

An Astonishing Title

Emanuele Cosenza
e.cosenza3@studenti.unipi.it

Riccardo Massidda
r.massidda@studenti.unipi.it

ML course, 2019/2020.

January 8, 2020

Type A project.

Abstract

Multi-layer perceptron with different regulations techniques to avoid overfitting issues. The model selection and the assessment of the learning process are validated by using the cross-validation method.

Introduction

The presence of different techniques to improve the performances of an artificial neural network, requires the use of formal methods to validate the effectiveness of the proposed improvements. Implementing from the ground up both the network and the validation techniques has lead to the execution of different experiments to motivate the choices made during the whole software life cycle.

The proposed neural network is a multilayer perceptron designed to be user-configurable as much as possible, without forcing any design choice in the code but instead allowing a big variety of combinations to be tested independently. The learning of the weights in the network is done by using the back-propagation algorithm¹, some variations have been introduced in the update rule to permit some enhancements.

One of this variation, is given by the possibility of using the momentum technique² to accelerate the learning process. Also the L2 regularization, known also as Tikhonov regularization, requires adding a penalty for undesirable weights in the update rule.

The model also offers the possibility of early stopping³, since it is a recognized

good regularization technique, and furthermore it reduces the computational time by not learning for more epochs than required.

To perform the model selection has been implemented a mechanism to automatically execute grid search over a family of possible models. Each relevant model generate by iterating on the grid can then be validated by using an hold-out approach or cross-validation depending on the requirements of the experiment. If the early stopping technique is used during training, the cross-validation approach also computes the mean of the stop epochs to approximate the optimal epochs number to train the final model.

The estimation of the risk, or model assessment, to evaluate the generalization power of the selected model can be computed by using a separate test set or by the double cross-validation algorithm. Also in this case depending on the nature of the experiment one of this approaches, or even both, have been chosen.

To expect the achievement of generalization all of the experiments assume a certain degree of smoothness in the source producing the data, respecting so the inductive bias of neural networks

Method

The numerical computational needs are addressed by NumPy⁴. The topology of the network is represented using a list of layers. The i -th weight matrix represents the weights connecting the nodes of the layer i to the nodes in the layer $i + 1$, that is the outgoing edges for the layer i and the ingoing edges for $i + 1$.

The prediction is coded by simply forwarding the input in the network. During the training phase the input is shuffled to avoid ordering bias^{???}, after this the scan of the data is done in a minibatching fashion. Grid search is implemented as a function capable to automatically perform the Cartesian product over the set of relevant values for each hyperparameter.

Experiments

The implementation of the network has been tested against the MONKS datasets, showing how a good balance between the hyper-parameters must be found using formal validation techniques.

Monk's Results

Cup Results

Conclusions

References

1. Rumelhart, D. E. & McClelland, J. L. *Parallel distributed processing: Explorations in the microstructure of cognition*. (MIT Press, 1986).
2. Goodfellow, I., Bengio, Y. & Courville, A. *Deep learning*. (The MIT Press, 2016).
3. Prechelt, L. Early stopping but when? 15.
4. Oliphant, T. E. *Guide to NumPy*. (Continuum Press, 2015).