Neuronal Spike Sorting for Multi-Day Multielectrode Array Recordings

John H. Abel

Department of Systems Biology, Harvard University

January 2017

Outline

Unique challenges of circadian spike sorting

○ Neurons appear and disappear

Unique challenges of circadian spike sorting

- Neurons appear and disappear
- Spike shape changes day-to-day

Unique challenges of circadian spike sorting

- Neurons appear and disappear
- Spike shape changes day-to-day
- Massive amounts of data

Addressing these issues

1 Binning into days for sorting / decision tree sorting

Addressing these issues

1 Binning into days for sorting / decision tree sorting
2 Correlating spike shape between days

Addressing these issues

1. Binning into days for sorting / decision tree sorting
2. Correlating spike shape between days
3. File formatting, creating a sorting tree based on a subsample of data (as little as 5%)

File setup

Manual

Reasoning
You need to put your files somewhere. Put them in a folder called `mcd`.
Within MCD, separate these files by days (e.g. 0day, 1day ...)
IMPORTANT: mcd files must be numbered sequentially (start with *0000.mcd).

Manual

Reasoning
You need to put your files somewhere. Put them in a folder called `mcd`.
Within MCD, separate these files by days (e.g. 0day, 1day ...)
IMPORTANT: mcd files must be numbered sequentially (start with *0000.mcd).

Requires: `experiment/`
Estimated time: <1 minute
Resulting directory: `experiment/mcd`

File format conversion

```
create_numpy_database.py
```

Reasoning

Loading .mcd files takes a long time.

Loading numpy files takes almost no time.

To take data from one electrode, you must load many .mcd files.

```
create_numpy_database.py
```

Reasoning
Loading .mcd files takes a long time.
Loading numpy files takes almost no time.
To take data from one electrode, you must load many .mcd files.

Requires: `experiment/mcd`
Estimated time: 1-2 hours total (less if parallel)
Resulting directory: `experiment/numpy_database`

Subsampling

```
subsample_mcd_array.py
```

Reasoning
Clustering can take a long time if trying to cluster everything at once.
After hundreds of thousands of spikes, everything begins to look the same.
Instead, cluster using a limited number (about 1 mcd file, tuneable) of spikes.

```
subsample_mcd_array.py
```

Reasoning
Clustering can take a long time if trying to cluster everything at once.
After hundreds of thousands of spikes, everything begins to look the same.
Instead, cluster using a limited number (about 1 mcd file, tuneable) of spikes.

Requires: `experiment/numpy_database`
Estimated time: <1 hour
Resulting directory: `experiment/subsampled_test_sets`

```
sort_electrodes.py
```

Reasoning
Using the subsampled data for an electrode, first identify the noise and remove it. Then sort resulting spikes to build a decision tree. Classify all spikes for the electrode based on this tree.

`sort_electrodes.py`

Reasoning
Using the subsampled data for an electrode, first identify the noise and remove it. Then sort resulting spikes to build a decision tree. Classify all spikes for the electrode based on this tree.

Requires: `experiment/subsampled_test_sets`
Estimated time: 1 hour per day
Resulting directory: `experiment/numpy_neurons`

Clustering: removing noise

Take subsamples, cluster using Gaussian Mixture Model (GMM), sort by expectation-maximization, delete the noise cluster.
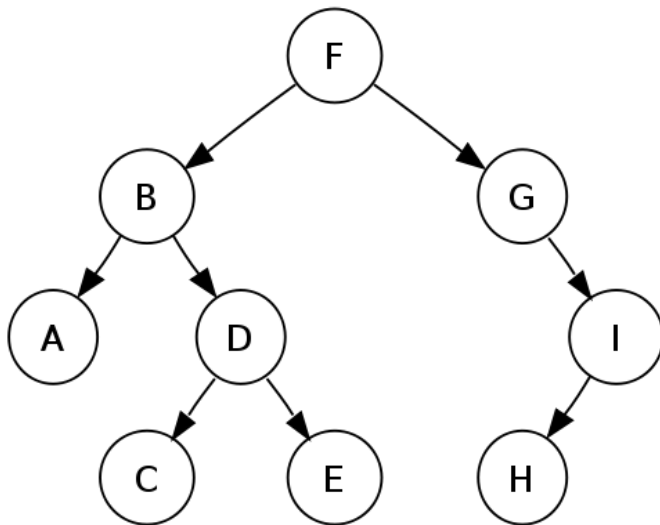
Clustering: building a classifier
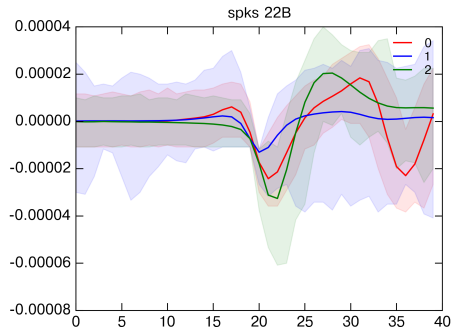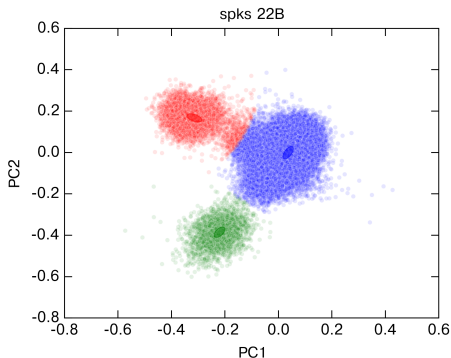
Decision tree clustering:

1 PCA, GMM cluster
2 EM sort
3 Take each cluster, PCA, GMM cluster
4 Take each cluster, EM sort
5 Continue until each dataset is a single cluster.

Record all PCA data and save it to construct a tree.
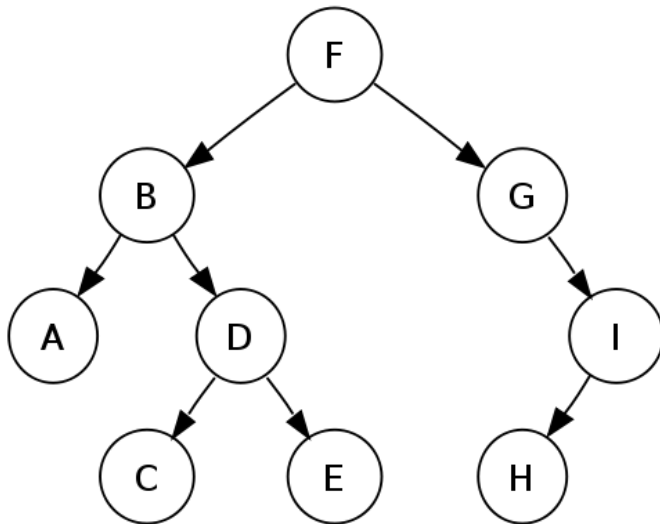
Clustering: building a classifier
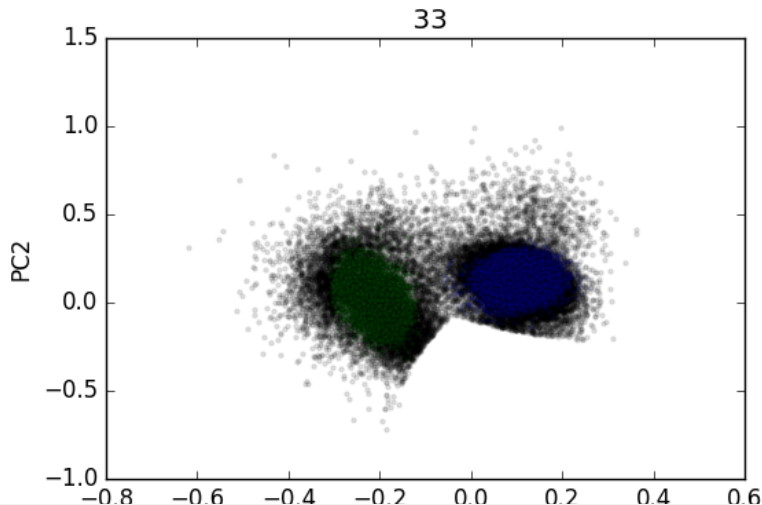
Decision tree clustering:

Take all data and sort based on the tree.

Clustering: Classification

Use mahalanobis distance to classify spikes in terminal clusters.
"False negatives are better than false positives."

33

Piecing it back together

```
combine_neurons.py
```

Reasoning
Correlate the spike shapes between different days across the experiment.
Some neurons come and go. We should respect that by not over-sorting.

Piecing it back together

`combine_neurons.py`

Reasoning
Correlate the spike shapes between different days across the experiment.
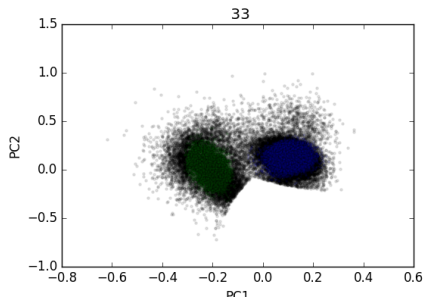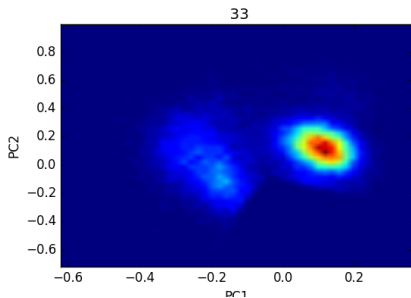Some neurons come and go. We should respect that by not over-sorting.

Requires: `experiment/numpy_neurons`
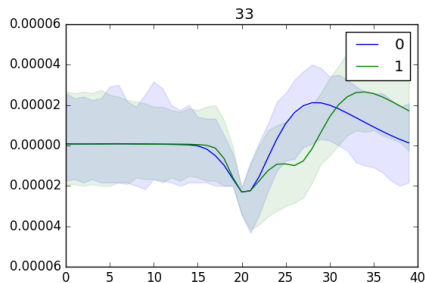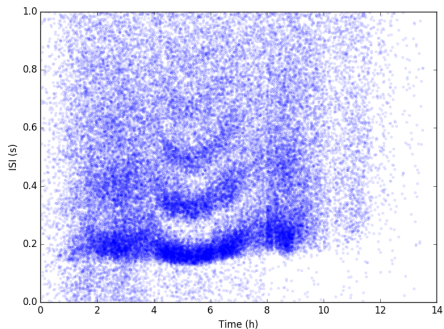Estimated time: 1-2 minutes
Resulting directory:
`experiment/numpy_neurons/combined_neurons`

Piecing it back together

github.com/JohnAbel/spikesort

Tips and tricks

- The information you need to set up will be at the top of each file
- Most errors are typos

Tips and tricks

- The information you need to set up will be at the top of each file
- Most errors are typos
- Folders that exist will refuse to be overwritten

Tips and tricks

- The information you need to set up will be at the top of each file
- Most errors are typos
- Folders that exist will refuse to be overwritten
- Don't upload your data to the server. Just bring a hard drive

Tips and tricks

- The information you need to set up will be at the top of each file
- Most errors are typos
- Folders that exist will refuse to be overwritten
- Don't upload your data to the server. Just bring a hard drive
- Don't fly blind! Check the sorting once it's finished and email me for help if you need. A variety of plots are generated.

Tips and tricks

- The information you need to set up will be at the top of each file
- Most errors are typos
- Folders that exist will refuse to be overwritten
- Don't upload your data to the server. Just bring a hard drive
- Don't fly blind! Check the sorting once it's finished and email me for help if you need. A variety of plots are generated.
- This isn't magic, and even though it works more often than not, it sometimes sorts badly.