Traversing Between Modes in Function Space for Fast Ensembling

EungGu Yun[1,2] Hyungi Lee[1] Giung Nam[1] Juho Lee[1,3]

[1]Korea Advanced Institute of Science and Technology (KAIST)

[2]Saige Research

[3]AITRICS

Abstract

- **Deep Ensemble (DE)** is a simple yet powerful way to improve the performance of deep neural networks.
- Using **mode connectivity**, one can efficiently collect ensemble parameters in low-loss subspaces.
- However, for inference, one should still execute multiple forward passes.
- We propose a novel framework “bridge network” to **reduce inference costs** using mode connectivity properties in function space.

Deep Ensemble [Lakshminarayanan et al., 2017] is a simple algorithm that ensembles multiple neural networks where each network is trained with different random seeds.

Benefits of DE:

- Simple to implement.
- Improves both accuracy and uncertainty calibration.
- Easy to parallelize.

Drawbacks of DE:

- Requires multiple training runs.
- Requires multiple forward passes for inference.

$\rightarrow$ Computational cost increases with the number of ensembles.

Mode Connectivity

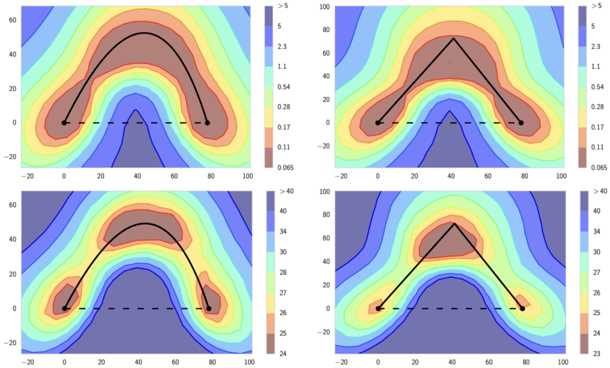

Figure 1: Two modes in the loss surface and the connecting subspace. Left: *Bezier curve*, Right: *Polygonal chain*. (Figure from Garipov et al. [2018])

Garipov et al. [2018] and Draxler et al. [2018] showed that **modes** (local optima) in the loss surface of a deep neural network **are connected** by relatively simple low-dimensional subspaces where the loss in the subspace retains low values.

Low-loss Subspaces

We focus on quadratic ***Bezier curves*** [Garipov et al., 2018]. Let $\boldsymbol{\theta}_i$ and $\boldsymbol{\theta}_j$ be two parameters of a neural network. The quadratic Bezier curve between them is defined as

$$\left\{(1-r)^2\boldsymbol{\theta}_i + 2r(1-r)\boldsymbol{\theta}_{i,j}^{(\text{be})} + r^2\boldsymbol{\theta}_j \mid r \in [0,1]\right\}, \tag{1}$$

where $\boldsymbol{\theta}_{i,j}^{(\text{be})}$ is a *pin-point* parameter characterizing the curve. A low-loss subspace connecting $(\boldsymbol{\theta}_i, \boldsymbol{\theta}_j)$ is found w.r.t. $\boldsymbol{\theta}_{i,j}^{(\text{be})}$ by minimizing

$$\int_0^1 \mathcal{L}\left(\boldsymbol{\theta}_{i,j}^{(\text{be})}(r)\right)\mathrm{d}r, \tag{2}$$

where $\boldsymbol{\theta}_{i,j}^{(\text{be})}(r)$ denotes the point at the position r of the curve,

$$\boldsymbol{\theta}_{i,j}^{(\text{be})}(r) = (1-r)^2\boldsymbol{\theta}_i + 2r(1-r)\boldsymbol{\theta}_{i,j}^{(\text{be})} + r^2\boldsymbol{\theta}_j, \tag{3}$$

and $\mathcal{L} : \boldsymbol{\Theta} \to \mathbb{R}$ is the loss function evaluating parameters.

Ensemble with Bezier curves

Let $\{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_m\}$ be a set of parameters independently trained as a deep ensemble. Then, for each pair $(\boldsymbol{\theta}_i, \boldsymbol{\theta}_j)$, we can construct a low-loss Bezier curve. For instance, choosing $r = 0.5$, we can collect $\boldsymbol{\theta}_{i,j}^{(\text{be})}(0.5)$ for all (i, j) pairs, and construct an ensembled predictor as

$$\frac{1}{m + \binom{m}{2}} \left(\sum_{i=1}^{m} f_{\boldsymbol{\theta}_i}(\boldsymbol{x}) + \sum_{i<j} f_{\boldsymbol{\theta}_{i,j}^{(\text{be})}(0.5)}(\boldsymbol{x}) \right). \tag{4}$$

While this strategy provide an effective way to increase the number of ensemble members, for inference, an additional $\mathcal{O}(m^2)$ number of forward passes are required.

Goal

Reduce the inference cost *using mode connectivity properties in function space.*

How: Directly approximate the outputs evaluated at the subspace with a small auxiliary network, which is called "**bridge network**".

Assumption: if two modes are connected by a simple subspace, we can predict the outputs corresponding to the parameters on the subspace using *only the outputs computed from the modes*.

The bridge network *lets us travel between modes in the function space.*

Method Overview

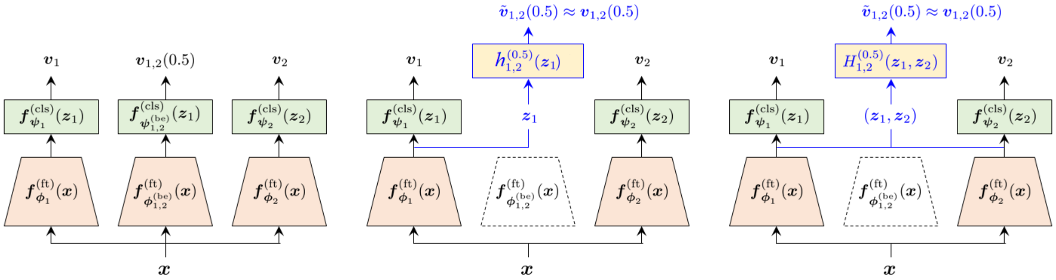

Figure 2: Ensembles with a Bezier curve (**left**), a type I bridge network (**center**), and a type II bridge network (**right**).

Assumption (revisited): if two modes are connected by a simple low-loss subspace (Bezier curve), then we can predict the outputs corresponding to the parameters on the subspace using only the information obtained from the modes.

If such mapping exists, we may learn them via **a lightweight neural network**.

- **Features**: $\boldsymbol{z}_i := f^{(\text{ft})}_{\phi_i}(\boldsymbol{x})$
- **Output**: $\boldsymbol{v}_i := f_{\boldsymbol{\theta}_i}(\boldsymbol{x}) = f^{(\text{cls})}_{\psi_i}(\boldsymbol{z}_i)$

- **Features (Bezier curve)**: $\boldsymbol{z}_{i,j}(r) := f_{\phi^{(\text{be})}_{i,j}(r)}(\boldsymbol{x})$
- **Output (Bezier curve)**: $\boldsymbol{v}_{i,j}(r) := f_{\boldsymbol{\theta}^{(\text{be})}_{i,j}(r)}(\boldsymbol{x}) = f^{(\text{cls})}_{\psi^{(\text{be})}_{i,j}(r)}(\boldsymbol{z}_{i,j}(r))$

We **reuse** features $\boldsymbol{z}_i$ to predict $\boldsymbol{v}_{i,j}(r)$ with a lightweight neural network, which lets us **directly move from $\boldsymbol{v}_i$ to $\boldsymbol{v}_{i,j}(r)$ in the function space**.

A bridge network is usually constructed with a Convolutional Neural Network (CNN) whose inference cost is much lower than that of $f_{\boldsymbol{\theta}_i}$.

Type I Bridge Networks

A type I bridge network $h^{(r)}_{i,j}$ takes a feature $\boldsymbol{z}_i$ from only one mode, and predicts

$$\boldsymbol{v}_{i,j}(r) \approx \tilde{\boldsymbol{v}}_{i,j}(r) = h^{(r)}_{i,j}(\boldsymbol{z}_i). \tag{5}$$

An ensembled prediction with the type I bridge network is then constructed as

$$\frac{1}{2}\left(\boldsymbol{v}_i + h^{(r)}_{i,j}(\boldsymbol{z}_i)\right), \tag{6}$$

whose inference cost is not much higher than that of $\boldsymbol{v}_i$. One can also connect $\boldsymbol{\theta}_i$ with multiple modes $\{\boldsymbol{\theta}_{j_1}, \dots, \boldsymbol{\theta}_{j_k}\}$, learn bridge networks between $(i, j_1), \dots, (i, j_k)$, and construct an ensemble

$$\frac{1}{1+k}\left(\boldsymbol{v}_i + \sum_{j=1}^{k} h^{(r)}_{i,j_k}(\boldsymbol{z}_i)\right). \tag{7}$$

Still, since the costs for $h^{(r)}_{i,j_k}$s are far lower than $\boldsymbol{v}_i$, the inference cost does not significantly increase.

Type II Bridge Networks

A type II bridge network $H_{i,j}^{(r)}$ between $(\boldsymbol{\theta}_i, \boldsymbol{\theta}_j)$ takes two features $(\boldsymbol{z}_i, \boldsymbol{z}_j)$, and predicts

$$\boldsymbol{v}_{i,j}(r) \approx \tilde{\boldsymbol{v}}_{i,j}(r) = H_{i,j}^{(r)}(\boldsymbol{z}_i, \boldsymbol{z}_j). \tag{8}$$

An ensembled prediction with the type II bridge network is then constructed as

$$\frac{1}{3}\left(\boldsymbol{v}_i + \boldsymbol{v}_j + H_{i,j}^{(r)}(\boldsymbol{z}_i, \boldsymbol{z}_j)\right), \tag{9}$$

where we construct an ensemble of three models with effectively two forward passes (for $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$). Similar to the type I bridge networks, we may construct multiple bridges between a single curves and use them together for an ensemble

$$\frac{1}{k + \binom{k}{2}}\left(\sum_{i=1}^{k} \boldsymbol{v}_i + \sum_{i<j\leq k} H_{i,j}^{(r)}(\boldsymbol{z}_i, \boldsymbol{z}_j)\right). \tag{10}$$

Correspondence

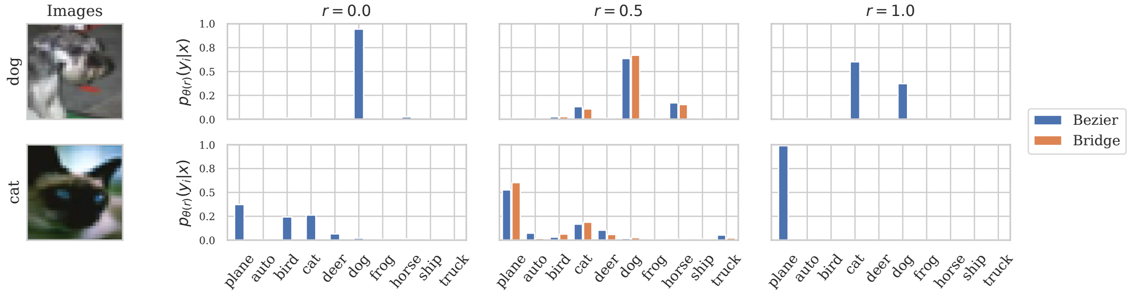

Figure 3: Bar plots in the third column show the class probability outputs of the bridge network (**orange**) and the base model with the Bezier parameters (**blue**) for given images displayed in the first column. We also depict the predicted outputs from the base model with θ_1 and θ_2 in the second and fourth columns, respectively.

Classification Performance (Type I)

Table 1: Performance improvement of the ensemble by adding type I bridges to the single base ResNet model on Tiny ImageNet and ImageNet datasets. $\text{Bridge}_{\text{sm}}$ and $\text{Bridge}_{\text{md}}$ denote the small and the medium versions of the bridge network based on their FLOPs.

Tiny ImageNet

Model	FLOPs (↓)	#Params (↓)	ACC (↑)	NLL (↓)	ECE (↓)	DEE (↑)
ResNet (DE-1)	× 1.000	× 1.000	62.66 ± 0.23	1.683 ± 0.009	0.050 ± 0.004	1.000
+ 1 $\text{Bridge}_{\text{sm}}$	× 1.050	× 1.057	64.58 ± 0.17	1.478 ± 0.006	0.025 ± 0.002	2.280 ± 0.086
+ 2 $\text{Bridge}_{\text{sm}}$	× 1.099	× 1.114	65.37 ± 0.13	1.421 ± 0.004	0.018 ± 0.002	3.087 ± 0.118
+ 3 $\text{Bridge}_{\text{sm}}$	× 1.149	× 1.171	**65.82** ± 0.10	**1.395** ± 0.003	**0.015** ± 0.001	**3.680** ± 0.133
+ 1 $\text{Bridge}_{\text{md}}$	× 1.180	× 1.206	65.13 ± 0.12	1.446 ± 0.002	0.034 ± 0.002	2.709 ± 0.049
+ 2 $\text{Bridge}_{\text{md}}$	× 1.359	× 1.412	66.29 ± 0.06	1.388 ± 0.004	0.025 ± 0.001	3.845 ± 0.171
+ 3 $\text{Bridge}_{\text{md}}$	× 1.539	× 1.618	**66.76** ± 0.09	**1.362** ± 0.003	**0.023** ± 0.001	**4.708** ± 0.209
DE-2	× 2.000	× 2.000	65.54 ± 0.25	1.499 ± 0.007	0.029 ± 0.003	2.000

ImageNet

Model	FLOPs (↓)	#Params (↓)	ACC (↑)	NLL (↓)	ECE (↓)	DEE (↑)
ResNet (DE-1)	× 1.000	× 1.000	75.85 ± 0.06	0.936 ± 0.003	0.019 ± 0.001	1.000
+ 1 $\text{Bridge}_{\text{sm}}$	× 1.061	× 1.071	76.46 ± 0.06	0.914 ± 0.000	0.012 ± 0.001	1.418 ± 0.034
+ 2 $\text{Bridge}_{\text{sm}}$	× 1.123	× 1.141	76.60 ± 0.06	0.907 ± 0.000	0.012 ± 0.001	1.537 ± 0.026
+ 3 $\text{Bridge}_{\text{sm}}$	× 1.184	× 1.212	**76.69** ± 0.04	**0.905** ± 0.000	**0.011** ± 0.001	**1.584** ± 0.021
+ 1 $\text{Bridge}_{\text{md}}$	× 1.194	× 1.222	77.03 ± 0.07	0.889 ± 0.001	**0.013** ± 0.000	1.881 ± 0.022
+ 2 $\text{Bridge}_{\text{md}}$	× 1.389	× 1.444	77.37 ± 0.07	0.876 ± 0.001	**0.013** ± 0.001	2.341 ± 0.076
+ 3 $\text{Bridge}_{\text{md}}$	× 1.583	× 1.665	**77.48** ± 0.03	**0.870** ± 0.000	**0.013** ± 0.000	**2.618** ± 0.062
DE-2	× 2.000	× 2.000	77.12 ± 0.04	0.883 ± 0.001	0.012 ± 0.001	2.000

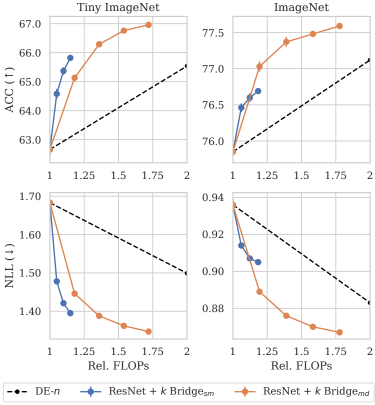

Figure 4: The cost-performance plots of type I bridges compared to DE on Tiny ImageNet and ImageNet datasets. On the basis of DE (black dashed line), the upper left is preferable in ACC, and the lower left is preferable in NLL.

Classification Performance (Type II)

Table 2: Performance improvement of the ensemble by adding type II bridges as members to existing DE ensembles on Tiny ImageNet and ImageNet datasets. $\text{Bridge}_{\text{sm}}$ and $\text{Bridge}_{\text{md}}$ denote the small and the medium versions of the bridge network based on their FLOPs.

Tiny ImageNet

Model	FLOPs (↓)	#Params (↓)	ACC (↑)	NLL (↓)	ECE (↓)	DEE (↑)
DE-4	× 4.000	× 4.000	67.50 ±0.11	1.381 ±0.004	0.018 ±0.001	4.000
+ 1 $\text{Bridge}_{\text{sm}}$	× 4.058	× 4.067	67.86 ±0.05	1.334 ±0.003	0.017 ±0.002	6.051 ±0.181
+ 2 $\text{Bridge}_{\text{sm}}$	× 4.117	× 4.135	68.12 ±0.09	1.311 ±0.005	0.015 ±0.001	8.174 ±0.465
+ 4 $\text{Bridge}_{\text{sm}}$	× 4.234	× 4.269	68.47 ±0.14	1.288 ±0.004	0.015 ±0.001	10.340 ±0.773
+ 6 $\text{Bridge}_{\text{sm}}$	× 4.351	× 4.404	**68.51** ±0.10	**1.278** ±0.003	**0.014** ±0.001	**11.268** ±0.871
+ 1 $\text{Bridge}_{\text{md}}$	× 4.198	× 4.226	68.00 ±0.11	1.333 ±0.003	**0.019** ±0.001	6.183 ±0.120
+ 2 $\text{Bridge}_{\text{md}}$	× 4.395	× 4.453	68.33 ±0.08	1.308 ±0.003	**0.019** ±0.001	8.489 ±0.481
+ 4 $\text{Bridge}_{\text{md}}$	× 4.791	× 4.906	68.61 ±0.05	1.281 ±0.004	0.021 ±0.003	10.897 ±0.800
+ 6 $\text{Bridge}_{\text{md}}$	× 5.186	× 5.359	**68.80** ±0.09	**1.269** ±0.003	0.021 ±0.001	**12.110** ±1.083
DE-5	× 5.000	× 5.000	67.90 ±0.14	1.354 ±0.003	0.019 ±0.001	5.000

ImageNet

Model	FLOPs (↓)	#Params (↓)	ACC (↑)	NLL (↓)	ECE (↓)	DEE (↑)
DE-4	× 4.000	× 4.000	77.87 ±0.04	0.851 ±0.001	0.012 ±0.001	4.000
+ 1 $\text{Bridge}_{\text{sm}}$	× 4.086	× 4.088	77.93 ±0.02	0.847 ±0.000	0.012 ±0.001	4.580 ±0.052
+ 2 $\text{Bridge}_{\text{sm}}$	× 4.172	× 4.176	78.00 ±0.04	**0.846** ±0.000	**0.011** ±0.000	4.739 ±0.052
+ 4 $\text{Bridge}_{\text{sm}}$	× 4.343	× 4.351	78.10 ±0.03	**0.846** ±0.000	**0.011** ±0.001	**4.768** ±0.041
+ 6 $\text{Bridge}_{\text{sm}}$	× 4.515	× 4.527	**78.12** ±0.05	**0.846** ±0.001	**0.011** ±0.001	4.659 ±0.037
+ 1 $\text{Bridge}_{\text{md}}$	× 4.243	× 4.256	78.14 ±0.03	0.839 ±0.000	**0.011** ±0.001	6.123 ±0.121
+ 2 $\text{Bridge}_{\text{md}}$	× 4.487	× 4.512	78.30 ±0.05	0.833 ±0.000	0.012 ±0.001	8.068 ±0.144
+ 4 $\text{Bridge}_{\text{md}}$	× 4.973	× 5.024	78.46 ±0.04	0.828 ±0.000	0.012 ±0.000	9.951 ±0.163
+ 6 $\text{Bridge}_{\text{md}}$	× 5.460	× 5.536	**78.56** ±0.09	**0.825** ±0.000	0.012 ±0.001	**10.760** ±0.202
DE-5	× 5.000	× 5.000	78.03 ±0.03	0.844 ±0.001	0.012 ±0.001	5.000

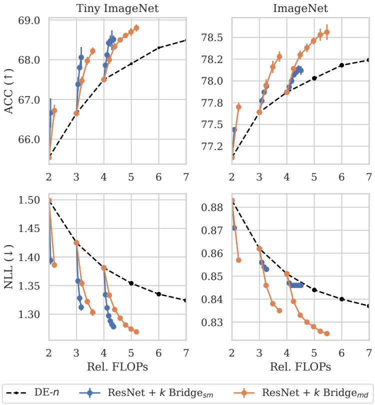

Figure 5: The cost-performance plots of type II bridges compared to DE on Tiny ImageNet and ImageNet datasets. On the basis of DE (black dashed line), the upper left is preferable in ACC, and the lower left is preferable in NLL.

Conclusion

We proposed a novel framework for efficient ensembling that reduces inference costs of ensembles with a lightweight network called **bridge networks**.

Through empirical validation, we show that

1. **Bridge networks can approximate outputs** of connecting subspaces quite accurately with minimal computation cost.
2. DEs augmented with bridge networks can significantly **reduce inference costs** without big sacrifice in performance.

References

F. Draxler, K. Veschgini, M. Salmhofer, and F. Hamprecht. Essentially no barriers in neural network energy landscape. In *Proceedings of The 35th International Conference on Machine Learning (ICML)*, 2018.

T. Garipov, P. Izmailov, D. Podoprikhin, D. Vetrov, and A. G. Wilson. Loss surfaces, mode connectivity, and fast ensembling of DNNs. In *Advances in Neural Information Processing Systems 31 (NeurIPS)*, 2018.

B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems 30 (NIPS)*, 2017.