

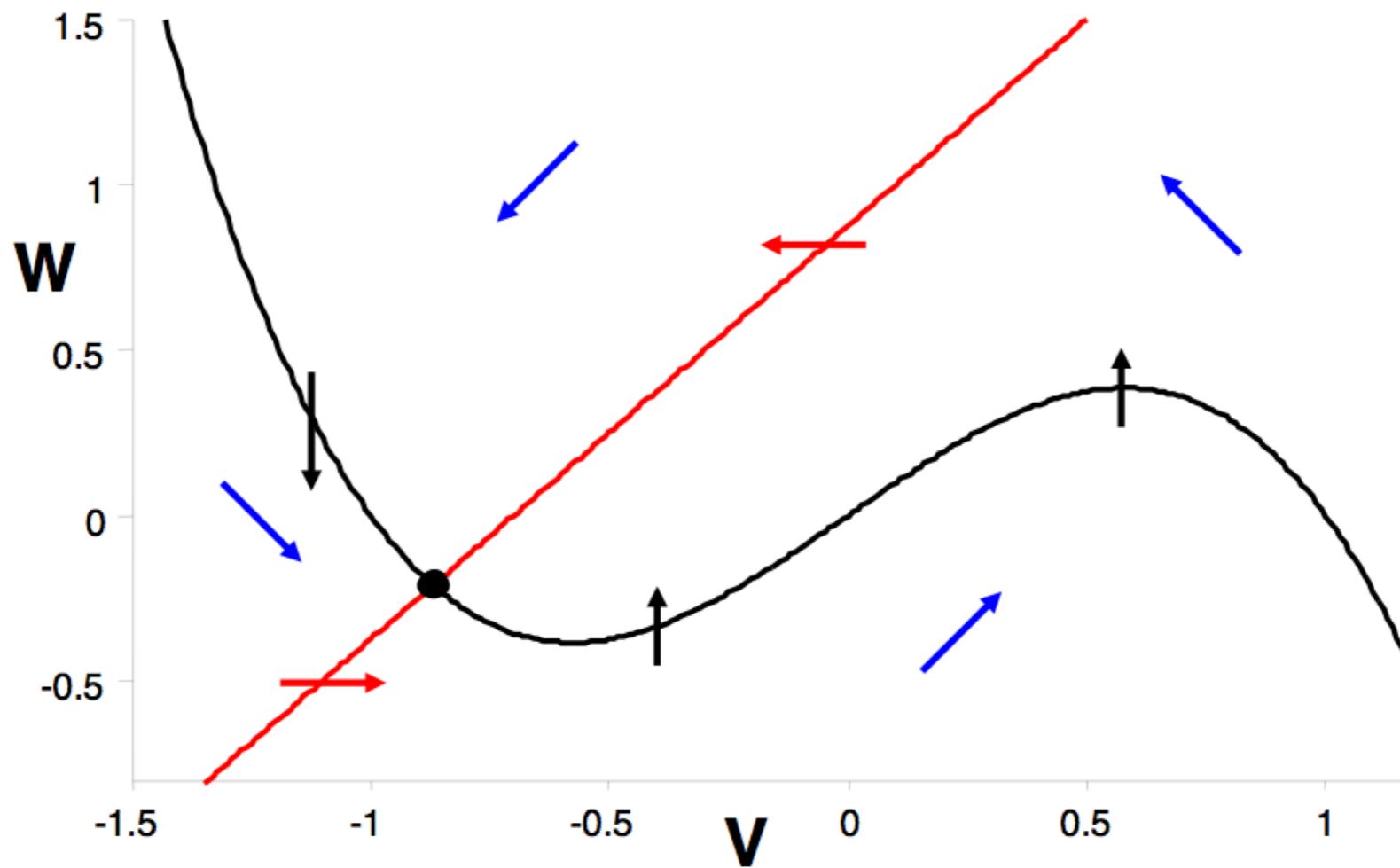
# *The Fitzhugh-Nagumo model:*



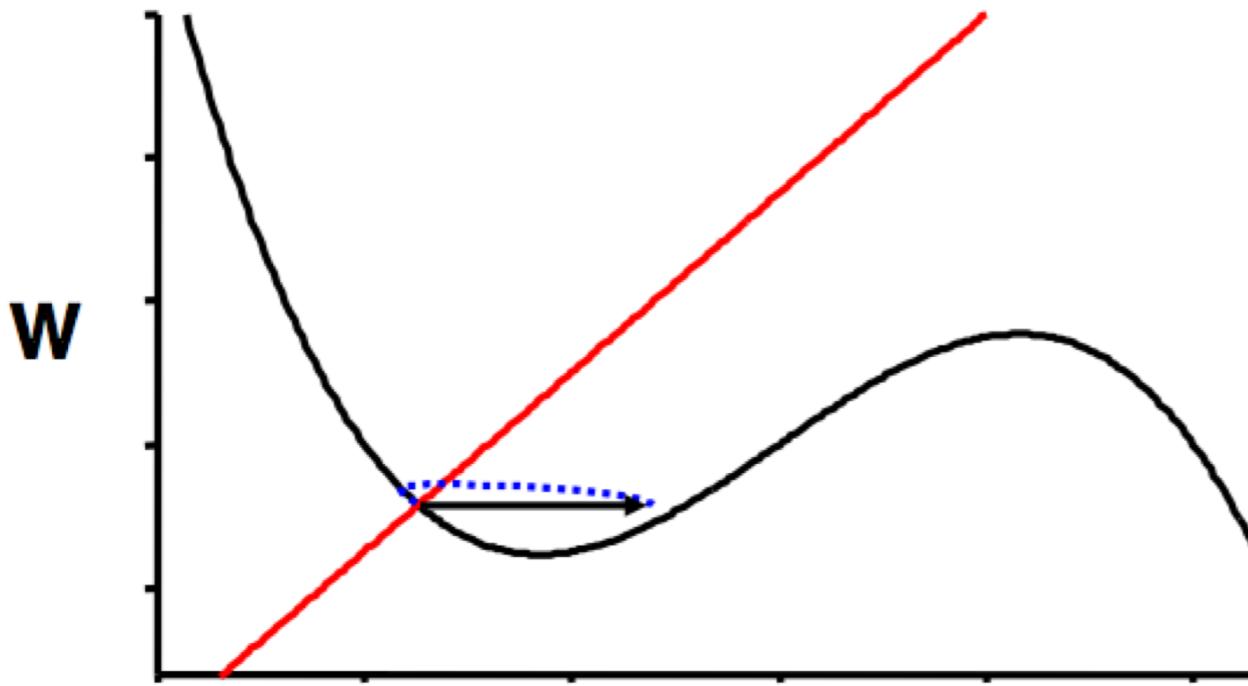
$$\frac{dV}{dt} = V - V^3 - W - I$$
$$\frac{dW}{dt} = 0.08*(V + 0.7 - 0.8W)$$



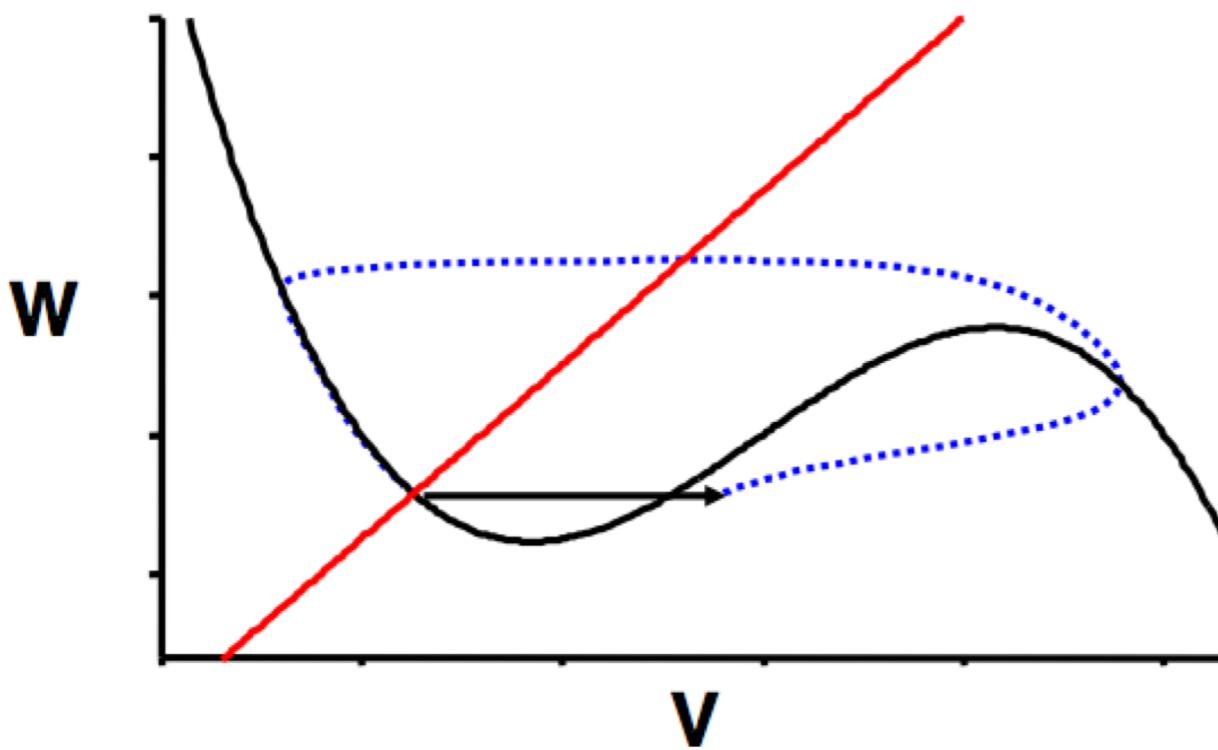
*The Fitzhugh-Nagumo model is a useful simplification:*



*The Fitzhugh-Nagumo model is a useful simplification:*

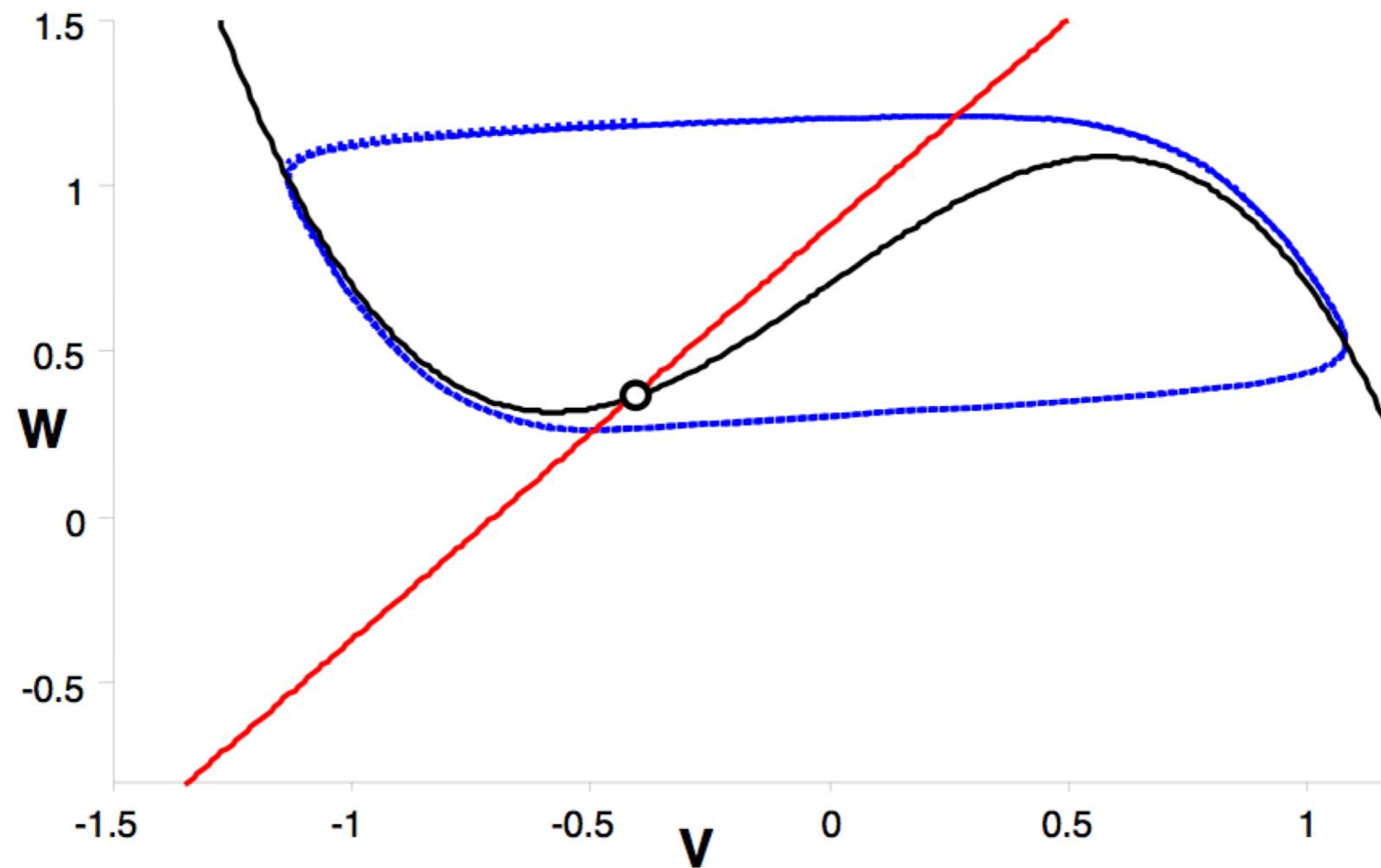


*return to baseline*



*spike!*

*Even explains sustained firing!*



# Lecture 2: Spike sorting

**Acknowledgements:** *Phil Nelson and Vijay Balasubramanian,  
U Penn*



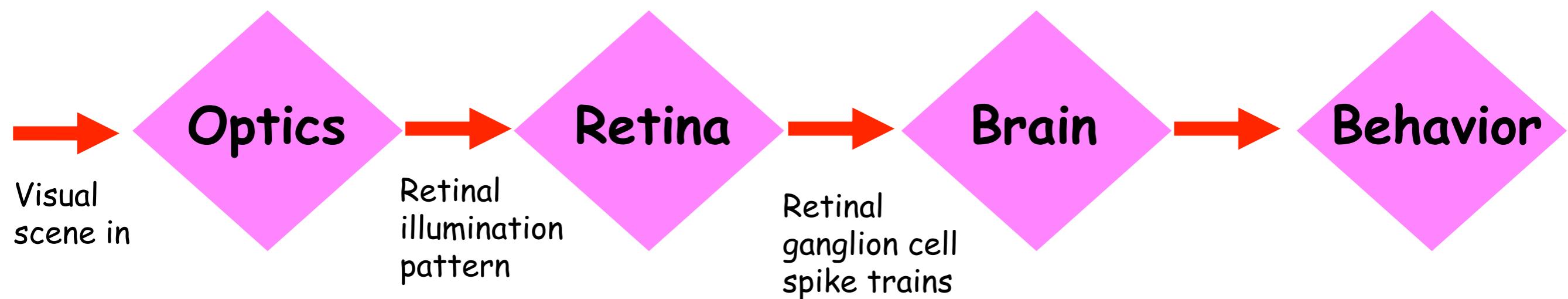
Courtesy of Phil Nelson, U Penn, and with apologies to B. Watterson

# Big Picture:

Traditional work: record from a small number of neurons via patch clamp. Work out circuit mechanisms this way.

New approaches: MEAs permit high throughput measurements from many neurons at once. This allows simultaneous measurements from many cells.

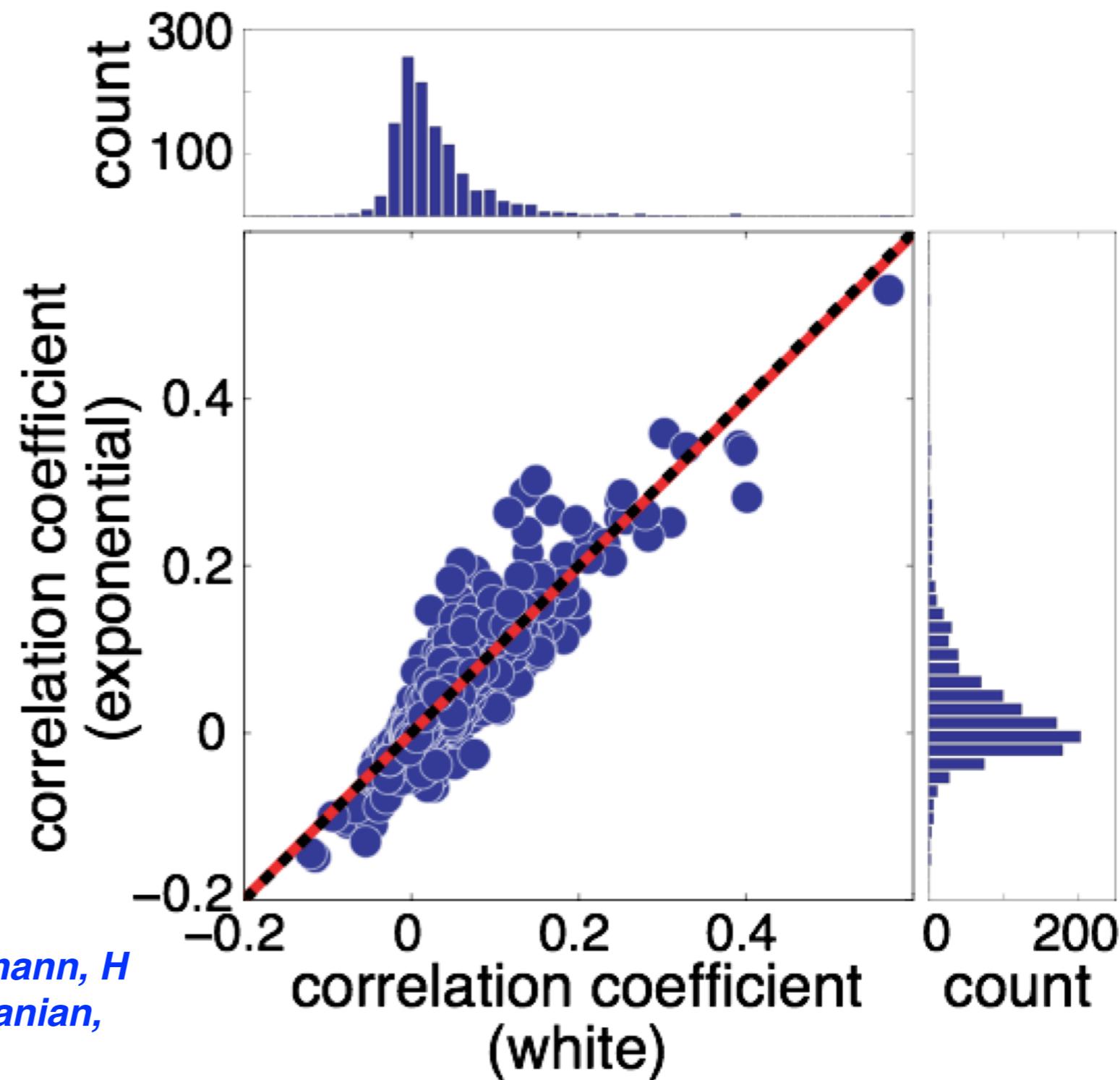
This is important for understanding population coding, cortical visual processing and behavior



# New questions:

e.g.: RETINA ADAPTS TO  
MAINTAIN INVARIANCE OF  
OUTPUT CORRELATIONS

The correlation between any two ganglion cells is nearly unchanged when we change the correlation strength in the stimulus.

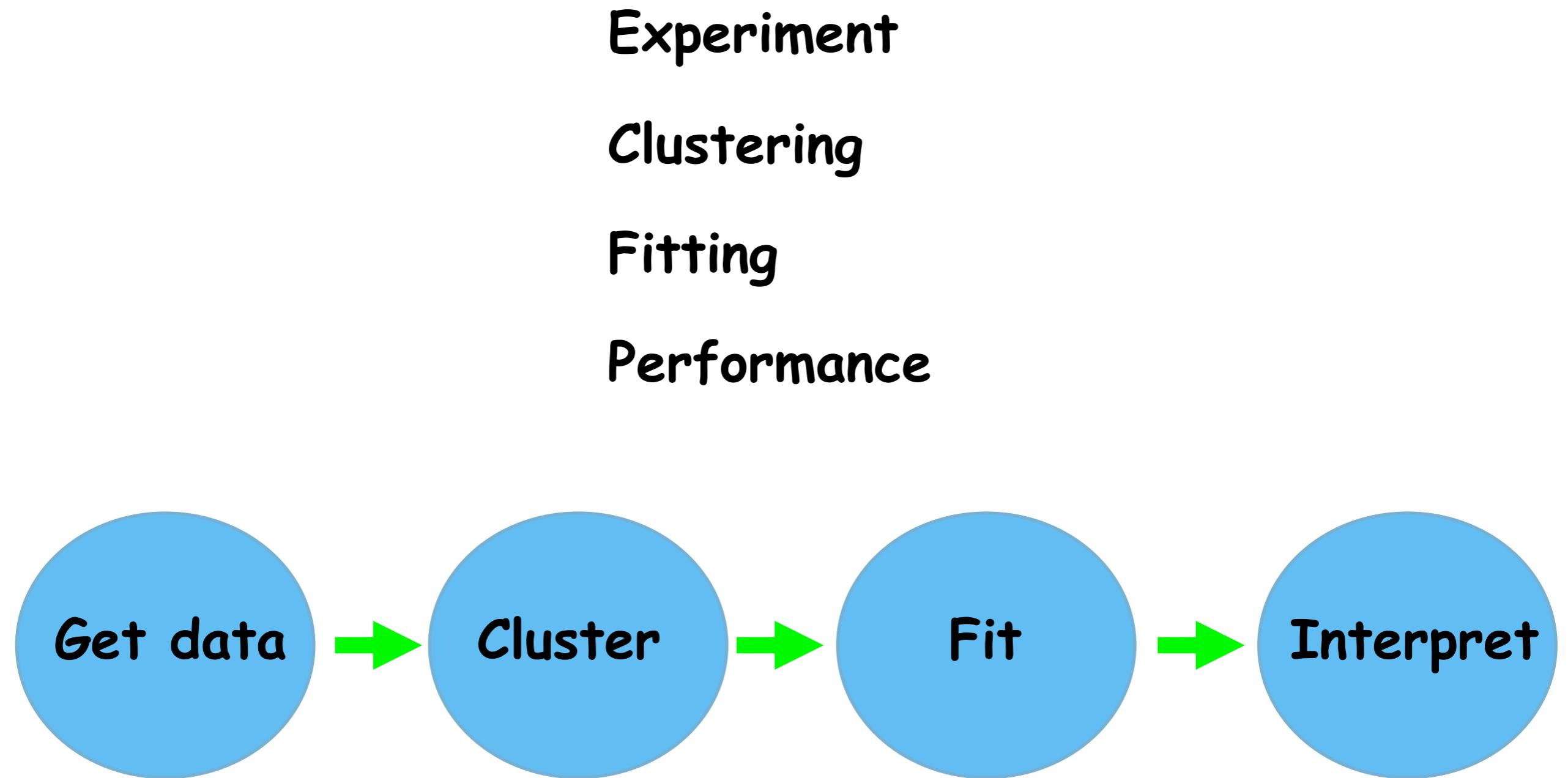


KD Simmons, JS Prentice, G Tkačik, J Homann, H Yee, SE Palmer, PC Nelson, V Balasubramanian, *JNeurosci* 2013.

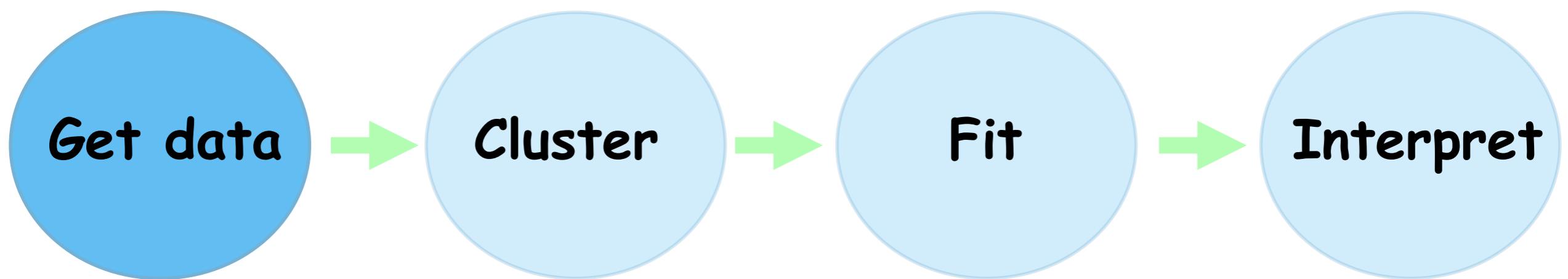
# Challenges:

1. Identifying cell types
2. Processing the huge amount of data
3. New kinds of questions and analyses
4. SPIKE SORTING

# Outline:

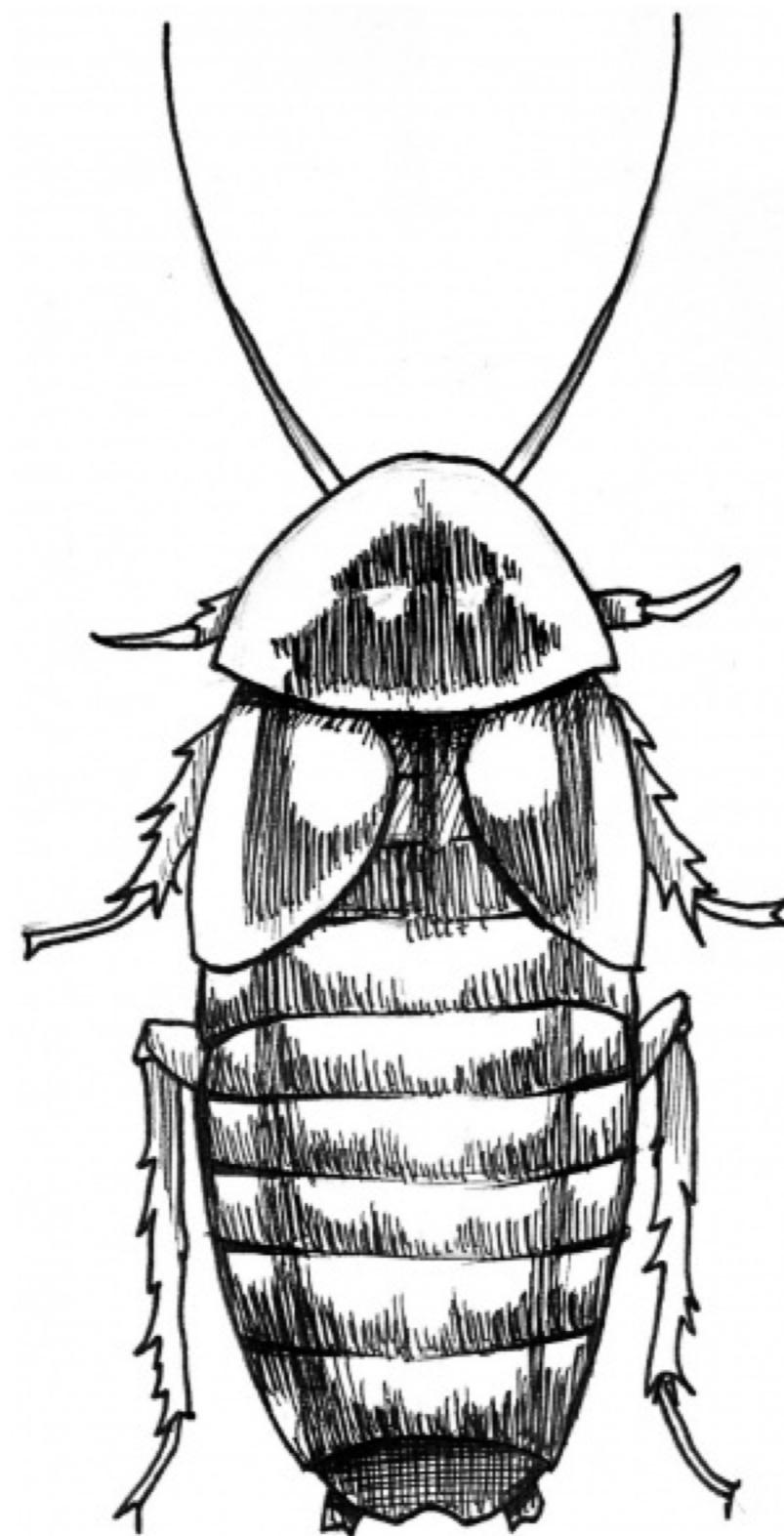


1. Experiment
2. Clustering
3. Fitting
4. Performance



*Single channel extracellular recording:*

***cockroach interlude***



## Cockroach Anatomy

---

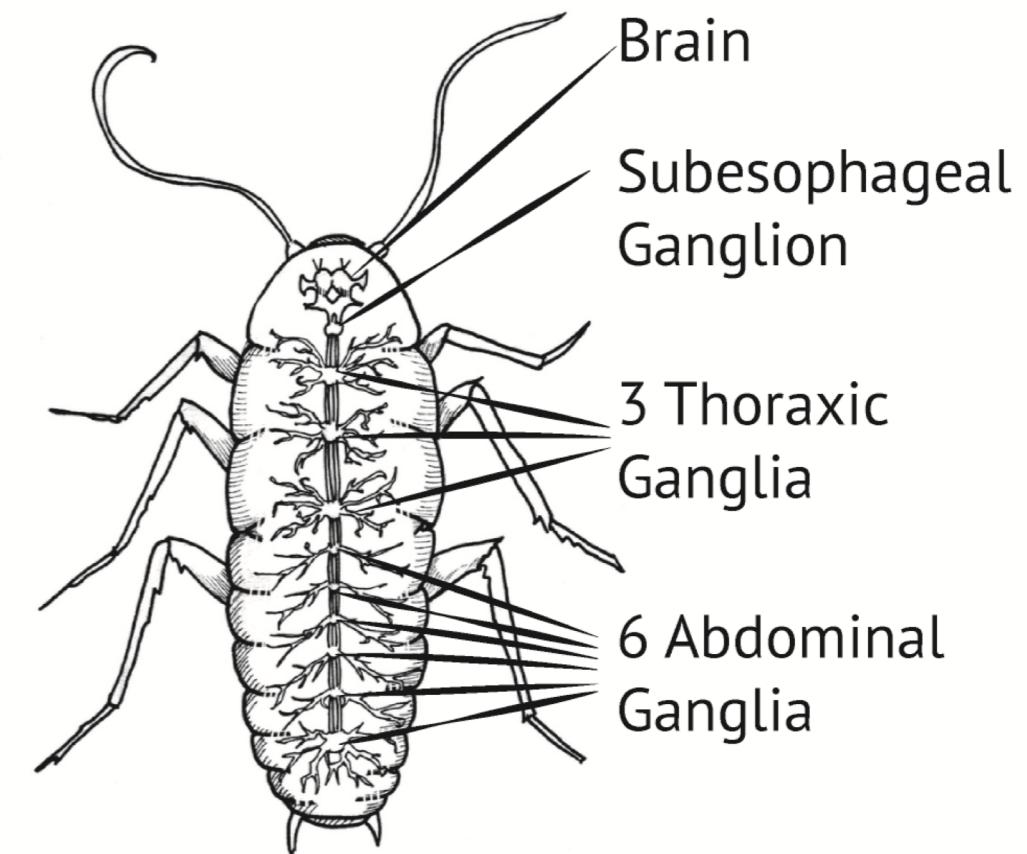
ABDOMEN THORAX HEAD



DORSAL VIEW

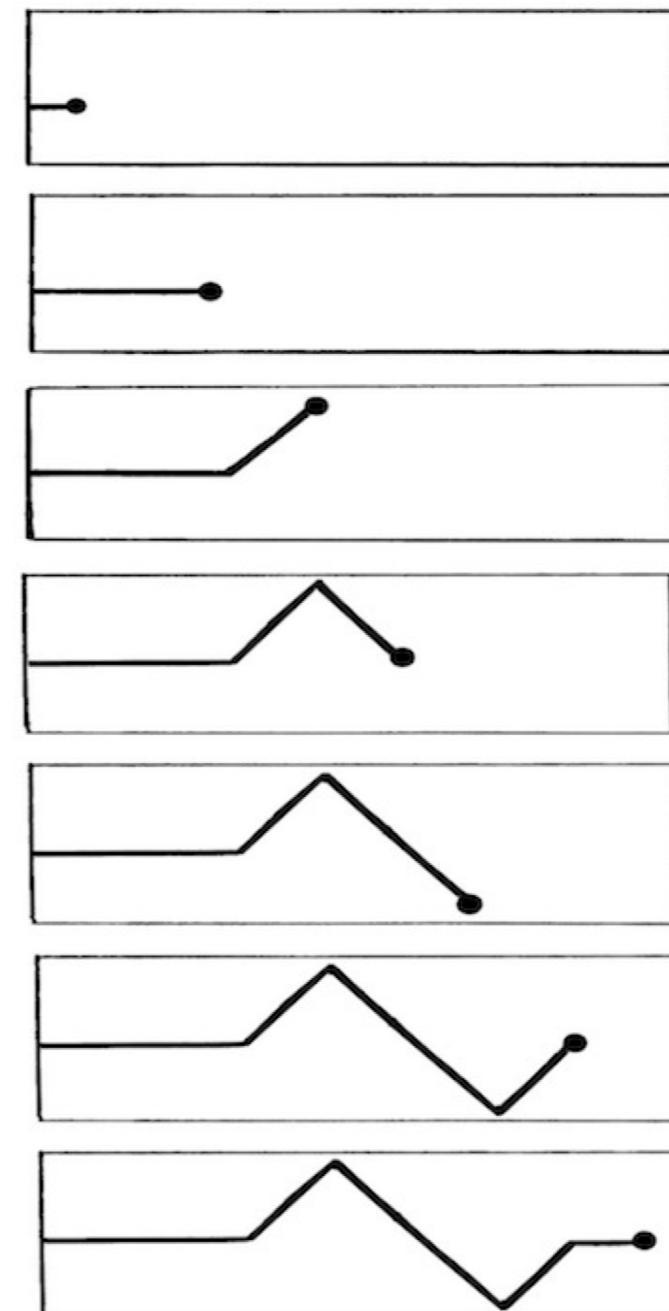
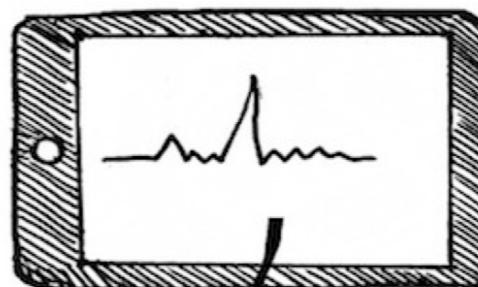
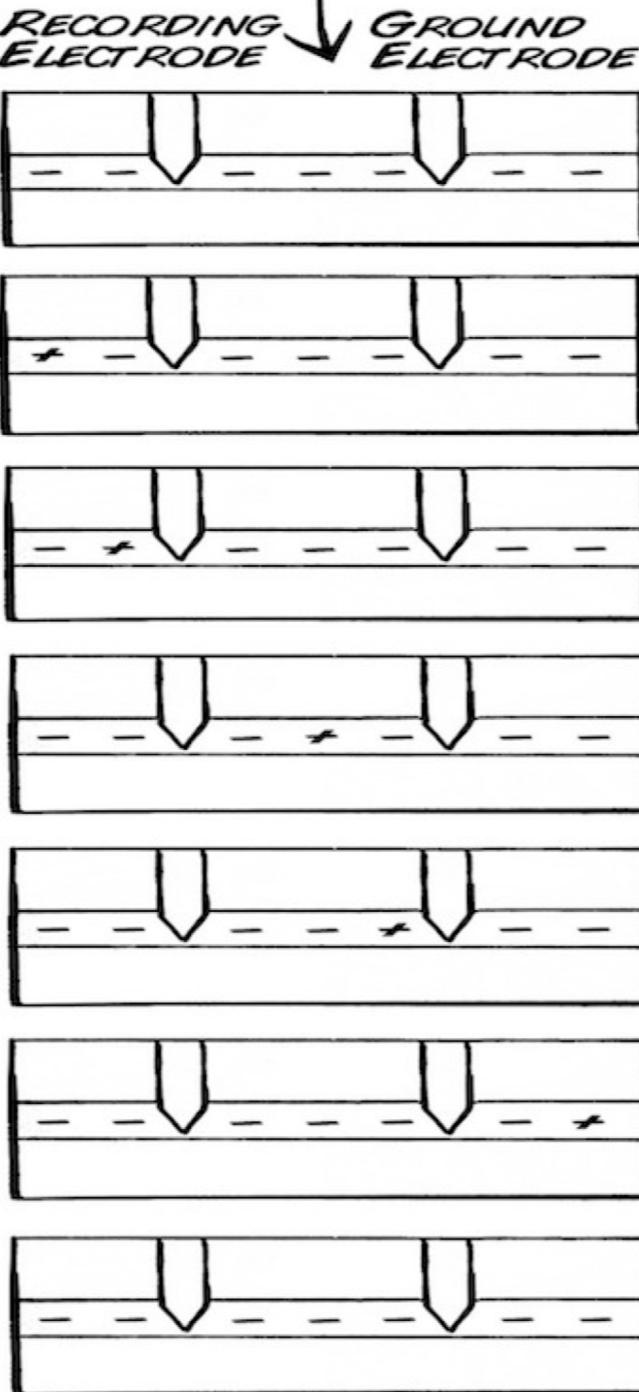


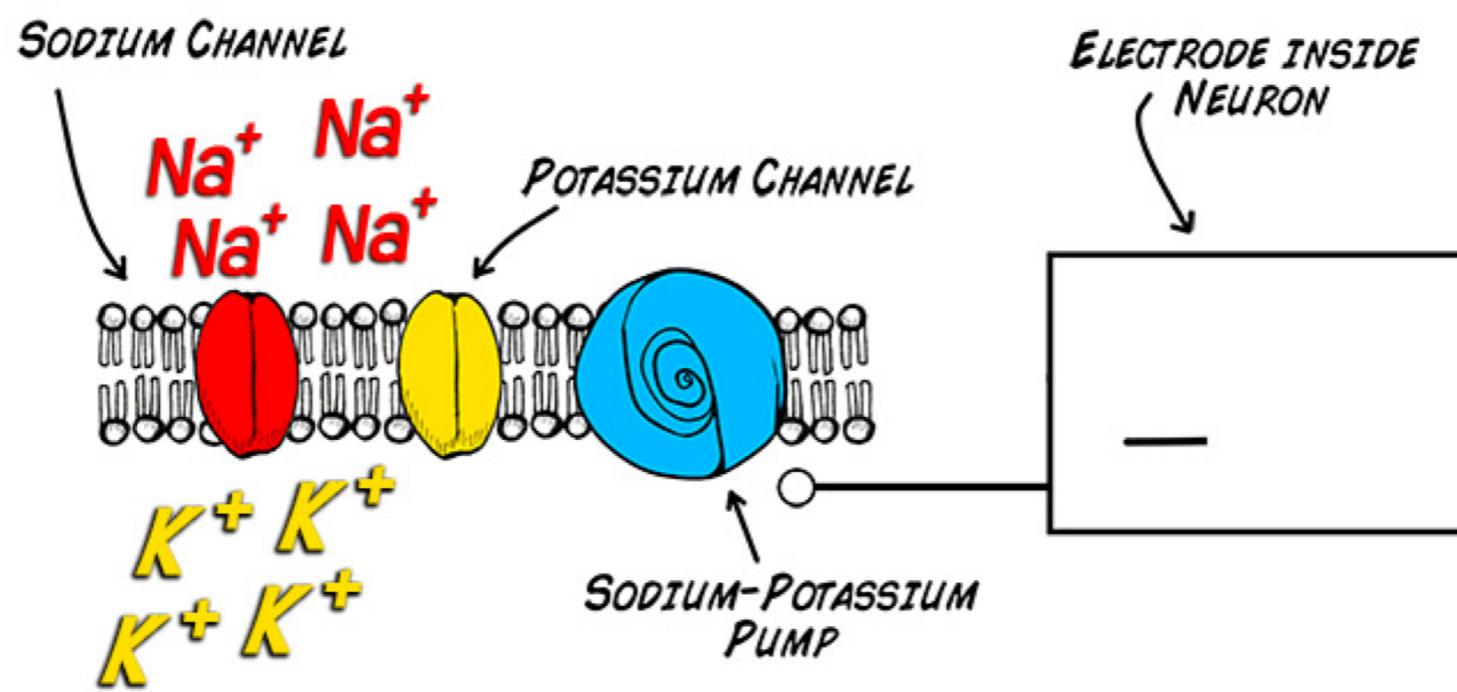
VENTRAL VIEW

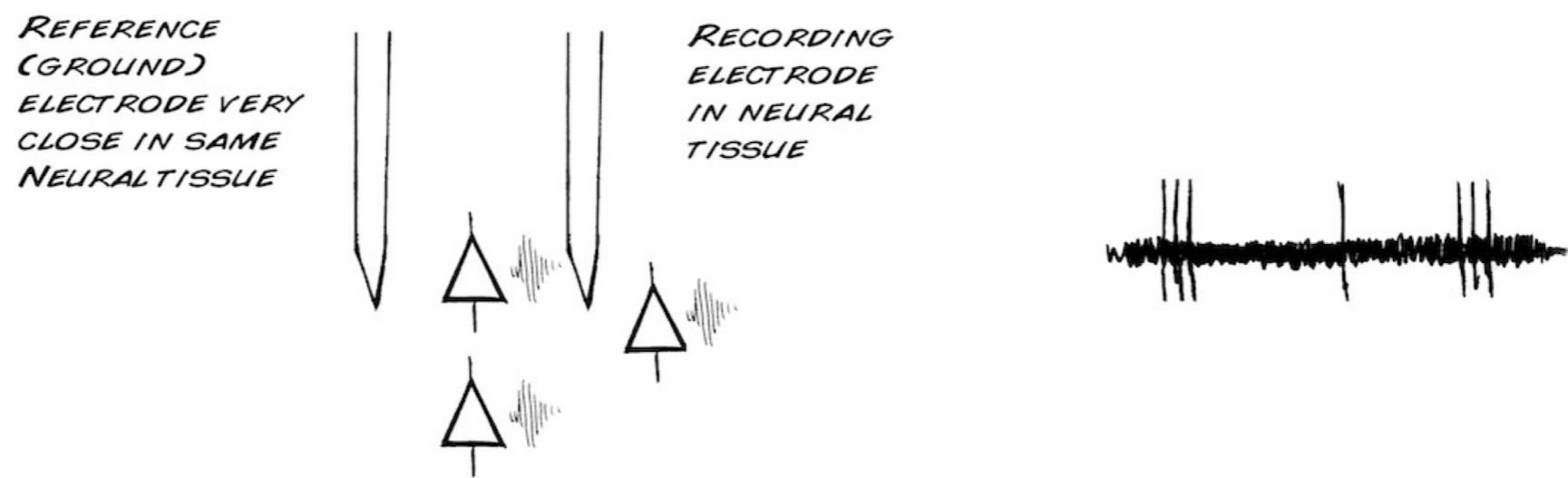
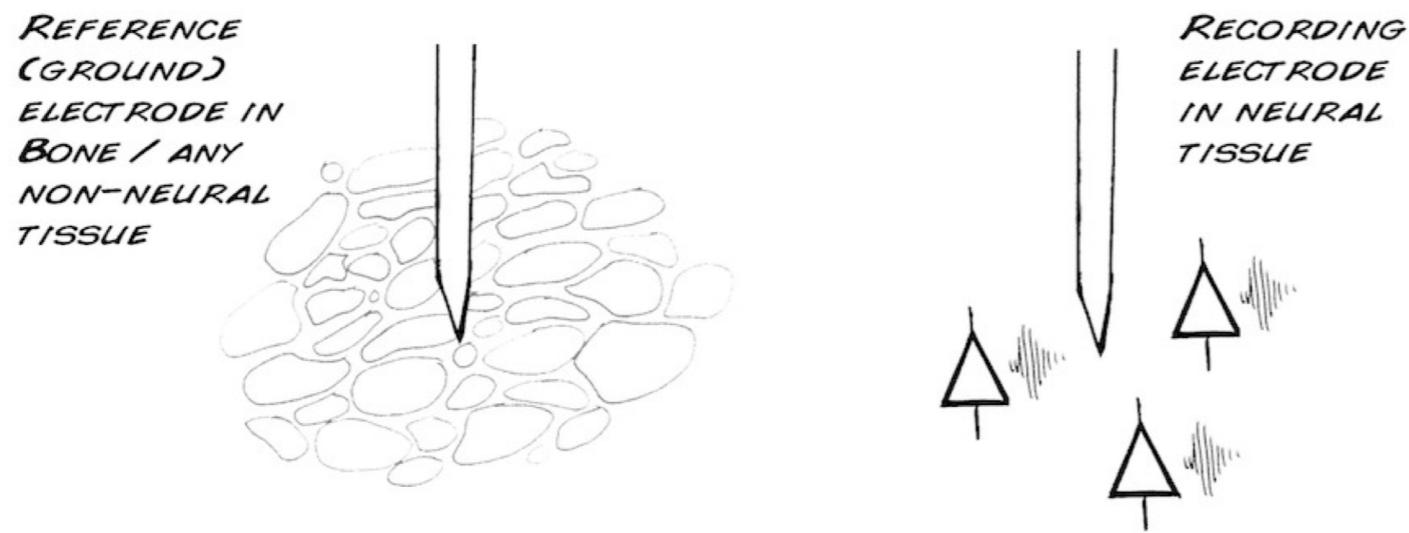
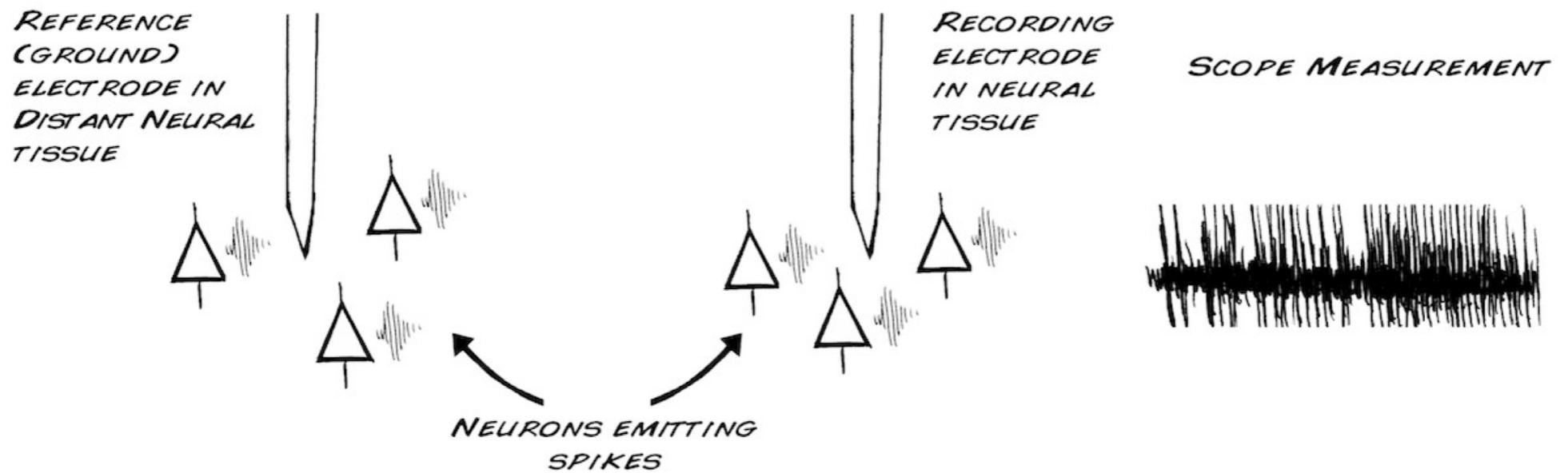


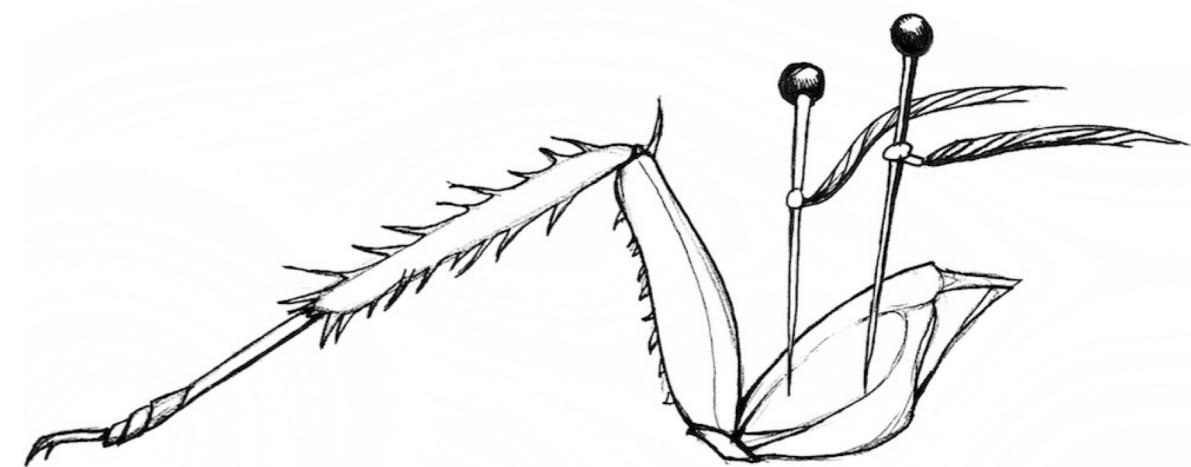
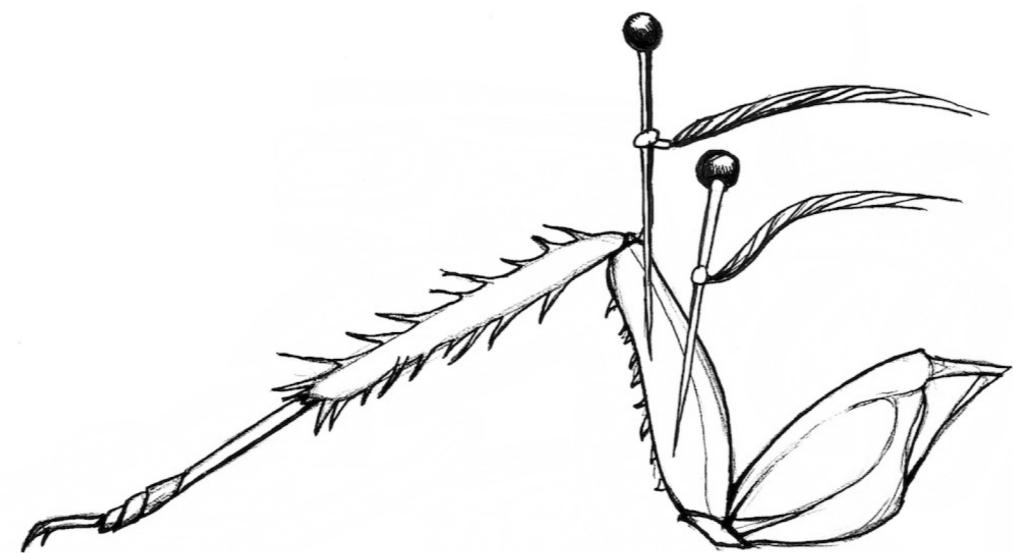
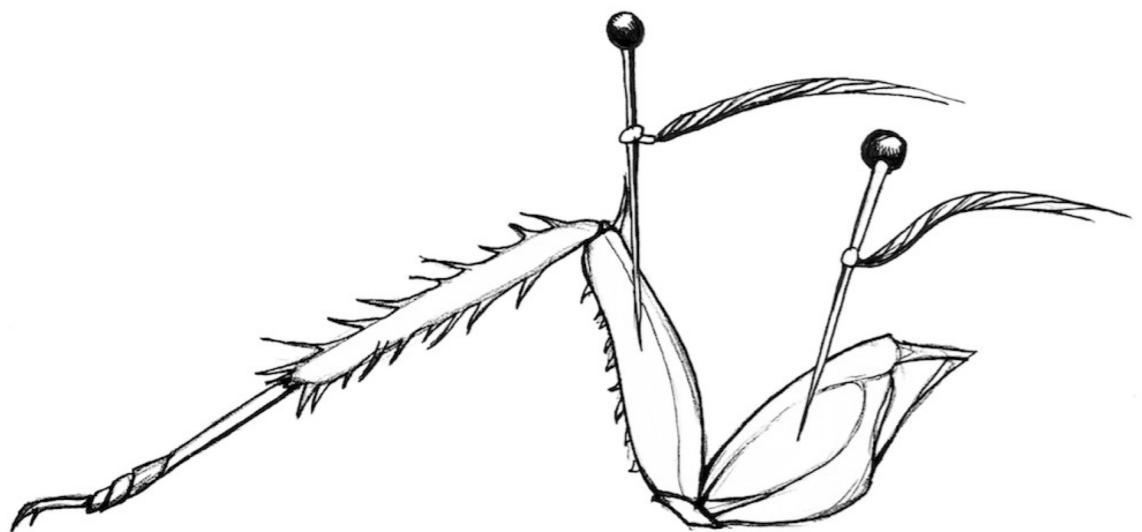
CENTRAL NERVOUS  
SYSTEM VIEW

# Recording from neurons

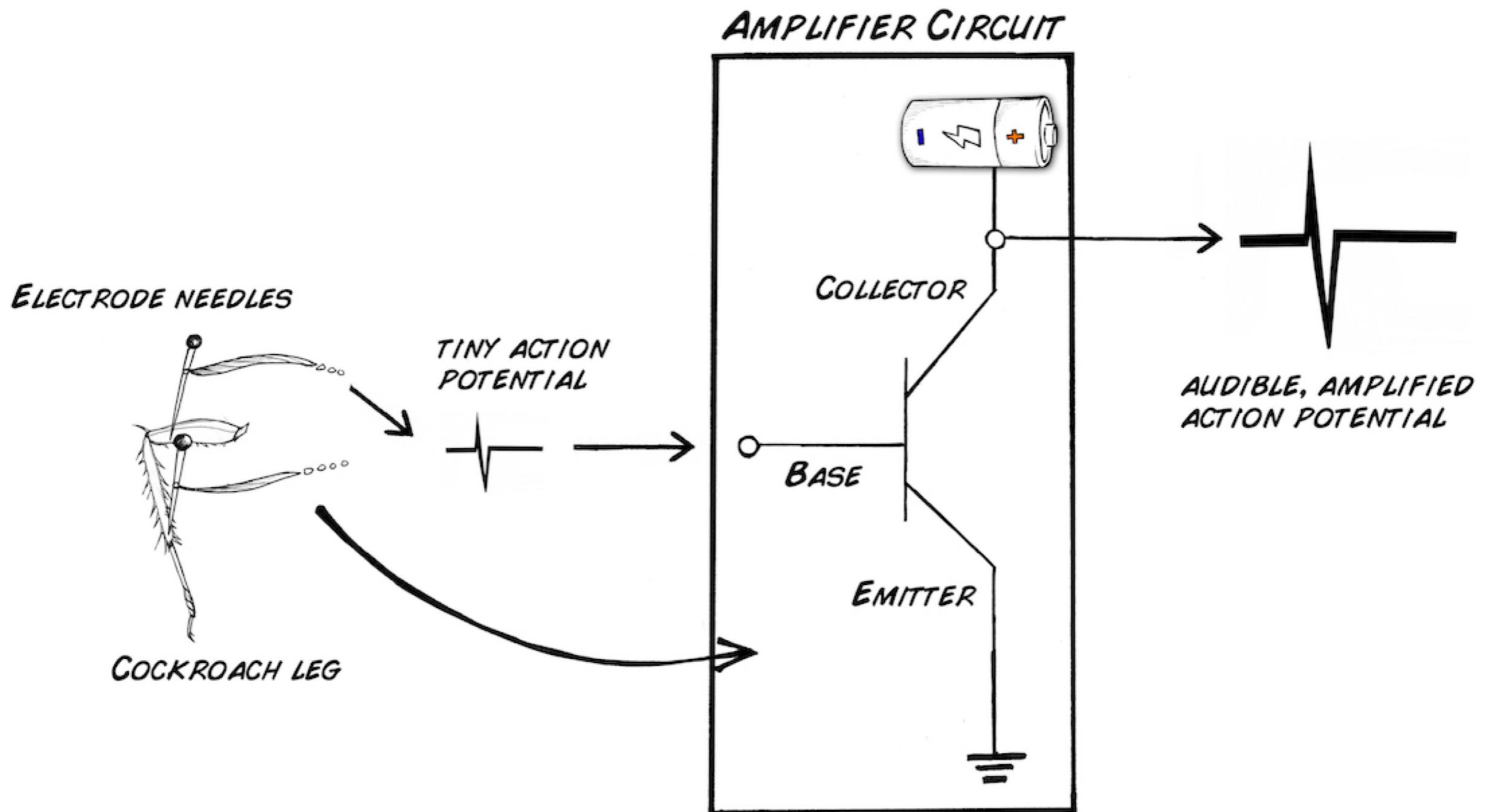








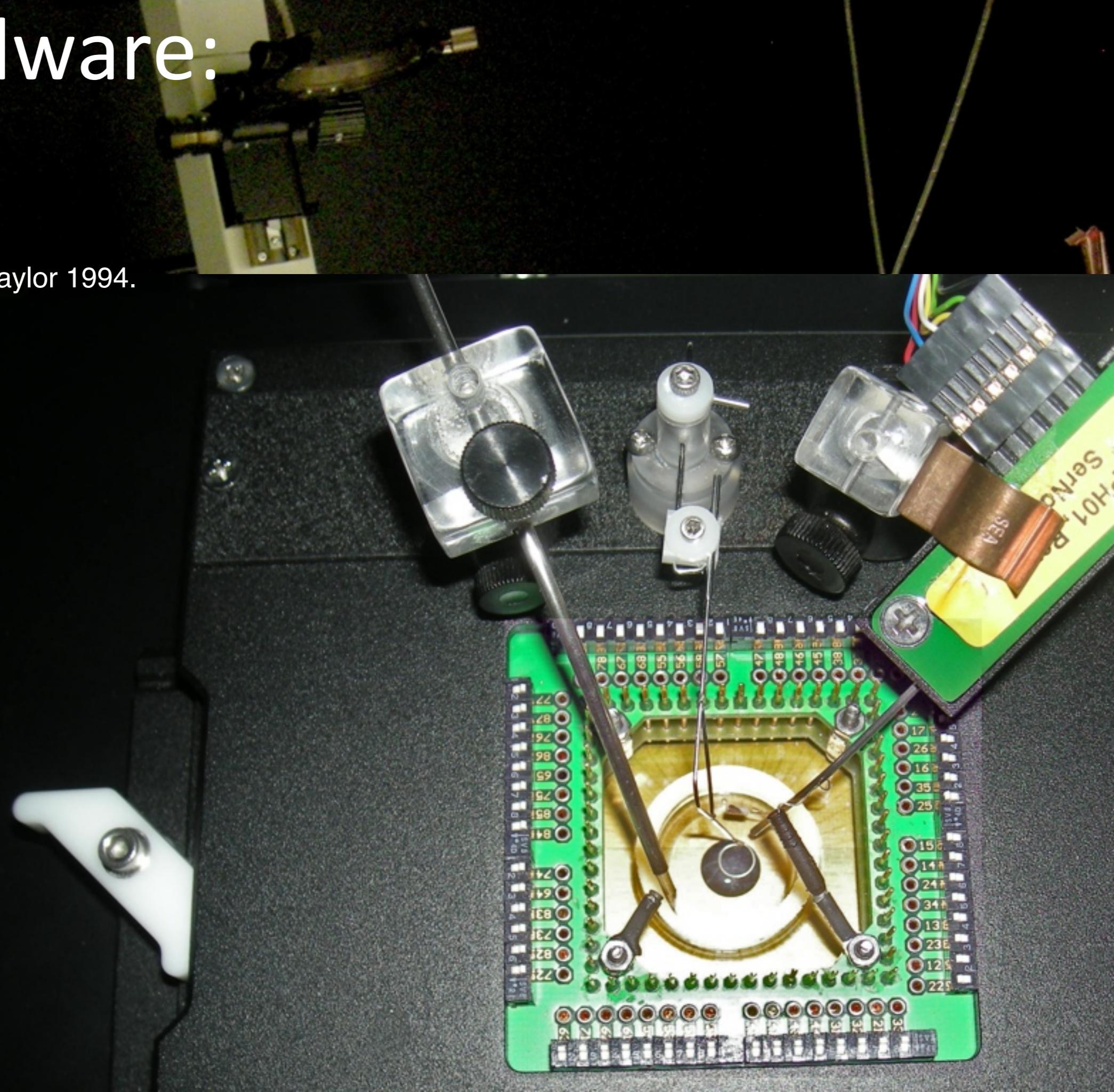
# *looking under the hood*



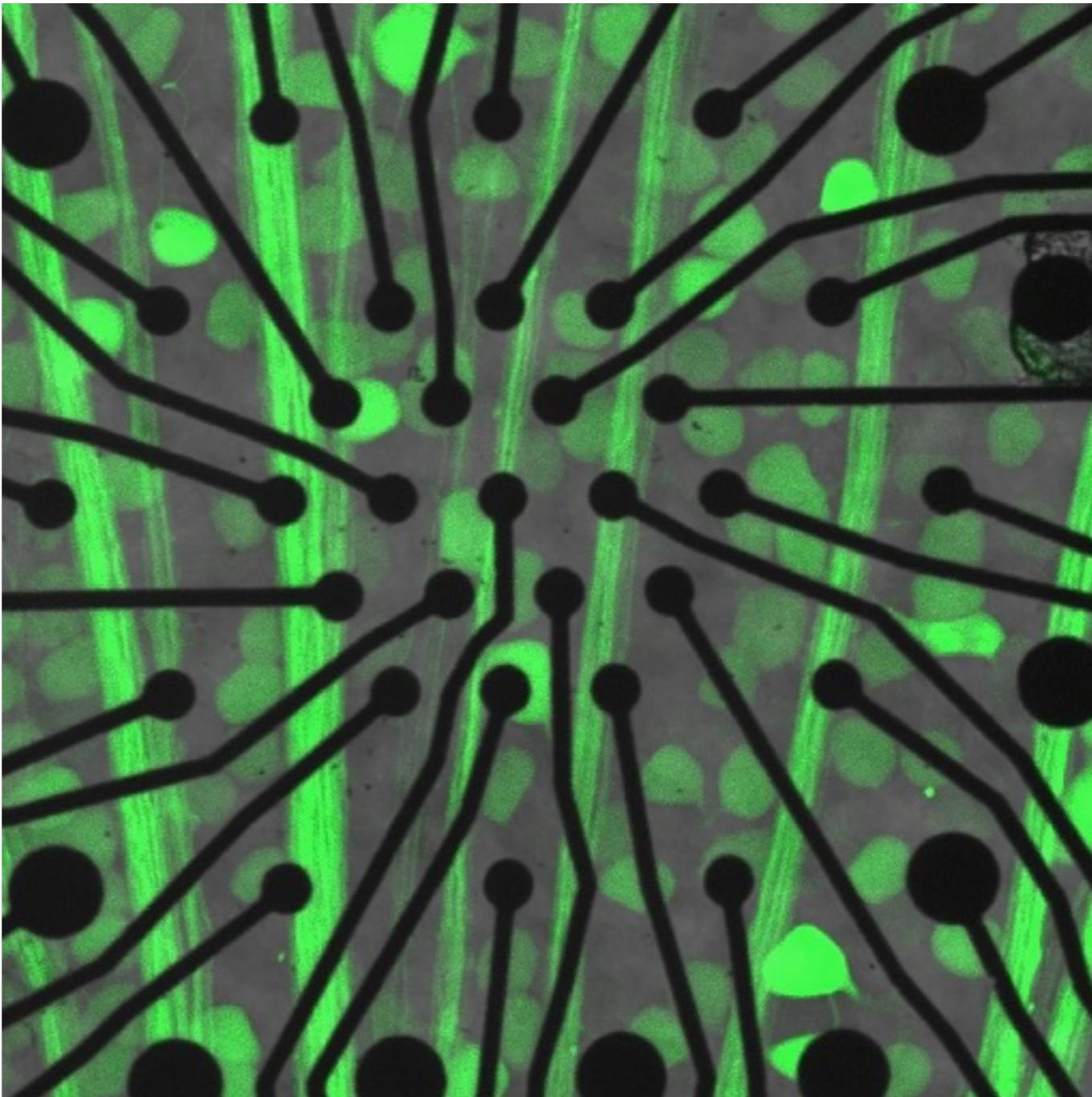
*back to more  
elaborate recordings*

# The hardware:

c.f. Meister, Pine, and Baylor 1994.



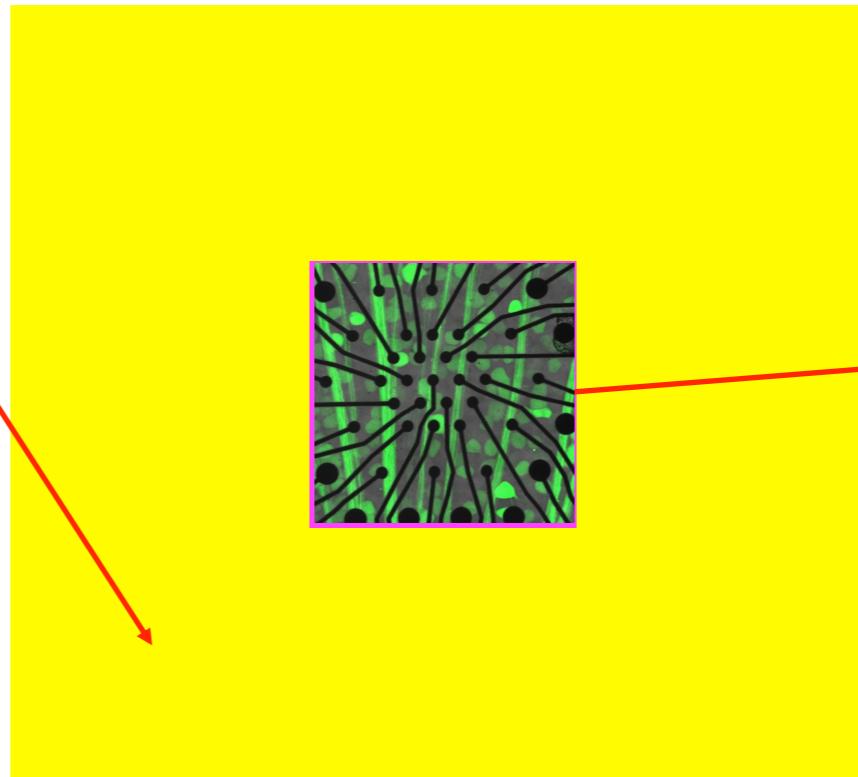
# What's in the dish?



from the Berry Lab, Princeton

# Typical stimulus setup:

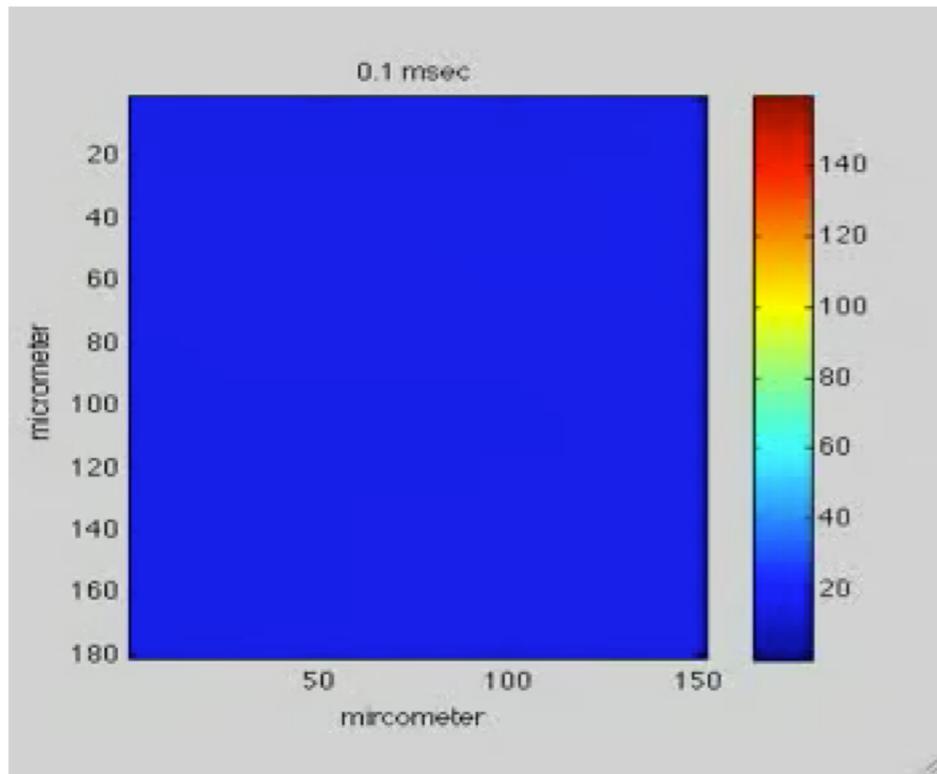
Field of illumination  
= e.g. 30x30 grid of stimulus checks, each 50 $\mu$ m on an edge.



Recording region  
= e.g., 5x6 array of electrodes spaced 30 $\mu$ m (similar to RGC spacing).

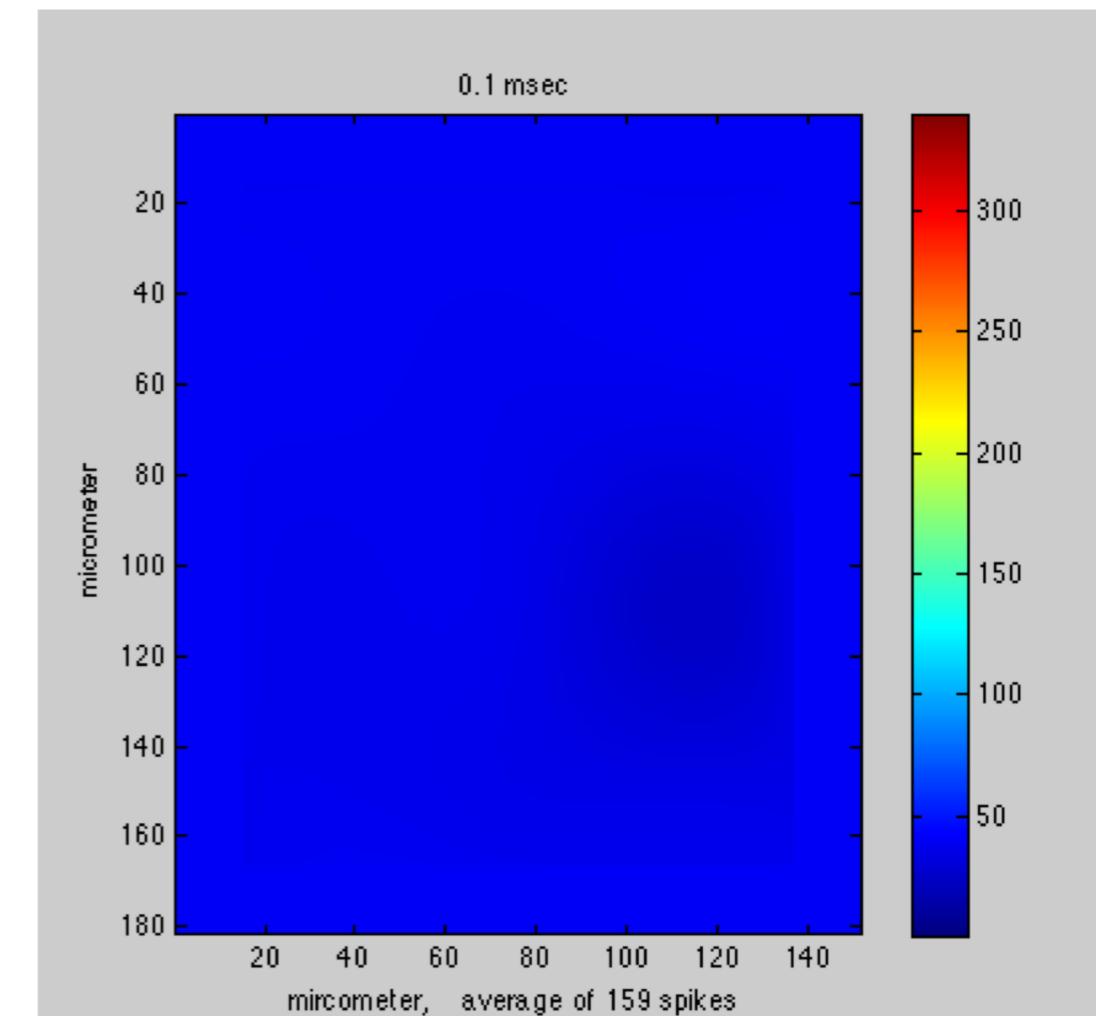
- Data taken at ~10kHz.
- Noise ~30 $\mu$ V. Big spikes ~400 $\mu$ V. Others go all the way down to the noise floor.
- Prior to analysis, filter out slow baseline drift.

# Simple events:



67 ms of data,  
viewed as a movie.  
[data have been smoothed]

Some spikes move across the array:



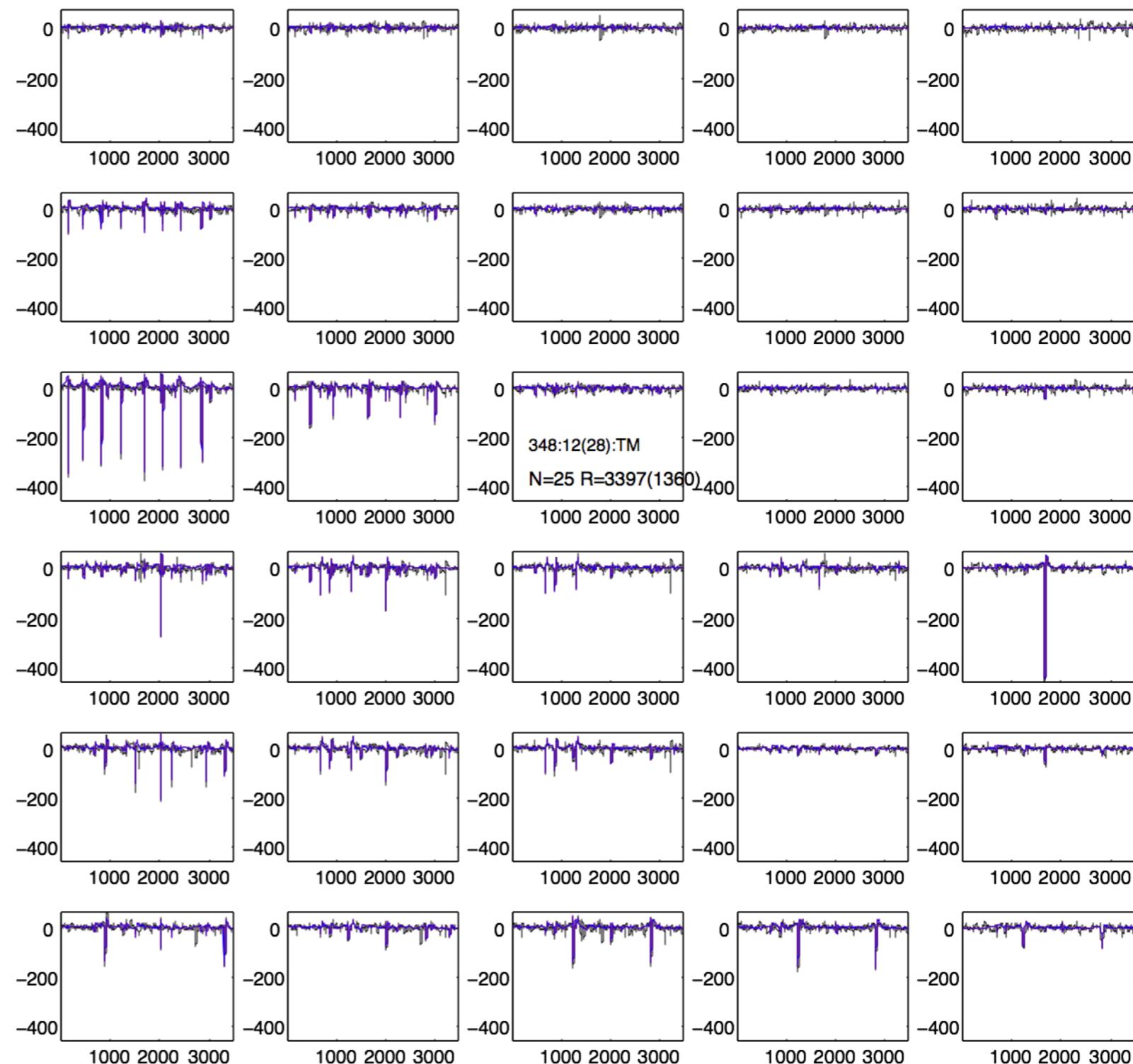
The **spike-sorting problem** is:  
Given raw data like these, convert  
to a list of discrete events (which  
cells fired at what times).

# Not-so-simple events:

Many complex events -- multiple overlapping spikes in many locations. And of course these may be the **most interesting ones!**

Mis-identification of overlapping spikes will lead to artificial correlations, errors in receptive field measurement etc.

When we graduate to very large arrays, **nearly all** events will involve overlaps in space & time!!



# Spike Sorting Desiderata:

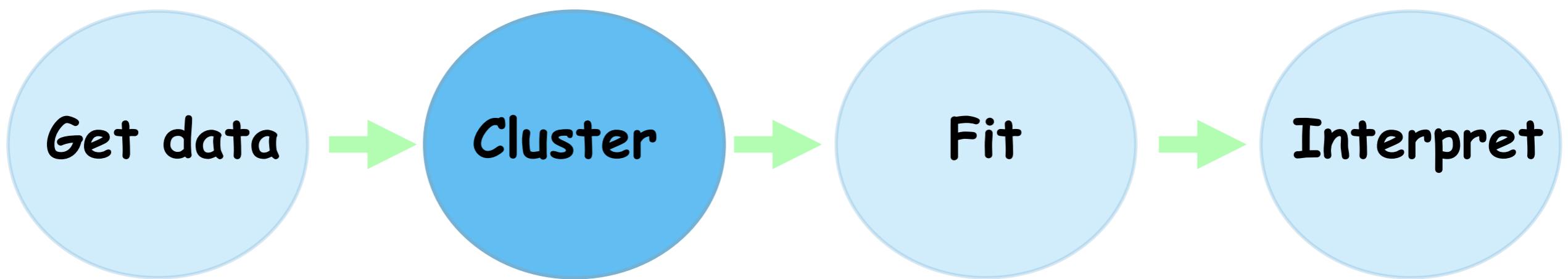
Be **FAST**. Maybe fast enough for on-the-fly diagnosis and repositioning of tissue.

Be **SCALABLE**, i.e. able to handle big arrays. Classic literature focused on 1-4 electrodes.

Be able to disassemble **OVERLAPPING SPIKES** and **BURSTS**.

**EASY HUMAN INTERVENTION**: Allow easy human scrutiny and correction -- no machine learning algorithm to date matches the judgement of expert humans.

1. Experiment
2. Clustering
3. Fitting
4. Performance



# Existing hands-on approach:

One natural choice of features: peak amplitudes in each of the channels.

One current method: consider all pairs of features ( $30 \times 29/2$  for 30 electrodes), view all events projected to each of these planes, find projections in which clusters are apparent, and manually cut until all clusters are separated.

**That's too much work for a human!** Especially when it later becomes  $1000 \times 999/2$  pairs for 1000 electrodes...

# Opportunities:

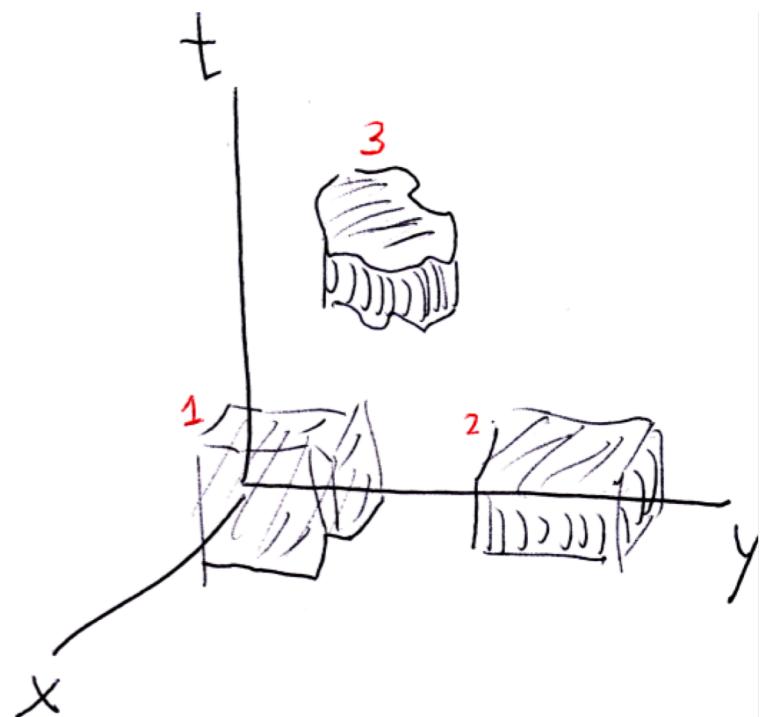
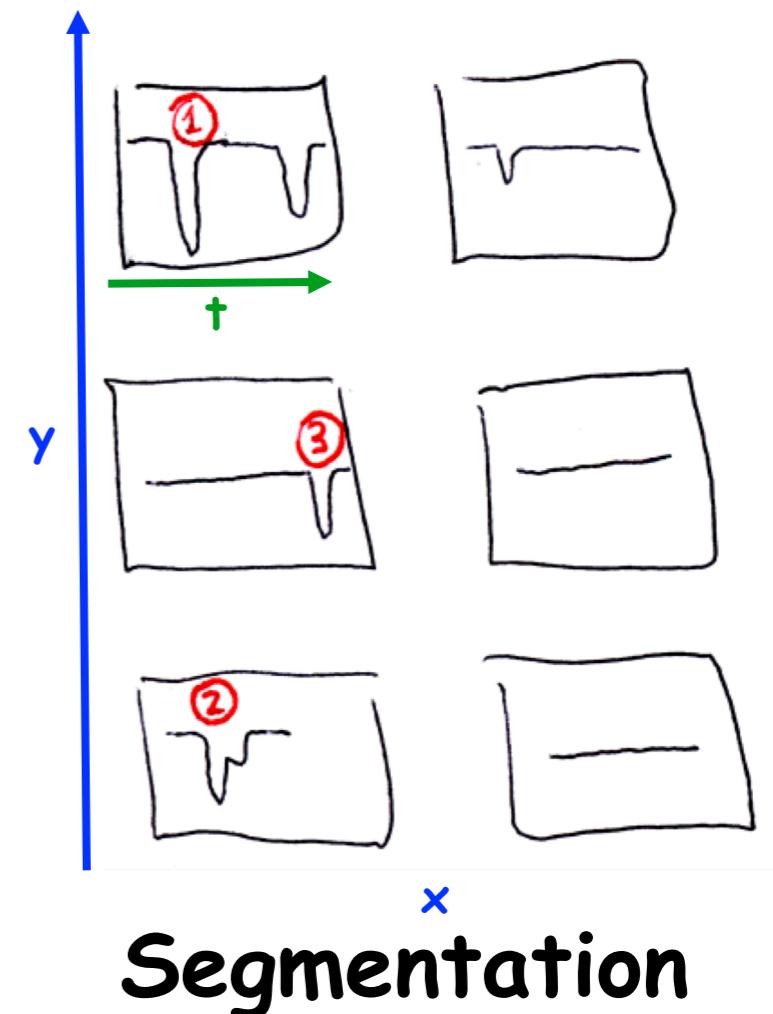
Each neuron has a **fixed spatial location**.

There are **not many overlaps in space and time**

The signals from different **neurons combine by linear superposition**.

Spikes from any given neuron **vary significantly in amplitude**, but not much else.

Most other **noise (variability) is pretty universal**, i.e. largely independent of the neuron

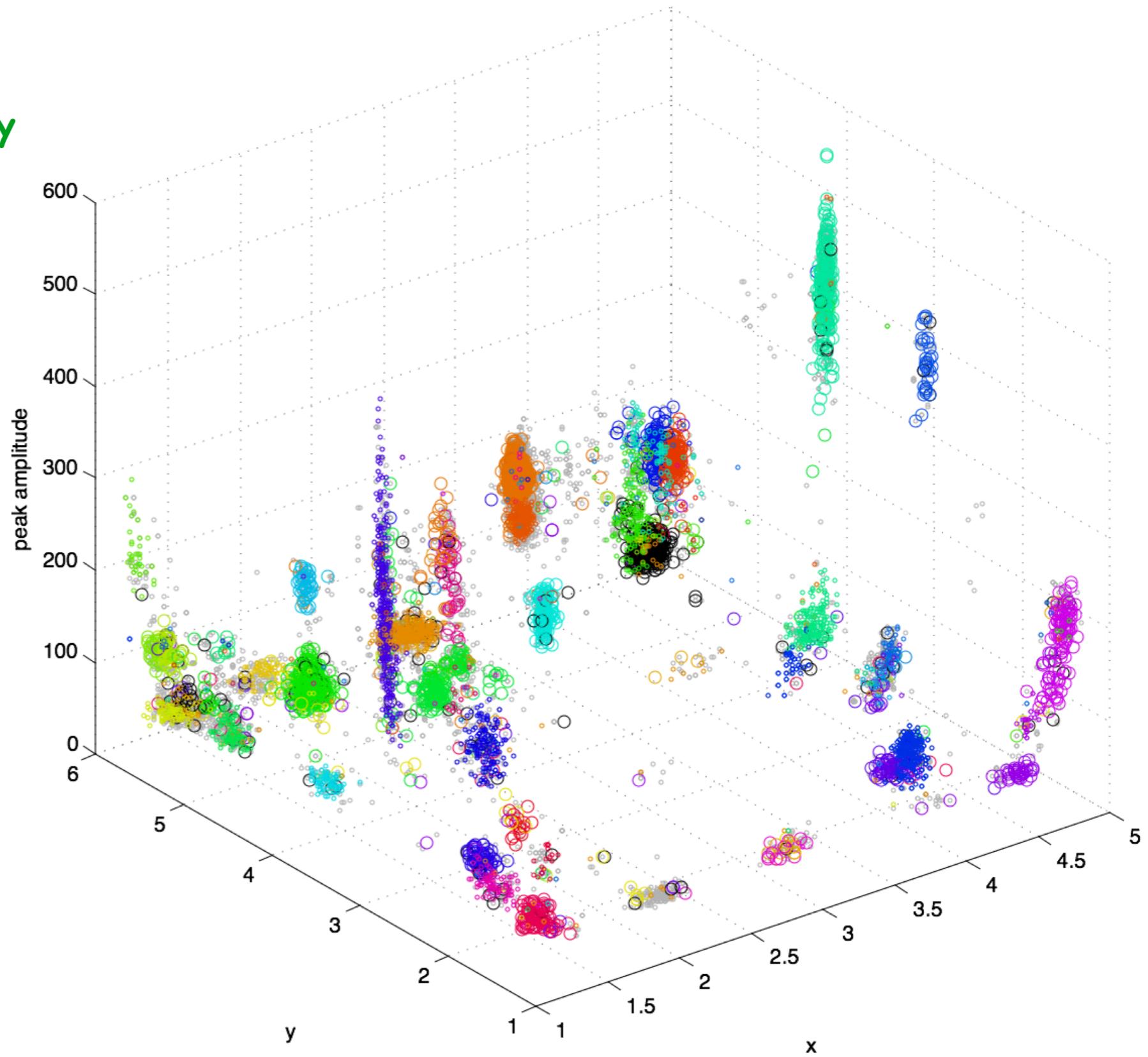


# Geographical clustering:

Feature 1: weighted average xy location for each event.

Feature 2: amplitude on the highest-amplitude channel

Cells have significantly different amplitude variability. We'll need to give each cell its own **amplitude profile** (i.e. prior probabilities of amplitudes).



# OPTICS algorithm:

We don't know a priori which are the best features to cluster on.

## Idea of OPTICS

1. Use entire waveform as our feature space.
2. Instead of directly seeking clusters, try to place points in a helpful linear sequence, one that makes it fast and easy for a human to cut into clusters.

Choose a point (spiking event). Move to its nearest neighbor, repeat to get a chain. Now simply show the actual waveforms in this sequence.

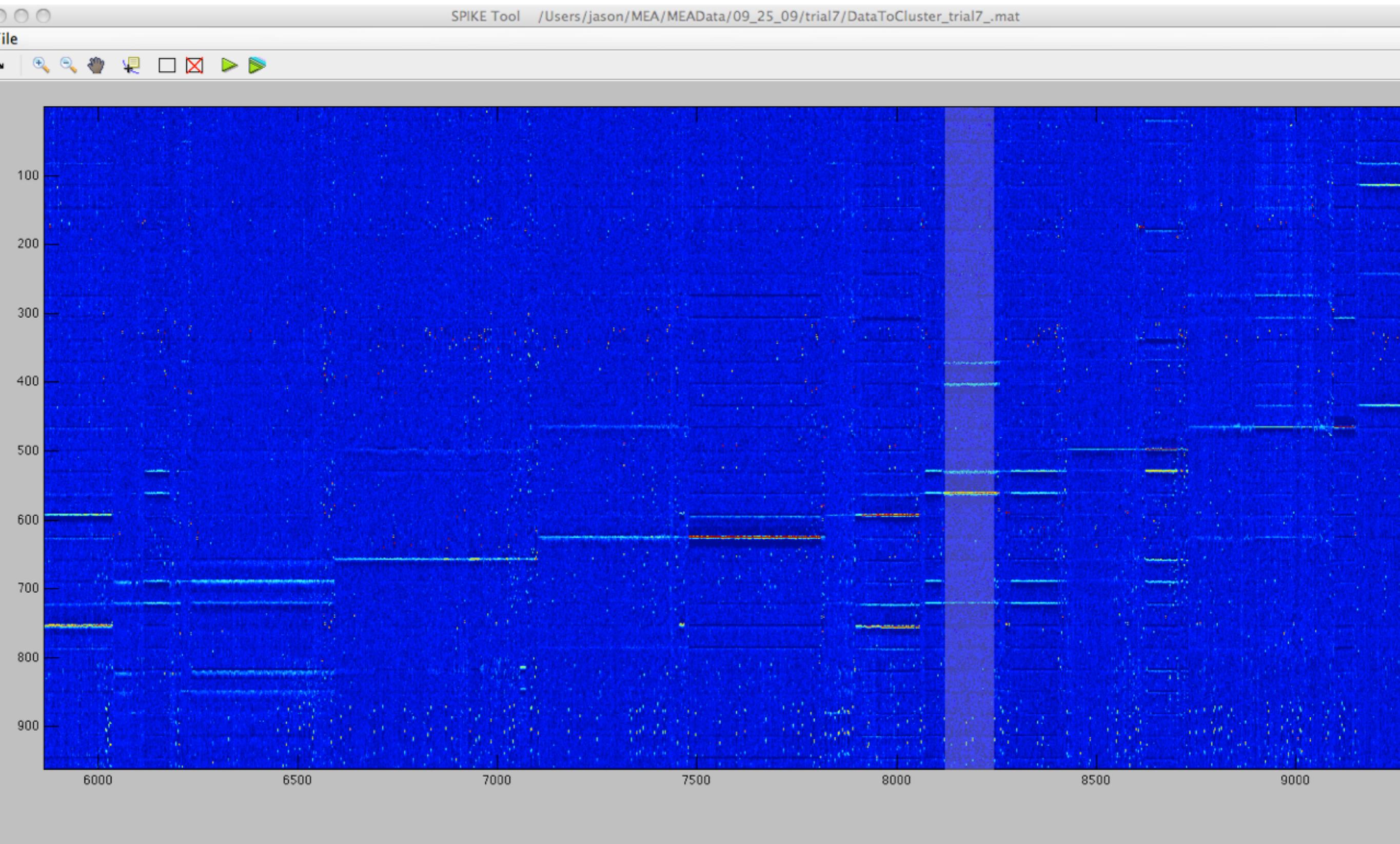
## Advantages

No assumptions on cluster shape and dimension. Easy and fast cluster cutting.

**OPTICS: Ordering Points To Identify the Clustering Structure**

*M. Ankerst, M. Breunig, H.-P. Kriegel, and J. Sander. 1999.*

GUI lets user clip out a cluster, then view the resulting average waveform as a movie.

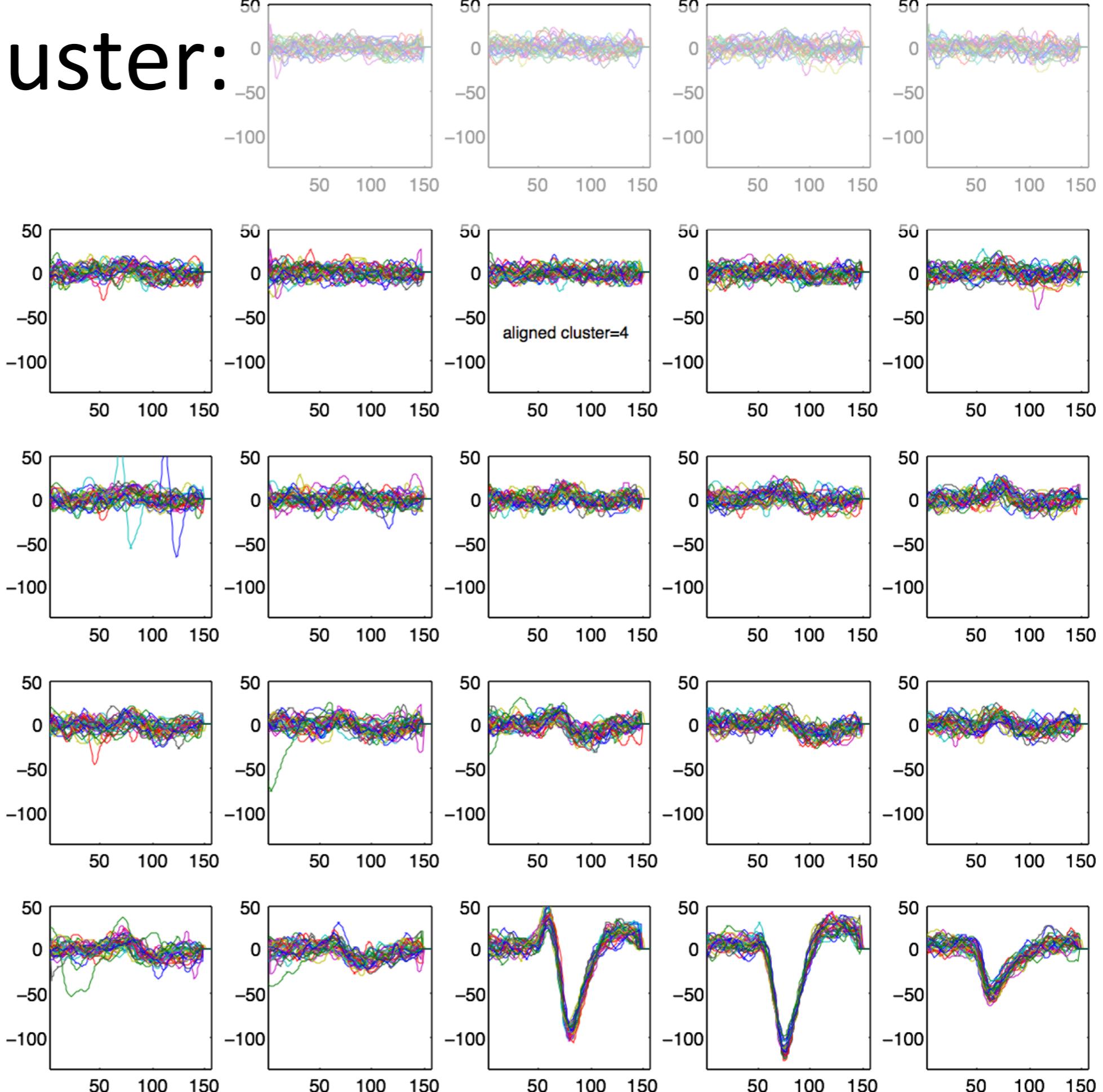


Jan Homann, Jason Prentice

# Typical cluster:

Superposing 50 traces chosen from 284 in this cluster shows that they really do all resemble each other.

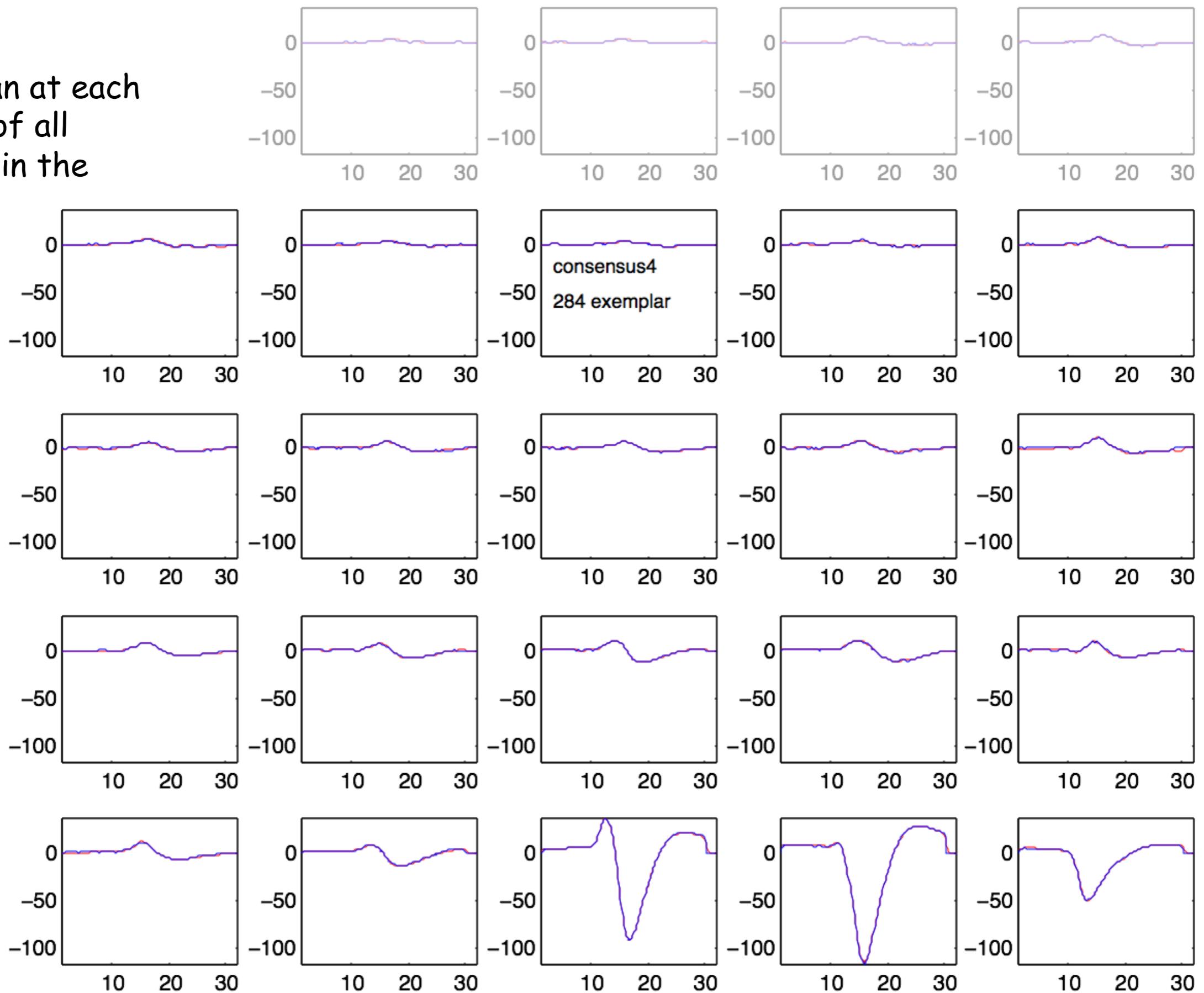
Average trace =  
“archetype waveform”



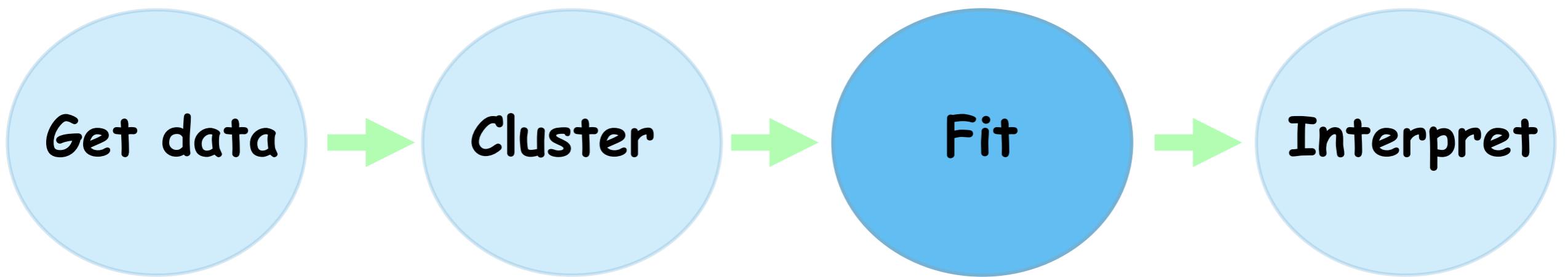
# *Resulting archetype waveform:*

**Blue:** Median at each time point of all waveforms in the cluster.

**Red:** Mean.



1. Experiment
2. Clustering
- 3. Fitting**
4. Performance



# Fit ideas:

We found archetypal waveforms ("templates"), representing single neurons.  
Now we want to fit these templates to the the data

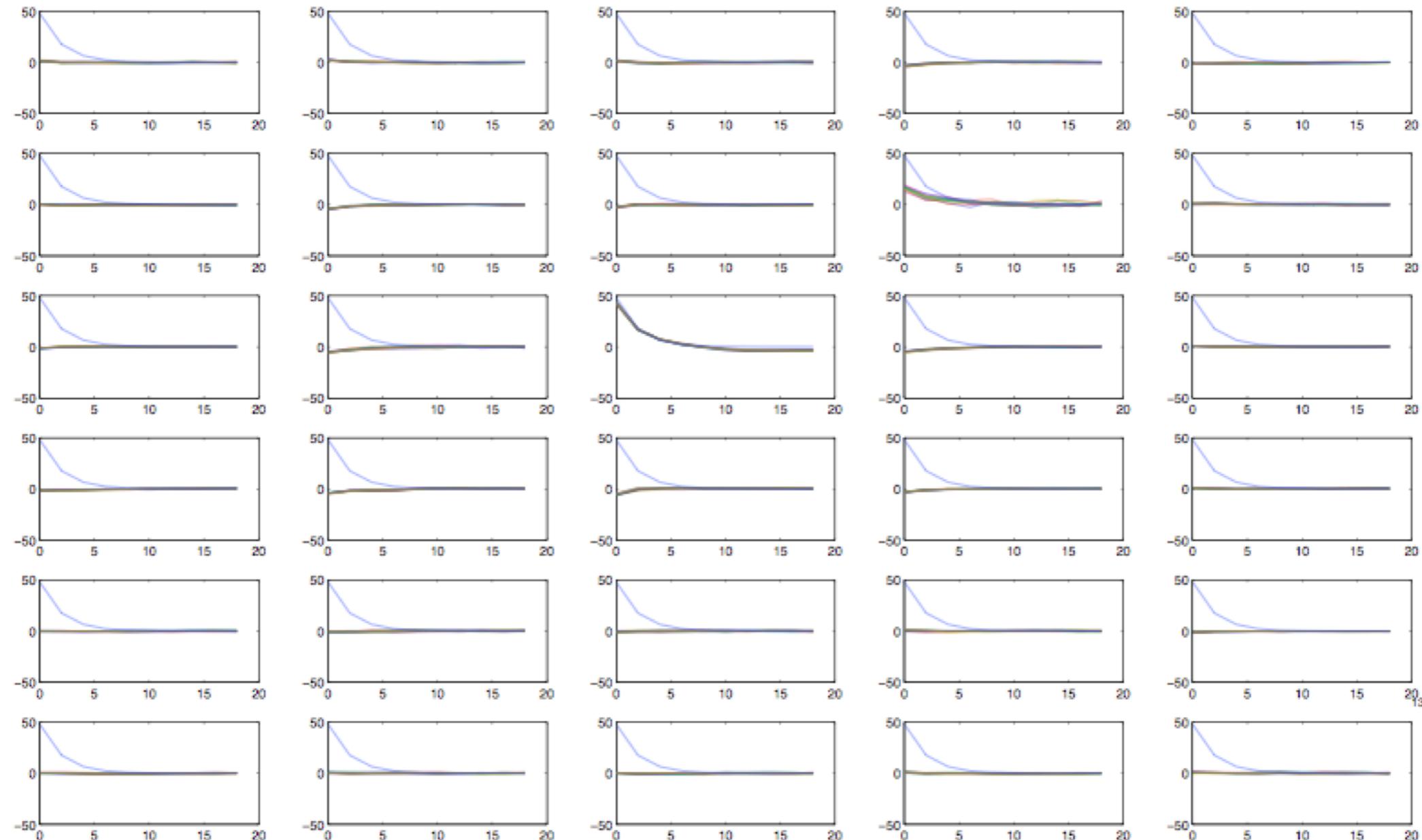
## ISSUES:

Spike variability, different for each neuron: assume this is largely in overall amplitude, and roughly Gaussian

Noise: electrical pickup, amplifier noise, unresolved activity of distant neurons, assume Gaussian and measure directly from segments without spikes. [Pouzat et. al. 2002].

# Noise covariance:

E.g., covariance of channel #13 with all other channels. (blue curve = an exponential function for comparison. Here channel 13 is mostly correlated with itself.



# Bayesian inference of spikes:

$$P(\text{spikes}|\text{data}) = P(\text{data}|\text{spikes}) \frac{P(\text{spikes})}{P(\text{data})}$$

**Spikes** = list of spike times for each cell

**Data** = observed waveforms on each electrode

**P(spikes)** = Prior probability of spikes at specific times with observed amplitudes

**P(data)** = Prior probability of the data. This is the same for any inferred set of spike times, so you can drop it.

$$P(\text{spikes}|\text{data}) \propto P(\text{data}|\text{spikes}) P(\text{spikes})$$

# Bayesian inference of spikes:

$$P(\text{spikes}|\text{data}) \propto P(\text{data}|\text{spikes}) P(\text{spikes})$$

Prior  $P(\text{spikes})$ : assume that for a single spike it has the form

$$P^{\text{cell}}(\mu) P^{\text{time}}(t) P^{\text{ampl}}(A|\mu)$$

$P^{\text{cell}}$  = popularity of this neuron,  $P^{\text{time}}$  = uniform in time,  $P^{\text{ampl}}$  = Gaussian distribution of amplitude. Get these from the data subset used in clustering.

Likelihood function  $P(\text{data} | \text{spikes})$ : assume that the data consist of the archetype, plus Gaussian noise measured from noise covariance.  
[Pouzat et. al. 2002]

Could have sophisticated priors reflecting cell-cell dependencies. Keeping things simple makes an algorithm FAST.

# Nuts and Bolts:

Start with a firing event and find its peak and find the spike (cell) with the **maximum likelihood** for the event.

Once we've found our best fit, we **subtract** the properly shifted and rescaled template from the data and **start over**.

[Segev, Goodhouse Puchalla & Berry, Nature Neuroscience \(2004\).](#)

Continue until either (a) the likelihood ratio for adding a spike is below threshold, or (b) the entire waveform's amplitude is below a voltage threshold.

1. Experiment
2. Clustering
3. Fitting
- 4. Performance**

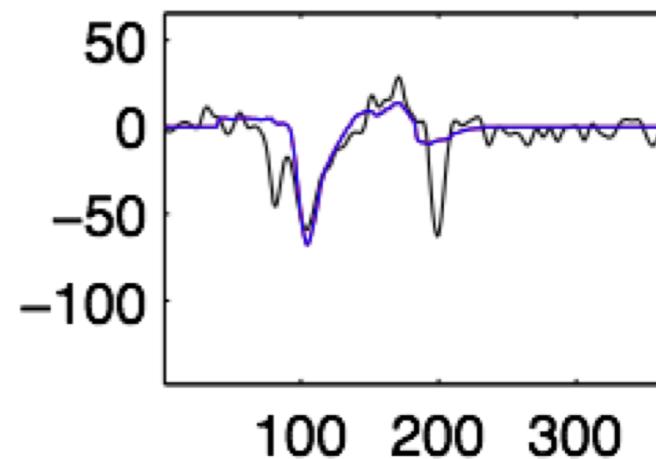
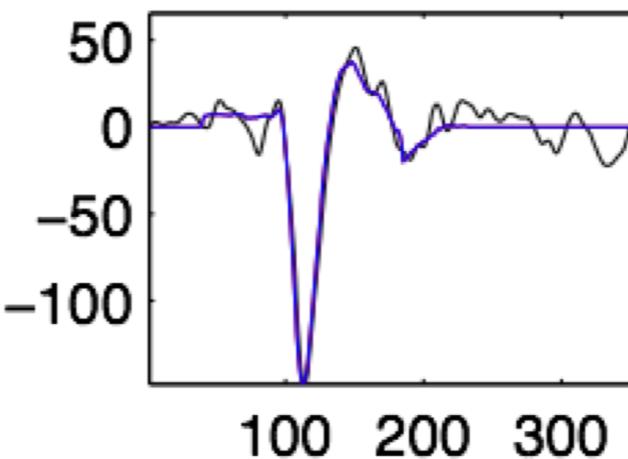
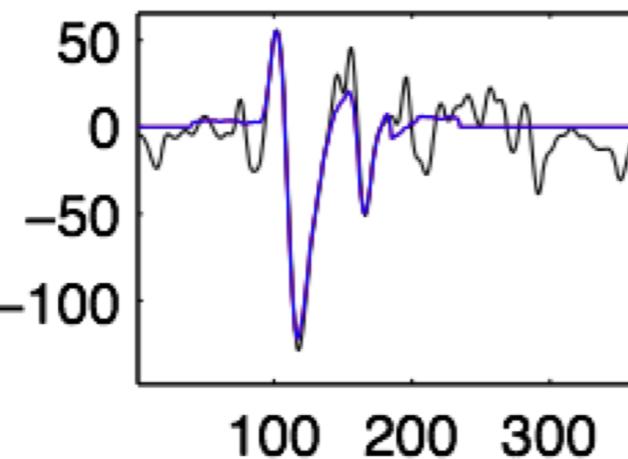
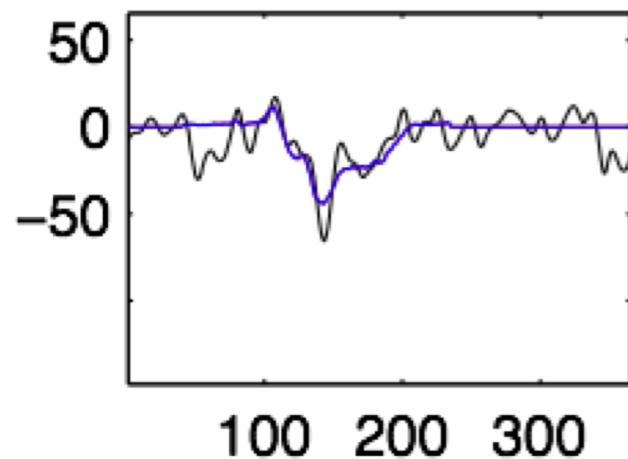
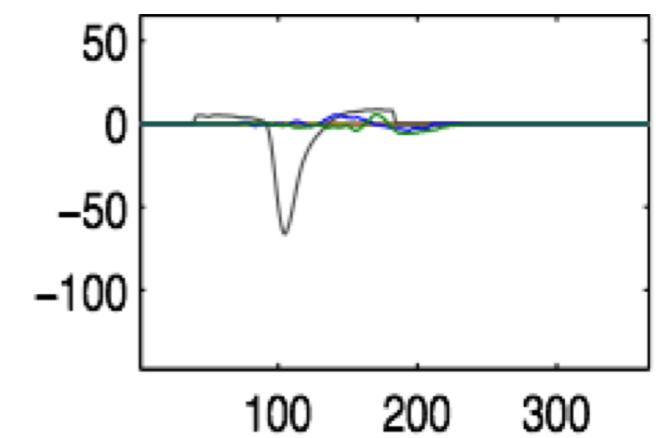
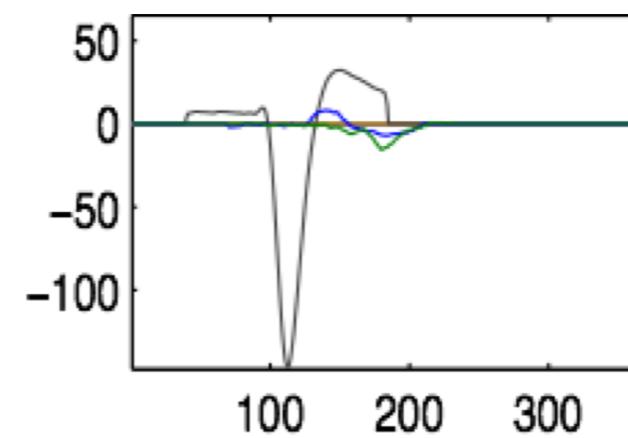
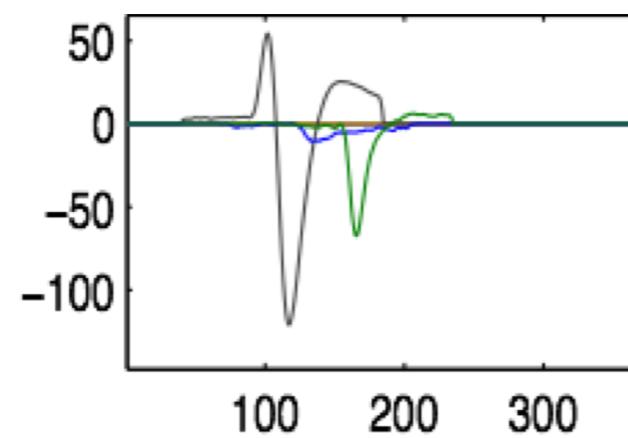
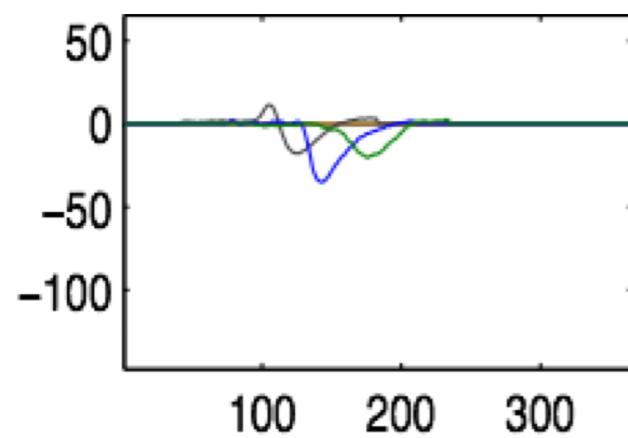
# Speedy:

The algorithm we have described (from Nelson and Balasubramanian) spends only about 30ms of 3-year old laptop time per fit spike.  
...and they have hardly begun to optimize it!

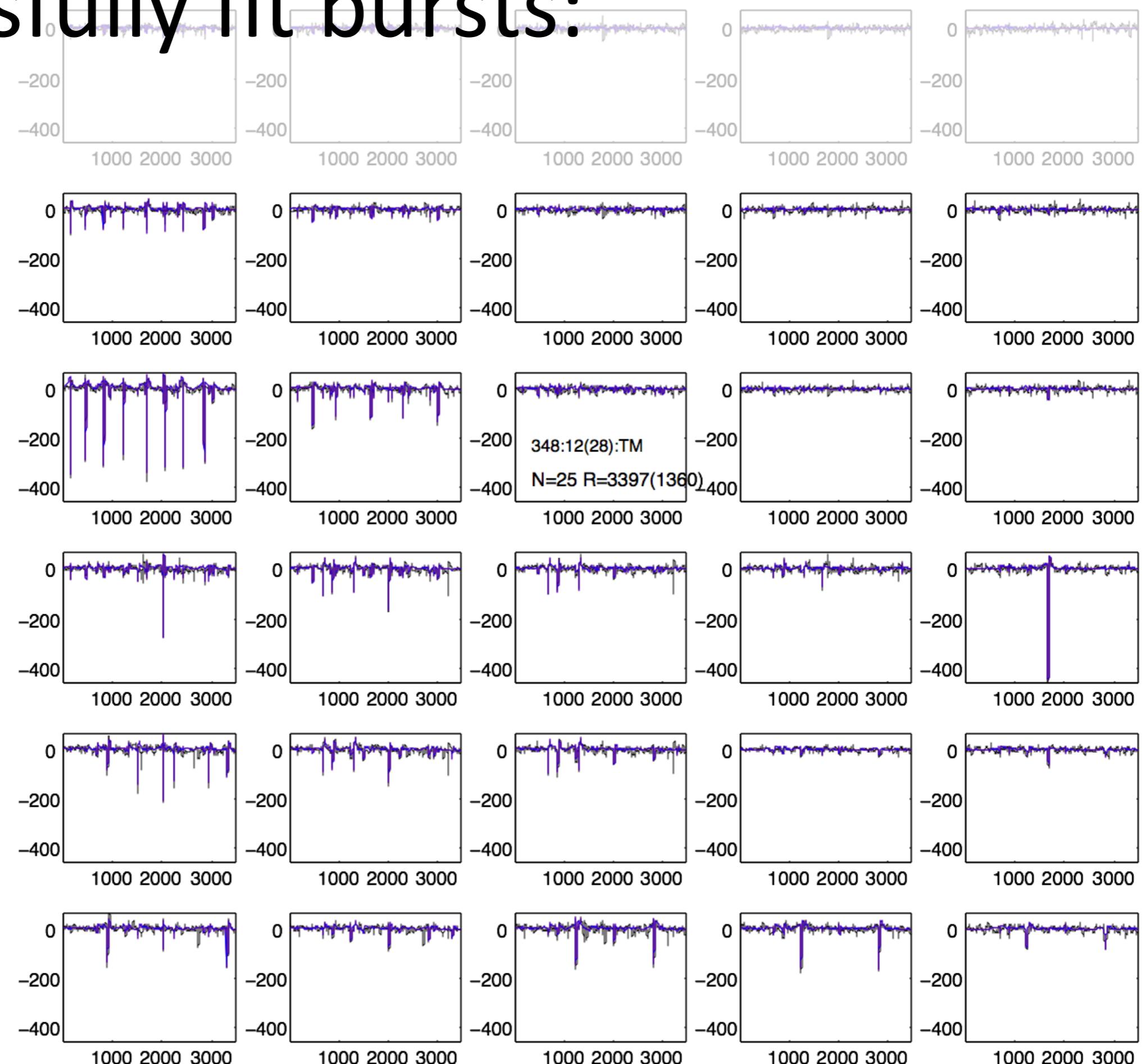
# Successfully fit overlaps:

Top: Closeup of four channels, showing three fit archetypes found by the algorithm.

Bottom: sum of those fits (color) versus actual data (black).



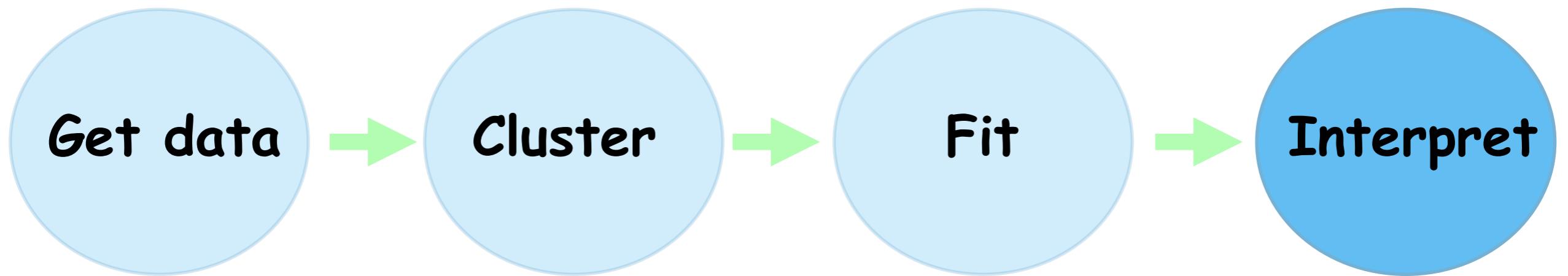
# Successfully fit bursts:



# Few refractory violations:

A neuron cannot fire twice in less than about 1.5 ms. If our fitting algorithm claims that a neuron did that, it has made a mistake. The algorithm does very well.

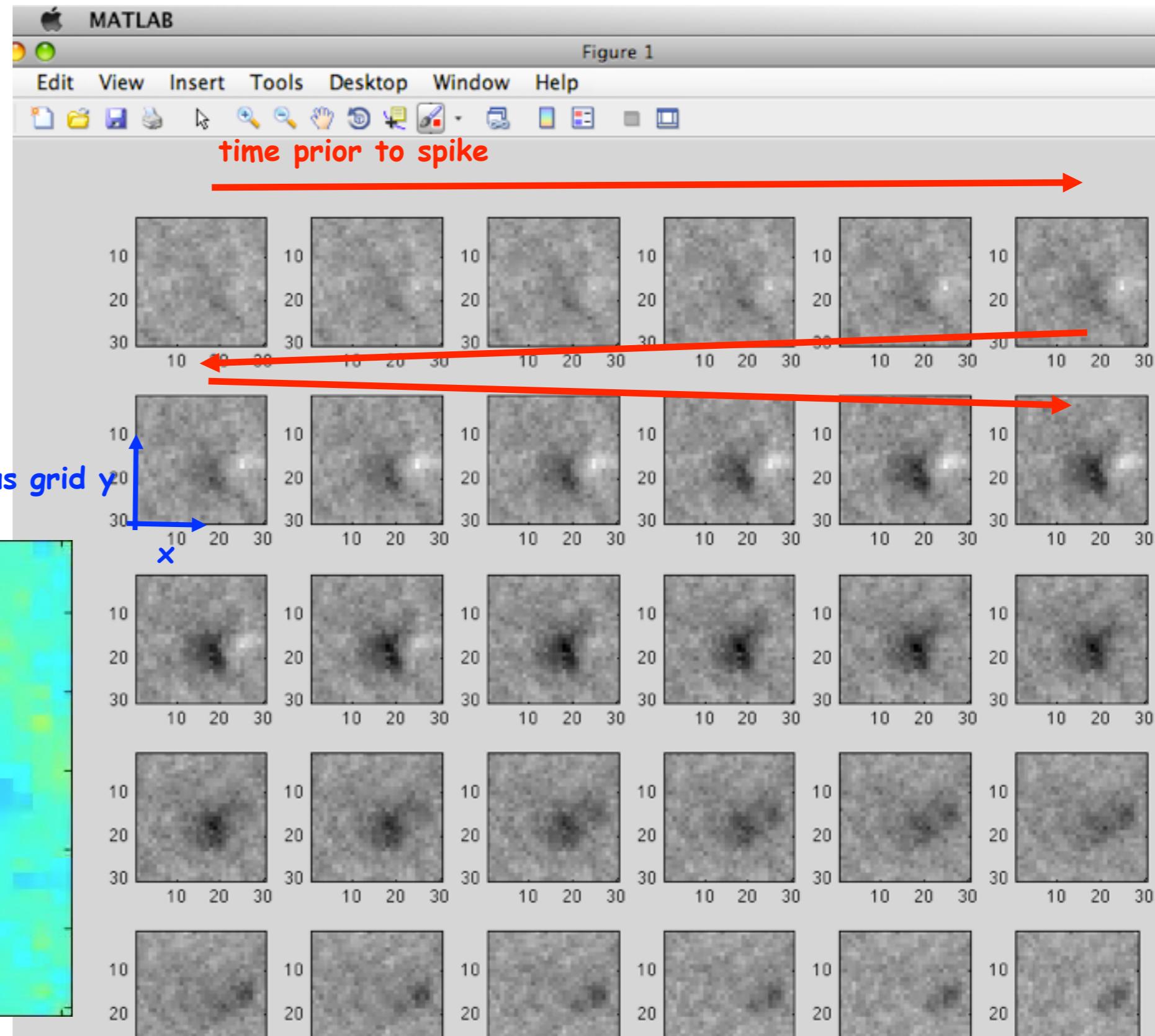
E.g. Of 49 found archetypes, 40 had fewer than 0.1% refractory violations. (7 had more than 1%.) This was not enforced by hand.



# Receptive fields:

Spatio-temporal  
Receptive Field (STRF).

MEA recording is high  
throughput:  
We get dozens of cells at  
once, can be hundreds.



# Receptive fields:

... and they tile the visual field. Here: simultaneously recorded RF centers of ON cells.



# Wrap:

For **experimentalists**: Spike sorting can be done efficiently and effectively and in a principled manner, and you should know what's under the hood.

For **theorists**: spike sorting is an interesting inference problem and lots of fun to think about. It needs to move out of the shadows of the "methods supplements", and the community should merge its best ideas to have a shared standard.

# Roads not taken:

Others have advocated a single, combined step that attempts to get clusters, spike IDs, and even the firing rates and dependences on stimuli all in a single huge optimization problem.

Most neuroscientists prefer a modular approach, in part because that way one can test each module.

Also, if desired one can replace one module but keep the other(s).

# Yet other methods:

MCMC model-switching  
PCA  
HMM  
SVM  
Annealing  
Optimal linear filter  
Wavelet analysis

# Metric used in clustering:

Each “event” from the segmentation analysis is up-sampled by  $\times 5$  to allow more precise alignment.

The metric is roughly Euclidean distance in  $(960 \times 5)$ -dimensional space.

More precisely, each voxel in the event is classified as noise/non-noise.

Only non-noise voxels contribute to the distance, which is then normalized by the number of contributing voxels.

There's also a normalization by the amplitude at the peak.

The main point is that the exact choice of metric isn't so critical, because the data are rich enough to make it easy to separate clusters (recall that 3D plot).

# More checks:

Spike train cross-correlation -- look for refractory violations.

Checked that residuals truly follow the same distribution as noise ("Mahalanobis squared distances are chi-squared distributed").

Prepared fake data, e.g. transpose our archetype waveforms to other spatial locations and sprinkle them, at random times, with added noise, then attempt fit.

Tested for long-term drift in spike shapes, firing rates over the whole experiment.