SYDE 556/750

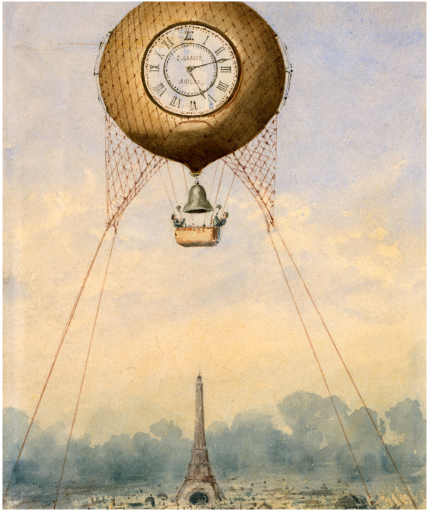
Simulating Neurobiological Systems
Lecture 4: Temporal Representations

Chris Eliasmith

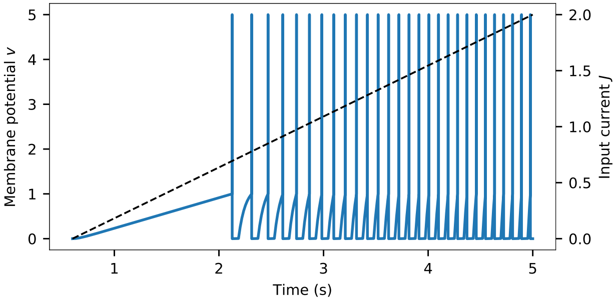
Sept 18 & 23, 2024

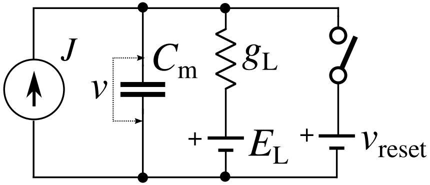
- Slide design: Andreas Stöckel
- Content: Terry Stewart, Andreas Stöckel, Chris Eliasmith

UNIVERSITY OF WATERLOO | FACULTY OF ENGINEERING



Reminder: The LIF Neuron





$$
\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t} v(t) &= -\frac{1}{\tau_\mathrm{RC}} \big(v(t) - J \big) \,, && \text{if } v(t) < 1 \,, \\
v(t) &\leftarrow \delta(t - t_\mathrm{th}) \,, && \text{if } t = t_\mathrm{th} \,, \\
v(t) &\leftarrow 0 \,, && \text{if } t > t_\mathrm{th} \text{ and } t \geq t_\mathrm{th} + \tau_\mathrm{ref} \,,
\end{aligned}
$$

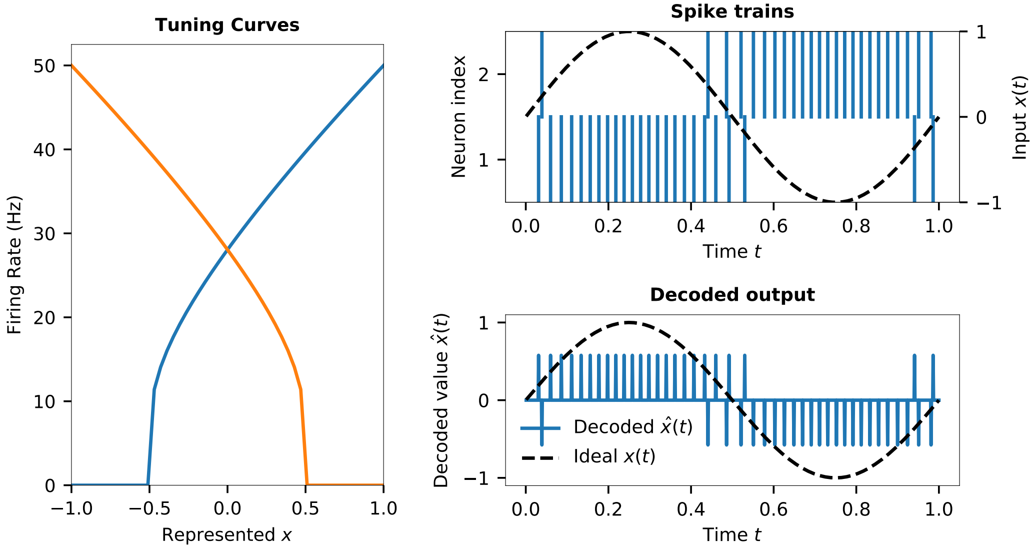
Temporal Decoding

- Our decoders to this point have ignored time, as we used a rate response function to calculate them.

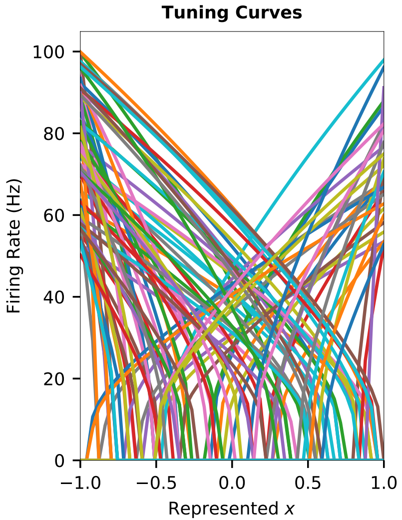
Temporal Decoding

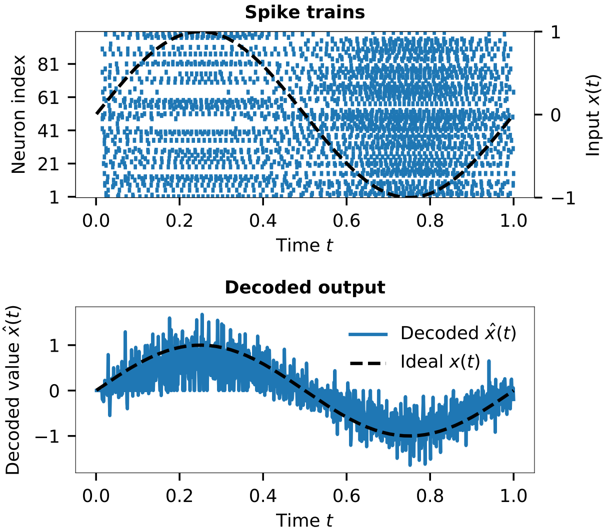
- Our decoders to this point have ignored time, as we used a rate response function to calculate them.
- What happens if we use those decoders with the spike trains generated by spiking LIF neurons?

Temporal Decoding of Two Neurons - Weighted Spikes



Temporal Decoding of One Hundred Neurons - Weighted Spikes





Temporal Decoding

- For population decoders, we needed to integrate their responses, $\mathbf{a}(\mathbf{x})$, over the represented variable, $\mathbf{x}$.

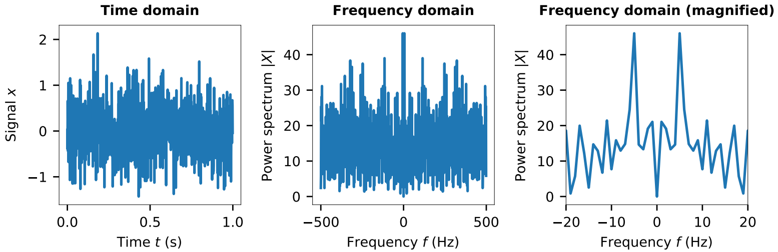
Temporal Decoding

- For population decoders, we needed to integrate their responses, $\mathbf{a}(\mathbf{x})$, over the represented variable, $\mathbf{x}$.
- For temporal decoders, we will likely want to integrate their responses, $\mathbf{a}(t)$, over the represented variable, $\mathbf{x}(t)$.

Temporal Decoding

- For population decoders, we needed to integrate their responses, $\mathbf{a}(\mathbf{x})$, over the represented variable, $\mathbf{x}$.
- For temporal decoders, we will likely want to integrate their responses, $\mathbf{a}(t)$, over the represented variable, $\mathbf{x}(t)$.
- What space do we want to sample to estimate the integrals?

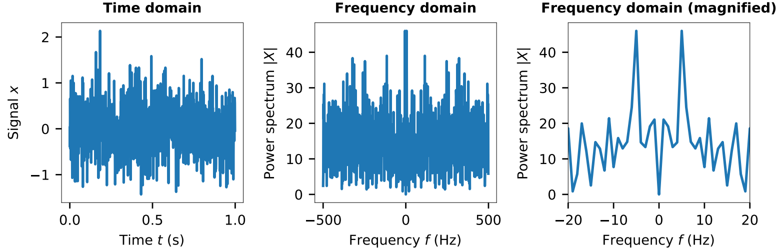
Random Signals



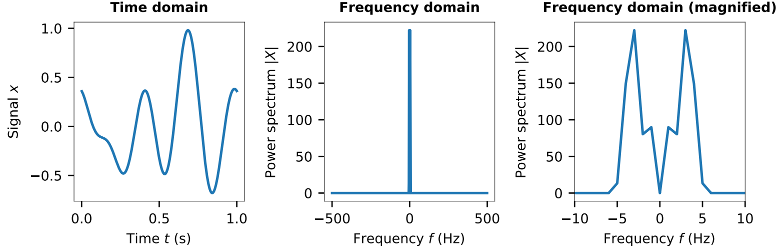
White Noise

(zero mean)

Random Signals

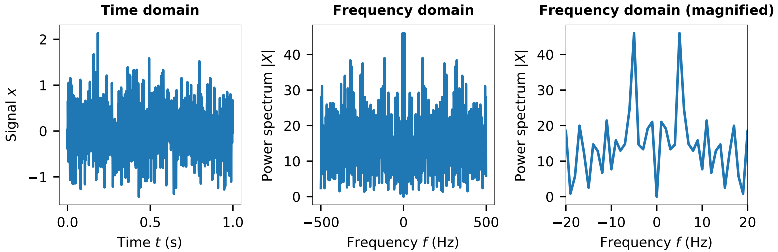


White Noise
(zero mean)

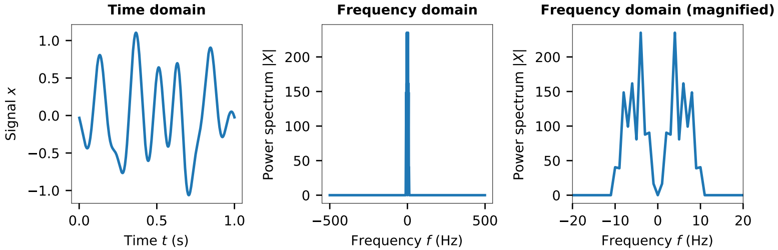


Bandlimited
White Noise
(zero mean,
5 Hz bandwidth)

Random Signals

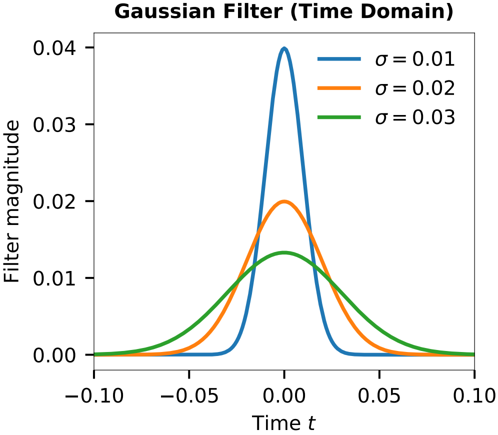


White Noise
(zero mean)



Bandlimited
White Noise
(zero mean,
10 Hz bandwidth)

Filtering by Convolution



Gaussian Filter

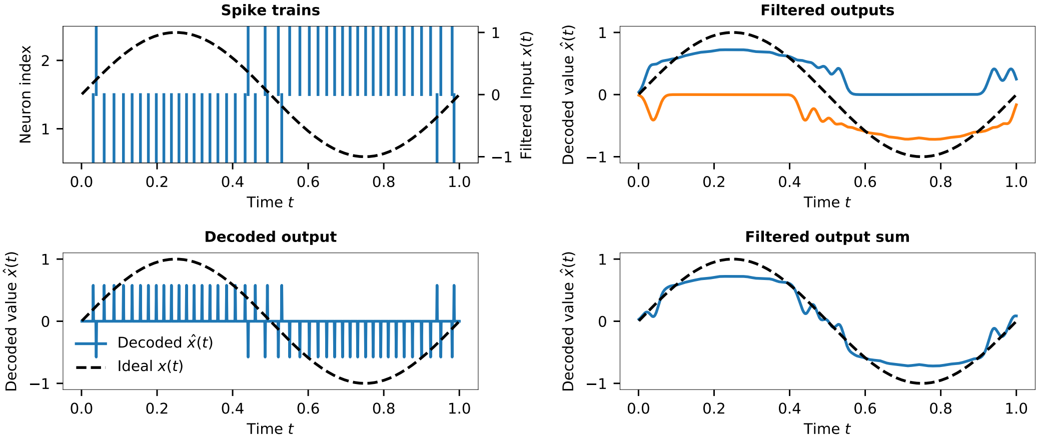
$$h(t) = c \exp\left(\frac{-t^2}{\sigma^2}\right)$$

where c chosen s.t. $\int_{-\infty}^{\infty} h(t)\,\mathrm{d}t = 1$

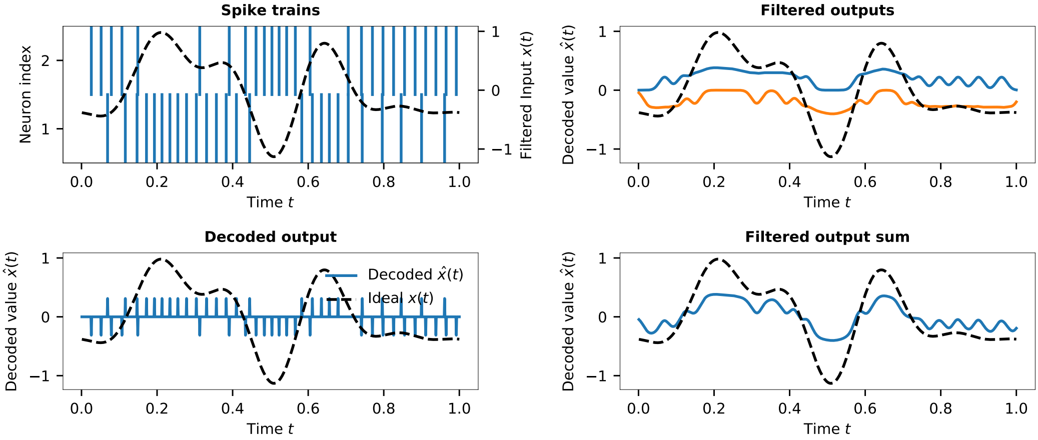
Convolution

$$(f * h)(t) = \int_{-\infty}^{\infty} f(t - t')h(t')\,\mathrm{d}t'$$

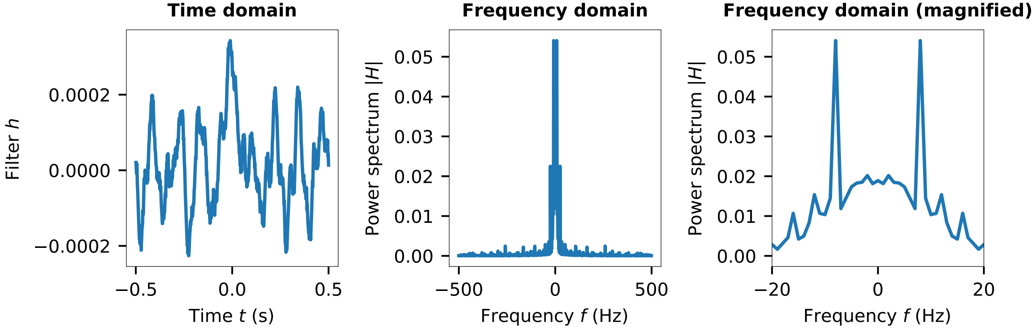
Filtering a Spike Train



Filtering a Spike Train for a Random Signal

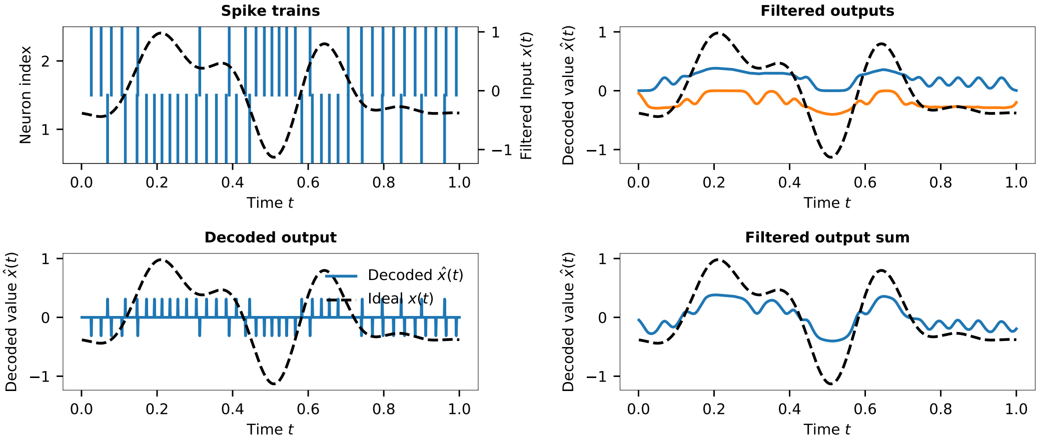


Optimal Filter

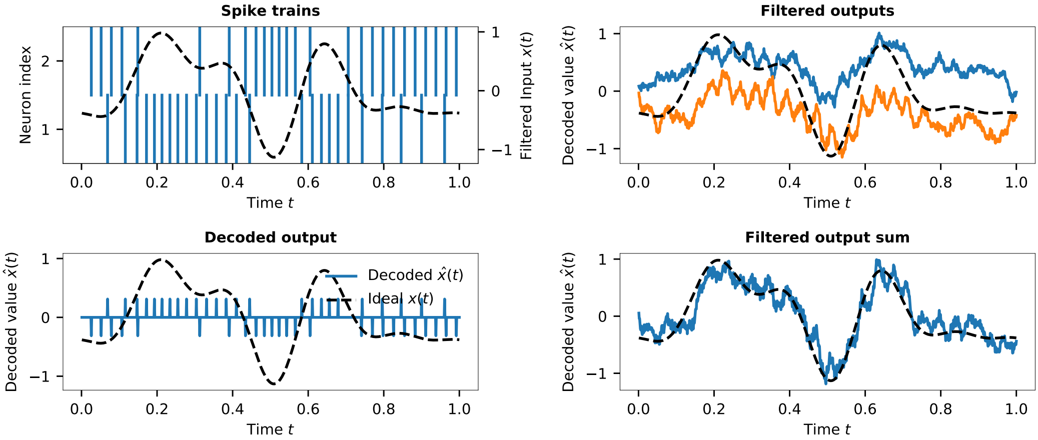


$$H(\omega) = \frac{X(\omega)\overline{R}(\omega)}{|R(\omega)|^2}$$

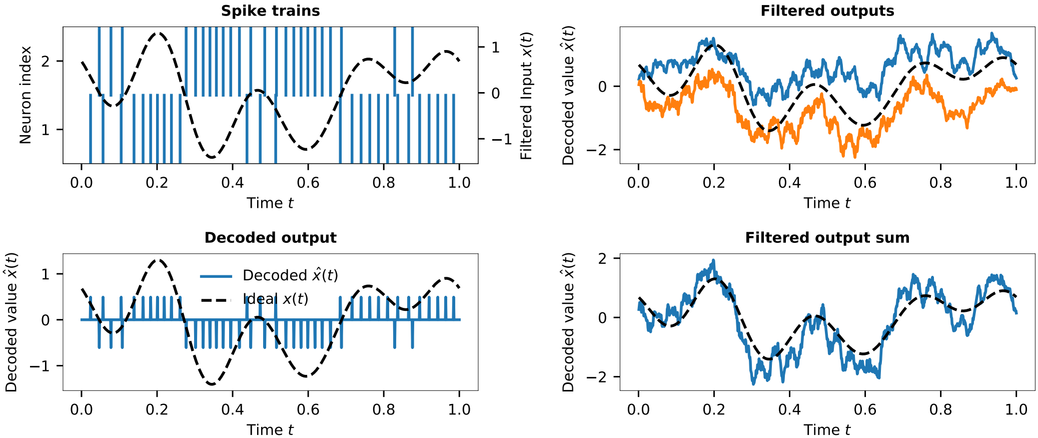
Filtering a Spike Train for a Random Signal (Optimal Filter)



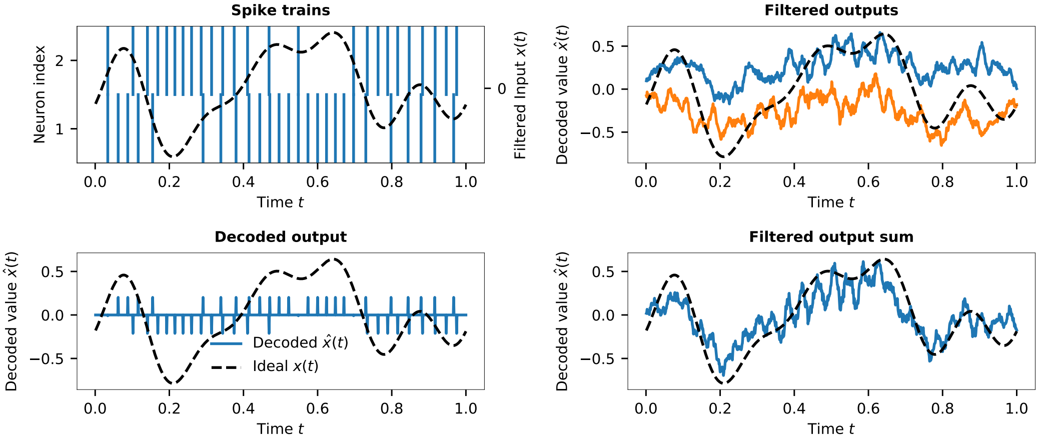
Filtering a Spike Train for a Random Signal (Optimal Filter)



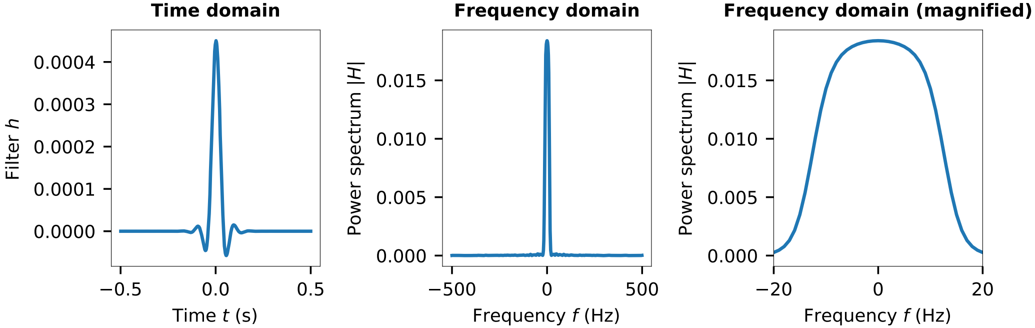
Filtering a Spike Train for a Random Signal (Optimal Filter)



Filtering a Spike Train for a Random Signal (Optimal Filter)

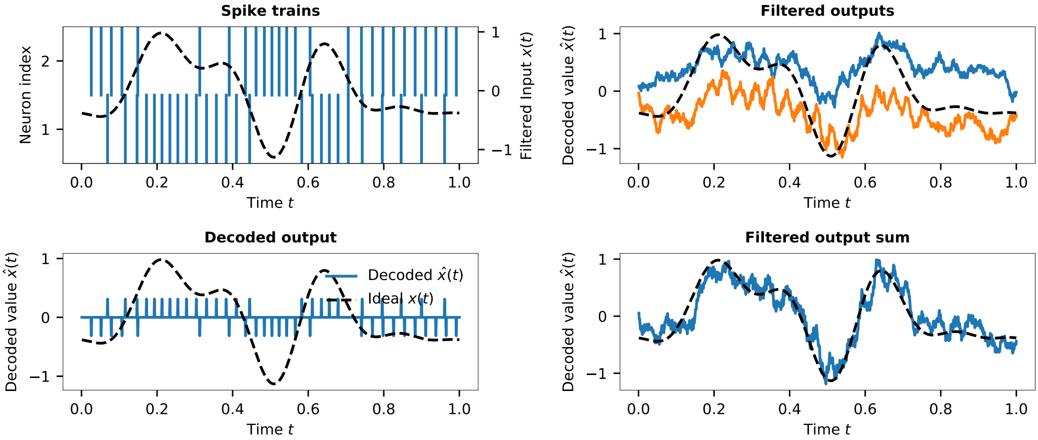


Optimal Filter (Improved)

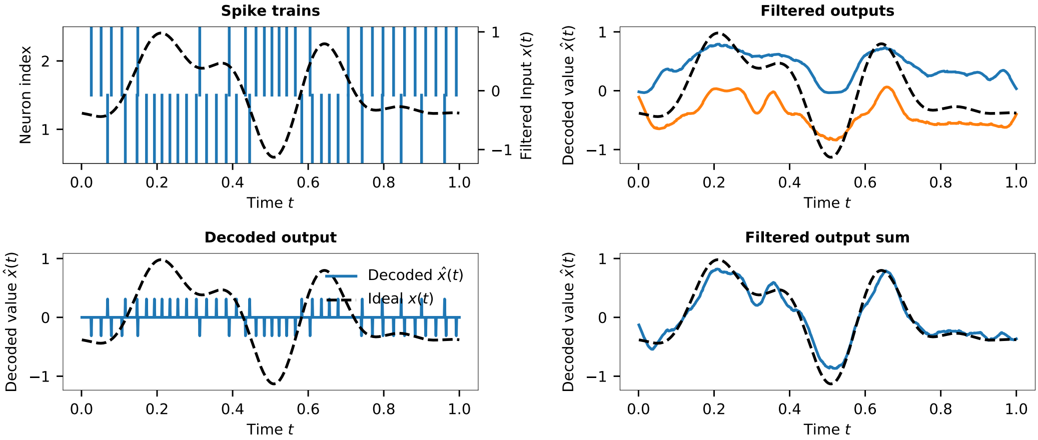


$$H(\omega) = \frac{X(\omega)\overline{R}(\omega) * W(\omega)}{|R(\omega)|^2 * W(\omega)}$$

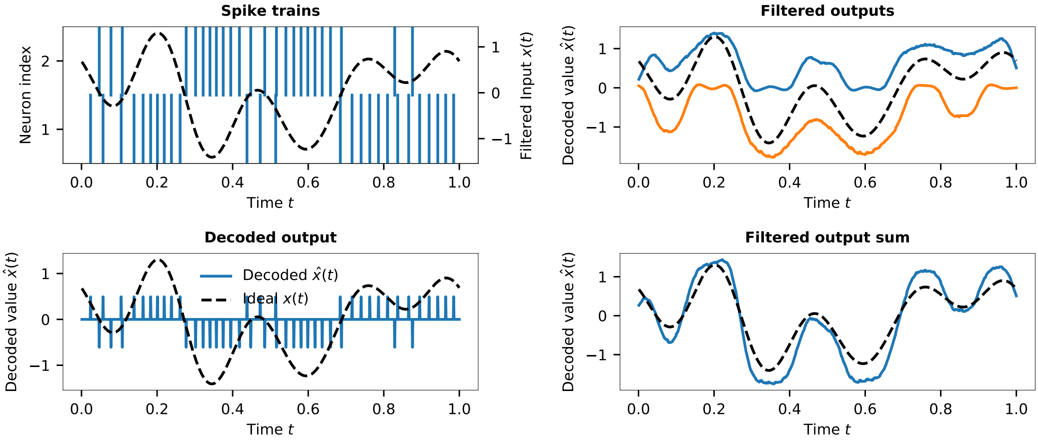
Filtering a Spike Train for a Random Signal (Improved Optimal Filter)



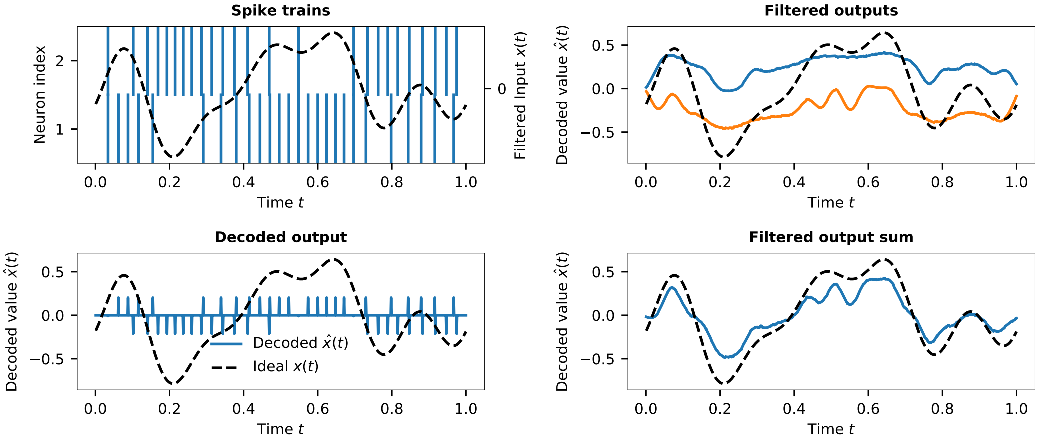
Filtering a Spike Train for a Random Signal (Improved Optimal Filter)



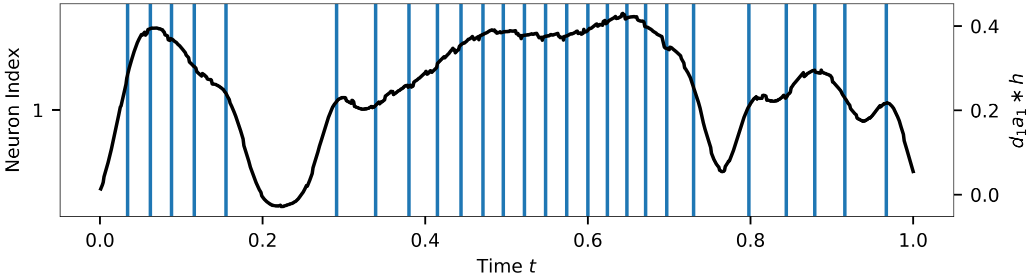
Filtering a Spike Train for a Random Signal (Improved Optimal Filter)



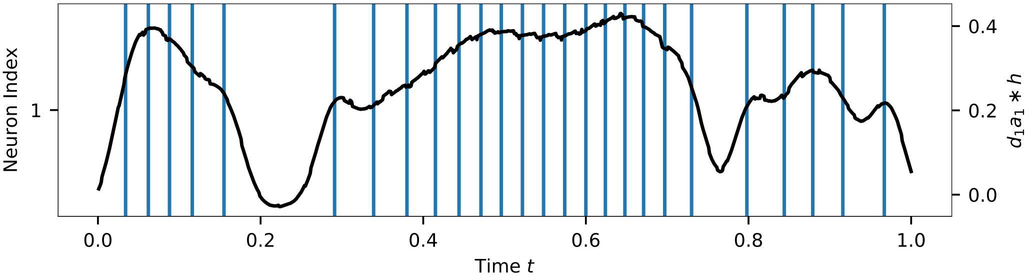
Filtering a Spike Train for a Random Signal (Improved Optimal Filter)



Pros and Cons of the Optimal Filter



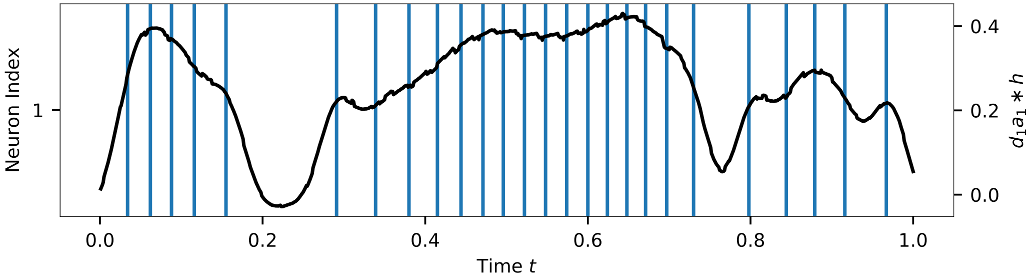
Pros and Cons of the Optimal Filter



Precise

Good for analysing data after the fact

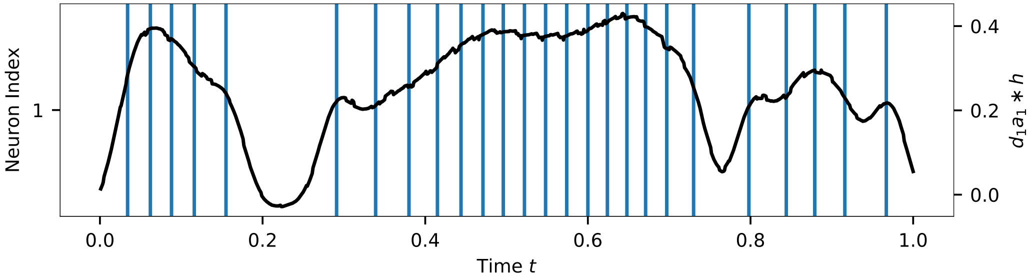
Pros and Cons of the Optimal Filter



Precise
Good for analysing data after the fact

Non-causal
Does not describe a biological process

Pros and Cons of the Optimal Filter

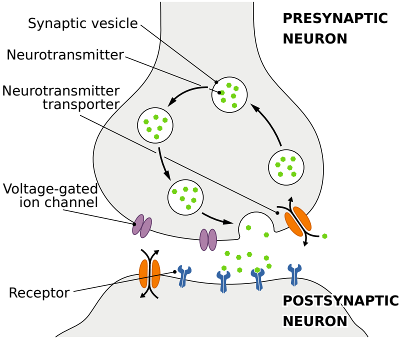


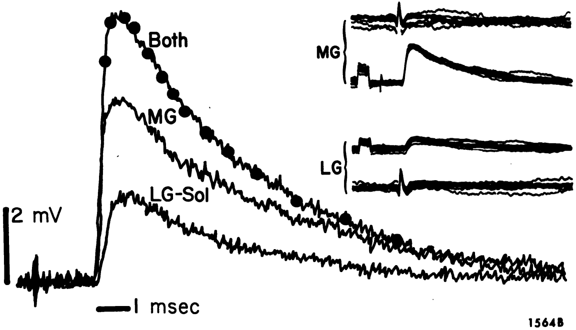
Precise
Good for analysing data after the fact

Non-causal
Does not describe a biological process

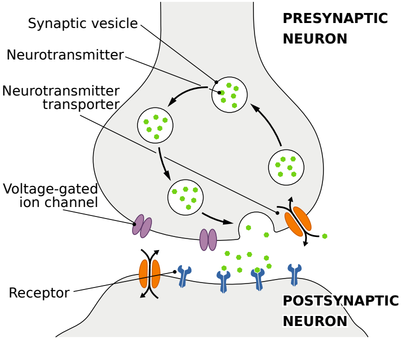
We need to find a mechanism that low-pass filters spikes over time!

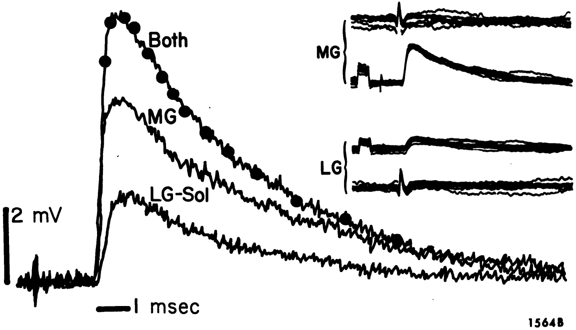
Synapses as Filters





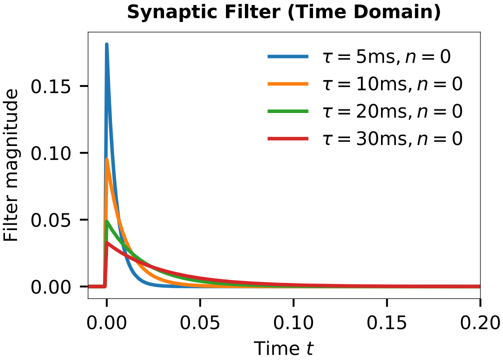
Synapses as Filters

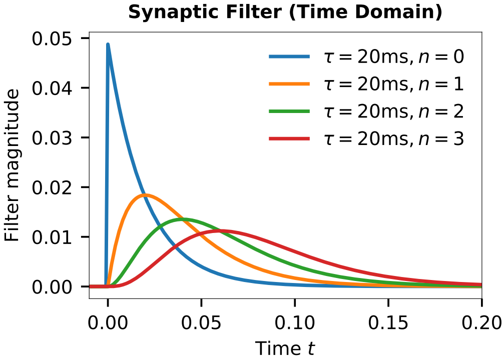




Post-synaptic currents (EPSCs, IPSCs) are low-pass filtered spike trains!

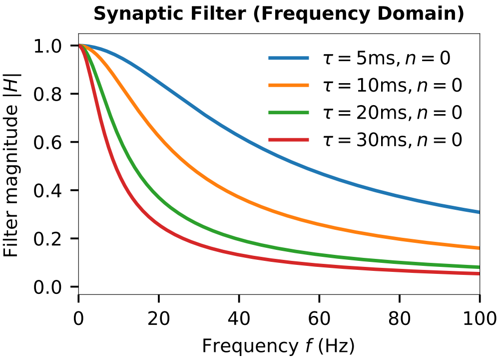
Exponential Low-Pass Filter (I)

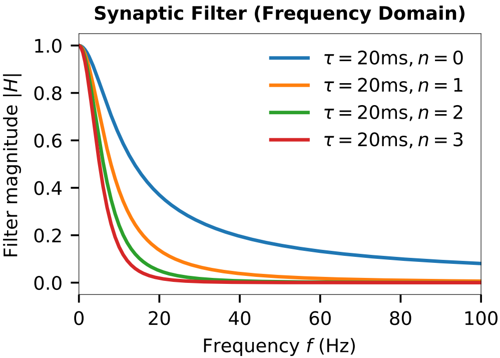




$$h(t) = \begin{cases} c^{-1} t^n \exp^{-t/\tau} & \text{if } t \geq 0\,, \\ 0 & \text{otherwise}\,, \end{cases} \qquad \text{where } c = \int_0^\infty t^n \exp^{-t/\tau} \, \mathrm{d}t\,.$$

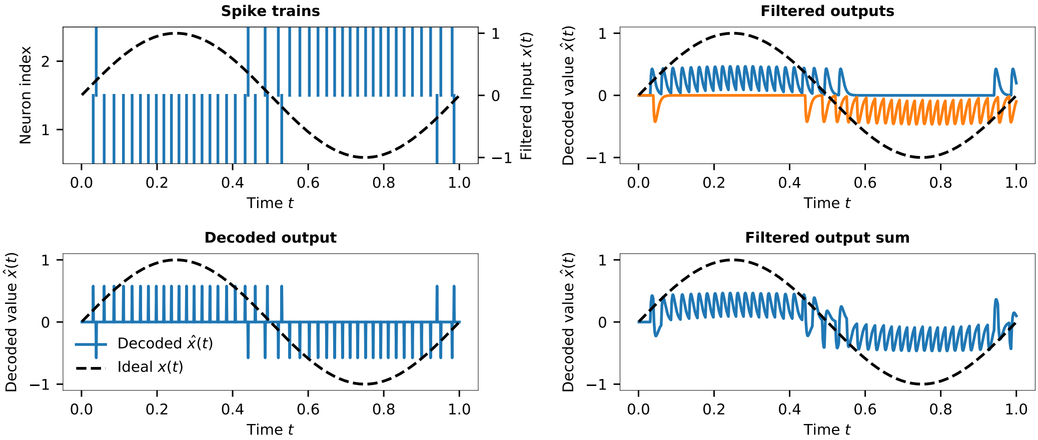
Exponential Low-Pass Filter (II)





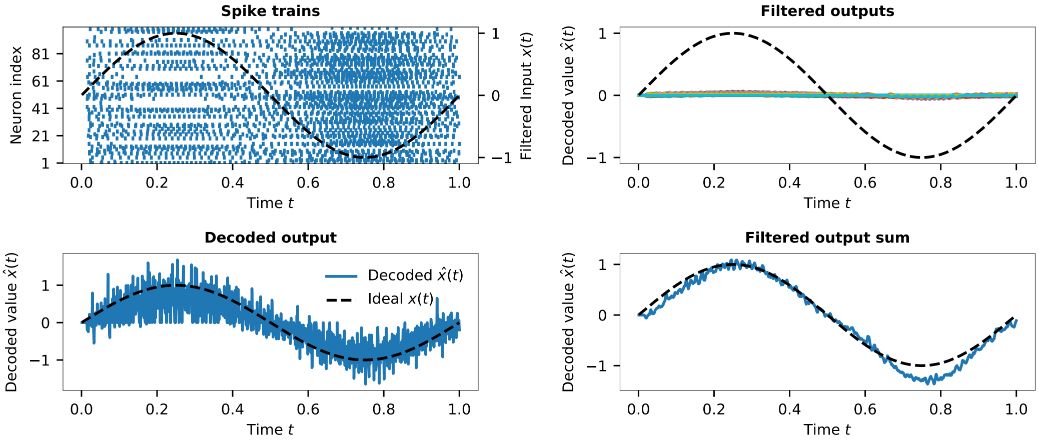
$$h(t) = \begin{cases} c^{-1} t^n \exp^{-t/\tau} & \text{if } t \geq 0\,, \\ 0 & \text{otherwise}\,, \end{cases} \qquad \text{where } c = \int_0^\infty t^n \exp^{-t/\tau} \, \mathrm{d}t\,.$$

Example: Synaptic Filter for Two Neurons



$$\tau = 5\,\mathrm{ms},\ n = 1$$

Example: Synaptic Filter for One Hundred Neurons



$$\tau = 5\,\text{ms}, n = 1$$

Image sources

Title slide
"Captive balloon with clock face and bell, floating above the Eiffel Tower, Paris, France."
Author: Camille Grávis, between 1889 and 1900.
From Wikimedia.