

Homework 3: Multi-Layer Perceptron**Part 4.**

We ran the data sample through mpl using each of the number of nodes in the hidden layer, as listed in the table below, using the default momentum of $\alpha=0.9$. Based on the table below, we can see that by using ten hidden layers we received one of the smallest mean errors, smallest standard deviation, and an average gap between max error and min error. Additionally, I must note that when running the data through mlp several times, using ten hidden layers always remained the lowest, or second to lowest, in ambiguity and mean error.

# Hidden Layers	1	2	3	5	10	25	50
Mean Error	259.4	258.2	257.7	257.2	257.2	259.11	257.87
Standard Deviation	11.15	9.98	8.6	8.9	7.9	9.8	9.5
Max Error	284.0	284.0	284.0	284.0	284.0	284.0	284.0
Min Error	232.0	232.0	232.0	232.0	232.0	232.0	232.0

Part 5.

We ran the data sample through mpl using each of the momentums, as listed in the table below, using the number of hidden nodes determined in part 4, ten hidden layer nodes. Based on the table below, we can see that by using $\alpha=0.7$ we received one of the smallest mean errors,

smallest standard deviation, and an average gap between max error and min error. Although using $\alpha=1.0$ has a smaller mean error, the standard deviation and gap between max and min errors are greater. Therefore there is too much variation between the errors. Additionally, I must note that when running the data through mlp several times, using $\alpha=0.7$ always remained the lowest, or second to lowest, in ambiguity and mean error. Therefore, even if in this table $\alpha=1.0$ seems to make sense, overall $\alpha=0.7$ seems like a better candidate.

α	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Mean Error	258.7	258.4	258.8	259.1	259.6	258.6	258.0	259.5	256.0	257.95
Standard Deviation	8.9	7.7	7.5	8.4	8.5	8.5	8.3	7.3	8.9	7.8
Max Error	277.0	277.0	277.0	277.0	277.0	277.0	277.0	277.0	277.0	277.0
Min Error	206.1	238.9	239.0	238.9	239.0	239.0	239.0	239.0	196.0	239.0

Part 6.

We ran trained, early stopped and ran the matrix creation several times and the best we obtained was:

Confusion Matrix	Negative	Positive
Predicted Neg	265	251
Predicted Pos	0	0

Predicted Correct: 51.4%

From the table above we can determine that:

TP: 0

FP: 0

TN: 265

FN: 251

Sensitivity (recall): 0

Specificity: 1

Accuracy: 0.51

The parameters we used to train the data set were nine input nodes (number of features), ten hidden layer nodes, one output layer node, learning rate of $\eta=0.3$ (as instructed), and a momentum rate of $\alpha=0.7$. We used nine input nodes because each node represents a feature in the data sample describing the indel and its influences on the gene product. We used ten nodes in the hidden layer because as noted in part 4 above, when we ran mlp using ten hidden layer nodes and the default momentum rate of $\alpha=0.9$ we received the lowest validation error, statistically speaking. We used only a single output layer node because there are only two categories, disease indels and functional neutral indels (which do not cause diseases), which can be represented as positive (1) or negative (0). Lastly, we used a momentum rate of $\alpha=0.7$ because as noted in part 5 above, when we ran mlp using ten hidden layer nodes and momentum rate of $\alpha=0.7$ we received the lowest validation error, statistically speaking.