EXPLORING THE SIMILARITY OF REPRESENTATIONS IN MODEL-AGNOSTIC META-LEARNING

Thomas Goerttler[1] & Klaus Obermayer[1,2]
[1]Technische Universität Berlin, Chair of Neural Information Processing, Germany
[2]Bernstein Center for Computational Neuroscience Berlin, Germany
Correspondence: thomas.goerttler@ni.tu-berlin.de

ABSTRACT

In past years model-agnostic meta-learning (MAML) has been one of the most promising approaches in meta-learning. It can be applied to different kinds of problems, e.g., reinforcement learning, but also shows good results on few-shot learning tasks. Besides their tremendous success in these tasks, it has still not been fully revealed yet, why it works so well. Recent work proposes that MAML rather reuses features than rapidly learns. In this paper, we want to inspire a deeper understanding of this question by analyzing MAML's representation. We apply representation similarity analysis (RSA), a well-established method in neuroscience, to the few-shot learning instantiation of MAML. Although some part of our analysis supports their general results that feature reuse is predominant, we also reveal arguments against their conclusion. The similarity-increase of layers closer to the input layers arises from the learning task itself and not from the model. In addition, the representations after inner gradient steps make a broader change to the representation than the changes during meta-training.

1 INTRODUCTION

In recent years, meta-learning, also learning to learn (Thrun & Pratt, 1998), has aroused interest in different fields of machine learning research. Meta-learning tries to use past experiences of similar problems to acquire a prior over model parameters or the learning procedure. Several researchers have used meta-learning to tackle the problem of few-shot learning (Santoro et al., 2016; Ravi & Larochelle, 2017; Finn et al., 2017) where novel tasks are trained only by seeing a few examples of each possible class. The models have to adapt quickly to the new problem task. Traditional supervised models fail because they cannot generalize on such a few examples and either do not learn the task or quickly overfit the training data.

Next to recurrent recurrent (Hochreiter et al., 2001) and attention-based (Vinyals et al., 2016) models, there exist also alternatives with a bi-level meta-learning setup e.g. MAML (Finn et al., 2017). Despite its success, it is still not fully revealed why and also how MAML learns so well. In Raghu et al. (2020) the question is proposed whether MAML successfully solves few-shot learning tasks mainly because of feature reuse or rapid learning. Feature reuse means that the meta-initialization is already so good that it does not have to be changed anymore significantly for the specific tasks. In contrast, rapid learning means that efficient significant changes are still done. By freezing a lot of layers and comparing the different layers by applying canonical correlation analysis (CCA) (Morcos et al., 2018), and centered kernel alignment (CKA) (Kornblith et al., 2019), they come to the answer that MAML rather reuses features.

We analyze the representation of few-shot learning models during training with representation similarity analysis (RSA) (Kriegeskorte et al., 2008) and reveal further insights into their representations. Similar to Raghu et al. (2020), we observe that the similarity of trained activations to activations before training is very high in early layers. However, we think that this is caused rather by the task itself than by the MAML learning procedure, as the same effect can be observed in CNNs for standard supervised learning. Moreover, our analysis also reveals that inner gradient steps lead to a more significant change in the representation than the updates in meta-training. Besides, we observe that most of the inner step optimization happens in the first gradient step.

2 RELATED WORK

Model-agnostic meta-learning (MAML) addresses the general problem of meta-learning (Thrun & Pratt, 1998), which includes few-shot learning. Next to recurrent-based (Hochreiter et al., 2001; Santoro et al., 2016; Ravi & Larochelle, 2017), and attention-based ((Vinyals et al., 2016; Mishra et al., 2018) meta-learning models, there also exist model-agnostic approaches (Maclaurin et al., 2015; Finn et al., 2017). Model-agnostic meta-learning (Finn et al., 2017) is a model-agnostic approach, which means it does not require a specific model. Since MAML requires second-order derivatives, there are several approaches that approximate the second-order derivatives, e.g., FOMAML (Finn et al., 2017), Reptile (Nichol et al., 2018), and iMAML (Rajeswaran et al., 2019) to overcome the need of calculating them.

Representation similarity analysis (RSA) is widely used in computational neuroscience to compare a computational or behavioral model with the brain response (Kriegeskorte et al., 2008). In Khaligh-Razavi & Kriegeskorte (2014), several (un)supervised vision models are used to show that supervised models are better for explaining IT cortical than unsupervised ones. In Cichy et al. (2017) the correlation of dynamics of the visual system is correlated with deep networks. RSA is also used in pure computational models (McClure & Kriegeskorte, 2016; Dwivedi & Roig, 2019; Mehrer et al., 2020): McClure & Kriegeskorte (2016) use RSA as a loss function for knowledge distillation, Dwivedi & Roig (2019) use it as a measure for the similarity between vision tasks, and Mehrer et al. (2020) analyze the consistency of neural networks with RSA. Dwivedi et al. (2020) propose a more general approach of comparing different activations, that RSA and CKA (Kornblith et al., 2019) - which is used in Raghu et al. (2020) to analyze MAML - are instantiations of.

3 ANALYSIS OF REPRESENTATION IN FEW-SHOT LEARNING

The idea of MAML (Finn et al., 2017) is to learn an initialization of a model using exact second-order methods across tasks that are sampled from the same distribution. The optimization is done in a bi-level meta-learning setup. The meta-optimization is done in the outer loop and can be described as

$$\theta^* := \underset{\theta \in \Theta}{\operatorname{argmin}} \frac{1}{M} \sum_{i=1}^{M} \mathcal{L}(in(\theta, \mathcal{D}_i^{tr}), \mathcal{D}_i^{test}) \tag{1}$$

where M is the number of task in a batch, $\mathcal{D}_i^{tr}$ and $\mathcal{D}_i^{test}$ are the training and test set of task i, $\mathcal{L}$ is the loss function and $in(\theta, \mathcal{D}_i^{tr})$ describes the inner loop. For every task i in a batch, the neural network is initialized with θ and optimized for one or a few training steps of gradient descent to obtain ϕ_i.

$$\phi_i \equiv in(\theta, \mathcal{D}_i^{tr}) = \theta - \alpha \nabla_\theta \mathcal{L}(\theta, \mathcal{D}_i^{tr}) \tag{2}$$

is an example of only using one gradient step.

In this paper, we apply RSA to the representations which are obtained by training MAML on few-shot learning tasks to reveal dissimilarities. These dissimilarities are obtained by comparing pairwise dissimilarities (e.g., by Euclidean distances) between representations associated with each pair of inputs. This results in Representation Dissimilarity matrices (RDMs). In a second step, the Spearman's correlation between different RDMs of different representations is calculated. Subtracting these from 1 leads to the dissimilarity scores of representations. In MAML, after every training step there exist a new initial representations parametrized by θ as well as new fine-tuned representations for every task i parametrized by ϕ_i of the meta-test task which are obtained after inner loop updates on the training (support) $\mathcal{D}_i^{tr}$ set. We analyze both the initial representation and the fine-tuned ones for several meta-test tasks. Further details on the application of RSA are described in Appendix B.

In our experiments, we use the same architecture which was proposed in Finn et al. (2017). In the analysis, we focus on 20-way 1-shot learning on the omniglot dataset, but results on other tasks (e.g., Mini-ImageNet) showed the same behaviour (see Appendix D). Details on the model are described in Appendix A. How to access the code can be found in Appendix E.

Additionally, we also apply the same network architecutre on MNIST for standard supervised classication to observe the similarity there.

(a) (b)

Figure 1: On (a), we see the mean dissimilarity (RSA with Euclidean distance) with error bars of the representations before finetuning compared to the random initilization at the beginning of meta-training of 50 test task. On (b), we see the dissimilarity of the representations before finetuning to the one 1000 training steps before (log-scaled).

(a) (b)

Figure 2: In this figure, the (dis)similarity of the MNIST experiments is seen. On (a), we use RSA and in (b) CKA as a (dis)similarity measure. Note that CKA measures the similarity, whereas RSA measures dissimilarity.

4 RESULTS

Comparision of similarity of early and late layers: In Figure 1a, we analyze the dissimilarity of the representations after 5 fine-tuning gradient steps on the test set of 50 meta-test tasks. After 60000 meta-training steps, we observe the same circumstance like Raghu et al. (2020): The similarity of the representations of early layers to its initialization is really high, whereas especially in the last layer, the activations change a lot. This makes sense because the last layer has to adapt to the labels of the novel task. Raghu et al. (2020) conclude from this observation that almost no learning happens in the inner loop. However, we also observe that this circumstance does not only hold after 60000 but also in earlier stages of the training process and even for the first ones. In fact, this even holds for standard classification with now pre-training of the initial θ. In a second experiment, we use the same network which is used for MAML on Omniglot for classification on MNIST (see Figure 2). Although the weights of early layers are completely random, the activations of early layers hardly change. To align with Raghu et al. (2020), we also calculated the similarity with CKA, which shows the same but even with higher similarity in early layers.

From these experiments, we conclude that even if no feature reusing is possible as weights are random, during fine-tuning, the activation of early layers stays more similar to their initializations than in later layers. Instead of deriving from this that no learning happens, we think that this is probably more likely an artifact of the model itself (early layers are farther away from the output layer and get smaller errors and gradients), or the network architecture is too dense. Therefore, we think this observation is not property-specific to MAML.

Comparision of meta-training to fine-tuning: Figure 1b shows the dissimilarity using RSA of the representation parametrized by θ of the five layers to the representation obtained by the network exactly 1000 training steps before. After several thousand training steps, we see that the dissimilarity has converged to a value and then only fluctuates around it for the rest of the training time. The dissimilarity of representations from the first pooling layers is close to 0, whereas the last pooling layers' dissimilarity fluctuates around about 0.1. This underlines the observation of before that there is more change in later layers than in early ones.

For further analysis, we sampled four tasks and evaluated the representation on a fifth one. The performance of them increasing in the first 10000 training iterations and after that only slowly in-

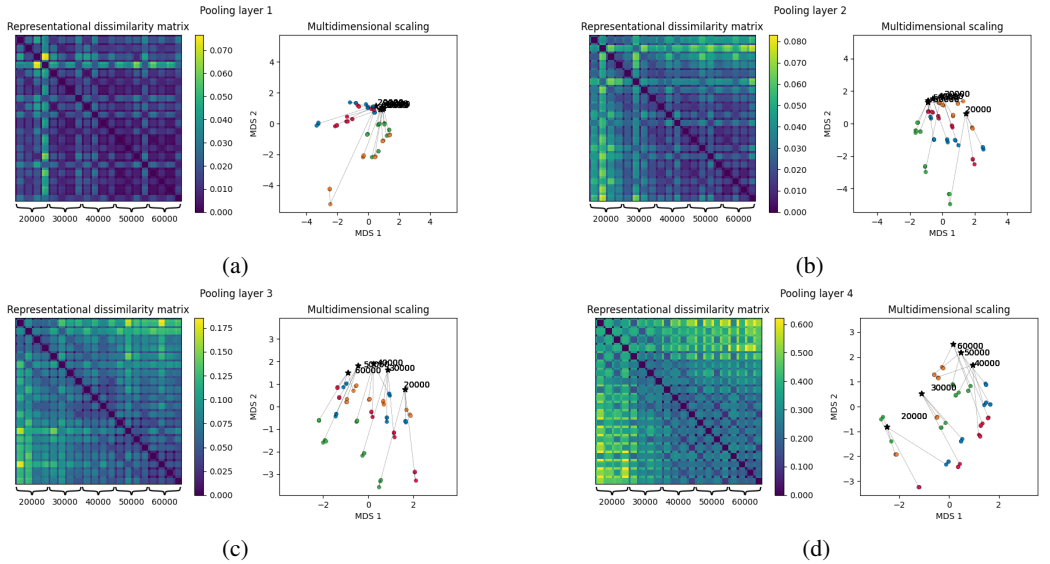

Figure 3: In this figure, the dissimilarity of the representations obtained by applying an evaluation task on the MAML model after 20000, 30000, 40000, 50000, and 60000 training steps are shown: on the left the RDM, on the right side after applying multidimensional scaling. For every step, we obtain the representation after updating for 0, 1, 5, and 10 inner steps connected by a dotted line.

creasing fluctuating around it (Further details in Appendix C) In Figure 3 we see the results of applying RSA on the representation before and after (1, 5, and 10) inner loop gradient steps of the four selected tasks for all layers after 20000, 30000, 40000, 50000, and 60000 training steps. We see that in every layer, all fine-tuned representations, even only after one inner step, are quite far away from their meta initialization by θ. Especially for the early layers, the learned representations in meta-optimization are closer to the representations 10000 training steps before than to their corresponding fine-tuned query tasks. This points to a rapid adaptation to the novel situation also in early pooling layers.

Although the representations after fine-tuning are similar to their starting representation in general, compared to the changes of updating during meta optimization (see 1b), the change is significantly larger. Interestingly the first inner step is responsible for most of the change in the representations. Trained for 5 inner steps, the representations after 5 or even 10 inner steps are not significant compared to the one after 1 step.Therefore, we think that rapidly learning also happens in early layers, even if their representation stays more similar to its initialization than in later layers.

5 Conclusion

In this paper, we applied RSA on both the initial but also the fine-tuned representation of MAML. On the one hand, our analysis confirmed that the representations in early layers do not change a lot compared to in later layers. However, this is also true in supervised classification. Therefore, this observation is not specific to MAML, but we think more likely caused by the architecture of neural networks.

In addition, we also showed that in early layers, the change of the representations in the inner loop of meta-testing is significantly larger than the representation change during training the meta-initialization. This reveals that MAML partially still quickly learns also in early layers.

In general, we think that the question of whether MAML reuses features or rapidly learns also depends on its definition. As there is no clear definition and measure for that, it is hard to agree finally. Nevertheless, our analysis shows the potential of using RSA to analyze the representations of MAML. We think that more questions, e.g., if a test task aims to a similar optimum during training, can be discussed in the future by looking at the representations of MAML models.

REFERENCES

Radoslaw Martin Cichy, Aditya Khosla, Dimitrios Pantazis, and Aude Oliva. Dynamics of scene representations in the human brain revealed by magnetoencephalography and deep neural networks. *NeuroImage*, 153:346–358, 2017.

Kshitij Dwivedi and Gemma Roig. Representation similarity analysis for efficient task taxonomy & transfer learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 12387–12396. Computer Vision Foundation / IEEE, 2019.

Kshitij Dwivedi, Jiahui Huang, Radoslaw Martin Cichy, and Gemma Roig. Duality diagram similarity: A generic framework for initialization selection in task transfer learning. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (eds.), *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XXVI*, volume 12371 of *Lecture Notes in Computer Science*, pp. 497–513. Springer, 2020.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pp. 1126–1135, 2017.

Sepp Hochreiter, A. Steven Younger, and Peter R. Conwell. Learning to learn using gradient descent. In *Artificial Neural Networks - ICANN 2001, International Conference Vienna, Austria, August 21-25, 2001 Proceedings*, pp. 87–94, 2001.

Seyed-Mahdi Khaligh-Razavi and Nikolaus Kriegeskorte. Deep supervised, but not unsupervised, models may explain it cortical representation. *PLOS Computational Biology*, 10(11):1–29, 11 2014. doi: 10.1371/journal.pcbi.1003915.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey E. Hinton. Similarity of neural network representations revisited. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3519–3529. PMLR, 2019.

Nikolaus Kriegeskorte, Marieke Mur, and Peter A. Bandettini. Representational similarity analysis-connecting the branches of systems neuroscience. *Frontiers in Systems Neuroscience*, 2:4, 2008.

Dougal Maclaurin, David Duvenaud, and Ryan P. Adams. Gradient-based hyperparameter optimization through reversible learning. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pp. 2113–2122, 2015.

Patrick McClure and Nikolaus Kriegeskorte. Representational distance learning for deep neural networks. *Frontiers Comput. Neurosci.*, 10:131, 2016.

Johannes Mehrer, Courtney J. Spoerer, Nikolaus Kriegeskorte, and Tim C. Kietzmann. Individual differences among deep neural network models. *Nature Communications*, (1):5725, 2020.

Nikhil Mishra, Mostafa Rohaninejad, Xi Chen, and Pieter Abbeel. A simple neural attentive meta-learner. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018.

Ari S. Morcos, Maithra Raghu, and Samy Bengio. Insights on representational similarity in neural networks with canonical correlation. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 5732–5741, 2018.

Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *CoRR*, abs/1803.02999, 2018.

Aniruddh Raghu, Maithra Raghu, Samy Bengio, and Oriol Vinyals. Rapid learning or feature reuse? towards understanding the effectiveness of MAML. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, 2020.

Aravind Rajeswaran, Chelsea Finn, Sham M. Kakade, and Sergey Levine. Meta-learning with implicit gradients. *CoRR*, abs/1909.04630, 2019.

Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy P. Lillicrap. Meta-learning with memory-augmented neural networks. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, pp. 1842–1850, 2016.

Sebastian Thrun and Lorien Y. Pratt. Learning to learn: Introduction and overview. In Sebastian Thrun and Lorien Y. Pratt (eds.), *Learning to Learn*, pp. 3–17. Springer, 1998.

Oriol Vinyals, Charles Blundell, Tim Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 3630–3638, 2016.

A MODEL DETAILS

In our experiments, we use the same architecture proposed in Finn et al. (2017). In our analysis, we focus on 20-way 1-shot learning on Omniglot in Section 4 and in the Appendix D on 5-way 1-shot on Mini-ImageNet.

For both datasets, we have 4 convolution hidden layers with a 3x3 filter and 64 filters for Omniglot and 32 filters for Mini-ImageNet, batch normalization, and ReLU activations. For Omniglot, the dimension of the strides is 2x2; for Mini-ImageNet, we use 2x2 max pooling instead. The output layer is a dense layer fed in a softmax. The images of Omniglot are downsampled to 28x28.

The network are trained for 60000 training steps with a batch size of 16 (Omniglot) and 4 (Mini-ImageNet), 5 inner loop updates, an inner learning rate of 0.1 (0.01 for Mini-ImageNet), and a meta-learning rate of 0.001 (AdamOptimizer).

For the MNIST experiement we use the same network as for Omniglot. Everything stays the same, only the batch size increases to 100 and we only traine for 200 training steps.

B DETAILS OF MEASURING THE DISSIMILARITY WITH RSA

RSA Representation dissimilarity analysis is organized in two steps. In the first step, we compute a representation dissimilarity matrix (RDM) for every representation we want to compare. This is done by calculating the pair-wise dissimilarity associated with each pair of inputs resulting in a representations' gram matrix. The resulting matrix is a symmetric matrix with the dimension of the number of inputs we consider. The dissimilarities are originally calculated by $1 - corr_{Pearson}(x, y)$ where x and y are representations; however, other (dis)similarity functions can be used. In our paper, we used the Euclidean distance.

In a second step, we again take pair-wise dissimilarities, but this time we take the flattened RDM matrices as the input obtained in the first step. For computational reasons, we only take the lower or upper triangle. This time the dissimilarity is calculated by $1 - corr_{Spearman}(x, y)$ where x and y are RDMs from step 1.

Experimental details Traditionally, the meta-test task is analyzed by the performance of the test (or query) set. As this query set differs across every test task, but RSA needs the same inputs aligned, we hold out the same specific evaluation task to compare different models after different training steps. In our experiment where we show the dissimilarity of four different tasks given the same initilization (Figure 3), we use always the same hold out task to calculate the dissimilarity.

C FURTHER RESULT ON OMNIGLOT

In Figure 4 the performance of the four task of Figure 3 after several training steps is shown. We observe they start fluctuating around a value after some thousand training steps increasing slightly. Also, the average performance across 600 tasks is rapidly increasing during the first 10000 iterations and then only slowly.

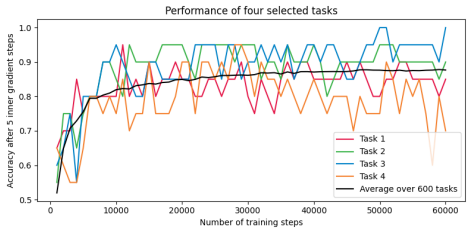

Figure 4: On this figure, we see the performance of MAML of four sampled tasks and optimized for five iterations in the inner loop.

In Figure 5 the representations of the head layer of the experiment on the omniglot dataset is shown. Not surprisingly we see, that every task tends to go the same region meaning that their representation is similar. This makes sense, as the output layer has to have very similar representations to classify similary.

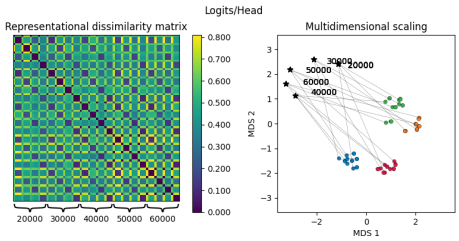

Figure 5: In this figure we show the representations of the output layer of the experiement in Section 4

.

D RESULTS ON MINI-IMAGENET

We also repeated the experiment on a more difficult dataset Mini-ImageNet (see Figure 6), and observed the same as for Omniglot.

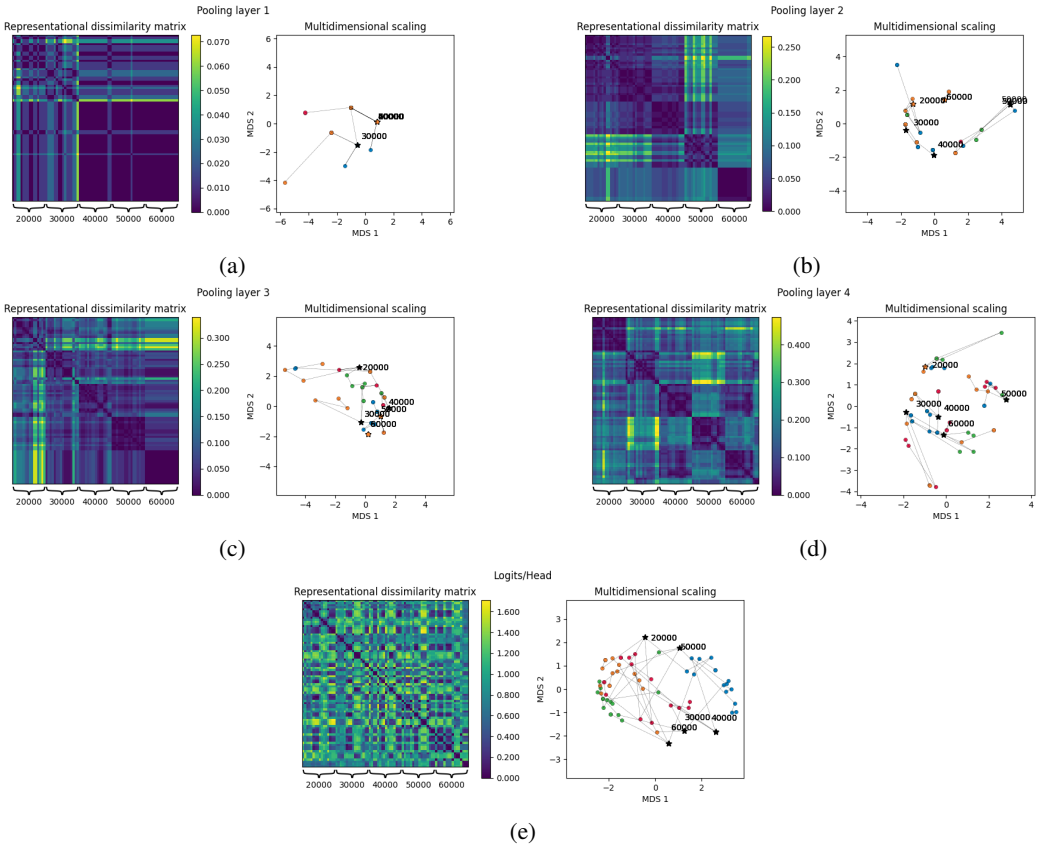

Figure 6: This figure shows the same as Figure 3, this time on Mini-ImageNet.

E ACCESS TO THE CODE

The code to reproduce all the results is available on GitHub[1]. For further details, please follow the instructions there. The MAML models are trained with the code from the original work on MAML (Finn et al., 2017)[2].

[1] https://github.com/ThomasGoerttler/similarity-analysis-of-maml
[2] https://github.com/cbfinn/maml