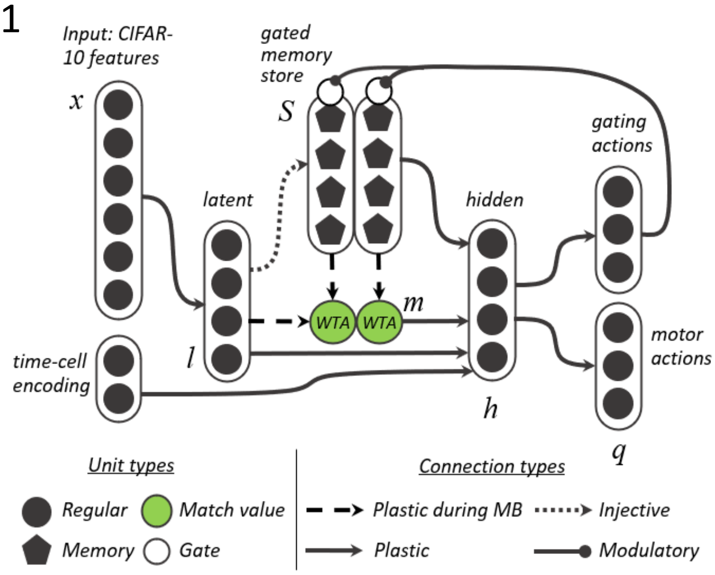
Learning to match Natural Scenes using Biologically Plausible Deep Reinforcement Learning

The ability to compare stimuli with those in memory is essential for intelligent behaviour. This ability can be studied, for example, with Delayed-Match-to-Sample tasks (DMS). Many contemporary models perform comparisons using a hard-wired mechanism. Here, we use recent work on self-organizing comparators, to demonstrate that a layer with anti-Hebbian plasticity can be embedded in a deep reinforcement learning (DRL) agent with flexible working memory to enable comparisons. We train the network with a biologically plausible learning reinforcement rule that relies on attentional feedback and global dopamine signalling to locally guide synaptic plasticity.

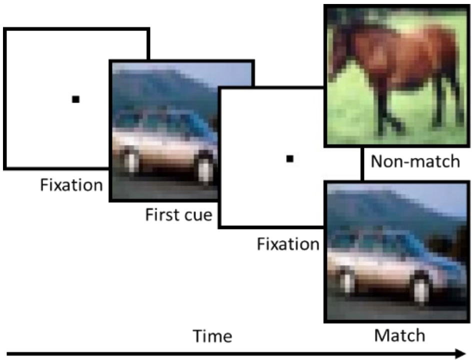
There are two phases in learning. In the first phase, we use an anti-Hebbian learning rule to train a layer that learns to match inputs to memories using motor babbling (MB). During MB, we sample the input space and learn a general matching function. In a second phase, we use the biologically plausible rule BrainProp to train the network to perform a Delayed Match-to-Sample task on CIFAR-10 pictures. Our results illustrate how plasticity during early developmental stages allows for the implementation of the fundamental capacity to match, whereas later phases can use a reinforcement learning scheme to learn advanced memory-guided behaviours that rely on these matching circuits.

1



Additional detail: In a DMS task, subjects are shown an image, hold it in memory during a delay and then match it to a second shown image, giving one response for a match and another one for mismatch. Evidence from monkey studies suggest that the prefrontal cortex plays a role in matching[1]. Current neural comparison models use hard-coded one-on-one connections[2], weights fixed to one[3], or shared weights[4] to compare an incoming stimulus to a stimulus in memory. Besides, the stimuli are represented by the scalar angle of motion direction[2] or short binary vectors[3]. Recently, a two-layer self-organizing comparator based on anti-Hebbian learning was shown to perform well in matching continuous-valued patterns[5]. Here we aimed to combine this matching method with a biologically plausible reinforcement learning scheme to enable a DRL agent to match high-dimensional features.

2



Methods: The network is a deep SARSA RL agent with an input layer x, two hidden layers (l, h) and an output layer q that gates the persistently active WM stores (S) (Fig. 1). The plastic connections are learned with a biologically plausible reinforcement learning rule[3]. Each cell in l is coupled to memory cells in S via a stable injective connection. The connections from l and S to the matching layers (m) are trained with an anti-Hebbian learning rule (AH). Lateral inhibition is modelled in m with a winner-takes-all mechanism, producing a scalar output activation at each timestep[5]. AH learning causes low values for correlated inputs and high values for uncorrelated inputs.

Fig. 1, Network architecture DLR agent. Multi-layer network with sigmoid activation complemented by gated memory and matching units. Units in the output layer select motor responses gate the memory content. Fig. 2, One trial of the four-step DMS task: Fixation, first sample, fixate again, during the presentation of a second sample press left for match or press right for mismatch.

Learning occurs during a first motor babbling phase and a second reward driven phase. During MB, the agent explores randomly[6]. Every *t* a random visual scene is presented. Motor and memory gating actions are selected at random, so that on average half of the memories are maintained and the matching process is learned. At the end of MB, AH switches off to stabilize matching. In the second phase, reinforcement learning is used to learn a DMS task with natural scenes (here we used 24 images) (Fig. 2). The input stimuli are activity vectors of the second fully connected layer in a CNN with 80% training accuracy (Caleb Woy, Kaggle) plus a time stamp inspired by time cell encoding[3]. If 85 out of the last 100 trials are correct, a new set of images is selected. After six switches, the stimulus set is changed to all stimuli combined and learning is completed.

Results: 25 agents were run. Within 3030 trials (s.d. 354), the connections of the matching layers self-organise such that a linear classifier correctly predicts 99% of matches/non-matches (Fig 3A). The DMS task is learned in 8458 trials (s.d. 5577, 68% of agents converge) and speeds up after the first level (Fig. 3B-C). A trial encompasses all four time-steps (Fig. 2), unless it is aborted earlier due to a wrong intermediate action. The newly extended deep architecture speeds up the learning of the DMS task by more than 50%[3] compared to a shallow agent with abstract inputs[3].

Discussion: We demonstrated that it is possible to train a matching scheme using unsupervised learning that can be used to compare sensory items to items in working memory. This can be combined with a DRL scheme that is biologically plausible. We propose a new role for the different phases of plasticity during development, where the role of early phases is to enable matching, such that it can support memory matching in later phases. Recent neurophysiological studies proposed dedicated neuronal mechanisms for matching[7] and it will be of interest to compare the properties of units in our matching network to the matching mechanisms implemented in the brain.

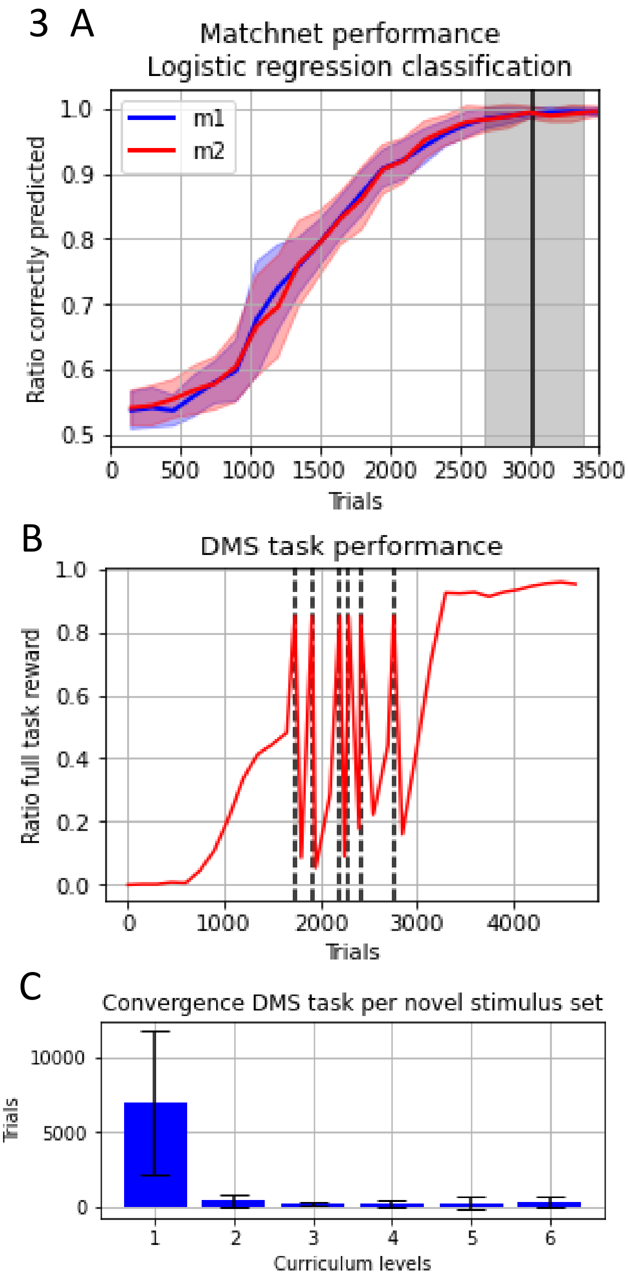


Fig. 3, Average performance of 25 agents. A, Running average of accuracy (250 trials) in the MB phase assessed with a linear classifier predicting a match/non-match. Training is complete when all matching modules (one for every memory store) perform above 99% (black line). Shaded areas indicate standard deviation. B, Example of DMS training: agent learns curriculum with different sets of stimuli (dashed lines indicate stimulus switch) within 3000 trials and sustains performance after. C, Performance DMS task: Average number of trials to achieve a level.

References: [1] Miller, E. K., Erickson, C. A., & Desimone, R. (1996). *Journal of Neuroscience*, *16*(16), 5154–5167. [2] Engel, T. A., & Wang, X. J. (2011). *Journal of Neuroscience*, 31(19), 6982–6996. [3] Kruijne, W., Bohte, S. M., Roelfsema, P. R., & Olivers, C. (2021). *Neural computation*, 33(1), 1–40. [4] Zagoruyko, S., & Komodakis, N. (2015). *IEEE Conference on Computer Vision and Pattern Recognition* (CVPR), 4353-4361. [5] Ludueña, G. A., & Gros, C. (2013). *Neural Computation*, 25(4), 1006–1028. [6] Piaget, J. (1952). The origins of intelligence in children (Second edi). International Universities Press. [7] Keller, G. B., & Mrsic-flogel, T. D. (2018). *Neuron*, 100(2), 424–435.