
Meta-Learner with Linear Nulling

Sung Whan Yoon
shyoon8@kaist.ac.kr

Jun Seo
tjwns0630@kaist.ac.kr

Jaekyun Moon
jmoon@kaist.edu

School of Electrical Engineering,
Korea Advanced Institute of Science and Technology (KAIST)

Abstract

We propose a meta-learning algorithm utilizing a linear transformer that carries out null-space projection of neural network outputs. The main idea is to construct an alternative classification space such that the error signals during few-shot learning are quickly zero-forced on that space so that reliable classification on low data is possible. The final decision on a query is obtained utilizing a null-space-projected distance measure between the network output and reference vectors, both of which have been trained in the initial learning phase. Among the known methods with a given model size, our meta-learner achieves the best or near-best image classification accuracies with Omniglot and *miniImageNet* datasets.

1 Introduction

Achieving human-like adaptability remains a daunting challenge for developers of artificial intelligence. In a related effort, researchers in the machine learning literature have tried to understand ways to learn quickly on a small number of training examples. Learning the ability to quickly learn a new task on low data is known as meta-learning, and typically two-level learning is employed: initial learning on large data sets representing widely varying tasks and few-shot learning on small amounts of unseen data to conduct a yet different task. Note that while the original meaning of meta-learning is to describe ability to conduct completely different sets of tasks without much new training in between [11, 12, 13, 15], many researchers have also used the term meta-learning as they developed classification methods involving disjoint classes of images between initial training and inference.

Significant advances have been made recently in this narrow sense of meta-learning, which is also often treated in the same context as few-shot learning. In particular, the prior work on combining a neural network with external memory, known as the memory-augmented neural network (MANN) [10], showed a notable progress. The Matching Network of [16] yields decisions based on matching the output of a network driven by a query sample to the output of another network fed by labeled samples; a training method was also introduced there which utilizes only a few examples per class at a time to match train and test conditions for a rapid learning. In yet another approach, the long short-term memory (LSTM) of [2] is trained to optimize another learner, which performs actual classification [8]. There, the parameters of the learner in few-shot learning are first set to the memory state of the LSTM, and then quickly learned based on the memory update rule of the LSTM, effectively capturing the knowledge from small samples. The model-agnostic meta-learner (MAML) of [1] sets up the model for easy fine-tuning so that a small number of gradient-based updates allows the model to yield a good generalization. A method dubbed the simple neural attentive meta-learner (SNAIL) combines an embedding neural network with temporal convolution and soft attention to draw from past experience [6]. The Prototypical Network of [14] makes decision by comparing the query output with the per-class cluster means in the embedding space while the network is learned via many rounds of episodic training.

In this paper, we propose a unique meta-learning algorithm utilizing linear transformation which allows classification in an alternative projection space to achieve improved few-shot learning. During the initial meta-learning phase, a special set of vectors that acts as references for classification is learned together with the embedding network. The linear transformer, which can be viewed as a simple null-space projector, zero-forces errors between the per-class average outputs of the embedding network and the reference vectors in the projection space. The construction of this projection space is episode-specific. The essence of our algorithm is that when we try to match the network outputs representing the images to classify with the references utilized for classification, they do not need to be close in the original embedding space as long as they are conditioned to match well in the projection space.

Our meta-learner exhibits competitive performances among existing meta-learners for Omniglot and *miniImageNet* image classification. In particular, in 20-way Omniglot experiments, our method gives near-best performance, second only to the SNAIL of [6] in both 1-shot and 5-shot results. For 1-shot experiments of *miniImageNet*, our meta-learner is again the second to SNAIL, which yields the best result among all existing methods to our knowledge, except for the task-dependent adaptive metric (TADAM) of [7] that requires an extra network. For the 5-shot testing of *miniImageNet*, however, our method yields the best accuracy for a given model size, beating both the Prototypical Network of [14] and the SNAIL method of [6], albeit by a small margin against the latter.

2 Meta-Learner with Linear Nulling

2.1 Model Description

Let us provide a quick conceptual understanding of the proposed algorithm. In processing a given episode of labeled images (support set \mathcal{S}) and queries during the initial meta-learning phase, the convolutional neural network (CNN) output average $\bar{\mathbf{g}}_k$ is first generated for each label. A support set \mathcal{S} contains pairs of images \mathbf{x}_n and matching labels y_n . Let S_k be the subset of \mathcal{S} with a particular label k . See Fig. 1. The model also makes use of a special set of vectors ϕ_k that are carried over from the last episode-processing stage (arbitrarily initialized when handling the first episode). The purpose of ϕ_k 's will be made clear shortly. Let Φ be the matrix collecting ϕ_k 's for all labels. A projection space M is then constructed such that the average CNN output vectors $\bar{\mathbf{g}}_k$ and the vectors in Φ are aligned closely in this space. The details of constructing M will be given below.

Now, each query input is applied to the embedding CNN and the resulting output is collected. When this is done for all queries (with different labels), both the CNN f_θ and the vectors in Φ are adjusted by comparing the query output with the reference vectors such that in-class similarities as well as out-of-class differences are maximized collectively in the projection space.

This process gets repeated for every remaining episode with new classes of images and queries during the initial learning phase. The vectors ϕ_k act as references for classification, and they are learned from one episode to next together with the embedding network. Note that the projection space M itself is not learned but computed anew in every episode, given newly obtained averages $\bar{\mathbf{g}}_k$ and the reference vectors Φ carried over from the last episode stage. The labels for the vectors in Φ do not need to change from one episode to another. M can be seen as playing a crucial role in providing an efficient form of episode-specific conditioning. By the time the initial meta-learning phase is over, an effective few-shot learning strategy gets built-in jointly in the embedding network and the reference vector set.

Now, as the few-shot learning phase begins, the learned parameters f_θ and Φ will have all been fixed. First, a new set of per-class average vectors $\bar{\mathbf{g}}_k$ corresponding to the given few-shot images are obtained. Then, a new projection space is formed using these average vectors as well as the reference vectors in Φ . Next, the network output for the test shot is compared with each of the reference vectors in the projection space for the final classification decision. The Euclidean distance is used here as distance measure and softmax is utilized as the activation function.

2.2 Construction of Null-Space

The projection space M is obtained as follows. Consider the training samples labeled k in a given episode. The average CNN outputs $\bar{\mathbf{g}}_k$ are obtained for all classes. As mentioned, we are to choose M such that $\bar{\mathbf{g}}_k$ and the matching reference weight vector ϕ_k are aligned closely when projected on

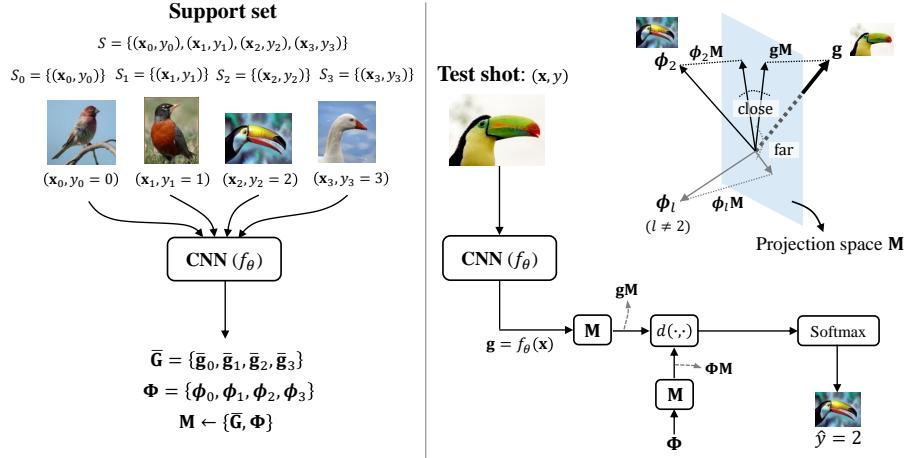


Figure 1: The proposed meta-learner with linear nulling

M itself. At the same time, we also want to make \bar{g}_k and the non-matching weights ϕ_l for all $l \neq k$ well-separated in the same projection space. An effective way to achieve this is to maximize

$$\Delta_k = \left\{ (N_c - 1)\phi_k - \sum_{l \neq k} \phi_l \right\} M M^T \bar{g}_k^T \quad (1)$$

which is to say that two matching vectors ϕ_k and \bar{g}_k yield a large inner product (i.e., well-aligned) on M while at the same time the non-matching vector sum $\sum_{l \neq k} \phi_l$ and \bar{g}_k have a small inner product. N_c is the number of labels. The class references ϕ_k 's are not relabeled from one episode to next.

We assume that M takes the form of a matrix whose columns are orthonormal unit vectors. Defining the difference or error vector $v_k = \{(N_c - 1)\phi_k - \sum_{l \neq k} \phi_l\} - \bar{g}_k$, maximization of (1) is done by taking M to be a subspace of the orthogonal complement of the space spanned by v_k in \mathbb{R}^D , where D is the length of v_k . In this way, the error vector is always forced to zero when projected on M . To consider all classes, M should be a subspace of the null-space of a matrix whose columns are N_c error vectors, i.e., $M \subseteq \text{null}(\{v_0, \dots, v_{N_c-1}\})$.

By focusing only on the null-space, we are essentially cutting out unnecessary subspaces for classification that contain non-orthogonal basis to any error vectors. With such an M , all error vectors are forced to zero when projected on itself. Note that a simple choice of $M = \text{null}(\{v_0, \dots, v_{N_c-1}\})$ is a perfectly good solution, which we employ in this paper. If D is larger than N_c , the projection space M always exists. Note that because the matching between the network output averages and the reference vectors occurs on the episode-specific projection space, ϕ_k 's do not need to be relabeled in every episode; the same label sticks to each reference vector throughout the initial learning phase.

3 Experiment Results

3.1 Settings for Omniglot and miniImageNet Few-shot Classification

Omniglot [5] is a set of images of 1623 handwritten characters from 50 alphabets with 20 examples for each class. We have used 28×28 downsized grayscale images and introduced class-level data augmentation by random angle rotation of images in multiples of 90° degrees, as done in prior works [10, 14, 16]. 1200 characters are used for training and test is done with the remaining characters. miniImageNet is a dataset suggested by Vinyals et al. for few-shot classification of colored images [16]. It is a subset of the ILSVRC-12 ImageNet dataset [9] with 100 classes and 600 images per each class. We have used the splits introduced by Ravi and Larochelle [8]. For experiment, we have used 84×84 downsized color images with a split of 64 training classes, 16 validation classes and 20 test classes. Data augmentation was not employed in the miniImageNet experiment. The Adam optimizer [4] with optimized learning-rate decay is used.

For fair comparison, we employ the same embedding CNN widely used in prior works. It is based on four convolutional blocks, each of which consists of a 3×3 2D convolution layer with 64 filters,

stride 1 and padding, a batch normalization layer [3], a ReLU activation and a 2×2 maxpooling, as done in [14, 16].

For 20-way Omniglot and 5-way *miniImageNet* classification, our meta-learner is trained with 60-way and 20-way episodes, respectively. In the test phase of 20-way Omniglot and 5-way *miniImageNet* classification, we have to choose 20 and 5 reference vectors among 60 and 20, respectively. In selecting only a subset of reference vectors for testing purposes, relabeling is done. For each average network output chosen in arbitrary order, the closest vector among the remaining ones in Φ is tagged with the matching label. The closeness measure is the Euclidean distance in our experiments. After choosing the closest reference vectors, the projection space M is obtained for few-shot classification. The experimental results of our meta-learner in Table 1 are based on 60-way initial learning for 20-way Omniglot and 20-way initial learning for 5-way *miniImageNet* classification.

3.2 Results

In Table 1, few-shot classification accuracies of our meta-learner with nulling (MLN) are presented. The performance in the 20-way Omniglot experiment is evaluated by the average accuracy over randomly chosen 1×10^4 test episodes with 5 query images for each class. On the other hand, the performance in 5-way *miniImageNet* is evaluated by the average accuracy and a 95% confidence interval over randomly chosen 3×10^4 test episodes with 15 query images for each class.

When we obtain the projection space, we normalize the all $\{(N_c - 1)\phi_k - \sum_{l \neq k} \phi_l\}$ and \bar{g}_k vectors by forcing the power to 1. Once the projection space is found, classification of the query is done by measuring the distance to the reference vectors without normalizing the vectors, since the query output is not normalized.

Prior meta-learners including the matching networks [16], MAML [1], the Prototypical Networks [14] and SNAIL [6] are compared. For 20-way Omniglot classification, MLN shows the second best performance for both 1- and 5-shot cases. Although the performance is not the best, the classification accuracies are fairly close to the best. For the 5-shot case of 5-way *miniImageNet* classification, our meta-learner achieves the best performance. We remark that for *miniImageNet*, the recently introduced TADAM of [7] actually shows the best performance among all known methods including our MLN. However, we opted not to include this method in the comparison table as it requires an extra network for the task-conditioning and thus comparison would not be fair.

Our meta-learner is perhaps closest in spirit to the Prototypical Network, among known prior methods, in that both methods rely on training references via repetitive episode-specific conditioning with the final query image compared against each of the references during classification. The key difference is that in the Prototypical Network, the generalization strategy is directly learned in the embedding network, whereas in our method a separate set of reference vectors is maintained which is learned together with the network. Employing a projection space is also a unique feature of MLN.

Table 1: Few-shot classification accuracies for 20-way Omniglot and 5-way *miniImageNet*

Methods	20-way Omniglot		5-way <i>miniImageNet</i>	
	1-shot	5-shot	1-shot	5-shot
Matching Nets [16]	88.2%	97.0%	$43.56 \pm 0.84\%$	$55.31 \pm 0.73\%$
MAML [1]	95.8%	98.9%	$48.70 \pm 1.84\%$	$63.15 \pm 0.91\%$
Prototypical Nets [14]	96.0%	98.9%	$49.42 \pm 0.78\%$	$68.20 \pm 0.66\%$
SNAIL [6]	97.64%	99.36%	$55.71 \pm 0.99\%$	$68.88 \pm 0.92\%$
MLN	97.33%	99.18%	$50.68 \pm 0.11\%$	$69.00 \pm 0.09\%$

4 Conclusion

In this work, we proposed a meta-learning algorithm aided by a linear transformer that performs null-space projection of the network output. Our algorithm uses linear nulling to shape the classification space where network outputs could be better classified. Our meta-learner achieves the best or near-best performance among known methods in various few-shot image classification tasks for a given network size.

Acknowledgments

This work is supported by the ICT R&D program of Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) [2016-0-005630031001, Research on Adaptive Machine Learning Technology Development for Intelligent Autonomous Digital Companion].

References

- [1] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pages 1126–1135, 2017.
- [2] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
- [3] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.
- [4] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [5] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.
- [6] Nikhil Mishra, Mostafa Rohaninejad, Xi Chen, and Pieter Abbeel. A simple neural attentive meta-learner. In *NIPS 2017 Workshop on Meta-Learning*, 2017.
- [7] Boris N Oreshkin, Alexandre Lacoste, and Pau Rodriguez. Tadam: Task dependent adaptive metric for improved few-shot learning. In *Advances in Neural Information Processing Systems*, 2018.
- [8] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *International Conference on Learning Representations*, 2017.
- [9] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.
- [10] Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy Lillicrap. Meta-learning with memory-augmented neural networks. In *International conference on machine learning*, pages 1842–1850, 2016.
- [11] Jürgen Schmidhuber. *Evolutionary principles in self-referential learning, or on learning how to learn: The meta-meta.... hook*. Diplomarbeit, Technische Universität München, München, 1987.
- [12] Jürgen Schmidhuber. A neural network that embeds its own meta-levels. In *Neural Networks, 1993., IEEE International Conference on*, pages 407–412. IEEE, 1993.
- [13] Jürgen Schmidhuber, Jieyu Zhao, and Marco Wiering. Shifting inductive bias with success-story algorithm, adaptive levin search, and incremental self-improvement. *Machine Learning*, 28(1):105–130, 1997.
- [14] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, pages 4080–4090, 2017.
- [15] Sebastian Thrun and Lorien Pratt. *Learning to learn*. Springer Science & Business Media, 2012.
- [16] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, pages 3630–3638, 2016.