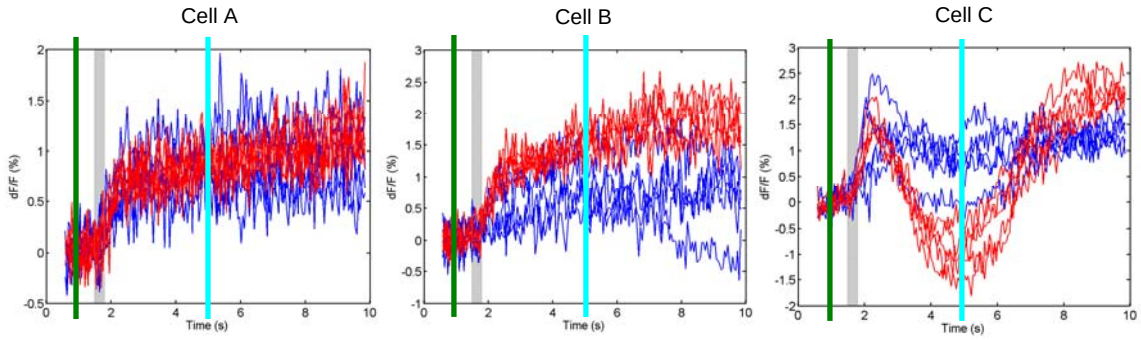
Supporting Online Material – LDA & PCA

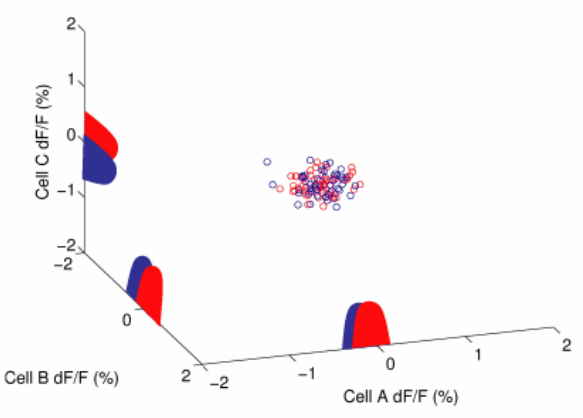
For readers unfamiliar with Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA), we wish to demonstrate a hopefully intuitive example. As described in the paper, our datasets are N-dimensional, where N is the number of neurons from which we record. First, we will show an example in which N = 3 to demonstrate LDA. Then we will show a 2D example of PCA. The goal is to develop an intuitive feel for what LDA and PCA do in a low dimensional example. We hope the reader can then imagine how the technique can be extended to higher dimensional datasets that are more difficult to visualize.

LDA

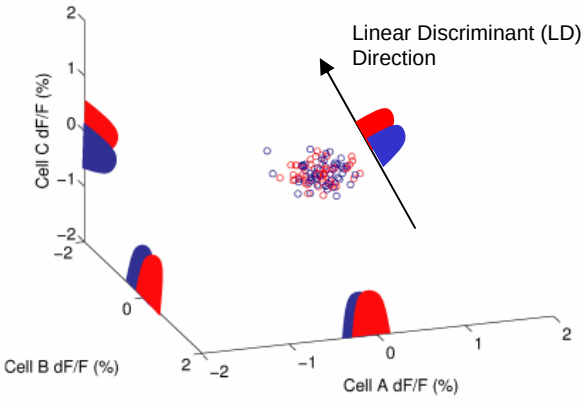
We begin by examining the raw signals from three neurons measured across 10 trials; 5 swimming trials (blue traces) and 5 crawling trials (red traces).



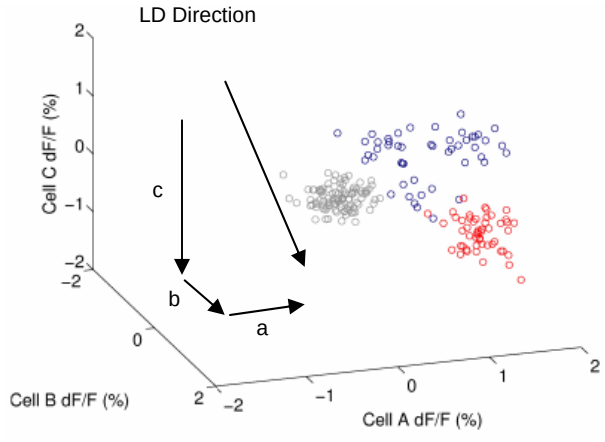
We are interested in when the trajectories for swimming and crawling trials have diverged in the 3D joint space of these three neurons. The 3D space is just a scatter plot of the optical data points from the neurons. First, we look at the data in a 500 ms window prior to the stimulus (highlighted by the green bars in the 3 plots above).



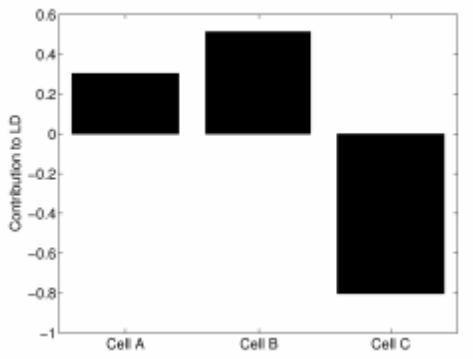
The distributions of swimming versus crawling points are shown along each axis. Since the window we chose is prior to the stimulus, none of these distributions are significantly different. The idea of the LDA is to find a line such that, when the data are projected onto it, the distributions are maximally separated. For example:



Again, the distributions along this line are not significantly different at this point in time. However, if we pick a later time window (the cyan colored bars in the 3 raw data plots), the behaviors are clearly distinguishable:



The linear discriminant (LD) direction tells us the optimal linear combination of the neurons to achieve maximal discrimination. This linear combination is defined by the slope (vectors a, b, and c above) of the LD:

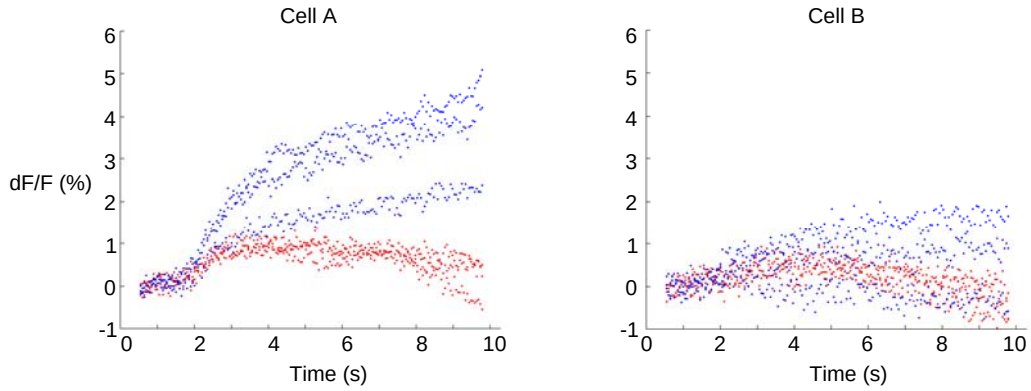


Since cell C does a good job of discriminating the two behaviors, it has a relatively large magnitude contribution to the slope of the LD. In contrast, cell A does not discriminate well, so it has a smaller contribution. Our basic technique then is to slide a window in time and estimate a LD at each time. We then perform an ANOVA on the data projected onto each line and ask when the distributions become significantly different. One of the main findings from the paper is that linear combinations of neurons can separate the swimming and crawling distributions earlier than single neurons.

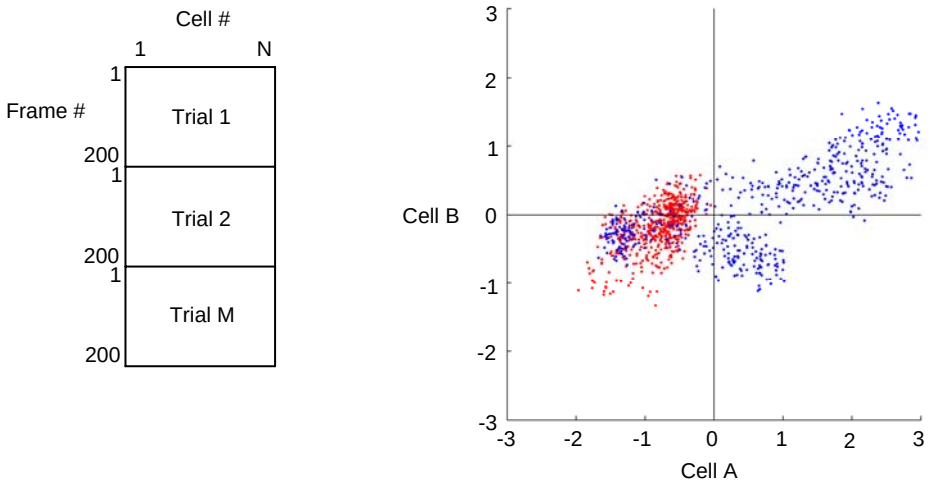
PCA

The LDA described above works well in low-dimensional spaces, but is susceptible to overfitting. The problem is that as more dimensions (neurons) are taken into consideration, more data points are needed to adequately sample the higher dimensional space. Since we were limited in the amount of data we could collect, we used PCA to reduce the dimensionality of our datasets prior to the LDA.

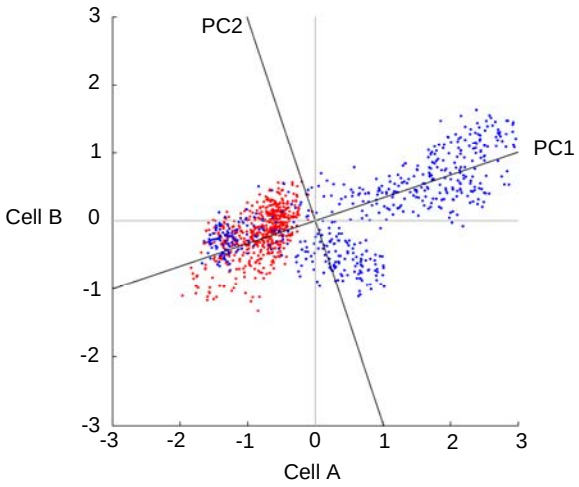
For this example, we will consider two neurons measured across 6 trials; 3 swimming trials (blue traces) and 3 crawling trials (red traces):



We organize the data into a matrix in which each column represents a neuron and the rows contain the time points for each trial. As mentioned above, these 2 neurons are considered to be dimensions (or axes) in a 2-D space. A scatter plot of the columns of the data matrix demonstrates this.

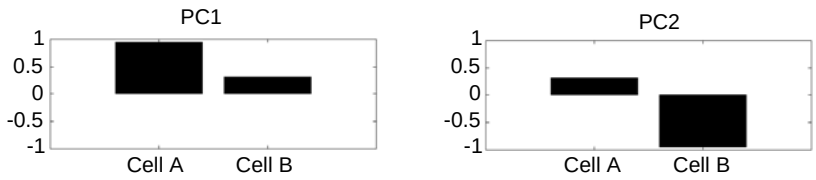


Notice that the activity of the 2 neurons is correlated. If the 2 neurons were uncorrelated, one would expect a circular cloud of data. PCA rotates the original axes to point in new directions. It chooses the new directions to lie along the directions of maximal covariance. These new directions are called the principal components (PCs). By rotating the original axes, it is hoped that the new PCs will provide a more parsimonious description of the data (i.e., we will ignore directions that do not contribute much to explaining the variance). The rotation that PCA found is shown by the two directions PC1 and PC2.

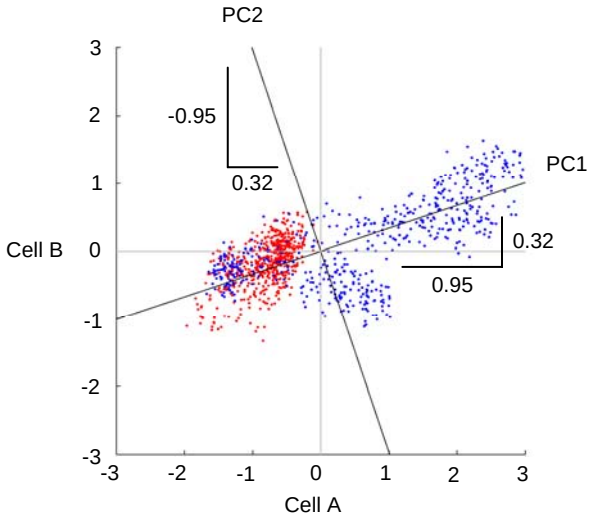


PC1 is always aligned along the direction of maximal covariance. PC2 is aligned along the direction of next maximal covariance, with the important constraint that it remain orthogonal to PC1. (The scatter plot was centered about the origin before the rotation).

What we call the PCs are actually the slopes of the new directions relative to the original axes. We choose to display them as bar graphs.



Each value in the bar graph is the contribution of each of the original axes to the new directions. When these values are combined, they are exactly the slopes of the new directions.

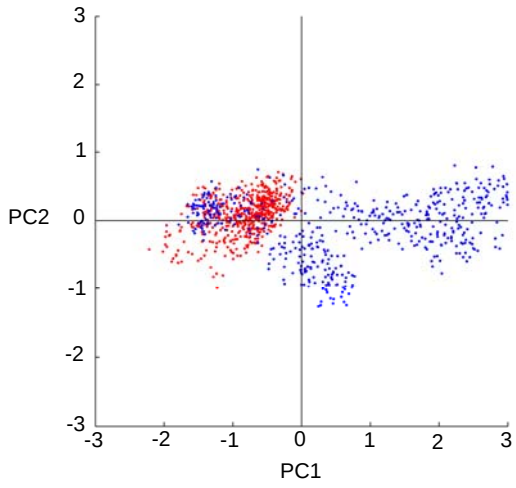


Note how PC1 has a larger value for cell A than cell B. This is because the direction of PC1 is oriented closer to the direction of cell A than cell B. This is what we mean when we refer to neurons with large magnitude contributions to a PC. Another way to express

this is that the PCs are linear combinations of the original directions. A neuron that does not influence the slope would then have zero contribution to the PC.

We hope that the reader can imagine how this generalizes to N dimensions. Each PC from an N-dimensional dataset contains the contributions of each of the N neurons. Also, by definition, PCA returns N PCs (N = 2 in the case above). When the datasets are of high dimensionality, the first few PCs can capture most of the overall variance. Suppose we record from 100 neurons (N = 100). PCA returns 100 PCs. The first 3 PCs (PC1, PC2, and PC3) might account for 90% of the variance in the 100-dimensional scatter plot. The remaining 10% of the variance is spread out over the remaining 97 PCs. Since each of these PCs explain little variance, they are discarded. Going back to our 2-dimensional example, PC1 explained 90% of the overall variance and PC2 explained 10%.

One important result from our analysis is not only that the first several PCs explain most of the variance, but that the new directions help separate the swimming from the crawling trials. We can see this by looking at the data in the rotated coordinate system.



Notice how PC1 separates the swimming and crawling data points. PC2, on the other hand, does not separate the behaviors well. By discarding PC2, we can reduce the dimensionality of the space, but still retain most of the interesting dynamics along PC1.

We now return to the idea of the LDA. Instead of performing the LDA on the raw data, we estimate LDs in a reduced PC space (to avoid overfitting). The slopes of these lines describe the relative contribution of each of the PCs to the discrimination. So the LD is a linear combination of PCs, each of which is a linear combination of neurons. Therefore, neurons contributing strongly to PCs that help separate swimming trials from crawling trials are candidate decision-making neurons. The issue of how many PCs and the bin width we use to estimate the LDs is addressed in Fig. S1.

We have attempted to graphically demonstrate the mechanics of LDA and PCA. There are many good texts that cover the mathematical background (*S1, S2*). Also, an excellent tutorial on PCA can be found at (*S3*).

Supplemental References

S1. Mardia K, Kent J, Bibby J, Multivariate Analysis (Academic Press, San Diego, CA, 1980).
S2. Duda RO, Hart PE, Stork DG, Pattern Classification (Wiley-Interscience, New York, ed. 2, 2000).
S3. Shlens J (2001) A Tutorial on Principal Component Analysis: Derivation, Discussion and Singular Value Decomposition, (http://www.snl.salk.edu/~shlens/pub/notes/pca.pdf)