Beyond Parameter Averaging in Model Aggregation

Pol G. Recasens[1,2] **Jordi Torres**[1,2] **Josep Ll. Berral**[2,1]
CROMAI
Barcelona Supercomputing Center (BSC)[1], Universitat Politècnica de Catalunya[2]
{pol.garcia,jordi.torres,josep.berral}@bsc.es

Søren Hauberg **Pablo Moreno-Muñoz**
Section for Cognitive Systems
Technical University of Denmark (DTU)
{sohau,pabmo}@dtu.dk

Abstract

The success of foundation models is strongly linked to scale, which has reinforced the interest in federated learning. With the prohibitive cost of training a large language model (LLM) in mind, little attention has been placed on reusing pre-trained models in collaborative training settings. Self-supervision has also played an important role in this success, but its emphasis has been primarily on data. This paper leverages Bayesian principles to bring self-supervision into the model aggregation toolbox. It introduces self-supervised Fisher merging, a framework that successfully merges models in parameter space without re-visiting data, opening a new door in model *reusability*. Experimental results build the foundation of our method on tractable linear models, and highlight its potential on aggregating neural networks.

1 Introduction

The proliferation of *foundation models* (Bommasani et al., 2021) in society, specifically large language models (LLM), puts the spotlight on decentralized training procedures carried out by multiple machines. This point has reinforced the interest in federated learning (FL) (Konen et al., 2016; McMahan et al., 2017), whose main success is to leverage computation and data from *private* devices (Abadi et al., 2016) and train a single global machine learning model. These two achievements represent the two major goals of FL in the last years: *i)* provide a robust learning paradigm that collectively trains a model from multiple clients and *ii)* avoid the centralisation of sensitive data, motivating new manners to communicate and aggregate locally trained parameters in a safe and fully private way.

In recent years, foundation models have revolutionized the field of machine learning, introducing new challenges to FL. One in particular is *critical*, as Brown et al. (2020) pointed out:

> "*Practical large-scale pre-training requires large amounts of computation, which is energy-intensive: training the GPT-3 175B consumed several thousand petaflop/s-days of compute during pre-training, (...). This means we should be cognizant of the cost and efficiency of such models.*"

Although the training of large models might cost millions of dollars, new repositories of pre-trained foundation models (Wolf et al., 2020) are available for public use, promoting flexible and reusable AI.

International Workshop on Federated Learning in the Age of Foundation Models in Conjunction with NeurIPS 2023 (FL@FM-NeurIPS'23). This workshop does not have official proceedings and this paper is non-archival.

While FL still offers multiple solutions to train these computationally expensive models from scratch, a third goal has emerged: *reusability*.

Reusing *assets* has been historically one of the core ideas in software development, mainly due to reasons of efficiency, where it is not usually desirable to duplicate time and resources for a task already solved. With the focus on the *model aggregation* step of FL, we are interested in the novel ideas brought around *reusable* pre-trained models (Raffel, 2023), which also comply with the two main principles in FL described above.

In this paper, we provide a new framework for model aggregation beyond the traditional parameter averaging used in FL. Additionally, we show that self-supervised learning might help on merging parameters from different pre-trained models. To deal with conditional predictions, we build on top of a recent aggregation method based on the Fisher information matrix (Matena and Raffel, 2022).

Background. Model aggregation is one of the most critical steps in FL, where standard approaches like parameter averaging (McMahan et al., 2017) has been shown to converge to good solutions. However, these often rely on the assumption that the client's data is i.i.d. sampled from the global data. To address issues like drastic degradation in performance whenever the previous assumption is not satisfied, other methods considered principles from *Bayesian inference* (Sharma et al., 2019; Chen and Chao, 2021).

The probabilistic view of FL aims to do model aggregation as an approximate maximisation of the joint likelihood of the models' posterior (Ashman et al., 2022; Matena and Raffel, 2022). This is often convenient as having densities over the different models' parameters facilitates aggregation. In practice, obtaining the *global* posterior is also equivalent to the computation of the log-marginal likelihood (LML) of the final model. It has recently been shown that *masked pre-training*, a variant of self-supervised learning (SSL), maximises according to an estimate of the LML (Moreno-Muñoz et al., 2023). For that reason, we are interested in exploring appropriate designs of SSL to perform model aggregation beyond parameter averaging in the Bayesian framework.

2 Self-supervised Learning for Model Aggregation

We consider the problem setting where we have K models built from neural networks (NN) parameterised by $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \cdots, \boldsymbol{\theta}_K$. The main goal behind *model aggregation* is to generate a single model with parameters $\boldsymbol{\theta}$ and similar capabilities as each one of the K *local* models. For convenience, we will refer to this one as the *global* model. The main difficulty behind this task is to perform model aggregation without revisiting any data.

A typical starting point to frame probabilistic model aggregation is to consider the maximisation of the joint posterior distribution. We assume each *local* posterior density $p(\boldsymbol{\theta}|\mathcal{D}_k)$ coming from a different FL client, such that

$$\boldsymbol{\theta}^* = \arg\max_{\theta} \sum_{k=1}^{K} \log p\left(\boldsymbol{\theta} \mid \mathcal{D}_k\right), \tag{1}$$

where $\mathcal{D}_k$ are the *local* data only observed by each kth client. We generally assume these data to be pairs of input-output observations $\{y_n, \boldsymbol{x}_n\}\ \forall n \in \{1, 2, \ldots, N\}$. Due to obtaining posterior densities over NN parameters is generally difficult — i.e. as it is in Bayesian NNs (Bishop, 1995; MacKay, 1995), approximations are often used. When such approximations are set via *isotropic* Gaussian distributions with identical variances, the optimal solution for $\boldsymbol{\theta}^*$ in Eq. (1) is known. This one equals the average of *all* model parameters and is given by $\boldsymbol{\theta}^* = 1/K \sum_k \boldsymbol{\theta}_k$. Using the convention of Matena and Raffel (2022), we will refer to this approach as *isotropic merging* in our empirical results.

Laplace Approximation and Fisher Merging. If we frame *model aggregation* with a different approach than *isotropic* Gaussians, we might end up considering Laplace approximation (LA) (MacKay, 2003; Daxberger et al., 2021). Specifically, LA approximates the posterior of NN's parameters using $\boldsymbol{\theta}_{\text{MAP}}$, such that

$$p(\boldsymbol{\theta} \mid \mathcal{D}) \approx \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\theta}_{\text{MAP}}, \Sigma) \quad \text{with} \quad \Sigma := (\nabla^2_\theta \mathcal{L}(\mathcal{D}|\boldsymbol{\theta})|_{\theta_{\text{MAP}}})^{-1}. \tag{2}$$

Since LA usually requires approximations to the (potentially indefinite) Hessian of the log-likelihood function (Daxberger et al., 2021; MacKay, 2003), the Fisher information matrix is one such choice.

This is defined as the variance of the gradient of the log-likelihood function respect to $\boldsymbol{\theta}$.

$$\mathbf{F}(\boldsymbol{\theta}) := \sum_{n=1}^{N} \mathbb{E}_{p_\theta(y|x_n)} \left[\nabla_\theta \log p_{\boldsymbol{\theta}} \left(y \mid \boldsymbol{x}_n \right) \nabla_\theta \log p_{\boldsymbol{\theta}} \left(y \mid \boldsymbol{x}_n \right)^\top \right]. \tag{3}$$

Since this approximation is quadratically large within the number of parameters, it is usually infeasible to compute, store, or invert. Thus, we typically apply further factorization assumptions. The most lightweight is a diagonal factorization which ignores off-diagonal elements (LeCun et al., 1989). Other approaches, like FEDBE (Chen and Chao, 2021), use a diagonal covariance matrix for the Gaussian posterior. More expressive alternatives are block-diagonal factorizations such as the Kronecker-factored approximate curvature (KFAC) method (Botev et al., 2017).

Following the spirit of Raffel (2023) and Daxberger et al. (2021), the recent aggregation method of Matena and Raffel (2022) introduces *Fisher merging*. Their idea is to construct Gaussian-distributed approximations for each kth model with mean $\boldsymbol{\theta}_k$ and precision $\mathbf{F}_k$. Since the *full* Fisher matrix takes $\mathcal{O}(|\boldsymbol{\theta}|^2)$ in memory, they empirically estimate the diagonal. Using additional hyperparameters, the final model aggregation is computed via a weighted average, which outperforms the traditional *isotropic merging*.

2.1 Self-supervised Fisher merging.

Inspired by the success of self-supervised learning (SSL) in *foundation models*, we propose a new merging method that recurrently benefits from *conditional independence*. The core idea is to avoid the diagonal Fisher approximation of Matena and Raffel (2022) to obtain a better performance in the *global* model aggregation step.

Inverted Bayes' Rule. Our starting point is slightly different to *Fisher merging*. In particular, we are interested in obtaining the *global* posterior solution according to the LML of the data. If we denote the marginal likelihood constant or *evidence* as $\mathcal{Z}_i = \int p(y_i|\boldsymbol{x}_i, \boldsymbol{\theta}) p_0(\boldsymbol{\theta}) d\boldsymbol{\theta}$ where $p_0(\boldsymbol{\theta})$ is the prior density over parameters, then we can compute the same constant for each *local* client. This is, $\mathcal{Z}_1, \mathcal{Z}_2, \ldots, \mathcal{Z}_K$ given the corresponding data $\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_K$. Now, under the assumption of i.i.d. observations, we can say that the exact *global* posterior can be obtained as

$$\log p(\boldsymbol{\theta} \mid \mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_K) = \sum_{k=1}^{K} \left(\log p_k(\boldsymbol{\theta}|\mathcal{D}_k) - \frac{k-1}{K} \log p_0(\boldsymbol{\theta}) \right) + \sum_k \log \mathcal{Z}_k - \log \mathcal{Z}, \tag{4}$$

where each log-posterior density from the kth client is denoted as $\log p_k(\boldsymbol{\theta}|\mathcal{D}_k)$. Moreover, as we are considering a *maximum likelihood* principle for the *global* posterior, the intractable LML constants in Eq. (4) can be ignored in our case, such that

$$\log p(\boldsymbol{\theta}|\mathcal{D}) = \sum_{k=1}^{K} \left(\log p(\boldsymbol{\theta}|\mathcal{D}_k) - \frac{k-1}{K} \log p_0(\boldsymbol{\theta}) \right) + \text{const.} \tag{5}$$

Thus, using Eq. (5), our main goal is now to find $\boldsymbol{\theta}^*$ such that $\boldsymbol{\theta}^* = \operatorname{argmax}_\theta \log p\left(\boldsymbol{\theta}|\mathcal{D}\right)$. The main difficulty appears as we want to maximise each one of the approximate $\log p(\boldsymbol{\theta}|\mathcal{D}_k)$ inherited from the *Fisher merging* method. Luckily, we can avoid diagonal approximations using a self-supervised learning variant known as masked pre-training (MPT).

Masked Pre-Training on the space of parameters. The idea behind MPT is to remove random input dimensions (also known as *masking*) in the observed objects and learn a model that accurately predicts the missing values. In BERT (Devlin et al., 2018), each object is usually considered as a D dimensional vector, where each feature (or *token*) is named as *token*. Given a random set of M tokens — i.e. originally set to 15% in BERT, MPT maximises according to the average of the following objective[1]

$$\log p(\boldsymbol{\theta}_{\mathcal{M}}|\boldsymbol{\theta}_{\mathcal{R}}, \mathcal{D}_k) = \mathbb{E}\left[\sum_{j \in \mathcal{M}} \log p(\boldsymbol{\theta}_j|\boldsymbol{\theta}_{\mathcal{R}}, \mathcal{D}_k) \right], \tag{6}$$

[1]We adapted the objective to work on the parameter variables $\boldsymbol{\theta}$ instead of observations. Notice that we want to avoid a *data* model aggregation in the context of FL.

where the average is w.r.t. the random masking $\pi \in \binom{D}{M}$. Thus, having access to a simpler average of conditional densities between parameters, we can exploit this point to build our merging loss. Particularly, we inherit the weighted sum between posterior and prior densities from Eq. (5), and we also average over the K *local* models and random masking patterns. Notice that our loss accepts mini-batching for stochastic optimization. See box below.

Conditional Precision and Fisher matrices. The last building block of our method is *how to obtain conditional densities* with Fisher-based precision matrices. For this, we primarily make use of the empirical Fisher estimates as in Matena and Raffel (2022) to build precision block matrices on the *masked* and *rest* dimensions. Having these, we exploit properties of Gaussian conditionals to obtain the conditional predictive mean and precision. It is important to notice that *conditional independence* is assumed among the masked tokens as in (Devlin et al., 2018).

▷ Self-supervised Fisher (SSF) merging loss:

$$\mathcal{L}_{\boldsymbol{\theta}} = -\mathbb{E}\left[\sum_{j\in\mathcal{M}}\left[\log p(\boldsymbol{\theta}_j^{(\pi)}|\boldsymbol{\theta}_{m+1:D-1}^{(\pi)}, \mathcal{D}_k)\right] - \frac{k-1}{K}\log p_0(\boldsymbol{\theta}_j^{(\pi)}|\boldsymbol{\theta}_{1:D-1}^{(\pi)})\right]. \tag{7}$$

3 Experiments

In this section, we explore the capabilities of SSF, and validate our proposed construction of each *local* posterior density. First, we lay the foundation with linear models, and then study different configurations of the method (e.g masking rate, number of random masks, merging steps) with small-size neural networks. Specifically, we train and merge numerous NN with the same architecture and same initialization, but different hyperparameters. This follows the same *model merging* set up, recently used in Matena and Raffel (2022) as a benchmark to compare different methodologies. Lastly, we scale-up the size of the models by using CONVNETS (LeCun et al., 1998) of 24K parameters pre-trained on MNIST and FASHION-MNIST datasets.

All the experiments on non-linear models follow the same SSF set-up: the maximization of the aggregated posterior distribution is composed of 5000 steps or *epochs*, and 20 random masking of parameters per step. We study how the number of *masked* parameters affects the merging performance, as reducing the masking size typically improves the computational cost of the method. We optimize the *global* model using stochastic gradient descent (SGD) and our loss (see Eq. 7), simulating standard NN training.

Once we have pre-trained each model, and before the merging, we compute and store the sum of the gradients and squared-gradients for a limited number of samples. We require the squared gradients as they define the diagonal of the Fisher matrix. On the other hand, we partially build the Fisher matrix *on-the-fly* by computing the outer product between the gradients respect to masked parameters. This process is made at each step to compute the conditional predictive probabilities for each *local* model.

3.1. Formal results in tractable linear models. To verify that our merging approach is based on a good approximation of the true posterior distribution, we first need a tractable probabilistic model. The Bayesian treatment of linear regression (Bishop, 2006) is the best choice as the selection of proper conjugate priors makes the posterior also Gaussian and exact. Thus, we assume the zero-mean isotropic Gaussian prior $p(\boldsymbol{w} \mid \alpha) = \mathcal{N}\left(\boldsymbol{w} \mid \mathbf{0}, \alpha^{-1}\mathbb{I}\right)$, and the corresponding posterior distribution is obtained as $p(\mathbf{w} \mid \mathbf{t}) = \mathcal{N}\left(\boldsymbol{w} \mid \boldsymbol{m}_N, \mathbf{S}_N\right)$, where

$$\begin{aligned} \boldsymbol{m}_N &= \mathbf{S}_N\left(\mathbf{S}_0^{-1}\boldsymbol{m}_0 + \beta\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t}\right) \\ \mathbf{S}_N^{-1} &= \mathbf{S}_0^{-1} + \beta\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}. \end{aligned} \tag{8}$$

In this first case, we are able to aggregate models using their exact posterior distribution. Thus, our analysis relies on a synthetic regression dataset, where three linear models are fitted on different overlapping subsets of the input data (non-i.i.d). The findings presented in Fig. 1 indicate that during the self-supervised merging process the parameters of the optimized model, or *global*, converge to the true ones. These are the ones fitted to the whole observed data. Moreover, since the posterior

distribution is usually intractable in non-linear settings, we approximated the posterior of each linear model with the Fisher matrix. In particular, we use it as a positive-definite approximation of the Hessian over the log-likelihood function, which defines the precision matrix. In linear regression it has the following closed-form expression,

$$\mathbf{F}(\boldsymbol{\theta}) = \boldsymbol{x}^\top \boldsymbol{x}/\sigma^2.$$

In this regard, Fig. 1 shows that throughout the self-supervised merging process, the parameters of the *global* converge toward the true parameters in *all* cases. This observation lays the groundwork for the applicability of our aggregation method to non-linear scenarios.

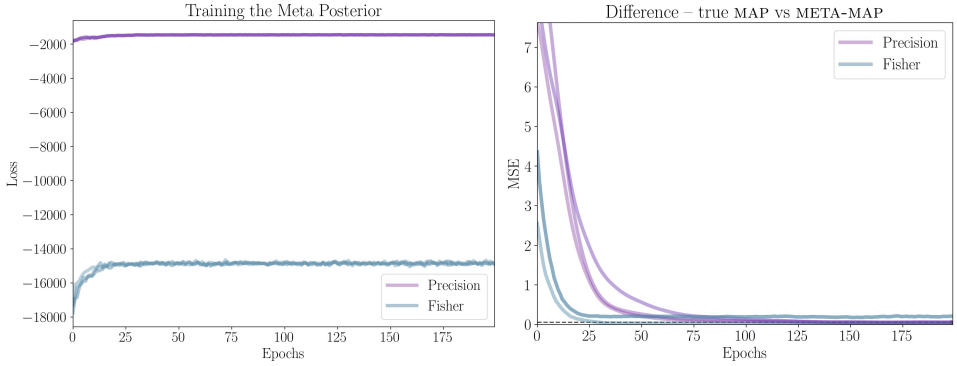

Figure 1: Asymptotic convergence of the optimized parameters to the true parameters during the aggregation process with linear models. (**Left**). Value of the loss, defined in Eq. (7), at each aggregation step. The negative loss is minimized during the aggregation, as it quantifies the probability of observing the *global* parameters given the posterior of each aggregated model. (**Right**). Mean-squared error between the parameters of the final model and the true parameters. Although using the Fisher matrix introduces a small bias, the aggregated model converges towards the true model.

3.2. Beyond linear models with small-size networks. One remaining question is whether our self-supervised merging method holds with non-linear models, such as neural networks within a large parameter space. It is worth to highlight that we aim to optimize a neural network from scratch without revisiting any data, maximizing the joint posterior distribution at each step. This is expressed with predictive conditionals over parameters which might number on the millions. The PINWHEEL classification (synthetic) dataset offers a promising initial step in our analysis and given our computational constraints. The studied models are neural networks, with identical architecture and initialization, and composed of 400 parameters. However, each model is trained with different hyperparameters. In Table 1 and Fig. 2, we indicate that our self-supervised method is able to optimize a randomly initialized model in parameter space *and* outperform the current state-of-the-art merging methods.

Table 1: Comparison of the different merging methodologies w.r.t. the number of aggregated models and SSF masking rate. The values corresponds to the test loss of the *global* model, and each row represents an independent set of experiments.

# MODELS	ISOTROPIC	FISHER	SSF 20% (ours)	SSF 30% (ours)	SSF 40% (ours)	SSF 50% (ours)
2	1.0762 ± 0.0023	0.7413 ± 0.0090	0.7396 ± 0.0099	0.7397 ± 0.0093	0.7397 ± 0.0079	0.7398 ± 0.0079
3	0.6658 ± 0.0158	0.2320 ± 0.0003	0.2265 ± 0.0005	0.2270 ± 0.0005	0.2276 ± 0.0005	0.2277 ± 0.0005
4	0.4342 ± 0.0037	0.1424 ± 0.0002	0.1397 ± 0.0002	0.1403 ± 0.0002	0.1403 ± 0.0002	0.1406 ± 0.0002
5	0.3085 ± 0.0002	0.1126 ± 0.0002	0.1108 ± 0.0002	0.1109 ± 0.0002	0.1117 ± 0.0002	0.1114 ± 0.0002

3.3. Bounding the cost for medium-size neural networks. With an eye on foundation models, where models might be composed of billions of parameters, we fix the number of parameters considered to construct each local density. This bounds the total cost of the method, that grows with the size of the model. To do so, we randomly select a window of parameters with a fixed length, from which we mask a random subset. Specifically, we select 5000 parameters and mask 800. Limiting the number of *rest* dimensions can be balanced with an increase of random masks per step, which might also help to stochastically explore further dimensions. Thus, we increase the number of permutations

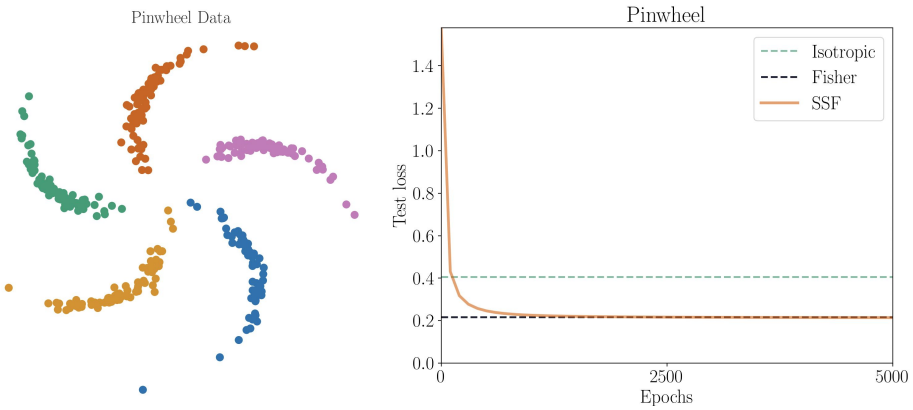

Figure 2: (**Left**). We generate a Pinwheel synthetic dataset composed of 5 different clusters, with 1000 samples per cluster and 2 latent dimensions per sample. (**Right**). Each classification model, defined as a three hidden layers neural network, is trained for a different number of epochs and with a different learning rate. After 5000 epochs of aggregation SSF outperforms the current state-of-the-art merging methods.

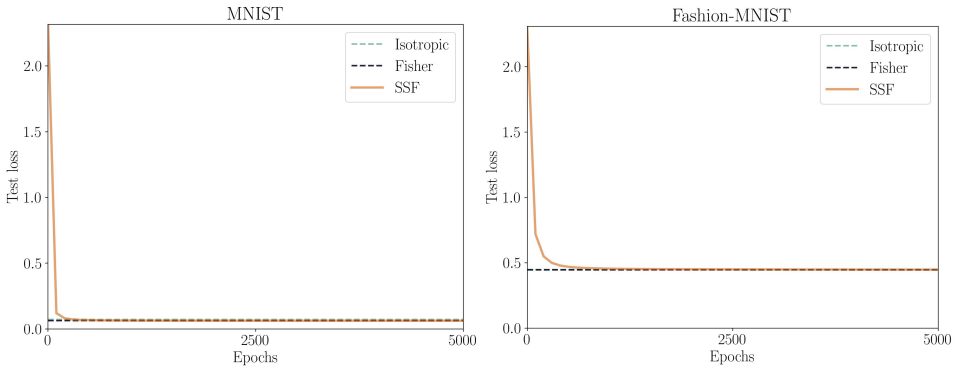

Figure 3: (**Left**). Comparison of the three techniques on merging 24k parameter CONVNETS trained with MNIST. We fix the number of parameters considered at each optimization step to bound the computational expenses of the optimization process. Despite not considering the whole parameter set, SSF optimizes a large model in parameter space, converging to an optimal solution. However, in this case SSF and Fisher merging obtain identical performance. (**Right**). Comparison of the three techniques on merging CONVNETS trained with FASHION-MNIST. The three merging methods lead to almost identical test losses, being Fisher merging slightly better than SSF.

.

per step to 200. In this analysis, we merge three CONVNET models of 24K parameters, trained on MNIST and FASHION-MNIST datasets from different hyperparameter configurations. In Fig. 3 we can observe how SSF converges towards Fisher merging. The reason for this might be that limiting the number of dimensions introduces a small bias in the conditioning.

Table 2: Comparison of the different merging methodologies w.r.t. MNIST and FASHION-MNIST datasets. In both cases we fix the number of parameters considered for the calculation of the loss, biasing the merging but bounding the computational cost.

DATASET	ISOTROPIC	FISHER	SSF 20 % (ours)
MNIST	0.0911 ± 0.0	0.0697 ± 0.0	0.0697 ± 0.0
FASHION-MNIST	0.4734 ± 0.0	0.4679 ± 0.0	0.4694 ± 0.0

3.4. Limitations. We have shown the positive results of improving the approximation given the local posterior densities, and we designed a *novel* self-supervision technique that aggregates models in parameter space. However, we have identified a clear trade-off — this advancement comes at

the expense of lacking a direct closed-form merging solution (e.g. parameter averaging), forcing an optimization process. Here, we identify two main computational problems. First, the predictive conditional for each local posterior is linked to the parameter size, due to the inherent quadratic cost of the Fisher matrix. Precisely, computing outer products between *masked* and *rest* parameters at each step of SSF might become challenging for larger models. Second, the optimization of large models is increasingly expensive when the model size grows, as it has been shown on recent large foundation models (Bommasani et al., 2021).

3.5. Future directions. We have already explored a possible solution for the first limitation, but improving the optimization of large models is out of the scope of this paper. We leave as future work the exploration of possible upgrades for both the method and the optimization, and also the integration of *ssf* in real FL scenarios. Perhaps, it has potential to serve as a form of *knowledge distillation* during the aggregation step, facilitating the *reusability* of pre-trained models.

4 Conclusion

In this paper, we proposed self-supervised Fisher merging, a *novel* aggregation technique that works beyond the standard parameter averaging. Inspired by the success of masked pre-training on recent foundation models and new links to the implicit maximization of the LML (Moreno-Muñoz et al., 2023), we provide a robust approximation to the global posterior by decomposing each local density as sum of conditionals over parameters. This clearly avoids the diagonalization of the Fisher matrix. We maximize the joint posterior throughout an optimization process, training models from scratch without re-visiting any data. We support our method with a formal study on tractable linear models, and with empirical results on merging small and medium-size neural networks.

Acknowledgements

This work was supported by a research grant (42062) from VILLUM FONDEN. This project has also received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement 757360). Our work was funded in part by the Novo Nordisk Foundation through the Center for Basic Machine Learning Research in Life Sciences (MLSS) with reference number NNF20OC0062606. This work has been partially financed by the EU-H2020 programme under grant agreement EU-H2020 GA.952179 and grant agreement EU-HORIZON GA.101095717. Also, it has been partially financed by Generalitat de Catalunya (AGAUR) under grant agreement 2021-SGR-00478.

References

M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang. Deep learning with differential privacy. In *ACM SIGSAC Conference on Computer and Communications Security*, pages 308–318, 2016.

M. Ashman, T. D. Bui, C. V. Nguyen, S. Markou, A. Weller, S. Swaroop, and R. E. Turner. Partitioned variational inference: A framework for probabilistic federated learning. *arXiv preprint arXiv:2202.12275*, 2022.

C. M. Bishop. Bayesian methods for neural networks. *Technical Report (Aston University)*, 1995.

C. M. Bishop. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

A. Botev, H. Ritter, and D. Barber. Practical gauss-newton optimisation for deep learning. In *International Conference on Machine Learning*, pages 557–565. PMLR, 2017.

T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:1877–1901, 2020.

H.-Y. Chen and W.-L. Chao. Fedbe: Making bayesian model ensemble applicable to federated learning. *International Conference on Learning Representations (ICLR)*, 2021.

E. Daxberger, A. Kristiadi, A. Immer, R. Eschenhagen, M. Bauer, and P. Hennig. Laplace redux-effortless bayesian deep learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 34, 2021.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

J. Konen, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

Y. LeCun, J. Denker, and S. Solla. Optimal brain damage. *Advances in neural information processing systems*, 2, 1989.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

D. J. MacKay. Bayesian neural networks and density networks. *Nuclear Instruments and Methods in Physics Research Section A*, 354(1):73–80, 1995.

D. J. MacKay. *Information theory, inference and learning algorithms*. Cambridge University Press, 2003.

M. S. Matena and C. A. Raffel. Merging models with Fisher-weighted averaging. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:17703–17716, 2022.

B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics (AISTATS)*, pages 1273–1282. PMLR, 2017.

P. Moreno-Muñoz, P. G. Recasens, and S. Hauberg. On masked pre-training and the marginal likelihood. *Advances in Neural Information Processing Systems (NeurIPS)*, 36, 2023.

C. Raffel. Building machine learning models like open source software. *Communications of the ACM*, 66(2): 38–40, 2023.

M. Sharma, M. Hutchinson, S. Swaroop, A. Honkela, and R. E. Turner. Differentially private federated variational inference. *2nd Privacy in Machine Learning Workshop @ NeurIPS*, 2019.

T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, et al. Transformers: State-of-the-art natural language processing. In *Conference on Empirical Methods in Natural Language Processing (EMNLP): System Demonstrations*, pages 38–45, 2020.