Efficacy of federated learning on genomic data: a study on the UK Biobank and the 1000 Genomes Project

Dmitry Kolobkov[1, 2, *, †], Satyarth Mishra Sharma[1, 3, *], Aleksandr Medvedev[1, 3], Mikhail Lebedev[1], Egor Kosaretskiy[1], and Ruslan Vakhitov[1]

[1]GENXT, Hinxton, UK
[2]Vavilov Institute of General Genetics
[3]Skolkovo Institute of Science and Technology
[†]Corresponding author, dmitry@genxt.network
[*]Equal contribution

January 24, 2023

Abstract

Combining training data from multiple sources increases sample size and reduces confounding, leading to more accurate and less biased machine learning models. In healthcare, however, direct pooling of data is often not allowed by data custodians who are accountable for minimizing the exposure of sensitive information. Federated learning offers a promising solution to this problem by training a model in a decentralized manner thus reducing the risks of data leakage. Although there is increasing utilization of federated learning on clinical data, its efficacy on genomic data has not been extensively studied. This study aims to contribute to the adoption of federated learning for genomic data by investigating its applicability in two scenarios: phenotype prediction on the UK Biobank data and ancestry prediction on the 1000 Genomes Project data. We show that federated models trained on data split into independent nodes achieve performance close to centralized models, even in the presence of significant inter-node heterogeneity. This paper describes the experiments and provides recommendations on strategies that should be used to reduce computational complexity or communication costs.

1 Introduction

1.1 Availability of genomic data

The last decade has seen a rapid increase in the amount of genomic data due to the improvement of sequencing technologies and the promise of big data studies in healthcare. With genotyping costs going down, the pool of genomic data also becomes less centralized as more organizations, both commercial, such as genetic testing companies, and non-profit, such as biobanks, accumulate vast collections of genomes. This decentralization, coupled with data-hungry genome-wide machine learning approaches, raises a need for collaboration between datasets. However, access to genomic data is usually restricted due to its sensitive nature and the harmful consequences of possible data leakage, such as deanonymization and genetic discrimination [1] [2] [3].

Data holders may share aggregated data such as summary statistics for genome-wide association studies (GWAS) to allow easy access with a reduced risk of exposing sensitive information. The summary statistics can be analyzed jointly via meta-analysis [4] [5]. However, summarizing involves a loss of information which affects model performance.

Non-profit holders of genomic data, such as biobanks [6] [7], typically allow researchers to access their data upon application approval. While in the case of a successful application, researchers get access to individual-level data, the approval process may take months and joint analysis with other data sources may be prohibited.

1.2 Phenotype prediction

Phenotype-from-genotype prediction aims to score an individual's genetic liability to a certain phenotype, usually, a disease, which can identify risk groups and assist diagnostics [8]. To build a predictive model, one has to obtain either individual-level data where each sample has two alleles for each included genetic variant (SNP) or summary-level data where each SNP has an allele frequency.

Models trained on individual-level data typically yield higher predictive performance as they learn the joint SNP distribution. However, this sensitive data can typically be accessed only with an approved research application.

On the other hand, summary-based models, or polygenic scores, are trained on publicly available GWAS-derived summary statistics and can even incorporate outputs of multiple GWAS using meta-analysis. However, polygenic score models are typically based on assumptions that reduce their applicability to samples with ancestry different from the ancestry of the training set. For instance, multiple studies show poor portability of polygenic scores to other ancestry groups [9] [10] [11]. This is caused by inter-population differences in allele frequency [12] and variant effect size [13], as well as different linkage disequilibrium patterns [14].

1.3 Ancestry prediction

Genetic ancestry prediction from SNPs has two common uses. First, it is a product that genetic testing companies, also called consumer genomics companies, sell to their customers [15]. Second, due to poor polygenic score cross-ancestry portability, per-ancestry summary statistics [16] and polygenic score [17] catalogs have been established. As self-reported ancestry is often noisy, ancestry prediction is a promising tool to be used in phenotype prediction and pharmacogenomics [18].

The task of predicting ancestry is closely related to inferring genetic population structure. As ancestry differences comprise the major part of human genetic variation, population structure is well described by top eigenvectors of the covariance matrix obtained from genetic variants. As a consequence, these principal components are commonly included in phenotype prediction models as covariates to control for population structure [19]. A common approach to estimate the ancestry of unlabeled samples is to project it onto the principal component space of a labeled reference panel [20], such as the 1000 Genomes Project [21].

1.4 Federated learning

Federated learning involves training a model locally on clients (data nodes) and sending parameter updates to a server (central hub) where these updates are aggregated into a new set of model parameters which are then sent back to the clients in the next round of training, also called a communication round [22]. Unlike conventional 'centralized' machine learning, federated learning does not require assembling data at a single location which saves communication costs and, more importantly, enhances data security since the client's data is not disclosed to the server nor to other clients. The server, however, should be trustworthy as the model updates the server has access to may leak information about the data [23]. A comprehensive review of the security aspects of federated learning is provided at [24]. A more horizontal approach in which clients send models to each other without a need for a server is called swarm learning [25].

Federated learning alone may not be enough to build a secure system. Additional mechanisms such as differential privacy [26], secure multiparty computation [27] and a trusted execution environment [28] can be utilized to provide additional security. We refer to [1] for a comprehensive survey of privacy-preserving mechanisms.

Federated learning features different strategies, which vary in aggregation methods on a server, in the training process on clients and in the communication frequency between a client and a server. Different strategies may be preferable in different data distribution scenarios: (i) cross-device (many clients with little data) or cross-silo (few clients with a lot of data); (ii) varying degrees of inter-client heterogeneity (dissimilarity); (iii) unavailable or straggling clients. This study considers the cross-silo scenario which is the most typical for genomic data where clients, such as hospitals, biobanks, and genetic testing companies

may be dissimilar due to differing genetic populations.

1.5 Federated learning for genomics and healthcare

Researchers have applied federated learning to a variety of healthcare data, including electronic health records [29], medical images [30] and wearables [31]. A number of surveys describe the applications, prospects, challenges and privacy concerns of federated learning in healthcare [32], [33], [34]. A number of privacy-preserving techniques for genomic data have been proposed, such as federated GWAS [35] [36] and federated PCA for GWAS [37]. However, to the best of our knowledge, the applicability of federated learning to make predictions from full-scale genomic data has not been extensively investigated. Training a federated model on genomic arrays poses additional challenges typical to omics data, such as the vast number of non-independent features (genetic variants). Thus, this paper contributes to the adoption of federated learning in genomics and potentially other omics domains.

In this paper, we analyze the efficacy of federated learning on genomic data in two experiments. First, we show the promise of federated models for phenotype prediction from the large-scale UK Biobank (UKB) genomic data. Next, we analyze federated learning strategies in more detail on the smaller-scale but more heterogeneous 1000 Genomes Project data. We investigate the model behavior on the clients and show the importance of frequent communication to achieve faster convergence in the presence of significant heterogeneity (client dissimilarity).

2 Results

2.1 Phenotype prediction from UK Biobank data

In this experiment, we mimic the situation where genomic data is stored in multiple large silos within the same country. We split the UK Biobank data into 19 datasets according to sample collection centers in different parts of the UK. Some inter-node heterogeneity is present due to the correlation between the UK's genetic population structure and geography [38]. After a standard quality control (QC), we reduced dimensionality by conducting GWAS on each node and selecting top SNPs. Then, we trained local, federated and centralized Lassonet neural networks (see Methods) of identical architecture with selected SNPs, sex and age as features. The experiment setup is visualized in Figure 1 and described in more detail in the Methods section.

Figure 2 displays test R^2 performance of six out of 19 local models, federated (FedAvg, 8 epochs in a communication round, see Methods) and centralized models for nine continuous phenotypes. The six displayed nodes were selected, prior to model training, to demonstrate the whole range of sample sizes. The considered phenotypes were selected based on previous heritability estimates [39]. On each node two local models were trained: one trained on top SNPs

from local GWAS and one trained on top SNPs from meta-GWAS (see Methods). The federated models used SNPs from meta-GWAS and the centralized models used SNPs from centralized GWAS. Local and centralized covariates-only (sex and age) models were also trained as a baseline.

We trained local and centralized models with (i) 'native' features, i.e. SNPs derived from local and centralized GWAS, to represent end-to-end solutions, and (ii) SNPs yielded by meta-GWAS so that local, centralized and federated models can be compared on exactly the same set of features. For centralized models, performance on centralized GWAS SNPs and meta-GWAS SNPs was very similar for all phenotypes, thus, only the former was included in the Figure 2. Local models tend to perform better on meta-GWAS SNPs compared to local GWAS SNPs, most likely because SNP selection via a local GWAS tends to yield more false positives due to an insufficient number of samples. For local models, we see a natural trend that performance improves as the node size grows. Federated models outperform all local models and get close to centralized models.

2.2 Ancestry prediction from 1000 Genomes data

To mimic the situation where genetic testing companies from different parts of the world collaboratively predict ancestry, we split the 1000 Genomes Project data into 5 isolated nodes based on sample superpopulation (African, Native American, East Asian, European, Southern Asian), thus getting high inter-node heterogeneity. After a standard QC, we reduced dimensionality by applying federated PCA [40] to pruned SNPs and then trained local, federated and centralized multilayer perceptrons (MLPs) of identical architecture. The experiment setup is visualized in Figure 3 and described in more detail in the Methods section.

Our goal here is to investigate the performance of federated models as a function of communication between the clients (nodes) and the server in the presence of significant cross-client heterogeneity. For this, we compare FedAvg strategies with a different number of epochs in a round of communication. The training process of federated models is displayed in Figure 4.

Figure 4a compares the validation loss of the centralized and federated models. Federated models show a clear trend that the more communication between the server and the clients, i.e. the more rounds and the fewer local epochs in each round, the faster the convergence is. For the centralized model, convergence was fast, however, it started from a higher loss value because of the random class assignment in a multiclass classification initialization: a centralized model solves a problem for 26 classes whereas each of the five superpopulation nodes has fewer classes (26 in total).

Figure 4b shows the evolution of the client training loss of the FedAvg model with 32 local epochs in a round of communication. Here, the peaks correspond to the initial evaluation of the aggregated parameters sent from the server to the client. Parameter aggregation on the server results in an increase of the local loss as the aggregated parameters are a weighted average of parameters

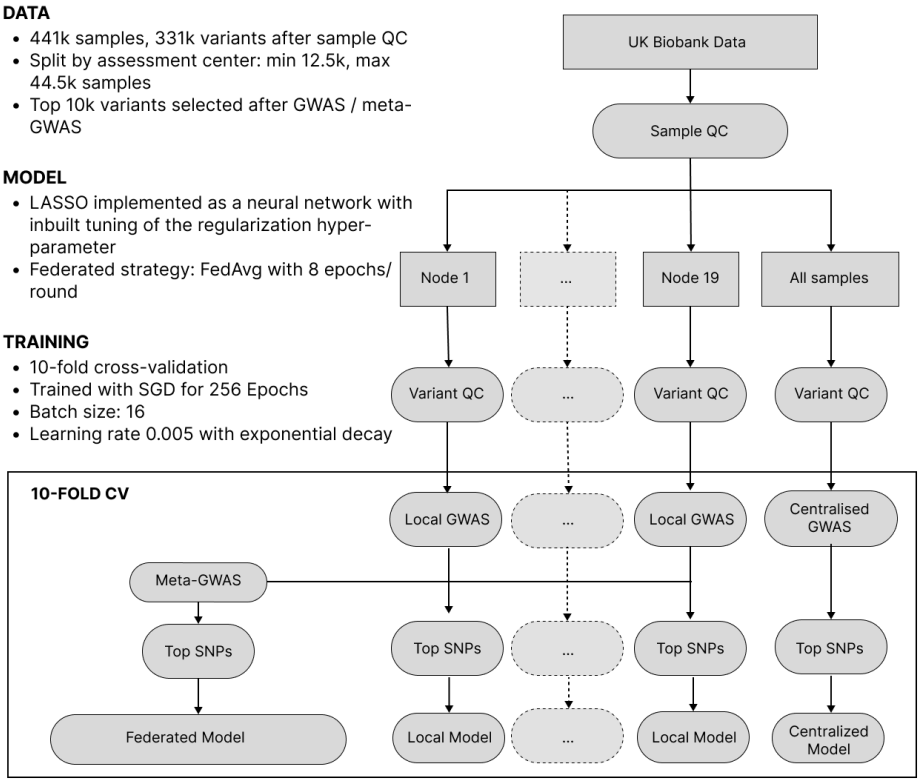


Figure 1: The workflow of phenotype prediction from UKB data. Data is split into nodes and undergoes per-node variant QC. Then, within a cross-validation loop dimensionality is reduced by selecting the most significant SNPs via a local or meta-GWAS. Finally, models are trained.

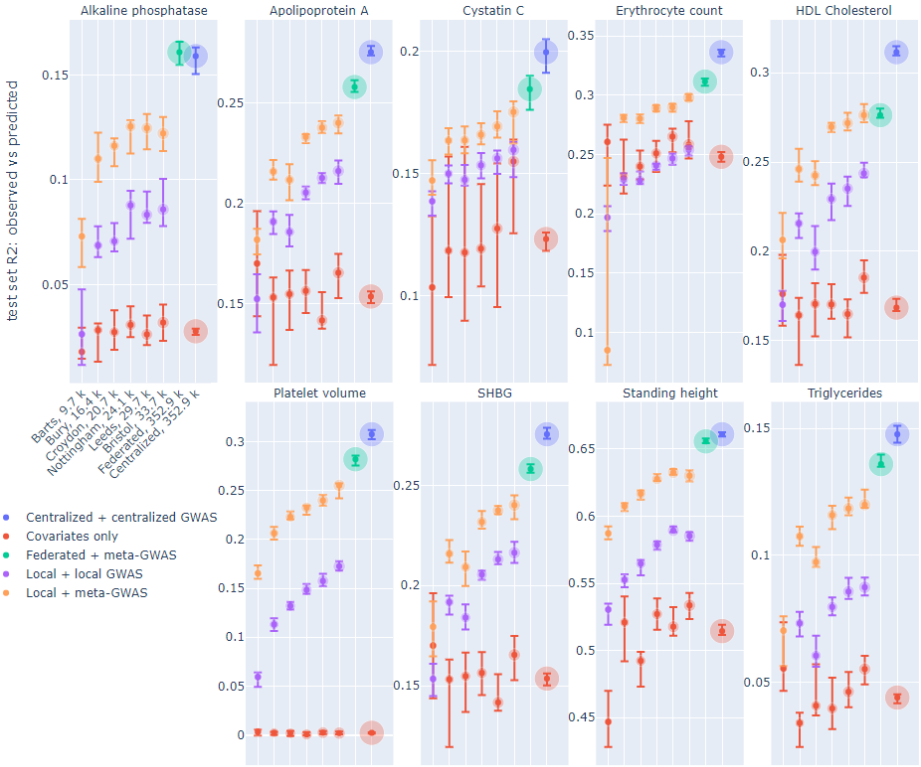


Figure 2: R^2 of observed vs predicted phenotype on a test set of local, federated and centralized models for phenotype prediction on UK Biobank data split by assessment center. Facets correspond to the phenotypes. The x-axis shows six of 19 local nodes, selected to represent a range of node sizes, in increasing size order, which is the same for each phenotype. The number of training samples in each node is indicated in the x-axis labels, which bubble size also corresponds to. Color corresponds to the model type. For each dot, the median and 80% confidence interval for 10-fold cross-validation is displayed.

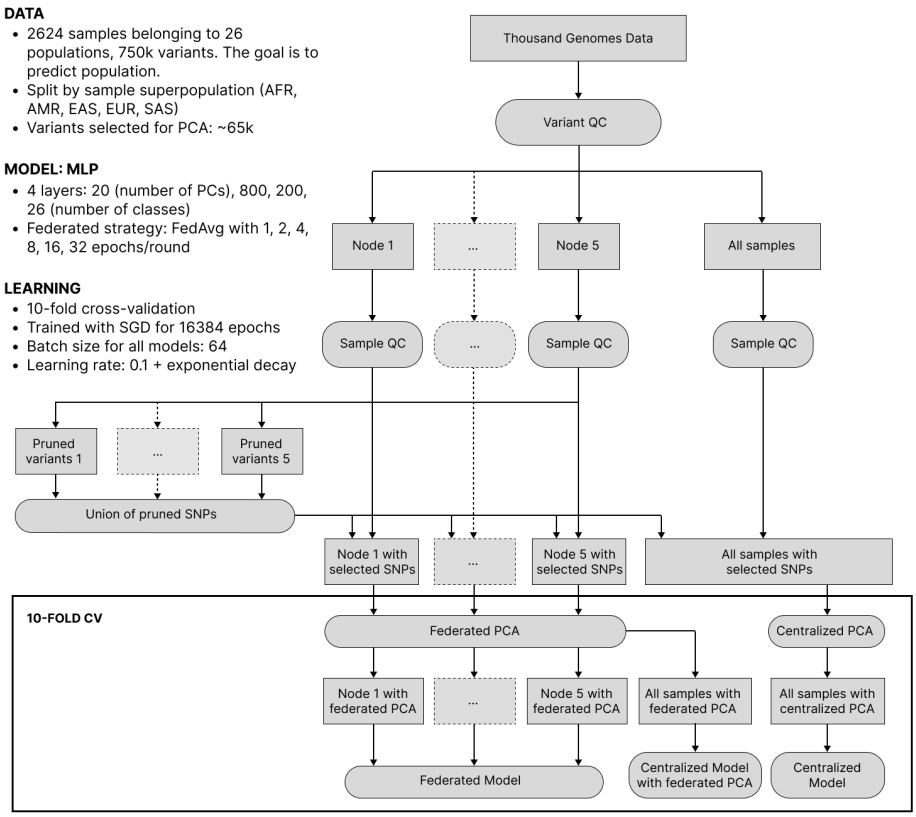
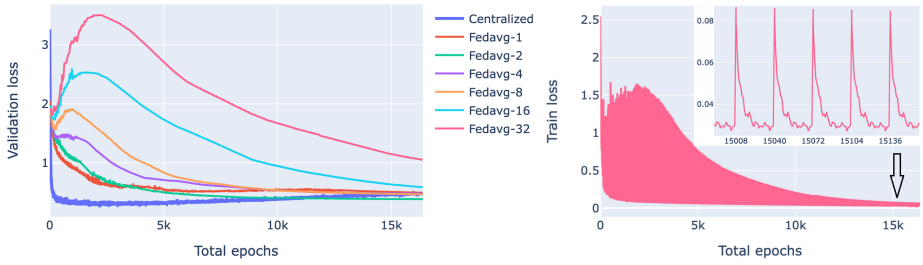


Figure 3: The workflow of ancestry prediction from 1000 Genomes data. Data was split into nodes according to sample superpopulation. The union of variants pruned on each node was taken for all nodes to have the same features. Federated / centralized PCA was used to further decrease dimensionality. Finally, federated and centralized models were trained.



(a) (b)

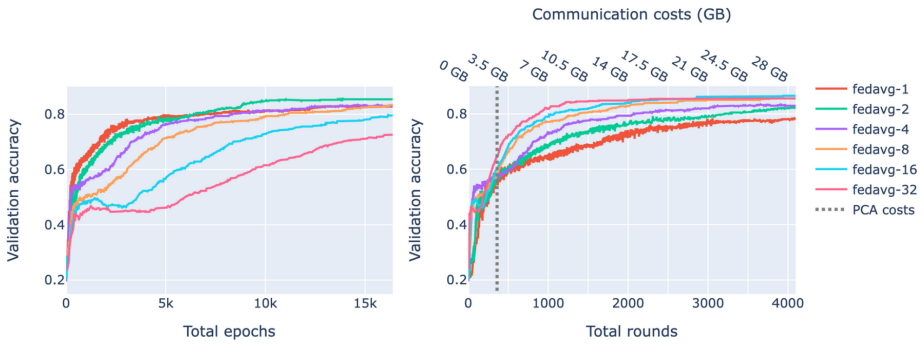
Figure 4: Loss behavior of federated and centralized models. FedAvg strategy with 1, 2, 4, 8, 16, 32 epochs/round was used. (a) Validation loss of federated models with a different number of communication rounds. For each model and epoch, a median loss over 10-fold cross-validation is shown. (b) The training process of a 32-epochs-in-round model on a client. Every 32nd epoch contains two points: one before the start of the epoch when the client receives parameters from the server and one after the first epoch of a round.

optimized on different data distributions (due to inter-node heterogeneity), then the loss starts decreasing as the model starts fitting to the local data.

2.3 Practical considerations of server-client communication

Figure 5 shows the accuracy of federated models as functions of the number of total epochs (computational complexity) and rounds (communication). In compliance with Figure 4a, Figure 5a shows that for heterogeneous data, increasing communication between the clients and the server leads to higher accuracy. On the other hand, Figure 5b shows that for a limited number of rounds, it is beneficial to train locally for a larger number of epochs. Thus, depending on what is the bottleneck of the system, communication or computational complexity, different federated learning strategies may be preferable.

A fully federated solution requires all data to be prepared in a federated manner as well. In the case of ancestry prediction, dimensionality reduction is usually conducted via PCA, thus, we first pruned SNPs as displayed in Figure 3 to decrease computational load and then utilized federated PCA using the P-stack algorithm as described in [40]. The amount of communication used in federated PCA linearly depends on the number of input SNPs which affects the model accuracy. Hence, if communication is limited, one can spend more on the PCA step by including more SNPs or use more communication rounds for model training. Figure 6 shows our rationale behind choosing the pruning parameters that determine the number of input SNPs for federated PCA. We also validate the federated PCA approach and show that the centralized classifier performs similarly on federated and centralized PCs.



(a) (b)

Figure 5: Validation accuracy as a function of complexity and communication. FedAvg strategy with 1, 2, 4, 8, 16, 32 epochs/round was used. Each shown value is a median over 10-fold cross-validation. (a) Accuracy of federated models as a function of the total number of epochs. (b) Accuracy of federated models as a function of the amount of communication between the server and the clients. The dashed line corresponds to the amount of communication used by federated PCA.

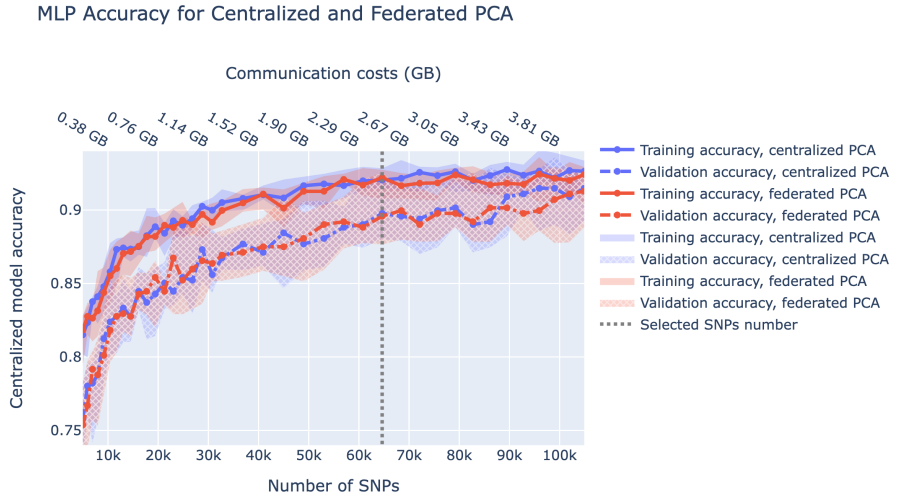


Figure 6: Centralized model accuracy as a function of the number of SNPs used for dimensionality reduction via centralized and federated PCA. Solid/dashed line corresponds to train/validation model accuracy, and blue/red corresponds to centralized/federated PCA. Shaded areas correspond to 80% confidence intervals based on 10-fold cross-validation. Vertical dashed lines correspond to the number of SNPs we chose to be used in downstream analysis.

3 Methods

This study aims to assess the applicability of federated learning to genomic data. In this paper, we consider only 'global' models that aim to perform well on all varieties of data split between isolated datasets. A survey of personalization methods for federated models is provided in [41].

Intuitively, we expect a good federated model to perform considerably better than the local models, trained on data in a single node, and slightly worse than a centralized model that trains on all of the data together. In this case, a federated model delivers all benefits of federated learning at the cost of a small reduction in performance compared to a centralized model.

3.1 Federated learning strategies

Here, we consider the FedAvg strategy [42] with a different number of epochs in a communication round. In our case, where the number of clients is low, the pseudocode is displayed in Algorithm 1:

Algorithm 1 FedAvg for a small number of clients. K is number of clients, n_k – number of samples on kth client, R – number of communication rounds, E – number of local epochs in a round, η – local learning rate

```
procedure SERVER-SIDE
    Initialize ω ← ω_0                                  ▷ Initialize model weights
    for r ∈ 1, .., R do
        for k ∈ 1, .., K do <in parallel>
            ω_r^k ← Client-side(k, ω)                   ▷ Parallel local optimization
        end for
        ω ← Σ_{k=1}^{K} (n_k / n) ω_r^k                 ▷ Weighted average of client weights
    end for
end procedure
procedure CLIENT-SIDE(k, ω)
    for i ∈ 1, .., E do
        for batch b do
            ω ← ω − η∇F(ω, b)                           ▷ Batch gradient descent
        end for
    end for
    return ω
end procedure
```

In the presence of significant inter-node heterogeneity, i.e. when the local data distribution on the clients is different from the global distribution, the global model tends to overfit to local data causing 'client drift' that slows or prevents convergence [43]. Client drift can be decreased by limiting the amount of training on a client in a single round, i.e. increasing communication between a client and a server. In this paper, we utilize FedAvg with different amounts of

server-client communication by varying the number of communication rounds K but keeping the total number of local epochs KE constant.

3.2 The UK Biobank dataset and data processing

An overview of our data processing pipeline for experiments on the UK Biobank dataset can be seen in Figure 1. All samples underwent quality control using PLINK [44] to remove individuals with an insufficient number of genotyped variants (6% missingness cutoff) and related samples with a KING [45] cutoff 0.0884 corresponding to second-degree relatives. For local and federated but not centralized models, the data is split into 19 datasets according to the data collection center (UKB data-field 54). Data collection centers with less than 10k samples were excluded. The size of the datasets after QC ranges from 12.1k (Barts) to 42.1k (Bristol), for a total of 441k samples. For each dataset separately, variant QC is conducted by filtering out variants by rarity (5% minor allele fraction cutoff), missingness (2% cutoff), and a Hardy-Weinberg equilibrium p-value threshold of 10e-6. Each dataset is split into 10 folds for cross-validation where at each time eight folds are used as training data, one as validation data (to be used for early stopping during model training and regularization parameter selection), and one as test data. Next, we reduce dimensionality by selecting the 10,000 most significant SNPs with a GWAS conducted on the training data using age, sex, and 20 principal components from SNPs as covariates. These 10,000 SNPs are combined with age and sex as the input features for our predictive model. For experiments with federated models, feature selection is done by performing a random-effects meta-analysis using PLINK to aggregate information from the GWAS reports of the individual datasets and then selecting the top 10,000 SNPs.

3.3 Phenotype prediction model

The best-performing phenotype prediction models on large datasets like UK Biobank are typically LASSO-based and are solved iteratively to save memory consumption [46, 47, 48]. LASSO is a linear model with L1 regularization, which performs feature selection by setting the weights of non-influential parameters to zero. A LASSO problem can be solved using coordinate descent [46] or gradient descent. We chose to solve LASSO using gradient descent because it can be easily implemented on top of existing deep learning and federated learning frameworks, such as PyTorch [49] and Flower [50].

However, with gradient descent, the LASSO problem can only be solved for a single value of the regularization parameter λ. Since λ determines model performance and generalization ability, one typically trains multiple models with different values of λ and selects the one with the best validation metric. We implemented this procedure as a linear neural network which efficiently trains LASSO models with a range of λ values in parallel and offers built-in model selection. We call this implementation Lassonet.

Centralized and local Lassonet models are trained using PyTorch and PyTorch Lightning [51] on the Zhores cluster node with Nvidia V100 GPU with 16GB VRAM and up to 160GB of RAM [52]. We used the SGD optimizer with learning rate 5e-3, learning rate decay 0.99, batch size 16 and trained Lassonet for 256 local epochs for each run on the UK Biobank data.

We implemented federated models using the Flower framework. Here, we aggregated validation loss from models with the same λ across clients each round, and then chose the model with the best validation loss to be evaluated on the test set.

3.4 The 1000 Genomes Project dataset and data processing

The 1000 Genomes array contains about 750 thousand genetic variants (SNPs) of 2624 samples of 26 genetic populations belonging to five superpopulations of East Asians (EAS), Southern Asians (SAS), Europeans (EUR), Africans (AFR) and Native Americans (AMR).

Our data processing workflow sketched in Figure 3 was performed in the following order. First, we conduct variant QC in PLINK keeping genetic variants with minor allele frequency $>5\%$ and missing call rates $<2\%$. Next, we split the samples into five isolated nodes according to sample superpopulations. Then, we conduct sample QC on each node separately in PLINK keeping non-related (KING relatedness cutoff 0.0884 that corresponds to second-degree relatives) samples with missing call rates $<6\%$.

Next, we reduced dimensionality by first pruning variants on each node separately in PLINK, then taking a union of the remaining variants across the nodes to get a single variant set (only variant IDs are communicated between nodes) yielding about 65 thousand SNPs. Then, data on each node was split into 10 folds for cross-validation where at each time eight folds are used as training data, one as validation data (to be used for early stopping during model training) and one as test data. Finally, we conducted federated PCA and, alternatively, centralized PCA on the training set and extracted the top 20 principal components. To reduce the dimensionality of validation and test sets, we projected them onto the training PC space. The influence of the pruning strictness and the federated vs centralized PCA is displayed in Figure 6.

3.5 Federated PCA for dimensionality reduction

The standard way to reduce dimensionality for ancestry-from-genotype prediction is by using the principal component transformation as it is well known that PCs of genetic variants retain genetic population structure in the dataset [53]. The federated models we use in this study require client datasets to have the same feature space, therefore the PCs have to be obtained collaboratively. Since computing PCs centrally discloses the data and therefore compromizes the purpose of downstream federated learning, we employ the federated PCA approach.

We utilized the P-STACK method as described in [40], which involves sending a local eigenvalues vector and an eigenvector matrix from each client to the server. Further, the server stacks local PCA components and then performs a singular value decomposition (SVD) of the obtained joint matrix. To perform an exact PCA, we used the maximum available number of eigenvectors on each client, which equals the number of client samples minus one. When the number of PCs is fixed, the size of the eigenvector matrix depends only on the number of genetic variants. We used PLINK to prune genetic variants, i.e. removed variants in close linkage disequilibrium. Pruning was conducted on each individual node and then a union of remaining on each node variants was taken. This allowed us to reduce communication costs for federated PCA and significantly shrink RAM consumption while running SVD on the server. When SVD is completed, the resulting eigenvector matrix is sent back to the clients after which each client is able to perform the PC transformation into the joint feature space.

3.6 Ancestry prediction model

After performing the Federated PCA on the 1000 Genomes dataset we use 20 PCs as features for a multi-layer perceptron neural network. It has two hidden layers and outputs raw scores of a sample belonging to the each of 26 populations. It has approximately 182K parameters.

We use a fully-connected neural network with two hidden layers of size 800 and 200, respectively, 20 input and 26 output neurons. The activation function is *selu* and the loss function is cross entropy. We trained it for 16384 local epochs with batch size 64, learning rate 0.1 and exponential learning rate decay with γ 0.9999. We trained it on CPU-only machines with 4 CPUs and 8-16 GB of RAM.

4 Discussion

The promise of federated learning for healthcare, and genomics in particular, is a result of two powerful trends. First, machine learning models require a lot of data to train and their applicability depends on the diversity of the training dataset. Training the model on diverse data obtained from multiple sources reduces confounding by population genetics, experimental design, etc. and generally improves performance on external data. Second, the growing awareness of the sensitivity of healthcare data and the harmful consequences of its leakage encourages data custodians to restrict data access, e.g. by requiring an application approval and then granting access only within a trusted research environment [54] [55]. This makes merging multiple datasets at a single location challenging, thus discouraging training conventional centralized models.

Nevertheless, the applicability of federated learning to genomic data has not been studied extensively. In this paper, we analyzed the behavior of federated models in two scenarios: phenotype-from-genotype prediction on the

UK Biobank data and ancestry-from-genotype prediction on the 1000 Genomes Project data. We first showed that federated models are almost as accurate as centralized models and considerably more accurate than local models for predicting multiple phenotypes from genomic data. Then, we moved to ancestry prediction using the 1000 Genomes dataset which features fewer samples but higher population diversity. By splitting the data by sample superpopulations we achieved high inter-node heterogeneity. We showed that in this setting, frequent communication between the server and the clients plays a crucial role in achieving fast convergence and showing performance similar to that of the centralized model. We also demonstrated that depending on whether computational time or communication is a bottleneck of the system, different FedAvg with different numbers of epochs in a round should be preferred.

In both of our experiments, the main reason for federated models not reaching the performance level of centralized models is data heterogeneity across the nodes or client dissimilarity. When a federated model trains on a client, it overfits to local data; then as fitted parameters from different clients get aggregated, the result may differ from the update of the corresponding centralized model, a phenomenon called client drift. Client drift can be decreased by increasing communication between the client and the server by decreasing the number of epochs of local training in a communication round (between parameter updates), as shown in Figures 4a and 5a.

Federated learning is a quickly developing field of research and new strategies continue to emerge, including those aiming to tackle client drift and improve convergence in case of high inter-node heterogeneity, such as SCAFFOLD [56] and FedDyn [57]. These novel strategies may be preferable if communication between the clients and the server is limited. However, they require additional testing, as FedAvg with frequent communication, one or two epochs in a round, is a difficult baseline to beat.

In both experiments, we used a single dataset artificially split into several independent nodes. On the one hand, this is an advantage as the uniform data collection process allows us to limit the influence of environmental and experimental confounders and focus on the relative performance of the models. On the other hand, in a real scenario, using multiple independently collected datasets may require additional work to unify features and outputs across datasets. For example, predictions from genomic data may require SNP imputation or another solution if different datasets have different sets of genetic variants; similarly, ancestry and phenotypes may be defined or collected differently in different experiments.

Currently, the vast majority of healthcare data comes from people of European descent which makes models trained on this data biased towards Europeans, adding to the healthcare inequality of people around the world [58] [59]. Thus, one of the main potential extensions of this study is to train federated phenotype-from-genotype predicting models across different ancestries. This could be achieved by establishing collaboration between data custodians, such as biobanks, in different countries and training a global federated model that accurately predicts phenotype across different ancestries, potentially even

in those absent from the training set. This global model can be subsequently personalized if the collaborating parties require the model to prioritize their data.

5 Conclusions

Despite its promise, the applicability of federated learning to genomics has not been thoroughly investigated. We filled this gap by training and analyzing federated models in two important scenarios: phenotype-from-genotype prediction and ancestry-from-genotype prediction. We showed that federated models consistently achieve high performance close to that of centralized models for the prediction of multiple phenotypes and ancestry, even in the presence of significant inter-node heterogeneity. For heterogeneous nodes, we investigated the dependency of federated strategies convergence on the amount of communication between the server and the nodes and provided recommendations on which strategy to choose if communication or computational time is a bottleneck. We also showed how federated prediction models can be integrated with federated data processing steps such as dimensionality reduction by federated PCA. This study encourages the adoption of federated models in healthcare, which has the potential to enable global data collaboration and train less biased models that represent diverse genetic ancestries.

6 Acknowledgements

The computations performed in this study on the UK Biobank dataset were done on the Zhores cluster [52], and we thank the CDISE HPC team for their assistance. This research has been conducted using the UK Biobank Resource under Application Number '43661'. We also thank Dmitry Yarotsky, Pavel Nikonorov and Bert Mouler for the valuable discussions.

References

[1] Luca Bonomi, Yingxiang Huang, and Lucila Ohno-Machado. "Privacy challenges and research opportunities for genomic data sharing". In: *Nature genetics* 52.7 (2020), pp. 646–654.

[2] Yann Joly et al. "Comparative approaches to genetic discrimination: chasing shadows?" In: *Trends in Genetics* 33.5 (2017), pp. 299–302.

[3] Carolyn Riley Chapman et al. "Genetic discrimination: emerging ethical challenges in the context of advancing technology". In: *Journal of Law and the Biosciences* 7.1 (2020), lsz016.

[4] Debashree Ray and Michael Boehnke. "Methods for meta-analysis of multiple traits using GWAS summary statistics". In: *Genetic epidemiology* 42.2 (2018), pp. 134–145.

[5] Evangelos Evangelou and John Ioannidis. "Meta-analysis methods for genome-wide association studies and beyond". In: *Nature Reviews Genetics* 14.6 (2013), pp. 379–389.

[6] Cathie Sudlow et al. "UK biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age". In: *PLoS medicine* 12.3 (2015), e1001779.

[7] Mitja I Kurki et al. "FinnGen: Unique genetic insights from combining isolated population and national health register data". In: *medRxiv* (2022).

[8] Cathryn M Lewis and Evangelos Vassos. "Polygenic risk scores: from research tools to clinical instruments". In: *Genome medicine* 12.1 (2020), pp. 1–11.

[9] Alicia R Martin et al. "Clinical use of current polygenic risk scores may exacerbate health disparities". In: *Nature genetics* 51.4 (2019), pp. 584–591.

[10] Deepti Gurdasani et al. "Genomics of disease risk in globally diverse populations". In: *Nature Reviews Genetics* 20.9 (2019), pp. 520–535.

[11] Florian Privé et al. "Portability of 245 polygenic scores when derived from the UK Biobank and applied to 9 ancestry groups from the same cohort". In: *The American Journal of Human Genetics* 109.1 (2022), pp. 12–23.

[12] Arun Durvasula and Kirk E Lohmueller. "Negative selection on complex traits limits phenotype prediction accuracy between populations". In: *The American Journal of Human Genetics* 108.4 (2021), pp. 620–631.

[13] Huwenbo Shi et al. "Population-specific causal disease effect sizes in functionally important regions impacted by selection". In: *Nature communications* 12.1 (2021), pp. 1–15.

[14] Tiffany Amariuta et al. "Improving the trans-ancestry portability of polygenic risk scores by prioritizing variants in predicted cell-type-specific regulatory elements". In: *Nature genetics* 52.12 (2020), pp. 1346–1354.

[15] Brianne E Kirkpatrick and Misha D Rashkin. "Ancestry testing and the practice of genetic counseling". In: *Journal of Genetic Counseling* 26.1 (2017), pp. 6–20.

[16] Annalisa Buniello et al. "The NHGRI-EBI GWAS Catalog of published genome-wide association studies, targeted arrays and summary statistics 2019". In: *Nucleic acids research* 47.D1 (2019), pp. D1005–D1012.

[17] Samuel A Lambert et al. "The Polygenic Score Catalog as an open database for reproducibility and systematic evaluation". In: *Nature Genetics* 53.4 (2021), pp. 420–425.

[18] Hsin-Chou Yang et al. "Genetic ancestry plays a central role in population pharmacogenomics". In: *Communications biology* 4.1 (2021), pp. 1–14.

[19] Alkes L Price et al. "Principal components analysis corrects for stratification in genome-wide association studies". In: *Nature genetics* 38.8 (2006), pp. 904–909.

[20] Florian Privé et al. "Efficient toolkit implementing best practices for principal component analysis of population genetic data". In: *Bioinformatics* 36.16 (2020), pp. 4449–4457.

[21] 1000 Genomes Project Consortium et al. "A global reference for human genetic variation". In: *Nature* 526.7571 (2015), p. 68.

[22] Qiang Yang et al. "Federated machine learning: Concept and applications". In: *ACM Transactions on Intelligent Systems and Technology (TIST)* 10.2 (2019), pp. 1–19.

[23] Jia Qi Lim and Chee Seng Chan. "From Gradient Leakage To Adversarial Attacks In Federated Learning". In: *2021 IEEE International Conference on Image Processing (ICIP)*. 2021, pp. 3602–3606. DOI: 10.1109/ICIP42928.2021.9506589.

[24] Viraaji Mothukuri et al. "A survey on security and privacy of federated learning". In: *Future Generation Computer Systems* 115 (2021), pp. 619–640.

[25] Stefanie Warnat-Herresthal et al. "Swarm Learning for decentralized and confidential clinical machine learning." In: *Nature* 594.7862 (June 2021), pp. 265–270. DOI: 10.1038/s41586-021-03583-3.

[26] Martin Abadi et al. "Deep learning with differential privacy". In: *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*. 2016, pp. 308–318.

[27] Ivan Damgård et al. "Multiparty computation from somewhat homomorphic encryption". In: *Annual Cryptology Conference*. Springer. 2012, pp. 643–662.

[28] Fan Mo et al. "PPFL: Privacy-preserving federated learning with trusted execution environments". In: *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services*. ACM, June 2021, pp. 94–108. ISBN: 9781450384438. DOI: 10.1145/3458864.3466628.

[29] Akhil Vaid et al. "Federated learning of electronic health records to improve mortality prediction in hospitalized patients with COVID-19: machine learning approach". In: *JMIR medical informatics* 9.1 (2021), e24207.

[30] Xiaoxiao Li et al. "Multi-site fMRI analysis using privacy-preserving federated learning and domain adaptation: ABIDE results". In: *Medical Image Analysis* 65 (2020), p. 101765.

[31] Yiqiang Chen et al. "FedHealth: A Federated Transfer Learning Framework for Wearable Healthcare". In: *IEEE Intelligent Systems* 35.4 (2020), pp. 83–93. DOI: 10.1109/MIS.2020.2988604.

[32] Madhura Joshi, Ankit Pal, and Malaikannan Sankarasubbu. "Federated Learning for Healthcare Domain-Pipeline, Applications and Challenges". In: *ACM Transactions on Computing for Healthcare* (2022).

[33] Jie Xu et al. "Federated learning for healthcare informatics". In: *Journal of Healthcare Informatics Research* 5.1 (2021), pp. 1–19.

[34] Nicola Rieke et al. "The future of digital health with federated learning". In: *NPJ digital medicine* 3.1 (2020), pp. 1–7.

[35] Reza Nasirigerdeh et al. "sPLINK: a federated, privacy-preserving tool as a robust alternative to meta-analysis in genome-wide association studies". In: *BioRxiv* (2020).

[36] Md Nazmus Sadat et al. "Safety: secure gwas in federated environment through a hybrid solution". In: *IEEE/ACM transactions on computational biology and bioinformatics* 16.1 (2018), pp. 93–102.

[37] Anne Hartebrodt et al. "Federated principal component analysis for genome-wide association studies". In: *2021 IEEE International Conference on Data Mining (ICDM)*. IEEE. 2021, pp. 1090–1095.

[38] Aman Agrawal et al. "Scalable probabilistic PCA for large-scale genetic variation data". In: *PLoS genetics* 16.5 (2020), e1008773.

[39] Nasa Sinnott-Armstrong et al. "Genetics of 35 blood and urine biomarkers in the UK Biobank". In: *Nature genetics* 53.2 (2021), pp. 185–194.

[40] Anne Hartebrodt and Richard Röttger. "Federated horizontally partitioned principal component analysis for biomedical applications". In: *Bioinformatics Advances* 2.1 (2022), vbac026.

[41] Viraj Kulkarni, Milind Kulkarni, and Aniruddha Pant. "Survey of personalization techniques for federated learning". In: *2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)*. IEEE. 2020, pp. 794–797.

[42] Brendan McMahan et al. "Communication-efficient learning of deep networks from decentralized data". In: *Artificial intelligence and statistics*. PMLR. 2017, pp. 1273–1282.

[43] Xiang Li et al. "On the convergence of fedavg on non-iid data". In: *arXiv preprint arXiv:1907.02189* (2019).

[44] Christopher C Chang et al. "Second-generation PLINK: rising to the challenge of larger and richer datasets". In: *Gigascience* 4.1 (2015), s13742–015.

[45] Ani Manichaikul et al. "Robust relationship inference in genome-wide association studies". In: *Bioinformatics* 26.22 (2010), pp. 2867–2873.

[46] Louis Lello et al. "Genomic Prediction of 16 Complex Disease Risks Including Heart Attack, Diabetes, Breast and Prostate Cancer". In: *Scientific Reports* 9.1 (Dec. 2019), pp. 1–16. ISSN: 20452322. DOI: `10.1038/s41598-019-51258-x`.

[47] Junyang Qian et al. "A fast and scalable framework for large-scale and ultrahigh-dimensional sparse regression with application to the UK biobank". In: *PLoS Genetics* 16.10 (2020), pp. 1–30. ISSN: 15537404. DOI: `10.1371/journal.pgen.1009141`. URL: `http://dx.doi.org/10.1371/journal.pgen.1009141`.

[48] Florian Prive et al. "Efficient analysis of large-scale genome-wide data with two R packages: Bigstatsr and bigsnpr". In: *Bioinformatics* 34.16 (2018), pp. 2781–2787. ISSN: 14602059. DOI: 10.1093/bioinformatics/bty185.

[49] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[50] Daniel J. Beutel et al. *Flower: A Friendly Federated Learning Research Framework*. 2020. DOI: 10.48550/ARXIV.2007.14390. URL: https://arxiv.org/abs/2007.14390.

[51] William Falcon et al. *PyTorchLightning/pytorch-lightning: 0.7.6 release*. May 2020. DOI: 10.5281/ZENODO.3828935. URL: https://zenodo.org/record/3828935 (visited on 01/18/2023).

[52] Igor Zacharov et al. ""Zhores" — Petaflops supercomputer for data-driven modeling, machine learning and artificial intelligence installed in Skolkovo Institute of Science and Technology". en. In: *Open Engineering* 9.1 (Oct. 2019), pp. 512–520. ISSN: 2391-5439. DOI: 10.1515/eng-2019-0059. URL: http://www.degruyter.com/view/j/eng.2019.9.issue-1/eng-2019-0059/eng-2019-0059.xml.

[53] Nick Patterson, Alkes L Price, and David Reich. "Population structure and eigenanalysis". In: *PLoS genetics* 2.12 (2006), e190.

[54] Esma Mansouri-Benssassi et al. "Machine Learning Models Disclosure from Trusted Research Environments (TRE), Challenges and Opportunities". In: *arXiv preprint arXiv:2111.05628* (2021).

[55] Sanaz Kavianpour et al. "Next-generation capabilities in trusted research environments: interview study". In: *Journal of Medical Internet Research* 24.9 (2022), e33720.

[56] Sai Praneeth Karimireddy et al. "Scaffold: Stochastic controlled averaging for federated learning". In: *International Conference on Machine Learning*. PMLR. 2020, pp. 5132–5143.

[57] Durmus Alp Emre Acar et al. "Federated learning based on dynamic regularization". In: *arXiv preprint arXiv:2111.04263* (2021).

[58] "Genetics for all". In: *Nature Genetics* 51.4 (Mar. 2019), pp. 579–579. DOI: 10.1038/s41588-019-0394-y. URL: https://doi.org/10.1038/s41588-019-0394-y.

[59] Alicia R Martin et al. "Current clinical use of polygenic scores will risk exacerbating health disparities". In: *Nature genetics* 51.4 (2019), p. 584.