The Hopfield-ish energy



Note in Scellier’s implementation, he treats i and j as though they’re layer numbers (because that implementation assuming layers only talk to other layers), but for a more general implementation, Wij should be a full weight matrix for every node to every other node

Flow of ideas:

Start with a neuron’s behavior, for instance a leaky-integrator:

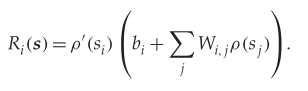


(or the continuous version)



* (ignore eta above, it’s noise which we can dismiss for now)
* Here, what’s said is that the change in state for neuron *h* is equal to some external-driving-force R(s) input to that neuron which depends on all the other neurons (*s)*

Now, according to this local update rule, the neuron states *s* will evolve into some kind of configuration where they become stable. But what is that configuration? Well, if we assume, say, that the driving force for each neuron R\_i(s) is just a bias term plus the weighted sum of outputs (***rho(s\_j)***) from its neighbors ***j***:



* Ignore rho’(s) above, it’s a not necessary for this discussion
* R(s) really is just the input to a given neuron
* rho(s) is really just the output from all the neuron’s connected neighbors

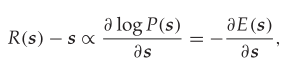
We will find that the configurations that *s* settles into correspond to the lowest-energy points of some as-yet-undetermined energy function *which is entirely determined by the neuron states* ***s*** *and how the neurons talk to each other, i.e.* ***R(s)***. In particular, if we can write that energy function s.t. the probability of those states are



then by definition



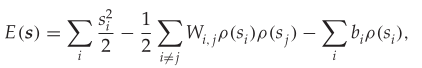
In particular, if we can write that energy function s.t.



Then the stationary states (= the final configuration = the fixed points = when R(s) == s) of ***s*** will correspond to local minima in the energy function and be stable there. We’re going to go a step farther here, and assume that finding these low-energy states is actually useful (without explanation!)

The energy function

If we just whip up the following energy function out of thin air, we see that it meets the criteria above:



Namely:

