NEURAL DATA ANALYSIS

ALEXANDER ECKER, PHILIPP BERENS, MATTHIAS BETHGE

COMPUTATIONAL VISION AND NEUROSCIENCE GROUP

NEURAL DATA ANALYSIS

Spike sorting

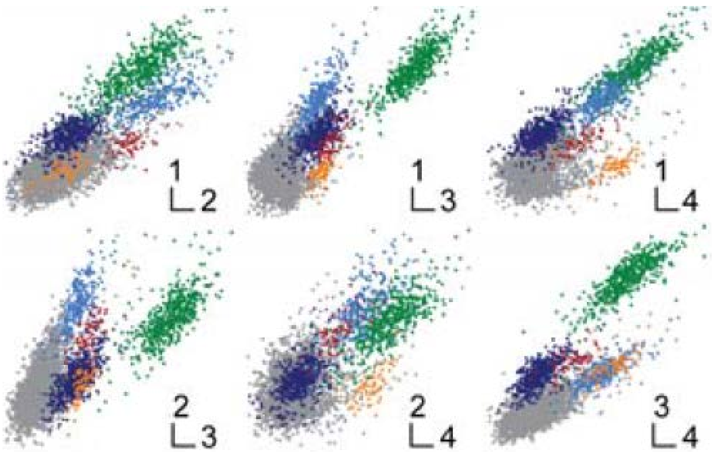
Firing rate estimation

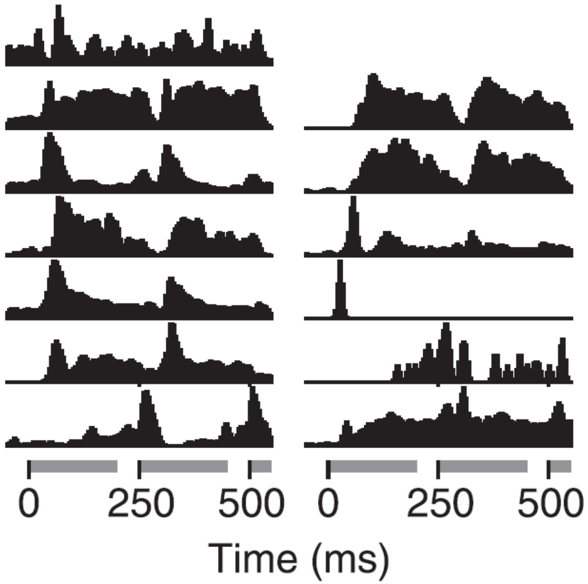
Correlation analysis

Spike pattern modeling

Spike train decoding

Information theory





Brown, Kass, Mitra (2004)

WHY IS THERE AN EXTRA COURSE?

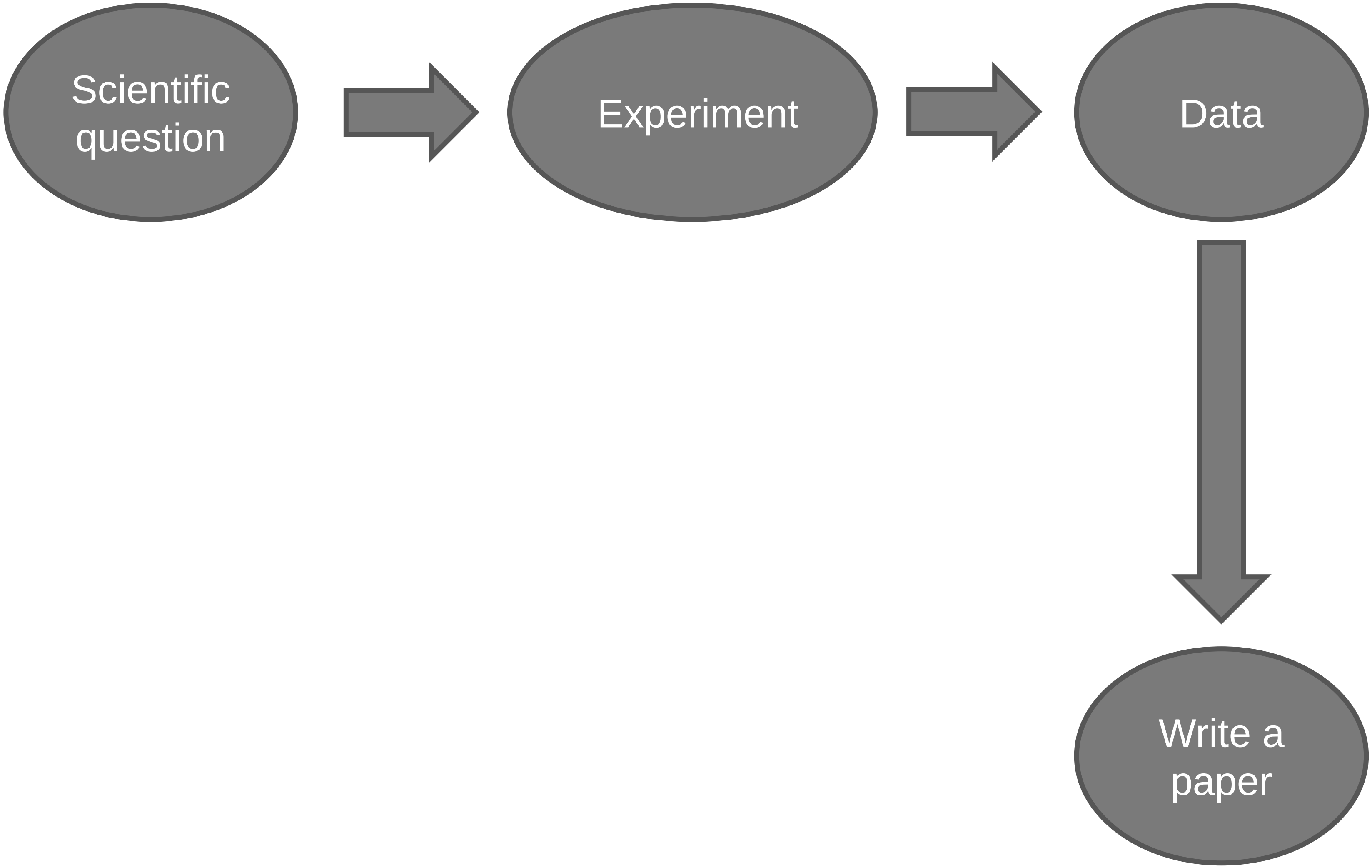
Machine Learning

Signal Processing

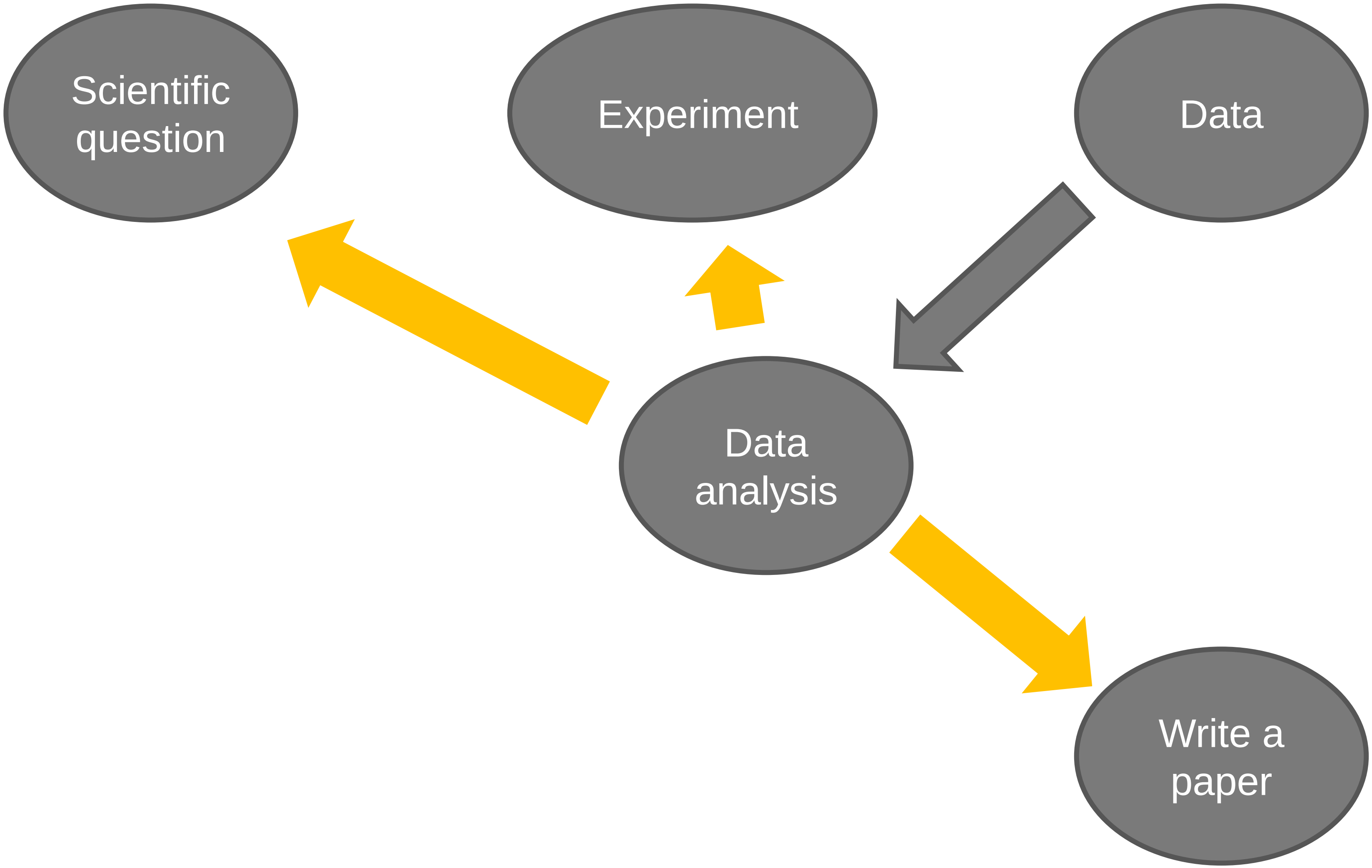
Neural Coding

Scientific Programming

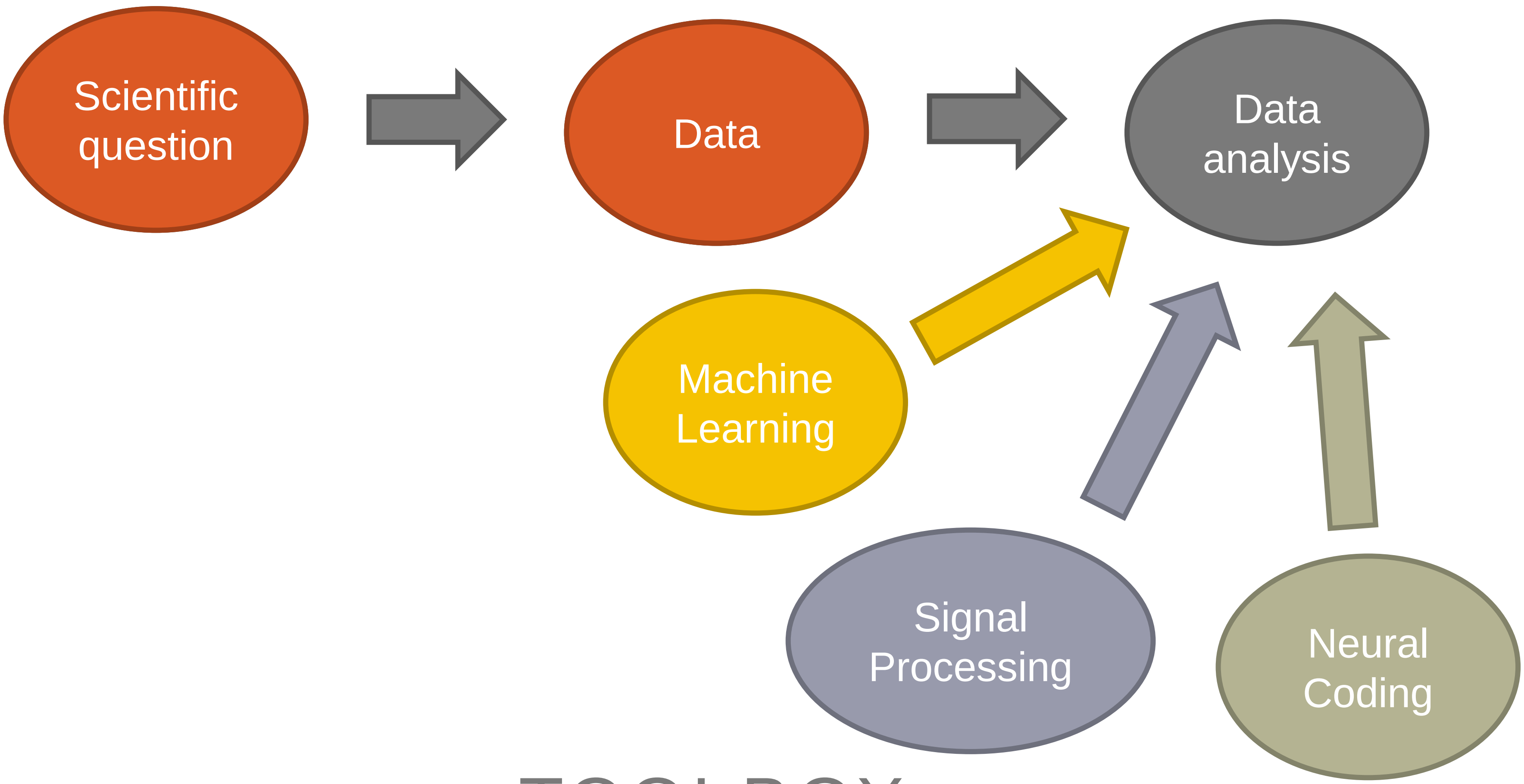
THE EXPERIMENTALISTS IDEAL



A DATA ANALYSIS CENTERED VIEW



HOW DOES DATA ANALYSIS WORK?



TOOLBOX

PHILOSOPHY

Look at examples

Look at raw data

Test algorithms on toy data

Ask yourself „Is this result possible? Does it make sense?“

The problem determines the techniques (not the other way around)

HOW DOES THIS COURSE WORK?

- **Lecture (60 min per week)**
  - Problem statement & possible solutions
  - Algorithmic details and application issues
  - Goal: working knowledge of important techniques
  - Attendance and punctuality mandatory!

- **Programming exercises (60% of grade)**
  - Work on real-world problems and do analysis
  - Implement algorithms discussed in lecture as model solution
  - Practise data analysis philosophy
  - Grading after class

HOW DOES THIS COURSE WORK?

- **Report (20% of grade)**
  - One group writes a detailed report each week
  - Problem statement, solutions, well-prepared figures, results, brief discussion of literature
- **„Current problems“ presentation (20% of grade)**
  - Short papers on current research questions
  - 15 min presentation (5.5. or 30.6.)
- **No exam**

PROGRAMMING EXERCISES

Groups of two students

Due Mondays, 12 PM (before lunch)

1. **Upload**
   Upload a single PDF containing all figures to ILIAS – no late submissions

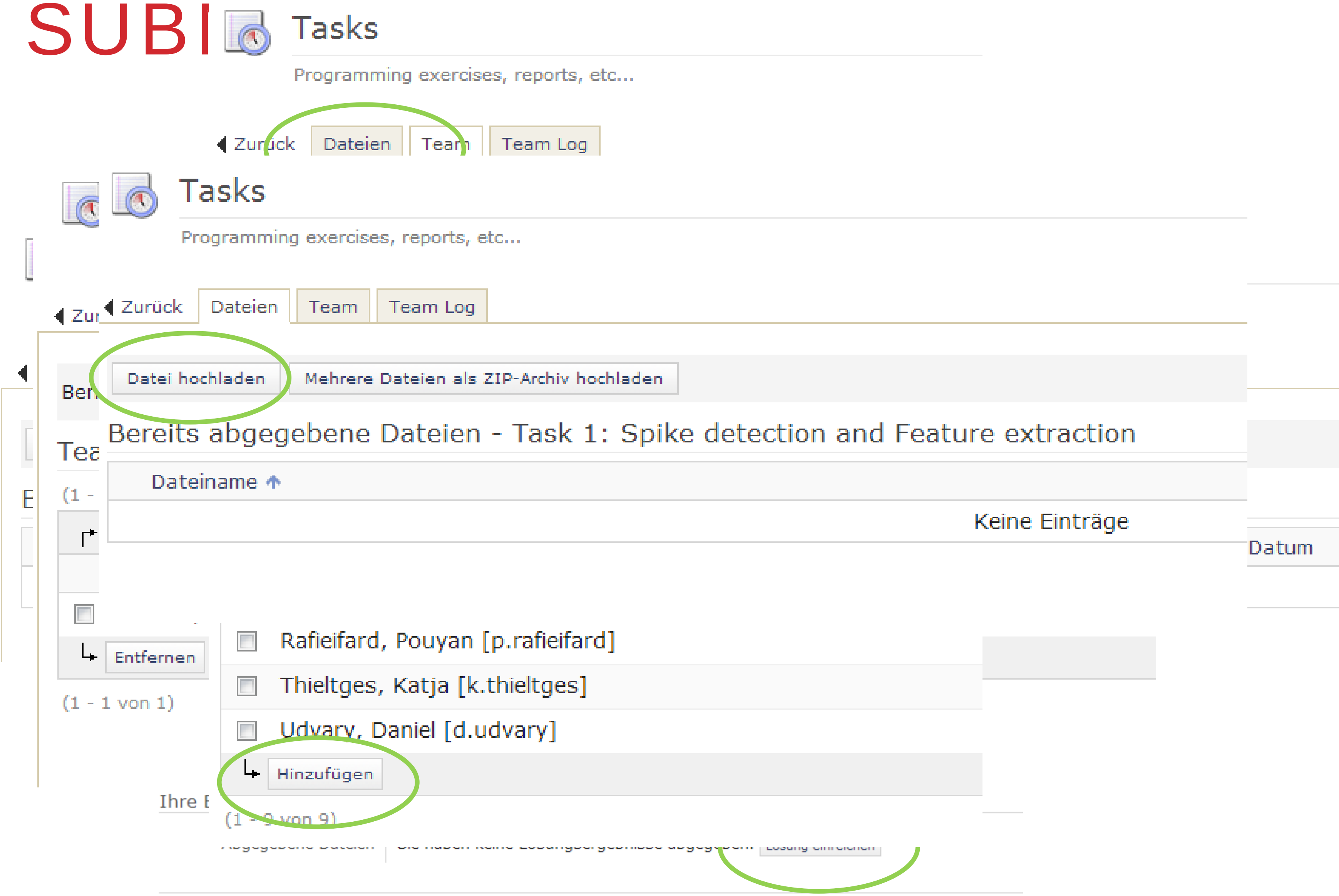
2. **In class demonstration (5-10 min)**
   “Interview” about results and implementation

   You must be able to explain the figures and the code (every student individually)

   If you are not around, it is your responsibility to arrange an alternate date!

   Figures must be ready! (i.e. Laptops switched on, Matlab started, Figures on screen)

SUBI



Tasks
Programming exercises, reports, etc...
Zurück
Dateien
Team
Team Log
Tasks
Programming exercises, reports, etc...
Zurück
Dateien
Team
Team Log
Datei hochladen
Mehrere Dateien als ZIP-Archiv hochladen
Bereits abgegebene Dateien - Task 1: Spike detection and Feature extraction
Dateiname
Keine Einträge
Datum
Entfernen
(1 - 1 von 1)
Rafieifard, Pouyan [p.rafieifard]
Thieltges, Katja [k.thieltges]
Udvary, Daniel [d.udvary]
Hinzufügen
(1 - 9 von 9)

POLICIES

No late submissions.

Use any available material, including ‚built-in‘ MATLAB functions (unless we specifically ask you not to do this).

Talk to others about the problems, but

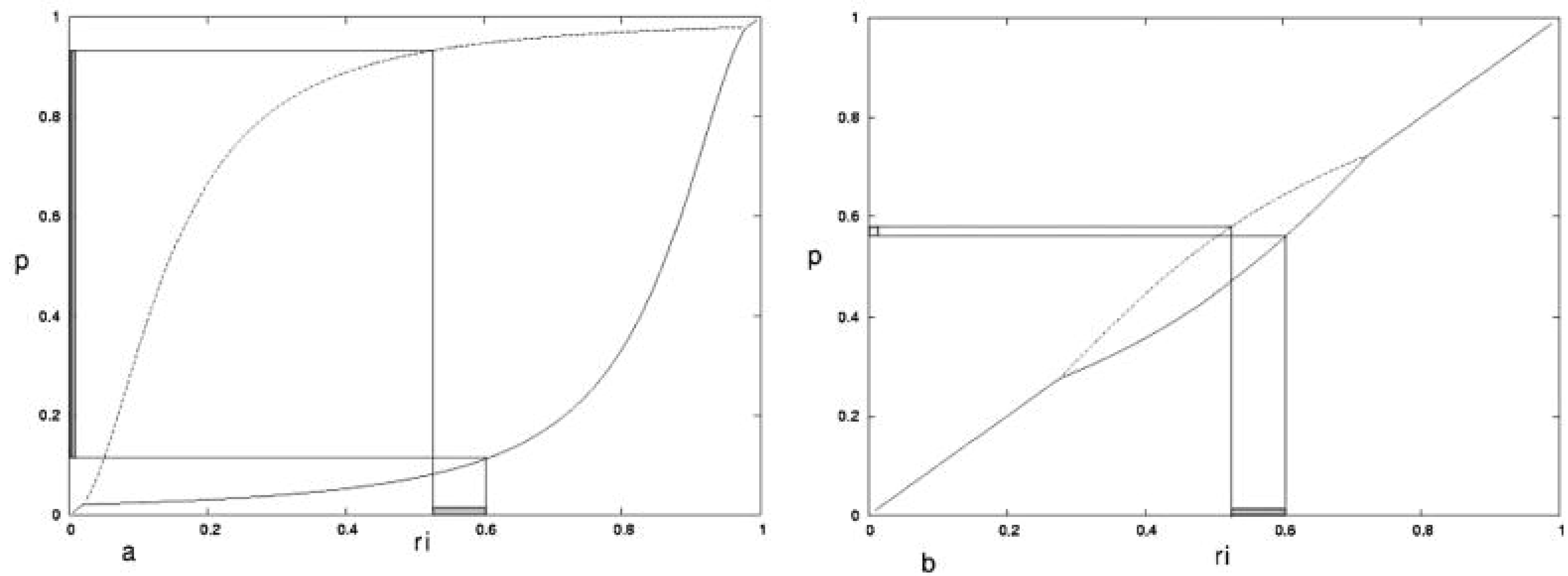
- credit any source in the code
- write your own code.

Work as a team, provided

- you disclose who did what
- every group member can explain each part.

If you are not happy with any part of the course, talk to us.

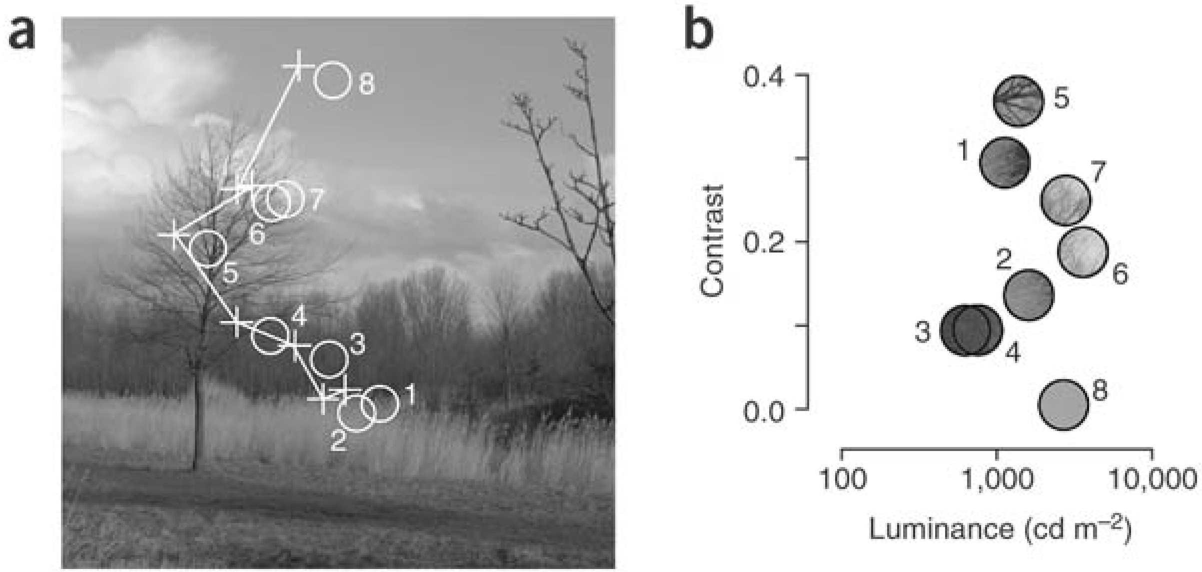
WHAT MAKES GOOD FIGURES?



Solutions of the inequalities 6.5 as a function of for C = 0.02 (a) and C = 0.2 (b). The solid lines show the limits of the solution (i.e., at equality) for the left inequality; the dashed lines are the equivalent for the right inequality. Acceptable p values are all those between the two curves, that is, above the full lines and below the dashed lines. For a chosen range of mean rates (shaded on the abscissa), the range of acceptable values (shaded on the ordinate) is significantly smaller for C = 0.2 than for C = 0.02. The spread of mean rates is zero for rates ri that are outside the range where the curves cross.

Niebur et al. (2007)

WHAT MAKES GOOD FIGURES



a. A sequence of fixations in a natural scene. The crosses indicate fixation locations and the circles represent the corresponding locations of an arbitrary receptive field (diameter: 1°). **b.** Enlargements of the image patches falling within the receptive field as a function of their r.m.s. contrast (ordinate) and average luminance (abscissa).

Mante et al. (2005)

READABLE CODE

Keep it simple!

Write as little and clean code as possible.

Comment your code, but don’t overdo it.

```
% load data
load(‘data.mat’)
```

Don’t worry about efficiency. Readability is more important.

Keep to the interface!

SPIKE SORTING

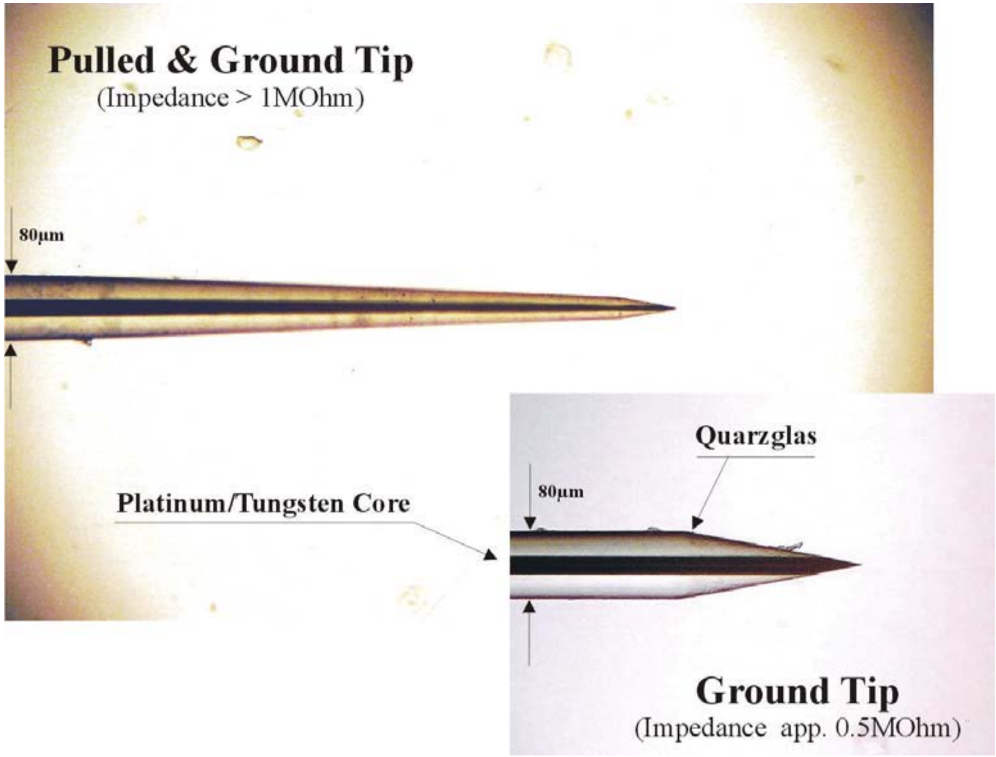
1ST PROBLEM

REVIEW: EXTRACELLULAR RECORDING ELECTRODES

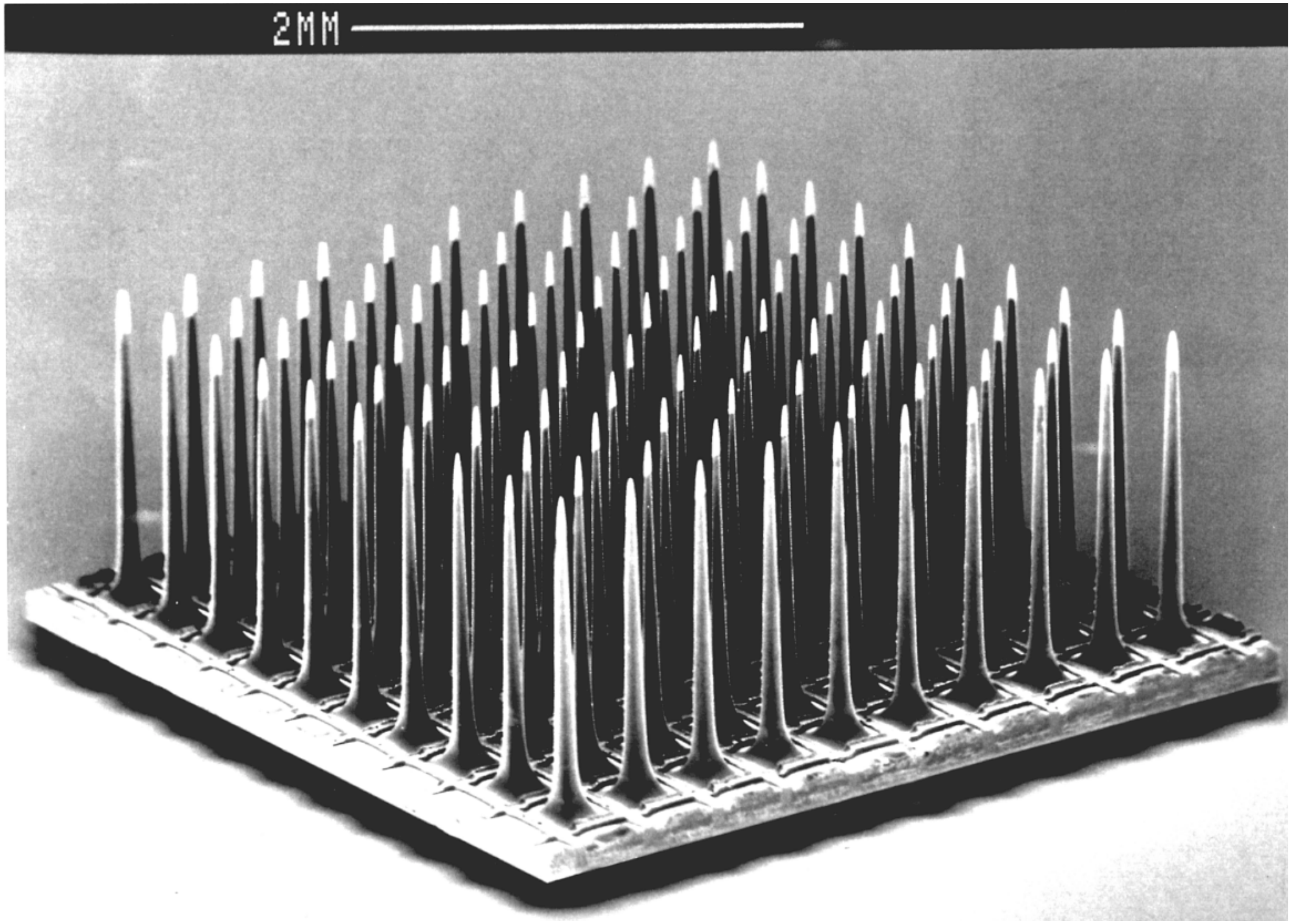
WHICH ONES DO YOU KNOW?

TUNGSTEN MICROELECTRODES

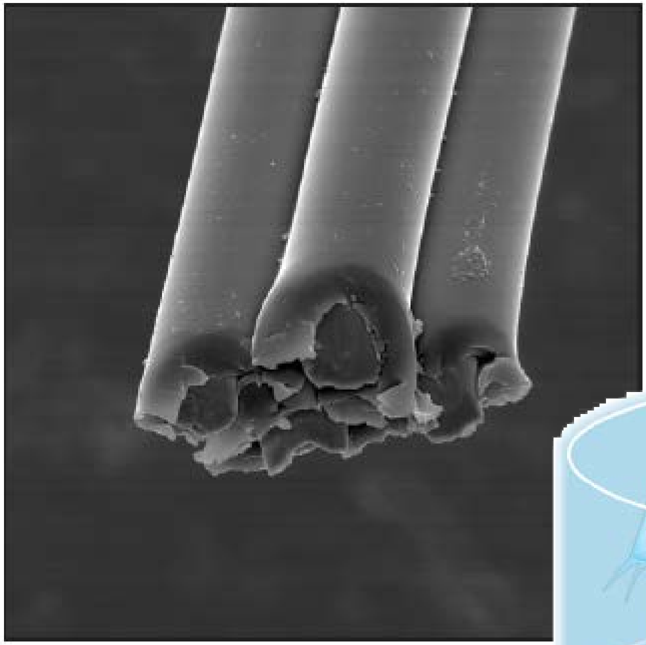
E.g. Thomas Recordings

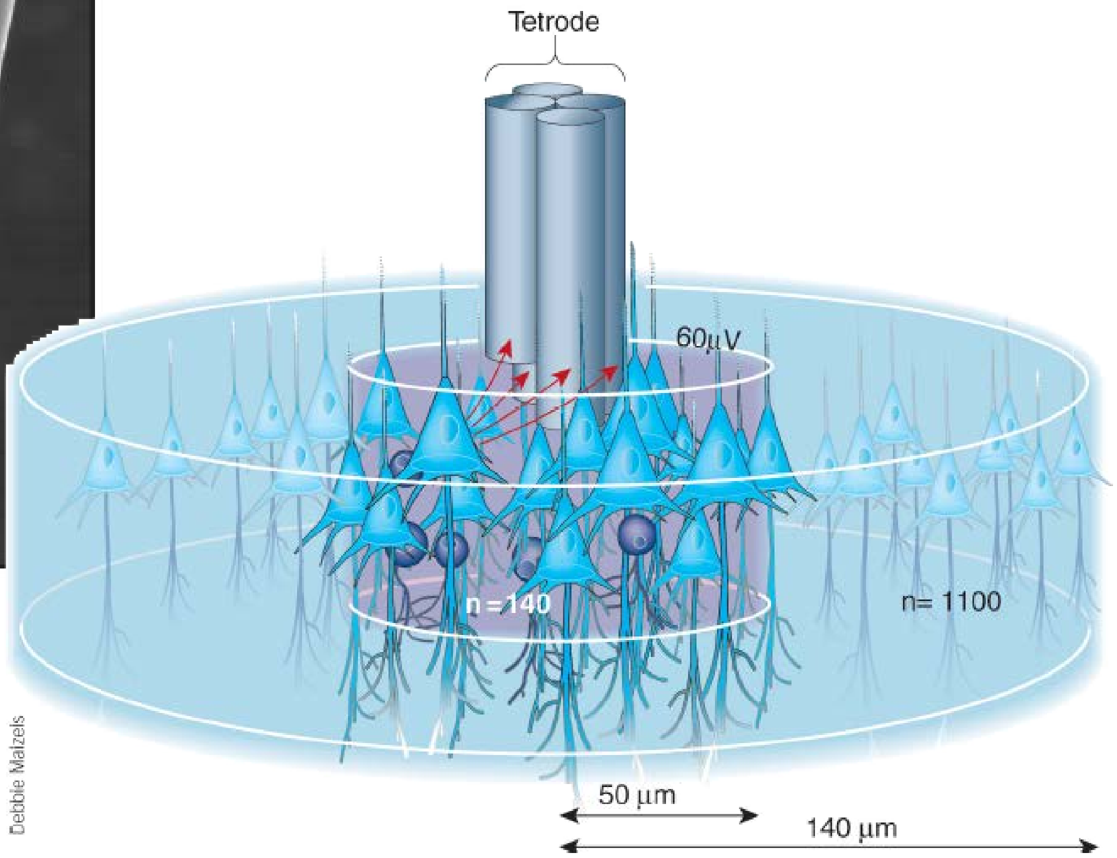


UTAH ARRAY



TETRODES

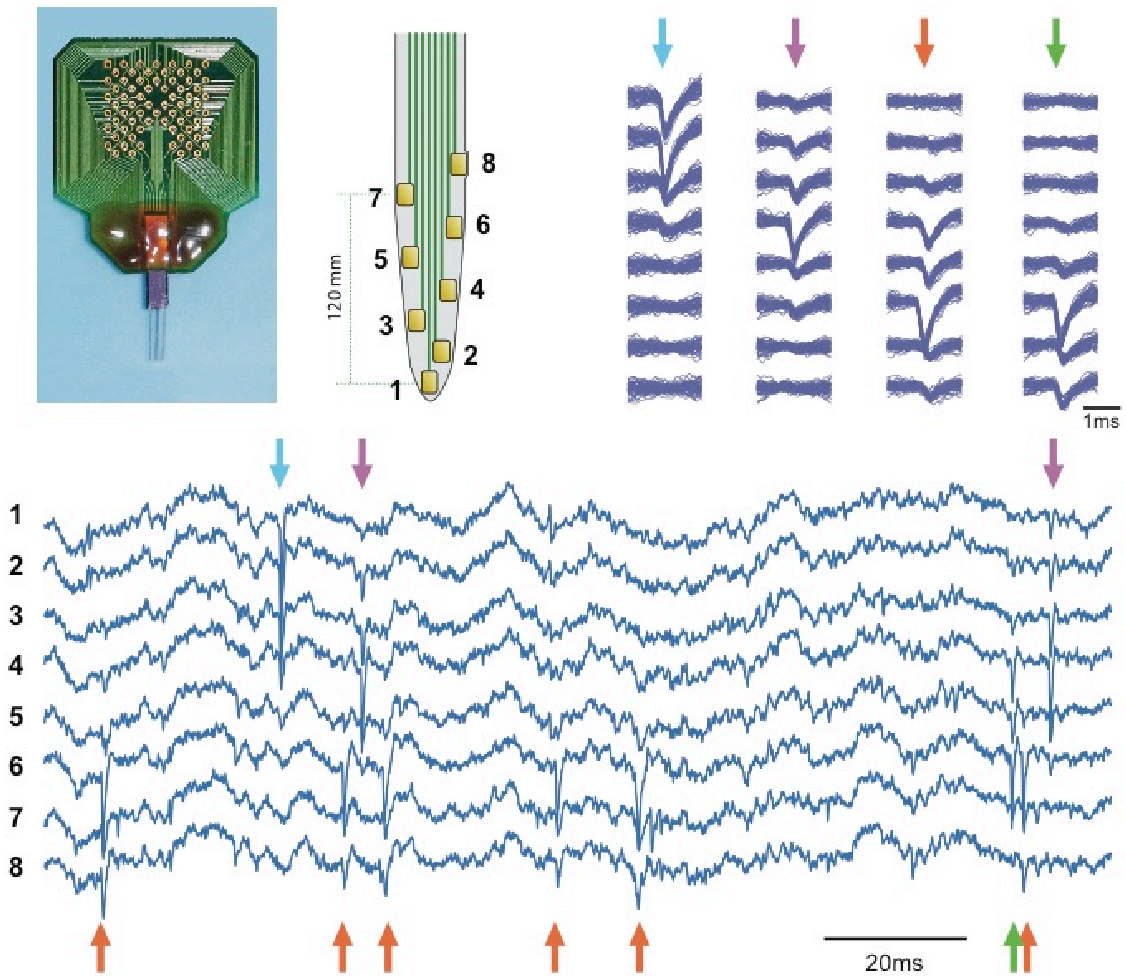


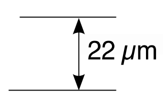


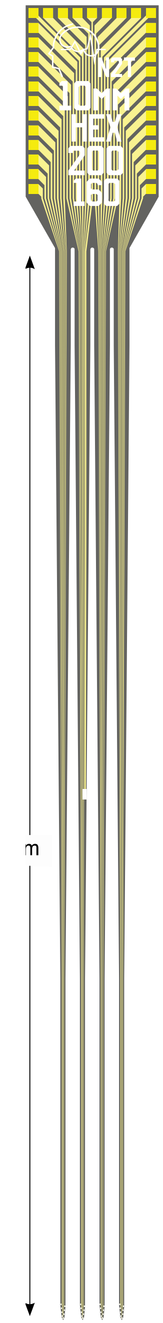
NeuroNexus

Buzsaki 4x8

SILICON PROBES

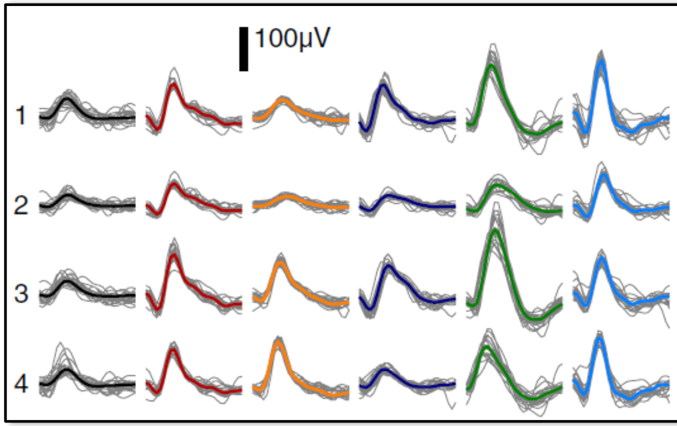


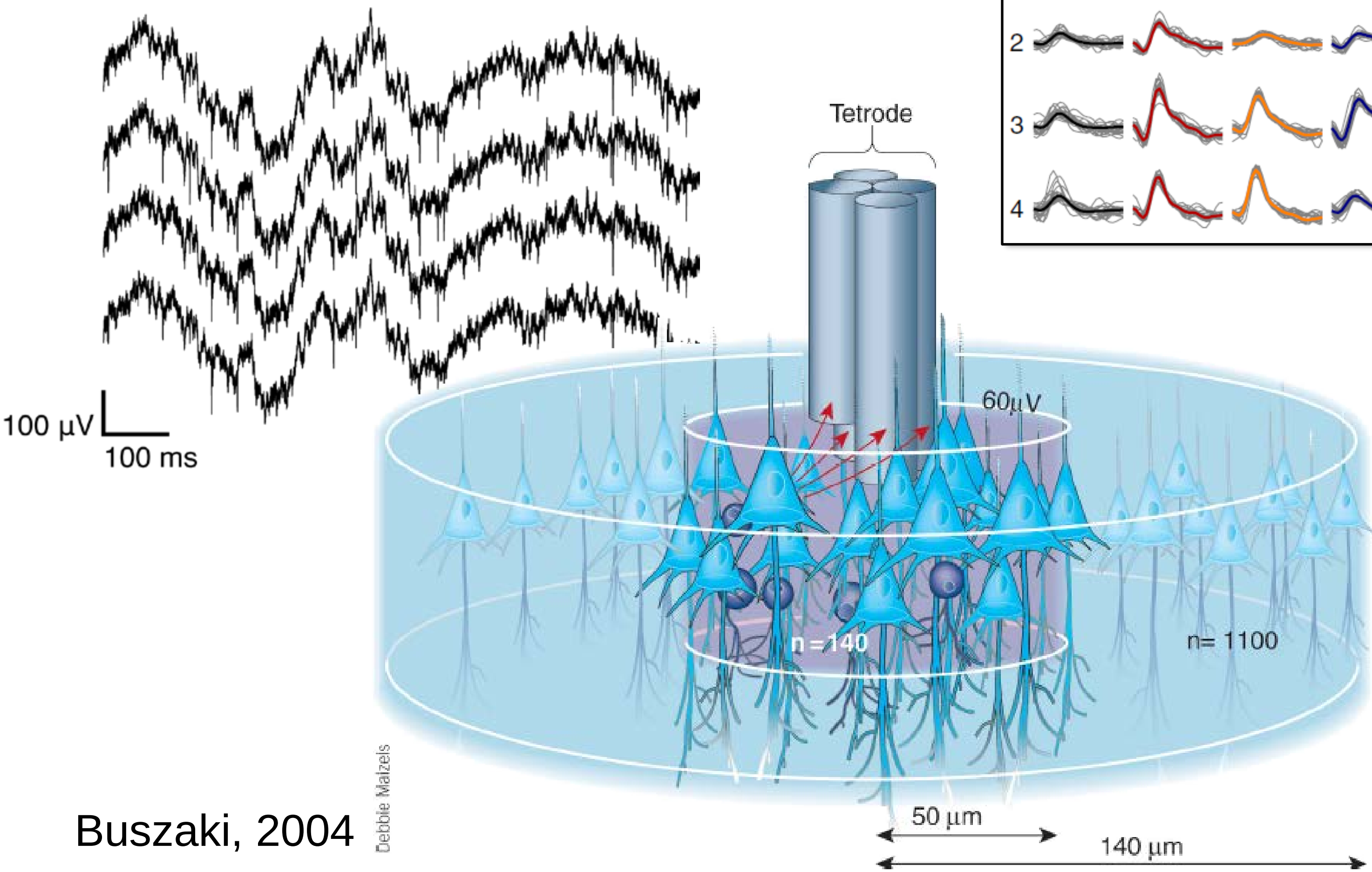




THE PROBLEM

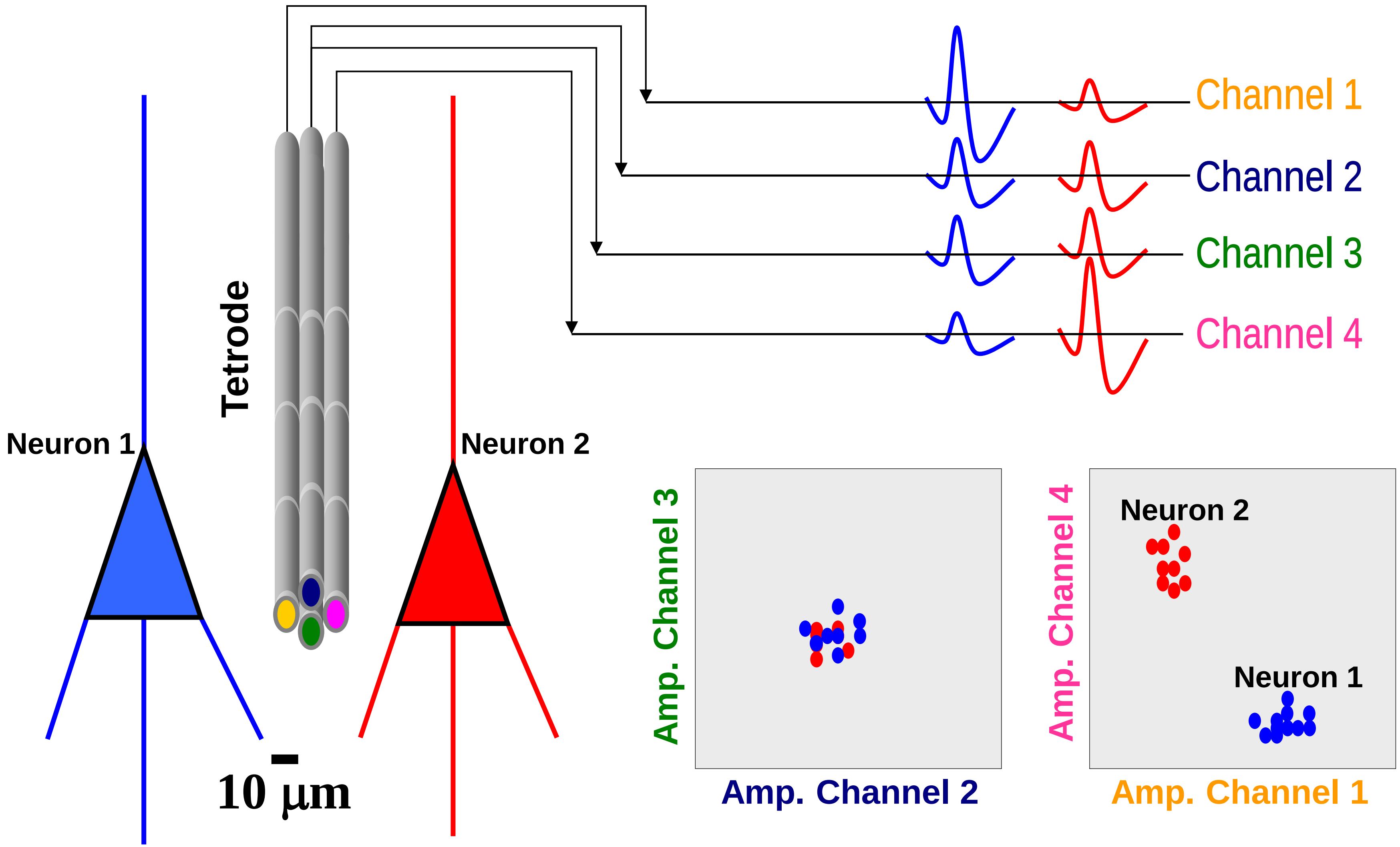
Ecker et al., 2010





Buszaki, 2004

TRIANGULATION



OUTLINE
SPIKE SORTING

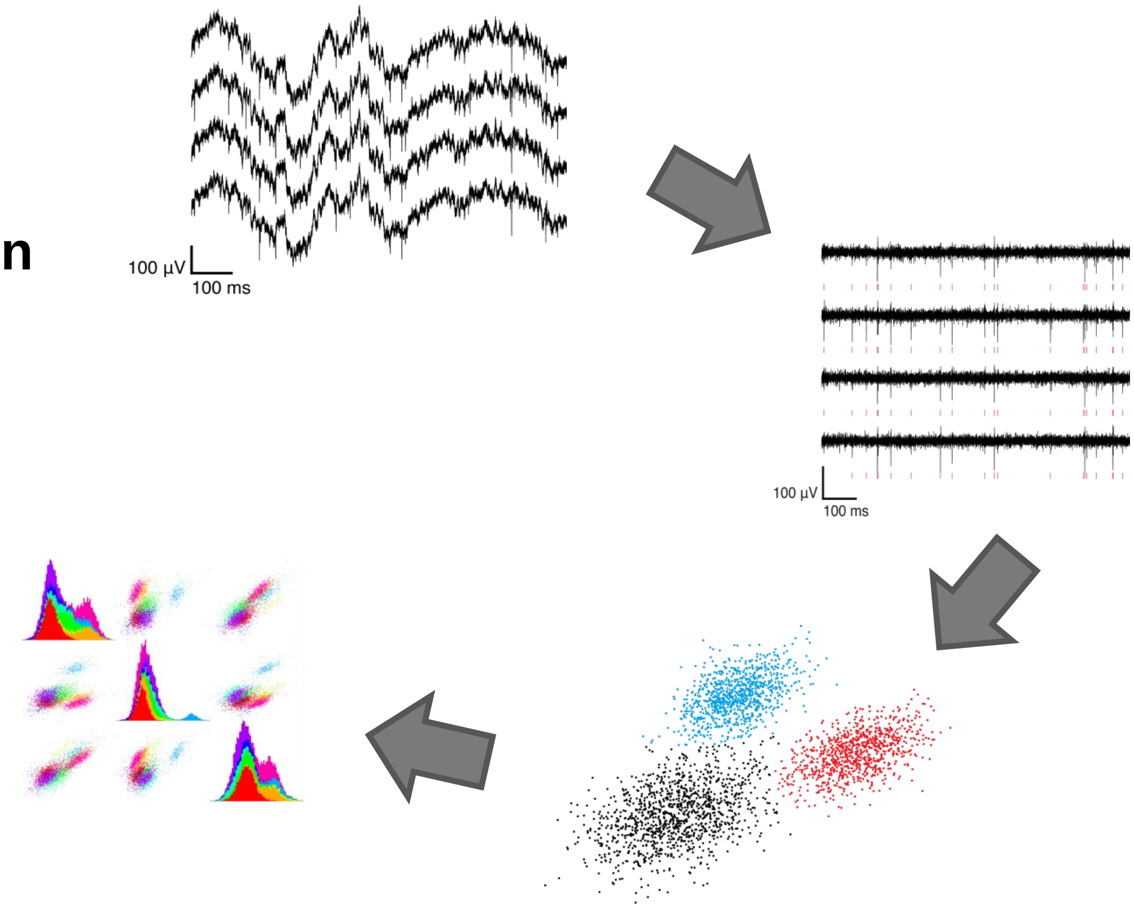
Raw data

Spike detection

Feature extraction

Clustering

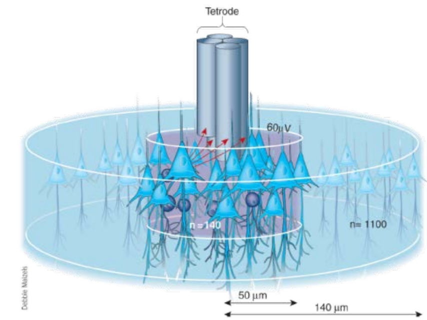
Verification

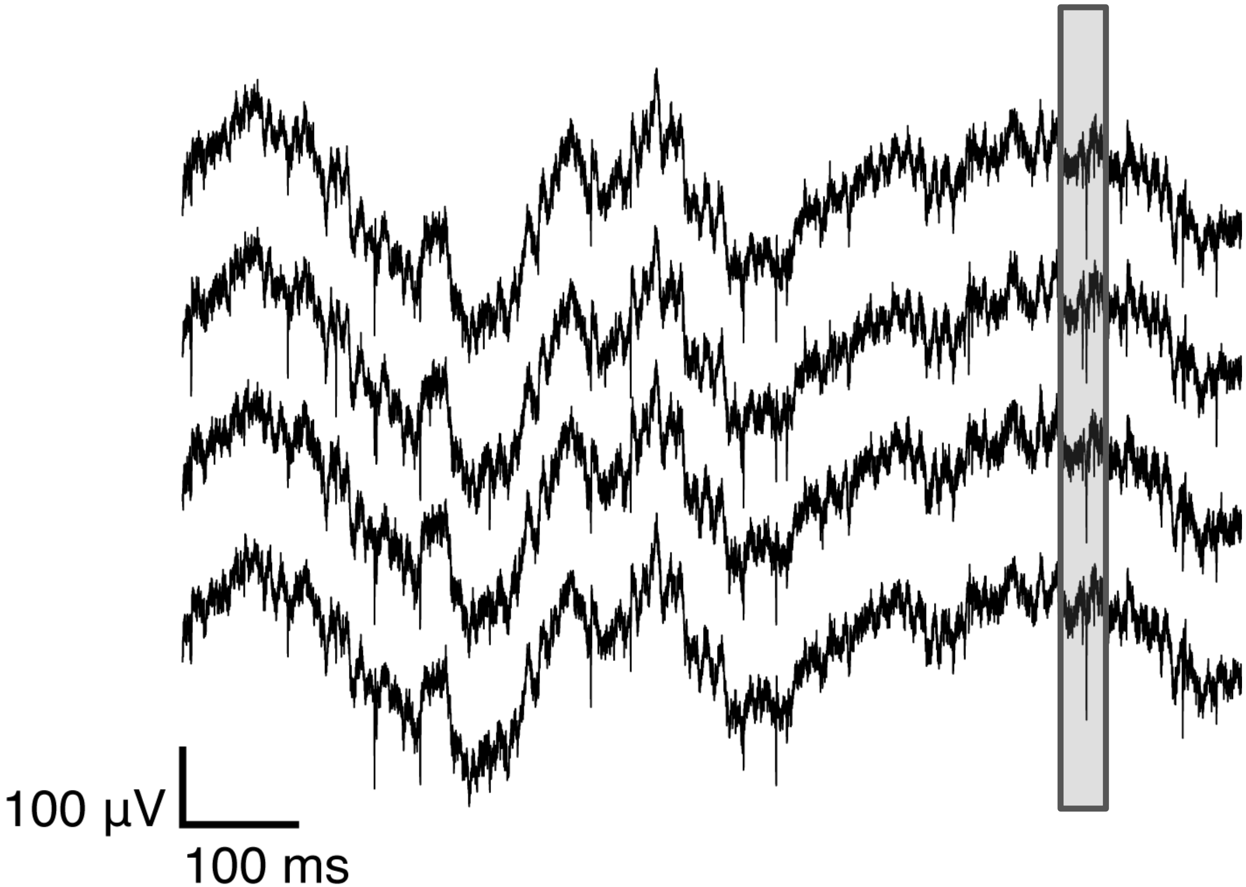


SPIKE DETECTION

TASK 1

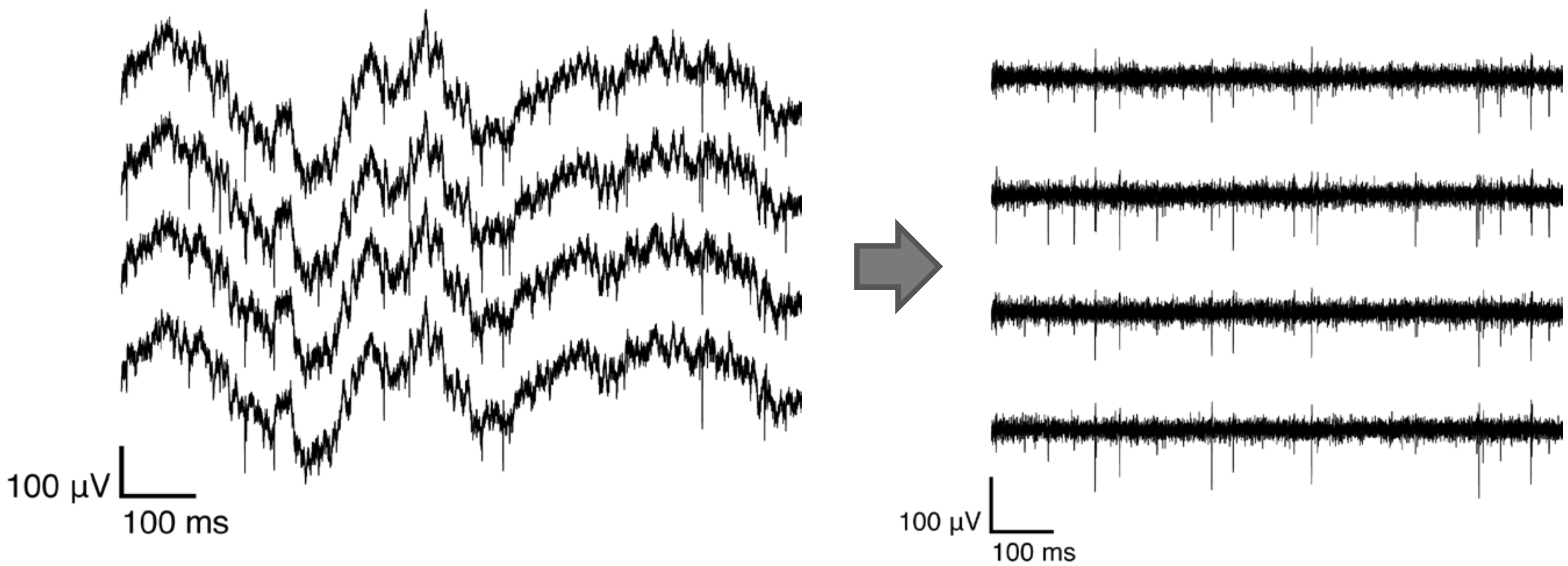
RAW SIGNAL





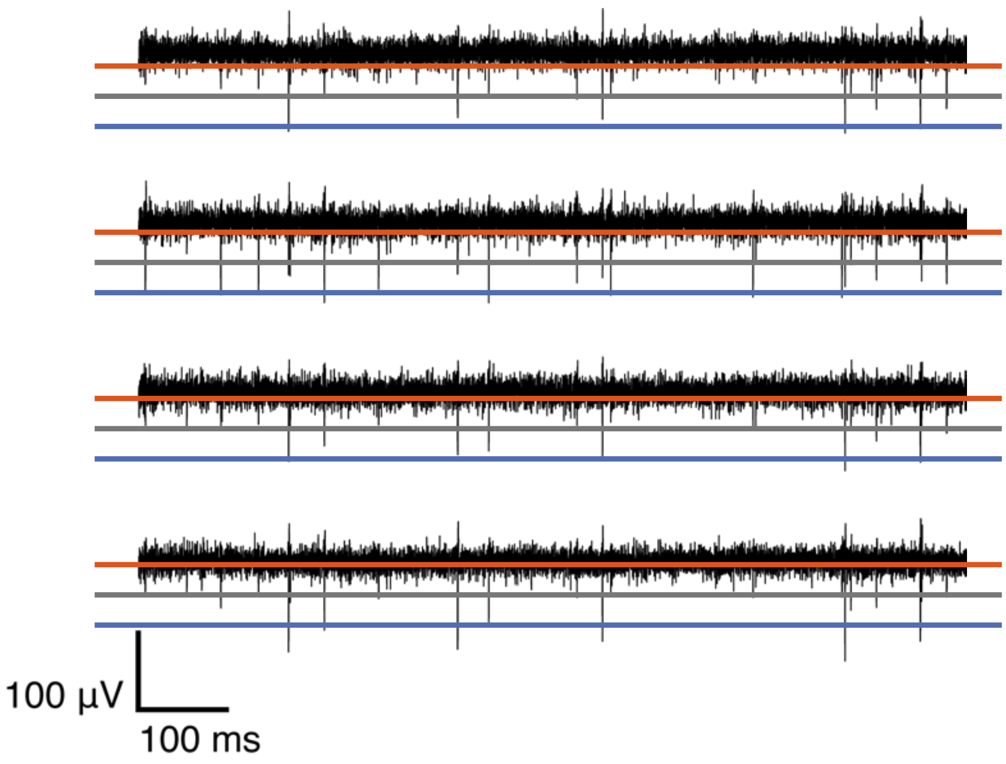
Tetrodes: 4 channels in parallel!

REMOVE LFP AND HIGH FREQUENCY NOISE




```
y = filterSignal(x, Fs);
```

SPIKE DETECTION

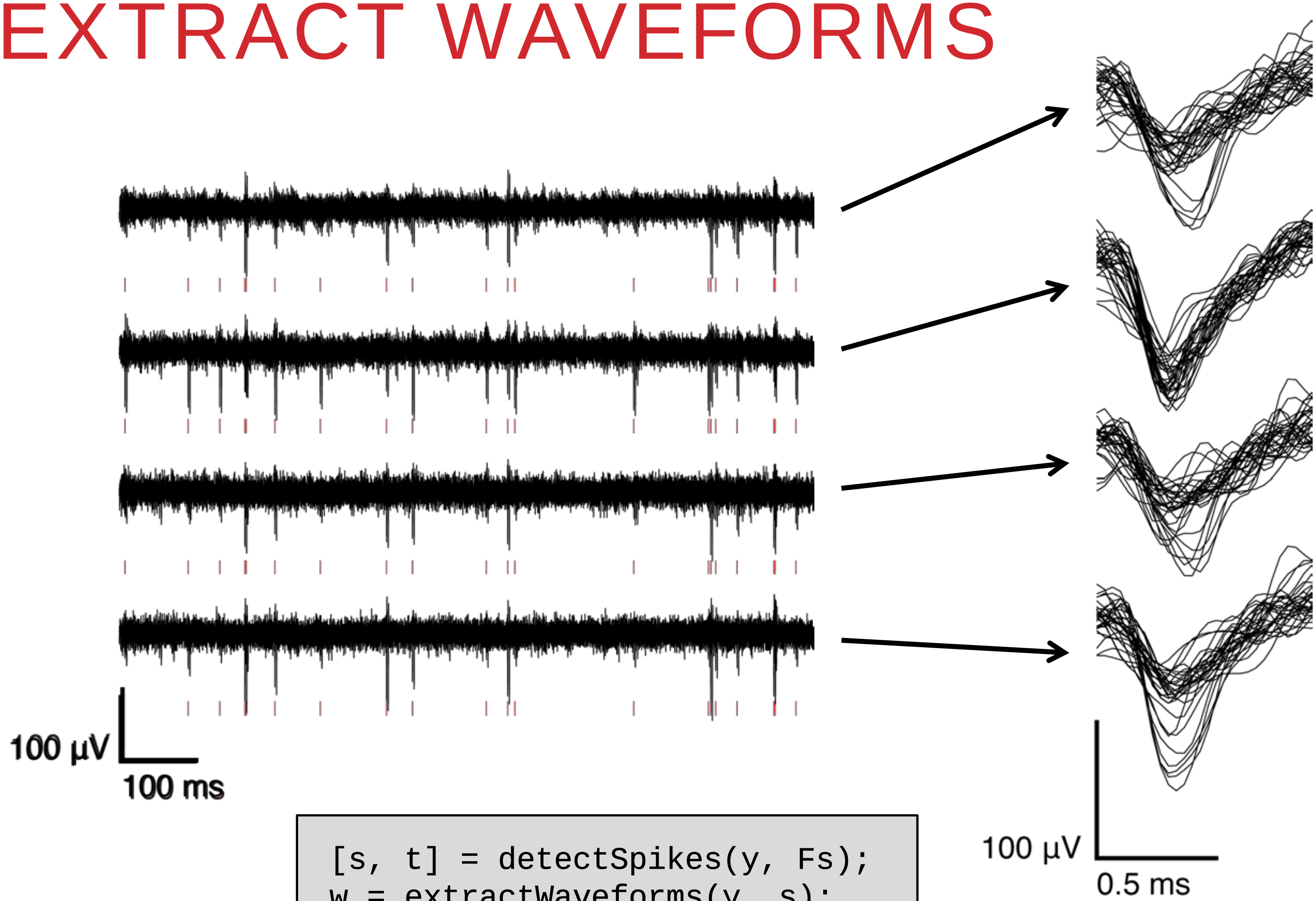


- **Threshold at $N\sigma$**
- **Low threshold**
- **High threshold**
- **Robust estimation of σ:**

$$\hat{\sigma} = median\left(\frac{|x|}{0.6745}\right)$$

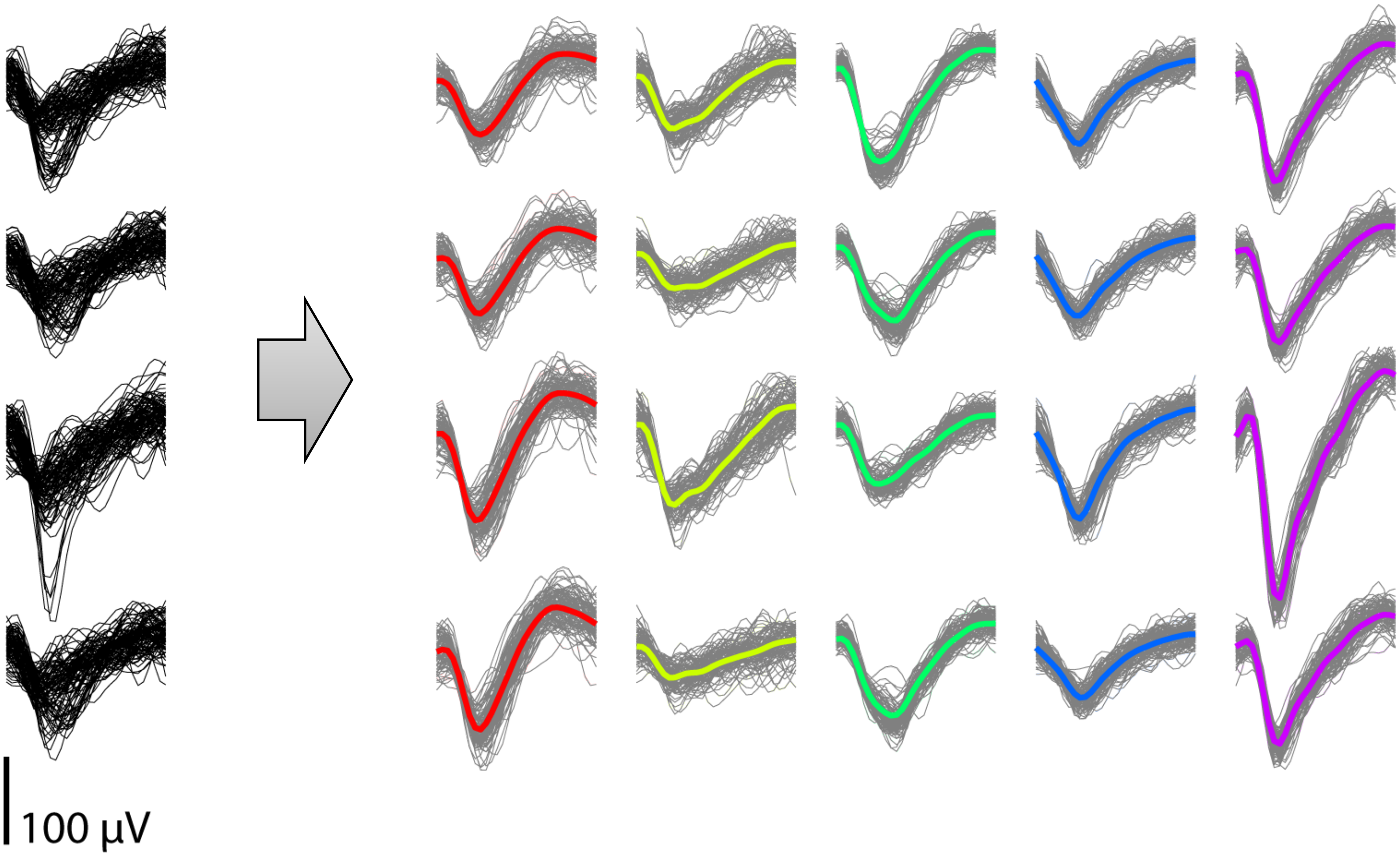
Quian-Quiroga 2004

DETECT SPIKES AND EXTRACT WAVEFORMS




```
[s, t] = detectSpikes(y, Fs);
w = extractWaveforms(y, s);
```

GOAL (EVENTUALLY): SPIKE SORTING



FEATURE EXTRACTION

TASK 1

PROBLEM: CURSE OF DIMENSIONALITY

4 channels × 32 samples = 128 dimensions

Problems:

- Computationally intractable
- Overfitting/regularization

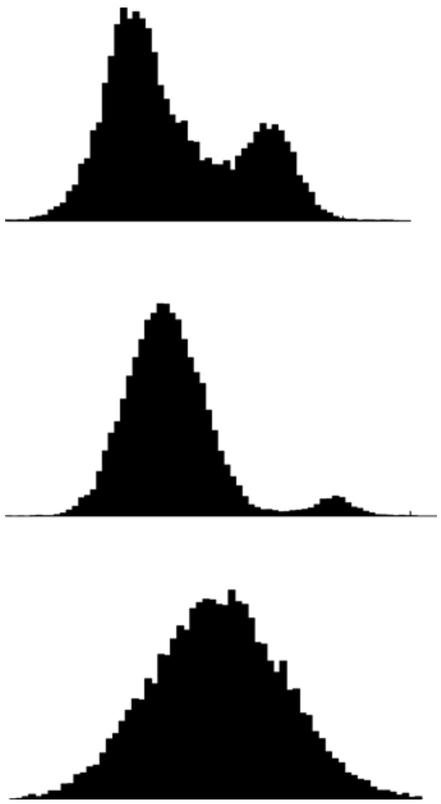
Solution: dimensionality reduction

WHICH FEATURES TO CHOOSE?

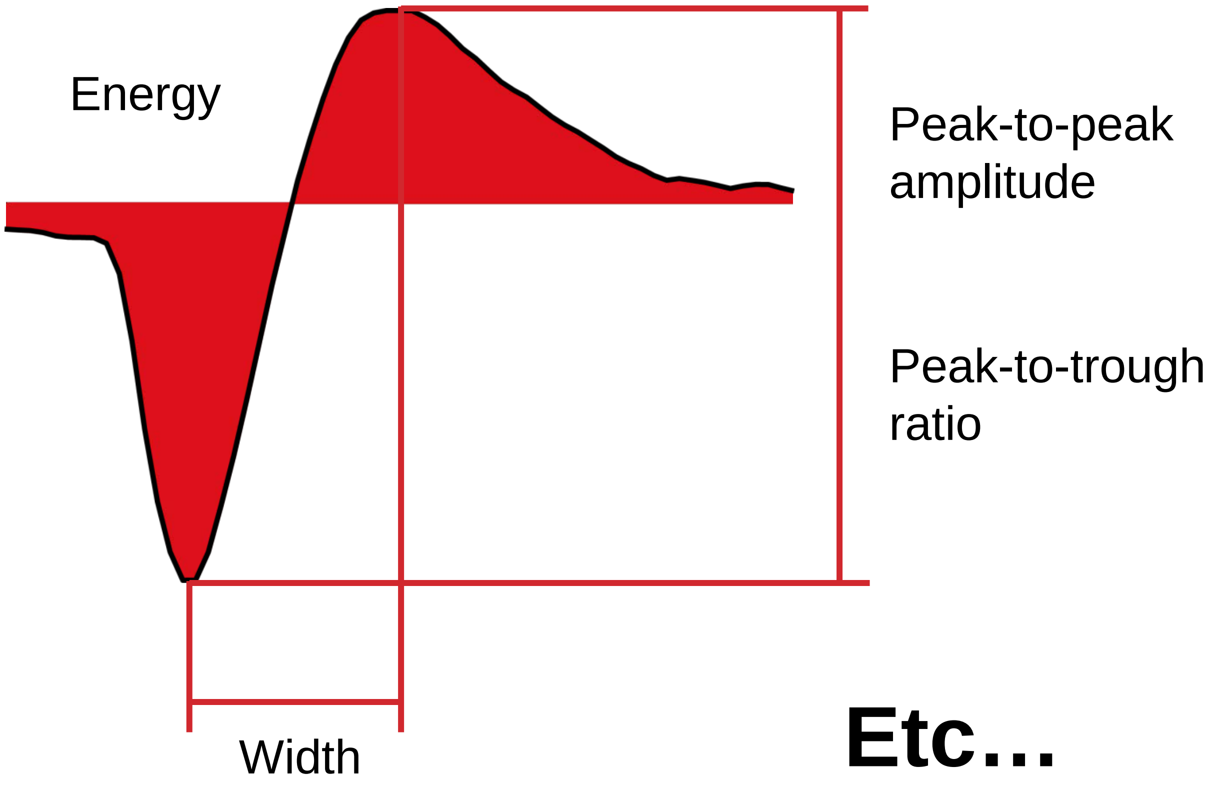
Goal: small set of features that discriminate between different neurons

Robust against noise?

What is good may depend on the spike sorting algorithm (shape of distribution)



“DESCRIPTIVE FEATURES”



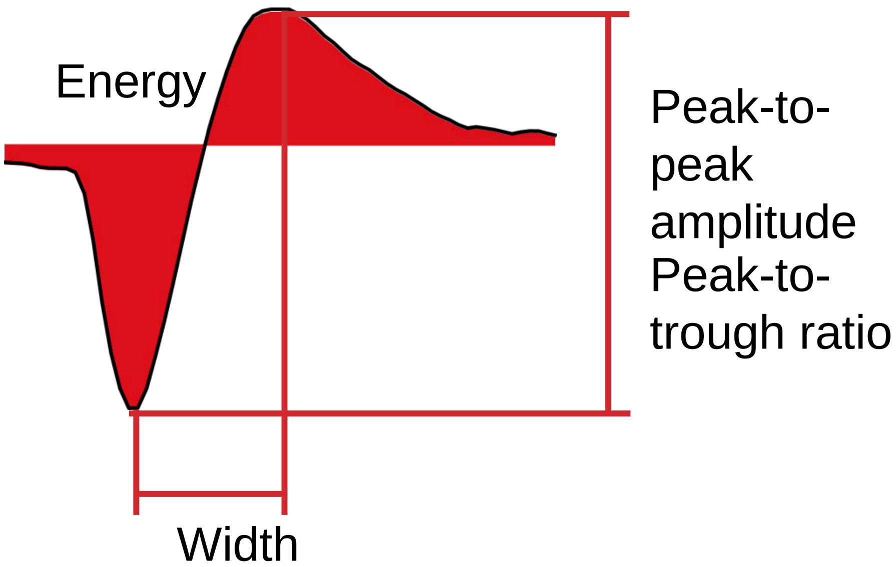
Etc…

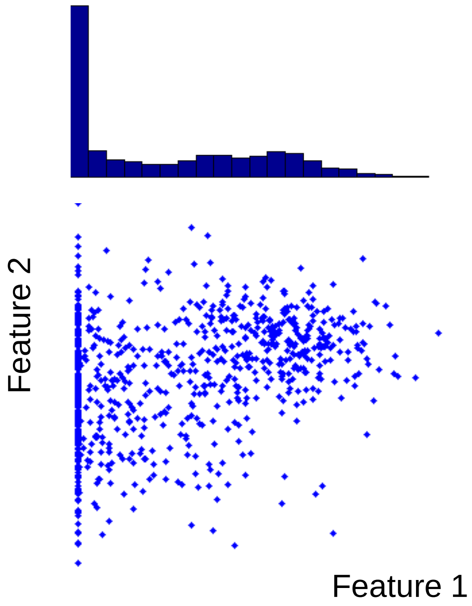
PROBLEMS

Some times undefined (no trough)

Non gaussian distributions

Correlated



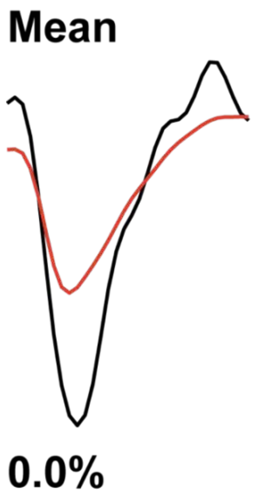


See also
Quian-Quiroga, 2004

PRINCIPAL COMPONENT ANALYSIS (PCA)

Finds an orthogonal basis for the data

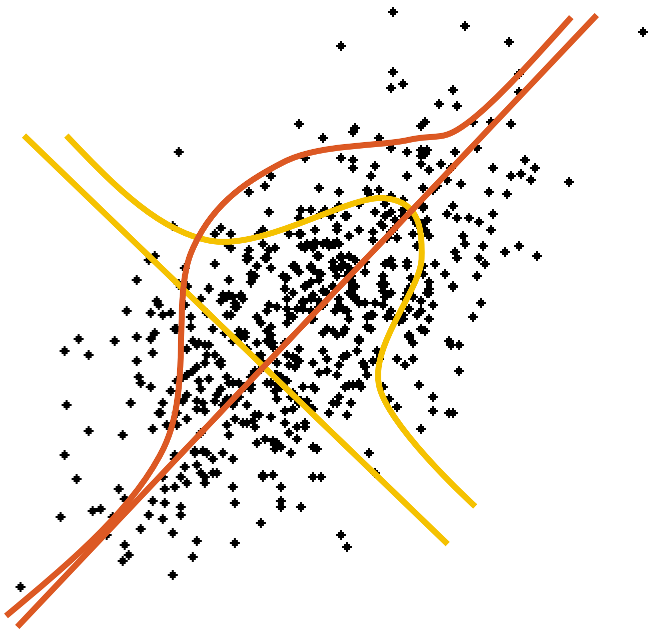
First PC is the direction of largest variance



→ A few components capture most of the variance

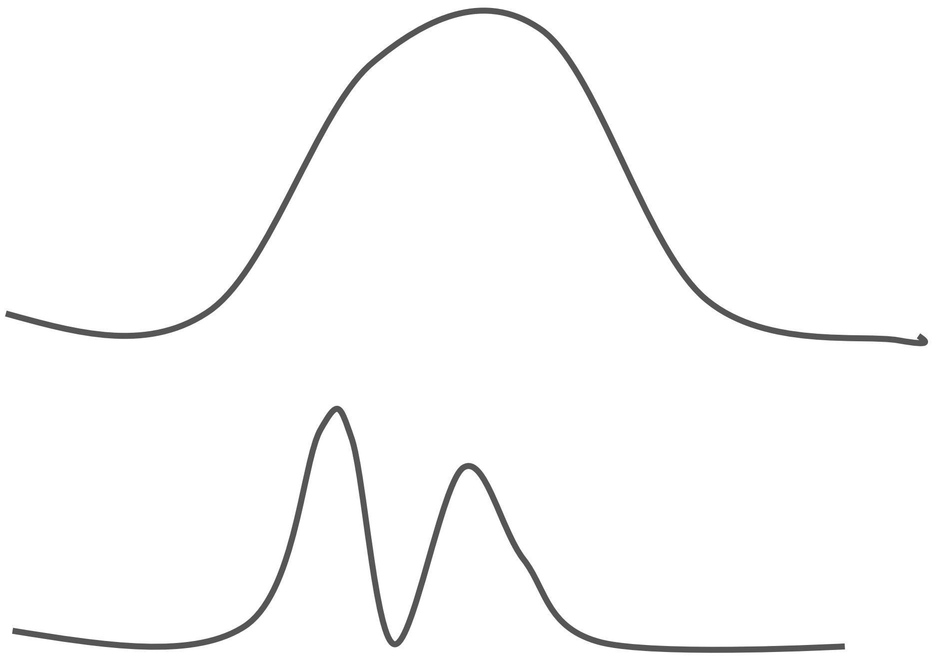
PCA

Goal: subspace capturing maximum variance



PROBLEM WITH PCA

Largest variance – multiple peaks



Larger variance

Better for clustering

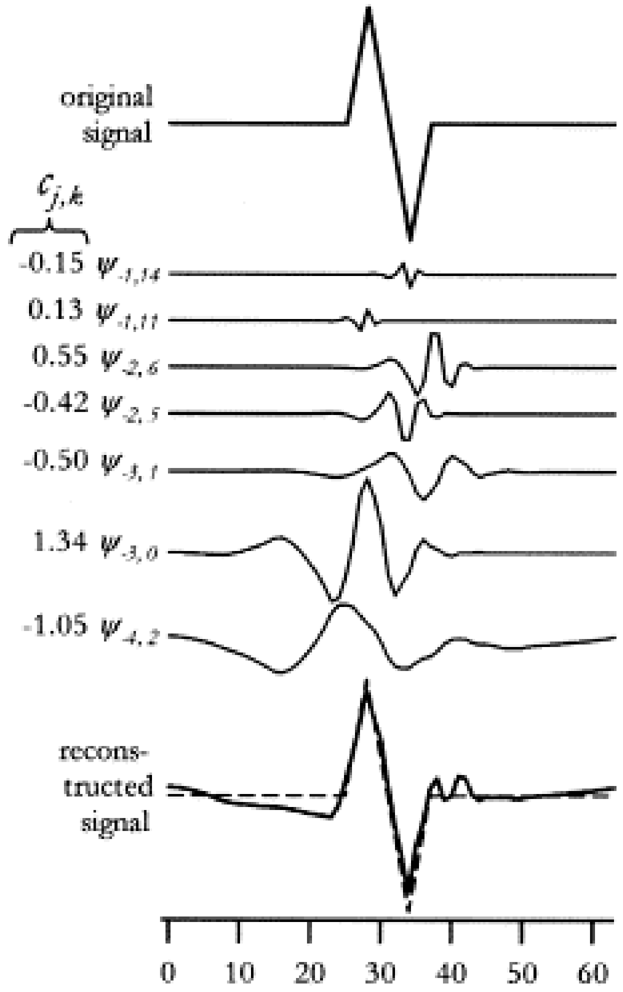
WAVELET BASIS

Quian Quiroga et al. (2004): Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering. *Neural Computation*, 16.

Letelier & Weber (2000): Spike sorting based on discrete wavelet transform coefficients. *Journal of Neuroscience Methods*, 101(2).

And many more.

Your ideas?



ALMOST DONE...

FORM GROUPS

1^{ST} REPORT

(PAPERS / REPORTS NEXT WEEK)