Supplementary of Probabilistic Fusion of Neural Networks that Incorporates Global Information

1. Permutation invariance of deep neural architecture

Permutation invariance of deep FCs We naturally extend Eq. (1) in main text to iteratively define deep FC network:

$$f_s(\boldsymbol{x}^{(n)}) = \sigma(\boldsymbol{x}^{(n-1)^{\mathrm{T}}} \mathbb{A}_s^{(n-1)^{\mathrm{T}}} \boldsymbol{W}_s^{(n)} \mathbb{A}_s^{(n)}), \tag{1}$$

where $n = 1, 2, \ldots, N$ is the layer index, $\boldsymbol{W}_s^{(0)}$ is a identity matrix indicating non-ambiguity in the ordering of input features $x = x_0$ and $\boldsymbol{W}_s^{(N)}$ is a identity matrix for the same purpose in output classes. Conventionally $\sigma(\cdot)$ is any non-linearity except for x_N where it is the identity function (or softmax if we want probabilities instead of logits). To perform matched aggregating of deep FCs acquired from S clients we need to align every layer of every client. Here we consider an iteratively (in layers) matched aggregating formulation which will be detailedly described in later subsections.

Permutation invariance of CNNs Before our work, there is seldom study that implement model fusion algorithms on CNNs. Therefore, we introduce permutation invariance of CNNs before applying model fusion methods on CNNs. To understanding permutation invariance of CNNs, the key observation is that instead of neurons, channels define the invariance. To be more specific, define $\mathrm{Conv}(\boldsymbol{x}, \boldsymbol{W}_s)$ as the convolutional operation on input data $\boldsymbol{x}$ with weights $\boldsymbol{W}_s \in \mathbb{R}^{C^{in} \times w \times h \times C^{out}}$, where C^{in}, C^{out} respectively are the numbers of input/output channels and w, h are the width and height of the filters. Applying any permutation to the output channel of the weights and then same permutation to the input channel of the subsequent layer will make the result of corresponding CNN's forward pass consistent:

$$f_s(\boldsymbol{x}^{(n)}) = \sigma(\mathrm{Conv}(\boldsymbol{x}^{(n-1)}, \mathbb{A}_s^{(n-1)^{\mathrm{T}}} \boldsymbol{W}_s^{(n)} \mathbb{A}_s^{(n)})). \tag{2}$$

To apply matched aggregating for the nth CNN layer we form inputs as $\{w_{sj} \in \mathbb{R}^D\}_{j=1}^{C_n^{out}}$, $j = 1, \cdots, J_s$, where D is the dimension of $\mathbb{R}^{C_n^{in} \times w \times h}$, to which the flattened $\mathbb{A}_s^{(n-1)^T} \boldsymbol{W}_s^{(n)}$ belongs. Similar to deep FCs, we can also iteratively perform matched aggregating on deep CNNs.

2. BBP and IBP

Beta-Bernoulli Process and Indian Buffet Process Denote Q as a random measure drawn from a Beta process: $Q|\gamma_0, H \sim \mathrm{BP}(1, \gamma_0 H)$, where γ_0 is the mass parameter, H is the base measure over some domain Ω such that $H(\Omega) = 1$. One can show that Q is a discrete measure with $Q = \sum_i q_i \delta_{\boldsymbol{\theta}_i}$, which can be characterized by an infinitely countable set of (weight, atom) pairs $(q_i, \boldsymbol{\theta}_i) \in [0, 1] \times \Omega$. The atoms $\boldsymbol{\theta}_i$ can be drawn i.i.d from H and the weights $\{q_i\}_{i=1}^{\infty}$ can be generated via a stick-breaking process (**?**): $q_1 \sim \mathrm{Beta}(\gamma_0, 1), q_i = \prod_{g=1}^{i} q_g$. Then subsets of atoms

in the random measure Q can be picked via a Bernoulli process. That is, each subset $\mathcal{T}_s$ for $s = 1, \cdots, S$ can be distributed via a Bernoulli process with base measure Q: $\mathcal{T}_s|Q \sim \text{BeP}(Q)$. Hence, subset $\mathcal{T}_s$ can also be viewed as a discrete measure $\mathcal{T}_s := \sum_i a_{si}\delta_{\boldsymbol{\theta}_i}$, which is formed by pairs $(a_{si}, \boldsymbol{\theta}_i) \in \{0,1\} \times \Omega$, where $a_{si}|q_i \sim \text{Bernoulli}(q_i), \forall i$ is a binary random variable indicating whether $\boldsymbol{\theta}_i$ belongs to subset $\mathcal{T}_s$. We call such collection of subsets a Beta-Bernoulli process (**?**).

The Indian buffet process (IBP) specifies distribution on sparse binary matrices (**?**). IBP involves a metaphor of a sequence of customers tasting dishes in an infinite buffet: the first customer tastes Poisson(γ_0) dishes, every subsequent sth customer tastes each dish that is previously selected with probability n_i/s, where $n_i = \sum_{s=1}^{S-1} a_{si}$, and then tastes Poisson(γ_0/s) new dishes. Marginalizing over Beta Process distributed Q above will induce dependencies among subsets and recover the predictive distribution $\mathcal{T}_S|\mathcal{T}_1, \cdots, \mathcal{T}_{S-1} \sim \text{BeP}(H\frac{\gamma_0}{S} + \sum_i \frac{n_i}{S}\delta_{\boldsymbol{\theta}_i})$. That is equivalent to the IBP.

3. Iteratively Layer-wise Matched Aggregation via GI-FNM

As the empirical study in Wang et al. (2020) demonstrates, directly applying the matching algorithms fails on deep architectures which are necessary to solve more complex tasks. Thus to alleviate this problem , we also extend GI-FNM to the following layer-wise matching scheme. Firstly, the server only collects the first layer weights from the clients and applies GI-FNM to acquire the first layer weights of the federated model. Then, the server broadcasts these weights to the clients, which proceed to train all consecutive layers on their datasets while keeping the matched layers frozen. Repeat this process until the last layer, where we make a weighted average based on class proportions of each client's data points. We summarize our layer-wise version of GI-FNM in Algorithm 1. The layer-wise approach requires communication rounds that equal to the number of layers in a neural network. Experimental results show that with layer-wise matching, GI-FNM performs well on the ConvNets even for U-nets which has a complex architecture. In the more challenging heterogeneous setting, GI-FNM outperforms FedAvg, FedProx trained with same number of communication rounds (5 for ConvNet and 19 for U-net).

4. Incorporating GI-FNM with Batch Normalization Layers

Although Wang et al. (2020) shows how to apply the PFNM to CNNs, it doesn't enable additional deep learning building blocks, e.g., batch normalization layer, to the matching algorithm. However, widely used deep CNNs such as U-net often contain batch normalization layer in their architectures. In this paper, we utilize a common setup which merges the batch normalization layer with a preceding convolution to incorporate GI-FNM with batch normalization layer.

Without loss of generality, we assume that the feature map size is the same as the filter size. Let $W_{\text{conv}} \in \mathbb{R}^{C^{out} \times (C^{in} \cdot w \cdot h)}$ and $b_{\text{conv}} \in \mathbb{R}^{C^{out}}$ be the parameters of the convolutional layer that precedes batch normalization. Given a input data $x \in \mathbb{R}^{(C^{in} \cdot w \cdot h)}$, the convolutional operator can be simply expressed as:

$$f = W_{\text{conv}} * x + b_{\text{conv}}. \tag{3}$$

Batch normalization (BN) is a popular method used in modern neural networks as it often reduces training time and potentially improves generalization. Given the outputting feature of

Algorithm 1 Iteratively Layer-wise GI-FNM

Require:
Collected local weights of N-layer architectures $\{\boldsymbol{W}_s^{(0)}, \cdots, \boldsymbol{W}_s^{(N-1)}\}_{s=1}^{S}$ from S clients;

Ensure:
New constructed global weights $\{\boldsymbol{W}^{(0)}, \cdots, \boldsymbol{W}^{(N-1)}\}$.

$n = 0$;
while layers $n \leq N$ **do**
  if n < N-1 **then**
    $\{\mathbb{A}_s\}_{s=1}^{S}$ = GI-PNM($\{\boldsymbol{W}_s^{(n)}\}_{s=1}^{S}$);
    $\boldsymbol{W}^{(n)} = \frac{1}{S}\sum_s \boldsymbol{W}_s^{(n)}\mathbb{A}_s^T$;
  else
    $\boldsymbol{W}^{(n)} = \sum_s \boldsymbol{p}_s \cdot \boldsymbol{W}_s^{(n)}$ where $\boldsymbol{p}_s$ is vector of fraction of data points with each label on worker s, and $\cdot$ denotes the dot product;
  end if
  for $s \in \{1, \cdots, S\}$ **do**
    $\boldsymbol{W}_s^{(n+1)} = \mathbb{A}_s \boldsymbol{W}_s^{(n+1)}$;
    Train $\{\boldsymbol{W}_s^{(n+1)}, \cdots, \boldsymbol{W}_s^{(N-1)}\}$ with $\boldsymbol{W}_s^{(n)}$ frozen;
  end for
  $n = n + 1$
end while

preceding convolutional layer, it can be normalized as follows:

$$\hat{f} = \gamma \frac{f - \mu}{\sqrt{\sigma^2 + \epsilon}} + \beta, \tag{4}$$

where μ and σ^2 are the mean and variance computed over a batch of feature, ϵ is a small constant included for numerical stability, γ is the scaling factor and β the shift factor. The parameters γ and β are slowly learned with gradient descent together with the other parameters of the network.

If we take W_{conv} and b_{conv} into the Eq. (4) we can get the new weights and bias as:

- weights: $W_{\text{BN}} = \gamma \cdot \frac{W_{\text{conv}}}{\sqrt{\sigma^2+\epsilon}}$;
- bias: $b_{\text{BN}} = \gamma \cdot \frac{(b_{\text{conv}} - \mu)}{\sqrt{\sigma^2+\epsilon}} + \beta$.

Thus the batch normalized feature can be directly obtained by:

$$\hat{f} = W_{\text{BN}} * x + b_{\text{BN}}. \tag{5}$$

By matching the fused weights W_{BN} and bias b_{BN}, we enable the batch normalization layer in GI-FNM.

5. Proofs

5.1. Proof of Proposition 1

Proof PFNM maximizes a posterior probability of the global atoms$\{\boldsymbol{\theta}_i\}_{i=1}^{\infty}$ and assignments of observed neural network weight estimates to global atoms$\{\boldsymbol{A}^s\}_{s=1}^{S}$. Given estimates of the client

weights $\{\boldsymbol{w}_{sj}$ for $j = 1,\ldots,J_s\}_{s=1}^{S}$, it has:

$$\max_{\{\boldsymbol{\theta}_i\},\{\boldsymbol{A}^s\}} P(\{\boldsymbol{\theta}_i\},\{\boldsymbol{A}^s\}|\{\boldsymbol{w}_{sj}\}) \propto P(\{\boldsymbol{w}_{sj}\}|\{\boldsymbol{\theta}_i\},\{\boldsymbol{A}^s\})P(\{\boldsymbol{A}^s\})P(\{\boldsymbol{\theta}_i\}), \tag{6}$$

by taking negative natural logarithm it can obtain:

$$\min_{\{\boldsymbol{\theta}_i\},\{\boldsymbol{A}^s\}} -\sum_i \Big(\sum_{s,j} \boldsymbol{A}^s_{i,j} \log(p(\boldsymbol{w}_{sj}|\sim\boldsymbol{\theta}_i)) + \log(q(\boldsymbol{\theta}_i))\Big) - \log(P(\{\boldsymbol{A}^s\})), \tag{7}$$

expand probability function of multi-dimensional Gaussian distributions in Eq. (7), it obtains:

$$\min_{\{\boldsymbol{\theta}_i\},\{\boldsymbol{A}^s\}} \frac{1}{2}\sum_i \left(\frac{||\hat{\boldsymbol{\theta}}_i - \boldsymbol{\mu}_0||^2}{\sigma_0^2} + (D+K)\log(2\pi\sigma_0^2) + \sum_{s,j} A^s_{i,j}\frac{||\boldsymbol{w}_{sj} - \hat{\boldsymbol{\theta}}_i||^2}{\sigma_s^2}\right) - \log(P(\{\boldsymbol{A}^s\})). \tag{8}$$

We now consider the first part of Eq. (8). Through the closed-form expression of $\{\boldsymbol{\theta}_i\}$ estimated according to the Gaussian-Gaussian conjugacy:

$$\hat{\boldsymbol{\theta}}_i = \frac{\boldsymbol{\mu}_0/\sigma_0^2 + \sum_{s,j} A^s_{i,j}\boldsymbol{w}_{sj}/\sigma_s^2}{1/\sigma_0^2 + \sum_{s,j} A^s_{i,j}/\sigma_s^2} \text{ for } i = 1,...,J, \tag{9}$$

where for simplicity we assume $\boldsymbol{\Sigma}_0 = \boldsymbol{I}\sigma_0^2$ and $\boldsymbol{\Sigma}_s = \boldsymbol{I}\sigma_s^2$, we can now cast first part of Eq. (8) with respect only to $\{\boldsymbol{A}^s\}_{s=1}^{S}$:

$$\begin{aligned}
&\frac{1}{2}\sum_i \left(\frac{||\hat{\boldsymbol{\theta}}_i - \boldsymbol{\mu}_0||^2}{\sigma_0^2} + (D+K)\log(2\pi\sigma_0^2) + \sum_{s,j} A^s_{i,j}\frac{||\boldsymbol{w}_{sj} - \hat{\boldsymbol{\theta}}_i||^2}{\sigma_s^2}\right) \\
\cong &\frac{1}{2}\sum_i \left(\langle\hat{\boldsymbol{\theta}}_i,\hat{\boldsymbol{\theta}}_i\rangle(\frac{1}{\sigma_0^2} + \sum_{s,j}\frac{A^s_{i,j}}{\sigma_s^2}) + (D+K)\log(2\pi\sigma_0^2) - 2\langle\hat{\boldsymbol{\theta}}_i, \sum_{s,j} A^s_{i,j}\frac{\boldsymbol{w}_{sj}}{\sigma_s^2}\rangle\right) \\
= &-\frac{1}{2}\sum_i \left(\frac{||\sum_{s,j}\boldsymbol{A}^s_{i,j}\frac{\boldsymbol{w}_{sj}-\boldsymbol{\mu}_0}{\sigma_s^2}||^2}{(1/\sigma_0^2 + \sum_{s,j}\boldsymbol{A}^s_{i,j}/\sigma_s^2} - (D+K)\log(2\pi\sigma_0^2)\right).
\end{aligned} \tag{10}$$

Partition Eq. (10) between $i = 1,...,J_{-s'}$ and $i = J_{-s'}+1,...,J_{-s'}+J_{s'}$, and because it is now solving for $\boldsymbol{A}^{s'}$, it can subtract terms independent of $\boldsymbol{A}^{s'}$:

$$\begin{aligned}
&\sum_i \left(\frac{||\sum_{s,j}\boldsymbol{A}^s_{i,j}\frac{\boldsymbol{w}_{sj}-\boldsymbol{\mu}_0}{\sigma_s^2}||^2}{(1/\sigma_0^2 + \sum_{s,j}\boldsymbol{A}^s_{i,j}/\sigma_s^2} - (D+K)\log(2\pi\sigma_0^2)\right) \\
\cong &\sum_{i=1}^{J_{-s'}}\left(\frac{||\sum_j \boldsymbol{A}^{s'}_{i,j}\frac{\boldsymbol{w}_{s'j}-\boldsymbol{\mu}_0}{\sigma_{s'}^2} + \sum_{s\in-s',j}\boldsymbol{A}^s_{i,j}\frac{\boldsymbol{w}_{sj}-\boldsymbol{\mu}_0}{\sigma_s^2}||^2}{1/\sigma_0^2 + \sum_j \boldsymbol{A}^{s'}_{i,j}/\sigma_{s'}^2 + \sum_{s\in-s',j}\boldsymbol{A}^s_{i,j}/\sigma_s^2} - \frac{||\sum_{s\in-s',j}\boldsymbol{A}^s_{i,j}\frac{\boldsymbol{w}_{sj}-\boldsymbol{\mu}_0}{\sigma_s^2}||^2}{1/\sigma_0^2 + \sum_{s\in-s',j}\boldsymbol{A}^s_{i,j}/\sigma_s^2}\right) \\
&+ \sum_{i=J_{-s'}+1}^{J_{-s'}+J_{s'}}\left(\frac{||\sum_j \boldsymbol{A}^{s'}_{i,j}\frac{\boldsymbol{w}_{s'j}-\boldsymbol{\mu}_0}{\sigma_{s'}^2}||^2}{1/\sigma_0^2 + \sum_j \boldsymbol{A}^{s'}_{i,j}/\sigma_{s'}^2}\right),
\end{aligned} \tag{11}$$

observe that $\sum_j \boldsymbol{A}^{s'}_{i,j} \in \{0,1\}$, i.e. it is 1 if some neuron from dataset s' is matched to global neuron i and 0 otherwise. Thus Eq. (11) can rewritten as a linear sum assignment problem:

$$\begin{aligned}\sum_{i=1}^{J_{-s'}}\sum_{j=1}^{J_{s'}} \boldsymbol{A}^{s'}_{i,j}\left(\frac{||\frac{\boldsymbol{w}_{s'j}-\boldsymbol{\mu}_0}{\sigma^2_{s'}}+\sum_{s\in-s',j}\boldsymbol{A}^{s}_{i,j}\frac{\boldsymbol{w}_{sj}-\boldsymbol{\mu}_0}{\sigma^2_s}||^2}{1/\sigma_0^2+1/\sigma^2_{s'}+\sum_{s\in-s',j}\boldsymbol{A}^{s}_{i,j}/\sigma^2_s}-\frac{||\sum_{s\in-s',j}\boldsymbol{A}^{s}_{i,j}\frac{\boldsymbol{w}_{sj}-\boldsymbol{\mu}_0}{\sigma^2_s}||^2}{1/\sigma_0^2+\sum_{s\in-s',j}\boldsymbol{A}^{s}_{i,j}/\sigma^2_s}\right)\\ +\sum_{i=J_{-s'}+1}^{J_{-s'}+J_{s'}}\sum_{j=1}^{J_{s'}}\boldsymbol{A}^{s'}_{i,j}\left(\frac{||\frac{\boldsymbol{w}_{s'j}-\boldsymbol{\mu}_0}{\sigma^2_{s'}}||^2}{1/\sigma_0^2+\sum_j\boldsymbol{A}^{s'}_{i,j}/\sigma^2_{s'}}\right).\end{aligned} \tag{12}$$

Then consider the second term of Eq. (8), by subtracting terms independent of $\boldsymbol{A}^{s'}$ it has:

$$\log(P(\boldsymbol{A}^{s'})) = \log(P(\boldsymbol{A}^{s'}|\boldsymbol{A}^{-s'})) + \log(P(\boldsymbol{A}^{-s'})). \tag{13}$$

First, it can ignore $\log(P(\boldsymbol{A}^{-s'}))$ since now are optimizing for $\boldsymbol{A}^{s'}$. Second, due to exchange ability of datasets (i.e. customers of the IBP), $\boldsymbol{A}^{s'}$ can always be treated as the last customer of the IBP. Denote $n_i^{-s'} = \sum_{-s',j}\boldsymbol{A}^{s'}_{i,j}$ as the number of times local weights were assigned to global atom i outside of group s'. Now it can obtain the following:

$$\begin{aligned}\log P(\boldsymbol{A}^{s'}|\boldsymbol{A}^{-s'}) \cong &\sum_{i=1}^{J_{-s'}}\left(\Big(\sum_{j=1}^{J_{s'}}\boldsymbol{A}^{s'}_{i,j}\Big)\log\frac{n_i^{-s'}}{S}+\Big(1-\sum_{j=1}^{J_{s'}}\boldsymbol{A}^{s'}_{i,j}\Big)\log\frac{S-n_i^{-s'}}{S}\right)\\ &-\log\left(\sum_{i=J_{-s'}+1}^{J_{-s'}+J_{s'}}\sum_{j=1}^{J_{s'}}\boldsymbol{A}^{s'}_{i,j}\right)+\left(\sum_{i=J_{-s'}+1}^{J_{-s'}+J_{s'}}\sum_{j=1}^{J_{s'}}\boldsymbol{A}^{s'}_{i,j}\right)\log\frac{\gamma_0}{J}.\end{aligned} \tag{14}$$

Eq. (14) thus can be rearranged as a linear sum assignment problem:

$$\sum_{i=1}^{J_{-s'}}\sum_{j=1}^{J_{s'}}\boldsymbol{A}^{s'}_{i,j}\log\frac{n_i^{-s'}}{S-n_i^{-s'}}+\sum_{i=J_{-s'}+1}^{J_{-s'}+J_{s'}}\sum_{j=1}^{J_{s'}}\boldsymbol{A}^{s'}_{i,j}\left(\log\frac{\gamma_0}{S}-\log(i-J_{-s'})\right). \tag{15}$$

Combining Eq. (12) and Eq. (5.1), we arrive at the cost specification shown in Eq. (2) of the main text. And this is induced in Yurochkin et al. (2019b). The assignment cost specification for finding $\{\mathbf{A}^{s'}\}$ is

$$\boldsymbol{C}^{s'}_{i,j} = \begin{cases} -\frac{\|\frac{\boldsymbol{\mu}_0}{\sigma_0^2}+\frac{\boldsymbol{w}_{s'j}}{\sigma^2_{s'}}+\sum_{s\in-s',j}\boldsymbol{A}^{s}_{i,j}\frac{\boldsymbol{w}_{sj}}{\sigma^2_s}\|^2}{\frac{1}{\sigma_0^2}+\frac{1}{\sigma^2_{s'}}+\sum_{s\in-s',j}\boldsymbol{A}^{s}_{i,j}\frac{1}{\sigma^2_s}}+\frac{\|\frac{\boldsymbol{\mu}_0}{\sigma_0^2}+\sum_{s\in-s',j}\boldsymbol{A}^{s}_{i,j}\frac{\boldsymbol{w}_{sj}}{\sigma^2_s}\|^2}{\frac{1}{\sigma_0^2}+\sum_{s\in-s',j}\boldsymbol{A}^{s}_{i,j}\frac{1}{\sigma^2_s}}-2\log\frac{n_i^{-s'}}{S-n_i^{-s'}}, i\le J_{-s'}\\ -\frac{\|\frac{\boldsymbol{\mu}_0}{\sigma_0^2}+\frac{\boldsymbol{w}_{s'j}}{\sigma^2_{s'}}\|^2}{\frac{1}{\sigma_0^2}+\frac{1}{\sigma^2_{s'}}}+\frac{\|\frac{\boldsymbol{\mu}_0}{\sigma_0^2}\|^2}{\frac{1}{\sigma_0^2}}+2\log\frac{i-J_{-s'}}{\gamma_0/S}, J_{-s'}<i\le J_{-s'}+J_{s'},\end{cases} \tag{16}$$

The norm is l_2-norm. Therefore, the following inequality is useful for us to rewrite the above cost specification in this study.

As declared in PFNM, global neuron $\boldsymbol{\theta}_i$ can be estimated according to the Gaussian-Gaussian conjugacy:

$$\hat{\boldsymbol{\theta}}_i = \frac{\boldsymbol{\mu}_0/\sigma_0^2+\sum_{s,j}\boldsymbol{A}^{s}_{i,j}\boldsymbol{w}_{sj}/\sigma^2_s}{1/\sigma_0^2+\sum_{s,j}\boldsymbol{A}^{s}_{i,j}/\sigma^2_s}, \tag{17}$$

where $i = 1, 2, \ldots, J$. When $\boldsymbol{A}_{i,j}^{s'}$ is unknown for local model s', the global neuron $\boldsymbol{\theta}_i$ can be estimated by local neurons excluding neurons in model s', that is

$$\tilde{\boldsymbol{\theta}}_i = \frac{\boldsymbol{\mu}_0/\sigma_0^2 + \sum_{s\in -s',j} \boldsymbol{A}_{i,j}^{s} \boldsymbol{w}_{sj}/\sigma_s^2}{1/\sigma_0^2 + \sum_{s\in -s',j} \boldsymbol{A}_{i,j}^{s}/\sigma_s^2}. \tag{18}$$

For simplicity, we use characters to denote items as follows.

$$\begin{aligned}
&\boldsymbol{\alpha} = \boldsymbol{\mu}_0/\sigma_0^2 + \sum_{s\in -s',j} \boldsymbol{A}_{i,j}^{s} (\boldsymbol{w}_{sj}/\sigma_s^2),\\
&\boldsymbol{\beta} = \boldsymbol{w}_{s'j}/\sigma_{s'}^2,\\
&C_1 = 1/\sigma_0^2 + \sum_{s\in -s',j} \boldsymbol{A}_{i,j}^{s} (1/\sigma_s^2),\\
&C_2 = 1/\sigma_{s'}^2.
\end{aligned}$$

Besides, we denote $COST_\eta$ as the cost defined by norm η. For example, $COST_{l_2}$ denote the cost function defined by l_2-norm.

When $i \le J_{-s'}$ and when the norm is l_2-norm, the first two terms of the cost could be rewritten as

$$\begin{aligned}
&-\frac{\|\frac{\boldsymbol{\mu}_0}{\sigma_0^2} + \frac{\boldsymbol{w}_{s'j}}{\sigma_{s'}^2} + \sum_{s\in -s',j} \boldsymbol{A}_{i,j}^{s} \frac{\boldsymbol{w}_{sj}}{\sigma_s^2}\|_2^2}{\frac{1}{\sigma_0^2} + \frac{1}{\sigma_{s'}^2} + \sum_{s\in -s',j} \boldsymbol{A}_{i,j}^{s} \frac{1}{\sigma_s^2}} + \frac{\|\frac{\boldsymbol{\mu}_0}{\sigma_0^2} + \sum_{s\in -s',j} \boldsymbol{A}_{i,j}^{s} \frac{\boldsymbol{w}_{sj}}{\sigma_s^2}\|_2^2}{\frac{1}{\sigma_0^2} + \sum_{s\in -s',j} \boldsymbol{A}_{i,j}^{s} \frac{1}{\sigma_s^2}}\\
=&-\frac{\|\boldsymbol{\alpha} + \boldsymbol{\beta}\|_2^2}{C_1 + C_2} + \frac{\|\boldsymbol{\alpha}\|_2^2}{C_1}\\
=&\frac{\|\frac{\boldsymbol{\alpha}}{C_1} - \frac{\boldsymbol{\beta}}{C_2}\|_2^2 - (1 + \frac{C_2}{C_1})\|\frac{\boldsymbol{\beta}}{C_2}\|_2^2}{\frac{C_1+C_2}{C_1 C_2}} \qquad (19)\\
=&\frac{\|\frac{\boldsymbol{\mu}_0/\sigma_0^2 + \sum_{s\in -s',j} \boldsymbol{A}_{i,j}^{s} (\boldsymbol{w}_{sj}/\sigma_s^2)}{1/\sigma_0^2 + \sum_{s\in -s',j} \boldsymbol{A}_{i,j}^{s} (1/\sigma_s^2)} - \boldsymbol{w}_{s'j}\|_2^2 - (1 + \frac{1/\sigma_{s'}^2}{1/\sigma_0^2 + \sum_{s\in -s',j} \boldsymbol{A}_{i,j}^{s} (1/\sigma_s^2)})\|\boldsymbol{w}_{s'j}\|_2^2}{\frac{\frac{1}{\sigma_0^2} + \sum_{s\in -s',j} \boldsymbol{A}_{i,j}^{s} (\frac{1}{\sigma_s^2}) + \frac{1}{\sigma_{s'}^2}}{\left(\frac{1}{\sigma_0^2} + \sum_{s\in -s',j} \boldsymbol{A}_{i,j}^{s} (\frac{1}{\sigma_s^2})\right)(\frac{1}{\sigma_{s'}^2})}} \qquad (20)\\
=&\frac{\|\tilde{\theta}_i - \boldsymbol{w}_{s'j}\|_2^2 - (1 + \frac{1/\sigma_{s'}^2}{1/\sigma_0^2 + \sum_{s\in -s',j} \boldsymbol{A}_{i,j}^{s} (1/\sigma_s^2)})\|\boldsymbol{w}_{s'j}\|_2^2}{\frac{\frac{1}{\sigma_0^2} + \sum_{s\in -s',j} \boldsymbol{A}_{i,j}^{s} (\frac{1}{\sigma_s^2}) + \frac{1}{\sigma_{s'}^2}}{\left(\frac{1}{\sigma_0^2} + \sum_{s\in -s',j} \boldsymbol{A}_{i,j}^{s} (\frac{1}{\sigma_s^2})\right)(\frac{1}{\sigma_{s'}^2})}}, \qquad (21)\\
&\qquad (22)
\end{aligned}$$

Similarly, when $J_{-s'} < i \le J_{-s'} + J_{s'}$ and the norm is l_2-norm, we use characters to denote items as follows:

$$\begin{aligned}
&\tilde{\boldsymbol{\alpha}} = \boldsymbol{\mu}_0/\sigma_0^2,\\
&\boldsymbol{\beta} = \boldsymbol{w}_{s'j}/\sigma_{s'}^2,\\
&\tilde{C}_1 = 1/\sigma_0^2,
\end{aligned}$$

$$C_2 = 1/\sigma_{s'}^2.$$

The first two terms of the cost could be rewritten as

$$\begin{aligned}
&-\frac{\|\frac{\boldsymbol{\mu}_0}{\sigma_0^2}+\frac{\boldsymbol{w}_{s'j}}{\sigma_{s'}^2}\|_2^2}{\frac{1}{\sigma_0^2}+\frac{1}{\sigma_{s'}^2}}+\frac{\|\frac{\boldsymbol{\mu}_0}{\sigma_0^2}\|_2^2}{\frac{1}{\sigma_0^2}}\\
=&-\frac{\|\tilde{\boldsymbol{\alpha}}+\boldsymbol{\beta}\|_2^2}{\tilde{C}_1+C_2}+\frac{\|\tilde{\boldsymbol{\alpha}}\|_2^2}{\tilde{C}_1}\\
=&\frac{\|\frac{\tilde{\boldsymbol{\alpha}}}{\tilde{C}_1}-\frac{\boldsymbol{\beta}}{C_2}\|_2^2-(1+\frac{C_2}{\tilde{C}_1})\|\frac{\boldsymbol{\beta}}{C_2}\|_2^2}{\frac{\tilde{C}_1+C_2}{\tilde{C}_1C_2}} \qquad (23)\\
=&\frac{\|\boldsymbol{\mu}_0-\boldsymbol{w}_{s'j}\|_2^2-(1+\frac{\sigma_0^2}{\sigma_{s'}^2})\|\boldsymbol{w}_{s'j}\|_2^2}{(\frac{1}{\sigma_0^2}+\frac{1}{\sigma_{s'}^2})(\sigma_0^2\sigma_{s'}^2)}. \qquad (24)
\end{aligned}$$

Let

$$\begin{aligned}
L_1 &= (1+\frac{1/\sigma_{s'}^2}{1/\sigma_0^2+\sum_{s\in -s',j}\boldsymbol{A}_{i,j}^s(1/\sigma_s^2)}),\\
L_2 &= \big(\frac{1}{\sigma_0^2}+\sum_{s\in -s',j}\boldsymbol{A}_{i,j}^s\frac{1}{\sigma_s^2}+\frac{1}{\sigma_{s'}^2}\big),\\
L_3 &= \big((\frac{1}{\sigma_0^2}+\sum_{s\in -s',j}\boldsymbol{A}_{i,j}^s\frac{1}{\sigma_s^2})\frac{1}{\sigma_{s'}^2}\big),\\
L_4 &= (1+\frac{\sigma_0^2}{\sigma_{s'}^2}),
\end{aligned}$$

we get the formulation in main text.

■

5.2. Proof of Proposition 3

Proof

Without loss of generality, we denote $\boldsymbol{w}_{s'j}$ by $\boldsymbol{w}$, and assume $\boldsymbol{w} = (w_0, 0, \ldots, 0)^T$ and $w_0 >= 0$. We also assume the mean of global neurons $\boldsymbol{\mu}_0 = 0$. Then $\|\boldsymbol{w}\|_2 = |w_0|$. From Eq. (**??**), we have

$$\begin{aligned}
&P\Big(\mathrm{KL}(q\|p_{\boldsymbol{\theta}_{i^*}}) \geq \mathrm{KL}(q\|p_{\boldsymbol{\theta}_i})\Big)\\
=&P\big(q(\boldsymbol{\theta}_{i^*}) \leq q(\boldsymbol{\theta}_i)\big)\\
=&P(\|\boldsymbol{\theta}_{i^*}\|_2 \geq \|\boldsymbol{\theta}_i\|_2)\\
=&P(\|\boldsymbol{\theta}_{i^*}\|_2 \geq \|\boldsymbol{\theta}_i\|_2 \mid \|\boldsymbol{\theta}_i\|_2 = R_1, \|\boldsymbol{\theta}_{i^*}\|_2 = R_2) \qquad (25)
\end{aligned}$$

We firstly prove the conclusion for $\epsilon \geq 2\|\boldsymbol{w}\|_2$. Suppose $\boldsymbol{\theta}_i$ and $\boldsymbol{\theta}_{i^*}$ satisfy Assumption (A), that is, $\|\boldsymbol{\theta}_i - \boldsymbol{w}\|_2 = \|\boldsymbol{\theta}_{i^*} - \boldsymbol{w}\|_2 + \epsilon$, then they are on two hyper-spheres whose radii differ by ϵ. For

hyper-sphere $\|\boldsymbol{\theta}_i - \boldsymbol{w}\|_2^2 = R_1^2$, the shortest distance to the origin is $R_1 - \|\boldsymbol{w}\|_2$. For hyper-sphere $\|\boldsymbol{\theta}_{i^*} - \boldsymbol{w}\|_2^2 = R_2^2 = (R_1 - \epsilon)^2$, the longest distance to the origin is $\|\boldsymbol{w}\|_2 + R_2 = \|\boldsymbol{w}\|_2 + (R_1 - \epsilon)$. Apparently, if the minimum l_2-norm of $\boldsymbol{\theta}_i$ is greater than the maximum l_2-norm of $\boldsymbol{\theta}_{i^*}$, that is $\epsilon \geq 2\|\boldsymbol{w}\|_2$ which is induced from $R_1 - \|\boldsymbol{w}\|_2 \geq \|\boldsymbol{w}\|_2 + (R_1 - \epsilon)$, there would be no possibility that $\|\boldsymbol{\theta}_{i^*}\|_2 \geq \|\boldsymbol{\theta}_i\|_2$. By Eq. (25), when $\epsilon \geq 2\|\boldsymbol{w}\|_2$, $P\Big(\mathrm{KL}(q\|p_{\boldsymbol{\theta}_{i^*}}) \geq \mathrm{KL}(q\|p_{\boldsymbol{\theta}_i})\Big) = 0$.

We now prove the conclusion for $\epsilon < 2\|\boldsymbol{w}\|_2$. We will start from dimension $n = 1$. Then we prove the situation when $n = 2$. In the end , we will use $B_1(\epsilon)$ and $B_2(\epsilon)$ to approximate the lower bound for $n = k$.

When $n = 1$, from $\|\boldsymbol{\theta}_i - \boldsymbol{w}\|_2 = \|\boldsymbol{\theta}_{i^*} - \boldsymbol{w}\|_2 + \epsilon$, we know $\boldsymbol{\theta}_i = R_1 + \boldsymbol{w}$ or $-R_1 + \boldsymbol{w}$, and $\boldsymbol{\theta}_{i^*} = R_2 + \|\boldsymbol{w}\|_2$ or $-R_2 + \|\boldsymbol{w}\|_2$. In all four combinations of $\boldsymbol{\theta}_i$ and $\boldsymbol{\theta}_{i^*}$, there are two combinations which satisfy that $\|\boldsymbol{\theta}_{i^*}\|_2 \geq \|\boldsymbol{\theta}_i\|_2$. Hence, we have

$$
\begin{aligned}
&P_1(\|\boldsymbol{\theta}_{i^*}\|_2 \geq \|\boldsymbol{\theta}_i\|_2 \mid \|\boldsymbol{\theta}_i\|_2 = R_1, \|\boldsymbol{\theta}_{i^*}\|_2 = R_2) \\
=&P(\boldsymbol{\theta}_i = \|\boldsymbol{w}\|_2 - R_1, \boldsymbol{\theta}_{i^*} = \|\boldsymbol{w}\|_2 + R_2) + P(\boldsymbol{\theta}_i = \|\boldsymbol{w}\|_2 - R_1, \boldsymbol{\theta}_{i^*} = \|\boldsymbol{w}\|_2 - R_2) \\
=&P(\boldsymbol{\theta}_i = \|\boldsymbol{w}\|_2 - R_1)P(\boldsymbol{\theta}_{i^*} = \|\boldsymbol{w}\|_2 + R_2) + P(\boldsymbol{\theta}_i = \|\boldsymbol{w}\|_2 - R_1)P(\boldsymbol{\theta}_{i^*} = \|\boldsymbol{w}\|_2 - R_2) \\
=&\frac{q(\|\boldsymbol{w}\|_2 - R_1)q(\|\boldsymbol{w}\|_2 + R_2) + q(\|\boldsymbol{w}\|_2 - R_1)q(\|\boldsymbol{w}\|_2 - R_2)}{\big(q(R_1 + \|\boldsymbol{w}\|_2) + q(-R_1 + \|\boldsymbol{w}\|_2)\big)\big(q(R_2 + \|\boldsymbol{w}\|_2) + q(-R_2 + \|\boldsymbol{w}\|_2\big)} \\
=&\frac{e^{-\frac{(\|\boldsymbol{w}\|_2 - R_1)^2 + (\|\boldsymbol{w}\|_2 + R_2)^2}{2\sigma_0^2}} + e^{-\frac{(\|\boldsymbol{w}\|_2 - R_1)^2 + (\|\boldsymbol{w}\|_2 - R_2)^2}{2\sigma_0^2}}}{\big(e^{-\frac{(\|\boldsymbol{w}\|_2 + R_1)^2}{2\sigma_0^2}} + e^{-\frac{(\|\boldsymbol{w}\|_2 - R_1)^2}{2\sigma_0^2}}\big)\big(e^{-\frac{(\|\boldsymbol{w}\|_2 + R_2)^2}{2\sigma_0^2}} + e^{-\frac{(\|\boldsymbol{w}\|_2 - R_2)^2}{2\sigma_0^2}}\big)} \\
=&\frac{e^{\frac{R_1\|\boldsymbol{w}\|_2 - R_2\|\boldsymbol{w}\|_2}{\sigma_0^2}} + e^{\frac{R_1\|\boldsymbol{w}\|_2 + R_2\|\boldsymbol{w}\|_2}{\sigma_0^2}}}{(e^{-R_1\|\boldsymbol{w}\|_2/\sigma_0^2} + e^{R_1\|\boldsymbol{w}\|_2/\sigma_0^2})(e^{-R_2\|\boldsymbol{w}\|_2/\sigma_0^2} + e^{R_2\|\boldsymbol{w}\|_2/\sigma_0^2})} \\
=&\frac{e^{\frac{R_1\|\boldsymbol{w}\|_2}{\sigma_0^2}}}{e^{-R_1\|\boldsymbol{w}\|_2/\sigma_0^2} + e^{R_1\|\boldsymbol{w}\|_2/\sigma_0^2}} \\
=&B_1.
\end{aligned}
\tag{26}
$$

Apparently, B_1 is a constant.

When $n = 2$, as shown in Fig. 1, there are three total cases. The first case contain three different parts, of which the green arc and light blue arc contain $\boldsymbol{\theta}_i$ that makes $P(\|\boldsymbol{\theta}_{i^*}\|_2 \geq \|\boldsymbol{\theta}_i\|_2) > 0$. The second case contains two parts, of which the light blue arc contain $\boldsymbol{\theta}_i$ that makes $P(\|\boldsymbol{\theta}_{i^*}\|_2 \geq \|\boldsymbol{\theta}_i\|_2) > 0$. And in the third case,$P(\|\boldsymbol{\theta}_{i^*}\|_2 \geq \|\boldsymbol{\theta}_i\|_2) = 0$ induced by $\epsilon \geq 2\|\boldsymbol{w}\|_2$. As we can see, case 2 is a special case of case 1 when the green arc disappears. And case 3 is a special case of case 1 when the green arc and light blue arc disappear. Therefore, we only need to estimate the lower bound function of $P(\|\boldsymbol{\theta}_{i^*}\|_2 \geq \|\boldsymbol{\theta}_i\|_2)$ in case 1.

As before, we let $\boldsymbol{w} = \boldsymbol{w}_{s'j} = (w_0, 0)^{\mathrm{T}}$. As we assumed before, $\|\boldsymbol{\theta}_i\|_2 = R_1$ and $\|\boldsymbol{\theta}_{i^*}\|_2 = R_1 - \epsilon = R_2$. For the sake of simplicity, suppose $\boldsymbol{\theta}_i$ and $\boldsymbol{\theta}_{i^*}$ can be represented by parameter function

$$
\boldsymbol{\theta}_i = \begin{cases} R_1\cos\alpha + w_0 \\ R_1\sin\alpha \end{cases}, \qquad \boldsymbol{\theta}_{i^*} = \begin{cases} R_2\cos\beta + w_0 \\ R_2\sin\beta \end{cases},
$$

where α and β belong to $[0, 2\pi]$. Then, we have

$$
P(\|\boldsymbol{\theta}_{i^*}\|_2 \geq \|\boldsymbol{\theta}_i\|_2 \mid R_1, R_2)
$$

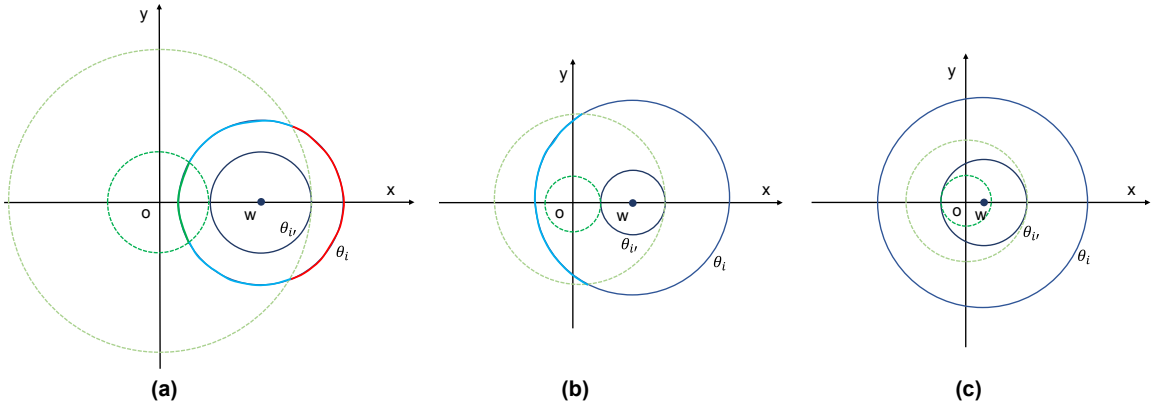


Figure 1: Three cases in estimating the lower bound of $P(\|\boldsymbol{\theta}_{i^*}\|_2 \geq \|\boldsymbol{\theta}_i\|_2)$ when $n = 2$. The first case contain three different parts, of which the green arc and light blue arc contain $\boldsymbol{\theta}_i$ that makes $P(\|\boldsymbol{\theta}_{i^*}\|_2 \geq \|\boldsymbol{\theta}_i\|_2) > 0$. The second case contains two parts, of which the light blue arc contain $\boldsymbol{\theta}_i$ that makes $P(\|\boldsymbol{\theta}_{i^*}\|_2 \geq \|\boldsymbol{\theta}_i\|_2) > 0$. And in the third case,$P(\|\boldsymbol{\theta}_{i^*}\|_2 \geq \|\boldsymbol{\theta}_i\|_2) = 0$ is induced by $\epsilon \geq 2\|\boldsymbol{w}\|_2$.

$$
\begin{aligned}
&= P\big((R_2\cos\beta + w_0)^2 + (R_2\sin\beta)^2 \geq (R_1\cos\alpha + w_0)^2 + (R_1\sin\alpha)^2 \mid R_1, R_2\big) \\
&= P(R_2^2 + 2w_0R_2\cos\beta \geq R_1^2 + 2w_0R_2\cos\alpha \mid R_1, R_2). \qquad (27)
\end{aligned}
$$

For the sake of briefness, we set $f_\epsilon(\alpha,\beta) = R_2^2 + 2w_0R_2\cos\beta - R_1^2 - 2w_0R_2\cos\alpha$, then Eq. (27) can be rewrote to

$$
\begin{aligned}
&P(f_\epsilon(\alpha,\beta) \geq 0 \mid R_1, R_2) \\
&= \int_{\tilde{\alpha}\in\Omega_\alpha} P\big(f_\epsilon(\alpha=\tilde{\alpha},\beta) \geq 0 \mid R_1, R_2\big) p(\tilde{\alpha} \mid R_1, R_2)\, d\tilde{\alpha} \\
&= \int_{\tilde{\alpha}\in\Omega_\alpha} \Big(\int_{\tilde{\beta}\in\Omega_\beta} P\big(f_\epsilon(\alpha=\tilde{\alpha},\beta=\tilde{\beta}) \geq 0 \mid R_1, R_2\big) p(\tilde{\beta} \mid R_2)\, d\tilde{\beta} \Big) p(\tilde{\alpha} \mid R_1)\, d\tilde{\alpha}, \qquad (28)
\end{aligned}
$$

where Ω_α and Ω_β are sets to which α and β belong. As shown in Fig. 2 (a), for the green arc of $\boldsymbol{\theta}_i$, $\boldsymbol{\theta}_i \in [\pi - \alpha_2, \pi + \alpha_1]$. And for fixed $\boldsymbol{\theta}_i$, Ω_β is $[0, 2\pi]$. α_1 can be computed via the Law of cosines, that is

$$
\alpha_1 = \arccos\frac{R_1^2 + w_0^2 - (w_0 - R_2)^2}{2R_1w_0} = \arccos\frac{R_1^2 - R_2^2 + 2R_2w_0}{2R_1w_0}. \qquad (29)
$$

As a consequence, for the green arc in Fig. 2 (a), we have

$$
\begin{aligned}
&P_g(f_\epsilon(\alpha,\beta) \geq 0 \mid R_1, R_2) \\
&= \int_{\pi-\alpha_1}^{\pi+\alpha_1} P\big(f_\epsilon(\alpha=\tilde{\alpha},\beta) \geq 0 \mid R_1, R_2\big) p(\tilde{\alpha} \mid R_1)\, d\tilde{\alpha} \\
&= \int_{\pi-\alpha_1}^{\pi+\alpha_1} p(\tilde{\alpha} \mid R_1)\, d\tilde{\alpha}. \qquad (30)
\end{aligned}
$$

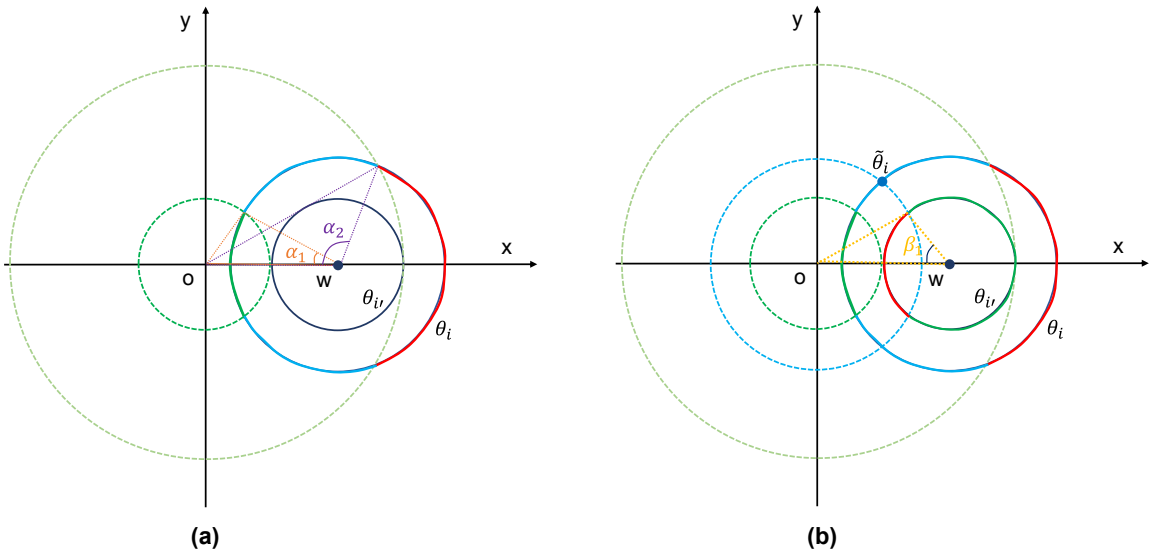


Figure 2: In the third case, there are two parts on the circle of $\boldsymbol{\theta}_i$, the green arc and light blue arc, which contain $\boldsymbol{\theta}_i$ that makes $P(\|\boldsymbol{\theta}_{i^*}\|_2 \geq \|\boldsymbol{\theta}_i\|_2) > 0$. For fixed $\boldsymbol{\theta}_i = \tilde{\boldsymbol{\theta}}_i$ on the green arc and light blue arc, the available $\boldsymbol{\theta}_{i^*}$ is on the smaller green arc.

For the light blue arc, we can also compute the α_2 in Fig. (2) (a) as

$$\alpha_2 = \arccos \frac{R_1^2 + w_0^2 - (w_0 + R_2)^2}{2R_1 w_0} = \arccos \frac{R_1^2 - R_2^2 - 2R_2 w_0}{2R_1 w_0}. \tag{31}$$

The Integration range of α is $[\pi - \alpha_2, \pi - \alpha_1] \cup [\pi + \alpha_1, \pi - \alpha_2]$. As shown Fig. (2) (b), for fixed $\tilde{\alpha}$, Ω_β is $[0, \pi - \beta_1] \cup [\pi + \beta_1, 2\pi]$. β_1 can also be computed by the Law of cosines as

$$\begin{aligned}
\beta_1 &= \arccos \frac{R_2^2 + w_0^2 - \|\boldsymbol{\theta}_i\|_2^2}{2R_2 w_0} \\
&= \arccos \frac{R_2^2 + w_0^2 - \left((R_1 \cos\tilde{\alpha} + w_0)^2 + (R_1 \sin\tilde{\alpha})^2\right)}{2R_2 w_0} \\
&= \arccos \frac{R_2^2 - R_1^2 - 2R_1 w_0 \cos\tilde{\alpha}}{2R_2 w_0}.
\end{aligned} \tag{32}$$

Consequently, for the light blue part, considering the symmetry of the integral, we have

$$\begin{aligned}
&P_b(f_\epsilon(\alpha, \beta) \geq 0 \,|\, R_1, R_2) \\
=&\int_{\tilde{\alpha} \in \Omega_\alpha} \left(\int_{\tilde{\beta} \in \Omega_\beta} P\big(f_\epsilon(\alpha = \tilde{\alpha}, \beta = \tilde{\beta}) \geq 0 \mid R_1, R_2\big) p(\tilde{\beta} \mid R_2)\, d\tilde{\beta} \right) p(\tilde{\alpha} \mid R_1)\, d\tilde{\alpha} \\
=&4 \int_{\pi - \alpha_2}^{\pi - \alpha_1} \left(\int_0^{\pi - \beta_1} P\big(f_\epsilon(\alpha = \tilde{\alpha}, \beta = \tilde{\beta}) \geq 0 \mid R_1, R_2\big) p(\tilde{\beta} \mid R_2)\, d\tilde{\beta} \right) p(\tilde{\alpha} \mid R_1)\, d\tilde{\alpha}
\end{aligned}$$

$$=4\int_{\pi-\alpha_2}^{\pi-\alpha_1}\Big(\int_{0}^{\pi-\beta_1} p(\tilde{\beta}\mid R_2)\,d\tilde{\beta}\Big)p(\tilde{\alpha}\mid R_1)\,d\tilde{\alpha}. \tag{33}$$

Thus,

$$\begin{aligned}
&P(f_\epsilon(\alpha,\beta)\geq 0\mid R_1,R_2)\\
=&P_g(f_\epsilon(\alpha,\beta)\geq 0\mid R_1,R_2)+P_b(f_\epsilon(\alpha,\beta)\geq 0\mid R_1,R_2)\\
=&\int_{\pi-\alpha_1}^{\pi+\alpha_1} p(\tilde{\alpha}\mid R_1)\,d\tilde{\alpha}+4\int_{\pi-\alpha_2}^{\pi-\alpha_1}\Big(\int_{0}^{\pi-\beta_1} p(\tilde{\beta}\mid R_2)\,d\tilde{\beta}\Big)p(\tilde{\alpha}\mid R_1)\,d\tilde{\alpha}.
\end{aligned} \tag{34}$$

By using Eq. (**??**), we can compute $p(\alpha)$ and $p(\beta)$ as follows.

$$\begin{aligned}
p(\alpha\,|\,R_1)&=p\big(\boldsymbol{\theta}_i(\alpha)\mid \|\boldsymbol{\theta}_i\|_2=R_1\big)\\
&=\frac{p\big(\boldsymbol{\theta}_i(\alpha)\big)}{p\big(\|\boldsymbol{\theta}_i\|_2=R_1\big)}\\
&=\frac{p(\boldsymbol{\theta}_i(\alpha))}{\int_{\|\boldsymbol{\theta}_i\|_2=R_1} p(\hat{\boldsymbol{\theta}}_i(\hat{\alpha}))\,d\hat{\boldsymbol{\theta}}_i(\hat{\alpha})}\\
&=\frac{p(\boldsymbol{\theta}_i(\alpha))}{\int_0^{2\pi} p(\hat{\boldsymbol{\theta}}_i(\hat{\alpha}))\,d\hat{\alpha}}
\end{aligned} \tag{35}$$

$$=\frac{e^{-R_1 w_0\cos\alpha/\sigma_0^2}}{\int_0^{2\pi} e^{-R_1 w_0\cos\hat{\alpha}/\sigma_0^2}d\hat{\alpha}}. \tag{36}$$

Similarly, we acquire

$$p(\beta\,|\,R_2)=\frac{e^{-R_2 w_0\cos\beta/\sigma_0^2}}{\int_0^{2\pi} e^{-R_2 w_0\cos\hat{\beta}/\sigma_0^2}d\hat{\beta}}. \tag{37}$$

For briefness, we let

$$\Delta_1=\int_0^{2\pi} e^{-R_1 w_0\cos\hat{\alpha}/\sigma_0^2}d\hat{\alpha},\Delta_2=\int_0^{2\pi} e^{-R_2 w_0\cos\hat{\beta}/\sigma_0^2}d\hat{\beta}.$$

Since $\cos x$ is lower bounded by $y=-x+1$ for $x\in[0,\pi]$, we have following upper bounds for Δ_1 and Δ_2:

$$\Delta_1\leq\int_0^{2\pi} e^{-R_1 w_0(-\hat{\alpha}+1)/\sigma_0^2}\,d\hat{\alpha}=\frac{\sigma_0^2}{R_1 w_0}e^{-R_1 w_0/\sigma_0^2}\big(e^{2\pi R_1 w_0/\sigma_0^2}-1\big), \tag{38}$$

$$\Delta_2\leq\int_0^{2\pi} e^{-R_2 w_0(-\hat{\beta}+1)/\sigma_0^2}\,d\hat{\beta}=\frac{\sigma_0^2}{R_2 w_0}e^{-R_2 w_0/\sigma_0^2}\big(e^{2\pi R_2 w_0/\sigma_0^2}-1\big). \tag{39}$$

Substitute the $p(\alpha\,|\,R_1)$ and $p(\beta\,|\,R_2)$ into Eq. (34), we get

$$\begin{aligned}
&P(f_\epsilon(\alpha,\beta)\geq 0\mid R_1,R_2)\\
=&\frac{1}{\Delta_1}\int_{\pi-\alpha_1}^{\pi+\alpha_1} e^{\frac{-R_1 w_0\cos\tilde{\alpha}}{\sigma_0^2}}\,d\tilde{\alpha}+\frac{1}{\Delta_1\Delta_2}\int_{\pi-\alpha_2}^{\pi-\alpha_1}\Big(\int_{0}^{\pi-\beta_1} e^{\frac{-R_2 w_0\cos\tilde{\beta}}{\sigma_0^2}}\,d\tilde{\beta}\Big)e^{\frac{-R_1 w_0\cos\tilde{\alpha}}{\sigma_0^2}}\,d\tilde{\alpha}
\end{aligned}$$

$$
\begin{aligned}
&\overset{(*)}{\geq} \frac{1}{\Delta_1}\int_{\pi-\alpha_1}^{\pi+\alpha_1} e^{\frac{R_1 w_0(\tilde{\alpha}+1-\pi)}{\sigma_0^2}}\, d\tilde{\alpha} + \frac{1}{\Delta_1\Delta_2}\int_{\pi-\alpha_2}^{\pi-\alpha_1}\Big(\int_0^{\pi-\beta_1} e^{\frac{R_2 w_0(\tilde{\beta}+1-\pi)}{\sigma_0^2}}\, d\tilde{\beta}\Big) e^{\frac{R_1 w_0(\tilde{\alpha}+1-\pi)}{\sigma_0^2}}\, d\tilde{\alpha} \\
&= \frac{\sigma_0^2 e^{\frac{(1-\pi)R_1 w_0}{\sigma_0^2}}}{R_1 w_0 \Delta_1}\int_{\frac{(\pi-\alpha_1)R_1 w_0}{\sigma_0^2}}^{\frac{(\pi+\alpha_1)R_1 w_0}{\sigma_0^2}} e^t\, dt + \frac{4\sigma_0^2 e^{\frac{(1-\pi)w_0(R_1+R_2)}{\sigma_0^2}}}{R_2 w_0 \Delta_1\Delta_2}\int_{\pi-\alpha_2}^{\pi-\alpha_1}\Big(\int_0^{\frac{(\pi-\beta_1)R_2 w_0}{\sigma_0^2}} e^{t'}\, dt'\Big) e^{\frac{R_1 w_0\tilde{\alpha}}{\sigma_0^2}}\, d\tilde{\alpha} \\
&= \frac{\sigma_0^2 e^{\frac{R_1 w_0}{\sigma_0^2}}}{R_1 w_0 \Delta_1}\Big(e^{\frac{\alpha_1 R_1 w_0}{\sigma_0^2}} - e^{-\frac{\alpha_1 R_1 w_0}{\sigma_0^2}}\Big) + \frac{4\sigma_0^2 e^{\frac{(1-\pi)w_0(R_1+R_2)}{\sigma_0^2}}}{R_2 w_0 \Delta_1\Delta_2}\int_{\pi-\alpha_2}^{\pi-\alpha_1}\Big(e^{\frac{(\pi-\beta_1)R_2 w_0}{\sigma_0^2}} - 1\Big) e^{\frac{R_1 w_0\tilde{\alpha}}{\sigma_0^2}}\, d\tilde{\alpha}, \qquad (40)
\end{aligned}
$$

where $(*)$ is induced by that the lower bound of $-\cos(x)$ is $y = x - \pi + 1$. We now substitute Δ_1 and Δ_2 into Eq. (40), we get

$$
\begin{aligned}
&P(f_\epsilon(\alpha,\beta) \geq 0 \mid R_1, R_2) \\
&\geq \frac{e^{\frac{2R_1 w_0}{\sigma_0^2}}}{e^{\frac{2\pi R_1 w_0}{\sigma_0^2}} - 1}\Big(e^{\frac{\alpha_1 R_1 w_0}{\sigma_0^2}} - e^{-\frac{\alpha_1 R_1 w_0}{\sigma_0^2}}\Big) \\
&\quad + \frac{4R_1 w_0}{\sigma_0^2}\frac{e^{\frac{(2-\pi)w_0(R_1+R_2)}{\sigma_0^2}}}{(e^{\frac{2\pi R_1 w_0}{\sigma_0^2}} - 1)(e^{\frac{2\pi R_2 w_0}{\sigma_0^2}} - 1)}\int_{\pi-\alpha_2}^{\pi-\alpha_1}\Big(e^{\frac{(\pi-\beta_1)R_2 w_0}{\sigma_0^2}} - 1\Big) e^{\frac{R_1 w_0\tilde{\alpha}}{\sigma_0^2}}\, d\tilde{\alpha} \\
&= \frac{e^{\frac{2R_1 w_0}{\sigma_0^2}}}{e^{\frac{2\pi R_1 w_0}{\sigma_0^2}} - 1}\Big(e^{\frac{\alpha_1 R_1 w_0}{\sigma_0^2}} - e^{-\frac{\alpha_1 R_1 w_0}{\sigma_0^2}}\Big) \\
&\quad + \frac{4R_1 w_0}{\sigma_0^2}\frac{e^{\frac{(2-\pi)w_0(2R_1-\epsilon)}{\sigma_0^2}}}{(e^{\frac{2\pi R_1 w_0}{\sigma_0^2}} - 1)(e^{\frac{2\pi(R_1-\epsilon)w_0}{\sigma_0^2}} - 1)}\int_{\pi-\alpha_2}^{\pi-\alpha_1}\Big(e^{\frac{(\pi-\beta_1)(R_1-\epsilon)w_0}{\sigma_0^2}} - 1\Big) e^{\frac{R_1 w_0\tilde{\alpha}}{\sigma_0^2}}\, d\tilde{\alpha} \\
&\geq \frac{e^{\frac{2R_1 w_0}{\sigma_0^2}}}{e^{\frac{2\pi R_1 w_0}{\sigma_0^2}} - 1}\Big(e^{\frac{\alpha_1 R_1 w_0}{\sigma_0^2}} - e^{-\frac{\alpha_1 R_1 w_0}{\sigma_0^2}}\Big) \\
&\quad + \frac{4R_1 w_0}{\sigma_0^2}\frac{e^{\frac{-(2\pi-4)R_1 w_0}{\sigma_0^2}}}{(e^{\frac{2\pi R_1 w_0}{\sigma_0^2}} - 1)(e^{\frac{2\pi R_1 w_0}{\sigma_0^2}} - 1)}\int_{\pi-\alpha_2}^{\pi-\alpha_1}\Big(e^{\frac{(\pi-\beta_1)(R_1-\epsilon)w_0}{\sigma_0^2}} - 1\Big) e^{\frac{R_1 w_0\tilde{\alpha}}{\sigma_0^2}}\, d\tilde{\alpha} \qquad (41) \\
&\geq \frac{4R_1 w_0}{\sigma_0^2}\frac{e^{\frac{-(2\pi-4)R_1 w_0}{\sigma_0^2}}}{(e^{\frac{2\pi R_1 w_0}{\sigma_0^2}} - 1)(e^{\frac{2\pi R_1 w_0}{\sigma_0^2}} - 1)}\int_{\pi-\alpha_2}^{\pi-\alpha_1}\Big(e^{\frac{(\pi-\beta_1)(R_1-\epsilon)w_0}{\sigma_0^2}} - 1\Big) e^{\frac{R_1 w_0\tilde{\alpha}}{\sigma_0^2}}\, d\tilde{\alpha}. \qquad (42)
\end{aligned}
$$

Since $\beta \in [0,\pi]$ and $\epsilon \in [0, R_1]$, the above equation is obviously non-negative. Now we will show that Eq. (42) is a monotonically decreasing function of ϵ.

According to $R_2 = R_1 - \epsilon$, Eq. (29) and Eq. (31), we have

$$
\alpha_1 = \arccos\frac{R_1^2 - (R_1-\epsilon)^2 + 2(R_1-\epsilon)w_0}{2R_1 w_0}
$$

$$= \arccos \frac{-[\epsilon - (R_1 - w_0)]^2 + (R_1 + w_0)^2 - 2R_1 w_0}{2R_1 w_0},$$

$$\begin{aligned}\alpha_2 &= \arccos \frac{R_1^2 - (R_1 - \epsilon)^2 - 2(R_1 - \epsilon) w_0}{2R_1 w_0} \\ &= \arccos \frac{-[\epsilon - (R_1 + w_0)]^2 + (R_1 + w_0)^2 - 2R_1 w_0}{2R_1 w_0}.\end{aligned}$$

Firstly, $\epsilon < R_1 < R_1 + w_0$ and $\arccos(x)$ is monotonically decreasing for $x \in [-1, 1]$. Thus, α_2 is a monotonically decreasing function of ϵ. Besides, from the three cases, we know that if $\alpha_1 > 0$, $w_0 - R_2 > R_1 - w_0$ must hold. This induce $\epsilon > 2(R_1 - w_0)$. Therefore, when $R_1 - w_0 < 0$, $\epsilon > R_1 - w_0$ because $\epsilon \geq 0$; when $R_1 - w_0 \geq 0$, $\epsilon > R_1 - w_0$ because $\epsilon > 2(R_1 - w_0)$. As a result, α_2 increases as ϵ increases and α_1 is a monotonically decreasing function of ϵ.

For the term of Eq. (42), the term integrated is non-negative. In addition, the integral range $[\pi - \alpha_1, \pi - \alpha_2]$ will expand as ϵ increases. Therefore, Eq. (42) is a monotonically decreasing function of ϵ.

We already proved (B) in the proposition holds for $n = 1, 2$. Now we need to prove (B) in the proposition holds for $\boldsymbol{\theta}_i$, $\boldsymbol{\theta}_{i^*} \in \mathbb{R}^n$ where $n > 2$. In the first place, we assume (B) holds for $n - 1$. This means

$$P_{n-1}(\|\boldsymbol{\theta}_{i^*}\|_2 \geq \|\boldsymbol{\theta}_i\|_2 \mid R_1, R_2) \geq B_{n-1}(\epsilon), \tag{43}$$

where $B_{n-1}(\epsilon)$ is monotonically decreasing and non-negative. For clearness, we write n dimensional $\boldsymbol{\theta}_i$, $\boldsymbol{\theta}_{i^*}$ as $\boldsymbol{\theta}_{i,n}$, $\boldsymbol{\theta}_{i^*,n}$ separately. Therefore,

$$\begin{aligned}\boldsymbol{\theta}_i &= (\theta_{i,n}^{(1)}, \theta_{i,n}^{(2)}, \ldots, \theta_{i,n}^{(n-1)}, \theta_{i,n}^{(n)})^{\mathrm{T}} = (\overline{\boldsymbol{\theta}}_{i,n-1}^{\mathrm{T}}, \boldsymbol{\theta}_{i,n}^{(n)})^{\mathrm{T}}, \\ \boldsymbol{\theta}_{i^*} &= (\theta_{i^*,n}^{(1)}, \theta_{i^*,n}^{(2)}, \ldots, \theta_{i^*,n}^{(n-1)}, \theta_{i^*,n}^{(n)})^{\mathrm{T}} = (\overline{\boldsymbol{\theta}}_{i^*,n-1}^{\mathrm{T}}, \boldsymbol{\theta}_{i^*,n}^{(n)})^{\mathrm{T}}.\end{aligned}$$

This induces that

$$\begin{aligned}\|\boldsymbol{\theta}_{i,n}\|_2^2 &= \|\overline{\boldsymbol{\theta}}_{i,n-1}\|_2^2 + \|\boldsymbol{\theta}_{i,n}^{(n)}\|_2^2, \\ \|\boldsymbol{\theta}_{i^*,n}\|_2^2 &= \|\overline{\boldsymbol{\theta}}_{i^*,n-1}\|_2^2 + \|\boldsymbol{\theta}_{i^*,n}^{(n)}\|_2^2.\end{aligned}$$

Using the above results, we have

$$\begin{aligned}
&P_n(\|\boldsymbol{\theta}_{i^*}\|_2 \geq \|\boldsymbol{\theta}_i\|_2 \,|\, R_1, R_2) \\
=&P_n(\|\boldsymbol{\theta}_{i^*}\|_2^2 \geq \|\boldsymbol{\theta}_i\|_2^2 \,|\, R_1, R_2) \\
=&P_n(\|\overline{\boldsymbol{\theta}}_{i^*,n-1}\|_2^2 + \|\boldsymbol{\theta}_{i^*,n}^{(n)}\|_2^2 \geq \|\overline{\boldsymbol{\theta}}_{i,n-1}\|_2^2 + \|\boldsymbol{\theta}_{i,n}^{(n)}\|_2^2 \,|\, R_1, R_2) \\
=&P_n(\|\boldsymbol{\theta}_{i^*}\|_2^2 \geq \|\boldsymbol{\theta}_i\|_2^2 \,|\, R_1, R_2, \|\overline{\boldsymbol{\theta}}_{i^*,n-1}\|_2^2 \geq \|\overline{\boldsymbol{\theta}}_{i,n-1}\|_2^2, \|\boldsymbol{\theta}_{i^*,n}^{(n)}\|_2^2 \geq \|\boldsymbol{\theta}_{i,n}^{(n)}\|_2^2) \\
&+ P_n(\|\boldsymbol{\theta}_{i^*}\|_2^2 \geq \|\boldsymbol{\theta}_i\|_2^2 \,|\, R_1, R_2, \|\overline{\boldsymbol{\theta}}_{i^*,n-1}\|_2^2 \leq \|\overline{\boldsymbol{\theta}}_{i,n-1}\|_2^2, \|\boldsymbol{\theta}_{i^*,n}^{(n)}\|_2^2 \geq \|\boldsymbol{\theta}_{i,n}^{(n)}\|_2^2) \\
&+ P_n(\|\boldsymbol{\theta}_{i^*}\|_2^2 \geq \|\boldsymbol{\theta}_i\|_2^2 \,|\, R_1, R_2, \|\overline{\boldsymbol{\theta}}_{i^*,n-1}\|_2^2 \geq \|\overline{\boldsymbol{\theta}}_{i,n-1}\|_2^2, \|\boldsymbol{\theta}_{i^*,n}^{(n)}\|_2^2 \leq \|\boldsymbol{\theta}_{i,n}^{(n)}\|_2^2) \\
\geq&P_n(\|\boldsymbol{\theta}_{i^*}\|_2^2 \geq \|\boldsymbol{\theta}_i\|_2^2 \,|\, R_1, R_2, \|\overline{\boldsymbol{\theta}}_{i^*,n-1}\|_2^2 \geq \|\overline{\boldsymbol{\theta}}_{i,n-1}\|_2^2, \|\boldsymbol{\theta}_{i^*,n}^{(n)}\|_2^2 \geq \|\boldsymbol{\theta}_{i,n}^{(n)}\|_2^2) \\
=&P_n(\|\overline{\boldsymbol{\theta}}_{i^*,n-1}\|_2^2 \geq \|\overline{\boldsymbol{\theta}}_{i,n-1}\|_2^2, \|\boldsymbol{\theta}_{i^*,n}^{(n)}\|_2^2 \geq \|\boldsymbol{\theta}_{i,n}^{(n)}\|_2^2 \,|\, R_1, R_2)
\end{aligned}$$

$$
\begin{aligned}
&= P_{n-1}(\|\overline{\boldsymbol{\theta}}_{i^*,n-1}\|_2^2 \geq \|\overline{\boldsymbol{\theta}}_{i,n-1}\|_2^2 \,|\, R_1, R_2) P_1(\|\boldsymbol{\theta}_{i^*,n}^{(n)}\|_2^2 \geq \|\boldsymbol{\theta}_{i,n}^{(n)}\|_2^2 \,|\, R_1, R_2) \\
&\geq B_{n-1}(\epsilon) B_1. \qquad (44)
\end{aligned}
$$

Repeat the recursion process, and choose the maximum function between $B_2(\epsilon)$ and $B_1(\epsilon)$, we get

$$
B_n(\epsilon) = \begin{cases} \max\{B_1^n, B_2(\epsilon)^{n/2}\}, \text{n is even}, \\ \max\{B_1^n, B_2(\epsilon)^{(n-1)/2} B_1(\epsilon)\}, \text{n is odd}, \end{cases} \qquad (45)
$$

where $B_1, B_2(\epsilon)$ are referred to Eq. (26) and Eq. (42) respectively. ■

5.3. Proof of Proposition 4

Proof Combine Eq. (13) in the main text with Eq. (17) it can cast optimization with respect to only $\{\boldsymbol{A}^s\}_{s=1}^S$,

$$
\min_{\{\boldsymbol{A}^s\}} \frac{1}{2} \sum_i \sum_{s,j} \boldsymbol{A}_{i,j}^s \left(\frac{||\sum_{s,j} \boldsymbol{A}_{i,j}^s \frac{\boldsymbol{w}_{sj} - \boldsymbol{\mu}_0}{\sigma_s^3}||^2}{(1/\sigma_0^2 + \sum_{s,j} \boldsymbol{A}_{i,j}^s / \sigma_s^2)^2} + (D+K)(\frac{\sigma_0^2}{\sigma_s^2} - 1 + \log \frac{\sigma_s^2}{\sigma_0^2}) \right), \qquad (46)
$$

We now consider the first part of Eq. (46). We partition it between $i = 1, \cdots, J_{-s'}$ and $i = J_{-s'} + 1, \cdots, J_{-s'} + J_{s'}$, and since we are solving for $\boldsymbol{A}^{s'}$, we can subtract terms independent of $\boldsymbol{A}^{s'}$ (we use $\cong$ to say that two objective functions are equivalent up to terms independent of the interested variables):

$$
\begin{aligned}
&\sum_i \sum_{s,j} \boldsymbol{A}_{i,j}^s \frac{\left\| \sum_{s,j} \boldsymbol{A}_{i,j}^s \frac{\boldsymbol{w}_{sj} - \boldsymbol{\mu}_0}{\sigma_s^3} \right\|^2}{(1/\sigma_0^2 + \sum_{s,j} \boldsymbol{A}_{i,j}^s / \sigma_s^2)^2} \\
&\cong \sum_{i=1}^{J_{-s'}} \left(\sum_j \boldsymbol{A}_{i,j}^{s'} \frac{\left\| \sum_j \boldsymbol{A}_{i,j}^{s'} \frac{\boldsymbol{w}_{s'j} - \boldsymbol{\mu}_0}{\sigma_{s'}^3} + \sum_{s \in -s',j} \boldsymbol{A}_{i,j}^s \frac{\boldsymbol{w}_{sj} - \boldsymbol{\mu}_0}{\sigma_s^3} \right\|^2}{(1/\sigma_0^2 + \sum_j \boldsymbol{A}_{i,j}^{s'} / \sigma_{s'}^2 + \sum_{s \in -s',j} \boldsymbol{A}_{i,j}^s / \sigma_s^2)^2} \right. \\
&\quad + \sum_{s \in -s',j} \boldsymbol{A}_{i,j}^s \frac{\left\| \sum_j \boldsymbol{A}_{i,j}^{s'} \frac{\boldsymbol{w}_{s'j} - \boldsymbol{\mu}_0}{\sigma_{s'}^3} + \sum_{s \in -s',j} \boldsymbol{A}_{i,j}^s \frac{\boldsymbol{w}_{sj} - \boldsymbol{\mu}_0}{\sigma_s^3} \right\|^2}{(1/\sigma_0^2 + \sum_j \boldsymbol{A}_{i,j}^{s'} / \sigma_{s'}^2 + \sum_{s \in -s',j} \boldsymbol{A}_{i,j}^s / \sigma_s^2)^2} \\
&\quad \left. - \sum_{s \in -s',j} \boldsymbol{A}_{i,j}^s \frac{\left\| \sum_{s \in -s',j} \boldsymbol{A}_{i,j}^s \frac{\boldsymbol{w}_{sj} - \boldsymbol{\mu}_0}{\sigma_s^3} \right\|^2}{(1/\sigma_0^2 + \sum_{s \in -s',j} \boldsymbol{A}_{i,j}^s / \sigma_s^2)^2} \right) + \sum_{i=J_{-s'}+1}^{J_{-s'}+J_{s'}} \left(\sum_j \boldsymbol{A}_{i,j}^{s'} \frac{\left\| \sum_j \boldsymbol{A}_{i,j}^{s'} \frac{\boldsymbol{w}_{s'j} - \boldsymbol{\mu}_0}{\sigma_{s'}^3} \right\|^2}{(1/\sigma_0^2 + \sum_j \boldsymbol{A}_{i,j}^{s'} / \sigma_{s'}^2)^2} \right),
\end{aligned} \qquad (47)
$$

observe that $\sum_j \boldsymbol{A}_{i,j}^{s'} \in \{0,1\}$, i.e., it equals to 1 if some neuron from dataset s' is matched to global neuron i and 0 otherwise. Thus Eq. (47) can be rewritten as a linear sum assignment problem:

$$
\sum_{i=1}^{J_{-s'}} \sum_{j=1}^{J_{s'}} \boldsymbol{A}_{i,j}^{s'} \left(\frac{\left\| \frac{\boldsymbol{w}_{s'j} - \boldsymbol{\mu}_0}{\sigma_{s'}^3} + \sum_{s \in -s',j} \boldsymbol{A}_{i,j}^s \frac{\boldsymbol{w}_{sj} - \boldsymbol{\mu}_0}{\sigma_s^3} \right\|^2}{(1/\sigma_0^2 + 1/\sigma_{s'}^2 + \sum_{s \in -s',j} \boldsymbol{A}_{i,j}^s / \sigma_s^2)^2} + \left(\frac{\left\| \frac{\boldsymbol{w}_{s'j} - \boldsymbol{\mu}_0}{\sigma_{s'}^3} + \sum_{s \in -s',j} \boldsymbol{A}_{i,j}^s \frac{\boldsymbol{w}_{sj} - \boldsymbol{\mu}_0}{\sigma_s^3} \right\|^2}{(1/\sigma_0^2 + 1/\sigma_{s'}^2 + \sum_{s \in -s',j} \boldsymbol{A}_{i,j}^s / \sigma_s^2)^2} \right.
$$

$$-\frac{\Big\| \sum\limits_{s\in -s',j} \boldsymbol{A}^{s}_{i,j} \frac{\boldsymbol{w}_{sj}-\boldsymbol{\mu}_0}{\sigma_s^3} \Big\|^2}{(1/\sigma_0^2+\sum\limits_{s\in -s',j} \boldsymbol{A}^{s}_{i,j}/\sigma_s^2)^2}\Big) n_i^{-s'}\Big) + \sum_{i=J_{-s'}+1}^{J_{-s'}+J_{s'}} \sum_{j=1}^{J_{s'}} \boldsymbol{A}^{s'}_{i,j} \Big(\frac{\| \frac{\boldsymbol{w}_{s'j}-\boldsymbol{\mu}_0}{\sigma_{s'}^3} \|^2}{(1/\sigma_0^2+\sum_j \boldsymbol{A}^{s'}_{i,j}/\sigma_{s'}^2)^2} \Big). \tag{48}$$

Then we consider the second term of Eq. (46). By subtracting terms independent of $\boldsymbol{A}^{s'}$ it has:

$$\sum_i \sum_j \boldsymbol{A}^{s'}_{i,j}(D+K)\Big(\frac{\sigma_0^2}{\sigma_{s'}^2} - 1 + \log\frac{\sigma_{s'}^2}{\sigma_0^2}\Big). \tag{49}$$

Because Eq. (49) is only varied by s', that means it adds equal cost in each item of $\{\widetilde{\boldsymbol{C}}^{s'}_{i,j}\}_{i,j}$, thus we can ignore it in the cost specification. From what has been discussed above, we obtain the assignment cost specification. ■

6. Experiment

6.1. Hyperparameters of training neural network

Hyperparameters of training neural networks in our experiments are shown in Table 1.

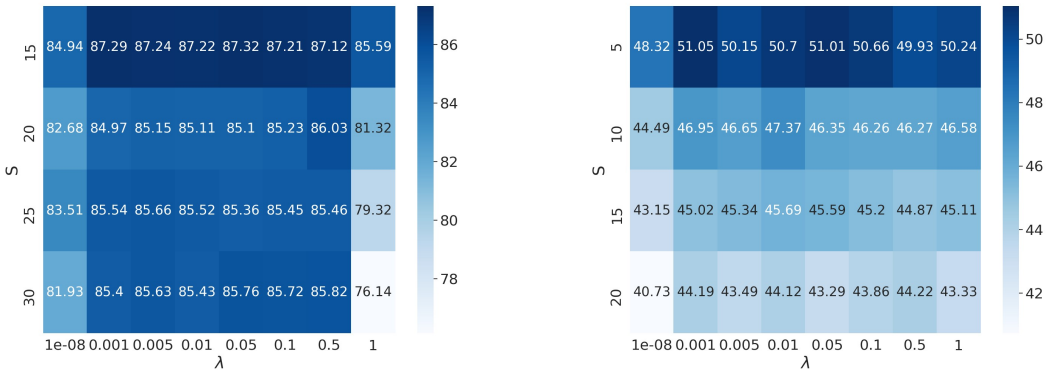
Table 1: Hyperparameter settings for training neural networks

	MNIST	CIFAR-10	CIMC
Model	FCNN	ConvNet	U-net
Optimizer	Adam	SGD	RMSprop
Learning rate	0.01	0.01	0.01
Size of minibatch	32	32	3
Epochs	10	10	3

6.2. Sensitivity Analysis

In GI-FNM, we induce a hyper-parameter λ, the weight of the KL-divergence term. Therefore, it is necessary to testify the sensitivity of λ. Here, we set λ to various positive values for GI-FNM applied on fusing FCNs and CNNs trained in MNIST and CIFAR10 respectively. We also consider the effect of number of clients. As shown in Fig. 3, the heat map indicates the accuracy on the test data for different data sets, and for various neural network types and number of clients, there is only tiny fluctuation of prediction accuracy for fused global model when $10^{-8} \leq \lambda \leq 1$. Although GI-FNM with $\lambda = 10^{-8}$ performs significantly worse than GI-FNM with $10^{-3} \leq \lambda \leq 0.5$, the performance of fused global model still maintains a high level. The performance of fused global model declines suddenly when $\lambda = 1$ but is still high . In summary, GI-FNM is robust on hyper-parameter λ under various conditions.

As we described in Sect. 4 of the main text, KL divergence term can be viewed as a regularization term related to selecting global neurons. From an optimization perspective, It is obvious that the performance of GI-FNM will close to PFNM when λ decreases to number around zero.



(a) MNIST multiple nets

(b) CIFAR10 multiple nets

Figure 3: **KL regularization coefficients sensitivity analysis**

This explains why the performance of GI-FNM declines when $\lambda = 10^{-8}$. In Sect. 4 of the main text , we also theoretically prove that KL term we induced can fix the drawback of PFNM, i.e., KL penalty has a higher probability of selecting a higher prior probability global neuron when the difference among distance of different global neurons to the local neuron is small, and it will select global neurons closer to local neurons as the original PFNM when the difference among distance of different global neurons to the local neuron is large. In other words, KL penalty can function theoretically only after the local neurons matched to the same global neuron were well selected by PFNM. This explains why the performance of GI-FNM declines suddenly when $\lambda = 1$.

References

Amir Beck. *First-order methods in optimization*. SIAM, 2017.

Davis Blalock, Jose Javier Gonzalez Ortiz, Jonathan Frankle, and John Guttag. What is the state of neural network pruning? *arXiv preprint arXiv:2003.03033*, 2020.

Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010.

Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541, 2006.

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

Yu Cheng, Duo Wang, Pan Zhou, and Tao Zhang. A survey of model compression and acceleration for deep neural networks. *arXiv preprint arXiv:1710.09282*, 2017.

Sebastian Claici, Mikhail Yurochkin, Soumya Ghosh, and Justin Solomon. Model fusion with Kullback–Leibler divergence. *arXiv preprint arXiv:2007.06168*, 2020.

Yuyang Deng, Mohammad Mahdi Kamani, and Mehrdad Mahdavi. Distributionally robust federated averaging. *arXiv preprint arXiv:2102.12660*, 2021.

Getao Du, Xu Cao, Jimin Liang, Xueli Chen, and Yonghua Zhan. Medical image segmentation based on U-net: A review. *Journal of Imaging Science and Technology*, 64(2):20508–1, 2020.

David B Dunson. Commentary: practical advantages of bayesian analysis of epidemiologic data. *American journal of Epidemiology*, 153(12):1222–1226, 2001.

Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. *Cited on*, 14(8):2, 2012.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*, pages 5132–5143. PMLR, 2020.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Harold W Kuhn. The Hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *arXiv preprint arXiv:1812.06127*, 2018.

Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3):50–60, 2020.

Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. *arXiv preprint arXiv:2006.07242*, 2020.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pages 1273–1282, 2017.

Mehryar Mohri, Gary Sivek, and Ananda Theertha Suresh. Agnostic federated learning. In *International Conference on Machine Learning*, pages 4615–4625, 2019.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

Jürgen Schmidhuber. Learning complex, extended sequences using the principle of history compression. *Neural Computation*, 4(2):234–242, 1992.

Sidak Pal Singh and Martin Jaggi. Model fusion via optimal transport. *Advances in Neural Information Processing Systems*, 33, 2020.

Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet S Talwalkar. Federated multi-task learning. In *Advances in Neural Information Processing Systems*, pages 4424–4434, 2017.

Romain Thibaux and Michael I. Jordan. Hierarchical Beta processes and the Indian buffet process. *Journal of Machine Learning Research*, 2(3):564–571, 2007.

Hongyi Wang, Mikhail Yurochkin, Yuekai Sun, Dimitris Papailiopoulos, and Yasaman Khazaeni. Federated learning with matched averaging. *arXiv preprint arXiv:2002.06440*, 2020.

Peng Xiao, Samuel Cheng, Vladimir Stankovic, and Dejan Vukobratovic. Averaging is probably not the optimum way of aggregating parameters in federated learning. *Entropy*, 22(3):314, 2020.

Mikhail Yurochkin, Zhiwei Fan, Aritra Guha, Paraschos Koutris, and XuanLong Nguyen. Scalable inference of topic evolution via models for latent geometric structures. *arXiv preprint arXiv:1809.08738*, 2018.

Mikhail Yurochkin, Mayank Agarwal, Soumya Ghosh, Kristjan Greenewald, and Nghia Hoang. Statistical model aggregation via parameter matching. *Advances in Neural Information Processing Systems*, 32:10956–10966, 2019a.

Mikhail Yurochkin, Mayank Agarwal, Soumya Ghosh, Kristjan Greenewald, Nghia Hoang, and Yasaman Khazaeni. Bayesian nonparametric federated learning of neural networks. In *International Conference on Machine Learning*, pages 7252–7261, 2019b.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3): 107–115, 2021.

Xinwei Zhang, Mingyi Hong, Sairaj Dhople, Wotao Yin, and Yang Liu. Fedpd: A federated learning framework with optimal rates and adaptivity to non-iid data. *arXiv preprint arXiv:2005.11418*, 2020.

Zhi-Hua Zhou. Why over-parameterization of deep neural networks does not overfit? *Science China Information Sciences*, 64(1):1–3, 2021.

Zhuangdi Zhu, Junyuan Hong, and Jiayu Zhou. Data-free knowledge distillation for heterogeneous federated learning. *arXiv preprint arXiv:2105.10056*, 2021.