NextGen

Genome-Centric Multimodal
Data Integration in Personalised
Cardiovascular Medicine.

Deliverable D3.1
Technical methods for federated genomic analysis

Grant Agreement Number: 101136962

Funded by
the European Union

UK Research and Innovation

The British associated partners of NextGen were funded by UK Research and Innovation (UKRI) under the UK government's Horizon Europe funding guarantee [grant agreements No. 10098097, No. 10104323]

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

Federal Department of Economic Affairs,
Education and Research EAER
**State Secretariat for Education,
Research and Innovation SERI**

NextGen	
Project full title	Next Generation Tools for Genome-Centric Multimodal Data Integration In Personalised Cardiovascular Medicine
Call identifier	HORIZON-HLTH-2023-TOOL-05-04
Type of action	RIA
Start date	01/01/2024
End date	31/12/2027
Grant agreement no	101136962

Funding of associated partners
The Swiss associated partners of the NextGen project were funded by the Swiss State Secretariat for Education, Research and Innovation (SERI). The British associated partners of NextGen were funded by UK Research and Innovation (UKRI) under the UK government's Horizon Europe funding guarantee [grant agreements No 10098097, No 10104323]

D3.5 – Synthetic datasets for testing and piloting-1	
Author(s)	Marco Scutari
Editor	Francesca Mangili
Participating partners	SUPSI
Version	1.0
Status	Final
Deliverable date	M12
Dissemination Level	PU - Public
Official date	2024-12-03
Actual date	2024-12-03

Disclaimer

This document contains material, which is the copyright of certain **NextGen** contractors, and may not be reproduced or copied without permission. All **NextGen** consortium partners have agreed to the full publication of this document if not declared “Confidential”. The commercial use of any information contained in this document may require a licence from the proprietor of that information. The reproduction of this document or of parts of it requires an agreement with the proprietor of that information., according to the provisions of the Grant Agreement nr. 101136962 and the Consortium Agreement.

Furthermore, a disclaimer will be included, indicating that: “Any communication or publication related to the action, whether produced collectively by the beneficiaries or individually in any format and through any medium, represents solely the views of the author, and the Commission bears no responsibility for any use that may be made of the information contained therein.”

The NEXTGEN consortium consists of the following partners:

No	PARTNER ORGANISATION NAME	ABBREVIATION	COUNTRY
1	UNIVERSITAIR MEDISCH CENTRUM UTRECHT	UMCU	NL
2	HIRO MICRODATACENTERS B.V.	HIRO	NL
3	EURECOM GIE	EURE	FR
4	JOHANN WOLFGANG GOETHE-UNIVERSITAET FRANKFURT AM MAIN	GUF	DE
5	KAROLINSKA INSTITUTET	KI	SE
6	HUS-YHTYMA	HUS	FI
7	UNIVERSITY OF VIRGINIA	UVA	US
8	KLINIKUM RECHTS DER ISAR DER TECHNISCHEN UNIVERSITAT MUNCHEN	TUM-Med	DE
9	HL7 INTERNATIONAL FOUNDATION	HL7	BE
10	MYDATA GLOBAL RY	MYDTA	FI
11	DATAPOWER SRL	DPOW	IT
12	SOCIETE EUROPEENNE DE CARDIOLOGIE	ESC	FR
13	WELLSPAN HEALTH	WSPAN	US
14	LIKE HEALTHCARE RESEARCH GMBH	LIKE	DE
15	NEBS SRL	NEBS	BE
16	THE HUMAN COLOSSUS FOUNDATION	HCF	CH
17	SCUOLA UNIVERSITARIA PROFESSIONALE DELLA SVIZZERA ITALIANA	SUPSI	CH
18	DRUG INFORMATION ASSOCIATION	DIA	CH
19	DPO ASSOCIATES SARL	DPOA	CH
20	QUEEN MARY UNIVERSITY OF LONDON	QMUL	UK
21	EARLHAM INSTITUTE	ERLH	UK

Document Revision History

DATE	VERSIO N	DESCRIPTION	CONTRIBUTIONS
17/12/2024	1.0	Complete draft.	SUPSI

Authors

AUTHOR/EDITOR	ORGANISATION
Marco Scutari	SUPSI
Francesca Mangili	SUPSI

Reviewers

REVIEWER	ORGANISATION

List of terms and abbreviations

ABBREVIATION	DESCRIPTION
AI	Artificial intelligence
EDHS	European Health DataSpace
FL	Federated (machine) learning
GWAS	Genome-wide association study
KPI	Key Performance Indicator
ML	Machine learning
MMIO	Multi-model integration object
PGS	Polygenic (risk) score
VCF	Variant call format
WP	Work Package

Table of contents

Summary

Federated learning leverages data across institutions to improve clinical discovery while complying with data-sharing restrictions and protecting patient privacy. As the evolution of biobanks in genetics and systems biology has proved, accessing more extensive and varied data pools leads to a faster and more robust exploration and translation of results. More widespread use of federated learning may have the same impact in bioinformatics, allowing access to many combinations of genotypic, phenotypic and environmental information that are undercovered or not included in existing biobanks. This document reviews the methodological issues that academic and clinical institutions must address before implementing it.

1 Introduction

Sharing personal information has been increasingly regulated in both the EU (with the GDPR; European Union, 2016) and the US (with the AI Act; U.S. Congress, 2020) to mitigate the personal and societal risks associated with their use, particularly in connection with machine learning and AI models (Cath *et al.*, 2018). These regulations make multi-centre studies and similar endeavours more challenging, impacting biomedical and clinical research.

Federated learning (FL; McMahan *et al.*, 2017;Ludwig and Baracaldo, 2022) is a technical solution to reducing the impact of these restrictions. FL allows multiple parties to train a global machine learning model collaboratively from the respective data without sharing the data themselves and without any meaningful model performance degradation. Instead, parties only share model updates, making it impractical to reconstruct personal information. This approach strengthens security by keeping sensitive information local, enhances privacy by minimising data exposure even between the parties involved, and limits the risk of data misuse by allowing each party to retain complete control over its data (Truong *et al.*, 2021). If enough parties are involved, FL also allows access to larger and more varied data pools than centralised biobanks can provide, resulting in moreaccurate and robust models than those produced by any individual party.

As a result, FL has proven to be a valuable tool for biomedical research that will gain further traction in the coming years. Its use has improved breast density classification models (accuracy up by 6%, generalisability up by 46%; Roth *et al.*, 2020), COVID-19 outcome prediction at both 24h and 72h (up 16% and 38%; Dayan *et al.*, 2021) and rare tumour segmentation (up by 23-33% and 15%; Pati *et al.*, 2022) compared to single-party analyses. A consortium of ten pharmaceutical companies found that FL improved structure-activity relationship (QSAR) models for drug discovery (both up 12% Heyndrickx *et al.*, 2023). Early-stage applications building predictive models from electronic health records (Brisimi *et al.*, 2018) have also confirmed no practical performance degradation compared to pooling data from all parties.

To achieve such results, a real-world implementation of FL must overcome several methodological, infrastructural and legal issues. However, biomedical FL literature reviews (Xu *et al.*, 2020; Chowdhury *et al.*, 2022, among others) are predominantly high-level and considered simulated rather than real-world implementations. Instead, we will cover federated methods designed explicitly for bioinformatics and discuss the infrastructure they need.

2 Federated Learning

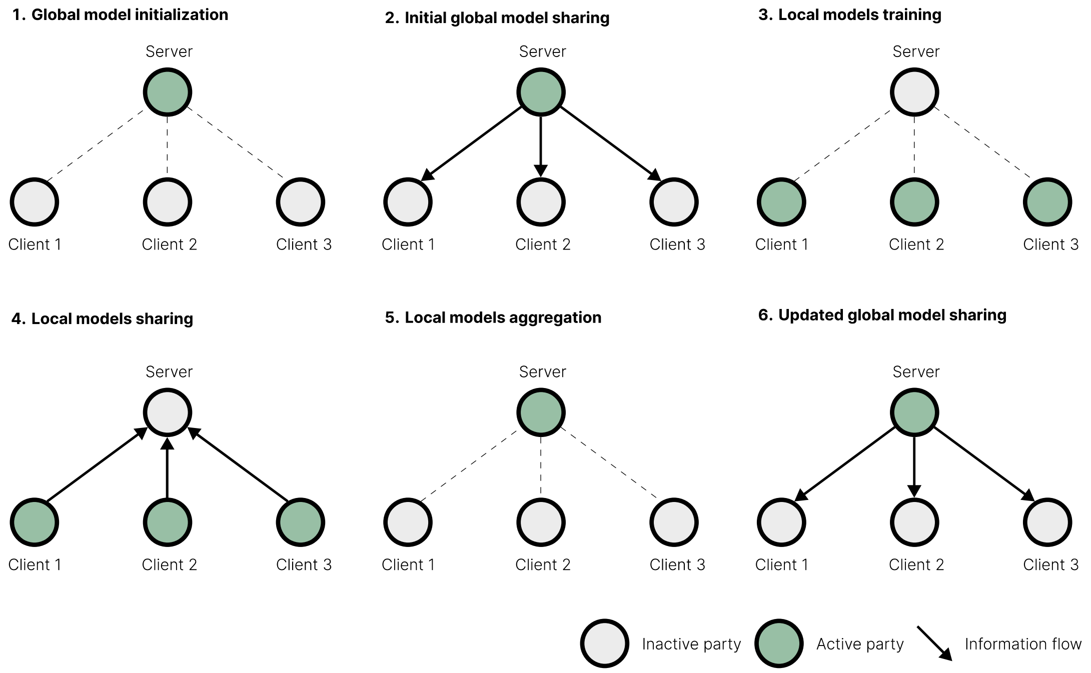


Figure 1: A typical FL workflow. (1) The central server initialises the global model. (2) The initial global model is distributed to participating parties, commonly called clients. (3) The model is trained locally on client data. (4) Clients share their locally trained models with the server. (5) The server aggregates the local models to update the global model. (6) The updated global model is shared with clients for the next training round. Steps (3) to (6) are repeated iteratively until a predefined stopping criterion is met. We highlight active and inactive parties at each step and the flow of information within the consortium.

FL is a collaborative approach to machine learning model training, where multiple data holders form a consortium to jointly train a shared model by exchanging model updates rather than raw data. Typically, FL involves data holders (called "clients") sharing their local contributions with a server (McMahan *et al.*, 2017) as outlined in Figure 1. The server then creates and shares back a global model, inviting the data holders to update and resubmit their contributions. This process is iterative and involves several rounds of model update exchanges. Unlike traditional centralised computing, FL does not store data in a central location. Instead, data remain under the control of the respective data owners at their sites, enhancing privacy.

FL has similarities with distributed computing, meta-analysis, and trusted research environments (TREs) but also has key differences, which we highlight below.

Distributed computing (Zomaya, 2006) divides a computational task among multiple machines to enhance processing speed and efficiency. Usually, it starts from a centrally managed data set spread across different machines and assumed to contain independent and identically distributed observations. Each machine is tasked to process a comparable quantity of data. In contrast, clients independently join FL with their locally-held data, which may vary significantly in quantity and distribution. While sharing some techniques with FL, distributed computing aims for computational efficiency and lacks the privacy focus that characterises FL.

On the other hand, meta-analysis (Toro-Dominguez *et al.*, 2021) aggregates previously completed studies' results using statistical methods to account for variations across studies, thus allowing researchers to synthesise findings without accessing raw data and preserve the privacy of individual data sets. Here, FL collaboratively trains a joint model using distributed data to iteratively update it while meta- analysis constructs it in a single step from pre-existing results.

TREs (Kavianpour *et al.*, 2022) provide access to data within a controlled, secure computing environment for conducting analyses, almost always disallowing data sharing. Some TREs have a centralised data location and governance; an example is the Research Analysis Platform (RAP), the TRE for the UK Biobank (UKB; Sudlow *et al.*, 2015). Others, such as FEGA (Federated European Genome-phenome Archive, 2024), are decentralised. Each data holder maintains their data locally; only the relevant data are securely transferred to the computing environment when the analysis is authorised. Unlike FL, the learning process is not distributed across the data holders. Thus, the tradeoff between TREs and FL is between a centralised, trusted entity with substantial compute that can place substantial restrictions on the analysis; and a consortium that ensures all parties apply governance guidelines, requires them to provide compute and trust each other, but can scale both data access and privacy guarantees.

2.1 Topologies

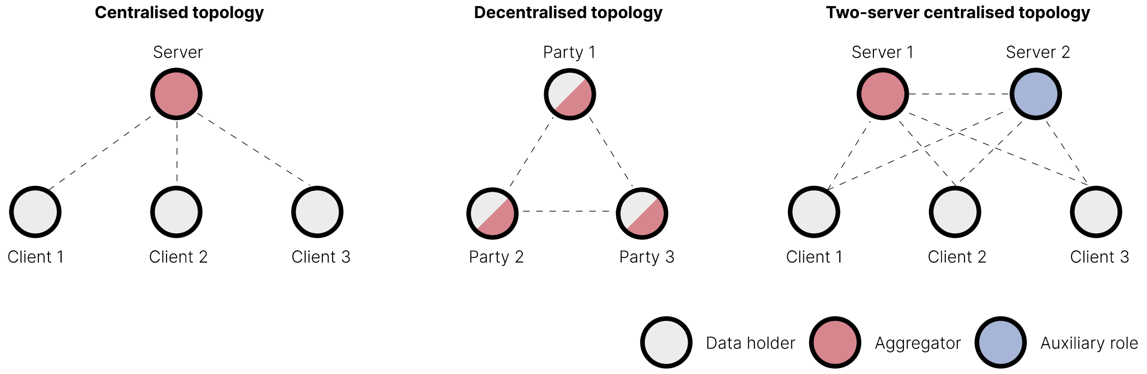


Figure 2: Different FL topologies. In centralised topologies, the data holders are typically referred to as clients, reflecting their interaction with a central server. In decentralised topologies, where no central entity exists, the participants are often called parties.

The parties' roles and how they are allowed to communicate between them define the topology of the FL consortium. Some examples are illustrated in Figure 2. The most common is the centralised topology, where multiple data-holding parties (the clients) collaboratively train a shared machine learning model through a central server (the aggregator) that iteratively collects model updates from each client, updates the global model and redistributes it back to the clients. Typically, clients do not communicate directly; they only communicate with the central server. In contrast, a decentralised topology (Beltran *et al.*, 2023) lacks a dedicated aggregation server. All consortium parties can potentially be model trainers and aggregators, interacting via peer-to-peer communication. Hybrid configurations include, for instance, using two servers: one server handles aggregation, while the other performs auxiliary tasks (Nasirigerdeh *et al.*, 2021). Clients can communicate with the servers, and servers can communicate between themselves, but clients cannot communicate with each other.

We will focus on the standard centralised topology and its two-server variant here because, to our knowledge, no bioinformatics applications use decentralised topologies.

2.2 Hardware and software

Hardware, software and models should be chosen with knowledge of the data and inputs from domain and machine learning specialists to design an effective machine learning pipeline (Scutari and Malvestio, 2023).

In terms of infrastructure, FL requires compute machines for each client and server. The optimal hardware configuration depends on the models to be trained; at a minimum, each client must be able to produce model updates from local data, and each server must be able to aggregate those updates and manage the consortium. Connection bandwidth is not necessarily critical: communications between clients and servers contain only a few megabytes of data, reaching 150MB only for large computer vision models, and can be made more compact through compression and model quantisation (Camajori Tedeschini *et al.*, 2022). On the other hand, latency may be a bottleneck if it limits the hardware utilisation.

As for software, several dedicated FL frameworks provide structured tools and environments for developing, deploying, and managing federated machine learning models (Riedel *et al.*, 2024). While some frameworks, such as Tensorflow Federated (TFF; Google, 2024), specialise in specific models, others support a broader range of approaches. Notable open-source examples include PySyft (Ziller *et al.*, 2021) and Flower (Beutel *et al.*, 2020). Both are supported by active communities and integrate with PyTorch to train complex models. PySyft is a multi-language library focusing on advanced privacy-preserving techniques, including differential privacy and homomorphic encryption. These features make it an excellent choice for sensitive tasks like genomic data analysis or secure sharing of proteomics data sets. Flower is an FL framework: its modular design and ease of customisation make it particularly useful for large-scale and multi-omics studies involving heterogeneous devices and clients. We will provide examples of their use in Section 3 before discussing frameworks designed explicitly for bioinformatics in Section 3.6.

Other frameworks target healthcare and biomedical applications but not bioinformatics specifically. For instance, OpenFL (Foley *et al.*, 2021) is designed to facilitate federated learning on sensitive EHRs and medical imaging data; it supports vertical FL and differential privacy in cross-silo FL but struggles with heterogeneous cross-device FL.

2.3 Usage scenarios: cross-device and cross-silo

FL applications take different forms in different domains. Many small, low-powered clients, such as wearable medical devices from the Internet of Things, may produce the data needed to train the federated machine learning model. Cross-device communications are often unreliable: passing lightweight model updates instead of raw data largely addresses connectivity issues and privacy risks.

FL may also involve a small number of parties, each possessing large amounts of sensitive data (Huang *et al.*, 2022), stored within their “data silos”. In this cross-silo scenario, common in healthcare and bioinformatics, the priority is to minimise the privacy risks associated with data sharing and comply with regulations. Minimising large data transfers is also computationally advantageous when modelling exceptionally large data, such as whole-genome sequences.

These two scenarios differ in how they handle model updates. In the cross-silo scenario, all (few) data holders in the consortium must participate in each update. In contrast, we can rely on a subset of (the many) data holders in the cross-device scenario because each holds a smaller share of the overall data. This article focuses on the cross-silo scenario, as nearly all bioinformatics applications fall within this framework.

2.4 Data partitioning and heterogeneity

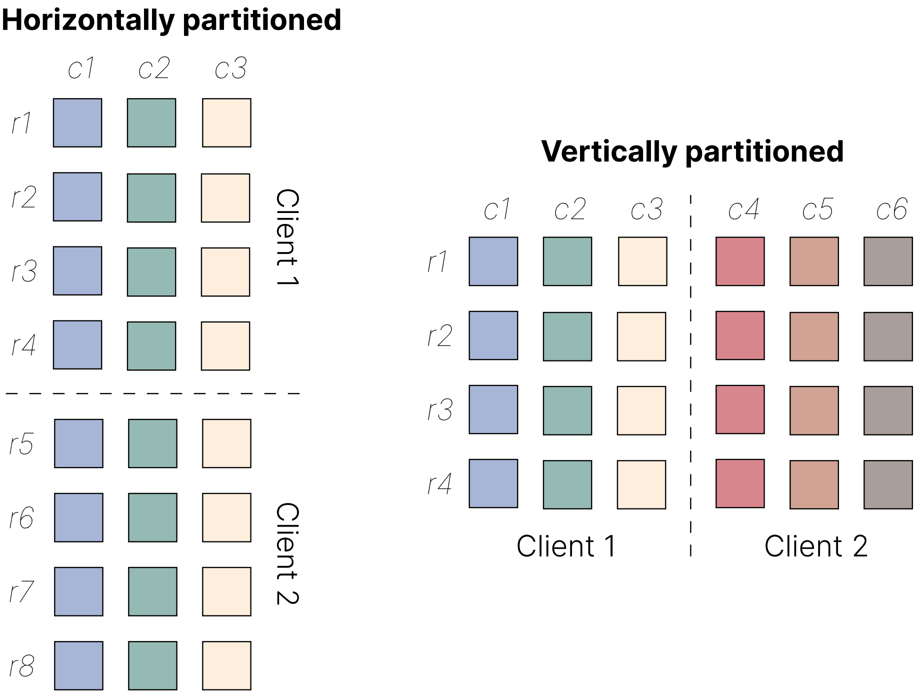


Figure 3: Illustration of horizontal and vertical FL data partitioning. In horizontal FL (left), clients hold data sets with the same features (c1-c3) but different subsets of samples (r1–r8). In vertical FL (right), clients hold data sets with different features (c1–c6) but the same set of samples (r1–r4).

Data may be partitioned along variable or observation lines: each party may record the same features for different individuals or different features describing the same individuals (Figure 3). In the first scenario, known as horizontal FL,[1] different parties may each possess genomic sequencing data from different individuals. In contrast, in vertical FL, one party may hold data from one omic type (say, genomic data), while another may have data from a different phenotype or omic type (say, proteomic data) for the same individuals. Horizontal FL is by far the most prevalent approach in bioinformatics.

Significant variations in sample size and feature distributions across data holders often exist. This heterogeneity allows FL to better capture the variability of the underlying population, resulting in transferrable models that generalise well (Sheller *et al.*, 2020). Clearly, if data holders collect observations from distinct populations, any federated model trained from them must be correctly specified to capture population structure and avoid bias in inference and prediction. If the populations are known, we can train targeted population-specific models alongside the global one (Tan *et al.*, 2022).

[1] This naming convention assumes that observations (features) are the rows (columns) of a tabular data set.

Otherwise, we can use clustering to identify them from the available data (Sattler *et al.*, 2021). Harmonising data across parties is fundamental to account for variations in measurements, definitions and distributions and is much more challenging than in meta-analysis because access to data is restricted (Camajori Tedeschini *et al.*, 2022).

2.5 Security and privacy

FL reduces privacy and security risks by design because it passes model updates between parties. However, it does not eliminate them completely.

In terms of privacy, deep learning models are the most problematic in machine learning because of their ability to memorise training data. They leak individual observations during training (through model updates; Geiping *et al.*, 2020), after training (through their parameters; Haim *et al.*, 2022) and during inference (membership attacks; Shokri *et al.*, 2017; Hu *et al.*, 2022). However, individual reidentification is a general issue for genetic data (Homer *et al.*, 2008) and all the models learned from them. For instance, (Cai *et al.*, 2015) has demonstrated that it is possible to identify an individual from the linear model learned in an association study from just 25 genes. Even basic infrastructure security measures and the distributed nature of the data make such identification difficult under the best circumstances. The privacy-enhancing techniques discussed in Section 2.6 can make such efforts wholly impractical.

As for security, we must consider different threat models, understanding what information requires protection, their vulnerabilities, and how to mitigate or respond to threats. Internal and external threats to the consortium should be treated equally with security in depth design and implementation decisions that consider parties untrusted. Security threats, such as membership attacks and model inversion attacks (Fredrikson *et al.*, 2015), can originate equally from parties and external adversaries that seek to abuse the model inference capabilities to extract information about the data. On the other hand, adversarial attacks are more likely to originate from consortium parties that seek to introduce carefully crafted data or model updates into the training process to produce a global model with undesirable behaviours. Some examples are data poisoning (Sun *et al.*, 2022), manipulation (Jagielski *et al.*, 2018) and Byzantine attacks (Li *et al.*, 2023a).

Encrypting communication channels, implementing strict authentication (to verify each party’s identity) and authorisation (to control which information and resources each party has access to or shares) schemes, and keeping comprehensive access logs for audit can secure any machine learning pipeline, including federated ones. Similarly, using an experiment tracking platform makes it possible to track data provenance, audit both the data and the training process and ensure the reproducibility of results (Scutari and Malvestio, 2023). These measures must be complemented by federated models resistant to these threats at training and inference time, as thoroughly discussed in Yin *et al.* (2021).

2.6 Privacy-enhancing techniques

Privacy-enhancing techniques improve the confidentiality of sensitive information during training. We summarise the most relevant below, illustrating them with a simple example in Figure 4.

Homomorphic encryption (HE; Gentry, 2009) is a cryptographic technique that enables computations to be performed directly on encrypted data (ciphertexts) without requiring decryption. The outcome of operations on ciphertexts matches the result of performing the same operations on the corresponding non-encrypted values (plaintexts) when decrypted. HE can be either fully homomorphic (FHE), which allows for arbitrary computations, and partially homomorphic (PHE), which supports only a specific subset of mathematical operations. For instance, the Paillier PHE scheme (Paillier, 1999) only supports additive operations on encrypted data. FHE requires considerable computational resources for encryption and decryption. PHE is less flexible but computationally more efficient, making it a common choice in practical applications.

Differential privacy (DP; Ficek *et al.*, 2021) is a mathematical framework designed to enable analyses to remain statistically consistent regardless of whether any specific individual's data is included or excluded. This property guarantees that sensitive information about individuals cannot be inferred from the results up to a preset "privacy budget" worth of operations. DP is typically implemented by introducing noise into the data (Schein *et al.*, 2019; Cai *et al.*, 2021), weight clipping in the training process (Abadi *et al.*, 2016; Jayaraman and Evans, 2019) or predictions (Nissim *et al.*, 2007; Dwork and Feldman, 2018) to obfuscate individual contributions. The amount of noise must be carefully calibrated to balance predictive accuracy and privacy within the analysis: too little noise undermines privacy, and too much reduces performance. This effect is more pronounced within specific subgroups underrepresented in the training set (Bagdasaryan *et al.*, 2019).

Secure multiparty computation (SMPC; Zhao *et al.*, 2019) is a peer-to-peer protocol allowing multiple parties to compute a function over their data collaboratively, similarly to Figure 2 (centre). Each data holder divides their data into random shares and distributes them among all parties in the consortium, thus ensuring that no single party can access the complete data set. The shares are then combined during the computation process, often with the assistance of a server, to produce the correct result while preserving data privacy. SMPC ensures high security with exact results and keeps data private throughout the computation process. However, SMPC is computationally intensive and requires peer-to-peer communication, leading to high communication overhead. Additionally, the complexity of the protocol increases with the number of participants, limiting scalability.

Another approach to securing FL is using an aggregator and a compensator server in a centralised two-server topology (Figure 2, right; Nasirigerdeh *et al.*, 2021). Each client adds a noise pattern to their local data, sharing the former with the compensator (which aggregates all noise patterns) and the latter with the aggregator (which aggregates the noisy data and trains the model). The aggregator then obtains the overall noise pattern from the compensator and removes it from the aggregated noisy data, allowing for denoised model training. This two-server approach is efficient: it requires neither extensive computation in the clients nor peer-to-peer communication. However, using two servers makes infrastructure more complex and requires trust in both servers not to collude to compromise the privacy of individual contributions.

3 Federated Learning in bioinformatics

Most FL literature focuses on general algorithms and is motivated by applications other than bioinformatics, such as digital twins for smart cities (Ramu *et al.*, 2022), smart industry (Zhang *et al.*, 2021) and open banking and finance (Long *et al.*, 2020). Even the clinical literature mainly focuses on different types of data and issues (Dayan *et al.*, 2021; van Rooden *et al.*, 2024). Here, we highlight and discuss notable examples of FL designed specifically for bioinformatics. They are all in the early stages of development, so their reliability, reproducibility, and scalability are open questions. However, they hint at the potential of FL to perform better than meta- analysis and single-client analyses on real-world data, comparing favourably to centralised data analyses in which data are pooled in a central location while addressing data sharing and use concerns.

3.1 Proteomics and differential gene expression

Proteomics studies the complex protein dynamics that govern cellular processes and their interplay with physiological and pathological states, such as cancer (Maes *et al.*, 2015), to improve risk assessment, early detection, diagnosis, prognosis, treatment selection, and patient monitoring. Differential expression analyses focus specifically on comparing expression levels across different conditions, tissues, or cell types to identify genes with statistically significant differences (Rodriguez-Esteban and Jiang, 2017).

In addition to the issues discussed in Section 2, FL in proteomics must overcome the challenge of integrating data from different platforms (Rieke *et al.*, 2020) while accounting for imbalanced samples and batch effects. Cai *et al.* (2022) produced a federated implementation of DEqMS (FedProt; Zhu *et al.*, 2020) for variance estimation in mass spectrometry-based data that successfully identifies top differentially-abundant proteins in two real-world data sets using label-free quantification and tandem mass tags.

Zolotareva *et al.* (2021) implemented a federated limma voom pipeline (Law *et al.*, 2014) on top of HyFed (Nasirigerdeh *et al.*, 2021), which uses the aggregator-compensator two-server topology we described earlier. This approach was showcased on two extensive RNA-seq datasets, proving robust to heterogeneity across clients and batch effects. Hannemann *et al.* (2024) trained a federated deep-learning model for cell type classification using both Flower and TFF and different architectures, with similar results.

3.2 Genome wide association studies

Genome-wide association studies (GWAS) aim to identify genomic variants statistically associated with a qualitative (say, a case-control label) or quantitative trait (say, body mass index). These studies mainly use regression models, which can be largely trained using general-purpose federated regression implementations with little modifications to address scalability and correct for population structure.

Li *et al.* (2022) has developed the most complete adaptation of these models to federated GWAS in the literature: it provides linear and logistic regressions with fixed and random effects and accounts for population structure via a genomic relatedness matrix. Wang *et al.* (2022) further provides a federated estimator for the genomic relatedness matrix. Finally, Li *et al.* (2024) describes the federated association tests for the genomic variants associated with this model. All these steps incorporate HE to ensure privacy in the GWAS.

As an alternative, Cho *et al.* (2024) build on REGENIE (Mbatchou *et al.*, 2021) to avoid using a genomic relatedness matrix and increase the scalability of GWAS while using MPC and HE to secure the data. Despite the overhead introduced by the encryption, this approach is efficient enough to work on a cohort of 401k individuals from the UK Biobank and 90 million SNPs in less than 5 hours.

3.3 Single-cell RNA sequencing

Single-cell RNA sequencing (scRNA-seq) measures gene expression at the cellular level rather than aggregating it at the tissue level as in bulk RNA sequencing, identifying the distinct expression profiles of individual cell populations within tissues (Hwang *et al.*, 2018; Papalexi and Satija, 2018).

Wang *et al.* (2024) developed scFed, a unified FL framework integrating four algorithms for cell type classification from scRNA-seq data: the ACTINN neural network (Ma and Pellegrini, 2020), explicitly designed for this task; a linear support vector machine; XGBoost based on Li *et al.* (2023b); and the GeneFormer transformer (Theodoris *et al.*, 2023). They evaluated scFed on eight data sets (human and non-human) evenly distributed among 2–20 clients, suggesting that the federated approach has a predictive accuracy comparable to that obtained by pooling the data and better than that in individual clients. However, the overhead during training increases with the number of clients, limiting the scalability to larger consortia.

3.4 Multi-omics

Proteomics, genomics, and transcriptomics capture different aspects of biological processes. Integrating large data sets from different omics offers deeper insights into their underlying mechanisms (Civelek and Lusis, 2014). Vertical FL allows multiple parties to combine various features of the same patients into multimodal omics data sets without exposing sensitive information (Liu *et al.*, 2024). For instance, Wang *et al.* (2023) trained a deep neural network with an adaptive optimisation module for cancer prognosis evaluation from multi-omics data. The neural network performs feature selection while the adaptive optimisation module prevents overfitting, a common issue in small high-dimensional samples (Rajput *et al.*, 2023). This method performs better than a single-omic analysis, but the improvement in predictive accuracy is strongly model-dependent.

Another example is Danek *et al.* (2024), who built a diagnostic model for Parkinson's disease: they provided a reproducible setup for federated multi-omics model training based on Flower. Despite using pre-processed, harmonised data, they showed that multi-omics data heterogeneity between clients brings only marginal performance improvements while evaluating several FL models for Parkinson's disease prediction.

3.5 Medical imaging

Medical imaging studies the human body's interior to diagnose abnormalities in its anatomy and physiology from digital images such as those obtained by radiography, magnetic resonance and ultrasound devices (Suetens, 2017). As a computer vision application, it must overcome challenges such as incomplete or inaccurate labelling and the normalisation of images from different scanners and different protocols.

However, medical imaging is the most common application of FL in the medical literature. As a result, protocols for image segmentation and diagnostic prediction are well documented in the literature (Chowdhury *et al.*, 2022). Notable case studies in the literature target breast cancer (Roth *et al.*, 2020), melanomas (Haggenmuller *et al.*, 2024), cardiovascular disease (Linardos *et al.*, 2022), COVID-19 (Yang *et al.*, 2021; Dayan *et al.*, 2021). Bdair *et al.* (2022) explored a federated labelling scheme in which clients produced ground-truth labels for skin lesions in a privacy-preserving manner, improving classification accuracy. Yan *et al.* (2023) also proposed an efficient scheme to use data sets mainly comprising unlabelled images, focusing on chest X-rays. Furthermore, Jiang *et al.* (2022) apply FL to learn a harmonised feature set from heterogeneous medical images that improves both classification and segmentation of histology and MRI scans.

3.6 Ready-to-use FL tools for bioinformatics

The need for user-friendly FL implementations of common bioinformatics workflows has driven the creation of secure collaborative analysis tools (Berger and Cho, 2019; Froelicher *et al.*, 2021; Wan *et al.*, 2022). Two notable examples are sfkit and FeatureCloud.

The sfkit framework (Mendelsohn *et al.*, 2023) facilitates federated genomic analyses by implementing GWAS, principal component analysis (PCA), genetic relatedness and a modular architecture to complement them as needed. It provides a web interface featuring a project bulletin board, chat functions, study parameter configurations and results sharing. State-of-the-art cryptographic tools for privacy preservation based on SMPC and HE ensure data protection.

FeatureCloud (Matschinske *et al.*, 2023) is an integrated solution from which end users without programming experience can build custom workflows. It provides modules to run on the clients and servers in the consortium. Unlike sfkit, FeatureCloud allows developers to easily implement and publish applications in its app store, including regression models, random forests and neural networks. Developers must also document how privacy guarantees are implemented in their apps.

4 How to conduct federation of specific operations in bioinformatics

This section provides practical insights to help a reader interested in building a federated and secure analogue of an existing bioinformatics algorithm. We focus on horizontal FL with the centralised topology from Figure 2 (left). Consider K different clients, each possessing a local data set $X_k, k=1,\ldots,K$ with n_k observations x_{ij}^k (for different individuals) and P features stored in columns (the same for all individuals). We denote a row (column) of the matrix X^k with x_{i*}^k (x_{*j}^k). This describes a distributed dataset of $N=\sum_{k=1}^{K} n^k$ observations:

$$X=\begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_k \end{bmatrix}.$$

The following sections assume that an FL consortium has been established, the necessary infrastructure is operational, and an appropriate FL framework has been selected and installed. It is also assumed that a secure sum protocol has been chosen, such as those described in Section 2.6 and Figure 4. The choice of a specific secure sum protocol may depend on several factors, including IT considerations (e.g., the availability of a specific FL topology that drives the choice), privacy risks, or scalability concerns, as discussed in Section 2.6. In the following sections, we provide a general description of operations based on a secure sum and operations based on federated averaging (FedAvg; McMahan *et al.*, 2017).

We chose Flower to explore these examples because it has a shallow learning curve for new FL users and provides a good balance between simplicity and flexibility when implementing custom algorithms. Riedel *et al.* (2024) also identified it as a promising framework because it has a large, active, and growing community of developers and scientists and because of the extensive tutorials and documentation. The secure sum protocol used in the examples is SecAgg+. This protocol combines encryption with SMPC and is particularly suitable for FL, as it provides high scalability and is robust to client dropouts. Scalability is ensured because, within the multiparty approach, each client interacts with only a fraction of the other clients. The algorithm scales linearly with the number of clients each client interacts with and linearly with the size of the vectors to be aggregated (Li *et al.*, 2021).

4.1 Sum-based computations

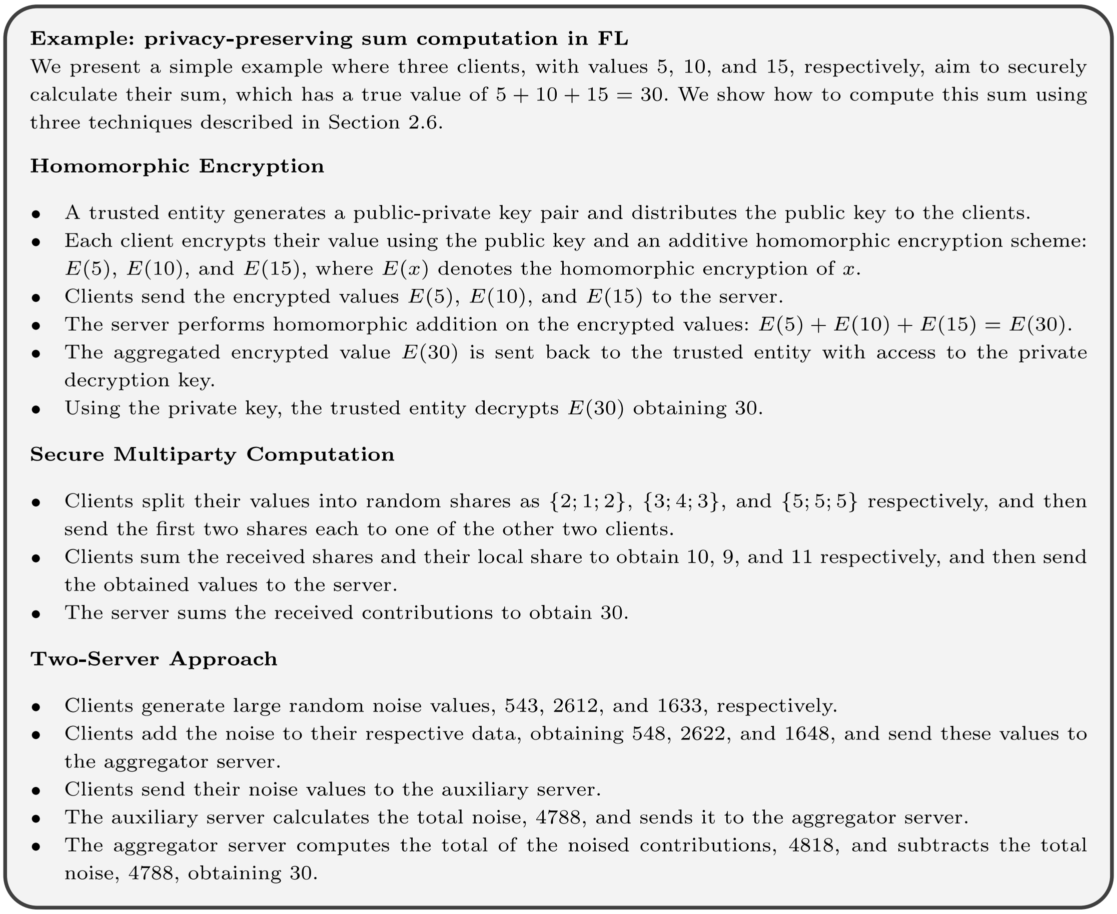
Let a^k be real numbers stored by individual clients. We define the secure sum of these numbers, performed through the selected secure sum protocol, as $\bigoplus_{k=1}^{K} n^k$. We can build on this simple sum to construct a wide range of operations. However, note that as the complexity of operations increases, the amount of information revealed to the server may also increase. Sum-based operations include:

- The overall sample size of the distributed data set as $N=\bigoplus_{k=1}^{K} n^k$ from the local sample sizes n^k.
- The mean value of the j-th feature, given N, as $M_j=\bigoplus_{k=1}^{K}\left[\sum_{i=1}^{n_k} x_{ij}^k\right]$. Each client computes the inner sum on their local data, whereas the outer one is a secure sum aggregated across clients by the server.
- The variance of the j-th feature, given N and M_j, as $V_j=\frac{1}{N-1}\bigoplus_{k=1}^{K}\left[\sum_{i=1}^{n_k}\left(x_{ij}^k-M_j\right)^2\right]$, which can be used it to standardise the j-th feature as $\left(x_{*\,j}^k-M_j\right)/\sqrt{\left(V_j\right)}$.
- The Pearson correlation coefficient of two features j and j', given M_j and $M_{j'}$, as $\rho_{j,j'}=\frac{\frac{1}{N-1}\bigoplus_{k=1}^{K}\sum_{i=1}^{n_k}\left(x_{ij}^k-M_j\right)\left(x_{ij'}^k-M_{j'}\right)}{\sqrt{V_j V_{j'}}}$.
- The matrix $X^T X$, as $X^T X=\bigoplus_{k=1}^{K}\left(X^k\right)^T X^k$, where ⊕ is a secure element-wise sum. For standardised datasets, this matrix is equivalent to the covariance matrix and is commonly used for PCA.

Beyond these general-purpose examples, many operations specific to bioinformatics pipelines also rely on simple sums. These operations are often straightforward generalisations or compositions of the examples introduced above.

For example, in differential gene expression studies, it may be useful to filter out weakly expressed genes. Weakly expressed genes can be defined as those whose expression values fall below a specified threshold t in, for instance, 70% of the samples. Let v^k be a vector belonging to client k, where each vector component represents the number of samples in which the expression level of the gene (e.g., counts) exceeds the threshold t. The server can securely calculate $v=\frac{1}{N}\bigoplus_{k=1}^{K} v^k$ and identify weakly expressed genes as those whose corresponding components of v are smaller than 0.7.

A fundamental preliminary step in a GWAS is identifying the minor allele and its frequency. Let a^k , c^k , g^k , t^k be vectors belonging to client k, where each component corresponds to a specific SNP. The components of a^k , c^k , g^k , t^k represent the number of samples in which alleles A, C, G, T are observed, respectively. The server can securely compute the aggregated allele counts across all clients as $a=\frac{1}{N}\bigoplus_{k=1}^{K} a^k$ and similarly c, g, t (where t can also be computed by difference from N and the other three vectors). For each SNP, the minor allele is determined by comparing the corresponding components of a, c, g, t: the allele with the smaller value is designated as the minor allele. This operation is crucial because the minor allele within a single client's population may differ from the minor allele when considering the whole distributed dataset. Ensuring a consistent definition of the minor allele across all clients is essential for reliable downstream analyses.



Example: privacy-preserving sum computation in FL

We present a simple example where three clients, with values 5, 10, and 15, respectively, aim to securely calculate their sum, which has a true value of $5 + 10 + 15 = 30$. We show how to compute this sum using three techniques described in Section 2.6.

Homomorphic Encryption

- A trusted entity generates a public-private key pair and distributes the public key to the clients.
- Each client encrypts their value using the public key and an additive homomorphic encryption scheme: $E(5)$, $E(10)$, and $E(15)$, where $E(x)$ denotes the homomorphic encryption of x.
- Clients send the encrypted values $E(5)$, $E(10)$, and $E(15)$ to the server.
- The server performs homomorphic addition on the encrypted values: $E(5) + E(10) + E(15) = E(30)$.
- The aggregated encrypted value $E(30)$ is sent back to the trusted entity with access to the private decryption key.
- Using the private key, the trusted entity decrypts $E(30)$ obtaining 30.

Secure Multiparty Computation

- Clients split their values into random shares as $\{2; 1; 2\}$, $\{3; 4; 3\}$, and $\{5; 5; 5\}$ respectively, and then send the first two shares each to one of the other two clients.
- Clients sum the received shares and their local share to obtain 10, 9, and 11 respectively, and then send the obtained values to the server.
- The server sums the received contributions to obtain 30.

Two-Server Approach

- Clients generate large random noise values, 543, 2612, and 1633, respectively.
- Clients add the noise to their respective data, obtaining 548, 2622, and 1648, and send these values to the aggregator server.
- Clients send their noise values to the auxiliary server.
- The auxiliary server calculates the total noise, 4788, and sends it to the aggregator server.
- The aggregator server computes the total of the noised contributions, 4818, and subtracts the total noise, 4788, obtaining 30.

Figure 4: Example of privacy-preserving sum computationin FL using three different techniques. Note that althoughdifferential privacy is described in Section 2.6, it is not included in this example, as it would not be suitable for such a calculation.

4.2 Federated averaging computations

FedAvg is a widely used algorithm for training deep neural networks in FL. It iteratively computes a weighted average of model parameters across clients, with weights proportional to the local sample sizes n^k. Thus, it can be applied to any parametric model, including linear models.

FedAvg proceeds as illustrated in Figure 4. The server first broadcasts an initial global model with parameters w_0. At each step of the algorithm, clients start with the global model wt and perform local updates to produce updated local models w_{t+1}^k. The global model is updated after each round of local training as the weighted sum of the local models:

$$w_{t+1} = \bigoplus_{k=1}^{K} \frac{n^k}{N} w_{t+1}^k.$$

where we use the secure sum ⊕ for aggregation (FedAvg is itself a sum-based operation). After aggregation, the updated global model is distributed back to the clients.

However, many bioinformatics pipelines rely on linear models rather than deep learning models. One commonly used model is logistic regression, which is applied in tasks such as differential gene expression analysis and GWAS. A federated implementation of logistic regression can be achieved by starting with a standard implementation and applying FedAvg, which aggregates the local models after a specified number of iterations performed by the local logistic regressions.

5 References

M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang. Deep Learning with Differential Privacy. In Proceedings of the 2016 ACM Sigsac Conference on Computer and Communications Security, pages 308–318, 2016.

E. Bagdasaryan, O. Poursaeed, and V. Shmatikov. Differential Privacy Has Disparate Impact on Model Accuracy. Advances in Neural Information Processing Systems, 32:15479–15488, 2019.

T. Bdair, N. Navab, and S. Albarqouni. Semi-Supervised Federated Peer Learning for Skin Lesion Classification. Machine Learning for Biomedical Imaging, 1(April 2022):1–37, 2022.

E. T. M. Beltran, M. Q. Perez, P. M. S. Sanchez, S. L. Bernal, G. Bovet, M. G. P´erez, G. M. Perez, and A. H. Celdran. Decentralized Federated Learning: Fundamentals, State of the Art, Frameworks, Trends, and Challenges. IEEE Communications Surveys & Tutorials, 5(4):2983–3013, 2023.

B. Berger and H. Cho. Emerging Technologies Towards Enhancing Privacy in Genomic Data Sharing. Genome Biology, 20:128, 2019.

D. J. Beutel, T. Topal, A. Mathur, X. Qiu, J. Fernandez-Marques, Y. Gao, L. Sani, K. H. Li, T. Parcollet, P. P. B. de Gusm˜ao, *et al.* Flower: A Friendly Federated Learning Research Framework, 2020. arXiv preprint arXiv:2007.14390.

T. S. Brisimi, R. Chen, T. Mela, A. Olshevsky, I. C. Paschalidis, and W. Shi. Federated Learning of Predictive Models From Federated Electronic Health Records. International Journal of Medical Informatics, 112:59–67, 2018.

K. Cai, X. Lei, J. Wei, and X. Xiao. Data Synthesis via Differentially Private Markov Random Fields. Proceedings of the Vldb Endowment, 14(11):2190–2202, 2021.

R. Cai, Z. Hao, M. Winslett, X. Xiao, Y. Yang, Z. Zhang, and S. Zhou. Deterministic Identification of Specific Individuals From Gwas Results. Bioinformatics, 31(11):1701–1707, 2015.

Z. Cai, R. C. Poulos, J. Liu, and Q. Zhong. Machine Learning for Multi-Omics Data Integration in Cancer. Iscience, 25(2), 2022.

B. Camajori Tedeschini, S. Savazzi, R. Stoklasa, L. Barbieri, I. Stathopoulos, M. Nicoli, and L. Serio. Decentralized Federated Learning for Healthcare Networks: A Case Study on Tumor Segmentation. IEEE Access, 10:8693–8708, 2022.

C. Cath, S. Wachter, B. Mittelstadt, M. Taddeo, and L. Floridi. Artificial Intelligence and the ‘Good Society’: the US, EU, and UK Approach. Science and Engineering Ethics, 24(2):505–528, 2018.

H. Cho, D. Froelicher, J. Chen, M. Edupalli, A. Pyrgelis, J. R. Troncoso-Pastoriza, J. Hubaux, and B. Berger. Secure and Federated Genome-Wide Association Studies for Biobank-Scale Datasets, 2024. bioRXiv preprint 10.1101/2022.11.30.518537v2.

A. Chowdhury, H. Kassem, N. Padoy, R. Umeton, and A. Karargyris. A Review of Medical Federated Learning: Applications in Oncology and Cancer Research. In Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 7th International Workshop, BrainLes 2021, 24th MICCAI Conference, pages 3–24, 2022.

M. Civelek and A. J. Lusis. Systems Genetics Approaches to Understand Complex Traits. Nature Reviews Genetics, 15(1):34–48, 2014.

B. P. Danek, M. B. Makarious, A. Dadu, D. Vitale, P. S. Lee, A. B. Singleton, M. A. Nalls, J. Sun, and F. Faghri. Federated Learning for Multi-Omics: A Performance Evaluation in Parkinson's Disease. Patterns, 5(3):100945, 2024.

I. Dayan, H. R. Roth, A. Zhong, A. Harouni, A. Gentili, A. Z. Abidin, A. Liu, A. B. Costa, B. J. Wood, C. Tsai, C. Wang, C. Hsu, C. K. Lee, P. Ruan, D. Xu, D. Wu, E. Huang, F. C. Kitamura, G. Lacey, G. C. de Antonio Corradi, G. Nino, H. Shin, H. Obinata, H. Ren, J. C. Crane, J. Tetreault, J. Guan, J. W. Garrett, J. D. Kaggie, J. G. Park, K. Dreyer, K. Juluru, K. Kersten, M. A. B. C. Rockenbach, M. G. Linguraru, M. A. Haider, M. AbdelMaseeh, N. Rieke, P. F. Damasceno, P. M. C. e Silva, P. Wang, S. Xu, S. Kawano, S. Sriswasdi, S. Y. Park, T. M. Grist, V. Buch, W. Jantarabenjakul, W. Wang, W. Y. Tak, X. Li, X. Lin, Y. J. Kwon, A. Quraini, A. Feng, A. N. Priest, B. Turkbey, B. Glicksberg, B. Bizzo, B. S. Kim, C. Tor-Diez, C. Lee, C. Hsu, C. Lin, C. Lai, C. P. Hess, C. Compas, D. Bhatia, E. K. Oermann, E. Leibovitz, H. Sasaki, H. Mori, I. Yang, J. H. Sohn, K. N. K. Murthy, L. Fu, M. R. F. de Mendonca, M. Fralick, M. K. Kang, M. Adil, N. Gangai, P. Vateekul, P. Elnajjar, S. Hickman, S. Majumdar, S. L. McLeod, S. Reed, S. Graf, S. Harmon, T. Kodama, T. Puthanakit, T. Mazzulli, V. L. de Lavor, Y. Rakvongthai, Y. R. Lee, Y. Wen, F. J. Gilbert, M. G. Flores, and Q. Li. Federated Learning for Predicting Clinical Outcomes in Patients with COVID-19. Nature Medicine, 27(10):1735–1743, 2021.

C. Dwork and V. Feldman. Privacy-preserving Prediction. In Proceedings of the 31st Conference On Learning Theory, pages 1693–1702, 2018.

European Union. General Data Protection Regulation (GDPR). Official Journal of the European Union, document 32016R0679, 2016.

Federated European Genome-phenome Archive. Federated European Genome-Phenome Archive (FEGA), 2024. URL https://ega-archive.org/ federated. Accessed: 2024-10-30.

J. Ficek, W. Wang, H. Chen, G. Dagne, and E. Daley. Differential Privacy in Health Research: A Scoping Review. Journal of the American Medical Informatics Association, 28(10):2269–2276, 2021.

C. N. Foley, A. M. Mason, P. D. Kirk, and S. Burgess. Mr-Clust: Clustering of Genetic Variants in Mendelian Randomization with Similar Causal Estimates. Bioinformatics, 37(4):531–541, 2021.

M. Fredrikson, S. Jha, and T. Ristenpart. Model Inversion Attacks That Exploit Confidence Information and Basic Countermeasures. In Proceedings of the 22nd ACM Sigsac Conference on Computer and Communications Security, pages 1322–1333, 2015.

D. Froelicher, J. R. Troncoso-Pastoriza, J. L. Raisaro, M. A. Cuendet, J. S. Sousa, H. Cho, B. Berger, J. Fellay, and J. Hubaux. Truly Privacy-Preserving Federated Analytics for Precision Medicine with Multiparty Homomorphic Encryption. Nature Communications, 12(1):5910, 2021.

J. Geiping, H. Bauermeister, H. Droge, and M. Moeller. Inverting Gradients-How Easy Is It to Break Privacy in Federated Learning? Advances in Neural Information Processing Systems, 33:16937–16947, 2020.

C. Gentry. Fully Homomorphic Encryption Using Ideal Lattices. In Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing, pages 169–178, 2009.

Google. TensorFlow Federated: Machine Learning on Decentralized Data, 2024. URL https://www.tensorflow.org/federated.

S. Haggenmuller, M. Schmitt, E. Krieghoff-Henning, A. Hekler, R. C. Maron, C. Wies, J. S. Utikal, F. Meier, S. Hobelsberger, F. F. Gellrich, M. Sergon, A. Hauschild, L. E. French, L. Heinzerling, J. G. Schlager, K. Ghoreschi, M. Schlaak, F. J. Hilke, G. Poch, S. Korsing, C. Berking, M. V. Heppt, M. Erdmann, S. Haferkamp, K. Drexler, D. Schadendorf,W.

Sondermann, M. Goebeler, B. Schilling, J. N. Kather, S. Frohling, and T. J. Brinker. Federated Learning for Decentralized Artificial Intelligence in Melanoma Diagnostics. JAMA Dermatology, 160 (3):303, 2024.

N. Haim, G. Vardi, G. Yehudai, O. Shamir, and M. Irani. Reconstructing Training Data From Trained Neural Networks. Advances in Neural Information Processing Systems, 35:22911–22924, 2022.

A. Hannemann, J. Ewald, L. Seeger, and E. Buchmann. Federated Learning on Transcriptomic Data: Model Quality and Performance Trade-Offs. In International Conference on Computational Science, pages 279–293, 2024.

W. Heyndrickx, L. Mervin, T. Morawietz, N. Sturm, L. Friedrich, A. Zalewski, A. Pentina, L. Humbeck, M. Oldenhof, R. Niwayama, P. Schmidtke, N. Fechner, J. Simm, A. Arany, N. Drizard, R. Jabal, A. Afanasyeva, R. Loeb, S. Verma, S. Harnqvist, M. Holmes, B. Pejo, M. Telenczuk, N. Holway, A. Dieckmann, N. Rieke, F. Zumsande, D. Clevert, M. Krug, C. Luscombe, D. Green, P. Ertl, P. Antal, D. Marcus, N. Do Huu, H. Fuji, S. Pickett, G. Acs, E. Boniface, B. Beck, Y. Sun, A. Gohier, F. Rippmann, O. Engkvist, A. H. Goller, Y. Moreau, M. N. Galtier, A. Schuffenhauer, and H. Ceulemans. MELLODDY: Cross-Pharma Federated Learning at Unprecedented Scale Unlocks Benefits in Qsar Without Compromising Proprietary Information. Journal of Chemical Information and Modeling, 64(7):2331–2344, 2023.

N. Homer, S. Szelinger, M. Redman, D. Duggan, W. Tembe, J. Muehling, J. V. Pearson, D. A. Stephan, S. F. Nelson, and D. W. Craig. Resolving Individuals Contributing Trace Amounts of DNA to Highly Complex Mixtures Using High-Density SNP Genotyping Microarrays. PLoS Genetics, 4(8):e1000167, 2008.

H. Hu, Z. Salcic, L. Sun, G. Dobbie, P. S. Yu, and X. Zhang. Membership Inference Attacks on Machine Learning: A Survey. ACM Computing Surveys, 54(11s):1–37, 2022.

C. Huang, J. Huang, and X. Liu. Cross-Silo Federated Learning: Challenges and Opportunities, 2022. arXiv preprint arXiv:2206.12949.

B. Hwang, J. H. Lee, and D. Bang. Single-Cell RNA Sequencing Technologies and Bioinformatics Pipelines. Experimental & Molecular Medicine, 50(8):1–14, 2018.

M. Jagielski, A. Oprea, B. Biggio, C. Liu, C. Nita-Rotaru, and B. Li. Manipulating Machine Learning: Poisoning Attacks and Countermeasures for Regression Learning. In 2018 IEEE Symposium on Security and Privacy (Sp), pages 19–35, 2018.

B. Jayaraman and D. Evans. Evaluating Differentially Private Machine Learning in Practice. In 28th Usenix Security Symposium (Usenix Security 19), pages 1895–1912, 2019.

M. Jiang, Z. Wang, and Q. Dou. Harmofl: Harmonizing Local and Global Drifts in Federated Learning on Heterogeneous Medical Images. Proceedings of the AAAI Conference on Artificial Intelligence, 36(1):1087–1095, 2022.

S. Kavianpour, J. Sutherland, E. Mansouri-Benssassi, N. Coull, and E. Jefferson. Next-Generation Capabilities in Trusted Research Environments: Interview Study. Journal of Medical Internet Research, 24(9):e33720, 2022.

C. W. Law, Y. Chen, W. Shi, and G. K. Smyth. Voom: Precision Weights Unlock Linear Model Analysis Tools for Rna-Seq Read Counts. Genome Biology, 15(2):1–17, 2014.

B. Li, P. Wang, Z. Shao, A. Liu, Y. Jiang, and Y. Li. Defending Byzantine Attacks in Ensemble Federated Learning: A Reputation-Based Phishing Approach. Future Generation Computer Systems, 147:136–148, 2023a.

K. H. Li, P. P. B. de Gusmao, D. J. Beutel, and N. D. Lane. Secure aggregation for federated learning in flower. In Proceedings of the 2nd ACM International Workshop on Distributed Machine Learning, pages 8–14, 2021.

Q. Li, Z. Wu, Y. Cai, Y. Han, C. M. Yung, T. Fu, and B. He. Fedtree: A Federated Learning System for Trees. In Proceedings of Machine Learning and Systems, 2023b.

W. Li, J. Tong, M. M. Anjum, N. Mohammed, Y. Chen, and X. Jiang. Federated Learning Algorithms for Generalized Mixed-Effects Model (GLMM) on Horizontally Partitioned Data From Distributed Sources. BMC Medical Informatics and Decision Making, 22(1), 2022.

W. Li, H. Chen, X. Jiang, and A. Harmanci. FedGMMAT: Federated Generalized Linear Mixed Model Association Tests. PLoS Computational Biology, 20(7):e1012142, 2024.

A. Linardos, K. Kushibar, S. Walsh, P. Gkontra, and K. Lekadir. Federated Learning for Multi-Center Imaging Diagnostics: A Simulation Study in Cardiovascular Disease. Scientific Reports, 12(1):3551, 2022.

Y. Liu, Y. Kang, T. Zou, Y. Pu, Y. He, X. Ye, Y. Ouyang, Y. Zhang, and Q. Yang. Vertical Federated Learning: Concepts, Advances, and Challenges. IEEE Transactions on Knowledge and Data Engineering, 36(7):3615–3634, 2024.

G. Long, Y. Tan, J. Jiang, and C. Zhang. Federated Learning for Open Banking, pages 240–254. Springer, 2020.

H. Ludwig and N. Baracaldo. Federated Learning: A Comprehensive Overview of Methods and Applications. Springer, 2022.

F. Ma and M. Pellegrini. Actinn: Automated Identification of Cell Types in Single Cell RNA Sequencing. Bioinformatics, 36(2):533–538, 2020.

E. Maes, I. Mertens, D. Valkenborg, P. Pauwels, C. Rolfo, and G. Baggerman. Proteomics in Cancer Research: Are We Ready for Clinical Practice? Critical Reviews in Oncology/Hematology, 96(3):437–448, 2015.

J. Matschinske, J. Spath, M. Bakhtiari, N. Probul, M. M. K. Majdabadi, R. Nasirigerdeh, R. Torkzadehmahani, A. Hartebrodt, B. Orban, S. Fejer, *et al.* The Featurecloud Platform for Federated Learning in Biomedicine: Unified Approach. Journal of Medical Internet Research, 25(1):e42621, 2023.

J. Mbatchou, L. Barnard, J. Backman, A. Marcketta, J. A. Kosmicki, A. Ziyatdinov, C. Benner, C. O'Dushlaine, M. Barber, B. Boutkov, L. Habegger, M. Ferreira, A. Baras, J. Reid, G. Abecasis, E. Maxwell, and J. Marchini. Computationally Efficient Whole-Genome Regression for Quantitative and Binary Traits. Nature Genetics, 53(7):1097–1103, 2021.

B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas. Communication-Efficient Learning of Deep Networks From Decentralized Data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, pages 1273–1282, 2017.

S. Mendelsohn, D. Froelicher, D. Loginov, D. Bernick, B. Berger, and H. Cho. Sfkit: A Web-Based Toolkit for Secure and Federated Genomic Analysis. Nucleic Acids Research, 51(W1):W535–W541, 2023.

R. Nasirigerdeh, R. Torkzadehmahani, J. Matschinske, J. Baumbach, D. Rueckert, and G. Kaissis. Hyfed: A Hybrid Federated Framework for Privacy-Preserving Machine Learning, 2021. arXiv preprint arXiv:2105.10545.

K. Nissim, S. Raskhodnikova, and A. Smith. Smooth Sensitivity and Sampling in Private Data Analysis. In Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing, pages 75–84, 2007.

P. Paillier. Public-Key Cryptosystems Based on Composite Degree Residuosity Classes. In International Conference on the Theory and Applications of Cryptographic Techniques, pages 223–238, 1999.

E. Papalexi and R. Satija. Single-Cell RNA Sequencing to Explore Immune Cell Heterogeneity. Nature Reviews Immunology, 18(1):35–45, 2018.

S. Pati, U. Baid, B. Edwards, M. Sheller, S. Wang, G. A. Reina, P. Foley, A. Gruzdev, D. Karkada, C. Davatzikos, C. Sako, S. Ghodasara, M. Bilello, S. Mohan, P. Vollmuth, G. Brugnara, C. J. Preetha, F. Sahm, K. Maier-Hein, M. Zenk, M. Bendszus, W. Wick, E. Calabrese, J. Rudie, J. Villanueva-Meyer, S. Cha, M. Ingalhalikar, M. Jadhav, U. Pandey, J. Saini, J. Garrett, M. Larson, R. Jeraj, S. Currie, R. Frood, K. Fatania, R. Y. Huang, K. Chang, C. Balana, J. Capellades, J. Puig, J. Trenkler, J. Pichler, G. Necker, A. Haunschmidt, S. Meckel, G. Shukla, S. Liem, G. S. Alexander, J. Lombardo, J. D. Palmer, A. E. Flanders, A. P. Dicker, H. I. Sair, C. K. Jones, A. Venkataraman, M. Jiang, T. Y. So, C. Chen, P. A. Heng, Q. Dou, M. Kozubek, F. Lux, J. Michalek, P. Matula, M. Kerkovsky, T. Koprivova, M. Dostal, V. Vyb´ıhal, M. A. Vogelbaum, J. R. Mitchell, J. Farinhas, J. A. Maldjian, C. G. B. Yogananda, M. C. Pinho, D. Reddy, J. Holcomb, B. C. Wagner, B. M. Ellingson, T. F. Cloughesy, C. Raymond, T. Oughourlian, A. Hagiwara, C. Wang, M. To, S. Bhardwaj, C. Chong, M. Agzarian, A. X. Falcao, S. B. Martins, B. C. A. Teixeira, F. Sprenger, D. Menotti, D. R. Lucio, P. LaMontagne, D. Marcus, B. Wiestler, F. Kofler, I. Ezhov, M. Metz, R. Jain, M. Lee, Y. W. Lui, R. McKinley, J. Slotboom, P. Radojewski, R. Meier, R. Wiest, D. Murcia, E. Fu, R. Haas, J. Thompson, D. R. Ormond, C. Badve, A. E. Sloan, V. Vadmal, K. Waite, R. R. Colen, L. Pei, M. Ak, A. Srinivasan, J. R. Bapuraj, A. Rao, N. Wang, O. Yoshiaki, T. Moritani, S. Turk, J. Lee, S. Prabhudesai, F. Moron, J. Mandel, K. Kamnitsas, B. Glocker, L. V. M. Dixon, M. Williams, P. Zampakis, V. Panagiotopoulos, P. Tsiganos, S. Alexiou, I. Haliassos, E. I. Zacharaki, K. Moustakas, C. Kalogeropoulou, D. M. Kardamakis, Y. S. Choi, S. Lee, J. H. Chang, S. S. Ahn, B. Luo, L. Poisson, N. Wen, P. Tiwari, R. Verma, R. Bareja, I. Yadav, J. Chen, N. Kumar, M. Smits, S. R.van der Voort, A. Alafandi, F. Incekara, M. M. J. Wijnenga, G. Kapsas, R. Gahrmann, J. W. Schouten, H. J. Dubbink, A. J. P. E. Vincent, M. J. van den Bent, P. J. French, S. Klein, Y. Yuan, S. Sharma, T. Tseng, S. Adabi, S. P. Niclou, O. Keunen, A. Hau, M. Valli`eres, D. Fortin, M. Lepage, B. Landman, K. Ramadass, K. Xu, S. Chotai, L. B. Chambless, A. Mistry, R. C. Thompson, Y. Gusev, K. Bhuvaneshwar, A. Sayah, C. Bencheqroun, A. Belouali, S. Madhavan, T. C. Booth, A. Chelliah, M. Modat, H. Shuaib, C. Dragos, A. Abayazeed, K. Kolodziej, M. Hill, A. Abbassy, S. Gamal, M. Mekhaimar, M. Qayati, M. Reyes, J. E. Park, J. Yun, H. S. Kim, A. Mahajan, M. Muzi, S. Benson, R. G. H. Beets-Tan, J. Teuwen, A. Herrera-Trujillo, M. Trujillo, W. Escobar, A. Abello, J. Bernal, J. Gomez, J. Choi, S. Baek, Y. Kim, H. Ismael, B. Allen, J. M. Buatti, A. Kotrotsou, H. Li, T. Weiss, M. Weller, A. Bink, B. Pouymayou, H. F. Shaykh, J. Saltz, P. Prasanna, S. Shrestha, K. M. Mani, D. Payne, T. Kurc, E. Pelaez, H. Franco-Maldonado, F. Loayza, S. Quevedo, P. Guevara, E. Torche, C. Mendoza, F. Vera, E. Rios, E. Lopez, S. A. Velastin, G. Ogbole, M. Soneye, D. Oyekunle, O. Odae-Oyibotha, B. Osobu, M. Shu'aibu, A. Dorcas, F. Dako, A. L. Simpson, M. Hamghalam, J. J. Peoples, R. Hu, A. Tran, D. Cutler, F. Y. Moraes, M. A. Boss, J. Gimpel, D. K. Veettil, K. Schmidt, B. Bialecki, S. Marella, C. Price, L. Cimino, C. Apgar, P. Shah, B. Menze, J. S. Barnholtz-Sloan, J. Martin, and S. Bakas. Federated Learning Enables Big Data for Rare Cancer Boundary Detection. Nature Communications, 13(1):7346, 2022.

D. Rajput, W. Wang, and C. Chen. Evaluation of a Decided Sample Size in Machine Learning Applications. BMC Bioinformatics, 24:48, 2023.

S. P. Ramu, P. Boopalan, Q. Pham, P. K. R. Maddikunta, T. Huynh-The, M. Alazab, T. T. Nguyen, and T. R. Gadekallu. Federated Learning Enabled Digital Twins for Smart Cities: Concepts, Recent Advances, and Future Directions. Sustainable Cities and Society, 79:103663, 2022.

P. Riedel, L. Schick, R. von Schwerin, M. Reichert, D. Schaudt, and A. Hafner. Comparative Analysis of Open-Source Federated Learning Frameworks-A Literature-Based Survey and Review. International Journal of Machine Learning and Cybernetics, 15:5257–5278, 2024.

N. Rieke, J. Hancox, W. Li, F. Milletari, H. R. Roth, S. Albarqouni, S. Bakas, M. N. Galtier, B. A. Landman, K. Maier-Hein, *et al.* The Future of Digital Health with Federated Learning. NPJ Digital Medicine, 3(1):1–7, 2020.

R. Rodriguez-Esteban and X. Jiang. Differential Gene Expression in Disease: A Comparison Between High-Throughput Studies and the Literature. BMC Medical Genomics, 10:1–10, 2017.

H. R. Roth, K. Chang, P. Singh, N. Neumark, W. Li, V. Gupta, S. Gupta, L. Qu, A. Ihsani, B. C. Bizzo, Y. Wen, V. Buch, M. Shah, F. Kitamura, M. Mendon¸ca, V. Lavor, A. Harouni, C. Compas, J. Tetreault, P. Dogra, Y. Cheng, S. Erdal, R. White, B. Hashemian, T. Schultz, M. Zhang, A. McCarthy, B. Min Yun, E. Sharaf, K. V. Hoebel, J. B. Patel, B. Chen, S. Ko, E. Leibovitz, E. D. Pisano, L. Coombs, D. Xu, K. J. Dreyer, I. Dayan, R. C. Naidu, M. Flores, D. Rubin, and J. Kalpathy-Cramer. Federated learning for breast density classification: A real-world implementation. In Domain Adaptation and Representation Transfer, and Distributed and Collaborative Learning: 2nd MICCAI Workshop, DART 2020, and 1st MICCAI Workshop, DCL 2020, pages 181–191. Springer, 2020.

F. Sattler, K. Muller, and W. Samek. Clustered Federated Learning: Model-Agnostic Distributed Multitask Optimization Under Privacy Constraints. IEEE Transactions on Neural Networks and Learning Systems, 32(8):3710–3722, 2021.

A. Schein, Z. S. Wu, M. Z. A. Schofield, and H. Wallach. Locally Private Bayesian Inference for Count Models. In Proceedings of the 36th International Conference on Machine Learning, pages 638–5648, 2019.

M. Scutari and M. Malvestio. The Pragmatic Programmer for Machine Learning: Engineering Analytics and Data Science Solutions. Chapman & Hall, 2023.

M. J. Sheller, B. Edwards, G. A. Reina, J. Martin, S. Pati, A. Kotrotsou, M. Milchenko, W. Xu, D. Marcus, R. R. Colen, and S. Bakas. Federated Learning in Medicine: Facilitating Multi-Institutional Collaborations Without Sharing Patient Data. Scientific Reports, 10(1), 2020.

R. Shokri, M. Stronati, C. Song, and V. Shmatikov. Membership Inference Attacks Against Machine Learning Models. In 2017 IEEE Symposium on Security and Privacy, pages 3–18, 2017.

C. Sudlow, J. Gallacher, N. Allen, V. Beral, P. Burton, J. Danesh, P. Downey, P. Elliott, J. Green, M. Landray, B. Liu, P. Matthews, G. Ong, J. Pell, A. Silman, A. Young, T. Sprosen,T. Peakman, and R. Collins. UK Biobank: An Open Access Resource for Identifying the Causes of a Wide Range of Complex Diseases of Middle and Old Age. PLoS Medicine, 12(3):e1001779, 2015.

P. Suetens. Fundamentals of Medical Imaging. Cambridge University Press, 3rd edition, 2017.

G. Sun, Y. Cong, J. Dong, Q. Wang, L. Lyu, and J. Liu. Data Poisoning Attacks on Federated Machine Learning. IEEE Internet of Things Journal, 9(13):11365–11375, 2022.

A. Z. Tan, H. Yu, L. Cui, and Q. Yang. Towards Personalized Federated Learning. IEEE Transactions on Neural Networks and Learning Systems, 34(12):9587–9603, 2022.

C. V. Theodoris, L. Xiao, A. Chopra, M. D. Chaffin, Z. R. Al Sayed, M. C. Hill, H. Mantineo, E. M. Brydon, Z. Zeng, X. S. Liu, *et al.* Transfer Learning Enables Predictions in Network Biology. Nature, 618(7965):616–624, 2023.

D. Toro-Dominguez, J. A. Villatoro-Garcia, J. Martorell-Marugan, Y. Roman-Montoya, M. E.Alarcon-Riquelme, and P. Carmona-Saez. A Survey of Gene Expression Meta-Analysis: Methods and Applications. Briefings in Bioinformatics, 22(2):1694–1705, 2021.

N. Truong, K. Sun, S. Wang, F. Guitton, and Y. Guo. Privacy Preservation in Federated Learning: An Insightful Survey From the GDPR Perspective. Computers & Security, 110:102402, 2021.

U.S. Congress. National Artificial Intelligence Initiative Act of 2020, 2020. Public Law No:

116-283, Division E.

S. M. van Rooden, S. D. van der Werff, M. S. M. van Mourik, F. Lomholt, K. L. Møller, S. Valk, C. dos Santos Ribeiro, A. Wong, S. Haitjema, M. Behnke, and E. Rinaldi. Federated Systems for Automated Infection Surveillance: A Perspective. Antimicrobial Resistance & Infection Control, 13(1), 2024.

Z. Wan, J. W. Hazel, E. W. Clayton, Y. Vorobeychik, M. Kantarcioglu, and B. A. Malin. Sociotechnical Safeguards for Genomic Data Privacy. Nature Reviews Genetics, 23(7):429–445, 2022.

Q. Wang, M. He, L. Guo, and H. Chai. Afei: Adaptive Optimized Vertical Federated Learning for Heterogeneous Multi-Omics Data Integration. Briefings in Bioinformatics, 24(5):bbad269, 2023.

S. Wang, M. Kim, W. Li, X. Jiang, H. Chen, and A. Harmanci. Privacy-Aware Estimation of Relatedness in Admixed Populations. Briefings in Bioinformatics, 23(6), 2022.

S. Wang, B. Shen, L. Guo, M. Shang, J. Liu, Q. Sun, and B. Shen. Scfed: Federated Learning for Cell Type Classification with scRNA-seq. Briefings in Bioinformatics, 25(1):bbad507, 2024.

J. Xu, B. S. Glicksberg, C. Su, P. Walker, J. Bian, and F. Wang. Federated Learning for Healthcare Informatics. Journal of Healthcare Informatics Research, 5(1):1–19, 2020.

R. Yan, L. Qu, Q. Wei, S. Huang, L. Shen, D. L. Rubin, L. Xing, and Y. Zhou. Label-Efficient Self-Supervised Federated Learning for Tackling Data Heterogeneity in Medical Imaging. IEEE Transactions on Medical Imaging, 42(7):1932–1943, 2023.

D. Yang, Z. Xu, W. Li, A. Myronenko, H. R. Roth, S. Harmon, S. Xu, B. Turkbey, E. Turkbey, X. Wang, W. Zhu, G. Carrafiello, F. Patella, M. Cariati, H. Obinata, H. Mori, K. Tamura, P. An, B. J. Wood, and D. Xu. Federated Semi-Supervised Learning for COVID Region Segmentation in Chest CT Using Multi-National Data From China, Italy, Japan. Medical Image Analysis, 70:101992, 2021.

X. Yin, Y. Zhu, and J. Hu. A Comprehensive Survey of Privacy-Preserving Federated Learning: A Taxonomy, Review, and Future Directions. ACM Computing Surveys, 54(6):1–36, 2021.

W. Zhang, D. Yang, W. Wu, H. Peng, N. Zhang, H. Zhang, and X. Shen. Optimizing Federated Learning in Distributed Industrial Iot: A Multi-Agent Approach. IEEE Journal on Selected Areas in Communications, 39(12):3688–3703, 2021.

C. Zhao, S. Zhao, M. Zhao, Z. Chen, C. Gao, H. Li, and Y. Tan. Secure Multi-Party Computation: Theory, Practice and Applications. Information Sciences, 476:357–372, 2019.

Y. Zhu, L. M. Orre, Y. Z. Tran, G. Mermelekas, H. J. Johansson, A. Malyutina, S. Anders, and J. Lehtio. DEqMS: A Method for Accurate Variance Estimation in Differential Protein Expression Analysis. Molecular & Cellular Proteomics, 19(6):1047–1057, 2020.

A. Ziller, A. Trask, A. Lopardo, B. Szymkow, B. Wagner, E. Bluemke, J. Nounahon, J. Passerat-Palmbach, K. Prakash, N. Rose, *et al.* Pysyft: A Library for Easy Federated Learning. Federated Learning Systems: Towards Next-Generation AI, pages 111–139, 2021.

O. Zolotareva, R. Nasirigerdeh, J. Matschinske, R. Torkzadehmahani, M. Bakhtiari, T. Frisch, J. Spath, D. B. Blumenthal, A. Abbasinejad, P. Tieri, *et al.* Flimma: A Federated and Privacy-Aware Tool for Differential Gene ExpressionAnalysis. Genome Biology, 22(1):1–26, 2021.

A. Y. Zomaya. Parallel Computing for Bioinformatics and Computational Biology: Models, Enabling Technologies, and Case Studies. John Wiley & Sons, 2006.