Foundations of Data Mining ($2IMM20$)
Homework Assignment $3B$
Hyperparameters Optimization

Decebal Mocanu

Release date: October 16, 2019.
Guided Self-Study session: October 23, 2019.
Deadline: November 1, 2019.

Preamble

This homework can be solved using Python. Please submit your solutions in a *.zip file[1] through Canvas. The solution needs to contain a report in the pdf format to discuss and present your results. Optionally it can contains also the files with code, but the code will not be used for grading. State clearly in the report the group number and the group members.

Problem

In this homework, you are requested to perform hyperparameters optimization to try understanding the behavior of feed forward neural networks. Starting from the MLP implementation that you made for HW3A and the dataset used for HW3A please solve each of the following subproblems and give clear explanations for your solutions.

I. Hyperparameters Optimization (50 points)

1. (10 points) *Parameter initialization.* Choose the best performing model from HW3A in terms of classification accuracy. For this model, initialize all its parameters (connection weights and biases) to zero and retrain it using all the other settings from HW3A. Report the performance (e.g. loss function over training epochs, classification accuracy and confusion matrix on the validation set after training) of this new trained model. Do you observe any interesting behavior for this new trained model? Discuss the performance of this new trained model in comparison with the performance of the best model from HW3A.

2. (10 points) *Learning rate vs parameter initialization.* Retrain several times for the same fixed amount of epochs (e.g. 100 - this value is your choice) the best model from HW3A using exactly the same settings as in HW3A, with the exception of the learning rate and the initialization of the connection weights and biases which have to be different for each retraining. Start with a very small learning rate (e.g. 0.0001) and after that gradually increase it (e.g. next values can be .001, 0.01, ...). Initialize the connection weights and biases by sampling from a normal distribution $\mathcal{N}(0, \sigma^2)$. σ^2 has to take the following values $\{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$. Make a heatmap[2] where the x-axis represents σ^2, the $y-axis$ represents the learning rate, while the colors represent the accuracy[3] obtained on the validation set after each training. Each element in the heatmap matrix represents practically the accuracy obtained with a specific learning rate and a specific initialization. Discuss the heatmap results.

[1] no rar, no 7z, no tar, no tgz, no nonsense, just send a zip.
[2] https://en.wikipedia.org/wiki/Heat_map
[3] For convenience, please use 0 for the cases when Python gives error during training.

3. (10 points) *Activations.* Choose the worst (one which still trains and does not give errors during training) and the best performing models from Subproblem 2 in terms of classification accuracy. For both models, visualize the hidden neurons activations for each of the two hidden layer three times (after connection initialization before to start the training process, at half of the training process, and at the end of the training process). Visualization hints (it is optional to follow them, please feel free to be creative): you can compute the activations on the validation set where the data points are ordered per class; you can use a heatmap where x-axis represents the hidden neuron id, the y-axis represents the data points id, and the colors represent the activations. Discuss the results.

4. (20 points) *Hyperparameters Optimization.* On the dataset from HW3A please train a MLP model on which you perform hyperparameters optimization. For this specific subproblem, you are not constraint any-more to use just two hidden layers, and 10 neurons per hidden layer. You are free to vary any hyperparameter (e.g. learning rate, number of hidden neurons per layer, type of activation function) you consider to be important to maximize the accuracy on the validation set. Please make a Parallel Coordinates Plot[4] to study the effect of the hyperparameters choice on the performance. The colors of the lines which connect various hyperparameter settings represent the accuracy on the validation set. Discuss the results. For the worst and best performing models, please make a plot with the values of the loss function computed separately over the training and validation sets respectively. Discuss the generalization performance (non exclusive suggestions: overfitting, underfitting) of these two extreme models.

   You can find an example for classification accuracy of a Parallel Coordinates Plot here
   https://docs.microsoft.com/en-us/azure/machine-learning/service/how-to-tune-hyperparameters

II. Peer Review paragraph (0 points)

Finally, each group member must write a single paragraph outlining their opinion on the work distribution within the group. Did every group member contribute equally? Did you split up tasks in a fair manner, or jointly worked through the exercises. Do you think that some members of your group deserve a different grade from others?

Success!

[4] https://en.wikipedia.org/wiki/Parallel_coordinates