Computational neuroscience
Recording from the brain
Recording from the brain
Reading out the neural code
Reading out the neural code
Reading out the neural code
What is the neural code?
What is the neural code?

Single neurons

Populations
Encoding and decoding

*Encoding*: how does a stimulus cause a pattern of responses?

• what are the responses and what are their characteristics?
• neural models:
  -- from stimulus to response
  -- descriptive $\leftrightarrow$ mechanistic models

*Decoding*: what do these responses tell us about the stimulus?

• Implies some kind of decoding algorithm
• How to evaluate how good our algorithm is?
Encoding and decoding

\[
P(\text{response} | \text{stimulus}) \quad \text{encoding}
\]
\[
P(\text{stimulus} | \text{response}) \quad \text{decoding}
\]

- What is response?
- What is stimulus?
- What is the function $P$?
Tuning curves

Nonlinear function: \( r = g(s) \)

Gaussian tuning curve of a cortical (V1) neuron
Tuning curves

Hand reaching direction

Nonlinear function: \( r = g(s) \)

Cosine tuning curve of a motor cortical neuron
Nonlinear function: \( r = g(s) \)

Sigmoidal tuning curve of a V1 cortical neuron
Map of feature selectivity in primary visual cortex


©2008 by American Physiological Society
“Tuning curves”

What is $s$?
Tuning curves

What is \( s \)?
Building up complex selectivity
Basic coding model: linear filtering

Spatial filter: \[ r = \int\int f(x,y) I(x_0-x, y_0-y) \, dx \, dy \]

retina

Visual cortex
Basic coding model: temporal filtering

Linear filter: $r(t) = \int s(t-\tau) f(\tau) \, dt$
Basic coding model: temporal filtering

Linear filter: \( r(t) = \int s(t-\tau) f(\tau) \, d\tau \)

...shortcomings?
Next most basic coding model

Linear filter & nonlinearity: \( r(t) = g(\int s(t-\tau) f(\tau) \, d\tau) \)
How to find the components of this model
Reverse correlation: the spike-triggered average

Spike-conditional ensemble

\[ s \]

\[ + \]

\[ + \]

\[ + \]

\[ ... \]

spike-triggered average
The spike-triggered average
More generally, one can conceive of the action of the neuron or neural system as *selecting a low dimensional subset* of its inputs.

\[ P(\text{response} \mid \text{stimulus}) \rightarrow P(\text{response} \mid s_1, s_2, \ldots, s_n) \]

Start with a very high dimensional description (eg. an image or a time-varying waveform) and pick out a small set of relevant dimensions.

\[ S(t) = (S_1, S_2, S_3, \ldots, S_n) \]
Linear filtering

Linear filtering = convolution = projection

Stimulus feature is a vector in a high-dimensional stimulus space
Determining linear features from white noise
How to find the components of this model

\[ s(t) \rightarrow f_1 \rightarrow \text{feature} \rightarrow S^*f_1 \rightarrow \text{decision function} \rightarrow r(t) \]
Determining the nonlinear input/output function

The input/output function is:

\[ P(\text{spike}|\text{stimulus}) \]

which can be derived from data using Bayes’ rule:

\[ P(\text{spike}|s_1) = \frac{P(s_1|\text{spike})P(\text{spike})}{P(s_1)} \]
Tuning curve: \[ P(\text{spike}|s) = \frac{P(s|\text{spike}) P(\text{spike})}{P(s)} \]
Next most basic coding model

Linear filter & nonlinearity: $r(t) = g(\int f(t-\tau) s(\tau) \, d\tau)$

...shortcomings?
Less basic coding models

Linear filters & nonlinearity: \( r(t) = g(f_1 * s, f_2 * s, \ldots, f_n * s) \)
Determining linear features from white noise

Spike-triggered average

Spike-conditional distribution

Gaussian prior stimulus distribution

Covariance
Identifying multiple features

The covariance matrix is

\[ C_{ij} = \langle S(t_{\text{spike}} - t_i)S(t_{\text{spike}} - t_j) \rangle - \bar{S}(t_i)\bar{S}(t_j) - \langle I(t - t_i)I(t - t_j) \rangle \]

Properties:

• The number of eigenvalues significantly different from zero is the number of relevant stimulus features

• The corresponding eigenvectors are the relevant features (or span the *relevant subspace*)

Bialek et al., 1988; Brenner et al., 2000; Bialek and de Ruyter, 2005
A toy example: a filter-and-fire model

Let’s develop some intuition for how this works: a filter-and-fire threshold-crossing model with AHP

- Spiking is controlled by a single filter, \( f(t) \)
- Spikes happen generally on an upward threshold crossing of the filtered stimulus

⇒ expect 2 **relevant features**, the filter \( f(t) \) and its time derivative \( f'(t) \)

Keat, Reinagel, Reid and Meister, Predicting every spike. Neuron (2001)
Covariance analysis of a filter-and-fire model
Let’s try it

Example: rat somatosensory (barrel) cortex
Rasmus Petersen and Mathew Diamond, SISSA

Record from single units in barrel cortex
White noise analysis in barrel cortex

Spike-triggered average:
Is the neuron simply not very responsive to a white noise stimulus?
Covariance matrices from barrel cortical neurons

Prior

Spike-triggered

Difference
Eigenspectrum from barrel cortical neurons

Eigenspectrum

Leading modes
Eigenspectrum from barrel cortical neurons

Eigenspectrum

Leading modes
Input/output relations from barrel cortical neurons

Input/output relations wrt first two filters, alone:

and in quadrature:
Less significant eigenmodes from barrel cortical neurons

How about the other modes?

Next pair with +ve eigenvalues

Pair with -ve eigenvalues
Negative eigenmode pair

Input/output relations for negative pair

Firing rate decreases with increasing projection: suppressive modes
Salamander retinal ganglion cells perform a variety of computations.
Almost filter-and-fire-like
Not a threshold-crossing neuron
Complex cell like

Stimulus

Eigenvalue

Mode Index

Projection onto Mode 2

Projection onto Mode 1

Spike-Triggered Average

Time before Spike (s)
Bimodal: two separate features are encoded
When have you done a good job?

• When the tuning curve over your variable is *interesting*.

• How to quantify interesting?
When have you done a good job?

Tuning curve: \[ P(\text{spike}|s) = \frac{P(s|\text{spike}) P(\text{spike})}{P(s)} \]

Boring: spikes unrelated to stimulus

Interesting: spikes are selective

Goodness measure: \[ D_{KL}(P(s|\text{spike}) | P(s)) \]
Maximally informative dimensions

*Sharpee, Rust and Bialek, Neural Computation, 2004*

Choose filter in order to maximize $D_{KL}$ between spike-conditional and prior distributions.
Choose filter in order to maximize $D_{KL}$ between spike-conditional and prior distributions

Equivalent to maximizing mutual information between stimulus and spike

Does not depend on white noise inputs

Likely to be most appropriate for deriving models from natural stimuli
Finding relevant features

1. Single, best filter determined by the first moment
2. A family of filters derived using the second moment
3. Use the entire distribution: information theoretic methods

Removes requirement for Gaussian stimuli
Less basic coding models

Linear filters & nonlinearity: \( r(t) = g(f_1*s, f_2*s, \ldots, f_n*s) \)

...shortcomings?
GLM: \( r(t) = g(f_1 s + f_2 r) \)

Less basic coding models

GLM: \( r(t) = g(f*s + h*r) \)

...shortcomings?
GLM: \( r(t) = g(f_1 \ast s + h_1 \ast r_1 + h_2 \ast r_2 + \ldots) \)

...shortcomings?