Robust Mean Estimation for Few-Shot Classification Meta Learning

Nicholas Petrocelli
Department of Computer Science
Brown Universiry
Providence, RI 02912
nicholas_petrocelli@brown.edu

Abstract

A key innovation of the past 5 years in few-shot image classification was the development of nearest-centroid based meta-learning algorithms. We examine the robustness of one such algorithm, Meta-Baseline [Chen et al., 2021], and find that it is vulnerable to random corruption of 20% of the support examples at inference time when evaluated on MiniImageNet [Vinyals et al., 2016] and TieredImageNet [Triantafillou et al., 2018] in the 5-shot and 20-shot environments. We find that classifying using the Euclidean distance is particularly vulnerable, with an accuracy drop of 37.7% on average across datasets and shots compared to the cosine similarity's 14.5%. We attempt to improve Meta-Baseline's robustness by employing Cheng et al. [2020]'s robust mean estimation algorithm with Euclidean distance classification, and find that this reduces this accuracy loss by 34 percentage points on average for MiniImageNet and 9 percentage points for TieredImageNet. We conclude that this is an effective proof of concept for this technique and offer multiple straightforward avenues for future work to improve upon our result. The codebase for this project can be found at https://github.com/nickpetrocelli/meta-baseline-extension.

1 Introduction

One of the most studied problems in machine learning is that of *image classification*, where a model is trained to categorize input images into particular classes based on their content. Traditionally, training a model to classify images requires large amounts of annotated class-image pairs; for example, the current best performing model on the ImageNet [Deng et al., 2009], CoCa [Yu et al., 2022], trains on the JFT-3B dataset [Zhai et al., 2022], which contains almost 3 billion such pairs. In contrast, humans can learn to classify many categories of images after viewing just a few examples. This contrast inspired the field of *few-shot learning*, where models are designed to classify images after seeing only a few examples of each potential class. Today, many high-performing few-shot classification algorithms make use of a *meta-learning* paradigm, where the model "learns to learn" by transferring knowledge from training tasks to learn a new task at inference time. Specifically, a popular class of algorithms first learns an image embedding space by learning to classify a set of base classes at training time, then classifies novel classes at inference time by computing the centroid of the few support examples for each class in the embedding space and classifying query images based on the nearest centroid. In this project, we specifically study Meta-Baseline [Chen et al., 2021], a recent algorithm that reaches competitive performance with modern few-shot baselines by applying this paradigm.

A potential weakness of Meta-Baseline is that it relies heavily on the support examples for each category being representative of their class. This makes Meta-Baseline vulnerable to corruption of

Preprint. Under review.

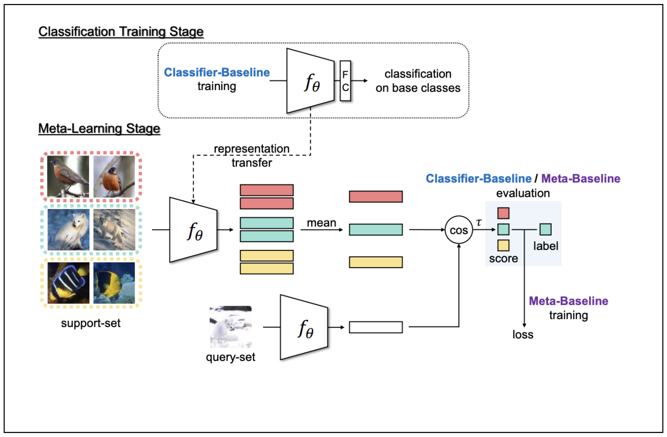

Figure 1: Figure 1 from Chen et al. [2021], demonstrating architecture of Meta-Baseline and 2-phase training procedure.

the support examples, as any corrupted support example could arbitrarily alter the centroid. Here, we examine one potential avenue to make Meta-Baseline more robust against such attacks: calculating the class centroids via high-dimensional robust mean estimation. Specifically, we use Cheng et al. [2020]'s algorithm, which uses gradient descent to calculate an estimate of the mean in a relatively straightforward manner.

We test this combination of algorithms on MiniImageNet [Vinyals et al., 2016] and TieredImageNet [Triantafillou et al., 2018] in the 5-shot and 20-shot settings. We find that applying Cheng et al. [2020]'s algorithm does significantly improve Meta-Baseline's robustness when 20% of the support examples are corrupted at test time. We find that when classifying using the Euclidean distance, the loss in accuracy between the un-corrupted and corrupted settings is reduced by 34 percentage points on average for MiniImageNet and 9 percentage points on average for TieredImageNet. We also find that robust mean estimation does not reduce Meta-Baseline's performance in un-corrupted settings on either dataset. However, we also note substantial runtime issues with the robust mean estimation algorithm, and detail clear areas in which future work can improve upon our results to produce a more practical form of this algorithm.

2 Background

2.1 Few-shot image classification

In traditional supervised image recognition, a model is trained on some set of N in-domain classes, with access to a large number of examples during training time. The model is then evaluated on the same set of N classes at testing time using a testing set that consists of novel examples of those classes.

In contrast, the few-shot image recognition task pursued in this paper and its predecessors places additional constraints on the testing environment. Here, a model is still trained on some set of N in-domain classes, but at testing time is evaluated on a set of query examples from n unseen classes after seeing only k examples of each class, with q query examples per class. This is referred to as "n-way, k-shot" learning.

2.2 Meta-Baseline and distance metrics

In this paper and its preceding work, a model is trained to perform the few-shot classification task using a *meta-learning* paradigm. Meta-learning is defined by Lemke et al. [2015] as any model that "...include[s] a learning subsystem, which adapts with experience...extracted in a previous learning episode on a single dataset, and/or from different domains or problems." The specific meta-learning method that we extend here is *Meta-Baseline* from Chen et al. [2021]. Meta-Baseline relies on a simple 2-phase learning process (Figure 1) that leads to good few-shot generalization in practice:

1. First, a backbone classifier is trained on the N in-domain classes using standard supervised learning procedures, thereby learning an embedding space suitable for image classification.
2. Second, the encoder from this classifier is transferred to a meta-learning model, which fine-tunes the embedding space by repeatedly sampling n classes, $k \times n$ support examples, and $q \times n$ query examples from the training data and thereby simulating the few-shot learning task. Each one of these simulations is called a *training episode*.

In order to classify each query example during a few-shot episode, the Meta-Baseline model first computes an embedding for each of the support examples for each class and then computes a centroid c_i for each class $i \in [1 \dots n]$ [Chen et al., 2021]. Then, to perform classification on some query example, the model simply computes an embedding for the example and outputs the class of the nearest centroid. Importantly, Chen et al. [2021] defines a centroid as a point in the embedding space equidistant from all k support examples according to some distance metric. Chen et al. [2021] utilizes the cosine similarity metric for this purpose. However, other methods that employ nearest-centroid classification sometimes use different metrics, such as Prototypical Networks [Snell et al., 2017] which utilizes the Euclidean distance. In particular, Snell et al. [2017] also proved that performing nearest-centroid classification using the Euclidean distance is equivalent to performing mixture-density estimation using spherical gaussians as the densities, where the estimated mean of each density is the Euclidean mean of the support vectors. Thus, for Euclidean distance nearest-centroid classification to be robust, we require some means of obtaining a good estimate for the centroid in terms of ℓ_2 norm in a corrupted-data setting.

2.3 Robust Mean Estimation

2.3.1 ϵ-corruption

Following earlier works in the field of robust statistics such as Huber [1964] and Diakonikolas et al. [2019], we say our model is *robust* if it can deliver accurate results in an ϵ-corrupted setting, i.e. when some fraction $0 < \epsilon < 0.5$ of the support examples have been removed by some adversary and replaced with arbitrary points. More specifically, we wish for our models to minimize the drop in accuracy when evaluated in an ϵ-corrupted setting versus an uncorrupted one, and maintain comparable accuracy to base Meta-Baseline when evaluated in an uncorrupted setting.

2.3.2 Robust Mean Estimation using Projected Gradient Descent

For an algorithm to compute this estimate, we chose Cheng et al. [2020]'s gradient descent-based algorithm. Like previous works in high-dimensional robust mean estimation such as Cheng et al. [2019], this algorithm provides excellent error guarantees - specifically, that the difference in ℓ_2 norm between the output estimate and the mean of the uncorrupted sample set is $O(\epsilon\sqrt{\log(1/\epsilon)})$ when the uncorrupted set of samples is at least $N = \tilde{\Omega}(d/\epsilon^2)$ in size and the number of iterations is $\tilde{O}(Nd^3/\epsilon)$, where d is the dimensionality of the sample vectors [Cheng et al., 2020]. However, the main advantage of this algorithm is that it is much more straightforward to implement than previous approaches, as it does not require a nearly-linear time Semi-Definite Program solver as in Cheng et al. [2019] nor any heuristics as in Diakonikolas et al. [2017].

Despite these advantages, it must be made clear that in the few-shot setting, we necessarily lack the number of samples needed for the error guarantees in Cheng et al. [2020] to take effect with high-dimensional data ($d = 512$ in our case). However, while we may not have provable guarantees, there are two intuitions that might indicate that the algorithm will be nonetheless effective:

1. One reason why a large number of samples is needed is so that the covariance of the uncorrupted samples resembles that of the underlying data distribution [Cheng et al., 2019]. However, because we are specifically fine-tuning the underlying embedding space on the few-shot classification task, the covariance of the sample embeddings may be well behaved even without a large number of samples.
2. The algorithm in Cheng et al. [2020] assumes that the underlying data distribution from which the samples are drawn is a spherical Gaussian with unknown mean; however, this assumption is already implicitly made by the act of performing nearest-centroid classification using the ℓ_2 norm, as Snell et al. [2017] proved. Thus, utilizing the algorithm in Cheng et al. [2020] does not introduce any additional assumptions on the sample distribution.

3 Methods

3.1 Datasets and backbone

Here, we train and evaluate the extended Meta-Baseline on two datasets: MiniImageNet, from Vinyals et al. [2016], and TieredImageNet, from Triantafillou et al. [2018], both of which are subsets of ImageNet [Deng et al., 2009]. Here, we use the same training and testing class splits as in Chen et al. [2021]. While both datasets are subsets of ImageNet, the TieredImageNet presents a more difficult task for the model, both because it contains more classes to sample from (608 vs. 100 for MiniImageNet [Triantafillou et al., 2018]), but also because the split ensures that the training and testing classes are from distinct semantic categories [Triantafillou et al., 2018]. For each dataset, we train a ResNet-12 model [He et al., 2016] to act as the backbone classifier and provide the transferred embedding space for meta-learning. Training hyperparameters for these classifiers are identical to those in Chen et al. [2021].

Chen et al. [2021] also trains and evaluates Meta-Baseline on two additional datasets: Imagenet-800, a randomly split 800-base-class subset of ImageNet of their own creation, and Meta-Dataset from Triantafillou et al. [2019]. We did not train or evaluate on these datasets, however, due to a lack of the necessary compute to train the ResNet-18 and ResNet-50 [He et al., 2016] backbones used by Chen et al. [2021] on these datasets.

3.2 Experimental settings

3.2.1 Few-shot settings

Here, as we are evaluating a new method of finding the class centroids when multiple support examples are present for each class, we evaluate the model in the 5-shot and 20-shot settings, differing from Chen et al. [2021]'s 1- and 5-shot settings. We chose to run experiments with an increased number of samples in order to determine if there is a difference in effectiveness of the algorithm between differing sample sizes that are each still below the $\tilde{\Omega}(d/\epsilon^2)$ derived by Cheng et al. [2020].

3.2.2 Corruption setting

Here, we evaluate the robustness of our model at testing time only, and therefore assume that the training data has not been ϵ-corrupted. We do this both because it is standard for ImageNet and its derivatives [Deng et al., 2009] to be hand-annotated and therefore would be extremely unlikely to be attacked by an adversary, and also because running the robust mean estimation algorithm substantially slowed evaluation time, so running it during training was not practical with our available time and compute (discussed further in Section 6). When evaluating for robustness, we corrupt 20% of the support set for each class (i.e. $\epsilon = 0.2$) by adding a Gaussian noise tensor to each corrupted sample, with a standard deviation of 20 for each color value in each pixel. We do this in order to simulate random noisy disruptions to the input images such as dropped and corrupted pixels. These disruptions have been shown to significantly reduce the performance of standard classification models such as ConvNets [Momeny et al., 2021]. While noising the data in this way is potentially more extreme than real-world noise, we chose this method as it was straightforward to implement and sufficient to create enough corruption to affect the efficacy of Meta-Baseline.

3.3 Application of Robust Mean Estimation

As we restrict the corruption setting to only the support examples for each class, this is where we apply the algorithm from Cheng et al. [2020]. In each evaluation run where we are evaluating the algorithm's effectiveness, we run the algorithm for 10 iterations in order to calculate the centroid for each class. We use 10 iterations as this was the suggested starting point from Cheng et al. [2020]'s implementation of the algorithm; unfortunately, we did not have time to perform a hyperparameter search for the most effective number of iterations, as we discuss further in section 6.

In each experiment where we performed robust mean estimation, we performed nearest-centroid classification of the query examples using the euclidean distance, rather than the cosine similarity. While Chen et al. [2021] found that the Euclidean distance is less effective than the cosine similarity when performing nearest-centroid classification using Meta-Baseline, we were interested in determining if

Table 1: Results of ϵ-corruption of the support set on MiniImageNet [Vinyals et al., 2016].

Method	Shot	Corrupted?	Accuracy	Delta
Cosine	5	No	0.792 ± 0.006	
		Yes	0.626 ± 0.008	-0.166 ± 0.014
	20	No	0.848 ± 0.002	
		Yes	0.686 ± 0.008	-0.162 ± 0.010
Euclid.	5	No	0.776 ± 0.005	
		Yes	0.246 ± 0.002	-0.530 ± 0.007
	20	No	0.848 ± 0.003	
		Yes	0.222 ± 0.003	-0.626 ± 0.006

the robust mean estimation algorithm would improve the performance of using the Euclidean distance, and if its effectiveness for model robustness would differ between the two methods. However, we unfortunately ran out of time before we could perform evaluation runs using robust mean estimation with cosine similarity estimation.

3.4 Training Meta-Baseline

To implement the above experimental settings, we trained four versions of Meta-Baseline on each of the two datasets: one fine-tuned using the cosine similarity and one fine-tuned using the Euclidean distance for each of the 5-shot and 20-shot settings, respectively. Aside from the shot number and distance metric used, training hyperparameters were identical to those in Chen et al. [2021].

4 Results

4.1 Testing Details

As stated previously, all tests where the support set is corrupted are run with $\epsilon = 0.2$. All evaluations are run 5-way, with 10 testing epochs consisting of 200 batches of 4 testing episodes, for a total of 800 testing episodes per evaluation. All results are reported here as the average top-1 accuracy of the best testing epoch ± the 95% confidence interval for that epoch.

In evaluations where robust mean estimation (RME) is used, the algorithm is run for 10 iterations with $\epsilon = 0.2$.

4.2 Results of ϵ-corruption

First, we replicate Chen et al. [2021]'s 5-shot results and examine the effects of ϵ-corruption of the support set.

On MiniImageNet [Vinyals et al., 2016] we find that in the 5-shot un-corrupted environment the cosine similarity metric outperforms the Euclidean distance by about 1.6%, aligning with Chen et al. [2021]'s results. Interestingly, however, this advantage vanishes in the 20-shot un-corrupted setting, where the two metrics perform nearly identically. In the corrupted environment, we find that the cosine similarity metric is fairly robust on its own, with an accuracy drop of about 16% in both the 5-shot and 20-shot environments. In contrast, the Euclidean distance is crippled in the corrupted environment, with accuracy drops of 53% and 62% in the 5-shot and 20-shot scenarios, respectively. Curiously, the Euclidean distance performs worse in a corrupted 20-shot environment than in a corrupted 5-shot environment. A possible explanation for this effect is that even though the proportion of outlier to inlier samples is the same in the 5-shot and 20-shot cases, the absolute number of outliers is higher in the 20-shot case. As each outlier can corrupt the covariance of the sample distribution in different directions, this would allow multiple outliers with the same noise distribution to have more cumulative effect on the sample mean than one corrupted sample could. However, the Euclidean norm still performs better than random guessing in both cases (20% accuracy). Full results on MiniImageNet [Vinyals et al., 2016] can be found in Table 1.

On TieredImageNet [Triantafillou et al., 2018], we find that the cosine similarity vastly outperforms the Euclidean distance, with the 5-shot un-corrupted accuracy of the cosine similarity being about double that of the 20-shot un-corrupted Euclidean distance, 83% vs. 41%. Chen et al. [2021] did not

Table 2: Results of ϵ-corruption of the support set on TieredImageNet [Triantafillou et al., 2018]. Note the greater under performance of the Euclidean distance compared to results on MiniImageNet [Vinyals et al., 2016]

Method	Shot	Corrupted?	Accuracy	Delta
Cosine	5	No	0.825 ± 0.006	
		Yes	0.699 ± 0.007	-0.127 ± 0.013
	20	No	0.859 ± 0.005	
		Yes	0.735 ± 0.002	-0.124 ± 0.007
Euclid.	5	No	0.341 ± 0.006	
		Yes	0.210 ± 0.001	-0.131 ± 0.007
	20	No	0.413 ± 0.005	
		Yes	0.201 ± 0.000	-0.212 ± 0.005

Table 3: Results running robust mean estimation via Cheng et al. [2020]'s algorithm on the support set for MiniImageNet [Vinyals et al., 2016] and TieredImageNet [Triantafillou et al., 2018] and classifying using the Euclidean Distance. Note the substantial reduction in accuracy loss between the un-corrupted and corrupted environments.

Dataset	Shot	Corrupted?	Accuracy	Delta
Mini	5	No	0.773 ± 0.006	
		Yes	0.651 ± 0.003	-0.122 ± 0.009
	20	No	0.847 ± 0.003	
		Yes	0.504 ± 0.004	-0.343 ± 0.007
Tiered	5	No	0.351 ± 0.006	
		Yes	0.315 ± 0.006	-0.036 ± 0.012
	20	No	0.425 ± 0.005	
		Yes	0.297 ± 0.003	-0.128 ± 0.008

report the performance of the Euclidean distance on TieredImageNet, so this result is new and defies straightforward explanation. As was stated in section 3.1, the TieredImageNet dataset is generally considered a more difficult task for few-shot image classification models than MiniImageNet, which could explain the Euclidean distance's under performance; however, this explanation does not account for the cosine similarity's high performance on this dataset, where it outperforms its prior results on MiniImageNet, replicating Chen et al. [2021]'s result. We leave further exploration of this for future work.

In the corrupted setting on TieredImageNet, the cosine similarity proves to be even more robust than on MiniImageNet, losing about 12% of its accuracy in both the 5- and 20- shot cases. On the other hand, the Euclidean distance is reduced to essentially random guessing, with about 20% accuracy in both the 5- and 20-shot cases. Full results can be found in Table 2.

4.3 Results with robust mean estimation

Next, we evaluate the effect of running Cheng et al. [2020]'s robust mean estimation algorithm on the support set while classifying using the Euclidean distance on both MiniImageNet and TieredImageNet. In all cases, we note either no change or a slight improvement in accuracy in the un-corrupted environment compared to using the Euclidean distance without robust mean estimation, and a significant increase in accuracy in the corrupted environment. The results are especially stark on TieredImageNet, where robust mean estimation rescues the Euclidean distance from random guessing to only having an accuracy loss of 3.6% in the 5-shot case. Still, the Euclidean distance with RME is outperformed by the cosine similarity without RME in all cases except for the 5-shot corrupted case on MiniImageNet, where the Euclidean distance with RME slightly overperforms with 65% accuracy vs. cosine similarity's 63%. Full results can be found in Table 3.

5 Discussion

Overall, our results show that nearest-centroid based few-shot learning algorithms for image classification are vulnerable to attacks against their support sets, even when these attacks come in the form of random noise rather than adversarially selected examples. Algorithms that utilize the Euclidean distance for nearest-centroid measurement appear to be especially vulnerable to such attacks, as Meta-Baseline consistently loses over 50% of its accuracy on MiniImageNet and is reduced to random guessing on TieredImageNet when its support sets are attacked. In contrast, the cosine similarity seems inherently more robust, as a Meta-Baseline that utilizes it never dropped below 60% accuracy in any of our evaluations.

Our hypothesis for why the cosine similarity is more robust is that the cosine similarity likely inherently ignores certain alterations to the covariance of the support set, due to it being invariant to the magnitude of the vectors being compared. Thus, for any corruption to alter the cosine similarity between two examples, it would need to alter the angle of the embedding, not simply its magnitude, and this may be less probable of an occurrence with random Gaussian corruptions on the original input images. Further work might seek to test this hypothesis by directly corrupting the embedding vectors' angle and magnitude, rather than simply altering the raw images.

Our other major result is that despite not having any error guarantees on datasets of this size, the robust mean estimation algorithm from Cheng et al. [2020] does improve the performance of Meta-Baseline in a corrupted environment, at least when evaluating using the Euclidean distance for nearest-centroid estimation. In all evaluations, utilizing robust mean estimation significantly reduced the accuracy deficit between the un-corrupted and corrupted tests, especially on TieredImageNet where running RME saved Meta-Baseline from random guessing. While there still exist several barriers to this being applied at scale in a real-world environment (see Section 6), this is an important proof of concept demonstrating that algorithms such as Cheng et al. [2020]'s can be used in a few-shot setting, and leaves ample opportunity for future work to further improve on these results.

6 Limitations and future work

Due to time and compute constraints on this project, there are a number of questions we have left unanswered that would be straightforward for future work to explore. Along with the issues discussed below, a lack of available compute kept us from evaluating on Imagenet-800 [Chen et al., 2021] or Meta-Dataset [Triantafillou et al., 2019] as we could not train the required ResNet-18 or ResNet-50 [He et al., 2016] backbone classifiers.

A key issue is that we derived our implementation of Cheng et al. [2020]'s algorithm by porting their implementation into Python via the NumPy [Harris et al., 2020] library. While this allowed us to perform the necessary operations to execute the algorithm (such as top-1 eigenvalue decomposition), this required executing the algorithm on CPU rather than GPU. This, alongside the algorithm's inherent complexity, greatly slowed down evaluation: evaluations without RME took about 45 minutes on an NVIDIA Titan RTX GPU, whereas evaluations with RME took about 36 hours. Future work should seek to develop means to parallelize the algorithm in order to increase evaluation speed.

The primary evaluation that we were unable to perform was running Cheng et al. [2020]'s algorithm while performing classification using the cosine similarity. This would be straightforward to implement, as Chen et al. [2021]'s method for calculating the centroid via cosine similarity first computes the mean of the support embeddings in terms of ℓ_2 norm. As we found the cosine similarity to be inherently more robust than the Euclidean distance, we believe that augmenting cosine similarity evaluation with robust mean estimation will lead to the most robust version of the Meta-Baseline model.

Another key issue left unaddressed here due to time is that we did not perform any hyperparameter search on the number of iterations for the robust mean estimation algorithm. We executed the algorithm for only 10 iterations in every evaluation, far below the $\tilde{O}(Nd^3/\epsilon)$ required by Cheng et al. [2020] for their error guarantees. We hypothesize that running the algorithm for additional iterations would further improve the robustness of Meta-Baseline, but doing so would be impractical unless the above efficiency improvements were achieved.

One of the key constraints of this work is that we only evaluated the use of Cheng et al. [2020]'s algorithm during testing, and did not employ it during the meta-learning phase of Meta-Baseline's training. Our results suggest that employing robust mean estimation during training would not have affected our results while training on an un-corrupted dataset, as testing with robust mean estimation did not significantly alter the model's accuracy on un-corrupted testing data. However, future work should confirm this hypothesis, as well as examine the effects of corrupted training data and the efficacy of robust mean estimation during training at mitigating those effects.

Finally, there are multiple issues that would challenge this system in a real-world environment that we left unexamined here. First is that applying Cheng et al. [2020] requires a prior assumption on the value of ϵ, i.e. the fraction of data that is corrupted. While here we assumed that the algorithm's ϵ was exactly equal to the true fraction of corrupted data for simplicity, future work should test the effects of over- or under-estimating ϵ in a corrupted-data environment. Additionally, here we used random corruption of the data for simplicity; many other works on robustness, however, instead focus on developing adversarial examples, e.g. Subramanya and Pirsiavash [2022]. Future work should test if Cheng et al. [2020]'s algorithm also improves robustness against such examples.

7 Related Work

7.1 High-dimensional robust mean estimation

This work relies heavily on previous research on high-dimensional robust statistics, most prominently Cheng et al. [2020]'s algorithm for high-dimensional robust mean estimation, which utilizes a form of projected gradient descent that Cheng et al. [2020] prove is optimal at all critical points. This work itself follows from earlier robust mean estimation methods, such as Diakonikolas et al. [2017] which relies on filtering heuristics to eliminate outliers and Cheng et al. [2019] which formulates the robust mean estimation problem as a semi-definite program.

7.2 Meta-learning

Here, we rely on Chen et al. [2021]'s Meta-Baseline as our few-shot classification model. This model is one of the latest in a line of research focused on performing nearest-centroid classification in a learned embedding space; such techniques originated with Prototypical Networks [Snell et al., 2017] and were developed further in works such as Tian et al. [2020]. This method contrasts other meta-learning techniques such as MAML [Finn et al., 2017] in its simplicity, and is competitive with contemporaneous state of the art methods for few-shot classification, such as Constellation Networks [Xu et al., 2021].

7.3 Robust classification and meta-learning

The field of robust image classification has become an increasingly active area of research in recent years. Momeny et al. [2021] develops a method for adapting convolutional neural networks to be noise-robust through additional noise mapping and adaptive resizing layers. Subramanya and Pirsiavash [2022] also focuses on few-shot methods similar to Meta-Baseline, but their techniques rely on adversarial training and transfer learning to inure models to adversarially-selected examples. Kong et al. [2020] focuses on robust meta-learning across differently-sized tasks where some tasks are corrupted to varying degrees, and tests its methods in a constrained mixed linear regression environment.

As far as we are aware, applying robust mean estimation for robustness in image classification or meta-learning is novel.

Acknowledgments and Disclosure of Funding

Many thanks to Dr. Yu Cheng, who taught the seminar this project was completed for and who provided code samples for the robust mean estimation algorithm that were essential for getting the project over the finish line. Thanks also to Gidaris and Komodakis [2018] and Lee et al. [2019] who provided prepackaged versions of the MiniImageNet and TieredImageNet splits used in Chen et al. [2021], respectively. Finally, many thanks to the GridEngine admins, without whose work none of our evaluations would have been possible.

As a student project, this had no funding sources.

References

Yinbo Chen, Zhuang Liu, Huijuan Xu, Trevor Darrell, and Xiaolong Wang. Meta-baseline: Exploring simple meta-learning for few-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9062–9071, 2021.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 29, 2016.

Eleni Triantafillou, Hugo Larochelle, Jake Snell, Josh Tenenbaum, Kevin Jordan Swersky, Mengye Ren, Richard Zemel, and Sachin Ravi. Meta-learning for semi-supervised few-shot classification. 2018.

Yu Cheng, Ilias Diakonikolas, Rong Ge, and Mahdi Soltanolkotabi. High-dimensional robust mean estimation via gradient descent. In *International Conference on Machine Learning*, pages 1768–1778. PMLR, 2020.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022.

Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12104–12113, 2022.

Christiane Lemke, Marcin Budka, and Bogdan Gabrys. Metalearning: a survey of trends and technologies. *Artificial intelligence review*, 44(1):117–130, 2015.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30, 2017.

Peter J. Huber. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 35 (1):73–101, 1964. ISSN 00034851. URL `http://www.jstor.org/stable/2238020`.

Ilias Diakonikolas, Gautam Kamath, Daniel Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robust estimators in high-dimensions without the computational intractability. *SIAM Journal on Computing*, 48(2):742–864, 2019.

Yu Cheng, Ilias Diakonikolas, and Rong Ge. High-dimensional robust mean estimation in nearly-linear time. In *Proceedings of the thirtieth annual ACM-SIAM symposium on discrete algorithms*, pages 2755–2771. SIAM, 2019.

Ilias Diakonikolas, Gautam Kamath, Daniel M. Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Being robust (in high dimensions) can be practical. ICML'17, page 999–1008. JMLR.org, 2017.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Utku Evci, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Swersky, Pierre-Antoine Manzagol, et al. Meta-dataset: A dataset of datasets for learning to learn from few examples. *arXiv preprint arXiv:1903.03096*, 2019.

Mohammad Momeny, Ali Mohammad Latif, Mehdi Agha Sarram, Razieh Sheikhpour, and Yu Dong Zhang. A noise robust convolutional neural network for image classification. *Results in Engineering*, 10:100225, 2021.

Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020. doi: 10.1038/s41586-020-2649-2. URL `https://doi.org/10.1038/s41586-020-2649-2`.

Akshayvarun Subramanya and Hamed Pirsiavash. A simple approach to adversarial robustness in few-shot image classification. *arXiv preprint arXiv:2204.05432*, 2022.

Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B Tenenbaum, and Phillip Isola. Rethinking few-shot image classification: a good embedding is all you need? In *European Conference on Computer Vision*, pages 266–282. Springer, 2020.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.

Weijian Xu, Yifan Xu, Huaijin Wang, and Zhuowen Tu. Attentional constellation nets for few-shot learning. In *International Conference on Learning Representations*, 2021.

Weihao Kong, Raghav Somani, Sham Kakade, and Sewoong Oh. Robust meta-learning for mixed linear regression with small batches. *Advances in neural information processing systems*, 33: 4683–4696, 2020.

Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4367–4375, 2018.

Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10657–10665, 2019.