

richardwu.ca

# CS 489/689 COURSE NOTES

ADVANCED TOPICS IN CS (NEURAL NETWORKS)

JEFF ORCHARD • WINTER 2019 • UNIVERSITY OF WATERLOO

---

Last Revision: January 17, 2019

## Table of Contents

<b>1</b>	<b>January 7, 2019</b>	<b>1</b>
1.1	Simulating neurons and the Hodgkin-Huxley model . . . . .	1
<b>2</b>	<b>January 9, 2019</b>	<b>3</b>
2.1	Leaky Integrate-and-Fire (LIF) model . . . . .	3
<b>3</b>	<b>January 11, 2019</b>	<b>4</b>
3.1	Sigmoid neurons . . . . .	4
3.2	Rectified Linear Unit (ReLU) . . . . .	6
<b>4</b>	<b>January 14, 2019</b>	<b>6</b>
4.1	Synapses . . . . .	6
<b>5</b>	<b>January 16, 2019</b>	<b>8</b>
5.1	Connection weights . . . . .	8
5.2	Euler's method . . . . .	9
5.3	Supervised Learning . . . . .	9

---

### Abstract

These notes are intended as a resource for myself; past, present, or future students of this course, and anyone interested in the material. The goal is to provide an end-to-end resource that covers all material discussed in the course displayed in an organized manner. These notes are my interpretation and transcription of the content covered in lectures. The instructor has not verified or confirmed the accuracy of these notes, and any discrepancies, misunderstandings, typos, etc. as these notes relate to course's content is not the responsibility of the instructor. If you spot any errors or would like to contribute, please contact me directly.

## 1 January 7, 2019

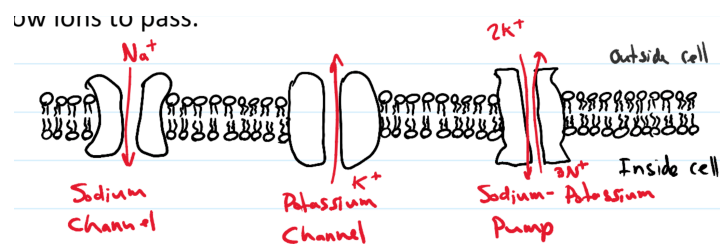
### 1.1 Simulating neurons and the Hodgkin-Huxley model

To construct neural networks we must first simulate how a biological neuron works.

Ions (positively and negatively charged molecules with excess protons and electrons, respectively) exist outside and inside of a cell and may be moved across the cell membrane.

There exist sodium and potassium **channels** which permit  $\text{Na}^+$  and  $\text{K}^+$  ions to move across the cell membrane, respectively.  $\text{K}^+$  channels move  $\text{K}^+$  ions out of the cell whereas  $\text{Na}^+$  channels move  $\text{Na}^+$  ions into the cell.

Sodium-potassium **pumps** exchange 3  $\text{Na}^+$  inside the cell for 2  $\text{K}^+$  ions outside the cell. This in effect creates a negative charge inside the cell.



**Figure 1.1:** Cell membrane with  $\text{Na}^+/\text{K}^+$  channels and a sodium-potassium pump. Ions move across the membrane via the channels and pump.

The difference in charge across the membrane induces a voltage difference called the **membrane potential**.

The **action potential** is a spike of electrical activity in neurons. This electrical burst travels along the neuron's **axon** to its **synapse** where it passes signals to other neurons.

The **Hodgkin-Huxley** model describes how the action potential is effected. Note that both  $\text{Na}^+$  and  $\text{K}^+$  ion channels are voltage-dependent:  $\text{Na}^+$  and  $\text{K}^+$  move according to the membrane potential as the channels open and close with the membrane potential.

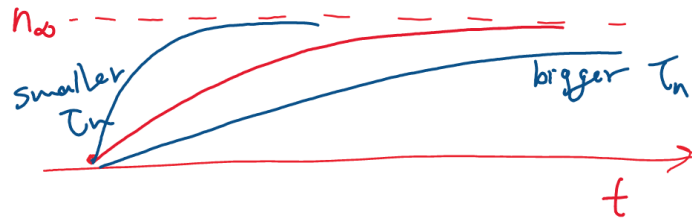
Let  $V$  be the membrane potential. A neuron usually keeps a membrane potential of around  $-70\text{mV}$ .

The fraction of  $\text{K}^+$  channels that are open is  $n^4$ , where

$$\frac{dn}{dt} = \frac{1}{t_n(V)} (n_\infty(V) - n)$$

where  $t_n(V)$  is the time constant and  $n_\infty(V)$  is the equilibrium solution constant, which are empirically calculated (they're both functions of  $V$  however).

Note that each  $\text{K}^+$  channel is controlled by four gates wherein the probability of one gate being open is  $n$ , hence the probability of all gates being open is  $n^4$ .



**Figure 1.2:**  $n$  in fraction of K<sup>+</sup> channels open over time (for a fixed  $V$ ). The blue graphs correspond to larger and smaller time constants  $\tau_n$ .

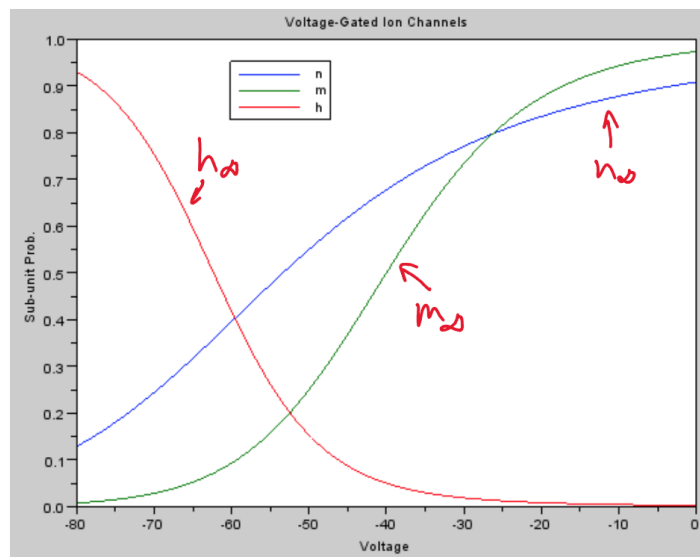
The fraction of Na<sup>+</sup> channels that are open is  $m^3h$ , where (similar to above)

$$\frac{dm}{dt} = \frac{1}{t_m(V)}(m_\infty(V) - m)$$

$$\frac{dh}{dt} = \frac{1}{t_h(V)}(h_\infty(V) - h)$$

similar to above, we can interpret this as the Na<sup>+</sup> channel is controlled by three gates with probability  $m$  being open and one gate with probability  $h$  being open.

If we measure empirically the equilibrium solutions for each of  $n, m, h$  over various voltage, we get logistic-like curves



We can thus express the membrane potential as a differential equation in terms of the fraction of K<sup>+</sup> and Na<sup>+</sup> channels open:

$$C \frac{dV}{dt} = J_{in} - g_L(V - V_L) - g_{Na}m^3h(V - V_{Na}) - g_Kn^4(V - V_K)$$

where each term corresponds to:

$J_{in}$  input current (from other neurons)

$g_L(V - V_L)$  current from “leakiness”

$g_{Na}m^3h(V - V_{Na})$  current from Na<sup>+</sup> channels

$g_K n^4 (V - V_K)$  current from K<sup>+</sup> channels

each  $g_X$  term corresponds to the max conductance for each of the sources, and each  $V_X$  term corresponds to the zero-current potential for each source.  $C$  corresponds to the capacitance of the neuron.

If we solve the above DE for  $V$  with various input potential  $J_{in}$  over time, we can see that increasing the input potential will cause the voltage to spike rapidly and successively which is the **action potential**.

## 2 January 9, 2019

### 2.1 Leaky Integrate-and-Fire (LIF) model

While the HH model already simplifies a neuron to a 4-D nonlinear system, we can further simplify it. We note that the presence of the spike is the most important takeaway and the shape (due to the K<sup>+</sup> and Na<sup>+</sup> channels) are less important.

The **leaky integrate-and-fire (LIF) model** models only the sub-threshold membrane potential but not the spike itself. We express it as

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

Note that  $g_L = \frac{1}{R}$  where  $R$  is the resistance, thus we have

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

$$\tau_m \frac{dV}{dt} = V_{in} - (V - V_L)$$

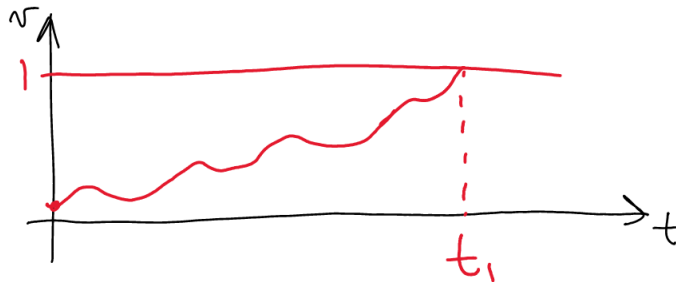
$$\tau_m = RC \quad RJ_{in} = V \text{ (Ohm's law)}$$

if  $V < V_{th}$  (threshold potential). If we let  $v = \frac{V - V_L}{V_{th} - V_L}$  for  $v < 1$ , then we have

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

(note that unlike HH, our simplified time constant  $\tau_m$  is not a function of  $v$ ).

If we integrate the DE for a given input voltage until  $v$  reaches 1 i.e. the threshold voltage of the cell is reached at time  $t_1$ , we see the membrane potential climbs in an irregular pattern until time  $t_1$



after which a spike is recorded and we reset the voltage to 0 again (after which we solve the DE for the next spike). There is a refractory period before it can spike again.

What is the firing rate if we held  $v_{in}$  constant? We need to solve for the DE analytically

**Claim.** We claim  $v(t) = v_{in}(1 - e^{-\frac{t}{\tau_m}})$  is a solution to  $\tau_m \frac{dv}{dt} = v_{in} - v$  where  $v(0) = 0$ .

*Proof.* Substitute and show that LHS = RHS. □

If  $v_{in} > 1$  (our threshold for firing), then our LIF neuron will spike. To solve for the firing rate, we need to solve for the time the spike occurs (as a function of  $v_{in}$ ).

The firing time  $t_{isi}$  is

$$t_{isi} = \tau_{ref} + t^*$$

where  $\tau_{ref}$  is the refractory time constant and  $t^*$  is the time for  $v$  to reach 1.

We need to find  $t^*$  where  $v(t^*) = 1$ . From our above solution

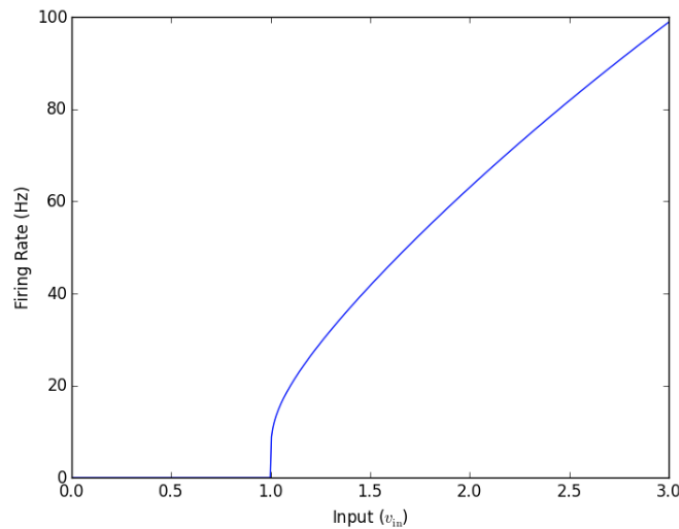
$$\begin{aligned} v(t^*) = 1 &= v_{in}(1 - e^{-\frac{t^*}{\tau_m}}) \\ \Rightarrow t^* &= -\tau_m \ln(1 - \frac{1}{v_{in}}) \quad v_{in} > 1 \end{aligned}$$

So  $t_{isi} = \tau_{ref} - \tau_m \ln(1 - \frac{1}{v_{in}})$  for  $v_{in} > 1$ .

Thus the steady-state firing rate for constant  $v_{in}$  is  $\frac{1}{t_{isi}}$  or

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

Typical values for *cortical neurons* are  $\tau_{ref} = 0.002s$  (2ms) and  $\tau_m = 0.02s$  (20ms) which has the following firing rates as a function of  $v_{in}$



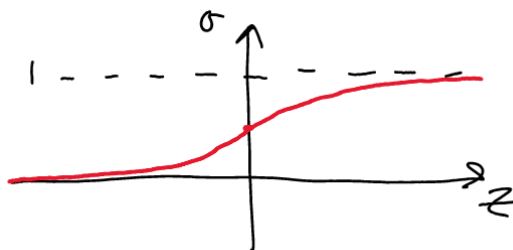
### 3 January 11, 2019

#### 3.1 Sigmoid neurons

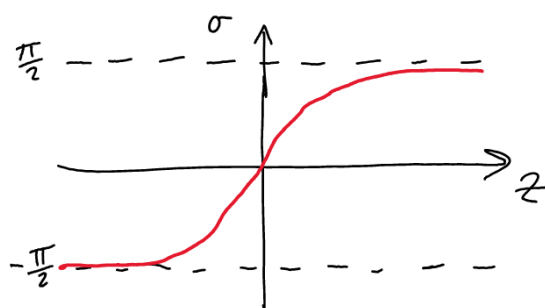
As we've seen before the activity of a neuron is low/zero when the input is low, and the activity goes up and approaches some maximum as the input increases. This behaviour can be represented by **activation functions**:

**Logistic curve**

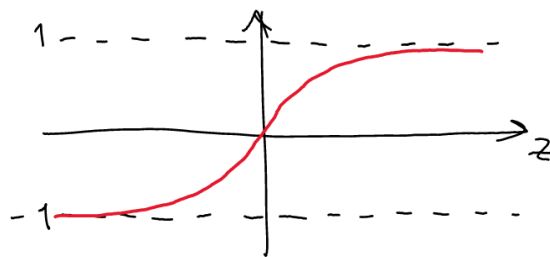
$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

**Arctan**

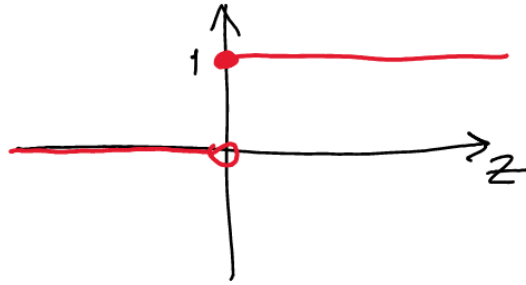
$$\sigma(z) = \arctan(z)$$

**Hyperbolic tangent**

$$\sigma(z) = \tanh(z)$$

**Threshold**

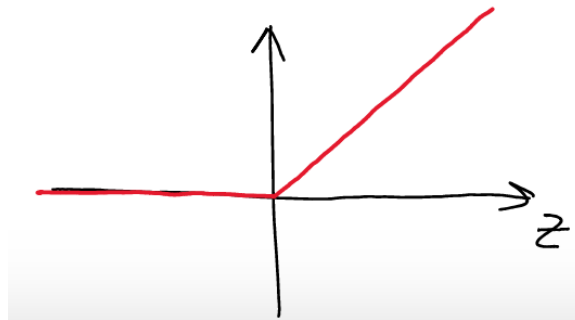
$$\sigma(z) = \begin{cases} 0 & \text{if } z < 0 \\ 1 & \text{if } z \geq 0 \end{cases}$$



### 3.2 Rectified Linear Unit (ReLU)

The ReLU function is simply a line that gets capped at zero below zero.

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

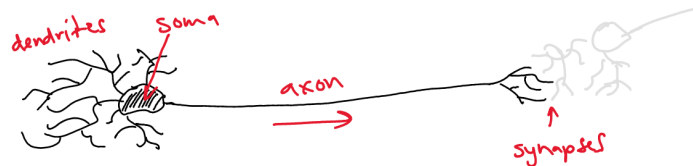


## 4 January 14, 2019

### 4.1 Synapses

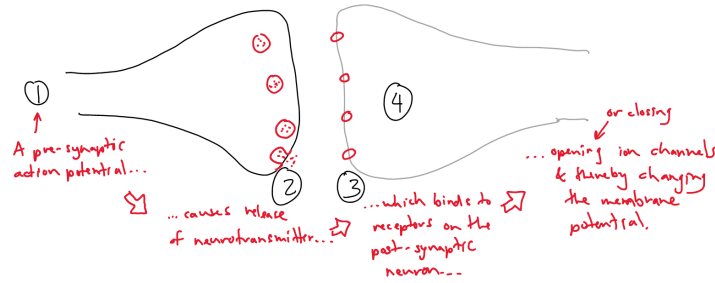
We want to understand how neurons pass on information and how to model the communication channels.

The input of a neuron comes from multiple other neurons. When a neuron fires an action potential, the wave of electrical activity travels along its axon.



The junction between the axon and dendrites of two communicating neuron is called a **synapse**.

A **pre-synaptic** action potential causes the release of **neurotransmitters** into adjacent synapses which bind to receptors on the **post-synaptic** neuron. This in turn opens or closes ion channels in the post-synaptic neuron thereby changing membrane potential and causing the action potential to propagate.



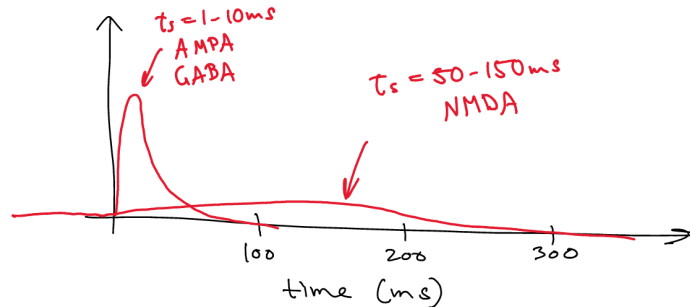
While the action potential is very fast, the synapse process can take from 10ms to over 300ms.

If we represent the time with constant  $\tau_s$  then the **post-synaptic current (PSC)** (or post-synaptic potential (PSP)) entering the post-synaptic neuron is

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

where  $k$  is a normalization constant such that  $\int_0^\infty h(t) dt = 1$  i.e.  $k = \frac{1}{n! \tau_s^{n+1}}$  (we will later scale this to the appropriate current levels).

Note that when  $n = 0$  we have exponential decay from time  $t = 0$ . When  $n = 1$  (which is more realistic) we have a gamma-like distribution with  $\alpha > 1$ . The  $\tau_s$  constant also influences the shape: as  $\tau_s$  increases, the more “drawn out” the post-synaptic current



Multiple spikes (from multiple action potentials) form a **spike train** which is modelled as a sum of Dirac delta functions  $a(t) = \sum_p \delta(t - t_p)$  where the Dirac delta function is defined as

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

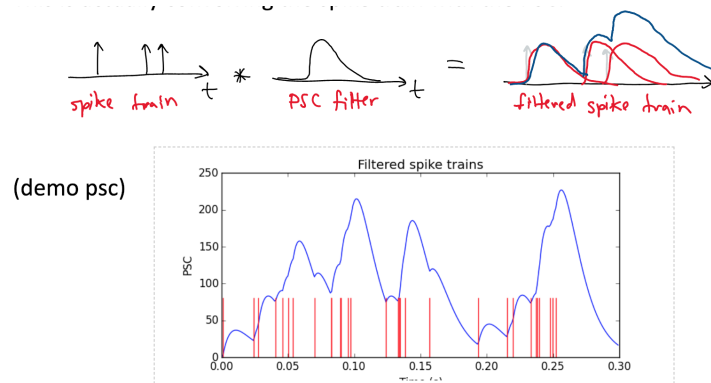
and

$$\int_{-\infty}^{\infty} \delta(t) dt = 1$$

$$\int_{-\infty}^{\infty} f(t) \delta(s - t) dt = f(s)$$

To combine our PSC filter/function with a spike train, we can simply take the convolution of the spike train (sum of Dirac deltas) and the PSC to form a **filtered spike train**





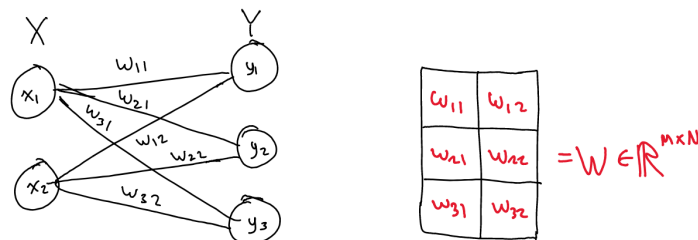
## 5 January 16, 2019

### 5.1 Connection weights

The total current induced on a *particular* post-synaptic neuron varies widely depending on:

- number and sizes of synapses (there may be multiple synapses between with multiple post-synaptic neurons)
- amount and type of neurotransmitter
- number and type of receptors
- etc.

We combine all these factors into a single number: the **connection weight** (which could be negative or inhibitory rather than excitatory). The total input is thus a *weighted sum* of filtered spike trains from pre-synaptic neurons. The weight from neuron  $A$  to  $C$  is denoted as  $w_{CA}$ . In general, for  $N$  pre-synaptic neurons ( $X$ ) and  $M$  post-synaptic neurons ( $Y$ ) we can represent the weights as an  $M \times N$  **weight matrix**



If we represent the neuron activities in neurons  $X$  and neurons  $Y$  as vectors

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

then we can compute the linear input  $\vec{z}$  to the nodes in  $Y$

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

where  $\vec{b}$  are the biases for the nodes in  $Y$ .

**Aside.** Biases can represent approximately constant noise from other neurons that is not a function of the upstream activities. It could also represent a baseline where some types of neurons have a propensity of firing even with little activity.

Finally after activation (spike) we have

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

Another way to introduce the bias is using  $\hat{W}$  where

$$\hat{W} = \begin{bmatrix} W & \vec{b} \end{bmatrix}$$

which we can re-write

$$W\vec{x} + \vec{b} \rightarrow \hat{W} \begin{bmatrix} \vec{x} \\ 1 \end{bmatrix}$$

## 5.2 Euler's method

Recall that  $h(t) = kt^n e^{-\frac{t}{\tau_s}}$ . For  $n = 0$  we have  $h(t) = \frac{1}{\tau_s} e^{-\frac{t}{\tau_s}}$  which is simply exponential decay with rate  $\tau_s$ .

**Claim.** This also happens to be the solution to the DE

$$\tau_s \frac{dh}{dt} = -h$$

*Proof.* Substitute and show  $LHS = RHS$ . □

We essentially have an initial value problem (IVP) where  $\frac{ds}{dt} = \frac{-s}{\tau_s}$  and  $s(0) = \frac{1}{\tau_s}$ . We can solve this DE or any first order DE numerically with **Euler's method**:

---

### Algorithm 1 Euler's method for $n = 0$ input current

---

```

1:  $s_0 \leftarrow s(0)$ 
2:  $\Delta t$  is time step size
3:  $t \leftarrow 0$ 
4: for  $i = 1, 2, \dots$  do
5:    $\frac{ds}{dt} \leftarrow \frac{-s_{i-1}}{\tau_s}$  ▷ (slope)
6:    $s_i \leftarrow s_{i-1} + \Delta t \frac{ds}{dt} = s_{i-1} (1 - \frac{\Delta t}{\tau_s})$  ▷ (step)
7:   for each pre-synaptic neuron  $n$  do
8:     if a spike arrived from neuron  $n$  at current time  $t$  then
9:        $s_i \leftarrow s_i + \frac{1}{\tau_s} w_n$ 
10:   $t \leftarrow t + \Delta t$ 

```

---

where after  $m = \frac{T}{\Delta t}$  steps (where  $T$  is our desired timepoint)  $s_m$  represents the total input current (from spike train) from all pre-synaptic neurons at time  $T$ .

## 5.3 Supervised Learning

If we have a network with connection weights, how do we *adjust* the network to output what we want?