Laboratory work 7
Robotics II: Control, Modeling and Learning with Laboratory

Ulugbek Alibekov
Robotics and Mechatronics
Nazarbayev University
ulugbek.alibekov@nu.edu.kz

***Abstract*—This laboratory report contains the results and findings of obtaining the inverse kinematics without the robot model. For that a deep learning toolbox of the Matlab was used. After several tries, the appropriate feed-forward neural network was implemented with 4 hidden layers. It was able to get mean error of 0,045. The actual and predicted points were plotted to illustrate the accuracy.**

I. Activation Function

First of all, it was necessary to understand which activation function suits better for this task. Out of available choices, the $tanh$ and $relu$ activation functions were considered. Then the model consisting of 1 hidden layer with 512 neurons was trained. The results are below.

Fig. 1. The RMSE and loss graphs for $tanh$ activation function

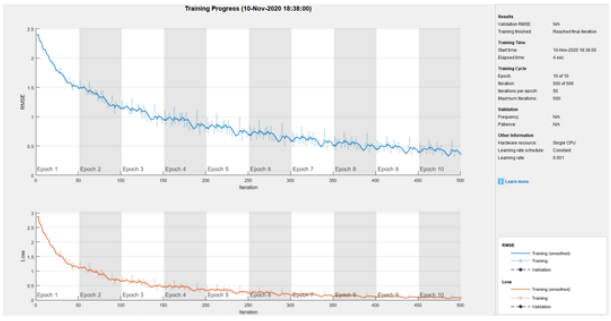

Fig. 2. The RMSE and loss graphs for $relu$ activation function

As it can be seen from the graphs, the $relu$ activation function is a better choice for this problem. To be more precise, when the model tried to predict the inverse kinematics points of the robot the mean error for the $tanh$ was 1.49, whereas for $relu$ this value was around 0.28. That would be the starting point.

TABLE I
Different Neural Network architectures

H-Layer 1	H-Layer 2	H-Layer 3	H-Layer 4	Mean Error
512	256	128	64	0.079
256	128	128	64	0.065
512	256	256	256	0.053
512	512	256	256	0.045

II. The structure of the model

It was decided to use 4 hidden layers, since the larger number resulted in overfitting while the model with lower number of layers could not generalize the problem correctly. Based on this, different number of neurons at each hidden layer was tested. The general rule is that the number of the neurons at the previous hidden layer should be larger than in the current one. Also, the number of neurons at each layer should usually be divisible by 8. As the result, several configurations were tested and the results are given in the table 1.

The last model showed the best performance with mean error of 0.045. The visualization of the predicted and actual points are given below.

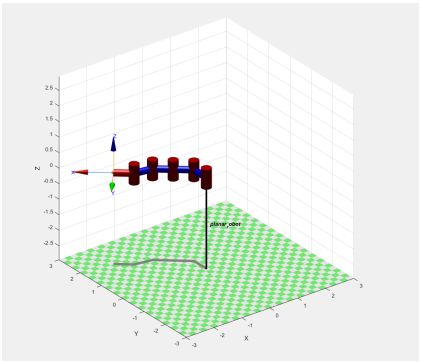

Fig. 3. Actual and Predicted points for the robot (They coincide)

Here

$$XYZ_{actual} = -1,93; 1.83; 0$$
$$XYZ_{predicted} = -1.94; 1.86; 0$$