# Class-wise Metric Scaling for Improved Few-Shot Classification

## Appendix

## A. Network architectures

**Conv4** contains four blocks, where each block is composed of 64 channels with $3*3$ convolutional filters followed by batch normalization, ReLU, and $2*2$ max-pooling. The input image size is $3*84*84$, and output feature maps have a size of $64*5*5$.

**ResNet12** contains four residual blocks, with each block having three $3*3$ convolutional layers followed by 2*2 max-pooling. Each convolutional layer has a 3*3 kernel, followed by batch normalization and leaky ReLU of 0.1. The first block includes 64 feature channels, which are doubled at each subsequent block, and final output feature maps have 512 channels. For the $3*84*84$ input size, the output feature maps have a size of $512*5*5$.

**Low-rank bilinear pooling (LBP) Layer** utilizes two $1*1$ convolutional layers for feature dimension reduction. We set feature dimension $d$ as 1024 and 8192 for the Conv4 and ResNet12, respectively.

## B. Implementation details

In the training stage, we use the SGD optimizer with the Nesterov momentum 0.9, where a weight decay of $1e^{-4}$ is applied to the model parameters except for the metric scalars. The learning rate is initialized with 0.1. We train 100 epochs on the tieredImagenet with learning rate dropped by 0.1 every 25 epochs, while 400 epochs for *mini*-ImageNet and cross-domain benchmarks with learning rate dropped by 0.1 every 100 epochs. Learnable metric scalars $s_b$ are initialized by 20 and 5 for cosine similarity and Euclidean distance, respectively. Conventional data augmentation, including random crop, horizontal flip, and color jitter, are applied to training images. As for the validation set, 300 1-shot and 5-shot tasks are sampled for hyper-parameter and model selections with the best 1-shot and 5-shot validation performances, respectively. The validation interval is set to 1 epoch for tieredImgaeNet and 5 epochs for other datasets.

In the testing stage, we evaluate 5-way 1-shot and 5-shot generalization performance on the novel set. We sample 5000 tasks with 15 queries per class for all experiments and report the average accuracy with a corresponding $95\%$ confidence interval. For both support and query sets, ten crops and horizontal flip are used. In both types of the semi-supervised inference mode, the proportion $\tau$ of cherry-picking is set to 0.5. For solving the convex optimization problem, we adopt the BFGS algorithm with the backtracking line search, which indicates high-efficiency for the low-dimensional optimization problem and just needs a few iteration steps to obtain the optimal solution.