Assignment 2: Analogue Hopfield network (v3)

Neural Computation 2015-2016. Mark van Rossum

26th November 2015

Practical info

Organize your answers according to the questions; don't merge them. Plots should include axis labels and units (either on the plot, or mentioned in the text), see my web page link. Some answers will require units as well.

You will find that some questions are quite open-ended. In order to receive full marks for those you will need to do more than running a simulation and making a plot. Instead, you should substantiate your explanations and claims, for instance by doing additional simulations or mathematical analysis. However, core-dumping (just writing down all you can think of) is discouraged, and incorrect claims can reduce marks. It should not be necessary to consult scientific literature, but if you do use additional literature, cite it. There will be a to-be-determined normalization factor between the number of points scored and the resulting percentage mark.

Copying results is absolutely not allowed and can lead to severe punishment. It's OK to ask for help from your friends. However, this help must not extend to copying code, results, or written text that your friend has written, or that you and your friend have written together. I assess you on the basis of what you are able to do by yourself. It's OK to help a friend. However, this help must not extend to providing your friend with code or written text. If you are found to have done so, a penalty will be assessed against you as well.

Deadline will be announced via email and the website. Hand in a paper copy to ITO (if you are out of town an email with a PDF to me is fine). Late policies are strict and are stated at `www.inf.ed.ac.uk/student-services/teaching-organisation/for-taught-students/coursework-and-projects`. In case of illness etc, contact your personal tutor (CC me if you want).

Model and setup

In this assignment we consider a rate based network with $N = 100$ units. The network is used as a model for a piece of cortex which can embed some memories. The units are described by their firing rates $r(t)$. We first assume fully linear units, so that

$$\tau \frac{dr_i(t)}{dt} = -r_i(t) + \sum_j w_{ij} r_j(t) + s_i$$

where s_i is the stimulus.

There are $M = 5$ different stimulus vectors, labelled with p. So $\mathbf{s}^p$ is stimulus p (bold-face denotes a N-dimensional vector). They are drawn from an exponential distribution, $P(s_i^p) = \exp(-s_i^p)$. In Octave/Matlab it is easiest to create a matrix with all stimuli at once, `smat=exprnd(1,N,M)`.

Questions

Question 1 (5 points) Suppose the weights of the recurrent connections are all the same and equal to w_0. What is the maximal value for w_0 in that case for the network to be stable. Confirm your result with a stimulated network.

From now on, suppose that the weights of the connections are given by $w_{ij} = c\frac{1}{M}\sum_{p=1}^{M}(s_i^p - \bar{s})(s_j^p - \bar{s})$, where $\bar{s}$ denotes the average s, but remove the diagonal terms, i.e. `w=w-diag(diag(w))` in Octave/Matlab. This way the stimuli are stored in the network.

Question 2 (5 points) Examine the distribution of synaptic weights, i.e. the distribution of the matrix elements w_{ij}.

Question 3 (10 points) What is the maximal value for c in this case for the network to be stable? Explore numerically how this depends on N. Compare with theory (which is perhaps only possible in the limit of large N...)

In case the network is stable, the steady state is given by $\mathbf{r} = W.\mathbf{r} + \mathbf{s}^p$, which for small c becomes $\mathbf{r} = \mathbf{s} + W.\mathbf{s}$. Assume this latter equation for Question 4&5.

Question 4 (5 points) In this approximation examine the statistics of $\mathbf{r}$, in terms of its mean and variance when a learned stimulus is presented. Also introduce a set of stimuli that have the same statistics as the learned stimuli, but are not stored in the the weight matrix (i.e. novel patterns). Examine the statistics of $\mathbf{r}$ in that case as well.

Question 5 (10 points) Calculate the expected variance of $\mathbf{r}$ theoretically for both learned and novel patterns.

Next, we study whether the network can sustain attractor memory states. Hereto simulate

$$\tau\frac{dr_i(t)}{dt} = -r_i(t) + g(\sum_j w_{ij}r_j(t) + s_i)$$

where $g(x) = \frac{2}{1+\exp(-x+2)}$. To allow for attractor states determine the maximal c for which the linear network is stable (as above) and then multiply each element of W by a factor 5, i.e. multiply c by 5.

Take a simulation time-step of about $\tau/10$. Simulate the evolution of the network for times $t = 0..10\tau$, while a stimulus is present ($\mathbf{s} = \mathbf{s}^p$) , then continue simulating a bit longer, e.g. $t = 10\tau \ldots 20\tau$, while the stimulus is removed ($\mathbf{s} = \mathbf{0}$).

Question 6 (10 points) Simulate the network and calculate the (average) Pearson correlation between the stimulus and the final network state. Note you will need to run the simulation a couple of times, as the results can differ from run to run. How can the performance be improved?