COMP6248 REPRODUCIBILITY CHALLENGE ACCELERATING SGD WITH MOMENTUM FOR OVERPARAMETERIZED LEARNING

Jacob Neale (jmdn1g17@soton.ac.uk)
Qingyang Chen (qc3m20@soton.ac.uk)
Meiyu Zhao (mz1f20@soton.ac.uk)

ABSTRACT

This report aims to reimplement a new optimiser named Momentum-added Stochastic Solver (MaSS) which can accelerate SGD with momentum for over-parameterized learning. The selected paper pointed out that there is a non-acceleration issue in the SGD with Nesterov which can be solved by MaSS with a compensation term. The reimplemention includes verifying the performance of MaSS and comparing it with other optimisers such as SGD, SGD+Nesterov, and Adam.

1 INTRODUCTION OF THE SELECTED PAPER

The selected paper aims at introducing the MaSS optimizer to solve the non-acceleration issue in the SGD+Nesterov algorithm. The paper points out that MaSS outperforms SGD, SGD+Nesterov and Adam both in optimization and generalization on different architectures of deep neural networks including convolutional networks and ResNet. It has been proved through experimentation that the MaSS algorithm is guaranteed to never have a slower convergence rate than SGD. Furthermore, in the same Gaussian setting MaSS matches the optimal accelerated full-gradient Nesterov rate. The pseudo code of MaSS is as follows:

Algorithm 1 Momentum added Stochastic Solver (MaSS)

1: **Require**: Step size η_1, secondary step size η_2, acceleration parameter $\gamma \in (0,1)$.
2: **Initialize**: $u_0 = w_0$.
3: **while** not converged **do**
4: $\quad w_{t+1} \leftarrow u_t - \eta_1 \widetilde{\nabla} f(u_t)$,
5: $\quad u_{t+1} \leftarrow (1+\gamma) w_{t+1} - \gamma w_t - \eta_2 \widetilde{\nabla} f(u_t)$.
6: **end while**
7: **Output**: weight w_t

2 TARGET QUESTIONS

The stochastic version of Nesterov's acceleration method (SGD+Nesterov) is widely used to train modern machine learning models in practice due to the good reputation of acceleration of the deterministic Nesterov's method. However, there may be a non-accelerating issue in this method. This is because to make sure of convergence, the step size of SGD+Nesterov has to be much smaller than the optimal step size of SGD, which negates the benefits brought by the momentum term. The author of the selected paper pointed out that an algorithm named Momentum-added Stochastic Solver (MaSS) can solve this non-accelerating issue by making SGD+Nesterov have the same size of steps as SGD. The target questions of this reimplemention aims to prove that MaSS is indeed better than other optimisers, including SGD, SGD+Nesterov, and Adam, and evaluating the convergence speed and accuracy of MaSS on the real data.

3 EXPERIMENTAL METHODOLOGY

We redeployed the methods used in the selected paper to compare the results of reimplementing and original paper. The optimisation performance of SGD, SGD+Nesterov, Adam, and MaSS has been compared on the

following real data:
1. Using a fully connected network (FCN) to classify the MNIST dataset.
2. Using a convolutional neural network (CNN) and ResNet to classify the CIFAR-10 dataset. What needs to be noticed is that a batch normalization (BN) layer was inserted after each convolution computation, and the dropout layer in the fully-connected phase was removed.
3. Linear regression on the MNIST dataset.

4 IMPLEMENTATION

According to the available source code provided by the original paper, we reimplemented the MaSS optimizer and used different learning rates to evaluated the performance. Note that only the optimiser code (MaSS optimiser) was copied from the original paper, and FCN, CNN and Linear regression as well as evaluations and experiments of all models were reimplemented for the purpose of this paper. In addition, we chose the typical momentum parameter (y=0.9) for both SGD+Nesterov and MaSS. All algorithms are implemented with mini-batches of size 64 for neural network training. On both CNN and ResNet-32 models, we initialize the learning rate of SGD, SGD+Nesterov, and MaSS by using the same value. The learning rate of Adam was set of the common default setting 0.001. The total epochs that models were trained for is 100.

5 ANALYSIS AND DISCUSSION

5.1 TESTING CONVERGENCE SPEED

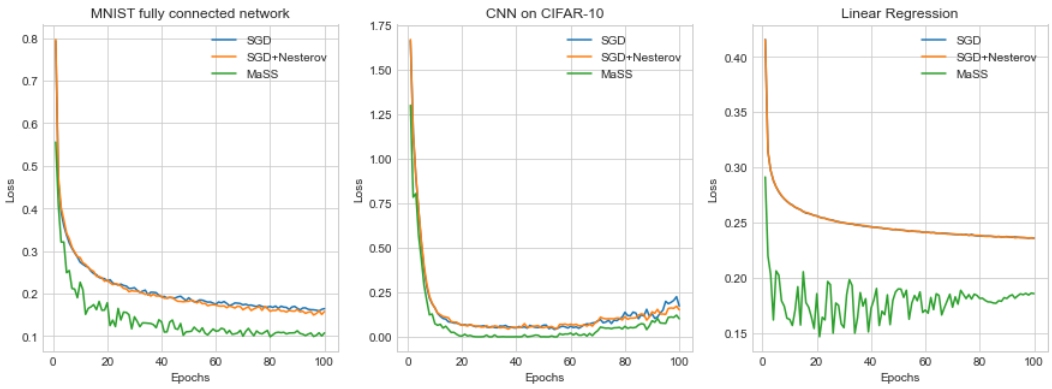

Figure 1: Comparison of the loss value of SGD, SGD+Nesterov, and MaSS on (left) fully connected neural network, (middle) convolutional neural network, and (right) linear regression.

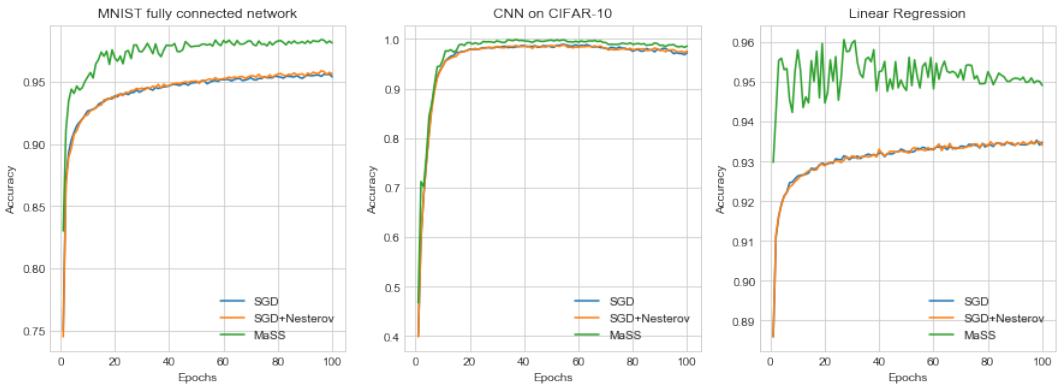

Figure 2: Comparison of the accuracy value of SGD, SGD+Nesterov, and MaSS on (left) fully connected neural network, (middle) convolutional neural network, and (right) linear regression.

The loss curves and accuracy curves of SGD, SGD+Nesterov and MaSS on the real optimise task are shown in figure 1 and figure 2. It is clear to see that MaSS indeed accelerates and converges on the real tasks. More specifically, MaSS improved the optimiser's performance a lot when using the fully connected network (FCN) to classify the MNIST dataset and using linear regression to classify the MNIST dataset. MaSS also has a slight improvement in optimisation when using the convolutional neural network (CNN) to classify the CIFAR-10 dataset. In this experiment, we also confirmed that the non-acceleration issue of SGD+Nesterov proposed by the author of the original paper does exist. It can be seen from the figure that in the three real optimisation tasks, the performance of SGD+Nesterov is almost the same as that of SGD and the general performance of MaSS is better than them with a smaller loss and greater accuracy. Therefore, we can say that at the same number of iterations, the MaSS has the best optimisation performance, which is consistent with the results in the original paper. MaSS indeed solves the non-acceleration issue of SGD+Nesterov and non-convex optimisation problems on neural networks.

5.2 Influence on algorithms with different learning rates

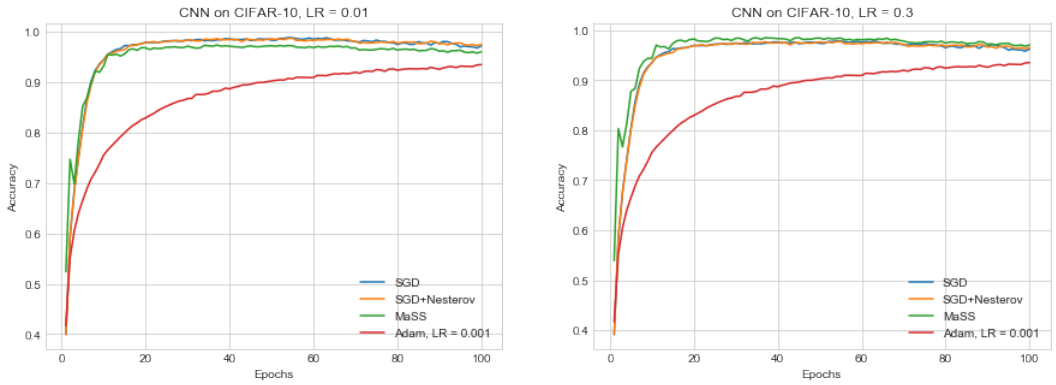

Figure 3: Comparison of the classification accuracy of SGD, SGD+Nesterov, MaSS and Adam(under the default) on CNN within different learning rate (left)lr=0.01 (right) lr=0.3

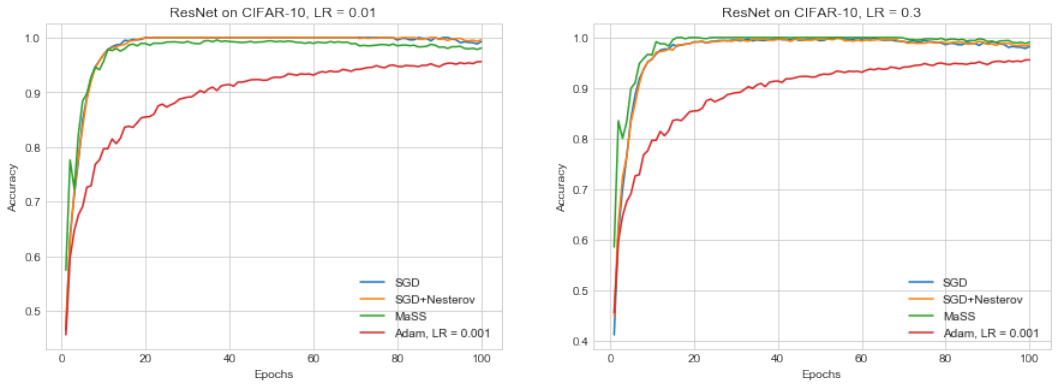

Figure 4: Comparison of the classification accuracy of SGD, SGD+Nesterov, MaSS and Adam(under the default) on ResNet within different learning rate (left)lr=0.01 (right) lr=0.3

Table 1: Comparison of the classification accuracy of these algorithms on the test set of CIFAR-10 (average of 3 independent runs).

	η	SGD	SGD+Nesterov	MaSS	Adam
CNN	0.01	97.24%	97.57%	96.08%	65.71%
	0.3	96.24%	96.57%	97.09%	
RESNET	0.01	99.24%	99.57%	98.09%	96.80%
	0.3	98.24%	98.58%	99.09%	

The classification accuracy of MaSS is evaluated and compared with SGD, SGD+Nesterov, and Adam on different neural network models (CNN and ResNet). In each task, the learning rates of SGD, SGD+Nesterov, and MaSS are the same, while the learning rate of Adam is set to default. Apart from comparing the performance of these 4 optimisers, we also change the learning rate to see the influence of increasing learning rate on SGD, SGD+Nesterov, and MaSS. The result has been shown in figure 3, figure 4, and table 1.

It can be seen that the MaSS optimizer has the best performance when the learning rate is 0.3. While when the learning rate is 0.01, the MaSS has worse accuracy than both SGD and SGD+Nesterov which disagrees with the results of the original paper: The MaSS algorithm always has higher accuracy than SGD and SGD+Nesterov. In addition, the paper result shows that increasing the learning rate can improve the performance of SGD and MaSS, but decreases that of SGD+Nesterov. While table 1 shows that increasing the learning rate can only improve the MaSS's performance slightly, and for the other two algorithms it has a negative impact. As for the Adam optimizer, we can say that it has significantly different performance in the different neural network models, for example in this classification problem, Adam has much better accuracy on ResNet than on the CNN. By contrast, MaSS's performance is much more stable.

6 CONCLUSION

As the original paper pointed out that when the batch size is small, MaSS can optimize the real tasks with a faster convergence speed, which solves the non-accelerated problem in SGD+Nesterov and non-convex optimisation problems on neural networks. In this aspect, the same result is obtained as in the original paper. However, by testing the accuracy of the validation dataset, we observe that the MaSS optimizer does not always performs better than SGD and SGD+Nesterov, especially when the learning rate is lower, which differs slightly from the result of the original paper. In conclusion, fully utilising a combination of sources allowed for a successful reimplementation and reproduction of the original paper in its core aspects of proving MaSS outperforms SGD, SGD+Nesterov and Adam.

7 REFERENCES

[1]Liu, C. and Belkin, M., 2020. Accelerating SGD with momentum for over-parameterized learning.
[2]Team, K., 2020. Keras Documentation: Optimizers. [online] Keras.io. Available at: https://keras.io/api/optimizers/ [Accessed 25 May 2020].

8 APPENDIX

A GITHUB REPOSITORY

The GitHub repository of the reimplemented code and experiments can be found at:

https://github.com/COMP6248-Reproducability-Challenge/
COMP6248-Reproducibility-Challenge-MaSS-Optimiser