Deep Learning HW1

Tumino Stefano

April 2024

We tried to implement the network suggested from the paper, composed by one hidden layer with two hidden units, and just one output. This was done using fully conntected layers.

The used loss function is the Mean Square Error, while the optimizer is the classical Stochastic Gradient Descent using a weigth decay of 0.9 and a learning rate of 0.1.

The batch size is equal to the size of the dataset, which is 64. As a test set we use the whole train-set, due to the lack of data. In the end, the parameters are initialized uniformly between -0.3 and 0.3.

From the implementation of the network suggested from the paper, we can notice that the obtained model is nothing else than a dummy classifier, because classify all the elements with the same class.

This is probably due to the high umbalancing of the data, which will contain just 64 rows, from which just 8 are simmetric, which means 12.5% of the dataset. We have also that the parameters seems to converge all to the same value.

This problem is solvable increasing the learning rate, this because the loss function is probably contains several local minima. Increase the learning rate can increase the probability to go out from the local minima, but give also the possibility to bounce without reaching the global minimum. To help this process to reach better minima, we can use the momentum which will reduce the bouncing through the walls.

We can see that using this approach the model is able to reach the maximum accuracy and obtain simmetrical parameters. These, of course, are different from the ones found from the paper, but they are consistent with their results.

So we can conclude that the parameters and strategies suggested from the paper are not enough to find the requested results, because the model will be stuck into local minima. The solutions can be found using some strategies which

push the parameters away from the minima introducing some noise, e.g. which could be a smaller batch, but in this case we probably should balance the dataset otherwise we could obtain batches without simmetric values, or, like seen before, introducing the momentum.