Experimental Report on Hyper-parameter Optimization

Xuedong Shang

No Institute Given

Abstract.

1 Introduction

The algorithms being considered here for the moment are Hyperband [1], a Bayesian-based approach TPE [2], two hierarchical approaches HOO [3], HCT [4], and a baseline method random search. Current datasets used are MNIST dataset and some datasets from UCI machine learning dataset archive.

2 MNIST

We consider here logistic regression, multi-layer perceptron and convolutional neural networks as classifiers to be hyper-optimized. This part of code (code for classifers with eventual usage of GPU) is based on code available at `http://deeplearning.net/`.

Hyper-parameters The hyper-parameters to be optimized are listed below in Table 1, 2 and 3. For logistic regression, the hyper-parameters to be considered are learning rate and mini-batch size (since we are doing mini-batch SGD). For MLP, we take into account an additional hyper-parameter which is the l_2 regularization factor. For CNN (or LeNet), we take into account the number of kernels used in the two convolutional-pooling layers.

Hyper-parameter	**Type**	**Bounds**
learning_rate	$\mathbb{R}^+$	$\left[10^{-3}, 10^{-1}\right]$ (log-scaled)
batch_size	$\mathbb{N}^+$	$[1, 1000]$

Table 1. Hyper-parameters to be optimized for logistic regression with SGD.

Dataset The MNIST dataset is pre-split into three parts: training set D_{train}, validation set D_{valid} and test set D_{test}.

Hyper-parameter	Type	Bounds
learning_rate	$\mathbb{R}^+$	$\left[10^{-3}, 10^{-1}\right]$ (log-scaled)
batch_size	$\mathbb{N}^+$	$[1, 1000]$
l_2_reg	$\mathbb{R}^+$	$\left[10^{-4}, 10^{-2}\right]$ (log-scaled)

Table 2. Hyper-parameters to be optimized for MLP with SGD.

Hyper-parameter	Type	Bounds
learning_rate	$\mathbb{R}^+$	$\left[10^{-3}, 10^{-1}\right]$ (log-scaled)
batch_size	$\mathbb{N}^+$	$[1, 1000]$
k_2	$\mathbb{N}^+$	$[10, 60]$
k_1	$\mathbb{N}^+$	$[5, k_2]$

Table 3. Hyper-parameters to be optimized for CNN with SGD.

Resource Allocation The type of resource considered here is the number of epochs, where one epoch means a pass of training through the whole training set using SGD. Note that this is similar to the original Hyperband paper where one unit of resources corresponds to 100 mini-batch iterations for example. One epoch may contain a various number of mini-batch iterations depending on the mini-batch size.

3 UCI Datasets

References

1. Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., & Talwalkar, A. (2016). Hyperband: A novel bandit-based approach to hyperparameter optimization. arXiv preprint arXiv:1603.06560.
2. Bergstra, J. S., Bardenet, R., Bengio, Y., & Kgl, B. (2011). Algorithms for hyper-parameter optimization. In Advances in neural information processing systems (pp. 2546-2554).
3. Bubeck, S., Munos, R., Stoltz, G., & Szepesvri, C. (2011). X-armed bandits. Journal of Machine Learning Research, 12(May), 1655-1695.
4. Azar, M. G., Lazaric, A., & Brunskill, E. (2014). Online stochastic optimization under correlated bandit feedback. In Proceedings of the 31st International Conference on Machine Learning (ICML-14) (pp. 1557-1565).