Homework #5: Synaptic Plasticity
(NEURL-GA 3042, Fall 2022)
Due date: Sunday November 6

Problem 1. Generate correlated N-dimensional input data with zero mean and some set covariance. Algorithmically, an arbitrary covariance matrix can be generated by sampling a $N \times N$ random matrix $\mathbf{A}$, where each entry is drawn independently from a gaussian distribution $\sim \mathcal{N}\left(0, \frac{\eta^2}{N}\right)$ with zero mean and variance η^2/N. Using this random matrix, one can construct a covariance matrix as $\Sigma = \mathbf{A}\mathbf{A}^\top$ (We do this because a covariance needs to be symmetric/ positive definite). To sample individual input datapoints from this distribution one samples N-dimensional vectors, $\boldsymbol{\epsilon}$, with entries independently drawn from $\epsilon_i \sim \mathcal{N}(0,1)$ for i=1:N; then multiplies the outcome by $\mathbf{A}$, $\mathbf{x} = \mathbf{A}\boldsymbol{\epsilon}$.

Note 1: This procedures follows the known property of gaussian variables that if $\boldsymbol{\epsilon}$ *is normally distributed with mean* μ *and covariance* K *then the random variable* $\mathbf{x} = \mathbf{A}\boldsymbol{\epsilon}$ *is also normal with mean* $\mathbf{A}\mu$ *and covariance* $\mathbf{A}K\mathbf{A}^\top$.

(a) In the first instance, assume the number of inputs is $N = 2$, $\eta = 1$. Feed a stream of independently sampled input datapoints (for at least 1000 steps, but may need longer for convergence) into a linear neuron and train the feedforward synapses by Oja's rule, with a small learning rate, 0.001. Initialize weights at random from zero mean unit variance gaussian. Check that over learning the weights end up aligning with the first principal component (i.e. the eigenvector with the largest eigenvalue of Σ): visualize the data as a scatter plot; plot the first PC of the data in red; plot the projection of the weight vector onto the first PC in blue. Generate such a plot at different stages of learning.

(b) Plot the norm of the weight vector as a function of time to see the effects of normalization.

(c) repeat (a-b) with a higher dimensional input space $N = 10$, using the first two input dimensions for the scatter plot.

Note 2: for debugging purposes it may be useful to initialize the weights to the answer and check that the weights remain correct after multiple iterations of applying *the learning rule (formally, we're checking that the PC is a fixed point for the learning dynamics).*

Problem 2. Implement nearest-neighbor STDP for a single synapse (modification only depends on pairs of a postsynaptic spike and a presynaptic spike with the shortest time difference). Keeping the presynaptic input to Poisson draws with a fixed firing rate 10Hz, independently vary the firing rate of the postsynaptic Poisson sequence in the range 0 to 25Hz. Measure the average change in weight as a function of postsynaptic firing rate for parameters $A_+ = 1.03$, $A_- = -0.51$, $\tau_+ = 14$ms, $\tau_- = 34$ms. For each estimate (i.e. postsynaptic firing rate) compute the average change over many realizations of pre- and post-synaptic draws, using 10 sec long trials for each realization.