GENERALIZED DISTRIBUTION CALIBRATION FOR FEW-SHOT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Few shot learning is an important problem in machine learning as large labelled datasets take considerable time and effort to assemble. Most few-shot learning algorithms suffer from one of two limitations— they either require the design of sophisticated models and loss functions, thus hampering interpretability; or employ statistical techniques but make assumptions that may not hold across different datasets or features. Developing on recent work in extrapolating distributions of small sample classes from the most similar larger classes, we propose a Generalized sampling method that learns to estimate few-shot distributions for classification as weighted random variables of all large classes. We use a form of covariance shrinkage to provide robustness against singular covariances due to overparameterized features or small datasets. We show that our sampled points are close to few-shot classes even in cases when there are no similar large classes in the training set. Our method works with arbitrary off-the-shelf feature extractors and outperforms existing state-of-the-art on miniImagenet, CUB and Stanford Dogs datasets by 3% to 5% on 5way-1shot and 5way-5shot tasks and by 1% in challenging cross domain tasks.

1 INTRODUCTION

Few-shot learning (FSL) refers the problem of learning from datasets where only a limited number of examples (typically, one to tens per class or a problem in general) are available for training a machine learning model. FSL has gained importance over the years since obtaining large labelled datasets requires a significant investment of time and resources. There is a rich history of research into various methodologies for FSL, two primary ones being model development approaches and representation learning. (Wang et al., 2020).

Model development approaches aim at capturing the data distribution so that new points can be sampled to improve few-shot classification accuracy (Park et al., 2020), (Wang et al., 2019), (Chen et al., 2019b), (Zhang et al., 2019). They have typically relied on complex models and loss functions to understand data from only a few examples, which limits the interpretability of the model and hampers generalization. Representation learning, on the other hand, aims at identifying feature transformations which can allow simple statistical techniques like nearest neighbor and bayesian classification generalize on few-shot tasks for novel classes (Chen et al., 2020), (Xue & Wang, 2020). Starting from early works (Miller et al., 2000) and building to recent methodologies (Yang et al., 2021), (Zhang et al., 2021b), representation learning strategies have relied on simple statistical assumptions that may not hold across diverse datasets.

Recently, (Yang et al., 2021) showed that classes which are semantically similar in meaning are also correlated in the means and covariances of their feature distributions. They used the statistics of classes with plentiful datapoints (called base classes) to learn the distribution of classes with a few datapoints (called novel classes). Despite some limiting assumptions, their method, called Distribution Calibration, outperformed much more complex models that relied on non-parametric optimization and generative models (Park et al., 2020), (Wang et al., 2019), (Chen et al., 2019b), (Zhang et al., 2019). Building on this idea and inspired by traditional statistics, we present a rigorous generalized sampling method for Few-Shot Learning, called DC+ that outperforms existing state-of-the-art classification accuracy without introducing additional statistical assumptions or complex generative models. Our main contributions are:

1. Introducing a principled approach to estimate novel class mean and covariance from the moments of a random variable weighted by the distance between the novel and the base classes,
2. Incorporating the statistical technique of variance shrinkage, which not only helps increase the accuracy but also stabilizes covariance estimation in cases when the feature extractor is over-parametrized (a common occurrence in modern deep learning models),
3. Extending the applicability of the statistical sampling approach to arbitrary feature extractors by introducing general Gaussianization transformations,
4. Presenting a single scaling parameter in Euclidean distance weighting that mitigates the need to search among Euclidean, Mahalanobis, and generalized distances for novel class estimation, and

Combined, our contributions put statistical sampling approach on a sound foundation, close open questions in earlier research, and demonstrate 3% to 5% improvement over competitive states-of-the-art for 5way-1shot and 5way-5shot tasks for miniImageNet (Ravi & Larochelle, 2017), CUB (Welinder et al., 2010), StanfordDogs (Khosla et al., 2011), highest level classes of tieredImagenet (Ren et al., 2018) and 1% improvement Cross Domain 5way-1shot task of miniImagenet $\rightarrow$ CUB.

2 Related Works

Learning good features or manipulating the features to help generalize few-shot tasks is an active research area (Hou et al., 2019) (Hao et al., 2019), (Li et al., 2019a). Miller et al. (2000) proposed a congealing process which learned a joint distribution among all available classes. This distribution could then be used as a prior knowledge for constructing efficient few-shot classifiers. Chen et al. (2020) showed that by pre-training first on entire base classes, few shot classification accuracies on novel classes could be improved with a simple meta-learning on nearest-centroid based classification algorithm. Their main focus was on the pretraining methods which could improve a cosine based classifier on the centroids of the extracted features. DC+ can be applied on top of these feature extractor techniques to further explore improvements.

To correct bias in centroid or prototype estimations, several rectification methods were proposed—RestoreNet (Xue & Wang, 2020) transforms the feature space to move the images closer to their true centroids. Our proposed method approximates this transformation with scaled Euclidean distances and weighted random variables for novel classes, without any additional learnable parameters. Zhang et al. (2021a) proposed a 4-step method consisting of learning base class details as priors and then using these priors for correcting bias in novel prototype estimation. They then jointly fine tuned the feature extractor and bias corrector. Our method does not have this multi-step process and works with off-the-shelf feature extractors. Liu et al. (2020c) attempted to reduce the bias in distance estimation through reducing intra-class bias by label propagation and cross-class bias through shifting features. Their method works in the transductive setting where entire data, including the query set is consumed without label information. Our method does not need additional unlabelled data from the query set. Again their method improves upon Prototypical Networks (Snell et al., 2017) whereas we show that our method can be applied with any feature extractor.

DC+ can be broadly categorized as a data augmentation method. Several methods have been proposed earlier in this space. Antoniou & Storkey (2019) learn few-shot tasks by randomly labelling a subset of images and then augmenting this subset with different techniques like random crops, flips etc. Park et al. (2020) tried to transfer variance between different classes in order to simulate the query examples that can be encountered during test. Other works like Wang et al. (2019) and Liu et al. (2020b) utilized the intra-class variance to perform augmentation. These methods leverage complex neural networks with large number of learnable parameters to generate new examples. Our method does not require any additional learnable parameter and uses simple statistical techniques while still outperforming all previous deep learning methods.

Closest to DC+, Yang et al. (2021) estimated the novel class distributions based on their similarity with the base classes. Their method implicitly assumed that the base classes were semantically independent of each other when constructing covariance estimates, did not consider the similarity strength between the base and novel classes when estimating novel class statistics, and could not be applied to arbitrary off-the-shelf feature extractors (with activation functions different from `relu`)

and large feature dimensions often capable of producing ill-defined covariances. Our method does not make independence assumptions, leverages similarity information in the base classes, and can be applied to any off-the-shelf feature extractor.

3 PROPOSED APPROACH

3.1 PROBLEM DEFINITION

Few-shot classification problems are defined as Nway-Kshot classification tasks $\mathcal{T}$ (Vinyals et al., 2016) where given a small support set $\mathbb{S}$ of features $\tilde{\mathbf{x}}$ and labels y, $\mathbb{S} = \{(\tilde{\mathbf{x}}_i, y_i)\}_{i=1}^{N\times K}, \tilde{\mathbf{x}}_i \in \mathbb{R}^d, y_i \in \mathbb{C}$, consisting of K points from N classes, the model should correctly classify a query set $\mathbb{Q} = \{(\tilde{\mathbf{x}}_i, y_i)\}_{i=N\times K+1}^{N\times K+N\times q}$ with q points from each of the N classes in the support set. The entire dataset $\mathbb{D}$, is divided into $\mathbb{C}_b$ base, $\mathbb{C}_v$ validation and $\mathbb{C}_n$ novel classes such that $\mathbb{C}_b \cap \mathbb{C}_v \cap \mathbb{C}_n = \phi$ and $\mathbb{C}_b \cup \mathbb{C}_v \cup \mathbb{C}_n = \mathbb{C}$. The goal is to train a model with tasks $\mathcal{T}$ sampled from $\mathbb{C}_b$ and use $\mathbb{C}_v$ for hyperparameter tuning, where each task $\mathcal{T}$ is an Nway-Kshot classification problem on N unique classes of the set under consideration, for example base, validation set $\mathbb{C}_b, \mathbb{C}_v$ here. The performance of few-shot learning algorithms is reported as the average accuracy on the query set $\mathbb{Q}$ of tasks $\mathcal{T}$ sampled from $\mathbb{C}_n$.

3.2 ALGORITHM

Our proposed methodology, DC+, is outlined in Algorithm 1. In the following subsections, we incrementally go through the steps of DC+ in detail.

3.2.1 GAUSSIANIZATION OF THE DATA

Following Yang et al. (2021), our sampling methodology assumes that the input features follow a multivariate normal distribution. There are many methods of data gaussianization like Tukey's Ladder of Powers (Tukey, 1977), Yeo-Johnson Transformation (Weisberg, 2001), and Iterative Gaussianization (Chen & Gopinath, 2000). In our experiments, we observed that Tukey's Ladder of Powers outperformed other methods, but the fractional powers and log transform could only be applied to non-negative features (feature extractors which have `relu` and equivalent activation functions in their final layers). Expanding the applicability of the sampling method to arbitrary feature extractors and activation functions in deep learning models, we make a choice on the transformation of input features denoted by random variable $\mathbf{x}$ as per equation 1 below,

$$\hat{\mathbf{x}} = \begin{cases} tukey(\mathbf{x}) & if\ \mathbf{x} \geq 0\ \ always \\ yeo\_johnson(\mathbf{x}) & otherwise \end{cases} \tag{1}$$

We give the definitions of $tukey$ and $yeo\_johnson$ in Appendix A.

3.2.2 PROPOSED RANDOM VARIABLE

We extrapolate the distribution of a given novel class as a weighted average of the distributions of k closest base classes. Formally, if $\tilde{\mathbf{x}} \in \mathbb{R}^d$ is a d dimensional support point from a novel class, and $\mathbf{X}_i \in \mathbb{R}^d$ is a random variable denoting points of base class i, then we compose a random variable $\mathbf{X}'$ representing our estimate of that novel class as

$$\mathbf{X}' = \frac{\tilde{\mathbf{x}} + \sum_{i\in\mathbb{S}_k} w_i \mathbf{X}_i}{1 + \sum_{i\in\mathbb{S}_k} w_i}, \tag{2}$$

where w_i are the weights assigned to the closest base classes in $\mathbb{S}_k$. The associated mean and the covariance of this random variable are (since $\tilde{\mathbf{x}}$ is a constant vector, it does not affect the covariance in $\mathbf{\Sigma}'$),

$$\boldsymbol{\mu}' = \frac{\tilde{\mathbf{x}} + \sum_{i\in\mathbb{S}_k} w_i \boldsymbol{\mu}_i}{1 + \sum_{i\in\mathbb{S}_k} w_i} \qquad \mathbf{\Sigma}' = cov(\mathbf{X}') \tag{3}$$

where $\boldsymbol{\mu}_i = \mathbb{E}[\mathbf{X}_i]$. There are many ways of estimating w_i. One of the simplest is to look at the distance of the novel point from the base classes. In particular, we find the k closest base classes to

$\tilde{\mathbf{x}}$ in $\mathbb{S}_k$,

$$d_i = ||\tilde{\mathbf{x}} - \boldsymbol{\mu}_i||^2, i \in \mathbb{C}_b \tag{4}$$

$$\mathbb{S}_k = \{i | - d_i \in topk(-d_i), i \in \mathbb{C}_b\} \tag{5}$$

Based on d_i (for alternative distance formulations, see Appendix B), we construct the weights w_i of each base class in $\mathbb{S}_k$ as,

$$w_i = \frac{1}{1 + d_i^m}, i \in \mathbb{S}_k \tag{6}$$

where m is a hyperparameter that helps in decaying w_i as a function of d_i and gives us control in the relative weights of the classes in $\mathbb{S}_k$. This form of weighted variable estimation is reminiscent of (though not the same as) inverse distance weighting, which is widely used to estimate unknown functions at interpolated points (Shepard, 1968), (Łukaszyk, 2004) It is worth noting that $\lim_{d_i \to 0} w_i = 1$. Hence as the base class i moves closer to the support point $\tilde{\mathbf{x}}$, it gains increasing weight until $\boldsymbol{\mu}_i$ overlaps with $\tilde{\mathbf{x}}$, at which point the weight is 1, same as DC method.

3.2.3 SHRINKING THE COVARIANCE

When the number of data points in base class $\mathbf{X}_i$ is less than the feature dimensions, i.e. $|\mathbf{X}_i| < d$, $cov(\mathbf{X}_i) = \boldsymbol{\Sigma}_i$ is in general non-invertible, which prohibits constructing a normal distribution with $\boldsymbol{\Sigma}'$ and poses a serious limitation since most off-the-shelf deep learning feature extractors have large feature dimensions (Zagoruyko & Komodakis, 2016), (Szegedy et al., 2015). We propose a variant of covariance shrinkage (Van Ness, 1980), (Friedman, 1989) to stabilize the $\boldsymbol{\Sigma}'$ in equation 3 against singularities by introducing two hyperparameters that dictate the relative strength of the dimension variances and the off-diagonal covariance interactions as,

$$\boldsymbol{\Sigma}'_s = \boldsymbol{\Sigma}' + \alpha_1 \sigma_1 \mathbf{I} + \alpha_2 \sigma_2 (\mathbf{1} - \mathbf{I}), \tag{7}$$

where σ_1 is the average diagonal variance and σ_2 is the average off-diagonal covariance of $\boldsymbol{\Sigma}'$.

3.2.4 SAMPLING THE NOVEL CLASS

With $\boldsymbol{\mu}', \boldsymbol{\Sigma}'_s$ now formulated, we have both the mean and covariance to represent the ground truth distribution of the novel class associated with the support point $\tilde{\mathbf{x}}$. Hence we can sample n points from this extrapolated distribution and append them to the existing support set,

$$\mathbb{D}_y = \{(\mathbf{x}, y) | \mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}', \boldsymbol{\Sigma}'_s)\} \tag{8}$$

where y denotes the class of $\tilde{\mathbf{x}}$. Steps from Section 3.2.1 to 3.2.4 are repeated for every point $\tilde{\mathbf{x}}$ in the support set $\mathbb{S}$, which has N classes with K points in each class constituting an Nway-Kshot classification task $\mathcal{T}$. Hence the total dataset $\mathbb{D}$ after augmentation can be written as,

$$\mathbb{D} = \cup \{\mathbb{D}_y \forall (\tilde{\mathbf{x}}, y) \in \mathbb{S}\} \cup \mathbb{S} \tag{9}$$

$\mathbb{D}$ can be used to construct a classifier like Logistic Regression, SVM etc. and the performance is reported as the average accuracy on the query set $\mathbb{Q}$ in $\mathcal{T}$.

4 EXPERIMENTS

In this section we compare the performance of DC+ with other states-of-the-art, show that our sampled points are closer to query data than DC, give theoretical insights with empirical results on the generalization improvement of DC+, and perform an ablation study to show the effectiveness of each component in our method.

4.1 IMPLEMENTATION DETAILS

4.1.1 DATASETS

We compare our proposed method with existing states-of-the-art on **miniImagenet** (Ravi & Larochelle, 2017), **CUB** (Welinder et al., 2010) and **Stanford Dogs** (Khosla et al., 2011). The details of the train/test/validation splits of each dataset is given in Appendix D. To show that our

Algorithm 1 DC+: Generalized Sampling Method for Few-shot learning

Require: Base class features $\mathbf{X}_i \in \mathbb{R}^d, i \in \mathbb{C}_b$
Require: Support set features $\mathbb{S} = \{(\tilde{\mathbf{x}}_j, y_j)\}_{j=1}^{N \times K} : \tilde{\mathbf{x}}_j \in \mathbb{R}^d, y_j \in \mathbb{C}_n$
  for $(\tilde{\mathbf{x}}_j, y_j) \in \mathbb{S}$ **do**
    Gaussianize $\tilde{\mathbf{x}}$ with equation 1
    Find the nearest k base classes using d_i, $\mathbb{S}_k$ (equation 4, 5)
    Calculate weights w_i for each base class $i \in \mathbb{S}_k$ (equation 6)
    Use w_i to find the weighted random variable $\mathbf{X}'$ (equation 2)
    Calculate $\boldsymbol{\mu}'$, $\boldsymbol{\Sigma}'$ (equation 3)
    Shrink $\boldsymbol{\Sigma}'$ to get $\boldsymbol{\Sigma}'_s$ (equation 7)
    Sample features for class y_j using $\{(\mathbf{x}, y_j) | \mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}', \boldsymbol{\Sigma}'_s)\}$ (equation 8)
  end for
  Construct $\mathbb{D}$ by appending all sampled features to the support set $\mathbb{S}$ (equation 9)
  Train a logistic regression classifier on $\mathbb{D}$ (Details in Appendix C)
  Report the accuracy on query set $\mathbb{Q} = \{(\tilde{\mathbf{x}}_j, y_j)\}_{j=N \times K+1}^{N \times K+N \times q}$

method gives superior performance when the base classes are dissimilar to the novel classes, we evaluate our method on **Cross Domain** dataset by training on tasks sampled from one distribution and evaluating on a different distribution. Specifically, we follow Patacchiola et al. (2020) and show results on **miniImagenet** $\rightarrow$ **CUB**, i.e. train split from miniImagenet and test/validation split from CUB. We also compare our method with DC (Yang et al., 2021) on a **meta-tieredImagenet** dataset of the 34 broad categories from tieredImagenet, split into 20 base, 8 novel and 6 validation classes, as laid out in Ren et al. (2018). Note that there is a high dissimilarity between the base and novel/validation classes in this meta-tieredImagenet as seen in Table 6 of Appendix D.

4.1.2 Feature Extractor

We used a WRN-28-10 (Zagoruyko & Komodakis, 2016) feature extractor trained using S2M2 Method (Mangla et al., 2020) for miniImagenet, CUB, Stanford Dogs and meta-tieredImagenet experiments. For our cross domain results on miniImagenet $\rightarrow$ CUB, we used a Conv-4 backbone to compare our results with other states-of-the-art as done in Patacchiola et al. (2020). Details of our feature extractor training can be found in Appendix E.

The extracted features were 640 dimensional for miniImagenet and CUB which had 600 and 44 points in each base class. Feature dimensions were 1600 for miniImagenet $\rightarrow$ CUB experiments with 600 points in each base class. Note that such mismatch between the feature dimensions and the number of datapoints leads to singularities when trying to estimate the covariance matrix. Our shrinkage method takes care of these pathological situations. To show that our proposed algorithm works equally well for non-singular cases, we extracted 64 dimensional representation for Stanford Dogs and meta-tieredImagenet which had 148 and 12950 points in each of their base classes.

To demonstrate that our gaussianization approach (Section 3.2.1) generalizes, our features in mini-Imagenet and CUB come from a penultimate `relu` layer hence are all non negative. For Stanford Dogs and meta-tieredImagenet, we project 640 dimensional features from WRN-28-10 to 64 dimensions in the penultimate layer and hence the features span in both positive and negative regions.

4.1.3 Hyperparameter Search

To circumvent combinatorial explosion from grid search of a larger number of hyperparameters, we used optuna (Akiba et al., 2019) library for tuning our hyperparameters $\beta, m, k, \alpha_1, \alpha_2, n$. All tunings are 1000 trials long with a `Median Pruner`. We give complete details of our optuna setting and picking the best hyperparameters in Appendix F.

4.2 Comparison with states-of-the-art

Tables 1 and 2 summarize the performance of our proposed DC+ with existing states-of-the-arts. We report the average classification accuracy of 5000 random tasks $\mathcal{T}$ sampled from the novel classes

Table 1: Comparing the results of our proposed algorithm DC+ on Stanford Dogs and CUB with 95% confidence intervals. Best results highlighted in bold.

	Stanford Dogs		CUB	
Methods	5way-1shot	5way-5shot	5way-1shot	5way-5shot
RelationNet (Sung et al., 2018)	43.33 ± 0.42	55.23 ± 0.41	62.45 ± 0.98	76.11 ± 0.69
adaCNN (Munkhdalai et al., 2018)	41.87 ± 0.42	53.93 ± 0.44	56.57 ± 0.47	61.21 ± 0.42
PCM (Wei et al., 2019)	28.78 ± 2.33	46.92 ± 2.00	42.10 ± 1.96	62.48 ± 1.21
CovaMNet (Li et al., 2019c)	49.10 ± 0.76	63.04 ± 0.65	60.58 ± 0.69	74.24 ± 0.68
DN4 (Li et al., 2019b)	45.41 ± 0.76	63.51 ± 0.62	52.79 ± 0.86	81.45 ± 0.70
PABN+cpt (Huang et al., 2021)	45.65 ± 0.71	61.24 ± 0.62	63.56 ± 0.79	75.35 ± 0.58
LRPABN+cpt (Huang et al., 2021)	45.72 ± 0.75	60.94 ± 0.66	63.63 ± 0.77	76.06 ± 0.58
MML (Chen et al., 2021)	59.05 ± 0.68	75.59 ± 0.51	63.86 ± 0.67	80.73 ± 0.46
DC (Yang et al., 2021)	-	-	79.56 ± 0.87	90.67 ± 0.35
DC+ (Ours)	**65.35 ± 0.61**	**80.56 ± 0.45**	**84.57 ± 0.48**	**93.46 ± 0.25**

Table 2: Comparing the results of our proposed algorithm DC+ on miniImagenet and Cross Domain miniImagenet → CUB tasks with 95% confidence intervals. Best results highlighted in bold.

	miniImagenet		miniImagenet → CUB	
Methods	5way-1shot	5way-5shot	5way-1shot	5way-5shot
MatchingNet (Vinyals et al., 2016)	43.56 ± 0.84	55.31 ± 0.73	36.98 ± 0.06	50.72 ± 0.36
ProtoNet (Snell et al., 2017)	54.16 ± 0.82	73.68 ± 0.65	33.27 ± 1.09	52.16 ± 0.17
MAML (Finn et al., 2017)	48.70 ± 1.84	63.11 ± 0.92	34.01 ± 1.25	48.83 ± 0.62
RelationNet (Sung et al., 2018)	50.44 ± 0.82	65.32 ± 0.70	37.13 ± 0.20	51.76 ± 1.48
Baseline++ (Chen et al., 2019a)	51.87 ± 0.77	75.68 ± 0.63	39.19 ± 0.12	**57.31 ± 0.11**
DKT (Patacchiola et al., 2020)	49.73 ± 0.07	64.00 ± 0.09	40.22 ± 0.54	56.40 ± 1.34
E3BM (Liu et al., 2020d)	63.80 ± 0.40	80.29 ± 0.25	-	-
Negative-Cosine (Liu et al., 2020a)	62.33 ± 0.82	80.94 ± 0.59	-	-
Meta Variance Transfer (Park et al., 2020)	-	67.67 ± 0.70	-	-
DC (Yang et al., 2021)	68.57 ± 0.55	82.88 ± 0.42	35.08 ± 0.55	50.81 ± 0.43
DC+ (ours)	**73.00 ± 0.50**	**87.22 ± 0.33**	**41.08 ± 0.53**	54.69 ± 0.41

$\mathbb{C}_n$ along with their 95% confidence intervals. The exact value of each hyperparameter that give the reported accuracy along with other candidate hyperparameters are given in Appendix G.

We observe that a logistic regression constructed on our sampled data achieves consistent performance improvements of 3% (CUB 5way-5shot) to 5% (CUB 5way-1shot) compared to the 2nd best method. We also achieve 1% improvement on challenging cross domain 5way-1shot classification of miniImagenet → CUB. Finally, for the case when base classes and novel classes share very little similarity (as is the case for the meta-tieredImagenet), we outperform DC by more than 9% for 5way-5shot task as shown in Table 3.

Note that these accuracy improvements are derived from a simple statistical model that leverages just 3 additional hyperparameters compared to the 2nd best method of DC Yang et al. (2021). We also do not make any assumptions about the feature values, the feature dimensions, and show these improvements across different cases of singular (miniImagenet, CUB, miniImagenet → CUB) and non-singular covariances (Stanford Dogs, meta-tieredImagenet).

4.3 Visualization of Sampled Points

In Figure 1 we compare the t-SNE representation (van der Maaten & Hinton, 2008) of sampled points for both DC (Yang et al., 2021) and our method DC+ to visualize whether our sampled points are closer to the ground truth as indicated quantitatively from results in Tables 1, 2, 3. We can see that our method produces clusters of sampled points which are more compact and overlap with more query points than the sampled points of DC. We analyse this in the next section 4.4. For more visualizations refer to Appendix H.

Table 3: Comparing the results of our proposed algorithm DC+ on meta-tieredImagenet 5way-1shot and 5way-5shot tasks with 95% confidence intervals. The results below are reported for 5000 tasks sampled from the novel classes.

	meta-tieredImagenet	
Methods	5way-1shot	5way-5shot
No Method (baseline)	20.00 ± 0.00	20.00 ± 0.00
DC (Yang et al., 2021)	39.54 ± 0.64	47.11 ± 0.78
DC+ (Ours)	**43.51 ± 0.50**	**56.79 ± 0.64**

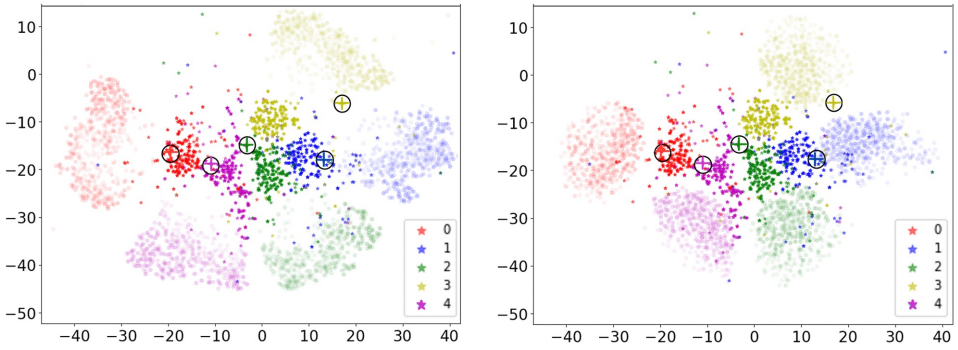

Figure 1: t-SNE visualization for 5 randomly sampled novel classes for DC (left) and our method DC+ (right) in miniImagenet dataset. The support points are indicated with a '+' sign within black circles, sampled points are semi-transparent indicated with a 'o' sign, and the query points are opaque denoted by '*'. Note that our sampled points are on average closer to the query points.

4.4 WHY SHOULD DC+ GENERALIZE BETTER THAN DC?

To motivate the generalization and performance improvements of DC+, we start with a comparison of the weighting scheme between DC and DC+. Figure 2 (left) shows the estimated position of $\mathbf{X}'$ as the closest base class moves away from $\tilde{\mathbf{x}}$ for the simple case of one base class. The unweighted random variable $\mathbf{X}'$ in DC can be written as $\mathbf{X}' = (\tilde{\mathbf{x}} + \sum_{i \in \mathbb{S}_k} \mathbf{X}_i)/(1+k) = (\tilde{\mathbf{x}} + \mathbf{X}_1)/2$, for k=1.

Hence as $\boldsymbol{\mu}_i = E(\mathbf{X}_1)$ moves away from $\tilde{\mathbf{x}}$, $\mathbf{X}'$ moves further apart in DC as seen in Figure 2. Thus even though the method borrows stable values of mean and covariance from the base class in question, the values themselves are far from the support point, which acts as a proxy location for the query points during training. In contrast, with a higher m in DC+, the weight w_i assigned to base class i drops polynomially with distance from $\tilde{\mathbf{x}}$ and $\mathbf{X}'$ saturates at $\tilde{\mathbf{x}}$ when the base class is far from $\tilde{\mathbf{x}}$. Hence our sampled points should be much closer to $\tilde{\mathbf{x}}$ giving us better generalization than DC.

To confirm this hypothesis, we visualize the sampled points for $k = 1$ in miniImagenet $\rightarrow$ CUB experiment in Figure 2. We chose this dataset to also show how a higher m can help offset noise from base classes by minimizing their w_i in estimating the distribution $\mathbf{X}'$ when they are not similar to the novel class of $\tilde{\mathbf{x}}$. We can see that points produced by DC and $m = 0, 1$ are far from the support point $\tilde{\mathbf{x}}$. As m increases, the cluster moves closer to $\tilde{\mathbf{x}}$ where at $m = 4$ the cluster overlaps with $\tilde{\mathbf{x}}$. Our peformance improvement of 5% on 5way-1shot and 4% on 5way-5shot tasks of Cross Domain miniImagenet $\rightarrow$ CUB experiments compared to DC further confirms this hypothesis.

This same trend can also be seen in the results of meta-tieredImagenet in Table 3 where DC+ converges on a high value of m ($m = 2.25$ in 5way-1shot and $m = 2$ in 5way-5shot) giving weights to base classes that are 10 to 100 smaller than miniImagenet, and 100 times smaller than DC. We further analyse this case and visualize the histogram of these weights comparing them with DC, and weights from miniImagenet experiments in Appendix I.

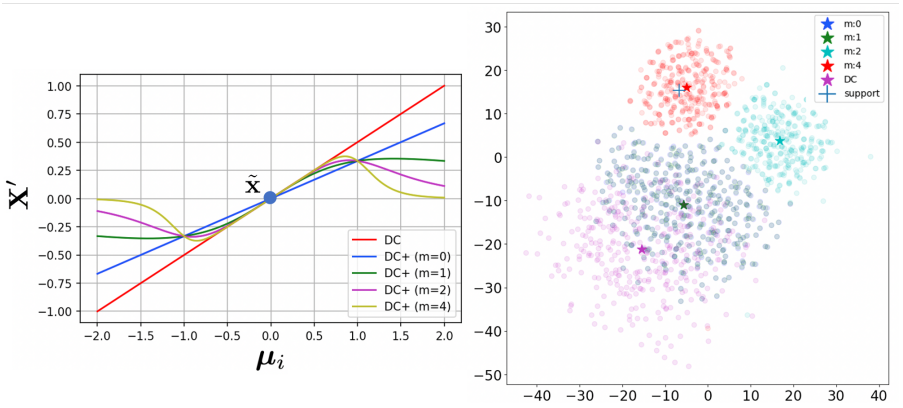

Figure 2: **Left**: Novel class estimate $\mathbf{X}'$ as a function of $\boldsymbol{\mu}_i$ for a 1 dimensional example. $\tilde{\mathbf{x}}$ is at origin. Note that as base class i moves away from $\tilde{\mathbf{x}}$, the error in DC's estimation of $\mathbf{X}'$ accumulates. With $m = 4$, DC+ still produces $\mathbf{X}'$ close to $\tilde{\mathbf{x}}$. **Right**: t-SNE plots of sampled points along with their mean (denoted by '*') confirming analysis in **Left** figure. Note that the sampled points of $m = 4$ overlaps with $\tilde{\mathbf{x}}$. Sampled clusters of $m = 0, 1$ overlap with each other.

4.5 Ablation Study

We ran a 5way-1shot ablation study on the Stanford Dogs dataset to estimate the effect of each hyperparameter of our method. For each experiment, the hyperparameter range were defined as,

$$m \in [low = 0, high = 3, step = 0.25] \qquad k \in [low = 1, high = 20, step = 1]$$
$$\alpha_1 \in [low = 0, high = 600, step = 100] \qquad \alpha_2 \in [low = 0 \times \alpha_1, high = 600 \times \alpha_1, step = 100]$$

We fixed $n = 750$, and the total number of trials in our `optuna` hyperparameter search to 100 so that the accuracy improvements with the addition of each hyperparameter were independent of the search time. For each trial, the number of tasks $\mathcal{T}$ was also fixed to 200.

Table 4 summarizes the results of our study and shows that each component of DC+ is important in giving improvements over state-of-the-art. For the Gaussianization step, the final logistic regression model was trained on one data point from each novel class in a 5way classification task. Notice that $\beta = 1$ in our study implies that raw features from 64 dimensional linear layer give best accuracy. The results for $m = 1$ show that introducing weighted random variables to model novel class distributions helps even without tuning m. Further control on decaying w_i as a function of d_i as discussed in Section 4.4 with a tuned m helps in achieving state-of-the-art.

Table 4: Ablation study of DC+ on Stanford Dogs 5way-1shot task showing the change in accuracy (with 95% confidence interval over 200 tasks) as each component of our model is added incrementally. The cumulative improvement in accuracy is a significant 4% compared to the baseline.

Step	Fixed Hyperparameter	Tuned Hyperparameter	5way-1shot
Gaussianization	Baseline, only hyperparameter is β	β=1	60.44 ± 0.98%
Top-k Selection	α_1=0, α_2=0, m=0	k=1	61.27 ± 0.97% (↑0.8%)
Distance-weighted random variable	α_1=0, α_2=0, m=1	k=1	61.98 ± 0.98% (↑0.7%)
Shrinkage (with α_1)	α_2=0, m=1	k=4, α_1=100	63.46 ± 0.95% (↑1.5%)
Shrinkage (with α_1 and α_2)	m=1	k=12,α_1=400, α_2=200α_1	63.92 ± 0.97% (↑0.4%)
All parameters tuned simultaneously	None	k=10,α_1=400, α_2=100α_1,m=1.5	64.33 ± 0.99% (↑0.4%)

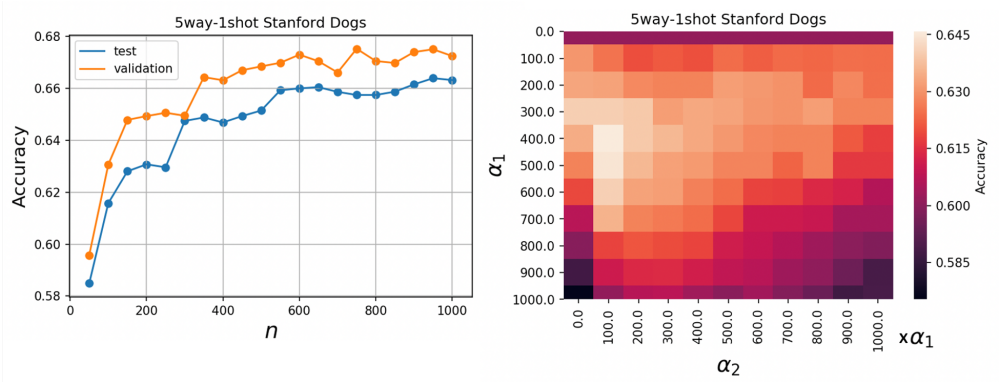

Figure 3: **Left**: Accuracy v/s n the number of sampled features for validation and novel classes in 5way-1shot Stanford Dogs dataset. **Right**: Accuracy v/s α_1, α_2 in 5way-1shot Stanford Dogs dataset. The accuracy for each point for both n and α_1, α_2 experiment is an average over 200 tasks.

4.5.1 Effect of k

We investigated whether the weight decay parameter m can remove the need for the hyperparameter k, since increasing m reduces the weight w_i of a base class that is farther from a given novel point. In Table 4 we see that the best k equals 10 for the case when all hyperparameters are tuned together. This indicates that k and m work together to estimate novel class statistics (otherwise the best value would be $k = 30 = |\mathbb{C}_b|$, number of base classes in Stanford Dogs dataset). To see how much the introduction of k is helping the accuracy, we tuned all hyperparameters in Stanford Dogs 5way-1shot task and find that without k, the accuracy was 63.46% with $m = 1.5, \alpha_1 = 600, \alpha_2 = 0$. This was 0.9% lower than the best result with all k, α_1, α_2, m tuned in the last row of Table 4.

4.5.2 Effect of n, α_1, α_2

The value of n, the number of sampled points, was fixed to 750 in all our experiments. Figure 3 shows the trend of accuracy v/s n in our ablation study on Stanford Dogs 5way-1shot task. We can see that the accuracy keeps increasing with slight noise until $n = 750$ after which it saturates. It indicates that the sampling more points after a certain number does not cover additional query points in this specific feature space.

From Table 4 we see that α_1, α_2 play a key role in improving accuracy. In Figure 3 the heatmap of accuracy versus α_1 and α_2 shows that accuracy increases with increasing α_1, α_2 and then starts dropping beyond $\alpha_1 > 200$. This could be due to overlapping clusters of sampled points with a higher covariance, where the decision boundaries start overlapping.

5 Conclusion

In this work, we have proposed a principled approach to estimate novel class distributions by formulating a similarity-based weighted random variable of closest base classes. We showed that incorporating statistical techniques of covariance shrinkage and Gaussianization not only generalize our method (DC+) to arbitrary pretrained feature extractors, but also increase the accuracy over state-of-the-art significantly. Our experiments effectively demonstrate cumulative performance improvements of 1% to 9% over DC including challenging cross domain tasks. Exploring the trade-off between more hyperparameters and accuracy along with generalizations to different tasks like non-Gaussian distributions can be productive avenues to pursue.

References

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.

Antreas Antoniou and Amos J. Storkey. Assume, augment and learn: Unsupervised few-shot meta-learning via random labels and data augmentation. *CoRR*, abs/1902.09884, 2019. URL http://arxiv.org/abs/1902.09884.

A. Bhattacharya. On a measure of divergence between two statistical populations defined by their probability distributions. *Bulletin of the Calcutta Mathematical Society*, 35:99–109, 1943.

Haoxing Chen, Huaxiong Li, Yaohui Li, and Chunlin Chen. Multi-level metric learning for few-shot image recognition, 2021.

Scott Saobing Chen and Ramesh A. Gopinath. Gaussianization. In Todd K. Leen, Thomas G. Dietterich, and Volker Tresp (eds.), *Advances in Neural Information Processing Systems 13, Papers from Neural Information Processing Systems (NIPS) 2000, Denver, CO, USA*, pp. 423–429. MIT Press, 2000. URL https://proceedings.neurips.cc/paper/2000/hash/3c947bc2f7ff007b86a9428b74654de5-Abstract.html.

Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. *CoRR*, abs/1904.04232, 2019a. URL http://arxiv.org/abs/1904.04232.

Yinbo Chen, Xiaolong Wang, Zhuang Liu, Huijuan Xu, and Trevor Darrell. A new meta-baseline for few-shot learning. *CoRR*, abs/2003.04390, 2020. URL https://arxiv.org/abs/2003.04390.

Zitian Chen, Yanwei Fu, Yinda Zhang, Yu-Gang Jiang, Xiangyang Xue, and Leonid Sigal. Multi-level semantic feature augmentation for one-shot learning. *IEEE Transactions on Image Processing*, 28(9):4594–4605, Sep 2019b. ISSN 1941-0042. doi: 10.1109/tip.2019.2910052. URL http://dx.doi.org/10.1109/TIP.2019.2910052.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1126–1135. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/finn17a.html.

Jerome H. Friedman. Regularized discriminant analysis. *Journal of the American Statistical Association*, 84(405):165–175, 1989. ISSN 01621459. URL http://www.jstor.org/stable/2289860.

Fusheng Hao, Fengxiang He, Jun Cheng, Lei Wang, Jianzhong Cao, and Dacheng Tao. Collect and select: Semantic alignment metric learning for few-shot learning. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8459–8468, 2019. doi: 10.1109/ICCV.2019.00855.

Ruibing Hou, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. Cross attention network for few-shot classification, 2019.

Huaxi Huang, Junjie Zhang, Jian Zhang, Jingsong Xu, and Qiang Wu. Low-rank pairwise alignment bilinear network for few-shot fine-grained image classification. *IEEE Transactions on Multimedia*, 23:1666–1680, 2021. ISSN 1941-0077. doi: 10.1109/tmm.2020.3001510. URL http://dx.doi.org/10.1109/TMM.2020.3001510.

Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Li Fei-Fei. Novel dataset for fine-grained image categorization. In *First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition*, Colorado Springs, CO, June 2011.

Hongyang Li, David Eigen, Samuel F. Dodge, Matthew D. Zeiler, and Xiaogang Wang. Finding task-relevant features for few-shot learning by category traversal. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–10, 2019a.

Wenbin Li, Lei Wang, Jinglin Xu, Jing Huo, Yang Gao, and Jiebo Luo. Revisiting local descriptor based image-to-class measure for few-shot learning. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7253–7260, 2019b. doi: 10.1109/CVPR.2019.00743.

Wenbin Li, Jinglin Xu, Jing Huo, Lei Wang, Yang Gao, and Jiebo Luo. Distribution consistency based covariance metric networks for few-shot learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):8642–8649, Jul. 2019c. doi: 10.1609/aaai.v33i01.33018642. URL https://ojs.aaai.org/index.php/AAAI/article/view/4885.

Bin Liu, Yue Cao, Yutong Lin, Qi Li, Zheng Zhang, Mingsheng Long, and Han Hu. Negative margin matters: Understanding margin in few-shot classification, 2020a.

Jialun Liu, Yifan Sun, Chuchu Han, Zhaopeng Dou, and Wenhui Li. Deep representation learning on long-tailed data: A learnable embedding augmentation perspective. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pp. 2967–2976. Computer Vision Foundation / IEEE, 2020b. doi: 10.1109/CVPR42600.2020.00304. URL https://openaccess.thecvf.com/content_CVPR_2020/html/Liu_Deep_Representation_Learning_on_Long-Tailed_Data_A_Learnable_Embedding_Augmentation_CVPR_2020_paper.html.

Jinlu Liu, Liang Song, and Yongqiang Qin. Prototype rectification for few-shot learning, 2020c.

Yaoyao Liu, Bernt Schiele, and Qianru Sun. An ensemble of epoch-wise empirical bayes for few-shot learning, 2020d.

S. Łukaszyk. A new concept of probability metric and its applications in approximation of scattered data sets. *Computational Mechanics*, 33(4):299–304, January 2004. doi: 10.1007/s00466-003-0532-2.

Puneet Mangla, Nupur Kumari, Abhishek Sinha, Mayank Singh, Balaji Krishnamurthy, and Vineeth N Balasubramanian. Charting the right manifold: Manifold mixup for few-shot learning. In *The IEEE Winter Conference on Applications of Computer Vision*, pp. 2218–2227, 2020.

Erik G. Miller, Nicholas E. Matsakis, and Paul A. Viola. Learning from one example through shared densities on transforms. In *2000 Conference on Computer Vision and Pattern Recognition (CVPR 2000), 13-15 June 2000, Hilton Head, SC, USA*, pp. 1464–1471. IEEE Computer Society, 2000. doi: 10.1109/CVPR.2000.855856. URL https://doi.org/10.1109/CVPR.2000.855856.

Tsendsuren Munkhdalai, Xingdi Yuan, Soroush Mehri, and Adam Trischler. Rapid adaptation with conditionally shifted neurons. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3664–3673. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/munkhdalai18a.html.

Seong-Jin Park, Seungju Han, Ji-Won Baek, Insoo Kim, Juhwan Song, Hae Beom Lee, Jae-Joon Han, and Sung Ju Hwang. Meta variance transfer: Learning to augment from the others. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 7510–7520. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/park20b.html.

Massimiliano Patacchiola, Jack Turner, Elliot J. Crowley, Michael F. P. O'Boyle, and Amos J. Storkey. Bayesian meta-learning for the few-shot setting via deep kernels. In *NeurIPS*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/b9cfe8b6042cf759dc4c0cccb27a6737-Abstract.html.

Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL https://openreview.net/forum?id=rJY0-Kcll.

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

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *CoRR*, abs/1409.0575, 2014. URL http://arxiv.org/abs/1409.0575.

Donald Shepard. A two-dimensional interpolation function for irregularly-spaced data. In *Proceedings of the 1968 23rd ACM National Conference*, ACM '68, pp. 517–524, New York, NY, USA, 1968. Association for Computing Machinery. ISBN 9781450374866. doi: 10.1145/800186.810616. URL https://doi.org/10.1145/800186.810616.

Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 4077–4087, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/cb8da6767461f2812ae4290eac7cbc42-Abstract.html.

Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H. S. Torr, and Timothy M. Hospedales. Learning to compare: Relation network for few-shot learning. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pp. 1199–1208. Computer Vision Foundation / IEEE Computer Society, 2018. doi: 10.1109/CVPR.2018.00131. URL http://openaccess.thecvf.com/content_cvpr_2018/html/Sung_Learning_to_Compare_CVPR_2018_paper.html.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision, 2015.

John W. Tukey. *Exploratory data analysis*. Addison-Wesley series in behavioral science : quantitative methods. Addison-Wesley, 1977. ISBN 0201076160. URL https://www.worldcat.org/oclc/03058187.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. URL http://jmlr.org/papers/v9/vandermaaten08a.html.

John Van Ness. On the dominance of non-parametric Bayes rule discriminant algorithms in high dimensions. *Pattern Recognition*, 12(6):355–368, January 1980. doi: 10.1016/0031-3203(80)90012-6.

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

Yaqing Wang, Quanming Yao, James T. Kwok, and Lionel M. Ni. Generalizing from a few examples: A survey on few-shot learning. *ACM Comput. Surv.*, 53(3):63:1–63:34, 2020. doi: 10.1145/3386252. URL https://doi.org/10.1145/3386252.

Yulin Wang, Xuran Pan, Shiji Song, Hong Zhang, Cheng Wu, and Gao Huang. Implicit semantic data augmentation for deep networks. *CoRR*, abs/1909.12220, 2019. URL http://arxiv.org/abs/1909.12220.

Xiu-Shen Wei, Peng Wang, Lingqiao Liu, Chunhua Shen, and Jianxin Wu. Piecewise classifier mappings: Learning fine-grained learners for novel categories with few examples. *IEEE Transactions on Image Processing*, 28(12):6116–6125, Dec 2019. ISSN 1941-0042. doi: 10.1109/tip.2019.2924811. URL http://dx.doi.org/10.1109/TIP.2019.2924811.

Sanford Weisberg. Yeo-johnson power transformations, 2001.

P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology, 2010.

Wanqi Xue and Wei Wang. One-shot image classification by learning to restore prototypes. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):6558–6565, Apr. 2020. doi: 10.1609/aaai.v34i04.6130. URL https://ojs.aaai.org/index.php/AAAI/article/view/6130.

Shuo Yang, Lu Liu, and Min Xu. Free lunch for few-shot learning: Distribution calibration. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=JWOiYxMG92s.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *CoRR*, abs/1605.07146, 2016. URL http://arxiv.org/abs/1605.07146.

Baoquan Zhang, Xutao Li, Yunming Ye, Zhichao Huang, and Lisai Zhang. Prototype completion with primitive knowledge for few-shot learning, 2021a.

Jiacheng Zhang, Weishuo Liu, Zhicheng Zhao, and Fei Su. Distribution estimation based pseudo-feature library generation for few-shot image classification. In *2021 IEEE International Conference on Multimedia Expo Workshops (ICMEW)*, pp. 1–6, 2021b. doi: 10.1109/ICMEW53276.2021.9455950.

Jian Zhang, Chenglong Zhao, Bingbing Ni, Minghao Xu, and Xiaokang Yang. Variational few-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

APPENDIX

A TUKEY AND YEO-JOHNSON

Tukey transformation on a random vector $\mathbf{x}$ with parameter β is defined as,

$$tukey(\mathbf{x}) = \begin{cases} \mathbf{x}^{\beta} & \beta \neq 0 \\ \log(\mathbf{x}) & \beta = 0 \end{cases} \tag{10}$$

Yeo-Johnson transformation on a random vector $\mathbf{x}$ with parameter β is defined as,

$$yeo\_johnson(\mathbf{x}) = \begin{cases} ((\mathbf{x}+1)^{\beta} - 1)/\beta & \beta \neq 0, \mathbf{x} \geq 0 \\ \log(\mathbf{x}+1) & \beta = 0, \mathbf{x} \geq 0 \\ -[(-\mathbf{x}+1)^{2-\beta} - 1]/(2-\beta) & \beta \neq 2, \mathbf{x} < 0 \\ -\log(-\mathbf{x}+1) & \beta = 2, \mathbf{x} < 0 \end{cases} \tag{11}$$

B EFFECT OF DISTANCES

For normally distributed features, the probability of a novel point $\tilde{\mathbf{x}}$ being in class i with distribution $\mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ can be written quite generally as,

$$p(class = i|\tilde{\mathbf{x}}) \approx p(i) \frac{1}{\sqrt{\mid \boldsymbol{\Sigma}_i \mid}} \exp \frac{-1}{2} (\tilde{\mathbf{x}} - \boldsymbol{\mu}_i)^T \Sigma_i^{-1} (\tilde{\mathbf{x}} - \boldsymbol{\mu}_i), \tag{12}$$

Taking $\ln$ on both sides of this probability, and assuming that each base class is equally probable in priors $p(i)$, we get a natural distance between $\tilde{\mathbf{x}}$ and $\boldsymbol{\mu}_i$ as:

$$d_i = \ln \mid \boldsymbol{\Sigma}_i \mid + (\tilde{\mathbf{x}} - \boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1} (\tilde{\mathbf{x}} - \boldsymbol{\mu}_i) \tag{13}$$

equation 13 is the squared Mahalanobis distance between $\tilde{\mathbf{x}}$ and $\boldsymbol{\mu}_i$ along with an added $\ln$ term. Replacing $\boldsymbol{\Sigma}_i^{-1} = \mathbf{I}$ for all i, our d_i reduces to squared Euclidean distance,

$$d_i = (\tilde{\mathbf{x}} - \boldsymbol{\mu}_i)^T (\tilde{\mathbf{x}} - \boldsymbol{\mu}_i) \tag{14}$$

The above formulations assume that both $\tilde{\mathbf{x}}, \boldsymbol{\mu}_i$ come from the same distribution. Keeping this in mind, we also derive a metric called squared δ distance with δ as a hyperparameter in Formulation 1 (Section B.1) as,

$$d_i = (\mathbf{I}\tilde{\mathbf{x}} - \delta \mathbf{I}\boldsymbol{\mu}_i)^T (\mathbf{I}\tilde{\mathbf{x}} - \delta \mathbf{I}\boldsymbol{\mu}_i) \tag{15}$$

Table 5: Accuracy with different distance measures in Stanford Dogs 5way-1shot task. Note that $m = 1$ in all distances except the scaled Euclidean distance in the last row.

Distance (d_i)	5way-1shot
Squared Mahalanobis with log, equation 13	63.95%
Squared Euclidean, equation 14	64.33%
Squared δ distance, equation 15	64.57%
Scaled Euclidean $m = 1.5$	64.58%

Experimenting with these different distance metrics, in Table 5, we observe no advantage of using Mahalanobis distance or squared δ distance over a simpler Euclidean distance when defining d_i in Section 3.2.2. We also see that the scaled Euclidean with m as a hyperparameter gives improved accuracy compared to Euclidean with $m = 1$. Hence, we propose using our scaled Euclidean distance (equation 6) instead of a searching for multiple distance metrics in the feature space to improve accuracy.

B.1 FORMULATION 1: GENERAL DISTANCE: SQUARED δ

Consider two points $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^d$ sampled from different multivariate distributions $\mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1), \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1), \boldsymbol{\mu}_1, \boldsymbol{\mu}_2 \in \mathbb{R}^d, \boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2 \in \mathbb{R}^{d\times d}$. Multiple distance measures can be constructed between the populations, e.g., the Bhattacharya distance (Bhattacharya, 1943). Here, we consider a general distance of the form

$$\begin{aligned} d &= (\mathbf{L}^{-1}\mathbf{x}_1 - \mathbf{M}^{-1}\mathbf{x}_2)^T(\mathbf{L}^{-1}\mathbf{x}_1 - \mathbf{M}^{-1}\mathbf{x}_2) \\ &where, \;\; \mathbf{L}\mathbf{L}^T = \boldsymbol{\Sigma}_1, \quad \mathbf{M}\mathbf{M}^T = \boldsymbol{\Sigma}_2 \end{aligned} \tag{16}$$

Here $\mathbf{L}, \mathbf{M} \in \mathbb{R}^{d\times d}$ are the cholesky decompositions of $\boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2$ respectively. If $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2$, equation 16 reduces to squared Mahalanobis distance. If $\mathbf{x}_1$ is the support point $\tilde{\mathbf{x}}$ and $\mathbf{x}_2$ is the mean of base class $\mathbf{X}_i$, then $\boldsymbol{\Sigma}_2 = \boldsymbol{\Sigma}_i =$covariance of the base class i. The covariance of the distribution of $\tilde{\mathbf{x}}$ is unknown since we are trying to estimate it. Assuming $\boldsymbol{\Sigma}_1 = \psi\mathbf{I}$, i.e. a diagonal covariance, we get,

$$\begin{aligned} d_i =& (\mathbf{L}^{-1}\tilde{\mathbf{x}} - \mathbf{M}^{-1}\boldsymbol{\mu}_i)^T(\mathbf{L}^{-1}\tilde{\mathbf{x}} - \mathbf{M}^{-1}\boldsymbol{\mu}_i) \\ =& (\sqrt{\psi}\mathbf{I}\tilde{\mathbf{x}} - \mathbf{M}^{-1}\boldsymbol{\mu}_i)^T(\sqrt{\psi}\mathbf{I}\tilde{\mathbf{x}} - \mathbf{M}^{-1}\boldsymbol{\mu}_i) \\ & where, \;\; \mathbf{L}\mathbf{L}^T = \psi\mathbf{I}, \quad \mathbf{M}\mathbf{M}^T = \boldsymbol{\Sigma}_i \end{aligned}$$

From our experiments we observed that the off-diagonal covariances in the base classes of mini-Imagenet, CUB and Stanford Dogs, were at least 2 orders of magnitudes smaller than the diagonal variance. Hence assuming $\boldsymbol{\Sigma}_i = \sigma\mathbf{I}$, i.e. a diagonal matrix with constant variance, we get,

$$\begin{aligned} d_i =& (\sqrt{\psi}\mathbf{I}\tilde{\mathbf{x}} - \sqrt{\sigma}\mathbf{I}\boldsymbol{\mu}_i)^T(\sqrt{\psi}\mathbf{I}\tilde{\mathbf{x}} - \sqrt{\sigma}\mathbf{I}\boldsymbol{\mu}_i) \\ =& \psi(\mathbf{I}\tilde{\mathbf{x}} - \frac{\sqrt{\sigma}}{\sqrt{\psi}}\mathbf{I}\boldsymbol{\mu}_i)^T(\mathbf{I}\tilde{\mathbf{x}} - \frac{\sqrt{\sigma}}{\sqrt{\psi}}\mathbf{I}\boldsymbol{\mu}_i) \\ =& \psi(\mathbf{I}\tilde{\mathbf{x}} - \delta\mathbf{I}\boldsymbol{\mu}_i)^T(\mathbf{I}\tilde{\mathbf{x}} - \delta\mathbf{I}\boldsymbol{\mu}_i) \;\; , where \;\; \delta = \frac{\sqrt{\sigma}}{\sqrt{\psi}} \end{aligned} \tag{17}$$

Since ψ is common to all base classes $\mathbf{X}_i$, it does not affect our closest base class calculation. Hence dropping we ψ we get a general distance as,

$$d_i = (\mathbf{I}\tilde{\mathbf{x}} - \delta\mathbf{I}\boldsymbol{\mu}_i)^T(\mathbf{I}\tilde{\mathbf{x}} - \delta\mathbf{I}\boldsymbol{\mu}_i) \tag{18}$$

where δ is a hyperparameter that can be optimized for accuracy.

C DETAILS OF LOGISTIC REGRESSION

We performed Logistic Regression using `torch` library. For all datasets– miniImagenet, CUB, StanfordDogs and for all experiments, the following hyperparameters were used,

$$\begin{aligned} \text{batch size} &= 1024, \\ \text{epochs} &= 200, \\ \text{learning rate} &= 0.08, \\ \text{optimizer} &= \text{Stochastic Gradient Descent } (\texttt{torch.optim.SGD}), \\ \text{scheduler} &= \text{None}, \\ \text{Loss Function} &= \text{Cross Entropy } (\texttt{torch.nn.CrossEntropyLoss}), \\ \text{Loss Regularization} &= \text{None} \end{aligned}$$

D DATASETS

miniImagenet (Ravi & Larochelle, 2017) is derived from the ILSVRC-12 dataset (Russakovsky et al., 2014). It contains 100 different classes with 600 examples per class. The images sizes are $84 \times 84 \times 3$. We follow the train, validation, test split of 64 base, 16 validation and 20 novel classes as done in previous work of Ravi & Larochelle (2017).

CUB (Welinder et al., 2010) is a fine grained classification dataset consisting of different bird species. Each class has varying number of examples in this dataset so we take the minimum available number of 44 examples from each of 200 classes. The image sizes are again $84 \times 84 \times 3$. Following Chen et al. (2019a) we split train, validation, test as 100 base, 50 validation and 50 novel classes respectively.

Stanford Dogs (Khosla et al., 2011) is another fine grained classification dataset of dogs species derived from ILSVRC-12 dataset. Here again each class has varying number of points so we take 100 points from each of 120 classes. Following existing state-of-the-art results of Chen et al. (2021), we use train, validation and test splits of 70 base, 20 validation and 30 novel classes respectively.

Cross Domain datasets: To show that our method gives superior performance even when the base classes are dissimilar to the novel classes, we evaluate our proposed method by training on tasks sampled from one distribution and evaluating on a different distribution. Specifically, we follow Patacchiola et al. (2020) and show results on **miniImagenet** $\rightarrow$ **CUB**, i.e. train split from miniImagenet and test/val split from CUB. We also compare our method against DC (Yang et al., 2021) on a **meta-tieredImagenet** of 34 broad categories from tieredImagenet, split into 20 base, 8 novel and 6 validation classes, as laid out in Ren et al. (2018). Note that there is a high dissimilarity between the base and novel/validation classes in this meta-tieredImagenet as seen in Table 6.

Table 6: 34 broad categories of tieredImagenet dataset forming a meta-tieredImagenet. We can see that there is a high dissimilarity between the base and novel classes here. The only similar class between base and novel is 'working dog' and 'hound, hound dog'.

#Num	Base class	Novel class	Validation class
1	protective covering, protective cover, protect	obstruction, obstructor, obstructer, impediment	durables, durable goods, consumer durables
2	Garment	geological formation, formation	motor vehicle, automotive vehicle
3	building, edifice	solid	machine
4	establishment	substance	furnishing
5	electronic equipment	vessel	mechanism
6	game equipment	aquatic vertebrate	sporting dog, gun dog
7	Tool	working dog	
8	Craft	insect	
9	ungulate, hoofed mammal		
10	musical instrument, instrument		
11	Primate		
12	feline, felid		
13	hound, hound dog		
14	Terrier		
15	snake, serpent, ophidian		
16	Saurian		
17	passerine, passeriform bird		
18	aquatic bird		
19	restraint, constraint		
20	instrument		

E FEATURE EXTRACTOR BACKBONE

We used S2M2 Method (Mangla et al., 2020) to train a WRN-28-10 (Zagoruyko & Komodakis, 2016) feature extractor for miniImagenet, CUB, Stanford Dogs and meta-tieredImagenet. The backbone was first allowed to overfit for 400 epochs on all base classes to minimize the classification + self supervised rotation loss. Next the decision boundaries were smoothened using the validation classes with a cosine classifier on the feature extractor until the loss on validation classes stopped improving. For our cross domain results on miniImagenet $\rightarrow$ CUB, we used identical setting as (Patacchiola et al., 2020) with a Conv-4 backbone.

F HYPERPARAMETER SEARCH METHODOLOGY

We used `optuna` (Akiba et al., 2019) library to tune our hyperparameters $\beta, m, k, \alpha_1, \alpha_2, n$. For all datasets (miniImagenet, CUB, Stanford Dogs, meta-tieredImagenet and miniImagenet $\rightarrow$ CUB) the search space for β, m, k, n was,

$$\beta \in [low = 0, high = 10, step = 0.25],\ m \in [low = 0, high = 3, step = 0.25],$$
$$k \in [low = 2, high = |\mathbb{C}_b|, step = 2],\ n \in [low = 100, high = 1000, step = 50]$$

where $|\mathbb{C}_b|$ denotes the number of base classes.

For miniImagenet and CUB we set

$$\alpha_1 \in [low = 0, high = 10000, step = 1000], \alpha_2 \in \{0, 0.1, 1, 10, 100\} \times \alpha_1$$

whereas for Stanford Dogs, meta-tieredImagenet, and miniImagenet $\rightarrow$ CUB,

$$\alpha_1 \in [low = 0, high = 1000, step = 100], \alpha_2 \in [low = 0, high = 1000, step = 100] \times \alpha_1$$

Note that a larger range of α_2 was needed in Stanford Dogs owing to the average off-diagonal covariance of $\mathbf{\Sigma}'$ (equation 3) being 2 orders of magnitude smaller than the average variance.

All hyperparameters were jointly tuned using `TPESampler` in `optuna`. For each hyperparameter setting sampled, we evaluated its average accuracy over 200 random tasks $\mathcal{T}$ sampled from the validation classes $\mathbb{C}_v$. During this phase, we pruned any hyperparameter setting which had less than median accuracy after 100 runs using `MedianPruner`. Once every hyperparameter setting was validated, we picked the top-3 candidates from this experiment and ran these specific hyperparameters for 5000 random tasks sampled from the novel classes, i.e. $\mathcal{T} \sim \mathbb{C}_n$. We report our accuracy and mean confidence of the best candidate in comparison with state-of-the-art in Tables 1, 2. Accuracies of all 3 candidates can be found in Appendix G.

G TUNED HYPERPARAMETERS

In Table 7, we give the accuracy of each of the top 3 candidates from our `optuna` hyperparameter search. Note than the first row of every setting (5way-1shot or 5way-5shot) is the result we used for comparing to other methodologies in Tables 1, 2.

Table 7: Accuracy for each of top 3 candidates during our `optuna` hyperparameter search. The accuracies are reported after evaluating on 5000 random tasks sampled from the novel classes, i.e. $\mathcal{T} \sim \mathbb{C}_n$

Dataset	Setting	Accuracy	Hyperparameters
miniImagenet	5way-1shot	73.00 +- 0.50	$m = 1, k = 8, \alpha_1 = 3000, \alpha_2 = 10\alpha_1, \beta = 0.5, n = 750$
		72.80 +- 0.43	$m = 1, k = 8, \alpha_1 = 6000, \alpha_2 = 10\alpha_1, \beta = 0.5, n = 750$
		72.30 +- 0.51	$m = 1, k = 8, \alpha_1 = 2000, \alpha_2 = 10\alpha_1, \beta = 0.5, n = 750$
	5way-5shot	87.22 +- 0.33	$m = 3, k = 30, \alpha_1 = 9000, \alpha_2 = 10\alpha_1, \beta = 0.5, n = 750$
		86.91 +- 0.32	$m = 3, k = 30, \alpha_1 = 8000, \alpha_2 = 10\alpha_1, \beta = 0.5, n = 750$
		86.82 +- 0.39	$m = 3, k = 30, \alpha_1 = 7000, \alpha_2 = 10\alpha_1, \beta = 0.5, n = 750$
CUB	5way-1shot	84.57 +- 0.48	$m = 1, k = 4, \alpha_1 = 8000, \alpha_2 = 10\alpha_1, \beta = 0.5, n = 750$
		84.49 +- 0.50	$m = 1, k = 6, \alpha_1 = 10000, \alpha_2 = 10\alpha_1, \beta = 0.5, n = 750$
		84.22 +- 0.46	$m = 1, k = 4, \alpha_1 = 9000, \alpha_2 = 10\alpha_1, \beta = 0.5, n = 750$
	5way-5shot	93.46 +- 0.25	$m = 2, k = 4, \alpha_1 = 5000, \alpha_2 = 10\alpha_1, \beta = 0.5, n = 750$
		93.16 +- 0.21	$m = 1, k = 10, \alpha_1 = 2000, \alpha_2 = 100\alpha_1, \beta = 0.5, n = 750$
		93.15 +- 0.26	$m = 1, k = 16, \alpha_1 = 3000, \alpha_2 = 100\alpha_1, \beta = 0.5, n = 750$
StanfordDogs	5way-1shot	65.35 +- 0.61	$m = 1.5, k = 10, \alpha_1 = 400, \alpha_2 = 100\alpha_1, \beta = 1, n = 750$
		65.12 +- 0.61	$m = 1.75, k = 10, \alpha_1 = 500, \alpha_2 = 100\alpha_1, \beta = 1, n = 750$
		64.91 +- 0.58	$m = 1.5, k = 12, \alpha_1 = 500, \alpha_2 = 100\alpha_1, \beta = 1, n = 750$
	5way-5shot	80.56 +- 0.45	$m = 1.5, k = 12, \alpha_1 = 300, \alpha_2 = 200\alpha_1, \beta = 1, n = 750$
		80.12 +- 0.46	$m = 1.25, k = 12, \alpha_1 = 300, \alpha_2 = 400\alpha_1, \beta = 1, n = 750$
		80.01 +- 0.41	$m = 1.25, k = 12, \alpha_1 = 100, \alpha_2 = 200\alpha_1, \beta = 1, n = 750$
miniImagenet $\rightarrow$ CUB	5way-1shot	41.08 +- 0.53	$m = 0.5, k = 64, \alpha_1 = 400, \alpha_2 = 100\alpha_1, \beta = 0.5, n = 750$
		39.11 +- 0.51	$m = 0.5, k = 62, \alpha_1 = 500, \alpha_2 = 100\alpha_1, \beta = 0.5, n = 750$
		39.02 +- 0.58	$m = 2.5, k = 10, \alpha_1 = 400, \alpha_2 = 100\alpha_1, \beta = 0.5, n = 750$
	5way-5shot	54.69 ± 0.41	$m = 0.5, k = 18, \alpha_1 = 100, \alpha_2 = 100\alpha_1, \beta = 0.5, n = 750$
		52.12 +- 0.45	$m = 0.5, k = 12, \alpha_1 = 100, \alpha_2 = 100\alpha_1, \beta = 0.5, n = 750$
		49.14 +- 0.44	$m = 1.5, k = 18, \alpha_1 = 100, \alpha_2 = 200\alpha_1, \beta = 0.5, n = 750$
meta-tieredImagenet (tieredImagenet)	5way-1shot	43.51 ± 0.50	$m = 2.25, k = 8, \alpha_1 = 100, \alpha_2 = 100\alpha_1, \beta = 1, n = 750$
		41.12 +- 0.61	$m = 2.0, k = 10, \alpha_1 = 100, \alpha_2 = 100\alpha_1, \beta = 1, n = 750$
		40.01 +- 0.58	$m = 1.5, k = 12, \alpha_1 = 200, \alpha_2 = 100\alpha_1, \beta = 1, n = 750$
	5way-5shot	56.79 ± 0.64	$m = 2.0, k = 10, \alpha_1 = 100, \alpha_2 = 100\alpha_1, \beta = 1, n = 750$
		53.13 +- 0.66	$m = 2.25, k = 12, \alpha_1 = 200, \alpha_2 = 100\alpha_1, \beta = 1, n = 750$
		50.01 +- 0.61	$m = 1.25, k = 12, \alpha_1 = 100, \alpha_2 = 200\alpha_1, \beta = 1, n = 750$

H ADDITIONAL T-SNE VISUALIZATION

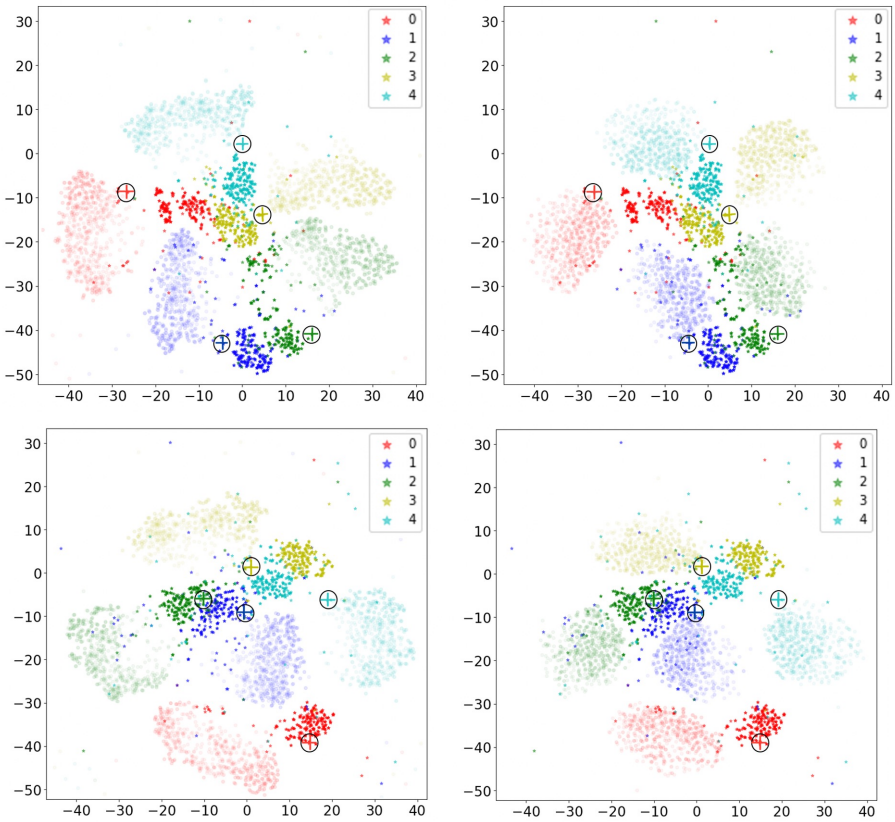

Figure 4: t-SNE visualization for 5 randomly sampled novel classes for DC (left) and our method DC+ (right) for miniImagenet, supplementing the visualization in the main text section. The support points are indicated with a '+' sign within black circles, sampled points are semi-transparent indicated with a 'o' sign, and the query points are opaque denoted by '*'. Note that our sampled points (o) are on average closer to the ground truth query points (*).

I META-TIEREDIMAGENET

We used the same S2M2 method for training the backbone feature extractor, the feature dimensions were set to 64 with no activation function. Hyperparameter search ranges were,

$\beta \in [low = 0, high = 10, step = 0.25]$, $m \in [low = 0, high = 3, step = 0.25]$,

$k \in [low = 2, high = |\mathbb{C}_b|, step = 2]$

$\alpha_1 \in [0, 1000, step = 100]$, $\alpha_2 \in [0, 1000, step = 100] \times \alpha_1$

In Table 3, we see that DC+ also outperforms DC and baseline (no method) on meta-tieredImagenet even when there is a high dissimilarity between the base and novel/validation classes. The hyperparameters corresponding to these results were,

DC+ 5way-1shot $m = 2.25, k = 8, \alpha_1 = 100, \alpha_2 = 100, \beta = 1, n = 750$

DC+ 5way-5shot $m = 2, k = 10, \alpha_1 = 100, \alpha_2 = 100, \beta = 1, n = 750$

DC 5way-1shot $k = 2, \alpha = 1000, \beta = 1, n = 750$

DC 5way-5shot $k = 2, \alpha = 900, \beta = 1, n = 750$

An explanation as to why our method outperforms DC on this meta dataset is because of the high m ($m = 2.25$ in 5way-1shot and $m = 2$ in 5way-5shot) our method discovers. With such high m,

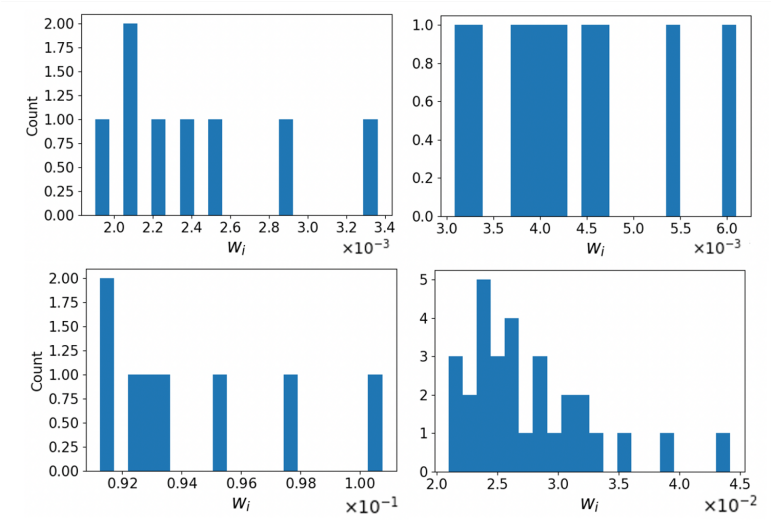

Figure 5: Histogram visualization of weights w_i for meta-tieredImagenet (top) and miniImagenet (bottom) for both 5way-1shot (left) and 5way-5shot (right) tasks. Note how the weights in meta-tieredImagenet are at least an order of magnitude smaller than miniImagenet.

Table 8: w_i for k base classes in meta-tieredImagenet experiment

Method	5way-1shot	5way-5shot
DC+	[0.0034, 0.0030, 0.0025, 0.0024, 0.0023, 0.0022, 0.0021, 0.0019]	[0.0064, 0.0056, 0.0049, 0.0047, 0.0044, 0.0042, 0.0041, 0.0039, 0.0034, 0.0032]
DC	[0.33, 0.33]	[0.33, 0.33]

our w_i is 100 times smaller than the weights assigned by DC method as seen in Figure 5. Hence very small contribution of the k base classes are extrapolated in DC+ method while calculating the calibrated $\boldsymbol{\mu}'$, $\boldsymbol{\Sigma}'$, compared to DC where a hard contribution of 1/(k+1) exists.

Figure 5 and Table 8 shows the weights of the k base classes calculated by our method DC+ and DC method averaged over 5 random selection of 5way-K shot task (random selection of novel classes and support points)

To put these w_i into perspective, the w_i for best hyperparameters in miniImagenet are 10 times larger as shown in Figure 5 and Table 9, ($m = 1, k = 8$ for 5way-1shot, and $m = 3, k = 30$ for 5way-5shot) showing that the classes are much more similar in miniImagenet than in the current meta-tieredImagenet.

Table 9: w_i for k base classes in miniImagenet experiment

Method	5way-1shot	5way-5shot
DC+	[0.1008, 0.0979, 0.0953, 0.0935, 0.0928, 0.0924, 0.0916, 0.0912]	[0.0442, 0.0393, 0.0355, 0.0332, 0.0323, 0.0318, 0.0307, 0.0303, 0.0298, 0.0291, 0.0287, 0.0281, 0.0273, 0.0267, 0.0264, 0.0261, 0.0258, 0.0255, 0.0253, 0.0250 0.0244, 0.0242, 0.0239, 0.0236, 0.0234, 0.0231, 0.0225, 0.0218, 0.0214, 0.0210]