*Journal of the American Medical Informatics Association*, 26(5), 2019, 392–403
doi: 10.1093/jamia/ocy180
Advance Access Publication Date: 20 March 2019
Research and Applications

Research and Applications

# Fair compute loads enabled by blockchain: sharing models by alternating client and server roles

**Tsung-Ting Kuo,[1] Rodney A. Gabriel,[1,2] and Lucila Ohno-Machado[1,3]**

[1]UCSD Health Department of Biomedical Informatics, University of California, San Diego, La Jolla, California, USA, [2]Department of Anesthesiology, University of California, San Diego, San Diego, California, USA, and [3]Division of Health Services Research & Development, VA San Diego Healthcare System, La Jolla, California, USA

Corresponding Author: Lucila Ohno-Machado, MD, PhD, 9500 Gilman Dr, San Diego, CA, USA (lohnomachado@ucsd.edu)

Received 16 October 2018; Revised 16 October 2018; Editorial Decision 28 November 2018; Accepted 2 December 2018

## ABSTRACT

**Objective**: Decentralized privacy-preserving predictive modeling enables multiple institutions to learn a more generalizable model on healthcare or genomic data by sharing the partially trained models instead of patient-level data, while avoiding risks such as single point of control. State-of-the-art blockchain-based methods remove the "server" role but can be less accurate than models that rely on a server. Therefore, we aim at developing a general model sharing framework to preserve predictive correctness, mitigate the risks of a centralized architecture, and compute the models in a fair way. **Materials and Methods**: We propose a framework that includes both server and "client" roles to preserve correctness. We adopt a blockchain network to obtain the benefits of decentralization, by alternating the roles for each site to ensure computational fairness. Also, we developed GloreChain (Grid Binary LOgistic REgression on Permissioned BlockChain) as a concrete example, and compared it to a centralized algorithm on 3 healthcare or genomic datasets to evaluate predictive correctness, number of learning iterations and execution time. **Results**: GloreChain performs exactly the same as the centralized method in terms of correctness and number of iterations. It inherits the advantages of blockchain, at the cost of increased time to reach a consensus model. **Discussion**: Our framework is general or flexible and can also address intrinsic challenges of blockchain networks. Further investigations will focus on higher-dimensional datasets, additional use cases, privacy-preserving quality concerns, and ethical, legal, and social implications. **Conclusions**: Our framework provides a promising potential for institutions to learn a predictive model based on healthcare or genomic data in a privacy-preserving and decentralized way.

**Key words**: blockchain distributed ledger technology, privacy-preserving predictive modeling, batch machine learning, clinical information systems, decision support systems

## INTRODUCTION

Healthcare and genomics are some of the most important types of data for cross-institution predictive modeling that estimates patient outcomes by analyzing observed data and generating scientific evidence using data from multiple institutions.[1–13] Specifically, as the volume of the shared healthcare or genomic data increases, the "learned" model (ie, the parameters identified by the learning algorithms) becomes more generalizable, thus improves the predictive correctness for each of the participating institutions. Initiatives such as ClinVar[4,5] aim at the same goal and allow researchers to perform case-based predictions.

To avoid potential risks of improper protected health information (PHI) data disclosure in direct data sharing,[6–11] several privacy-preserving algorithms, such as GLORE[12] and EXPLORER,[13] have been proposed to exchange only the model (ie, aggregated parameters) but not the observation-level patient data. As shown in Figure 1A, such methods are mostly based on a centralized architecture (ie, they require a central server as an intermediary) that includes 2

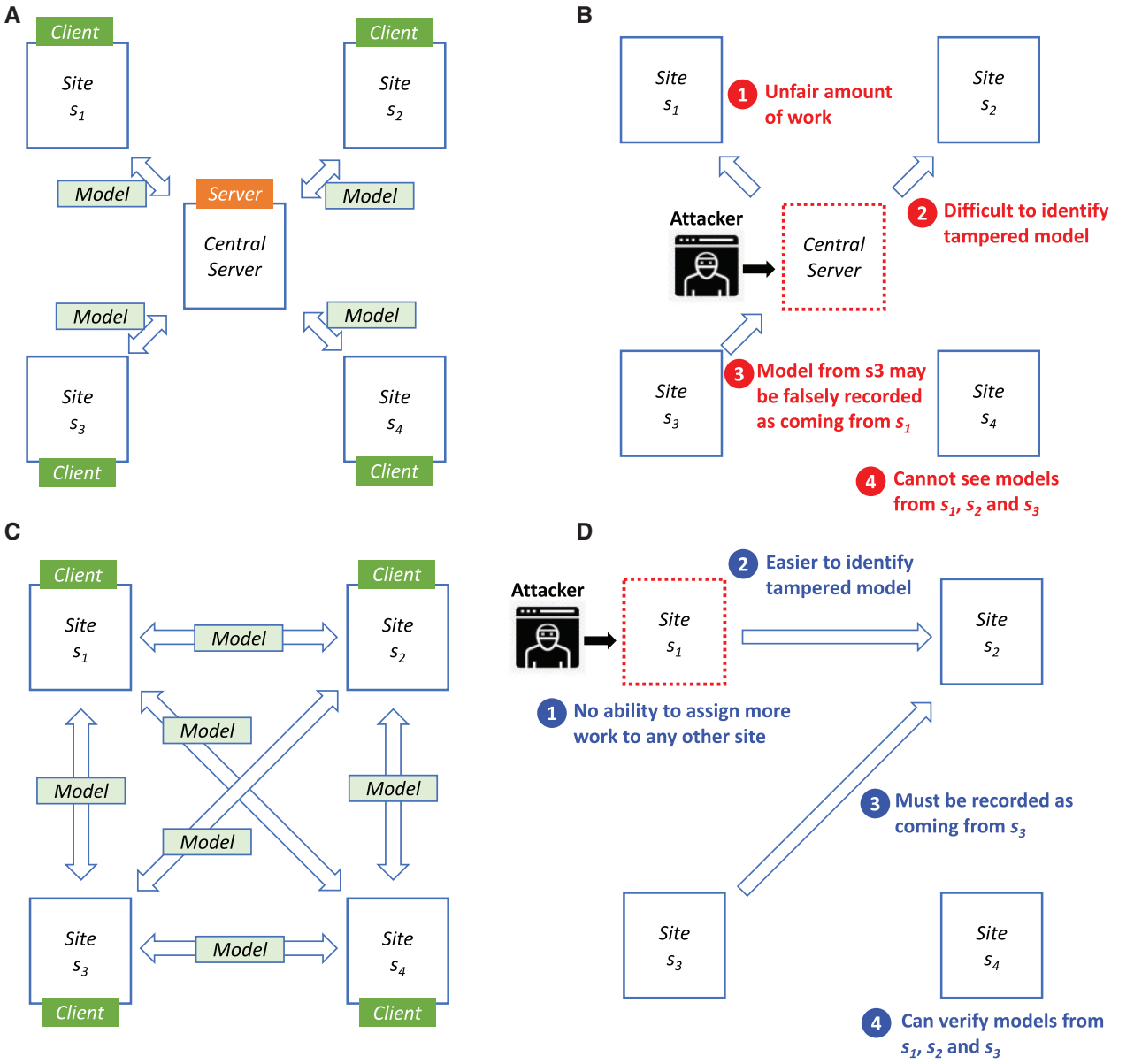


**Figure 1.** Comparison of privacy-preserving learning methods. (A) Basic idea of learning on a centralized architecture (eg, GLORE). Each site computes its local model and submits the model to the central server. The central server combines the models into a global one, and submits the models back to each site. This process continues until the global model converges. (B) Types of attack related to a centralized architecture. The central server is taken over by an attacker during the model learning process. The potential risks are as follows: (1) single point of control: because the attacker now controls the learning process, a site ($s_1$ in this example) may be assigned more computational work than the others; (2) mutable data/records: if the attacker submits a falsified or fabricated global model to a site (eg, $s_2$), no other sites can detect this misconduct, because the transparency about the models is limited; (3) change provenance: the attacker may change the ownership of the local model submitted by a site (eg, $s_3$) to another site (eg, $s_1$); and (4) partial visibility: a site ($s_4$) can only access its own local models and the global models, and therefore cannot see the models from the other sites ($s_1$, $s_2$, and $s_3$). (C) Basic idea of learning on a decentralized architecture (eg, GloreChain). There is no central server. Instead, each site exchanges its models to improve the predictive correctness on a peer-to-peer blockchain network. (D) Reducing risks by using a decentralized architecture. Site $s_1$ is taken over by an attacker. The risks described in panel B can be mitigated as follows: (1) no single point of control: although the attacker can change or even stop the computational work on site $s_1$, the loading of the other sites cannot be easily raised because the learning process is not controlled by $s_1$; the attacker cannot overload any other site as would be the case in panel B; (2) immutable data or records: if the attacker who took over $s_1$ tries to submit a falsified or fabricated model to another site (eg, $s_2$), all other sites (eg, $s_3$ and $s_4$) will receive the tampered model on the blockchain network, and thus the probability of detecting such misconduct increases; (3) data provenance: the attacker cannot claim that a model generated by $s_3$ came from $s_1$, because on a blockchain network the source of each model is recorded and is verifiable; and (4) complete visibility: a site (eg, $s_4$) can easily see all models from all sites, because on a blockchain network every site has a full copy of all models.

roles: a central server (the "server" role) and multiple sites (the "client" role). The server controls the learning process and combines the models generated by the clients. An analog is the shared computation with people's personal computers, such as the Fight AIDS @ Home project.[14] Although these approaches preserve privacy during the learning process, such a centralized architecture may be vulnerable to attacks (Figure 1B) due to its single point of control, mutable data or records, change provenance trail, and partial visibility.[15–18]

To mitigate these risks, one plausible idea is to adopt a decentralized architecture, as shown in Figure 1C. Without a central server (ie, no server role), the benefits of the decentralized solution include: no single point of control, immutable data or records, ascertained data provenance, and complete visibility (Figure 1D). Based on this idea, recent studies such as ModelChain[19] and ExplorerChain[20] proposed to combine privacy-preserving learning with the blockchain technology,[15–17,21–28] which is a distributed chain of transaction blocks and has been proposed by many researchers for various healthcare or genomic applications.[18,29–60] ModelChain/ExplorerChain execute learning algorithms on the peer-to-peer permissioned blockchain network (ie, participants are preauthorized) and utilize the metadata of the transaction to disseminate the partially trained models. In the design of ModelChain and ExplorerChain, the server role is removed to achieve decentralization (Figure 2A). Without a server role to oversee and combine the local models, ModelChain/ExplorerChain are actually approximate methods to integrate the learning algorithm with blockchain,[20] and thus have reduced correctness when compared with centralized algorithms such as EXPLORER.[12]

Owing to the importance of the predictive correctness for cross-institutional healthcare or genomic modeling, the inclusion of both server and client roles can ensure the learning algorithm performs as well as the centralized methods. On the other hand, the fairness of the compute loads sharing should also be considered. That is, every site should share the loads of acting as the server in a fair way. Such fairness is important because the server site is responsible for the combination of the partial models from all sites, and the combination process involves non-negligible computational costs and thereby high energy consumption.

### Objective

We sought to develop a general privacy-preserving predictive model sharing framework that achieves 3 goals: (1) preserves the predictive correctness, (2) mitigates the risks of a centralized architecture, and (3) computes the models in a fair way, facilitating cross-institution healthcare or genomic studies and quality improvement initiatives.

## MATERIALS AND METHODS

In our proposed framework, to achieve the first goal of preserving predictive correctness, every site serves as both a server and client while computing the model (Figure 2B). With this approach, the high level of correctness can be preserved because the learning process is exactly the same as that of centralized learning. To achieve the second goal of mitigating the risks of centralized architecture, we adopt the peer-to-peer blockchain technology, and can thus inherit the benefits of blockchain (eg, no single point of control). To achieve the third goal of computing the models in a fair way, we propose to adopt the round-robin approach in which every site alternatively serves as server for each learning iteration, as shown in Figure 3. This approach can avoid computational unfairness, such as what happens when one or few sites acts significantly more times as the server than the other sites. Note that permissioned blockchain platforms such as MultiChain[22,61] utilize the round-robin approach as a fast and low computational cost consensus protocol to mitigate the slow and high energy consumption of blockchain.

To examine our proposed blockchain-enabled fair compute loads framework, we developed GloreChain (Grid Binary LOgistic REgression on Permissioned BlockChain) as a concrete example. The 3 main components of GloreChain—batch model learning, blockchain data/network, and consensus learning algorithm—are described sequentially in the following 3 subsections, followed by the details of our implementation.

### The GLORE batch model learning

There are 2 major types of privacy-preserving predictive modeling algorithms. The first type includes online methods (eg, EXPLORER)[12] that update the model using partial data sequentially and focus on the efficiency of the retraining process (ie, when the data are updated, the model does not need to be completely retrained). The second type includes batch methods (eg, GLORE)[13] that update the model using all data at the same time and focus on learning a model that is exactly the same as the one trained by "traditional" logistic regression (ie, disseminate all data to a single server first and then perform learning).

Although our framework is general and can adopt both online and batch methods while still being decentralized, for GloreChain, we adopt the batch learning algorithm GLORE.[13] This is because we focus on obtaining a high-level of predictive correctness. GLORE applies the Newton-Raphson method,[62] and decomposes the derivatives of the log-likelihood function to estimate model coefficients.[13] We denote the client and the server parts of the learning algorithms in GLORE as GLORE-Client and GLORE-Server, respectively.

### The blockchain data and network

GloreChain utilizes the metadata of the transaction to disseminate partially trained predictive models (ie, a set of aggregated numerical coefficients or parameters) and relevant meta information. The design of GloreChain is shown in Figure 4. As shown in Figure 4A, each transaction represents an action of a model, and stores the partial model with meta information in the metadata of the transaction. Table 1 describes the details of each data field stored in the metadata of the transaction on the blockchain. Only partially trained models are disseminated on-chain, and the observation-level patient data are stored off-chain, to improve privacy protection. Also, there is no transaction amount (ie, the coin value transferred within every transaction is 0), and thus GloreChain takes advantage of only the distributed data ledger but not of the cryptocurrency aspect of blockchain.

Regarding the type of the underlying blockchain network, a permissioned one (ie, only permitted participants can join the network) is suitable for GloreChain. We only store the aggregated partial models on-chain, but a permissioned blockchain network provides an additional layer of privacy protection. It should also be noted that GloreChain provides nonfinancial incentives (ie, improved predictive power) instead of financial ones (ie, coins) for each participating site to contribute to the computation (eg, learning the models, creating the blocks, and verifying the transactions). In a complex network, however, not all sites participate in a study, so it is important that all institutions (ie, nodes in the network) know

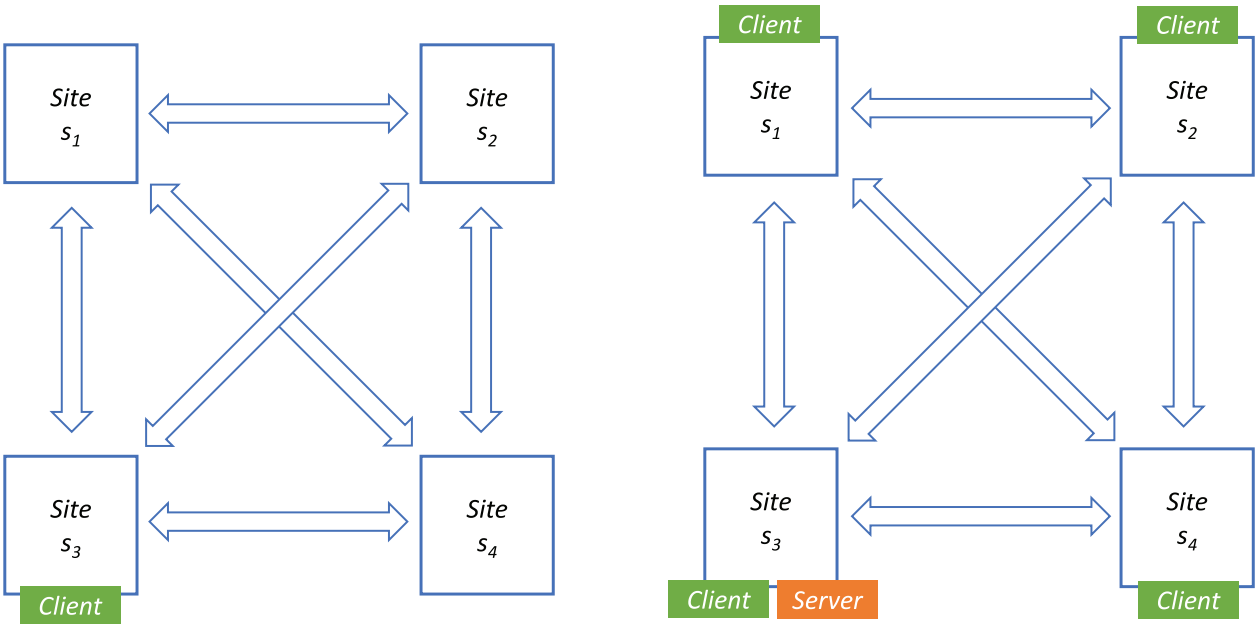


**Figure 2.** Comparison of the methods to integrate predictive modeling in a decentralized architecture. (A) Inclusion of only the "client" role to approximate model learning. One of the sites (eg, $s_3$) serves as a client at each learning iteration. (B) Inclusion of both client and "server" roles to perform exact model computation. Every site serves as a "client," and 1 of the sites (eg, $s_3$) serves as the "server," at each learning iteration.

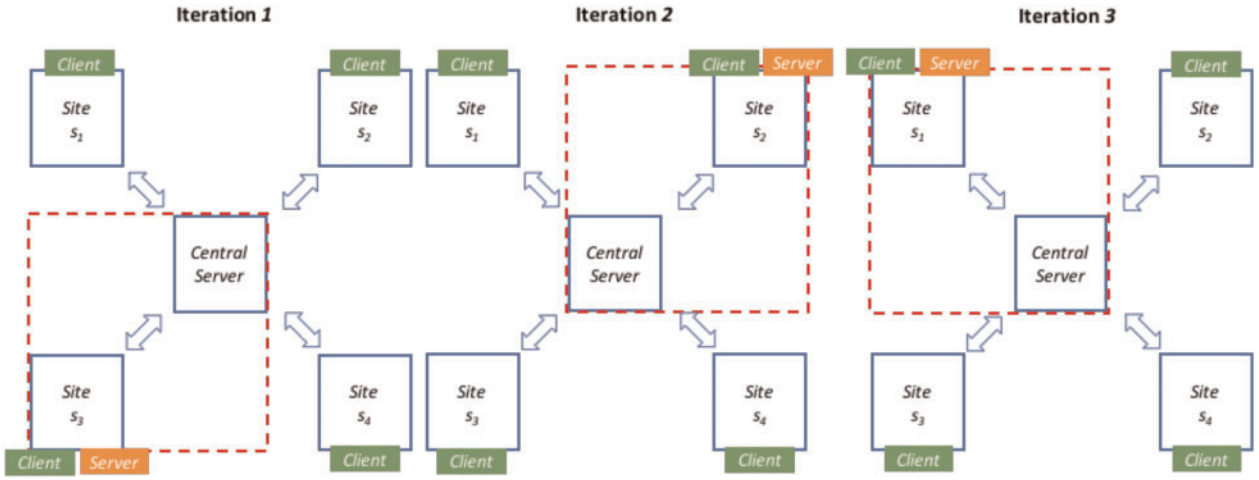


**Figure 3.** An illustration of the alternating client-server method. At each learning iteration, 1 site (eg, $s_3$, $s_2$ and $s_1$) serves as the "server" in an alternating manner.

how to build a model and are thus not dependent on a single institution to act as a "server."

## The Proof-of-Equity consensus learning algorithm

We designed a new algorithm, Proof of Equity (PoE), to determine a fair order for each site to serve as the server in a round-robin way. The PoE algorithm determines the order alphabetically using the unique name or identifier that represents a site. Each site first submits to the blockchain its unique name or identifier (eg, "San Diego Hospital" and "Davis Hospital"), and then every site retrieves the names or identifiers from blockchain to determine the order, alphabetically (eg, "Davis Hospital" → "San Diego Hospital"). Thereafter, the learning process starts, and each site follows the predetermined order to serve as the server for each iteration, in a round-robin way (eg, "Davis Hospital" → "San Diego Hospital" → "Davis Hospital" → "San Diego Hospital" →. . .). This process continues until the model converges or the maximum number of learning iterations is reached. We then regard the final predictive model as the consensus model.

A running example of the PoE algorithm is shown in Figure 5, while the details are described in Algorithm 1 (the high-level PoE algorithm in GloreChain). There are 4 hyperparameters in the algorithm: the polling time period $\Delta$, the waiting time period $\Theta$, the maximum iteration $\Omega$, and the total number of participating sites $N$. The size of the model (including mean and variance) is $O(m^2)$, where $m$ is the total number of covariates.

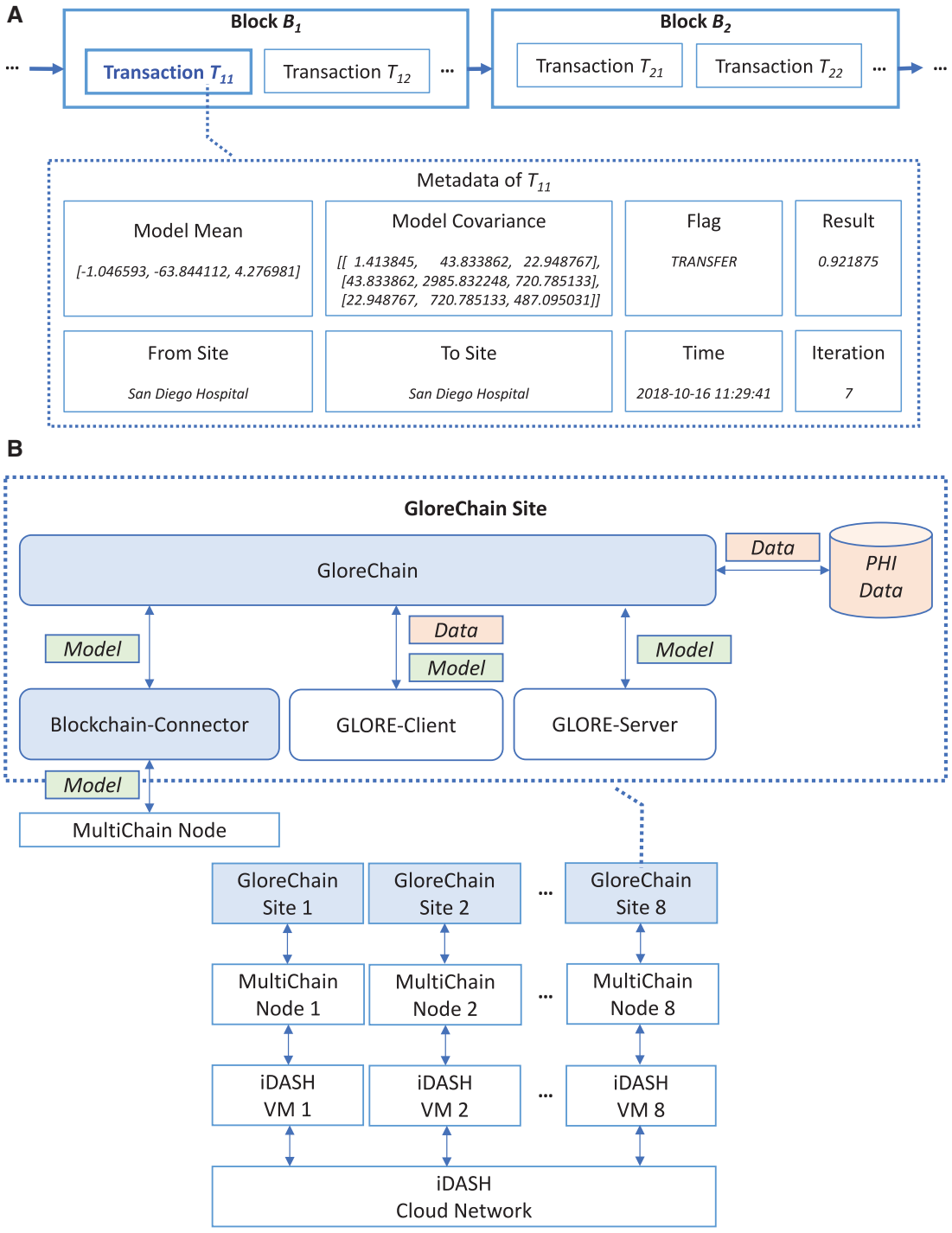


**Figure 4.** The design of GloreChain. (A) The simplified blocks and transactions. The details of the data fields stored in the metadata of the transaction are described in Table 1. (B) The implementation architecture of GloreChain, based on an 8-site configuration. The Blockchain-Connector is a software component we programmed as an interface of the GloreChain and the underlying MultiChain blockchain platform. PHI: protected health information; VM: virtual machine.

## Algorithm 1

**Input**: The local data $D$, the polling time period $\Delta$, the waiting time period $\Theta$, the maximum iteration $\Omega$, and the total number of participating sites $N$.

**Output**: The consensus batch predictive model $M$.

**Step 1.** Submit an initial transaction to the blockchain.

**Step 2.** Check the blockchain every time $\Delta$ until the initial transactions from all $N$ sites are received, and determine the learning order alphabetically using the unique name or identifier of each site.

**Step 3.** Wait for time $\Theta$ to let every site determine the learning order.

**Step 4.** Initialize the global model $G$ by setting all coefficients to zeroes.

**Table 1.** The on-chain data of GloreChain demonstrating an example of the transfer action of a model.

| Field | Description | Possible Values | Example |
|---|---|---|---|
| **Model Mean** | The mean vector of the GLORE partial model[13] | A numerical vector with its length equals $m+1$ | *[−1.046593, −63.844112, 4.276981]* |
| **Model Covariance** | The variance-covariance matrix of the GLORE partial model[13] | A numerical *(m + 1) x (m + 1)* square symmetric matrix | *[[1.413845, 43.833862, 22.948767], [43.833862, 2985.832248, 720.785133], [22.948767, 720.785133, 487.095031]]* |
| **Flag** | The type of action a site has taken to the model | *UNKNOWN*, *INITIALIZE*, *UPDATE*, *TRANSFER*, *CONSENSUS*, *TEST*, *CLEAR* | *TRANSFER* |
| **Result** | The value of the evaluation metric when the learning process is complete | A numerical value between 0 and 1 | *0.921875* |
| **From Site** | The site that has submitted the model | A unique name or identifier representing the site | *San Diego Hospital* |
| **To Site** | The site that will receive the model | A unique name or identifier representing the site | *San Diego Hospital* |
| **Time** | The time that the site submitted the model | A timestamp | *2018–10–16 11: 29: 41* |
| **Iteration** | The current iteration of the learning process | A non-negative integer | *7* |

In this example, $m = 2$ is the number of the covariates in the dataset. The partial model of GLORE contains both Model Mean (eg, the $m + 1 = 3$-dimensional vector) and Model Covariance (eg, the $(m + 1)$ x $(m + 1) = 3$ x 3 matrix), while the final model is the consensus mean vector.[13] The Flag is *TRANSFER*, representing the submission of a global model to the blockchain (in contrast to *UPDATE*, representing the submission of a local model). The Result indicates the value of the evaluation metric for correctness (eg, the full area under the receiver operating characteristic curve).[63–65] In this round, San Diego Hospital acts as the "server," and thus it is both the From Site and To Site. The Time field stands for the timestamp of the transaction, and the Iteration is the number of learning iterations

**Step 5.** Wait for time $\Theta$ to let every site initialize its global model.

**Step 6.** Loop until $G$ converges or the maximum iteration $\Omega$ is reached.

Step 6.1. Wait for time $\Theta$, update the local model $L$ using $G$ and local data $D$ through the GLORE-Client learning algorithm, and submit $L$ to the blockchain.

Step 6.2. If this site is the next server site, check the blockchain every time $\Delta$ until all $N$ local models are received, update model $G$ using all local models received through the GLORE-Server learning algorithm, and then submit $G$ to the blockchain.

Step 6.3. If this site is not the next server site, check the blockchain every time $\Delta$ until the next $G$ is found.

**Step 7.** The consensus model is $M = G$.

## The implementation of GloreChain

The architecture of GloreChain is demonstrated in Figure 4B. GloreChain, as well as the blockchain-connector and the two GLORE components (ie, GLORE-Client and GLORE-Server, as introduced in Section 3.1) were written in Java. The code of GLORE was refactored to the application programming interface that can be called by GloreChain, while the original modeling capabilities remained the same. Note that PHI was only used to compute the local model (ie, GLORE-Client) and was not disseminated to the blockchain network.

Based on a recent survey of the blockchain platforms,[66] we adopted MultiChain[22,61] in GloreChain, because (1) MultiChain is based on the popular Bitcoin Blockchain[25,67] and (2) MultiChain is developed to serve as a general-purpose permissioned blockchain network.[20] We used its default parameters and consensus protocol (ie, Mining Diversity)[22,61] in our implementation. The computation environment for GloreChain is the iDASH (integrating Data for Analysis, Anonymization, and SHaring) cloud network,[68,69] a private cloud network compliant with requirements from the Health Insurance Portability and Accountability Act. We simulated the multisite scenarios (2, 4, and 8 sites in our experiment) on iDASH virtual machines (VMs). Each VM included 2 of Intel Xeon 2.30 GHz CPUs, 8 GB of RAM, and 100GB of storage.

## Datasets

We evaluated GloreChain on 3 healthcare or genomic test datasets, and each of them had a binary outcome. First, the Edinburg (Edin) dataset[70] was used to predict whether myocardial infarction was present, based on phenotyping features. Next, the cancer biomarkers (CA) dataset[71] was used to predict whether the cancer was present based on biomarkers (ie, CA-19 and CA-125). Finally, the total hip arthroplasty (THA) dataset[20,72] was used to predict whether a patient's actual hospital length of stay was greater than the expected length of stay (3 days) for the unilateral primary THA surgery at University of California, San Diego (UCSD), based on various covariates including demographics, osteoarthritis grade, surgical approach, and comorbidities. Table 2 summarizes the statistics of the 3 datasets.

These test datasets were used to evaluate privacy-preserving predictive models in the previous studies.[12,13,20] For the THA dataset, the Institutional Review Board at UCSD approved this research with Project Number 171344X on February 9, 2018. Also, the Human Research Protections Program at UCSD exempted the informed consent requirement, because the THA dataset was defined as Health Insurance Portability and Accountability Act de-identified and contained no sensitive patient health information.

## Experiment settings

The goal of our experiments was to evaluate whether GloreChain, containing the alternating server and client roles, could in practice provide the exact same predictive power as centralized methods,

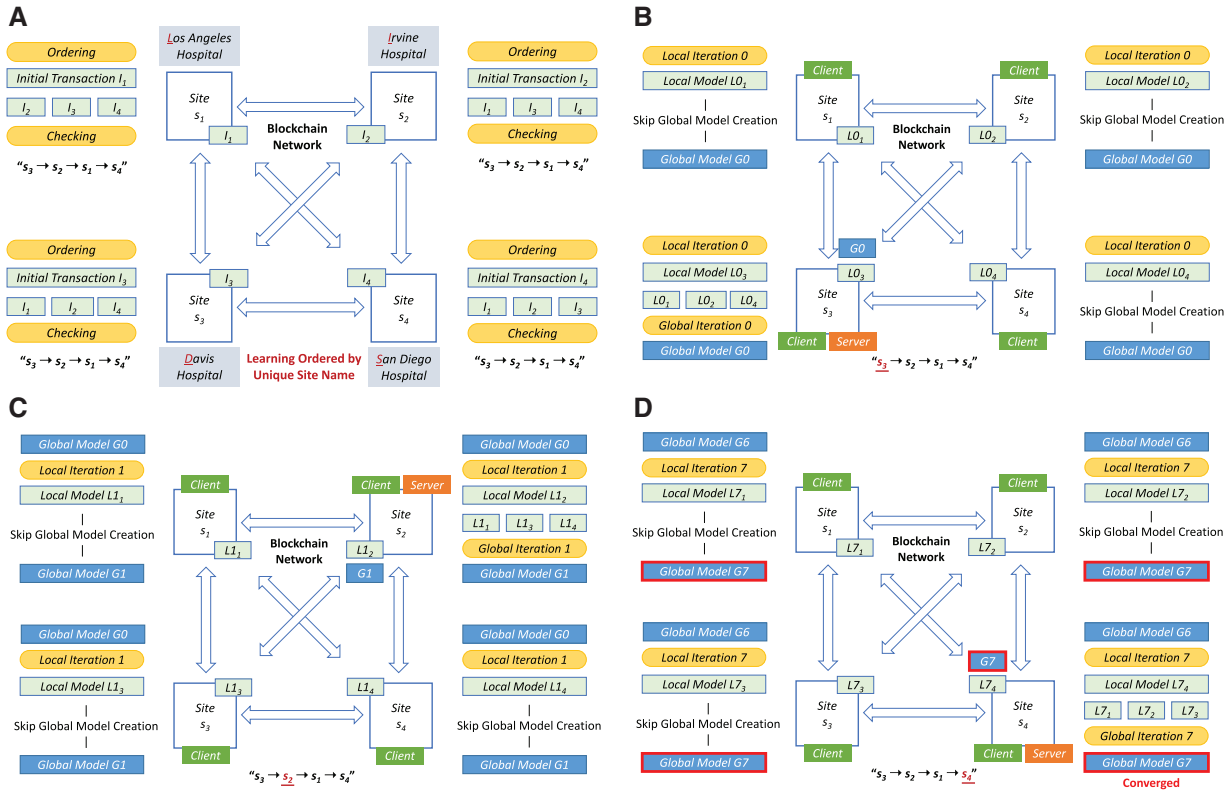


**Figure 5.** An example of the Proof-of-Equity (PoE) algorithm (ie, model consensus protocol). The notations are as follows: $I_s$ denotes the initial transaction on site $s$ for determining the order, $Li_s$ is the local model at iteration $i$ on site $s$, and $Gi$ is the global model in iteration $i$. (A) Order determination. Each site $s$ first starts the ordering process by submitting an initial transaction $I_s$, which includes the unique name/identifier of the site (Los Angeles Hospital, Irvine Hospital, Davis Hospital, and San Diego Hospital in our example), to the blockchain. Then, each site collects the initial transactions from all sites, and then starts the *checking* process by ordering the unique name or identifier in an alphabetical order (eg, Davis Hospital → Irvine Hospital → Los Angeles Hospital → San Diego Hospital, or $s_3 \rightarrow s_2 \rightarrow s_1 \rightarrow s_4$). In this round-robin way, the same order is determined on each site (eg, $s_3 \rightarrow s_2 \rightarrow s_1 \rightarrow s_4 \rightarrow s_3 \rightarrow s_2 \rightarrow s_1 \rightarrow s_4 \rightarrow \ldots$). (B) Initialization of the predictive model. Next, each site starts to compute the initial local model (in *local iteration 0*), and submits its local model (ie, $L0_s$ on site $s$) to the blockchain. Then, based on the predetermined order, the first "server" site ($s_3$) computes the initial global model (in *global iteration 0*), and submits the global model ($G0$) to the blockchain. Note that, every site serves as a "client," while only $s_3$ acts as the "server," in this initial learning process. Therefore, sites $s_1$, $s_2$, and $s_4$ do not perform global model computation to avoid duplicated work and unnecessary energy consumption. (C) The first iteration. At iteration 1, each site first computes its local model (ie, $L1_s$ on site $s$) based on the initial global model ($G0$), and submits $L1_s$ to the blockchain. Then, the next server site ($s_2$) computes the global model ($G1$) and submits $G1$ to the blockchain. Other sites ($s_1$, $s_3$, and $s_4$) skip the global model computation and serve as clients only. (D) The final or consensus iteration. The learning process repeats until a converged model is identified or the maximum number of iterations is reached. For the example shown in panel D, at iteration 7, site $s_4$ acts as the server, and after the global model computation it determines that the global model ($G7$) converged. That is, $G7$ is very close to $G6$, the global model in the previous iteration, based on the convergence criterion ($10^{-6}$ precision in our experiments). In this case, $s_4$ submits this consensus predictive model $G7$ to the blockchain, and the PoE algorithm is complete.

while obtaining the benefits of the underlying decentralized blockchain network. Therefore, we compared GloreChain with GLORE,[13] the state-of-the-art batch learning method, in our experiments. The convergence criterion was set to $10^{-6}$ precision, the same value described in Wu et al,[13] for both methods.

The hyperparameters for GloreChain were configured as the following: the polling time period $\Delta = 1$ (second), the waiting time period $\Theta = 30$ (seconds), the maximum iteration $\Omega = 20$, and the total number of participating sites $N = 2$, 4, or 8. We selected the time period hyperparameters $\Delta$ and $\Theta$ according to the learning speed and the network latency based on a previous study.[20] The maximum iteration $\Omega$ was selected based on the average iteration numbers for convergence (ie, 7 and 12 iterations for the Edin and the CA datasets, respectively[13]) Also, we checked the latest $N$ transactions with the size of the transaction metadata >20 in the PoE algorithm. Note that a waiting time was added in each iteration for GLORE for the purpose of synchronization. We chose this per-iteration waiting time in GLORE to be the same as the value of $\Theta$ (ie, 30 seconds) in GloreChain, to retain the fairness of the comparison.

We evenly and randomly split each dataset for the 2-, 4-, and 8-site scenarios. For each site, the dataset was randomly divided into 80% and 20% training and the testing records, respectively. Note that each training or testing dataset preserved a similar class distribution as described in Table 2, and contained at least 1 positive and 1 negative record. The full area under the receiver-operating characteristic curve (AUC)[63–65] on the test datasets was adopted as our evaluation measure. We used the averaged test AUC among all $N$ sites as the result for the predictive correctness.

We repeated the previously mentioned process (ie, data splitting, model learning, and averaged test AUC computation) over 30 trials. We also compared consensus iterations and the execution time of GloreChain and GLORE. To collect the results, a pausing time of 240 seconds was added between each trial for both GloreChain and GLORE, and this per-trial waiting time was deducted in the execution time computation. It should be noted that, for the $N$ site config-

**Table 2.** Statistics of the test datasets evaluated in our experiments, including the percentage of the positive or negative classes

| Dataset | Edin | CA | THA |
|---|---|---|---|
| **Description** | Myocardial infarction | Cancer | Length of hospitalization >3 days |
| **Covariates** | 9 | 2 | 34 |
| **Observations** | 1253 | 141 | 960 |
| **Class Distribution (Positive/ Negative)** | 0.219/0.781 | 0.638/0.362 | 0.278/0.722 |
| **Outcome** | Presence of disease | Presence of cancer | Hospital length of stay for total hip arthroplasty is >3 days |

The values for the myocardial infarction (Edin) and cancer biomarker (CA) datasets are reproduced from Wang et al[12] and the values for the length of hospitalization (total hip arthroplasty [THA]) dataset are reproduced from Kuo et al.[20]

**Table 3.** The predictive correctness and number of learning iterations for the myocardial infarction (Edin), CA, and length of hospitalization (THA) datasets for the 2-, 4-, and 8-site scenarios

| Dataset | n | Correctness (AUC) | | | | Number of Iterations | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | GLORE | | GloreChain | | GLORE | | GloreChain | |
| | | Mean | SD | Mean | SD | Mean | SD | Mean | SD |
| **Edin** | 2 | 0.965 | 0.013 | 0.965 | 0.013 | 6.967 | 0.183 | 6.967 | 0.183 |
| | 4 | 0.962 | 0.010 | 0.962 | 0.010 | 7.000 | 0.000 | 7.000 | 0.000 |
| | 8 | 0.959 | 0.013 | 0.959 | 0.013 | 7.433 | 2.373 | 7.433 | 2.373 |
| **CA** | 2 | 0.893 | 0.054 | 0.893 | 0.054 | 11.633 | 0.49 | 11.633 | 0.490 |
| | 4 | 0.866 | 0.071 | 0.866 | 0.071 | 11.833 | 0.379 | 11.833 | 0.379 |
| | 8 | 0.900 | 0.058 | 0.900 | 0.058 | 11.800 | 0.407 | 11.800 | 0.407 |
| **THA** | 2 | 0.736 | 0.034 | 0.736 | 0.034 | 17.500 | 5.686 | 17.500 | 5.686 |
| | 4 | 0.741 | 0.046 | 0.741 | 0.046 | 17.500 | 5.686 | 17.500 | 5.686 |
| | 8 | 0.722 | 0.040 | 0.722 | 0.040 | 17.000 | 6.103 | 17.000 | 6.103 |

The evaluation metric for correctness is the averaged full AUC among $N$ sites for 30 trials.

AUC: area under the receiver-operating characteristic curve; CA: cancer biomarkers; Edin: Edinburg; THA: total hip arthroplasty.

uration, GLORE actually requires $N+1$ VMs (ie, $N$ VMs for the clients and 1 VM for the server), while GloreChain requires exactly $N$ VMs.

## RESULTS

The comparison results for predictive correctness and number of learning iterations are shown in Table 3. For both correctness and iterations, the results of GloreChain are exactly the same as the ones from GLORE. In general, the mean AUC, the standard deviation of AUC, and the mean of the iterations remained stable in different scenarios (ie, 2, 4, and 8 sites) for the same dataset.

On the other hand, the standard deviations of an iteration in some combinations of datasets and scenarios (ie, Edin for 8 sites, and THA for 2, 4, and 8 sites) were relatively high. Therefore, we further investigated the iterations per trial for each dataset and for each scenario. As shown in Figure 6, 1 of the 30 trials for Edin and the 8-site configuration reaches the maximum iteration Ω (ie, 20). Also, for the THA dataset, most of the trials reach the maximum iteration, while some of the trials converged in 5 iterations. Although increasing the maximum iteration Ω can potentially improve the predictive power, our goal is to evaluate whether GloreChain performs exactly the same as GLORE in terms of correctness, instead of improving the predictive power for each dataset. Also, current correctness results are already comparable to ones shown in the previous studies.[12,13,20] Therefore, we kept the hyperparameter Ω as 20 in our experiments.

Table 4 illustrates the total and the per-iteration execution time results. GloreChain has higher total execution time when compared with GLORE because of the additional time requirement of the PoE algorithm to determine the learning order of the institutions at the beginning of the modeling process, as described in Section 3.3. Also, GloreChain had higher running time (about 3–8 times slower than GLORE in terms of the per-iteration running time), because GloreChain adopts a blockchain network and therefore incurs additional time to create and synchronize the transactions and blocks.

GLORE used 3, 5, and 9 VMs for the 2-, 4-, and 8-site scenarios, respectively, while GloreChain used exactly 2, 4 and 8 VMs for the corresponding scenarios. Therefore, the actual difference of the aggregated running time on all VMs is smaller. For example, the per-iteration running time for the Edin dataset and the 2-site setting of GLORE and GloreChain are 0.024 and 0.092, respectively (GloreChain takes 4 times longer than GLORE). However, considering the aggregated running time by multiplying the number of VMs (3 for GLORE and 2 for GloreChain) in the same dataset and scenario, GloreChain takes only 2.5 times longer than GLORE.

## DISCUSSION

Based on the results, GloreChain, with its alternating server and client roles, had exactly the same predictive power as well as number of learning iterations when compared with GLORE. Additionally, GloreChain possessed the advantages of adopting the blockchain technology, such as no single point of control. The cost of inheriting these additional benefits from blockchain technology was a slight increase in execution time (including both running and synchronization time) when compared with an equivalent approach that did not use blockchain. This slight increase was minimal when compared with what would have been the energy consumption if we had decided to utilize the typical Proof-of-Work protocol[25,67] that made Bitcoin blockchain famous worldwide for promoting the use of computer "farms" by block miners.[73,74] We used instead a low energy-consumption protocol (ie, Mining Diversity) that was sufficiently secure to run GloreChain on a permissioned blockchain network (eg, MultiChain).[22,61] In fact, we used fewer machines to run GloreChain than we would have used had we followed other approaches (eg, GLORE) because the central server and its backups were not needed due to the dis-intermediation and redundancy features provided by blockchain technology.

Although we adopted GLORE in GloreChain, the core privacy-preserving learning can be replaced by any other centralized algorithm. That is, our framework, containing both server and client roles, provides a general and flexible framework that supports any client-server–based algorithms. In related methods, such as those used in ModelChain/ExplorerChain, the server is replaced by a consensus algorithm to determine the learning order using model errors, stored on-chain as a field in the metadata of the transaction.[19,20] Therefore, ModelChain/ExplorerChain can only integrate online

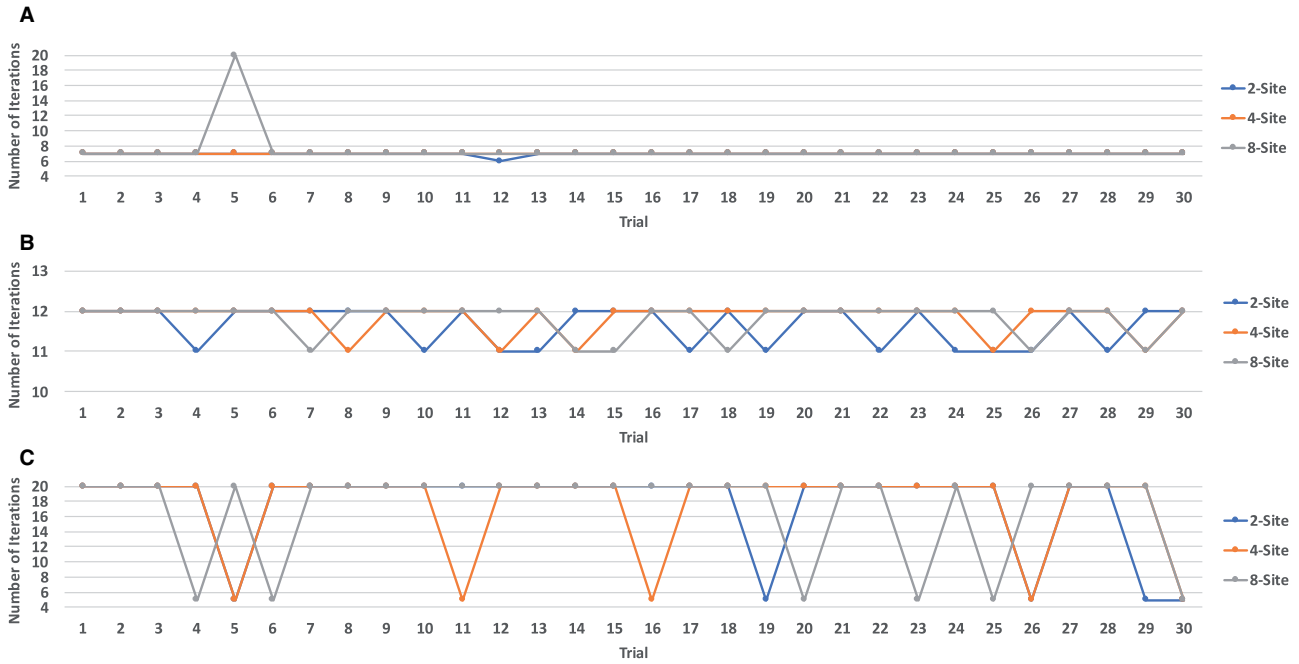


**Figure 6.** The pretrial iterations on the 3 scenarios (2, 4, and 8 sites) for each of the 30 trials, with the maximum iteration Ω set to 20. (A) Edinburg (Edin) dataset, (B) cancer biomarkers (CA) dataset, (C) total hip arthroplasty (THA) dataset.

**Table 4.** The execution time (in seconds) for *N* sites in 30 trials.

| Dataset | n | Overall | | | | Per Iteration | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | GLORE | | GloreChain | | GLORE | | GloreChain | |
| | | Total | Running | Total | Running | Total | Running | Total | Running |
| **Edin** | 2 | 239.196 | 0.168 | 342.757 | 0.638 | 34.334 | 0.024 | 49.200 | 0.092 |
| | 4 | 240.158 | 0.219 | 350.964 | 0.644 | 34.308 | 0.031 | 50.138 | 0.092 |
| | 8 | 253.139 | 0.228 | 367.316 | 0.760 | 34.055 | 0.031 | 49.415 | 0.102 |
| **CA** | 2 | 379.094 | 0.178 | 488.326 | 0.717 | 32.587 | 0.015 | 41.976 | 0.062 |
| | 4 | 385.145 | 0.182 | 505.684 | 0.774 | 32.547 | 0.015 | 42.734 | 0.065 |
| | 8 | 384.121 | 0.192 | 509.440 | 0.856 | 32.553 | 0.016 | 43.173 | 0.073 |
| **THA** | 2 | 555.322 | 0.275 | 687.953 | 1.701 | 31.733 | 0.016 | 39.312 | 0.097 |
| | 4 | 555.301 | 0.302 | 702.545 | 1.908 | 31.731 | 0.017 | 40.145 | 0.109 |
| | 8 | 540.313 | 0.357 | 691.771 | 2.883 | 31.783 | 0.021 | 40.692 | 0.170 |

All measurements are in seconds and are averaged over *N* sites. The total time (ie, the "real" time in the Linux system) includes both running time (ie, the "user + sys" time in the Linux system) and synchronization time (ie, total minus running time). We divided the total time by the mean of iteration (as shown in Table 3) to compute the per-iteration time. The additional per-trial pausing time (240 seconds) was deducted in the computation.

CA: cancer biomarkers; Edin: Edinburg; THA: total hip arthroplasty.

learning algorithms with blockchain. GloreChain does not require model errors to determine the order during the learning process, and instead includes a field "Result" to store the value of AUC for computing the overall predictive correctness results (Figure 4A and Table 1).

The 3 intrinsic challenges of the blockchain networks[18–20]: transparency or confidentiality, speed or scalability, and the threat of a 51% attack (ie, the blockchain network is taken over by the majority of malicious nodes), are not critical for GloreChain because: (1) GloreChain only disseminates partial models and not PHI on the blockchain, and thus minimizes the risk of confidentiality breaches; (2) the execution time of GloreChain is large (40–50 seconds per iteration) when compared with the transaction time of the underlying blockchain platform (eg, 1000 transactions per second at a maximum for MultiChain),[75] and the difference may be much larger for the use cases on big data; and (3) GloreChain is based on a permissioned blockchain network and the participating nodes are preauthorized, therefore mitigating the risk of a 51% attack.

In the PoE algorithm, we used the alphabetical order of the unique site name or identifier to determine the next server site, and more sophisticated methods may be applied so that the institutions at the top of the order do not work more than others. For example, a different order may be assigned every *N* iterations (eg, "$s_3 \rightarrow s_2 \rightarrow s_1 \rightarrow s_4$," followed by "$s_4 \rightarrow s_1 \rightarrow s_2 \rightarrow s_3$"). Although our method is relatively simple, in the long run, the computational cost and energy consumption for each site can still be as "fair" as more

complicated methods. Note that our framework is based on a "semitrust" assumption, where every site would like to improve the predictive correctness by sharing the aggregated partial model, but might be "curious" about other sites. Moreover, we assume syntactic and semantic interoperability for each site (ie, they adopted the same data format, meaning, and standards).

Regarding the hyperparameters, the polling time period $\Delta$ decides the balance of system responsiveness and network burden, the waiting time period $\Theta$ determines execution speed while considering potential network latency, and the maximum iteration $\Omega$ represents the trade-off between predictive correctness and learning efficiency. These hyperparameters can be adjusted based on the actual use case scenarios. Also, the size of the metadata of the transaction is about 5KB for THA (ie, the largest dataset in our experiment), and is way below the default size limit of MultiChain (ie, 2 MB).

It should also be noted that, although we implemented GloreChain based on MultiChain, the GloreChain framework itself is platform independent, and can adopt other blockchain platforms such as Ethereum[76] or Hyperledger[77] by changing the BlockchainConnector (as shown in Figure 4B). Finally, the deployment on the iDASH private cloud compute environment also improved the security level of the permissioned blockchain network.

### Limitations

The limitations for this study are as follows: (1) our framework was not evaluated on very high-dimensional datasets, which can create large predictive models and therefore have impacts on the size of the metadata of the transactions, the speed to disseminate partial models over the blockchain network, and the additional process to adjust the hyperparameters; (2) several use case situations were not tested, such as nonrepresentative samples, very different data distribution among all sites, and low-quality models due to poor data; (3) privacy-preserving concerns, such as those often raised by differential privacy advocates (ie, the potential privacy breach because of the very small data size in some sites),[78] were not fully studied; and (4) more investigations about the potential ethical, legal, and social implications arising from decentralized computer system access are yet to be conducted.

## CONCLUSION

By including both server and client roles in each learning iteration, adopting blockchain as the underlying peer-to-peer network, and alternating the roles for each site in a round-robin way, our proposed framework preserves the prediction correctness, obtains the benefits of decentralization, and ensures the computational fairness for predictive model building. GloreChain, an example of our framework, demonstrates the capability to reach exactly the same predictive power and number of learning iterations as the state-of-the-art centralized method. Considering the critical and sensitive nature of healthcare or genomic data, the exchange of additional execution time for benefits such as no single point of control may be considered attractive by some institutions. Although more evaluations and refinements are warranted, our framework provides a promising potential for multiple institutions to collaboratively learn a healthcare or genomic predictive model in a privacy-preserving and decentralized way, using blockchain platform that are maintained by a large community of software developers worldwide, as opposed to custom software.

## FUNDING

The study was supported by National Institutes of Health grants K99HG009680 and R00HG009680 (T-TK) and R01GM118609 (LO-M); and the National Human Genome Research Institute of the National Institutes of Health and R00HG009680. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

## AUTHOR CONTRIBUTORS

T-TK contributed in conceptualization, data curation, formal analysis, funding acquisition, investigation, methodology, project administration, resources, software, validation, visualization, and writing (original draft). RAG contributed in data curation and writing (review & editing). LO-M contributed in conceptualization, funding acquisition, project administration, resources, supervision, and writing (review and editing).

## ACKNOWLEDGEMENTS

The authors would like to thank Xiaoqian Jiang, PhD, for very helpful discussions, as well as the UCSD Health Department of Biomedical Informatics Development Team for the technical support of the iDASH cloud infrastructure.

*Conflict of interest statement.* None declared.

## REFERENCES

1. Navathe AS, Conway PH. Optimizing health information technology's role in enabling comparative effectiveness research. *Am J Manag Care* 2010; 16 (12 Suppl HIT): SP44–7.
2. Wicks P, Vaughan TE, Massagli MP, Heywood J. Accelerated clinical discovery using self-reported patient data collected online and a patient-matching algorithm. *Nat Biotechnol* 2011; 29 (5): 411–4.
3. Grossman JM, Kushner KL, November EA, Lthpolicy PC. *Creating Sustainable Local Health Information Exchanges: Can Barriers to Stakeholder Participation Be Overcome?* Washington, DC: Center for Studying Health System Change; 2008.
4. ClinVar. https://www.ncbi.nlm.nih.gov/clinvar/. Accessed June 1, 2017.
5. Landrum MJ, Lee JM, Benson M, *et al*. ClinVar: public archive of interpretations of clinically relevant variants. *Nucleic Acids Res* 2016; 44 (D1): D862–68.
6. Sweeney L. *Uniqueness of Simple Demographics in the U.S.* Population: Technical Report. Pittsburgh, PA: Carnegie, Mellon University; 2000.
7. Golle P. Revisiting the uniqueness of simple demographics in the US population. In: *Proceedings of the 5th ACM Workshop on Privacy in Electronic Society*. New York: ACM; 2006: 77–80.
8. El Emam K, Hu J, Mercer J, *et al*. A secure protocol for protecting the identity of providers when disclosing data for disease surveillance. *J Am Med Inform Assoc* 2011; 18 (3): 212–7.
9. Loukides G, Denny JC, Malin B. The disclosure of diagnosis codes can breach research participants' privacy. *J Am Med Inform Assoc* 2010; 17 (3): 322–7.
10. Vaszar LT, Cho MK, Raffin TA. Privacy issues in personalized medicine. *Pharmacogenomics* 2003; 4 (2): 107–12.
11. Calloway SD, Venegas LM. The new HIPAA law on privacy and confidentiality. *Nurs Adm Q* 2002; 26 (4): 40–54.
12. Wang S, Jiang X, Wu Y, Cui L, Cheng S, Ohno-Machado L. Expectation propagation logistic regression (explorer): distributed privacy-preserving online model learning. *J Biomed Inform* 2013; 46 (3): 480–96.
13. Wu Y, Jiang X, Kim J, Ohno-Machado L. Grid Binary LOgistic REgression (GLORE): building shared models without sharing data. *J Am Med Inform Assoc* 2012; 19 (5): 758–64.

14. ScrippsResearchInstituteOlsonLaboratory. Fight AIDS @ Home. http://fightaidsathome.scripps.edu/. Accessed August 31, 2018.
15. McConaghy T, Marques R, Müller A, *et al*. BigchainDB: a scalable blockchain database. 2016. https://www.bigchaindb.com/whitepaper/. Accessed July 30, 2016.
16. Pilkington M. Blockchain technology: principles and applications. In: Olleros FX, Zhegu M, eds. *Research Handbook on Digital Transformations*. Cheltenham, UK: Edward Elgar; SSRN; 2016. https://ssrn.com/abstract=2662660. Accessed July 30, 2016.
17. Xu X, Pautasso C, Zhu L, *et al*. The blockchain as a software connector. In: *13th Working IEEE/IFIP Conference on Software Architecture (WICSA), 2016*. Piscataway, NJ: Institute of Electrical and Electronics Engineers; 2016: 182–91.
18. Kuo T-T, Kim H-E, Ohno-Machado L. Blockchain distributed ledger technologies for biomedical and health care applications. *J Am Med Inform Assoc* 2017; 24 (6): 1211–20.
19. Kuo T-T, Hsu C-N, Ohno-Machado L. ModelChain: Decentralized Privacy-Preserving Healthcare Predictive Modeling Framework on Private Blockchain Networks. ONC/NIST Use of Blockchain for Healthcare and Research Workshop. Gaithersburg, MD: ONC/NIST; 2016.
20. Kuo T-T, Gabriel RA, Ohno-Machado L. EXpectation Propagation LOgistic REgRession on Permissioned BlockCHAIN (ExplorerChain): decentralized privacy-preserving online healthcare/genomics predictive model learning. 2018. https://doi.org/10.5281/zenodo.1492820. Accessed December 1, 2018.
21. Luu L, Narayanan V, Baweja K, Zheng C, Gilbert S, Saxena P. SCP: a computationally-scalable Byzantine consensus protocol for blockchains. Cryptology ePrint Archive. Report 2015/1168; 2015.
22. Greenspan G. MultiChain private blockchain—white paper. 2015. http://www.multichain.com/download/MultiChain-White-Paper.pdf. Accessed July 30, 2016.
23. Bissias G, Ozisik AP, Levine BN, Liberatore M. Sybil-resistant mixing for bitcoin. In: Proceedings of the 13th Workshop on Privacy in the Electronic Society. New York: Association for Computing Machinery; 2014: 149–58.
24. McConaghy T. *Blockchain, Throughput, and Big Data. Bitcoin Startups Berlin*, 2014. http://trent.st/content/2014-10-28%20mcconaghy%20-%20blockchain%20big%20data.pdf. Accessed July 30, 2016.
25. Nakamoto S. Bitcoin: A peer-to-peer electronic cash system. 2008. https://bitcoin.org/bitcoin.pdf. Accessed July 30, 2016.
26. Miller A, LaViola JJ Jr. Anonymous byzantine consensus from moderately hard puzzles: a model for bitcoin. 2014. https://nakamotoinstitute.org/static/docs/anonymous-byzantine-consensus.pdf. Accessed: July 30, 2016.
27. Meiklejohn S, Pomarole M, Jordan G, *et al*. A fistful of bitcoins: characterizing payments among men with no names. In: *2013 Internet Measurement Conference*. New York: Association for Computing Machinery; 2013: 127–40.
28. Garay J, Kiayias A, Leonardos N. The bitcoin backbone protocol: analysis and applications. In: *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Sofia, Bulgaria: Springer; 2015: 281–310.
29. McKernan KJ. The chloroplast genome hidden in plain sight, open access publishing and anti-fragile distributed data sources. *Mitochondrial DNA A DNA Mapp Seq Anal* 2016; 27 (6): 4518–9.
30. Shrier AA, Chang A, Diakun-Thibault N, *et al. Blockchain and health IT: algorithms, privacy, and data. ONC/NIST Use of Blockchain for Healthcare and Research Workshop*. Gaithersburg, MD: ONC/NIST; 2016.
31. Kuo T-T, Ohno-Machado L. Healthcare/genomics Blockchain: applications in privacy-preserving predictive model development. *Paper presented at: NHGRI Research Training and Career Development Annual Meeting*; March 18–20, 2018; Los Angeles, CA. Data Analysis and Coordinating Center for the NHGRI-supported Training Consortium, St. Louis, Missouri.
32. Kuo T-T, Ohno-Machado L. A tutorial about biomedical and healthcare blockchain. Paper presented at: AMIA Annual Symposium; November 3, 2018; San Francisco, CA. American Medical Informatics Association, Bethesda, MD.
33. Mettler M. Blockchain technology in healthcare: the revolution starts here. In: *2016 IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom)*. Munich, Germany: IEEE; 2016: 1–3.
34. Angraal S, Krumholz HM, Schulz WL. Blockchain technology: applications in health care. *Circ Cardiovasc Qual Outcomes* 2017; 10 (9): e003800.
35. Ribitzky R, Clair JS, Houlding DI, *et al*. Pragmatic, interdisciplinary perspectives on blockchain and distributed ledger technology: paving the future for healthcare. *Blockchain Healthc Today* 2018;1.
36. Clauson KA, Breeden EA, Davidson C, Mackey TK. Leveraging blockchain technology to enhance supply chain management in healthcare. *Blockchain Healthc Today* 2018;1.
37. Mamoshina P, Ojomoko L, Yanovich Y, *et al*. Converging blockchain and next-generation artificial intelligence technologies to decentralize and accelerate biomedical research and healthcare. *Oncotarget* 2018; 9 (5): 5665.
38. Juneja A, Marefat M. Leveraging blockchain for retraining deep learning architecture in patient-specific arrhythmia classification. In: *2018 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*. Piscataway, NJ: Institute of Electrical and Electronics Engineers; 2018: 393–7.
39. Dubovitskaya A, Xu Z, Ryu S, Schumacher M, Wang F. Secure and trustable electronic medical records sharing using blockchain. *AMIA Annu Symp Proc* 2017; 2017: 650–9. American Medical Informatics Association, Bethesda, MD.
40. Choudhury O, Sarker H, Rudolph N, et al. Enforcing Human Subject Regulations using Blockchain and Smart Contracts. *Blockchain Healthc Today* 2018; 1.
41. Patel V. A framework for secure and decentralized sharing of medical imaging data via blockchain consensus. *Health Informatics J* 2018; 1460458218769699.
42. Ernst & Young LLP. Blockchain in health. How distributed ledgers can improve provider data management and support interoperability 2016. https://www.hyperledger.org/wp-content/uploads/2016/10/ey-blockchain-in-health.pdf. Accessed November 16, 2018.
43. Ivan D. Moving toward a blockchain-based method for the secure storage of patient records. *ONC/NIST Use of Blockchain for Healthcare and Research Workshop*. Gaithersburg, MD: ONC/NIST; 2016.
44. Yue X, Wang H, Jin D, Li M, Jiang W. Healthcare data gateways: found healthcare intelligence on blockchain with novel privacy risk control. *J Med Syst* 2016; 40 (10): 218.
45. Culver K. Blockchain technologies: a whitepaper discussing how the claims process can be improved. *ONC/NIST Use of Blockchain for Healthcare and Research Workshop*. Gaithersburg, MD: ONC/NIST; 2016.
46. Attili S, Ladwa SK, Sharma U, Trenkle AF. Blockchain: the chain of trust and its potential to transform healthcare—our point of view. *ONC/NIST Use of Blockchain for Healthcare and Research Workshop*. Gaithersburg, MD: ONC/NIST; 2016.
47. Vian K, Voto A, Haynes-Sanstead K. A blockchain profile for medicaid applicants and recipients. *ONC/NIST Use of Blockchain for Healthcare and Research Workshop*. Gaithersburg, MD: ONC/NIST; 2016.
48. Healthbank.coop. HealthBank. https://www.healthbank.coop/. Accessed December 20, 2016
49. Linn LA, Koo MB. Blockchain for health data and its potential use in health IT and health care related research. *ONC/NIST Use of Blockchain for Healthcare and Research Workshop*. Gaithersburg, MD: ONC/NIST; 2016.
50. Goldwater J. The use of a blockchain to foster the development of patient-reported outcome measures. *ONC/NIST Use of Blockchain for Healthcare and Research Workshop*. Gaithersburg, MD: ONC/NIST; 2016.
51. Topol EJ. Money back guarantees for non-reproducible results? *BMJ* 2016; 353: i2270.
52. Brodersen C, Kalis B, Leong C, *et al. Blockchain: Securing a New Health Interoperability Experience*. United States: Accenture LLP; 2016.

53. Ekblaw A, Azaria A, Halamka JD, Lippman A. A case study for blockchain in healthcare: "MedRec" prototype for electronic health records and medical research data. *ONC/NIST Use of Blockchain for Healthcare and Research Workshop*. Gaithersburg, MD: ONC/NIST; 2016.
54. Azaria A, Ekblaw A, Vieira T, Lippman A. MedRec: using blockchain for medical data access and permission management. In: *International Conference on Open and Big Data (OBD)*. Vienna, Austria: IEEE; 2016: 25–30.
55. Griggs KN, Ossipova O, Kohlios CP, Baccarini AN, Howson EA, Hayajneh T. Healthcare blockchain system using smart contracts for secure automated remote patient monitoring. *J Med Syst* 2018; 42 (7): 130.
56. Dagher GG, Mohler J, Milojkovic M, Marella PB. Ancile: privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology. *Sustain Cities Soc* 2018; 39: 283–97.
57. Kaur H, Alam MA, Jameel R, Mourya AK, Chang V. A proposed solution and future direction for blockchain-based heterogeneous medicare data in cloud environment. *J Med Syst* 2018; 42 (8): 156.
58. Nebula genomics. https://www.nebula.org/. Accessed August 29, 2018.
59. Nebula genomics white paper. https://nebulas.io/docs/NebulasTechnicalWhitepaper.pdf. Accessed August 29, 2018.
60. LunaDNA. https://www.lunadna.com/. Accessed August 29, 2018.
61. MultiChain. MultiChain open platform for blockchain applications. http://www.multichain.com/. Accessed December 16, 2016.
62. Minka TP. A comparison of numerical optimizers for logistic regression. https://tminka.github.io/papers/logreg/minka-logreg.pdf. Accessed August 10, 2018.
63. Lasko TA, Bhagwat JG, Zou KH, Ohno-Machado L. The use of receiver operating characteristic curves in biomedical informatics. *J Biomed Inform* 2005; 38 (5): 404–15.
64. Hanley JA, McNeil BJ. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology* 1982; 143 (1): 29–36.
65. Davis J, Goadrich M. The relationship between precision-recall and ROC curves. In: *Proceedings of 23rd International Conference on Machine Learning (ICML)*. New York: ACM; 2006: 233–40.
66. Kuo T-T, Zavaleta Rojas H, Ohno-Machado L. Comparison of blockchain platforms: a systematic review and healthcare examples. *J Am Med Inform Assoc* 2019; 26 (5): 462–78.
67. Bitcoin. https://bitcoin.org/en/. Accessed February 11, 2019.
68. Ohno-Machado L, Bafna V, Boxwala A, *et al*. iDASH. Integrating data for analysis, anonymization, and sharing. *J Am Med Inform Assoc* 2012; 19 (2): 196–201.
69. Ohno-Machado L. To share or not to share: that is not the question. *Sci Transl Med* 2012; 4 (165): 165cm15.
70. Kennedy RL, Burton AM, Fraser HS, McStay LN, Harrison RF. Early diagnosis of acute myocardial infarction using clinical and electrocardiographic data at presentation: derivation and evaluation of logistic regression models. *Eur Heart J* 1996; 17 (8): 1181–91.
71. Zou KH, Liu A, Bandos AI, Ohno-Machado L, Rockette HE. *Statistical Evaluation of Diagnostic Performance: Topics in ROC Analysis*. Boca Raton, FL: CRC Press; 2011.
72. Sharma BS, Swisher MW, Doan CN, Khatibi B, Gabriel RA. Predicting patients requiring discharge to post-acute care facilities following primary total hip replacement: does anesthesia type play a role? *J Clin Anesth* 2018; 51: 32–6.
73. Dinges C. *Forecast of bitcoin–can it become a major currency or is it just another bubble?* SSRN. 2018. http://dx.doi.org/10.2139/ssrn.3110445. Accessed November 16, 2018.
74. Sompolinsky Y, Zohar A. Bitcoin's underlying incentives. *Commun ACM* 2018; 61 (3): 46–53.
75. MultiChain 1.0 beta 2 and 2.0 roadmap. https://www.multichain.com/blog/2017/06/multichain-1-beta-2-roadmap/. Accessed July 21, 2017.
76. Buterin V. A next-generation smart contract and decentralized application platform. 2014. https://github.com/ethereum/wiki/wiki/White-Paper. Accessed July 31, 2016.
77. Hyperledger architecture, volume 1: introduction to hyperledger business blockchain design philosophy and consensus. Vol 1. https://www.hyperledger.org/wp-content/uploads/2017/08/HyperLedger_Arch_WG_Paper_1_Consensus.pdf. Accessed November 5, 2017.
78. Dwork C. Differential privacy. In: *ICALP 2006: Automata, Lanugages, and Processing*. New York: Springer; 2006: 1–12.