Programming Assignment # 3

Aswin Ramachandran
aswin.ramachandran@okstate.edu

TrainPA3:

Objective: To train the Player in playing Black Jack card game using neural network.

The training program was implemented using both the TLU threshold functions i.e. the Sigmoid function and the TanH function. The statistical information having mean, max, min and standard deviation of the player wins Vs the Generation are plotted. It took about 7500 seconds to complete the training for 100,000 generations.

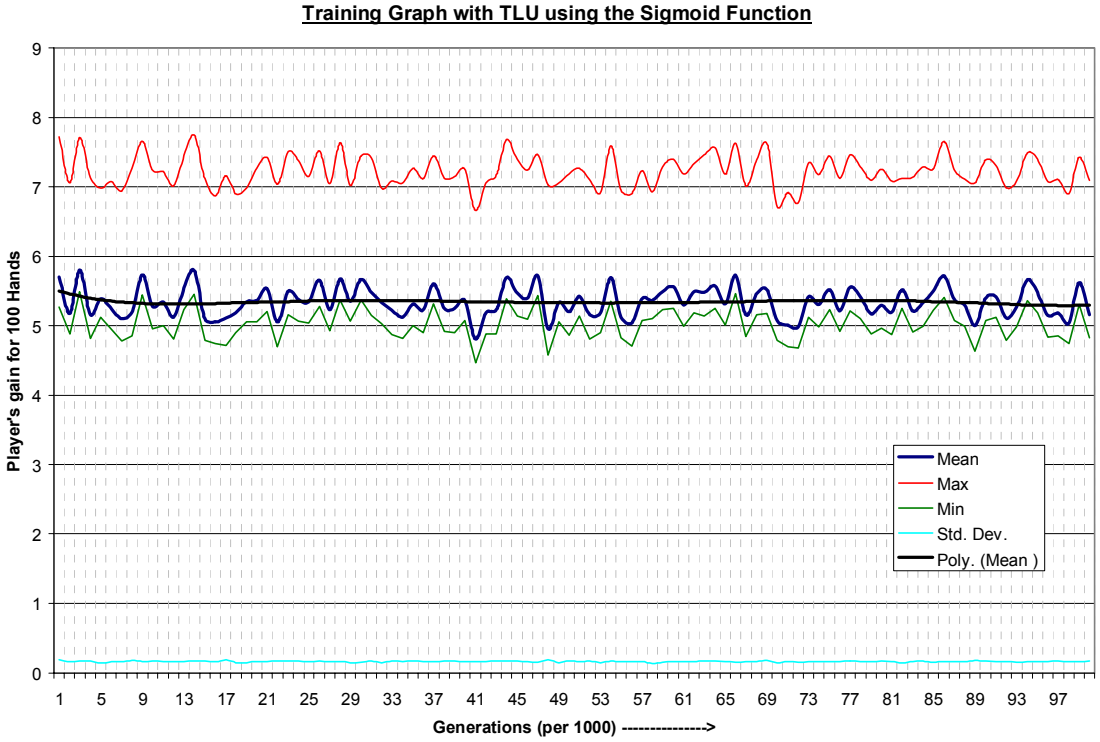
To improve the player's behavior and his winning rate**, *10 separate single input nodes*** both for the dealer and the player were assigned. The TLU implemented in the *Unary Node* representation uses a TanH function for threshold. The statistical results indicate a better trained player as the no. of wins is relatively higher than when trained using multiple input nodes.

Eliminating the Worst Generations:

Initial executions showed that the player deteriorates in learning partially due to poor results in certain generations that to contribute to the weights of the next generations. To avoid this, if the mean loss for the player is less than 5 (i.e. -5) for 100 hands then that particular generation is eliminated and a new set of 20 children are generated for the same generation (the generation count is not incremented if it's weights are discarded). This scheme is kind of applying inductive bias while training the player and did achieve a good improvement in the results.

Below are the Fitness Vs Generation plots. The plots are averaged for every 1000 generations over the total range of 100,000 generations.

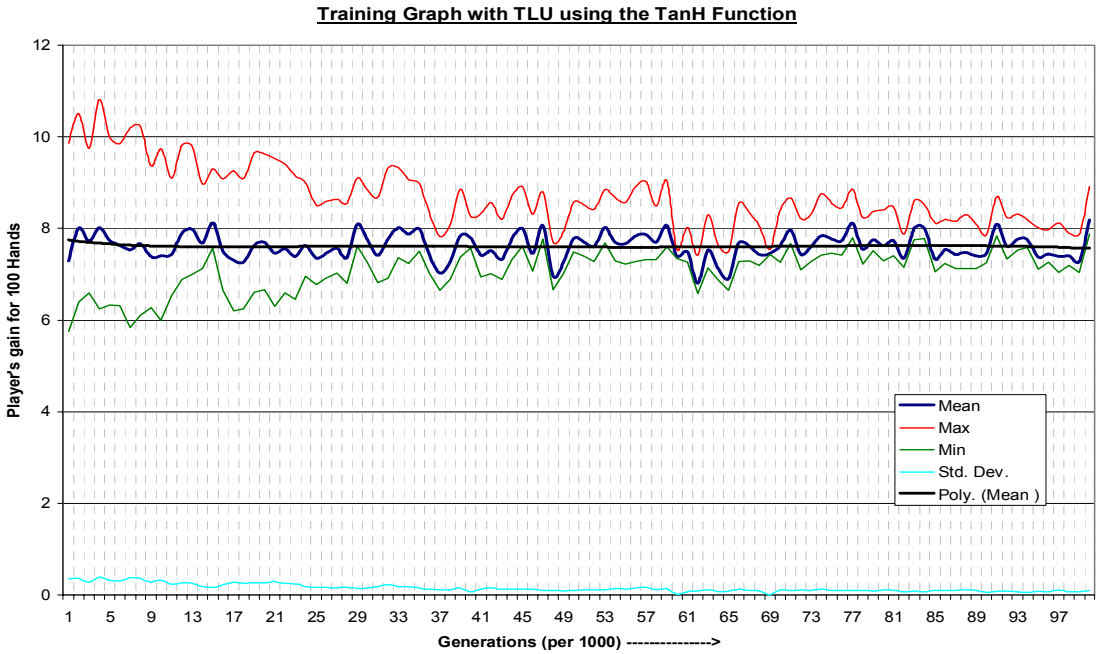
1. TLU using Sigmoid Function:



Comments:

The plot (1) shows the values of player's win (fitness) has he gets trained in each generation using the Sigmoid function. Though no significant training of player is noted from the plot, it does show some improvement for each generation.

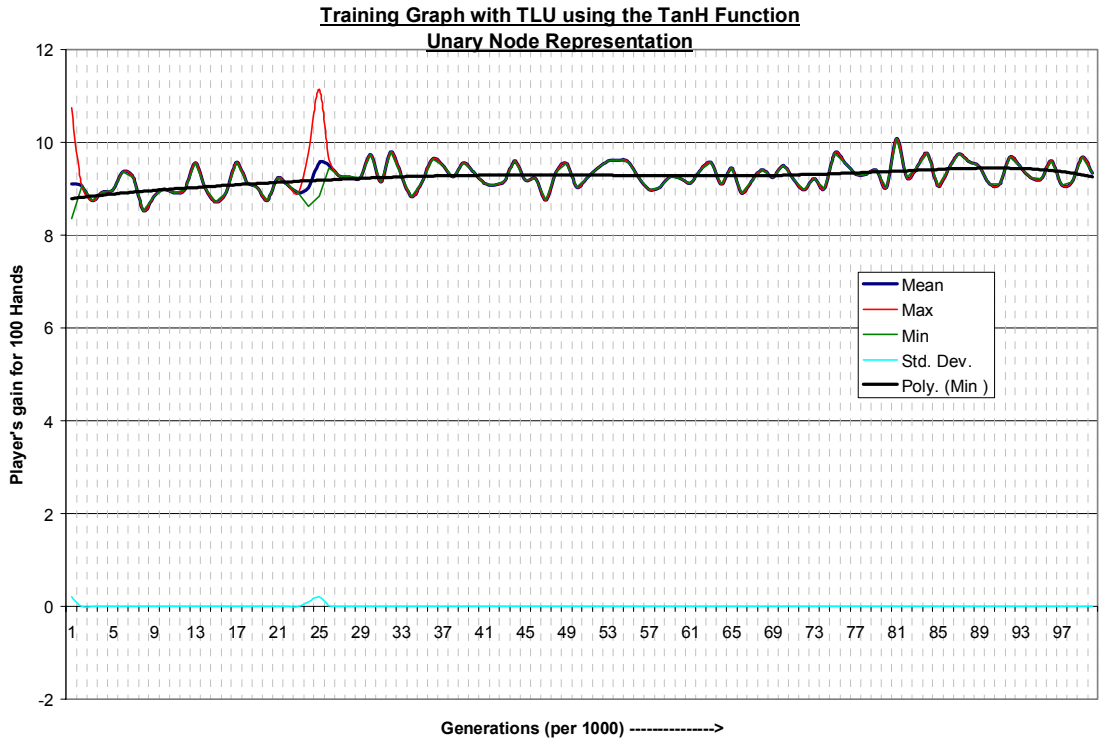
2. TLU using TanH function:



Comments:

The plot (2) shows the fitness curve of the player using the TanH function. The player seems to have a higher winning rate when trained using this threshold function.

3. Unary Node Representation:



Comment:

The fitness curve shown in plot (3) when the player is trained using Unary Node representation shows a significant increase in the winning rate compared to the previous two methods. But, after certain generations (30,000) the results show the max. and min. value have converged which may be a result of crowding.

Problems encountered:

The 'gamma' value used while generating the children was a cause of concern. It did cause some crowding effecting in the player results i.e. even after several generations the weights are so adjusted that the training data always falls into a particular classification. Hence, the 'gamma' value is made very small to avoid this effect.

TestPA3:

Command Line Input Arguments for TestPA3:

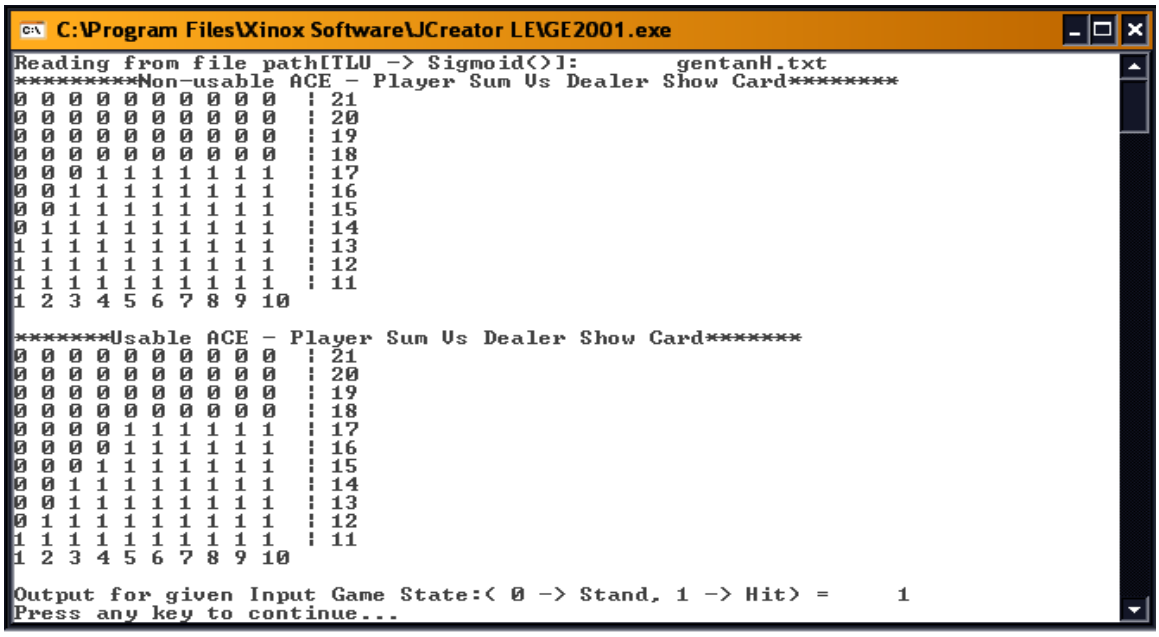
<inNN> <DealerValue> <PlayerValue> <AceValue>

For example,

C:\\genTanH.txt 12 2 0

The path *inNN* may not work if there is a space in the path line.

1. TLU using TanH function as Threshold.



2. TLU using the Sigmoid-function as Threshold.