An Empirical Study of Meta-Learning: a step towards rigorously understanding meta-learning algorithms

Brando Miranda
University of Illinois Urbana-Champaign
miranda9@illinois.edu

Abstract

It has been recently observed that a good embedding is all we needed to solve many few-shot learning benchmarks. In addition, other work has strongly suggested that MAML mostly works via this same method: by learning a good embedding. This highlights our lack of understanding of what meta-learning algorithms are doing and when they work. In this work we provide preliminary results that shed some light towards understanding meta-learning algorithms better. In particular we identify 3 interesting properties: 1) It's possible to define a synthetic task that results in higher degree of meta-adaptation, thus suggesting that current few-shot learning benchmarks might not have the properties needed for the success of meta-learning algorithms 2) meta-overfitting occurs when the number of classes (or concepts) are finite and this issue disappears once the task has an unbounded number of concepts 3) more adaptation for MAML does not necessarily result in representations that have adapted more or even perform better. Finally, we suggest that to understand meta-learning algorithms better it is imperative that we go beyond tracking only absolute performance and in addition formally quantify the degree of meta-learning and track both metrics together. Reporting results in future work this way should help us identify the sources of meta-overfitting more accurately and hopefully design more flexible meta-learning algorithms. In the appendix we also discuss that quantifying AI safety too is important but is left as future work.

1. Introduction

Few-shot learning is a research challenge that assess an AI model's capacity to quickly adapt to new tasks or new environments. This has been the leading area where AI researchers apply meta-learning algorithms - where a strategy that learns-to-learn quickly is likely to be the most promising. However, it was recently shown by [24] that a model that only has a good embedding is able to match and beat many modern sophisticated meta-learning algorithms. In addition there seems to be growing evidence that this is a real phenomena e.g. [3, 5, 9, 13]. Furthermore, carefully analysis of the representations learned by MAML [10] (on few-shot learning tasks) reveal that this algorithm mainly works by learning a feature that is re-usable for many tasks [23] (i.e. what we are calling a good embedding) in modern few-shot learning benchmarks.

These discoveries reveal a lack of understanding on when meta-learning algorithms work. This is the main motivation for this work. In particular our contributions are:

1. It's possible to define a synthetic task that results in higher degree of meta-adaptation, thus suggesting that current few-shot learning benchmarks might not have the properties needed for the success of meta-learning algorithms
2. Meta-overfitting occurs when the number of classes (or concepts) are finite and this issue disappears once the task has an unbounded number of concepts
3. More adaptation for MAML does not necessarily result in representations that have adapted more or even perform better.

Note: this paper's main goal is to summarize and partially expand on the results presented for the research project presented for the class Learning to Learning [2] with professor Wang at UIUC. For the project video presentation see [18]. Thus, a recapitulation of meta-learning and related techniques has been omitted and will be assumed, but the following are good resources [1, 2].

2. Related Work

Previous work has shown that having a good representation is sufficient to achieve high meta-accuracy on modern few-shot learning tasks (e.g. mini-Imagenet, tiered-Imagenet, Cifar FS, FC100, Omniglot, etc) [24]. In addition related work shows variants of models that primarily rely on a good embedding that support these claims [3, 5, 9, 13].

However, in depth analysis of meta-learning algorithms or there adaptations are lacking. The main work we are aware that does carry this analysis is [23] and to some degree [25] (though their main goal is to propose a large scale few-shot learning benchmark).

3. Unified Framework for studying Meta-learning and Absolute performance

We propose that future work in meta-learning should not only reports absolute performance but quantify and report the degree of meta-learning their algorithms have. This is crucial not only for better understanding of meta-learning algorithms but also because if our eventual goal is Artificial General Intelligence (AGI), then we must make deliberate efforts to measure and define it an actionable way. In addition, such a metric can be useful to be able to diagnose plausible causes of meta-overfitting. For example, a high degree of meta-learning *coupled* with a high generalization gap between meta-train and meta-test errors would suggest meta-overfitting. This would be extremely useful as this might suggest fixes for meta-overfitting (e.g. regularizing the meta-learner, getting more meta-experience through data or larger set of concepts to learn from, and more).

In this work me make a humble but valuable first step - inspired by [23] - by defining the degree of meta-learning by measuring the normalized degree of change in the representation after using meta-adaptation A i.e.

$$ML(f) = Diff(f, A(f))$$

in this work we set $ML(f)$ to be Canonical Correlation Analysis (CCA) [21] and (squared) Normalized Euclidean Distance (NED) [12]. We also hope that in the future a metric for AI safety is ubiquitously reported as proposed in [19].

4. Benchmarks that require meta-learning

4.1. Motivation

Our goal in this section is to define a benchmark that requires meta-learning (and not only a good embedding) to be solved effectively. The idea is to measure the CCA and (squared) NED of a MAML representation after adaptation for our new benchmark. The hope is that if models trained on this benchmark have a higher CCA than 0.1 (CCA value form previous work [23]) then it is good evidence this new benchmark benefits of meta-learning and can be detected at a higher degree than previous work [23]. For more details of further experiments that be needed to make these inferences conclusive see the future work section in the appendix.

4.2. Synthetic task that requires meta-learning

4.2.1 Overview and Goal

The main idea is to sample functions to be approximated such that the final layer needs little of no adaption but the feature layers require a large amount of adaption. This type of task would forcibly require that the meta-learner learns a representation that requires the feature layer to change by a lot to achieve good meta-test performance. Thus, to perform well, not only would it be good to adapt all the layers at meta-evaluation time but additional performance might be obtained from an initial (meta-learned) representation that can be changed flexibly to perform well on any task. Therefore we believe this type of task is a sufficient for meta-learning to occur. In other words, tasks must not all have the same shared representation to be solved for meta-learning to be useful and detectable.

4.2.2 Definition

We propose the first set of benchmarks to be regression functions (but can be easily extend to classification as briefly explained in the appendix). Therefore one task will be a specific function sampled from a distribution of similar functions. An example task can be seen in figure 1 We will choose the family of functions to be fully connected neural networks (FCNN) with a fixed amount of layers (in our experiments we used a total of 4 fully connected layers). The process to sampled one function is as followings: we will have two pair of parameters, one to sample the parameters for the representation layer and one to sample the parameters for the final layer using a Gaussian distribution. We denote the first generation parameters with $(\mu^{(1)}, \sigma^{(1)})$ and the latter $(\mu^{(2)}, \sigma^{(2)})$. Then each task is sampled as follows:

- Sample the representation parameters $w^{(l)} \sim N(\mu^{(1)}, \sigma^{(1)})$ for each layer $l \in [L-1]$
- Sample the final layer parameters $w^{(L)} \sim N(\mu^{(2)}, \sigma^{(2)})$

The idea is that $\sigma^{(1)} < c \cdot \sigma^{(2)}$ (for some $c \in R$) so that the variance in tasks is due to the representation and therefore adapting the representation layer is necessary. We also hope this property can be exploited by the meta-learning algorithm during meta-training. Note that the scientific challenge of the study is to find the constant c for a benchmark(s) such that the constants allow for the properties we hope to observe to be noticeable Although, in practical the actual value for c might not important.

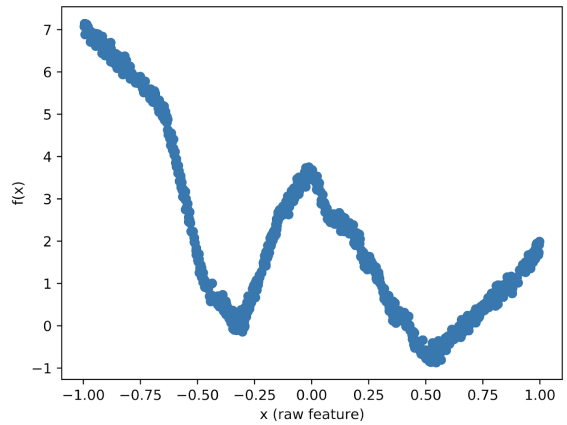

Figure 1. This plot shows an example function regression task.

4.2.3 Results on Benchmark that requires meta-learning

In this section we report the different amounts of CCA and (squared) NED exhibited on different benchmarks we created.

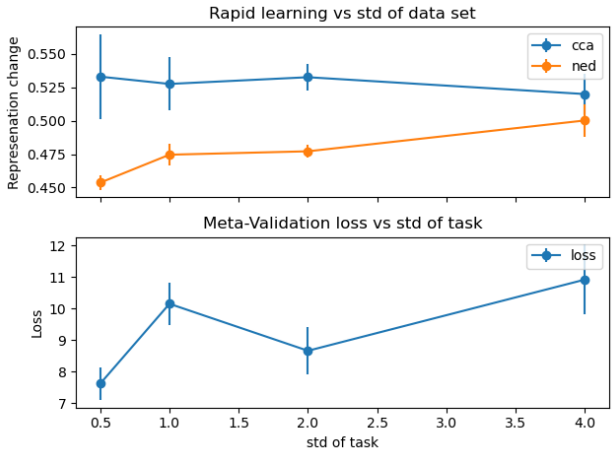

Figure 2. Shows the correlation between degrees of meta-learning (measured with squared NED and CCA) as the standard deviation of the task increases. We also show the meta-validation loss vs the standard deviation of the task. The meta-validation loss increases because as the representation layers are more different the tasks are harder to adapt to resulting in higher losses. The models used for each point in the plot are models selected from early stopping (using the meta-validation MSE loss). The models are the same architecture as the target function (4 layers fully connected neural network) with ReLU activation function.

Figure 2 is the most important plot in this section as it shows that the degree of meta-learning is higher than previous work [23]. We want to emphasise that our value of CCA is around of 0.53 ± 0.2 which is much higher than that of previous work of approximately 0.1 and is statistically significant. Note this higher squared NED/CCA than 0.1 was observed across all of our experiments (over 10 different benchmarks), even with models that had overfitted and were close to converging (e.g. see figure 7). This is suggestive that these tasks require meta-learning. In addition we notice squared NED increases as the standard deviation of the representation layer increases, however, this has to be investigated further as it's unclear if it's a consistent observation. We do believe further experiments when $\sigma^{(1)} < \sigma^{(2)}$ is essential for our results to be conclusive (ideally squared NED/CCA decreases to zero as $\sigma^{(1)} \to 0$).

In addition, figure 6 shows that as the number of inner steps for the MAML meta-learner increases, the NED/CCA does not increase. This suggests that a few number of steps were required to adapt this representation. This might suggest that MAML did find a good representation. However, the representations were trained with MAML with 1 inner step which might mean MAML found an optimal representation requiring 1 step (or few). Further studies are needed to disentangle these competing hypothesis but it is interesting to see the squared NED/CCA remained much higher than previous work [23].

5. Meta-overfitting

In this section we discuss the large gap between meta-train and meta-test/validation loss that we observed - what we term *meta-overfitting*. This meta-generalization gap increases as the models were meta-trained further. We suggest to track degree of meta-learning to diagnose possible causes for meta-overfitting.

5.1. Finite number of tasks

In this work we observed that when the number of tasks (functions) are finite (200 for this work) we consistently observed meta-overfitting as shown in figure 3. This was observed in over $30 - 50$ experiments with a finite number of tasks.

In addition, meta-overfitting was observed in real few-shot learning benchmarks as shown in figure 5 with mini-Imagenet. With the standard Pytoch ResNet18 one can observe a meta-generalization gap of about 30% and on a state of the art ResNet12 [24] a meta-generalization gap of about 20%.

5.2. Infinite number of tasks

We believe it is important to highlight that meta-overfitting was not observed when the number of tasks is unbounded as shown in figure 4. This suggests that when the number of tasks are unbounded but sampled from a related set of tasks, meta-learning algorithms can leverage their power to adapt without meta-overfitting.

To measure the amount of meta-learning we also did the NED/CCA analysis as in section 3 and observed a value of

about 0.65 (in a statistically significant way). This implies that the degree of meta-learning is higher when the number of tasks is unbounded. We predict that a model without meta-learning (e.g. without MAML) would perform at chance and thus predict meta-learning is necessary to solve such a benchmark (in fact it is unclear how to train it with an unbounded number of tasks).

5.3. How do we go forward if meta-overfitting exists?

We believe that the results in this section shows that the current benchmarks need to be changed for them to be useful benchmarks for meta-learning. As an initial suggestion, we suggest that the number of image classes increases by considerably until the probability that 4 classes are common between any two N-way,K-shot tasks is small. We believe this is a good suggestion because although $Choose(64, 5) = 7,624,512$ might seem like a large number of tasks, there being only 64 different classes implies there is a high probability that two tasks will have a substantial amount of sharing of image classes. Making this argument precise and coming up with a good definition for task similarity for real few-shot learning vision tasks will be important future work.

We also want to argue that although it might seem useful to have a finite number of tasks and design meta-algorithms that don't meta-overfit, we believe a more promising direction is to identify conditions were meta-algorithms are needed especially because humans are able to learn from an infinite pool of concepts and use them in rich ways [16, 15]. Therefore although meta-overfitting is a interesting concept to study, we believe also designing tasks that mimic more closely were humans require meta-learning might be a more promising path for achieving machines that reason like humans (and quantify general intelligence in the process [19]).

6. Effects of more meta-adaptation

In this section we report the results obtained when we increase the number of inner steps in MAML. In particular we track the the meta-loss and degree of meta-learning (measured via squared NED and CCA).

Figure 6 shows that as the number of inner steps increases both meta-learning and performance become saturated. This suggest that MAML seems to be robust to meta-overfitting as the number of inner steps increases. Understanding why this happens would be interesting, although, other work shows that deep neural networks have been observed to be resilient to overfitting with SGD [22] implying this might not be entirely surprising.

In addition, the loss of a 4-layered fully connected neural network model meta-trained with MAML with *no adaptation* has an error of about 22.0 - which is much higher than the models adapted with further inner steps. Further steps

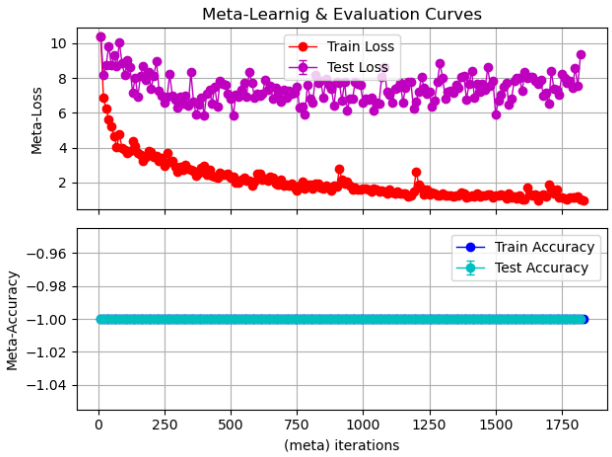

Figure 3. Shows meta-overfitting when the number of tasks (functions) is finite at 200 tasks. The curve is the learning curve for a 4-layered fully connected neural network trained with MAML [10] (using episodes [10]). It is particularly interesting to note that the meta-validation loss increases as the meta-iterations increases while the meta-train loss decreases. We use a (large) meta-batch size of 75 for meta-evaluation and meta-train to decrease the noise during training. This is a regression task so the blue curves can be ignored.

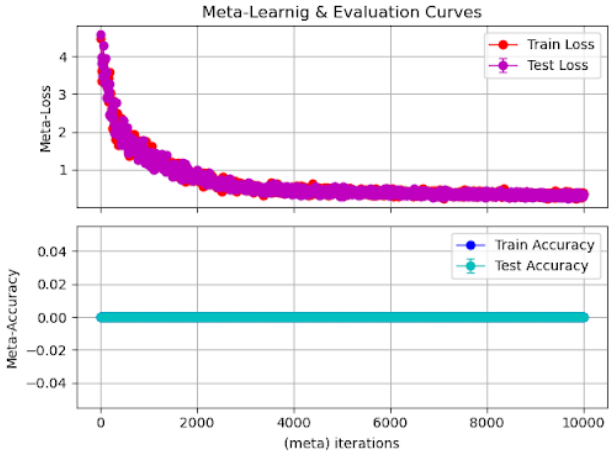

Figure 4. Shows that meta-overfitting does not occur and in fact perfect meta-generalization occurs when the number of tasks (functions) is unbounded when training with MAML. In other words the meta-train and meta-validation error are indistinguishable and decrease together as the meta-iterations increases. This benchmark was created using the sinusoidal task suggested as in [10]. For better comparison with previous experiments using our synthetic tasks it will be interesting to repeat those experiments but one of the benchmarks we suggest. This is a regression task so the blue curves can be ignored.

decreases the loss to about 9.0 providing further support that the meta-learner is crucial to perform well on this task. Note however, that adaption of the final layer to convergence as

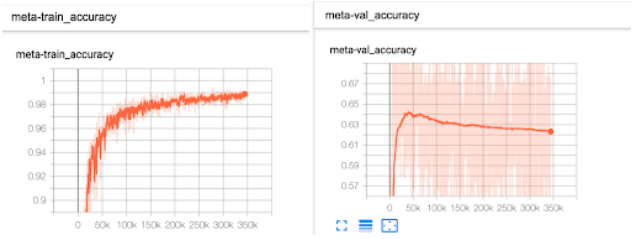

Figure 5. Shows that meta-overfitting is a real phenomenon even in real few-shot learning benchmarks. This is the standard Pytorch ResNet18 trained to convergence on the mini-Imagenet benchmark. Note that the noise of the meta-accuracy is due to having a meta-batch size of 1 to speed up experiments. We consistently saw that increases in meta-batch size lead to decreases noise in the learning curves but we didn't re-run these experiments since it can take up to a week to reproduce on a Quadro RTX 6000 using Pytorch libraries torchmeta and higher [8, 11].

in [24] would be an interesting comparison since that would compare the loss of a model using only a good representation and a model adapted with all it's layers.

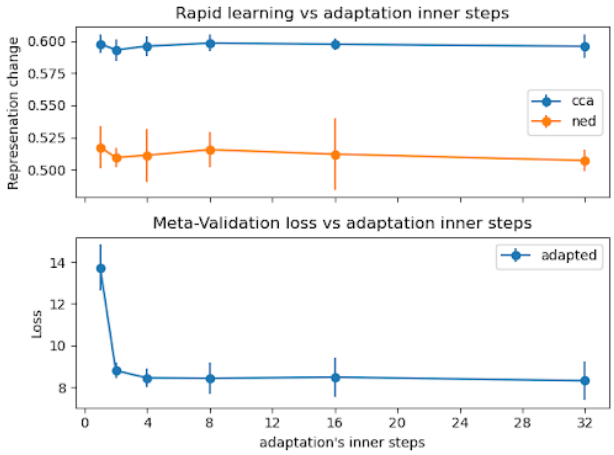

Figure 6. Shows the correlation between degree of meta-learning (measured with squared NED and CCA) as the number of inner steps in MAML increases. The models used are models selected using early stopping using the meta-validation set e.g. models with the minimum meta-validation MSE loss. This is the model used for figure 3.

7. Discussion

We believe that these results are exciting. We show that the synthetic tasks we suggested consistently showed higher degree of meta-learning measured by two different metrics. However, it will be important to investigate why the degree of meta-learning does not seem to increase as the task difficulty is increased. In addition, further control experiments where the meta-learning representation used for adaptation is obtained from supervised pre-trained could reveal if MAML is truly doing rapid learning compared to a representation that has no such prior built into it (but crucially that also performs well). We also suggest a more in depth analysis of the regime for $\sigma^{(1)} < 1.0$ where we expect the degree of meta-learning to eventually be zero (since eventually all synthetic tasks should be using approximately the same representation).

It would also be interesting to reproduce these experiments with few-shot learning benchmarks with real images. However, a similar role to increasing $\sigma^{(1)}$ would be required - something that measures similarity of two N-way, K-shot tasks. The best similarity measure we have is the probability that two N-way, K-shot tasks share at least 4 image classes. A more sophisticated development of measures quantifying task similarity for N-way, K-shot classification tasks would be fascinating and crucial.

We believe the meta-overfitting phenomenon is the most consistent result we discovered and was reproduced with over $30 - 50$ experiments. This was also observed with experiments done with pytorch ResNet models and ResNet models from [24] on mini-Imagenet. This is a real phenomenon and the synthetic benchmarks reveal a strong correlation with the number of tasks available. Interesting experiments could be to plot the generalization gap (with a synthetic classification task) and demonstrate it decreases as the number of tasks decreases (although we have the case when the number of tasks is infinite and meta-overfitting is not observed). In addition, we believe verifying these findings with many few-shot learning tasks would be interesting. One possibility is taking a union of many vision classification tasks and re-scaling all images to be of the size of mini-Imagenet. Then we would redo the experiment 4 but with this benchmark composed of real images.

For discussion of future work see the future work section in the appendix.

8. Conclusion

From these results we argue that the best way to go forward is to identify the situation were meta-learning is needed. We believe its better to design better benchmarks that inherently require meta-learning. In particular we believe efforts to deliberately identify what is needed to achieve human level intelligence and tracking it in few-shot leanring benchmarks is a better path than trying to fix the meta-overfitting problem without a deliberate goal in mind.

The experiments show that when the benchmarks have properties that require meta-learning (e.g. number of concepts are unbounded or tasks require different representations) meta-learning is detectable and even has perfect meta-generalization. We also showed more adaptation is not always helpful but also doesn't meta-overfit. We also showed that it is important to track and quantify degree of meta-learning and it's relation to absolute performance.

We hope these results motivate future work to make more deliberate efforts for designing benchmarks that require human level meta-learning.

9. Broader Impact

9.1. Quantifying general intelligence through meta-learning

There is valuable efforts that try to make benchmarks which require higher level cognition e.g. [26]. An example of work that tries to quantify AGI and proposes a benchmark is [6]. We believe the second approach is likely to have more impact in the long run because it also deliberately quantifies general intelligence. We believe that suggesting benchmarks without clearly specifying the long term goal or measuring the metric we are trying to optimize is a suboptimal approach. However, we do believe grounding benchmarks on tasks that humans are able to perform is a good idea but suggest to augment these proposals with metrics and explicit discussions of general intelligence.

Another approach we believe has high potential is program synthesis [4] and theorem proving [20, 7] because humans create higher abstractions that are composed and resue thus suggesting to meta-learning might be taking place. We believe that higher level cognition tasks are a challenging to assess meta-learning algorithms.

9.2. Quantifying AI safety

We also believe quantifying and tracking metrics for AI safety as early as possible is crucial. Few-shot learning is likely one of simplest - and arguably the atomic buildings blocks for general intelligence. We believe AI safety could be enriched if research community deliberately tracks, discusses and report it in all it's research - especially in meta-learning research. For a brief proposal see [19].

9.3. Summary

We hope that this discussion inspires the AI community - but especially the meta-learning research community - to always report their progress using, what we will call the "the big three" [19]:

1. the score for absolute performance (to ensure usefulness)
2. the score for general skill acquisition (to ensure flexibility and general intelligence)
3. the AI safety score (to ensure positive outcome).

10. Acknowledgements

We'd like to *especially* acknowledge the fantastic discussion that lead to this work with Professor Sanmi Koyejo. In addition, we'd like to acknowledge professor Y. Wang for discussions and hosting such a insightful course as CS 598 at UIUC. For the short video presentation of this work see [18] We'd like to thank Intel for providing our team with access to their Academic Cluster Environment (ACE). The compute resources and support from their staff for their ACE cluster were essential to the successful completion of our project. In addition, this work utilizes resources supported by the National Science Foundation's Major Research Instrumentation program, grant 1725729, as well as the University of Illinois at Urbana-Champaign [14]. We would also like to acknowledge the CS 598 staff and students for awarding this work the best class project for the graduate course CS 598 Learning to Learn (on December 2020) [2]. Although we are not releasing our code (or data yet), we'd like to acknowledge the work and authors of torchmeta and higher [8, 11] for making their code available and answering ours questions in their project's repository.

11. Appendix

11.1. Analysis of meta-overfitted models

In this section we discuss the properties of meta-overfitted models. We do this both as a function of the standard deviation of the representation layer $\sigma^{(1)}$ (in figure 7) and the number inner steps (figure 8).

Figure 8 shows how ReLU models were only trained until 5 inner steps, as any further made the errors explode in magnitude (errors above 10^{21}). Surprisingly however, the degree of meta-learning (as measured with squared NED and CCA) remained higher than 0.1 from previous work [23]. However, note how the degree of meta-learning had a high variance and was less stable 7 than our other plots that used models selected form early stopping.

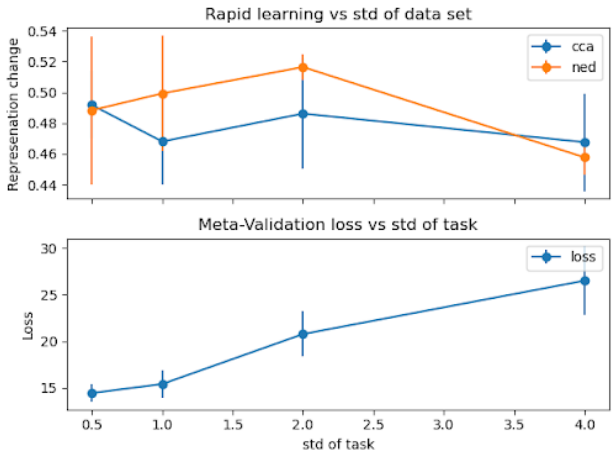

Figure 7. Shows how ReLU overfitted models show more noisy CCA/NED values. Note it is still above 0.1 compared to previous work [23].

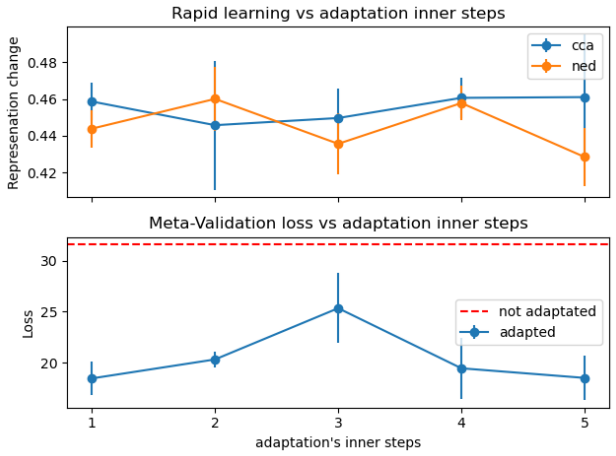

Figure 8. Shows that we could only train MAML up to 5 inner steps before it exploded to error with magnitude above 10^{21}. This shows meta-overfitted models with ReLU are unstable.

11.2. Analysis of Architectures with Sigmoid function

One hypothesis is that sigmoid models might allow better meta-learning when using MAML [10]. This is because during the meta-training phase the backward pass through the gradient operation would be present rather than being zero (since ReLU models are locally linear [10]).

The conclusion seems to be that sigmoid neural networks do exhibit a higher degree of meta-learning since $0.6 > 0.5$ (in a statistically significant way) as can bee seen in figure 5. Note that we did do additional experiments on mini-Imagenet that are yet to be analyzed.

11.3. Role of Backbone on meta-accuracy

In this section we describe the relation of the depth of a Pytorch ResNet model with the meta-test accuracy. The motivation for these experiments is that if we can close the gap on mini-Imagenet to over 90% by only increasing the back bone depth then this would provide strong evidence that such benchmarks really only need a good embedding. However, we discovered that for the ResNets used in [24] it seems that accuracy saturates at 80% (results not shown in paper) but when using the Pytorch models we see meta-overfitting and decreasing meta-test error 9. This suggests that even this simple scenario of few-shot learning still has still space for meta-learning to be a solution.

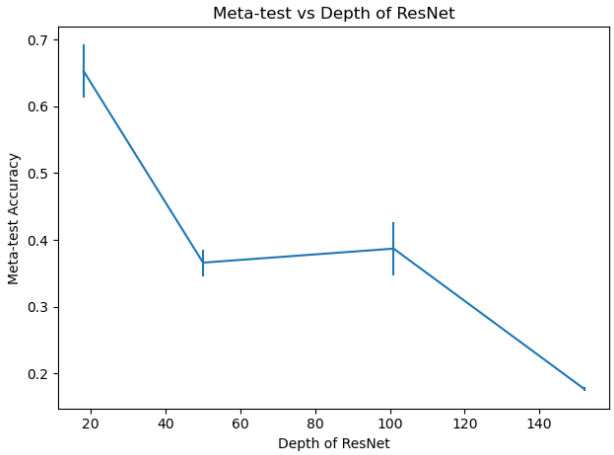

Figure 9. Shows that as the backbone of the Pytorch ResNets increases to 152 the meta-accuracy on mini-Imagenet decreases. These models were trained with supervised union training is in [24]. The meta-adaption algorithm used logistic regression and was adapted to convergence on the final layer as in [24]. When using the Pytorch ResNet models instead of the special ResNets designed for mini-imagenet [24] we observe see that the meta-accuracy decreases 9.

11.4. Analysis of meta-learned initialization

In this section we an experiment where fix the ResNet18 meta-learned initialization and use the adaption that only adapts the final later as in [24]. The results in figure 10 are mixed but it is interesting to note MAML with no inner steps performs worse than a random neural network. This result is interesting because this is very similar to supervised pre-raining in that no meta-learner is present during training but instead of seeing all 64 images it sees 5 randomly (but uses no meta-learner). We would have expected that the initialization obtained would have been equivalent to one with supervised pre-training. Since they are not it shows a MAML is at the very least capable of learning a representation that is invariant to concept permutation.

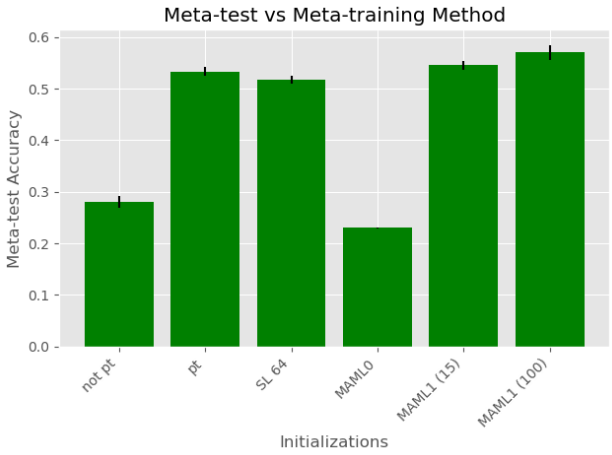

Figure 10. Shows relation of meta-test accuracy with models with a different meta-learned initialization. PT stands for Pre-trained on Imagenet. Not Pt stands for a random model. All models are ResNet18s from PyTorch. SL 64 stands for supervised union pre-training on mini-imagenet using all 64 labels during meta-training. MAML0 stands for only using episodic meta-training (i.e. MAML with zero inner steps). MAML1 (15) and MAML1 (100) stand for training using MAML with a query set of size 15 to 100. The meta-adaptation is the same as in [24] (training logistic regression in the final layer to convergence).

11.5. Training with zero number of inner steps

We believe it is an interesting observation that MAML with 0 inner steps (MAML0) (i.e. only using episodic meta-training) resulted in very different meta-learned initialization compared to MAML with 1 inner step (MAML1) on mini-Imagenet. Previous work observed that supervised pre-training [24] with all 64 images during meta-training results in a strong baseline. With this in mind it is natural to ask: what is the difference between seeing all 64 images during supervised pre-training or seeing only 5 using episodic training? With this in mind we trained MAML0 and obtained a model that performs at chance. Figure 11 compares MAML0 with MAML1 to show that MAML0 obtains a model that has a very high meta-training loss. Additionally, figure 10 shows such an initialization performed even worse than random. This is surprising but it seems that meta-learned initialization with MAML1 learn at least a model that is invariant to permutation of the order of the classes. Unfortunately, this result seems to only be reproducible in classification since training MAML0 in a synthetic regression task did converge to have model with low meta-train loss 12. This suggests future studies would be interesting to disentangle the casual factors.

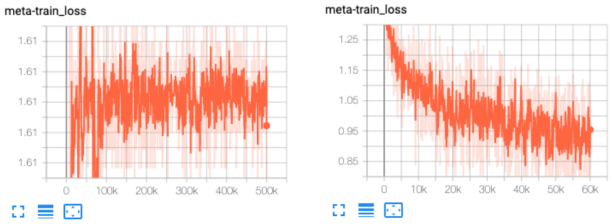

Figure 11. Compares MAML0 (only episodic training) vs MAML1 (MAML with 1 inner step). MAML0 remains close to chance with a high loss while MAML1 converges. This suggests MAML0 is not equivalent to supervised pre-training and that MAML1 does learns a representation that is invariant to class order permutation.

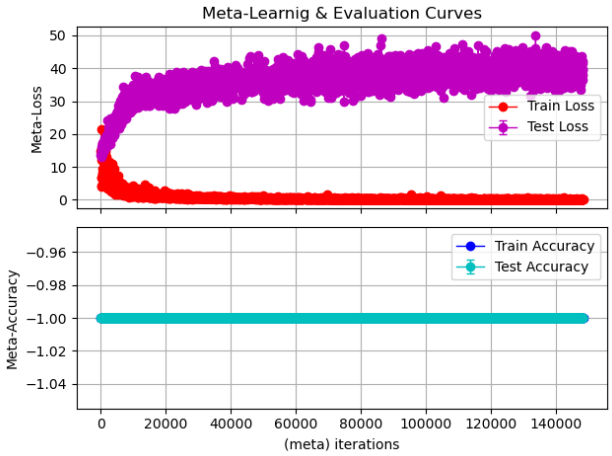

Figure 12. Shows MAML0 (only episodic training) getting zero meta-train loss (red curve) for a synthetic regression task. This suggests that meta-learning in regression and classification might not be enitrely equivalent. Note that meta-overfitting is still observed (purple curve). This is a regression task so the blue curves can be ignored.

11.6. Tips and tricks for episodic meta-training

From our experiments we suggest the following (when episodic meta-training [10]):

1. Use a large number of query examples e.g. greater than popular 15 (since they often speeds up convergence of the meta-learning algorithm).

2. A large meta-batch size (since it's important to be able to have a low level of noise when tracking the meta-validation error/loss for doing early stopping). We found empirically for $75-100$ tasks to be a good meta-batch size.

3. Episodic training as suggested in [10] is expensive and takes at least a week to train on mini-Imagenet on a Quadro RTX 6000 using torchmeta and higher [8, 11], so these suggestions are important.

11.7. Future work

11.7.1 Summary

1. We need to compare the amount of rapid learning (measured via CCA) more carefully in the case where $\sigma^{(1)} < \sigma^{(2)}$ bellow the current 0.5 (since this case is where a fixed embedding is enough to solve a task sampled form our synthetic benchmarks).

2. We need to compare the following inequality: $CCA(A(f_{maml}), f_{maml}) > CCA(A(f_{sl}), f_{sl})$. If the inequality holds then it's true that the rapid learning of f_{maml} is larger than that of f_{sl}, since showed a larger representation change.

3. Defining a synthetic benchmark that is a classification problem that also requires meta-learning (or rapid learning with MAML).

4. We also hope to construct a (real) benchmark from images that requires meta-learning. Formally, we propose a good start would be a benchmark were the probability of two task having the same class be small, otherwise we are more likely to see overfitting. Alternatively, a benchmark that requires the tasks to be different by at least requiring a different representation. We believe compositionality is an ideal benchmark since this would allow sophisticated re-use of lower level representations and simultaneously have an unbounded number of tasks. Humans are able to richly and flexibly cope with both. Additionally, it would be interesting to be able to quantify the distance between two different N-way, K-shot tasks to make these ideas more rigorous.

5. An interesting benchmark with a large number of classes with real images is taking the union of many vision classification tasks and re-scaling all images to be of the size of mini-Imagenet.

6. Plotting the meta-generalization gap (with a synthetic classification task) and demonstrate it decreases as the number of tasks increases would be interesting (note however we already have the limiting case when the number of tasks is unbounded and the meta-generalization gap is zero).

7. An interesting experiment would be to train a deep neural network with the episodic training (but without the MAML inner loop) but have an unbounded number of tasks and see if the test error keeps increases (or stays at chance as observed when this is done with mini-Imagent 11).

8. An interesting hypothesis to investigate is if meta-learning algorithms get representation that are optimal for their respective meta-learner (or adaptation rule). If this is true it means methods like [24] can be improved by making the entire pipeline differential and learning it end-to-end [17].

9. Test meta-learning algorithms in domains were higher level cognition is required and thus compositionality is essential e.g. program synthesis [4] and theorem proving [20, 7].

10. Deliberately design an AI safety measure as proposed in [19] for few-shot learning.

11. Propose a robust and widely accepted general intelligence metric that is applicable for many environments and tasks - in particular for few-shot learning. We believe deliberate efforts for general intelligence are important.

11.7.2 Proposal on Synthetic classification task that possibly require meta-learning

Synthetic tasks that use classification instead of regression are not hard to define. Two possible alternatives are: 1) a mixture of Gaussians but the standard deviation controls the radius of limit of how far the classes can be from each other 2) another option is the similar as with a mixture of Gaussians but have the (vector) samples be weights of a Neural Networks (so that the goal is to identify from which Neural Network data is coming from)

References

[1] CS 330 Deep Multi-Task and Meta Learning. 1

[2] CS598: Learning to Learn. 1, 6

[3] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A Closer Look at Few-shot Classification. *7th International Conference on Learning Representations, ICLR 2019*, 2019. 1

[4] Xinyun Chen, Chen Liang, Adams Wei Yu, Dawn Song, and Denny Zhou. Compositional generalization via neural-symbolic stack machines. (NeurIPS), 2020. 6, 9

[5] Yinbo Chen, Xiaolong Wang, Zhuang Liu, Huijuan Xu, and Trevor Darrell. A New Meta-Baseline for Few-Shot Learning. Technical report. 1

[6] François Chollet. On the Measure of Intelligence. 2019. 6

[7] Maxwell Crouse, Ibrahim Abdelaziz, Bassem Makni, Spencer Whitehead, Cristina Cornelio, Pavan Kapanipathi, Kavitha Srinivas, Veronika Thost, Michael Witbrock, and Achille Fokoue. A Deep Reinforcement Learning Approach to First-Order Logic Theorem Proving. Technical report. 6, 9

[8] Tristan Deleu, Tobias Würfl, Mandana Samiei, Joseph Paul Cohen, and Yoshua Bengio. Torchmeta: A Meta-Learning library for PyTorch, 2019. Available at: https://github.com/tristandeleu/pytorch-meta. 5, 6, 9

[9] Guneet S. Dhillon, Pratik Chaudhari, Avinash Ravichandran, and Stefano Soatto. A Baseline for Few-Shot Image Classification. 2019. 1

[10] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. 2017. 1, 4, 7, 8, 9

[11] Edward Grefenstette, Brandon Amos, Denis Yarats, Phu Mon Htut, Artem Molchanov, Franziska Meier, Douwe Kiela, Kyunghyun Cho, and Soumith Chintala. Generalized inner loop meta-learning. *arXiv preprint arXiv:1910.01727*, 2019. 5, 6, 9

[12] PTDS (https://stats.stackexchange.com/users/68112/ptds). Definition of normalized euclidean distance. Cross Validated. URL:https://stats.stackexchange.com/q/136232 (version: 2020-12-12). 2

[13] Shaoli Huang and Dacheng Tao. All you need is a good representation: A multi-level and classifier-centric representation for few-shot learning. 2019. 1

[14] Volodymyr Kindratenko, Dawei Mu, Yan Zhan, John Maloney, Sayed Hadi Hashemi, Benjamin Rabe, Ke Xu, Roy Campbell, Jian Peng, and William Gropp. HAL: Computer System for Scalable Deep Learning. In *ACM International Conference Proceeding Series*. Association for Computing Machinery, 2020. 6

[15] Brenden M. Lake, Ruslan Salakhutdinov, and Joshua B. Tenenbaum. The Omniglot challenge: a 3-year progress report, 2019. 4

[16] Brenden M. Lake, Tomer D. Ullman, Joshua B. Tenenbaum, and Samuel J. Gershman. Building Machines That Learn and Think Like People. *Behavioral and Brain Sciences*, 40, 2016. 4

[17] Brando Miranda. DiMO : Differentiable Model Optimization and metaDiMO. *Illinois Digital Environment for Access to Learning and Scholarship (IDEALS)*, 2019. 9

[18] Brando Miranda. An empirical study of the properties of meta-learning - presentation. *Illinois Digital Environment for Access to Learning and Scholarship (IDEALS)*, 2020. 1, 6

[19] Brando Miranda. Establishing the foundations of Meta-learning - a Proposal. *Illinois Digital Environment for Access to Learning and Scholarship (IDEALS)*, 2020. 2, 4, 6, 9

[20] Brando Miranda. Sketching: a Cognitively inspired Compositional Theorem Prover that Learns to Prove - a Proposal. *Illinois Digital Environment for Access to Learning and Scholarship (IDEALS)*, 2020. 6, 9

[21] Ari S Morcos, ‡ Deepmind, Maithra Raghu, Samy Bengio, and Google Brain. Insights on representational similarity in neural networks with canonical correlation. Technical report. 2

[22] Tomaso Poggio, Kenji Kawaguchi, Qianli Liao, Brando Miranda, Lorenzo Rosasco, Xavier Boix, Jack Hidary, and Hrushikesh Mhaskar. Theory of Deep Learning III: explaining the non-overfitting puzzle. 2017. 4

[23] Aniruddh Raghu, Maithra Raghu, Samy Bengio, and Google Brain. Rapid Learning or Feature Reuse? Towards Understanding the Effectiveness of MAML. Technical report. 1, 2, 3, 7

[24] Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B. Tenenbaum, and Phillip Isola. Rethinking Few-Shot Image Classification: a Good Embedding Is All You Need?, 2020. 1, 3, 5, 7, 8, 9

[25] Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Utku Evci, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Swersky, Pierre-Antoine Manzagol, and Hugo Larochelle. Meta-Dataset: A Dataset of Datasets for Learning to Learn from Few Examples. 2019. 2

[26] Ramakrishna Vedantam, Arthur Szlam, Maximilian Nickel, Ari Morcos, and Brenden Lake. CURI: A Benchmark for Productive Concept Learning Under Uncertainty. Technical report. 6