A Prototype Spiking Neural Network

Nathan Wycoff

September 23, 2018

1 Existing Models

1.1 The Leaky Integrate and Fire Neuron

The leaky linear integrate and fire neuron (LIF) is a first order system of ODE's:

$$V_i' = -aV_i + I_i(t); i \in \{1, \dots, h\}$$

There is also a "reset condition": if $V_i(\hat{t}) = v_t$, with v_t a known threshold value, then $\lim_{\epsilon \to 0^+} V(\hat{t} + \epsilon) = v_r$, with v_r a reset value, generally just 0.

Let the set F_i represent $\{t : V_i(t) = v_t\}$.

Then we can write the input current as such:

$$I_i(t) = \sum_{j=1}^{h} w_{i,j} \sum_{\hat{t} \in F_j} \alpha(t - \hat{t})$$

with α a kernel function which decays rather quickly. For our purposes, let's start with $\alpha(\Delta t) = a(\exp \frac{-\Delta t}{b} - \exp \frac{-\Delta t}{c})\Theta(\Delta t)$ with Θ representing the heaviside function, constants a, b, c known.

See "./python/lif_neuron.py" for an implementation using forward Euler.

Initial values are usually chosen arbitrarily. For our purposes, we will set $V_0 = v_r$, and further $v_r = 0$.

1.2 The Inverse Problem

Spiking neuron models have many uses. Our focues will be on using them to perform statistical computation to solve learning problems: classification, regression, planning, etc. This involves presenting the network with an input, and demanding that it produce as close as possible to a prespecified spike train.

That is to say, some subset of the network will be called "output nerons". For each of these output neurons i, there will be a list of desired spike times $t_{i,f}^d$. the inverse problem is to discover weights $w_{i,j}$ as described above, throughout the *entire network* (even though we are only monitoring output neurons), such that the spike times of output neurons is close to the desired spike times in some sense. For now, let's use sum of squared differences, so the objective is $\sum_{i \in \mathcal{O}} \sum_{f \in \mathcal{D}_i} (t_{i,f}^a - t_{i,f}^d)^2$, where $\mathcal{O}$ is the index set of output neurons, and $\mathcal{D}_i$ is the index set of firing times for neuron i.

2 Model Description

Each neuron's potential is described as a second order ODE:

$$\begin{aligned} V''(t) = &-aV(t)+ \\ &b_1 \mathbb{1}_{V(t)>\alpha_1} \mathbb{1}_{V(t)<\alpha_2} \mathbb{1}_{V'(t)>0}+ \\ &- b_2 \mathbb{1}_{V(t)>\alpha_2}+ \\ &- b_3 \mathbb{1}_{V(t)>\alpha_3} \mathbb{1}_{V'(t)<0} \mathbb{1}_{V(t)<\alpha_2}+ \\ &- b_4 \mathbb{1}_{V(t)>\alpha_1} \mathbb{1}_{V'(t)<0} \mathbb{1}_{V(t)<\alpha_3}+ \\ &- b_5 \mathbb{1}_{V(t)<\alpha_4} \end{aligned} \tag{1}$$

Each line describes a separate component of the model.

The first line tells us that the potential decays if there is no input from presynaptic neurons.

The second line causes acceleration if the potential gets too close to zero. The idea is that we don't want to potential to every cross below 0.

The third line causes acceleration when the potential crosses the threshold (α_1).

The fourth line decelerates when the potential reaches the peak (α_2).

The fifth line continues the deceleration throughout the spiking region (between α_1 and α_2).

3 Choosing Model Parameters

There are many parameters associated with this model that need to be decided upon.

Let's begin with the b_i's. It is first desireable that the velocity of the potential is approximately the same after the spike as it was before. This may be achieved by setting $b_1\Delta_1 + b_3\Delta_3 - b_4\Delta_4 = 0$ where Δ_i is the corresponding distance that the force acts upon.

4 Model Dynamics

4.1 Constant Stimulation

For a single neuron with constant input current, the subthreshold potential takes this form:

$$v(t) = I_0(1 - e^{-t/\tau})$$