

## Task 3: Identification of single units

**Due date: Monday, May 4, 11.59am**

### Tasks

Not every cluster in the fitted mixture model corresponds to a single neuron's spikes (i.e. is a 'single unit'). We will explore different ways of identifying single units and telling them apart from multi unit activity. In all your plots, color-code the clusters consistently.

1. **Visual inspection of clusters.** The most obvious candidates for single units are clusters that you can identify visually as being well separated from the rest. You can use the plotting function you implemented last week as a first tool to identify putative single units. Of course there may be other less obvious cases in the data.
2. **Visual inspection of waveforms.**  
*Figure 1: Plot the average waveforms and 100 examples from each cluster. This is a good sanity check and helps to identify potential artifacts such as electrical noise or clusters representing overlapping spikes (they often have large variance and few spikes).*
3. **Auto/cross-correlation.** Implement a function to calculate the auto/cross-correlograms of all clusters/pairs of clusters called `correlogram()`. Correlograms are useful mostly for two reasons:
  - a. To identify clusters that represent multi unit activity. Neurons have a refractory period: after firing a spike they cannot fire another spike within a period of time (typically at least 2–5 ms, depending on the cell type). Thus, if multiple cells contribute to one cluster, it won't have a refractory period.
  - b. To identify two (or more) clusters that represent the same single unit: if this is the case the cross-correlogram of the two clusters should show the refractory period, since it consists of spikes from only one cell, which cannot occur too close to each other.

*Figure 2: Plot a matrix with cross- and auto-correlograms like shown in the lecture. Use a bin size of 0.5 ms and a range of  $\pm 20$  ms. Which auto-correlograms show a refractory period? Which cross-correlograms do?*

4. **Cluster Separation.** Implement linear discriminant analysis to visualize how well each cluster is separated from its neighbors in the high-dimensional space in the function `separation()`. Project the spikes of each pair of clusters onto the axis that optimally separates those two clusters and plot the histograms.

*Figure 3: Plot a matrix with pairwise separation plots, showing the histogram of the points in both clusters projected on the axis best separating the clusters (as shown in the lecture).*

5. **Identify putative single units.** Use all of the above tools to identify all putative single units in the dataset. Which ones are these and why?

### **Tips**

- To facilitate comparison of different clusters, use the same y axis range when plotting (where appropriate; for instance when plotting the waveforms).
- An auto-correlogram is simply the cross-correlogram of a neuron with itself, except that for plotting you may want to remove the zero-lag bin.
- Since we're interested only in the shape of the correlograms you don't need to normalize them.
- The calculation of cross-correlograms is usually fastest if you implement it in a single loop over all spike times and compute the cross-correlogram simultaneously for all pairs of clusters.
- It can be useful to implement the plotting functions in a way that you can (optionally) plot only a selected subset of the clusters.