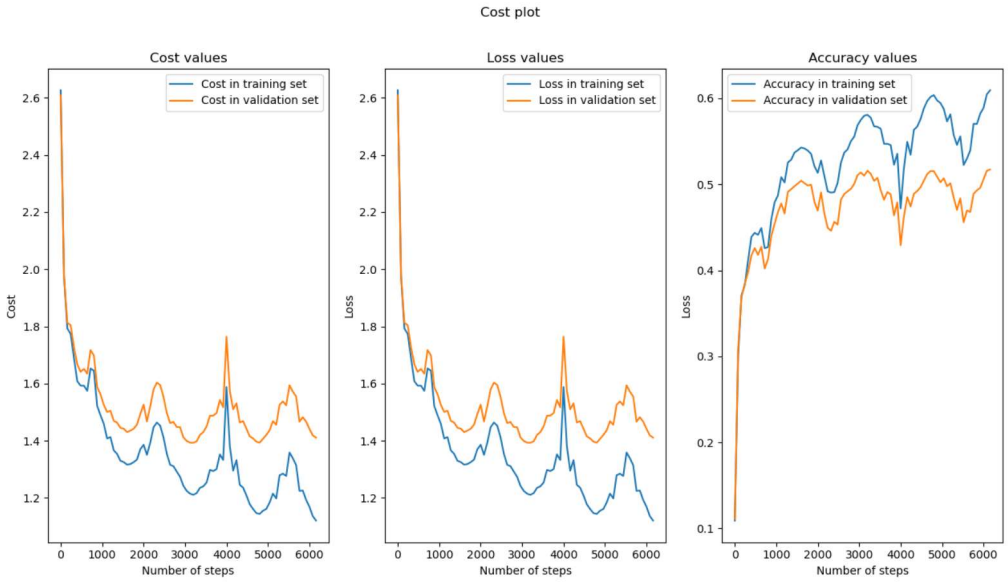
Assignment 2: Deep Learning in Data Science

Sebastián Barbas Laina

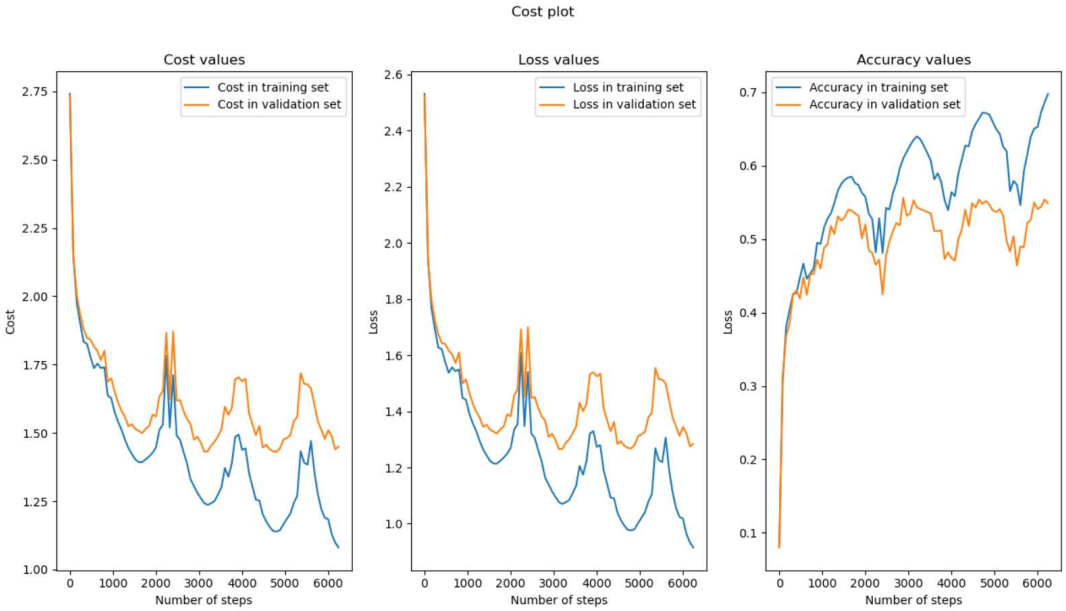
ssbl@kth.se

The first improvement done was to do a graph search to find the best hyperparameters. In order to find this, several hyperparameter values where tested while also doing the coarse to fine shrinkage penalty. For the shrinkage penalty, first a coarse search was done, where 10 random samples were tested within the range between 1e-1 and 1e-5. Afterwards, the two lambdas that gave the biggest validation accuracies where picked, set as the maximum and minimum values for the fine lambda search and another ten random values between these limits were picked. For the rest of the hyperparameters, a grid search was done. For the batch size the tested values were 50 or 100, for the number of cycles the values were 2, 4 and 6 and for the number of steps per cycle: 500 and 800. The search was done with all the dataset as the training dataset and 1000 features from the five sets for training as the validation dataset. The best hyperparameters found were: 100 for the batch size, 4 for the number of cycles, 0.00026227 for the shrinkage penalty and 800 for the number of steps for half a cycle, achieving an accuracy of 51.03% in the test dataset and the following cost and loss values:

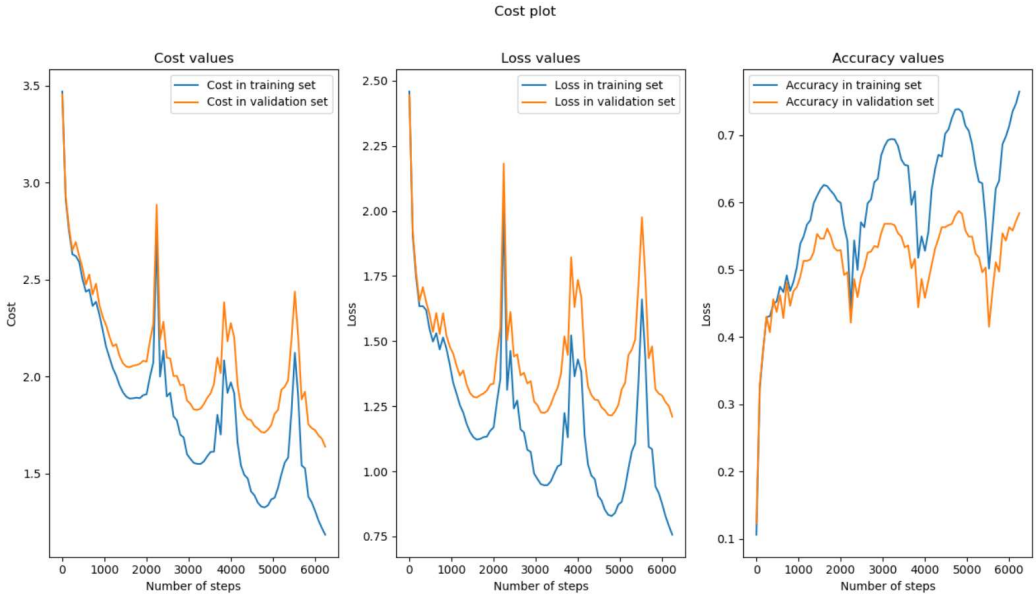


The next experiment performed was with 200 and 1000 nodes in the hidden layer, both with a shrinkage penalty of 0.001 and 4 cycles.

For the experiment with 200 nodes, we had an accuracy of 55.19% in the test dataset of and a loss and cost function of:

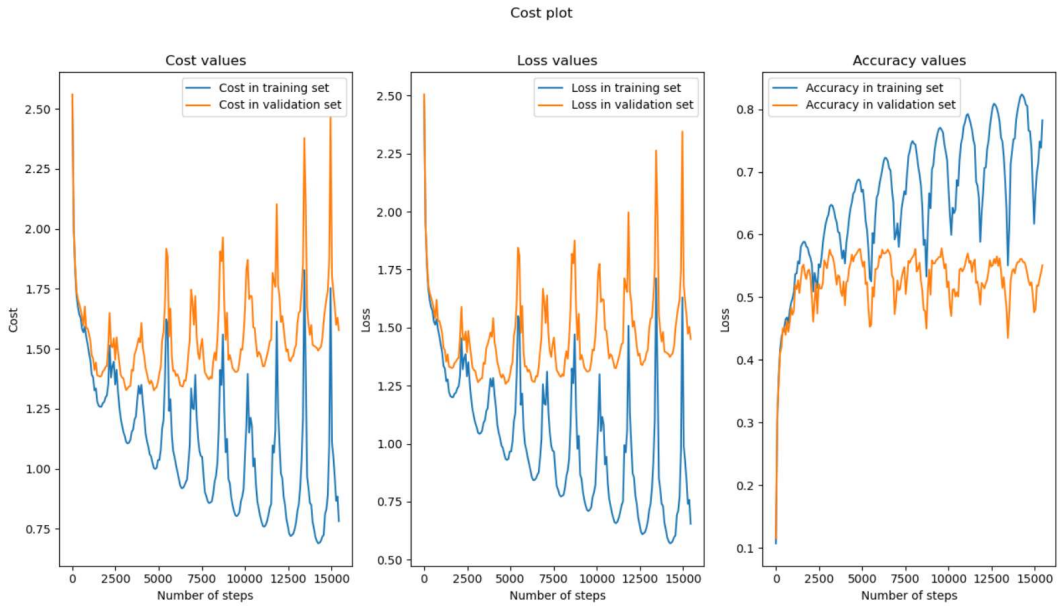


If the same experiment was repeated with 1000 nodes, an accuracy of 56.64% was achieved with a cost and loss function of:



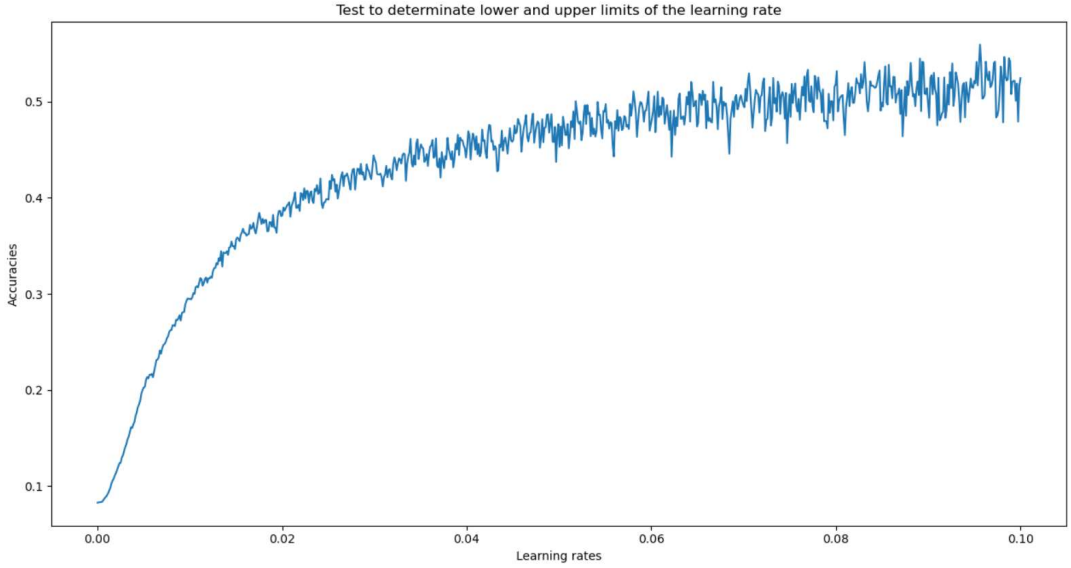
For better tuning, a graph search with a random search could have been done in order to improve the accuracies, although this was discarded due to computational times.

The final improvement was performing an ensemble learning with 200 nodes in the hidden layer. For this method, ten cycles of learning were done. In each cycle, at the end of it, a weak classifier is saved. Once finished the cyclic learning, ten weak learners will have been saved and then all the testing samples are classified with each classifier. Then, the most voted is selected as the correct class. The accuracy obtained was of 35.74%. During the training, this was the cost and loss function obtained:



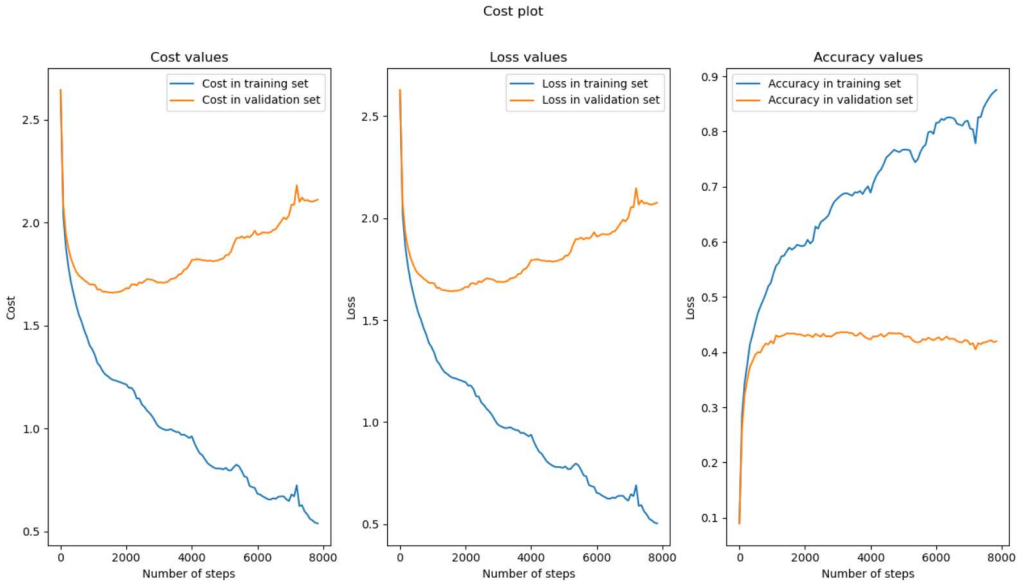
Part 2:

According to Leslie. N Smith (1), in order to determine the upper and lower bounds of the cyclical learning, a training over an ascending value of learning rates had to be done. After this,



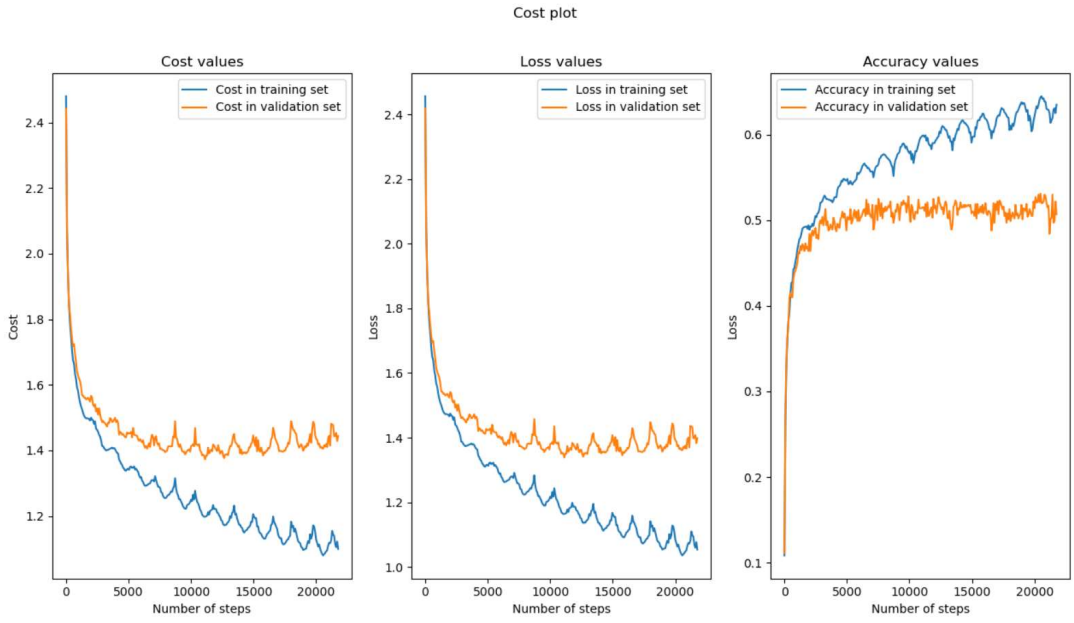
According to the same paper, the maximum value of the cyclic learning rate should be when the accuracy curve starts to lose its steepness and the lower value either when the curve starts to increase sharply or applying $\min = high/n$ with n being 3 or 4 (being this a rule of thumb). The maximum value should be around 0.019, as the curve stops growing as steep as before. As for the lower boundary, the curve starts gaining accuracy with a very los value, so around 0.00225.

With these new values, the training results like this using data-batch 1 as training batch and training batch 2 as validation:



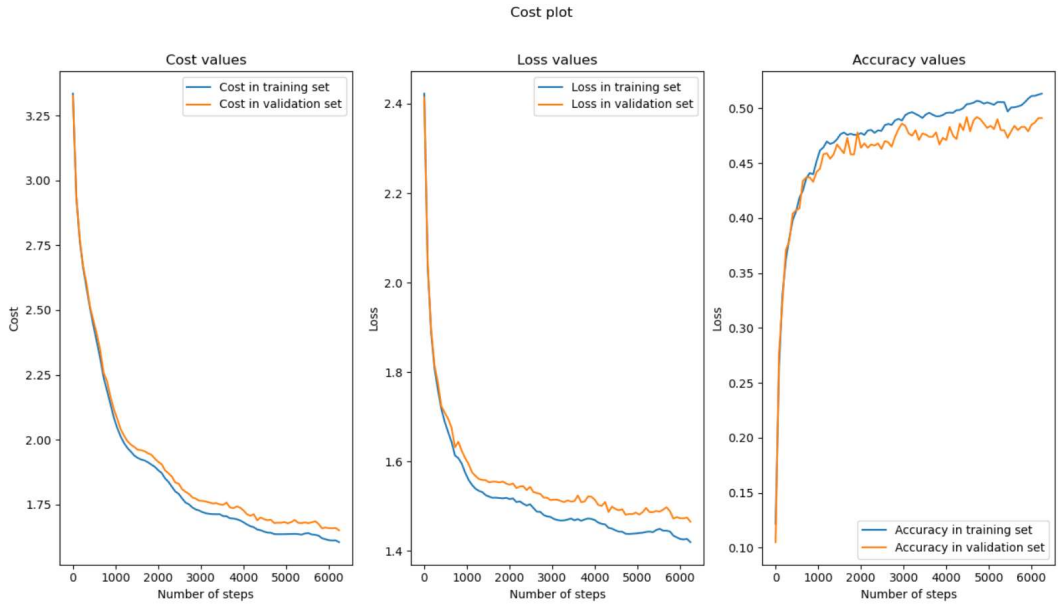
We can see that there is overfitting.

If the network learns for the whole dataset as training except for the last thousand features, which will be used as validation dataset, then the cost and loss functions are the following, with an accuracy in the test dataset of 50.17%:

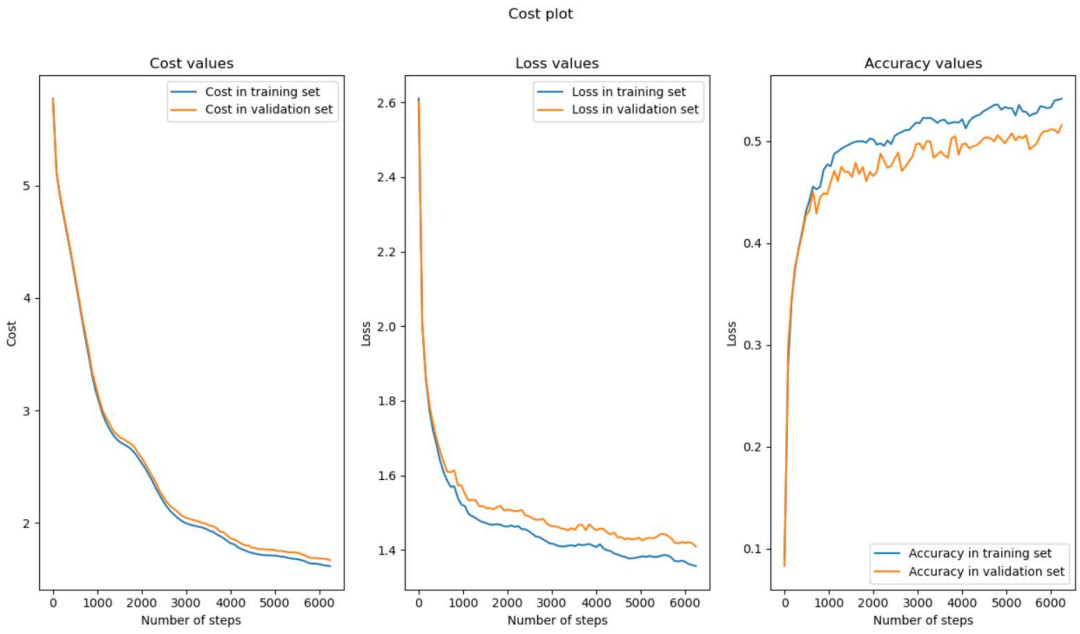


To improve these results, a new value for the shrinkage penalty had to be found via a coarse to fine random search ,which yielded an optimum value for the shrinkage penalty of 0.01504577.

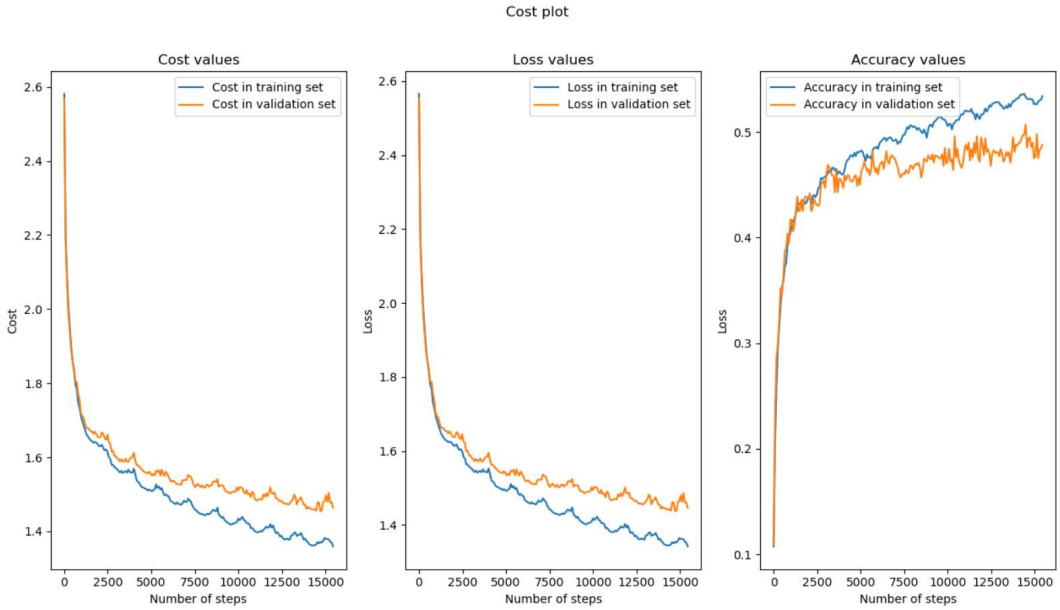
After changing this parameter value, the obtained performance was of 49.01%



With 200 nodes in the hidden layer, we obtain an accuracy of 51.23% and the following graph:

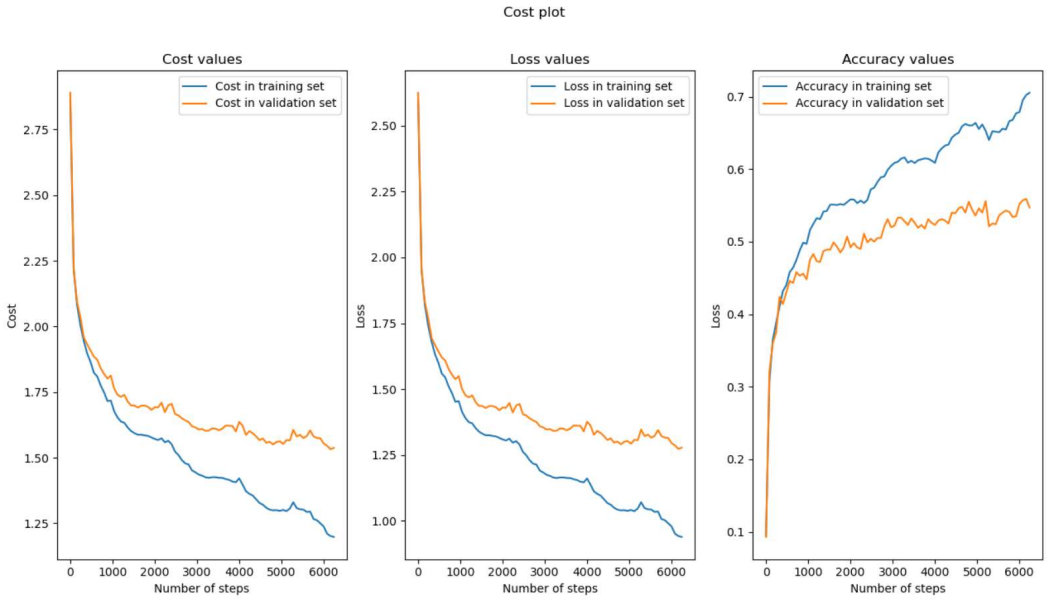


Another interesting test which could be performed is training the model that overfitted with dropout. After this, the obtained cost and loss graphs was:



And the accuracy on the test dataset was of 48.66%.

The final test which is going to be performed is for 1000 nodes in the hidden layer, achieving an accuracy of 54%:



The hyperparameters for the one thousand nodes were the ones previously derived from the first neural network. A better search could have been done but due to computational costs it was not done.

References:

1. **Smith, Leslie N.** Cyclical learning rates for training neural. 2015. arXiv:1506.01186 [cs.CV].