

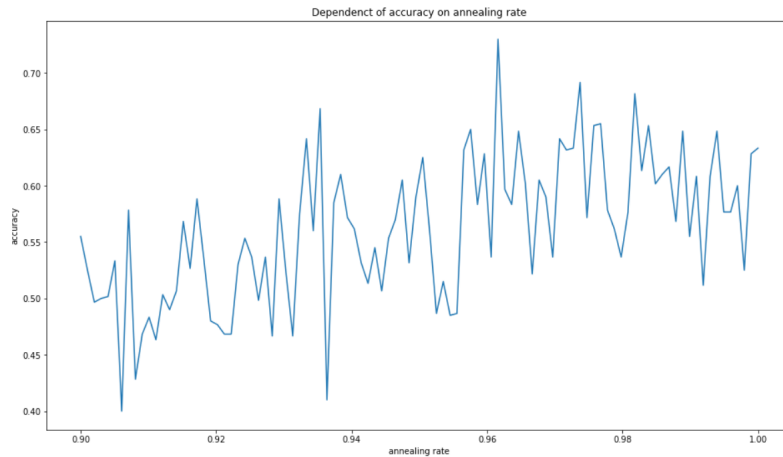
PMLDL - HW 1 report

Nikolay Gaivoronskiy

All the code was written on Python and neural network part was written using PyTorch.

1 Task 1

To evaluate the models, I splitted the dataset into train and validation sets, with proportion 80% and 20% for train and validation sets respectively. I defined a model as a two dense linear layer model. The input to the model are the features of a single object and the output of the model are the raw scores for each of the classes. The loss function being optimized during training is CrossEntropy and the loss for validation is Accuracy. Firstly, I trained the model via backpropagation. In 10 epochs the accuracy became 1, i.e. traditional method works perfectly. As far as training via simulated annealing is concerned, I ran it with different values of annealing rate, running several times with different random seeds and taking average for a fixed annealing rate. Here is the chart of accuracy on a validation set given different annealing rates:



It can be seen from the graph that the annealing performance highly depends on random seed.

2 Task 2

In this task, I took 30 most populated cities and tried to find Hamiltonian cycle of smallest length using simulated annealing. It was noticed that only for small annealing rates the convergence was fast.