

Bayesian Federated Learning: A Survey

Longbing Cao^{1,2}, Hui Chen¹, Xuhui Fan³, Joao Gama⁴, Yew-Soon Ong⁵ and Vipin Kumar⁶

¹University of Technology Sydney, Australia

²Macquarie University, Australia

³University of New Castle, Australia

⁴INESC TEC - LIAAD, Portugal

⁵Nanyang Technological University, Singapore

⁶University of Minnesota, USA

Abstract

Federated learning (FL) demonstrates its advantages in integrating distributed infrastructure, communication, computing and learning in a privacy-preserving manner. However, the robustness and capabilities of existing FL methods are challenged by limited and dynamic data and conditions, complexities including heterogeneities and uncertainties, and analytical explainability. *Bayesian federated learning* (BFL) has emerged as a promising approach to address these issues. This survey presents a critical overview of BFL, including its basic concepts, its relations to Bayesian learning in the context of FL, and a taxonomy of BFL from both Bayesian and federated perspectives. We categorize and discuss client- and server-side and FL-based BFL methods and their pros and cons. The limitations of the existing BFL methods and the future directions of BFL research further address the intricate requirements of real-life FL applications.

1 Introduction

Decentralized AI (DeAI) and machine learning (DeML) address new demands for engaging decentralized edge devices, nodes and servers to undertake secure edge-level and distributed AI and learning tasks [Cao, 2022c; McMahan *et al.*, 2017]. A critical DeAI and ML technique is federated learning (FL) [Kairouz *et al.*, 2021; Yang, 2021], which ensures decentralized learning on local data without sharing but with properties including privacy, security, heterogeneity, and personalization. FL has seen significant developments to leverage distributed and centralized ML techniques, catering to centralized, decentralized, heterogeneous, or personalized settings and requirements with privacy preservation. These include algorithms for preserving privacy [Liu *et al.*, 2022; Elgabli *et al.*, 2022], reducing communication consumption [Zhang *et al.*, 2021], handling system heterogeneity [Zong *et al.*, 2021], and addressing personalized requirements [T Dinh *et al.*, 2020; Fallah *et al.*, 2020].

However, FL faces various fundamental challenges in enabling DeAI and DeML for real-world applications. First, client data on edge nodes and devices may be very limited

and it may be costly to obtain labelled samples. Second, client conditions and behaviors are often dynamic, presenting strong uncertainties. Last but not least, decentralized applications are non-IID, involving heterogeneities of devices, behaviors, goals, and data and their interactions [Cao, 2022b]. These are challenges facing existing FL algorithms and inspire a promising direction - Bayesian Federated Learning (BFL) [Zhang *et al.*, 2022], which integrates the advantages of Bayesian learning (BL) into FL. In existing BFL methods, BL approaches incorporate prior knowledge about data to leverage a limited number of samples and learn distributions over FL parameters and statistical heterogeneity to quantify uncertainties and dynamics. Further, the strengths of FL in handling privacy, communication and heterogeneity are fused with BL. Consequently, BFL enables more robust, well-calibrated and dynamic predictions for safety- and privacy-critical applications. BFL has demonstrated various applications, such as for FinTech, driverless cars, Industry 4.0, medical diagnosis, and differential privacy [Achituve *et al.*, 2021; Snell and Zemel, 2021; Kendall and Gal, 2017; Blundell *et al.*, 2015; Triastcyn and Faltings, 2019; Yang, 2021].

BFL shows a promising potential to substantially expand the existing FL and address complex real-world requirements for DeAI, DeML and FL. Here, we systematically review the relevant work on BFL, conduct a critical analysis of their pros and cons, and present the challenges and opportunities for comprehensive BFL research. First, we briefly discuss the transfer from FL and BL to their integrative BFL. Then, a BFL taxonomy offers a structure of BFL in terms of both BL methods and FL research issues. We discuss and compare the advantages and disadvantages of different methods for client- and server-side BFL and various categories of BFL from the FL perspective. Lastly, we discuss the gaps and future directions of BFL in addressing broad-reaching, more realistic and actionable [Cao and Zhang, 2006; Cao, 2013] FL, BL, DeAI, and DeML settings and tasks and real-world FL scenarios and requirements.

2 From Federated and Bayesian Learning to Bayesian Federated Learning

We summarize FL concepts, topics and challenges. Then, BL methods and advantages are discussed. BFL integrates BL into FL addressing FL challenges using BL advantages.

2.1 Federated Learning and Challenges

FL concepts. Federated learning initially aims to address privacy leakage in distributed learning systems [Yang, 2021; Kairouz *et al.*, 2021]. In general, an FL system has a server and multiple clients and utilizes an iterative learning process through server-client communications. In each communication round, the server trains a global model and sends its model parameters to participating clients. Each client trains its local model on local data. Clients share model parameters (rather than local data) with the server, which pools and aggregates these local updates to update its global model for the next communication round. This process iterates until convergence or it reaches some stopping conditions.

Typically, FL achieves an objective function as follows:

$$\min_{\mathbf{w}} F(\mathbf{w}) = \sum_{m=1}^M \frac{|D_m|}{|D|} F_m(\mathbf{w}), \quad (1)$$

$$\text{where } F_m(\mathbf{w}) = \frac{1}{|D_m|} \sum_i l(\mathbf{x}_i, y_i; \mathbf{w}). \quad (2)$$

Here, M denotes the number of activated clients, D and D_m denote the aggregated dataset from all participating clients and the dataset of the m -th client, respectively, \mathbf{w} denotes the model parameters of Bayesian neural networks and $F_m(\mathbf{w})$ denotes the empirical risk of the m -th client with the loss function l over each instance (\mathbf{x}_i, y_i) [Chen and Chao, 2022].

FL research topics. FL has seen diversified rapid developments, which include the following categories. (1) Privacy-preserving FL [Liu *et al.*, 2022; Elgabli *et al.*, 2022] for differential privacy; (2) Communication-efficient FL [Lin *et al.*, 2022; Li *et al.*, 2022] addressing communication bottlenecks, [Dinh *et al.*, 2020] on resource allocation, [Zang *et al.*, 2022] on sparsified and compressed communication, and [Liu *et al.*, 2020] for propagating channels; (3) Heterogeneous and personalized FL [Ghari and Shen, 2022; T Dinh *et al.*, 2020] to address heterogeneities of local clients; and (4) FL optimization with different settings [Liu *et al.*, 2021; Malinovskiy *et al.*, 2020] for federated SGD, [McMahan *et al.*, 2017] for federated averaging, [Chen *et al.*, 2021] for dynamic aggregation, and [Durmus *et al.*, 2021] for dynamic regularization. In addition, there are many other (1) FL tasks, such as hierarchical FL, non-IID FL, unsupervised to semi-supervised FL, multitask FL, robust FL, fair and unbiased FL; and (2) FL application settings, such as blockchained FL, multimodal FL, secure FL, energy-aware and green FL.

FL challenges. However, these previous work on FL still faces various challenges as mentioned in the introduction. In particular, FL suffers from uncertain, dynamic, limited, and unsupervised problems and also data. These techniques also cannot provide robust and analytically explainable results.

2.2 Bayesian Learning and Advantages

BL concepts. Bayesian learning (BL) builds on Bayes' theorem [Benavoli and de Campos, 2021; Sun *et al.*, 2020]. It aims at learning the posterior distribution $p(\mathbf{w} | D)$ for parameter \mathbf{w} based on its prior distribution $p(\mathbf{w})$ and the corresponding likelihood $p(D | \mathbf{w})$ on observed evidence D . BL

then learns the posterior as follows:

$$p(\mathbf{w} | D) = \frac{p(D | \mathbf{w})p(\mathbf{w})}{p(D)}, \quad (3)$$

It thus converts a prior probability $p(\mathbf{w})$ into a posterior probability $p(\mathbf{w} | D)$ with the likelihood $p(D | \mathbf{w})$ for parameter \mathbf{w} on observed evidence D .

BL methods. Bayes' theorem has been widely applied in various settings and has generated numerous BL methods [Jospin *et al.*, 2022; Gershman and Blei, 2012] for tasks such as Bayesian classification, regression, clustering, representation, and optimization. In addition, these can be categorized in terms of modeling mechanisms and settings, learning tasks, and application scenarios, for example (1) Bayesian classification, such as naive Bayes and Bayesian belief networks for classification or optimization; (2) Bayesian approximation, inference and optimization, such as Laplace's approximation, expectation propagation, and variational approximation and Markov chain Monte Carlo (MCMC) methods such as Gibbs sampling; (3) hierarchical Bayesian models with hierarchical variable/parameter dependence, such as Beta-Poisson models; (4) dynamic BL such as Bayesian nonparametric models; (5) BL for other settings and tasks, such as Bayesian continual learning, Bayesian model averaging, and Bayesian posterior decomposition; and (6) hybrid BL methods, such as Bayesian neural networks, and Bayesian model ensemble.

BL advantages. BL shows various advantages. (1) *Quantifying uncertainty* over probability distributions rather than specific values, which captures the model or epistemic uncertainty of model parameters [Kendall and Gal, 2017]. This can justify the reliability of results, such as for safety-critical applications. (2) *Enhancing robustness* by evidence-based likelihood estimates, which learns the parameter distributions and regularizes the model for more robust parameters. (3) *Improving performance on limited data* using prior distribution for each model parameter, which captures prior knowledge about the data and problem. This can greatly improve modeling performance particularly with limited data [Sun *et al.*, 2019].

2.3 BFL: Bayesian Federated Learning

The discussed challenges facing FL can be mostly overcome by taking advantage of BL to address various real-life needs. Consequently, Bayesian federated learning (BFL) has been explored to incorporate BL principles and advantages into FL frameworks and tasks for stronger model robustness and learning improved performance on small-scale data [Zhang *et al.*, 2022]. BFL could lead to more robust, better explainable, and higher performance in handling uncertainties and process-oriented (rather than point-based) challenges. Such advantages could benefit various applications with strong uncertainties, such as estimating financial market dynamics, medical conditions, infectious diseases, financial crisis, and natural disaster.

Although no unified definitions are available for BFL, Figure 1 illustrates (1) the general framework of BFL, which integrates mechanisms of FL and BL; and (2) a general iterative learning process of BFL. Differing from pure FL, a BFL system learns global posterior $p(\mathbf{w} | D)$ for the server and local

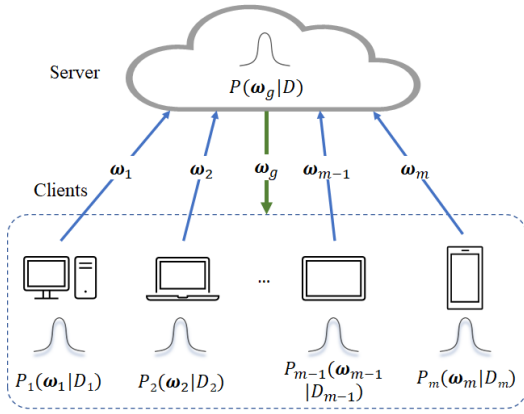


Figure 1: The framework and iterative learning process of Bayesian federated learning (BFL).

posterior $p_m(\mathbf{w}_m | D_m)$ for each participating client m on its local data D_m . A general or localized prior $p(\mathbf{w})$ applies to all clients. BFL incorporates Bayesian principles into FL for the server:

$$p(\mathbf{w}_g | D) = \frac{p(D | \mathbf{w}_g)p(\mathbf{w}_g)}{p(D)}, \quad (4)$$

and clients:

$$p(\mathbf{w}_m | D_m) = \frac{p(D_m | \mathbf{w}_m)p(\mathbf{w}_m)}{p(D_m)}. \quad (5)$$

\mathbf{w}_g and \mathbf{w}_m represent the parameters of the server and the m -th client respectively, $p(D)$ denotes a normalization constant. In non-personalized BFL, \mathbf{w}_g and \mathbf{w}_m are equivalent.

Accordingly, a BFL model converts the FL objective function Eq. (1) to BFL for a global loss, aligned local losses, or a mixed loss with settings (e.g., privacy or security preservation, or communication efficiency) and regularization:

$$\arg \min_{\mathbf{w}_m, \mathbf{w}_g} \sum_{m=1}^M l(\mathbf{w}_m; D_m) + \alpha \|\mathbf{w}_m, \mathbf{w}_g\|. \quad (6)$$

Then, the posteriors $p(\mathbf{w}|D)$ can be approximated by Bayesian optimization, for example, variational inference (VI). Accordingly, Eq. (6) can be converted to an ELBO-like objective \mathcal{L}_D , the true posterior $p(\mathbf{w}_m, \mathbf{w}_g | D)$ is approximated by a variational distribution $q_\theta(\mathbf{w}_m, \mathbf{w}_g)$ estimated on the evidence D with parameters θ :

$$\arg \min_{\mathbf{w}_m, q_\theta(\mathbf{w}_m)} KL(q_\theta(\mathbf{w}_m) \| p(\mathbf{w}_m)) - \mathbb{E}_{q_\theta(\mathbf{w}_m)} [\log p(D_m | \mathbf{w}_m)]. \quad (7)$$

KL refers to the Kullback–Leibler divergence. Such an objective function determines the optimization of a BFL system in fulfilling client or global objectives, or their mixture; and distinguishes generic FL from personalized FL.

3 BFL Taxonomy and Analyses

Here, we present a taxonomy of BFL and conduct critical analyses of representative BFL methods from both Bayesian and federated perspectives.

3.1 BFL Taxonomy

BFL tasks and methods can be categorized per aspects such as FL settings or BL methods. First, taking a Bayesian perspective, we categorize BFL in terms of FL architectures: client-side BFL and server-side BFL. Client-side BFL focuses on learning local models on client nodes using Bayesian methods, which involves representative methods: (1) Federated Bayesian privacy (FBP), (2) Bayesian neural networks (BNNs) for local models, (3) Bayesian optimization (BO) for local optimization, and (4) Bayesian nonparametric (BNP) models for dynamic FL. Server-side BFL aggregates local updates for global models using Bayesian methods, with typical methods including (1) Bayesian model ensemble (BME) for aggregation, (2) Bayesian posterior decomposition (BPD), and (3) Bayesian continual learning (BCL). Then, from the FL perspective, we can categorize BFL methods into heterogeneous, hierarchical, dynamic, personalized, and hybrid BFL, etc. Figure 2 shows a taxonomy of BFL and its connections to BL and FL, respectively.

3.2 Client-side BFL

Various Bayesian methods address client requirements and learn local models. Representative client-side methods include FBP, BNN, BO and BNP for different client-side requirements and objectives.

Federated Bayesian Privacy

Earlier FL models focus on privacy-preserving federated updating, communication, and aggregation, with typical progress on differential privacy [Elgabli *et al.*, 2022]. Various differential privacy methods involve simple statistics and object or output perturbation. By considering the randomness of local data, Bayesian differential privacy (BDP) [Triastcyn and Faltings, 2019] ensures client privacy, instance privacy and their joint privacy by a privacy loss accounting method. In [Gu *et al.*, 2021], KL-divergence quantifies Bayesian privacy loss during data restoration, forming a federated deep learning for private passport (FDL-PP) method against FL restoration attacks.

Discussion. Both BDP and FDL-PP relax constraints on existing FL differential privacy by incorporating uncertainty. However, they cannot handle complex FL and BL settings and privacy-preserving requirements.

Bayesian Neural Networks for FL Local Models

Bayesian neural networks (BNNs) [Blundell *et al.*, 2015] combine Bayesian inference with neural networks. BNNs are incorporated into FL in various ways. pFedBayes [Zhang *et al.*, 2022] uses a BNN to train local models in each communication round. Its objective is formulated as a two-level optimization problem. For each client, pFedBayes uses VI to approximate the posterior distribution of local model parameters by minimizing the loss function of the local model. On the server, pFedBayes aggregates local models and minimizes the averaged loss functions of clients. pFedBayes [Kairouz *et al.*, 2021] updates local parameters again after updating them using BNN for each client, resulting in a localized global model to overcome the challenge of non-IID data. FedPPD

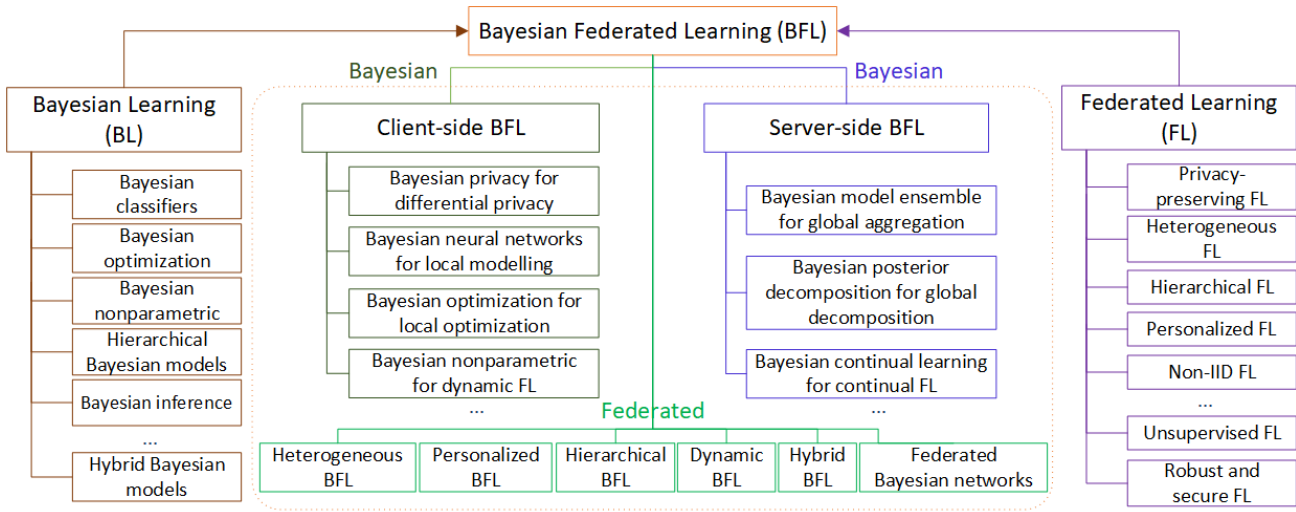


Figure 2: Taxonomy of Bayesian federated learning (BFL) and its connections to Bayesian learning and federated learning. The right and left panels show the mechanisms of FL and BL to support BFL, respectively. The middle panel consists of BFL methods in terms of Bayesian (client- and server-side) and federated perspectives.

[Bhatt *et al.*, 2022] also applies a BNN to train local models. The key difference between FedPPD and pFedBayes lies in the methods of approximating posterior distributions of model parameters. pFedBayes uses VI while FedPPD uses MCMC. FedPPD utilizes a Bayesian dark knowledge method to distill the posterior prediction distribution into a single deep neural network (DNN) for each client and then sends the resulting teacher model (approximate maximum posterior sample) and student model (approximate posterior predictive distribution) to the server for aggregation.

Discussion. BNNs train local models and enable FL to quantify local uncertainty while using DNNs for task learning, thus improving FL robustness. BNNs also improve the FL learning performance on limited data. However, BNNs also bring challenges to FL. First, the BFL with BNNs involves huge computational and memory costs for training local models, especially when the scale of local model parameters is immense. Second, it may be challenging to choose appropriate prior distributions for local model parameters particularly when we cannot estimate complex relationships between model outputs and parameters.

Bayesian Optimization for FL Local Optimization

Bayesian optimization (BO) is a sequential optimization approach, often used to tune the hyperparameters of DNNs. Various methods involve BO for FL. In [Dai *et al.*, 2020], a federated Bayesian optimization (FBO) setting uses BO to optimize local models in each communication round. The FTS algorithm for FBO uses a Gaussian process (GP) as the surrogate for modelling the objective and acquisition functions by Thompson sampling. Random Fourier features are used to approximate GP for scalability and information exchange (the parameters of random Fourier features are shared in each communication round). FTS has a strict convergence guarantee even for non-IID data. In [Zang *et al.*, 2022], BO supports FL for traffic flow prediction (TFP), and like FTS

and other BO algorithms, GP is used as an ideal objective function surrogate. Unlike FTS, since FBO for TFP utilizes BO dynamically to adjust the weights of local models for aggregation, FBO for TFP does not suffer from performance degradation when encountering heterogeneous data. Moreover, FBO for TFP involves an expected improvement instead of Thompson sampling as the acquisition function of the models without addressing the scalability problem of GP.

Discussion. Applying BO to FL achieves a relatively robust learning performance for non-IID local datasets due to the inherent properties of BO. Compared with traditional optimization algorithms, BO is simpler and more convenient to implement. However, the practicality of FBO is challenging for models with substantial data points. Moreover, the slow convergence rate of FBO is open to address.

Bayesian Nonparametric models for Dynamic FL

Bayesian nonparametric (BNP) models enable dynamic learning [Gershman and Blei, 2012]. In BFL, BNP models apply a Gaussian or Beta-Bernoulli process (BBP). In GP-based FL, pFedGP [Achituv *et al.*, 2021] utilizes a personalized GP classifier to train a local model for each client and shares a kernel function for all clients. FedLoc [Yin *et al.*, 2020] also uses GP to train local models for regression tasks. Unlike pFedGP, FedLoc cannot effectively deal with non-IID data for FL [Li *et al.*, 2020]. Different from these methods, FedCor [Tang *et al.*, 2022] uses GP to predict the loss change and then selects the clients that need to be activated in each communication round according to the loss change. FedCor is only applicable for the cross-silo FL framework, where the learning performance decays significantly for non-IID data.

In BBP-based FL, PFNM [Yurochkin *et al.*, 2019] uses BBP to find the matched subsets of neurons among local models. However, PFNM is only applicable to a simple feedforward neural network structure. Then, a layer-wise matching algorithm FedMA based on BBP extends PFNM

to other neural network structures [Wang *et al.*, 2020] such as convolutional neural networks (CNNs) and long short-term memory (LSTM). However, neither PFNM nor FedMA can achieve ideal learning performance on non-IID data.

Discussion. Since model complexity can adapt to data, BNP methods can train local models more flexibly than parametric methods [Orbanz and Teh, 2010]. However, because the complexity of BNP models grows with an increase in data, BNP for FL raises the computational power of clients with a large amount of data.

3.3 Server-side BFL

Various server-side BFL methods implement the global aggregation or decomposition of all updated local models for clients in each communication round. We introduce Bayesian methods BME and BPD. Further, BFL works in other settings, such as continual learning using BCL.

Bayesian Model Ensemble for FL Aggregation

BL models approximate posterior distributions over model parameters using stochastic (e.g., MCMC) or deterministic (e.g., VI) [Kingma and Welling, 2014; Wu *et al.*, 2023] methods. For stochastic methods, the posterior distribution can be approximated by random sampling [Murphy, 2022]. Hence, each sample can be viewed as a base learner for the Bayesian model ensemble (BME).

In BME-based FL, FedBE [Chen and Chao, 2021] first utilizes the parameters of clients to construct a global posterior distribution (Gaussian or Dirichlet) in the aggregation phase of each communication round. MCMC then samples from this global posterior distribution to obtain the ensemble model, which is used on unlabeled data to obtain a pseudo-labeled dataset. Finally, a stochastic weighted averaging algorithm distills the global model on this pseudo-labeled data. Similarly, FedPPD introduces three aggregation algorithms with one serving ensemble similar to FedBE. However, in the local learning phase of each communication round, FedBE still uses the point estimation method to obtain the local model parameters of clients, while FedPPD utilizes a BNN.

Discussion. BME helps to implement Bayesian inference on the server using the information of all clients more effectively to prevent model performance degradation. This works especially for local models on non-IID data. Compared with other approximation methods, the sampling method is simpler and more accurate. However, obtaining an accurate ensemble model requires as many samples as possible, increasing the computational power of devices. Also, we cannot obtain a specific global parameter directly by BME. Other methods for distilling an appropriate global parameter are required for client learning in the next communication round.

Bayesian Posterior Decomposition for FL Decomposition

For some ML tasks, it is essential yet challenging to decompose a model into a combination of sub-models that can be handled more easily than the original model [Abrial, 2009]. This requires model decomposition. For Bayesian learning, Bayesian posterior decomposition (BPD) serves this purpose by decomposing the posterior of the global model to a combination of local models.

For BPD-based FL, by imposing strong constraints on uniform prior and local data independence, FedPA [Al-Shedivat *et al.*, 2021] decomposes global posterior distribution into the product of local posterior distributions of clients in each communication round. A concrete expression of parameters of the global posterior distribution can be obtained by federated least squares. Since the direct calculation of parameters from the global posterior will incur high computational and communication costs, calculating the global posterior is converted into an optimization problem and solved by sampling. Since the independent approximations of local posteriors cannot guarantee an accurate global posterior approximation, FedEP [Guo *et al.*, 2023] extends FedPA to approximate a global model using an expectation propagation method. QLSA [Vono *et al.*, 2022] uses the same method as in FedPA to decompose the global distribution. QLSA mainly performs the update of clients per the quantised Langevin stochastic dynamics and sends the resulting compressed gradient to the server to control communication consumption in FL. Differing from these models, FOLA also uses BPD to decompose the global posterior approximation into the product of local posterior distributions with weighted terms, which does not require any strong constraints. VIRTUAL [Corinzia *et al.*, 2019] also uses BPD for aggregation on the server. To avoid catastrophic forgetting, its global posterior distribution is decomposed into the product of the global posterior distribution of the previous communication round and local posterior distributions (ratios of clients) of the current communication.

Discussion. Most FL methods use naive parameter averaging FedAVG [McMahan *et al.*, 2017] for model aggregation, which causes performance degradation when local data involves statistical heterogeneity. However, the stability of the BPD-based FL models on heterogeneous data can be significantly improved by global posterior model decomposition. More importantly, BPD also enables better interpretability of FL models. However, BPD introduces challenges to FL. First, BPD may require other algorithms to assist model learning, which may incur a large computational overhead or even be intractable. Second, some decomposition methods may involve strong constraints, which are usually not feasible when solving practical problems.

Bayesian Continual Learning for Continual FL

Continual learning updates models over a variety of datasets, which may change over time in a sequential manner [Nguyen *et al.*, 2018]. Bayesian learning can facilitate the property of continual learning and use the model posterior as the prior of the next task to achieve Bayesian continual learning (BCL).

For BCL-based FL, FOLA [Liu *et al.*, 2021] uses the product of local posteriors to obtain a global posterior rather than a simple mixture of local posteriors for clients in each communication round for more robust learning on non-IID data. The resulting global posterior will then be sent to all clients at the next communication round as a prior. Similar to FOLA, in each communication round of pFedBayes, the prior distribution of a local model is replaced by the global distribution of the previous communication round to improve the learning performance and interpretability. Nevertheless, there are two main differences between FOLA and pFedBayes. First, the

global distribution of pFedBayes is merely obtained by means of an incremental local model averaging, while the global distribution of FOLA is obtained by the product of local posteriors. Second, the local model for each client in pFedBayes is a BNN, while FOLA is a classic neural network.

Discussion. By applying BCL to FL, we can leverage the information aggregated from previous communication rounds. Such an online learning approach often results in better learning performance. However, a complex prior distribution may bring substantial computational overhead to modeling and result in limited performance improvement. It is thus challenging to arrive at a trade-off between using BCL and assuming a suitable prior distribution for FL.

3.4 BFL from the Federated Perspective

BFL research can also be categorized per other FL aspects as discussed in Sec. 2.1. Accordingly, we review specific FL tasks using heterogeneous, personalized, hierarchical, dynamic and hybrid BFL methods. Further, federated Bayesian networks implement BL using FL or in a federated structure.

Heterogeneous BFL. Heterogeneous BFL handles heterogeneous clients with different prior distributions, parameter distributions, or posterior distributions for individual clients. For example, [Kotelevskii *et al.*, 2022] addresses heterogeneous clients using a mixed-effect model for each client, which includes fixed effect (common representation) and random effect (personalized representation). FedBE alleviates the heterogeneity between clients by fitting a posterior distribution for all possible global models. However, existing research generally assumes all local models share the same global model architecture with minor adjustments on priors or parameter distributions. These cannot handle complex heterogeneities, such as heterogeneous structures and relations of clients [Li *et al.*, 2020; Cao, 2022d].

Personalized BFL. Personalized BFL defines a personalized model for each client to deal with its distinct data distributions. For example, pFedBayes uses a personalized model for each client by minimizing the KL divergence between the global model and each updated local model. Instead, pFedGP constructs a personalized GP classifier for each client on its local data. However, existing methods may focus on client heterogeneity but not personalize model parameters, learning tasks or objectives for each client [Tan *et al.*, 2022].

Hierarchical BFL. Hierarchical BFL addresses scenarios where client features or edge nodes may be grouped into another layer, involving hierarchical FL structures. To the best of our knowledge, no work extends BL to hierarchical FL. In [Chen *et al.*, 2022], a hierarchical Bayesian model captures intra- and inter-client uncertainties to optimize hyperparameters: initial value, learning rate, and a number of (early-stop) steps. Nonetheless, existing research ignores complex interactions and couplings between client features or edge nodes, leading to poor performance [Briggs *et al.*, 2020].

Dynamic BFL. Dynamic BFL handles evolving, temporal, or random characteristics, settings or tasks, which may also involve drifts (shifts, changes, variations) of client features, node features, or client-server interactions over time or

other dimensions. These require dynamic BL mechanisms, such as updating priors, the number of clients, feature dynamics, and posterior contributions. To address the change in client contributions, the FBO for TFP uses BO to dynamically adjust the weights of clients in each communication round. FOLA uses BCL to dynamically adjust the prior distributions of clients to accelerate model convergence. However, these methods generally overlook the dynamic components of FL, resulting in slow model convergence and a local optimal solution [Chen *et al.*, 2021] and other gaps.

Hybrid BFL. Hybrid BFL is useful to address FL with mixed client structures, features, tasks, priors, posteriors, etc. In FedPPD, a BNN trains local models for FL. FedBE utilizes a model ensemble to improve the robustness of FL. While hybrid BFL models may be able to handle diverse problems, limited existing work is available [Huang *et al.*, 2021].

Federated Bayesian networks. In this review, we exclude research on federated Bayesian networks which use FL for BL. Examples are [Ng and Zhang, 2022] which collectively learns a Bayesian network from partitioned data. Note that one may interchangeably use FBN for BFL, as in [Corinzia *et al.*, 2019; Kassab and Simeone, 2022].

Discussion. The related research on these BFL topics is very limited, immature, incomplete and imbalanced. As discussed above, some of the settings, requirements or tasks have not been explored yet. In the following section, we further discuss these and the future directions of BFL research.

4 Gaps and Directions

BFL has demonstrated significant progress and potential for improving FL on limited, dynamic and uncertain data. However, the existing BFL research also shows significant limitations and gaps in addressing theoretical and practical FL requirements, problems, and challenges.

4.1 Gap Analyses

Table 1 summarizes and compares both client- and server-based BFL methods discussed in Section 3. Their main limitations and gaps can be summarized as follows.

First, existing BFL methods have various limitations. Typical issues include high computational costs and low communication efficiency. Most BFL models involve strong constraints on (1) FL settings such as client independence and heterogeneity, privacy, resources, and communication costs, and (2) BL settings such as prior distributions and uncertainty of clients. High computational and communication overheads limit the applications of BFL, particularly for decentralized, cross-device and personalized FL tasks. The oversimplified settings and approaches limit their capacity and applications.

Second, existing BFL exhibits weak-to-no capabilities of handling complex interactions and heterogeneities in FL applications with non-IID data. Real-life FL systems are non-IID, involving comprehensive non-IIDnesses [Cao, 2022d; Cao, 2022b; Cao, 2016]. Examples include heterogeneous, interactive, coupled and hierarchical entities, features, relations, and structures within and between clients and communities and between clients and servers [Cao, 2022d]. Existing

personalized FL and BFL only weaken these non-IIDnesses through strategies such as neutralizing heterogeneities across clients, e.g., by unified global optimal parameters or simplified Gaussian processes. The statistical heterogeneity between clients requires each client to learn its personalized optimal parameters. Some devices and nodes may interact and couple with each other, while existing FL and BFL overlook their client couplings [Cao, 2022d; Cao, 2022b; Cao, 2016; Pang *et al.*, 2017]. Most of existing references on non-IID FL do not address the above non-IIDnesses or significantly simplify these challenges [Cao, 2022d].

In addition, BFL models still suffer from weak accuracy, robustness and adaption. These may be attributed to poor-to-weak priors, heterogeneous, dynamic and hybrid clients or tasks, weak optimization, or inappropriate updating and aggregation mechanisms. There are also gaps in implementing local and global optimization in various settings, such as heterogeneous, hybrid, dynamic, and decentralized FL and for communication, updating, and aggregation. Regarding dynamic BFL, gaps exist in modeling evolving, drifting, non-stationary, or even unlimited scenarios over time or other dimensions (such as value domain, or state space). In summary, the limited research on BFL is at its early stage and focuses on simple applications of classic and main BL settings and methods in simplified FL scenarios, tasks or applications.

4.2 Research Directions

On one hand, intricate real-world FL scenarios, requirement and applications challenge existing FL theories and systems, which could inspire promising BFL research issues and opportunities. On the other hand, BFL could transform FL and BL research and systems to a new generation. The new-generation BFL could include but may not be limited to: variational, non-IID, hierarchical, weakly-constrained, computation- and communication-efficient, hybrid, and actionable BFL theories and applications, and BFL under complex task, network and data conditions.

Variational FL. Variational deep learning has made significant progress by integrating VI with deep neural learning [Wu *et al.*, 2023]. Variational FL may expand the existing BFL research for (1) complex stochastic client/server conditions or settings; (2) diverse and efficient VI mechanisms in heterogeneous, hierarchical and hybrid FL settings; (3) large-scale and dynamic variational FL; and (4) variational FL under non-IID settings as discussed below.

Non-IID BFL. Original BFL mechanisms, architectures and models are required to address non-IIDnesses [Cao, 2022b] in FL systems. Examples include (1) heterogeneous data structures, distributions, priors and parameterizations across clients, devices and nodes; (2) personalized client requirements, tasks and objectives, and sample/client importance difference; (3) interactive and coupled clients, communities and edge nodes, message sharing between clients; (4) interactions and couplings between clients and server; and (5) nonstationary, evolving, adaptive, and drifting client/server conditions. This goes beyond existing heterogeneous and personalized FL focusing on limited and neutralized heterogeneities of clients without client couplings.

BFL on complex network conditions. Real-life federated networks may involve complex settings or conditions. For example, new BFL theories may need to handle (1) disconnected nodes, unstably connected clients, siloed clients, and hybrid connections with online and offline devices; (2) unavailable, unstable or forgettable clients or nodes, or unseen domains; (3) coupled and interactive networks such as with cross-domain, cross-silo, cross-client or cross-node (client node or edge network) conditions; (4) unaligned or conflicting network conditions, such as with unaligned clients (where clients share inconsistencies such as on features, distributions, or communication), client conflict, domain conflict, or objective conflict (conflicts may exist in various real-life scenarios, e.g., contrary or inconsistent circumstances, distinct objectives, or different evaluation measures); (5) insecure networks such as with data or model poisoning attacks; (6) hierarchical networks, such as client clusters, multiple client communities, hierarchical centralized servers, and mixing decentralized and centralized client/server structures; and (7) mixing asynchronous and synchronous communications.

BFL on complex data characteristics. FL applications often involve complex data, challenging the existing BFL and FL capacity. They include (1) low-quality data such as noisy, irregular and unaligned client data; (2) weak data such as with small, sparse and insufficient evidence; (3) changing data such as with covariate features, feature shift, prior shift, or concept drift; (4) mixed data such as with multiple domains, communities, modalities, structures and distributions, and multi-granular data such as with client and structure granularity; and (5) mixed labelled data with unlabelled to partially labelled clients. These require new BFL theories, e.g., for data-augmented, evolving BNP networks, and mixed-supervision BL networks and optimization for FL.

Weakly-constrained BFL. Existing FL and BFL involve various and often strong constraints, such as on client conditions, privacy, security, resources, client-server interactions, communication, prior and posterior distribution, parameter and message sharing, and optimization. These conflict with the diversified FL reality, where there may be siloed and coupled clients, some client features may be unbalanced and sparse, and data may be mixed, evolving and hybridized. These require substantially new and more flexible BFL theories and models catering for specific requirements.

Computation- and communication-efficient BFL. Both existing FL and BFL models face the significant challenges of high computation and communication costs and low efficiency, although intensive efforts have been made to address these. Computation- and communication-efficient BFL requires more efficient, energy-aware and scalable learning theories, sampling, optimization methods, client updating, server aggregation, parameter and message sharing, and back-propagation mechanisms.

Hybrid BFL. Real-life FL applications may involve (1) mixed features, modalities, and data sources; (2) hybrid priors and distributions of client data; (3) multiple to hybrid FL tasks; (4) mixing centralized and decentralized clients/server; and (5) mixing BFL with learning paradigms, etc. These require new BFL theories, architectures, and mechanisms for

Categories	BL	BFL models	Application	Advantages	Disadvantages
Client-side	FBP	BDP [Triastcyn and Faltings, 2019]	Medical images	Communication efficiency, high accuracy	Strong constraints
	FBP	FDL-PP [Gu <i>et al.</i> , 2021]	Wireless communication	Low complexity, high accuracy	High computational cost, IID
	BNN	pFedBayes [Zhang <i>et al.</i> , 2022]	Finance, medicine	Limited data, non-IID	Strong constraints
	BNN	FedPPD [Bhatt <i>et al.</i> , 2022]	Face perception, medical test	Weak constraints, non-IID	High computational cost
	BO	FTS [Dai <i>et al.</i> , 2020]	Human activity recognition	Communication efficiency, non-IID	Low convergence rate
	BO	FBO for TFP [Zang <i>et al.</i> , 2022]	Traffic flow prediction	Communication efficiency, non-IID	Poor scalability
	BNP	pFedGP [Achituve <i>et al.</i> , 2021]	Health care, legal	Computational efficiency, non-IID	High computational cost
	BNP	FedLoc [Yin <i>et al.</i> , 2020]	Outdoor vehicle navigation	Data privacy, high accuracy	High computational cost, IID
	BNP	FedCor [Tang <i>et al.</i> , 2022]	Precision medicine	Communication efficiency, fast convergence	Limited scenarios, IID
	BNP	PFNM [Yurochkin <i>et al.</i> , 2019]	Health care, finance	Communication efficiency	Limited scenarios, IID
	BNP	FedMA [Wang <i>et al.</i> , 2020]	Fingerprinting	Communication efficiency	High computational cost, IID
	Server-side	BME	FedBE [Chen and Chao, 2021]	Target localization	Deeper neural networks, non-IID
BME		FedPPD [Bhatt <i>et al.</i> , 2022]	Face perception, medical test	Model uncertainty, weak constraints, non-IID	High computational cost
BCL		FOLA [Liu <i>et al.</i> , 2021]	Medical AI	Aggregation error, local forgetting, non-IID	High computational cost
BCL		pFedBayes [Zhang <i>et al.</i> , 2022]	Finance, medicine	Limited data, non-IID	Strong constraints
BPD		FedPA [Al-Shedivat <i>et al.</i> , 2021]	Pedestrian tracking	Computational efficiency, non-IID	Strong constraints
BPD		FedEP [Guo <i>et al.</i> , 2023]	Disease detection	Communication efficiency, non-IID	Strong constraints
BPD		QLSD [Vono <i>et al.</i> , 2022]	Autonomous driving	Communication efficiency, non-IID	Strong constraints
BPD		FOLA [Liu <i>et al.</i> , 2021]	Medical AI	Aggregation error, local forgetting, non-IID	High computational cost
BPD		VIRTUAL [Corinzia <i>et al.</i> , 2019]	Smart keyboards	Communication efficiency, non-IID	High computational cost

Table 1: Categorization and comparison of various methods for Bayesian federated learning (BFL) from a Bayesian perspective.

hybrid Bayesian learning, multi-task BFL, multi-source BFL, multimodal BFL, hybrid FL client updating, server pooling and aggregation, client-server interaction, and optimization methods. Other hybrid BFL areas include hybridizing BFL (1) with other learning systems, such as for Bayesian federated transfer learning, reinforced BFL, ensemble BFL, and BFL for anomaly detection; and (2) with various communication and computing settings, such as compressed BFL, encrypted BFL, asynchronous BFL, blockchained BFL, decentralized BFL, Bayesian federated edge learning, over-the-air BFL, sparse BFL, adaptive BFL, BFL with message sharing across devices or communities, and cross-silo BFL.

Actionable BFL. Actionable BFL requires extra functional and nonfunctional requirements, settings or performance to ensure the actionability [Cao and Zhang, 2006; Cao, 2013; Cao, 2022a] of FL systems in the real world. This requires BFL theories and systems to support functions or capabilities such as fairness, unbiasedness, robustness, resilience, security, safety, responsibility, verifiability, explainability, and ethics. The evaluation of BFL algorithms and systems would have to consider both technical (such as statistical significance) and domain-driven business-oriented (such as impact on business) aspects and measures, and possibly both objective and subjective evaluation measures driven by the domain knowledge and factors, etc. [Cao *et al.*, 2010].

These opportunities could directly or naturally benefit from addressing intrinsic FL issues, utilizing stronger BL mechanisms, or aiming for better capacity and performance. In fact, more exciting opportunities exist in exploring intricate FL demand and challenges in the real world, and focusing on seamlessly integrating BL theories into FL settings.

5 Conclusion

By applying Bayesian learning to the federated learning framework, Bayesian federated learning has become an important learning paradigm to handle various FL challenges and requirements for more robust uncertainty learning. While existing BFL methods exhibit significant progress and potential in learning with limited data and uncertainties, various

technical gaps and hence opportunities remain. Fundamental BFL research is required to handle stochastic, heterogeneous, nonstationary, interactive, hierarchical, imbalanced, unlabeled, personalized, and hybrid challenges and requirements for robust BFL theories and actionable FL applications.

References

- [Abrial, 2009] Jean-Raymond Abrial. Event model decomposition. *Technical report/ETH*, 626, 2009.
- [Achituve *et al.*, 2021] Idan Achituve, Aviv Shamsian, Aviv Navon, Gal Chechik, and Ethan Fetaya. Personalized federated learning with gaussian processes. 34:8392–8406, 2021.
- [Al-Shedivat *et al.*, 2021] Maruan Al-Shedivat, Jennifer Gillenwater, Eric Xing, and Afshin Rostamizadeh. Federated learning via posterior averaging: A new perspective and practical algorithms. In *ICLR*, 2021.
- [Benavoli and de Campos, 2021] Alessio Benavoli and Casio P. de Campos. Bayesian independence test with mixed-type variables. In *DSAA 2021*, pages 1–13. IEEE, 2021.
- [Bhatt *et al.*, 2022] Shreyansh P. Bhatt, Aishwarya Gupta, and Piyush Rai. Bayesian federated learning via predictive distribution distillation. *ArXiv*, abs/2206.07562:1–15, 2022.
- [Blundell *et al.*, 2015] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *ICML*, pages 1613–1622, 2015.
- [Briggs *et al.*, 2020] Christopher Briggs, Zhong Fan, and Peter Andras. Federated learning with hierarchical clustering of local updates to improve training on non-iid data. In *IJCNN*, pages 1–9. IEEE, 2020.
- [Cao and Zhang, 2006] Longbing Cao and Chengqi Zhang. Domain-driven actionable knowledge discovery in the real world. In *PAKDD 2006*, pages 821–830, 2006.
- [Cao *et al.*, 2010] Longbing Cao, Philip S Yu, Chengqi Zhang, and Yanchang Zhao. *Domain Driven Data Mining*. Springer, 2010.

- [Cao, 2013] Longbing Cao. Combined mining: Analyzing object and pattern relations for discovering and constructing complex yet actionable patterns. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, 3(2):140–155, 2013.
- [Cao, 2016] Longbing Cao. Non-iid recommender systems: A review and framework of recommendation paradigm shifting. *Engineering*, 2(2):212–224, 2016.
- [Cao, 2022a] Longbing Cao. Beyond auttml: Mindful and actionable AI and autoai with mind and action. *IEEE Intell. Syst.*, 37(5):6–18, 2022.
- [Cao, 2022b] Longbing Cao. Beyond i.i.d.: non-IID thinking, informatics, and learning. *IEEE Intell Syst*, 37(4):5–17, 2022.
- [Cao, 2022c] Longbing Cao. Decentralized AI: Edge intelligence and smart blockchain, metaverse, web3, and desc. *IEEE Intell Syst*, 37(3):6–19, 2022.
- [Cao, 2022d] Longbing Cao. Non-IID federated learning. *IEEE Intell Syst*, 37(2):14–15, 2022.
- [Chen and Chao, 2021] Hong-You Chen and Wei-Lun Chao. Fed{BE}: Making Bayesian model ensemble applicable to federated learning. In *ICLR*, 2021.
- [Chen and Chao, 2022] Hong-You Chen and Wei-Lun Chao. On bridging generic and personalized federated learning for image classification. In *ICLR*, 2022.
- [Chen et al., 2021] Shengbo Chen, Cong Shen, Lanxue Zhang, and Yuanmin Tang. Dynamic aggregation for heterogeneous quantization in federated learning. *IEEE Trans. Wirel. Commun.*, 20(10):6804–6819, 2021.
- [Chen et al., 2022] Huili Chen, Jie Ding, Eric William Tramel, Shuang Wu, Anit Kumar Sahu, Salman Avestimehr, and Tao Zhang. Self-aware personalized federated learning. In *NeurIPS*, 2022.
- [Corinzia et al., 2019] Luca Corinzia, Ami Beuret, and Joachim M Buhmann. Variational federated multi-task learning. *arXiv preprint arXiv:1906.06268*, 2019.
- [Dai et al., 2020] Zhongxiang Dai, Bryan Kian Hsiang Low, and Patrick Jaillet. Federated bayesian optimization via thompson sampling. 33:9687–9699, 2020.
- [Dinh et al., 2020] Canh T Dinh, Nguyen H Tran, Minh NH Nguyen, Choong Seon Hong, Wei Bao, Albert Y Zomaya, and Vincent Gramoli. Federated learning over wireless networks: Convergence analysis and resource allocation. *IEEE ACM Trans Netw*, 29(1):398–409, 2020.
- [Durmus et al., 2021] Alp Emre Durmus, Zhao Yue, Matas Ramon, Mattina Matthew, Whatmough Paul, and Saligrama Venkatesh. Federated learning based on dynamic regularization. In *ICLR*, 2021.
- [Elgabli et al., 2022] Anis Elgabli, Chaouki Ben Issaid, Amrit Singh Bedi, Ketan Rajawat, Mehdi Bennis, and Vaneet Aggarwal. FedNew: A communication-efficient and privacy-preserving newton-type method for federated learning. In *ICML*, pages 5861–5877, 2022.
- [Fallah et al., 2020] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. 33:3557–3568, 2020.
- [Gershman and Blei, 2012] Samuel J Gershman and David M Blei. A tutorial on bayesian nonparametric models. *J Math Psychol*, 56(1):1–12, 2012.
- [Ghari and Shen, 2022] Pouya M. Ghari and Yanning Shen. Personalized online federated learning with multiple kernels. In *NeurIPS*, 2022.
- [Gu et al., 2021] Hanlin Gu, Lixin Fan, Bowen Li, Yan Kang, Yuan Yao, and Qiang Yang. Federated deep learning with bayesian privacy. *CoRR*, abs/2109.13012, 2021.
- [Guo et al., 2023] Han Guo, Philip Greengard, Hongyi Wang, Andrew Gelman, Yoon Kim, and Eric P Xing. Federated learning as variational inference: A scalable expectation propagation approach. *ICLR*, 2023.
- [Huang et al., 2021] Anbu Huang, Yang Liu, Tianjian Chen, Yongkai Zhou, Quan Sun, Hongfeng Chai, and Qiang Yang. Starfl: Hybrid federated learning architecture for smart urban computing. *ACM Trans Intell Syst Technol*, 12(4):1–23, 2021.
- [Jospin et al., 2022] Laurent Valentin Jospin, Hamid Laga, Farid Boussaid, Wray Buntine, and Mohammed Benamoun. Hands-on bayesian neural networks—a tutorial for deep learning users. *IEEE Comput Intell Mag*, 17(2):29–48, 2022.
- [Kairouz et al., 2021] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Found. Trends Mach. Learn.*, 14(1–2):1–210, 2021.
- [Kassab and Simeone, 2022] Rahif Kassab and Osvaldo Simeone. Federated generalized bayesian learning via distributed stein variational gradient descent. *IEEE Trans. Signal Process.*, 70:2180–2192, 2022.
- [Kendall and Gal, 2017] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *NIPS*, volume 30, page 5580–5590, 2017.
- [Kingma and Welling, 2014] Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. *ICLR*, pages 1–14, 2014.
- [Kotelevskii et al., 2022] Nikita Yurevich Kotelevskii, Maxime Vono, Alain Durmus, and Eric Moulines. Fedpop: A bayesian approach for personalised federated learning. In *NeurIPS*, 2022.
- [Li et al., 2020] Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *IEEE Signal Process Mag*, 37(3):50–60, 2020.
- [Li et al., 2022] Zhize Li, Haoyu Zhao, Boyue Li, and Yuejie Chi. SoteriaFL: A unified framework for private federated learning with communication compression. In *NeurIPS*, 2022.

- [Lin *et al.*, 2022] Shiyun Lin, Yuze Han, Xiang Li, and Zhihua Zhang. Personalized federated learning towards communication efficiency, robustness and fairness. In *NeurIPS*, 2022.
- [Liu *et al.*, 2020] Lumin Liu, Jun Zhang, SH Song, and Khaled B Letaief. Client-edge-cloud hierarchical federated learning. In *ICC*, pages 1–6, 2020.
- [Liu *et al.*, 2021] Liangxi Liu, Feng Zheng, Hong Chen, Guo-Jun Qi, Heng Huang, and Ling Shao. A bayesian federated learning framework with online laplace approximation. *arXiv preprint arXiv:2102.01936*, 2021.
- [Liu *et al.*, 2022] Ken Liu, Shengyuan Hu, Steven Wu, and Virginia Smith. On privacy and personalization in cross-silo federated learning. In *NeurIPS*, 2022.
- [Malinovskiy *et al.*, 2020] Grigory Malinovskiy, Dmitry Kovalev, Elnur Gasanov, Laurent Condat, and Peter Richtarik. From local SGD to local fixed-point methods for federated learning. In *ICML*, pages 6692–6701, 2020.
- [McMahan *et al.*, 2017] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *AISTATS*, pages 1273–1282, 2017.
- [Murphy, 2022] Kevin P. Murphy. *Probabilistic Machine Learning: An introduction*. MIT Press, 2022.
- [Ng and Zhang, 2022] Ignavier Ng and Kun Zhang. Towards federated bayesian network structure learning with continuous optimization. In *AISTATS 2022*, volume 151, pages 8095–8111, 2022.
- [Nguyen *et al.*, 2018] Cuong V Nguyen, Yingzhen Li, Thang D Bui, and Richard E Turner. Variational continual learning. In *ICLR*, 2018.
- [Orbanz and Teh, 2010] Peter Orbanz and Yee Whye Teh. Bayesian nonparametric models. *Encyclopedia of machine learning*, 1, 2010.
- [Pang *et al.*, 2017] Guansong Pang, Longbing Cao, Ling Chen, and Huan Liu. Learning homophily couplings from non-iid data for joint feature selection and noise-resilient outlier detection. In *IJCAI 2017*, pages 2585–2591, 2017.
- [Snell and Zemel, 2021] Jake Snell and Richard Zemel. Bayesian few-shot classification with one-vs-each poly-gamma augmented gaussian processes. In *ICLR*, 2021.
- [Sun *et al.*, 2019] Shengyang Sun, Guodong Zhang, Jiabin Shi, and Roger Grosse. Functional variational Bayesian neural networks. In *ICLR*, 2019.
- [Sun *et al.*, 2020] Wenli Sun, Changgee Chang, and Qi Long. Joint bayesian variable selection and graph estimation for non-linear SVM with application to genomics data. In *DSAA 2020*, pages 315–323. IEEE, 2020.
- [T Dinh *et al.*, 2020] Canh T Dinh, Nguyen Tran, and Josh Nguyen. Personalized federated learning with moreau envelopes. 33:21394–21405, 2020.
- [Tan *et al.*, 2022] Alysa Ziyang Tan, Han Yu, Lizhen Cui, and Qiang Yang. Towards personalized federated learning. *IEEE Trans Neural Netw Learn Syst*, 2022.
- [Tang *et al.*, 2022] Minxue Tang, Xuefei Ning, Yitu Wang, Jingwei Sun, Yu Wang, Hai Li, and Yiran Chen. FedCor: Correlation-based active client selection strategy for heterogeneous federated learning. In *CVPR*, pages 10102–10111, 2022.
- [Triastcyn and Faltings, 2019] Aleksei Triastcyn and Boi Faltings. Federated learning with bayesian differential privacy. In *IEEE BigData*, pages 2587–2596, 2019.
- [Vono *et al.*, 2022] Maxime Vono, Vincent Plassier, Alain Durmus, Aymeric Dieuleveut, and Eric Moulines. QLSD: Quantised langevin stochastic dynamics for bayesian federated learning. In *AISTATS*, pages 6459–6500, 2022.
- [Wang *et al.*, 2020] Hongyi Wang, Mikhail Yurochkin, Yuekai Sun, Dimitris Papailiopoulos, and Yasaman Khazaeni. Federated learning with matched averaging. In *ICLR*, 2020.
- [Wu *et al.*, 2023] Zhangkai Wu, Longbing Cao, and Lei Qi. e-VAE: Evolutionary variational autoencoder. *arXiv preprint arXiv:2301.00011*, 2023.
- [Yang, 2021] Qiang Yang. Toward responsible AI: an overview of federated learning for user-centered privacy-preserving computing. *ACM Trans. Interact. Intell. Syst.*, 11(3-4):32:1–32:22, 2021.
- [Yin *et al.*, 2020] Feng Yin, Zhidi Lin, Qinglei Kong, Yue Xu, Deshi Li, Sergios Theodoridis, and Shuguang Robert Cui. FedLoc: Federated learning framework for data-driven cooperative localization and location data processing. *IEEE Signal Process Mag*, 1:187–215, 2020.
- [Yurochkin *et al.*, 2019] Mikhail Yurochkin, Mayank Agarwal, Soumya Ghosh, Kristjan Greenewald, Nghia Hoang, and Yasaman Khazaeni. Bayesian nonparametric federated learning of neural networks. In *ICML*, pages 7252–7261, 2019.
- [Zang *et al.*, 2022] Lu Zang, Yang Qin, and Ruonan Li. Traffic flow prediction based on federated learning with joint pca compression and bayesian optimization. In *SMC*, pages 3330–3335, 2022.
- [Zhang *et al.*, 2021] Xinwei Zhang, Mingyi Hong, Sairaj Dhople, Wotao Yin, and Yang Liu. FedPD: A federated learning framework with adaptivity to non-iid data. *IEEE Trans. Signal Process.*, 69:6055–6070, 2021.
- [Zhang *et al.*, 2022] Xu Zhang, Yinchuan Li, Wenpeng Li, Kaiyang Guo, and Yunfeng Shao. Personalized federated learning via variational bayesian inference. In *ICML*, pages 26293–26310, 2022.
- [Zong *et al.*, 2021] Huixuan Zong, Qing Wang, Xiaofeng Liu, Yinchuan Li, and Yunfeng Shao. Communication reducing quantization for federated learning with local differential privacy mechanism. In *IEEE/CIC ICC*, pages 75–80, 2021.