Neural Encoding

Mark van Rossum

School of Informatics, University of Edinburgh

January 2015

Overview

Understanding the *neural code*

- Encoding: Prediction of neural response to a given stimulus
- Decoding (homunculus):
  - Given response, what was the stimulus?
  - Given firing pattern, what will be the motor output? (Important for prosthesis)
- Measuring information rates
- Books:
  [Rieke et al., 1996] (a very good book on these issues),
  [Dayan and Abbott, 2002] (chapters 2 and 3) [Schetzen, 2006],
  [Schetzen, 1981] (review paper on method)

The neural code

Understanding the *neural code* is like building a dictionary.

- Translate from outside world (sensory stimulus or motor action) to internal neural representation
- Translate from neural representation to outside world
- Like in real dictionaries, there are both one-to-many and many-to-one entries in the dictionary (think of examples)
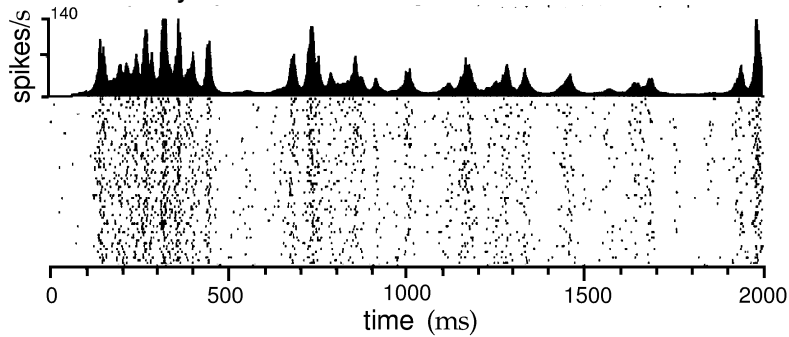
Encoding: Stimulus-response relation

Predict response *r* to stimulus *s*. Black box approach.
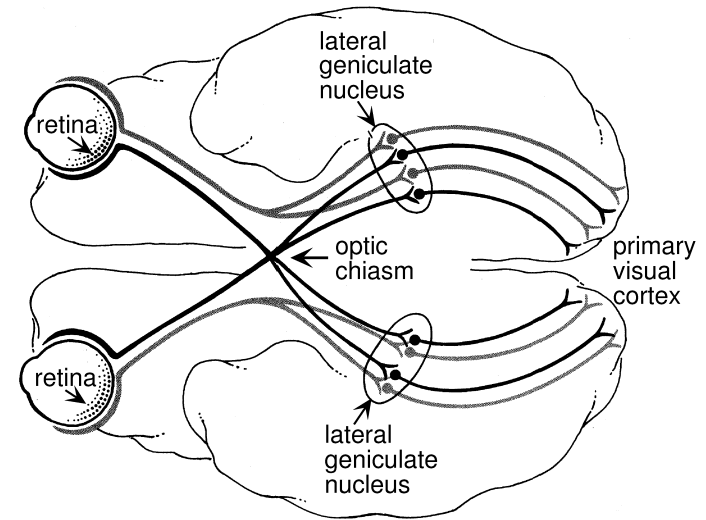This is a supervised learning problem, but:

- Stimulus *s* can be synaptic input or sensory stimulus.
- Responses are noisy and unreliable: Use probabilities.
- Typically many input (and sometimes output) dimensions
- Reponses are non-linear[1]
  - Assume non-linearity is weak. Make series expansion?
  - Or, impose a parametric non-linear model with few parameters
- Need to assume causality and stationarity (system remains the same). This excludes adaptation!

---

[1]Linear means: $r(\alpha s_1 + \beta s_2) = \alpha r(s_1) + \beta r(s_2)$ for all α, β.

Response consists of spikes. Spikes are (largely) stochastic. Compute rates by trial-to-trial average and hope that system is stationary and noise is really noise.
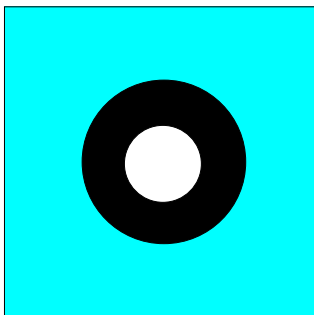

Initially, we try to predict r, rather than predict the spikes. (Note, there are methods to estimate most accurate histogram from data).
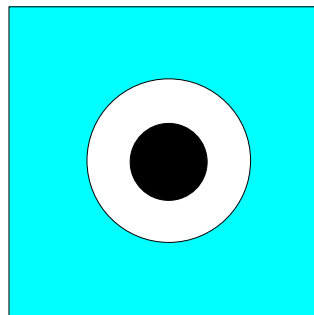

[Figure: Dayan and Abbott, 2001, after Nicholls et al, 1992]
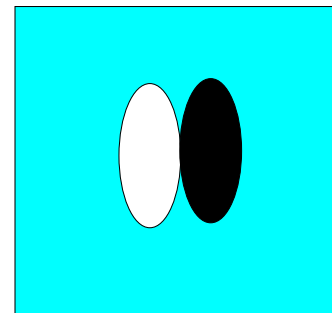
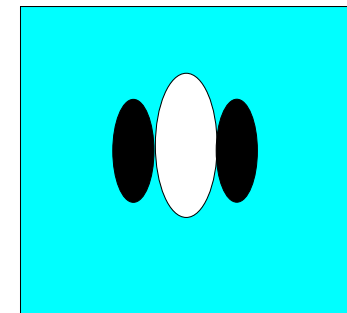On-centre off-surround   Off-centre on-surround

Also colour opponent cells

Odd   Even

- Simple cells, modelled by Gabor functions
- Also complex cells, and spatio-temporal receptive fields
- Higher areas
- Other pathways (e.g. auditory)

Not all cells are so simple...

The methods work well under limited conditions and for early sensory systems. But intermediate sensory areas (eg. IT) do things like face recognition. Very non-linear; hard with these methods.
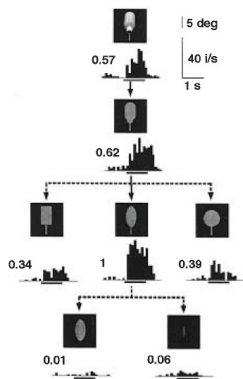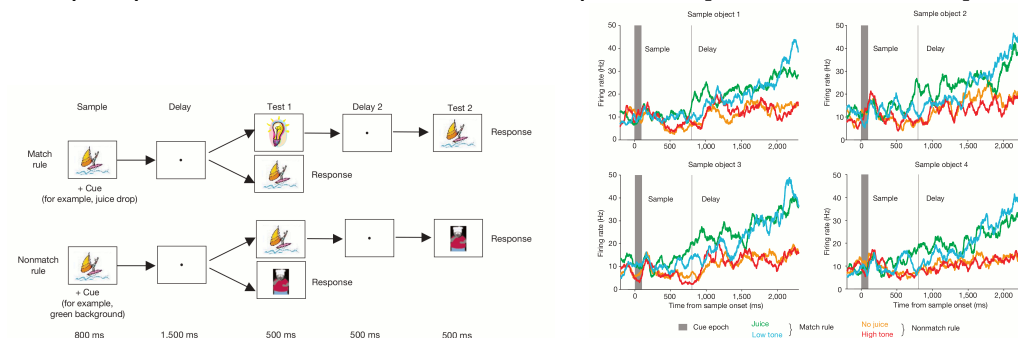

Figure 4.2
Example of the reduction process to determine the feature critical for the activation of individual cells. The responses were averaged over ten repetitions of the stimuli. Horizontal bars below histograms indicate the duration of stimulus presentation; numbers to the left, the magnitude of the maximum response. This cell was recorded from TE. (From Tanaka 1996.)

Not all cells are so simple...

In even higher areas the receptive field (RF) is not purely sensory. Example: pre-frontal cells that are task dependent [Wallis et al., 2001]
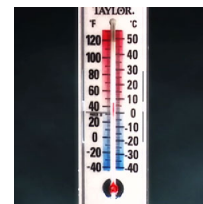
Overview

- Volterra and Wiener expansions
- Spike-triggered average & covariance
- Linear-nonlinear-Poisson (LNP) models
- Integrate & fire and Generalized Linear models
- Networks

Simple example

A thermometer: temperature T gives response $r = g(T)$, r measures cm mercury

- $g(T)$ is monotonic, $g^{-1}(r)$ probably exists
- Could be somewhat non-linear
- Could in principle be noisy.
- Will not indicate instantaneous T, but its recent history. For example $r(t) = g\left(\int dt' T(t') k(t - t')\right)$
- k is called a (filter) kernel. The argument of $g()$ is a convolution $T \star k \equiv \int dt' T(t') k(t - t')$
- Note, if $k(t) = \delta(t)$ then $r(t) = g(T(t))$

Volterra Kernels

Inspiration from Taylor expansion:
$r(s) = r(0) + r'(0)s + \frac{1}{2}r''(0)s^2 + \ldots = r_0 + r_1 s + \frac{1}{2}r_2 s^2 + \ldots$
But include temporal response (Taylor series with memory)

$$
\begin{aligned}
r(t) \;=\; & h_0 \\
& + \int_0^\infty d\tau_1 h_1(\tau_1) s(t - \tau_1) \\
& + \int_0^\infty \int_0^\infty d\tau_1 d\tau_2 \quad h_2(\tau_1, \tau_2) s(t - \tau_1) s(t - \tau_2) \\
& + \ldots h_3(\tau_1, \tau_2, \tau_3) \ldots
\end{aligned}
$$

Note, $h_2(\tau_1, \tau_2) = h_2(\tau_2, \tau_1)$
Hope that

- $\lim_{\tau_i \to \infty} h_k(\tau_j) = 0$
- h_k is smooth,
- h_k small for large k.

Noise and power spectra

- At each timestep draw an independent sample from a zero-mean Gaussian
  $\langle s(t_1) \rangle = 0$, $\langle s(t_1) \ldots s(t_{2k+1}) \rangle = 0$
  $\langle s(t_1) s(t_2) \rangle = C(t_1 - t_2) = \sigma^2 \delta(t_1 - t_2)$,
  $\langle s(t_1) s(t_2) s(t_3) s(t_4) \rangle =$
  $\sigma^4 [\delta(t_1 - t_2)\delta(t_3 - t_4) + \delta(t_1 - t_3)\delta(t_2 - t_4) + \delta(t_1 - t_4)\delta(t_2 - t_3)]$
- The noise is called *white* because in the Fourier domain all frequencies are equally strong.
- The powerspectrum of the signal and the autocorrelation are directly related via Wiener-Kinchin theorem.

$$
S(f) = 4 \int_0^\infty C(\tau) \cos(2\pi f \tau)
$$

Wiener Kernels

Wiener kernels are a rearrangement of the Volterra expansion, used when $s(t)$ is Gaussian white noise with $\langle s(t_1) s(t_2) \rangle = \sigma^2 \delta(t_1 - t_2)$
0th and 1st order Wiener kernels are identical to Volterra

$$
\begin{aligned}
r(t) \;=\; & g_0 \\
& + \int_0^\infty d\tau_1 g_1(\tau_1) s(t - \tau_1) \\
& + \left[\int_0^\infty \int_0^\infty d\tau_1 d\tau_2 g_2(\tau_1, \tau_2) s(t - \tau_1) s(t - \tau_2) \right. \\
& \left. - \sigma^2 \int_0^\infty d\tau_1 g_2(\tau_1, \tau_1) \right] + \ldots
\end{aligned} \tag{1}
$$

Estimating Wiener Kernels

To find kernels, stimulate with Gaussian white noise

$$
\begin{aligned}
g_0 &= \langle r \rangle \\
g_1(\tau) &= \frac{1}{\sigma^2} \langle r(t) s(t - \tau) \rangle \qquad \text{(correlation)} \\
g_2(\tau_1, \tau_2) &= \frac{1}{2\sigma^4} \langle r(t) s(t - \tau_1) s(t - \tau_2) \rangle \; (\tau_1 \neq \tau_2)
\end{aligned}
$$

- In Wiener, but not Volterra, expansion successive terms are independent. Including a quadratic term won't affect the estimation of the linear term, etc.
- Technical point [Schetzen, 1981] : Lower terms do enter in higher order correlations, e.g.
  $\langle r(t) s(t - \tau_1) s(t - \tau_2) \rangle = 2\sigma^4 g_2(\tau_1, \tau_2) + \sigma^2 g_0 \delta(\tau_1 - \tau_2)$
- The predicted rate is given by Eq.(1).

Model:

$$r(n\Delta t) = g_0 + \sum_{i=0}^{L-1} g_{1i}s((n-i)\Delta t) + \ldots$$

- The predicted rate can be <0.
- In biology, unlike physics, there is no obvious small parameter that justifies neglecting higher orders. Check the accuracy of the approximation post hoc.

Averaging and ergodicity

- $\langle x \rangle$ formally means an average over many realizations over the random variables of the system (both stimuli and internal state). This definition is good to remember when conceptual problems occur.
- An ergodic system visits all realizations if one waits long enough. That means one can measure from a system repeatedly and get the true average.

- In discrete time this is just linear/polynomial regression
- Solve e.g. to minimize squared error, $E = (\mathbf{r} - \mathbf{Sg})^{\mathsf{T}}(\mathbf{r} - \mathbf{Sg})$.
- E.g. $L = 3$, $\mathbf{g} = (g_0, g_{10}, g_{11}, g_{12})^T$ and $\mathbf{r} = (r_1, r_2, \ldots, r_n)^T$

$$S = \begin{pmatrix} 1 & s_1 & s_0 & s_{-1} \\ 1 & s_2 & s_1 & s_0 \\ \vdots & & & \vdots \\ 1 & s_n & s_{n-1} & s_{n-2} \end{pmatrix}$$

S is a $n \times (1 + L)$ matrix ('design matrix')

- The least-squares solution $\hat{\mathbf{g}}$ for any stimulus (differentiate E wrt. $\mathbf{g}$):

$$\hat{\mathbf{g}} = (S^T S)^{-1} S^T \mathbf{r}$$

- Note that on average for Gaussian noise $\langle S^T S \rangle_{ij} = n\delta_{ij}(\sigma^2 + (1 - \sigma^2)\delta_{i1})$
- After substitution we obtain

$$\hat{g}_0 = \frac{1}{n}\sum_{i=1}^{n} r_i = \langle r \rangle$$

$$\hat{g}_{1j} = \frac{1}{\sigma^2}\frac{1}{n}\sum_{i=1}^{n} s_{i-j}r_i = \frac{1}{\sigma^2}corr(s, r)$$

- Note parallel with continuous time equations.

Convolution becomes simple multiplication in Fourier domain
Assume the neuron is purely linear ($g_j = 0$, $j > 1$),
otherwise Fourier representation is not helpful

- $r(t) = r_0 + s * g_1$
- $\langle s(t)r(t + \tau) \rangle = \langle sr_0 \rangle + \langle s(t)g_1 * s \rangle$
- Now $g_1(\omega) = \frac{\langle rs \rangle(\omega)}{\langle ss \rangle(\omega)}$

For Gaussian white noise $\langle ss \rangle(\omega) = \sigma^2$ (note, that $\langle s \rangle = 0$)

- So $g_1(\omega) = \frac{1}{\sigma^2}\langle rs \rangle(\omega)$
- g_1 can be interpreted as *impedance* of the system
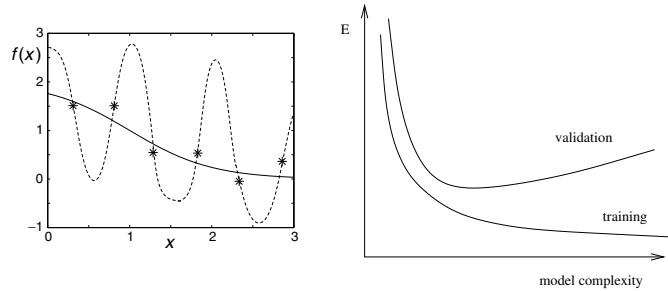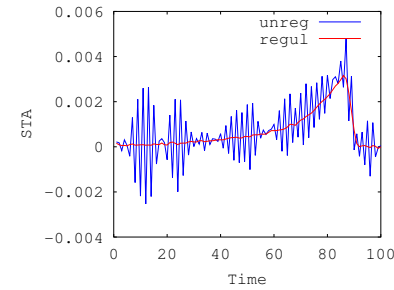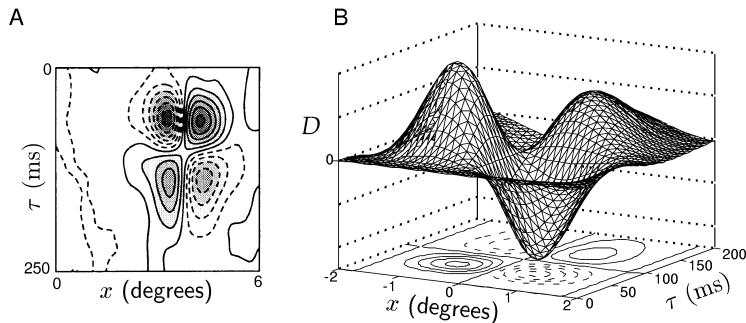
Regularization

Figure: Over-fitting: Left: The stars are the data points. Although the dashed line might fit the data better, it is over-fitted. It is likely to perform worse on new data. Instead the solid line appears a more reasonable model. Right: When you over-fit, the error on the training data decreases, but the error on new data increases. Ideally *both* errors are minimal.

Regularization

- Fits with many parameters typically require regularization to prevent over-fitting
- Regularization: punish fluctuations (smooth prior)
- Non-white stimulus, Fourier: $g_1(\omega) = \frac{\langle rs \rangle(\omega)}{\langle ss \rangle(\omega) + \lambda}$ (prevent division by zero as $\omega \to \infty$)
- In time-domain: $\hat{\mathbf{g}} = (S^T S + \lambda I)^{-1} S^T \mathbf{r}$
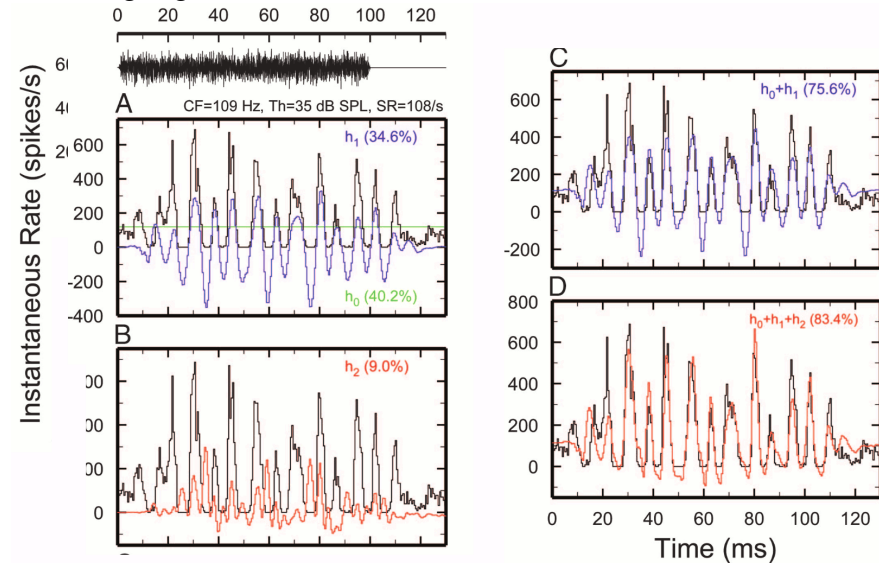- Set λ by hand

Spatio-temporal kernels

[Dayan and Abbott, 2002]

- Kernel can also be in spatio-temporal domain.
- This V1 kernel does not respond to static stimulus, but will respond to a moving grating ([Dayan and Abbott, 2002]§2.4 for more motion detectors)

Higher-order kernels

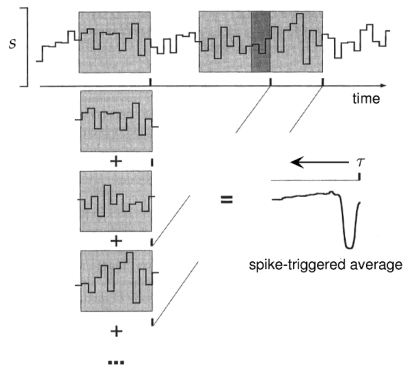Including higher orders leads to more accurate estimates.

Chinchilla auditory system [Temchin et al., 2005]

Wiener Kernels for spiking neurons: Spike triggered average (STA)

Spike times t_i, $r(t) = \sum \delta(t - t_i)$
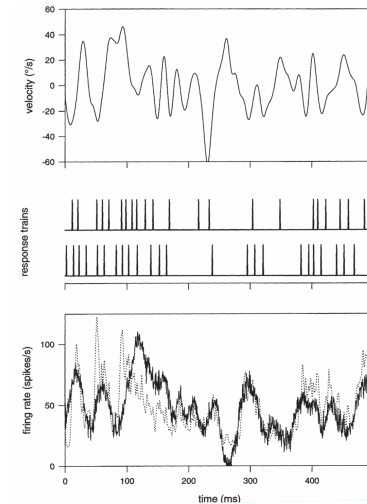
$g_1(\tau) = \frac{1}{\sigma^2} \langle r(t) s(t - \tau) \rangle = \frac{1}{\sigma^2} \sum_{t_i} s(t_i - \tau)$

spike-triggered average

For *linear* systems the most effective stimulus of a given power ($\int dt\, s(t)^2 = c$) is $s_{opt}(t) \propto g_1(-t)$
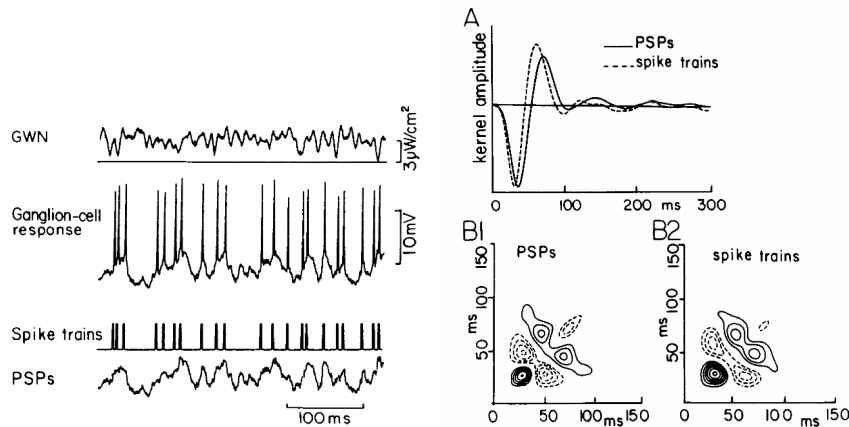
Wiener Kernels for spiking neurons

Application on H1 neuron [Rieke et al., 1996]. Prediction (solid), and actual firing rate (dashed). Prediction captures the slow modulations, but not faster structure. This is often the case.

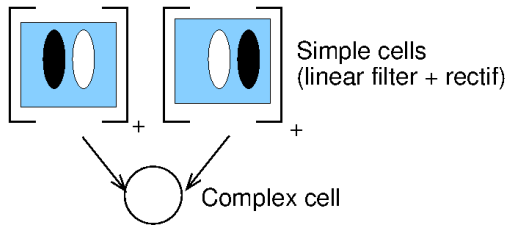# Wiener Kernels: Membrane potential vs spikes

Not much difference; spike triggered is sharper (transient detector).
Retinal ganglion cell [Sakai, 1992]

Spike triggered covariance STC

$r(t) = p(spike|\mathbf{s}) = g_0 + \mathbf{g}_1^T \mathbf{s} + \mathbf{s}^T G_2 \mathbf{s} + \dots$

- Stimulate with white noise to get spike-triggered covariance G_2 (STC)
  $G_2(\tau_1, \tau_2) \propto \sum_{t_i} s(t_i - \tau_1) s(t_i - \tau_2)$
- G_2 is a symmetric matrix, so its eigenvectors $\mathbf{e_i}$ form an orthogonal basis, $G_2 = \sum_{i=1}^{N} \lambda_i \mathbf{e}_i \mathbf{e}_i^T$.
  $r(t) = g_0 + \mathbf{g}_1^T \mathbf{s} + \sum_{i=1}^{N} \lambda_i (\mathbf{s}^T \mathbf{e}_i)^2 + \dots$
- To simplify, look for largest eigenvalues of G_2, as they will explain the largest fraction of variance in $r(t)$ (cf PCA).

Simple cells
(linear filter + rectif)

+   +

Complex cell

Red dots indicate spikes,
x-axis luminance of bar
1, y-axis luminance of
bar 2.

For complex cells $g_1 = 0$ (XOR in disguise).
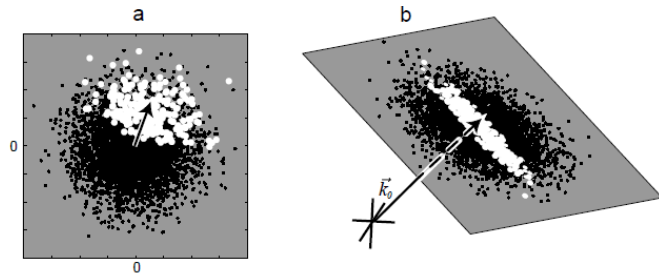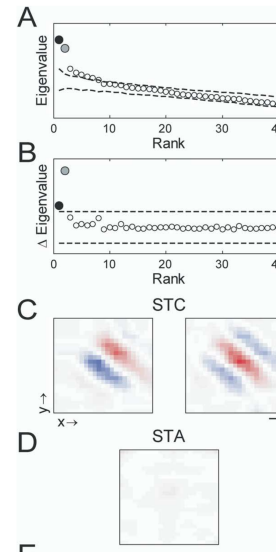First non-zero term is G_2.

Complex cell [Touryan et al., 2005]

Figure 1: Geometric depiction of spike-triggered analyses. **a**, Spike-triggered averaging with two-dimensional stimuli. Black points indicate raw stimuli. White points indicate stimuli eliciting a spike, and the STA (black vector), which provides an estimate of $\vec{k}_0$, corresponds to their center of mass. **b**, Spike-triggered covariance analysis of suppressive axes. Shown are a set of stimuli lying on a plane perpendicular to the excitatory kernel, $\vec{k}_0$. Within the plane, stimuli eliciting a spike are concentrated in an elliptical region. The minor axis of the ellipse corresponds to a suppressive stimulus direction: stimuli with a significant component along this axis are less likely to elicit spikes. The stimulus component along the major axis of the ellipse has no influence on spiking.

[Schwartz et al., 2001]

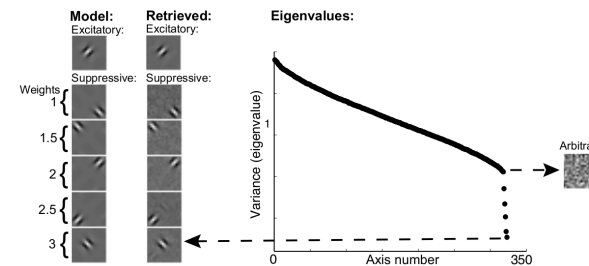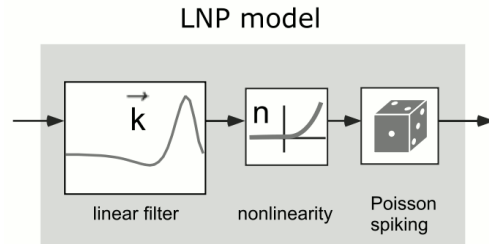Also lowest variance eigenvalues might be indicative, namely of
suppression

Figure 2: Estimation of kernels from a simulated model (equation 2). Left: Model kernels. Right: Sorted eigenvalues of covariance matrix of stimuli eliciting spikes (STC). Five eigenvalues fall significantly below the others. Middle: STA (excitatory kernel) and eigenvectors (suppressive kernels) associated with the lowest eigenvalues.

[Schwartz et al., 2001] (simulated neuron)
See [Chen et al., 2007] for real data.

LNP models

An alternative to including higher and higher orders, is to impose a
linear-nonlinear-Poisson (LNP) model.

LNP model

[Pillow et al., 2005]

$r(t) = n(\int dt' s(t') k(t - t'))$

- Linear: spatial and temporal filter kernel **k**
- Non-linear function giving output spike probability:
  halfwave rectification, saturation, ...
- Poisson spikes $p_{spike}(t) = \lambda(t)$ (quite noisy)

Data on Poisson variability

To test for Poisson with varying rate (inhomogeneous Poisson
process), measure variance across trials.

V1 monkey [Bair, 1999]

Fly motion cell
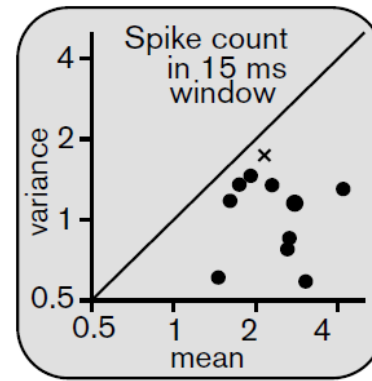[de Ruyter van Steveninck et al., 1997]

Estimation of LNP models

[Chichilnisky, 2001]:

- If $n()$ were linear, we would just use STA
- It turns out we can still use the STA if $p(\mathbf{s})$ is radially symmetric, e.g. Gaussian
- Let f_t denote the firing probability at time t given stimulus $\mathbf{s}_t$. Do STA:
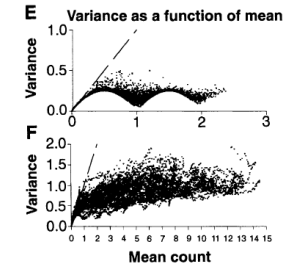
$$
\tilde{\mathbf{k}} = \frac{\sum_{t=1}^{T} \mathbf{s}_t f_t}{\sum_{t=1}^{T} f_t} = \frac{\frac{1}{T}\sum_{t=1}^{T} \mathbf{s}_t f_t}{\frac{1}{T}\sum_{t=1}^{T} f_t}
$$
$$
= \frac{1}{\langle f \rangle} \sum_{\mathbf{s}} \mathbf{s} p(\mathbf{s}) n(\mathbf{k} \cdot \mathbf{s})
$$

- For every **s** there is a $\mathbf{s}^*$ of equal probability with a location symmetric about **k**.

$$
\tilde{\mathbf{k}} = \frac{1}{2\langle f \rangle} \sum_{\mathbf{s}, \mathbf{s}^*} [\mathbf{s} p(\mathbf{s}) n(\mathbf{k} \cdot \mathbf{s}) + \mathbf{s}^* p(\mathbf{s}^*) n(\mathbf{k} \cdot \mathbf{s}^*)]
$$
$$
= \frac{1}{2\langle f \rangle} \sum_{\mathbf{s}, \mathbf{s}^*} (\mathbf{s} + \mathbf{s}^*) n(\mathbf{k} \cdot \mathbf{s}) p(\mathbf{s})
$$

use that, $\mathbf{k} \cdot \mathbf{s} = \mathbf{k} \cdot \mathbf{s}^*$, and $\mathbf{s} + \mathbf{s}^* \propto \mathbf{k}$. Hence

$$
\tilde{\mathbf{k}} \propto \mathbf{k}
$$

Continuous case: Bussgang's Theorem

[Bussgang, 1952; Dayan and Abbott (2002) p 83-84]
$s \sim N(\mathbf{0}, \sigma^2 I)$

$$
\begin{aligned}
\tilde{k}_i &= \frac{1}{\sigma^2} \int d\mathbf{s}\, p(\mathbf{s}) s_i n(\mathbf{k}.\mathbf{s}) \\
&= \frac{1}{\sigma^2} \prod_{j \neq i} \int ds_j P(s_j) \int ds_i \frac{s_i}{\sqrt{2\pi\sigma^2}} e^{-s_i^2/2\sigma^2} n(\mathbf{k}.\mathbf{s}) \\
&= \prod_{j \neq i} \int ds_j P(s_j) \int_{-\infty}^{\infty} ds_i \frac{1}{\sqrt{2\pi\sigma^2}} e^{-s_i^2/2\sigma^2} \frac{d}{ds_i} n(\mathbf{k}.\mathbf{s}) \\
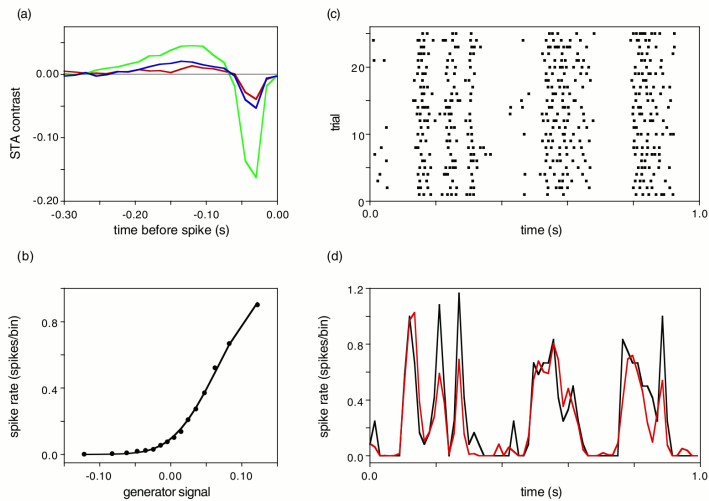&= k_i \int d\mathbf{s}\, p(\mathbf{s}) n'(\mathbf{k}.\mathbf{s}) = const\ k_i
\end{aligned}
$$

'const' same for all i, so $\tilde{\mathbf{k}} \propto \mathbf{k}$

Estimation of LNP models

- It is easy to estimate $n()$ once $\mathbf{k}$ is known
- Unlike linear case, STA does not minimize error anymore
- Can extend analysis from circularly to elliptically symmetric $p(\mathbf{s})$ [Paninski]

LNP results

[Chichilnisky, 2001]
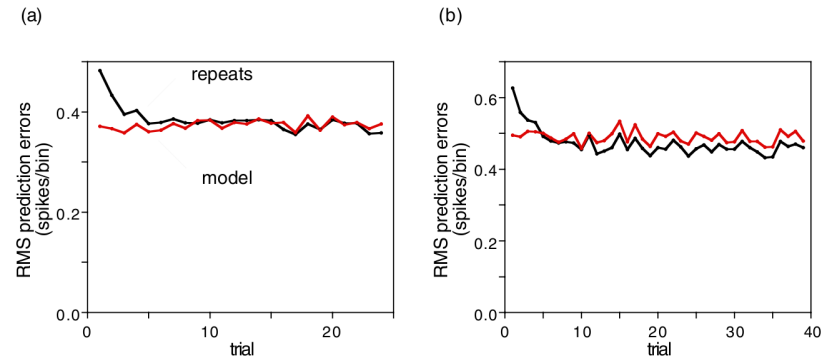Colors are the kernels for the different RGB channels

LNP results

Figure 6. (*a*) RMS error of predictions of spike counts over time as a function of number of repeated trials for the macaque retinal ganglion cell shown in figure 5. The red trace shows the RMS difference between observed spike counts and model predictions. The black trace shows the RMS difference between observed spike counts and predictions from repeated trials. (*b*) The same plot for a second macaque ganglion cell. $\Delta t = 15$ ms.
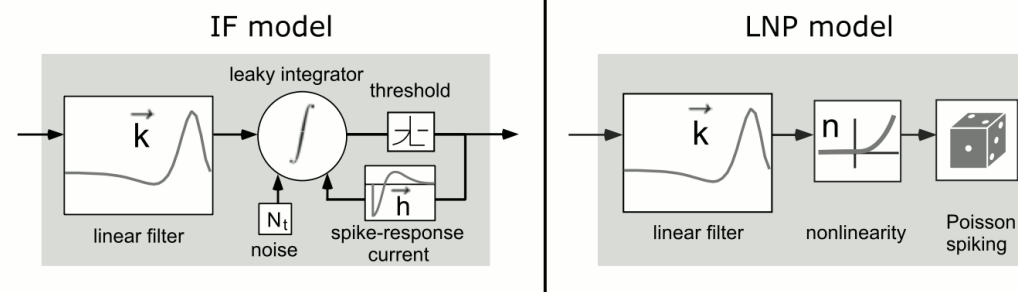
Good fit on coarse time-scale [Chichilnisky, 2001], but spike times are off(below).

- Above LNP model is for linear cell, e.g. for a simple cell
- For a complex cell, use two filters with squaring non-linearities, and add these before generating spikes
- In general (c.f. additive models in statistics)

$$r(t) = n(\sum_i f(\mathbf{k}_i \cdot \mathbf{s}))$$

[Pillow et al., 2005]

- Parameters are the **k** and **h** kernels
- **h** can include reset and refractoriness
- For standard I&F: $h(t) = \frac{1}{R}(V_T - V_{reset})\delta(t)$.
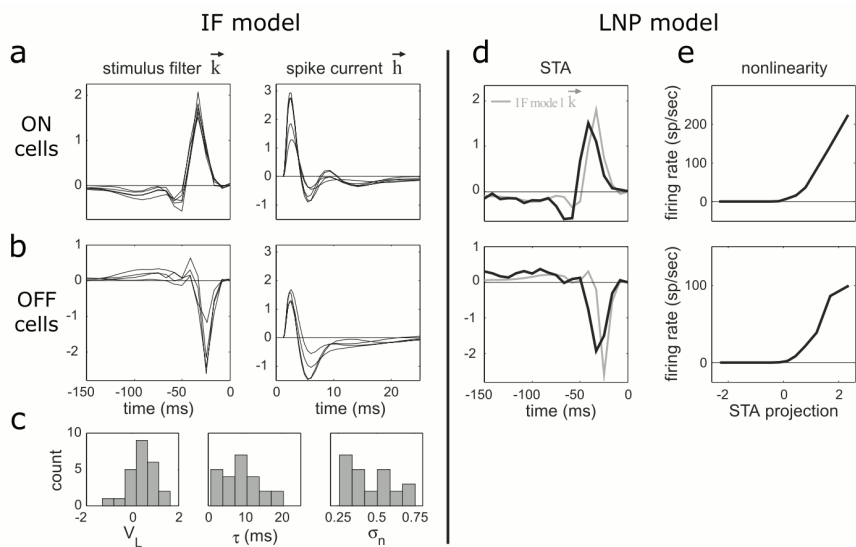
Figure 2. Parameters obtained from fits to RGC data for the IF model (*a–c*) and the LNP model (*d, e*). *a*, Filters $\vec{k}$ and spike–response currents $\vec{h}$ obtained for five ON cells in one retina. *b*, Corresponding filters for four OFF cells. *c*, Histograms of model scalar parameters and for all 24 cells in three retinas. *d*, Comparison of linear filters for the IF model (gray) and LNP model (black) for one ON cell (top) and one OFF cell (bottom). *e*, Measured LNP point nonlinearities for converting filter output to instantaneous spike rate.
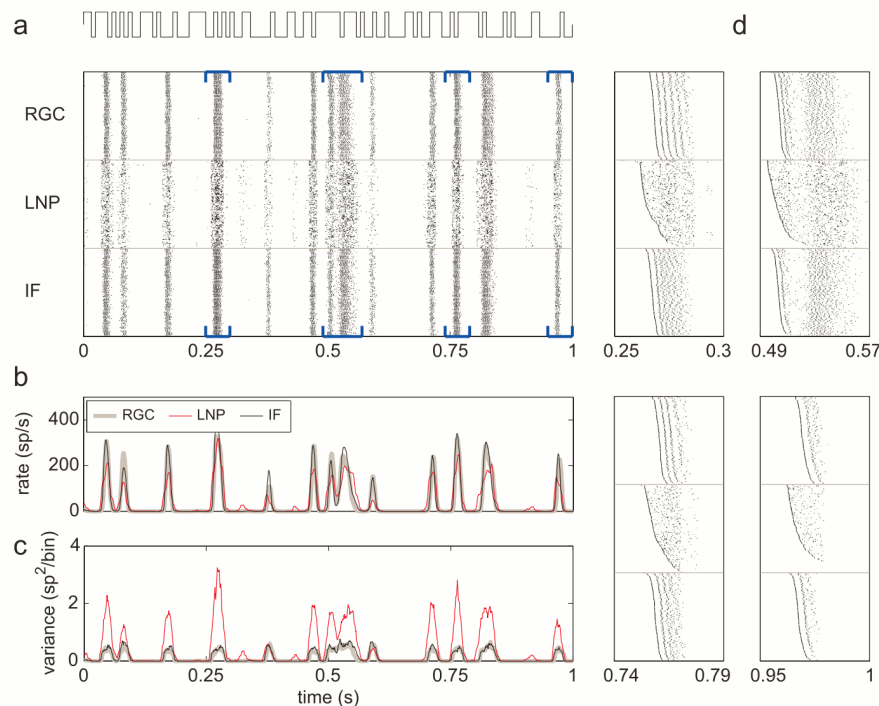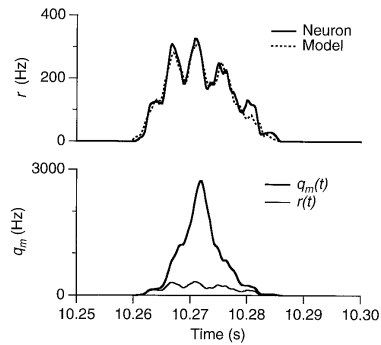
[Pillow et al., 2005] Fig 2

[Pillow et al., 2005] Fig 3

(Thick line in bottom graph: firing rate without refractoriness).
[Berry and Meister, 1998].
The reset after a spike reduces the rate and increase the precision of single spikes.
This substantially increases coded information per spike
[Butts et al., 2007].

Stimulate neuron and model as:

$$C\frac{dV(t)}{dt} = I_m(V, t) + I_{noise} + I_{in}(t) \tag{2}$$

Measure I_m using that

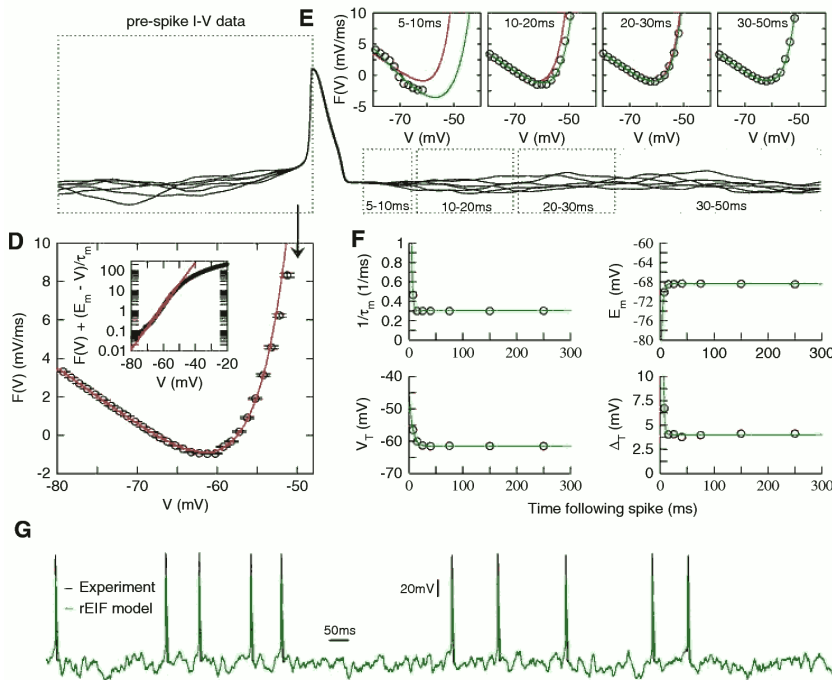$$I_m(V, t) + I_{noise} = -I_{in}(t) + C\frac{dV(t)}{dt} \tag{3}$$

Note, for linear I&F: $I_m(V, t) = \frac{1}{R}(V_{rest} - V)$
But, more realistically:
- non-linearities from spike generation (Na & K channels)
- after-spike changes (adaptation)
Used in a competition to predict spikes generated with current injection
(easier than a sensory stimulus).

[Badel et al., 2008]

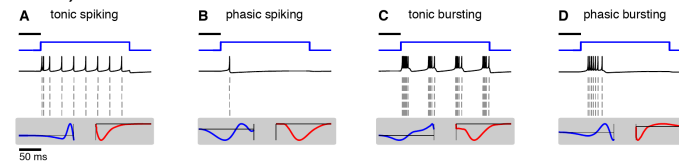## Generalized linear models

[Gerstner et al., 2014](Ch.10)

- Least square fit: $y = \mathbf{a}.\mathbf{x}$, Gaussian variables.
- IF (and LNP) fitting can be cast in GLMs, a commonly used statistical model $y = f(\mathbf{a}.\mathbf{x})$
- Note that $f()$ is a probability, hence not Gaussian distributed.
- Spike probability at any time is given by 'hazard function' $f(V_m)$ which includes spike feedback
  $f(V_m(t)) = f[(\mathbf{k} * s)(t) + \sum_i \mathbf{h}(t - t_i)]$.
- Can also include terms proportional to s^k. It is the linearity in the parameters that matters.
  (cf polynomial fitting, $y = \sum_{k=0}^{K} a_k x^k$)

Generalized linear models

- Log likelihood for small bins (t_i: observed spikes)
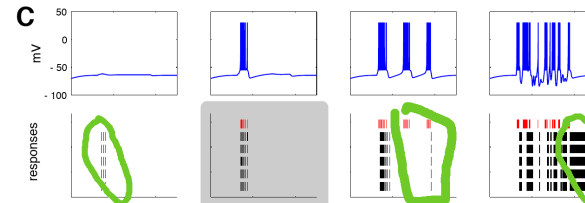
$$\log \mathcal{L} = \log \prod_t P_{spiketrain}(t)$$

$$= \sum_{t_i} \log f(V_m(t_i)) + \sum_{t \neq t_i} \log[1 - f(V_m(t))dt]$$

$$= \sum_{t_i} \log f(V_m(t_i)) - \int f(V_m(t))dt$$

- When f is convex and $\log(f)$ is concave in parameters, e.g. $f(x) = [x]_+$, or $f(x) = \exp(x)$ then $\log \mathcal{L}$ is concave, hence global max (easy fit).
- Regularization is typically required.

Generalized linear models

Can also capture bursting neurons (blue: input filter, red: feedback filter).

But generalization can fail badly (green):

From [Weber and Pillow, 2017].

Even more complicated models

A retina + ganglion cell model with multiple adaptation stages [van Hateren et al., 2002]
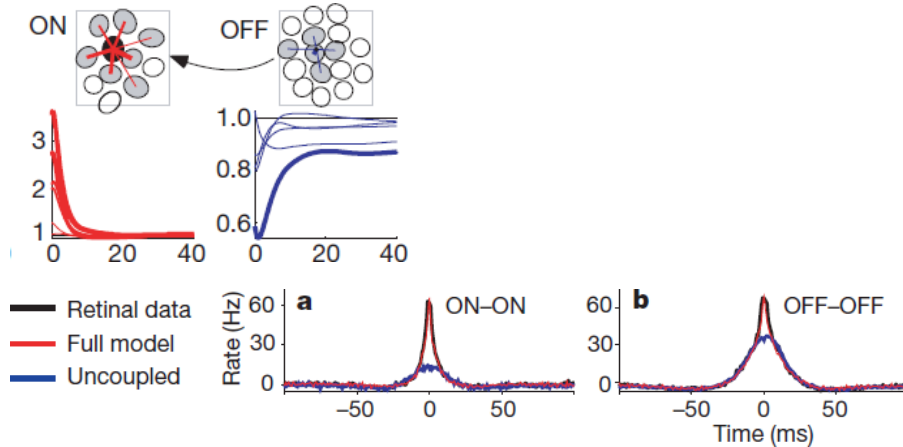

But how to fit?...

Network models

Generalization to networks.

- Unlikely to have data from all neurons
- Predict of cross-neuron spike patterns and correlations
- Correlations are important for decoding (coming lectures)
- Estimate 'functional coupling', $O(N \times N)$ parameters
- Uses small set of basis functions for kernels

[Pillow et al., 2008]

Network models

ON   OFF

— Retinal data
— Full model
— Uncoupled

Rate (Hz) **a** ON–ON   **b** OFF–OFF

Time (ms)

Note uncoupled case still correlations due to RF overlap, but less sharp. [Pillow et al., 2008]

Summary

Predicting neural responses
In order of decreasing generality

- Wiener kernels: systematic, but require a lot of data for higher orders
- Spike-triggered models: simple, but still leads to negative rates.
- LNP model: fewer parameters, spiking output, but no timing precision
- More neurally inspired models (I&F, GLM with spike feedback): good spike timing, but require regularization
- Biophyical models: in principle very precise, but in practice unwieldy

References I

Badel, L., Lefort, S., Berger, T. K., Petersen, C. C. H., Gerstner, W., and Richardson, M. J. E. (2008).
Extracting non-linear integrate-and-fire models from experimental data using dynamic I-V curves.
Biol Cybern, 99(4-5):361–370.

Bair, W. (1999).
Spike timing in the mammalian visual system.
Curr Opin Neurobiol, 9(4):447–453.

Berry, M. J. and Meister, M. (1998).
Refractoriness and Neural Precision.
J Neurosci, 18:2200–2211.

Butts, D. A., Weng, C., Jin, J., Yeh, C.-I., Lesica, N. A., Alonso, J.-M., and Stanley, G. B. (2007).
Temporal precision in the neural code and the timescales of natural vision.
Nature, 449(7158):92–95.

Chen, X., Han, F., Poo, M.-M., and Dan, Y. (2007).
Excitatory and suppressive receptive field subunits in awake monkey primary visual cortex (V1).
Proc Natl Acad Sci U S A, 104(48):19120–19125.

Chichilnisky, E. J. (2001).
A simple white noise analysis of neuronal light responses.
Network, 12:199–203.

Dayan, P. and Abbott, L. F. (2002).
Theoretical Neuroscience.
MIT press, Cambridge, MA.

References II

de Ruyter van Steveninck, R. R., Lewen, G. D., Strong, S. P., Koberle, R., and Bialek, W. (1997).
Reproducibility and variability in neural spike trains.
Science, 275:1805–1809.

Gerstner, W., Kistler, W., Naud, R., and Paninski, L. (2014).
Neuronal Dynamics: From Single Neurons, to networks and models of cognition.
Cambridge University Press.

Pillow, J. W., Paninski, L., Uzzell, V. J., Simoncelli, E. P., and Chichilnisky, E. J. (2005).
Prediction and Decoding of Retinal Ganglion Cell Responses with a Probabilistic Spiking Model.
J Neurosci, 23:11003–11013.

Pillow, J. W., Shlens, J., Paninski, L., Sher, A., Litke, A. M., Chichilnisky, E. J., and Simoncelli, E. P. (2008).
Spatio-temporal correlations and visual signalling in a complete neuronal population.
Nature, 454(7207):995–999.

Rieke, F., Warland, D., de Ruyter van Steveninck, R., and Bialek, W. (1996).
Spikes: Exploring the neural code.
MIT Press, Cambridge.

Sakai, H. M. (1992).
White-noise analysis in Neurophyiology.
Physiol Rev, 72:491–505.

Schetzen, M. (1981).
Nonlinear system modeling based on the Wiener theory.
Proc IEEE, 69(12):1557–1573.

Schetzen, M. (2006).
The Volterra and Wiener Theories of Nonlinear Systems.
Krieger publishers.

Schwartz, O., Chichilnisky, E. J., and Simoncelli, E. P. (2001).
Characterizing neural gain control using spike-triggered covariance.
NIPS, 14:269–276.

Temchin, A. N., Recio-Spinos, A., van Dijk, P., and Ruggero1, M. A. (2005).
Wiener Kernels of Chinchilla Auditory-Nerve Fibers: Verification Using Responses to Tones, Clicks, and Noise and Comparison With Basilar- Membrane Vibrations.
J Neurophysiol, 93:3635–3648.

Touryan, J., Felsen, G., and Dan, Y. (2005).
Spatial Structure of Complex Cell Receptive Fields Measured with Natural Images.
Neuron, 45:781–791.

van Hateren, J. H., Rüttiger, L., Sun, H., and Lee, B. B. (2002).
Processing of Natural Temporal Stimuli by Macaque Retinal Ganglion Cells.
J Neurosci, 22:9945–9960.

Wallis, J., Anderson, K. C., and Miller, E. K. (2001).
Single neurons in prefrontal cortex encode abstract rules.
Nature, 441:953–957.

Weber, A. I. and Pillow, J. W. (2017).
Capturing the dynamical repertoire of single neurons with generalized linear models.
Neural computation, 29(12):3260–3289.