Incremental Few-Shot Meta-Learning via Indirect Discriminant Alignment - Supplementary Material

Qing Liu[1], Orchid Majumder[2], Alessandro Achille[2], Avinash Ravichandran[2], Rahul Bhotika[2], and Stefano Soatto[2]

[1] Johns Hopkins University
[2] Amazon Web Services

1 Contents

As referenced in the main paper, in the supplementary material, we

- Provide the full expression for the loss function defined in Eq. 9 for Prototypical Networks (PN) [2] in Section 2.
- Compare the performance between Indirect Discriminant Alignment (IDA) and Exemplar Incremental Meta-Learning (EIML) for all three datasets - MiniImageNet, TieredImageNet and DomainImageNet with PN as the meta-learner.
- Show performance of IDA on different tasks by varying λ. Along the same lines, we show the performance of EIML as we vary the number of exemplars stored from the old task distribution.
- Provide additional empirical results on DomainImageNet dataset to better portray the impact of the domain gap.
- Compare the performance of different meta-learning algorithms when the "shots" and "ways" vary between meta-training and few-shot testing.

2 Full expression for the loss function

For PN, we have the following functions:

$$\begin{aligned} \chi(z, c) &:= -\|z - c\|^2, \\ \psi(\mathcal{D}_\tau, k) &:= \tfrac{1}{|\mathcal{C}_k|} \textstyle\sum_i \delta_{y_i,k} z_i, \\ \mathcal{C}_\tau &= \psi(\mathcal{D}_\tau, k)_{k=1}^K. \end{aligned}$$

Using these choices, we can rewrite $f_w(y = k|x, \mathcal{C})$ as

$$f_w(y = k|x, \mathcal{C}) = \frac{e^{-\|z-c_k\|^2}}{\sum_j e^{-\|z-c_j\|^2}}. \tag{1}$$

Applying the negative logarithm on both sides we get

$$-\log(f_w(y=k|x,\mathcal{C})) = \|z-c_k\|^2 + \log\sum_j e^{-\|z-c_j\|^2} \tag{2}$$

$$= \|z-c_k\|^2 + \mathrm{LSE}(z,\mathcal{C}), \tag{3}$$

where $\mathrm{LSE}(z,\mathcal{C}) = \log\sum_{c\in\mathcal{C}} e^{-\|z-c\|^2}$.

The new model weights w_{t+1} can then be obtained by solving the following optimization problem

$$\arg\min_{w_{t+1}} \mathcal{L}(w_{t+1}) = \arg\min_{w_{t+1}} L(w_{t+1};\mathcal{E}) + \lambda\ \mathrm{IDA}_{\mathcal{E}}(\phi_{w_{t+1}}|\phi_{w_t};\mathcal{C}^t), \tag{4}$$

where the Indirect Discriminant Alignment (IDA) loss is defined by

$$\mathrm{IDA}_{\mathcal{E}}(\phi_{w_{t+1}}|\phi_{w_t};\mathcal{C}^t) = \mathbb{E}_{x\sim\mathcal{E}^{\tau'},\hat{\mathcal{C}}\sim\mathcal{C}^t}\big[\mathrm{KL}(f_{w_{t+1}}(y|x,\hat{\mathcal{C}}))||f_{w_t}(y|x,\hat{\mathcal{C}})\big]. \tag{5}$$

Here

$$f_{w_{t+1}}(y=k|x,C_\tau^t) = \frac{e^{-\|\phi_{w_{t+1}}(x)-c_k^t\|^2}}{\sum_j e^{-\|\phi_{w_{t+1}}(x)-c_j^t\|^2}}, \tag{6}$$

$$f_{w_t}(y=k|x,C_\tau^t) = \frac{e^{-\|\phi_{w_t}(x)-c_k^t\|^2}}{\sum_j e^{-\|\phi_{w_t}(x)-c_j^t\|^2}}. \tag{7}$$

Eq (7) shows the discriminant calculated using: the old model embeddings, the old class centers and the input as new classes. Similarly (6) shows the discriminant calculated using: the new model embeddings, the old class centers and the input as new classes.

The cross-entropy loss $L(w;\mathcal{E})$ can be rewritten explicitly in the case of prototypical networks as:

$$L(w;\mathcal{E}) = \frac{1}{N_\tau}\sum_\tau \frac{1}{|\mathcal{E}_\tau|}\sum_{(x_i,y_i)\in\mathcal{E}_\tau} -\log p_w^\tau(y_i|x_i) \tag{8}$$

$$= \frac{1}{N_\tau}\sum_\tau \frac{1}{|\mathcal{E}_\tau|}\sum_{(x_i,y_i)\in\mathcal{E}_\tau} \|z_i - c_{y_i}^\tau\|^2 + \mathrm{LSE}(z_i,\mathcal{C}). \tag{9}$$

We use this as our loss when we meta-train with PN. The loss for ECM is identical, with the caveat that the class identities c_τ^t are not sample means but are instead learned via optimization [1].

3 Comparison of Exemplar Incremental Meta-Learning and Indirect Discriminant Alignment

In Table 1, we show the performance comparison between EIML and IDA on all datasets we considered — both in the 5-shot 5-way and 1-shot 5-way setup

Table 1: Comparison of EIML and IDA across multiple datasets

dataset	Method	1-shot 5-way			5-shot 5-way		
		Old classes	New classes	Unseen classes	Old classes	New classes	Unseen classes
MiniImageNet	EIML	$\mathbf{68.95 \pm 0.50}$	71.43 ± 0.52	54.86 ± 0.42	$\mathbf{90.20 \pm 0.20}$	86.91 ± 0.25	74.39 ± 0.50
	IDA	66.54 ± 0.49	$\mathbf{71.92 \pm 0.51}$	$\mathbf{55.22 \pm 0.43}$	89.14 ± 0.21	$\mathbf{87.32 \pm 0.25}$	$\mathbf{75.11 \pm 0.31}$
TieredImageNet	EIML	72.50 ± 0.51	69.44 ± 0.52	58.42 ± 0.50	88.93 ± 0.27	86.25 ± 0.32	77.97 ± 0.42
	IDA	$\mathbf{72.65 \pm 0.51}$	$\mathbf{70.17 \pm 0.53}$	$\mathbf{58.71 \pm 0.50}$	$\mathbf{89.03 \pm 0.27}$	$\mathbf{86.91 \pm 0.31}$	$\mathbf{78.40 \pm 0.42}$
DomainImageNet	EIML	$\mathbf{48.48 \pm 0.44}$	$\mathbf{42.27 \pm 0.41}$	43.91 ± 0.42	$\mathbf{83.23 \pm 0.25}$	$\mathbf{65.81 \pm 0.40}$	69.99 ± 0.36
	IDA	42.57 ± 0.41	38.95 ± 0.39	$\mathbf{44.88 \pm 0.43}$	81.26 ± 0.27	65.78 ± 0.41	$\mathbf{70.36 \pm 0.36}$

— using prototypical network as the meta-learner. While we expected EIML to perform better than IDA on all scenarios due to availability of additional samples from the old task distribution, we actually observed that IDA outperforms EIML on unseen classes and performs equally well on few-shot tasks containing new classes. However, as expected, EIML performs better when it comes to handling tasks from the old task distribution.

4 Varying λ in Indirect Discriminant Alignment

By varying λ for IDA (Eq. 9), we can maintain a trade-off between the IDA loss and the standard meta-learning loss. In this experiment, we investigate how changing the value of λ affects the model's performance on old, new and the unseen test set. We investigated this in the 5-shot 5-way setup for MiniImageNet using PN. We chose $\lambda \in \{0.0, 0.25, 0.5, 0.75, 1.0, 2.0, 5.0, 10.0\}$. All numbers reported in the main paper were with $\lambda = 1.0$. The outcome of this experiment can be visualized in Fig. 1 where we can see that as λ is increased, the model's performance on old tasks improves while its performance deteriorates on the new set of tasks. This is intuitive as IDA loss adds a constraint for the incremental model to be similar to the model learned from old tasks and increasing its contribution in the overall loss enhances the model's ability to perform better on old tasks. However, for the same reason, it diminishes the impact of standard meta-learning loss and hampers performance on new tasks. At $\lambda = 0.0$, the model only trains with the standard meta-learning loss and its performance is identical to our baseline named "Fine-Tuning (FT)". At λ value of 10.0, IDA loss dominates and the performance is similar (not fully identical though) to our "No Update (NU)" baseline which is the model trained only using old tasks. While measuring performance on the unseen clases, we observe that the ideal value of λ lies somewhat between the values of λ that provide the best performance on the old tasks and the values of λ that work best for the new tasks. The farther λ values go away from this range, it degrades the model's performance on the unseen tasks.

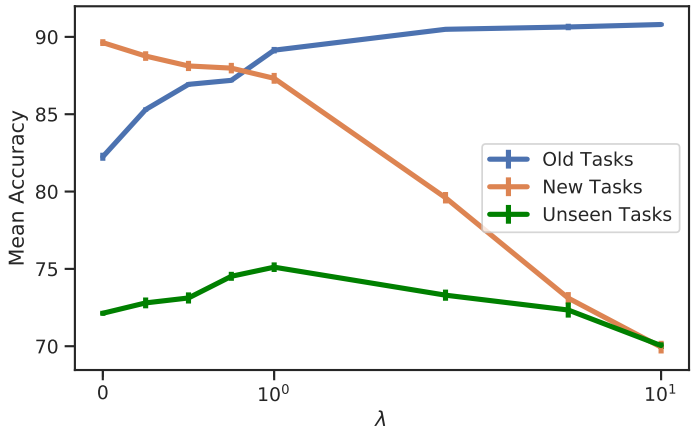

Fig. 1: Performance of the model on different few-shot tasks: old, new and unseen with λ varying from 0.0 to 10.0. The figure shows the mean accuracy averaged over 2000 episodes. The error bars indicate the 95% confidence interval.

5 Varying number of samples in Exemplar Incremental Meta-Learning

While using exemplars in EIML to retain information about the old tasks for meta-training our model incrementally, we kept the number of exemplars fixed to 15. We investigate the effect of storing more exemplars per class to see if it helps improve performance on old, new or unseen tasks. We investigated this in a 5-shot 5-way setup for MiniImageNet using PN with number of examples $\in \{15, 30, 60, 120\}$. The outcome of this experiment is shown in Fig. 2. This shows that increasing the number of exemplars per class does not yield much positive advantage on any of the task segments i.e., old, new or unseen.

6 Additional experimental results on DomainImageNet

In Tables 2, 3, 4, 5, 6 & 7 we report some additional experimental results on the DomainImageNet dataset which explain the effects of domain gap better while using the baseline methods and our proposed algorithm. In our original paper, in Table 4 and Table 5, we showed the result of applying Incremental Meta-Learning (IML) on DomainImageNet for 5-shot 5-way setup using PN and ECM where the model is first trained using natural object classes and then incrementally trained using man-made object classes. Here we provide results for the same scenario in a 1-shot 5-way setup and also the results for PN and ECM in both 1-shot 5-way and 1-shot 5-way setup by reversing the domains i.e., the model is first trained

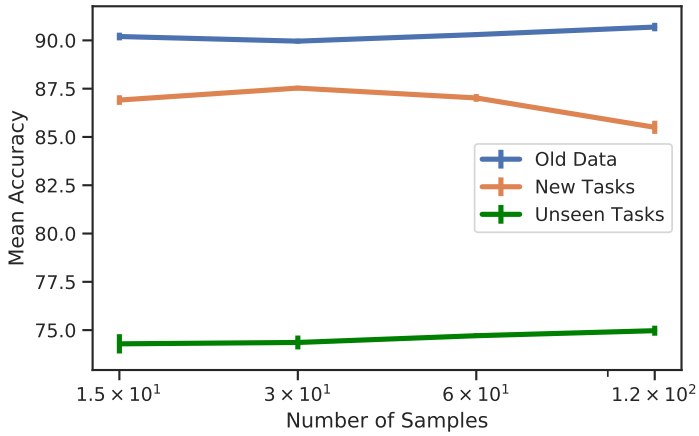

Fig. 2: Performance of the EIML model with different number of exemplars. The figure shows the mean accuracy averaged over 2000 episodes. The error bars indicate the 95% confidence interval.

Table 2: Results of 1-shot 5-way classification accuracy of our method on different sets of DomainImageNet using PN as the meta-learning algorithm. Here old tasks are from natural images and incremental tasks are from man-made images.

Model	Old classes from old domain (32)	New classes from new domain (32)	Unseen classes from old domain (20)	Unseen classes from new domain (20)	Unseen classes from both domains (40)
NU	56.32 ± 0.50	32.88 ± 0.34	37.90 ± 0.37	33.91 ± 0.33	39.56 ± 0.40
FT	37.12 ± 0.36	$\mathbf{61.34 \pm 0.56}$	32.81 ± 0.32	$\mathbf{46.61 \pm 0.43}$	42.91 ± 0.42
DFA	37.20 ± 0.37	60.23 ± 0.55	32.32 ± 0.32	46.12 ± 0.44	42.20 ± 0.41
IDA	$\mathbf{42.57 \pm 0.41}$	59.31 ± 0.55	$\mathbf{35.01 \pm 0.35}$	45.55 ± 0.43	$\mathbf{43.91 \pm 0.42}$
Paragon	62.23 ± 0.50	67.25 ± 0.53	40.06 ± 0.39	50.78 ± 0.45	54.57 ± 0.48

using tasks consisting of classes from man-made images and then incrementally trained using natural images.

7 Meta-Learning algorithms and their sensitivity to "shot" and "ways"

We investigate the sensitivity of PN and ECM with respect to "shot" and "ways". This result is shown in Table 8 and Table 9. From these tables, we see that PN is sensitive to the number of "shots" and "ways" it is being trained on. We see a drop in performance when the shots are changed between meta-training and few-shot testing. We also notice for PN, training using a larger number of "ways"

Table 3: Results of 1-shot 5-way classification accuracy of our method on different sets of DomainImageNet using ECM as the meta-learning algorithm. Here old tasks are from natural images and incremental tasks are from man-made images.

Model	Old classes from old domain (32)	New classes from new domain (32)	Unseen classes from old domain (20)	Unseen classes from new domain (20)	Unseen classes from both domains (40)
NU	72.58 ± 0.40	39.68 ± 0.37	45.33 ± 0.39	40.43 ± 0.34	46.72 ± 0.39
FT	48.54 ± 0.40	$\mathbf{76.46 \pm 0.43}$	39.37 ± 0.35	$\mathbf{54.93 \pm 0.41}$	51.64 ± 0.43
DFA	49.83 ± 0.40	74.63 ± 0.43	40.36 ± 0.36	54.80 ± 0.41	52.32 ± 0.43
IDA	$\mathbf{69.82 \pm 0.41}$	62.56 ± 0.44	$\mathbf{46.22 \pm 0.39}$	49.83 ± 0.40	$\mathbf{53.03 \pm 0.41}$
Paragon	70.10 ± 0.42	73.70 ± 0.44	46.81 ± 0.40	55.19 ± 0.41	57.97 ± 0.42

Table 4: Results of 5-shot 5-way classification accuracy of our method on different sets of DomainImageNet using PN as the meta-learning algorithm. Here old tasks are from man-made images and incremental tasks are from natural images.

Model	Old classes from old domain (32)	New classes from new domain (32)	Unseen classes from old domain (20)	Unseen classes from new domain (20)	Unseen classes from both domains (40)
NU	91.75 ± 0.18	50.67 ± 0.37	70.94 ± 0.35	45.04 ± 0.34	62.57 ± 0.44
FT	71.41 ± 0.31	$\mathbf{84.40 \pm 0.30}$	57.11 ± 0.32	59.61 ± 0.43	63.40 ± 0.38
DFA	73.51 ± 0.31	83.31 ± 0.31	58.87 ± 0.32	59.64 ± 0.42	64.36 ± 0.37
IDA	$\mathbf{88.39 \pm 0.23}$	81.24 ± 0.34	$\mathbf{70.83 \pm 0.34}$	$\mathbf{60.82 \pm 0.39}$	$\mathbf{71.13 \pm 0.38}$
Paragon	90.35 ± 0.21	87.11 ± 0.27	73.03 ± 0.34	62.44 ± 0.42	75.27 ± 0.38

Table 5: Results of 1-shot 5-way classification accuracy of our method on different sets of DomainImageNet using PN as the meta-learning algorithm. Here old tasks are from man-made images and incremental tasks are from natural images.

Model	Old classes from old domain (32)	New classes from new domain (32)	Unseen classes from old domain (20)	Unseen classes from new domain (20)	Unseen classes from both domains (40)
NU	72.45 ± 0.51	33.41 ± 0.34	51.48 ± 0.48	32.32 ± 0.31	43.88 ± 0.45
FT	46.08 ± 0.42	$\mathbf{61.72 \pm 0.56}$	39.90 ± 0.39	42.38 ± 0.43	44.85 ± 0.43
DFA	48.35 ± 0.44	58.13 ± 0.56	40.66 ± 0.40	41.30 ± 0.41	44.04 ± 0.42
IDA	$\mathbf{58.98 \pm 0.47}$	58.90 ± 0.56	$\mathbf{46.33 \pm 0.44}$	$\mathbf{43.14 \pm 0.42}$	$\mathbf{48.33 \pm 0.44}$
Paragon	73.31 ± 0.50	68.54 ± 0.55	52.29 ± 0.45	44.28 ± 0.45	58.11 ± 0.50

helps to increase the performance. However, we notice for ECM, the performance remains the same irrespective of what configuration the meta-learning model was trained with. This shows that the ECM is a better meta-learning algorithm for IML as it provides the flexibility to use any "shot" and "ways" to begin with. These results use the feature extractor proposed in the original PN [2] paper instead of ResNet-12 to train these various scenarios in a short time. All models

Table 6: Results of 5-shot 5-way classification accuracy of our method on different sets of DomainImageNet using ECM as the meta-learning algorithm. Here old tasks are from man-made images and incremental tasks are from natural images.

Model	Old classes from old domain (32)	New classes from new domain (32)	Unseen classes from old domain (20)	Unseen classes from new domain (20)	Unseen classes from both domains (40)
NU	59.58 ± 0.40	91.91 ± 0.16	73.04 ± 0.32	53.20 ± 0.37	68.12 ± 0.40
FT	$\mathbf{87.66 \pm 0.24}$	72.11 ± 0.31	61.39 ± 0.31	$\mathbf{63.87 \pm 0.40}$	67.91 ± 0.35
DFA	82.45 ± 0.31	77.01 ± 0.29	62.91 ± 0.32	62.31 ± 0.42	67.79 ± 0.36
IDA	79.80 ± 0.33	$\mathbf{90.43 \pm 0.19}$	$\mathbf{74.04 \pm 0.31}$	61.31 ± 0.39	$\mathbf{72.60 \pm 0.37}$
Paragon	85.66 ± 0.28	89.44 ± 0.20	74.60 ± 0.32	66.12 ± 0.39	76.92 ± 0.35

Table 7: Results of 1-shot 5-way classification accuracy of our method on different sets of DomainImageNet using ECM as the meta-learning algorithm. Here old tasks are from man-made images and incremental tasks are from natural images.

Model	Old classes from old domain (32)	New classes from new domain (32)	Unseen classes from old domain (20)	Unseen classes from new domain (20)	Unseen classes from both domains (40)
NU	42.27 ± 0.38	79.97 ± 0.40	53.49 ± 0.42	37.67 ± 0.34	49.74 ± 0.42
FT	$\mathbf{73.37 \pm 0.47}$	51.59 ± 0.40	42.22 ± 0.35	$\mathbf{47.09 \pm 0.42}$	48.64 ± 0.41
DFA	66.49 ± 0.50	56.71 ± 0.42	43.68 ± 0.36	46.29 ± 0.43	49.12 ± 0.41
IDA	60.64 ± 0.46	$\mathbf{76.91 \pm 0.41}$	$\mathbf{54.02 \pm 0.42}$	44.01 ± 0.39	$\mathbf{53.16 \pm 0.42}$
Paragon	68.86 ± 0.47	73.87 ± 0.43	53.91 ± 0.42	48.00 ± 0.42	58.00 ± 0.43

Table 8: Performance of different models trained using PN [2] on different few-shot sets with varying "ways" and "shots". Mean accuracy averaged over 2000 episodes is shown here.

Model	1-shot			5-shot		
	5-way	10-way	20-way	5-way	10-way	20-way
1-shot 5-way	49.35	33.78	21.99	65.63	49.60	35.76
1-shot 10-way	51.47	35.65	23.54	68.45	52.80	38.93
5-shot 5-way	47.09	31.94	20.68	69.09	53.52	39.78
5-shot 10-way	45.83	30.89	19.90	69.93	54.77	41.05
Range	5.63	4.76	3.64	4.30	5.18	5.29

were trained using identical hyperparameter settings (batch size, learning rate, optimizer etc.).

Table 9: Performance of different models trained using ECM [1] on different few-shot sets with varying "ways" and "shots". Mean accuracy averaged over 2000 episodes is shown here.

Model	1-shot			5-shot		
	5-way	10-way	20-way	5-way	10-way	20-way
1-shot 5-way	50.13	35.15	23.48	68.40	53.30	39.68
1-shot 10-way	50.91	35.69	23.84	69.62	54.34	40.53
5-shot 5-way	51.15	35.95	24.08	69.50	54.40	40.96
5-shot 10-way	50.78	35.44	23.73	69.64	54.46	40.72
Range	1.0224	0.7921	0.6009	1.2455	1.1688	1.2748

References

1. Ravichandran, A., Bhotika, R., Soatto, S.: Few-shot learning with embedded class models and shot-free meta training. In: International Conference on Computer Vision (2019)
2. Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. In: Advances in Neural Information Processing Systems. pp. 4077–4087 (2017)