Computational Neuroscience - Project 6

Principal Component Analysis

Kadir Berat YILDIRIM

Dec 2022

Introduction

In this project, our aim is to get familiar with principal component analysis (pca) which is used for dimensionality reduction while also exploring an application of this technique for spike-sorting neuronal waveforms. We build our own primitive spike sorter using pca to analyze extracellular data from recordings in the primary motor cortex of a macaque monkey.

Data is from the primary motor cortex of a macaque monkey, which contains waveforms and time stamps - waveforms in millivolts and timestamps in milliseconds.

Principal component analysis is a useful technique when dealing with large datasets. It basically is a projection of a higher-dimensional space into a lower dimensional space while preserving as much information as possible. In some fields, collected data can have dimensions on the thousands scale and manipulating the data in this way is not desirable for multitude of reasons - such as large computing time or storage problems. However, we can't just arbitrarily ignore dimensions either as we might lose some of the information we are trying to capture. Principal component analysis is a common method used to manage this trade-off. The idea is that we can select the 'most important' dimensions, and keep those, while throwing away the ones that contribute mostly noise.

Reducing the dimension does not mean that we throw away one of the correlated dimensions and keep the other. It is important to note that the dimension chosen was not one of the original two: in general, it won't be, because that would mean our variables were uncorrelated to begin with. We can also see that the direction of the principal component is the one that maximizes the variance of the projected data. This means that we are 'keeping as much information as possible'. So in that sense, principal components analysis is just a coordinate transformation.

Principal component is the dimension that maximizes the variance of the projected data.

The data we have contains waveforms from 2 different days. We first apply PCA to the first day's waveform and see how many principal components explain how much of the variance. Then choose the first 2 components and plot the waveforms in the PCA space, that is the transformed data. Then form a template waveform by selecting a circular region inside 2 clusters and compare the second day's waveform with this template and its own template.

Analysis

First Day's Waveform

In matlab, I start by loading the waveform data, then using *pca* command on the first day's waveforms. This command also creates the covariance matrix from the raw data so we can directly give it the waveform data. A variable I have called *latent* - that is returned from the *pca* command - contains variances, which can also be described as eigenvalues of the covariance matrix. In figure 1, we see percentage of the variance explained by taking more and more principal components. Note that *pca* command also sorts the eigenvectors and values in a descending order so for example first principal axis explains the most variance, while second one explains the second most variance in the data.

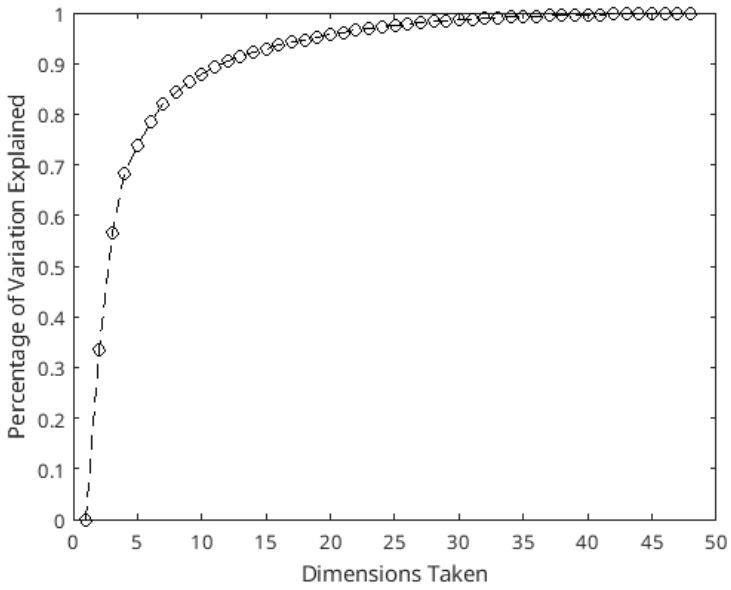

Figure 1: Visualization of how much variability of the data is represented as we take more and more principal components. Small number of dimensions can represent a huge number of variability in our data - i.e. taking 2 principal components cover 56 percent of the total variability.

Taking first 2 principal components would together explain 56.71 percent of the total variance, and taking the first 3 would explain 68.42 percent variance. As we take more and more components in the PC space, we explain more and more variability in the data. But we also increase the dimensions that we work with. Because of that, we may want to limit the number of axes we take from PC space. I have worked with first 2 principal components of the PC space in this project.

Numerically, first principal component explains 0.3354 percent variance while second one contains 0.2317 percent of the total variance.

I first plot the data in the PC space as a 3D histogram to decide on the clusters, as can be seen in figure 2.

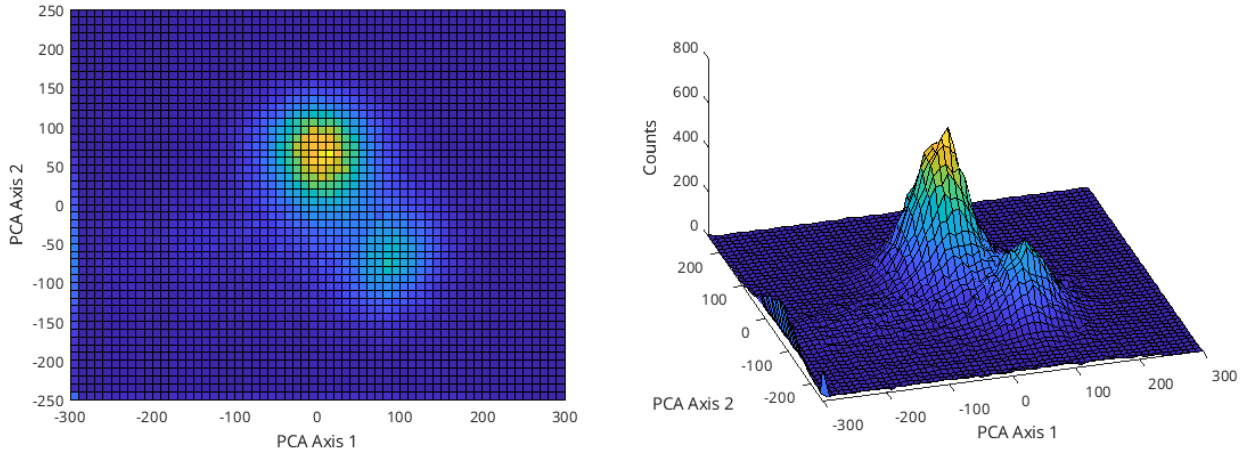

Figure 2: First day's waveform data plotted in the PC space, taking the first 2 principal components only.

On figure 2, on the left we see 'bird-eye view' of the PC space and on the right is 1 angle of the 3D view of the same space. There seems to be 2 clusters that can be separated from each other manually; one with a center of roughly (12.5, 75) and a radius of 65, and the other with a center of (87.5, -67.5) and a radius of 37.5. Figure 3 shows these clusters marked with a circle with these values.

Even though I have manually decided on the clusters, this could be done programmatically.

After the clusters are marked, in order to form a template waveform, I find the points - separately - that are inside both of those circles. Then, for all points that are in those circles, by multiplying with the transpose of the rotation matrix, we go back to the initial space/coordinates and average out across all channels. This way, we create an average template to distinguish the first neuron's firings. Figure 4 shows the template for both of the clusters.

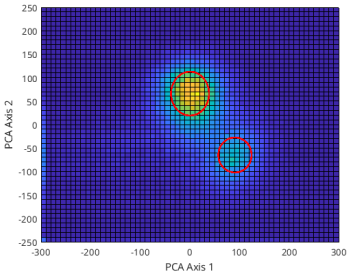

Figure 3: First 2 PC space clusters marked with red circles.

What can be understood from figure 4 at first glance is that these 2 neurons are measured in different amplitudes in different channels. Then the templates are a way to decide which of them has fired. One way to achieve that information is to use RMSE (root mean squared error), where for a given waveform (or measurement), we can calculate the RMSE for all the points of those clusters and see how they vary. Then, a we could take their *mean* + 2 *std* for example, to decide whether a given waveform is from that cluster or not (of course this would mean that the neuron related to that cluster has fired).

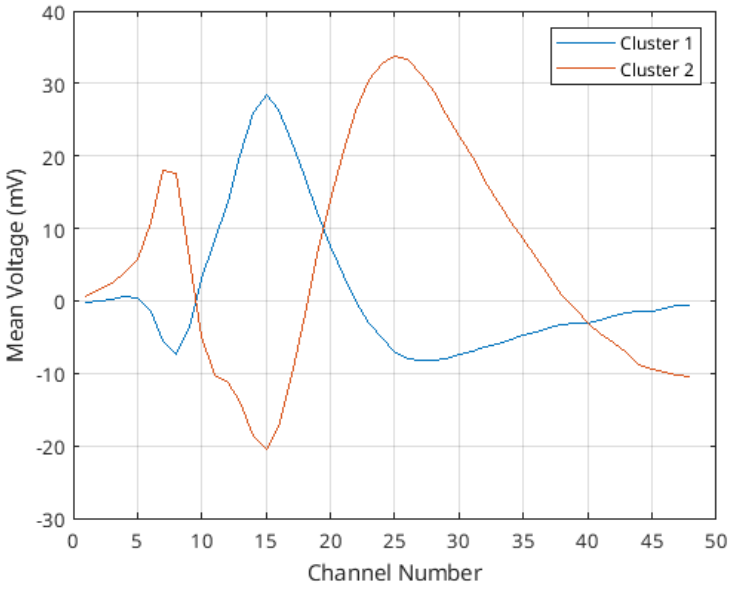

Figure 4: Template waveforms for both of the clusters found in the PC space.

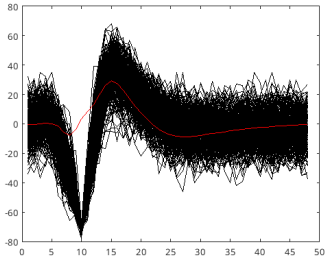

Figure 5: All waveforms plotted black and the template plotted red for the first cluster.

For cluster 1, mean RMSE is 14.21 with a standard deviation of 3.27. For cluster 2, mean RMSE is 13.70 with a standard deviation of 4.32.

If we were to take $Mean + 2\sigma = 20.75$ as the maximum RMSE value for the first cluster's template (that is, if a waveform's RMSE value compared to this template exceeds this value, we say its not from this cluster, or its not neuron 1), we can cover around 95.4% of the true firings of neuron 1 (assuming a normal distribution). For the second cluster, same value is calculated as 22.34. These values are calculated by taking RMSE for every waveform inside its own cluster with respect to the template waveform.

In figure 5, we see the template created from the PC space and all of the waveforms that have lower RMSE (with respect to this template) than 20.75 (first cluster's maximum RMSE value I am taking).

Second Day's Waveform

We can now take the waveform data and compare it with the first day's data to see if they are the same neurons or if there are any changes. Here we compare the projection of the second day's data to the first day's PC space, with the second day's own PC space.

To begin with, figure 6 shows the PC space of second day's waveform data. Compared to the figure 2, this already looks different, as the clusters are not in the same locations. Moreover, there seems to be a change present in the counts, which indicate either less waveforms are captured by the clusters or waveforms are more spread across the cluster, making the cluster's radius bigger. Further analysis can show which is true.

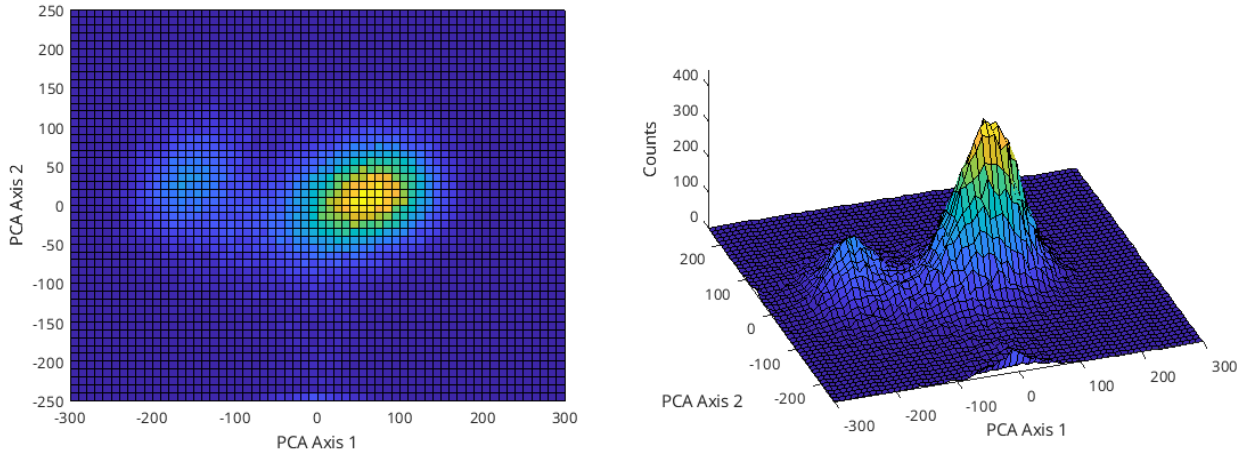

Figure 6: Second day's waveform data plotted in the PC space, taking the first 2 principal components only.

The first 2 principal components of the day 2 analysis can explain 52.29% of the variance in the data, as can be seen in figure 7.

Following the same steps, I first mark the clusters and determine center position and radius for both of them. There appears to be 2 clusters present again, which is expected, one with center position (60 10) and radius 30 and the other with center position (-160 30) and radius again 30. Note again that these are determined by examining by eye and a programmatic approach would be better for professional use.

After marking these clusters, I again find all the waveforms that lie inside these clusters and plot their average to form the template waveform for both of them. Then, using the projection data, I find the template there with the first day's own center and radius. Figure 8 shows 2 cluster's difference between the projected data and its own PCA.

Even though they do not lie perfectly on top of each other, their shapes do look similar. In fact, comparing the projected cluster 1 template with day 2 PCA cluster 1 template by using RMSE gives 7.31, which is below the thresholds for any waveform calculated before. RMSE between cluster 2 lines is 9.04 which is still quite low. We can conclude that these are the same neurons, but there may be a distortion happening in the data that effects the shapes of the templates.

This change in the data may be because of changes in the electrode location or structure. In the class, we have discussed that it is best to recalibrate the equipments that work with implanted electrodes regularly to account for these changes.

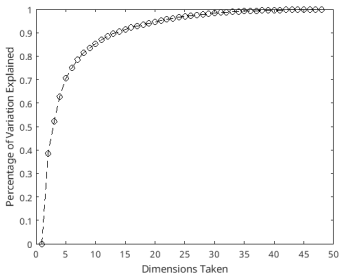

Figure 7: Percentage explained by principal components for day 2 waveform data's PCA.

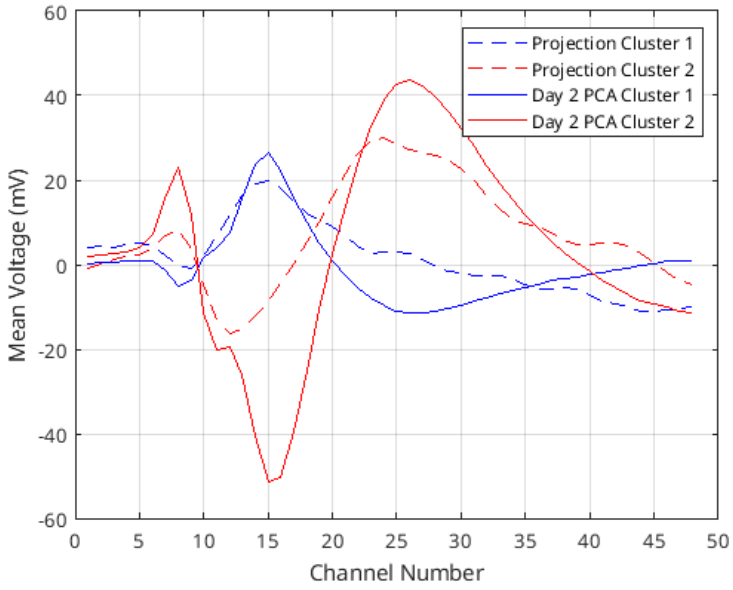

Figure 8: Template waveforms calculated for projection of day 2 onto day 1 PC space and day 2 data's own PCA.