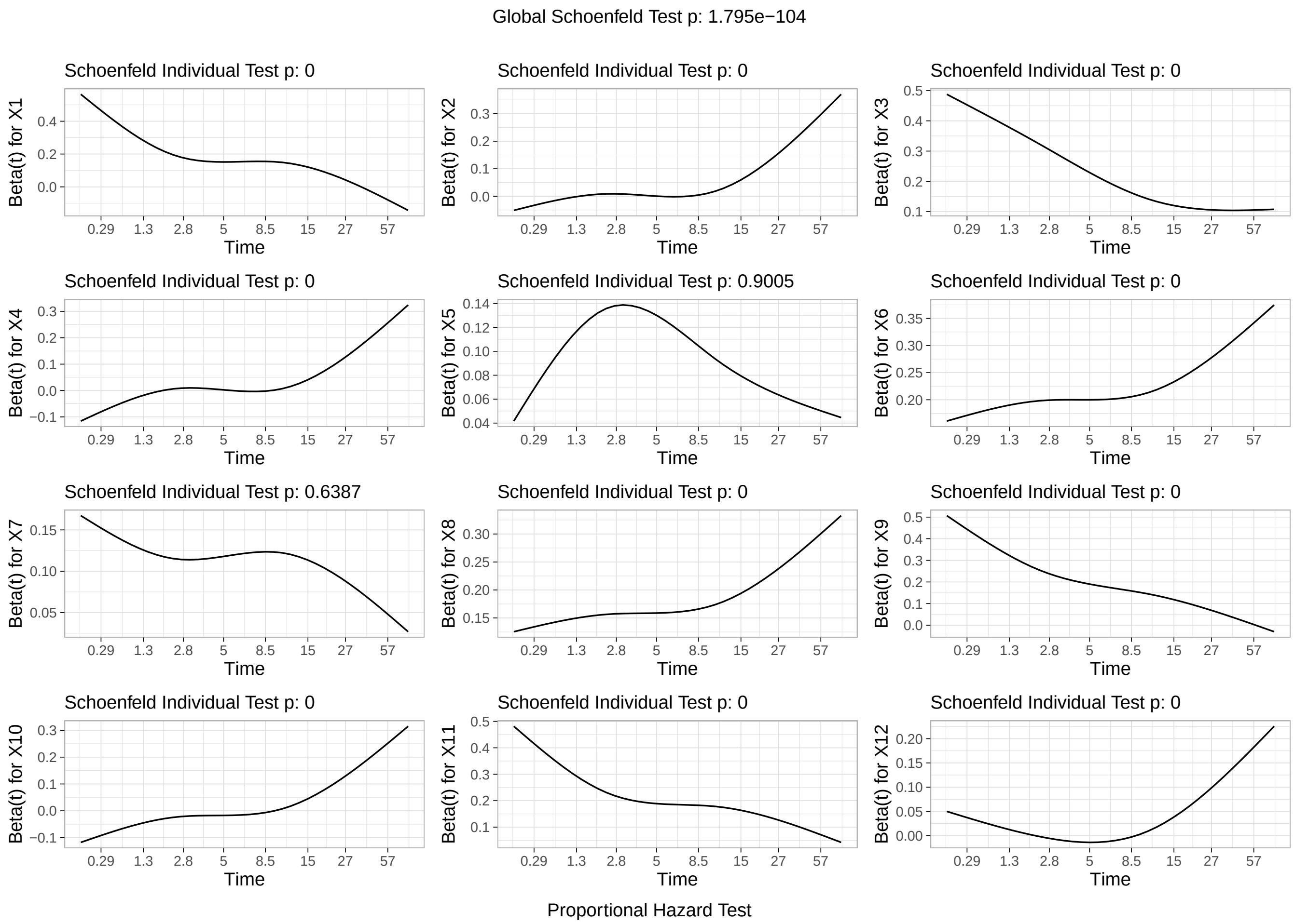

Figure 4. PH assumption test (non-constant lines) on DSDUC dataset for Client 1

observed that adapting pseudo values in our models help to handle censoring well and to make more accurate survival prediction.

4.2 Comparing performance under the violation of linearity and PH assumptions

From table 4 (Table 5 in main paper), it is observed that the PH assumption gets violated for most of the covariates in all the clients in the DSDUC dataset. The global test is also significant for all the clients, which indicates a strong client-level violation of the PH assumption. In the DSDUC dataset, the covariates also have shown nonlinear functional form (shown in figure 3. The `LinearPH` performs worst under the violation of both linearity and PH assumptions, which is expected as the model makes both assumptions. Our proposed models, especially `FedPAttn` perform significantly better than the baseline models, which assume either linearity or PH assumption (`LinearPH` and `NNph`). We want to highlight that our best-performing model `FedPAttn` performs even better than the baseline model `NNnph` (designed for relaxing the PH assumption) for almost all clients and shows a significantly better Brier score on Centralized test data. Our models capture the time-varying covariate effect on survival prediction by learning the time-specific weights in the output layer and, thus, are not affected by the violation of the PH assumption. Furthermore, neural networks used in our models can capture the nonlinearity in data well, which results in superior performance, especially compared to linear survival models.

4.3 Additional Results 1: Comparing locally trained and federated FedPDNN model

In order to show how much improvement the federated FedPseudo framework achieves over the locally trained pseudo value-based models, we consider the following three training setups:
(1) Locally train the `FedPDNN` models for each client with local Jackknife pseudo values as the response variable in the models. **(2)** Locally train the `FedPDNN` models for each client with federated pseudo values as the response variable in the models. **(3)** Federated training of the `FedPDNN` models with federated pseudo values as the response variable in the models.
From the Table 5, we have the following findings:
(a) Using federated pseudo values improves the performance of the locally trained models, which use the local Jackknife pseudo values

Table 2. Performance comparisons [mean(sd)] of the models on the real survival datasets. Our best-performing models show statistically significant improvement over baseline models (in 17 out of 24 cases for LinearPH), in 17 out of 24 cases for NNnph), in 17 out of 24 cases for NNph), and in 20 out of 24 cases for DeepHit).

Metric	Dataset	Setup	Model											
			Baseline Models					Our Models with Non-DP FPV			Our Models with DP FPV			DP-FedPDNN
			LinearPH	RSF	NNnph	NNph	DeepHit	FedPDNN	FedPLSTM	FedPAttn	FedPDNN	FedPLSTM	FedPAttn	
C-Index (↑ better)	METABRIC	**Centralized**	0.63(0.001)^a	0.60(0.002)^a	0.66(0.009)^a	0.63(0.008)^a	0.65(0.034)^a	0.66(0.004)	**0.67(0.001)**	**0.67(0.001)**	**0.67(0.004)**	0.66(0.002)	**0.67(0.002)**	0.66(0.007)
		IID	0.63(0.004)^a	0.62(0.005)^a	0.65(0.019)^b	0.63(0.006)^a	**0.66(0.017)**	0.64(0.017)	0.65(0.009)	**0.66(0.007)**	0.65(0.014)	0.64(0.041)	**0.66(0.010)**	0.64(0.014)
		Non-IID	0.65(0.003)^a	0.63(0.006)^a	**0.68(0.014)**	0.65(0.020)^a	**0.68(0.015)**	0.67(0.008)	0.67(0.016)	0.67(0.014)	0.67(0.013)	**0.68(0.010)**	0.67(0.0011)	0.55(0.014)
	SUPPORT	**Centralized**	0.60(0.001)^a	0.60(0.001)^a	**0.61(0.004)**	**0.61(0.004)**	0.60(0.008)^a	**0.61(0.002)**	**0.61(0.004)**	**0.61(0.001)**	**0.61(0.002)**	0.60(0.002)	**0.61(0.001)**	**0.61(0.003)**
		IID	0.60(0.001)^a	**0.61(0.001)**	0.60(0.006)^a	**0.61(0.002)**	0.59(0.007)^a	0.60(0.002)	0.60(0.003)	**0.61(0.003)**	0.60(0.002)	0.60(0.003)	**0.61(0.003)**	0.60(0.002)
		Non-IID	0.60(0.002)^a	**0.61(0.002)**	**0.61(0.004)**	**0.61(0.004)**	0.60(0.008)^a	0.60(0.002)	0.60(0.004)	0.60(0.002)	0.60(0.000)	0.60(0.003)	**0.61(0.001)**	0.59(0.007)
	GBSG	**Centralized**	0.66(0.005)^a	0.66(0.006)^a	0.66(0.008)^a	**0.67(0.008)**	0.61(0.055)^a	0.66(0.003)	**0.67(0.002)**	**0.67(0.002)**	**0.67(0.003)**	**0.67(0.001)**	0.66(0.003)	0.66(0.010)
		IID	0.64(0.008)^a	**0.68(0.004)**	0.64(0.011)^a	0.66(0.010)^a	0.63(0.016)^a	0.66(0.012)	0.66(0.002)	0.66(0.007)	0.66(0.007)	0.66(0.007)	0.67(0.009)	0.66(0.012)
		Non-IID	**0.63(0.007)**	**0.63(0.022)**	0.57(0.013)^a	0.60(0.019)^c	0.57(0.031)^a	0.58(0.028)	0.61(0.008)	0.58(0.014)	0.59(0.020)	0.61(0.007)	0.57(0.010)	0.57(0.035)
	META-HD	**Centralized**	0.65(0.008)^a	0.58(0.012)^a	0.59(0.026)^a	0.63(0.034)^a	0.64(0.010)^a	**0.69(0.003)**	0.59(0.037)	0.67(0.012)	**0.69(0.004)**	0.62(0.053)	0.67(0.011)	0.63(0.015)
		IID	0.65(0.012)^a	0.55(0.005)^a	0.65(0.009)^a	**0.66(0.007)**	0.64(0.025)^a	**0.66(0.006)**	0.57(0.017)	0.64(0.003)	0.65(0.005)	0.57(0.027)	0.64(0.009)	0.65(0.003)
		Non-IID	**0.67(0.007)**	0.55(0.015)^a	0.65(0.013)	**0.67(0.007)**	0.66(0.007)	0.64(0.017)	0.59(0.014)	0.61(0.023)	0.64(0.010)	0.57(0.024)	0.59(0.013)	0.63(0.038)
Win/Total Cases			2/12	4/12	3/12	6/12	2/12	3/12	3/12	5/12	4/12	2/12	6/12	1/12
Brier Score (↓ better)	METABRIC	**Centralized**	0.19(0.007)^a	0.30(0.004)^a	**0.18(0.003)**	0.20(0.009)^a	0.19(0.002)^a	**0.18(0.001)**	0.19(0.005)	**0.18(0.001)**	0.19(0.001)	0.19(0.003)	0.19(0.003)	**0.18(0.003)**
		IID	**0.18(0.001)**	0.29(0.003)^a	**0.18(0.003)**	0.19(0.003)^a	0.20(0.002)^a	0.19(0.003)	**0.18(0.005)**	**0.18(0.004)**	0.19(0.005)	0.19(0.006)	0.19(0.000)	0.19(0.004)
		Non-IID	**0.18(0.002)**	0.30(0.001)^a	**0.18(0.003)**	0.19(0.008)^a	0.20(0.005)^a	**0.18(0.002)**	**0.18(0.005)**	**0.18(0.003)**	**0.18(0.001)**	**0.18(0.004)**	0.19(0.006)	0.19(0.006)
	SUPPORT	**Centralized**	0.20(0.001)^a	0.22(0.001)^a	0.20(0.001)^a	0.20(0.002)^a	0.21(0.002)^a	0.20(0.001)	0.20(0.001)	**0.19(0.000)**	**0.19(0.001)**	0.20(0.001)	**0.19(0.000)**	**0.19(0.001)**
		IID	**0.20(0.001)**	0.22(0.001)^a	**0.20(0.003)**	**0.20(0.003)**	0.23(0.004)^a	**0.20(0.002)**	**0.20(0.001)**	**0.20(0.002)**	**0.20(0.001)**	**0.20(0.001)**	**0.20(0.002)**	**0.20(0.002)**
		Non-IID	**0.19(0.001)**	0.22(0.001)^a	0.20(0.002)^a	0.20(0.003)^a	0.22(0.002)^a	**0.19(0.001)**	**0.19(0.001)**	**0.19(0.001)**	**0.19(0.001)**	**0.19(0.001)**	**0.19(0.001)**	0.20(0.007)
	GBSG	**Centralized**	0.20(0.011)^a	0.26(0.003)^a	0.19(0.003)^a	0.19(0.007)^a	0.23(0.001)^a	**0.18(0.001)**	**0.18(0.001)**	**0.18(0.001)**	**0.18(0.000)**	**0.18(0.001)**	**0.18(0.001)**	**0.18(0.001)**
		IID	0.19(0.002)^a	0.25(0.003)^a	0.19(0.002)^a	0.19(0.003)^a	0.23(0.001)^a	0.19(0.002)	0.19(0.001)	**0.18(0.003)**	0.19(0.002)	0.19(0.003)	0.19(0.002)	**0.18(0.003)**
		Non-IID	**0.21(0.004)**	0.23(0.004)^a	0.23(0.004)^a	0.22(0.005)^a	0.23(0.002)^a	**0.21(0.005)**	0.22(0.003)	0.22(0.014)	**0.21(0.002)**	0.23(0.011)	0.22(0.020)	0.24(0.008)
	META-HD	**Centralized**	0.27(0.114)^a	0.32(0.008)^a	0.20(0.003)^a	0.25(0.075)^a	0.20(0.007)^a	0.20(0.010)	0.20(0.005)	0.21(0.022)	0.20(0.008)	0.20(0.008)	**0.19(0.008)**	0.20(0.002)
		IID	0.22(0.010)^a	0.32(0.006)^a	0.23(0.009)^a	0.22(0.002)^a	0.21(0.032)^a	**0.19(0.005)**	0.20(0.002)	0.20(0.002)	**0.19(0.004)**	0.20(0.002)	0.20(0.003)	0.20(0.004)
		Non-IID	0.22(0.006)^a	0.30(0.005)^a	0.24(0.008)^a	0.22(0.004)^a	**0.18(0.004)**	**0.18(0.006)**	0.19(0.003)	0.19(0.007)	**0.18(0.004)**	0.19(0.002)	**0.18(0.003)**	0.22(0.060)
Win/Total Cases			5/12	0/12	4/12	1/12	1/12	8/12	5/12	8/12	8/12	4/12	6/12	5/12

Wilcoxon signed-rank test - statistically significant codes: 0 'a' 0.001 'b' 0.01 'c' 0.05 'd' 0.1 ' ' 1, (Read '*' p as significant at $(p \times 100)\%$ level of significance). The test is performed to compare our best-performing model with the baseline models for each case.

as the response variable in the models, especially in terms of Brier scores and in the Non-IID setting.

(b) Our federated FedPDNN models significantly improve the client-level performance obtained by locally trained models on the test set of the local clients' data.

(c) Federated FedPDNN models with federated pseudo values obtain the best Brier scores in all four datasets and the best C-Index on GBSG and META-HD datasets. On the non-IID settings of the METABRIC and SUPPORT datasets, federated FedPDNN models show slightly worse performance in terms of the C-index. However, in the IID setting, those models give better performance than the locally trained models.

The federated training allows the models to learn from different clients' data simultaneously and, thus, helps to obtain more generalized and superior performance. We can conclude that federated learning is crucial when data are collected in decentralized clients and especially when the distribution of decentralized datasets is different (non-IID). The locally trained models fail to learn generalized representation from different clients' data, resulting in poor performance.

4.4 Additional Results 2: Comparing locally trained and federated Cox-based models

We also compare the performances of the locally trained and federated Cox-based models with PH assumption; `LinearPH` and `NNph`. Table 6 shows that federated Cox-based models outperform the locally trained Cox-based models in IID settings, whereas federated Cox-based models fail to show improvement over locally trained Cox-based models in non-IID settings of the real-world survival datasets. On the other hand, our federated FedPDNN model with federated pseudo values achieves consistent improvement over locally trained models in both IID and non-IID settings of real-world survival datasets.

Table 3. Performance comparisons [mean (sd)] of the models on the simulated datasets with different censoring settings. Higher C-index and Lower Brier scores indicate better performance.

Setup	Metric	Model	TC IID	TC Non-IID	IC IID	IC Non-IID	CC25 IID	CC25 Non-IID	CC50 IID	CC50 Non-IID	CC75 IID	CC75 Non-IID	DSDEC	DSDUC
Centralized	C-Index	LinearPH	0.51 (0.000)	0.66 (0.000)	0.51 (0.001)	0.64 (0.000)	0.52 (0.000)	0.62 (0.000)	0.53 (0.000)	0.63 (0.000)	0.52 (0.000)	0.62 (0.000)	0.53 (0.001)	0.55 (0.001)
		NNph	**0.77 (0.004)**	**0.92 (0.002)**	**0.78 (0.004)**	**0.90 (0.001)**	0.79 (0.002)	**0.88 (0.003)**	0.79 (0.003)	**0.89 (0.002)**	0.79 (0.003)	**0.88 (0.002)**	0.50 (0.011)	0.63 (0.007)
		NNnph	0.75 (0.009)	0.91 (0.007)	0.76 (0.004)	**0.90 (0.004)**	0.78 (0.002)	**0.88 (0.004)**	0.78 (0.003)	0.88 (0.003)	0.78 (0.004)	**0.88 (0.007)**	**0.71 (0.003)**	**0.73 (0.004)**
		FedPDNN	0.76 (0.006)	0.88 (0.004)	0.76 (0.010)	0.88 (0.004)	0.84 (0.010)	0.87 (0.005)	0.81 (0.016)	0.87 (0.006)	0.77 (0.017)	0.86 (0.010)	0.68 (0.006)	0.72 (0.001)
		FedPLSTM	0.76 (0.003)	0.89 (0.003)	0.77 (0.001)	0.89 (0.002)	**0.86 (0.002)**	0.87 (0.001)	**0.85 (0.004)**	0.86 (0.003)	0.80 (0.005)	0.86 (0.004)	0.67 (0.009)	0.72 (0.006)
		FedPAttn	**0.77 (0.006)**	0.89 (0.006)	0.77 (0.005)	0.89 (0.002)	0.85 (0.001)	0.86 (0.004)	**0.85 (0.007)**	0.85 (0.006)	**0.81 (0.003)**	0.84 (0.008)	0.68 (0.009)	0.71 (0.003)
	Brier Score	LinearPH	0.15 (0.000)	0.14 (0.000)	0.14 (0.000)	0.14 (0.000)	0.18 (0.000)	0.19 (0.000)	0.22 (0.000)	0.22 (0.000)	0.22 (0.000)	0.22 (0.000)	0.24 (0.000)	0.23 (0.000)
		NNph	**0.10 (0.004)**	**0.05 (0.002)**	**0.09 (0.001)**	0.05 (0.002)	0.10 (0.001)	0.06 (0.002)	0.11 (0.003)	**0.07 (0.002)**	**0.11 (0.001)**	**0.08 (0.002)**	0.24 (0.001)	0.21 (0.001)
		NNnph	0.11 (0.004)	0.06 (0.002)	0.10 (0.001)	0.06 (0.002)	0.10 (0.001)	0.07 (0.003)	0.11 (0.002)	**0.07 (0.003)**	0.12 (0.003)	**0.08 (0.002)**	0.22 (0.011)	0.19 (0.009)
		FedPDNN	0.11 (0.007)	**0.05 (0.005)**	0.11 (0.005)	**0.04 (0.003)**	0.04 (0.007)	**0.05 (0.007)**	0.08 (0.015)	0.09 (0.007)	0.13 (0.010)	0.13 (0.008)	0.22 (0.005)	**0.16 (0.004)**
		FedPLSTM	0.11 (0.005)	**0.05 (0.002)**	0.10 (0.003)	**0.04 (0.001)**	**0.02 (0.003)**	0.06 (0.005)	**0.05 (0.005)**	0.09 (0.006)	0.12 (0.003)	0.12 (0.005)	**0.20 (0.005)**	**0.16 (0.007)**
		FedPAttn	**0.10 (0.002)**	**0.05 (0.003)**	0.10 (0.005)	**0.04 (0.002)**	**0.02 (0.002)**	0.06 (0.006)	**0.05 (0.002)**	0.09 (0.006)	0.12 (0.001)	0.13 (0.006)	**0.20 (0.004)**	0.17 (0.006)
Federated	C-Index	LinearPH	0.51 (0.000)	0.66 (0.000)	0.51 (0.000)	0.64 (0.000)	0.52 (0.000)	0.62 (0.000)	0.53 (0.000)	0.63 (0.000)	0.52 (0.001)	0.62 (0.000)	0.53 (0.001)	0.55 (0.001)
		NNph	**0.78 (0.004)**	**0.91 (0.001)**	**0.78 (0.002)**	**0.89 (0.002)**	0.78 (0.001)	**0.87 (0.001)**	**0.79 (0.001)**	**0.88 (0.000)**	**0.79 (0.002)**	**0.87 (0.002)**	0.56 (0.003)	0.64 (0.012)
		NNnph	0.77 (0.003)	0.90 (0.003)	0.77 (0.002)	**0.89 (0.004)**	0.78 (0.001)	0.86 (0.001)	0.78 (0.001)	0.87 (0.002)	0.78 (0.003)	0.86 (0.006)	**0.69 (0.001)**	**0.72 (0.002)**
		FedPDNN	0.76 (0.004)	0.88 (0.007)	0.77 (0.004)	0.88 (0.003)	**0.79 (0.001)**	**0.87 (0.003)**	0.78 (0.001)	0.86 (0.006)	0.77(0.004)	0.83 (0.020)	0.63 (0.020)	0.69 (0.007)
		FedPLSTM	0.75 (0.003)	0.90 (0.002)	0.77 (0.006)	**0.89 (0.001)**	0.78 (0.001)	**0.87 (0.001)**	**0.79 (0.001)**	0.87 (0.002)	0.78 (0.002)	**0.87 (0.002)**	0.65 (0.007)	0.69 (0.004)
		FedPAttn	0.76 (0.005)	0.89 (0.002)	0.76 (0.004)	**0.89 (0.001)**	0.78 (0.001)	**0.87 (0.001)**	**0.79 (0.001)**	0.84 (0.006)	**0.79 (0.001)**	0.86 (0.003)	0.66 (0.002)	0.70 (0.003)
	Brier Score	LinearPH	0.15 (0.000)	0.15 (0.000)	0.14 (0.000)	0.14 (0.000)	0.18 (0.000)	0.19 (0.001)	0.22 (0.000)	0.22 (0.000)	0.22 (0.000)	0.22 (0.000)	0.26 (0.002)	0.25 (0.000)
		NNph	**0.09 (0.002)**	0.06 (0.002)	**0.09 (0.001)**	0.06 (0.001)	0.10 (0.001)	0.07 (0.001)	0.12 (0.001)	**0.08 (0.002)**	**0.12 (0.001)**	**0.09 (0.002)**	0.27 (0.007)	0.24 (0.003)
		NNnph	0.10 (0.003)	0.06 (0.003)	0.10 (0.001)	0.06 (0.001)	0.11 (0.001)	0.08 (0.002)	0.12 (0.001)	0.09 (0.002)	0.13 (0.002)	0.10 (0.001)	0.26 (0.003)	0.23 (0.002)
		FedPDNN	0.11 (0.007)	**0.05 (0.001)**	0.11 (0.005)	**0.05 (0.002)**	**0.09 (0.001)**	0.06 (0.003)	**0.11 (0.001)**	**0.08 (0.002)**	0.14 (0.002)	0.12 (0.001)	0.21 (0.007)	**0.17 (0.001)**
		FedPLSTM	0.12 (0.008)	**0.05 (0.001)**	0.11 (0.003)	**0.05 (0.002)**	**0.09 (0.002)**	0.06 (0.002)	0.12 (0.002)	**0.08 (0.001)**	0.14 (0.003)	0.12 (0.001)	**0.20 (0.006)**	**0.17 (0.004)**
		FedPAttn	0.10 (0.003)	**0.05 (0.001)**	0.11 (0.005)	**0.05 (0.002)**	**0.09 (0.000)**	**0.05 (0.001)**	**0.11 (0.001)**	**0.08 (0.001)**	0.14 (0.002)	0.12 (0.003)	0.21 (0.004)	**0.17 (0.004)**

4.5 Additional Results 3: Comparing performance in extreme non-IID setting

We replicate a non-IID setting by stratifying the data across the clients based on the non-overlapping quantile of the event time of the population where the first and last client, respectively, sees the shortest and longest survivals [13]. Table 7 shows that our models perform similarly or better than the baseline survival models. Even though DeepHit shows good Brier Score values in most of the dataset except for SUPPORT, it performs worst in terms of the C-Index.

4.6 Additional Results 4: Performance of DP-FedPLSTM

In this model, we employ differential privacy during both federated pseudo values computation and FL training of the FedPLSTM model. Table 8 demonstrates that DP-FedPLSTM performs similarly to the other DP-version (DP-FedPDNN) in terms of C-index and Brier Score.

4.7 Additional Results 5: Comparing training time of the models

We compare the training time of the models in both centralized and federated (IID and Non-IID) settings on real survival datasets on the same hyperparameter settings (e.g., fixed learning rate, batch size, communication round, number of clients, local epochs, and patience). Table 9 shows that our models require similar or less training time than the baseline model. In the high-dimensional META-HD data, our models take significantly less training time than the baseline models. However, the FedPAttnis more computationally expensive than the other proposed models due to its complex architecture.

ACKNOWLEDGMENTS

This work is supported by grants 1948399 and 2238743 from the US National Science Foundation (NSF).

REFERENCES

[1] Enrique Barrajon and Laura Barrajon. 2020. Effect of right censoring bias on survival analysis. *arXiv preprint arXiv:2012.08649* (2020).

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

Table 4. Significance test for checking the violation of the proportional hazard (PH) assumption and client-wise performance comparison of the models on the DSDUC dataset.

	Covariates	Client 1	Client 2	Client 3	Client 4	Client 5	Client 6	Client 7	Client 8	Client 9	Client 10	
Proportional Hazard Test	X1	130.3^{a}	131.7^{a}	142.7^{a}	123.2^{a}	124.6^{a}	92.4^{a}	68.7^{a}	47.4^{a}	28.7^{a}	12.0^{a}	
	X2	88.3^{a}	67.4^{a}	69.4^{a}	50.8^{a}	43.6^{a}	23.4^{a}	19.1^{a}	9.8^{b}	14.8^{a}	6.5^{b}	
	X3	57.1^{a}	66.6^{a}	65.5^{a}	70.9^{a}	65.5^{a}	33.8^{a}	30.5^{a}	12.9^{a}	6.0^{c}	1.3	
	X4	71.5^{a}	49.7^{a}	53.5^{a}	49.4^{a}	46.3^{a}	38.3^{a}	31.0^{a}	20.9^{a}	27.3^{a}	17.4^{a}	
	X5	0.02	1.48	0.59	4.79^{c}	3.8^{c}	2.20	5.1^{c}	1.50	1.10	0.01	
	X6	55.7^{a}	44.08^{a}	46.42^{a}	40.04^{a}	33.0^{a}	20.4^{a}	19.1^{a}	9.4^{b}	5.22^{c}	5.0^{c}	
	X7	0.22	4.59^{c}	3.13^{d}	15.87^{a}	17.84^{a}	9.7^{b}	10.3^{b}	6.8^{b}	4.58^{c}	2.40	
	X8	39.3^{a}	41.0^{a}	30.6^{a}	43.4^{a}	36.9^{a}	20.0^{a}	26.1^{a}	11.1^{a}	13.2^{a}	8.3^{a}	
	X9	97.8^{a}	100.6^{a}	99.0^{a}	117.5^{a}	114.1^{a}	89.2^{a}	73.2^{a}	45.0^{a}	29.3^{a}	12.0^{a}	
	X10	109.6^{a}	111.7^{a}	81.4^{a}	89.4^{a}	80.0^{a}	48.1^{a}	48.3^{a}	30.8^{a}	28.3^{a}	16.6^{a}	
	X11	38.4^{a}	48.0^{a}	48.3^{a}	51.0^{a}	48.1^{a}	38.0^{a}	33.4^{a}	18.9^{a}	14.5^{a}	6.3^{b}	
	X12	46.3^{a}	41.20^{a}	27.6^{a}	19.5^{a}	13.1^{a}	4.3^{c}	5.1^{c}	0.79	1.59^{a}	0.15	
	Overall	523.9^{a}	575.8^{a}	537.3^{a}	605.4^{a}	573.1^{a}	401.4^{a}	369.6^{a}	232.7^{a}	197.4^{a}	101.8^{a}	
Metric	**Model**	**Client 1**	**Client 2**	**Client 3**	**Client 4**	**Client 5**	**Client 6**	**Client 7**	**Client 8**	**Client 9**	**Client 10**	**Overall**
C-Index ↑	LinearPH	0.74	0.70	0.65	0.61	0.53	0.57	0.60	0.59	0.68	0.66	0.55
	NNph	0.85	0.83	0.78	0.68	0.58	0.65	0.67	0.74	0.84	0.86	0.63
	NNnph	0.87	0.85	0.79	0.73	**0.70**	0.74	0.73	0.80	**0.86**	0.86	**0.72**
	FedPDNN	**0.88**	0.85	**0.82**	0.72	0.69	**0.75**	0.73	0.78	0.85	**0.87**	0.70
	FedPLSTM	0.87	**0.86**	**0.82**	**0.74**	**0.70**	0.73	0.73	0.79	0.85	0.86	0.69
	FedPAttn	0.87	0.85	**0.82**	0.73	**0.70**	0.74	**0.76**	**0.81**	**0.86**	**0.87**	0.70
Brier Score ↓	LinearPH	0.16	0.20	0.21	0.24	0.24	0.22	0.23	0.24	0.24	0.24	0.25
	NNph	0.09	0.12	0.13	0.20	0.23	0.21	0.22	0.20	0.16	0.12	0.24
	NNnph	0.07	0.12	0.15	0.21	0.22	0.19	0.23	0.16	0.15	0.13	0.23
	FedPDNN	0.07	0.11	0.12	0.17	0.20	**0.15**	0.19	0.15	0.15	0.11	**0.17**
	FedPLSTM	0.09	**0.09**	0.11	**0.16**	**0.19**	0.18	0.23	0.15	0.15	0.11	**0.17**
	FedPAttn	**0.06**	**0.09**	**0.10**	**0.16**	**0.19**	**0.15**	**0.18**	**0.12**	**0.13**	**0.10**	**0.17**

Proportional hazard test (Chi-square) - statistically significant codes: 0 ’a’ 0.001 ’b’ 0.01 ’c’ 0.05 ’d’ 0.1 ’ ’ 1, (Read p ’*’ as significant at p% level of significance)

Table 5. Performance comparisons of the locally trained and federated FedPDNN models evaluated on the test set of the local clients’ data. Higher C-index and Lower Brier scores indicate better performance.

Data	Metric	Setup	IID					Non-IID				
			Client 1	Client 2	Client 3	Client 4	Client 5	Client 1	Client 2	Client 3	Client 4	Client 5
METABRIC	**C-Index**	Local pseudo value + Locally trained FedPDNN	**0.66**	0.57	0.53	0.44	0.62	0.56	0.42	0.55	0.41	0.46
		Federated pseudo value + Locally trained FedPDNN	0.64	0.63	0.52	0.62	0.65	**0.58**	**0.45**	0.55	**0.61**	**0.69**
		Federated pseudo value + Federated FedPDNN	0.61	**0.66**	**0.65**	**0.69**	**0.65**	0.56	0.38	**0.62**	0.60	**0.69**
	Brier	Local pseudo value + Locally trained FedPDNN	0.19	0.25	0.24	0.24	0.19	0.49	0.61	0.54	0.38	0.22
		Federated pseudo value + Locally trained FedPDNN	0.19	0.19	0.22	0.19	**0.18**	0.02	0.04	0.22	0.22	0.08
		Federated pseudo value + Federated FedPDNN	**0.19**	**0.16**	**0.19**	**0.18**	0.20	**0.01**	**0.04**	**0.06**	**0.04**	**0.08**
SUPPORT	**C-Index**	Local pseudo value + Locally trained FedPDNN	0.59	0.59	**0.62**	0.40	**0.63**	**0.53**	0.47	0.51	0.56	**0.67**
		Federated pseudo value + Locally trained FedPDNN	0.61	0.60	0.57	0.61	0.60	0.50	**0.49**	**0.53**	0.56	0.60
		Federated pseudo value + Federated FedPDNN	**0.61**	**0.60**	0.59	**0.61**	0.61	0.49	0.48	0.52	**0.58**	0.62
	Brier	Local pseudo value + Locally trained FedPDNN	0.19	0.25	0.24	0.24	0.20	0.33	0.31	0.36	0.43	0.06
		Federated pseudo value + Locally trained FedPDNN	0.19	0.21	0.21	**0.18**	0.21	0.20	0.00	0.04	0.15	0.07
		Federated pseudo value + Federated FedPDNN	**0.19**	**0.20**	**0.20**	0.19	**0.20**	**0.00**	**0.00**	**0.03**	**0.15**	**0.07**
GBSG	**C-Index**	Local pseudo value + Locally trained FedPDNN	0.63	0.53	0.57	0.70	0.66	0.49	0.49	0.47	NA	NA
		Federated pseudo value + Locally trained FedPDNN	0.63	0.53	0.57	0.69	0.65	0.49	0.45	0.43	NA	NA
		Federated pseudo value + Federated FedPDNN	**0.66**	**0.64**	**0.60**	**0.70**	**0.67**	**0.51**	**0.56**	**0.54**	NA	NA
	Brier	Local pseudo value + Locally trained FedPDNN	0.20	0.22	0.19	0.20	0.18	0.39	0.49	0.25	NA	NA
		Federated pseudo value + Locally trained FedPDNN	0.20	0.22	0.19	0.20	0.17	0.24	**0.10**	0.13	NA	NA
		Federated pseudo value + Federated FedPDNN	**0.19**	**0.20**	**0.19**	**0.20**	**0.18**	**0.05**	0.14	**0.12**	NA	NA
META-HD	**C-Index**	Local pseudo value + Locally trained FedPDNN	0.54	0.70	**0.66**	0.62	0.60	**0.55**	0.45	0.39	0.42	0.61
		Federated pseudo value + Locally trained FedPDNN	0.57	0.58	0.52	0.60	0.62	0.48	0.48	0.45	0.55	0.54
		Federated pseudo value + Federated FedPDNN	**0.63**	**0.70**	0.64	**0.63**	**0.68**	0.49	**0.53**	**0.64**	**0.59**	**0.64**
	Brier	Local pseudo value + Locally trained FedPDNN	0.22	0.19	0.20	0.17	0.20	0.51	0.34	0.58	0.46	0.18
		Federated pseudo value + Locally trained FedPDNN	0.21	0.24	0.22	0.17	0.22	0.01	0.18	0.05	0.09	0.18
		Federated pseudo value + Federated FedPDNN	**0.20**	**0.14**	**0.19**	**0.16**	**0.17**	**0.01**	**0.04**	**0.03**	**0.05**	**0.11**

[3] John A Foekens et al. 2000. The urokinase system of plasminogen activation and prognosis in 2780 breast cancer patients. *Cancer research* 60, 3 (2000), 636–643.

[4] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[5] Kaggle. 2022. *Breast Cancer METABRIC with Omics data*. Accessed: 2022-07-28.

[6] Jared L Katzman, Uri Shaham, Alexander Cloninger, Jonathan Bates, Tingting Jiang, and Yuval Kluger. 2018. DeepSurv: personalized treatment recommender

Table 6. Performance comparisons of the locally trained and federated Cox-based models with PH assumption evaluated on the test set of the local clients' data. Higher C-index and Lower Brier scores indicate better performance.

Dataset	Metric	Setup	IID Client 1	IID Client 2	IID Client 3	IID Client 4	IID Client 5	Non-IID Client 1	Non-IID Client 2	Non-IID Client 3	Non-IID Client 4	Non-IID Client 5
METABRIC	C-Index	Locally trained LinearPH	0.64	0.65	**0.62**	0.63	0.61	**0.60**	0.56	0.63	0.63	0.73
		Federated LinearPH	0.64	**0.67**	0.60	0.63	**0.64**	0.53	**0.59**	**0.64**	**0.64**	**0.77**
		Locally trained NNph	0.53	0.55	**0.66**	0.54	0.58	**0.47**	0.52	0.54	0.47	0.66
		Federated NNph	**0.69**	**0.69**	0.59	**0.62**	**0.64**	0.46	**0.59**	**0.65**	**0.65**	**0.68**
	Brier	Locally trained LinearPH	0.19	0.17	0.19	0.18	0.19	**0.01**	**0.04**	0.07	**0.02**	**0.07**
		Federated LinearPH	**0.18**	**0.16**	0.19	0.18	0.19	0.08	0.06	0.07	0.09	0.23
		Locally trained NNph	0.28	0.30	0.43	0.26	0.26	**0.04**	0.08	0.14	**0.07**	**0.11**
		Federated NNph	**0.17**	**0.18**	**0.19**	**0.18**	**0.19**	0.07	**0.06**	**0.08**	0.09	0.23
SUPPORT	C-Index	Locally trained LinearPH	0.60	0.60	0.57	0.60	0.61	0.52	0.51	0.52	0.60	0.69
		Federated LinearPH	0.60	0.60	0.57	**0.61**	0.61	0.52	0.51	0.52	**0.61**	0.69
		Locally trained NNph	0.59	0.57	0.54	0.58	0.60	0.51	**0.53**	**0.55**	0.60	0.62
		Federated NNph	**0.60**	**0.60**	**0.59**	**0.61**	**0.62**	**0.52**	0.52	0.51	0.60	**0.65**
	Brier	Locally trained LinearPH	0.20	0.20	0.22	0.20	0.21	0.00	0.00	0.03	**0.07**	**0.06**
		Federated LinearPH	0.20	0.20	**0.21**	**0.19**	0.21	0.00	0.00	0.03	0.08	0.13
		Locally trained NNph	0.22	0.23	0.25	0.20	0.24	0.00	0.00	**0.03**	0.09	**0.08**
		Federated NNph	**0.21**	**0.20**	**0.20**	**0.19**	**0.20**	0.00	0.00	0.04	0.09	0.14
GBSG	C-Index	Locally trained LinearPH	0.66	0.59	0.60	0.62	**0.72**	0.56	0.59	0.49	NA	NA
		Federated LinearPH	0.66	0.59	0.60	**0.63**	0.70	**0.57**	0.59	**0.53**	NA	NA
		Locally trained NNph	0.56	0.64	0.57	0.61	0.61	**0.58**	0.57	**0.59**	NA	NA
		Federated NNph	**0.62**	**0.66**	**0.59**	**0.66**	**0.73**	0.56	**0.60**	0.51	NA	NA
	Brier	Locally trained LinearPH	0.19	0.21	**0.18**	0.20	0.19	**0.05**	0.10	**0.05**	NA	NA
		Federated LinearPH	0.19	**0.20**	0.19	0.20	0.19	0.09	0.10	0.07	NA	NA
		Locally trained NNph	0.27	0.22	0.25	0.40	0.32	**0.06**	0.10	0.06	NA	NA
		Federated NNph	**0.21**	**0.19**	**0.20**	**0.21**	**0.18**	0.09	**0.09**	0.06	NA	NA
META-HD	C-Index	Locally trained LinearPH	0.62	**0.63**	**0.64**	0.58	0.57	**0.58**	0.54	0.57	0.50	**0.67**
		Federated LinearPH	**0.69**	0.50	0.61	**0.60**	**0.66**	0.49	**0.59**	**0.62**	**0.59**	0.51
		Locally trained NNph	0.62	0.60	0.55	0.61	0.46	**0.54**	0.50	0.58	0.47	**0.62**
		Federated NNph	**0.68**	**0.62**	**0.63**	**0.66**	**0.62**	0.50	**0.54**	**0.61**	**0.59**	0.53
	Brier	Locally trained LinearPH	0.24	**0.16**	0.19	**0.18**	**0.20**	**0.01**	**0.05**	**0.06**	**0.03**	**0.11**
		Federated LinearPH	**0.20**	0.23	0.19	0.20	0.22	0.02	0.06	0.11	0.14	0.41
		Locally trained NNph	**0.22**	**0.16**	0.22	0.18	0.21	0.01	0.09	**0.06**	**0.17**	**0.11**
		Federated NNph	0.23	0.27	**0.21**	**0.16**	**0.18**	0.01	**0.06**	0.17	0.29	0.61

Table 7. Performance comparisons [mean(sd)] of the models in extreme non-IID setting (The data across the clients are stratified based on the non-overlapping quantile of the event time of the population where the first and last client respectively sees the shortest and longest survivals [13]) on the real survival datasets.

Metric	Dataset	Baseline Models: LinearPH	Baseline Models: NNnph	Baseline Models: NNph	Baseline Models: DeepHit	Our Models with Non-DP Pseudo Values: FedPDNN	Our Models with Non-DP Pseudo Values: FedPLSTM	Our Models with Non-DP Pseudo Values: FedPAttn	Our Models with DP Pseudo Values: FedPDNN	Our Models with DP Pseudo Values: FedPLSTM	Our Models with DP Pseudo Values: FedPAttn	DP-FedPDNN
C-Index	METABRIC	0.60(0.013)	0.56(0.021)	0.56(0.022)	0.55(0.023)	0.59(0.054)	**0.62(0.007)**	0.60(0.017)	0.58(0.040)	0.61(0.023)	**0.62(0.020)**	0.55(0.014)
	SUPPORT	**0.58(0.005)**	0.55(0.020)	0.56(0.013)	0.54(0.011)	0.56(0.028)	**0.58(0.009)**	**0.58(0.019)**	**0.58(0.003)**	**0.58(0.004)**	0.57(0.015)	0.55(0.043)
	GBSG	0.62(0.002)	0.56(0.031)	0.57(0.021)	0.59(0.025)	0.60(0.033)	0.63(0.006)	0.61(0.012)	0.63(0.008)	**0.66(0.010)**	0.64(0.004)	0.51(0.030)
	META-HD	**0.57(0.011)**	0.56(0.024)	0.56(0.023)	0.54(0.021)	**0.57(0.038)**	0.54(0.010)	0.57(0.026)	0.55(0.044)	0.53(0.016)	0.55(0.013)	0.56(0.037)
Brier Score	METABRIC	0.23(0.013)	0.26(0.020)	0.25(0.023)	**0.22(0.019)**	**0.22(0.018)**	0.23(0.018)	**0.22(0.028)**	0.24(0.042)	0.24(0.017)	0.25(0.025)	0.27(0.030)
	SUPPORT	0.27(0.047)	0.27(0.024)	0.26(0.025)	0.32(0.055)	0.24(0.021)	**0.21(0.007)**	0.23(0.029)	0.27(0.041)	0.22(0.013)	0.23(0.009)	0.25(0.012)
	GBSG	0.22(0.002)	0.22(0.005)	0.22(0.002)	**0.21(0.001)**	0.31(0.064)	0.31(0.004)	0.29(0.048)	0.22(0.013)	0.23(0.014)	0.24(0.033)	0.22(0.007)
	META-HD	0.32(0.023)	0.25(0.006)	0.33(0.026)	**0.22(0.010)**	0.25(0.019)	0.23(0.021)	**0.22(0.010)**	0.24(0.021)	0.26(0.026)	0.25(0.035)	0.23(0.022)

Table 8. Performance of DP-FedPLSTM (DP is employed during federated pseudo values calculation and on the FedPLSTM model in FL training) on the real survival datasets.

Setup	METABRIC C-Index	METABRIC Brier Score	SUPPORT C-Index	SUPPORT Brier Score	GBSG C-Index	GBSG Brier Score	META-HD C-Index	META-HD Brier Score
Centralized	0.66 (0.004)	0.19 (0.001)	0.60 (0.002)	0.20 (0.002)	0.67 (0.002)	0.18 (0.001)	0.58 (0.024)	0.20 (0.001)
IID	0.64 (0.012)	0.18 (0.002)	0.60 (0.002)	0.20 (0.001)	0.66 (0.004)	0.18 (0.001)	0.57 (0.017)	0.20 (0.000)
Extreme Non-IID	0.53 (0.009)	0.21 (0.010)	0.58 (0.010)	0.21 (0.010)	0.63 (0.020)	0.21 (0.014)	0.50 (0.059)	0.21 (0.005)

system using a Cox proportional hazards deep neural network. *BMC medical research methodology* 18, 1 (2018), 1–12.

Table 9. Comparing training time of the models in both centralized and federated (IID and Non-IID) settings.

Dataset	Setup	Baseline Models				Our Models with Non-DP Pseudo Values		
		LinearPH	NNnph	NNph	DeepHit	FedPDNN	FedPLSTM	FedPAttn
METABRIC	Centralized	11.2	13.4	21.3	11.5	14.0	20.8	21.9
	IID	23.0	22.2	25.4	28.2	23.3	26.9	35.9
	Non-IID	26.8	28.2	31.9	40.0	24.4	26.3	32.7
SUPPORT	Centralized	28.6	15.0	28.2	16.8	28.4	70.2	63.6
	IID	45.5	47.1	52.2	195.7	55.6	72.3	91.6
	Non-IID	63.2	55.8	76.9	308.5	56.1	72.1	95.4
GBSG	Centralized	10.8	14.9	21.3	11.2	13.1	17.8	19.4
	IID	31.6	30.3	34.4	29.7	24.5	27.1	34.1
	Non-IID	25.9	24.8	28.7	31.8	23.9	27.8	33.2
META-HD	Centralized	95.7	95.3	91.9	82.7	92.2	85.5	219.2
	IID	900.9	905.1	920.7	971.9	240.6	282.3	699.6
	Non-IID	1049.7	998.0	1006.9	1062.8	236.3	279.1	694.1

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

[8] Luís Meira Machado and Susana Faria. 2014. A simulation study comparing modeling approaches in an illness-death multi-state model. (2014).

[9] Md Mahmudur Rahman et al. 2022. Fair and Interpretable Models for Survival Analysis. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1452–1462.

[10] Md Mahmudur Rahman, Koji Matsuo, Shinya Matsuzaki, and Sanjay Purushotham. 2021. Deeppseudo: Pseudo value based deep learning models for competing risk analysis. In *The Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI-21)*.

[11] M Schumacher, G Bastert, H Bojar, K Hübner, M Olschewski, W Sauerbrei, C Schmoor, C Beyerle, RL Neumann, and HF Rauschecker. 1994. Randomized 2 x 2 trial evaluating hormonal treatment and the duration of chemotherapy in node-positive breast cancer patients. German Breast Cancer Study Group. *Journal of Clinical Oncology* 12, 10 (1994), 2086–2093.

[12] Zhaohong Sun, Wei Dong, Jinlong Shi, Kunlun He, and Zhengxing Huang. 2021. Attention-Based Deep Recurrent Model for Survival Prediction. *ACM Transactions on Computing for Healthcare* 2, 4 (2021), 1–18.

[13] Dekai Zhang, Francesca Toni, and Matthew Williams. 2022. A Federated Cox Model with Non-Proportional Hazards. *arXiv preprint arXiv:2207.05050* (2022).

[14] Lili Zhao et al. 2020. Deep neural networks for survival analysis using pseudo values. *IEEE journal of biomedical and health informatics* 24, 11 (2020), 3308–3314.