# Instance Specific Data Augmentation for Meta-Learning

Eric Tang and Pranay Agrawal

Stanford University
{eatang,pranagra}@stanford.edu

## Extended Abstract

In our work, we experiment with the use of an instance-based augmentation method in a meta-learning setting. Past experiments in traditional supervised settings have shown that learning an input-specific augmentation can outperform global augmentations that are sampled independently from the input data. Furthermore, data augmentation has been shown to improve performance in meta-learning settings. To our knowledge, however, limited work has been performed on instance-based augmentation in meta-learning settings. To this end, we benchmark the performance of learning instance-based cropping augmentations on the few shot datasets CIFAR-FS and mini-ImageNet, which are designed for K-shot learning. Training in a fully end-to-end manner, we apply InstaAug on a meta-learning framework using ProtoNet with a ResNet-12 backbone on 5-way 1-shot, 5-shot, and 10-shot for CIFAR-FS and 5-way 1-shot on mini-ImageNet (due to limited resources). Our results show that a learned invariance module during meta training is able to outperform a random baseline. Furthermore, our results support past work on standard global augmentations that suggest query augmentation is the most effective and that support and shot augmentation can harm performance. However, unlike past work in traditional supervised settings, we find that instance-based augmentation in meta-learning settings performs worse (2-3% absolute) than CutMix. Taking a closer look at the results, we notice that the generalization gap between training and validation accuracy is similar and large across no augmentation, random cropping, and instance cropping but minimal for CutMix. As meta-learning suffers greatly from overfitting, we conclude that stronger regularization is needed for meta-learning when compared to traditional supervised learning settings. Finally, we analyze applying instance-based augmentations at meta-test time, and find that unlike in the supervised setting, applying instance-based data augmentations to generate additional image views at test time does not improve classifier accuracy. We suggest that this is a result of the instance-based augmentation module being fit to the training task distribution rather than the test task distribution.

## 1 Introduction

Data augmentation for image classification has become vital as it offers a simple way to boost performance. By augmenting the data, one can increase the dataset while reducing overfitting during training. Existing efforts involve global augmentation where an augmentation such as cropping or rotation is sampled independently with no reliance on the input data. However, global augmentation has limitations in that it may over-exploit or under-exploit the transformation of an image, accidentally transforming the image to a different label or limiting the diversity of a label, respectively. In Figure 1 from Miao et. al, rotating the number "6" transforms it into a "9", or color jittering from yellow to green transforms a lemon into a pro.

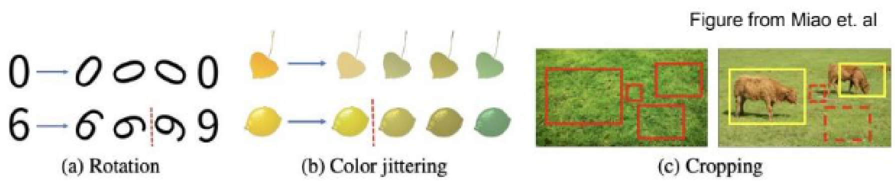


**Fig. 1.** The augmentation of some images can increase the diversity and size of the dataset, but the same augmentation on a different image can transform the label to a different label. For example, 0 is rotation invariant whereas 6 is not.

More recent work has been performed on learning input-specific augmentation. Unlike a global augmentation, by training a learned invariance module, we can sample a distribution dependent on the input for our augmentations. InstaAug, which we use as a framework for our results, learns input-dependent transformations that can be trained end-to-end with a downstream model. By learning a better distribution over augmentations it is possible to improve the distribution of training data and improve overall performance.

Furthermore, data augmentation in meta-learning settings has also received increasing attention to boost few-shot performance, motivated by the insight that meta-learners are particularly vulnerable to overfitting, due to both potential overfitting at the task level (overfitting to the support set), as well as potentially overfitting to the distribution of training tasks (Rajendran et al. (2020)). Prior work on helping this overfitting phenomenon has included work on augmenting tasks by applying random modifications to images (Liu et al. (2020); Santoro et al. (2016)) and mixing labels (Rajendran et al. (2020)). However, to our knowledge, there is limited work performed with learning input-dependent augmentations for meta-learning. As capturing these local invariances has shown to improve performance in traditional supervised settings, our goal is to experiment if instance-specific data augmentation is viable for meta-learning. Using CIFAR-FS and mini-ImageNet, two datasets for benchmarking few-shot meta-learning performance, we attempt to learn instance-based data augmentations for the meta-learning setting.

## 2 Related Work

### 2.1 Learning to Augment

There exists a body of work focusing on learning augmentations from training data for a fixed supervised learning task. Benton et al. (2020) proposes learning a distribution over augmentations for a specific task via parameterization of a set of augmentations and end-to-end training. Miao et al. (2022) further proposes parameterizing the distribution of augmentations based on the input image and training end to end. Other works like Zhou et al. (2020) and Cheung & Yeung (2022) suggest learning the augmentation module on a separate validation set before applying it to a training set for learning. However, to our knowledge, there are no prior works suggesting learning instance-specific augmentations in a meta-learning setting.

### 2.2 Data Augmentation for meta-learning

Previous work from Ni et al. (2020) on applying data augmentation for meta-learning has shown improvements in generalization from applying methods such as MixUp and CutMix. There also exists work that discusses the topic of meta-augmentation, in which additional tasks are created from the sampled training task distribution by either modifying images, as in Santoro et al. (2016) and Liu et al. (2020), or by modifying labels, as in Rajendran et al. (2020). In our work, we intend to focus on exploring the case of modifying the input images via instance-specific augmentations for task, query, and shot augmentation in a setting most similar to Ni et al. (2020), but with learned rather than randomly sampled augmentations.

## 3 Instance Specific Data Augmentation for meta-learning

In this section, we describe how we adapt Instance Specific Data Augmentation (Miao et al. (2022)) for the meta-learning setting.

### 3.1 meta-learning Problem Setup

We follow the standard N-way K-shot meta-learning problem setup, where given a dataset $\mathcal{D}$ we sample a task $T_i = \{D_i^s, D_i^q\}$ by first sampling $N$ classes, then sampling $K$ examples from each task in order to form a support set $D_i^s$, and finally sampling some number of additional examples from each of the $N$ classes to form a query set $D_i^q$.

For modeling, we use the ProtoNet (Snell et al. (2017)), where the aim is to learn a set of model parameters $\theta$ used to compute a feature vector $f_\theta(x)$ for a given input $x$. The prototype of each of the $N$ classes in the support set is defined as the mean of the $K$ feature vectors belonging to the class. Classification on examples from the query set is then carried out by computing the nearest prototype to query set feature vectors. We aim to minimize the negative log-likelihood of the query data, which can be calculated using the cross-entropy loss.

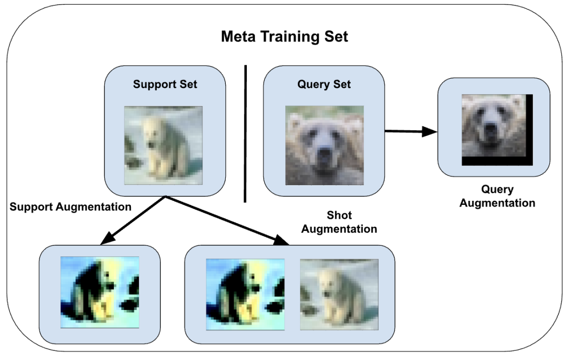


**Fig. 2.** Meta training support, shot, and query augmentation. Support and query augmentation replace the original image with the augmented image for the respective set, whereas shot augmentation appends the augmented image to the support set.

### 3.2 Data Augmentation for meta-learning Setup

Ni et al. (2020) suggests 4 categories of augmentation that can be applied to the meta-learning pipeline - support, query, task, and shot augmentation. As shown in Figure 2, support augmentation replaces the support training images with the transformed image, shot augmentation appends augmented support training images to the original set of support training images, and query augmentation transforms the query training images. We omit task augmentation (in which the applied label-preserving data augmentation is used to generate additional tasks to sample from during meta-training), in our experiments, and leave it as a direction for future exploration.

### 3.3 Instance Specific Data Augmentation for meta-learning Setup

We learn an instance-specific data augmentation as described in Miao et al. (2022), which involves training an augmentation module together with a meta-learning model in an end-to-end fashion in order to minimize the expected loss while enforcing the diversity of the augmented images. The invariance model $\phi$, which is a trainable neural network, sits between inputs $x$ from the meta-training set, and the meta-learning classifier $f$. At meta-training time, given an input, we sample a transformation $\tau \sim p(\tau; \phi(x))$ to generate an augmented sample from either the support or the query set. We then follow Miao et al. (2022) in solving the constrained optimization problem

$$min_{f,\phi}\mathbb{E}_{x,y\sim p_{data}}[\mathbb{E}_{\tau\sim p(\tau;\phi(x)}[\mathcal{L}(f(\tau(x),y)]]$$
$$\text{s.t. } \mathbb{E}_{x,y\sim p_{data}}[\mathbb{H}[p(\tau;\phi(x)]] \in [H_{min}, H_{max}]$$

This is possible by parameterizing $p$ appropriately, solving for entropy in closed form, and using the following Lagrangian function, where the value of $\lambda$ is tuned based on

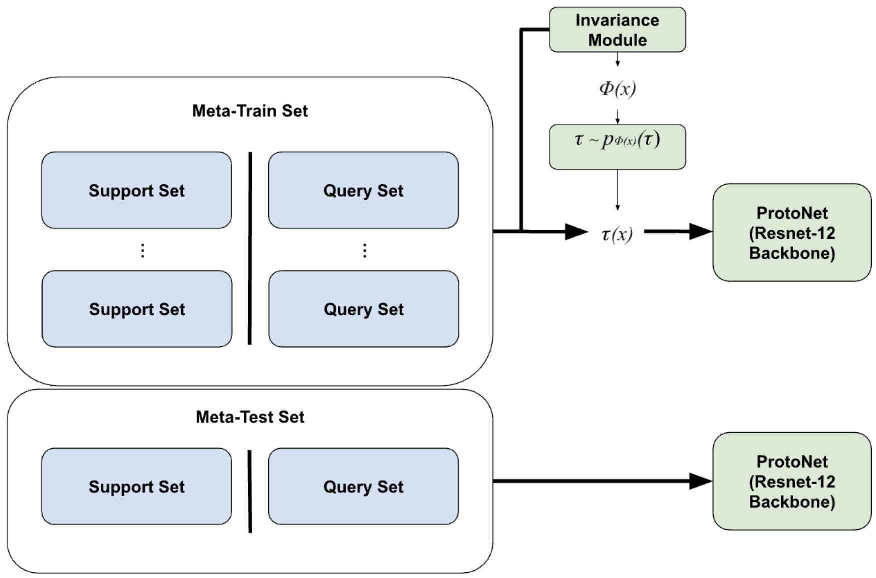


**Fig. 3.** Instance-specific data augmentation for the meta-learning pipeline. Trains the invariance module and then samples this distribution to generate $\tau(x)$ using a ProtoNet with a ResNet-12 backbone.

the current entropy (increasing $\lambda$ if average entropy over a batch is lower than $H_{min}$, and decreasing $\lambda$ if average entropy is higher than $H_{max}$).

$$\mathbb{E}_{x,y\sim p_{data}}[\mathbb{E}_{\tau\sim p(\tau;\phi(x))}[\mathcal{L}(f(\tau(x),y)]] - \lambda\mathbb{E}_{x,y\sim p_{data}}[\mathbb{H}[p(\tau;\phi(x)]] \qquad (1)$$

We note that since we are in a meta-learning setting where the model parameters are backpropagated through the meta-training query examples, the $x$ in the above expressions corresponds to the sampled query data $x \sim D_i^q$. Thus, in order to perform gradient descent on the learned invariance module, augmentation must always be performed on the query data, since ProtoNet only computes gradient updates on the query set. This means that shot and support augmentation are unable to be considered separately from query augmentation with our proposed setup. However, considering that Ni et al. (2020) find that support and shot augmentation may degrade performance, and find that query augmentations tend to improve accuracy, this is unlikely to have impacted our findings.

### 3.4 Test Time Instance Specific Augmentations

Test time data augmentation, in which multiple data augmentations are performed on test images, and predictions are averaged across the sampled augmentations, has shown to improve performance for traditional supervised learning, being used in the AlexNet and ResNet papers (Krizhevsky et al. (2012); He et al. (2015)). We investigate using our learned invariance model to generate additional views of meta-test query set images

$x \sim D_i^q$ by sampling $n$ different transformations $\tau_i$ from $p(\tau; \phi(x))$, and averaging over the outputs to produce a more robust classifier.

# 4 Experiments and Results

We primarily experiment on learning instance-based cropping on the CIFAR-FS dataset (Bertinetto et al. (2018)). For our meta-learning model, we use a ProtoNet with a ResNet-12 backbone. We also run additional experiments on Mini-ImageNet (Vinyals et al. (2016)).

## 4.1 Datasets

The CIFAR-FS dataset consists of 32x32 images from 100 classes with 600 examples in each class, sampled from the CIFAR-100 dataset. Similarly, the Mini-ImageNet dataset consists of 84x84 images from 100 classes with 600 examples in each class, sampled from the ImageNet dataset. Both datasets split their 100 classes into 64 classes for training, 16 classes for validation, and 20 classes for testing.

## 4.2 Experiment Details

All models are trained for 40 epochs, with 8000 episodes per epoch, and batches of 8 tasks. We also fix the number of classes per task, $N$, as 5. Both the learned invariance module and the ProtoNet backbone are trained with standard SGD optimizers, with the invariance module using a fixed learning rate of $1e-5$, and the meta-learning backbone using an initial learning rate of $0.1$, with nesterov momentum and weight decay. Validation set accuracies are calculated over 2000 episodes, and with 15 query examples per class.

| Method | Mode | 1-Shot | 5-Shot | 10-Shot |
|---|---|---|---|---|
| No Aug | - | 58.84 ± 0.56 | 74.47 ± 0.42 | 78.37 ± 0.37 |
| CutMix | Query | **64.08 ± 0.57** | **79.14 ± 0.38** | **81.54 ± 0.36** |
| InstaCrop | Query | 60.91 ± 0.55 | 76.64 ± 0.39 | 79.32 ± 0.37 |
| RandomCrop | Query | 59.41 ± 0.56 | 75.06 ± 0.39 | 79.01 ± 0.36 |
| InstaCrop | Support + Query | 58.62 ± 0.56 | 75.13 ± 0.39 | 77.08 ± 0.38 |
| RandomCrop | Support + Query | 50.56 ± 0.53 | 64.55 ± 0.45 | 70.97 ± 0.40 |
| InstaCrop | Shot + Query | 59.68 ± 0.55 | 75.67 ± 0.41 | 78.47 ± 0.37 |
| RandomCrop | Shot + Query | 57.30 ± 0.53 | 71.69 ± 0.41 | 77.22 ± 0.38 |

**Table 1.** Instance Based Cropping (InstaCrop) Augmentations on CIFAR-FS for 1,5,10 Shot learning with a ProtoNet using a ResNet-12 backbone.

**CIFAR-FS Instance Cropping Settings** We first experiment on the CIFAR-FS dataset. We attempt using our instance-based cropping augmentation module for query augmentation, query + support augmentation, and query + shot augmentation in the 1-Shot, 5-Shot, and 10-Shot settings. We benchmark our model against a baseline of no augmentation and global random cropping with scale $[0.2, 1]$. For the cropping invariance module, we additionally have to set $H_{min}$ and $H_{max}$, which enforce constraints on the diversity of the learned crops. After a brief sweep, we found $[3, 3.5]$ to be the best setting, which is in line with the findings from Miao et al. (2022).

**CIFAR-FS Instance Cropping Results** From Table 1 we can see that the invariance module is able to learn a better distribution over the space of possible crops than random cropping, with higher accuracies for instance-based cropping when applied to all three of query, query + shot, and query + support augmentation compared to the random baseline. In a meta-learning setting, since the invariance module is learning augmentations just over the distribution of training tasks, we hypothesize that the improvement in performance comes from the module learning the most effective crops to enhance the pool of training time query images, which allows the meta learner to more effectively generalize at meta test time.

The gap between random and instance cropping is especially noticeable for query + support augmentation, suggesting that the learned invariance module is able to learn to avoid crops that entirely exclude the object in the image or that could result in a different label, which is especially important for augmentations on the support images.

Additionally, we find that instance-based cropping is able to consistently outperform the baseline of no augmentation when applied to just the query set. For query + support and query + shot augmentation, instance cropping leads to either equivalent or slightly worse accuracy, which is in line with findings for random augmentations from Ni et al. (2020). In particular, we hypothesize that shot augmentation could contribute to overfitting issues on the support set, canceling out any potential benefit from the learned instance augmentation.

However, we find that unlike in the traditional supervised setting, CutMix drastically outperforms the best instance-based cropping method, with around an additional 2-3% gain in performance relative to instance-based cropping on the query set. We hypothesize that data augmentations with relatively stronger regularizing effects such as CutMix are even more effective for meta-learning due to the increased potential for overfitting in the meta-learning process. Cropping is typically combined with additional augmentations such as random masking and color jittering for additional regularization - future directions for instance-based augmentations for meta-learning could include learning instance-based augmentations for other augmentations in parallel with the cropping module for a meta-learning setting.

We can see the difference in the generalization gap between methods in Figure 4, where there is clear overfitting in the no augmentation setting, slight improvements in the generalization gap for random cropping and instance-based cropping, and almost no generalization gap for CutMix, leading to the highest relative accuracy.

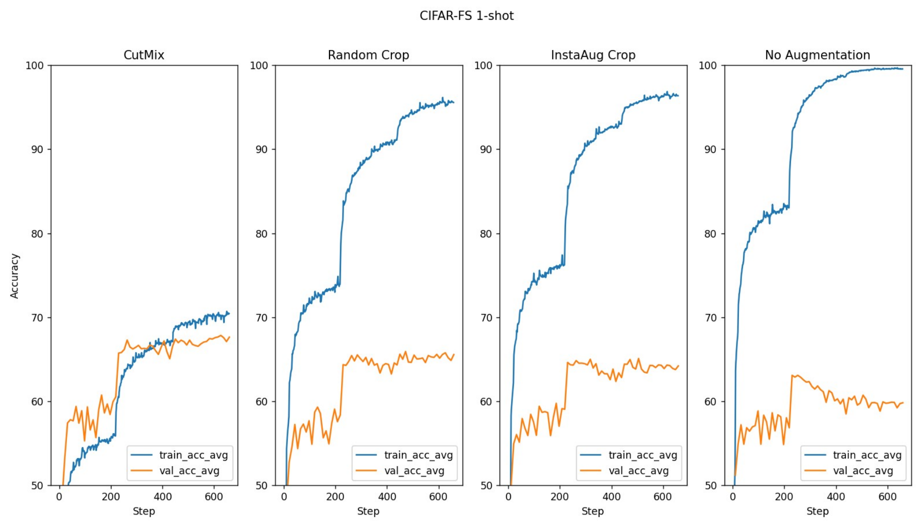


**Fig. 4.** CIFAR-FS 1-shot average training accuracy (blue) and average validation accuracy (orange) for CutMix, random crop, InstaAug crop, and no augmentation on the query set. All runs additionally used task augmentation with rotation. CutMix performed the best and the generalization gap for CutMix is much less when compared to random crop, InstaAug crop, and no augmentation.

| Method | Mode | 1-Shot |
|---|---|---|
| No Aug | - | 61.84 ± 0.53 |
| CutMix | Query | **66.14 ± 0.52** |
| InstaCrop | Query | 62.88 ± 0.52 |
| RandomCrop | Query | 62.84 ± 0.52 |

**Table 2.** 1-Shot performance of Instance Based Cropping on Mini-ImageNet

**Mini-ImageNet Instance Cropping Results** Due to limited time and increased computational requirements of Mini-ImageNet, we were unable to run as many experiments for Mini-ImageNet. We used the same scale parameterization for random cropping and the same entropy parameters as the the CIFAR-FS Experiments. We find a similar trend in that CutMix drastically improves over the no augmentation baseline, while both cropping methods improve just marginally over the baseline.

### 4.3 Test Time Data Augmentation

We additionally experiment with using our learned instance-based augmentation at test time to generate additional views over meta-test query images. In Table 3, we can see that using our learned invariance module for test time data augmentation leads to a decrease in accuracy, with the trend showing that additional views using the instance-based cropping lead to further decreases in accuracy. On the other hand, increasing the number of views using random cropping shows the opposite trend, with accuracy increasing as a function of the number of views. We hypothesize that since the instance cropping module learns the underlying distribution of the classes in the meta-training task set, it does not generalize well when applied to test time samples when compared to a random cropping baseline. This supports our earlier hypothesis that the instance cropping module successfully learns the underlying training task distribution, which is why it outperforms the random baseline when applied to the training query set.

| Method | Mode | Number of Samples | Accuracy |
|---|---|---|---|
| Augmentation No Test Aug | - | 1 | 76.64 ± 0.39 |
| Instance Cropping | Query | 2 | 75.55 ± 0.4 |
| Instance Cropping | Query | 5 | 73.87 ± 0.4 |
| Instance Cropping | Query | 10 | 72.99 ± 0.41 |
| Random Cropping | Query | 2 | 76.86 ± 0.39 |
| Random Cropping | Query | 5 | 77.01 ± 0.39 |
| Random Cropping | Query | 10 | 77.12 ± 0.39 |

**Table 3.** CIFAR-FS Test Time Data Augmentation - evaluations carried out in a 5-shot 5-way setting.

## 5 Conclusions and Future Work

Global data augmentation for meta-learning is known to improve accuracy, but there is limited experimentation performed on instance-based augmentation in a meta-learning setting. From our results, we support past findings for global augmentations that instance-based support and shot augmentation can harm performance while instance-based query augmentations can improve performance. We also find that using instance-based cropping for query augmentation performs better than random cropping, suggesting that it is valuable for an instance-based data augmentation to learn the underlying training task distribution, however, we find that CutMix still results in higher accuracy. We hypothesize that stronger regularizers are needed for meta data augmentation compared to traditional supervised settings.

Additionally, we find that unlike in the supervised setting, applying instance-based data augmentations to generate additional image views at test time does not improve classifier accuracy. We suggest that this is a result of the instance-based augmentation module being fit to the training task distribution rather than the test task distribution.

As a result of our findings, one avenue of future work could explore additional instance-based augmentations that are “stronger” in nature (i.e. random masking) or the effect on performance by stringing together multiple instance-based augmentations. Furthermore, one could also investigate instance-based data augmentations across different datasets and meta-learning model types.

# Bibliography


Gregory Benton, Marc Finzi, Pavel Izmailov, and Andrew Gordon Wilson. Learning invariances in neural networks, 2020. URL https://arxiv.org/abs/2010.11882.

Luca Bertinetto, João F. Henriques, Philip H. S. Torr, and Andrea Vedaldi. Meta-learning with differentiable closed-form solvers. *CoRR*, abs/1805.08136, 2018. URL http://arxiv.org/abs/1805.08136.

Tsz-Him Cheung and Dit-Yan Yeung. Adaaug: Learning class- and instance-adaptive data augmentation policies. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=rWXfFogxRJN.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. URL http://arxiv.org/abs/1512.03385.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

Jialin Liu, Fei Chao, and Chih-Min Lin. Task augmentation by rotating for meta-learning. *CoRR*, abs/2003.00804, 2020. URL https://arxiv.org/abs/2003.00804.

Ning Miao, Emile Mathieu, Yann Dubois, Tom Rainforth, Yee Whye Teh, Adam Foster, and Hyunjik Kim. Learning instance-specific data augmentations. *arXiv preprint arXiv:2206.00051*, 2022.

Renkun Ni, Micah Goldblum, Amr Sharaf, Kezhi Kong, and Tom Goldstein. Data augmentation for meta-learning. *CoRR*, abs/2010.07092, 2020. URL https://arxiv.org/abs/2010.07092.

Janarthanan Rajendran, Alex Irpan, and Eric Jang. Meta-learning requires meta-augmentation. 2020. https://doi.org/10.48550/ARXIV.2007.05549. URL https://arxiv.org/abs/2007.05549.

Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy P. Lillicrap. Meta-learning with memory-augmented neural networks. In *ICML*, pp. 1842–1850, 2016. URL http://proceedings.mlr.press/v48/santoro16.html.

Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. *CoRR*, abs/1703.05175, 2017. URL http://arxiv.org/abs/1703.05175.

Oriol Vinyals, Charles Blundell, Timothy P. Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. *CoRR*, abs/1606.04080, 2016. URL http://arxiv.org/abs/1606.04080.

Fengwei Zhou, Jiawei Li, Chuanlong Xie, Fei Chen, Lanqing Hong, Rui Sun, and Zhenguo Li. Metaaugment: Sample-aware data augmentation policy learning. *CoRR*, abs/2012.12076, 2020. URL https://arxiv.org/abs/2012.12076.