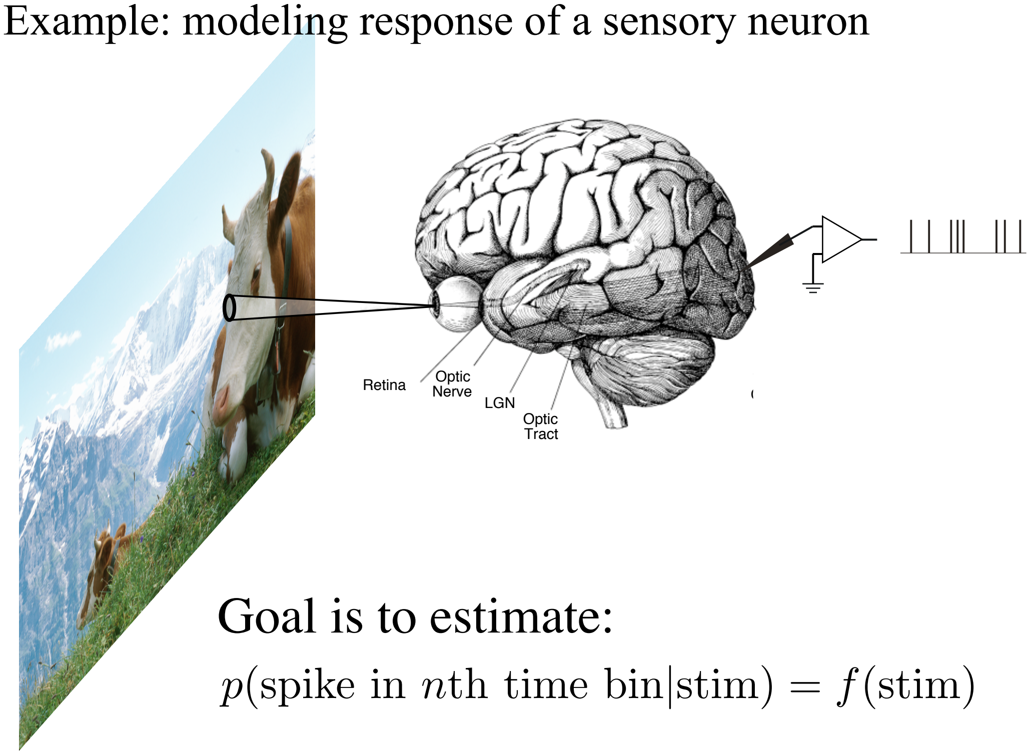
Fitting models to data

- How do we estimate parameters?
  - formulate model + objective function
  - optimize
- How good is fit?
  - bias
  - variance
  - model failures

Example: modeling response of a sensory neuron

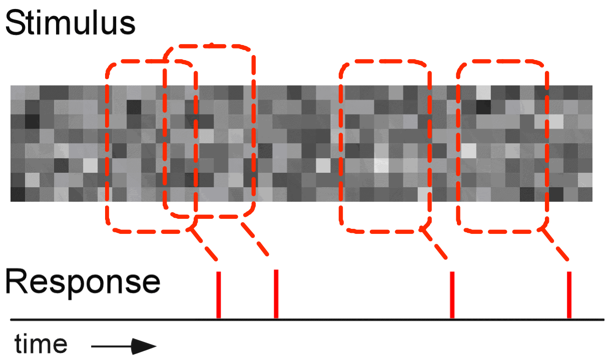


Goal is to estimate:

$p(\text{spike in } n\text{th time bin}|\text{stim}) = f(\text{stim})$

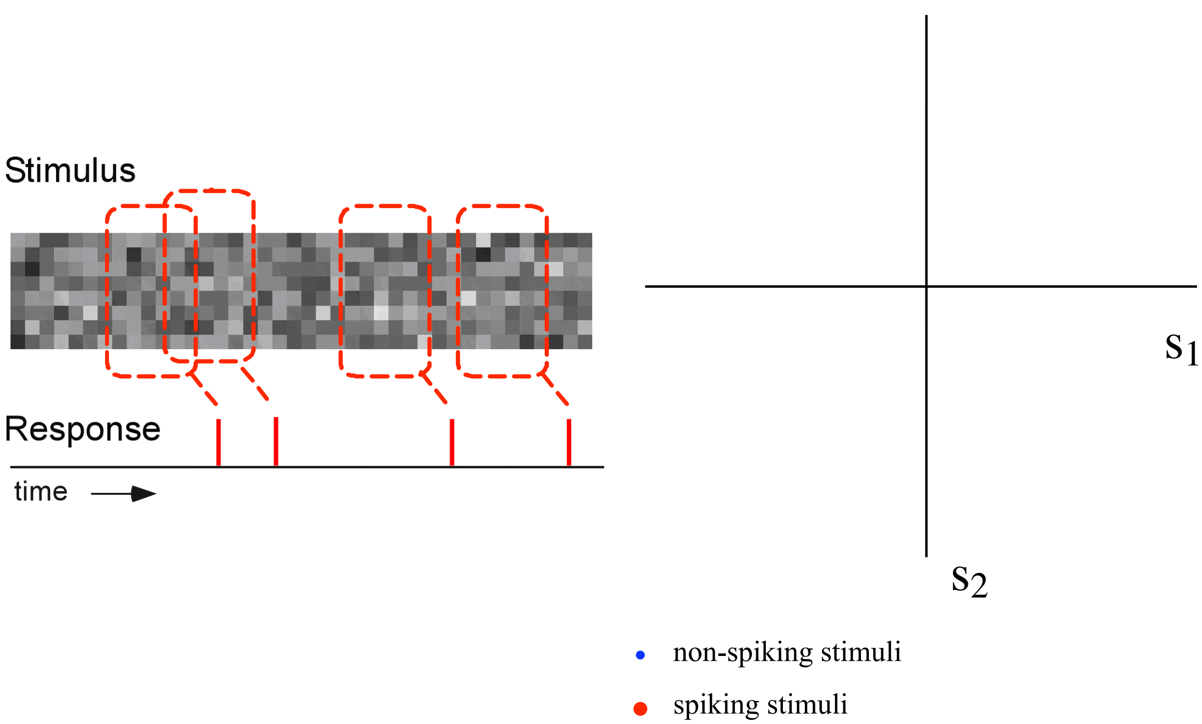
Geometric view

1D stimulus over time
(e.g., flickering bars)

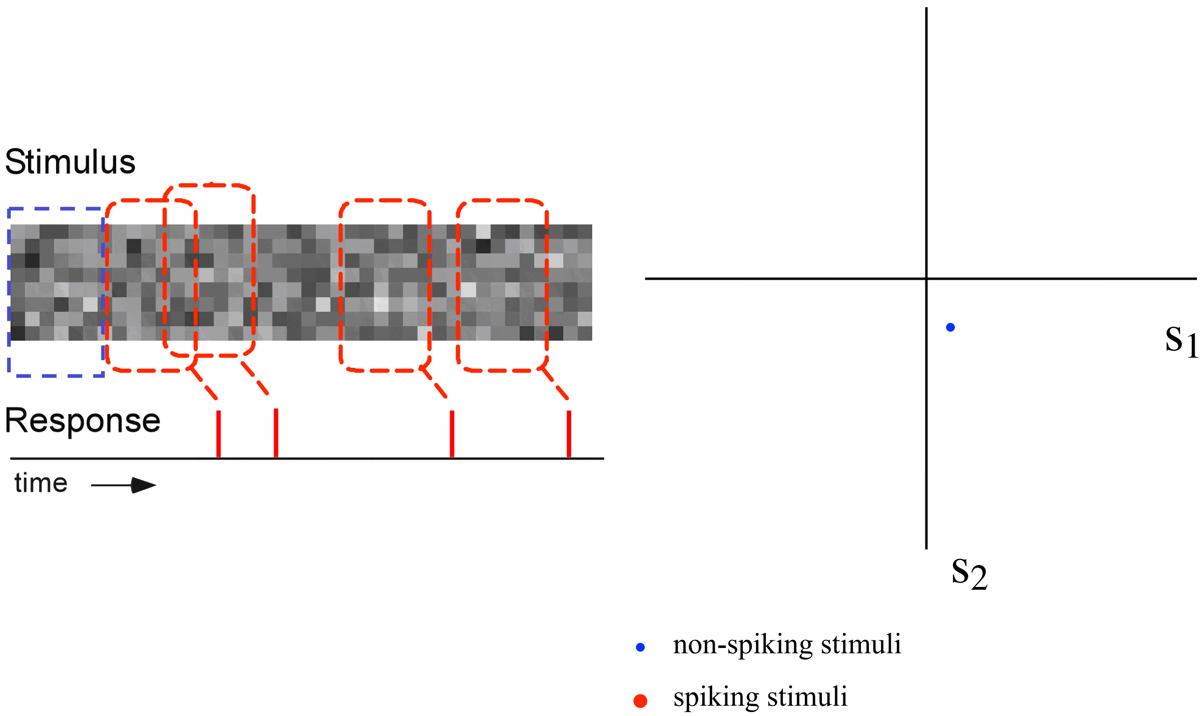


- 8 x 6 stimulus block
  = 48-dimensional vector

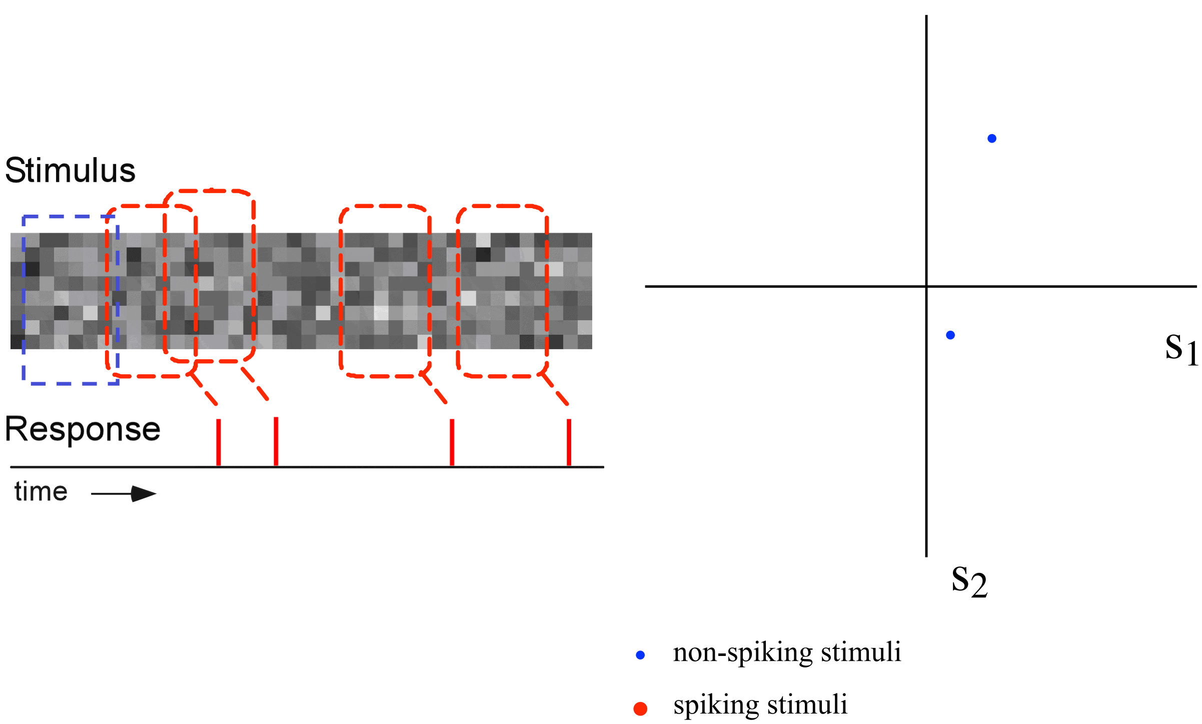
Geometric picture



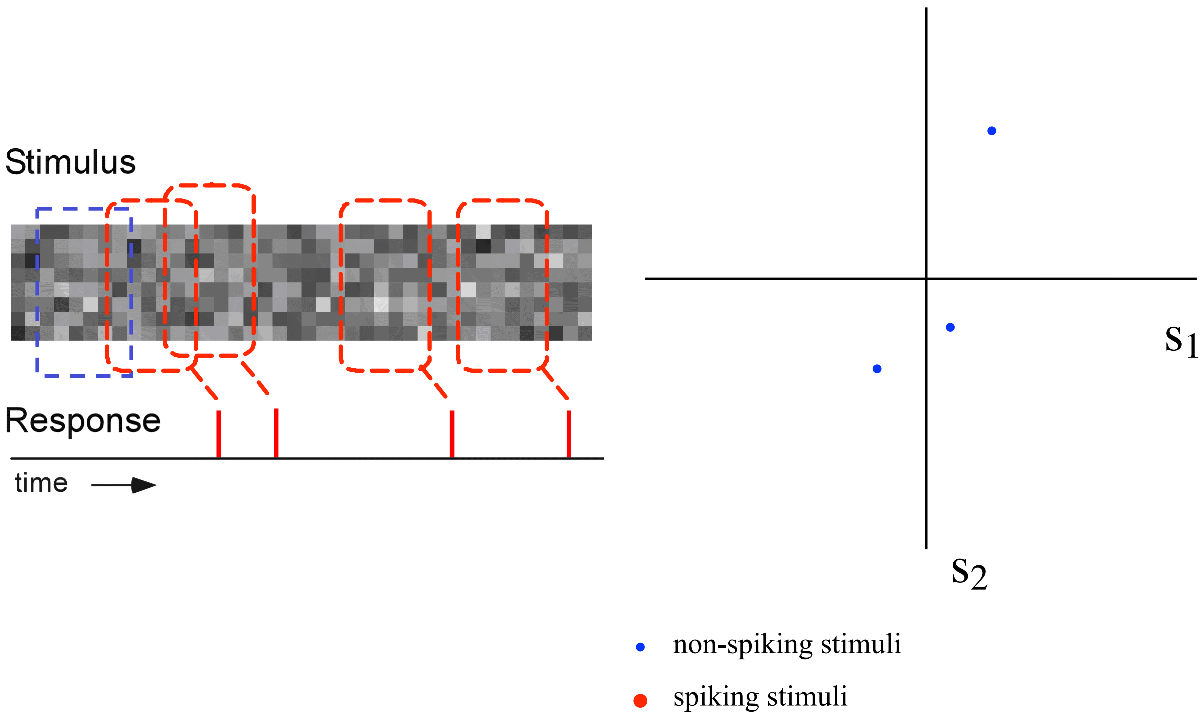
Geometric picture



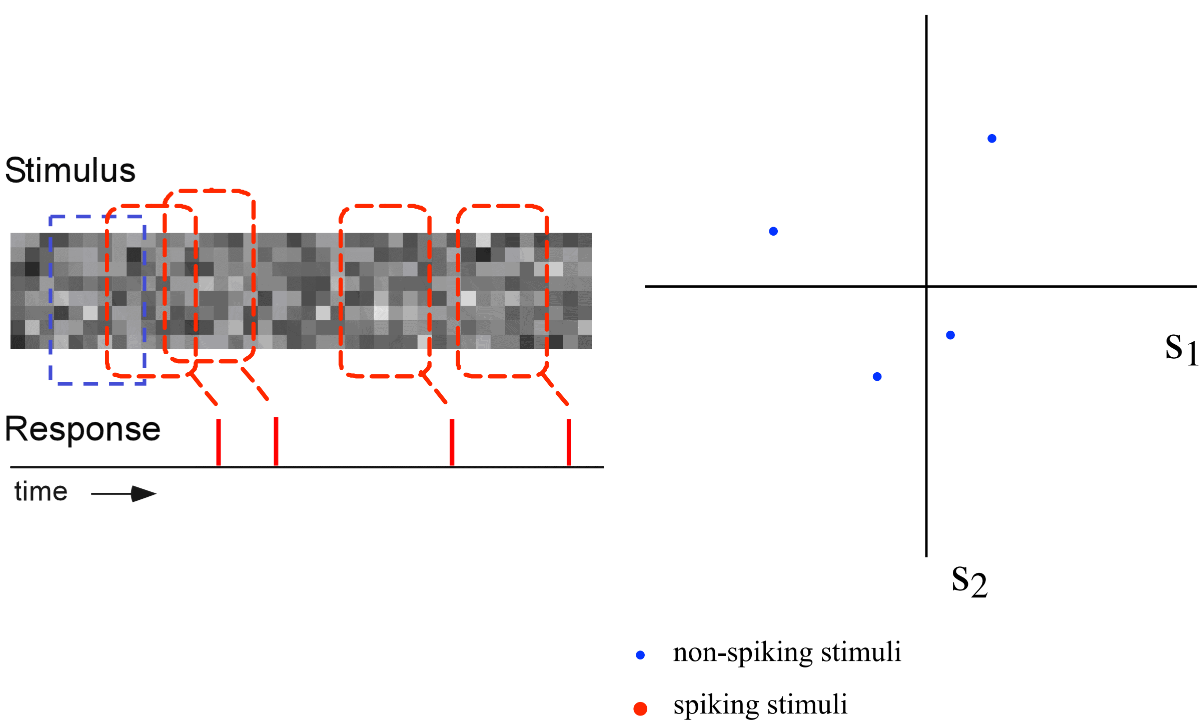
Geometric picture



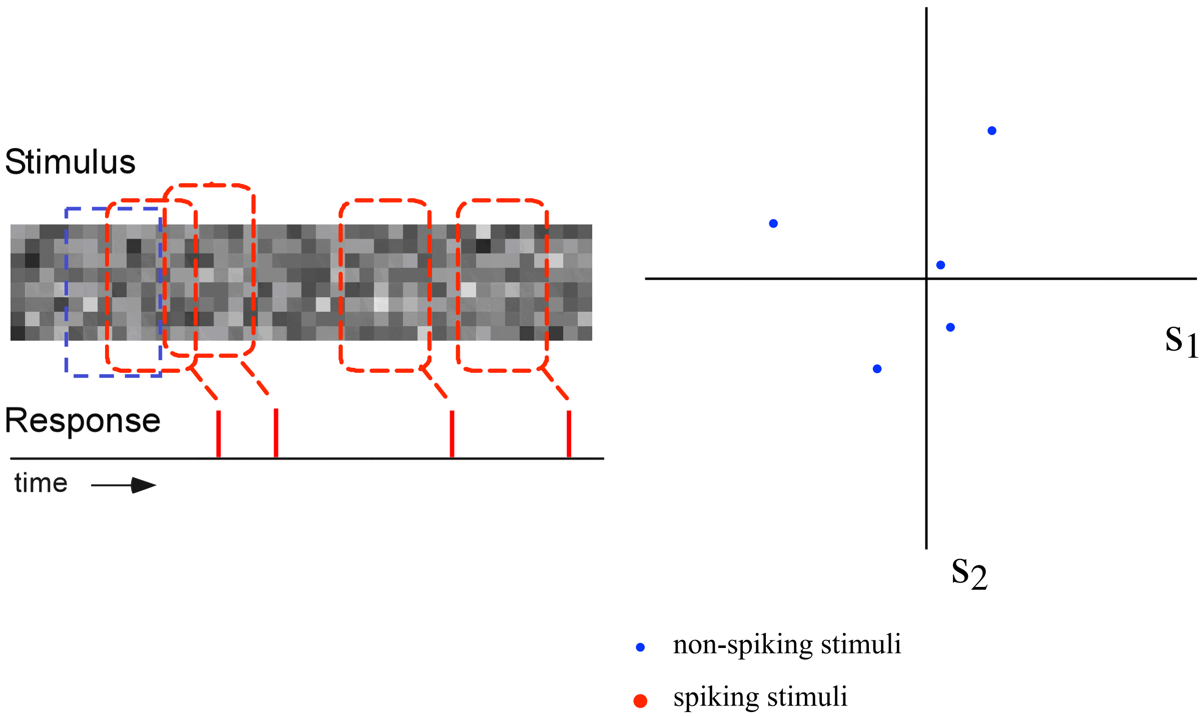
Geometric picture



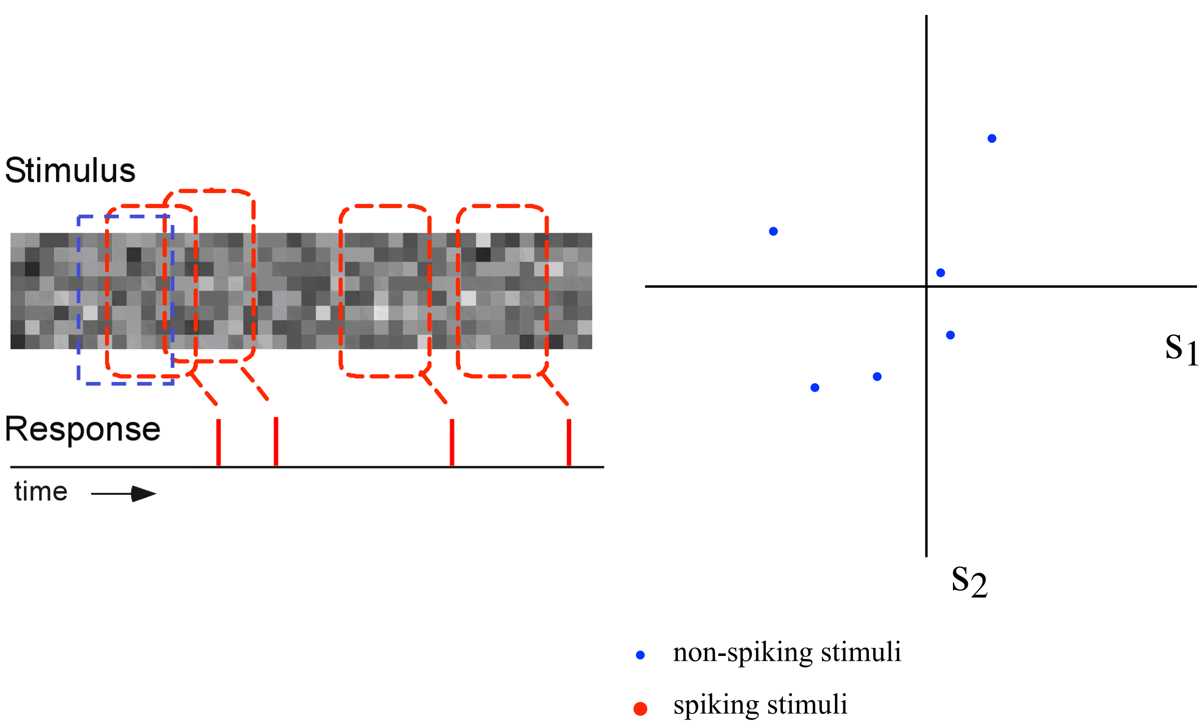
Geometric picture



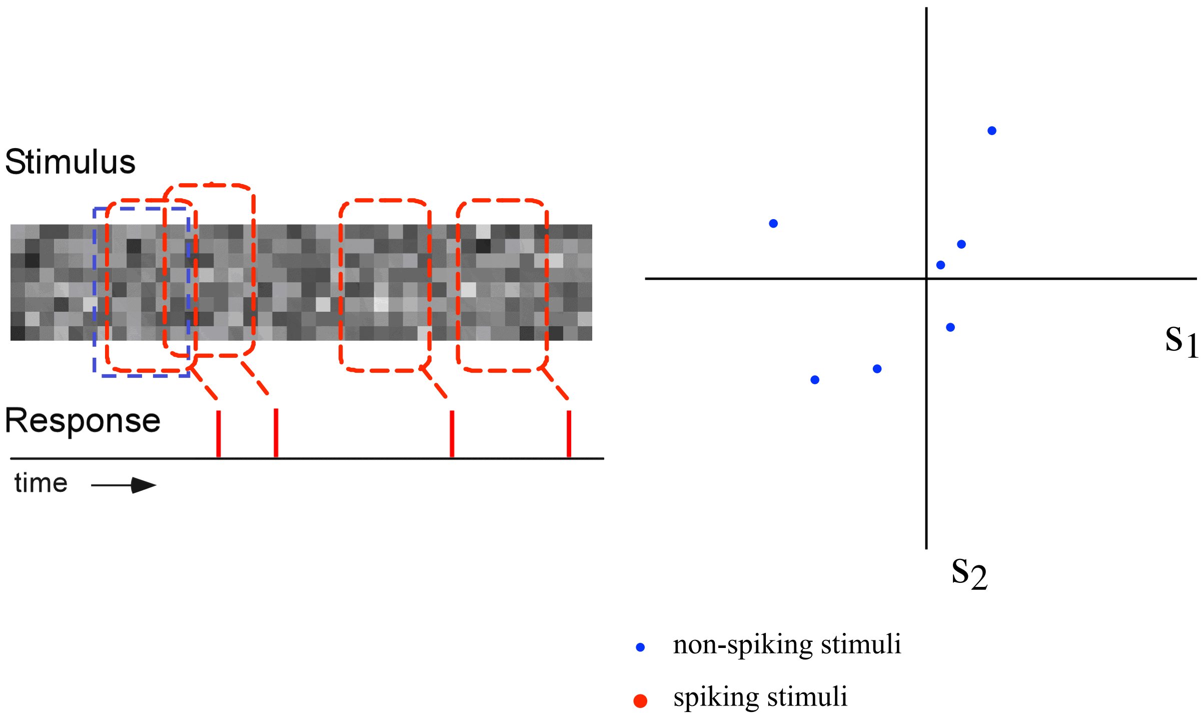
Geometric picture



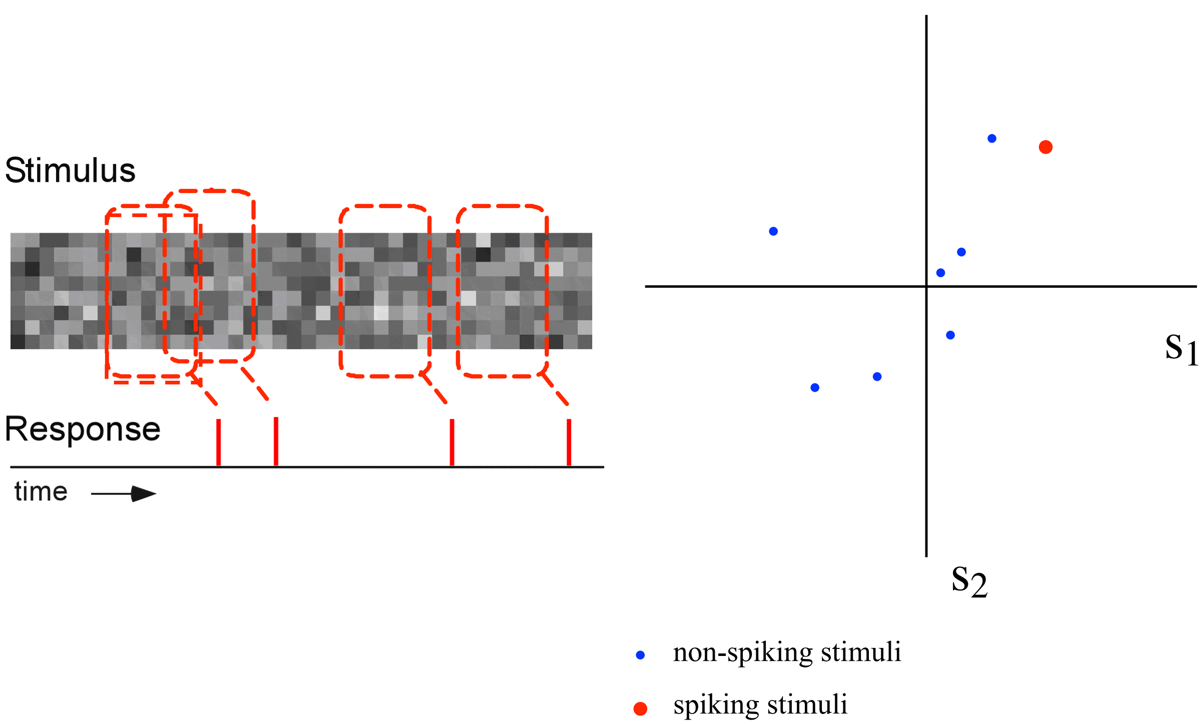
Geometric picture



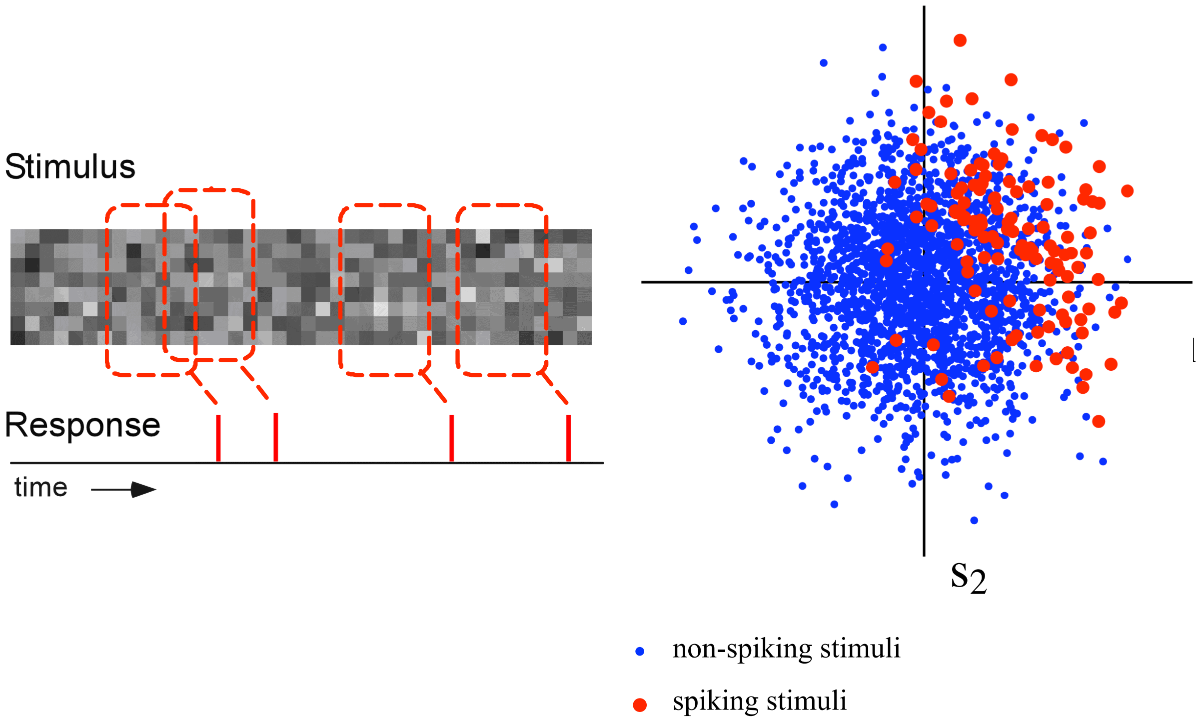
Geometric picture



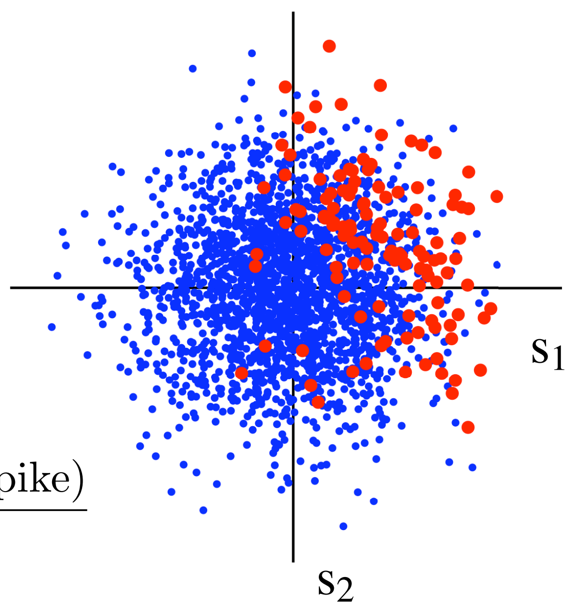
Geometric picture



Geometric picture



Response is captured by relationship between the distribution of red points (spiking stim) and blue+red points (all stim), expressed in terms of Bayes' rule:



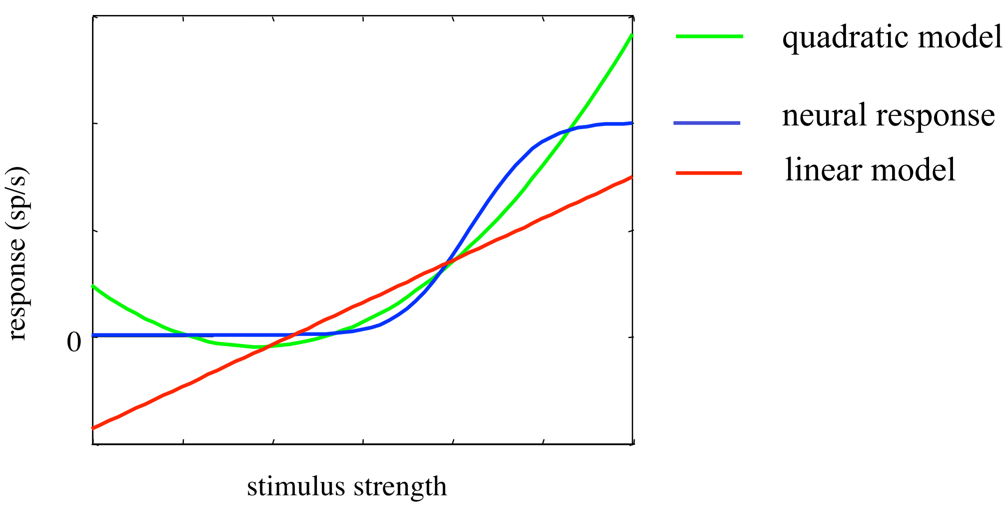
$$P(\text{spike}|\vec{s}) = \frac{P(\vec{s}\,|\text{spike})P(\text{spike})}{P(\vec{s})}$$

Cannot estimate directly ("curse of dimensionality").
We need a **model**

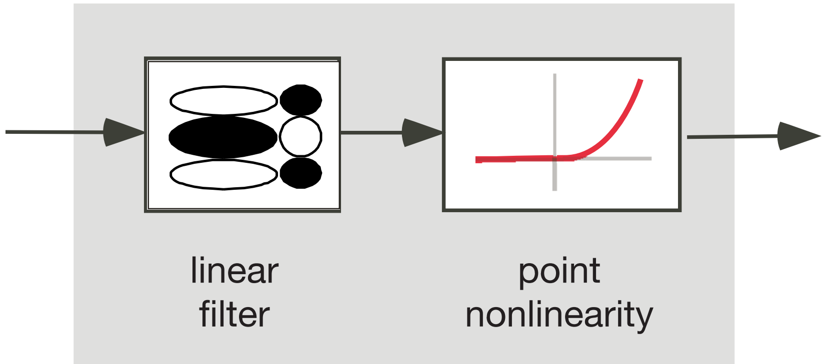
Some tractable model options

- Low-order polynomial [Volterra '13; Wiener '58; DeBoer and Kuyper '68; ...]
- Low-dimensional subspace [Bialek '88; Brenner etal '00; Schwartz etal '01; Touryan and Dan '02; ...]
- Recursive linear with exponential nonlinearity [Truccolo etal '05; Pillow etal '05]

Low-order polynomial model

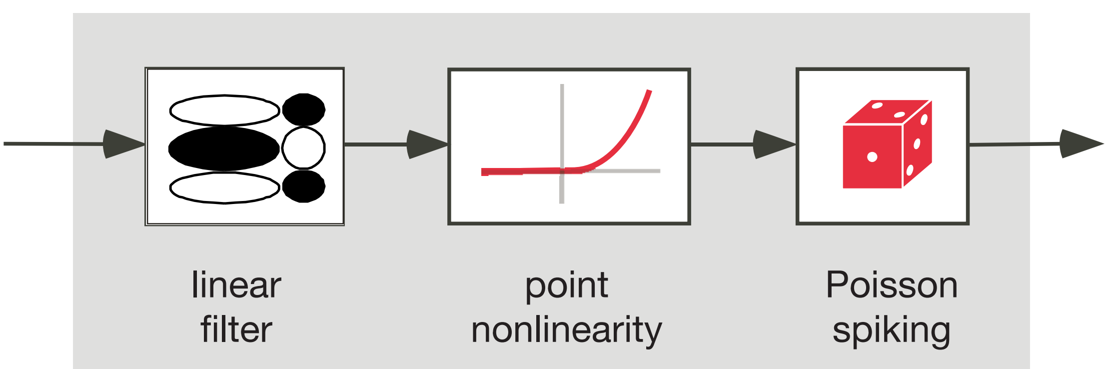


Example: LN cascade model



- Threshold-like nonlinearity => linear classifier
- Classic model for Artificial Neural Networks
    - McCullough & Pitts (1943), Rosenblatt (1957), etc
- No spikes (output is firing rate)

LNP cascade model



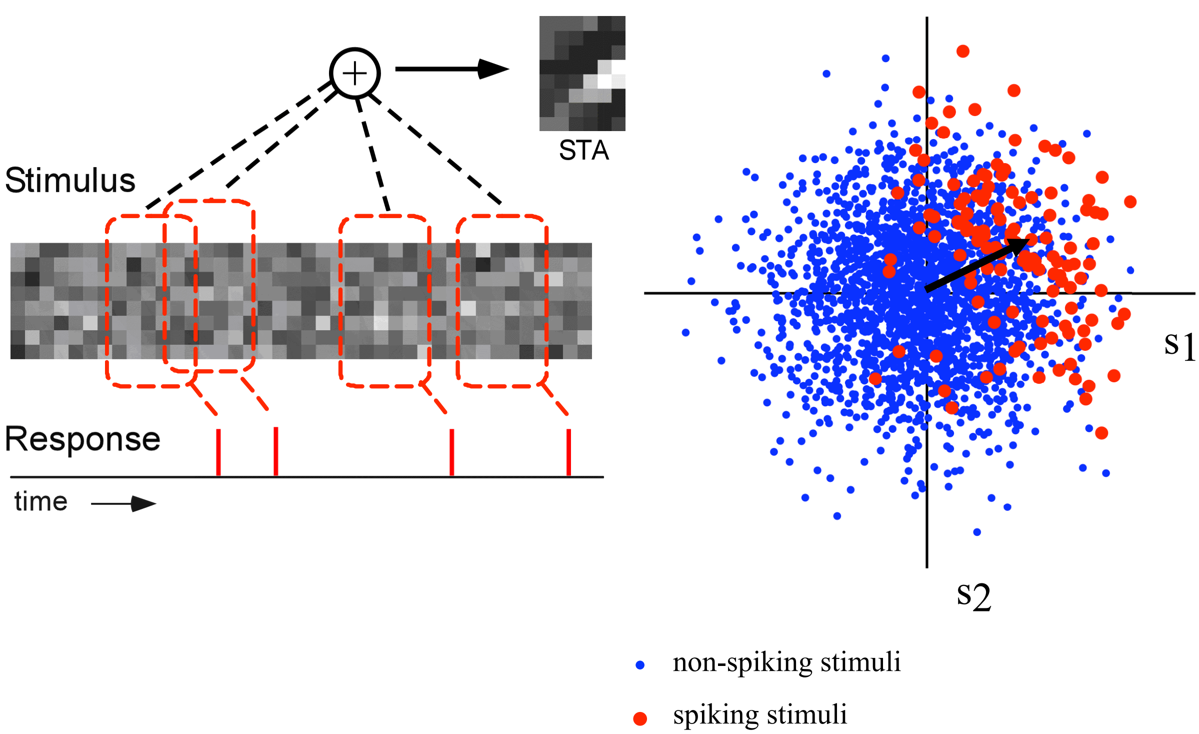
- Simplest descriptive spiking model
- Easily fit to (extracellular) data
- Descriptive, and interpretable (although *not* mechanistic)

Simple LNP fitting

- Assuming:
  - stochastic stimuli, spherically distributed
  - average of spike-triggered ensemble (STA) is shifted from that of raw ensemble
- The STA gives an **unbiased** estimate of w (for any f).
- For exponential f, this is the ML estimate!

- Bussgang 52; de Boer & Kuyper 68

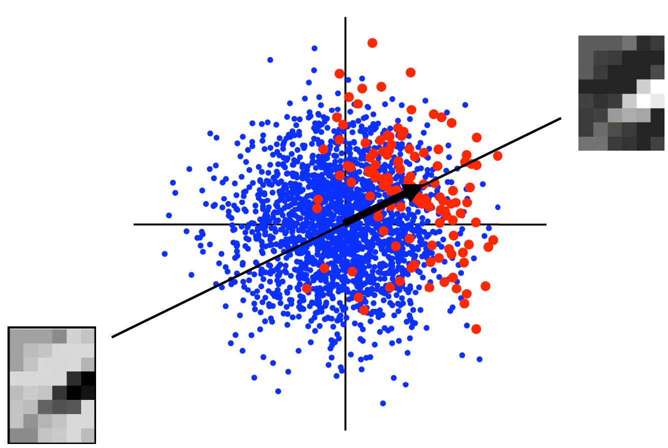
Computing the STA



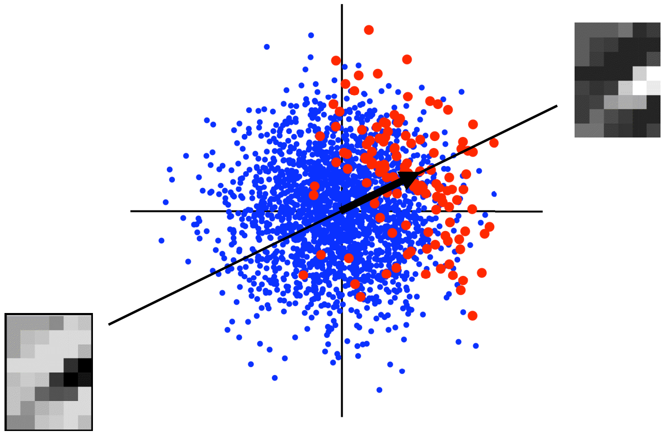
The STA provides an unbiased estimate of the linear filter in an LNP model

[geometric proof on board]

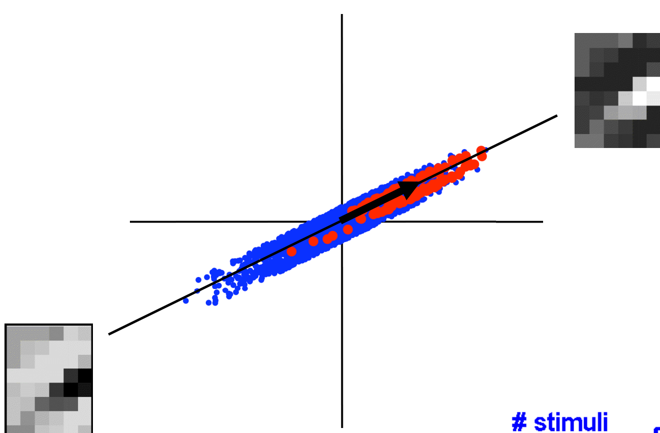
STA corresponds to a “direction” in stimulus space



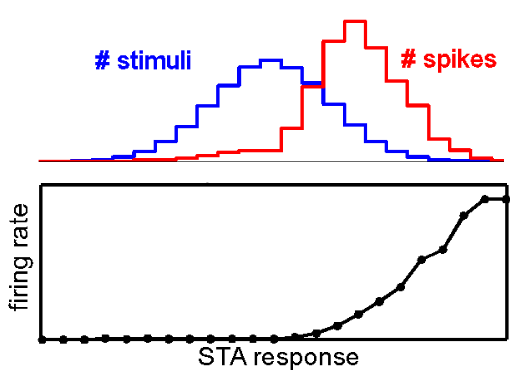
Projecting onto the STA



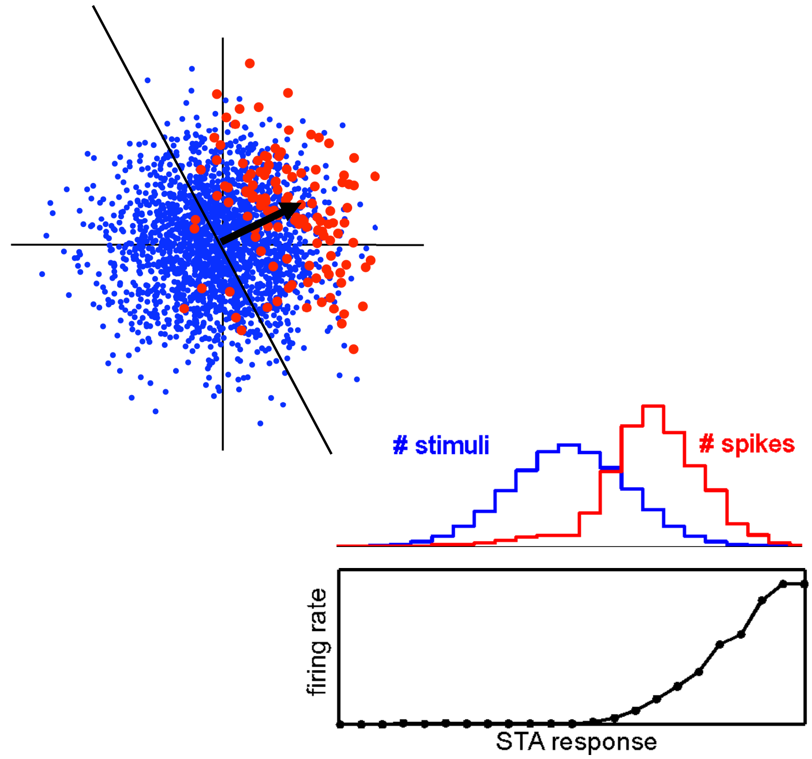
Solving for nonlinearity nonparametrically



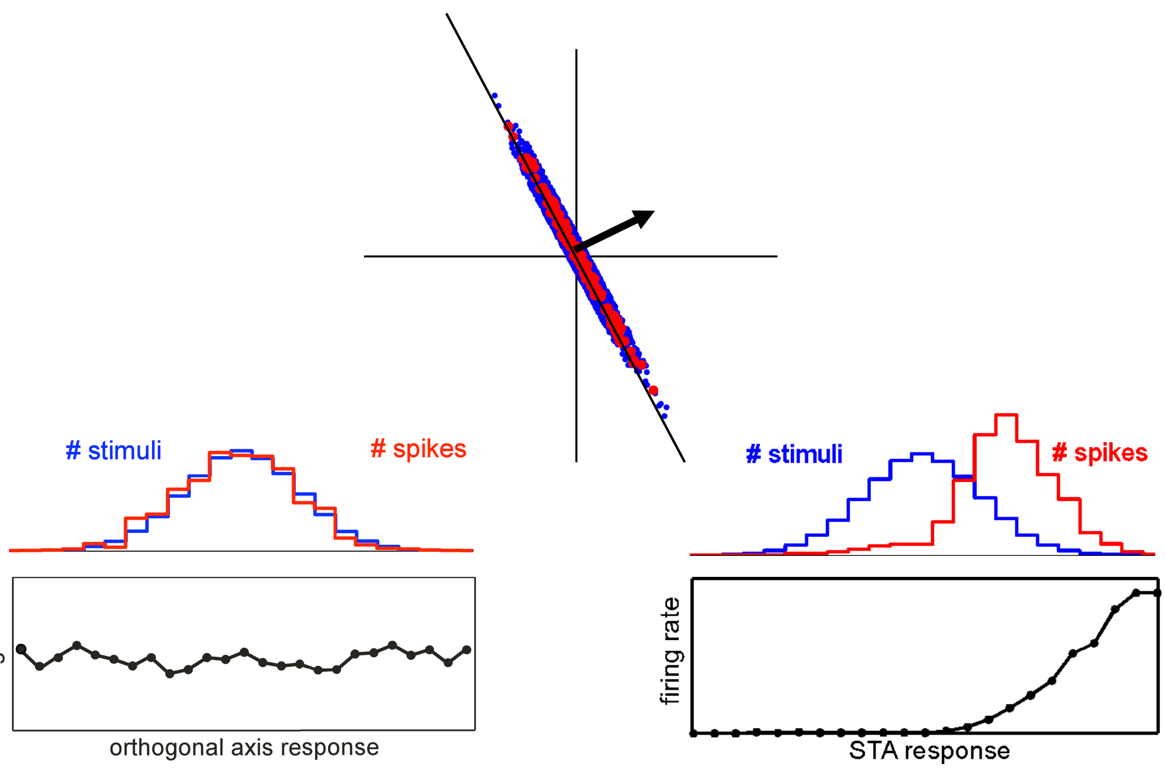
$$P(\text{spike}|\vec{s}) = \frac{P(\vec{s}\ |\text{spike})P(\text{spike})}{P(\vec{s})}$$



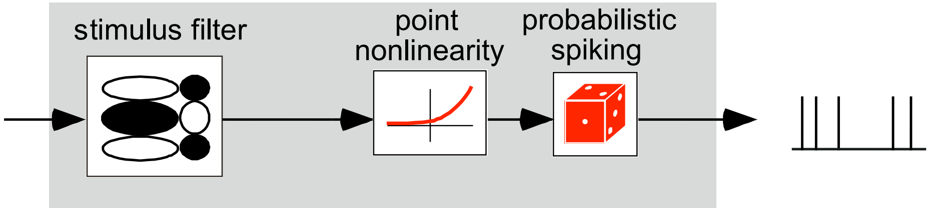
Projecting onto an axis orthogonal to the STA

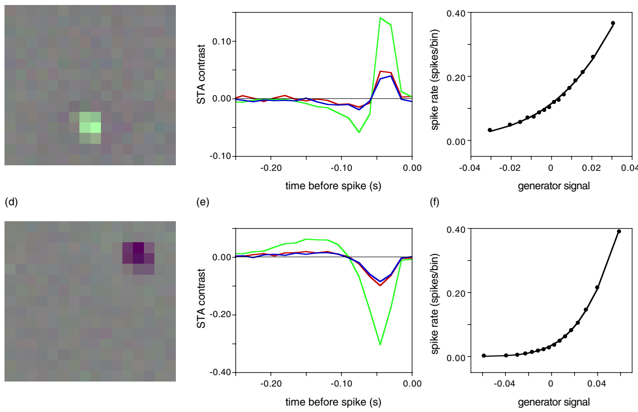


Projecting onto an axis orthogonal to the STA

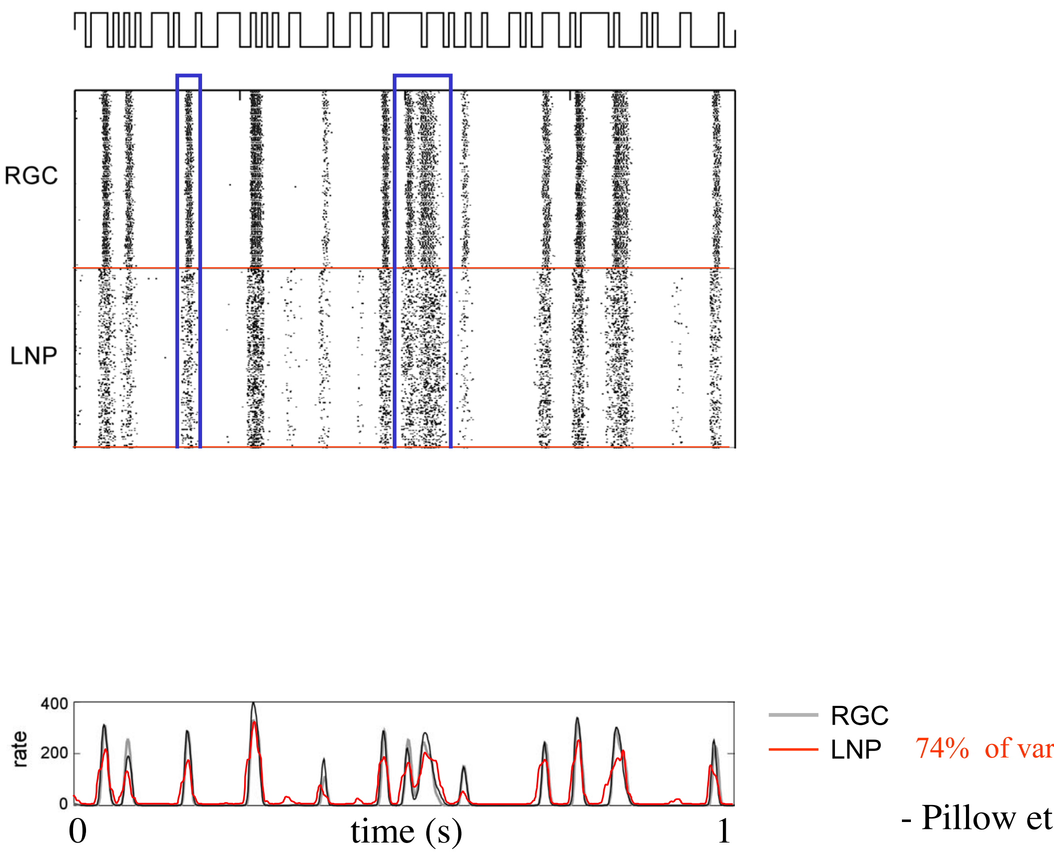


LNP model



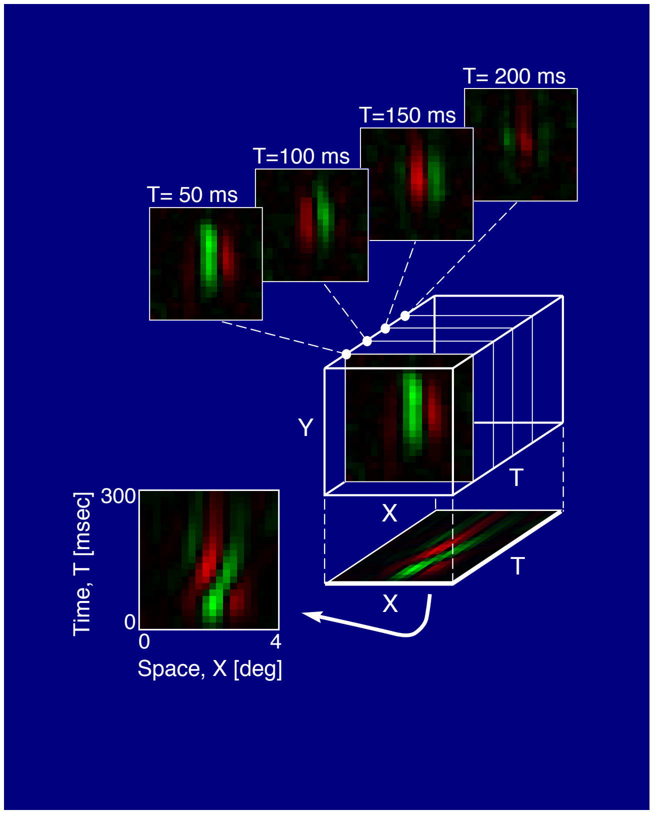


[Chichilnisky & Kalmer, 2002]

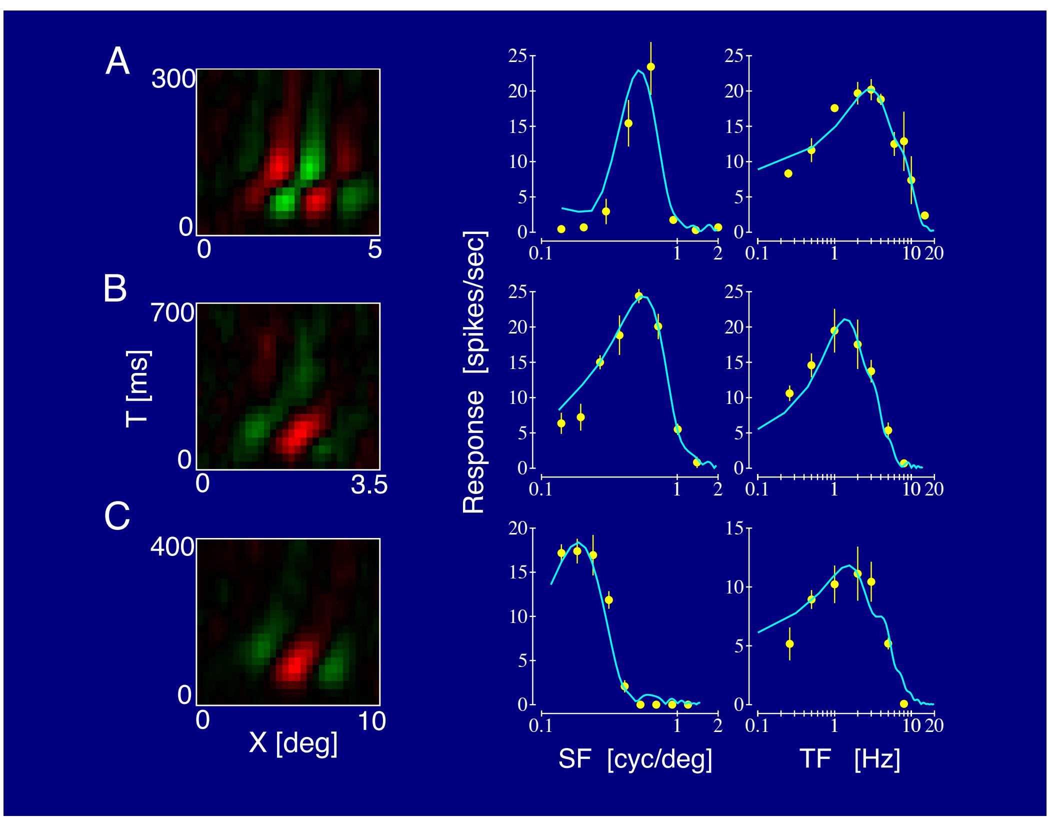


- Pillow etal, 2004

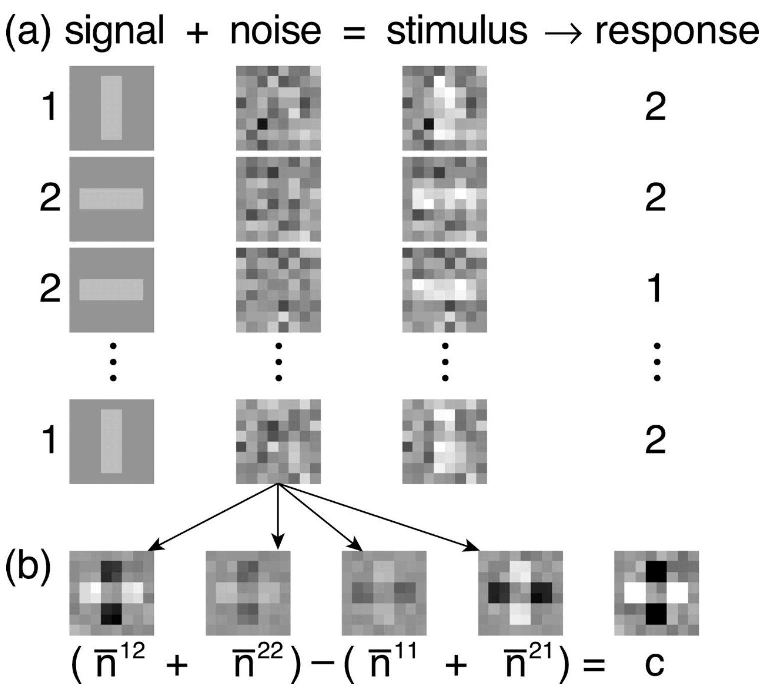
V1
simple
cell



- Ozhawa, etal

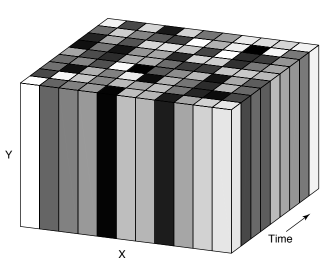


The standard method of calculating a classification image.

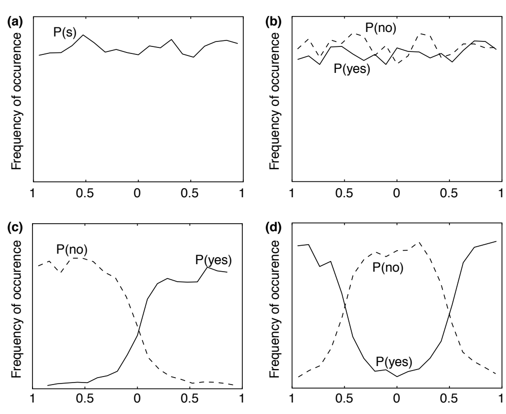


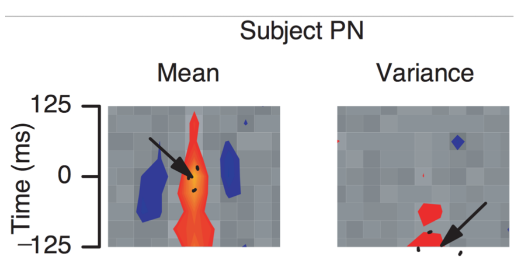
Murray R F J Vis 2011;11:2

Stimuli: 11x9 movie of bars, uniform random intensities



Task:
Decide if center bar of middle frame is brighter or darker than the mean





Simulation:
a) raw stim distribution
b) cond. dist. for irrelevant bar
c) cond. dist. for linear response model
d) cond. dist. for quadratic (contrast) response

[Neri & Heeger, 2002]

ML estimation of LNP

If $f_\theta(\vec{k} \cdot \vec{x})$ is convex (in argument and theta),
and *log* $f_\theta(\vec{k} \cdot \vec{x})$ is concave,
the likelihood of the LNP model is convex
(for all data, $\{n(t), \vec{x}(t)\}$)

Examples: $e^{(\vec{k} \cdot \vec{x}(t))}$

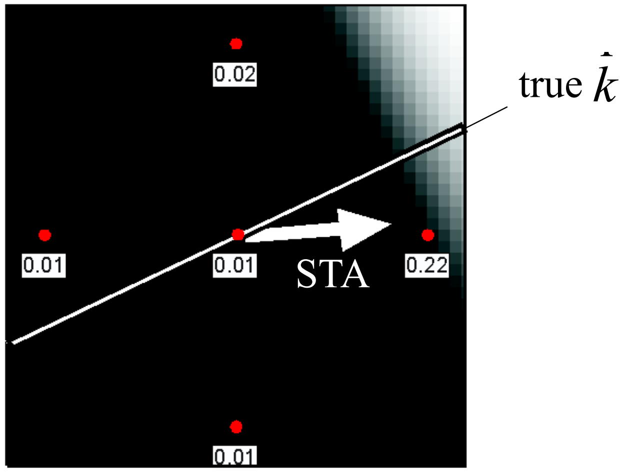
$(\vec{k} \cdot \vec{x}(t))^\alpha, \quad 1 < \alpha < 2$

[Paninski, '04]

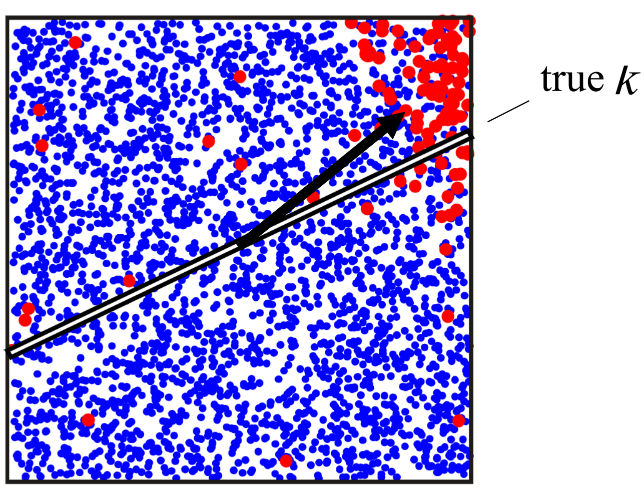
Sources of STA estimation error

- Non-spherical stimuli (can cause biases)
- Finite data (convergence goes as 1/N)
  [Paninski, '03]
- Model failures. Examples:
  - symmetric nonlinearity (causes no change in STE mean)
  - response not captured by 1D linear projection
  - spike history dependence (non-Poisson)

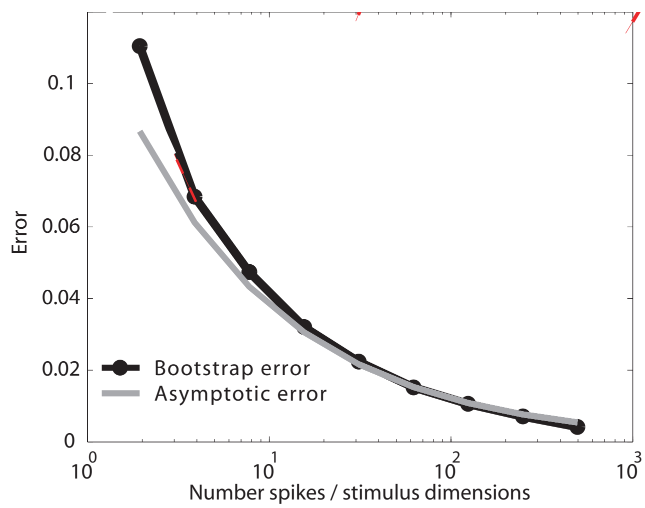
Example 1:
"sparse" noise



Example 2:
uniform noise



Variance behavior of STA



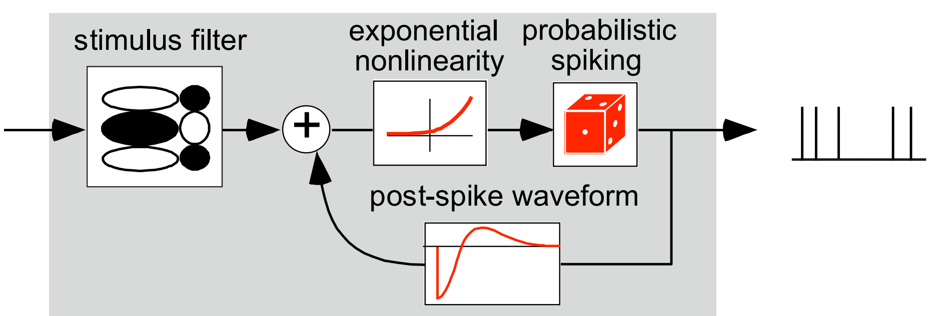
LNP summary

- LNP is the defacto standard descriptive model for early sensory neurons, and is implicit in much of the vision and audition literature
- Accounts for basic “receptive field” properties
- Accounts for basic spiking properties (rate code)
- Easily fit to data
- Easily interpreted
- BUT, non-mechanistic, and exhibits striking failures (esp. beyond early sensory/motor areas)

LNP limitations

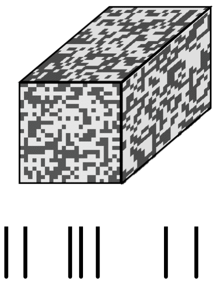
- Neural response depends on spike history
  => introduce spike history feedback
- Symmetric nonlinearities and/or multi-dimensional front-end (e.g., V1 complex cells)
  => spike-triggered covariance, subspace analyses
- White noise doesn't drive mid- to late-stage neurons well
  => build LNP on top of an "afferent" model

Recursive LNP



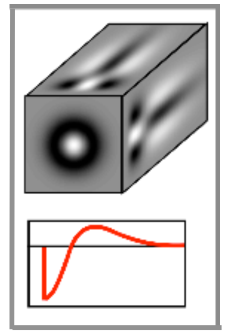
[Truccolo et al '05; Pillow et al '05]

stimulus &
spike train

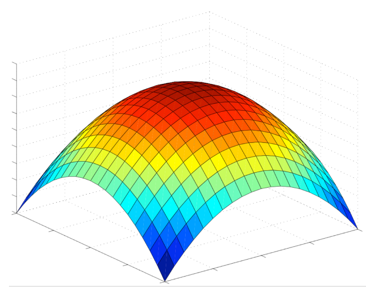


maximize
likelihood

model
parameters

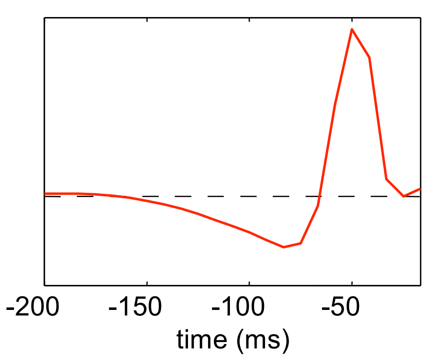


Critical: Likelihood function has no local maxima [Paninski 04]

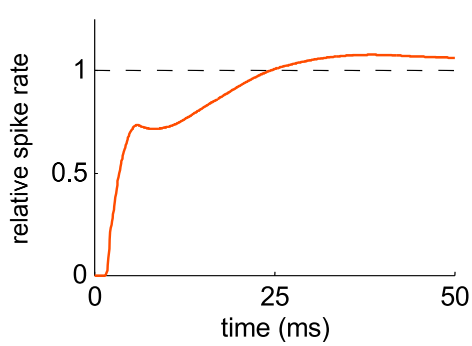


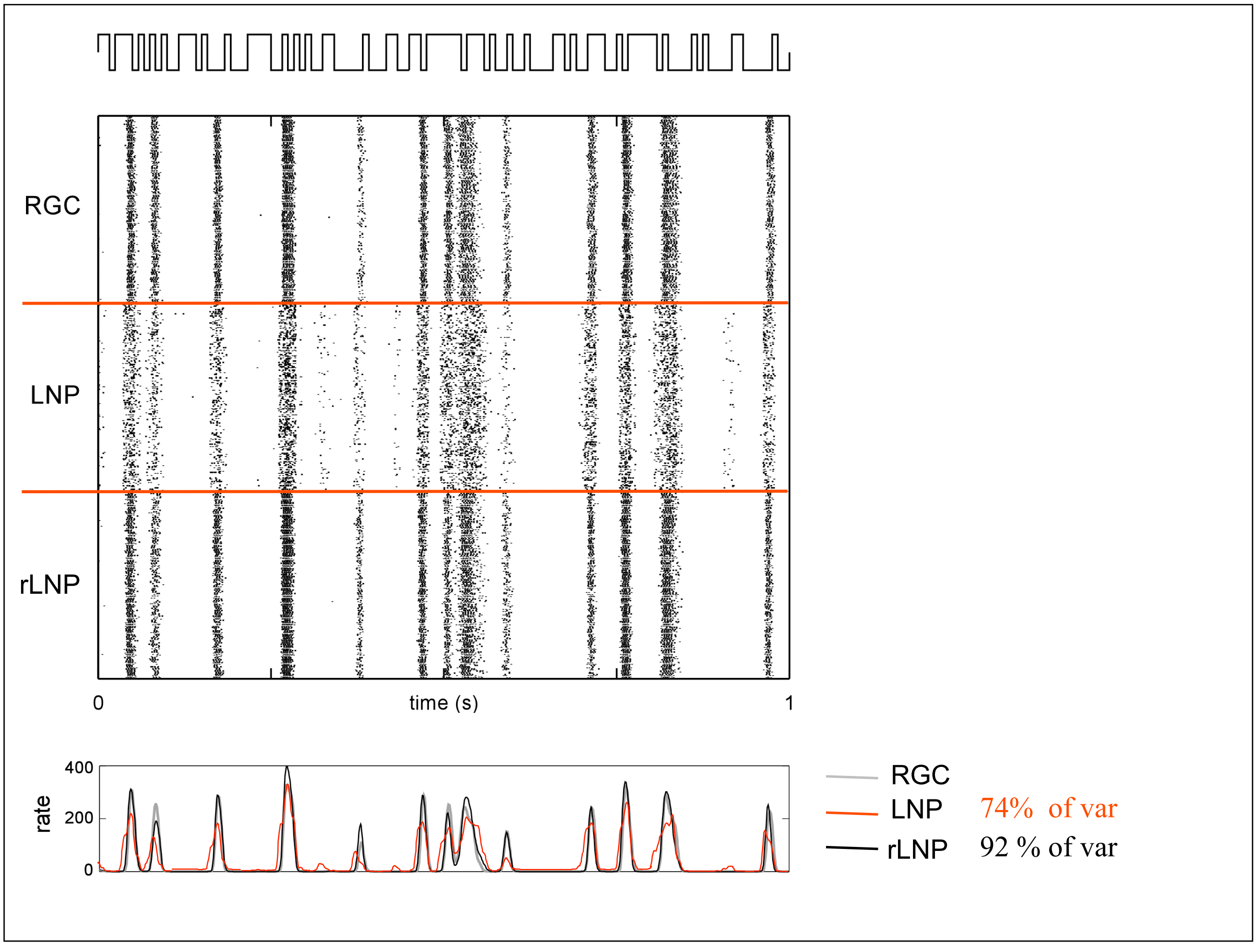
Example ON cell

temporal filter

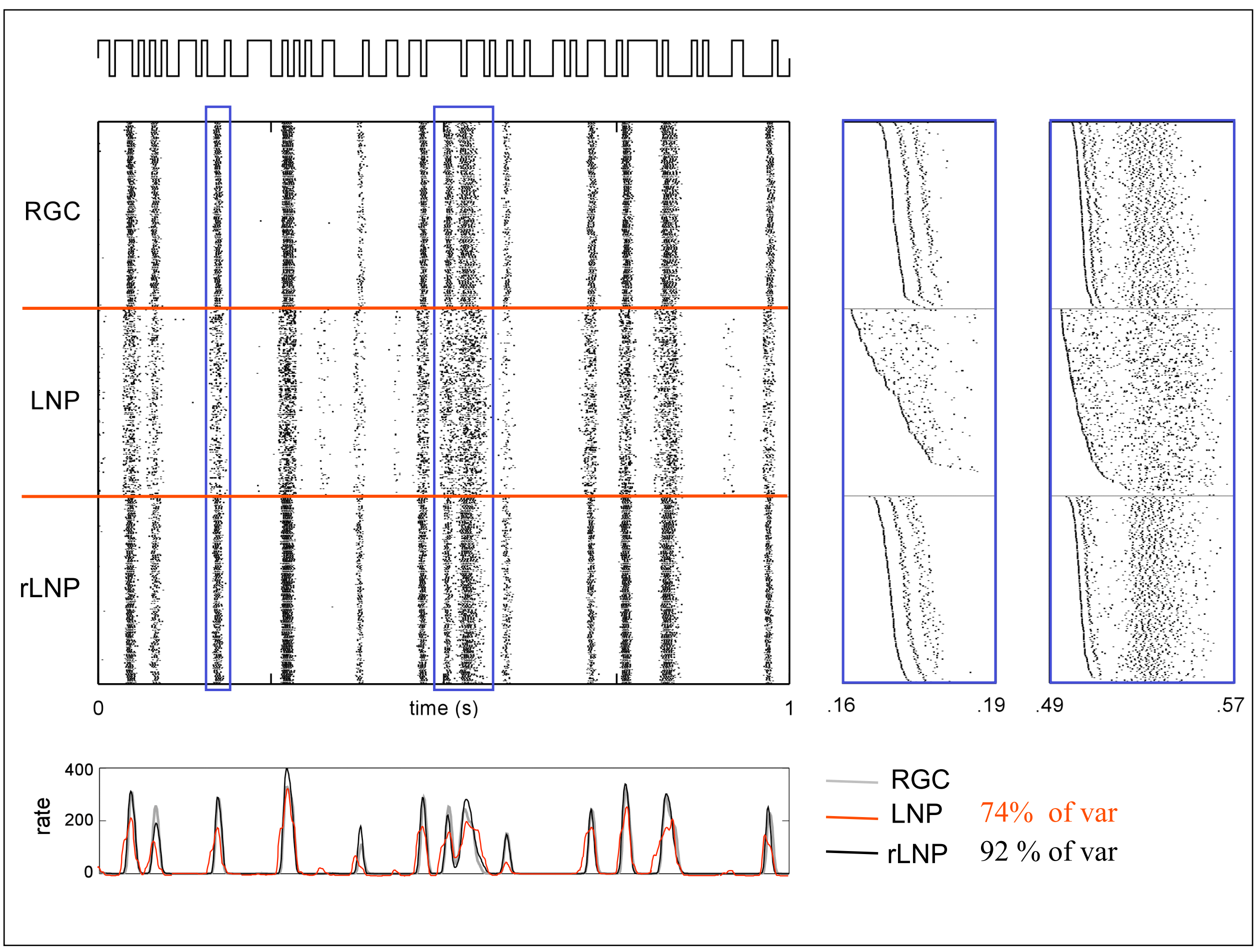


post-spike waveform

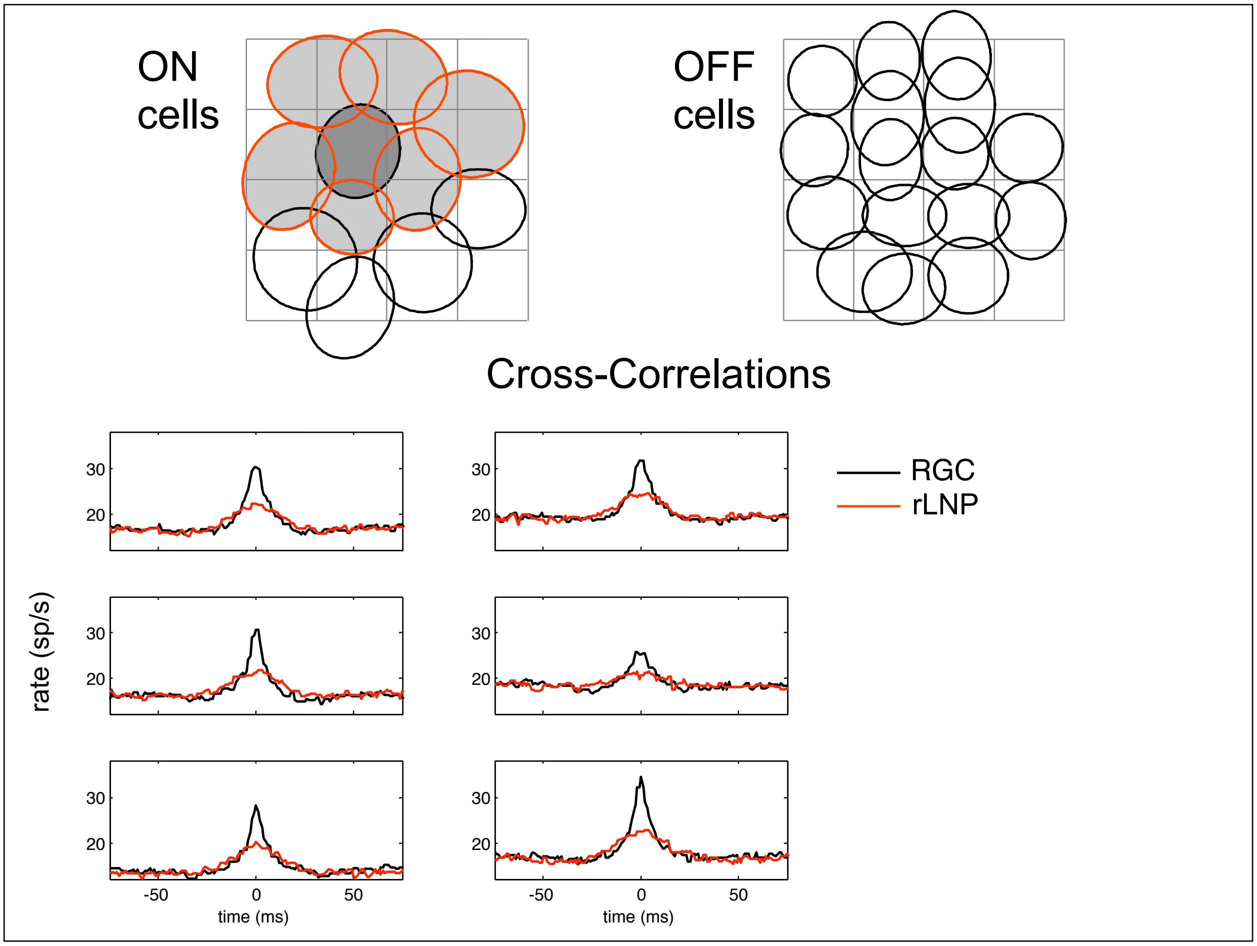


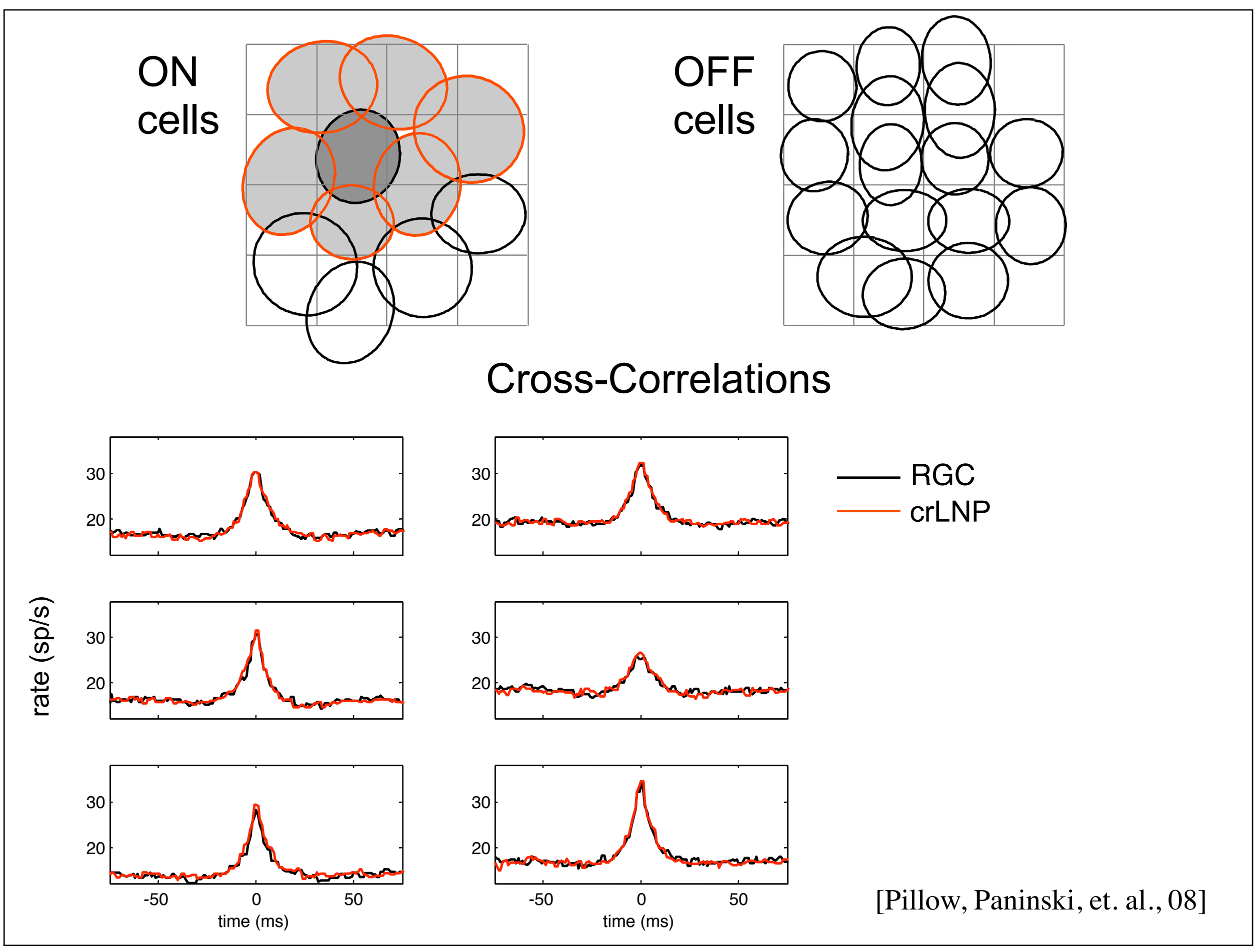


RGC
LNP
rLNP
0
time (s)
1
400
200
0
rate
RGC
LNP
74% of var
rLNP
92 % of var

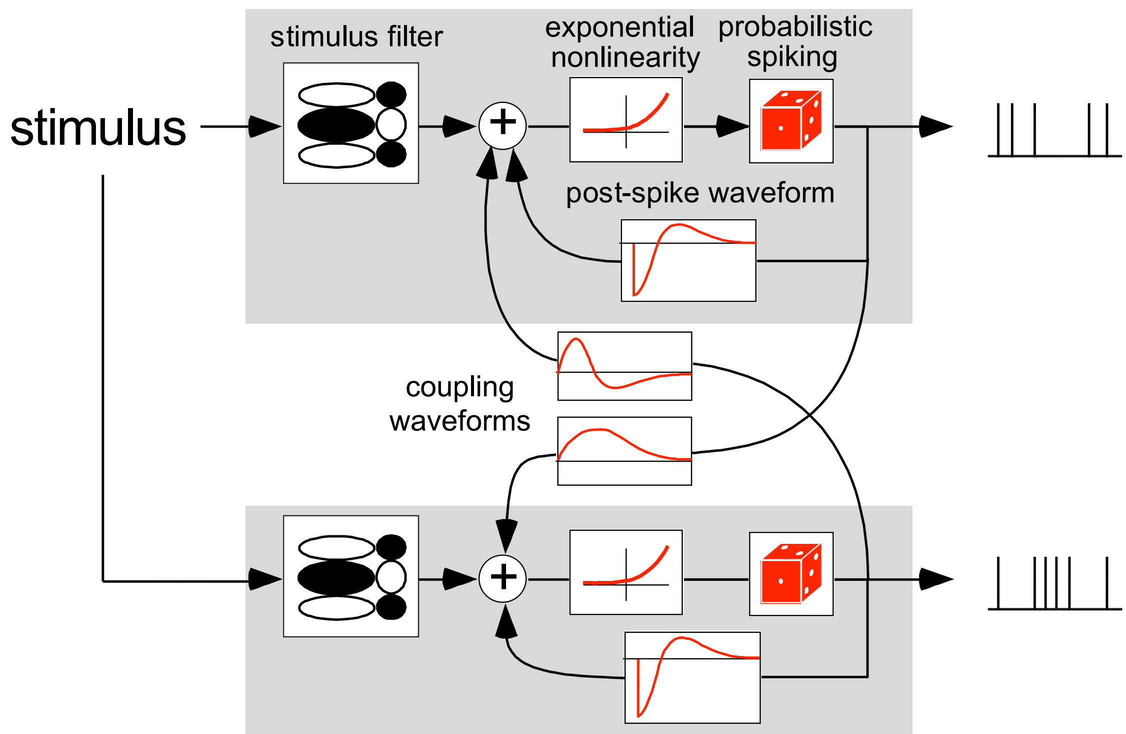


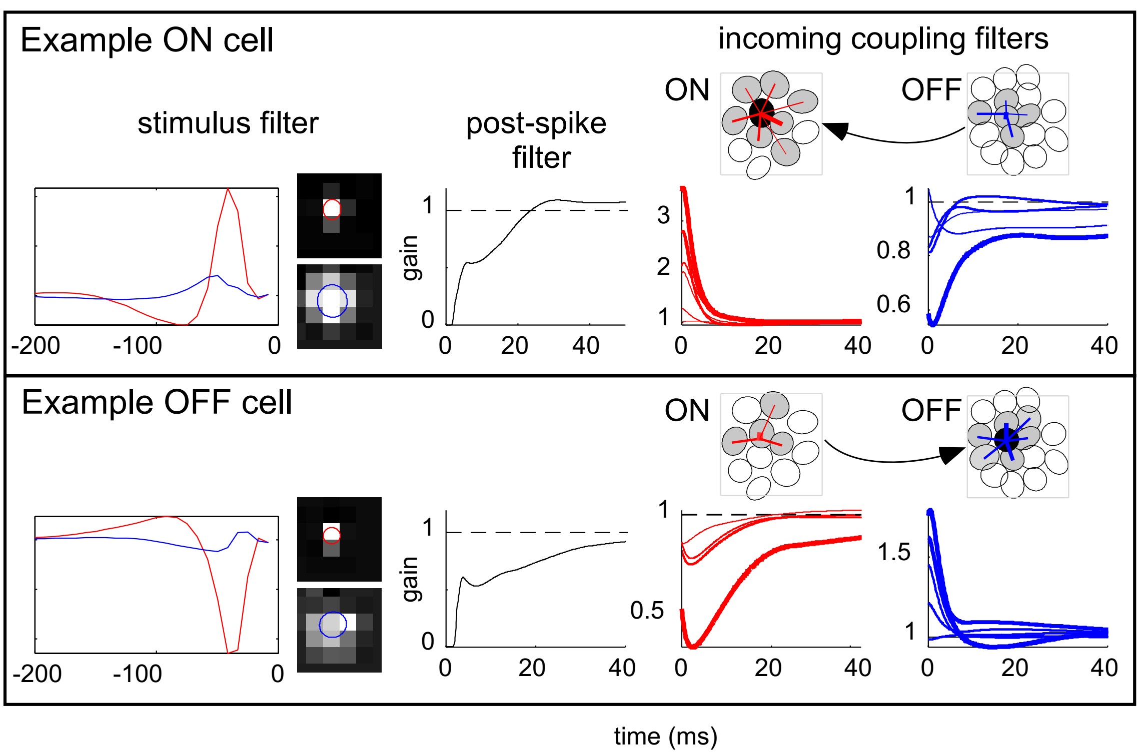
RGC
LNP
rLNP
0
time (s)
1
.16
.19
.49
.57
400
200
0
rate
RGC
LNP
74% of var
rLNP
92 % of var


ON
cells
OFF
cells
Cross-Correlations
rate (sp/s)
30
20
-50
0
50
time (ms)
RGC
rLNP

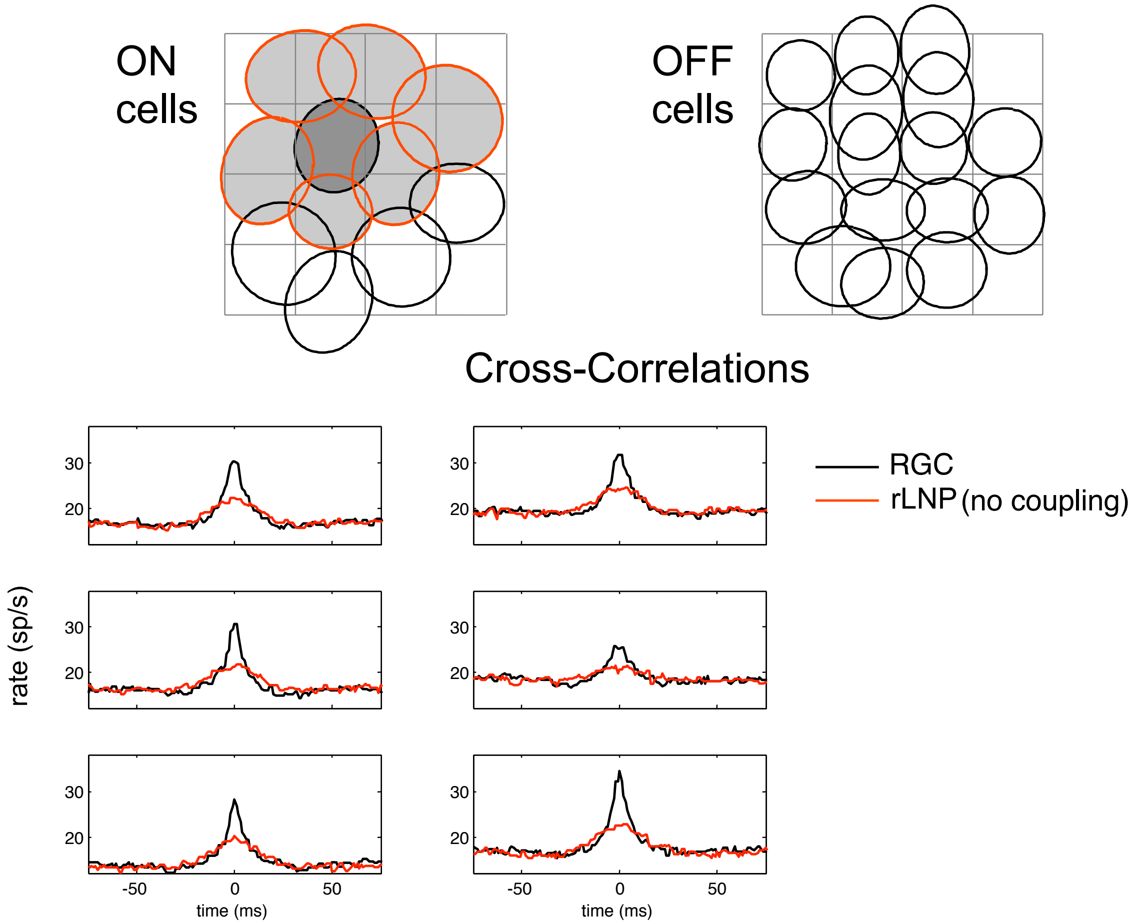

ON
cells
OFF
cells
Cross-Correlations
rate (sp/s)
30
20
-50
0
50
time (ms)
RGC
crLNP
[Pillow, Paninski, et. al., 08]

Coupled recursive LNP (crLNP)

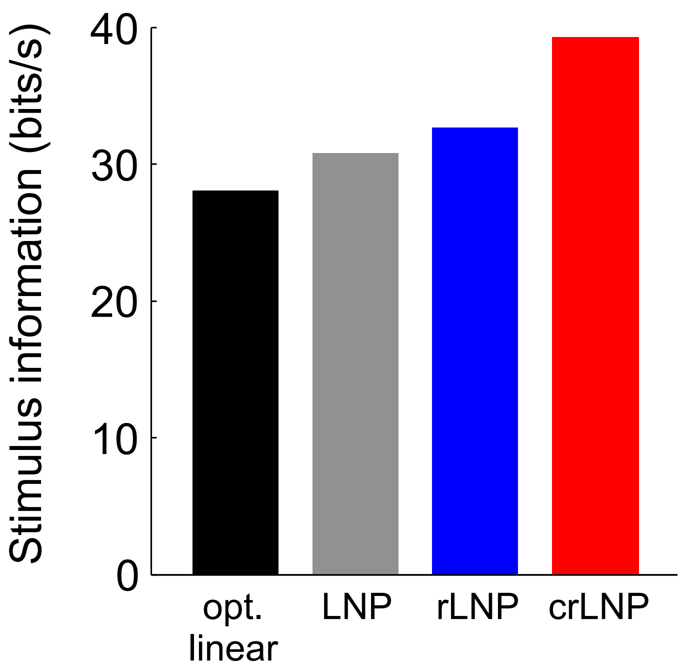




[Pillow, Paninski, et. al., 08]



Decoding



[Pillow, Paninski, et. al., 08]