Summary of Changes and Authors' Response to Reviewers' Comments

Article: TMLR Paper2081 - "Federated Variational Inference: Towards Improved Personalization and Generalization"

We would like to thank the reviewers for taking the time in reading the manuscript and providing valuable comments contributing to the improvement of the manuscript. In what follows, we present a detailed point-by-point response for all the comments made by the reviewers. At the end, we attach the submitted manuscript with the highlighted changes.

Reviewer BQTq

Comments: *It feels as though the paper is split into two distinct parts. In the first, the authors motivate the use of a hierarchical probabilistic model to handle data heterogeneity across clients, and VI for handling the challenge of inference. The generalisation bounds section is a nice addition; however, my main concern is in the correct handling of the hyper parameters γ and τ. For anything other than γ, τ = 1 (8) does not lower-bound the evidence, so is not an ELBO as described. The authors should make this clear. Further, I suspect that Corollary 1 is only valid for $\gamma = \tau$ also? I could be incorrect on this—clarification from the authors would be appreciated.*

We appreciate the reviewer's insightful question. While Equation (8) is indeed a generalization of ELBO (specifically, ELBO when $\gamma = \tau = 1$), Corollary 1 generalizes this bound to settings where $\gamma, \tau \neq 1$. Note that while Equation (8) may not be specifically a lower bound on the evidence, it is an upper bound on the True risk which is equally useful in practice.

In what follows we prove that the upper bound on the True risk holds for any $\gamma > 0$ and $\tau > 0$. Given Equation (9)'s validity for any $\frac{1}{\eta} > 0$, setting $\frac{1}{\eta} = \min\{\gamma, \tau\}$, allows us to derive:

$$
\begin{aligned}
&\overbrace{\mathsf{E}_{\mathcal{D}}[-\log\big(\mathsf{E}_{q(\theta,\beta^c|X,Y)}[\ell(Y|X,\theta,\beta^c)\big)]]}^{\text{True risk}} \leq \\
&\overbrace{\mathsf{E}_{X,Y}[\mathsf{E}_{q(\theta,\beta^c|X,Y)}[-\log(\ell(Y|X,\theta,\beta^c))]]}^{\text{Empirical risk}} + \frac{1}{\eta}\overbrace{D_{\mathrm{KL}}(q(\theta,\beta^c|X,Y)\|\pi(\theta,\beta^c))}^{\text{KL divergence}} + \frac{1}{\eta}\text{Slack} \\
&= \mathsf{E}_{X,Y}[\mathsf{E}_{q(\theta,\beta^c|X,Y)}[-\log(\ell(Y|X,\theta,\beta^c))]] + \frac{1}{\eta}\overbrace{D_{\mathrm{KL}}(q(\theta|X,Y)\|t(\theta))}^{\text{Global KL divergence}} \\
&+ \frac{1}{\eta}\overbrace{D_{\mathrm{KL}}(q(\beta^c|\theta,X,Y)\|r(\beta^c))}^{\text{Local KL divergence}} + \frac{1}{\eta}\text{Slack} \\
&\leq \mathsf{E}_{X,Y}[\mathsf{E}_{q(\theta,\beta^c|X,Y)}[-\log(\ell(Y|X,\theta,\beta^c))]] + \gamma\overbrace{D_{\mathrm{KL}}(q(\theta|X,Y)\|t(\theta))}^{\text{Global KL divergence}} \\
&+ \tau\overbrace{D_{\mathrm{KL}}(q(\beta^c|\theta,X,Y)\|r(\beta^c))}^{\text{Local KL divergence}} + \frac{1}{\eta}\text{Slack}. \qquad (1)
\end{aligned}
$$

In addressing the reviewer's inquiry, we have included this explanation at the end of Section 4 (highlighted in blue) for further clarity.

In the second section, the authors describe their FedVI algorithm. Although one can piece together the parts of this algorithm to relate it back to the hierarchical model and variational approximation described in Section 3, this connection is far from explicit.

We appreciate the reviewer's feedback regarding the clarity of the connection between the FedVI algorithm (Section 5) and the hierarchical model/variational approximation (Section 3). We acknowledge that this link could be made more explicit.

To address this, we have added the following introductory paragraph at the beginning of Section 5, directly relating the algorithm to the theoretical framework:

The main goal of this section is to go through the details of our primary theoretical assumptions and model architecture for implementing an instance of our proposed hierarchical generative model and evaluating it. We note that the model architecture proposed in this section is one of infinity many architectures that is compatible with our theoretical model. This section will outline how the algorithm's steps correspond to the specific components of the hierarchical model and variational approximation presented in Section 3.

We believe this addition, along with the revised introduction, significantly improves the understanding of how the FedVI algorithm serves as a practical implementation of our theoretical findings.

1. *The likelihood is parameterised by combining the outputs of two different functions, one using global and the other using local parameters. (a) make this clear and (b) how does this combination happen?*

   We thank the reviewer for this suggestion. While there are multiple ways that one *could* merge the predictions, we found that the simplest way was to add them together. This treats the local predictions as modifications to the global predictions in Logit space. To improve clarity for the reviewer, we have added an explanation as a footnote on page 8.

2. *The approximate posterior over local parameters, which itself should be clearly defined, is amortised given a subset of the observed data (it would be nice for connections to be drawn to other methods that use amortised inference, such as VAEs). (a) If this is what makes it "stateless", then this should be made clear. (b) Also, if this is the key contribution, which I think it is, then highlight it.*

   The reviewer's suggestion is greatly appreciated. We agree that the connection to other amortized models such as VAEs should be made clear and have mentioned this connection in the paper. This is indeed the element that makes the model "stateless", since performing this inference during training/evaluation removes the need to maintain a set of local parameters for each client. We had attempted to highlight this benefit of our inference procedure in several places throughout the paper, but had fallen short of considering it a key contribution, since we had deemed it more of a "relative benefit to our approach". Following the reviewer's suggestion, we have further emphasized it by adding the following paragraph to Section 2:

FedVI's statelessness stems from its use of an "amortized posterior inference model." This concept shares similarities with how Variational Autoencoders (VAEs) [1] perform inference. In VAEs, an encoder network maps input data to a latent space. FedVI takes a set of examples and infers the posterior over the local parameters, effectively capturing the personalized features of each client's data. This dynamic encoding allows for continuous adaptation and personalization as new clients join and contribute their data.

(c) I'm not convinced that this is actually doing VI (i.e. targeting the ELBO), as the expectation in (8) is now over the approximate posterior given the support set and the likelihood is only evaluated at a non overlapping query set. This feels closer to the neural process ML objective (e.g. Foong et al. (2020): Meta-learning stationary stochastic process prediction with convolutional neural processes, section 3.2.).

The reviewer's comment provided valuable insight regarding the potential deviation from standard VI and the similarity to the neural process ML objective. We acknowledge that splitting data into non-overlapping support and query sets could indeed compromise the validity of the ELBO bound.

While we initially followed the approach in FedRecon [2], we recognize the importance of adhering to proper VI principles. Therefore, we conducted additional experiments where we combined the support and query sets, effectively avoiding the data split. Encouragingly, these experiments yielded similar performance results, suggesting that the original approach did not significantly impact the overall findings.

(3) How the posterior constructor model operates as a set function on the set of global features of the query set. Something that would help here is clearly defining the space in which the intermediate variables (e.g. $R^g_{k,q}$) exist.

We thank the reviewer for their question regarding the posterior constructor model's operation and the space of intermediate variables. We apologize for the confusion caused by our previous wording.

To clarify, the posterior constructor model does not operate as a set function on the global features of the query set. Its input consists of the first d_g elements (global features) of the embedding of each data sample in the support set. This distinction is important, as the model operates on a batch of individual data embeddings rather than sets of features.

As you suggested, we have now explicitly defined the space of intermediate variables in Section 5. Specifically:

- $R^g_{k,s} \in \mathbb{R}^{d_g}, R^l_{k,s} \in \mathbb{R}^{d_l}$ represent the global and local features, respectively, for the support set of client k.
- $R^g_{k,q} \in \mathbb{R}^{d_g}, R^l_{k,q} \in \mathbb{R}^{d_l}$ represent the global and local features, respectively, for the query set of client k.

Here, d is the embedding size, d_g is is the number of global features, and d_l is the number of local features, with $d = d_l + d_g$.

We believe these clarifications address the reviewer's concerns and provide a more accurate understanding of the model's operation and the relevant variable spaces.

3. *I understand the notion of "stateless" to mean "no local parameters". It would be great as to why this is beneficial in this case. Citations to use cases in the introduction would help, and clarity on why it's an issue to store local parameters vs. computing them using a global model (aren't these then stored locally anyway to obtain the local predictions?).*

   We agree with the reviewer's definition of statelessness, where clients do not retain local parameters. Our method aligns with this by using a latent set of local parameters that are marginalized during prediction, but not directly stored by clients. This stateless approach avoids the "cold start" problem for new or infrequent clients, who might otherwise have poorly initialized local parameters. In contrast, stateful methods can create a participation gap, favoring clients with more training history. Our approach addresses this by treating local parameter estimation as an inference subproblem, enabling personalization without requiring clients to maintain a state.

   While stateful methods can offer advantages in scenarios with consistent client participation, our stateless approach aligns well with the practical realities of many real-world federated learning (FL) deployments. As highlighted in Table 1 of [3], statelessness is a defining characteristic of cross-device FL, emphasizing its importance in this context. Note that in practical cross device FL, clients participate in only a few rounds or less, leading to potentially stale or untrained local parameters that negate the benefits of statefulness. Our method is well-suited for such dynamic FL environments, where maintaining client state is challenging. Notably, most existing personalization attempts ([4–7]) rely on stateful setups, whereas ours does not.

   I'm also somewhat confused by "local parameters remain on clients" in section 3.1—isn't this exactly what you're trying to avoid by being stateless?

   We agree with the reviewer that section 3.1 as written was confusing as to the treatment of "local parameters". We have revised this section to read "Global parameters update at the server end after each training round, while local parameters are deleted after each round." in order to note that we do not maintain their state between rounds.

4. *There is no description of the baseline methods. Are the architectures used the same as the global model? At the bare minimum these should be included in the appendix.*

   We appreciate the reviewer's feedback regarding the baseline method descriptions. In the revised manuscript, we've added the following paragraph as a detailed breakdown of the baseline architectures in Section 5:

   KNN-Per [8] achieves personalized federated learning by combining a global MobileNet-V2 model with local k-nearest neighbors models based on shared data representations, demonstrating improved accuracy and fairness compared *to other methods. FedPA [9] reimagines federated learning as a posterior inference problem, proposing a novel algorithm that utilizes MCMC for efficient local inference and communication, employing CNN models for FEMNIST and ResNet-18 for CIFAR-100. FedEP [10] similarly reformulates federated learning as a variational inference problem, using an expectation propagation algorithm to refine approximations to the global posterior, and also employs CNN models for FEMNIST and ResNet-18 for CIFAR-100. APFL [6] offers a communication-efficient federated learning algorithm that adaptively combines local and global models to learn personalized models, utilizing various models (logistic regression, CNN, MLP) on diverse datasets, including FEMNIST and CIFAR-100. ClusteredFL [7] is designed for clustered users, iteratively estimating user clusters and optimizing their model parameters, demonstrating strong performance in various settings. Finally, FedRep [5] learns a shared representation and unique local heads for each client, using 5-layer CNNs for CIFAR and a 2-layer MLP for FEMNIST, while DITTO [11] introduces a simple personalization mechanism within a multi-task learning framework, utilizing CNNs, logistic regression, and linear SVMs for different datasets.*

5. *"No assumptions are made about clients' data generating distributions". I'm not sure what you mean by this, as you're using a very specific CNN architecture and categorical distribution which indicates that you are making assumptions.*

   We apologize for the confusion. Our intent was to remind the reader that (as mentioned in line 5 of equation 1), that we have not assumed that the data generating distributions for each client are the same. We of course do make assumptions about the support of the predictive and generative distributions, and the dimensionality of the event space. We have clarified that statement to read "We assume that clients' data generating distributions have the same support, but that they can be different from one another." to be more specific about our assumptions.

We would like to thank the reviewer for the constructive comments, which have really helped us to improve the paper.

Reviewer QSyx

Comments:

Strengths: I believe the motivation, both the personalization and the statelessness, for the work is well justified and something that would definitely be of interest for the TMLR audience. Also the connection between you generalization bound in eq. (9) and the ELBO itself is an interesting realization. The empirical evaluation seems to suggest that the proposed model outperforms the existing approach, even with some margin.

Weaknesses in Writing: The writing of the paper needs to improve quite significantly. I do not quite understand for example how the scores computed by the local and global model are "merged" before updating the model. This would be a crucial bit of information for understanding how the local predictions can help in personalization.

Weaknesses in Experiments: So do I understand correctly, that if you set $\tau = 0$ in your experiments, then you don't get any regularization from the local priors? If so, I *guess the proposed setting wouldn't differ from having a single global model (assuming $\tau = 0$)? Now, looking at Fig. 5a, while there are some points where the generalization gap is smaller than with $\tau = 0$, it seems to be quite hard to predict how different $\tau > 0$ values work. Would it be possible to somehow repeat the inference multiple times to get less noisy estimates on how the test errors behave as a function of τ ? I'm mainly worried that since the FEMNIST is most likely the data set that has some strong heterogeneity among the clients, and if we cannot clearly witness the regularization there, then it might be hard to say if the regularization actually has some statistically significant effect.*

1. *Further experiments on whether $\tau = 0$ for the FEMNIST leads to significantly larger generalization gap than $\tau > 0$.*

   We conducted additional experiments, running the FEMNIST scenario five times with different seed values $(0, 10, 20, 30, 40)$. We then calculated the average and variance of both hold-out and participating test accuracy across these runs.

   The results, presented in the revised Figure 1, provide a smoother visualization of the trend. As the figure demonstrates, we still observe a noticeably larger generalization gap for $\tau = 0$ compare to $\tau > 0$ (As the horizontal axis in Figure 1 is semi-logarithmic, test accuracy results of $\tau = 0$ are shown at point $\tau = 10^{-12}$). This supports our initial findings and further emphasizes the importance of the KL divergence term in promoting generalization, even with increased experimental rigor.

   We appreciate the reviewer's insightful suggestion, which has helped us strengthen the robustness of our results and provide a clearer understanding of the impact of τ on model performance.

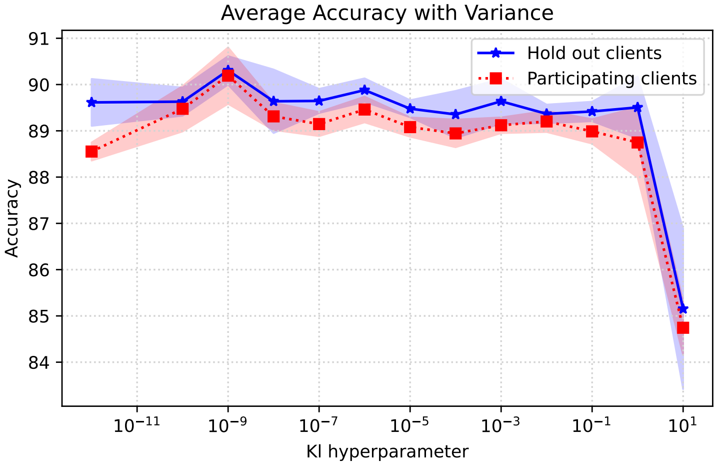


Figure 1: Participating and non-participating test accuracy vs. KL hyperparameter τ on FEMNIST dataset.

2. *Clarify how the merging of local and global predictions is done.*

   We thank the reviewer for this suggestion. While there are multiple ways that one *could* merge the predictions, we found that the simplest way was to add them together. This treats the local predictions as modifications to the global predictions in Logit space. To improve clarity for the reviewer, we have added an explanation as a footnote on page 8.

3. *What τ value was used for the results in Table 1 and Fig. 4?*

   We used $\tau = 10^{-9}$ for FEMNIST, and $\tau = 10^{-3}$ for CIFAR-100 in both table 1 and Figure 4, since each performed best in our experiments. In order to make this clearer, we have mentioned these values in the captions of Table 1 and Figure 4.

4. *Page 4: "the best approximation", in general this is not true since the quality of the approximation is completely dependent on the chosen surrogate distribution.*

   In order to address the reviewer's concern, we have changed this statement to read "This minimization process provides the best approximation for the intractable posterior distribution, under the chosen family of surrogates". This highlights that, while this minimization may not yield the best approximation to the true posterior over all distributions, it does produce the closest surrogate to the true posterior (in terms of the KL divergence between them), out of the chosen family of surrogates.

5. *Page 4: The formula in the end: I guess this is actually negative ELBO?*

   We thank the reviewer for their comment. We have revised the paper as suggested.

6. *Corollary 1: Please clarify what is the "slight generalization" over the Germain et al. 2016. To me, the bound in equation (9) looks very much the same as the bound in the prior work.*

   We appreciate the reviewer's question regarding Corollary 1. We acknowledge that the bound in Equation (9) appears similar to the bound in [12]. However, our work offers a slight generalization in the following way:

   - Non-IID Data: While Theorem 3 in [12] explicitly assumes IID empirical data samples, our corollary extends the applicability to non-IID cases. This is particularly relevant in federated learning scenarios where data distributions can vary across clients.
   - Proof Adaptation: In the proof of our corollary, we demonstrate that the core argument in [12] does not strictly require the IID assumption for empirical data. This allows us to derive a similar bound even when data samples originate from different distributions.

   We believe this generalization, though seemingly subtle, is an important contribution as it broadens the theoretical foundations of PAC-Bayes bounds in the context of federated learning and other non-IID settings.

About the local parameters. On the 7th point of the listing in page 7, you say that "Both local and global parameters get updated through back propagation ...". So I guess as local parameters you mean the μ_k and σ_k, and not the β (which is titled as local parameters on eq (18))?

We appreciate the reviewer's attention to detail regarding the terminology of local parameters. The reviewer is correct that in Equation (18), β_k is referred to as the local parameters. Our method constructs a distribution over these parameters, using (μ_k and σ_k) and then samples β_k from that distribution. However, all of these are inferred from the global reconstruction model. It is the global parameters of that model which are updated through backpropagation. We have changed the wording of the seventh point on page 8 in the revised manuscript to make this explicit. Note that by updating the global parameters, we are implicitly updating the local parameters (since we use the reconstruction model to infer them).

7. *When you say that "The local and global predictions are merged to get the predictions", how is the merging done?*

   We thank the reviewer for this suggestion. While there are multiple ways that one *could* merge the predictions, we found that the simplest way was to add them together. This treats the local predictions as modifications to the global predictions in Logit space. To improve clarity for the reviewer, we have added an explanation as a footnote on page 8.

   The samples fed to both classifiers are the same (just different parts of the feature vector) right?

   Yes, this is correct. The same samples are fed to both classifiers, but different portions of the feature vector are used. As detailed in the "Data partitioning" section (pages 7-8), we split the feature vector into global and local components. In our experiments, we found that using a larger proportion of global features (80%) compared to local features (20%) yielded the best performance. Specifically, when the embedding model's last layer has a dimension of $d = 128$, the first 102 features are considered global, and the remaining 26 are local.

8. *I guess Figure 4 is not referred anywhere in the main text?*

   We thank the reviewer for bringing this to our attention. While we initially provided a brief explanation in the caption, we understand that this was insufficient. To address this, we have added the following explanation to the "Evaluation Results and Discussion" section:

   Figure 4 illustrates the non-participating test accuracy on FEMNIST ($\tau = 10^{-9}$) and CIFAR-100 ($\tau = 10^{-3}$) over 1500 *rounds of training, providing a visual representation of the results reported in Table 1.*

9. *About Figure 5: If I have understood correctly, the τ parameter controls the level of local regularization through scaling the KL divergence between the variational posterior and the prior for β parameter. Could you clarify, would the $\tau = 0$ correspond to the actual ELBO objective? Or is there some scaling issue*

between the data and the prior that might lead to $\tau \neq 0$ being the correct scale? If not, then it would be interesting to have more discussion why the test performance gap seems to somewhat high for the $\tau = 0$ in the FEMNIST case, as I would imagine it should lead to better generalization that the $\tau < 1$.

We appreciate the reviewer's insightful question regarding the role of the τ parameter and its relationship to the ELBO objective. That is correct that τ controls the level of local regularization by scaling the KL divergence between the variational posterior and the prior for the β parameter.

Our objective function in Equation (8) is indeed a generalization of the ELBO, where setting $\tau = \gamma = 1$ recovers the standard ELBO. Setting $\tau = 0$ effectively removes the KL divergence term, resulting in maximum likelihood estimation (MLE).

This distinction explains the higher generalization gap observed for $\tau = 0$ in both FEMNIST and CIFAR-100 experiments (Figure 5). While MLE can sometimes lead to overfitting, the KL divergence term in the ELBO acts as a regularizer, promoting better generalization. Therefore, values of $\tau > 0$, which maintain the KL divergence term, generally exhibit superior performance and result is smaller generalization gaps.

We hope this explanation clarifies the behavior of τ and its impact on generalization.

10. *Appendix C: in the first line of eq. (24), it seems like you are writing $\ell(Y|X) = E_{q(\theta,\beta^c|X,Y)}[\ell(Y|X,\theta,\beta^c)]$. I assume that the $\ell(Y|X)$ and $\ell(Y|X,\theta,\beta^c)$ are the evidence and likelihood. Is this really true for variational q? Wouldn't you have that $\ell(Y|X) = E_{\pi(\theta,\beta^c)}[\ell(Y|X,\theta,\beta^c)]$? Or if q is not the variational approximation, can you please clarify what it is?*

    We appreciate the reviewer's insightful question regarding Equation (24) in Appendix C. Throughout this paper $\ell(Y|X)$ stands for marginalized likelihood over the surrogate posterior so $\ell(Y|X) = E_{q(\theta,\beta^c|X,Y)}[\ell(Y|X,\theta,\beta^c)]$ is correct. To clarify our notation, we use $\nu(X)$ to denote the prior data generating distribution. Therefore, we would show the evidence by $\nu(Y|X) = E_{\pi(\theta,\beta^c)}[\ell(Y|X,\theta,\beta^c)]$.

11. *Appendix C: what is the $||$ after the 2nd $\leq$ (close to equation 24)? Is it supposed to be KL?*

    We thank the reviewer for spotting the typo in Appendix C. The symbol "$||$" after the second "$\leq$" should indeed be "D_{KL}". We have corrected this in the updated manuscript.

12. *Appendix C: I believe the last expectation in the Donsker Varadhan inequality should have a log around it (talking about the expression after "After that we use the Donsker-Varadhan inequality which says").*

    We appreciate the reviewer's careful attention to detail. The typo has been corrected in the revised version.

13. *Some of the references look a bit odd. There are a lot of uncapitalized letters, as well as missing venues.*

    We thank the reviewer for pointing out the reference formatting inconsistencies. We apologize for this oversight and have carefully reviewed our bib file to identify and correct the issues. In the updated manuscript, all references now include venues, ensuring clarity and adherence to formatting guidelines.

14. *Possibly relevant cite: Han Guo, Philip Greengard, Hongyi Wang, Andrew Gelman, Yoon Kim, Eric P. Xing: Federated Learning as Variational Inference: A Scalable Expectation Propagation Approach. ICLR 2023*

    We are grateful for the reviewer for suggesting a relevant reference. We have incorporated this reference in Section 5 and added FedEP to the baseline comparisons in Table 1, strengthening the evaluation of our method.

The reviewer's insightful comments have significantly strengthened our paper, and we are grateful for their feedback.

Reviewer WAV2

Comments:

Strengths: (i) The paper is well-written and well-motivated in general. Improving the effectiveness of FL is a promising research direction. (ii) The presented idea behind FedVI is easy to follow. Both theoretical and experimental results are presented to justify the method.

Weaknesses: (i) It is not clear about the extra computation and communication incurred by FedVI. By looking at the major algorithm box, FedVI seems to be much more complex compared to FedAvg. The extra computation and communication costs have not been explicitly studied in the experiments. (ii) The experimental scales are too small. It would be essential to demonstrate the effectiveness of FedVI on larger-scale datasets, e.g., ImageNet. Some similar baseline methods are missing, e.g., [1] https://arxiv.org/abs/2010.05273 [2] https://arxiv.org/abs/2302.04228.

1. *Adding experiments on the extra computation and communication overheads of FedVI. And study the end-to-end wall clock time comparisons between FedVI and other baseline methods.*

   We found the reviewer's suggestion to be particularly valuable to analyze the computation and communication overheads of FedVI compared to other baseline methods.

   To assess the complexity of FedVI algorithm as compared to FedAvg, we conducted experiments where we removed the posterior reconstruction and local classifier components from FedVI, effectively turning it into FedAvg. We then compared the runtime of both algorithms on the FEMNIST dataset for 200 rounds using an NVIDIA A100 GPU (all the other hyperparameters are similar to what is reported in the main manuscript).

   The results presented in Figures 2 and 3 demonstrate comparable training and evaluation runtimes for both FedAvg and FedVI. FedVI achieves an average training runtime of 71.35 seconds per round and evaluation runtime of 18.96 seconds per round. In comparison, FedAvg training requires 71.74 seconds per round and evaluation takes 18.39 seconds per round on average. While FedVI incorporates additional components compared to FedAvg, our experiments demonstrate that these complexities do not significantly impact runtime. This is because the added components do not significantly contribute to the computational overhead, as the primary bottleneck is the embedding model. Similarly, although FedVI requires communicating slightly more global parameters (posterior constructor parameters), this does not notably affect runtime. These results underscore that FedVI's personalization benefits come at a minimal cost in terms of computational and communication efficiency.

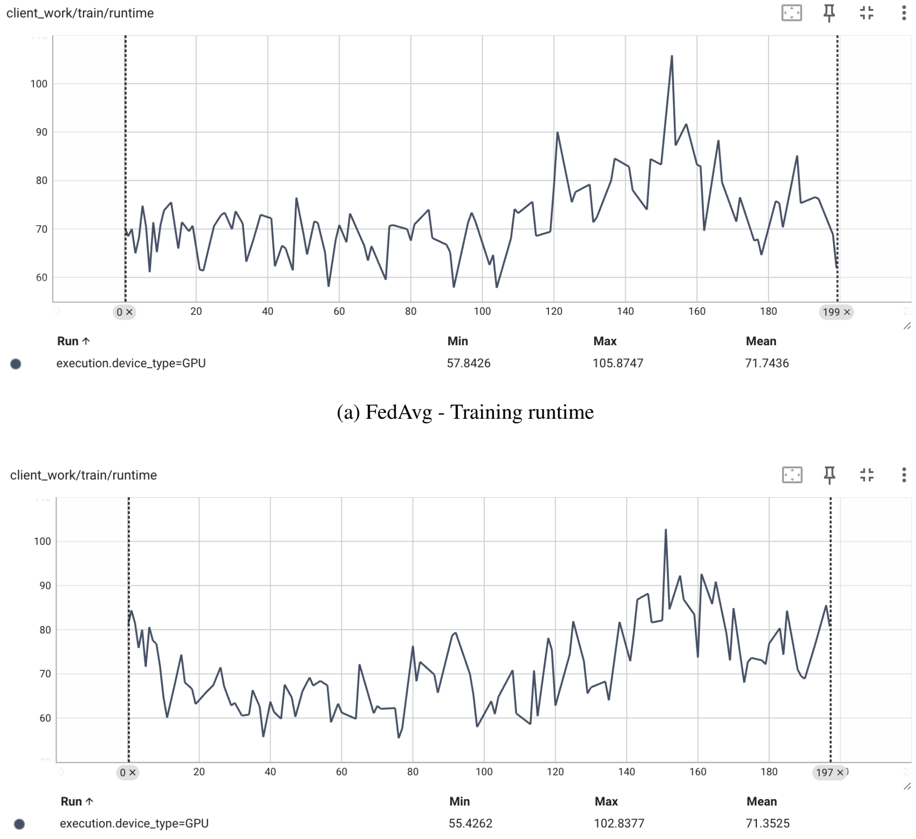
We acknowledge the reviewer's request for comparisons with other baselines. While a direct runtime comparison is challenging without re-implementing those methods, we can offer some insights. Since the posterior constructor model and local classifier do not represent computational bottlenecks for FedVI, its runtime is expected to be similar to or potentially even less than that of other baseline methods. This is because those baselines typically involve personalization components that may introduce additional computational or communication overhead compared to FedAvg.

We hope this analysis provides valuable insights into FedVI's computational and communication efficiency.

2. *The experimental scales are too small. It would be essential to demonstrate the effectiveness of FedVI on larger-scale datasets, e.g., ImageNet.*

   We thank the reviewer for suggesting the evaluation of FedVI on ImageNet and other large-scale datasets. While we acknowledge the general value of such experiments, we believe that applying FedVI to ImageNet might not provide the desired insights for the following reasons:

   - Federated Nature: ImageNet is not inherently structured for federated learning (FL). Adapting it to an FL setting would require simulating an artificial client-based distribution, which might not accurately reflect real-world FL scenarios. This could potentially distort the evaluation of our personalized method, as its effectiveness relies on the natural heterogeneity and distribution shifts present in genuine FL data. Therefore, we believe that evaluating on datasets specifically designed for FL or those with inherent client-based structures (such as FEMNIST) would provide a more reliable demonstration of our method's capabilities.
   - Baseline Comparisons: Our primary focus here lies in personalized federated learning, where existing baselines are rarely evaluated on datasets of that scale. This makes direct comparisons with other FL methods on ImageNet challenging. We believe the chosen datasets, specifically FEMNIST, effectively demonstrate FedVI's effectiveness in a personalized FL setting and exhibit inherent heterogeneity and client-based distribution shifts, which represent the core challenges that FedVI addresses.



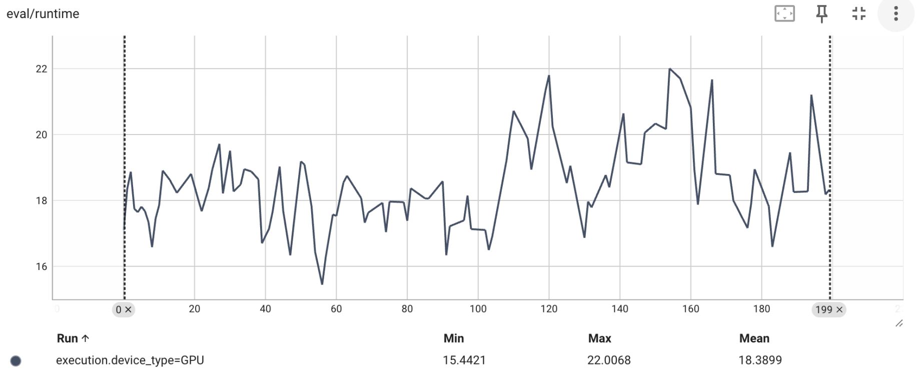
Run ↑	Min	Max	Mean
● execution.device_type=GPU	57.8426	105.8747	71.7436

(a) FedAvg - Training runtime

Run ↑	Min	Max	Mean
● execution.device_type=GPU	55.4262	102.8377	71.3525

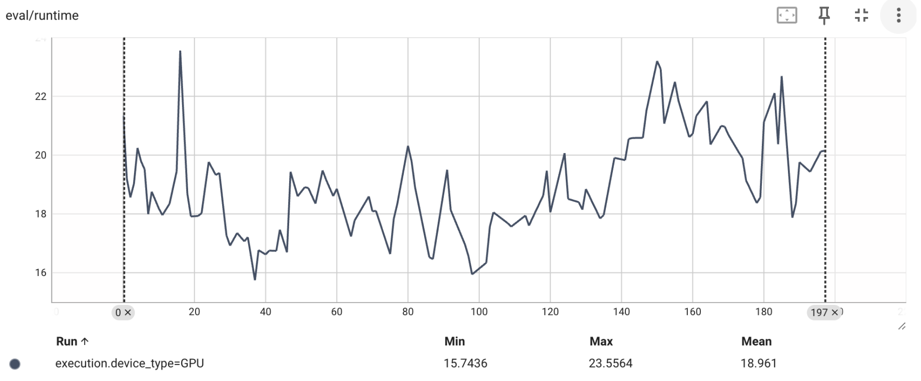
(b) FedVI - Training runtime

Figure 2: Training runtime of FedVI and FedAvg experiments on FEMNIST dataset.



	Run ↑	Min	Max	Mean
●	execution.device_type=GPU	15.4421	22.0068	18.3899

(a) FedAvg - Evaluation runtime



	Run ↑	Min	Max	Mean
●	execution.device_type=GPU	15.7436	23.5564	18.961

(b) FedVI - Evaluation runtime

Figure 3: Evaluation runtime of FedVI and FedAvg experiments on FEMNIST dataset.

3. *Add comparisons between FedVI and baseline methods like:*

   [1] https://arxiv.org/abs/2010.05273 and [2] https://arxiv.org/abs/2302.04228.

   Following the reviewer's helpful suggestion, we have integrated the recommended references on FedEP and FedPA into Section 5. Additionally, we have included these methods as baselines in Table 1 to provide a more comprehensive evaluation.

We would like to thank the reviewer for the helpful comments, which have greatly improved the paper.

References

[1] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[2] K. Singhal, H. Sidahmed, Z. Garrett, S. Wu, K. Rush, and S. Prakash, "Federated reconstruction: Partially local federated learning," 2021. [Online]. Available: https://arxiv.org/abs/2102.03448

[3] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings *et al.*, "Advances and open problems in federated learning," *Foundations and Trends® in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021.

[4] X. Zhang, Y. Li, W. Li, K. Guo, and Y. Shao, "Personalized federated learning via variational bayesian inference," in *International Conference on Machine Learning*. PMLR, 2022, pp. 26 293–26 310.

[5] L. Collins, H. Hassani, A. Mokhtari, and S. Shakkottai, "Exploiting shared representations for personalized federated learning," in *International conference on machine learning*. PMLR, 2021, pp. 2089–2099.

[6] Y. Deng, M. M. Kamani, and M. Mahdavi, "Adaptive personalized federated learning," *arXiv preprint arXiv:2003.13461*, 2020.

[7] A. Ghosh, J. Chung, D. Yin, and K. Ramchandran, "An efficient framework for clustered federated learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 19 586–19 597, 2020.

[8] O. Marfoq, G. Neglia, R. Vidal, and L. Kameni, "Personalized federated learning through local memorization," in *International Conference on Machine Learning*. PMLR, 2022, pp. 15 070–15 092.

[9] M. Al-Shedivat, J. Gillenwater, E. Xing, and A. Rostamizadeh, "Federated learning via posterior averaging: A new perspective and practical algorithms," *arXiv preprint arXiv:2010.05273*, 2020.

[10] H. Guo, P. Greengard, H. Wang, A. Gelman, Y. Kim, and E. P. Xing, "Federated learning as variational inference: A scalable expectation propagation approach," *arXiv preprint arXiv:2302.04228*, 2023.

[11] T. Li, S. Hu, A. Beirami, and V. Smith, "Ditto: Fair and robust federated learning through personalization," in *International conference on machine learning*. PMLR, 2021, pp. 6357–6368.

[12] P. Germain, F. Bach, A. Lacoste, and S. Lacoste-Julien, "Pac-bayesian theory meets bayesian inference," in *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29. Curran Associates, Inc., 2016. [Online]. Available: https://proceedings.neurips.cc/paper/2016/file/84d2004bf28a2095230e8e14993d398d-Paper.pdf

Federated Variational Inference: Towards Improved Personalization and Generalization

Anonymous authors
Paper under double-blind review

Abstract

Conventional federated learning algorithms train a single global model by leveraging all participating clients' data. However, due to heterogeneity in client generative distributions and predictive models, these approaches may not appropriately approximate the predictive process, converge to an optimal state, or generalize to new clients. We study personalization and generalization in stateless cross-device federated learning setups assuming heterogeneity in client data distributions and predictive models. We first propose a hierarchical generative model and formalize it using Bayesian Inference. We then approximate this process using Variational Inference to train our model efficiently. We call this algorithm *Federated Variational Inference (FedVI)*. We use PAC-Bayes analysis to provide generalization bounds for FedVI. We evaluate our model on FEMNIST and CIFAR-100 image classification and show that FedVI beats the state-of-the-art on both tasks.

1 Introduction

Federated Learning (FL) (McMahan et al., 2016) allows training machine learning models on decentralized datasets, avoiding the need to aggregate data on a central server due to privacy concerns. In FL, the central server oversees a global model distributed to clients who conduct local training, and the model updates are aggregated to iteratively improve the global model.

In simple and idealized settings, FL can approximate centralized training with similar theoretical guarantees, as seen in FedSGD (McMahan et al., 2016). However, real-world cross-device FL scenarios, such as those in (Reddi et al., 2020; Wang et al., 2021), often diverge from these ideal conditions. Practical FL implementations involve multiple local training steps to minimize communication overhead. Client participation is typically uneven, with some contributing more data and others not participating at all. Additionally, the non-Independently and Identically Distributed (non-IID) nature of client datasets, stemming from distinct data generation processes, challenges theoretical guarantees, leads to performance disparities between participating and non-participating clients (Yuan et al., 2022), and complicates training high-performing models in practical FL setups.

Modern approaches address this challenge by either modifying the local loss to converge to a global solution (Li et al., 2020) or using personalized models to handle local distribution shifts (Zhang et al., 2022). Approaches for personalization have often focused on stateful FL setups, where clients are revisited throughout training and thus can update a locally stored model (Karimireddy et al., 2019; Wang et al., 2021). However, many production scenarios are effectively stateless, since individual clients only rarely contribute to training, and local models may be either stale or non-existent. Few studies have concentrated on personalization in this context. Those that have (Singhal et al., 2021), require clients to possess labeled examples for personalization.

This paper explores personalization in stateless cross-device FL setups and introduces Federated Variational Inference (FedVI), an algorithm which utilizes Variational Inference (VI) to enable models to generalize and personalize across diverse client data, even for untrained clients. The key contributions encompass (i) proposing a hierarchical generative model rooted in mixed effects models for cross-device federated setups, (ii) offering generalization bounds through Probably Approximately Correct (PAC)-Bayes analysis, (iii) introducing FedVI algorithm, inspired by the theoretical approach, which provides a simplified experimental approximation and can be implemented by the existing FL frameworks, and (iv) demonstrating the superior performance of FedVI on two federated datasets, FEMNIST and CIFAR-100, compared to previous state-of-the-art methods.

2 Related Work

Bayesian FL: To tackle statistical heterogeneity in FL, various studies have employed Bayesian methods to incorporate domain knowledge and aid convergence. Early attempts (Thorgeirsson & Gauterin, 2020; Chen & Chao, 2020) focused on model aggregation, either to retain uncertainty in model parameters, or to weight parameter updates proportional to performance. Zhang et al. (2022) instead attempts to use a Bayesian Neural Network (BNN) approximated with VI to train a global model using a Kullback–Leibler (KL) regularizer which induces convergence similar to the proximal term in FedProx (Li et al., 2020). While their local models can, in principle, personalize by deviating from the global model, they realistically require stateful settings with significant labeled data on clients in order to do so. Kotelevskii et al. (2022) casts personalized FL as mixed effects regression, and attempts to model the inherent heterogeneity in this setting explicitly using Stochastic Gradient Langevin Dynamics (Welling & Teh, 2011). Our proposed method assumes a similar generative process to Kotelevskii et al. (2022) but instead uses VI to efficiently infer the posterior, as well as place a bound on the predictive risk to induce generalization to new clients (Germain et al., 2016).

Stateful FL: There is a rich body of literature on personalization in cross device FL (Corinzia et al., 2019; Ghosh et al., 2020; Chen & Chao, 2021; Collins et al., 2021; Deng et al., 2020; Li et al., 2021; Hassan et al., 2023). Many previous methods focus on stateful settings, where local parameters are stored on clients and maintained across training rounds. However, as emphasized in Table 1 of (Kairouz et al., 2021), statelessness is a key characteristic of cross-device FL, highlighting its practical significance. Therefore, we focus on stateless settings, where maintaining up-to-date local states on each client is not feasible. This is similar to the setting considered by (Marfoq et al., 2022), who uses K-nearest neighbors to account for client distributional shift. While this is a robust means of dealing with both input and output distributional shift, it requires clients to possess labeled examples for every class (which is unrealistic in real-world setups), and cannot be used outside of classification problems. FedVI's statelessness stems from its use of an *amortized posterior inference model*. This concept shares similarities with how Variational Autoencoders (VAEs) (Kingma & Welling, 2013b) perform inference. In VAEs, an encoder network maps input data to a latent space. FedVI takes a set of examples and infers the posterior over the local parameters, effectively capturing the personalized features of each client's data. This dynamic encoding allows for continuous adaptation and personalization as new clients join and contribute their data.

Meta Learning: There is a significant amount of prior work that studies connections between personalized FL and Model-Agnostic Meta-Learning (MAML) approaches (Finn et al., 2017; Singhal et al., 2021; Fallah et al., 2020; Collins et al., 2021; Lin et al., 2020; Chen et al., 2018). The main idea behind these works is to find an initial global shared model that the existing or new clients can adapt to their own dataset by performing a few steps of gradient descent with respect to their local data. FedRecon (Singhal et al., 2021) is also motivated by MAML and considers a partially local federated learning setting, where only a subset of model parameters (known as global parameters) will be aggregated and trained globally for fast reconstruction of the local parameters. Our work can be considered as an extension of FedRecon. Unlike this work, we also provide a means of reconstructing local parameters [1] without access to labeled data.

3 Methods

3.1 Hierarchical Generative Model

Let us consider a stateless cross-device federated setup with multiple clients and a central server, where randomly selected client subsets participate in each training round. In this setup, we categorize each client's model parameters as global (θ) and local (β_k for $k \in [c]^2$) parameters, with c representing the total number of clients. Global parameters update at the server end after each training round, while local parameters are

[1] The detailed procedure for reconstructing the local parameters can be found in Section 5.

[2] In this paper we represent the set of $\{1, \ldots, c\}$ by $[c]$.

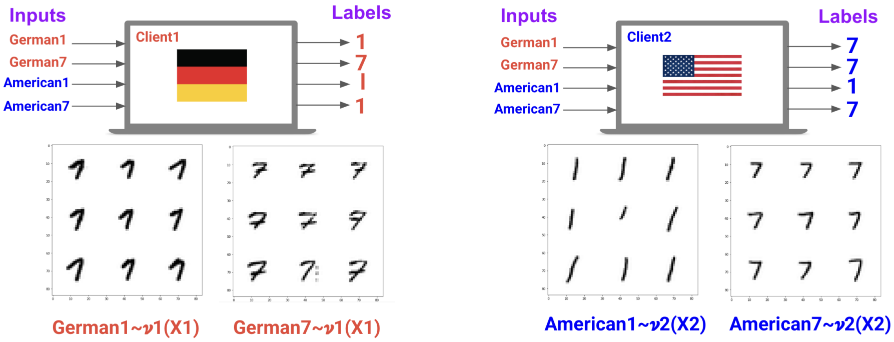


Figure 1: Illustration of diverse data generation and predictive models in cross-device FL.

deleted after each round. Global parameters are drawn from the prior distribution $t(\Theta)$, while each client's local parameters are independent samples from the local prior $r(B_k)$. Additionally, data may not exhibit IID characteristics among clients, *i.e.*, $x_{ik} \sim \nu_k(X_k)$ for $i \in [n_k]$ and $k \in [c]$, where n_k is the total number of data samples at client k. Moreover, each client may have a distinct predictive distribution. Although all clients share the same likelihood distribution family $\ell(Y_k|f(\theta, \beta_k, x_{ik}))$, the distribution varies based on β_k, making it different for each client.

The above setup is a prototypical example of a mixed effects model (Demidenko, 2013), commonly employed for predicting a continuous random variable using multiple independent factors, including both random and fixed, and incorporating repeated measurements from the same observational unit. Mixed effects models (Demidenko, 2013) have a well-established foundation. By framing our setup within this context, we can leverage existing theoretical insights in this field. To summarize, we propose the following hierarchical data generating process:

$$\begin{aligned}
&\theta \sim t(\Theta) \\
&\text{for } k \in [c]: \\
&\quad \beta_k \sim r(B_k) \\
&\quad \text{for } i \in [n_k]: \\
&\qquad x_{ik} \sim \nu_k(X_k) \\
&\qquad y_{ik} \sim \ell(Y_k|f(\theta, \beta_k, x_{ik})),
\end{aligned} \tag{1}$$

where $f : \Phi \times \mathcal{B}_k \times \mathcal{X}_k \to \mathcal{Z}_k$ is a deterministic function (e.g., DNN) mapping what we know to the latent space $\mathcal{Z}_k$, which is the parameter space of our distribution over outcomes, $\ell(.)$.

For a more intuitive grasp of varying data generation processes and predictive distributions, consider the Federated EMNIST dataset (FEMNIST; Figure 1), where each client's dataset consists of numbers and letters handwritten by that client. Each client's input data reflects their unique writing style; for instance, a German client may include a horizontal middle bar when writing sevens, whereas an American client may not. Likewise, the German client may add a hood to the number 1, while the American client may not. This describes the difference in data generating distributions. This also illustrates that each client may have different predictive distributions: the American client may see the German's 1 as a 7, while the German client may see the American's 1 as a lowercase "l". Thus their predictive distributions are in direct conflict with each other. A purely global model cannot accommodate this diversity and must incorporate some level of local adjustments to accurately represent the data generation process. Our proposed algorithm explicitly assumes this data generating process. Note that this assumption reduces in special cases to existing FL setups, such as IID predictive distributions ($r(B_k) = \delta(B_k - \beta)$), or IID data generating processes ($\nu_k(X_k) = \nu(X_k)$). In the following section, we detail how we use VI to efficiently infer the model parameters.

3.2 Training Objective

In this section, our goal is to present a step-by-step definition of the objective function that is meant to be minimized throughout the training process. We begin by calculating the estimated probability density function of labels given input data, denoted as $\hat{p}(\{y^{n_k}\}^c) \overset{\text{def}}{=} p(\{y^{n_k}\}^c|\{x^{n_k}\}^c)$, following a similar marginalization approach as (Watanabe, 2018):

$$\hat{p}(\{y^{n_k}\}^c) \overset{\text{def}}{=} \int_\theta \int_{\beta_c} \cdots \int_{\beta_1} p(\theta, \beta^c|\{y^{n_k}, x^{n_k}\}^c)\ell(\{y^{n_k}\}^c|f(\theta, \{\beta_k, x^{n_k}\}^c)), \tag{2}$$

where $\beta^c \overset{\text{def}}{=} \{\beta_k\}^c \overset{\text{def}}{=} \{\beta_k : k \in [c]\}$, $x^{n_k} \overset{\text{def}}{=} \{x_i : i \in [n_k]\}$, $y^{n_k} \overset{\text{def}}{=} \{y_i : i \in [n_k]\}$, $\{x^{n_k}\}^c \overset{\text{def}}{=} \{x_{ik} : i \in [n_k], k \in [c]\}$, and $\{y^{n_k}\}^c \overset{\text{def}}{=} \{y_{ik} : i \in [n_k], k \in [c]\}$.

Therefore, for calculating $\hat{p}(\{y^{n_k}\}^c)$ it is required to calculate the posterior probability of model parameters given the training data which is equal to:

$$p(\theta, \beta^c|\{y^{n_k}, x^{n_k}\}^c) = \frac{p(\theta, \beta^c, \{y^{n_k}\}^c|\{x^{n_k}\}^c)}{p(\{y^{n_k}\}^c|\{x^{n_k}\}^c)}. \tag{3}$$

Assuming that the prior distribution of the global parameters, $t(\theta)$, the prior distribution of the local parameters, $r(\beta_k)$, and the likelihood distribution of each client, $\ell(y^{n_k}|f(\theta, \beta_k, x^{n_k}))$, are independent we calculate the numerator of Equation 3 as:

$$\begin{aligned} p(\theta, \beta^c, \{y^{n_k}\}^c|\{x^{n_k}\}^c) &= p(\theta, \{\beta_k, \{y_{ik}\}_{i\in[n_k]}\}_{k\in[c]}|\{x_{ik}\}_{k\in[c], i\in[n_k]}) \\ &= t(\theta) \prod_{k\in[c]} r(\beta_k) \prod_{k\in[c]} \prod_{i\in[n_k]} \ell(y_{ik}|f(\theta, \beta_k, x_{ik})) \\ &= t(\theta) \prod_{k\in[c]} \left(r(\beta_k) \prod_{i\in[n_k]} \ell(y_{ik}|f(\theta, \beta_k, x_{ik})) \right) \\ &= t(\theta) r(\beta^c) \ell(\{y^{n_k}\}^c|f(\theta, \beta^c, \{x^{n_k}\}^c)). \end{aligned} \tag{4}$$

Moreover, the denominator of Equation 3 can be written as:

$$p(\{y^{n_k}\}^c|\{x^{n_k}\}^c) = \int_\theta \int_{\beta_c} \cdots \int_{\beta_1} p(\theta, \beta^c, \{y^{n_k}\}^c|\{x^{n_k}\}^c). \tag{5}$$

Unfortunately this integral is not only infeasible to compute, but also mathematically intractable. Consequently, this makes the whole posterior intractable.

To address the problem of the intractable posterior distribution, a tractable surrogate distribution, denoted as $q(\theta, \beta^c|\{y^{n_k}, x^{n_k}\}^c)$, is approximated using VI. By formulating a specific lower bound on the marginal distribution known as the evidence lower bound (ELBO), which is equivalent to the KL divergence between the posterior and surrogate distributions (Equation 6), the best surrogate distribution can be obtained by minimizing the ELBO. This minimization process provides the best approximation for the intractable posterior distribution, $p(\theta, \beta^c|\{y^{n_k}, x^{n_k}\}^c)$, under the chosen family of surrogates. The notation $D_{\text{KL}}(q\|p)$ represents the KL divergence between two distributions p and q, and detailed derivations of Equation 6 are available in Appendix A.

$$-\log p(\{y^{n_k}\}^c|\{x^{n_k}\}^c) \leq \min_q D_{\text{KL}}(q(\theta, \beta^c|\{y^{n_k}, x^{n_k}\}^c)\|p(\theta, \beta^c, \{y^{n_k}\}^c|\{x^{n_k}\}^c)). \tag{6}$$

By asserting factorization, we define the surrogate as a parametric distribution as:

$$q(\theta, \beta^c|\{y^{n_k}, x^{n_k}\}^c) \overset{\text{def}}{=} q_\lambda(\theta|\{y^{n_k}, x^{n_k}\}^c) \prod_{k\in[c]} q_\lambda(\beta_k|\theta, y^{n_k}, x^{n_k}) \overset{\text{def}}{=} q_\lambda(\theta|\{y^{n_k}, x^{n_k}\}^c) q_\lambda(\beta^c|\theta, \{y^{n_k}, x^{n_k}\}^c), \tag{7}$$

where λ is the parameter set that uniquely defines the surrogate distribution. Therefore, the objective function for training the proposed hierarchical model is a generalization of the negative ELBO (specifically, negative ELBO when $\gamma = \tau = 1$), which can be written as follows using the definition of KL divergence, logarithm properties, and the multiplication rule in probability.

$$\begin{aligned}\mathcal{J}(\lambda;\gamma,\tau) &= D_{\mathrm{KL}}(q_\lambda(\theta,\beta^c|\{y^{n_k},x^{n_k}\}^c)\|p(\theta,\beta^c,\{y^{n_k}\}^c|\{x^{n_k}\}^c))\\ &= \sum_{k\in[c]}\sum_{i\in[n_k]} \overbrace{\mathbb{E}_{q_\lambda(\theta|\{y^{n_k},x^{n_k}\}^c)q_\lambda(\beta_k|\theta,y^{n_k},x^{n_k})}\left[-\log \ell(y_{ik}|f(\theta,\beta_k,x_{ik}))\right]}^{\text{Per Datum Expected Loss}}\\ &\quad+\underbrace{\gamma D_{\mathrm{KL}}(q_\lambda(\theta|\{y^{n_k},x^{n_k}\}^c)\|t(\theta))}_{\text{Global Regularizer}}+\sum_{k\in[c]}\tau\,\underbrace{\mathbb{E}_{q_\lambda(\theta|\{y^{n_k},x^{n_k}\}^c)}[D_{\mathrm{KL}}(q_\lambda(\beta_k|\theta,y^{n_k},x^{n_k})\|r(\beta_k))]}_{\text{Local Regularizer}},\end{aligned} \tag{8}$$

where γ, τ, $t(\theta), r(\beta_k)$, and the functional form of $q_\lambda(\theta,\beta^c|\{y^{n_k},x^{n_k}\}^c)$ are left as hyper parameters. The details of this derivation are provided in Appendix B. In the following section we explain how minimizing this objective function is equivalent to minimizing an upper bound on the generalization error.

4 Generalization Bounds

As mentioned earlier, we utilize a generalization of the negative ELBO as our objective function to train the hierarchical model. Minimizing this function ideally reduces the training dataset error (empirical risk). However, our primary aim is to minimize the error on unseen datasets (generalization error or true risk) for better generalization. To achieve this, we conduct a PAC-Bayes analysis, leveraging the results presented in Theorem 3 of (Germain et al., 2016). We introduce a slightly generalized version of this theorem in the form of the following corollary, enabling us to compute a generalization bound for the true risk of our model, under the assumption of non-IID empirical data samples.

Corollary 1 *Given a distribution $\mathcal{D}$ over $\mathcal{X}\times\mathcal{Y}$, a hypothesis set $\mathcal{F}=\{\theta,\beta^c\}$, a loss function $\ell:\mathcal{F}\times\mathcal{X}\times\mathcal{Y}\to\mathbb{R}$, a prior distribution $\pi(\Theta,B^c)=t(\Theta)r(B^c)$ over $\mathcal{F}$, a $\delta\in(0,1]$ and a real number $\eta>0$, with probability at least $1-\delta$ over the choice of $(\{x^{n_k}\}^c,\{y^{n_k}\}^c)\overset{\text{def}}{=}(X,Y)\sim\mathcal{D}$, for any $q(.)$ on $\mathcal{F}$ we have:*

$$\begin{aligned}&\overbrace{\mathbb{E}_{\mathcal{D}}[-\log\left(\mathbb{E}_{q(\theta,\beta^c|X,Y)}[\ell(Y|X,\theta,\beta^c))\right]]}^{\textit{True risk}}\le\\ &\overbrace{\mathbb{E}_{X,Y}[\mathbb{E}_{q(\theta,\beta^c|X,Y)}[-\log(\ell(Y|X,\theta,\beta^c))]]}^{\textit{Empirical risk}}+\frac{1}{\eta}\Bigg[\overbrace{D_{\mathrm{KL}}(q(\theta,\beta^c|X,Y)\|\pi(\theta,\beta^c))}^{\textit{KL divergence}}\\ &\underbrace{+\log\left(\tfrac{1}{\delta}\mathbb{E}_{X,Y}\left[\mathbb{E}_{\pi(\theta,\beta^c)}\left[\exp\left(\eta\mathbb{E}_{\mathcal{D}}[-\log(\ell(Y|X,\theta,\beta^c))]-\eta\mathbb{E}_{X,Y}[-\log(\ell(Y|X,\theta,\beta^c))]\right)\right]\right]\right)}_{\textit{Slack term}}\Bigg].\end{aligned}\tag{9}$$

Where $\mathbb{E}_{X,Y}[\log(\ell(Y|X,\theta,\beta^c))]=\frac{1}{\sum_{k=1}^{c}n_k}\sum_{k=1}^{c}\sum_{i=1}^{n_k}[\log(\ell(y_{ik}|x_{ik},\theta,\beta_k))]$ and $\mathbb{E}_{\mathcal{D}}[.]=\mathbb{E}_{(X,Y)\sim\mathcal{D}}[.]$.

Sketch of Proof: This corollary's proof closely follows Theorem 3 in Germain et al. (2016). We establish it using Jensen's inequality, Donsker-Varadhan change of measure inequality, and Markov's inequality. Additional details can be found in Appendix C.

Having obtained the generalization bound in Equation 9, we observe that it equals the negative ELBO (Equation 8) (for $\eta = 1$) plus a constant slack term, unrelated to the surrogate posterior distributions. Note that given Equation 9 validity for any $\eta > 0$, setting $\frac{1}{\eta}=\min\{\gamma,\tau\}$ allows us to consider Equation 8 plus a slack term as an upper bound on the True risk. Consequently, as long as this slack term remains finite, minimizing Equation 8 with respect to the surrogate distribution is equivalent to minimizing the generalization error with respect to the surrogate distribution. Thus we conclude that, assuming a finite slack term and with probability greater than $1-\delta$, minimizing Equation 8 should improve the generalization of our model.

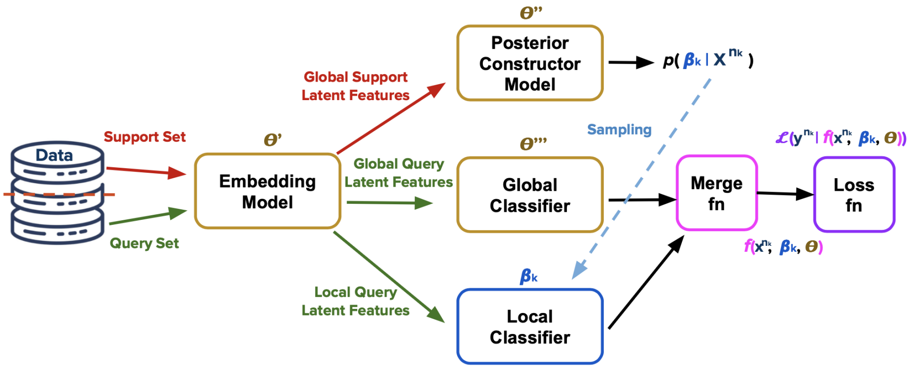


Figure 2: Our proposed model architecture implementing FedVI.

5 Implementation and Experimental Evaluation

The main goal of this section is to go through the details of our primary theoretical assumptions and model architecture for implementing an instance of our proposed hierarchical generative model and evaluating it. We note that the model architecture proposed in this section is one of infinity many architectures that is compatible with our theoretical model. This section will outline how the algorithm's steps correspond to the specific components of the hierarchical model and variational approximation presented in Section 3.

Distributions: For the prior distribution of the local parameters, we assume a normal distribution with zero mean and variance equal to that given by the initialization scheme (e.g. Glorot & Bengio, 2010; Glorot et al., 2011; He et al., 2015). We assume that clients' data generating distributions have the same support, but that they can be different from one another. We use a categorical distribution as our likelihood, where the logits generated by a deep neural network parameterized by θ and β_k (described below). To simplify implementation, we use a point estimate for the global posterior. This is equivalent to assuming the hyper parameter of the global KL divergence is equal to zero, *i.e.,* in Equation 8 we have $\gamma = 0$. Moreover, to make sure that the KL divergence between the global posterior and the global prior, $D_{\mathrm{KL}}(q_\lambda(\theta|\{y^{n_k}, x^{n_k}\}^c)\|t(\theta))$, is finite we assume that the global posterior is a very narrow normal distribution, but still finite, while the global prior can be any finite function.

Tasks: We evaluate FedVI algorithm on two different datasets, FEMNIST[3] (Caldas et al., 2019) (62-class digit and character classification) and CIFAR-100[4] (Krizhevsky et al., 2009) (100-class classification). FEMNIST is particularly relevant since it has a naturally different data generative distribution for each client. Although CIFAR-100 data is synthetically partitioned using a hierarchical Latent Dirichlet Allocation (LDA) process (Li & McCallum, 2006) and distributed among clients, we evaluate FedVI on this dataset as well to show the superiority of our method on a more complicated classification task.

Model Architecture: There are infinitely many model architectures which could implement our method. The architecture that we chose in our experiments is illustrated in Figure 2 and summarised in Algorithm 1. The mathematical notations that are used in both Figure 2 and Algorithm 1 are as follows:

[3] `https://www.tensorflow.org/federated/api_docs/python/tff/simulation/datasets/emnist/load_data`
[4] `https://www.tensorflow.org/federated/api_docs/python/tff/simulation/datasets/cifar100/load_data`

$$\begin{aligned}
D_k &\overset{\text{def}}{=} \{x^{n_k}, y^{n_k} : x^{n_k} \in \mathcal{X}_k,\ y^{n_k} \in \mathcal{Y}_k,\ k \in [c]\} && \text{(input dataset of client } k\text{)} && (10)\\
\mathcal{X}_k &\overset{\text{def}}{=} \mathbb{R}^{i \times i \times j} && \text{(input space; whitened images)} && (11)\\
\mathcal{Y}_k &\overset{\text{def}}{=} [\zeta] && \text{(label space)} && (12)\\
\mathbf{E}_{\theta'}(.) &: \mathcal{X}_k \rightarrow \mathbb{R}^d && \text{(embedding model; relu-convnet with dropout)} && (13)\\
\mathbf{P}_{\theta''}(.) &: \mathbb{R}^{d_g} \rightarrow \mathbb{R}^{(2 \cdot d_l + 1) \cdot |\mathcal{Y}_k|} && \text{(posterior constructor model; relu-mlp)} && (14)\\
\mathbf{G}_{\theta'''}(.) &: \mathbb{R}^{d_g} \rightarrow \mathbb{R}^{|\mathcal{Y}_k|} && \text{(global classifier; one dense layer)} && (15)\\
\mathbf{L}_{\beta_k}(.) &: \mathbb{R}^{d_l} \rightarrow \mathbb{R}^{|\mathcal{Y}_k|} && \text{(local classifier; one dense layer)} && (16)\\
\theta &= \theta' \cup \theta'' \cup \theta''' && \text{(global parameters)} && (17)\\
\beta_k & && \text{(local parameters of client } k\text{),} && (18)
\end{aligned}$$

where for FEMNIST we have $i = 28$, $j = 1$, and $|\mathcal{Y}_k| = \zeta = 62$, and for CIFAR-100 $i = 32$, $j = 3$, and $|\mathcal{Y}_k| = \zeta = 100$, for $k \in [c]$. For both datasets the embedding size $d = 128$ and the number of local and global features are equal to $d_l = 26$ and $d_g = 102$, respectively.

Our proposed model architecture consists of four separate modules: an embedding model, $\mathbf{E}_{\theta'}(.)$, which encodes the input as a vector, a posterior reconstruction model, $\mathbf{P}_{\theta''}(.)$, which predicts the posterior over local parameters, a classifier parameterized by global parameters, $\mathbf{G}_{\theta'''}(.)$, and a classifier implemented by local parameters, $\mathbf{L}_{\beta_k}(.)$, generated by sampling from the reconstructed posterior. The global parameters serve the purpose of classifying input data samples by considering their global features shared among all clients. On the other hand, the local parameters play a distinct role in refining the classification outcome by accounting for the unique local features specific to each individual client. Our model follows the stateless definition outlined in Table 1 of (Kairouz et al., 2021), eliminating the necessity to retain prior client states for parameter updates. Clients are not required to store updated global parameters; instead, the server aggregates and transmits averaged updates for upcoming rounds. Furthermore, clients can avoid the need to store updated local parameters by employing the posterior constructor model in each round to reconstruct the local parameter distribution, allowing them to derive local parameters through sampling from this reconstructed posterior distribution.

Implementation: We implement our FedVI algorithm in TensorFlow Federated (TFF) and scale up the implementation to NVIDIA Tesla V100 GPUs for hyperparameter tuning. For FEMNIST dataset with 3400 clients we consider the first 20 clients as non-participating users which are held-out in training to better measure generalization as in (Yuan et al., 2022). At each round of training we select 100 clients uniformly at random without replacement, but with replacement across rounds. For CIFAR-100 with 500 training clients, we set the data of the first 10 clients as held-out data and select 50 clients uniformly at randomly at each round. We train FedVI algorithm on both FEMNIST and CIFAR-100 for 1500 rounds and at each round of training we divide both datasets into mini-batches of 256 data samples and used mini-batch gradient descent algorithm to optimize the objective function. The training procedure for each client k at round t, outlined in Algorithm 1, is as follows. Further details regarding each step are explained subsequently:

1. Each client k partitions its input data, D_k, over the batch dimension into support and query sets, $D_{k,s}$ and $D_{k,q}$, using the data split function, $S(.)$. Similar to FedRecon (Singhal et al., 2021), the support set is used to reconstruct the local parameters and the query set is used to make predictions. Note that the support set we use can be unlabeled, and that the two sets need not be disjoint. However, we use disjoint sets in our experiments since (Singhal et al., 2021) found that it improved their model performance.

2. Both support and query sets are fed into the embedding model, $\mathbf{E}_{\theta'}(.)$, to extract vector representations of the data, *i.e.,* $R_{k,s} \in \mathbb{R}^d$ and $R_{k,q} \in \mathbb{R}^d$.

3. The representation for both support and query sets are further split over their features axis into global and local features, *i.e.,* $(R^g_{k,s} \in \mathbb{R}^{d_g}, R^l_{k,s} \in \mathbb{R}^{d_l})$ and $(R^g_{k,q} \in \mathbb{R}^{d_g}, R^l_{k,q} \in \mathbb{R}^{d_l})$, for $d = d_g + d_l$, using the feature split function $F(.)$, as illustrated in Figure 3.

Algorithm 1 FedVI Training

Input: set of global parameters θ, data split function $S(.)$, feature split function $F(.)$, embedding model $\mathbf{E}_{\theta'}(.)$, posterior constructor model $\mathbf{P}_{\theta''}(.)$, global classifier $\mathbf{G}_{\theta'''}(.)$, local classifier $\mathbf{L}_{\beta_k}(.)$, merge function $f(.)$, client update algorithm $U(.)$.

```
Server Executes:
$\theta^{(0)} \leftarrow$ (initialize $\theta$)
for each round t do
    $\mathcal{S}^{(t)} \leftarrow$ (randomly sample $c$ clients)
    for each client $k \in \mathcal{S}^{(t)}$ in parallel do
        $(\Delta_k^{(t)}, n_k) \leftarrow$ ClientUpdate($k$, $\theta^{(t)}$)
    end for
    $n = \sum_{k \in \mathcal{S}^{(t)}} n_k$
    $\theta^{(t+1)} \leftarrow \theta^{(t)} + \alpha_s \sum_{k \in \mathcal{S}^{(t)}} \frac{n_k}{n} \Delta_k^{(t)}$
end for

ClientUpdate:
$(D_{k,s}, D_{k,q}) \leftarrow S(D_k)$
$R_{k,s} \leftarrow \mathbf{E}_{\theta'}(x^{n_{k,s}}, \theta'^{(t)})$
$R_{k,q} \leftarrow \mathbf{E}_{\theta'}(x^{n_{k,q}}, \theta'^{(t)})$
$(R^g_{k,s}, R^l_{k,s}) \leftarrow F(R_{k,s})$
$(R^g_{k,q}, R^l_{k,q}) \leftarrow F(R_{k,q})$
$(\mu_k, \sigma_k) \leftarrow \mathbf{P}_{\theta''}(R^g_{k,s}, \theta''^{(t)})$
$\beta_k^{(t)} \leftarrow$ sample($\mathcal{N}(\mu_k, \sigma_k)$)
$O^g_k \leftarrow \mathbf{G}_{\theta'''}(R^g_{k,q}, \theta'''^{(t)})$
$O^l_k \leftarrow \mathbf{L}_{\beta_k}(R^l_{k,q}, \beta_k^{(t)})$
$\theta_k^{(t)} \leftarrow U(f(O^g_k, O^l_k), y^{n_{k,q}})$
$\Delta_k^{(t)} \leftarrow \theta_k^{(t)} - \theta^{(t)}$
$n_k \leftarrow |D_{k,q}|$
return $(\Delta_k^{(t)}, n_k)$ to the server
```

4. The global features of the support set, $R^g_{k,s} \in \mathbb{R}^{d_g}$, are used to reconstruct the mean and variance of the local posterior, *i.e.,* $(\mu_k \in \mathbb{R}^{d_l \cdot |\mathcal{Y}_k|}, \sigma_k \in \mathbb{R}^{d_l \cdot |\mathcal{Y}_k|})$, through the posterior constructor model, $\mathbf{P}_{\theta''}(.)$. The local parameters, $\beta_k^{(t)}$, are generated by sampling from this posterior.

5. The global features of the query set, $R^g_{k,q} \in \mathbb{R}^{d_g}$, are passed to the global classifier, $\mathbf{G}_{\theta'''}(.)$, to get the global predictions, $O^g_k \in \mathbb{R}^{|\mathcal{Y}_k|}$, and the local features of the query set, $R^l_{k,q} \in \mathbb{R}^{d_l}$, and local parameters, $\beta_k^{(t)}$, are passed to the local classifier, $\mathbf{L}_{\beta_k}(.)$, to get the local modifications to the global predictions, $O^l_k \in \mathbb{R}^{|\mathcal{Y}_k|}$.

6. The local and global predictions are merged[5] to get the predictions. The log-likelihood is then computed between these predictions and labels and added to the KL divergence between local posterior and prior.

7. Global parameters get updated through back propagation over the loss function that is calculated in the previous step. This indirectly updates the local parameters, as they are inferred from the posterior constructor model, which is parameterized by the global parameters. Then the local update of the global parameters, $\Delta_k^{(t)}$, along with the number of query data samples at client k, n_k, are returned to the server.

8. The server aggregates all client updates and calculates the global update of the global parameters, $\theta^{(t+1)}$, and shares them with all clients $k \in \mathcal{S}^{(t+1)}$ for the next round of training.

Data Partitioning: First we note that for both FEMNIST and CIFAR-100 datasets, at each epoch we consider the first 50% of each mini-batch as the support set and the other 50% as the query set (*i.e*, for a mini-batch with 256 data samples the first 128 samples belong to the support set and the rest belong to query set). For the global-local features split, we found that using a larger number of global features (80%) than local features (20%) performed best. More specifically, in these experiments that the dimension of the last layer of the embedding model is equal to $d = 128$, the first 102 features are considered as the global features and the rest of 26 features are local features.

[5] While there are multiple ways that one could merge the predictions, we found that the simplest way was to add them together. This treats the local predictions as modifications to the global predictions in Logit space.

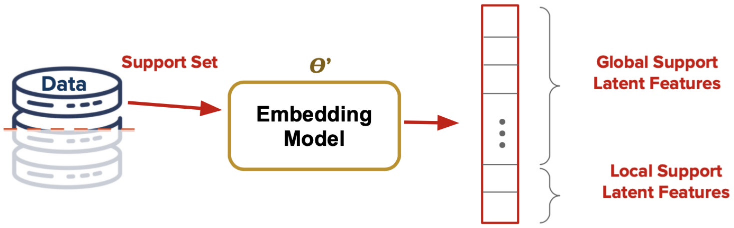


Figure 3: An illustration of the division of the data into support and query sets, as well as the division into global and local features.

Embedding Model: In our experiments the embedding model, $\mathbf{E}_{\theta'}(.)$, is a relu convnet. For FEMNIST experiment we consider the convolutional model with 2 convolution layers that is described in Table 4 of (Reddi et al. (2020)) paper (without the top layer) and is parameteraized by the global parameters. the detailed structure of this embedding model is as the following.

For FEMNIST: $\mathbf{E}_{\theta'}(.)$ = *conv(32)* → *relu* → *conv(64)* → *relu* → *maxpool(2,2)* → *dropout(0.25)* → *flatten* → *dense(128)* → *dropout(0.5)*

We choose a convolutional embedding model for CIFAR-100 as well, which is similar to FEMNIST embedding model, but having 5 convolution layers instead. The detailed structure is as follows.

For CIFAR-100: $\mathbf{E}_{\theta'}(.)$ = *conv(32)* → *relu* → *conv(64)* → *relu* → *conv(128)* → *relu* → *conv(256)* → *relu* → *conv(512)* → *relu* → *maxpool(2,2)* → *dropout(0.25)* → *flatten* → *dense(128)* → *dropout(0.5)*

Posterior Constructor Model: The posterior constructor model, $\mathbf{P}_{\theta''}(.)$, is an MLP with three (dense) layers that takes the global features of the output of $\mathbf{E}_{\theta'}(.)$ as input and generates mean, variance, and bias of the posterior.

For both FEMNIST and CIFAR-100: $\mathbf{P}_{\theta''}(.)$ = *dense(256)* → *relu* → *dense(256)* → *relu* → *dense*$((2 \times 26 + 1) \times |\mathcal{Y}_k|)$

Global and Local Classifiers: For both FEMNIST and CIFAR-100 experiments global classifier is one dense layer with $|\mathcal{Y}_k|$ units and no activation function, parameterized by the global parameters, and the local classifier is one dense layer similar to the global classifier, but parameterized by the local parameters.

Optimizers: We use Stochastic Gradient Descent (SGD) for our client optimizer and SGD with momentum for the server optimizer for all experiments (Reddi et al., 2020). We set the client learning rate equal to 0.03 for CIFAR-100 and 0.02 for FEMNIST dataset, and server learning rate equal to 3.0 with momentum 0.9 for both FEMNIST and CIFAR-100 datasets.

Evaluation Results and Discussion: We evaluate our proposed FedVI algorithm against state-of-the-art personalized FL method, KNN-Per (Marfoq et al., 2022), as well as FedPA (Al-Shedivat et al., 2020), FedEP (Guo et al., 2023) (using highest reported values), FedAvg+ (Chen & Chao, 2021), ClusteredFL (Ghosh et al., 2020), DITTO (Li et al., 2021), FedRep (Collins et al., 2021), APFL (Deng et al., 2020), and FedAvg (McMahan et al., 2016). Results for the baseline methods (except for FedPA and FedEP) are taken from (Marfoq et al., 2022).

KNN-Per (Marfoq et al., 2022) achieves personalized federated learning by combining a global MobileNet-V2 model with local k-nearest neighbors models based on shared data representations, demonstrating improved accuracy and fairness compared to other methods. FedPA (Al-Shedivat et al., 2020) reimagines federated learning as a posterior inference problem, proposing a novel algorithm that utilizes MCMC for efficient local inference and communication, employing CNN models for FEMNIST and ResNet-18 for CIFAR-100. FedEP (Guo et al., 2023) similarly reformulates federated learning as a variational inference problem, using an expectation propagation algorithm to refine approximations to the global posterior, and also employs CNN models for FEMNIST and ResNet-18 for CIFAR-100. APFL (Deng et al., 2020) offers a communication-efficient federated learning algorithm that adaptively combines local and global models to learn personalized models, utilizing various models (logistic regression, CNN, MLP) on diverse datasets, including FEMNIST

Table 1: Test accuracy of the participating/non-participating clients. FedVI results are reported for $\tau = 10^{-9}$ for FEMNIST, and $\tau = 10^{-3}$ for CIFAR-100.

Dataset	FedAvg	FedAvg+	ClusteredFL	DITTO	FedRep	APFL	FedPA	FedEP	KNN-Per	**FedVI**
FEMNIST	83.4/83.1	84.3/84.2	83.7/83.2	84.3/83.9	85.3/85.4	84.1/84.2	87.3/NA	86.6/NA	88.2/88.1	**90.3/90.6**
CIFAR-100	47.4/47.1	51.4/50.8	47.2/47.1	52.0/52.1	53.2/53.5	51.7/49.1	46.3/NA	50.7/NA	55.0/56.1	**59.1/58.7**

and CIFAR-100. ClusteredFL (Ghosh et al., 2020) is designed for clustered users, iteratively estimating user clusters and optimizing their model parameters, demonstrating strong performance in various settings. Finally, FedRep (Collins et al., 2021) learns a shared representation and unique local heads for each client, using 5-layer CNNs for CIFAR and a 2-layer MLP for FEMNIST, while DITTO (Li et al., 2021) introduces a simple personalization mechanism within a multi-task learning framework, utilizing CNNs, logistic regression, and linear SVMs for different datasets.

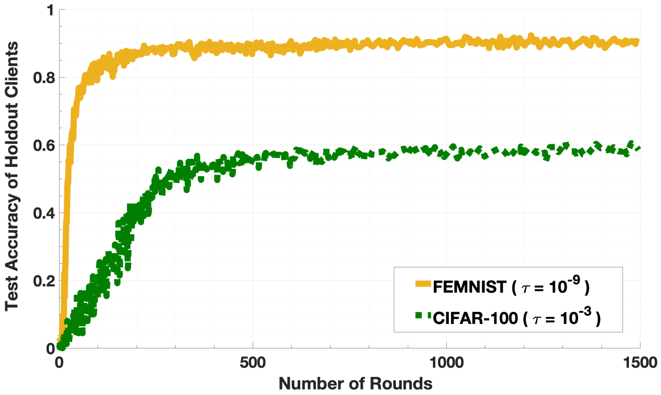
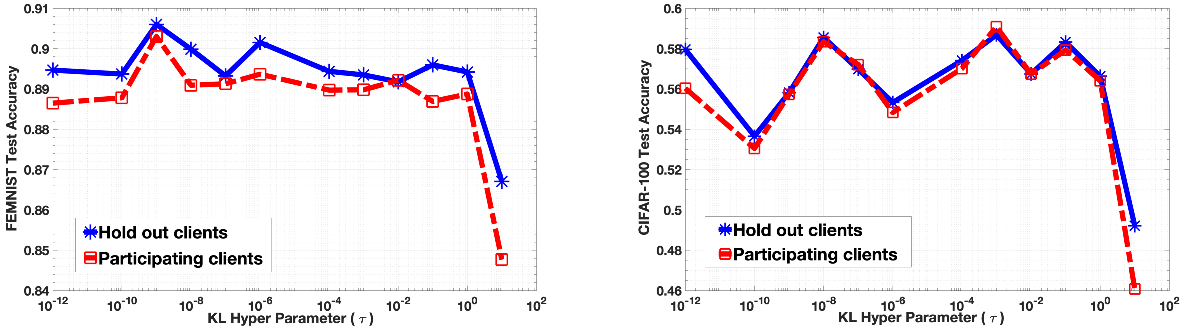


Figure 4: Non-participating test accuracy of FEMNIST ($\tau = 10^{-9}$) and CIFAR-100 ($\tau = 10^{-3}$) over 1500 rounds of training.

The performance of FedVI algorithm and other methods on the local test dataset of each client (unseen data at training) are provided in Table 1 for participating and non-participating (completely unseen during training) clients. All of the reported values are average weighted accuracy with weights proportional to local dataset sizes. To ensure the robustness of our reported results for FedVI, we average test accuracy across the last 100 rounds of training. Figure 4 illustrates the non-participating test accuracy on FEMNIST ($\tau = 10^{-9}$) and CIFAR-100 ($\tau = 10^{-3}$) over 1500 rounds of training, providing a visual representation of the results reported in Table 1.

Figure 5a shows the average test accuracy over the last 100 FEMNIST training rounds for a range of KL hyperparameter τ, from 10^{-12} to 10 (As the horizontal axis of both figures in Figure 5 are semi-logarithmic, test accuracy results of $\tau = 0$ are shown at point $\tau = 10^{-12}$). Notably, $\tau = 10^{-9}$ outperforms others, achieving higher accuracy with a smaller generalization gap compared to $\tau = 0$.

Figure 5b displays the average test accuracy over the last 100 rounds in CIFAR-100, with varying KL hyperparameter τ. Notably, $\tau = 10^{-3}$ achieves the highest accuracy for both participating and non-participating clients. Comparing $\tau = 0$ to other values ($\tau \neq 0$) reveals that minimizing KL divergence reduces the gap in participation test accuracy, as anticipated. Note that our objective function in Equation 8 is indeed a generalization of the ELBO, where setting $\tau = \gamma = 1$ recovers the standard ELBO. Setting $\tau = 0$ effectively removes the KL divergence term, resulting in maximum likelihood estimation (MLE). While MLE can sometimes lead to overfitting, the KL divergence term in the ELBO acts as a regularizer, promoting better generalization. Therefore, values of $\tau > 0$, which maintain the KL divergence term, generally exhibit superior performance and result is smaller generalization gaps. Furthermore, comparing this figure to Figure 5a, it's evident that the difference in test accuracy between $\tau = 0$ and $\tau = 10^{-9}$ in the FEMNIST experiment is significantly larger than the difference between $\tau = 0$ and $\tau = 10^{-3}$ in the CIFAR-100 experiment. This suggests that minimizing KL divergence is more critical for FEMNIST than for CIFAR-100. One possible explanation is that in FEMNIST, each client's data generation distribution naturally differs, while in CIFAR-100, data is synthetically partitioned and distributed among clients.



(a) FEMNIST

(b) CIFAR-100

Figure 5: Participating and non-participating test accuracy vs. KL hyperparameter τ.

6 Conclusion and Future Work

This work addresses personalization in stateless cross-device federated setups through the introduction of FedVI, a novel algorithm grounded in mixed effects models and trained using VI. We establish generalization bounds for FedVI through PAC-Bayes analysis, present a novel architecture, and implement it. Evaluation on FEMNIST and CIFAR-100 datasets demonstrates that FedVI outperforms state-of-the-art methods in both cases. It is worth noting that in this paper, we employed a narrow normal distribution as the posterior for global parameters. However, in future research, we intend to explore more generalized distributions to enhance the modeling capabilities. Additionally, the model architecture presented in Figure 2 is just one of several possible architectures that align with our theoretical hierarchical model. In upcoming work we will focus on refining these architectures to optimize performance and explore their potential for achieving even better results.

References

Maruan Al-Shedivat, Jennifer Gillenwater, Eric Xing, and Afshin Rostamizadeh. Federated learning via posterior averaging: A new perspective and practical algorithms. *arXiv preprint arXiv:2010.05273*, 2020.

Sebastian Caldas, Sai Meher Karthik Duddu, Peter Wu, Tian Li, Jakub Konečný, H. Brendan McMahan, Virginia Smith, and Ameet Talwalkar. Leaf: A benchmark for federated settings, 2019.

Fei Chen, Mi Luo, Zhenhua Dong, Zhenguo Li, and Xiuqiang He. Federated meta-learning with fast convergence and efficient communication. *arXiv: Learning*, 2018. URL `https://api.semanticscholar.org/CorpusID:209376818`.

Hong-You Chen and Wei-Lun Chao. Fedbe: Making bayesian model ensemble applicable to federated learning. *arXiv preprint arXiv:2009.01974*, 2020.

Hong-You Chen and Wei-Lun Chao. On bridging generic and personalized federated learning for image classification. *arXiv preprint arXiv:2107.00778*, 2021.

Liam Collins, Hamed Hassani, Aryan Mokhtari, and Sanjay Shakkottai. Exploiting shared representations for personalized federated learning. In *International conference on machine learning*, pp. 2089–2099. PMLR, 2021.

Luca Corinzia, Ami Beuret, and Joachim M Buhmann. Variational federated multi-task learning. *arXiv preprint arXiv:1906.06268*, 2019.

Eugene Demidenko. *Mixed models: theory and applications with R*. John Wiley & Sons, 2013.

Yuyang Deng, Mohammad Mahdi Kamani, and Mehrdad Mahdavi. Adaptive personalized federated learning. *arXiv preprint arXiv:2003.13461*, 2020.

Alireza Fallah, Aryan Mokhtari, and Asuman E. Ozdaglar. Personalized federated learning: A meta-learning approach. *CoRR*, abs/2002.07948, 2020. URL `https://arxiv.org/abs/2002.07948`.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. *CoRR*, abs/1703.03400, 2017. URL `http://arxiv.org/abs/1703.03400`.

Pascal Germain, Francis Bach, Alexandre Lacoste, and Simon Lacoste-Julien. Pac-bayesian theory meets bayesian inference. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL `https://proceedings.neurips.cc/paper/2016/file/84d2004bf28a2095230e8e14993d398d-Paper.pdf`.

Avishek Ghosh, Jichan Chung, Dong Yin, and Kannan Ramchandran. An efficient framework for clustered federated learning. *Advances in Neural Information Processing Systems*, 33:19586–19597, 2020.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *International Conference on Artificial Intelligence and Statistics*, 2010.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In Geoffrey Gordon, David Dunson, and Miroslav Dudík (eds.), *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 315–323, Fort Lauderdale, FL, USA, 11–13 Apr 2011. PMLR. URL `https://proceedings.mlr.press/v15/glorot11a.html`.

Han Guo, Philip Greengard, Hongyi Wang, Andrew Gelman, Yoon Kim, and Eric P Xing. Federated learning as variational inference: A scalable expectation propagation approach. *arXiv preprint arXiv:2302.04228*, 2023.

Conor Hassan, Robert Salomone, and Kerrie Mengersen. Federated variational inference methods for structured latent variable models. *arXiv preprint arXiv:2302.03314*, 2023.

Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1026–1034, 2015.

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. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank J. Reddi, Sebastian U. Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning, 2019. URL `https://arxiv.org/abs/1910.06378`.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2013a. URL `https://arxiv.org/abs/1312.6114`.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013b.

Nikita Kotelevskii, Maxime Vono, Eric Moulines, and Alain Durmus. Fedpop: A bayesian approach for personalised federated learning, 2022. URL `https://arxiv.org/abs/2206.03611`.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450, 2020.

Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In *International conference on machine learning*, pp. 6357–6368. PMLR, 2021.

Wei Li and Andrew McCallum. Pachinko allocation: Dag-structured mixture models of topic correlations. In *Proceedings of the 23rd international conference on Machine learning*, pp. 577–584, 2006.

Yujie Lin, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Dongxiao Yu, Jun Ma, Maarten de Rijke, and Xiuzhen Cheng. Meta matrix factorization for federated rating predictions. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 981–990, 2020.

Othmane Marfoq, Giovanni Neglia, Richard Vidal, and Laetitia Kameni. Personalized federated learning through local memorization. In *International Conference on Machine Learning*, pp. 15070–15092. PMLR, 2022.

H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. 2016. doi: 10.48550/ARXIV.1602.05629. URL https://arxiv.org/abs/1602.05629.

Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečný, Sanjiv Kumar, and H. Brendan McMahan. Adaptive federated optimization, 2020. URL https://arxiv.org/abs/2003.00295.

Karan Singhal, Hakim Sidahmed, Zachary Garrett, Shanshan Wu, Keith Rush, and Sushant Prakash. Federated reconstruction: Partially local federated learning, 2021. URL https://arxiv.org/abs/2102.03448.

Adam Thor Thorgeirsson and Frank Gauterin. Probabilistic predictions with federated learning. *Entropy*, 23 (1):41, 2020.

Jianyu Wang, Zachary Charles, Zheng Xu, Gauri Joshi, H Brendan McMahan, Maruan Al-Shedivat, Galen Andrew, Salman Avestimehr, Katharine Daly, Deepesh Data, et al. A field guide to federated optimization. *arXiv preprint arXiv:2107.06917*, 2021.

Sumio Watanabe. *Mathematical theory of Bayesian statistics*. CRC Press, 2018.

Max Welling and Yee W Teh. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pp. 681–688, 2011.

Honglin Yuan, Warren Richard Morningstar, Lin Ning, and Karan Singhal. What do we mean by generalization in federated learning? In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=VimqQq-i_Q.

Xu Zhang, Yinchuan Li, Wenpeng Li, Kaiyang Guo, and Yunfeng Shao. Personalized federated learning via variational bayesian inference. In *International Conference on Machine Learning*, pp. 26293–26310. PMLR, 2022.

Appendices

A Derivations of Equation 6

Here we provide the detailed derivations of Equation 6 which are derived based on Section 2.2 of (Kingma & Welling, 2013a). The main goal of these derivations is to devise an upper bound on the negative logarithm of the intractable denominator of the posterior probability of model parameters, *i.e.,* $p(\theta, \beta^c|\{y^{n_k}, x^{n_k}\}^c) = p(\theta, \beta^c, \{y^{n_k}\}^c|\{x^{n_k}\}^c)/p(\{y^{n_k}\}^c|\{x^{n_k}\}^c)$, to be able to approximate $p(\theta, \beta^c|\{y^{n_k}, x^{n_k}\}^c)$, in a tractable way. For this purpose, we consider an arbitrary distribution $q(\theta, \beta^c|\{y^{n_k}, x^{n_k}\}^c)$ as a surrogate for the posterior. Since the KL divergence of two distributions is always non-negative, we can use the KL divergence between the true posterior and our surrogate to devise an obvious and trivial upper bound on $-\log p(\{y^{n_k}\}^c|\{x^{n_k}\}^c)$ as the initial step in Equation 19. As the minimum of a non-negative number is always non-negative, we replace the KL divergence with its minimum value with respect to the surrogate distribution $q(\theta, \beta^c|\{y^{n_k}, x^{n_k}\}^c)$, to make this upper bound as tight as possible (Equation 20). Moreover, since $-\log p(\{y^{n_k}\}^c|\{x^{n_k}\}^c)$ is independent of the surrogate distribution, we move this term inside the minimum as shown in Equation 21. The rest of the proof comes from the definition of KL divergence, the multiplication rule of probability, and properties of logarithms. For the sake of simplicity in notation we have $\{y^{n_k}, x^{n_k}\}^c \overset{\text{def}}{=} X, Y$ in the following equations.

$$-\log p(Y|X) \leq -\log p(Y|X) + \overbrace{D_{\text{KL}}(q(\theta, \beta^c|X, Y)\|p(\theta, \beta^c|X, Y))}^{\text{Always} \geq 0.} \tag{19}$$

$$\Rightarrow -\log p(Y|X) \leq -\log p(Y|X) + \overbrace{\min_q D_{\text{KL}}(q(\theta, \beta^c|X, Y)\|p(\theta, \beta^c|X, Y))}^{\text{Always} \geq 0.} \tag{20}$$

$$\Rightarrow -\log p(Y|X) \leq \min_q -\log p(Y|X) + D_{\text{KL}}(q(\theta, \beta^c|X, Y)\|p(\theta, \beta^c|X, Y)) \tag{21}$$

$$\begin{aligned} &= \min_q \mathbb{E}_{q(\theta,\beta^c|X,Y)}[-\log p(Y|X) + \log \frac{q(\theta, \beta^c|X, Y)}{p(\theta, \beta^c|X, Y)}] \\ &= \min_q \mathbb{E}_{q(\theta,\beta^c|X,Y)}[\log \frac{q(\theta, \beta^c|X, Y)}{p(\theta, \beta^c|X, Y)p(Y|X)}] \\ &= \min_q \mathbb{E}_{q(\theta,\beta^c|X,Y)}[\log \frac{q(\theta, \beta^c|X, Y)}{p(\theta, \beta^c, Y|X)}] \\ &= \min_q D_{\text{KL}}(q(\theta, \beta^c|X, Y)\|p(\theta, \beta^c, Y|X)). \end{aligned} \tag{22}$$

B Derivations of Equation 8

We provide details for Equation 8, which is derived based on the definition of KL divergence, properties of logarithms, and the multiplication rule of probability. In the following equations $\{y^{n_k}, x^{n_k}\}^c \overset{\text{def}}{=} X, Y$ for the simplicity in notations.

$$\begin{aligned} p(Y|X) &= \frac{p(\theta, \beta^c, Y|X)}{p(\theta, \beta^c|X, Y)} = \frac{p(\theta, \beta^c, Y|X)}{p(\theta, \beta^c|X, Y)} \times \frac{q(\theta, \beta^c|X, Y)}{q(\theta, \beta^c|X, Y)} \\ &= \frac{p(\theta, \beta^c, Y|X)}{q(\theta, \beta^c|X, Y)} \times \frac{q(\theta, \beta^c|X, Y)}{p(\theta, \beta^c|X, Y)} \\ &= \frac{t(\theta)r(\beta^c)\ell(Y|f(\theta, \beta^c, X))}{q_\lambda(\theta|X, Y)q_\lambda(\beta^c|\theta, X, Y)} \times \frac{q(\theta, \beta^c|X, Y)}{p(\theta, \beta^c|\theta, X, Y)} \\ &\Rightarrow -\log(p(Y|X)) = -\log(\ell(Y|f(\theta, \beta^c, X))) \\ &+ \log(\frac{q_\lambda(\theta|X, Y)}{t(\theta)}) + \log(\frac{q_\lambda(\beta^c|\theta, X, Y)}{r(\beta^c)}) - \log(\frac{q(\theta, \beta^c|X, Y)}{p(\theta, \beta^c|\theta, X, Y)}) \end{aligned}$$

$$
\begin{aligned}
&\Rightarrow \mathbb{E}_{q(\theta,\beta^c|X,Y)}[-\log(p(Y|X))] = -\log(p(Y|X)) \\
&= \mathbb{E}_{q(\theta,\beta^c|X,Y)}[-\log(\ell(Y|f(\theta,\beta^c,X)))] + \mathbb{E}_{q(\theta,\beta^c|X,Y)}[\log(\frac{q_\lambda(\theta|X,Y)}{t(\theta)})] \\
&+ \mathbb{E}_{q(\theta,\beta^c|X,Y)}[\log(\frac{q_\lambda(\beta^c|\theta,X,Y)}{r(\beta^c)})] - \mathbb{E}_{q(\theta,\beta^c|X,Y)}[\log(\frac{q(\theta,\beta^c|X,Y)}{p(\theta,\beta^c|X,Y)})] \\
&\Rightarrow -\log(p(Y|X)) + D_{\mathrm{KL}}(q(\theta,\beta^c|X,Y)\|p(\theta,\beta^c|X,Y)) \\
&= \mathbb{E}_{q(\theta,\beta^c|X,Y)}[-\log(\ell(Y|f(\theta,\beta^c,X)))] \\
&+ \mathbb{E}_{q_\lambda(\beta^c|\theta,X,Y)}[D_{\mathrm{KL}}(q_\lambda(\theta|X,Y)\|t(\theta))] + \mathbb{E}_{q_\lambda(\theta|X,Y)}[D_{\mathrm{KL}}(q_\lambda(\beta^c|\theta,X,Y)\|r(\beta^c))] \\
&= \overbrace{\mathbb{E}_{q(\theta,\beta^c|X,Y)}[-\log(\ell(Y|f(\theta,\beta^c,X)))]}^{\text{Expected Loss}} \\
&+ \underbrace{D_{\mathrm{KL}}(q_\lambda(\theta|X,Y)\|t(\theta))}_{\text{Global Regularizer}} + \underbrace{\mathbb{E}_{q_\lambda(\theta|X,Y)}[D_{\mathrm{KL}}(q_\lambda(\beta^c|\theta,X,Y)\|r(\beta^c))]}_{\text{Local Regularizer}}
\end{aligned}
$$

$$
= \mathbb{E}_{q(\theta,\beta^c|X,Y)}[-\log(\ell(Y|f(\theta,\beta^c,X)))] + D_{\mathrm{KL}}(q(\theta,\beta^c|X,Y)\|t(\theta)r(\beta^c))) \tag{23}
$$

C Proof of Corollary 1

The proof of this corollary is derived from the proof of Theorem 3 in (Germain et al. (2016)). More specifically, Equation 24 comes from Jensen inequality, Equation 25 is a result of Donsker-Varadhan change of measure inequality, and Equation 26 comes from Morkov's inequality.

$$
\begin{aligned}
&\eta\mathbb{E}_{\mathcal{D}}[-\log\big(\ell(Y|X)\big)] = \eta\mathbb{E}_{\mathcal{D}}[-\log\big(\mathbb{E}_{q(\theta,\beta^c|X,Y)}[\ell(Y|X,\theta,\beta^c)\big)]] \\
&\leq \eta\mathbb{E}_{\mathcal{D}}[\mathbb{E}_{q(\theta,\beta^c|X,Y)}[-\log\big(\ell(Y|X,\theta,\beta^c)\big)]] && (24) \\
&\leq \eta\mathbb{E}_{X,Y}[\mathbb{E}_{q(\theta,\beta^c|X,Y)}[-\log\big(\ell(Y|X,\theta,\beta^c)\big)]] \\
&+ D_{\mathrm{KL}}(q(\theta,\beta^c|X,Y)\|\pi(\theta,\beta^c)) \\
&+ \log\left(\mathbb{E}_{\pi(\theta,\beta^c)}[\exp\left(\eta\mathbb{E}_{\mathcal{D}}[-\log(\ell(Y|X,\theta,\beta^c))] - \eta\mathbb{E}_{X,Y}[-\log(\ell(Y|X,\theta,\beta^c))]\right)]\right) && (25) \\
&\overset{\leq}{w.p >} 1-\delta \quad \eta\mathbb{E}_{X,Y}[\mathbb{E}_{q(\theta,\beta^c|X,Y)}[-\log\big(\ell(Y|X,\theta,\beta^c)\big)]] + D_{\mathrm{KL}}(q(\theta,\beta^c|X,Y)\|\pi(\theta,\beta^c)) \\
&+ \log\big(\tfrac{1}{\delta}\mathbb{E}_{X,Y}\mathbb{E}_{\pi(\theta,\beta^c)}\big[\exp\left(\eta\mathbb{E}_{\mathcal{D}}[-\log(\ell(Y|X,\theta,\beta^c))] - \eta\mathbb{E}_{X,Y}[-\log(\ell(Y|X,\theta,\beta^c))]\right)\big]\big) && (26)
\end{aligned}
$$

We note that as opposed to Theorem 3 in (Germain et al., 2016), we did not assume the empirical data samples (X, Y) are derived IID from a data distribution and interestingly this proof, which is a slightly revised version of the proof of Theorem 3 in (Germain et al., 2016), is correct for non-IID empirical data samples as well. The rationale behind this is that none of the steps in the aforementioned proof relies on the IID property of the empirical data samples. More specifically, this proof starts with calculating the true risk, $\mathbb{E}_{\mathcal{D}}$, and moving the logarithm inside the expected value using Jensen inequality. After that we use the Donsker-Varadhan inequality which says $\mathbb{E}_q[\phi(f)] < D_{\mathrm{KL}}(q\|\pi) + \log(\mathbb{E}_\pi[e^{\phi(f)}])$ (Germain et al., 2016). To use this inequality we define $\phi(f) = \mathbb{E}_{\mathcal{D}} - \mathbb{E}_{X,Y}$. The crucial aspect of this proof is the Donsker-Varadhan inequality, which holds true for any function $\phi(f) = \mathbb{E}_{\mathcal{D}} - \mathbb{E}_{X,Y}$ and whether the data we used to compute the empirical risk, $\mathbb{E}_{X,Y}$, is IID or not, doesn't affect its validity. Finally, the last inequality is the Morkov's inequality that does not need IID assumption as well.