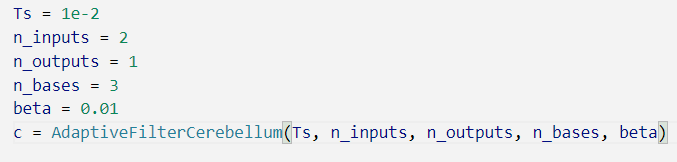
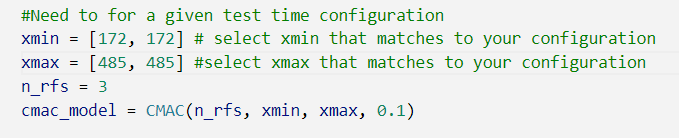
**Indications about the training / testing process by using the Adaptive Filter**

* Before comparing the accuracy of the combined version Adaptive Filter (AF) plus Neural network (NN) vs the standalone Neural Network, train the AF (do not train the NN anymore) with about 10 targets. This is because, initially, the prediction values of the AF are very small (for example, initial predictions are around 0.0000008, and after the training they are around 0.1, 0.05, 0.08, etc.).
* In the initial training of AF, choose the angular difference between target and robot initial position larger than 20 degrees. Under these conditions, future AF predictions can be improved.
* To check the performance of AF, use different location of targets and initial robot positions than the initial robot positions and target locations that will be used for testing. Compare the performance of the AF vs the NN.

The following parameters have been used for AF:

**Indications about the training / testing process by using the Adaptive Filter or CMAC**

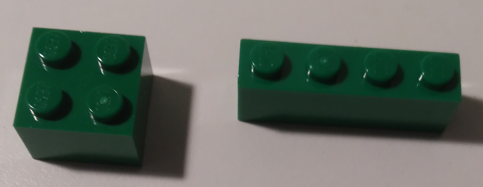
* Before comparing the accuracy of the combined version *CMAC plus NN* vs the standalone *NN*, train the CMAC (do not train the NN anymore) with about 10 random targets.
* In the initial training of CMAC, choose the angular difference between target and robot initial position larger than 20 degrees. Under these conditions, future CMAC predictions can be improved.
* To rigorously check the performance of CMAC, use different target location and initial robot positions for training and for testing. These values will be used for comparing the CMAC vs NN.

The following parameters have been used for CMAC:

* The maximum and minimum values for the coordinates required for the CMAC can be obtained by setting the Y angle of the robot to -90 and 90 when the X angle is placed as described in the previous document uploaded on Learn.

**We have observed that**:

* When the x coordinate difference between target and robot is less than or equal to five, it considers that the target is reached by the robot.
* The maximum number of iterations in any control loop is set to 20 assuming that convergence to target will be happened at or before 20 iterations.
* It is better to use thin green boxes as shown in the picture below for fast convergence and to increase the accuracy during the reaching task.



**Comparison Cerebellum (CMAC or AF) vs NN**

During the testing, the table in the next page has been created to compare the standalone NN version with the combined version NN + CMAC and with the combined version NN + AF. Here the initial position of the robot and the target position is given in angular positions of Y joint when the robot is placed in front of the webcam in the same way that we used for training.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| idx | Robot position | Target position | Number of iterations to converge | | | Error at convergence | | |
| NN only | NN + CMAC | NN + AF | NN only | NN + CMAC | NN + AF |
| 1 |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |  |  |
| 8 |  |  |  |  |  |  |  |  |
| 9 |  |  |  | | | | | |
| 10 |  |  |  |  |  |  |  |  |
| 11 |  |  |  |  |  |  |  |  |
| 12 |  |  |  |  |  |  |  |  |
| 13 |  |  |  |  |  |  |  |  |
| 14 |  |  |  |  |  |  |  |  |
| 15 |  |  |  |  |  |  |  |  |
| 16 |  |  |  |  |  |  |  |  |
| 17 |  |  |  |  |  |  |  |  |
| 18 |  |  |  |  |  |  |  |  |
| 19 |  |  |  |  |  |  |  |  |
| 20 |  |  |  |  |  |  |  |  |
| 21 |  |  |  |  |  |  |  |  |
| 22 |  |  |  |  |  |  |  |  |
| 23 |  |  |  |  |  |  |  |  |
| 24 |  |  |  |  |  |  |  |  |
| 25 |  |  |  |  |  |  |  |  |
| 26 |  |  |  |  |  |  |  |  |
| 27 |  |  |  |  |  |  |  |  |
| 28 |  |  |  |  |  |  |  |  |