Reducing Communication Cost in Federated Transfer Learning: An Investigation of S. Li, T. Cai and R. Duan (2023).

Mengqi Xu

Aug 7, 2024

1 Introduction

Precision medicine, also referred to as personalized medicine, is a groundbreaking approach to healthcare that customizes medical treatment based on the unique characteristics of each patient, incorporating genetic, environmental, and lifestyle factors (Ashley, 2016). Currently, the increasing availability of large-scale biomedical data, including electronic health records (EHR) and health surveys, allows for the development of precise personalized risk prediction models in a cost-effective manner (Li et al., 2020).

However, a significant challenge persists for demographic sub-populations that are underrepresented in precision medicine research (Kraft et al., 2018; West, Blacksher, and Burke, 2017). For instance, in the UK Biobank, more than 95% of participants are of European ancestry (Sudlow et al., 2015). Consequently, the effectiveness of many existing genetic risk prediction models in non-European populations is considerably lower than in European-ancestry populations (Duncan et al., 2019). The primary issue is that conventional methods either indiscriminately combine all data, failing to tailor models to specific populations, or rely on data from the specific population for prediction, where the sample size is often insufficient.

Therefore, transfer learning has been recently employed to tackle these issues by transferring shared knowledge from diverse populations to an underrepresented population, enabling comparable model performance with significantly less training data (Weiss, Khoshgoftaar, and Wang, 2016). Figure 1 provides a general illustration of the transfer learning concept.

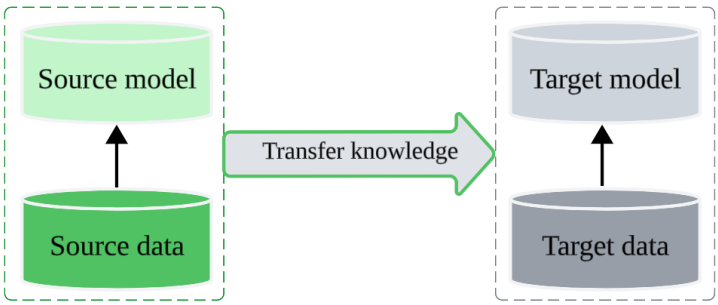

Figure 1: *Transfer learning illustration.*

In addition to incorporating information from various source populations, multi-institutional data integration can increase the sample size of underrepresented populations and data diversity (McCarty et al., 2011). In recent years, numerous research consortia and data networks have been established to support multi-institutional data integration. For instance, the electronic Medical Records and Genomics (eMERGE)

Network is a consortium of ten sites investigating the use of EHR systems for genomic research (Gottesman et al., 2013). However, integrating data across multiple institutions often cannot involve individual-level data due to data privacy concerns (van der Haak et al., 2003) and challenges related to data storage, management, and computation (Kushida et al., 2012). Federated learning, also known as collaborative learning, can be used to deal with multi-institutional health records and biobank data by incorporating summary-level information from different institutions. This method focuses on settings where multiple entities collaboratively train a model while keeping their data decentralized (Wikipedia, 2019). Figure 2 illustrates the federated learning framework.

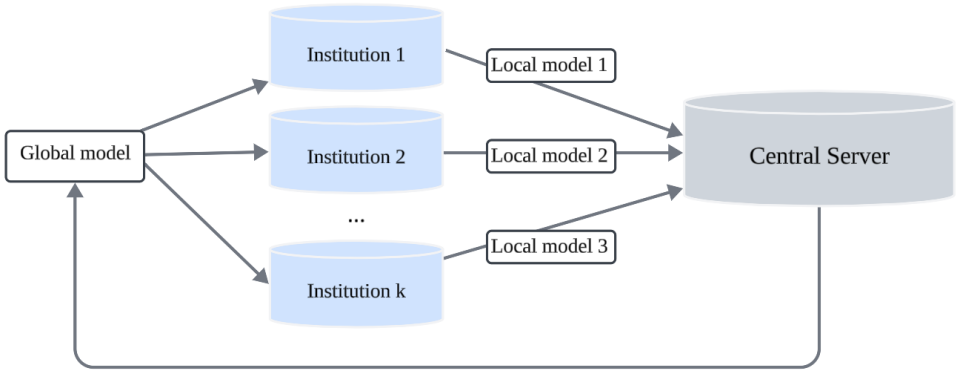

Figure 2: *Federated learning illustration.*

Consider a scenario where we have data from all populations across multiple sites (Figure 3, Li et al., 2023). As shown in Figure 3, each site stores data from all populations, and we aim to predict the targeted underrepresented population (the grey "Target" in Figure 3) by (1) combining shared knowledge across populations and (2) integrating data from multiple sites. This setting presents several challenges: First, individual-level information cannot be shared across sites, making it impossible to pool all health/biobank data together. Second, there are two types of heterogeneity: one is the conditional distribution $f(y|x)$ across populations, and the other is the marginal distribution $f(x)$ across institutions (sites).

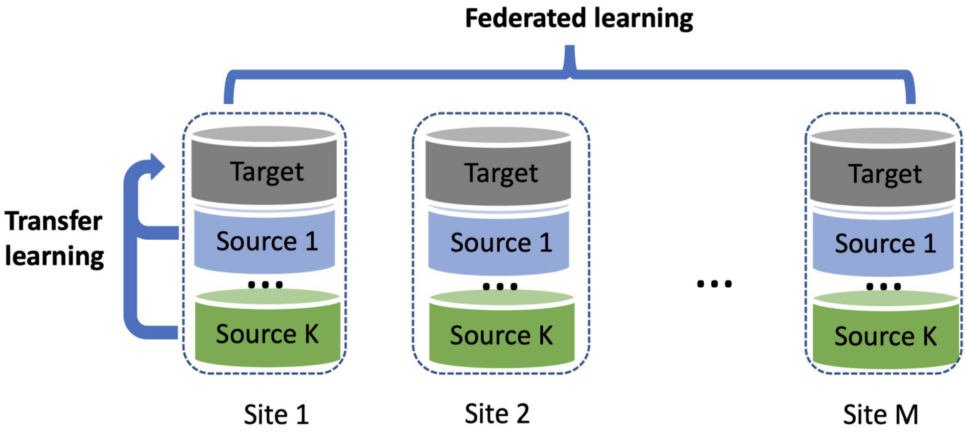

Figure 3: *A schematic illustration of the federated transfer learning framework and the problem setting (Li et al., 2023).*

In existing studies, transfer learning methods primarily focus on settings where individual-level data can

be shared. For instance, transfer learning in high-dimensional linear models (Bastani, 2020; Li, Cai, and Li, 2022) and high-dimensional generalized linear models (Tian and Feng, 2021). These individual data-based methods cannot be directly applied to federated settings due to data sharing constraints and potential heterogeneity across sites. In addition, many federated learning methods focus on settings where the true models are consistent across studies, such as Chen and Xie (2014), Lee et al. (2017), Li, Lin, and Li (2013), and Wang et al. (2019a). To enhance efficiency, surrogate likelihood approaches (Duan et al., 2019; Duan et al., 2020; Jordan, Lee, and Yang, 2019) and distributed multitask learning approaches (Liu et al., 2021; Cai, Liu, and Xia, 2022) have been proposed. However, these methods cannot be easily adapted to federated transfer learning settings where the goal is to train a model for a target population while addressing both data sharing constraints and heterogeneity.

Thus, Li et al (2023) propose a FEderated Transfer Algorithm (FETA), which (1) employs transfer learning to transfer knowledge across different populations, addressing a critical issue in precision medicine where sample sizes from different populations can be highly unbalanced, and (2) utilizes federated learning to incorporate summary-level information from data across multiple sites, requiring only a small number of communications among participating sites and achieving performance comparable to pooled analysis. FETA can overcome data sharing constraints and heterogeneity across sites and populations.

This report aims to introduce the FEderated Transfer Algorithm (FETA) proposed by Li et al (2023), present a research question (details in the following sections), and conduct an investigation. Specifically, in Section 2, we describe the problem setting and notations. In Section 3, we discuss transfer learning and FETA within the high-dimensional Generalized Linear Model (GLM) framework. In Section 4, we propose a potential improvement to reduce the communication cost of FETA, and in Section 5, we conduct a simulation study to further investigate. Finally, in Section 6, we present discussions and future work.

2 Problem Setting and Notations

The federated transfer learning method FETA is based on sparse high-dimensional regression models (Tibshirani, 1996; Bickel, Ritov, and Tsybakov, 2009). As shown in Figure 3, assume there are N subjects in total from $K+1$ populations. The underrepresented population of interest is treated as the target population ($k = 0$), while the other K populations are treated as source populations, indexed by $k = 1, ..., K$. Data for all subjects are stored at M different sites, where individual-level data cannot be shared across sites. Consider the case where the number of populations K is finite but the number of sites M is allowed to grow as the total sample size grows to infinity.

Denote

- $\mathcal{N}^{(m,k)}$: the index sets of the data from the k-th population in the m-th site.
  - We assume $\mathcal{N}^{(m,k)} \cap \mathcal{N}^{(m,k')} = \emptyset$ for any $0 \leq k < k' \leq K$.
- $n^{(m,k)} = |\mathcal{N}^{(m,k)}|$: the corresponding sample size of the k-th population in the m-th site.
- $N^{(k)} = \sum_{m=1}^{M} n^{(m,k)}$; $N = \sum_{k=0}^{K} N^{(k)}$.

We are particularly interested in the challenging scenario $N^{(0)} \ll N$, where the target population has a very small sample size compared to the total populations. Note that at certain sites the relative sample compositions can be arbitrary. For example, some sites may not have data from certain populations, i.e., $n^{(m,k)} = 0$ for some but not all m for $1 \leq m \leq M$.

We consider the high-dimensional setting where the number of predictors $p \gg N^{(0)}$ and N. For the i-th subject, we observe an outcome $y_i \in \mathbb{R}$ and a set of p predictors $\mathbf{x}_i \in \mathbb{R}^p$, including the intercept term. Assume that the target data at the m-th site, $\{\mathbf{x}_i, y_i\}_{i \in \mathcal{N}^{(m,0)}}$, follow a generalized linear model

$$g\{\mathbb{E}(y_i \mid \mathbf{x}_i)\} = \mathbf{x}_i^\top \boldsymbol{\beta}$$

with a canonical link function $g(\cdot)$ and a negative log-likelihood function

$$L^{(m,0)}(\boldsymbol{\beta}) = \sum_{i \in \mathcal{N}^{(m,0)}} \left\{ \psi(\mathbf{x}_i^\top \boldsymbol{\beta}) - y_i \cdot \mathbf{x}_i^\top \boldsymbol{\beta} \right\}$$

for some unknown parameter $\boldsymbol{\beta} \in \mathbb{R}^p$ and $\psi(\cdot)$ uniquely determined by $g(\cdot)$.

Similarly, the k-th source population at the m-th site, $\{\mathbf{x}_i, y_i\}_{i \in \mathcal{N}^{(m,k)}}$, also follow a generalized linear model

$$g\{\mathbb{E}(y_i \mid \mathbf{x}_i)\} = \mathbf{x}_i^\top \mathbf{w}^{(k)}$$

with negative log-likelihood

$$L^{(m,k)}(\mathbf{w}^{(k)}) = \sum_{i \in \mathcal{N}^{(m,k)}} \left\{ \psi(\mathbf{x}_i^\top \mathbf{w}^{(k)}) - y_i \cdot \mathbf{x}_i^\top \mathbf{w}^{(k)} \right\}$$

for some unknown parameter $\mathbf{w}^{(k)} \in \mathbb{R}^p$.

Our goal is to estimate β, using data from $K+1$ populations and M sites.

3 Method (Li et al, 2023)

3.1 Introduction of Transfer Learning

Consider a simple case where *individual-level data could be obtained* from each site. In this scenario, the common transfer learning method can be applied to estimate β using the following three steps:

Step 1: Fit a regression model in each source population. The source-specific parameter estimate from the kth population, denoted by $\hat{\mathbf{w}}^{(k)}$, can be obtained by minimizing the loss function $\sum_{m=1}^{M} L^{(m,k)}(\mathbf{b})$. In the high-dimensional setting, we apply Lasso regression, adding an L_1 penalty term to the optimization. In summary, for $k \in \{1, \ldots, K\}$,

$$\hat{\mathbf{w}}^{(k)} = \arg\min_{\mathbf{b} \in \mathbb{R}^p} \left\{ \frac{1}{N^{(k)}} \sum_{m=1}^{M} L^{(m,k)}(\mathbf{b}) + \lambda^{(k)} \|\mathbf{b}\|_1 \right\}. \tag{1}$$

Step 2: Calibrate each source model using data from the target population. In this step, we use the target data to adjust for the difference between $\boldsymbol{\beta}$ and $\mathbf{w}^{(k)}$:

$$\hat{\boldsymbol{\delta}}^{(k)} = \arg\min_{\mathbf{b} \in \mathbb{R}^p} \left\{ \frac{1}{N^{(0)}} \sum_{m=1}^{M} L^{(m,0)}(\hat{\mathbf{w}}^{(k)} + \mathbf{b}) + \lambda_\delta \|\mathbf{b}\|_1 \right\}, \tag{2}$$

for $k \in \{1, \ldots, K\}$. Threshold $\hat{\boldsymbol{\delta}}^{(k)}$ via $\check{\boldsymbol{\delta}}^{(k)} = \mathcal{H}_{(N^{(0)}/\log p)^{1/2}}(\hat{\boldsymbol{\delta}}^{(k)})$.

Step 3: Joint estimation using source and target data. We pool all the data to jointly estimate $\boldsymbol{\beta}$, adjusting for the estimated differences $\hat{\boldsymbol{\delta}}^{(k)}$ for data from the k-th source population:

$$\hat{\boldsymbol{\beta}} = \arg\min_{\mathbf{b} \in \mathbb{R}^p} \left\{ \frac{1}{N} \sum_{m=1}^{M} L^{(m,0)}(\mathbf{b}) + \frac{1}{N} \sum_{k=1}^{K} \sum_{m=1}^{M} L^{(m,k)}(\mathbf{b} - \check{\boldsymbol{\delta}}^{(k)}) + \lambda_\beta \|\mathbf{b}\|_1 \right\}. \tag{3}$$

In the above algorithm, $\lambda^{(k)}{}_{k=1}^{K}$, λ_δ, and λ_β are tuning parameters. This transfer learning algorithm leverages information from all data across sites, producing more reliable results than the target-only estimator $\hat{\beta}_{\text{target}} = \arg\min_{\mathbf{b}} \left\{ \sum_{m=1}^{M} L^{(m,0)}(\mathbf{b}) + \lambda_{\beta_{\text{target}}} \|\mathbf{b}\|_1 \right\}$. The algorithm also adjusts for differences between populations by introducing the calibration term $\hat{\boldsymbol{\delta}}^{(k)}$, instead of simply pooling all data together and training a unified model.

3.2 FETA: Extension to Federated Setting

However, in a federated setting where individual-level data cannot be shared across institutions, the above transfer learning estimator is not applicable. Note that, in the transfer learning method, individual-level information is only involved in calculating the loss functions $L^{(m,k)}(\mathbf{b})$, where $k = 0, ..., K$ and $m = 1, ..., M$. Therefore, Li et al (2023) consider approximating the loss function $L^{(m,k)}(\mathbf{b})$ using its second-order Taylor expansion:

$$\tilde{L}^{(m,k)}(\mathbf{b}; \overset{\circ}{\mathbf{b}}) = L^{(m,k)}(\overset{\circ}{\mathbf{b}}) + \sum_{m=1}^{M} (\mathbf{b} - \overset{\circ}{\mathbf{b}})^\top \nabla L^{(m,k)}(\overset{\circ}{\mathbf{b}}) + \frac{1}{2} \sum_{m=1}^{M} \nabla^2 L^{(m,k)}(\overset{\circ}{\mathbf{b}})(\mathbf{b} - \overset{\circ}{\mathbf{b}}) \otimes 2.$$

Now, the sites only need to share three sets of summary statistics:

- the current estimate $\overset{\circ}{\mathbf{b}}$,
- the score vector $\nabla L^{(m,k)}(\overset{\circ}{\mathbf{b}})$,
- and the Hessian matrix $\nabla^2 L^{(m,k)}(\overset{\circ}{\mathbf{b}})$,

so they can approximate the loss functions without requiring individual-level data. To summarize, for $k = 0, \ldots, K$, they define the combined **surrogate negative log-likelihood functions** for the k-th population:

$$\hat{\mathbf{R}}^{(k)}(\mathbf{b}; \overset{\circ}{\mathbf{b}}) = \frac{1}{2}(\mathbf{b} - \overset{\circ}{\mathbf{b}})^\top \hat{\mathbf{H}}^{(k)}(\overset{\circ}{\mathbf{b}})(\mathbf{b} - \overset{\circ}{\mathbf{b}}) + \left\langle \mathbf{b} - \overset{\circ}{\mathbf{b}}, \frac{1}{N^{(k)}} \nabla L^{(k)}(\overset{\circ}{\mathbf{b}}) \right\rangle, \tag{4}$$

where

$$\nabla L^{(k)}(\mathbf{b}) = \sum_{m=1}^{M} \nabla L^{(m,k)}(\mathbf{b}), \text{ and } \hat{\mathbf{H}}^{(k)}(\mathbf{b}) = \frac{1}{N^{(k)}} \sum_{m=1}^{M} \nabla^2 L^{(m,k)}(\mathbf{b}).$$

Then, following the three steps in the transfer learning method (in Section 3.1) but replacing the loss functions $\frac{1}{N^{(k)}} \sum_{m=1}^{M} L^{(m,k)}(\mathbf{b})$ with the surrogate losses $\hat{\mathbf{R}}^{(k)}(\mathbf{b}; \overset{\circ}{\mathbf{b}})$, they construct the FEderated Transfer Algorithm (FETA), as detailed in Algorithm 1.

They also proposed a robust version of FETA, which sequentially combines all the populations and uses a target validation dataset at the leading site to select the aggregated estimator with the best performance. Readers can refer to Algorithm 2 in Li et al (2023) for details.

Algorithm 1 Federated Transfer Learning Algorithm (FETA)

Input: Target population $\{X^{(m,0)}, y^{(m,0)}\}_{m=1}^{M}$ and source populations $\{\{X^{(m,k)}, y^{(m,k)}\}_{m=1}^{M}\}_{k=1}^{K}$.
Initial values: $\hat{\boldsymbol{\beta}}_0, \{\hat{\mathbf{w}}_0^{(k)}\}_{k=1}^{K}$.
for $t = 1, \ldots, T$ **do**

Threshold $\check{\mathbf{w}}_{t-1}^{(k)} = \mathcal{H}_{c_n}(\hat{\mathbf{w}}_{t-1}^{(k)})$ and $\check{\boldsymbol{\beta}}_{t-1} = \mathcal{H}_{c_n}(\hat{\boldsymbol{\beta}}_{t-1})$
for $m = 1, \ldots, M$ **do**

Transmit $\{\nabla L^{(m,0)}(\check{\boldsymbol{\beta}}_{t-1}), \{\nabla L^{(m,k)}(\check{\mathbf{w}}_{t-1}^{(k)})\}_{k=1}^{K}\}$ and $\{\nabla^2 L^{(m,0)}(\check{\boldsymbol{\beta}}_{t-1}), \{\nabla^2 L^{(m,k)}(\check{\mathbf{w}}_{t-1}^{(k)})\}_{k=1}^{K}\}$ to the leading site.

end

Combine the first- and second-order information $\nabla L^{(0)}(\check{\boldsymbol{\beta}}_{t-1})$, $\nabla L^{(k)}(\check{\mathbf{w}}_{t-1}^{(k)})$, $\hat{\mathbf{H}}^{(0)}(\check{\boldsymbol{\beta}}_{t-1})$, and $\hat{\mathbf{H}}^{(k)}(\check{\mathbf{w}}_{t-1}^{(k)})$. Compute

$$\hat{\mathbf{w}}_t^{(k)} = \arg\min_{\mathbf{b}\in\mathbb{R}^p} \left\{ \hat{\mathbf{R}}^{(k)}(\mathbf{b}; \hat{\mathbf{w}}_{t-1}^{(k)}) + \lambda^{(k)} \|\mathbf{b}\|_1 \right\}, \quad k = 1, \ldots, K.$$

$$\hat{\boldsymbol{\delta}}_t^{(k)} = \arg\min_{\boldsymbol{\delta}\in\mathbb{R}^p} \left\{ \hat{\mathbf{R}}^{(0)}(\hat{\mathbf{w}}_t^{(k)} + \boldsymbol{\delta}; \check{\boldsymbol{\beta}}_{t-1}) + \lambda_{\boldsymbol{\delta}} \|\boldsymbol{\delta}\|_1 \right\}, \quad k = 1, \ldots, K.$$

Let $\check{\boldsymbol{\delta}}_t^{(k)} = \mathcal{H}_{(N^{(0)}/\log p)^{1/2}}(\hat{\boldsymbol{\delta}}_t^{(k)}), \quad k = 1, \ldots, K.$
Combine all the populations:

$$\hat{\boldsymbol{\beta}}_{\mathbf{t}} = \arg\min_{\mathbf{b}\in\mathbb{R}^p} \left\{ \frac{N^{(0)}}{N} \hat{\mathbf{R}}^{(0)}(\mathbf{b}; \check{\boldsymbol{\beta}}_{\mathbf{t-1}}) + \sum_{k=1}^{K} \frac{N^{(k)}}{N} \hat{\mathbf{R}}^{(k)}(\mathbf{b} + \check{\boldsymbol{\delta}}_t^{(k)}; \check{\boldsymbol{\beta}}_{t-1}) + \lambda_{\boldsymbol{\beta}} \|\mathbf{b}\|_1 \right\}$$

end
Output: $\hat{\boldsymbol{\beta}}_T$

4 Reducing Cost of Sharing Hessian Matrices

4.1 Homogeneous Distribution of x: "Local Hessian" (Li et. al. 2023)

As mentioned in their paper, when the dimension p is very large, transmitting Hessian matrices from all sites results in high communication costs. Li et al (2023) propose another option to reduce these communication costs: when the distributions of covariate variables x are homogeneous across sites for a certain population, they can use the Hessian matrices from a leading site (denoted as the m^*-th site), with more sample sizes from all the $K+1$ populations, to approximate the Hessian matrices across all sites. In this way, only the leading site needs to transmit the Hessian matrices of all the $K+1$ populations, while other sites only share their gradient vectors. Using the empirical Hessian matrix obtained at the leading site, we can get a "local approximation" $\hat{\mathbf{R}}^{(\text{local},k)}(\mathbf{b}; \overset{\circ}{\mathbf{b}})$ of the surrogate loss $\hat{\mathbf{R}}^{(k)}(\mathbf{b}; \overset{\circ}{\mathbf{b}})$:

for $k = 0, \ldots, K$, denote

$$\hat{\mathbf{R}}^{(\text{local},k)}(\mathbf{b}; \overset{\circ}{\mathbf{b}}) = \frac{1}{2}(\mathbf{b} - \overset{\circ}{\mathbf{b}})^\top \hat{\mathbf{H}}^{(m^*,k)}(\overset{\circ}{\mathbf{b}})(\mathbf{b} - \overset{\circ}{\mathbf{b}}) + \langle \mathbf{b} - \overset{\circ}{\mathbf{b}}, \nabla L^{(k)}(\overset{\circ}{\mathbf{b}}) \rangle,$$

where

$$\hat{\mathbf{H}}^{(m^*,k)}(\overset{\circ}{\mathbf{b}}) = \frac{1}{n^{(m^*,k)}} \nabla^2 L^{(m^*,k)}(\overset{\circ}{\mathbf{b}})$$

is the empirical Hessian for the k-th population at $\overset{\circ}{\mathbf{b}}$ based on the samples at the leading site.

Algorithm 2 in the Appendix B shows the whole algorithm of the "local Hessian" method (Same as Algorithm B.1 from the Supplementary Material in Li et al (2023)).

4.2 Heterogeneous Distribution of x: "Local-diagonal" (Our Method)

However, what if the dimension p is very large and the distribution of covariate variables x is also heterogeneous across sites? Besides transmitting the Hessian matrices from the leading site, we can also consider transmitting the diagonal approximation of the Hessian matrix from other sites. This approach can still reduce the communication cost of transmitting all Hessian matrices and potentially improve accuracy compared to Algorithm 2, as it incorporates partial Hessian information from other sites. Now, the approximation $\hat{\mathbf{R}}^{(\text{diag},k)}(\mathbf{b};\overset{\circ}{\mathbf{b}})$ of the surrogate loss $\hat{\mathbf{R}}^{(k)}(\mathbf{b};\overset{\circ}{\mathbf{b}})$ is:

for $k = 0, \ldots, K$,

$$\hat{\mathbf{R}}^{(\text{diag},k)}(\mathbf{b};\overset{\circ}{\mathbf{b}}) = \frac{1}{2}(\mathbf{b}-\overset{\circ}{\mathbf{b}})^\top \hat{\mathbf{H}}^{(m^*,k)}_{\text{diag}}(\overset{\circ}{\mathbf{b}})(\mathbf{b}-\overset{\circ}{\mathbf{b}}) + \langle \mathbf{b}-\overset{\circ}{\mathbf{b}}, \nabla L^{(k)}(\overset{\circ}{\mathbf{b}})\rangle,$$

where

$$\hat{\mathbf{H}}^{(m^*,k)}_{\text{diag}}(\overset{\circ}{\mathbf{b}}) = \frac{1}{N^{(k)}}\sum_{m=1}^{M} \text{diag}(\nabla^2 L^{(m,k)}(\overset{\circ}{\mathbf{b}})) + \frac{1}{n^{(m,k)}}(\nabla^2 L^{(m^*,k)}(\overset{\circ}{\mathbf{b}}) - \text{diag}(L^{(m^*,k)}(\overset{\circ}{\mathbf{b}})))$$

is the approximated Hessian for the k-th population at $\overset{\circ}{\mathbf{b}}$. Intuitively, the diagonal elements of $\hat{\mathbf{H}}^{(m^*,k)}_{\text{diag}}$ are the same as the original FETA Hessian $\hat{\mathbf{H}}^{(m,k)}$, but the off-diagonal elements of $\hat{\mathbf{H}}^{(m^*,k)}_{\text{diag}}$ are the same as local Hessian $\hat{\mathbf{H}}^{(m^*,k)}$.

We can use $\hat{\mathbf{R}}^{(\text{diag},k)}(\mathbf{b};\overset{\circ}{\mathbf{b}})$ to substitute $\hat{\mathbf{R}}^{(k)}(\mathbf{b};\overset{\circ}{\mathbf{b}})$ in Algorithm 1, to obtain a federated transfer learning estimator $\hat{\beta}_T$. Compared with the "local Hessian" method, this method requires non-leading sites to share their diagonal Hessian as well, allowing us to obtain some "rough" information from non-leading sites at a low communication cost. This approach would be theoretically more robust when there is heterogeneity of $f(x)$ across sites.

5 Simulation Study: different levels of heterogeneity

5.1 Data Generation and Study Settings

In our simulation, we aim to investigate the performance of three different methods under varying levels of heterogeneity both across sites and populations: (1) Original FETA, (2) "Local Hessian": federated transfer learning using the Hessian from the leading site only, and (3) "local-diagonal": federated transfer learning using both the Hessian from the leading site and the diagonal Hessian from other sites.

We mimic genetic risk prediction in a federated network with $K = 1$ source population and $M = 2$ institutions (one leading site and one other site). At the leading site, we have $n^{(1,0)} = 100$ samples from the target population and $n^{(1,1)} = 500$ samples from the source population; at the other site, we have $n^{(2,0)} = 40$ samples from the target population and $n^{(2,1)} = 200$ samples from the source population. To illustrate the simulation data framework intuitively, we present Figure 4. We generate $p = 200$ predictors $\mathbf{x}_i$.

Generating covariates $\mathbf{x}_i$: We generate $\mathbf{x}_i$ to mimic genotype data, which takes values (0, 1, and 2). For simplicity, the distributions of $\mathbf{x}_i$ are the same within one site for both source and target populations. To generate $\mathbf{x}_i$ at each site m, we:

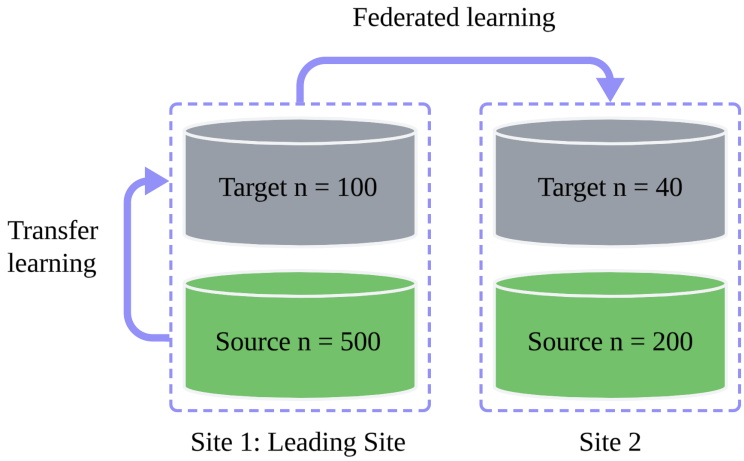

Figure 4: *An illustration of simulation data framework. In addition, we generate* $p = 200$ *predictors* $\mathbf{x}_i$.

1. Generate a p-dimensional multivariate Gaussian vector $\mathbf{z}_i \sim N_p(\mathbf{0}, \mathbf{\Sigma}_m)$, where $\mathbf{\Sigma}_m$ is a block-wise matrix with 20 blocks, each having dimensions 10×10. We set all 20 blocks to be the same, denoted by $\mathbf{B}_m$.
2. Randomly generate minor allele frequencies for the p genetic variants from $U(0, 0.5)$.
3. Obtain $\mathbf{x}_i$ by categorizing each z_i into 0, 1, and 2 based on the corresponding minor allele frequencies.

At the leading site, we set the block matrix $\mathbf{B}_{1,ij} = 0.5^{|i-j|}$. At the other site, we set the block matrix $\mathbf{B}_{2,ij} = (0.5 + b)^{|i-j|}$, where $b = 0, 0.2, 0.8$. As b increases, the difference of distributions $f(\mathbf{x}_i)$ *between the two sites* increases.

Generating outcome y_i**:** For each subject, we generate the binary outcome variable through a logistic regression model:

$$\text{logit}(\mathbb{E}\{y_i|\mathbf{x}_i\}) = \mathbf{x}_i^T \mathbf{b}_i,$$

where $\text{logit}(x) = \log\{x/(1-x)\}$. For the regression coefficients $\mathbf{b}_i$:

- If subject i is from the target population: $\mathbf{b}_i = \beta$, which has $s = 10$ nonzero entries generated from $U(-0.4, 0.4)$.
- If subject i is from the source population: $\mathbf{b}_i = \mathbf{w}^{(1)}$. For $j = 1, ..., p$, $w_j^{(1)} = \beta_j + \delta I(j \in H)$, where H is a random subset of $[p]$ with $|H| = h$ and δ is sampled from $\{-0.5, 0.5\}$.
  - This corresponds to the setting where a small number of genetic variants have relatively large differences in effect sizes across populations. The level of heterogeneity of $f(y_i|\mathbf{x}_i)$ *between populations* increases with h.

5.2 Result

We choose $h = \{10, 30, 50, 100\}$ and $b = \{0, 0.2, 0.8\}$, and compare AUCs from the three methods. The results are shown in Figure 5.

The results show: (1) As the difference in the conditional distribution $f(y|x)$ between the source population and target population increases, the AUCs of all three methods decrease. (2) As the difference in the

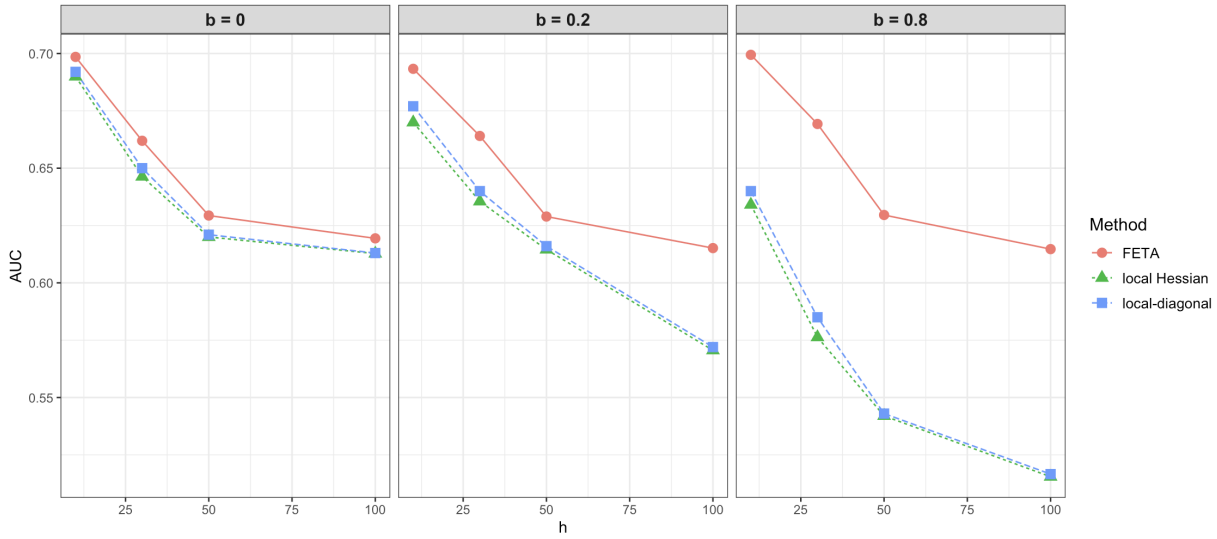

Figure 5: *Simulation result with 100 replications. The difference of the conditional distribution $f(y|x)$ between source and target population increases as h increases. The difference of the marginal distribution $f(x)$ between sites increases as b increases.*

marginal distribution $f(x)$ between sites increases, the local Hessian method performs worse than the original FETA method. (3) The "local-diagonal" method shows a slight improvement over the "local Hessian" method, though not significantly. This suggests that in our data, the off-diagonal elements of the Hessian matrix still play a significant role, given there are $p(p-1)$ off-diagonal elements.

We notice that even when $b = 0$, i.e., the generated distributions $f(x)$ are the same between sites, the "local Hessian" method still produces slightly lower AUCs than the original FETA. One explanation is the randomness of the simulation data. Using one Hessian to approximate the whole Hessian would reasonably result in slightly worse performance, but not by much.

6 Discussion

Li et al (2023) propose a federated transfer learning method that enhances prediction in underrepresented populations by integrating data from different study institutions. Their method overcomes data sharing constraints and accounts for population-level heterogeneity by allowing both the conditional distribution $f(y|x)$ and the marginal distribution $f(x)$ to vary across sites.

In addition to Lasso GLM, the FETA framework is flexible to other models. For example, we can change the penalty terms, e.g., l_2 penalties instead of l_1 penalties. Furthermore, some machine learning methods such as support vector machine (Stolpe, Bhaduri, and Das, 2016) can also be applied. However, some machine learning models may not be directly extended to the FETA framework. For example, random forest has not been formally studied in the distributed setting as far as we know.

Li et al (2023) propose a "local Hessian" method to reduce communication costs when the marginal distribution $f(x)$ is homogeneous across sites, by sharing only one local Hessian matrix from the leading site. In our simulation, this "local Hessian" method is not robust when the distribution $f(x)$ gradually becomes different across sites. To improve this, we propose a "local-diagonal" method: transmitting diagonal Hessian matrices from non-leading sites so that the Hessian matrix resembles the global Hessian more closely, while

still reducing the communication cost of the original FETA. Through our simulation, the "local-diagonal" method performs slightly better than the "local Hessian" method, though not significantly. This result is somewhat consistent with our expectations, although the performance is not particularly good, because the Hessian matrix still has $p(p-1)$ off-diagonal elements.

In the future, we can try to fit a density ratio model between each dataset and the target data at the leading site. Then we can use the leading target data to approximate the Hessian matrices of the other datasets through the density ratio tilting method (Duan, Ning, and Chen, 2022).

A Loss Function, Score Function and Hessian Matrix

A.1 General Formulas

Loss Function (Second-Order Expansion)

$$\widetilde{L}^{(m,k)}(b;\hat{b}) = L^{(m,k)}(\hat{b}) + \sum_{m=1}^{M}(b-\hat{b})^{\top}\nabla L^{(m,k)}(\hat{b}) + \frac{1}{2}\sum_{m=1}^{M}\nabla^2 L^{(m,k)}(\hat{b})(b-\hat{b})^{\otimes 2}.$$

Score Vector (First Derivative)
The score vector (gradient) with respect to $\mathbf{w}^{(k)}$ is given by:

$$\nabla L^{(m,k)}(\hat{b}) = \frac{\partial L^{(m,k)}(\mathbf{w}^{(k)})}{\partial \mathbf{w}^{(k)}} = \sum_{i\in\mathcal{N}^{(m,k)}}\left[\psi'(\mathbf{x}_i^{\top}\mathbf{w}^{(k)})\mathbf{x}_i - y_i\mathbf{x}_i\right],$$

where $\psi'(\cdot)$ is the derivative of $\psi(\cdot)$ with respect to its argument.

Hessian Matrix (Second Derivative)
The Hessian matrix (second derivative) with respect to $\mathbf{w}^{(k)}$ is given by:

$$\nabla^2 L^{(m,k)}(\hat{b}) = \frac{\partial^2 L^{(m,k)}(\mathbf{w}^{(k)})}{\partial \mathbf{w}^{(k)}\partial \mathbf{w}^{(k)\top}} = \sum_{i\in\mathcal{N}^{(m,k)}}\left[\psi''(\mathbf{x}_i^{\top}\mathbf{w}^{(k)})\mathbf{x}_i\mathbf{x}_i^{\top}\right],$$

where $\psi''(\cdot)$ is the second derivative of $\psi(\cdot)$ with respect to its argument.

A.2 Specific Case: Logistic Regression

For logistic regression, the function ψ is related to the logistic function. Specifically, the negative log-likelihood for logistic regression can be written as:

$$L^{(m,k)}(\mathbf{w}^{(k)}) = \sum_{i\in\mathcal{N}^{(m,k)}}\left\{\log(1+\exp(\mathbf{x}_i^{\top}\mathbf{w}^{(k)})) - y_i\cdot\mathbf{x}_i^{\top}\mathbf{w}^{(k)})\right\}.$$

Let

$$\mu(\mathbf{x}_i^{\top}\mathbf{w}^{(k)}) = \frac{\exp(\mathbf{x}_i^{\top}\mathbf{w}^{(k)})}{1+\exp(\mathbf{x}_i^{\top}\mathbf{w}^{(k)})}.$$

The score vector for logistic regression is:

$$\nabla L^{(m,k)}(\mathbf{w}^{(k)}) = \sum_{i\in\mathcal{N}^{(m,k)}}\left[\mu(\mathbf{x}_i^{\top}\mathbf{w}^{(k)})\mathbf{x}_i - y_i\mathbf{x}_i\right] = \sum_{i\in\mathcal{N}^{(m,k)}}\mathbf{x}_i\left[\mu(\mathbf{x}_i^{\top}\mathbf{w}^{(k)}) - y_i\right].$$

The Hessian matrix for logistic regression is:

$$\nabla^2 L^{(m,k)}(\mathbf{w}^{(k)}) = \sum_{i\in\mathcal{N}^{(m,k)}}\left[\mu(\mathbf{x}_i^{\top}\mathbf{w}^{(k)})(1-\mu(\mathbf{x}_i^{\top}\mathbf{w}^{(k)}))\mathbf{x}_i\mathbf{x}_i^{\top}\right].$$

B Algorithm 2

Algorithm 2 Federated transfer learning leveraging local Hessian

Input: Target population $\{X^{(m,0)}, y^{(m,0)}\}_{m=1}^{M}$ and source populations $\{\{X^{(m,k)}, y^{(m,k)}\}_{m=1}^{M}\}_{k=1}^{K}$. Initial values $\hat{\boldsymbol{\delta}}_0, \{\hat{\mathbf{w}}_0^{(k)}\}_{k=1}^{K}$.

for $t = 1, \ldots, T$ **do**

  Threshold $\check{\mathbf{w}}_{t-1}^{(k)} = \mathcal{H}_{c_n}(\hat{\mathbf{w}}_{t-1}^{(k)})$ and $\check{\boldsymbol{\delta}}_{t-1} = \mathcal{H}_{c_n}(\hat{\boldsymbol{\delta}}_{t-1})$.

  for $m = 1, \ldots, M$ **do**

    Transmit $\nabla L^{(m,0)}(\check{\boldsymbol{\delta}}_{t-1})$ and $\{\nabla L^{(m,k)}(\check{\mathbf{w}}_{t-1}^{(k)})\}_{k=1}^{K}$ to the leading site.

  end

  Compute the combined first-order information $\nabla L^{(0)}(\check{\boldsymbol{\delta}}_{t-1})$, $\nabla L^{(k)}(\check{\mathbf{w}}_{t-1}^{(k)})$ according to (2.4).

  In Algorithm 1, we replace $\hat{\mathbf{R}}^{(k)}(\mathbf{b}; \mathbf{b}')$ with $\hat{\mathbf{R}}^{(\text{local},k)}(\mathbf{b}; \mathbf{b}')$ and replace $\lambda^{(k)}, \lambda_{\boldsymbol{\delta}}, \lambda_{\boldsymbol{\delta}}$ with $\lambda_t^{(k)}, \lambda_{\boldsymbol{\delta},t}^{(k)}, \lambda_{\boldsymbol{\delta},t}$, respectively.

end

Output: $\hat{\boldsymbol{\delta}}_T$

References

[1] Ashley, E. A. (2016). Towards precision medicine. *Nat. Rev. Genet.* 17, 507–522.

[2] Li, S., Cai, T. T., & Li, H. (2023). Transfer learning in large-scale Gaussian graphical models with false discovery rate control. *J Am Stat Assoc.* 118(543):2171-2183.

[3] Kraft, S. A., Cho, M. K., Gillespie, K., et al. (2018). Beyond consent: Building trusting relationships with diverse populations in precision medicine research. *Am. J. Bioethics* 18, 3–20.

[4] West, K. M., Blacksher, E., & Burke, W. (2017). Genomics, health disparities, and missed opportunities for the nation's research agenda. *JAMA* 317, 1831–1832.

[5] Sudlow, C., Gallacher, J., Allen, N., et al. (2015). UK Biobank: An open access resource for identifying the causes of a wide range of complex diseases of middle and old age. *PLoS Med.* 12, e1001779.

[6] Duncan, L., Shen, H., Gelaye, B., Meijsen, J., Ressler, K., Feldman, M., Peterson, R., & Domingue, B. (2019). Analysis of polygenic risk score usage and performance in diverse human populations. *Nat. Commun.* 10, 1–9.

[7] Weiss, K., Khoshgoftaar, T. M., & Wang, D. (2016). A survey of transfer learning. *J. Big Data* 3, 1–40.

[8] McCarty, C. A., Chisholm, R. L., Chute, C. G., et al. (2011). The eMERGE network: A consortium of biorepositories linked to electronic medical records data for conducting genomic studies. *BMC Med. Genom.* 4, 1–11.

[9] Gottesman, O., Kuivaniemi, H., Tromp, G., Faucett, W. A., Li, R., Manolio, T. A., Sanderson, S. C., Kannry, J., Zinberg, R., et al. (2013). The electronic medical records and genomics (eMERGE) network: Past, present, and future. *Genet. Med.* 15, 761–771.

[10] Van der Haak, M., Wolff, A. C., Brandner, R., et al. (2003). Data security and protection in cross-institutional electronic patient records. *Int. J. Med. Inform.* 70, 117–130.

[11] Kushida, C. A., Nichols, D. A., Jadrnicek, R., et al. (2012). Strategies for de-identification and anonymization of electronic health record data for use in multicenter research studies. *Med. Care* 50, S82.

[12] Wikipedia Contributors. (2019). Federated learning. *Wikipedia, Wikimedia Foundation*

[13] Li, S., Cai, T. T., & Duan, R. (2023). Targeting underrepresented populations in precision medicine: A federated transfer learning approach. *Ann. Appl. Stat.* 17(4), 2970-2992.

[14] Bastani, H. (2020). Predicting with proxies: Transfer learning in high dimension. *Manage. Sci.* 67, 2657–3320.

[15] Li, S., Cai, T. T., & Li, H. (2022). Transfer learning for high-dimensional linear regression: Prediction, estimation and minimax optimality. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* 84, 149–173.

[16] Tian, Y., & Feng, Y. (2022). Transfer learning under high-dimensional generalized linear models. *J. Amer. Statist. Assoc.* 0, 1-14.

[17] Chen, X., & Xie, M. (2014). A split-and-conquer approach for analysis of extraordinarily large data. *Statist. Sinica* 24, 1655–1684.

[18] Lee, J. D., Liu, Q., Sun, Y., & Taylor, J. E. (2017). Communication-efficient sparse regression. *J. Mach. Learn. Res.* 18, Paper No. 5, 30.

[19] Li, R., Lin, D. K. J., & Li, B. (2013). Statistical inference in massive data sets. *Appl. Stoch. Models Bus. Ind.* 29, 399–409.

[20] Wang, X., Yang, Z., Chen, X., & Liu, W. (2019a). Distributed inference for linear support vector machine. *J. Mach. Learn. Res.* 20, Paper No. 113, 41.

[21] Duan, R., Boland, M. R., Moore, J. H., & Chen, Y. (2019). ODAL: A one-shot distributed algorithm to perform logistic regressions on electronic health records data from multiple clinical sites. *Pacific Symposium on Biocomputing* 30–41.

[22] Duan, R., Luo, C., Schuemie, M. J., et al. (2020). Learning from local to global: An efficient distributed algorithm for modeling time-to-event data. *J. Amer. Med. Inform. Assoc.* 27, 1028–1036.

[23] Jordan, M. I., Lee, J. D., & Yang, Y. (2019). Communication-efficient distributed statistical inference. *J. Amer. Statist. Assoc.* 114, 668–681.

[24] Liu, M., Xia, Y., Cho, K., & Cai, T. (2021). Integrative high dimensional multiple testing with heterogeneity under data sharing constraints. *J. Mach. Learn. Res.* 22, Paper No. 126, 26.

[25] Cai, T., Liu, M., & Xia, Y. (2022). Individual data protected integrative regression analysis of high-dimensional heterogeneous data. *J. Amer. Statist. Assoc.* 117, 2105–2119.

[26] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *J. Roy. Statist. Soc. Ser. B* 58, 267–288.

[27] Bickel, P. J., Ritov, Y., & Tsybakov, A. B. (2009). Simultaneous analysis of lasso and Dantzig selector. *Ann. Statist.* 37, 1705–1732.

[28] Duan, R., Ning, Y., & Chen, Y. (2022). Heterogeneity-aware and communication-efficient distributed statistical inference. *Biometrika* 109, 67–83.