

Original Paper

Privacy-Preserving Federated Survival Support Vector Machines for Cross-Institutional Time-To-Event Analysis: Algorithm Development and Validation

Julian Späth¹, MSc; Zeno Sewald², BSc; Niklas Probul¹, MSc; Magali Berland³, PhD; Mathieu Almeida³, PhD; Nicolas Pons³, PhD; Emmanuelle Le Chatelier³, PhD; Pere Ginès^{4,5,6,7}, MD, PhD; Cristina Solé^{4,5,6}, MD; Adrià Juanola^{4,5,6}, MD, PhD; Josch Pauling², PhD; Jan Baumbach¹, Prof Dr

¹Institute for Computational Systems Biology, University of Hamburg, Hamburg, Germany

²LipiTUM, Chair of Experimental Bioinformatics, TUM School of Life Sciences, Technical University of Munich, Freising, Germany

³MetaGenoPolis, INRAE, Université Paris-Saclay, Jouy-en-Josas, France

⁴Liver Unit, Hospital Clínic de Barcelona, Barcelona, Spain

⁵Institut d'Investigacions Biomèdiques August Pi i Sunyer (IDIBAPS), Barcelona, Spain

⁶Centro de Investigación en Red de Enfermedades hepáticas y Digestivas (CIBERehD), Madrid, Spain

⁷Faculty of Medicine and Health Sciences, University of Barcelona, Barcelona, Spain

Corresponding Author:

Julian Späth, MSc

Institute for Computational Systems Biology

University of Hamburg

Notkestrasse 9

Hamburg, 22607

Germany

Phone: 49 15750665331

Email: julian.alexander.spaeth@uni-hamburg.de

Abstract

Background: Central collection of distributed medical patient data is problematic due to strict privacy regulations. Especially in clinical environments, such as clinical time-to-event studies, large sample sizes are critical but usually not available at a single institution. It has been shown recently that federated learning, combined with privacy-enhancing technologies, is an excellent and privacy-preserving alternative to data sharing.

Objective: This study aims to develop and validate a privacy-preserving, federated survival support vector machine (SVM) and make it accessible for researchers to perform cross-institutional time-to-event analyses.

Methods: We extended the survival SVM algorithm to be applicable in federated environments. We further implemented it as a FeatureCloud app, enabling it to run in the federated infrastructure provided by the FeatureCloud platform. Finally, we evaluated our algorithm on 3 benchmark data sets, a large sample size synthetic data set, and a real-world microbiome data set and compared the results to the corresponding central method.

Results: Our federated survival SVM produces highly similar results to the centralized model on all data sets. The maximal difference between the model weights of the central model and the federated model was only 0.001, and the mean difference over all data sets was 0.0002. We further show that by including more data in the analysis through federated learning, predictions are more accurate even in the presence of site-dependent batch effects.

Conclusions: The federated survival SVM extends the palette of federated time-to-event analysis methods by a robust machine learning approach. To our knowledge, the implemented FeatureCloud app is the first publicly available implementation of a federated survival SVM, is freely accessible for all kinds of researchers, and can be directly used within the FeatureCloud platform.

(JMIR AI 2024;3:e47652) doi: [10.2196/47652](https://doi.org/10.2196/47652)

KEYWORDS

federated learning; survival analysis; support vector machine; machine learning; federated; algorithm; survival; FeatureCloud; predict; predictive; prediction; predictions; Implementation science; Implementation; centralized model; privacy regulation

Introduction

Accessing data to apply machine learning (ML) in biomedical settings is still challenging [1]. Large amounts of data exist in clinical settings but are scattered across numerous institutions. Due to strict privacy regulations, such as the General Data Protection Regulation (GDPR), this data cannot be easily shared or collected at a central institution [2]. This causes hurdles for cross-institutional biomedical analyses that depend on highly sensitive patient data. One example is time-to-event analysis, aiming to find parameters that prolong or shorten the time until a particular event, such as death, occurs [3]. In these studies, the event of interest does not necessarily occur for all samples, increasing the need for large sample sizes [4]. Until today, the need for large sample sizes and heterogeneous data for time-to-event studies is still mainly solved through traditional data sharing, leading to the central collection of various deidentified and anonymized data sets from different centers. Since using anonymized data in the training of ML models tends to weaken model performance [5], this comes with a tradeoff of data privacy and data quality, accelerating the need for alternative methods that keep data private and ensure the quality of the data [6].

In recent years, federated learning (FL) has become a feasible alternative to central data collection by enabling the training of models on distributed data sets. Instead of sharing sensitive data with a central institution, in FL, only insensitive model parameters are shared with a central aggregation server [7,8]. Therefore, each participating party calculates its own model with local model parameters on their local data. These local model parameters are then shared with the aggregator and aggregated into a global model. Afterward, the global model is shared again with each participant and can be updated in another iteration. The first and probably most widely used aggregation approach is the federated average [9], calculating the weighted mean of the exchanged model parameters. Besides using different aggregation approaches, FL can also be distinguished between horizontal and vertical learning, as well as cross-device and cross-silo learning. Horizontal learning describes FL on data with the same features but different samples, while vertical learning performs on the same samples but with different features between the participating parties. Cross-device FL trains models across millions of participants (such as mobile phones), cross-silo FL, on the other hand, focuses on a few clients only, such as hospitals or research institutes [10].

Especially in combination with privacy-enhancing techniques (PETs), model parameters can be exchanged securely, such that a global aggregator or potential attacker cannot even see the local parameters of each participant [11]. This secure exchange of model parameters is necessary to comply with the GDPR, as even local models can be considered personal data [12]. Therefore, FL enables the training on a significantly larger data set compared with single-institution scenarios. While federated algorithms still often struggle with communication efficiency,

the significantly increased amount of data can offset this performance issue, making FL a serious competitor to classical ML. Additionally, since FL models are trained on a larger variety of data, they typically generalize better than traditional ML models and even generalize faster in some cases [13,14]. Many FL approaches are already published for biomedical applications, such as medical imaging analysis, genome-wide association studies, or gene expression analysis [15-17].

In addition to federated ML approaches, several federated time-to-event analysis algorithms have been introduced recently and confirmed their high potential for privacy-preserving analyses [18-21]. However, existing approaches solely cover traditional statistical methods such as the estimation of survival functions and the Cox proportional hazards model. Modern ML algorithms for survival analysis, such as survival Support Vector Machines (SVMs), are not yet available in a federated fashion, even though SVMs belong to one of the most popular ML methods. If algorithms are not available in federated scenarios, this might be a reason why researchers chose not to perform FL, if their favorite algorithms are not available. Many well-performing centralized algorithms are challenging to translate to a federated scenario while keeping sensitive data private. Another limitation of FL is communication efficiency. FL algorithms need to exchange the intermediate statistics with a central aggregator, which is especially inefficient for algorithms with many iterations. This inefficiency even increases when adding secure aggregation schemes, such as additive secret sharing. This PET ensures that only masked and encrypted model parameters are shared with the aggregating party, securing the local models from data leakage [18].

To address the lack of availability of federated time-to-event methods, we propose a privacy-preserving, horizontally federated, cross-silo survival SVM based on the survival analysis package *scikit-survival* [22]. Compared with other existing time-to-event methods, such as the Cox proportional hazard model, the survival SVM allows an actual prediction of the time until an event happens. It can be used to predict the risk of individual samples, which is not possible in univariate time-to-event algorithms and is not the aim of the Cox proportional hazards model. Therefore, to the best of our knowledge, it is the first freely available federated survival prediction method. We implemented the algorithm as an app in the FeatureCloud platform to make it publicly accessible and to minimize the hurdles of FL infrastructure [23]. Based on a combination of FL and additive secret sharing, we show on 3 benchmark data sets, that our approach achieves highly similar results compared with central data analysis. Additionally, we apply it to a set of real-world microbiome data sets to demonstrate its applicability to original clinical data.

Methods

Here, we propose the adapted algorithm for the federated survival SVM, describe its implementation as a FeatureCloud app, and explain how we evaluated its performance.

Federated Survival SVM

We extended the regression objective of *scikit-survival*'s FastSurvivalSVM without ranking to be applicable in federated environments [24]. As shown in Figure 1, instead of calculating the sum of the squared ζ -function centrally, it is calculated at each site, with the feature vector x_i , the survival time $y_i > 0$, and the binary event indicator δ_i . Each site's local sum of squared ζ -function is then sent to a global aggregator and summed up to the global sum of squared ζ -function. The below equations show the central objective function and our corresponding federated objective function, with C being the set of all participating clients.

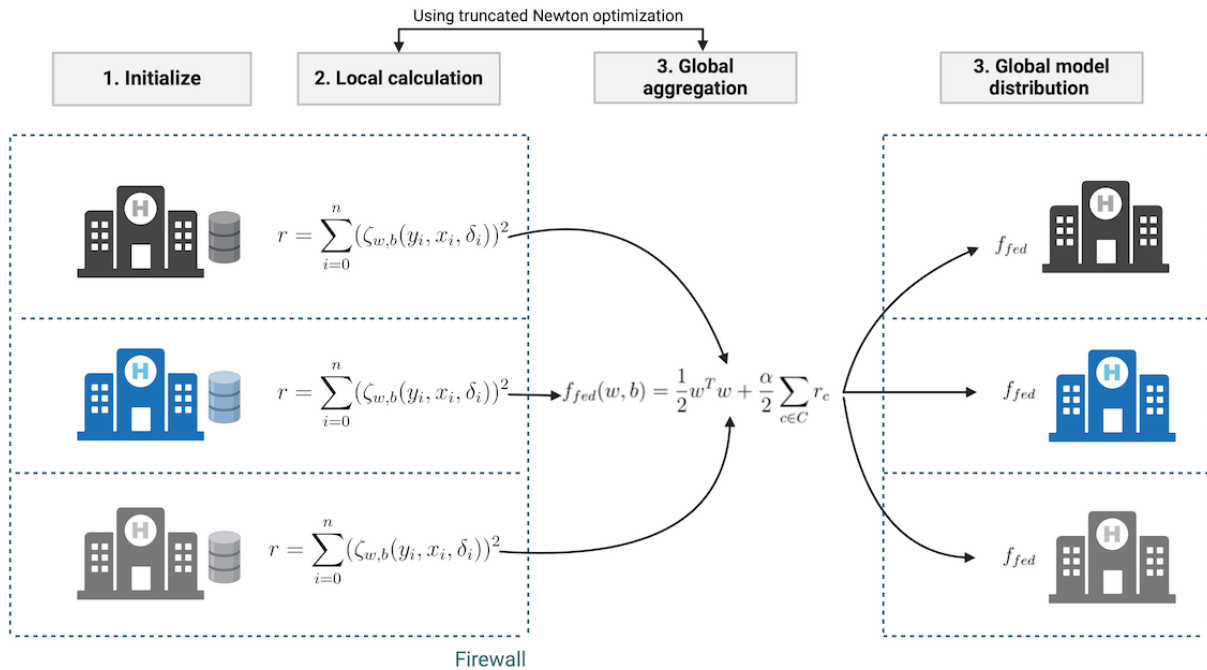
$$\operatorname{argmin}_{\omega, b} f_{\text{central}}(\omega, b) = \frac{1}{2} \omega^T \omega + \frac{\alpha}{2} \sum_{i=0}^n (\zeta_{\omega, b}(y_i, x_i, \delta_i))^2$$

$$\operatorname{argmin}_{\omega, b} f_{\text{fed}}(\omega, b) = \frac{1}{2} \omega^T \omega + \frac{\alpha}{2} \sum_{c \in C} \sum_{i=0}^{n_c} (\zeta_{\omega, b}(y_{c,i}, x_{c,i}, \delta_{c,i}))^2$$

$$\zeta_{\omega, b}(y_{c,i}, x_{c,i}, \delta_{c,i}) = \begin{cases} \max(0, y_i - \mathbf{w}^T \mathbf{x}_i - b) & \text{if } \delta_i = 0, \\ y_i - \mathbf{w}^T \mathbf{x}_i - b & \text{if } \delta_i = 1 \end{cases}$$

Mathematically, our federated formula leads to the same solution as the centralized calculation of the objective function. Similar to the centralized analysis, a truncated Newton method (such as Newton-CG) can be used to optimize the objective function. For this, in each iteration, the gradient and Hessian matrix of each client are also sent to the global aggregator to sum them up to the global gradient and Hessian matrix. To reduce potential privacy leakage from the exchanged data, the implementation of the federated algorithm should support a secure aggregation scheme that hides the locally exchanged data from attackers or the global aggregation server.

Figure 1. Federated calculation of a survival support vector machine (SVM). Each site calculates the sum of squares locally and sends it to the global aggregation server. The aggregation server aggregates the local sum of squares by summing them up to the global sum of squares. The objective function is minimized in a federated fashion by a truncated Newton approach. After convergence, the global model is distributed to all participating clients.



FeatureCloud

We developed an FL app on the FeatureCloud platform to make our approach publicly available. To develop this app, we used the app template and application programming interface provided by FeatureCloud [25]. Using the *scikit-survival* package and Python, we implemented our algorithm, put it into the FeatureCloud app template, and published it in the FeatureCloud artificial intelligence store. It can be used with other apps in a workflow or standalone using the platform. Our code is entirely open source.

In FeatureCloud, 1 participating client also takes the aggregating role and is called the coordinator. The app is implemented as a state machine, meaning that the app switches between states to perform different tasks. All states and their transitions are shown in Multimedia Appendix 1. After reading the local data and config files, minimizing the objective function using a federated Newton conjugate gradient is performed iteratively. Therefore, the local gradient and Hessian matrices are calculated and sent to the coordinator. The coordinator aggregates these data to obtain the global matrices, updates the weight vector ω , and broadcasts it to all clients. This is repeated until convergence.

A considerable advantage of the FeatureCloud platform is its native support of 2 very popular PETs, such as secure multiparty computation (SMPC). For applying SMPC, FeatureCloud supports a secure aggregation scheme for hiding locally exchanged parameters using additive secret sharing [26]. Through this, the exchanged local models are protected, and only the global aggregations are visible to attackers, clients, and the global aggregator. This is achieved by splitting the value that needs to be exchanged with the global aggregator into n shards, where n is the number of participating clients, and the sum of these n shards would result in the actual value [23]. Each shard is encrypted using a public key of each participant. These encrypted shards are shared with the global aggregator, sending them to the corresponding client holding the private key. The clients decrypt the received shards, sum them up, and send them back to the global aggregator, which sums up all received sums. This final sum results in the actual, nonhidden, global aggregate.

Ethical Considerations

According to German regulations, for our retrospective study performed on publicly available data or data with explicit consent, approval from an ethical committee was not required.

Evaluation

We evaluated our approach using the developed FeatureCloud app on 3 benchmark data sets, all available via the *scikit-survival* package. The breast cancer data set (BRCA) [27] contains the gene expression profiling of microarray experiments from 198 primary breast tumors, originally used to validate a 76-gene prognostic signature able to predict distant metastases in lymph node-negative patients with breast cancer. The German Breast Cancer Study Group 2 data set (GBSG2) [28] contains data from a multicenter randomized clinical trial to compare the effectiveness of 3 versus 6 cycles of cyclophosphamide, methotrexate, and fluorouracil on recurrence-free and overall

survival of 686 women. The observed parameters were hormonal therapy (yes or no), age of the patients, menopausal status (pre vs post), tumor size (in mm), tumor grade, number of positive tumor nodes, progesterone receptor (in fmol), and estrogen, as well as the censoring indicator and recurrence-free survival time (in days). The Worcester Heart Attack Study data set (WHAS500) [29] contains data from 500 patients with acute myocardial infarction, collected during thirteen 1-year periods. Parameters were age, gender, initial heart rate, initial systolic and diastolic blood pressure, body mass index, history of cardiovascular disease, atrial fibrillation, cardiogenic shock, congestive heart complications, complete heart block, myocardial infarction order and type, vital status, and total length of follow-up.

Additionally, we evaluated our algorithm on a recent, high-dimensional gut microbiome data set from the Hospital Clinic of Barcelona, containing data from 150 patients with liver cirrhosis [30]. The data set was aimed at assessing the predicting role of the gut microbiome for the survival of the patients in the context of liver cirrhosis, using shotgun metagenomic sequencing performed on fecal DNA isolated from stool samples. A former version of the data has been previously analyzed with a different methodology [30]. For this study, the Metagenomic Species Pangenome (MSP) was used to identify and quantify microbial species associated with the IGC2 reference catalog [31]. MSPs are clusters of coabundant genes (minimum size >100 genes) used as a proxy for microbial species, reconstructed from 1601 metagenomes to 1990 MSP species [32]. MSP abundances were estimated as the mean abundance of their 100 marker genes, as far as at least 20% of these genes are detected. The MSP abundance table was then normalized in each sample by dividing its abundance by the sum of MSP abundances detected in the sample. Further details regarding the data sets are shown in Table 1.

Table 1. Overview of all data sets. Our 4 evaluation data sets differ greatly in the number of samples, features, events, and censored individuals. Features indicate the number of clinical variables or microbial species abundance in the data set; median follow-up indicates the median follow-up time of the patients in days; events indicate the number of patients for whom the event of interest was observed during observation time; and censored indicates the number of patients for whom the event of interest was not observed during observation time.

Data set	Samples, n	Features, n	Median follow-up (days)	Events, n	Censored, n	End point
BRCA	198	84	4384.0	51	147	Presence of metastases
GBSG2	686	11	1084.0	299	387	Recurrence-free survival
WHAS500	500	16	631.5	215	285	Death
Microbiome	150	1995	416.0	51	99	Death

^aBRCA: breast cancer data set.

^bGBSG2: German Breast Cancer Study Group 2 data set.

^cWHAS500: Worcester Heart Attack Study data set.

We one-hot encoded nonbinary categorical features. For each data set, we created either 1 client (100%) as the centralized scenario, 3 clients (20%, 50%, and 30%) as the multicentric imbalanced scenario, and 5 clients (20% each) as the multicentric balanced scenario, and we split the data accordingly.

To evaluate the accuracy of our model, we used the Harrell concordance index, which was developed as a generalization of the area under the receiver operating characteristic curve for time-to-event models [33]. It corresponds to the probability of concordance between observed and predicted survival based on each pair of individuals. A c-index of 0.5 means that the model

performs as well as a random guess, and a c-index of 1.0 means that the model predicts perfectly well.

After preprocessing, we performed a 3×3 -fold cross-validation (CV) for a FeatureCloud workflow consisting of a federated normalization, the federated survival SVM, and a federated survival evaluation (c-index). We then compared our results with the centralized analysis of every client and the merged data set (simulating a central data collection). Centralized analysis was performed using *scikit-survival*'s FastSurvivalSVM with a rank ratio of 0, α of 0.0001, true fit intercept, and a maximum of 50 iterations. The same hyperparameters were used for the federated analysis, respectively.

Privacy

FeatureCloud supports several properties to increase the privacy and security of the computations. One important step is that FL projects can be only executed with invited participants. For this, a unique and secret code is needed to join the project. Every participant can see the workflow and each individually executed FeatureCloud app that will run in the workflow. As FeatureCloud apps are open source, even the executed code of the apps can be examined.

The execution of apps and workflows in FeatureCloud is containerized and strictly monitored. Due to the containerization, individual apps are not allowed to establish a connection to the internet, which prevents the extraction of data from malicious code. Even though the communication between clients does not contain sensitive patient information, it is RSA (Rivest–Shamir–Adleman) encrypted through the standard HTTPS protocol. This prevents unauthorized third parties from gaining insights into parameters exchanged during training.

Exchanged parameters from each individual site are masked through the secure aggregation scheme, hiding the intermediate statistics from other participating clients and the global aggregator. This efficiently addresses the problem of local models considered as personal data in GDPR [18].

Our federated survival SVM app currently uses a hybrid approach of SMPC and FL. This hybrid approach increases the privacy of the exchanged local parameters from both participants and potential attackers, as explained in the methods section.

Differential privacy (DP) [34] is not yet supported by FeatureCloud but is currently in development and could be added to the algorithm as an additional layer to improve privacy. However, as the app trains a linear model, it is less prone to overfit, reducing the surface for potential membership and attribute inference attacks [35]. In DP, noise is added to the model parameters during the training process to guarantee a mathematically quantifiable amount of privacy for each sample. While this comes with large advantages regarding privacy, the application of DP has also various weaknesses. The addition of noise lowers the performance of the model significantly,

especially when applying the amount of noise necessary for a meaningful level of privacy [36]. Further, this guarantee only is applicable for a limited number of interactions with the resulting model. As the final model is distributed to all participants, they can interact with the model arbitrarily, making the privacy guarantee void, thus not warranting an inclusion in this analysis.

A PET not supported by FeatureCloud currently is homomorphic encryption (HE), which allows the computation of the model on encrypted values, making sharing of data even more secure. While this is great in theory, it actually gains very little benefit in this analysis scenario. The data we share is already nonsensitive and through the use of SMPC, we can hide not only the data but the data's origin. This is why FeatureCloud currently supports SMPC instead of HE.

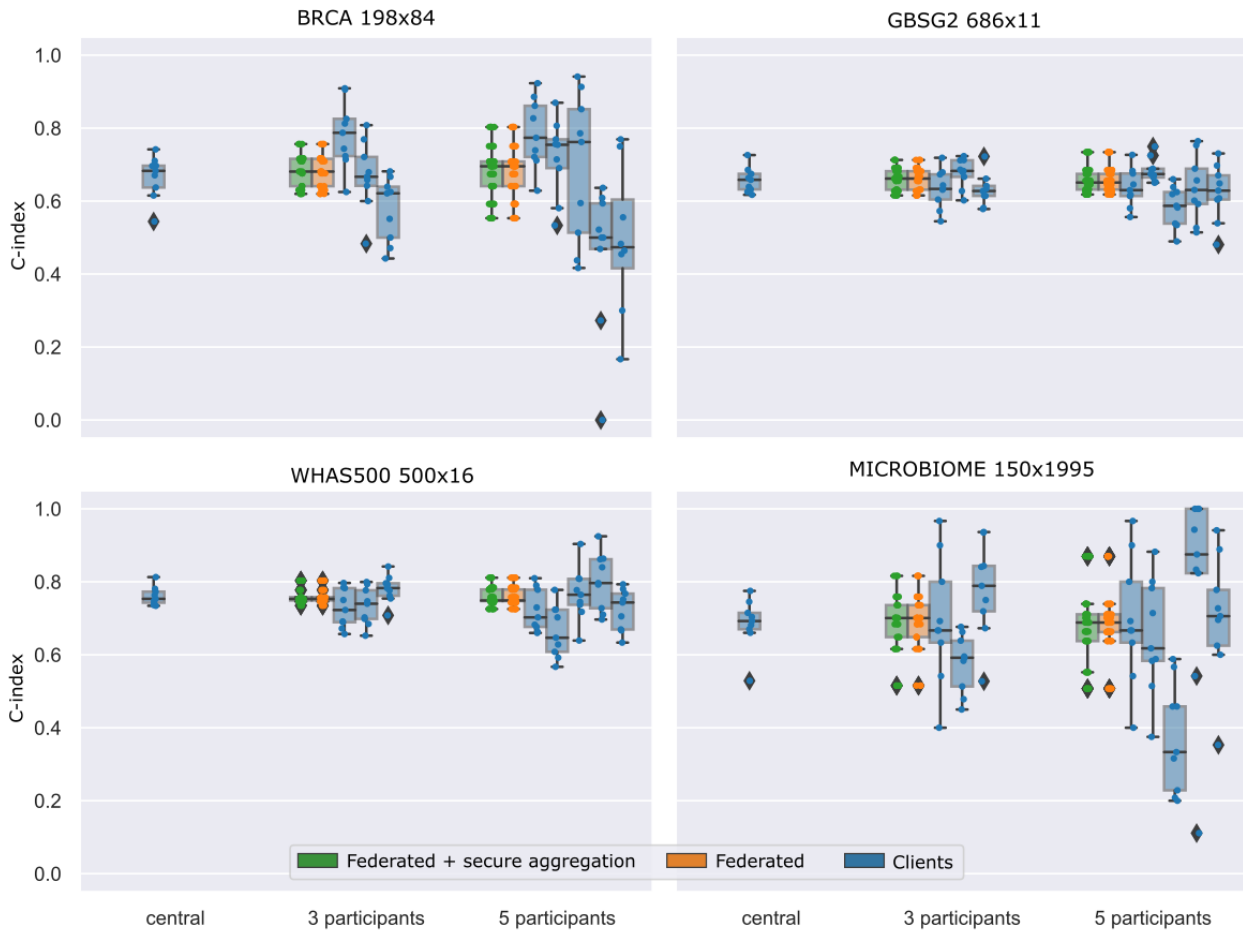
Our implementation of the federated survival SVM app uses all the functionalities offered by FeatureCloud and does not deviate from these best practices.

Results

Performance

Our workflow delivered a highly similar model performance and model parameters for all federated analyses compared with the ones performed on the corresponding centralized data sets. The resulting c-indices to estimate the performance of our time-to-event models are depicted in Figure 2 [33]. For each data set (subplot), we show a boxplot consisting of the evaluated c-index for each CV split of our federated workflow with secure aggregation (green), federated workflow without secure aggregation (orange), and centralized calculation for each individual client (blue). The CV results show that our federated as well as the federated and secure aggregation approach perform highly similar to the centralized estimates. The calculation of the federated c-index in FeatureCloud causes small deviances in the c-index between centralized and federated. This is because FeatureCloud calculates a local c-index and aggregates to the mean c-indices of all sites. Therefore, it does not lead to the same c-index as a central computation would. The mean c-indices for the 4 data sets are in the range between 0.658 (GDBG2) and 0.76 (WHAS500). In contrast to the accuracy, achieving very high c-indices is rather difficult and depends very much on the problem. In a bioinformatics context, the lowest c-index of 0.658 (GDBG2) can be considered as moderate. The model achieves discrimination between individuals with different survival outcomes. However, it might not be of clinical utility and needs further refinement. The c-index of 0.76 (WHAS500) on the other hand, can be considered as good and has predictive value. Improving the predictive value of the models and increasing c-index was out of the scope of this work. A complete table of the results is available in [Multimedia Appendix 2](#).

Figure 2. Comparison of federated and centralized analysis. The boxplots show the evaluated c-indices (3×3 -fold cross validation) of the central, 3 participants, and 5 participants analysis (rows). For each scenario, we compared the federated and secure aggregation approach (green), the federated-only approach (orange), and the performance of every single site (blue). BRCA: breast cancer data set; GBSG2: German Breast Cancer Study Group 2 data set; WHAS500: Worcester Heart Attack Study data set.



The model weights are nearly identical, with a maximum difference of only 0.001 and a mean difference of 0.0002 (Multimedia Appendices 1 and 3). These tiny differences between the weights of the central model and our model are negligible, as they do not change the overall prediction results and still lead to equal c-indices. The resulting model is therefore almost identical to the one that was trained on central data. A useful property of the linear survival SVM is, that the model weights can be used as a feature importance measure, which is also supported in our approach.

Besides calculating the feature importance from model weights directly, our federated survival SVM app uses Shapley additive explanations (SHAP), an explainable artificial intelligence framework for the interpretation of ML models [37]. Using SHAP, we compared the final models of the central, federated without secure aggregation, and federated with secure aggregation runs. For each data set, the SHAP shows highly similar model interpretations with a mean Pearson correlation of 0.991 between the central and the federated model without secure aggregation, and a mean Pearson correlation of 0.985 between the central model and the federated model with secure aggregation. A slightly worse correlation in the secure aggregation model is expected, as the masking of local parameters leads to floating-point issues. The worst correlation

is shown in the microbiome data set (0.964), which can be explained by the high correlation between features in this data set. The results of the SHAP correlation analysis are listed in Multimedia Appendix 4 and the corresponding SHAP beeswarm plots are available in Multimedia Appendix 5.

Our results further demonstrate the importance of large data sets, as the performance of the locally trained models on single clients (smaller sample size) shows a much higher variance than our federated models. If 5 institutes combine their small data sets, they can perform a much more reliable time-to-event analysis compared with isolated institutions. This further supports the high practical value of FL in real-world clinical time-to-event analysis, especially for institutions with small sample sizes, homogenous cohorts, or only a few patients with rare diseases.

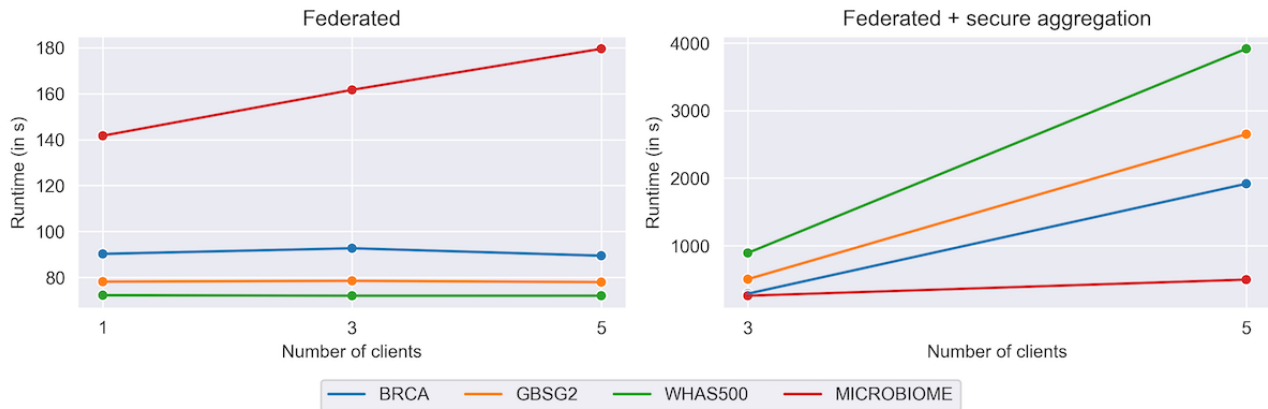
Runtime

As shown in Figure 3, the runtime largely depends on the data set. In the case of FL, the number of iterations and, therefore, the number of data exchanges are the bottleneck. While the federated-only approach has linear runtime, the runtime of federated and secure aggregation is much worse and increases with an increasing number of clients. As described in the FeatureCloud publication, providing better privacy by hiding

the exchanged parameters from the global aggregator, the simple additive secret sharing grows quadratic with the number of participants. Especially when many iterations and data

exchanges are needed, this has a bad influence on the runtime of the FL implementation.

Figure 3. Runtime analysis. The lines represent the runtime for each data set and the number of participating clients. The federated-only approach is depicted on the left, and the federated and secure aggregation approach is depicted on the right.



All results of the runtime analysis are shown in [Multimedia Appendix 6](#). Additionally, we performed the runtime analysis on a data set with a large sample size. As real-world time-to-event data sets are difficult to find, we used a synthetically generated, published data set from an example colon data set with 15,564 samples [38]. Our results show that our method scales well for large sample sizes, as the number of iterations is the bottleneck in FL ([Multimedia Appendix 7](#)).

FeatureCloud App

The app we developed can easily be used within the FeatureCloud platform. For this, a project coordinator creates a project, selects the app, and invites collaborators. Each participant installs FeatureCloud and joins the project. The app expects 2 CSV files as input, one for the training data and another for the test data. A config file can be used to define hyperparameters and other descriptors, such as the time and event label columns. After the federated computation has finished, each client receives the globally trained model as a pickle file, as well as a prediction file containing all predictions on the local test data set. The app can also be used in a FeatureCloud workflow, supporting various preprocessing methods, such as CV, normalization, feature selection, one-hot encoding, and subsequent evaluation of survival models using the c-index.

The requirements for running the survival SVM app are the same as for executing the FeatureCloud platform. It requires a stable internet connection to exchange the incentive model parameters with the central aggregator and to run the app on the website. Docker needs to be installed on a Mac, Linux, or Windows computer with the corresponding requirements for running Docker [39]. Moreover, enough memory should be available to process the data set. This depends mainly on the data set size, and not on the algorithm itself.

Discussion

Principal Findings

Our federated survival SVM has been demonstrated to offer a highly viable alternative to centralized data collection in a time-to-event analysis. It achieves comparable levels of accuracy without compromising the privacy of highly sensitive patient data. This makes it a compelling solution for organizations seeking to safeguard sensitive data while still gaining the benefits of advanced analysis and the application of ML. Through its availability as a FeatureCloud app, the platform takes care of deployment and federated infrastructures, making it directly usable with little programming knowledge. The results of the real-world microbiome data set are promising and show that FL might be an accelerator in microbiome research and the analysis of time-to-event microbiome data sets. Using FL combined with additive secret sharing, our approach can be currently considered GDPR compliant and, therefore, practically usable in real clinical time-to-event studies [12].

Comparison to Existing Work

Only a few federated survival analysis approaches were developed in recent years, such as the distributed Cox proportional hazards model WebDISCO or an approach for federated survival curves using multiparty HE [18,20]. In a recent study about privacy-aware multi-institutional time-to-event analysis, it was criticized that the existing work was mainly focusing on theoretical solutions, rather than practical [21]. Therefore, lack of usability was a huge issue that was addressed by the authors, who developed the platform "Partea" [21]. The platform supports the Kaplan-Meier estimator for survival curve estimation [40], Nelson-Aalen estimator for cumulative hazard ratios [41], and Cox proportional hazards model for survival regression [42]. Compared with "Partea," FeatureCloud does not only address the execution of FL algorithms, but also development. The FeatureCloud developer application programming interface for implementing FL algorithms that can be executed through FeatureCloud and published in the App Store is a huge advantage in terms of

development speed and also accessibility for the potential user group.

To our knowledge, the survival SVM FeatureCloud app is one of the first time-to-event analysis ML models implemented as a FL algorithm. This makes the accuracy (or c-index in our case) between the algorithms not directly comparable. However, similar to the existing solutions [20,21], our approach achieves almost identical results compared with the central algorithms.

Regarding runtime, univariate methods without iterations, such as Kaplan-Meier estimator, Nelson-Aalen estimator, or log-rank test are much more efficient in FL settings. However, these approaches cannot be used to analyze high dimensional data and multivariate settings. The efficiency of our approach is comparable to the iteratively trained Cox proportional hazard model, which is trained iteratively and requires communication and aggregation for every parameter update step.

Limitations

Our current approach does not support the more efficient ranking objective, as federated ranking is not trivial to implement. Instead, it is based on *scikit-survival*'s regression objective. Moreover, it solely supports the linear SVM and does not support the kernel SVM yet. Calculating a kernel matrix in a federated setting is not trivial, as it represents pairwise similarities (or distances) between the training data points. For supporting more complex, nonlinear relationships, this should be further investigated in the future. We still decided to implement and use a survival SVM in this work, as SVMs are very popular in health care and the only available time-to-event analysis ML model in *scikit-survival* that is not based on an ensemble approach. Ensemble models, such as random survival forests [43] or survival gradient boost, are both based on a set of survival trees. While ensemble models are also popular in time-to-event analysis, the federated aggregation of the local forests produces slightly worse results than centrally trained models in imbalanced scenarios [44]. A federated aggregation of each local tree, on the other hand, is computationally costly. The SVM in our implementation produces highly accurate results compared with central learning for model weights, c-index, and feature importance and can therefore lower the

burden of applying FL in health care (eg, microbiome analysis), as the participants can be sure that the results are equal to the ones they would obtain in a central setting.

FeatureCloud currently only supports a simple additive secret-sharing scheme, increasing runtime for calculations with many clients and iterations. This could be solved in the future by using a more efficient secret-sharing scheme, such as Shamir secret sharing, that is currently not supported by FeatureCloud [45]. By using FeatureCloud as the execution platform, our approach does not solve the still existing open problems of FL, such as fairness, debugging, and communication efficiency (especially when using secret sharing) [46]. Furthermore, there are attacks on FL architectures that cannot be prevented through the existing methods, such as privacy inference from the global model, and model or data poisoning [47]. It is therefore recommended to use the algorithms and FeatureCloud platform only with trusted parties.

Another limitation that comes from the FeatureCloud platform is data standardization. Data formatting and standards need to be discussed and determined in advance by the participants of the federated analysis. However, FeatureCloud provides the possibility to include federated data preprocessing applications in the workflow. While this does not remove the need for external communication of data standards, such as included features and naming conventions, it makes it straightforward to guarantee the same format and preprocessing for the used data before the actual model training process. Possible applications include imputation, normalization, train or test splitting, and CV [48,49].

Conclusions

In conclusion, we developed an open-source federated survival SVM that performs time-to-event analysis on geographically distributed data sets without sharing sensitive raw data. It is freely available in the FeatureCloud App Store. The trained models are almost identical compared with centrally trained survival SVMs. This extends the palette of existing federated time-to-event analysis approaches by another algorithm that can be applied to various problems.

Acknowledgments

This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement 826078. This publication reflects only the author's view and the European Commission is not responsible for any use that may be made of the information it contains (JB). This work was developed as part of the FeMAI project and is funded by the German Federal Ministry of Education and Research (BMBF) under grant 01IS21079 (NP) and by the Agence Nationale de la Recherche (ANR) under grant ANR-21-FAI1-0010. MB and MA were also supported by the grant ANR-11-DPBS-0001. JB was partially funded by his VILLUM Young Investigator Grant (13154). PG has received funds from the Instituto de Salud Carlos III through the Plan Estatal de Investigación Científica y Técnica y de Innovación, project references PI 16/00043 and PI 20/00579. These grants were cofunded by the European Regional Development Fund (FEDER) and also funded in part by an EU Horizon 2020 Programme (H2020-SC1-2016-RTD), LIVERHOPE (731875). JKP is funded by the Bavarian State Ministry of Education and the Arts in the framework of the Bavarian Research Institute for Digital Transformation (bidt, grant LipiTUM)

Data Availability

The data sets generated and analyzed during this study are available in the GitHub repository [50]. The code for the implementation of the federated survival SVM is available in the GitHub repository [51]. The microbiome data set is not publicly available due to privacy regulations but is available from the corresponding author on reasonable request.

Conflicts of Interest

CS has received speaking fees from Abbvie and Grifols. PG has received research funding from Gilead & Grifols. PG has consulted or attended advisory boards for Gilead, RallyBio, SeaBeLife, Merck, Sharp and Dohme (MSD), Ocelot Bio, Behring, Roche Diagnostics International and Boehringer Ingelheim, and received speaking fees from Pfizer.

Multimedia Appendix 1

State workflow of the survival support vector machine (SVM) FeatureCloud app and difference between coefficients.

[\[DOCX File, 244 KB-Multimedia Appendix 1\]](#)

Multimedia Appendix 2

C-indices of central, federated, and federated + secure aggregation analyses.

[\[XLSX File \(Microsoft Excel File\), 32 KB-Multimedia Appendix 2\]](#)

Multimedia Appendix 3

Coefficients of the trained survival support vector machines (SVMs).

[\[XLSX File \(Microsoft Excel File\), 243 KB-Multimedia Appendix 3\]](#)

Multimedia Appendix 4

Correlation of Shapley additive explanations (SHAP) values between central, federated, and federated + secure aggregation model.

[\[XLSX File \(Microsoft Excel File\), 10 KB-Multimedia Appendix 4\]](#)

Multimedia Appendix 5

Shapley additive explanations (SHAP) beeswarm plots for the different models.

[\[ZIP File \(Zip Archive\), 25020 KB-Multimedia Appendix 5\]](#)

Multimedia Appendix 6

Runtime of the federated survival support vector machine (SVM) training with 1, 3, and 5 clients.

[\[XLSX File \(Microsoft Excel File\), 11 KB-Multimedia Appendix 6\]](#)

Multimedia Appendix 7

Runtime of the federated survival support vector machine (SVM) with 1, 3, and 5 clients of a large sample size synthetic data set.

[\[XLSX File \(Microsoft Excel File\), 10 KB-Multimedia Appendix 7\]](#)

References

1. Adibuzzaman M, DeLaurentis P, Hill J, Benneyworth BD. Big data in healthcare—the promises, challenges and opportunities from a research perspective: a case study with a model database. *AMIA Annu Symp Proc.* 2017;2017:384-392. [Medline: [29854102](#)]
2. Vlahou A, Hallinan D, Apweiler R, Argiles A, Beige J, Benigni A, et al. Data sharing under the general data protection regulation: time to harmonize law and research ethics? *Hypertension.* 2021;77(4):1029-1035. [FREE Full text] [doi: [10.1161/HYPERTENSIONAHA.120.16340](#)] [Medline: [33583200](#)]
3. Greenhouse JB, Stangl D, Bromberg J. An introduction to survival analysis: statistical methods for analysis of clinical trial data. *J Consult Clin Psychol.* 1989;57(4):536-544. [doi: [10.1037//0022-006x.57.4.536](#)] [Medline: [2768615](#)]
4. Prinja S, Gupta N, Verma R. Censoring in clinical trials: review of survival analysis techniques. *Indian J Community Med.* 2010;35(2):217-221. [FREE Full text] [doi: [10.4103/0970-0218.66859](#)] [Medline: [20922095](#)]
5. Díaz JSP, García ÁL. Comparison of machine learning models applied on anonymized data with different techniques. *IEEE*; 2023. Presented at: 2023 IEEE International Conference on Cyber Security and Resilience (CSR); 31 July 2023 - 02 August 2023;618-623; Venice, Italy. URL: <https://ieeexplore.ieee.org/document/10224917> [doi: [10.1109/csr57506.2023.10224917](#)]
6. Antman E. Data sharing in research: benefits and risks for clinicians. *BMJ.* 2014;348:g237. [doi: [10.1136/bmj.g237](#)] [Medline: [24458978](#)]
7. Kairouz P, McMahan HB, Avent B, Bellet A, Bennis M, Nitin BA, et al. Advances and open problems in federated learning. In: *Foundations and Trends® in Machine Learning.* Boston, Massachusetts. Now Foundations and Trends; 2021;1-210.

8. Bonawitz K, Kairouz P, McMahan B, Ramage D. Federated learning and privacy: building privacy-preserving systems for machine learning and data science on decentralized data. *Queueing*. 2021;19(5):87-114. [FREE Full text] [doi: [10.1145/3494834.3500240](https://doi.org/10.1145/3494834.3500240)]
9. McMahan B, Ramage D. Federated learning: collaborative machine learning without centralized training data. Google Research. 2017. URL: <https://blog.research.google/2017/04/federated-learning-collaborative.html> [accessed 2024-02-13]
10. McMahan B, Moore E, Ramage D, Hampson S, Arcas BAY. Communication-efficient learning of deep networks from decentralized data. *PMLR*. 2017;54:1273-1282. Singh A, Zhu J, editors.
11. Torkzadehmahani R, Nasirigerdeh R, Blumenthal DB, Kacprowski T, List M, Matschinske J, et al. Privacy-preserving artificial intelligence techniques in biomedicine. *Methods Inf Med*. 2022;61(S 01):e12-e27. [FREE Full text] [doi: [10.1055/s-0041-1740630](https://doi.org/10.1055/s-0041-1740630)] [Medline: [35062032](https://pubmed.ncbi.nlm.nih.gov/35062032/)]
12. Brauneck A, Schmalhorst L, Majdabadi MMK, Bakhtiari M, Völker U, Saak CC, et al. Federated machine learning in data-protection-compliant research. *Nat Mach Intell*. 2023;5(1):2-4. Springer Science and Business Media LLC. [doi: [10.1038/s42256-022-00601-5](https://doi.org/10.1038/s42256-022-00601-5)]
13. Yang A, Ma Z, Zhang C, Han Y, Hu Z, Zhang W, et al. Review on application progress of federated learning model and security hazard protection. *Digit Commun Netw*. 2023;9(1):146-158. [FREE Full text] [doi: [10.1016/j.dcan.2022.11.006](https://doi.org/10.1016/j.dcan.2022.11.006)]
14. Asad M, Moustafa A, Ito T. Federated learning versus classical machine learning: a convergence comparison. *ArXiv*. Preprint posted online on 22 Jul 2021. [FREE Full text] [doi: [10.22541/au.162074596.66890690/v1](https://doi.org/10.22541/au.162074596.66890690/v1)]
15. Sheller MJ, Edwards B, Reina GA, Martin J, Pati S, Kotrotsou A, et al. Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data. *Sci Rep*. 2020;10(1):12598. [FREE Full text] [doi: [10.1038/s41598-020-69250-1](https://doi.org/10.1038/s41598-020-69250-1)] [Medline: [32724046](https://pubmed.ncbi.nlm.nih.gov/32724046/)]
16. Zolotareva O, Nasirigerdeh R, Matschinske J, Torkzadehmahani R, Bakhtiari M, Frisch T, et al. Flimma: a federated and privacy-aware tool for differential gene expression analysis. *Genome Biol*. 2021;22(1):338. [FREE Full text] [doi: [10.1186/s13059-021-02553-2](https://doi.org/10.1186/s13059-021-02553-2)] [Medline: [34906207](https://pubmed.ncbi.nlm.nih.gov/34906207/)]
17. Nasirigerdeh R, Torkzadehmahani R, Matschinske J, Frisch T, List M, Späth J, et al. sPLINK: a hybrid federated tool as a robust alternative to meta-analysis in genome-wide association studies. *Genome Biol*. 2022;23(1):32. [FREE Full text] [doi: [10.1186/s13059-021-02562-1](https://doi.org/10.1186/s13059-021-02562-1)] [Medline: [35073941](https://pubmed.ncbi.nlm.nih.gov/35073941/)]
18. Lu CL, Wang S, Ji Z, Wu Y, Xiong L, Jiang X, et al. WebDISCO: a web service for distributed cox model learning without patient-level data sharing. *J Am Med Inform Assoc*. 2015;22(6):1212-1219. [FREE Full text] [doi: [10.1093/jamia/ocv083](https://doi.org/10.1093/jamia/ocv083)] [Medline: [26159465](https://pubmed.ncbi.nlm.nih.gov/26159465/)]
19. Andreux M, Manoel A, Menuet R, Saillard C, Simpson C. Federated survival analysis with discrete-time cox models. *ArXiv*. Preprint posted online on 16 Jun 2020. [FREE Full text]
20. Froelicher D, Troncoso-Pastoriza JR, Raisaro JL, Cuendet MA, Sousa JS, Cho H, et al. Truly privacy-preserving federated analytics for precision medicine with multiparty homomorphic encryption. *Nat Commun*. 2021;12(1):5910. [FREE Full text] [doi: [10.1038/s41467-021-25972-y](https://doi.org/10.1038/s41467-021-25972-y)] [Medline: [34635645](https://pubmed.ncbi.nlm.nih.gov/34635645/)]
21. Späth J, Matschinske J, Kamanu FK, Murphy SA, Zolotareva O, Bakhtiari M, et al. Privacy-aware multi-institutional time-to-event studies. *PLOS Digit Health*. 2022;1(9):e0000101. [FREE Full text] [doi: [10.1371/journal.pdig.0000101](https://doi.org/10.1371/journal.pdig.0000101)] [Medline: [36812603](https://pubmed.ncbi.nlm.nih.gov/36812603/)]
22. Pölsterl S. scikit-survival: a library for time-to-event analysis built on top of scikit-learn. *J Mach Learn Res*. 2020;21(1):8747-8752. [FREE Full text]
23. Matschinske J, Späth J, Bakhtiari M, Probul N, Majdabadi MMK, Nasirigerdeh R, et al. The FeatureCloud platform for federated learning in biomedicine: unified approach. *J Med Internet Res*. 2023;25:e42621. [FREE Full text] [doi: [10.2196/42621](https://doi.org/10.2196/42621)] [Medline: [37436815](https://pubmed.ncbi.nlm.nih.gov/37436815/)]
24. Pölsterl S, Navab N, Katouzian A. Fast training of support vector machines for survival analysis. In: *Machine Learning and Knowledge Discovery in Databases*. Cham, Switzerland. Springer International Publishing; 2015;243-259.
25. FeatureCloud AI Developer API (1.1.0). FeatureCloud. URL: <https://featurecloud.ai/assets/api/redoc-static.html> [accessed 2024-01-13]
26. Cramer R, Damgard IB, Nielsen JB. *Secure Multiparty Computation and Secret Sharing*. Cambridge, England. Cambridge University Press; 2015.
27. Desmedt C, Piette F, Loi S, Wang Y, Lallemand F, Haibe-Kains B, et al. Strong time dependence of the 76-gene prognostic signature for node-negative breast cancer patients in the TRANSBIG multicenter independent validation series. *Clin Cancer Res*. 2007;13(11):3207-3214. [FREE Full text] [doi: [10.1158/1078-0432.CCR-06-2765](https://doi.org/10.1158/1078-0432.CCR-06-2765)] [Medline: [17545524](https://pubmed.ncbi.nlm.nih.gov/17545524/)]
28. Schumacher M, Bastert G, Bojar H, Hübner K, Olschewski M, Sauerbrei W, et al. Randomized 2 x 2 trial evaluating hormonal treatment and the duration of chemotherapy in node-positive breast cancer patients. German Breast Cancer Study Group. *J Clin Oncol*. 1994;12(10):2086-2093. [doi: [10.1200/JCO.1994.12.10.2086](https://doi.org/10.1200/JCO.1994.12.10.2086)] [Medline: [7931478](https://pubmed.ncbi.nlm.nih.gov/7931478/)]
29. Hosmer DW, Lemeshow S, May S. *Applied Survival Analysis: Regression Modeling of Time to Event Data: Second Edition*. New York, NY. John Wiley and Sons Inc; 2008.
30. Solé C, Guilly S, Da Silva K, Llopis M, Le-Chatelier E, Huelin P, et al. Alterations in gut microbiome in cirrhosis as assessed by quantitative metagenomics: relationship with acute-on-chronic liver failure and prognosis. *Gastroenterology*. 2021;160(1):206.e13-218.e13. [FREE Full text] [doi: [10.1053/j.gastro.2020.08.054](https://doi.org/10.1053/j.gastro.2020.08.054)] [Medline: [32941879](https://pubmed.ncbi.nlm.nih.gov/32941879/)]

31. Wen C, Zheng Z, Shao T, Liu L, Xie Z, Le Chatelier E, et al. Quantitative metagenomics reveals unique gut microbiome biomarkers in ankylosing spondylitis. *Genome Biol.* 2017;18(1):142. [FREE Full text] [doi: [10.1186/s13059-017-1271-6](https://doi.org/10.1186/s13059-017-1271-6)] [Medline: [28750650](https://pubmed.ncbi.nlm.nih.gov/28750650/)]
32. Oñate FP, Le Chatelier E, Almeida M, Cervino ACL, Gauthier F, Magoulès F, et al. MSPminer: abundance-based reconstitution of microbial pan-genomes from shotgun metagenomic data. *Bioinformatics.* 2019;35(9):1544-1552. [FREE Full text] [doi: [10.1093/bioinformatics/bty830](https://doi.org/10.1093/bioinformatics/bty830)] [Medline: [30252023](https://pubmed.ncbi.nlm.nih.gov/30252023/)]
33. Harrell FE, Califf RM, Pryor DB, Lee KL, Rosati RA. Evaluating the yield of medical tests. *JAMA.* 1982;247(18):2543-2546. [Medline: [7069920](https://pubmed.ncbi.nlm.nih.gov/7069920/)]
34. Dwork C. Differential privacy. In: *Automata, Languages and Programming.* Berlin, Heidelberg. Springer; 2006;1-12.
35. Yeom S, Giacomelli I, Fredrikson M, Jha S. Privacy risk in machine learning: analyzing the connection to overfitting. *IEEE*; 2018. Presented at: 2018 IEEE 31st Computer Security Foundations Symposium (CSF); July 09-12, 2018;268-282; Oxford, UK. URL: <https://ieeexplore.ieee.org/abstract/document/8429311/> [doi: [10.1109/csf.2018.00027](https://doi.org/10.1109/csf.2018.00027)]
36. Hsu J, Gaboardi M, Haebleren A, Khanna S, Narayan A, Pierce BC, et al. Differential privacy: an economic method for choosing epsilon. *IEEE*; 2014. Presented at: 2014 IEEE 27th Computer Security Foundations Symposium; July 19-22, 2014;398-410; Vienna, Austria. URL: <https://ieeexplore.ieee.org/abstract/document/6957125/> [doi: [10.1109/csf.2014.35](https://doi.org/10.1109/csf.2014.35)]
37. Lundberg SM, Lee SI. A unified approach to interpreting model predictions. USA. Curran Associates Inc; 2017. Presented at: Proceedings of the 31st International Conference on Neural Information Processing Systems Red Hook; 2017;4768-4777; NY, USA. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf
38. Smith A, Lambert PC, Rutherford MJ. Generating high-fidelity synthetic time-to-event datasets to improve data transparency and accessibility. *BMC Med Res Methodol.* 2022;22(1):176. [FREE Full text] [doi: [10.1186/s12874-022-01654-1](https://doi.org/10.1186/s12874-022-01654-1)] [Medline: [35739465](https://pubmed.ncbi.nlm.nih.gov/35739465/)]
39. Merkel D. Docker: Lightweight Linux Containers for Consistent Development and Deployment. Linux J Houston, TX. Belltown Media; 2014. URL: <https://www.seltzer.com/margo/teaching/CS508.19/papers/merkel14.pdf> [accessed 2024-03-06]
40. Kaplan EL, Meier P. Nonparametric estimation from incomplete observations. *J Am Stat Assoc.* 1958;53(282):457-481. [doi: [10.1080/01621459.1958.10501452](https://doi.org/10.1080/01621459.1958.10501452)]
41. Aalen O. Nonparametric inference for a family of counting processes. *Ann Statist.* 1978;6(4):701-726. [doi: [10.1214/aos/1176344247](https://doi.org/10.1214/aos/1176344247)]
42. Cox D. Regression models and life-tables. *J R Stat Soc.* 1972;34(2):187-202. [FREE Full text] [doi: [10.1007/978-1-4612-4380-9_37](https://doi.org/10.1007/978-1-4612-4380-9_37)]
43. Ishwaran H, Kogalur UB, Blackstone EH, Lauer MS. Random survival forests. *Ann Appl Stat.* 2008;2(3):841-860. [doi: [10.1214/08-aos169](https://doi.org/10.1214/08-aos169)]
44. Hauschild AC, Lemanczyk M, Matschinske J, Frisch T, Zolotareva O, Holzinger A, et al. Federated random forests can improve local performance of predictive models for various healthcare applications. *Bioinformatics.* 2022;38(8):2278-2286. [FREE Full text] [doi: [10.1093/bioinformatics/btac065](https://doi.org/10.1093/bioinformatics/btac065)] [Medline: [35139148](https://pubmed.ncbi.nlm.nih.gov/35139148/)]
45. Shamir A. How to share a secret. *Commun ACM.* 1979;22(11):612-613. [FREE Full text] [doi: [10.1145/359168.359176](https://doi.org/10.1145/359168.359176)]
46. Kairouz P, Brendan MH, Avent B, Bellet A, Bennis M, Bhagoji AN, et al. Advances and open problems in federated learning. *ArXiv. Preprint posted online on 9 Mar 2021.* [FREE Full text] [doi: [10.1561/9781680837896](https://doi.org/10.1561/9781680837896)]
47. Liu P, Xu X, Wang W. Threats, attacks and defenses to federated learning: issues, taxonomy and perspectives. *Cybersecurity.* 2022;5(1):1-19. [FREE Full text] [doi: [10.1186/s42400-021-00105-6](https://doi.org/10.1186/s42400-021-00105-6)]
48. Normalization app. FeatureCloud. 2022. URL: <https://featurecloud.ai/app/normalization> [accessed 2024-01-13]
49. Cross validation app. FeatureCloud. 2022. URL: <https://featurecloud.ai/app/cross-validation> [accessed 2024-01-13]
50. Späth J. julianspaeth / federated-survival-svm. GitHub. URL: <https://github.com/julianspaeth/federated-survival-svm> [accessed 2024-03-21]
51. Späth J. FeatureCloud / fc-survival-svm. GitHub. URL: <https://github.com/FeatureCloud/fc-survival-svm> [accessed 2024-03-21]

Abbreviations

- BRCA:** breast cancer data set
- CV:** cross-validation
- DP:** differential privacy
- FL:** federated learning
- GBSG2:** German Breast Cancer Study Group 2 data set
- GDPR:** General Data Protection Regulation
- HE:** homomorphic encryption
- ML:** machine learning
- MSP:** Metagenomic Species Pangenome
- PET:** privacy-enhancing technique

RSA: Rivest–Shamir–Adleman
SHAP: Shapley additive explanations
SMPC: secure multiparty computation
SVM: support vector machine
WHAS500: Worcester Heart Attack Study data set

Edited by K El Emam, B Malin; submitted 30.03.23; peer-reviewed by N Mungoli, S Nagavally, R Gorantla, D Gopukumar, X Jiang, Y Huang; comments to author 02.07.23; revised version received 06.08.23; accepted 10.02.24; published 29.03.24

Please cite as:

*Späth J, Sewald Z, Probul N, Berland M, Almeida M, Pons N, Le Chatelier E, Ginès P, Solé C, Juanola A, Pauling J, Baumbach J
Privacy-Preserving Federated Survival Support Vector Machines for Cross-Institutional Time-To-Event Analysis: Algorithm Development and Validation*

JMIR AI 2024;3:e47652

URL: <https://ai.jmir.org/2024/1/e47652>

doi: [10.2196/47652](https://doi.org/10.2196/47652)

PMID:

©Julian Späth, Zeno Sewald, Niklas Probul, Magali Berland, Mathieu Almeida, Nicolas Pons, Emmanuelle Le Chatelier, Pere Ginès, Cristina Solé, Adrià Juanola, Josch Pauling, Jan Baumbach. Originally published in JMIR AI (<https://ai.jmir.org>), 29.03.2024. This is an open-access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work, first published in JMIR AI, is properly cited. The complete bibliographic information, a link to the original publication on <https://www.ai.jmir.org/>, as well as this copyright and license information must be included.