



Neural Networks

CS 479



Jeff Orchard

Preface

Disclaimer Much of the information on this set of notes is transcribed directly/indirectly from the lectures of CS 479 during Winter 2021 as well as other related resources. I do not make any warranties about the completeness, reliability and accuracy of this set of notes. Use at your own risk.

Note that the course currently is CS 489, but in winter 2022, this course will become **CS 479**.

For any questions, send me an email via <https://notes.sibeliusp.com/contact/>.

You can find my notes for other courses on <https://notes.sibeliusp.com/>.

Sibelius Peng

Contents

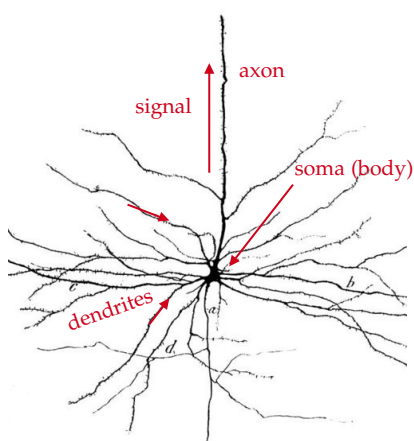
Preface	1
1 Neuron Models	3
1.1 Neurons	3
1.2 Neuron Membrane Potential	3
1.3 Hodgkin-Huxley Model	4
1.4 Leaky Integrate-and-Fire Model	7
1.5 Activation functions	9
1.6 Synapses	10
1.7 Connection Weight	11

Neuron Models

1.1 Neurons

A neuron is a special cell that can send and receive signals from other neurons.

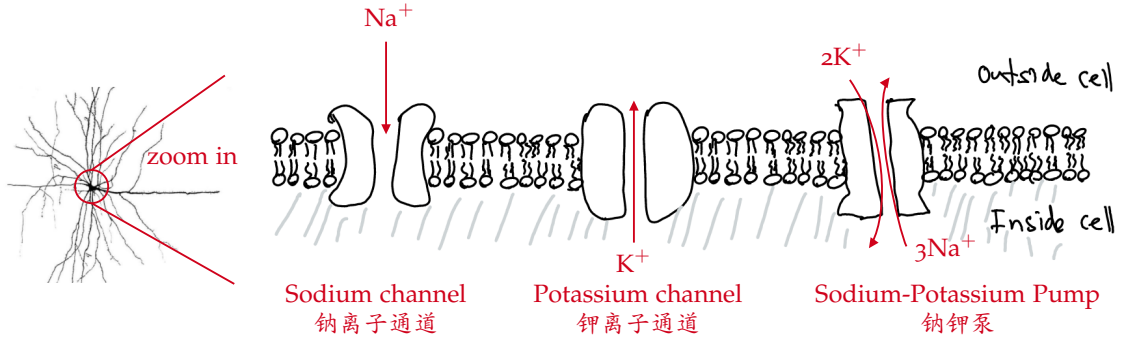
A neuron can be quite long, sending its signal over a long distance; up to **50m long!** But most are much shorter.



- Soma: 体细胞.
- Axon: 轴突. The electrical signal generated by soma travels along the axon.
- Dendrites: 树突. The electrical excitation is collected in the dendrites.
- Synapse: 突触. Structure that permits a neuron (or nerve cell) to pass an electrical or chemical signal to another neuron or to the target effector cell (wiki).

1.2 Neuron Membrane Potential

Ions(离子) are molecules or atoms in which the number of electrons (-) does not match the number of protons (+), resulting in a net charge. Many ions float around in your cells. The cell's membrane, a lipid bi-layer, stops most ions from crossing. However, ion channels embedded in the cell membrane can allow ions to pass.



Sodium-Potassium Pump exchanges 3 Na^+ ions inside the cell for 2 K^+ ions outside the cell.

- Causes a higher concentration of Na^+ outside the cell, and higher concentration of K^+ inside the cell.
- It also creates a net positive charge outside, and thus a net negative charge inside the cell.

This difference in charge across the membrane induces a voltage difference, and is called the **membrane potential**.

Neurons have a peculiar behaviour: they can produce a spike of electrical activity called an **action potential**(动作电位). This electrical burst travels along the neuron's *axon* to its *synapses*, where it passes signals to other neurons.

1.3 Hodgkin-Huxley Model

Alan Lloyd Hodgkin and Andrew Fielding Huxley received the Nobel Prize in Physiology or Medicine in 1963 for their model of an action potential (spike). Their model is based on the nonlinear interaction between membrane potential (voltage) and the opening and closing of Na^+ and K^+ ion channels.

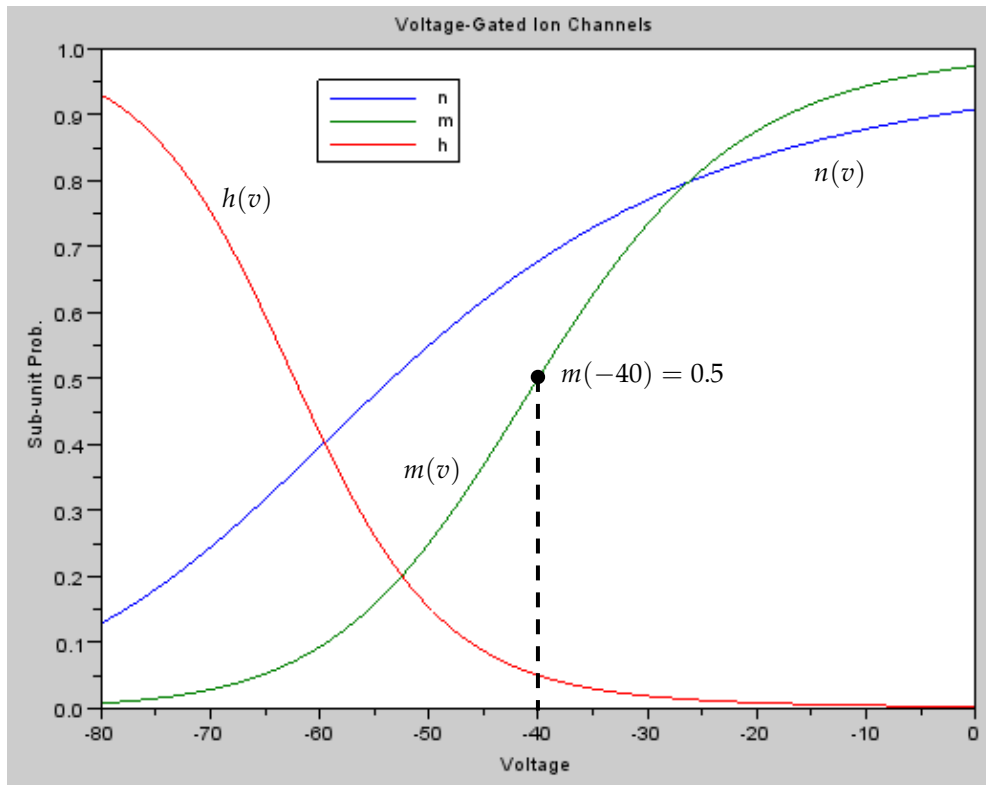
Both Na^+ and K^+ ion channels are voltage-dependent, so their opening and closing changes with the membrane potential.

Let V be the membrane potential. A neuron usually keeps a membrane potential of around -70mV .

The fraction of K^+ channels that are open is $n(t)^4$, where $\frac{dn}{dt} = \frac{1}{\tau_n(V)}(n_\infty(V) - n)$. Here n is a dynamic variable, and $n_\infty(V)$ is the equilibrium solution constant.

The fraction of Na^+ ion channels is $(m(t))^3h(t)$, where m and h are each themselves dynamic variables that also depend on the voltage.

$$\begin{aligned}\frac{dm}{dt} &= \frac{1}{\tau_m(V)}(m_\infty(V) - m) \\ \frac{dh}{dt} &= \frac{1}{\tau_h(V)}(h_\infty(V) - h)\end{aligned}$$



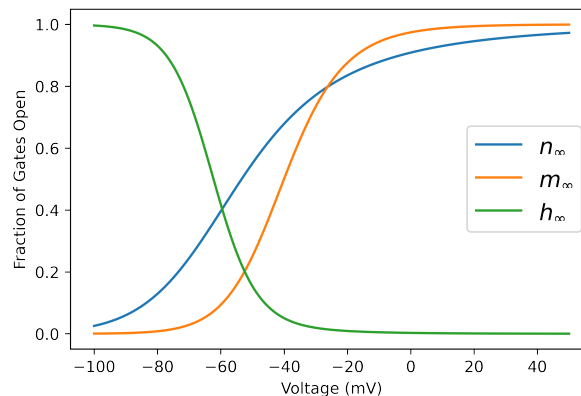
These two channels allow ions to flow into/out of the cell, inducing a current... which affects the membrane potential, V . Here is a differential equation which governs the membrane potential.

$$C \frac{dV}{dt} = J_{in} - \underbrace{g_L(V - V_L)}_{\text{leak current}} - \underbrace{g_{Na}m^3h(V - V_{Na})}_{\text{sodium current}} - \underbrace{g_Kn^4(V - V_K)}_{\text{potassium current}}$$

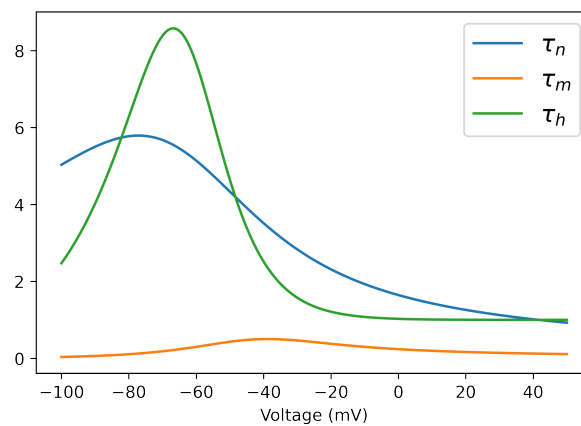
- C : capacitance.
- $\frac{dV}{dt}$: rate of change in voltage, or current.
- J_{in} : input current, usually from other neurons.
- V_L, V_{Na}, V_K : zero-current potentials.
- g_L, g_{Na}, g_K : max conductance.

This system of four differential equations (DEs) governs the dynamics of the membrane potential. Notice what happens when the input current is: negative, zero, slightly positive, very positive.

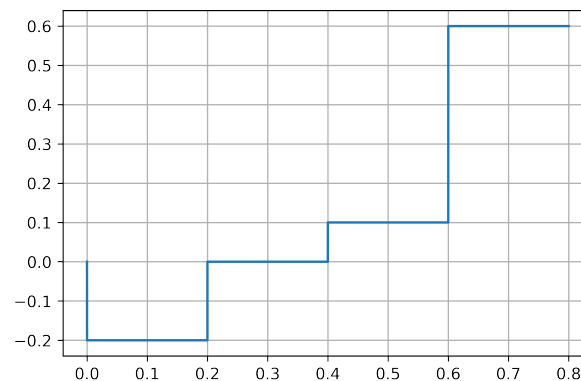
Here we can model this model in python. We have already seen these as functions of voltage.



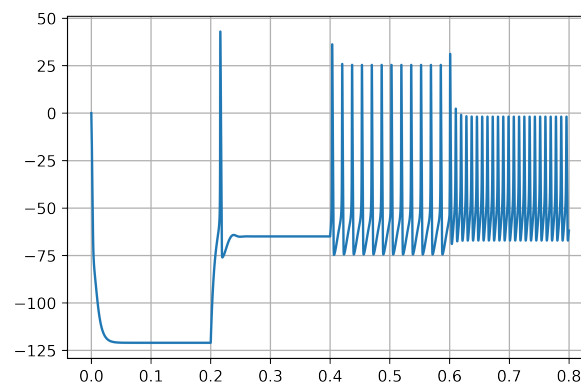
These are the τ 's in case you are interested.



Here is the input current. At the beginning, we have negative current, then way to 0.6, which is fed in to the model.



Then this is how neuron behaves.



At the beginning, membrane potential goes to around -120 . As we increase the input current, the membrane potential kinda goes higher. At 0.1, it's high enough that causes regular action potentials. As we increase input current even more, the action potentials continue to occur even faster. The firing rate of neurons goes up, the number of spikes per second goes up.

The HH model is already greatly simplified:

- a neuron is treated as a point in space
- conductances are approximated with formulas
- only considers K^+ , Na^+ and generic leak currents

- etc.

But to model a single action potential (spike) takes many time steps of this 4-D system. However, spikes are fairly generic, and it is thought that the *presence* of a spike is more important than its specific shape. So instead of modelling spikes themselves, we are going to offload that to some generic spike phenomenon and look at the sub-threshold membrane potential model that.

1.4 Leaky Integrate-and-Fire Model

The leaky integrate-and-fire (LIF) model only considers the sub-threshold membrane potential (voltage), but does NOT model the spike itself. Instead, it simply records when a spike occurs (i.e., when the voltage reached the threshold). So here is the model.

$$C \frac{dV}{dt} = J_{in} - g_L(V - V_L)$$

- C : capacitance.
- g_L : conductance and $g_L = \frac{1}{R}$ where R is resistance.
- J_{in} : input current.

If we multiply both sides by R , we get

$$\underbrace{RC}_{\tau_m} \frac{dV}{dt} = RJ_{in} - (V - V_L).$$

- τ_m : time constant which dictates how quick things happen.
- RJ_{in} : by Ohm's Law, let $V_{in} = RJ_{in}$.

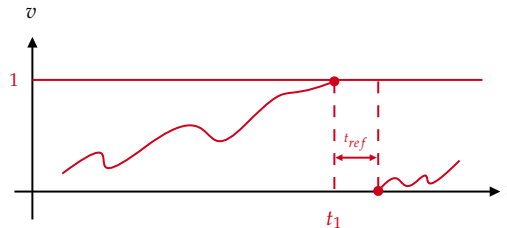
Thus, the voltage can be modelled as

$$\tau_m \frac{dV}{dt} = V_{in} - (V - V_L) \quad \text{for } V < V_{th}.$$

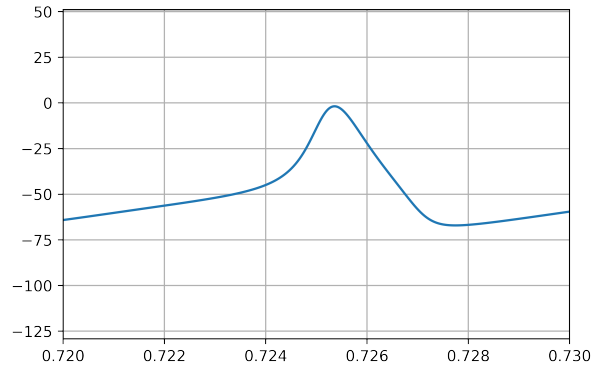
So this is the dynamics of the sub-threshold membrane potential. Change of variables: $v = \frac{V - V_L}{V_{th} - V_L}$, then $v \rightarrow 0$ if $v_{in} = 0$ and $v = 1$ is the threshold. Then we end up with a DE:

$$\tau_m \frac{dv}{dt} = v_{in} - v.$$

We integrate the DE for a given input current (or voltage) until v reaches the threshold value of 1. Then we record a spike at time t_1 . After it spikes, we wait a little bit, τ_{ref} , refractory time. It remains dormant during its refractory period, τ_{ref} (often just a few milliseconds). After that time, we integrate again from zero.



Let's put this in the context of the Hodgkin-Huxley model. If we zoom in on some little spikes here (between 0.72, 0.73). We can see as follows:



So the Hodgkin-Huxley model does model the spike itself.

LIF Firing Rate

Suppose we hold the input, v_{in} , constant. We can solve the DE analytically between spikes.

Claim

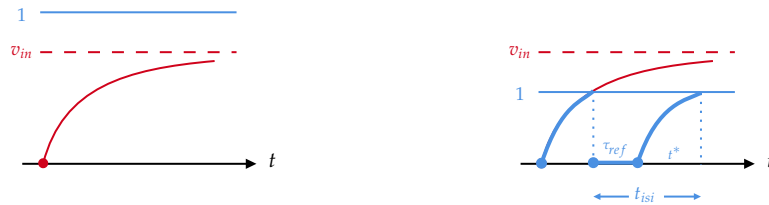
$v(t) = v_{in}(1 - e^{-\frac{t}{\tau}})$ is a solution of the IVP: $\tau \frac{dv}{dt} = v_{in} - v$, $v(0) = 0$.

Proof:

Plug in the solution to the DE and show LHS = RHS. □

What does the solution look like? It will approach v_{in} asymptotically.

Importantly, for the neuron to fire an potential, v_{in} has to be bigger than 1.



where t_{isi} stands for interspike interval.

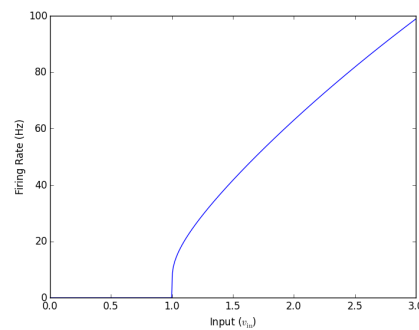
It can be shown that the steady-state firing rate for a constant input v_{in} is

$$G(v_{in}) = \begin{cases} \frac{1}{\tau_{ref} - \tau_m \ln(1 - \frac{1}{v_{in}})} & \text{for } v_{in} > 1 \\ 0 & \text{for } v_{in} \leq 1 \end{cases}$$

The graph plots the function above. It is called Tuning curve, because it tells us about how the neuron reacts to different input currents. In fact, eventually it would go asymptotic at a certain value.

Typical values for cortical neurons(神经元):

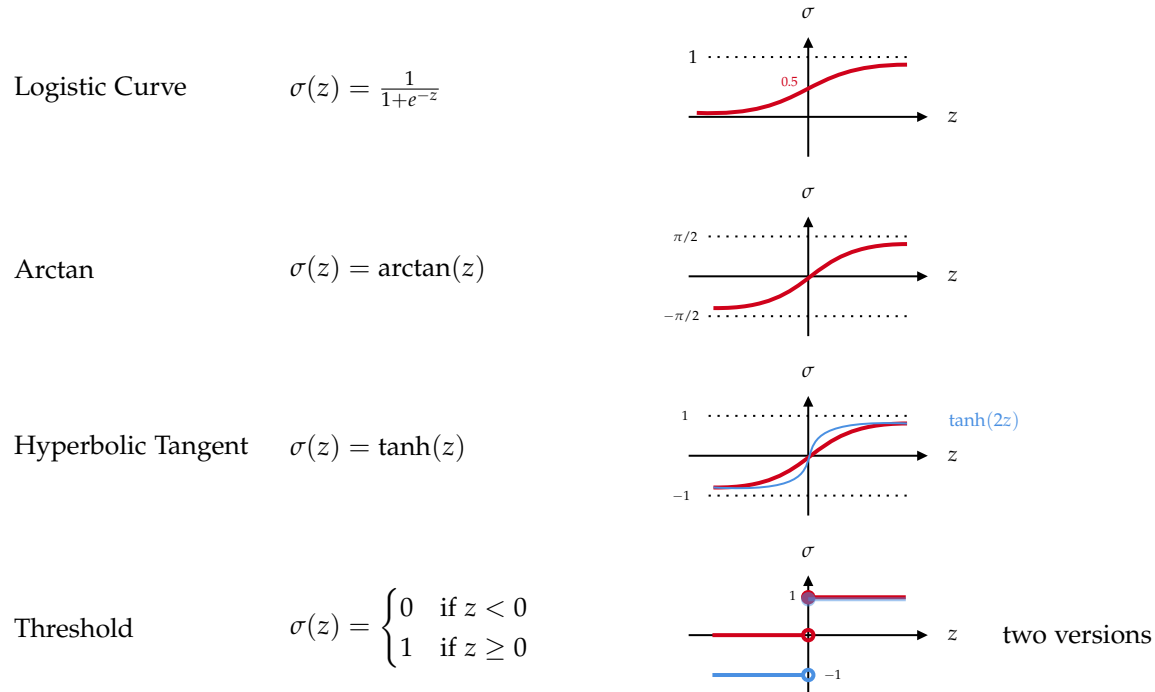
- $\tau_{ref} = 0.002s$
- $\tau_m = 0.02s$



Let's take a look at even simpler neurons.

1.5 Activation functions

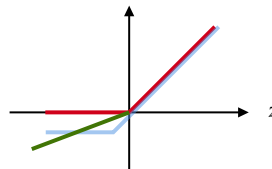
As we've seen, the activity of a neuron is very low, or zero, when the input is low, and the activity goes up and approaches some maximum as the input increases. This general behaviour can be represented by a number of different activation functions. In general, we call these sigmoidal shape.



Rectified Linear Unit (ReLU): This is just a line that gets clipped below at zero. **Leaky ReLU (LeReLU).** Another version in green, which changes the slope when negative/at the origin.

$$\text{ReLU}(z) = \max(0, z)$$

LeReLU

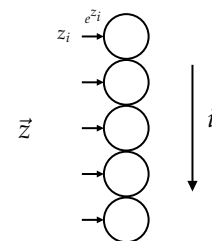


Multi-Neuron Activation Functions: Some activation functions depend on multiple neurons. Here are two examples.

SoftMax

SoftMax is like a probability distribution (or probability vector), so its elements add to 1. If \vec{z} is the drive (input) to a set of neurons, then

$$\text{SoftMax}(\vec{z})_i = \frac{e^{z_i}}{\sum_j e^{z_j}}$$



Then by definition, $\sum_i \text{SoftMax}(\vec{z})_i = 1$.

For example, $\vec{z} = [0.6, 3.4, -1.2, 0.05] \xrightarrow{\text{softmax}} \vec{y} = [0.06, 0.9, 0.009, 0.031]$

One-Hot

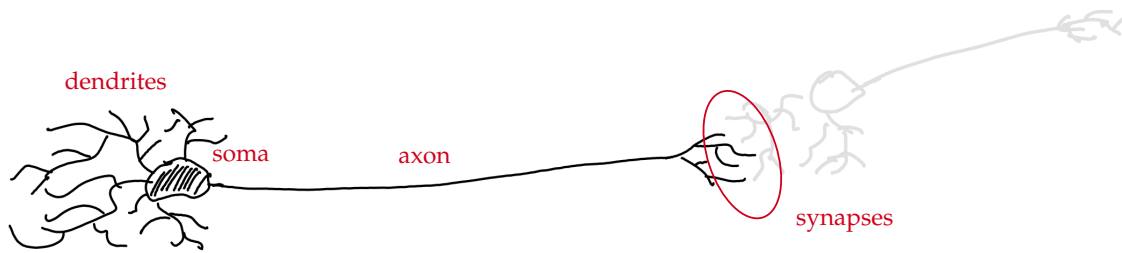
One-Hot is the extreme of the softmax, where only the largest element remains nonzero, while the others are set to zero.

For example, $\vec{z} = [0.6, 3.4, -1.2, 0.05] \xrightarrow{\text{one-hot}} \vec{y} = [0, 1, 0, 0]$

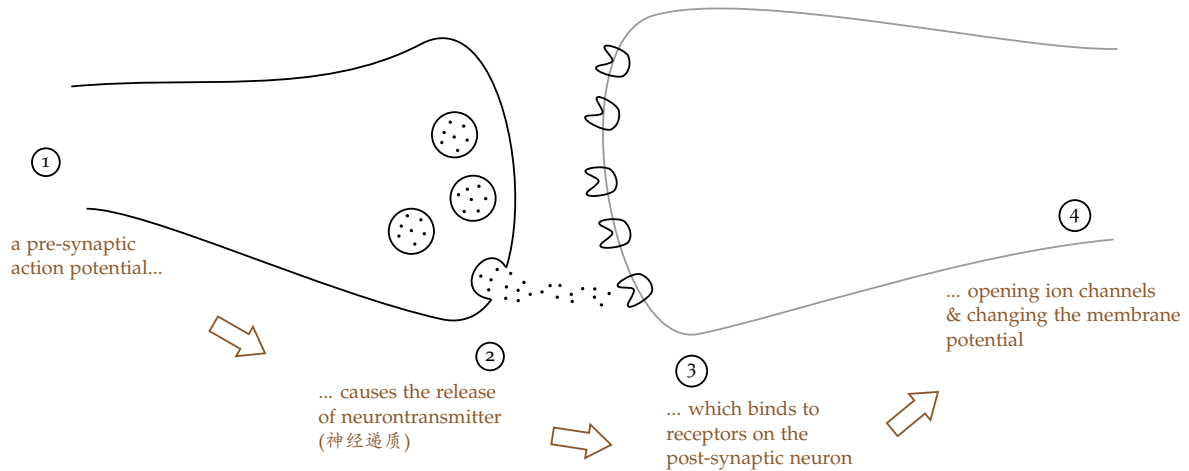
1.6 Synapses

To get an overview of how neurons pass information between them, and how we can model those communication channels.

So far, we've just looked at individual neurons, and how they react to their input. But that input usually comes from other neurons. When a neuron fires an action potential (the wave of electrical activity) travels along its axon.



The junction where one neuron communicates with the next neuron is called a synapse.

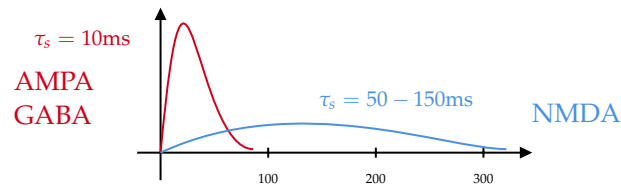


Even though an action potential is very fast, the synaptic processes by which it affects the next neuron takes time. Some synapses are fast (taking just about 10 ms), and some are quite slow (taking over 300 ms). If we represent that time constant using τ_s , then the current entering the post-synaptic neuron can be written

$$h(t) = \begin{cases} kt^n e^{-\frac{t}{\tau_s}} & \text{if } t \geq 0 \text{ for some } n \in \mathbb{Z}_{\geq 0} \\ 0 & \text{if } t < 0 \end{cases}$$

where k is chosen so that $\int_0^\infty h(t)dt = 1 \implies k = \frac{1}{n!\tau_s^{n+1}}$.

The reason we have a split at zero is because the spike arrives at the synapse at time $t = 0$, and then we are looking what's happening after that.



Some neurotransmitters are fast, like AMPA. Some are slow, like NMDA. The area under these curves are 1.

The function $h(t)$ is called a Post-Synaptic Current (PSC) filter, or (in keeping with the ambiguity between current and voltage) Post-Synaptic Potential (PSP) filter.

Multiple spikes form what we call a “spike train”, and can be modelled as a sum of Dirac delta functions,

$$a(t) = \sum_{p=1}^3 \delta(t - t_p)$$

if we have three spikes at t_1, t_2, t_3 .

Dirac Delta Function

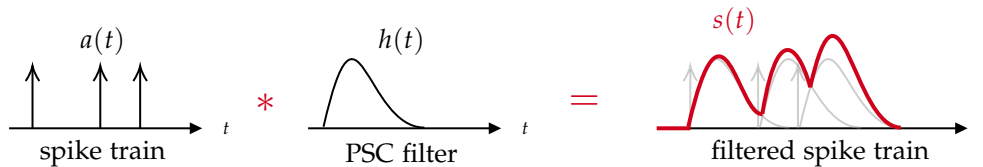
Dirac Delta Function is defined as

$$\delta(t) = \begin{cases} \infty & \text{if } t = 0 \\ 0 & \text{otherwise} \end{cases}$$

$$\text{and } \int_{-\infty}^{\infty} \delta(t) dt = 1 \text{ and } \int_{-\infty}^{\infty} f(t) \delta(T - t) dt = f(T).$$

How does a spike train influence the post-synaptic neuron?

Answer: You simply add together all the PSC filters, one for each spike. This is actually convolving the spike train with the PSC filter.



That is,

$$s(t) = (a * h)(t) = \sum_p h(t - t_p) = \text{sum of PSC filters, one for each spike}$$

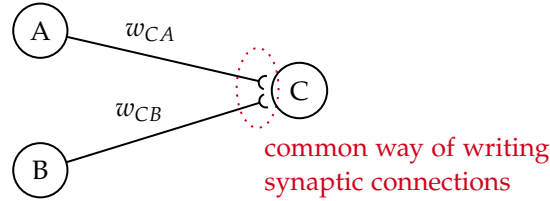
1.7 Connection Weight

The total current induced by an action potential onto a particular post-synaptic neuron can vary widely, depending on:

- the number and sizes of the synapses,

- the amount and type of neurotransmitter,
- the number and type of receptors,
- etc.

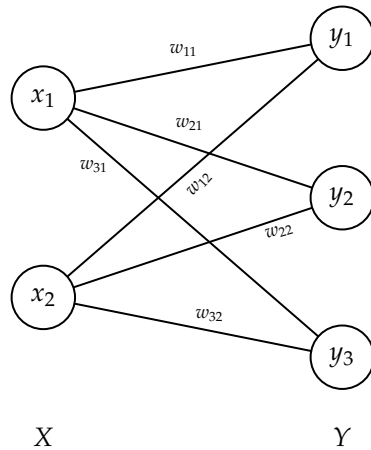
We can combine all those factors into a single number, the **connection weight**. Thus, the total input to a neuron is a weighted sum of filtered spike-trains.



Weight Matrices

When we have many pre-synaptic neurons, it is more convenient to use matrix-vector notation to represent the weights and activities.

Suppose we have 2 populations, X and Y , X has N nodes, Y has M nodes (neurons). If every node in X sends its output to every node in Y , then we will have a total of $N \times M$ connections, each with its own weight.



$$W = \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \end{bmatrix} \in \mathbb{R}^{M \times N}$$

Storing the neuron activities in vectors,

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \quad \vec{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}.$$

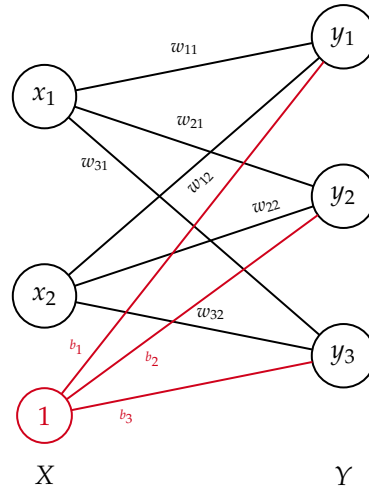
We can compute the input to the nodes in Y using

$$\vec{z} = W\vec{x} + \vec{b},$$

where \vec{b} holds the biases for the nodes (neurons) in Y . Bias is sort of a catch-all for influences on the neuron that are not accounted for the connections that we are modelling.

Thus $\vec{y} = \sigma(\vec{z}) = \sigma(W\vec{x} + \vec{b})$.

Another way to represent the biases, \vec{b} ,



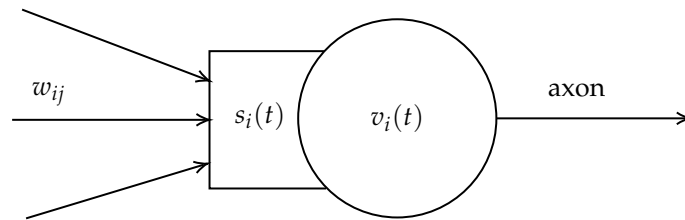
So $W\vec{x} + \vec{b} = [W|\vec{b}] \begin{bmatrix} \vec{x} \\ 1 \end{bmatrix} = \hat{W} \begin{bmatrix} \vec{x} \\ 1 \end{bmatrix}$.

Implementing Connections between Spiking Neurons

For simplicity, let $n = 0$: $h(t) = \frac{1}{\tau_s} e^{-\frac{t}{\tau_s}}$, which happens to be the solution of the IVP:

$$\tau_s \frac{ds}{dt} = -s, \quad s(0) = \frac{1}{\tau_s}.$$

Full LIF Neuron Model



Differential equations:

$$\begin{cases} \tau_m \frac{dv_i}{dt} = s_i - v_i & \text{if not refracting} \\ \tau_s \frac{ds_i}{dt} = -s_i \end{cases}$$

If v_i reaches 1 (threshold)...

1. start refractory period,
2. send spike along axon,
3. reset membrane potential v to 0.

If a spike arrives from neuron j , increase s_i : $s_i \leftarrow s_i + \frac{w_{ij}}{\tau_s}$. The amount of current that it injects into the post-synaptic neuron is proportional to the weight, and we divide it by τ_s , which is the normalizing factor so that the total amount of current that eventually gets injected is the weight.