Eyes are smarter than scientists believed

Ulisse Ferrari

ulisse.ferrari@gmail.com

INSTITUT DE LA VISION PARIS

Find the TD material at **http://oliviermarre.free.fr/**

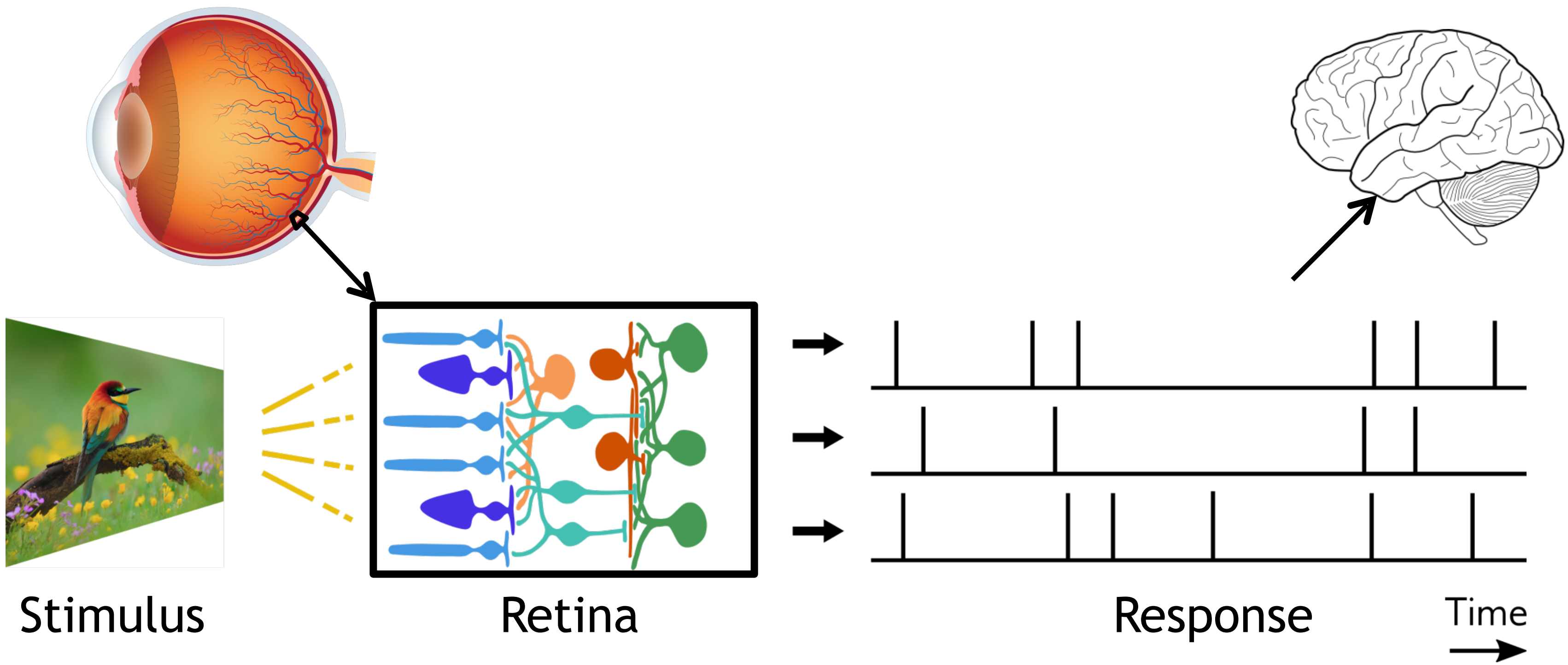
Eyes are smarter than scientists believed

1) The structure of the retina

2) Stimulus processing in the retina

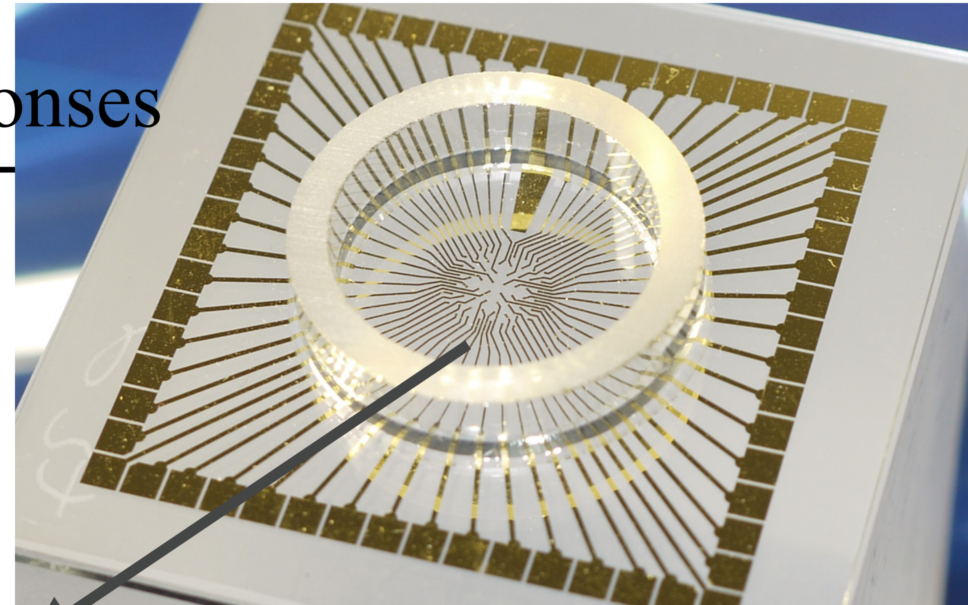
3) Predicting retinal light-response

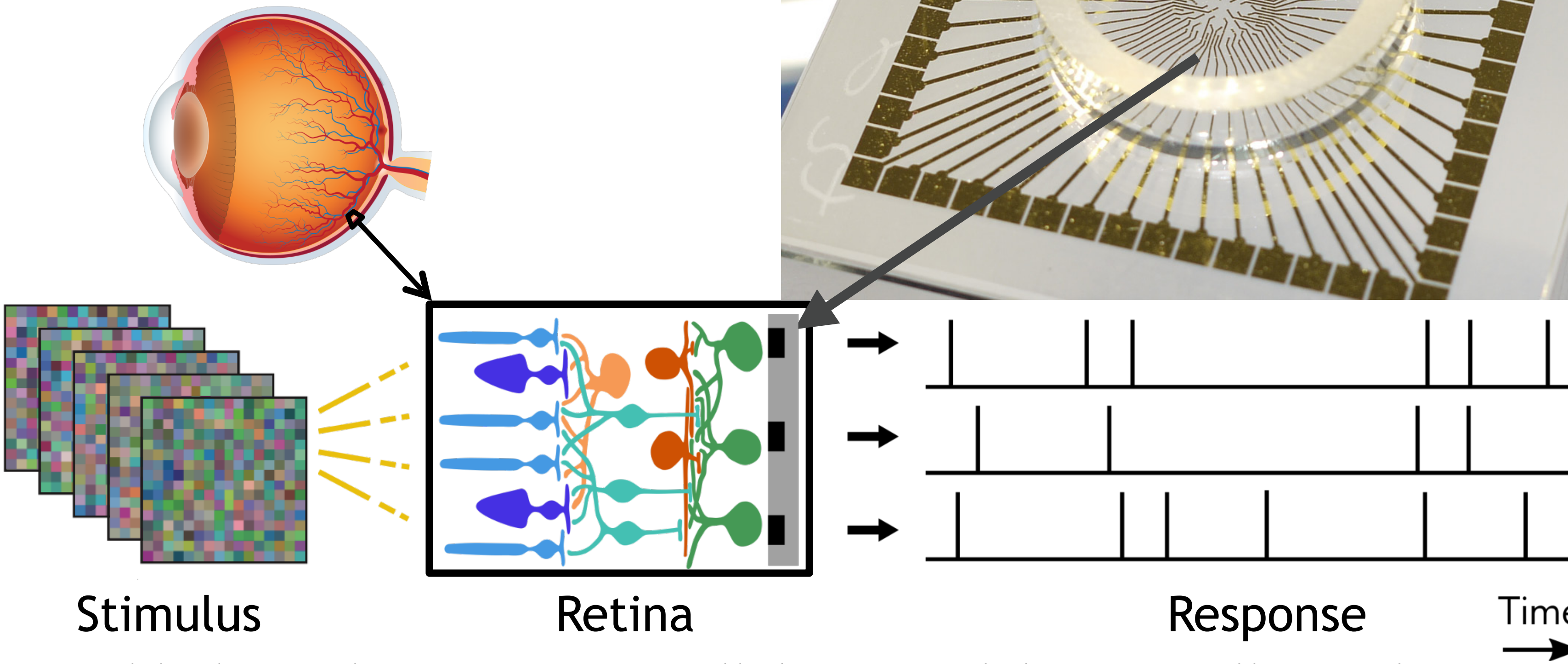
Predicting retinal light responses



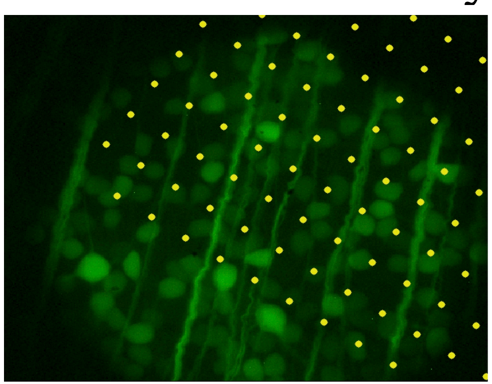
Goal: predict spiking times

Predicting retinal light responses

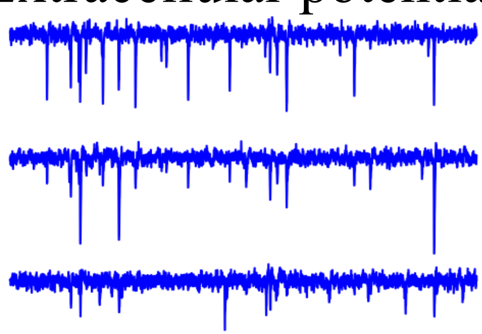




Multi-electrode array

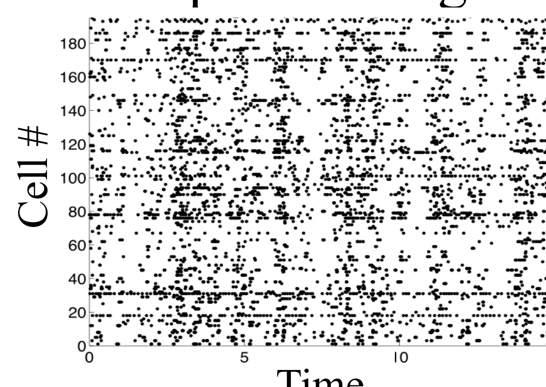


Extracellular potential



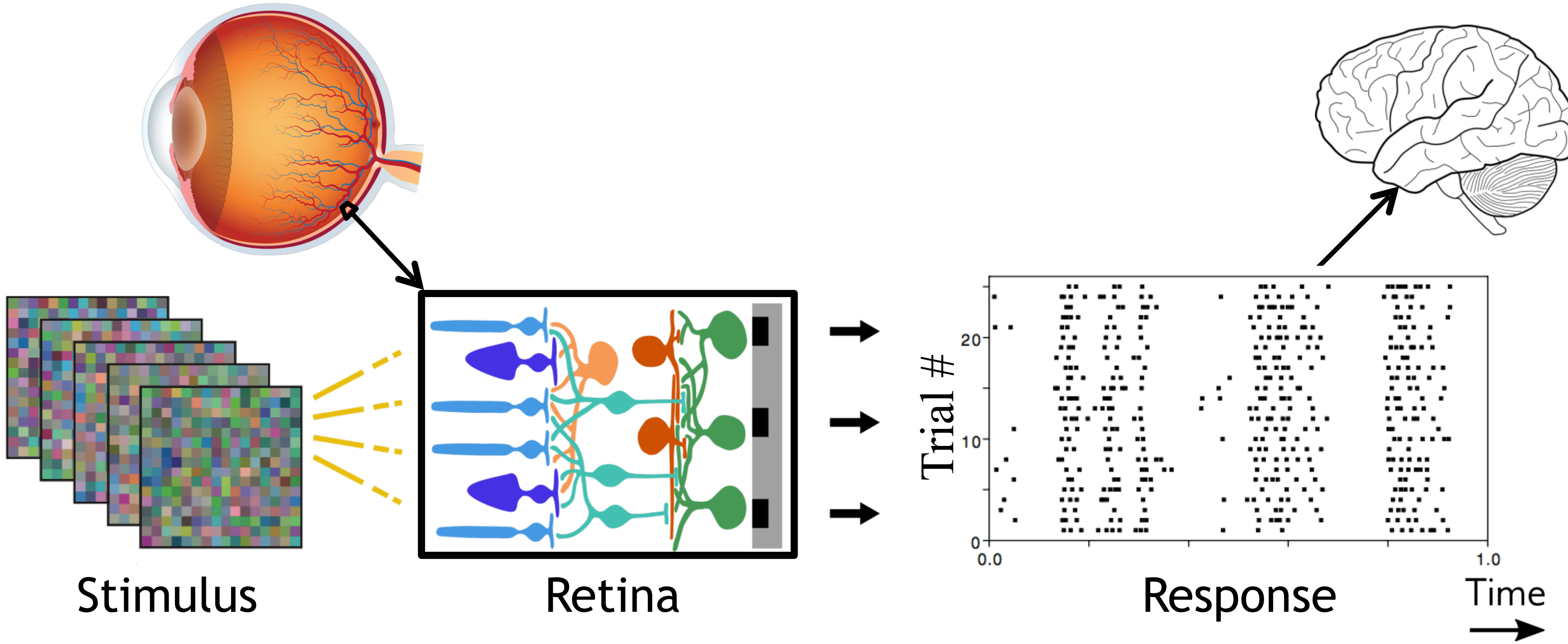
Marre '12; Yger '16

Spike sorting



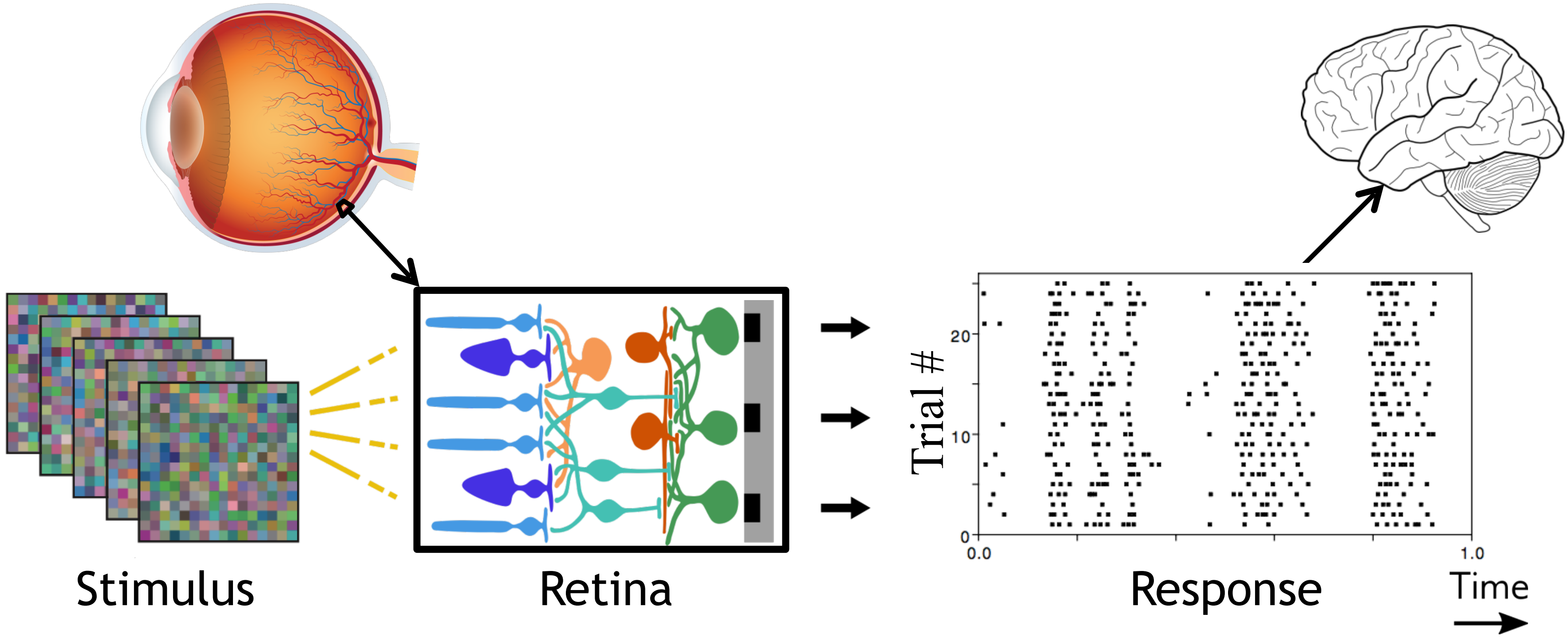
Predicting retinal light responses

Predicting retinal light responses



Response is reliable but noisy!

Predicting retinal light responses



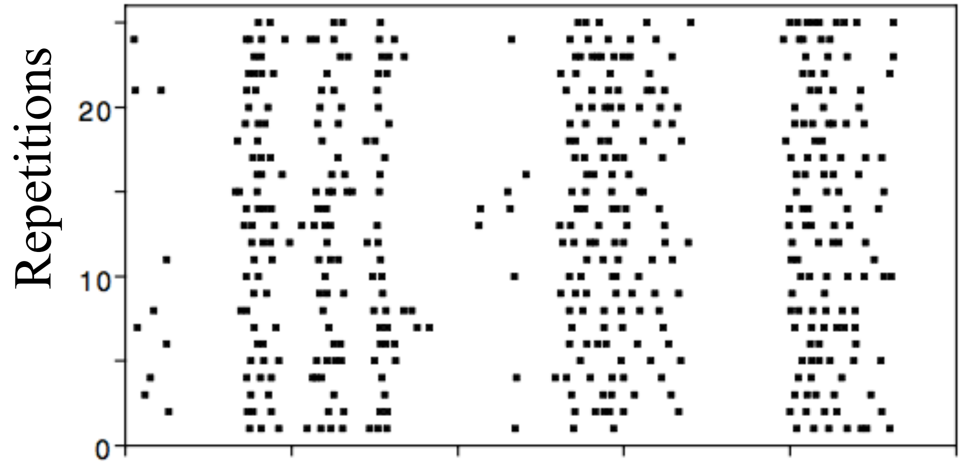
Response is reliable but noisy!

Average over trials

Predicting retinal light responses

Peri-Stimulus Time Histogram (PSTH)

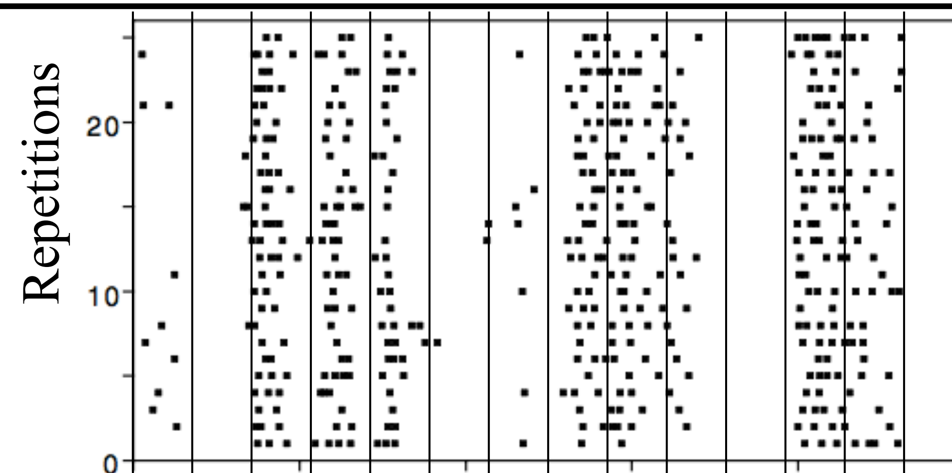
1) repeat video (trials)
2) align responses in time



Predicting retinal light responses

Peri-Stimulus Time Histogram (PSTH)

1) repeat video (trials)
2) align responses in time
3) bin responses at dt=25ms
4) spike counts n_t in each dt



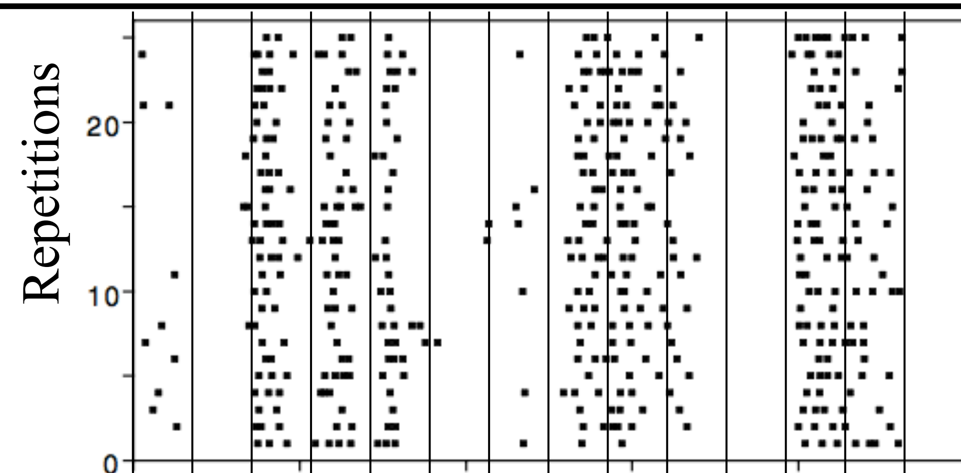
1	0	8	1	1	0	1	8	7	4	0	8	5	0
0	0	7	6	4	1	0	9	7	3	0	6	4	0
0	0	6	2	3	0	0	7	6	3	0	8	6	0
2	1	3	4	2	0	2	9	1	3	1	5	3	1

Predicting retinal light responses

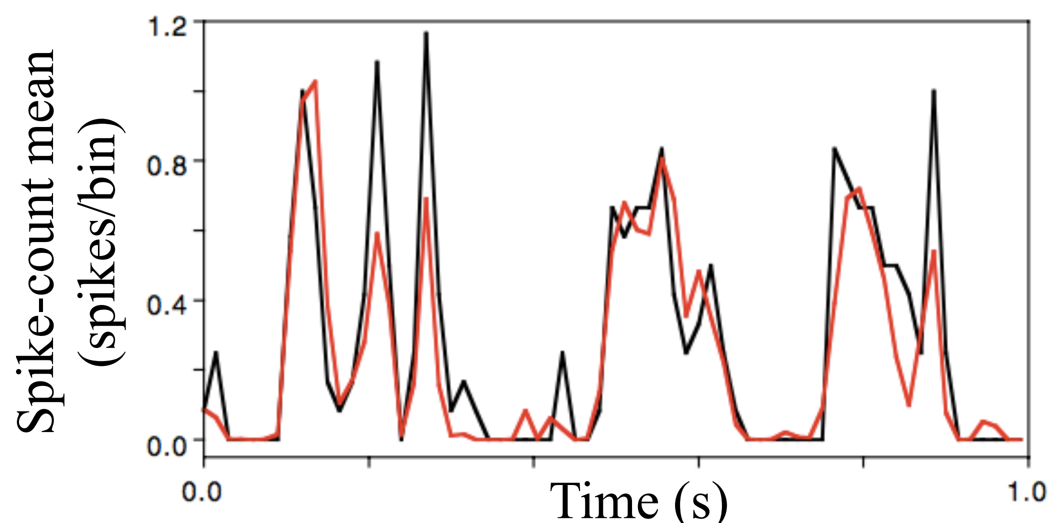
Peri-Stimulus Time Histogram (PSTH)

1) repeat video (trials)
2) align responses in time
3) bin responses at dt=25ms
4) spike counts n_t in each dt
5) average over trials
6) estimate spike rate

Can we predict $\langle n_t \rangle$?



1	0	8	1	1	0	1	8	7	4	0	8	5	0
0	0	7	6	4	1	0	9	7	3	0	6	4	0
0	0	6	2	3	0	0	7	6	3	0	8	6	0
2	1	3	4	2	0	2	9	1	3	1	5	3	1



Linear model

$$\tilde{S}_{xyt} = \left(S_{xyt} - \langle S \rangle \right) \Big/ \operatorname{std}(S)$$

$$f(t) = w * \tilde{S}(t) + b = \sum_{x,y,\tau} w_{x,y}(\tau)\ \tilde{S}_{x,y}(t - \tau) + b$$

Linear model

$$\tilde{S}_{xyt} = \left(S_{xyt} - \langle S \rangle \right) \Big/ \mathrm{std}(S)$$

$$f(t) = w * \tilde{S}(t) + b = \sum_{x,y,\tau} w_{x,y}(\tau)\ \tilde{S}_{x,y}(t-\tau) + b$$

Optimise w by minimising: $\quad \mathrm{MSE} = \dfrac{1}{2} \displaystyle\sum_t \big(f(t) - n(t) \big)^2$

Linear model

$$\tilde{S}_{xyt} = \Big(S_{xyt} - \langle S \rangle \Big) \Big/ \mathrm{std}(S)$$

$$f(t) = w * \tilde{S}(t) + b = \sum_{x,y,\tau} w_{x,y}(\tau)\ \tilde{S}_{x,y}(t-\tau) + b$$

Optimise w by minimising: $\quad \mathrm{MSE} = \dfrac{1}{2} \displaystyle\sum_t \big(f(t) - n(t) \big)^2$

After some algebra: $\quad w_{x,y}(\tau) = \displaystyle\sum_{\tau'} \mathrm{STA}_{x,y}(\tau')\ C^{-1}_{x,y}(\tau', \tau)$

Spike-Triggered Average

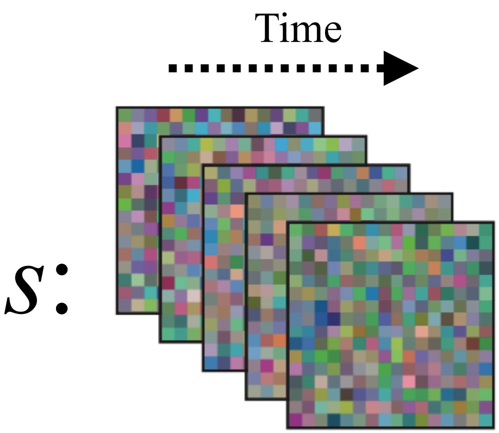
$$\mathrm{STA}_{x,y}(\tau) \equiv \frac{1}{T} \sum_t^T n(t)\ \tilde{S}_{x,y}(t-\tau)$$

Stimulus autocorrelation

$$C_{x,y}(\tau', \tau) \equiv \frac{1}{T} \sum_t^T \tilde{S}_{x,y}(t-\tau')\ \tilde{S}_{x,y}(t-\tau)$$

Linear model

$$C_{x,y}(\tau', \tau) = \delta(\tau - \tau')$$



After some algebra: $$w_{x,y}(\tau) = \sum_{\tau'} \mathrm{STA}_{x,y}(\tau')\ C^{-1}_{x,y}(\tau', \tau)$$

Spike-Triggered Average

$$\mathrm{STA}_{x,y}(\tau) \equiv \frac{1}{T} \sum_{t}^{T} n(t)\ \tilde{S}_{x,y}(t - \tau)$$

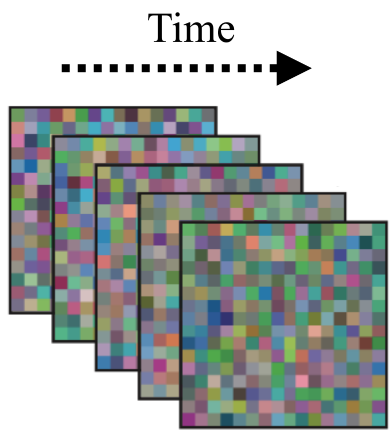
Stimulus autocorrelation

$$C_{x,y}(\tau', \tau) \equiv \frac{1}{T} \sum_{t}^{T} \tilde{S}_{x,y}(t - \tau')\ \tilde{S}_{x,y}(t - \tau)$$

Linear model

$$C_{x,y}(\tau', \tau) = \delta(\tau - \tau') \quad \Rightarrow \quad w_{x,y}(\tau) = \mathrm{STA}_{x,y}(\tau) = \equiv \frac{1}{T}\sum_{t}^{T} n(t)\ \tilde{S}_{x,y}(t-\tau)$$

S:



After some algebra: $$w_{x,y}(\tau) = \sum_{\tau'} \mathrm{STA}_{x,y}(\tau')\ C^{-1}_{x,y}(\tau', \tau)$$

Spike-Triggered Average

$$\mathrm{STA}_{x,y}(\tau) \equiv \frac{1}{T}\sum_{t}^{T} n(t)\ \tilde{S}_{x,y}(t-\tau)$$

Stimulus autocorrelation

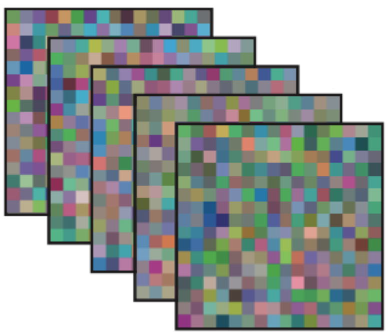
$$C_{x,y}(\tau', \tau) \equiv \frac{1}{T}\sum_{t}^{T} \tilde{S}_{x,y}(t-\tau')\ \tilde{S}_{x,y}(t-\tau)$$

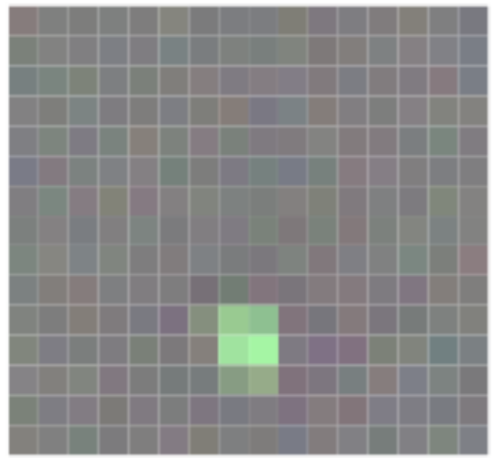
Linear model

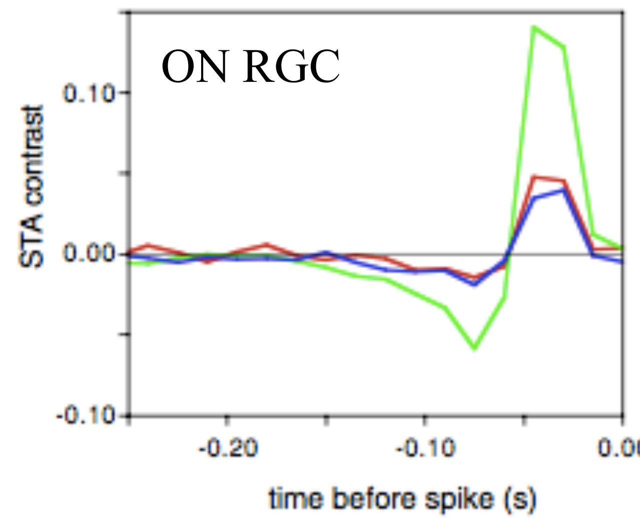
$$C_{x,y}(\tau', \tau) = \delta(\tau - \tau') \quad \Rightarrow \quad w_{x,y}(\tau) = \mathrm{STA}_{x,y}(\tau) = \equiv \frac{1}{T}\sum_{t}^{T} n(t)\ \tilde{S}_{x,y}(t - \tau)$$

Time

s:



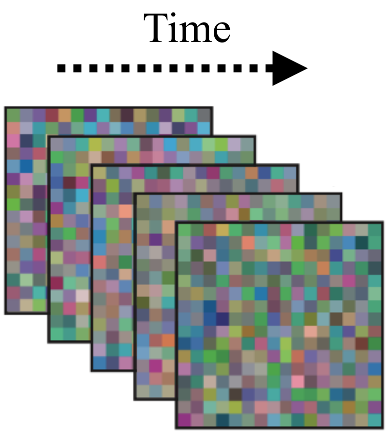


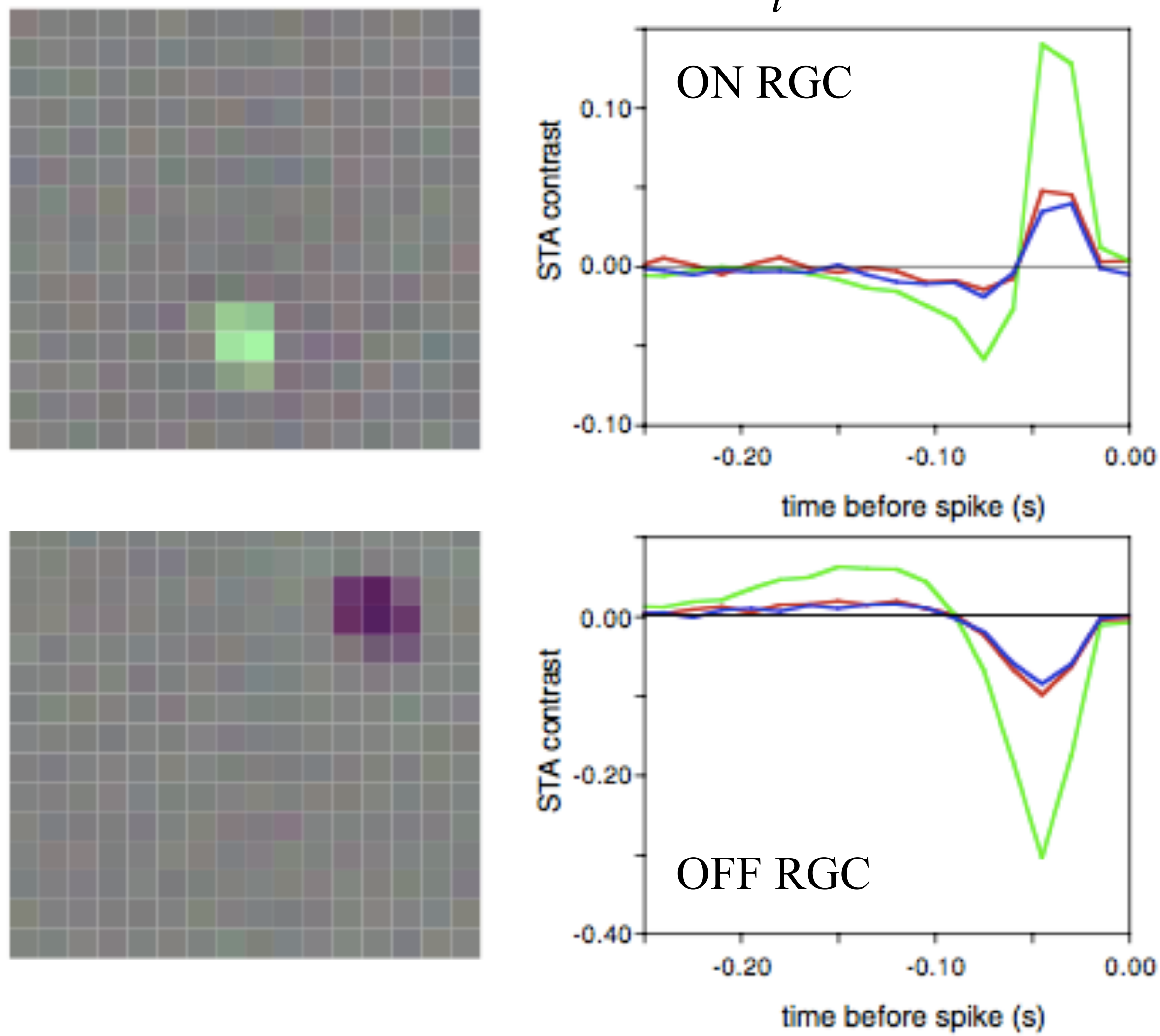


Linear model

$$C_{x,y}(\tau', \tau) = \delta(\tau - \tau') \quad \Rightarrow \quad w_{x,y}(\tau) = \mathrm{STA}_{x,y}(\tau) = \equiv \frac{1}{T}\sum_{t}^{T} n(t)\ \tilde{S}_{x,y}(t-\tau)$$

s:

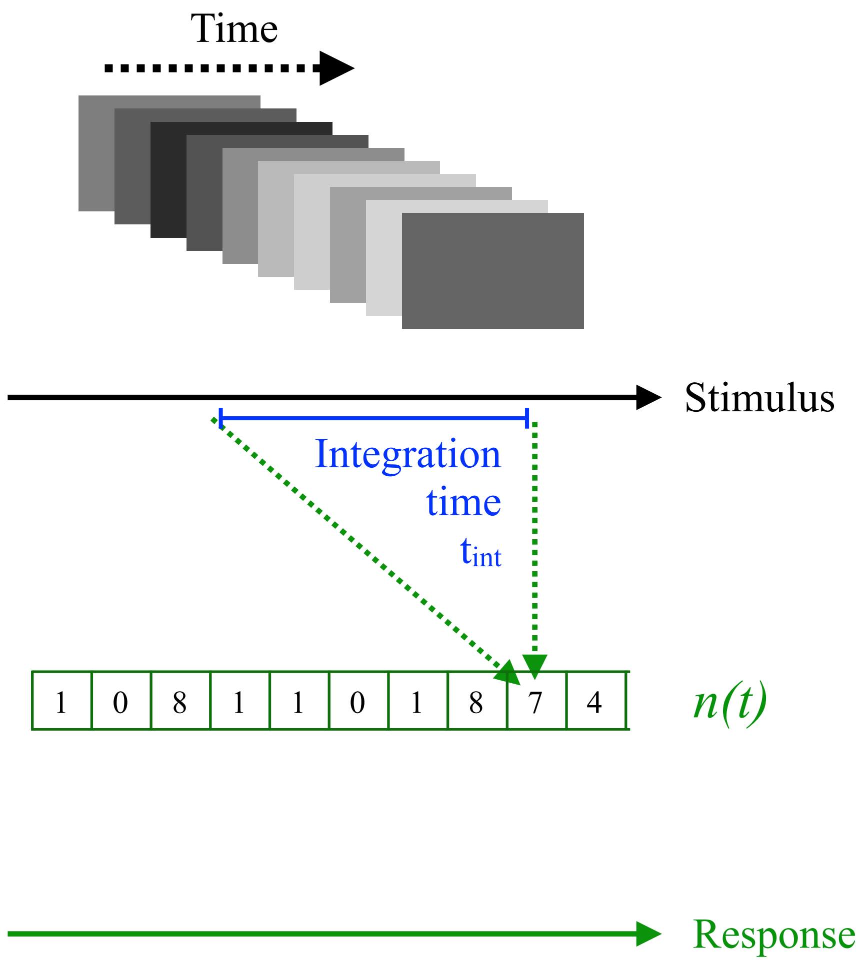




Linear model: full field case

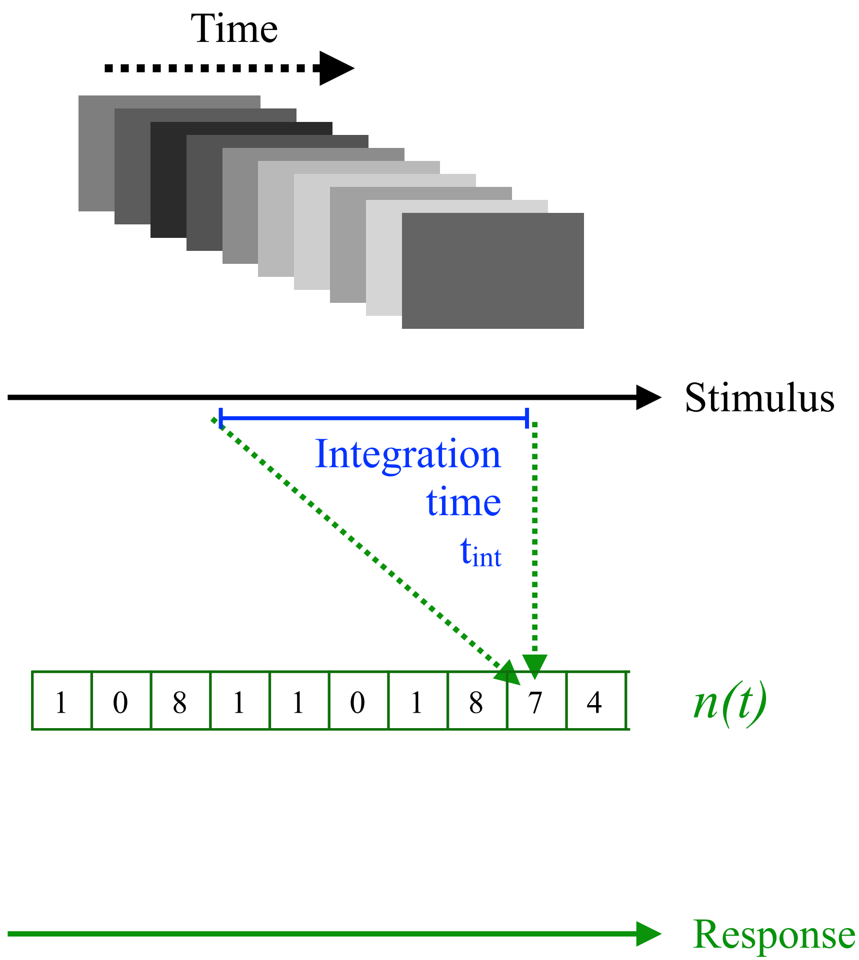
$$f(t) = w * \tilde{S}(t) + b = \sum_{\tau}^{t_{int}} w(\tau)\ \tilde{S}(t-\tau) + b$$

$$\mathrm{MSE} = \frac{1}{2}\sum_{t}\big(f(t) - n(t)\big)^2$$



Linear model: full field case

$$f(t) = w * \tilde{S}(t) + b = \sum_{\tau}^{t_{int}} w(\tau)\ \tilde{S}(t-\tau) + b$$



$$\mathrm{MSE} = \frac{1}{2} \sum_{t} \left(f(t) - n(t)\right)^2$$

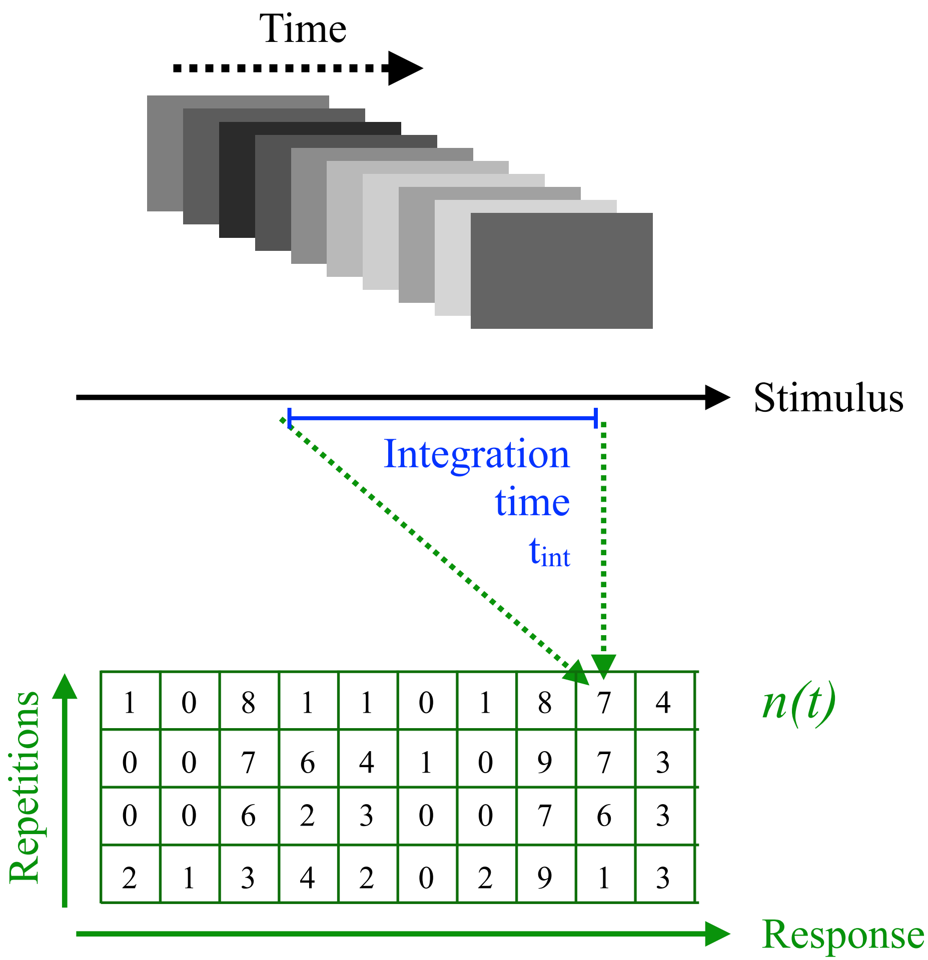
$$w(\tau) = \sum_{\tau'}^{t_{int}} \mathrm{STA}(\tau')\ C^{-1}(\tau', \tau)$$

$$C(\tau', \tau) \equiv \frac{1}{T} \sum_{t} \tilde{S}(t-\tau')\ \tilde{S}(t-\tau)$$

$$\mathrm{STA}(\tau) \equiv \frac{1}{T} \sum_{t} n(t)\ \tilde{S}(t-\tau)$$

Linear model: full field case

$$f(t) = w * \tilde{S}(t) + b = \sum_{\tau}^{t_{int}} w(\tau)\ \tilde{S}(t-\tau) + b$$



1	0	8	1	1	0	1	8	7	4
0	0	7	6	4	1	0	9	7	3
0	0	6	2	3	0	0	7	6	3
2	1	3	4	2	0	2	9	1	3

$$\mathrm{MSE} = \frac{1}{2} \sum_{t} \big(f(t) - n(t)\big)^2$$

$$w(\tau) = \sum_{\tau'}^{t_{int}} \mathrm{STA}(\tau')\ C^{-1}(\tau', \tau)$$

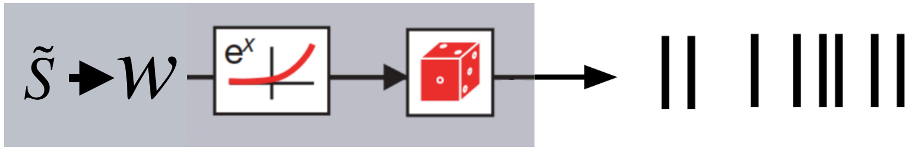
$$C(\tau', \tau) \equiv \frac{1}{T} \sum_{t} \tilde{S}(t-\tau')\ \tilde{S}(t-\tau)$$

$$\mathrm{STA}(\tau) \equiv \frac{1}{T} \sum_{t} n(t)\ \tilde{S}(t-\tau)$$

$$= \frac{1}{T} \sum_{t} \left(\frac{1}{R} \sum_{r} n^{r}(t)\right) \tilde{S}(t-\tau)$$

Linear-nonlinear Poisson model (LNP)

$$P_t(n) = \mathrm{Poisson}\left(n \mid f(t) \right) = \frac{f(t)^n}{n!} e^{-f(t)}$$

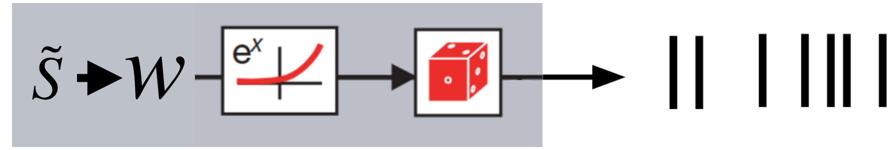


$$f(t) = \mathrm{NL}\left(w * \tilde{S} + b \right) \rightarrow \exp\left(w * \tilde{S} + b \right)$$

Where NL(x) can be exp(x)
or another nonlinear function

Linear-nonlinear Poisson model (LNP)

$$P_t(n) = \text{Poisson}\big(\, n \mid f(t)\,\big) = \frac{f(t)^n}{n!} e^{-f(t)}$$



$$f(t) = \text{NL}\left(w * \tilde{S} + b\right) \rightarrow \exp\left(w * \tilde{S} + b\right)$$

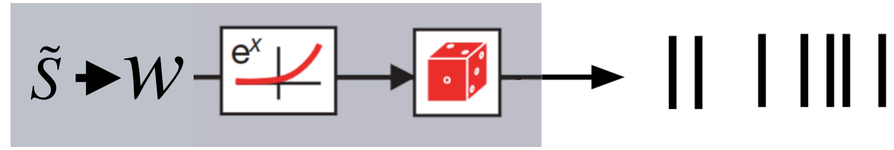
Where NL(x) can be exp(x) or another nonlinear function

Can be solved by log-likelihood maximisation:

$$\ell(w,b) = \sum_t \big(\, n(t)\log f(t) - f(t)\,\big)$$

Linear-nonlinear Poisson model (LNP)

$$P_t(n) = \mathrm{Poisson}\big(\, n \,|\, f(t) \,\big) = \frac{f(t)^n}{n!} e^{-f(t)}$$



$$f(t) = \mathrm{NL}\left(w * \tilde{S} + b\right) \rightarrow \exp\left(w * \tilde{S} + b\right)$$

Where NL(x) can be exp(x) or another nonlinear function

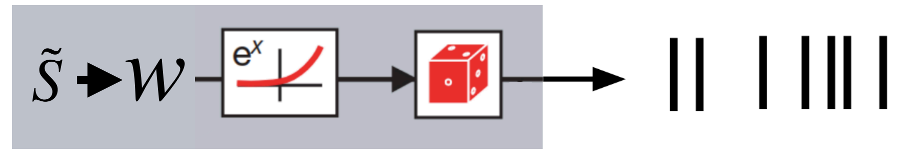
Can be solved by log-likelihood maximisation:

$$\ell(w, b) = \sum_t \big(\, n(t) \log f(t) - f(t) \,\big)$$

$$\frac{\partial \ell(w, b)}{\partial w(\tau)} = \sum_t \left[\left(\frac{n(t)}{f(t)} - 1 \right) \frac{\partial f(t)}{\partial w(\tau)} \right]$$

Linear-nonlinear Poisson model (LNP)

$$P_t(n) = \text{Poisson}(\, n \mid f(t) \,) = \frac{f(t)^n}{n!} e^{-f(t)}$$



$$f(t) = \text{NL}\left(w * \tilde{S} + b\right) \rightarrow \exp\left(w * \tilde{S} + b\right)$$

Where NL(x) can be exp(x) or another nonlinear function

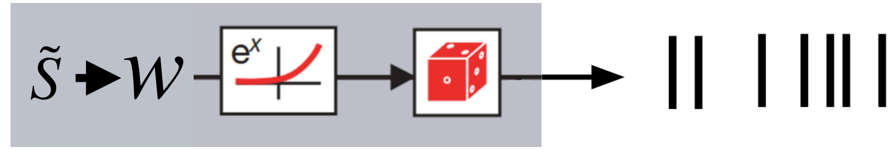
Can be solved by log-likelihood maximisation:

$$\ell(w, b) = \sum_t \left(\, n(t) \log f(t) - f(t) \,\right)$$

$$\frac{\partial \ell(w,b)}{\partial w(\tau)} = \sum_t \left[\left(\frac{n(t)}{f(t)} - 1 \right) \frac{\partial f(t)}{\partial w(\tau)} \right] = \sum_t \left[\left(\, n(t) - f(t) \,\right) \tilde{S}(t-\tau) \right]$$

Linear-nonlinear Poisson model (LNP)

$$P_t(n) = \mathrm{Poisson}\big(\, n \mid f(t) \,\big) = \frac{f(t)^n}{n!} e^{-f(t)}$$



$$f(t) = \mathrm{NL}\left(w * \tilde{S} + b\right) \rightarrow \exp\left(w * \tilde{S} + b\right)$$

Where NL(x) can be exp(x) or another nonlinear function

Can be solved by log-likelihood maximisation:

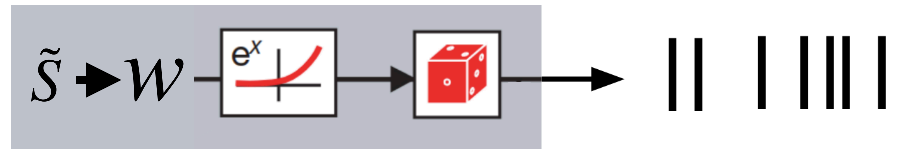
$$\ell(w,b) = \sum_t \big(\, n(t) \log f(t) - f(t) \,\big)$$

$$\frac{\partial \ell(w,b)}{\partial w(\tau)} = \sum_t \left[\left(\frac{n(t)}{f(t)} - 1 \right) \frac{\partial f(t)}{\partial w(\tau)} \right] = \sum_t \left[\big(\, n(t) - f(t) \,\big)\, \tilde{S}(t-\tau) \right]$$

$$\frac{\partial \ell(w,b)}{\partial b} = \sum_t \left[\big(\, n(t) - f(t) \,\big) \right]$$

Linear-nonlinear Poisson model (LNP)

$$P_t(n) = \text{Poisson}\big(\, n \mid f(t) \,\big) = \frac{f(t)^n}{n!} e^{-f(t)}$$



$$f(t) = \text{NL}\left(w * \tilde{S} + b\right) \rightarrow \exp\left(w * \tilde{S} + b\right)$$

Where NL(x) can be exp(x) or another nonlinear function

Can be solved by log-likelihood maximisation:

$$\ell(w,b) = \sum_t \big(\, n(t) \log f(t) - f(t) \,\big)$$

$$\frac{\partial \ell(w,b)}{\partial w(\tau)} = \sum_t \left[\left(\frac{n(t)}{f(t)} - 1 \right) \frac{\partial f(t)}{\partial w(\tau)} \right] = \sum_t \left[\big(\, n(t) - f(t) \,\big)\, \tilde{S}(t-\tau) \right]$$

$$\frac{\partial \ell(w,b)}{\partial b} = \sum_t \left[\big(\, n(t) - f(t) \,\big) \right]$$

Can be solved by steepest gradient

LNP model + L_2-Regularisation

To have a smooth *w*, we should minimise $\left(w(\tau) - w(\tau + 1) \right)^2$

LNP model + L_2-Regularisation

To have a smooth *w*, we should minimise $\left(w(\tau) - w(\tau+1) \right)^2$

$$\ell_\lambda(w,b) = \sum_t \left(n(t) \log f(t) - f(t) \right) - \frac{\lambda}{2} \sum_{\tau,\tau'} w(\tau)\, L(\tau,\tau')\, w(\tau')$$

LNP model + L_2-Regularisation

To have a smooth *w*, we should minimise $(\, w(\tau) - w(\tau+1)\,)^2$

$$\ell_\lambda(w,b) = \sum_t \Big(\, n(t)\log f(t) - f(t) \,\Big) - \frac{\lambda}{2}\sum_{\tau,\tau'} w(\tau)\, L(\tau,\tau')\, w(\tau')$$

Laplacian: $$L(\tau,\tau') = \begin{pmatrix} 2 & -1 & & & & & & \\ -1 & 4 & -1 & & & & 0 & \\ & -1 & 4 & -1 & & & & \\ & & -1 & \ddots & & & & \\ & & & & \ddots & -1 & & \\ & & & & -1 & 4 & -1 & \\ & 0 & & & & -1 & 4 & -1 \\ & & & & & & -1 & 2 \end{pmatrix}$$

LNP model + L_2-Regularisation

To have a smooth *w,* we should minimise $\big(w(\tau) - w(\tau+1) \big)^2$

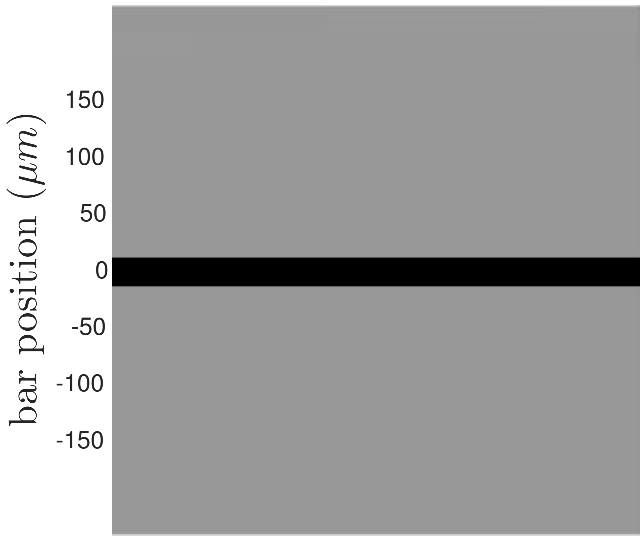
$$\ell_\lambda(w,b) = \sum_t \Big(n(t)\log f(t) - f(t) \Big) - \frac{\lambda}{2} \sum_{\tau,\tau'} w(\tau)\, L(\tau,\tau')\, w(\tau')$$

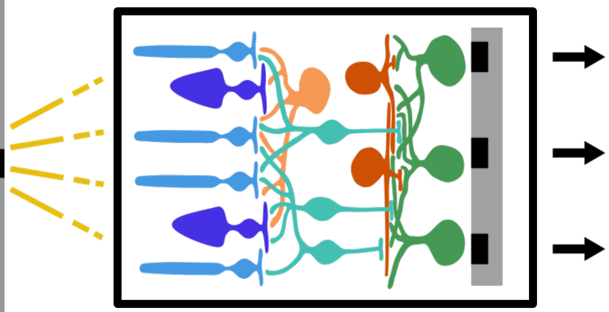
Laplacian: $$L(\tau,\tau') = \begin{pmatrix} 2 & -1 & & & & & & \\ -1 & 4 & -1 & & & 0 & & \\ & -1 & 4 & -1 & & & & \\ & & -1 & \ddots & & & & \\ & & & & \ddots & -1 & & \\ & & & & -1 & 4 & -1 & \\ & 0 & & & & -1 & 4 & -1 \\ & & & & & & -1 & 2 \end{pmatrix}$$

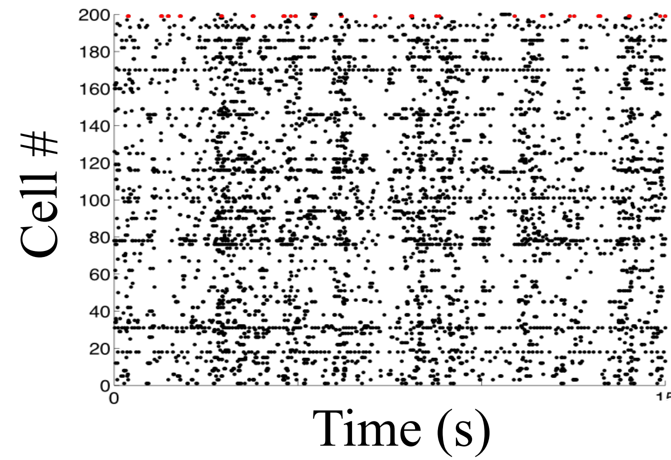
And the strength λ should be optimised over the validation set

Predicting retinal light responses to complex stimuli

What happens if the stimulus is more complex?

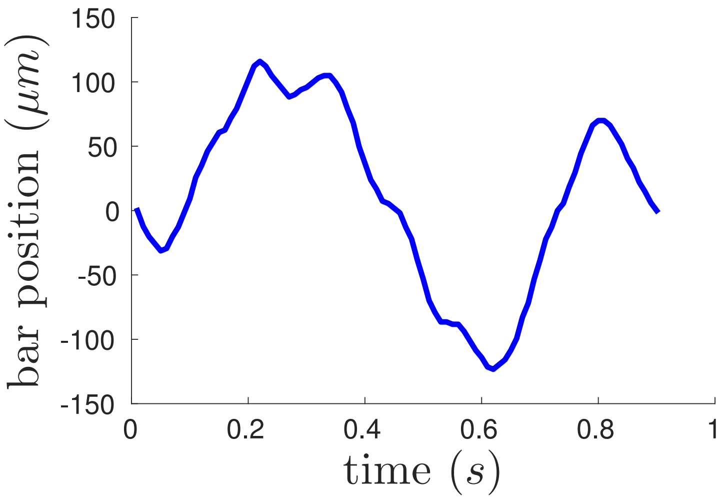






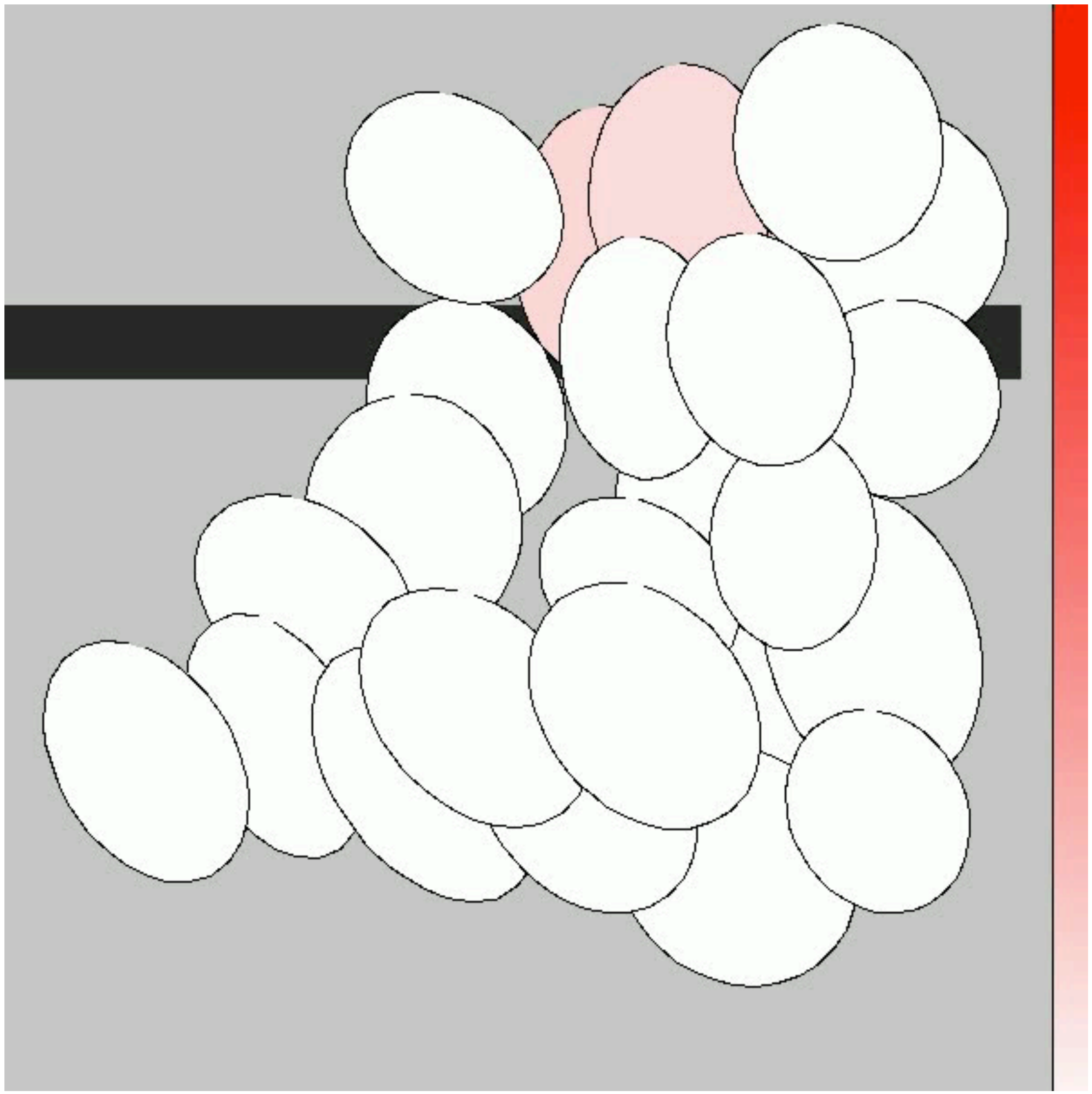
Relevant: test the tracking of moving object

Simple: fully described by one dimensional trajectory

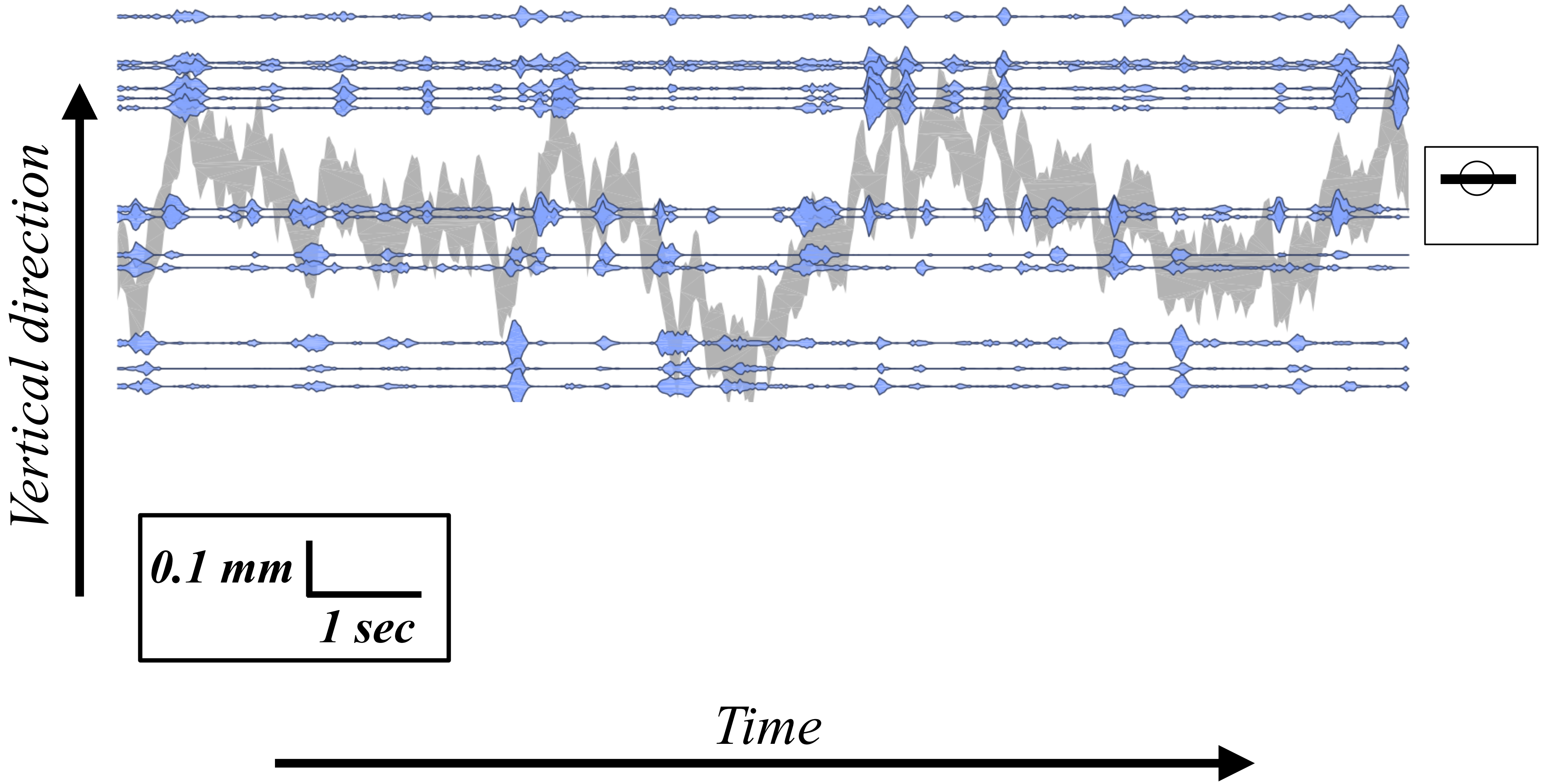


Predicting retinal light responses to complex stimuli

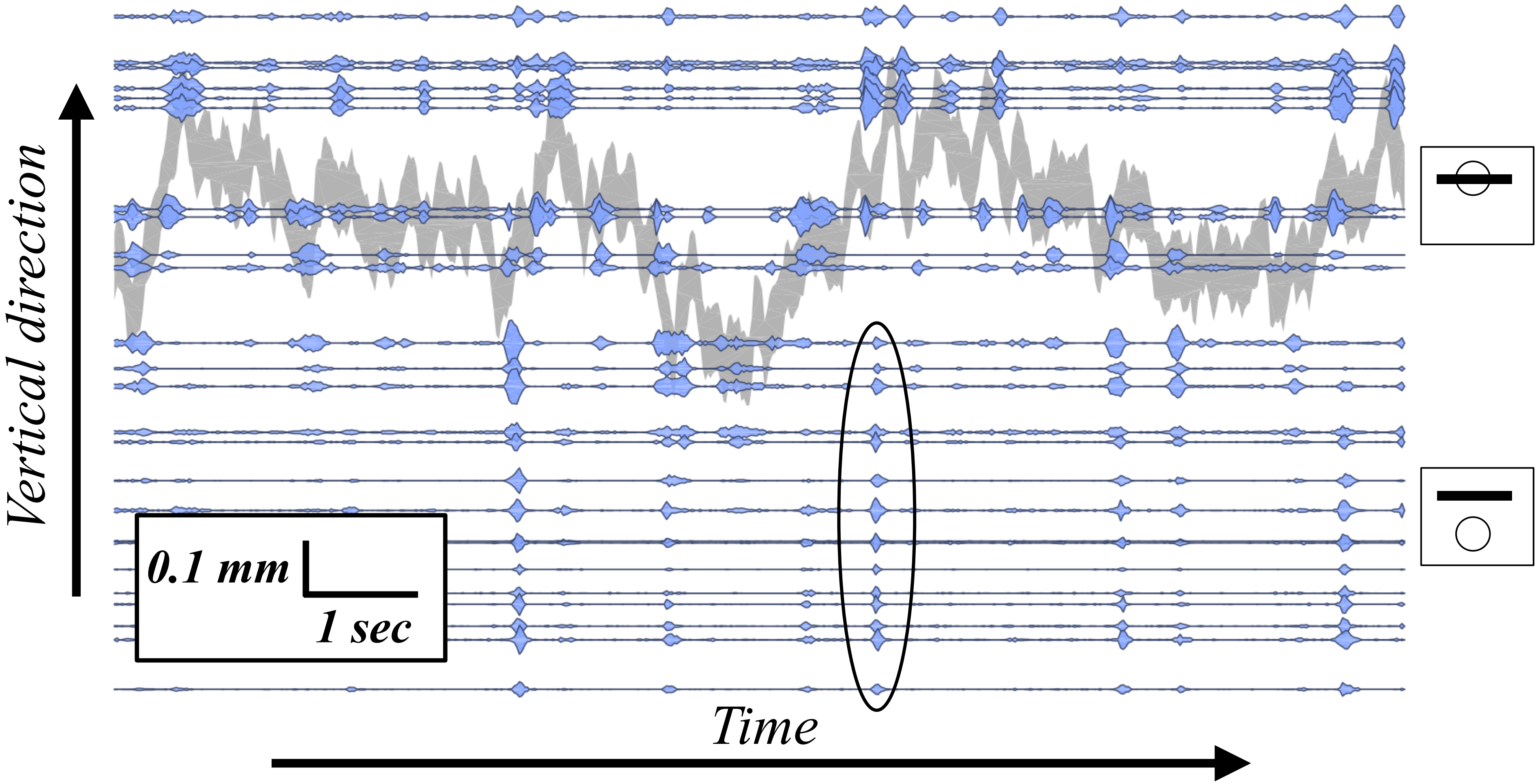
What happens if the stimulus is more complex?



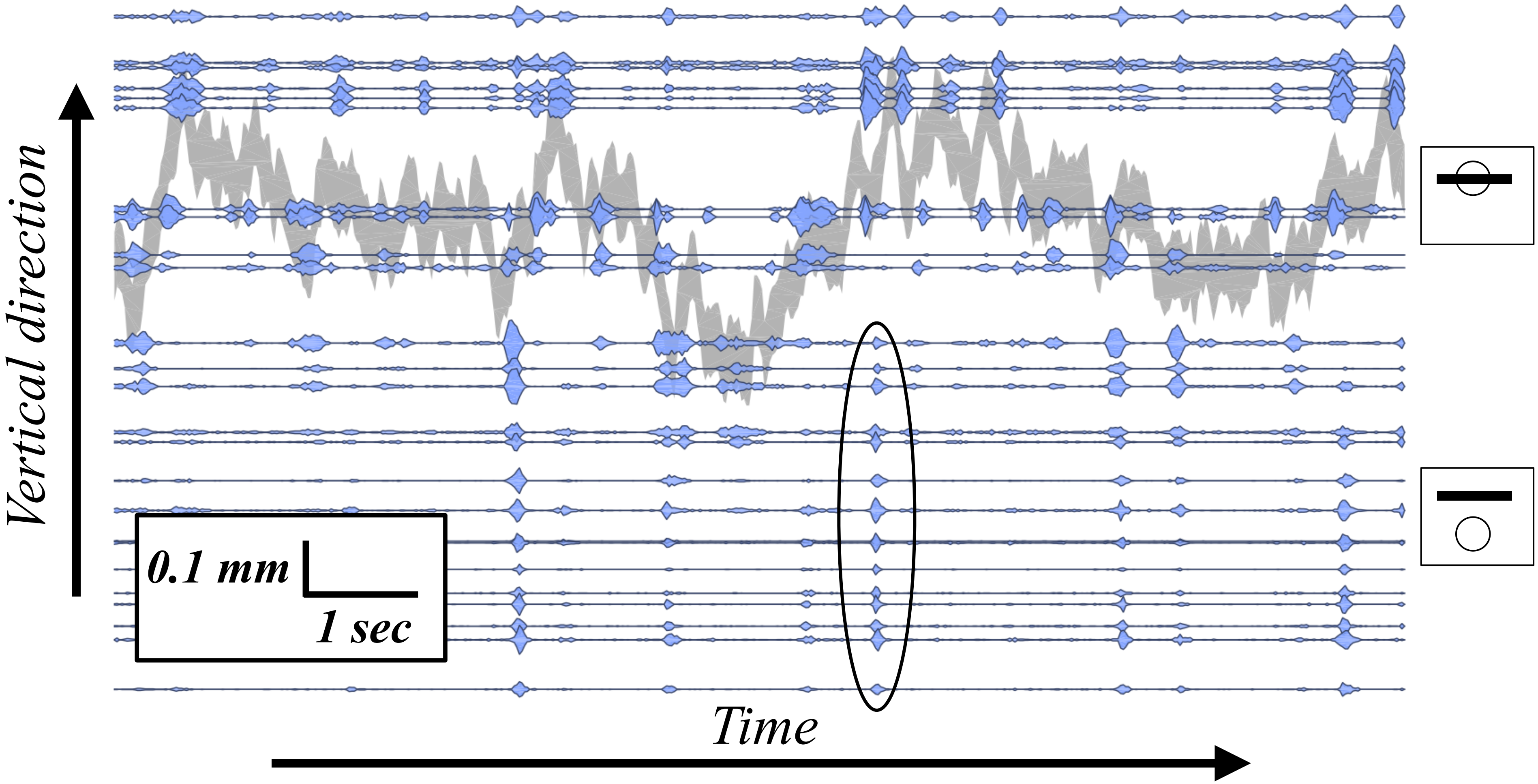
Predicting retinal light responses to complex stimuli



Predicting retinal light responses to complex stimuli



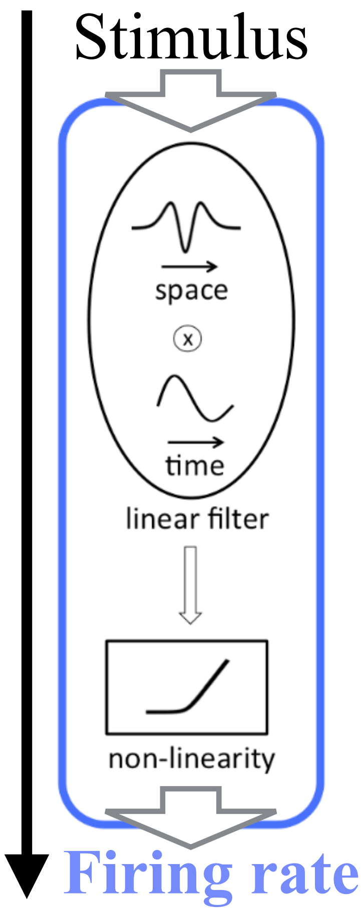
Predicting retinal light responses to complex stimuli

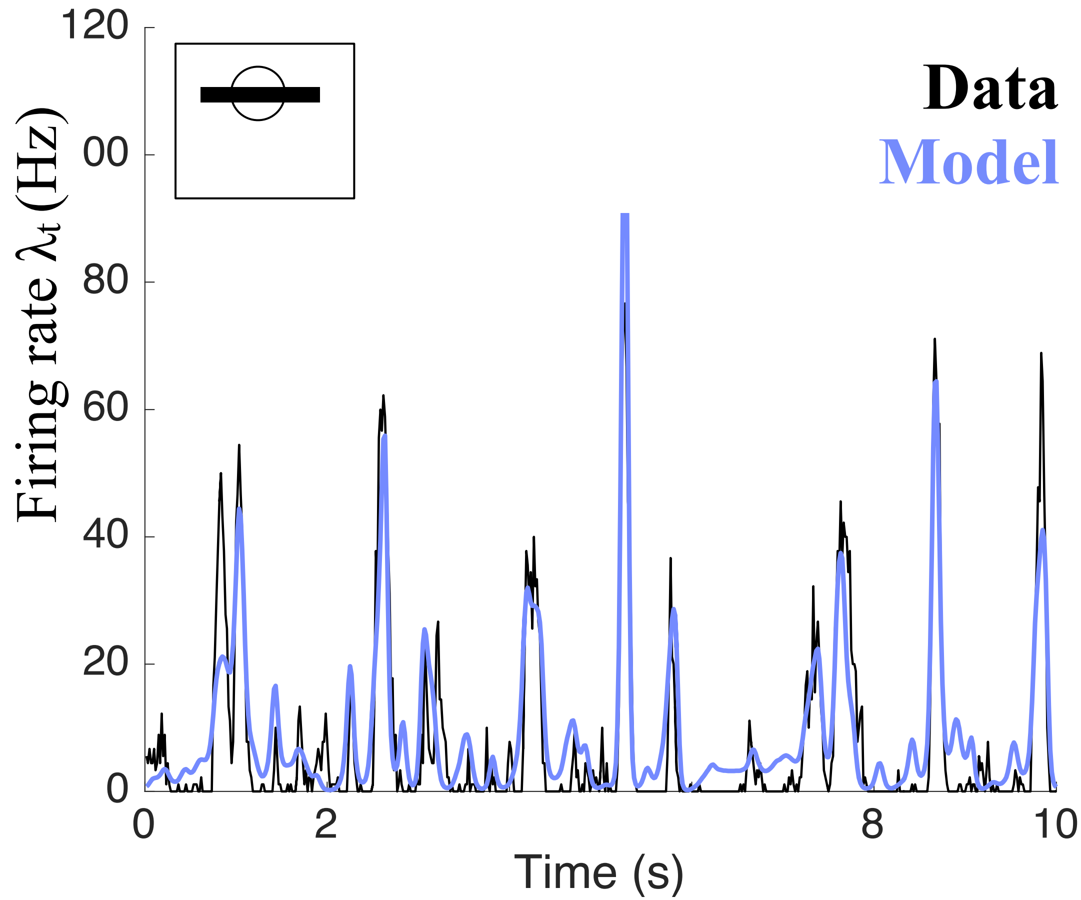


Same type of cell, but different computations!

Predicting retinal light responses to complex stimuli

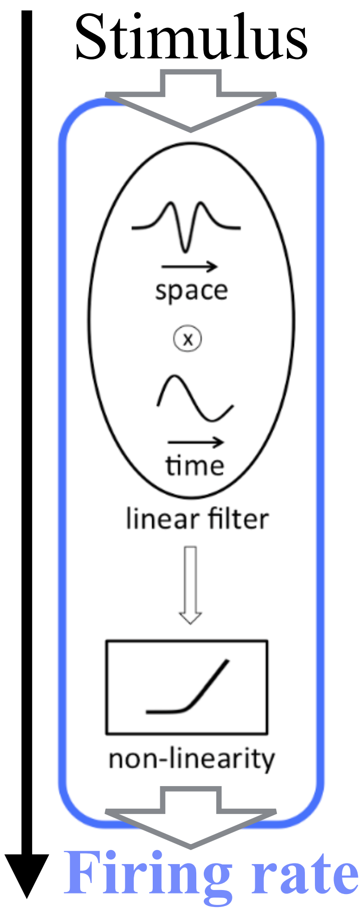
*A simple **LNP** model…*

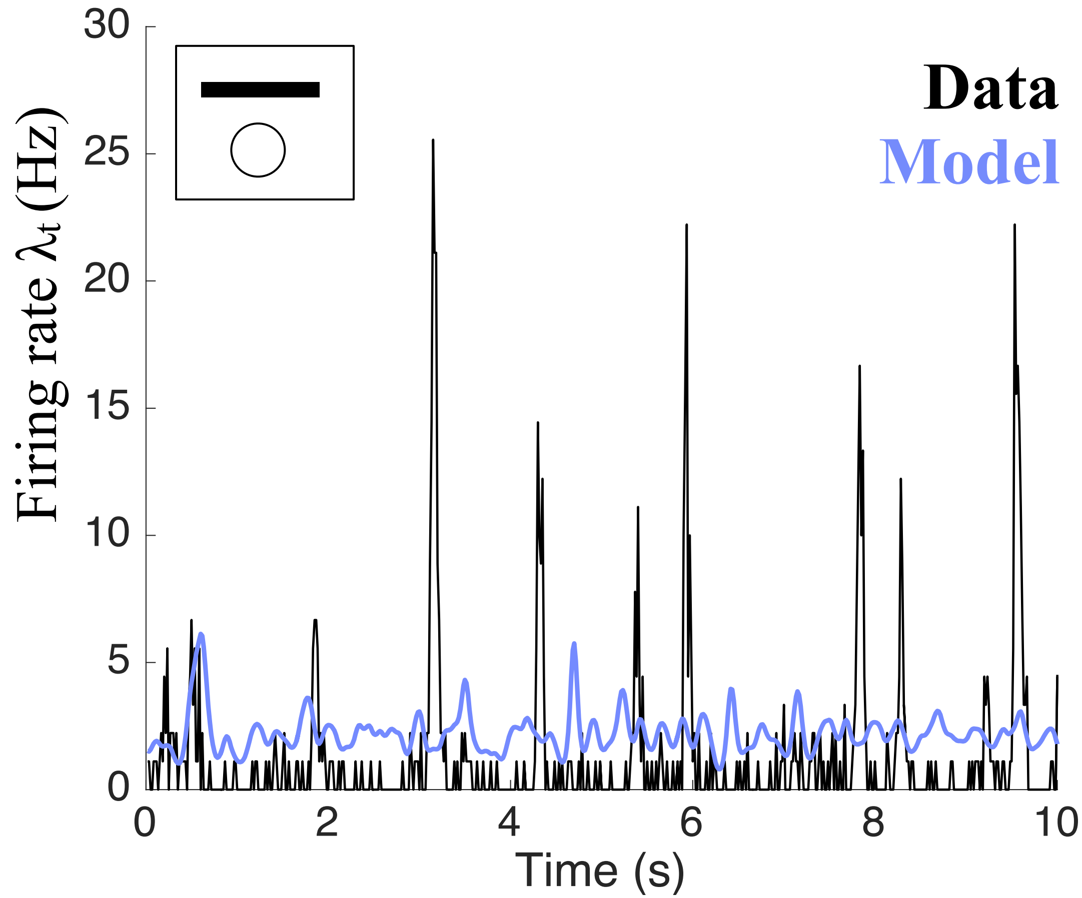




Predicting retinal light responses to complex stimuli

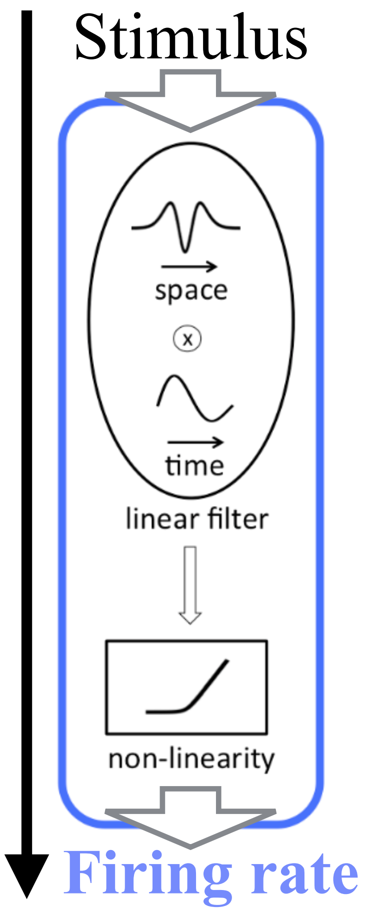
*A simple **LNP** model may not predict the response*

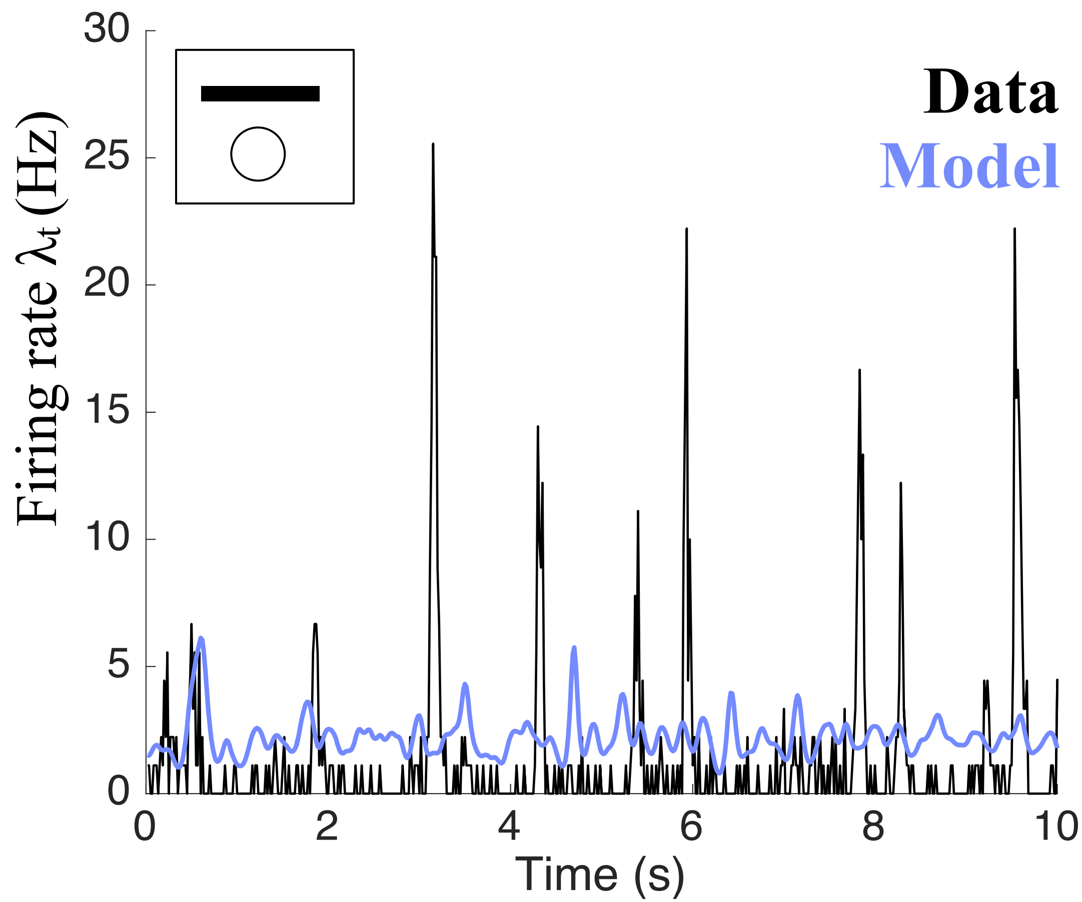




Predicting retinal light responses to complex stimuli

*A simple **LNP** model may not predict the response*

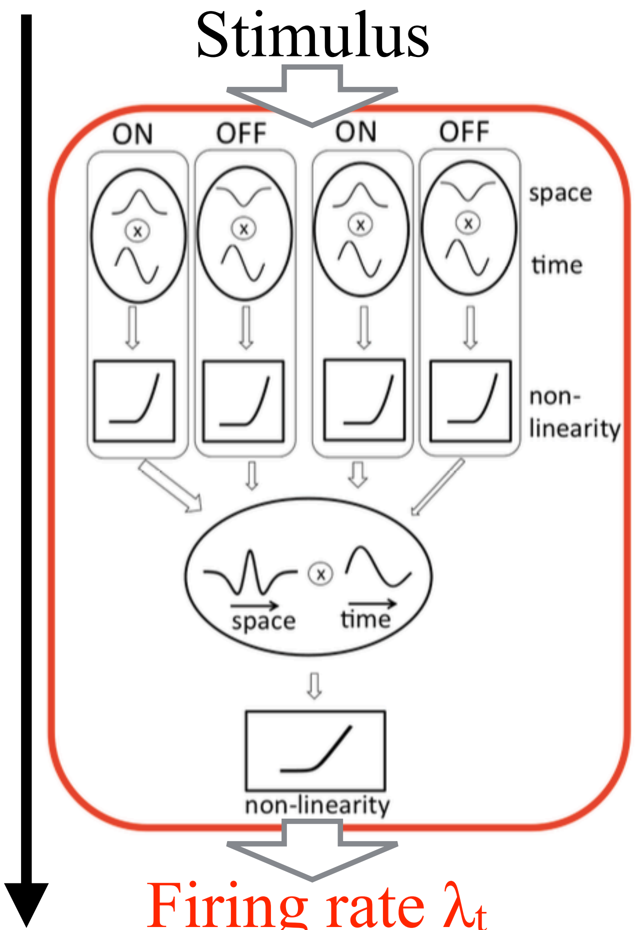




*A **LNP** model is not enough flexible!*

Predicting retinal light responses to complex stimuli

*An **highly non-linear** LN²P model…*



Sub-unit filter

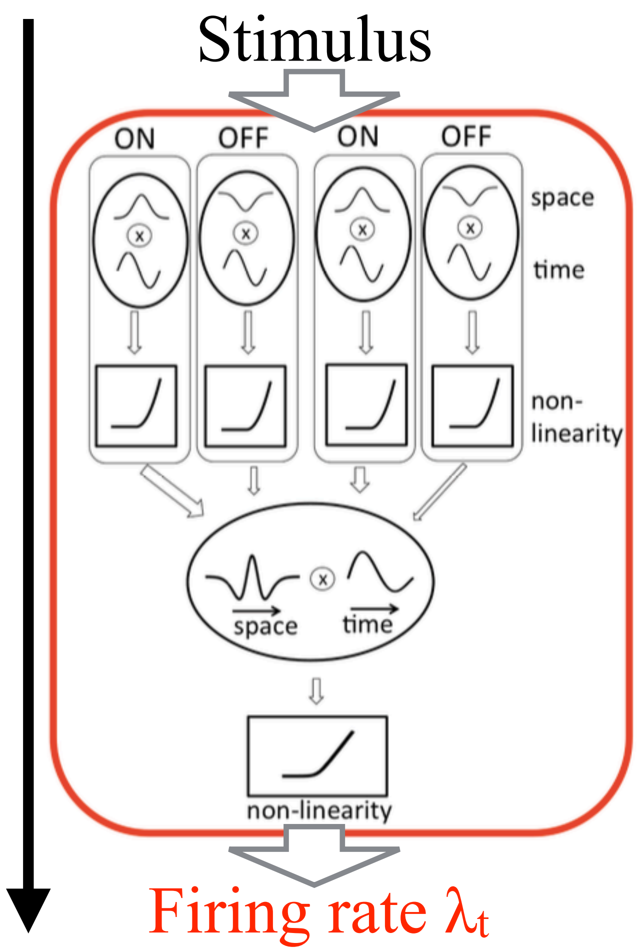
Nonlinear threshold

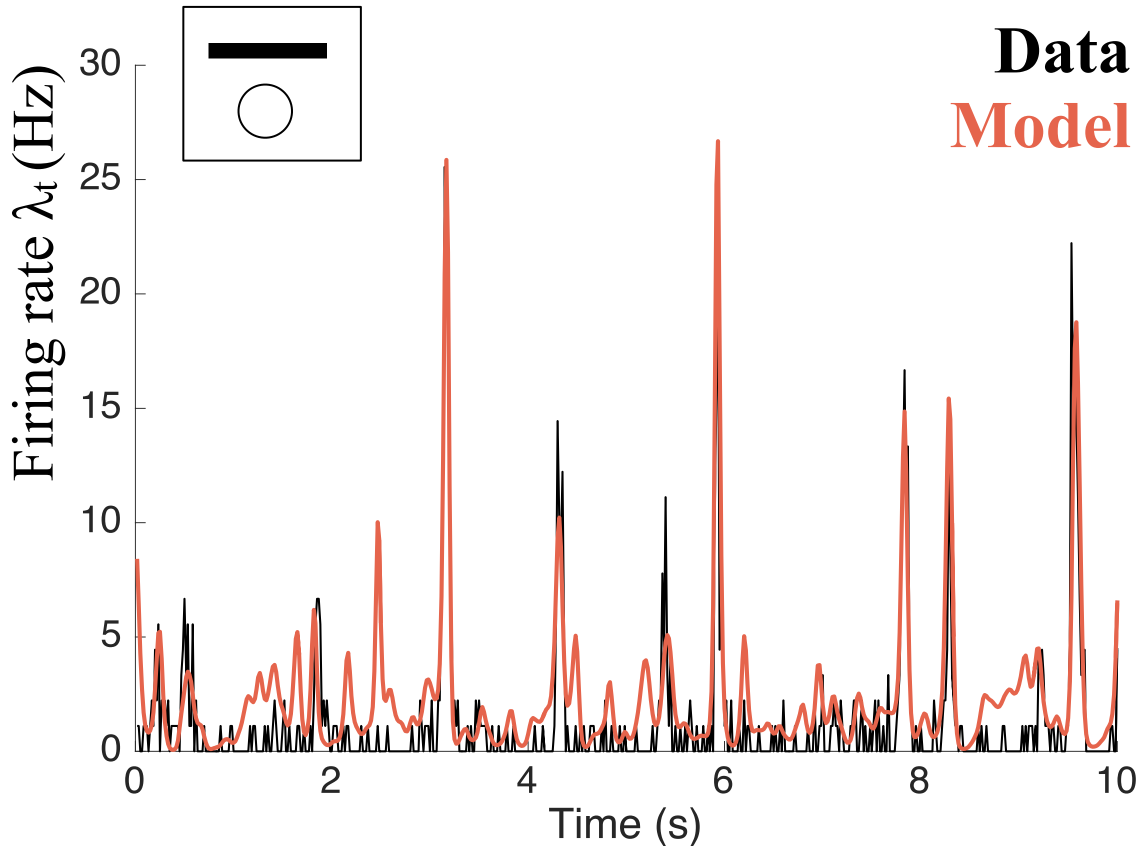
Receptive field filter

Nonlinear threshold

Predicting retinal light responses to complex stimuli

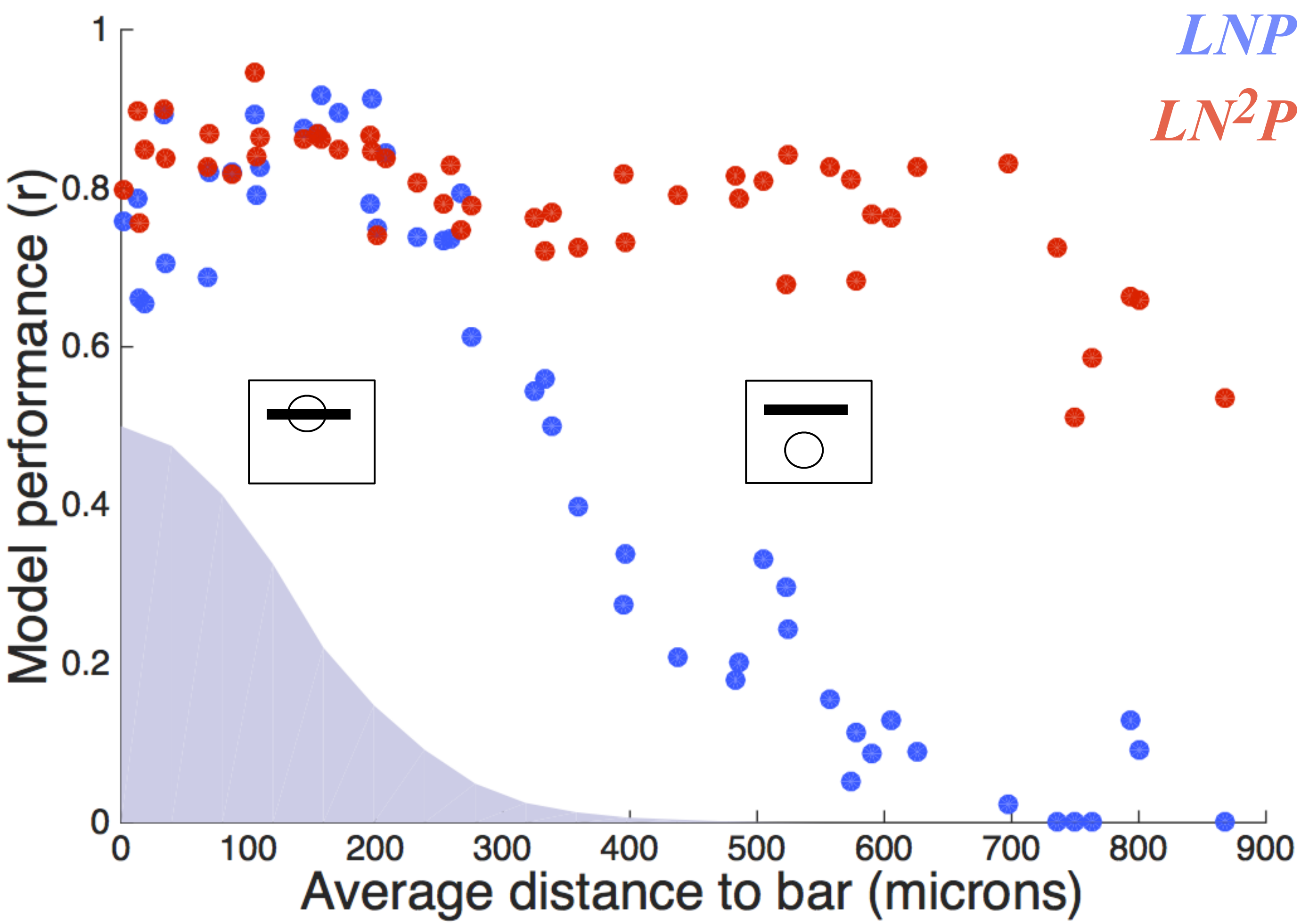
*An **highly non-linear** **LN²P** model predicts the response*





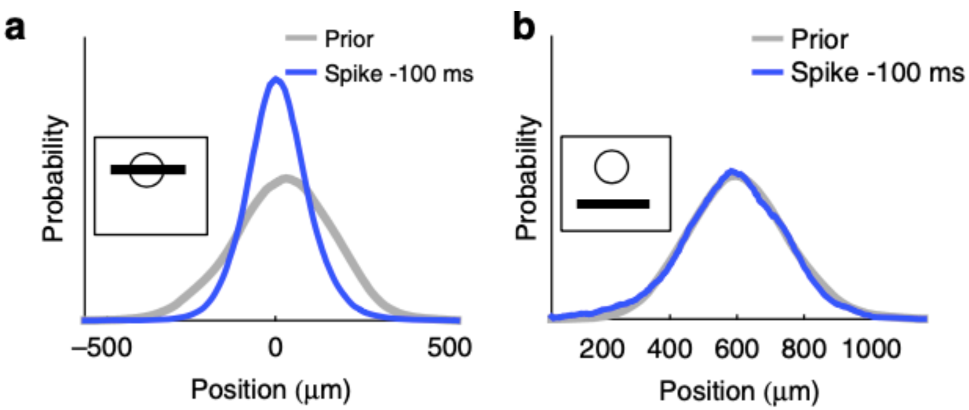
Predicting retinal light responses to complex stimuli

An ***highly non-linear*** *LN²P model predicts the response*



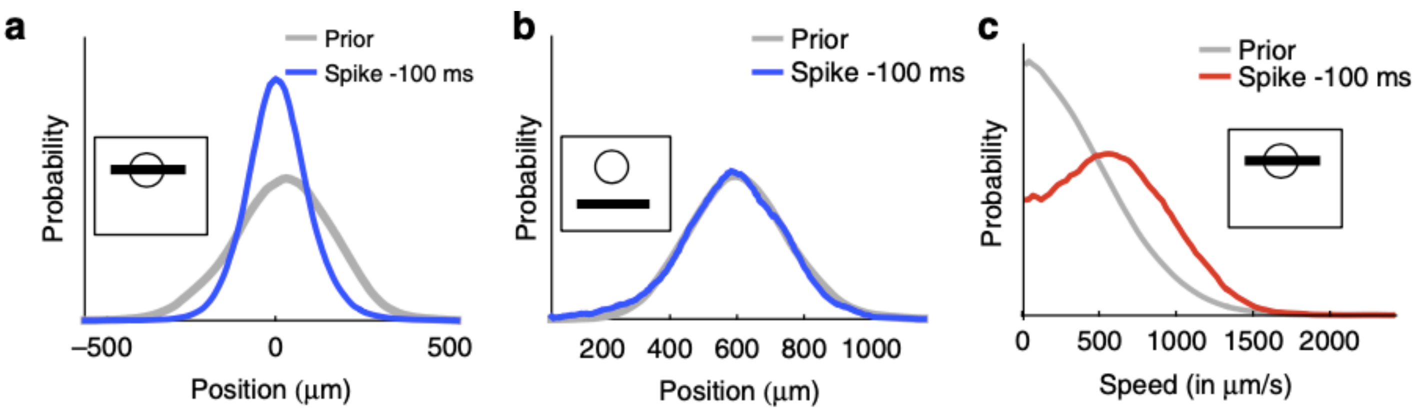
What are the neurons encoding?

How to quantify information carried by the spikes?



What are the neurons encoding?

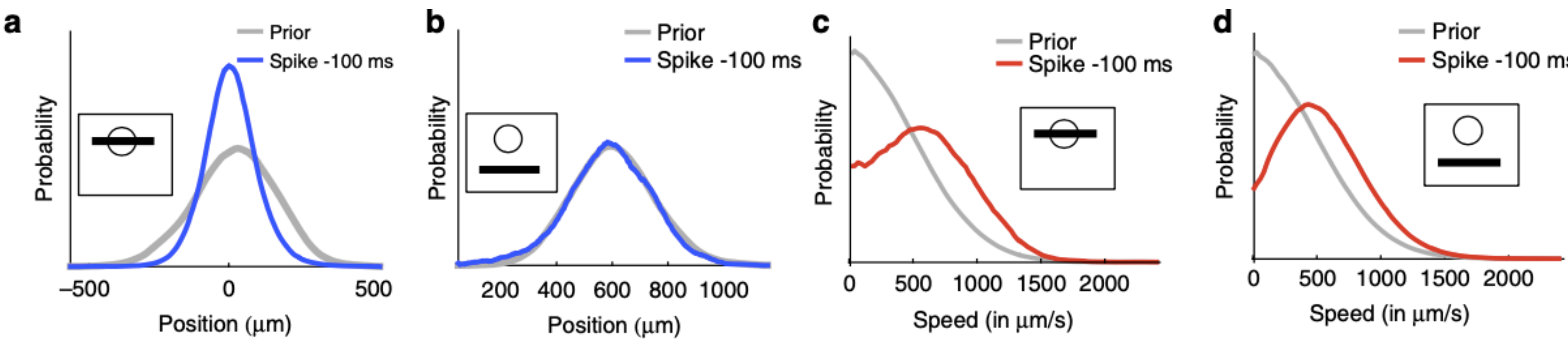
How to quantify information carried by the spikes?



Same can be done for speed…

What are the neurons encoding?

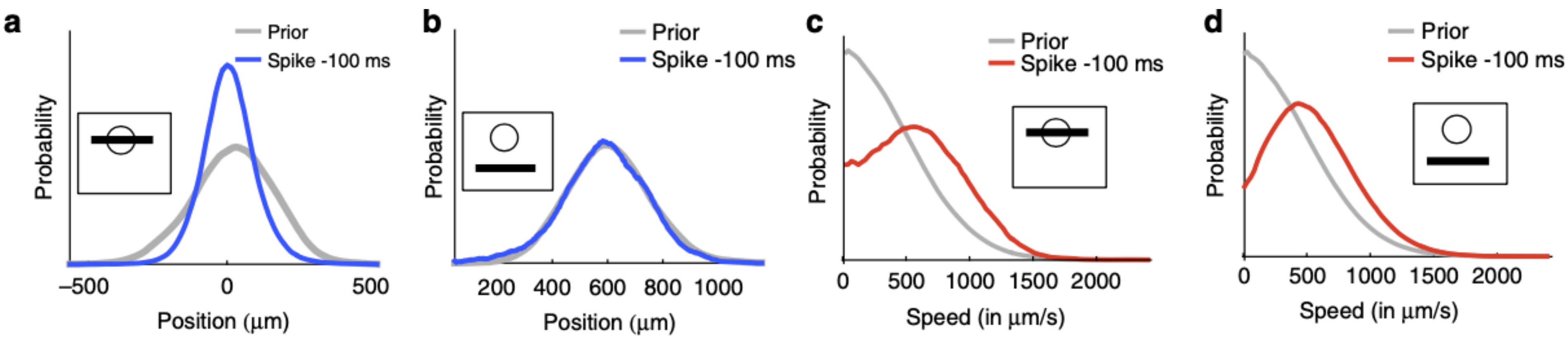
How to quantify information carried by the spikes?



Same can be done for speed…

What are the neurons encoding?

How to quantify information carried by the spikes?

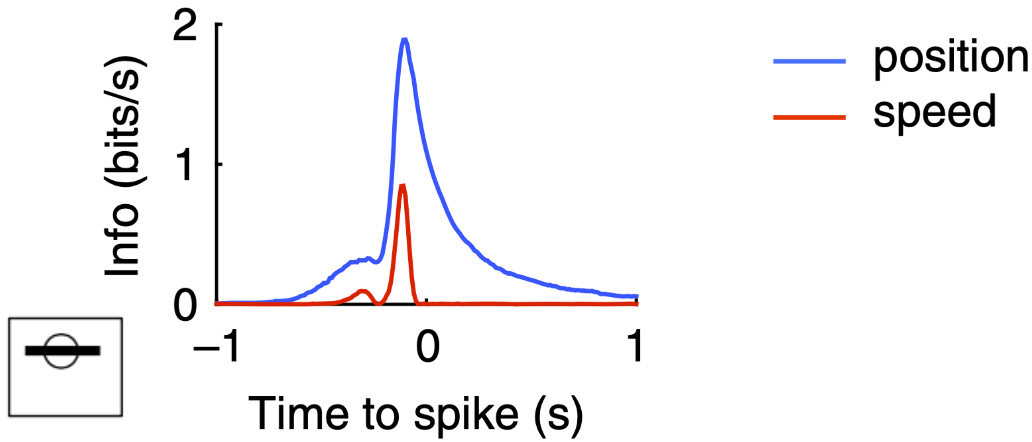


Mutual information between **pos** and **spikes**:

$$\mathcal{H}[\mathrm{pos}] - p(spike)\mathcal{H}[\,\mathrm{pos}\,|\,spike\,] - p(no\text{-}spike)\mathcal{H}[\,\mathrm{pos}\,|\,no\text{-}spike\,]$$

Same can be done for speed…

What are the neurons encoding?

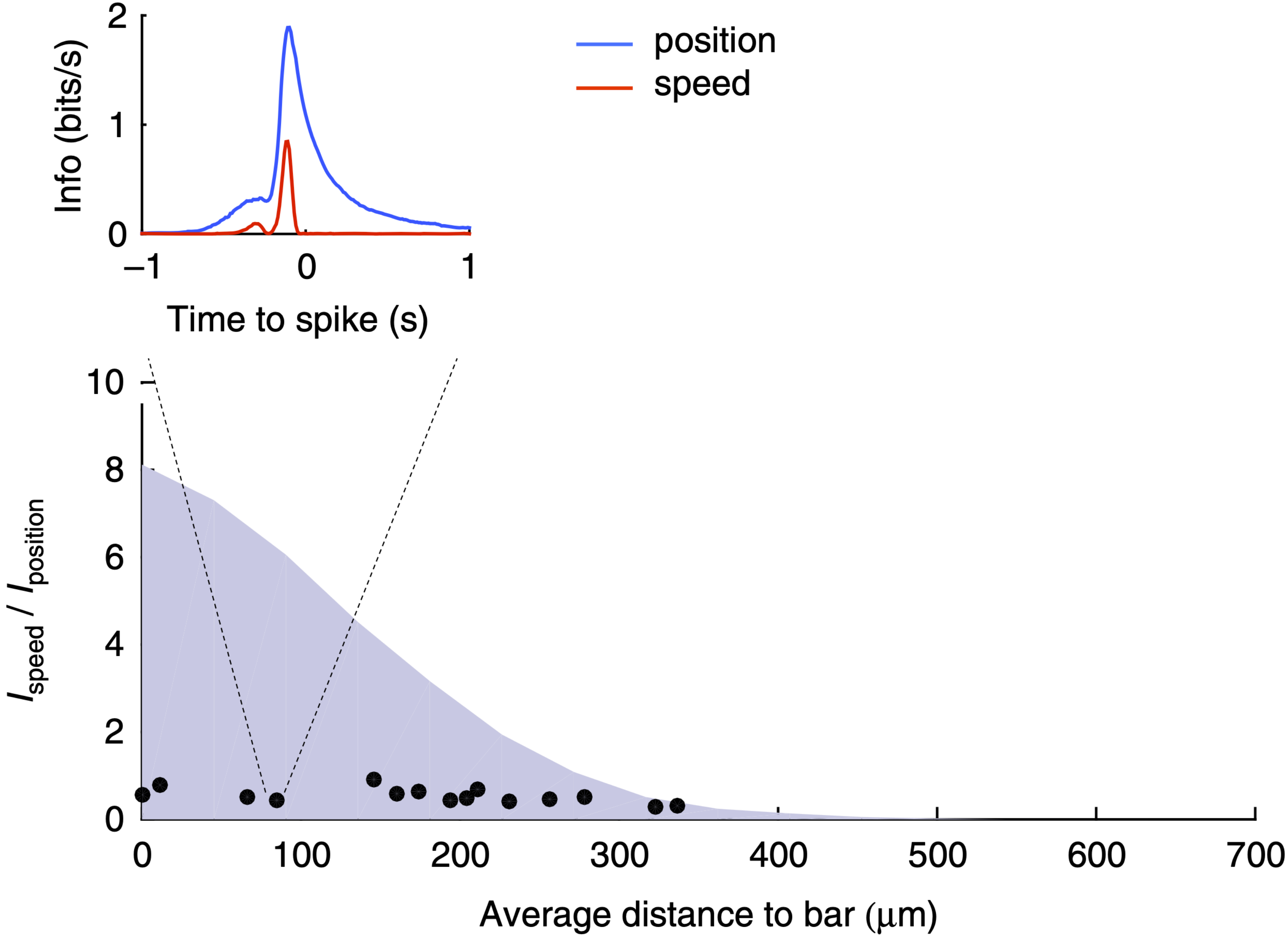


Mutual information between **pos** and **spikes**:

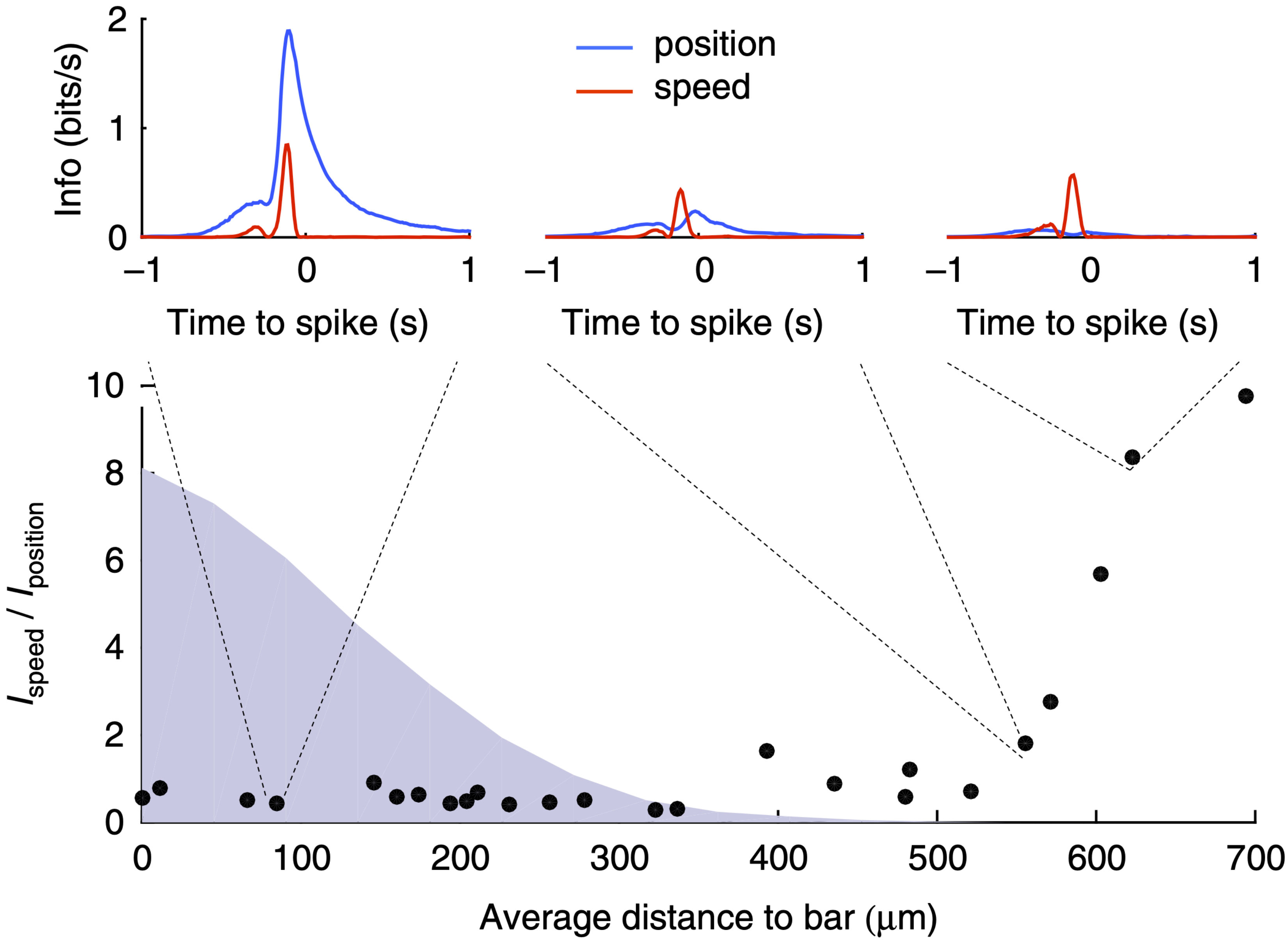
$$\mathcal{H}[\mathrm{pos}] - p(spike)\mathcal{H}[\,\mathrm{pos} \mid spike\,] - p(no\text{-}spike)\mathcal{H}[\,\mathrm{pos} \mid no\text{-}spike\,]$$

Same can be done for speed…

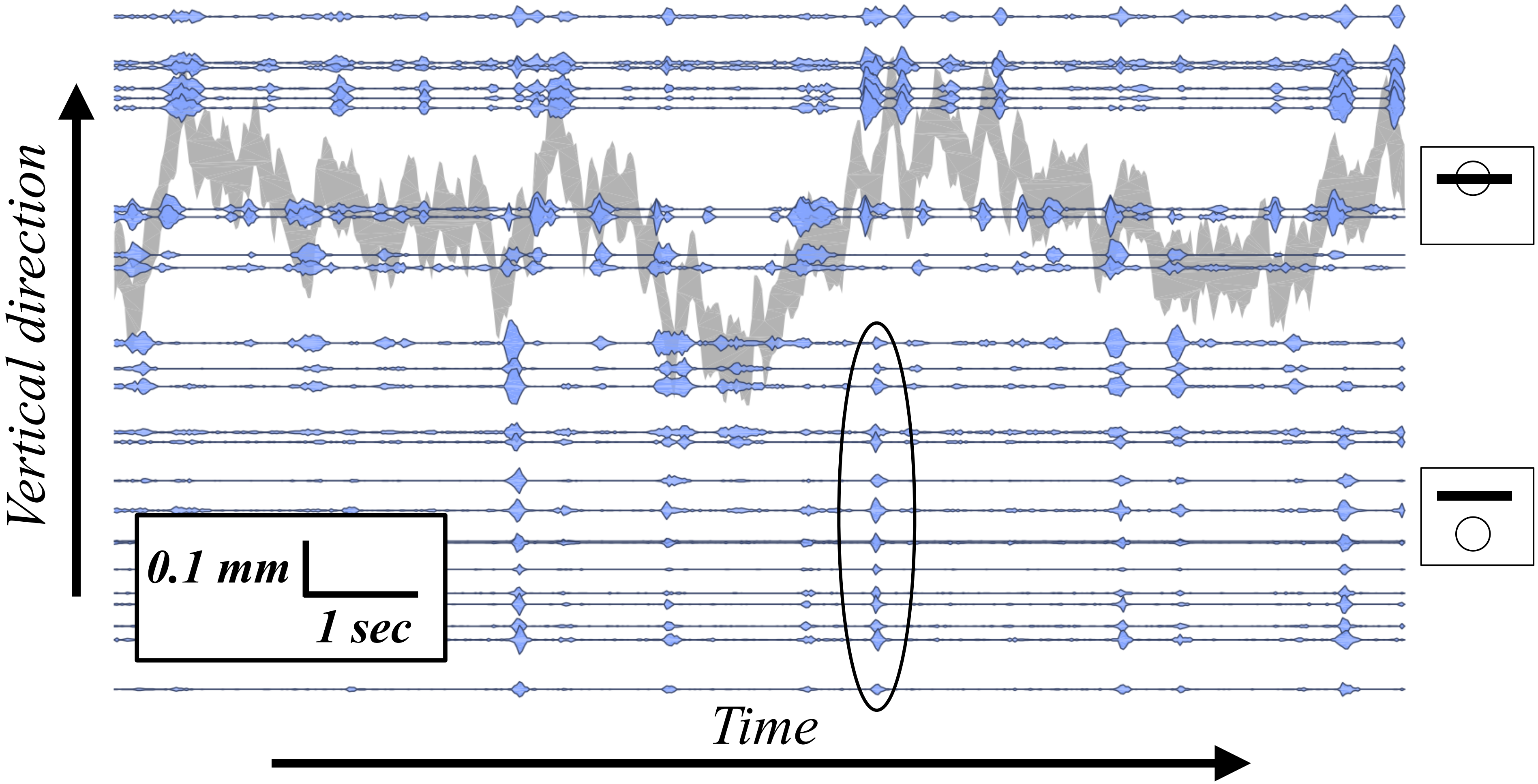
What are the neurons encoding?



What are the neurons encoding?



Predicting retinal light responses to complex stimuli



Same type of cell, but different computations!

Eyes are smarter than scientists believed

Ulisse Ferrari

ulisse.ferrari@gmail.com

INSTITUT DE LA VISION PARIS