COMP6248 REPRODUCIBILITY CHALLENAGE FREE LUNCH ON FEW-SHOT LEARNING

Authors: Zheyu Zhang , Yifei Wei , Wenhan Wei, Tiehan Yu
Email:{zz12g21, yw28n20, ww1g21, ty1g21}@soton.ac.uk

ABSTRACT

The few-shot learning is a challenging task and captures lots of attention due to its practicability where the model learns from only a limited number of samples. Yang et al. (2021) proposed a distribution calibration method which utilises the class from base classes where the data samples are sufficient to estimate the class distribution from the support set. This report aims to reproduce some of the results from this paper. Our reproduction code can be found here:https://github.com/COMP6248-Reproducability-Challenge/NAMO_Free_Lunch_For_Few_Shot_Learning; The github repository of original author is here: https://github.com/ShuoYang-1998/Few_Shot_Distribution_Calibration.

1 INTRODUCTION

1.1 PROBLEM STATEMENT

The standard few-shot classification method is stated below:
Given a dataset of data-label combos, $D = \{(x_i; y_i)\}$, where $x_i \in \Re^d$ is the sample's feature vector and $y_i \in C$ is the class (target) set. This collection of classes is separated into base classes C_b and novel classes C_n, where $C_b \cap C_n = \emptyset$. There are only a few labelled examples available for each task T to evaluate the model's quick adaption capacity or generalisation ability.
An N-way-K-shot task presented by Vinyals et al. (2016) is the most common method to construct a task, in which N classes are sampled from the novel set C_n and only K (e.g., 1 or 5) labelled samples are provided for each class. The model is assessed on another query set $Q = \{(x_i, y_i)\}_{i=N\times K+1}^{N\times K+N\times q}$, where each class in the job contains q test cases, and the few accessible labelled data is called support set $S = \{(x_i, y_i)\}_{i=1}^{N\times K}$. As a result, a model's performance is measured by its averaged accuracy over (the query set of) several jobs drawn from the novel classes.

1.2 TARGET QUESTIONS

The goal is to train a model using data from the N-way-K-shot support set such that it can generalise well on the selected query set. Our targets in this paper are:

- Reproducing the t-SNE scatter plot between the sample generated by the calibrated distribution and the ground truth samples to see how good the calibrated distribution estimates the real distribution.
- Verify the effect of the hyper-parameters, especially the value of power λ in the Tukey transformation stage.
- Validate the effect of the distribution calibration over the non-augmented data.
- Validate if the calibrated sampled support set produces a similar accuracy as the result from the paper over the query set.

1.3 DATASETS

CUB dataset is the most widely used benchmark for fine-grained image classification, which has 200 classes of birds with the size of 84 × 84 × 3. The paper follow Chen's work Antoniou & Storkey (2019), diving the dataset into 100 base classes and 50 novel classes.

miniImageNet data set has 100 classes. And for each class, contains 608 images. The size is also 84 × 84 × 3. Following Ravi and Larochelle's work Ravi & Larochelle (2016), diving the dataset into 64 base classes and 20 novel classes.

1.4 DISTRIBUTION CALIBRATION

The distribution calibration is using the C_b to calibrate the distribution of C_n, in order to augment the support set. Firstly, it uses the Tukey's Ladder of Powers transformation (equation (1)) to transfer the feature distribution more Gaussian-like.

$$\tilde{x} = \begin{cases} \mathbf{x}^{\lambda} & \text{if } \lambda \neq 0 \\ log(\mathbf{x}) & \text{if } \lambda = 0 \end{cases} \tag{1}$$

Then, it transfers the statistics from the base class with the mean and variance to the novel classes. The distance is calculated based on the Euclidean distance between the novel classes and the base classes with the feature space and the mean of the features with the formula (2).

$$\mathbb{S}_d = \left\{ - \left\| \mu_i - \tilde{x} \right\|^2 \mid i \in C_b \right\} \tag{2}$$

The mean and covariance of the distribution in novel class are calibrated by the statistics from the top k nearest base classes with formula (3).

$$\mathbb{S}_N = \left\{ i \mid - \left\| \boldsymbol{\mu}_i - \tilde{\boldsymbol{x}} \right\|^2 \in \operatorname{top} k \left(\mathbb{S}_d \right) \right\} \tag{3}$$

So the calibrated mean and variance can be updated with the formula (4).

$$\mu' = \frac{\sum_{i \in \mathbb{S}_N} \mu_i + \tilde{x}}{k+1}, \boldsymbol{\Sigma}' = \frac{\sum_{i \in \mathbb{S}_N} \boldsymbol{\Sigma}_i}{k} + \alpha \tag{4}$$

2 Experimental and re-implementation methodology

For the code posted by the author, it does not contain all the code reproduce all the results mentioned in the paper. Thus, some experiments are re-implemented such as the calibrated distribution. As the paper instructed, the 640 dimension data is projected to two-dimension using sklearn t-SNE method and visualized.

For datasets, we used CUB and miniImageNet as applied in the paper. The method proposed in the paper used a transfer learning method where features of the penultimate layer is extracted and classified by a linear classifier afterward. The training stage for these two datasets are extremely resource-consuming and time-consuming, especially for the large dataset like miniImageNet which requires $166GB$ space and endless training time for ECS GPU server. Fortunately, the default backbone model WideResNet_S2M2_Rotation (Zagoruyko & Komodakis, 2016) has been trained and the models pre-trained on the CUB and miniImageNet are provided in the author's source code. This model is leveraged in the following tests:

Both 5-way-1-shot and 5-way-5-shot setting are respectively tested among datasets. Moreover, the difference between linear classifiers are compared in terms of accuracy.

There are 3 main hyper-parameters mentioned in the paper, namely, the number of retrieved base class statistics k (equation(3)), Values of power λ in Tukey transformation (equation(1)), degree of dispersion of features α(equation(4)). Due to the time limit, only the effect of λ is tested. We set a range of λ and extend the experiments with dataset CUB which is not shown in the paper before. In terms of the hyper-parameters for the rest of the experiments, we use the same value as the paper for all datasets where the number of generated features $n = 750, k = 2, \lambda = 0.5, \alpha = 0.21$ for miniImageNet and 0.3 for CUB. The experimental details will be on the next section.

3 Reproduced experimental results and analysis

3.1 Distribution calibration

The results of distribution estimation are shown in this part. The triangular in the plots represent the support point of each class. In Figure 1, plot (a) (c) shown the estimate distribution of miniImageNet by using t-SNE and PCA separately . Plots (b) (e) show the origin miniImageNet data through the t-SEN and PCA separately. The plots (e),(d),(g), and (h) follow the same structure as above which (e),(g) was the estimated result. (f),(h) are the origin CUB data. From these results, it can be observed that the scalar of the estimate data in (a),(c),(e), and (g) is different from the original data in (b),(d),(f),(h). Compare (a),(e) with (b),(f), the sample data from the estimate distribution after t-SNE is different from the original data distribution by using t-SNE. We didn't get the same result as the original paper shown. Therefore, we try to use PCA to process these data, and the results of the estimated data and original data are still different. From (d) and (h), the mean and cov of the distribution are different. However, in the generated data, the mean and cov from the different classes are similar. We found the distribution of estimate and original data in the paper have the same scale and the support data shown in the same position. Therefore, we put the generated and original data together to make dimensional reduction by using t-SNE and PCA to force the support point shown in the same positions. The structure of Figure 2 was same as Figure 1. Although the support point shown in the same position, the estimate distribution is still totally different from the original distribution. And the PCA results show that the original data choose the wrong principal components. Therefore, we fail to reproduce the original paper distribution results. We think the estimate distribution is not similar to the original data.

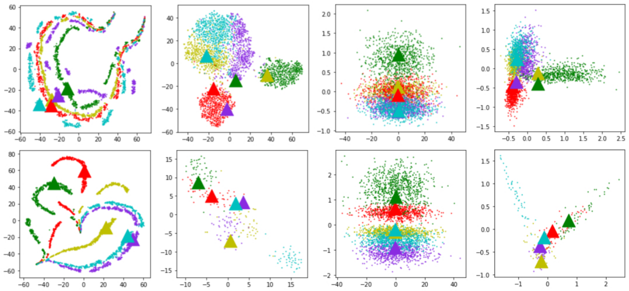


Figure 1: t-SNE and PCA visualization of query data and estimate data for CUB and miniImageNet

(a) (b) (c) (d)

(e) (f) (g) (h)

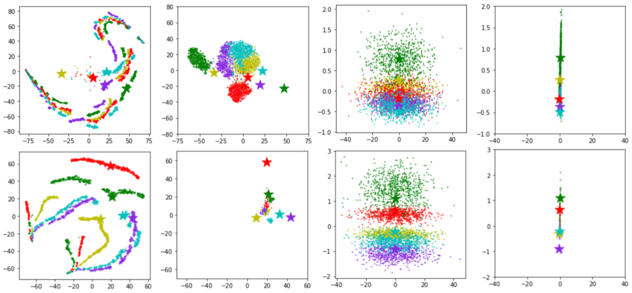
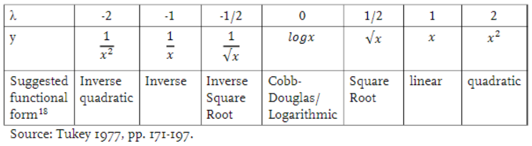


Figure 2: t-SNE and PCA visualization of query data and estimate data for CUB and miniImageNet

(i) (j) (k) (l)

(m) (n) (o) (p)



λ	-2	-1	-1/2	0	1/2	1	2
y	$\frac{1}{x^2}$	$\frac{1}{x}$	$\frac{1}{\sqrt{x}}$	$log x$	$\sqrt{x}$	x	x^2
Suggested functional form[18]	Inverse quadratic	Inverse	Inverse Square Root	Cobb-Douglas/ Logarithmic	Square Root	linear	quadratic

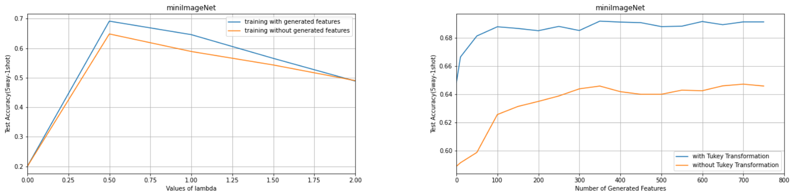
Source: Tukey 1977, pp. 171-197.

Figure 3: Tukey's Ladder of Powers (λ) Range

3.2 Tukey's transformation and hyper-parameter λ

The main calibration technique used in this paper is Tukey's Ladder of Powers(1), and we have attempted to reproduce the relationship between the hyperparameter (λ) and the test accuracy of the model in the paper (Figure 3 in the original). It is an excellent method for reducing the skewness of the data distribution and makes the data more like a normal distribution. There is no upper limit to the value of λ, but the usual range of values is [-2, 2]. When the data distribution has a positive skew it is usually calibrated by taking values ($\lambda <= 0$). For example, when ($\lambda = 0$) values taken Logarithmic function will have a large effect on larger values while smaller values will not change much. When the data distribution has a negative skew, the distribution is calibrated by an exponential transformation ($\lambda >= 2$). Lastly, ($\lambda = 1$) does not change the data distribution.

In the reproduction of this Figure[4a], we were unable to implement the experimental result with $\lambda <= 0$ using the source code and the code would report an error. However, we managed to reproduce the full case of miniImageNet $\lambda > 0$ and the result is almost identical to the original. We then replicated (Figure[4b]) the difference in accuracy between calibrated and uncalibrated when sampling different sample sizes to see if the results were consistent with the author. One point to note is that the distribution of data on the axes of the original author's plot is uneven, giving a visually confusing picture of the results. In the reproduction, we filled in all the missing data in the middle. The results show a slight increase in accuracy as the number of samples sampled becomes larger. The key, however, is whether calibration was performed or not, and the results are significantly worse without it. The final results were generally consistent.



(a) Reproduction of Figure 3 Left

(b) Reproduction of Figure 3 Right

Figure 4: Reproduction of Figure 3

3.3 Classification

Dataset	Classifier	N-way-K-shot	Test Accuracy	Expected Test Accuracy
CUB	LR	5-way-5-shot	90.18	90.67 ± 0.35
CUB	LR	5-way-1-shot	76.67	79.56 ± 0.87
CUB	SVM	5-way-5-shot	89.47	90.26 ± 0.98
CUB	SVM	5-way-1-shot	76.67	79.49 ± 0.33
miniImageNet	LR	5-way-5-shot	81.60	82.30 ± 0.34
miniImageNet	LR	5-way-1-shot	70.53	68.57 ± 0.55
miniImageNet	SVM	5-way-5-shot	81.33	82.88 ± 0.42
miniImageNet	SVM	5-way-1-shot	68.67	67.31 ± 0.83

Table 1: 5way1shot and 5way5shot classification accuracy (%) on miniImageNet and CUB with 95% confidence intervals.

We have reproduced the classification test accuracy results of the original authors on both datasets and they almost match the data provided.

4 Discussion and Conclusion

Our result reproduces most of the results stated in the original paper, except the distribution visualisation. The distribution scatter gives a very confusing plot differ from the promised Gaussian-like scatter, which informs a lack of reproducibility. Moreover, there are still some questions about the paper itself that remain:

- In the distribution calibration stage, this paper assumes that all classes follow a Multivariate Gaussian distribution, which is a pretty strong assumption. For datasets like miniImageNet, or CUB, it works. For realistic datasets, many of them may not follow a strict Gaussian-like distribution, which indicates a poor generalisation of harder datasets.
- The statistic transferring requires the calibrating base classes have sufficient data to form a decent Gaussian distribution. It also requires the classes between novel class and base class have high similarities. This also indicates a poor generalisation of smaller and more diverse datasets.

References

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

Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. 2016.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, koray kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

Shuo Yang, Lu Liu, and Min Xu. Free lunch for few-shot learning: Distribution calibration. 2021. doi: 10.48550/ARXIV.2101.06395. URL https://arxiv.org/abs/2101.06395.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.