2019-1 Deep Learning Homework #5

Fabian Tan Hann Shen (ID 20185112)

June 7, 2019

1.

(a)

To generate the sinusoidal wave, we have to configure the amplitude and the phase in the code. Therefore, I set the amplitude within 0.1 and 5.0, and the phase within 0 and π, as shown in the following code below.

```
if FLAGS.datasource == 'sinusoid':
            self.generate = self.generate_sinusoid_batch
            self.amp_range = config.get('amp_range', [0.1, 5.0])
            self.phase_range = config.get('phase_range', [0, np.pi])
            self.input_range = config.get('input_range', [-5.0, 5.0])
```

(b)

The model was trained with 70000 iterations and iterated for different K-shot example, which are 1-shot, 5-shot and 10 shot respectively. The loss is the mean-squared error between the prediction and the true value. Table. 1 shows the MSE for each K-shot on the sinusoidal function. As shown in the Table. 1, the performance of MAML using k=5 on the sinusoid function has the least MSE compared to other K-shot, ie. 1-shot and 10-shot. The mean square error value is 0.0782, it performs better for 5-shot, among these K-shot experiments.

The formula for Mean Squared Error:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{Y_i} - Y_i)^2$$

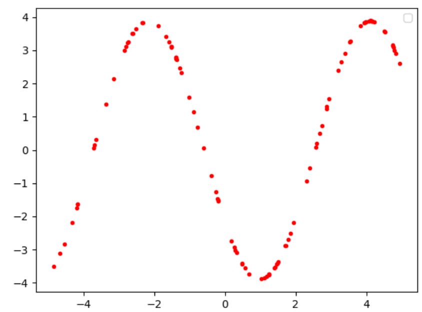


Figure 1: shows the sinusoidal graph of k=1 learning

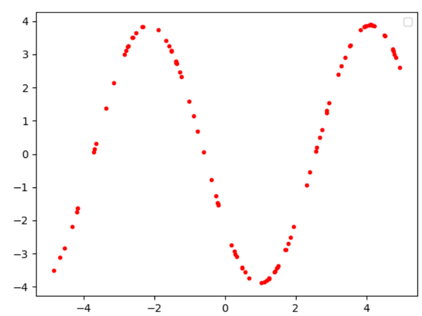


Figure 2: shows the sinusoidal graph of k=5 learning

Table 1: Mean Square Errors for each K-shot learning of the sinusoid data

Function		1-Way	
	1-shot	5-shot	10-shot
Sinusoid	0.5034 $\pm$ 0.1900	0.0782 $\pm$ 0.0390	0.9944 $\pm$ 0.2780

(c)

Sinusoidal graph for k=1,5 and 10 shot are shown in Figure 1, Figure 2 and Figure 3.

(d)

MAML's work for classification was performed on the Omniglot dataset, as referred to the transpose of MNIST dataset. Omniglot dataset comprises of 20

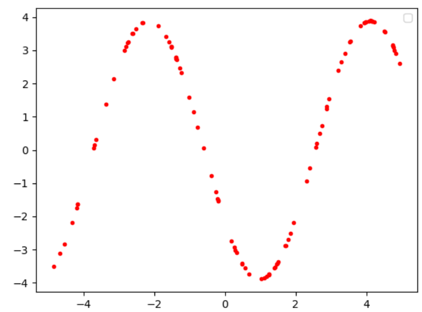


Figure 3: shows the sinusoidal graph of k=10 learning

instances of 1623 characters from 50 different alphabets. Each instance was drawn by different person. The problem of N-way classification is set up as: select N unseen classes, feed the model with different K-examples of each N classes.

In this experiment, 1-shot and 5-shot for each 5-way and 20-way learning is performed. Results comparing the k-shot and n-way on Omniglot are shown in Table. 2. The model was trained with 60000 iterations, with the meta batch size of 32 and learning rate starting with 0.4. As illustrated in Table. 2, we can see that the 5-way, 1-shot has better accuracy among the multiple experiments.

Under 1-shot experiment, it has 98.8±0.04% on 5-ways, compared to the 20-ways, which has only 88.4±0.5%. This experiment shows that it is able to produce better accuracy by feeding small number of classes to the model. However, for the 5-shot experiment, I was not able to attain the data successfully, because the learning of the parameter on the inner gradient update was stuck on the first few meta update iteration. Working on them by attempting to tune other parameters like learning rate, or optimizer and etc.

Table 2: Result for each K-shot learning on the Omniglot dataset. The ± shows the 95% confidence intervals over tasks

Dataset	**5-Ways Accuracy**		**20-Ways Accuracy**	
	1-shot	5-shot	1-shot	5-shot
Omniglot	$98.8 \pm 0.4\%$	-	$88.4 \pm 0.5\%$	-

2.

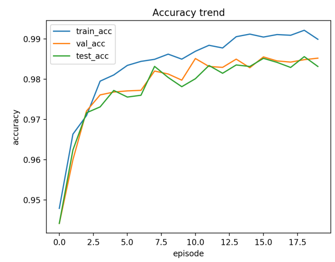
We also evaluate with another approach - Meta-SGD on the omniglot dataset and compare their performance. For an N-way,K shot classification task, N classes are sampled and then in each class sample K examples for training and 15 other for testing. The meta learner is updated for each batch of tasks. After performing meta-training, the model is tested with unseen classes from the meta-testing dataset.

Table.3 shows the accuracy and the loss on the Omniglot dataset with the meta-SGD approach. As shown in the table, it has the better accuracy when using 5-way compared to the 20-ways, which are nearly 0.99. It can be observed that feeding that model with small amount of classes can produce higher accuracy on the classification task. The model performs better when it is trained with 1-shot task during meta-training than 5-shot tasks. These are observed in both 5-way and 20-way classification. In addition, feeding the model with higher shot example, ie. 5-shot examples gives better accuracy. This can be seen in both 5-way and 20-way classification tasks.

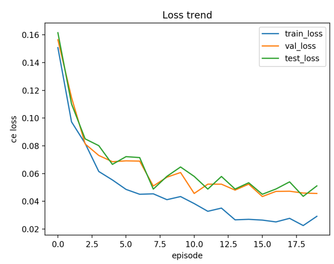
Table 3: Result for each N-way, K-shot on the Omniglot dataset with Meta-SGD

Metric	**5-Ways**		**20-Ways**	
	1-shot	5-shot	1-shot	5-shot
Accuracy	0.983	0.993	0.926	0.972
Loss	0.051	0.024	0.241	0.089

Whereas in Figure 4 to Figure 7 shows the accuracy and loss graph for each combination of N ways and K examples.

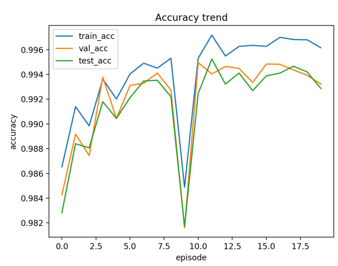


(a) Accuracy graph



(b) Loss graph

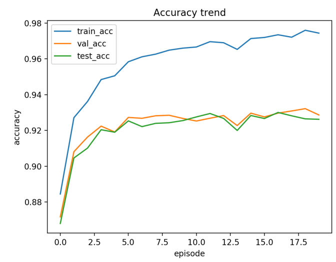
Figure 4: shows the accuracy/loss graph for n=5, k=1 learning



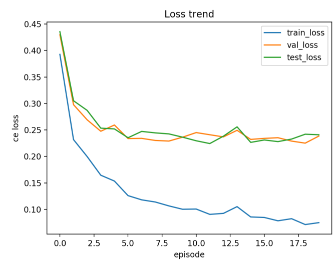
(a) Accuracy graph

(b) Loss graph

Figure 5: shows the accuracy/loss graph for n=5, k=5 learning

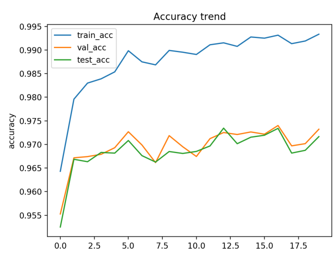


(a) Accuracy graph



(b) Loss graph

Figure 6: shows the accuracy/loss graph for n=20, k=1 learning



(a) Accuracy graph

(b) Loss graph

Figure 7: shows the accuracy/loss graph for n=20, k=5 learning