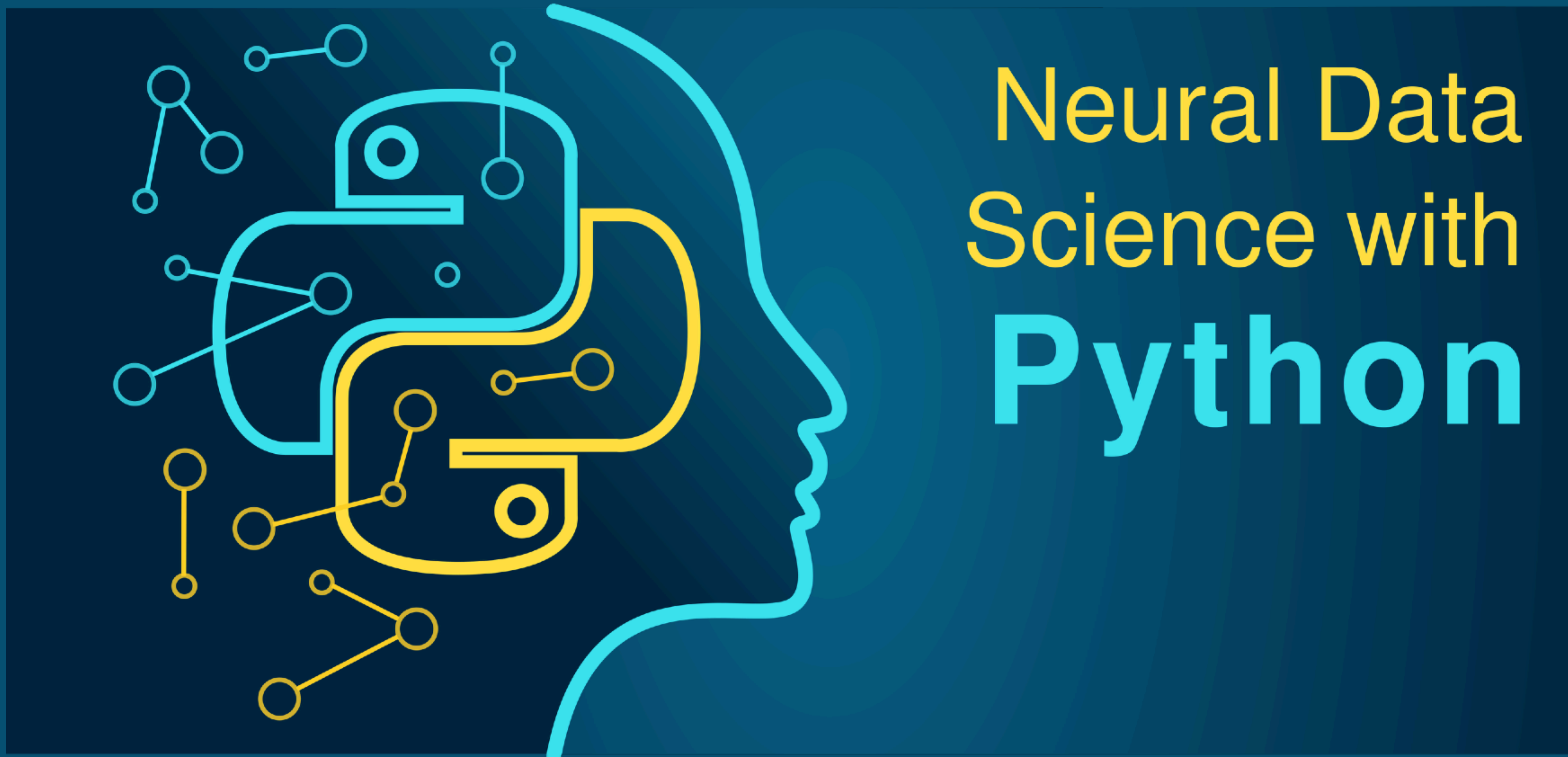
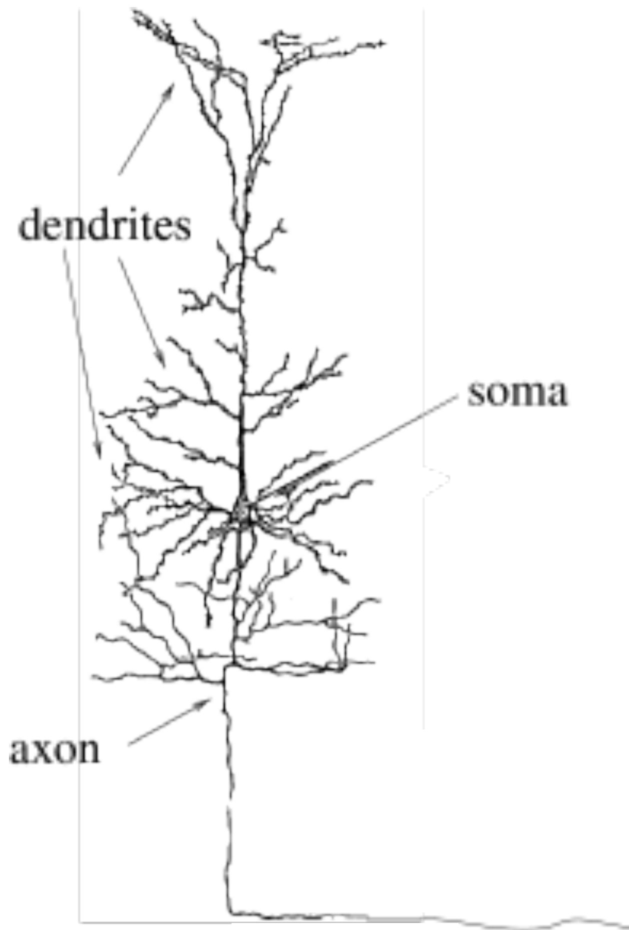


Biophysical neuron models: The Integrate-and-Fire neuron

Jonas Ranft
Institut de Biologie de l'ENS (IBENS)

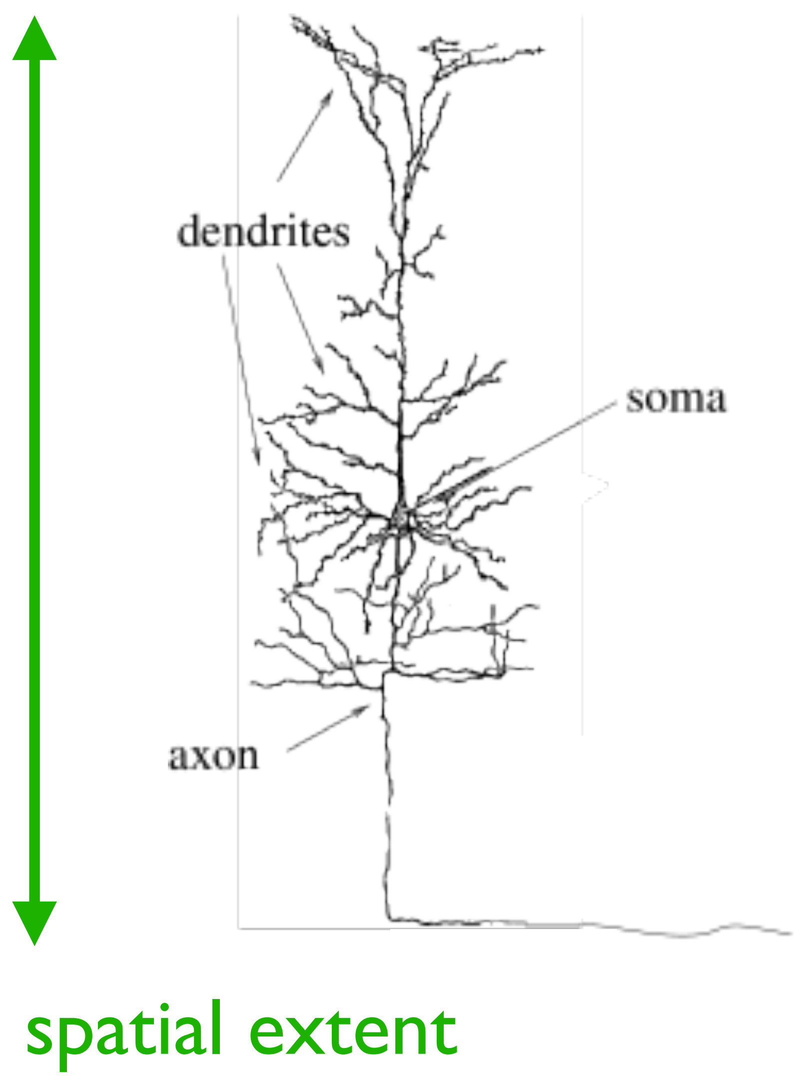
What is a biophysical model?

- Let's have a look at a typical cortical neuron:



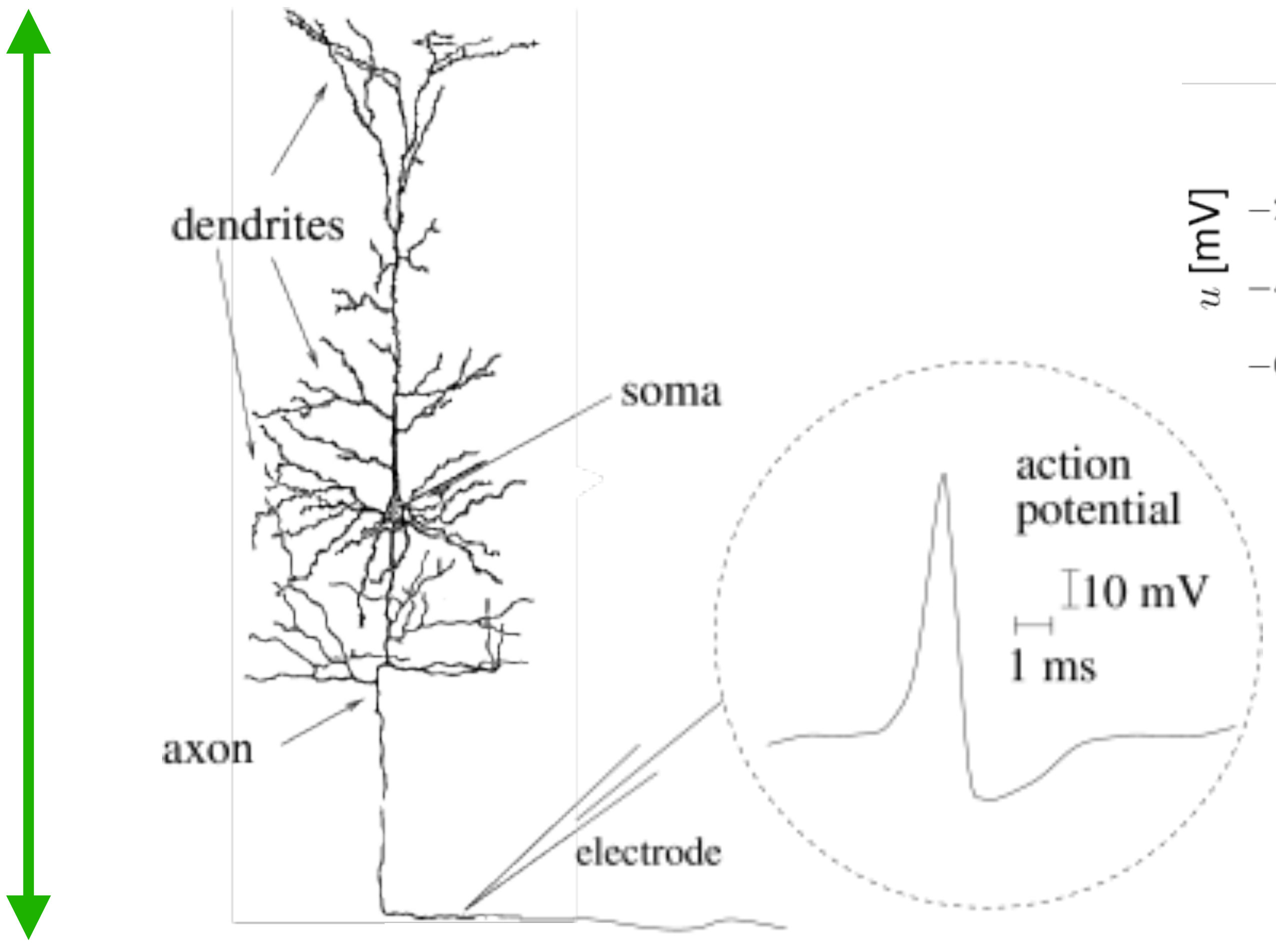
What is a biophysical model?

- Let’s have a look at a typical cortical neuron:

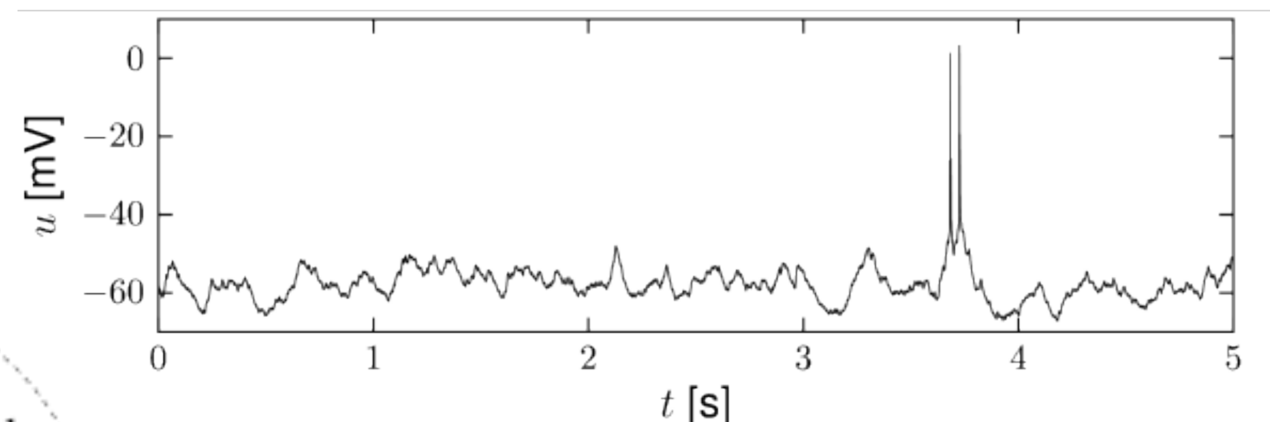


What is a biophysical model?

- Let’s have a look at a typical cortical neuron:

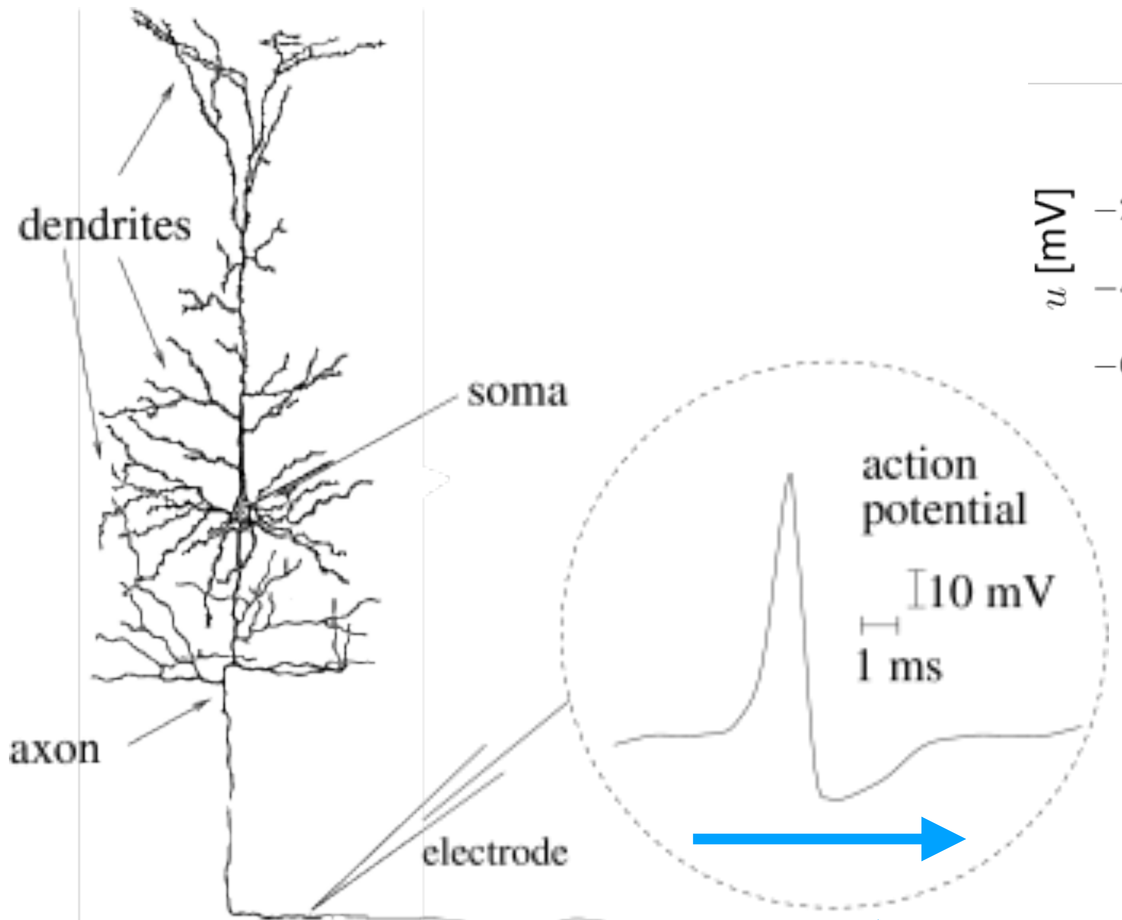


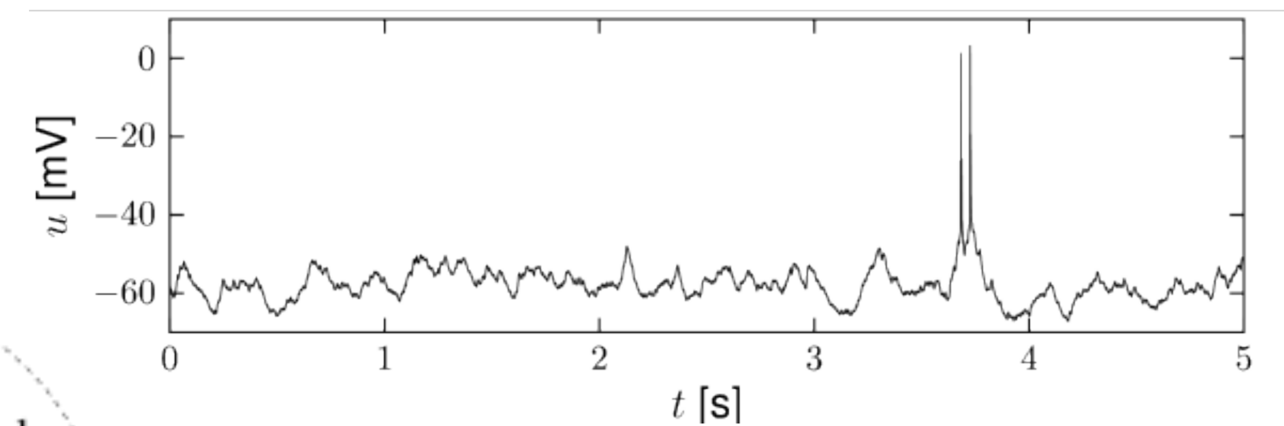
spatial extent



What is a biophysical model?

- Let’s have a look at a typical cortical neuron:



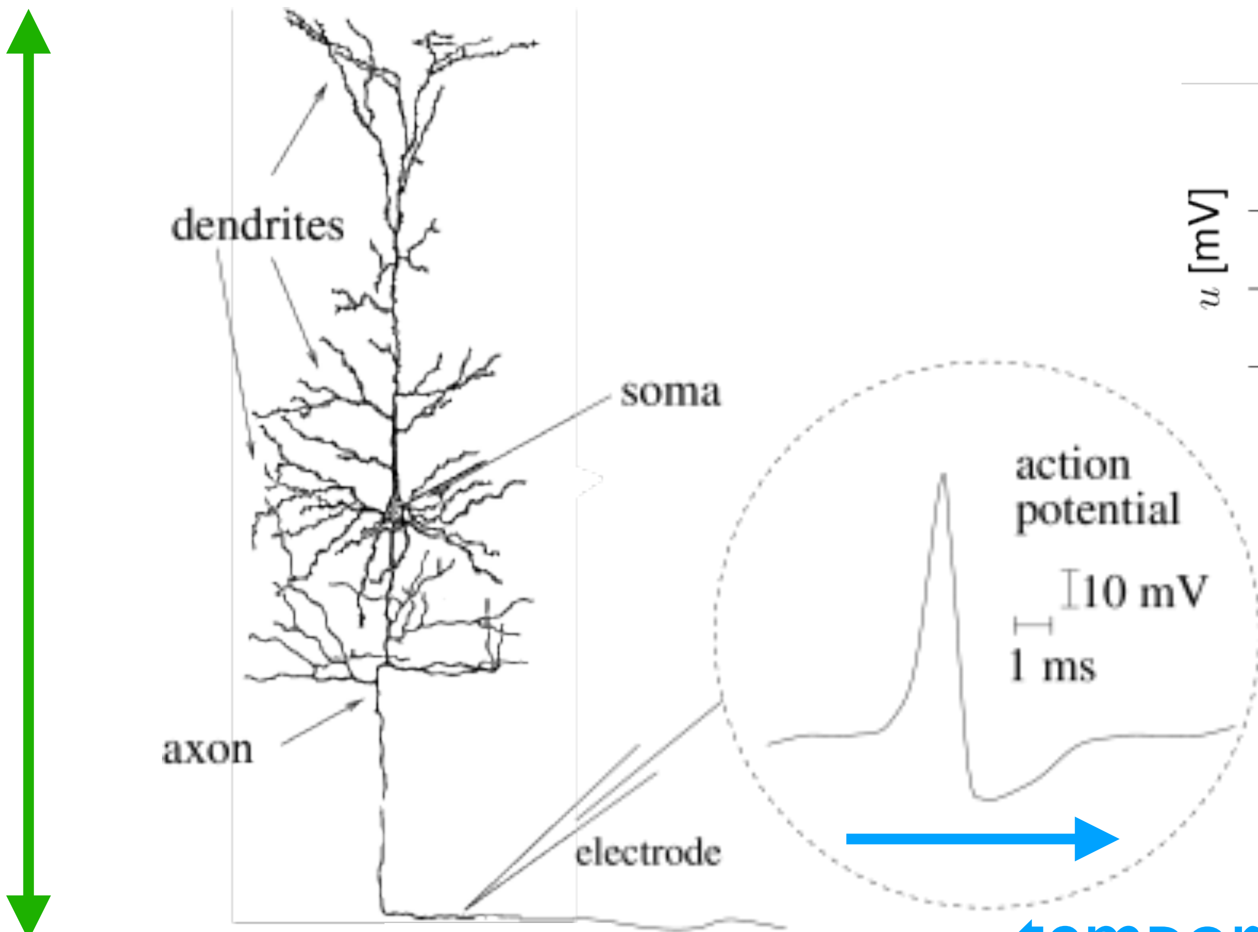


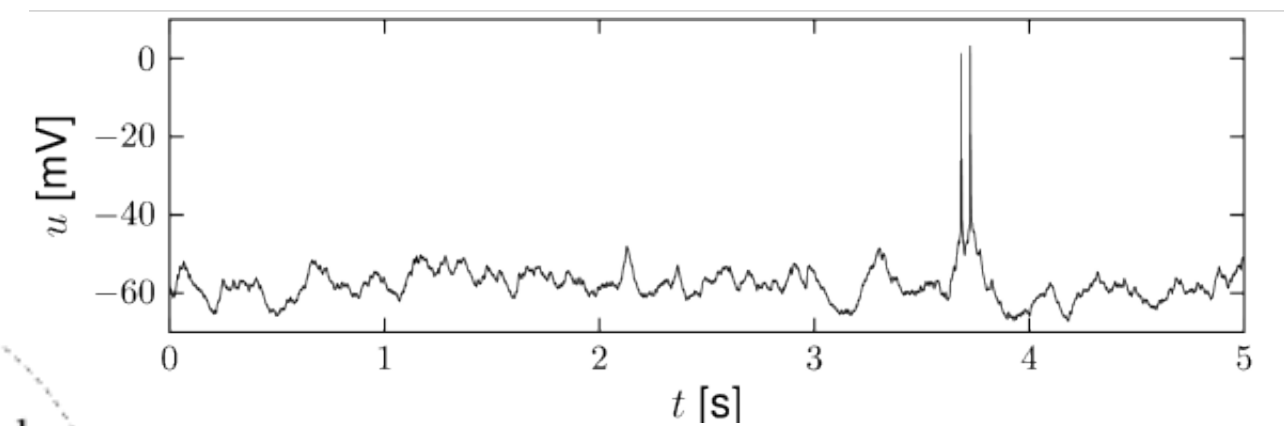
temporal dynamics

spatial extent

What is a biophysical model?

- Let's have a look at a typical cortical neuron:





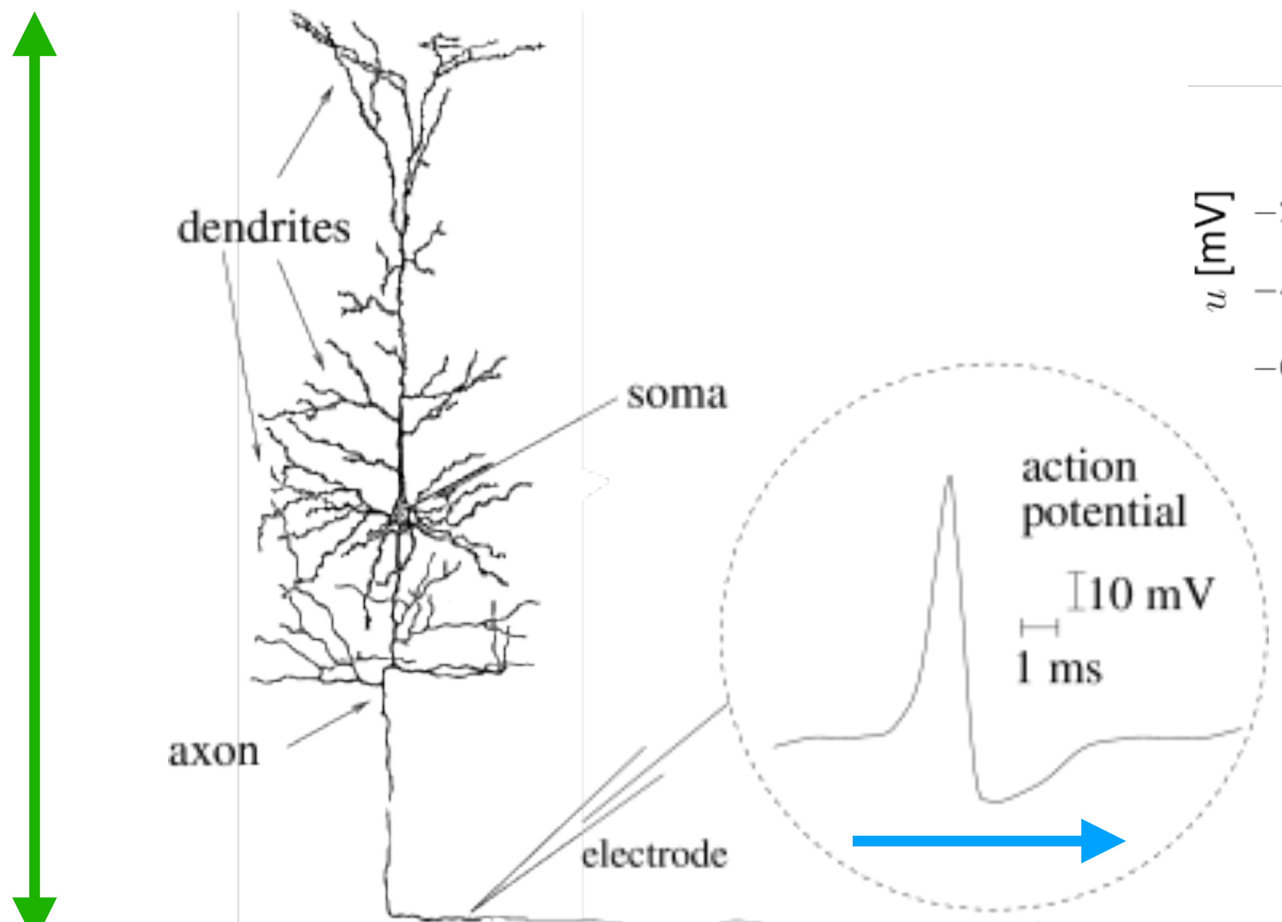
"the neuron's activity"

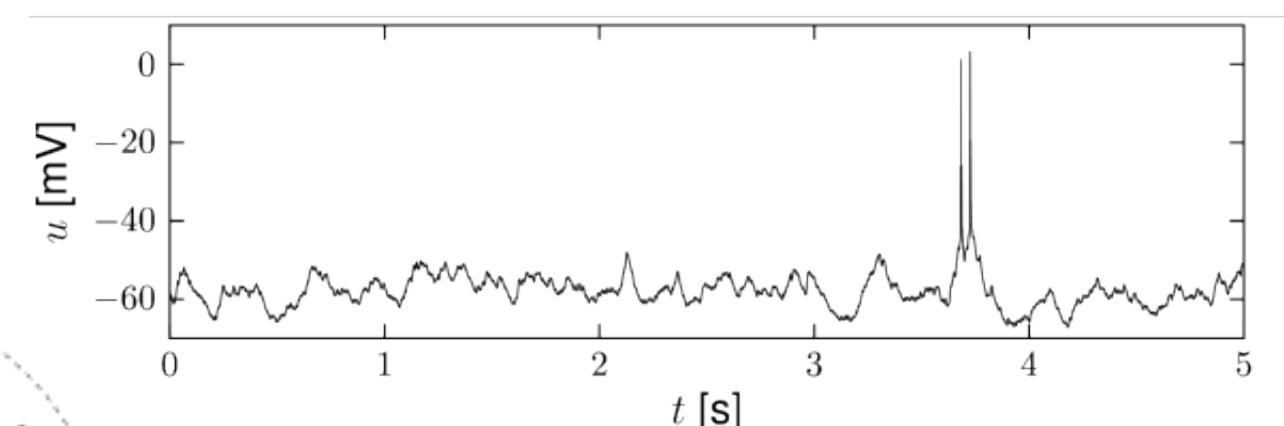
temporal dynamics

spatial extent

What is a biophysical model?

- Let's have a look at a typical cortical neuron:





"the neuron's activity"

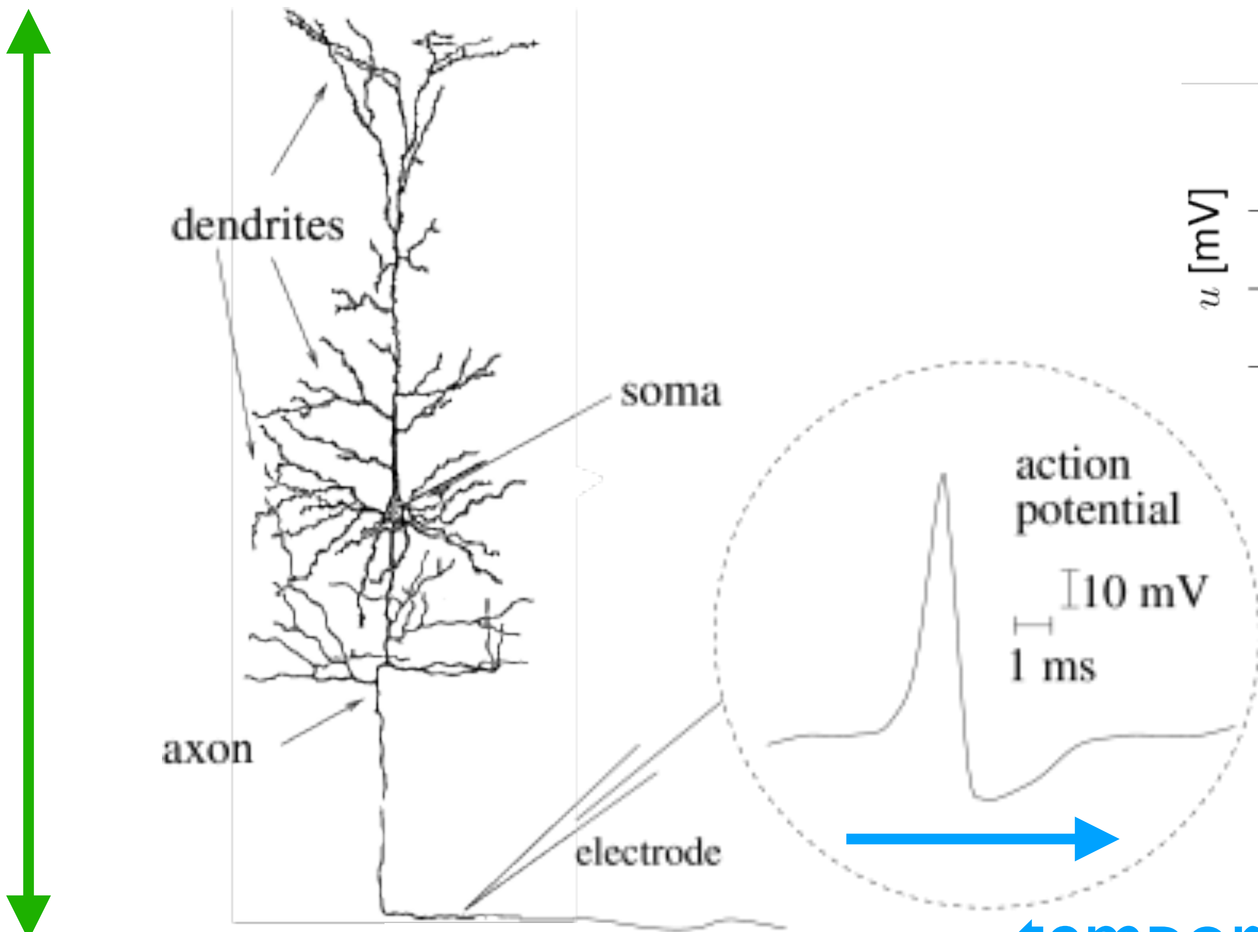
temporal dynamics

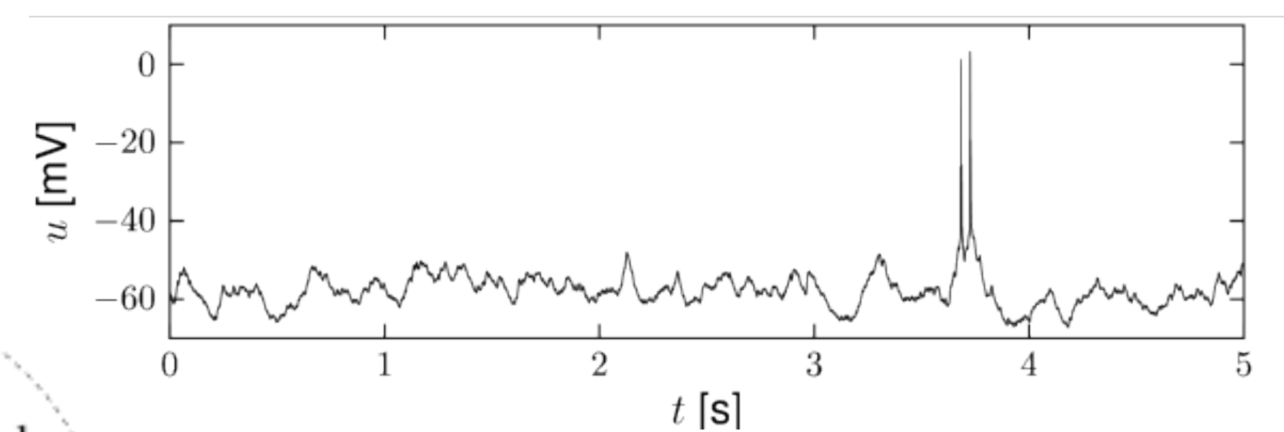
spatial extent

What does it mean to "model" neurons?

What is a biophysical model?

- Let’s have a look at a typical cortical neuron:





“the neuron’s activity”

temporal dynamics

spatial extent

What does it mean to “model” neurons?

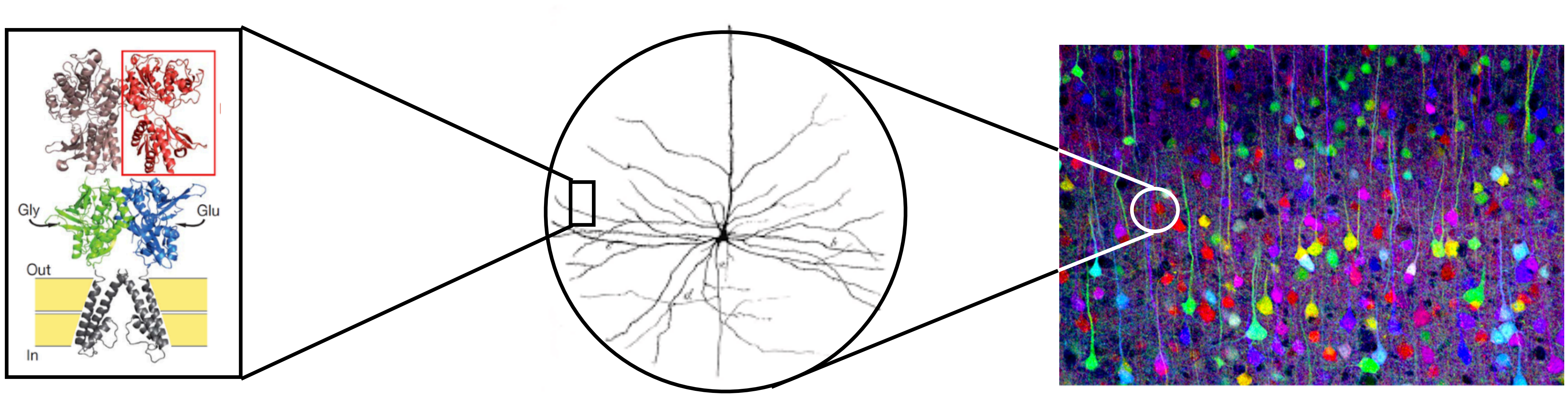
Why would we try to do so?

The power of models

1. Explain ("make sense of observations")
2. Predict ("let's test the theory")

The power of models

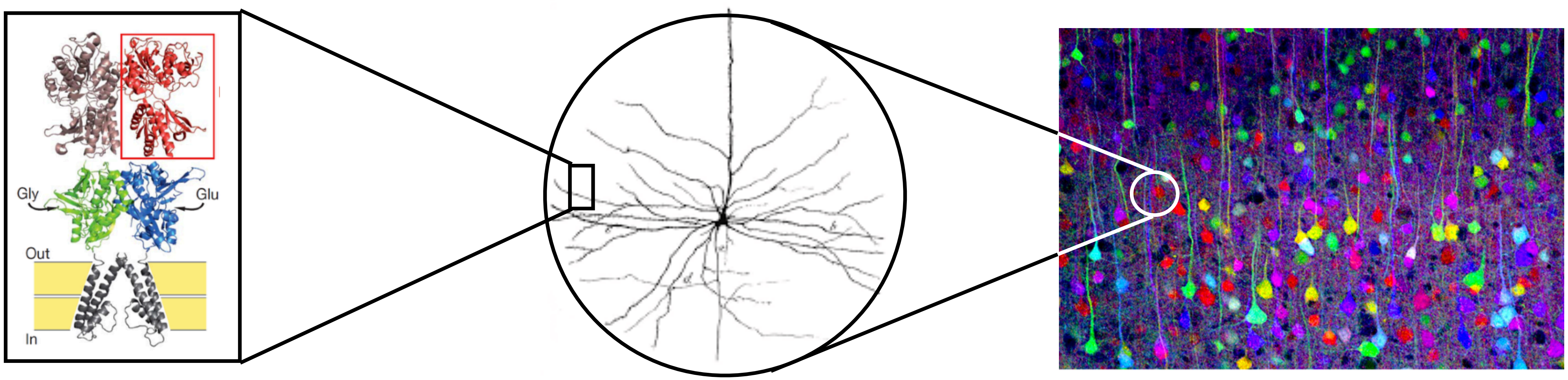
1. Explain (“make sense of observations”)
2. Predict (“let’s test the theory”)



NMDA receptor   pyramidal cell   cortical network

The power of models

1. Explain ("make sense of observations")
2. Predict ("let's test the theory")



NMDA receptor

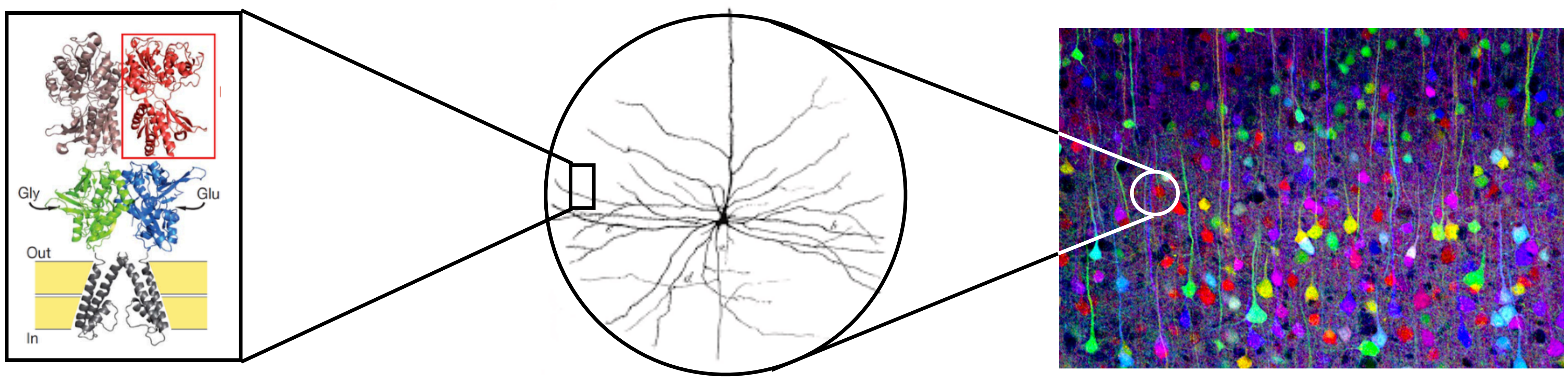
pyramidal cell

cortical network

- Every model is limited to a given spatial and temporal *scale*.

The power of models

1. Explain (“make sense of observations”)
2. Predict (“let’s test the theory”)



NMDA receptor

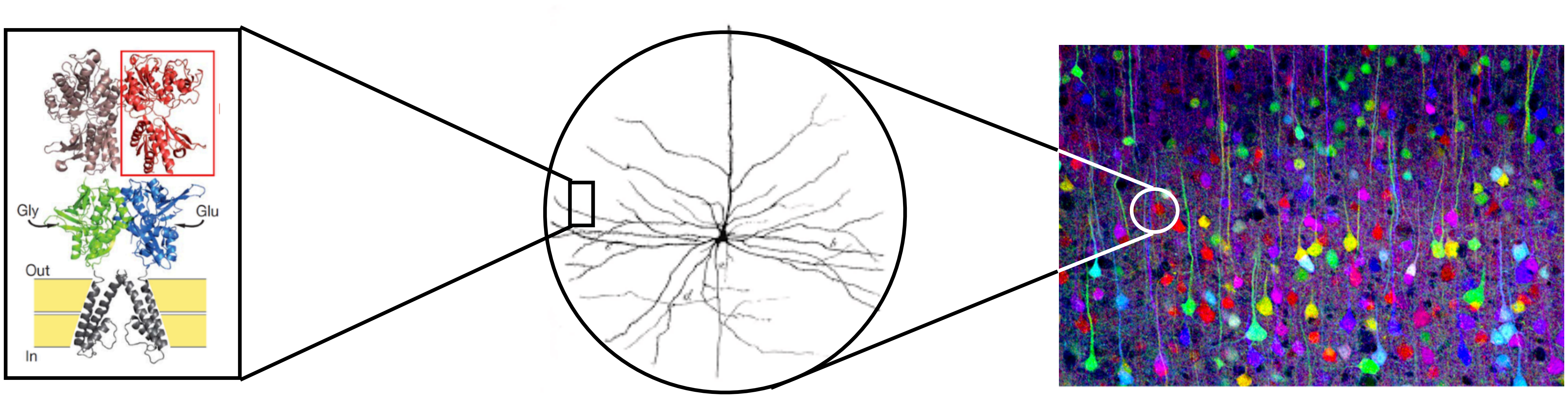
pyramidal cell

cortical network

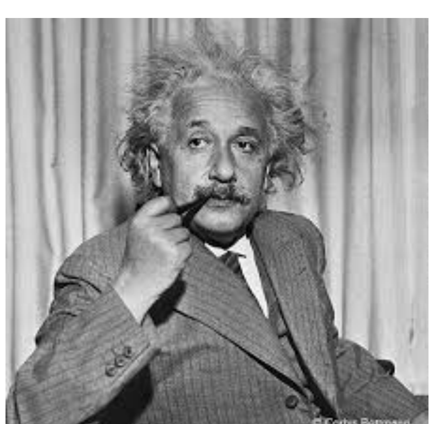
- Every model is limited to a given spatial and temporal *scale*.
- The scale depends on *what* we want to explain or predict.

The power of models

1. Explain ("make sense of observations")
2. Predict ("let's test the theory")



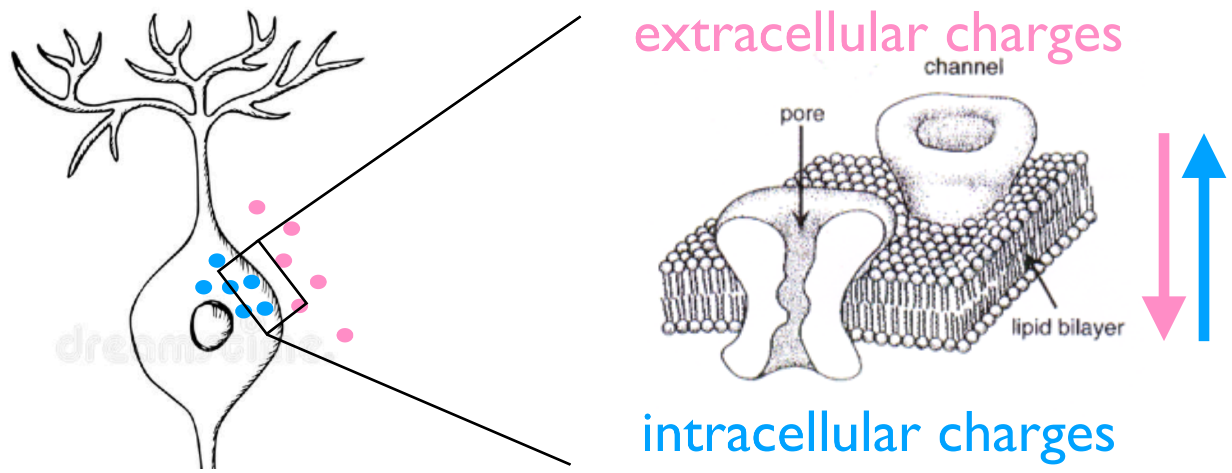
NMDA receptor   pyramidal cell   cortical network



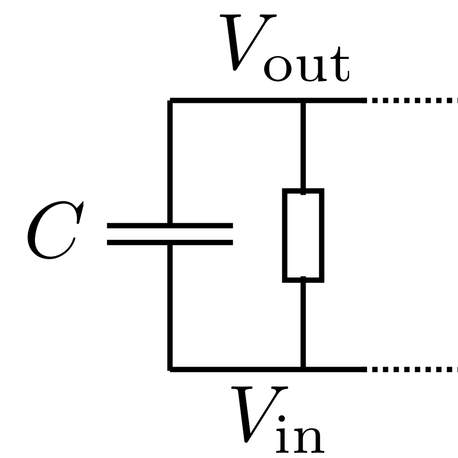
"Everything should be made as simple as possible, but no simpler."

Biophysical neuron models

- <u>Physical quantities</u>: voltages, currents (electric circuits)



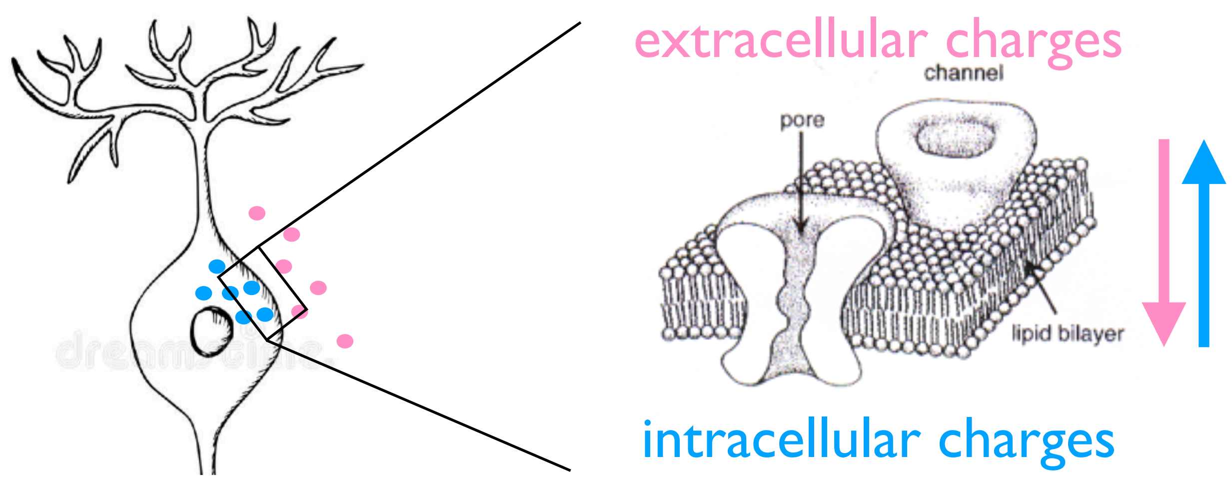
lipid bilayer acts like a capacitance



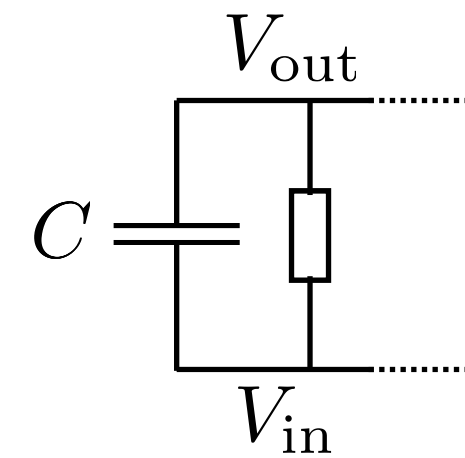
$$V_{\text{membrane}} = V_{\text{in}} - V_{\text{out}}$$

Biophysical neuron models

- <u>Physical quantities</u>: voltages, currents (electric circuits)



lipid bilayer acts like a capacitance



$$V_{\text{membrane}} = V_{\text{in}} - V_{\text{out}}$$

- <u>Description</u>: differential equations (mathematics)

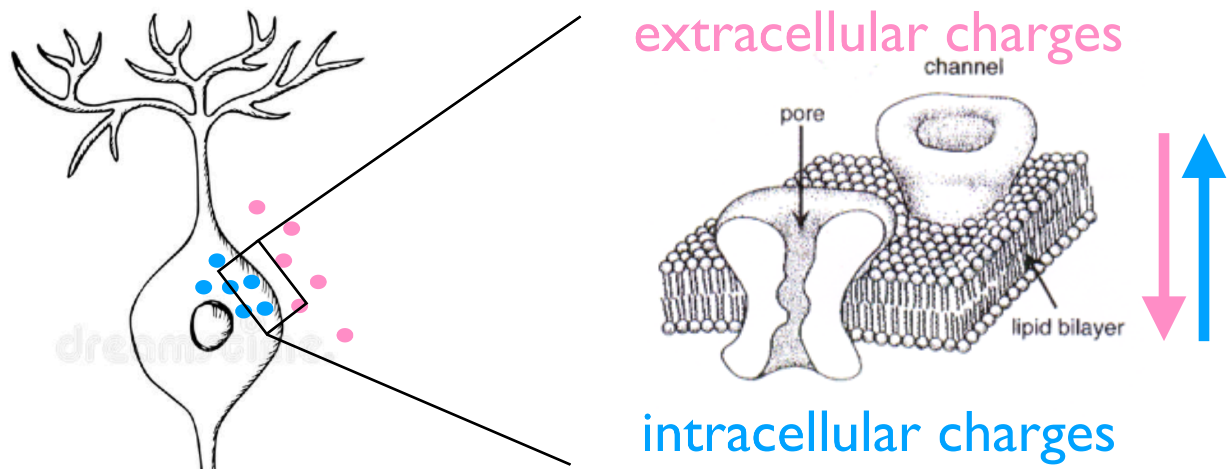
Ohm's law: $I_R = \dfrac{V}{R} = gV$

Capacitive current: $I_C = C\dfrac{\mathrm{d}V}{\mathrm{d}t}$

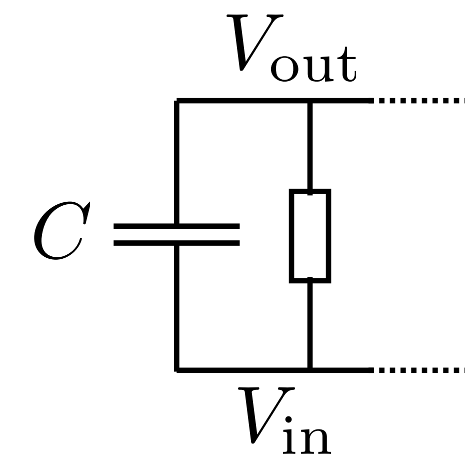
Kirchhoff's law: $I_1 + I_2 + I_3 + \cdots = 0$

Biophysical neuron models

- <u>Physical quantities</u>: voltages, currents (electric circuits)



lipid bilayer acts like a capacitance



$$V_{\mathrm{membrane}} = V_{\mathrm{in}} - V_{\mathrm{out}}$$

- <u>Description</u>: differential equations (mathematics)

Ohm's law: $I_R = \dfrac{V}{R} = gV$

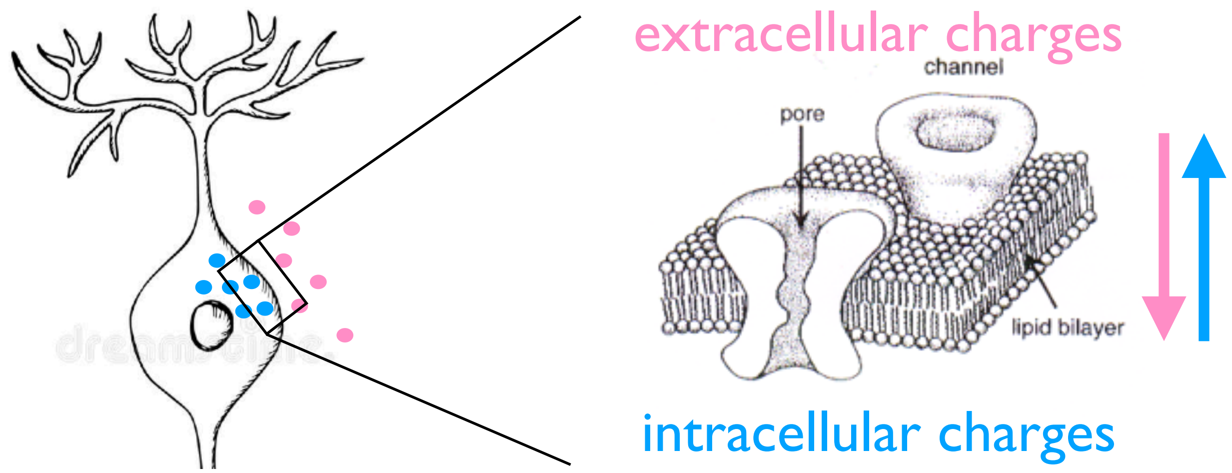
Capacitive current: $I_C = C\dfrac{\mathrm{d}V}{\mathrm{d}t}$

Kirchhoff's law: $I_1 + I_2 + I_3 + \cdots = 0$

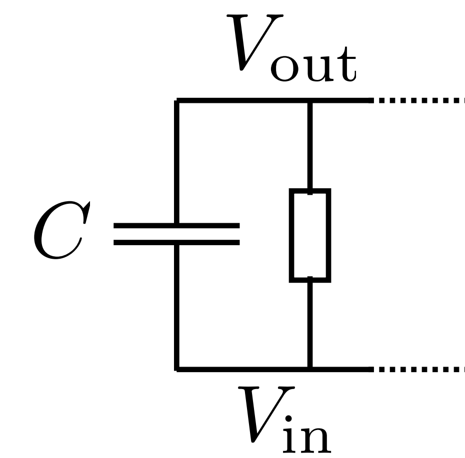
- <u>Methods</u>: analytical and numerical calculations, simulations

Biophysical neuron models

- Physical quantities: voltages, currents (electric circuits)



lipid bilayer acts like a capacitance



$$V_{\mathrm{membrane}} = V_{\mathrm{in}} - V_{\mathrm{out}}$$

- Description: differential equations (mathematics)

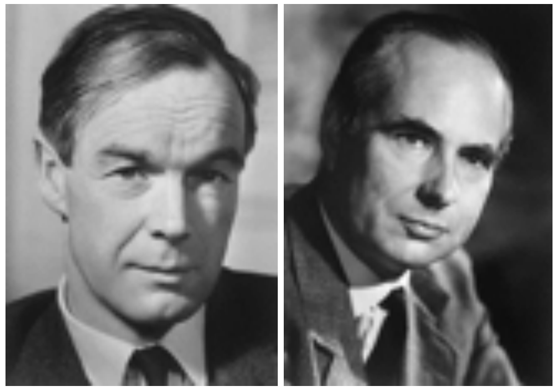
Ohm's law: $I_R = \dfrac{V}{R} = gV$

Capacitive current: $I_C = C\dfrac{\mathrm{d}V}{\mathrm{d}t}$

Kirchhoff's law: $I_1 + I_2 + I_3 + \cdots = 0$

- Methods: analytical and numerical calculations, simulations

TD



Hodgkin Huxley

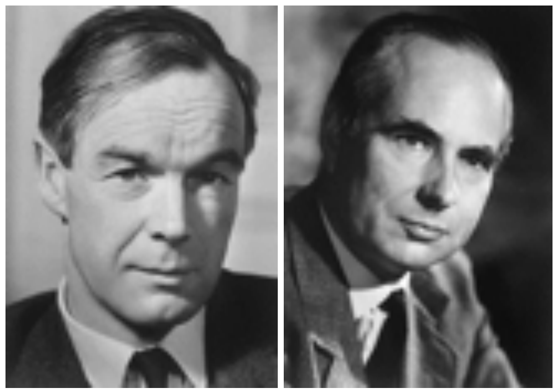
1. Hodgkin-Huxley model

Nobel prize 1963

Lapicque

2. Integrate-and-Fire model

first proposed in 1907



Hodgkin Huxley

1. Hodgkin-Huxley model

Nobel prize 1963

too complicated for today...

Lapicque

2. Integrate-and-Fire model

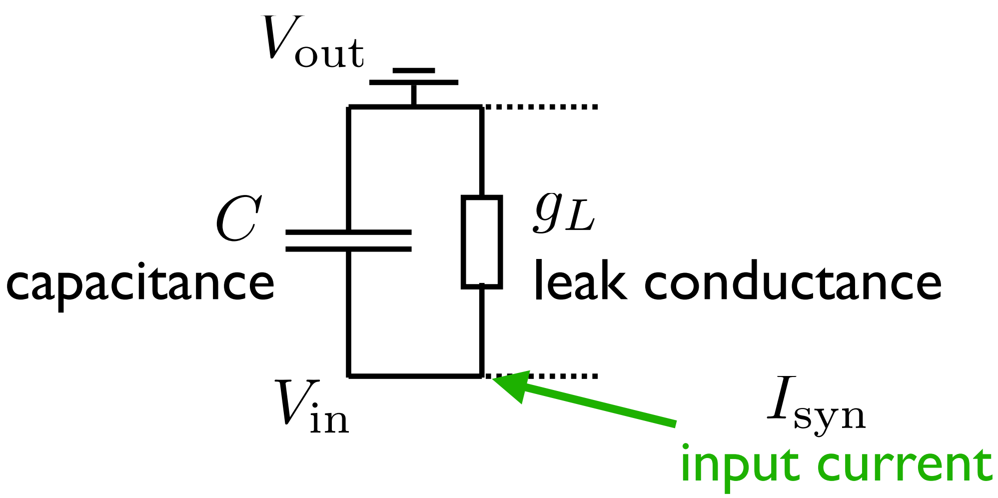
first proposed in 1907

The Integrate-and-Fire model

- **Aim:** Describe *spike generation* in response to *synaptic inputs*, using a model for the *membrane potential dynamics*

The Integrate-and-Fire model

- **Aim:** Describe *spike generation* in response to *synaptic inputs*, using a model for the *membrane potential dynamics*
- Consider only the passive electrical properties of the neuronal membrane:

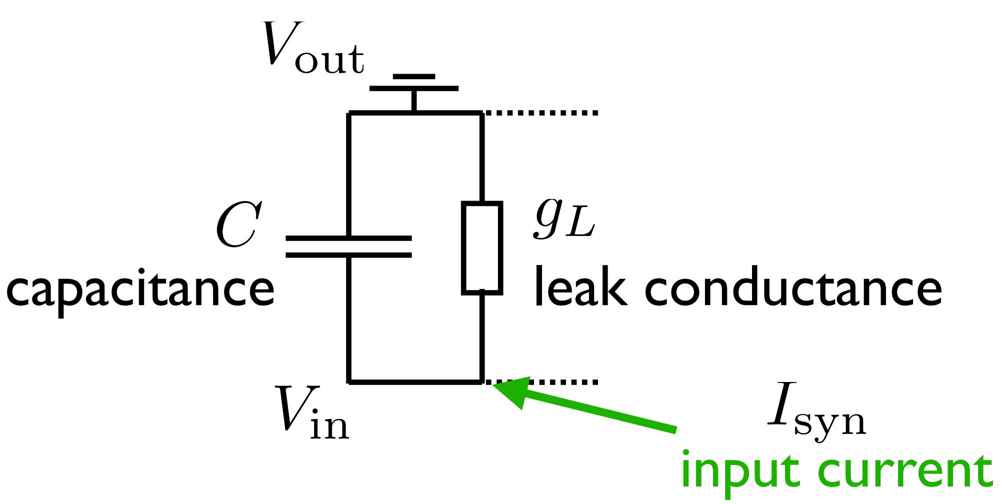


$$V = V_{\rm in} - V_{\rm out}$$

$$C\frac{dV}{dt} = g_L(E_L - V) + I_{\rm syn}$$

The Integrate-and-Fire model

- **Aim:** Describe *spike generation* in response to *synaptic inputs*, using a model for the *membrane potential dynamics*
- Consider only the passive electrical properties of the neuronal membrane:



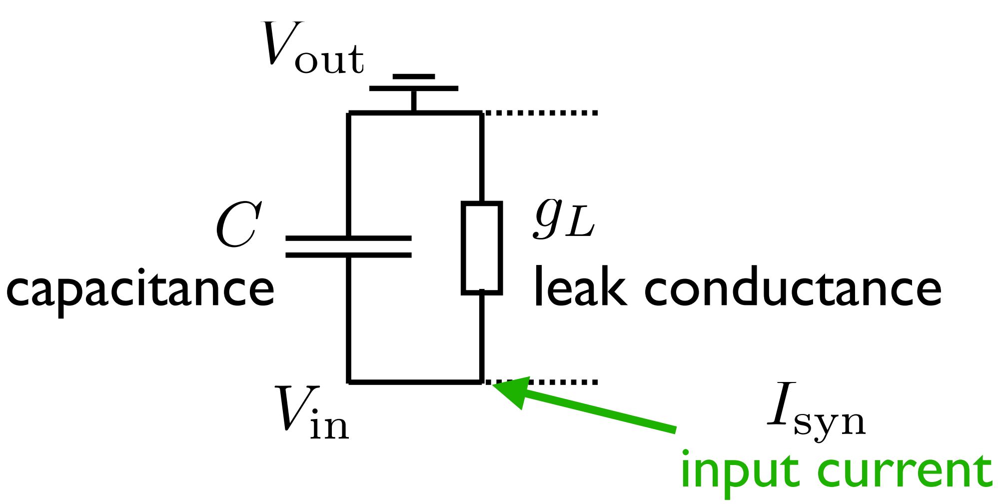
$$V = V_{\rm in} - V_{\rm out}$$

$$\boxed{C\frac{dV}{dt} = g_L(E_L - V) + I_{\rm syn}}$$

evolution equation for V(t)

The Integrate-and-Fire model

- **Aim:** Describe *spike generation* in response to *synaptic inputs*, using a model for the *membrane potential dynamics*
- Consider only the passive electrical properties of the neuronal membrane:



$$V = V_{\text{in}} - V_{\text{out}}$$

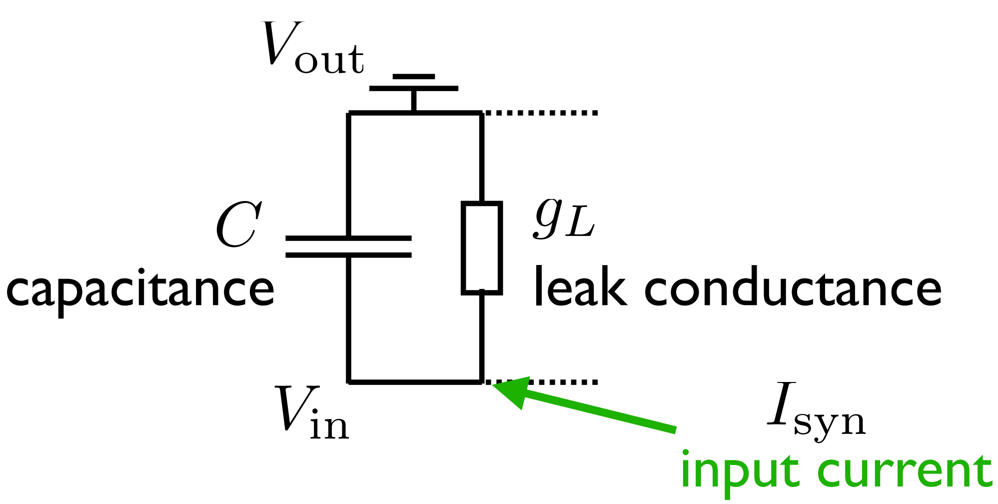
$$\boxed{C\frac{dV}{dt} = g_L(E_L - V) + I_{\text{syn}}}$$

evolution equation for V(t)

formal solution: $V(t) = E_L + \frac{1}{g_L \tau_m}\int_{-\infty}^{t} dt'\; e^{-\frac{t-t'}{\tau_m}} I_{\text{syn}}(t')$

The Integrate-and-Fire model

- **Aim:** Describe *spike generation* in response to *synaptic inputs*, using a model for the *membrane potential dynamics*
- Consider only the passive electrical properties of the neuronal membrane:



$$V = V_{\text{in}} - V_{\text{out}}$$

$$\boxed{C\frac{dV}{dt} = g_L(E_L - V) + I_{\text{syn}}}$$ *evolution equation for V(t)*

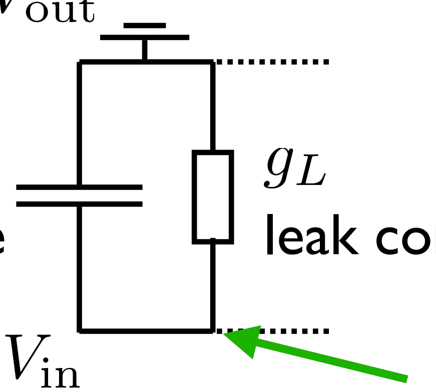
formal solution: $V(t) = E_L + \frac{1}{g_L \tau_m}\int_{-\infty}^{t} dt' \; e^{-\frac{t-t'}{\tau_m}} I_{\text{syn}}(t')$

→ *leaky* integrator with $\tau_m = C/g_L$ membrane time constant

The Integrate-and-Fire model

- **Aim:** Describe *spike generation* in response to *synaptic inputs*, using a model for the *membrane potential dynamics*
- Consider only the passive electrical properties of the neuronal membrane:

$$V = V_{\text{in}} - V_{\text{out}}$$



$$\boxed{C\frac{dV}{dt} = g_L(E_L - V) + I_{\text{syn}}}$$

evolution equation for V(t)

formal solution: $V(t) = E_L + \frac{1}{g_L \tau_m}\int_{-\infty}^{t} dt' \; e^{-\frac{t-t'}{\tau_m}} I_{\text{syn}}(t')$

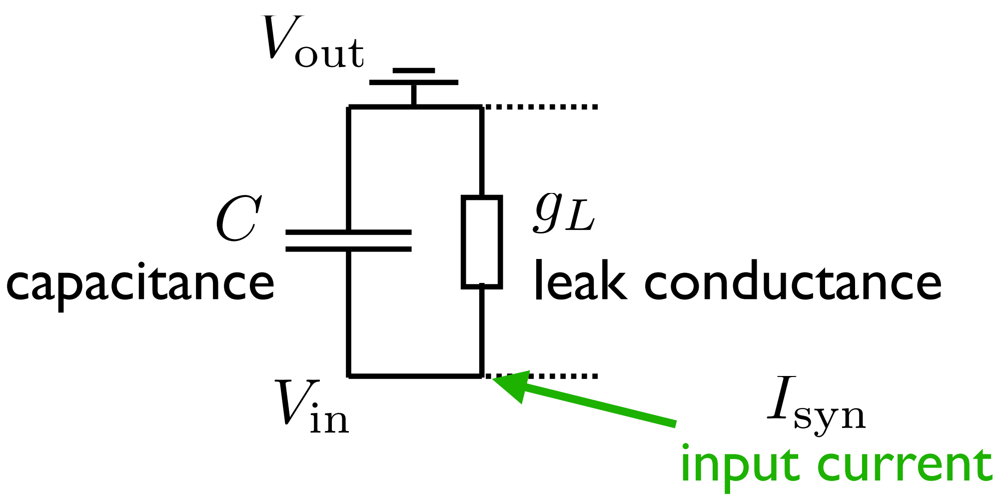
→ *leaky* integrator with $\tau_m = C/g_L$ membrane time constant

but: no spikes?!?

The Integrate-and-Fire model

- **Aim:** Describe *spike generation* in response to *synaptic inputs*, using a model for the *membrane potential dynamics*
- Consider only the passive electrical properties of the neuronal membrane:

$$V = V_{\text{in}} - V_{\text{out}}$$



$$\boxed{C \frac{dV}{dt} = g_L (E_L - V) + I_{\text{syn}}}$$ *evolution equation for V(t)*

formal solution: $V(t) = E_L + \frac{1}{g_L \tau_m} \int_{-\infty}^{t} dt' \; e^{-\frac{t-t'}{\tau_m}} I_{\text{syn}}(t')$

$\longrightarrow$ *leaky* integrator with $\tau_m = C/g_L$ membrane time constant

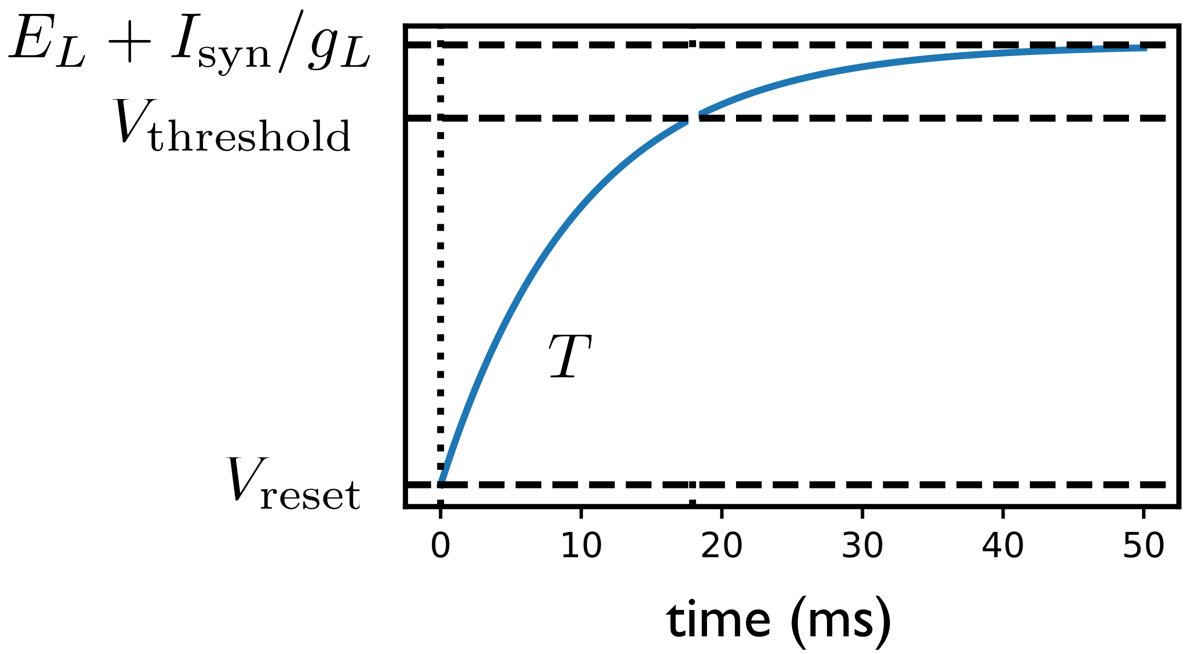
- *Spike emission* whenever the voltage crosses a threshold, followed by a reset:

$$\boxed{V(t) > V_{\text{threshold}} \quad \rightarrow \quad \text{spike} + V(t) = V_{\text{reset}}}$$

Analyzing the LIF model

LIF = Leaky Integrate-and-Fire

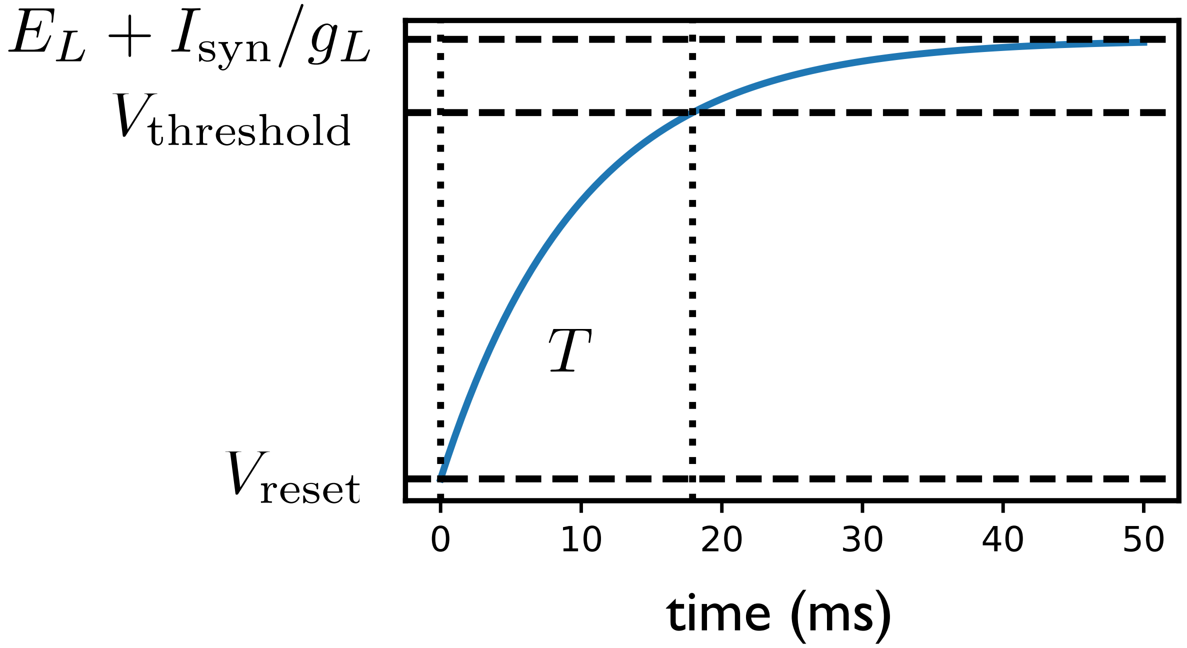
- We can e.g. calculate the firing rate as a function of the input current (if constant):



Analyzing the LIF model

LIF = Leaky Integrate-and-Fire

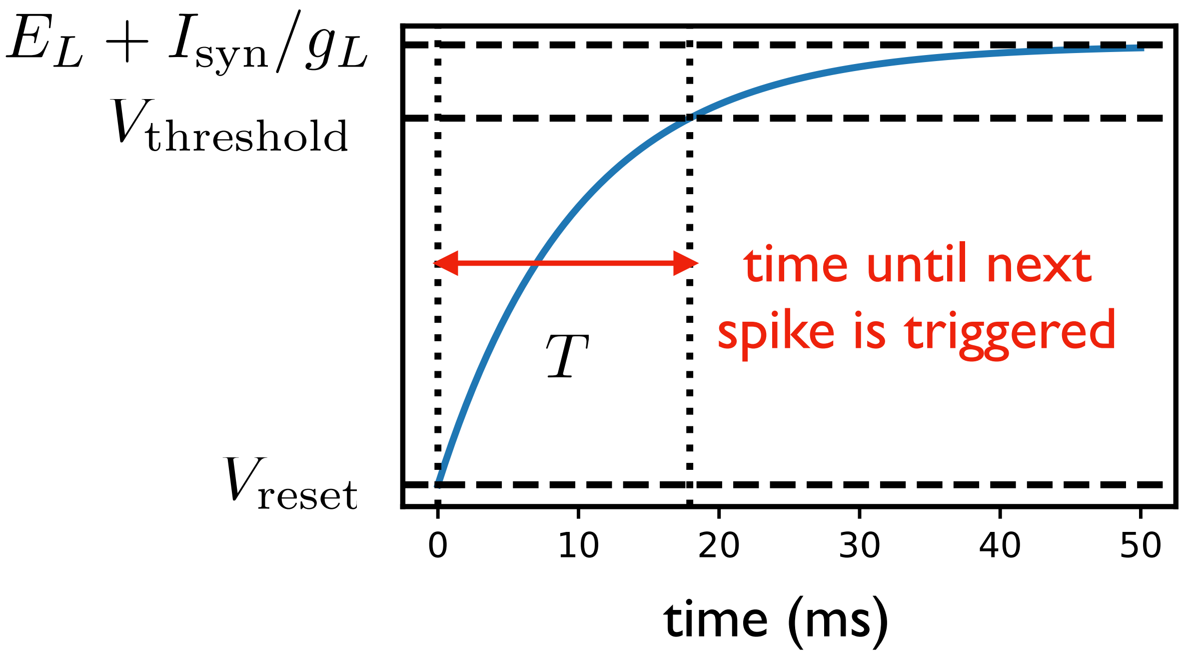
- We can e.g. calculate the firing rate as a function of the input current (if constant):



Analyzing the LIF model

LIF = Leaky Integrate-and-Fire

- We can e.g. calculate the firing rate as a function of the input current (if constant):

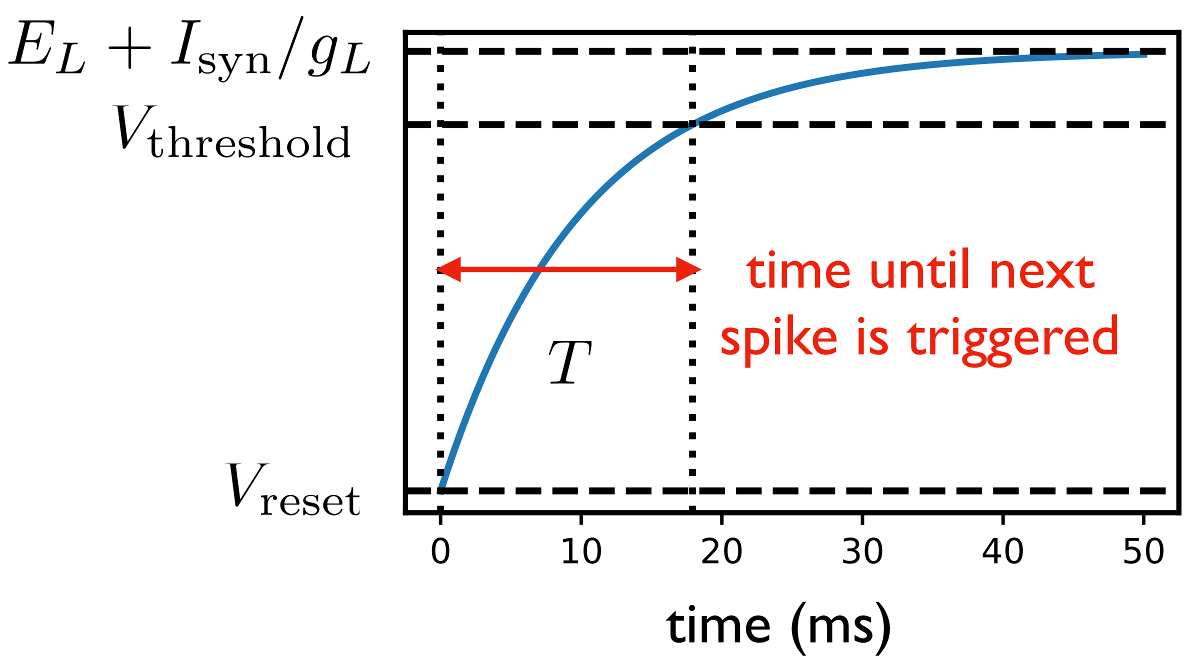


Analyzing the LIF model

LIF = Leaky Integrate-and-Fire

- We can e.g. calculate the firing rate as a function of the input current (if constant):

$$V(T) = V_{\mathrm{threshold}} \quad \Rightarrow \quad T = T(I_{\mathrm{syn}}, \ldots)$$

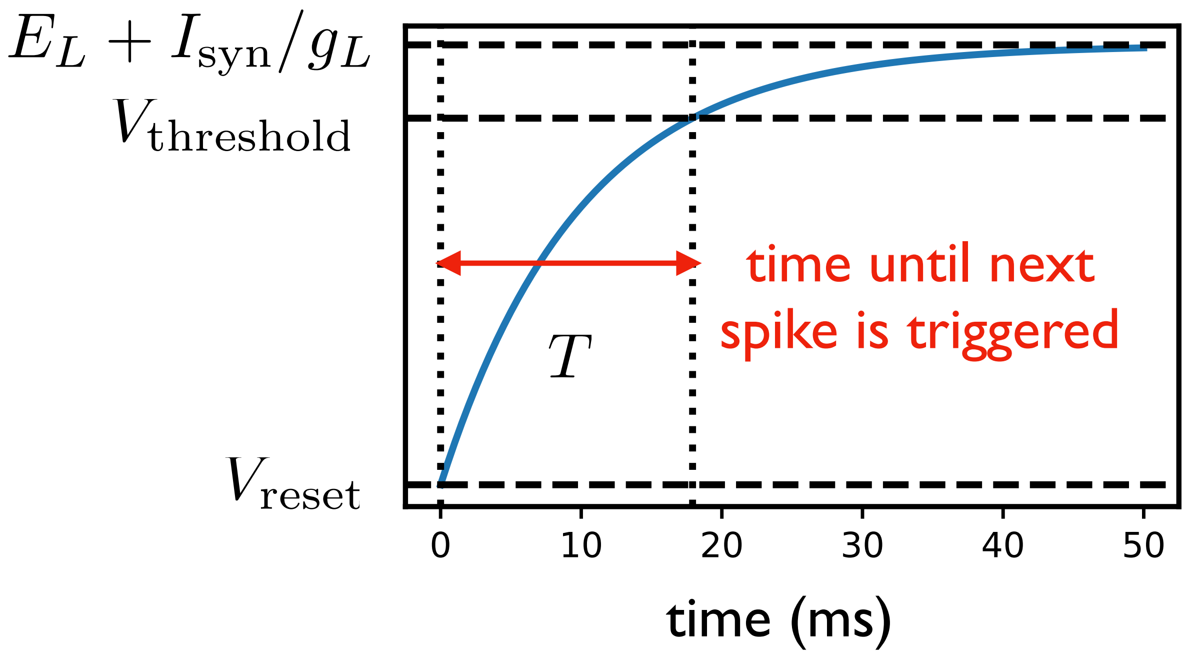


Analyzing the LIF model

LIF = Leaky Integrate-and-Fire

- We can e.g. calculate the firing rate as a function of the input current (if constant):

$$V(T) = V_{\text{threshold}} \quad \Rightarrow \quad T = T(I_{\text{syn}}, \ldots)$$



$$\boxed{\begin{aligned} f(I) &= \frac{1}{T(I)} \\ &= \frac{1}{\tau_m}\left(\log \frac{E_L + I/g_L - V_{\text{reset}}}{E_L + I/g_L - V_{\text{threshold}}}\right)^{-1} \end{aligned}}$$

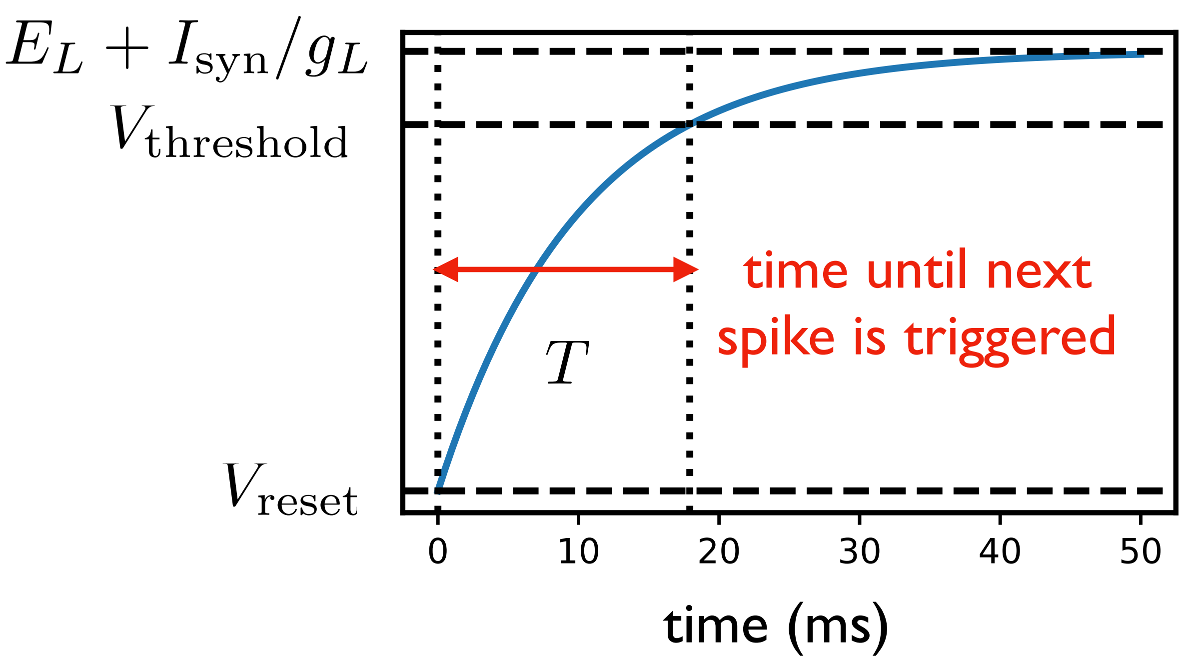
transfer function, or f-I curve

Analyzing the LIF model

LIF = Leaky Integrate-and-Fire

- We can e.g. calculate the firing rate as a function of the input current (if constant):

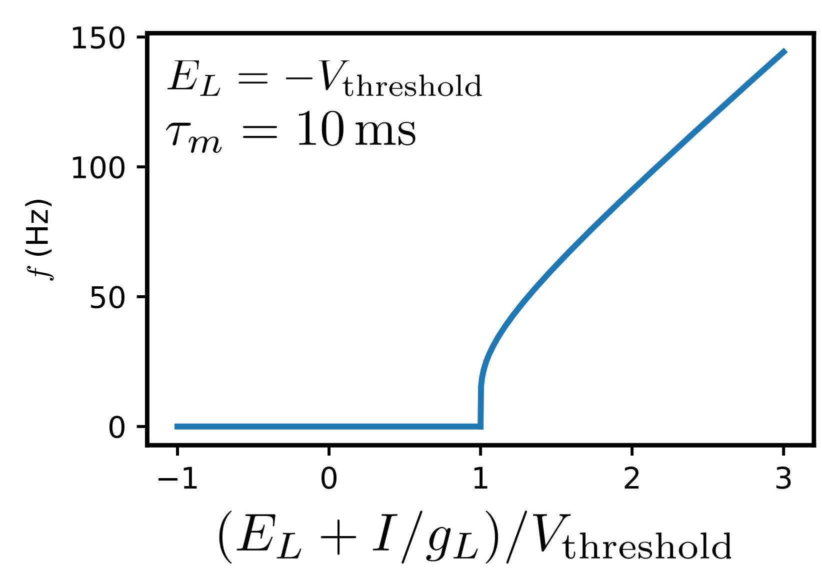
$$V(T) = V_{\text{threshold}} \quad \Rightarrow \quad T = T(I_{\text{syn}}, \ldots)$$



$$\boxed{\begin{aligned} f(I) &= \frac{1}{T(I)} \\ &= \frac{1}{\tau_m}\left(\log \frac{E_L + I/g_L - V_{\text{reset}}}{E_L + I/g_L - V_{\text{threshold}}}\right)^{-1} \end{aligned}}$$

transfer function, or f-I curve

- Above a sharp *firing threshold*, the firing rate increases with increasing input current.

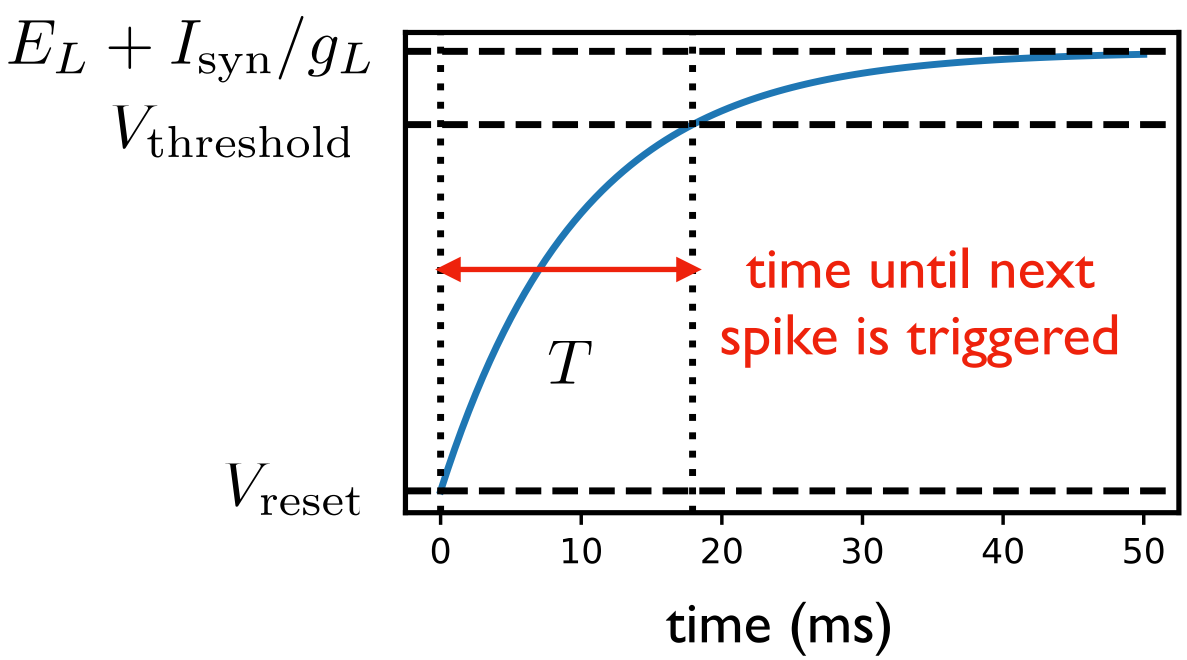


Analyzing the LIF model

LIF = Leaky Integrate-and-Fire

- We can e.g. calculate the firing rate as a function of the input current (if constant):

$$V(T) = V_{\text{threshold}} \quad \Rightarrow \quad T = T(I_{\text{syn}}, \ldots)$$

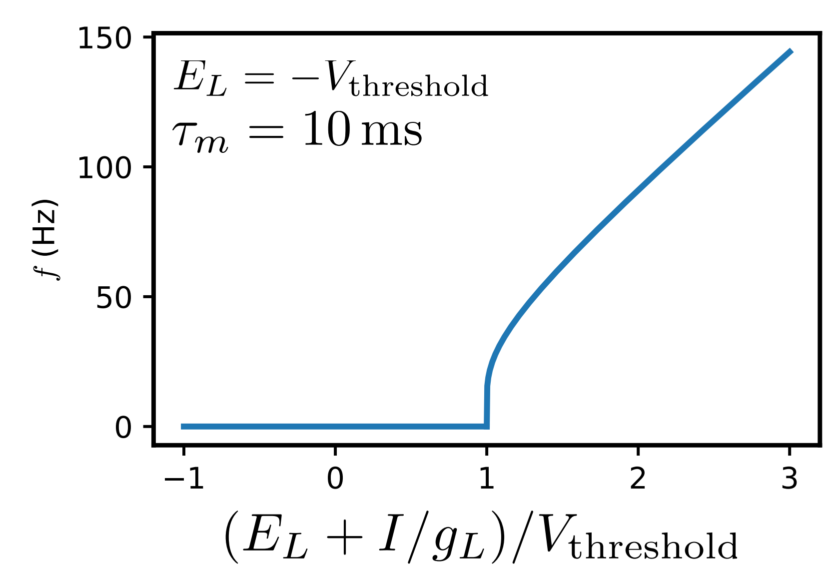


$$\boxed{\begin{aligned} f(I) &= \frac{1}{T(I)} \\ &= \frac{1}{\tau_m} \left(\log \frac{E_L + I/g_L - V_{\text{reset}}}{E_L + I/g_L - V_{\text{threshold}}} \right)^{-1} \end{aligned}}$$

transfer function, or f-I curve

- Above a sharp *firing threshold*, the firing rate increases with increasing input current.

Here, we considered a constant current, but of course input is typically time-dependent.



Where LIFs are useful

- Because of their relative simplicity, we can use many connected LIF neurons to model networks of spiking neurons:

$$C\frac{dV_i}{dt} = g_L(E_L - V_i) + I_{\mathrm{syn},i} \qquad I_{\mathrm{syn},i}(t) = \sum_j I_{ij}(t) \qquad i = 1, \ldots, N$$

sum over presynaptic neurons

Where LIFs are useful

- Because of their relative simplicity, we can use many connected LIF neurons to model networks of spiking neurons:

$$C\frac{dV_i}{dt} = g_L(E_L - V_i) + I_{\mathrm{syn},i} \qquad I_{\mathrm{syn},i}(t) = \sum_j I_{ij}(t) \qquad i = 1, \ldots, N$$

sum over presynaptic neurons

- Different possibilities to generate I_{syn} from presynaptic activity.

  <u>Simplest choice</u>: increment postsynaptic voltage after each spike

$$I_{ij}(t) = J_{ij} S_j(t)$$

weight matrix   spike train of neuron *j*

Where LIFs are useful

- Because of their relative simplicity, we can use many connected LIF neurons to model networks of spiking neurons:

$$C\frac{dV_i}{dt} = g_L(E_L - V_i) + I_{\mathrm{syn},i} \qquad I_{\mathrm{syn},i}(t) = \sum_j I_{ij}(t) \qquad i = 1, \ldots, N$$

sum over presynaptic neurons

- Different possibilities to generate I_{syn} from presynaptic activity.

<u>*Simplest choice*</u>*: increment postsynaptic voltage after each spike*

$$I_{ij}(t) = J_{ij} S_j(t)$$

weight matrix   spike train of neuron *j*

- Neuron integrates excitatory and inhibitory post-synaptic currents of ***several 1000 synapses***.

LIF allows further mathematical analysis when approximating I(t) with mean + noise term:

$$I(t) = I_0 + \sigma\xi(t)$$

Where LIFs are useful

- Because of their relative simplicity, we can use many connected LIF neurons to model networks of spiking neurons:

$$C\frac{dV_i}{dt} = g_L(E_L - V_i) + I_{\mathrm{syn},i} \qquad I_{\mathrm{syn},i}(t) = \sum_j I_{ij}(t) \qquad i = 1, \ldots, N$$

sum over presynaptic neurons

- Different possibilities to generate I_{syn} from presynaptic activity.

  <u>*Simplest choice*</u>*: increment postsynaptic voltage after each spike*

$$I_{ij}(t) = J_{ij} S_j(t)$$

weight matrix   spike train of neuron *j*

- Neuron integrates excitatory and inhibitory post-synaptic currents of ***several 1000 synapses***.

  LIF allows further mathematical analysis when approximating I(t) with mean + noise term:

$$I(t) = I_0 + \sigma\xi(t)$$

- For general, time-dependent *I(t)*   numerical integration!

Where LIFs are useful

- Because of their relative simplicity, we can use many connected LIF neurons to model networks of spiking neurons:

$$C\frac{dV_i}{dt} = g_L(E_L - V_i) + I_{\mathrm{syn},i} \qquad I_{\mathrm{syn},i}(t) = \sum_j I_{ij}(t) \qquad i = 1, \ldots, N$$

sum over presynaptic neurons

- Different possibilities to generate I_{syn} from presynaptic activity.

  <u>Simplest choice</u>: increment postsynaptic voltage after each spike

$$I_{ij}(t) = J_{ij} S_j(t)$$

weight matrix   spike train of neuron *j*

- Neuron integrates excitatory and inhibitory post-synaptic currents of ***several 1000 synapses***.

  LIF allows further mathematical analysis when approximating I(t) with mean + noise term:

$$I(t) = I_0 + \sigma \xi(t)$$

- For general, time-dependent *I(t)* $\longrightarrow$ numerical integration!