EMG Decomposition

Shakthi Visagan

November 16, 2018

1 Introduction

Electromyographic (EMG) signals are recorded using electrodes placed onto either muscle myofibrils or neurons and contain information about the motor unit action potential trains (MUAPTs) that compose them. Muscle contraction is controlled by the motor units and the action potentials sent by nearby neurons, so studying the MUAPTs from the EMG signal is helpful in studying disorders that affect muscle contractions. Essentially, our goal is to identify the individual MUAPTs given a raw EMG signal so that we can infer what information is being sent by α-motorneurons to the muscles.

2 EMG Signal Composition

The goal is to reduce EMG signals into the individual MUAPTs and this is based on some fundamental assumptions: firstly, that the entire aw EMG signal is comprised of noise and MUAPs. These MUAPs are essentially discharges of the individual motor units and secondly there exists a one to one connection between a characteristic MUAP and the motor unit that created it.

The EMG signal essentially records the action potentials sent by the muscle fibers along the membrane. This muscle fibre action potential (MFAP) is dependent on the radius of the fiber, the conduction velocity, and the distance from the electrode that is being used to detect the MFAP. These MUAPs are essentially a functional sum of the MFAPs of the comprised muscle fibers.

3 EMG Signal Decomposition

3.1 Filtering

The signal will contain frequencies which we will want to remove. We can do this by filtering by some form of band pass filtering to remove low and high frequencies. Sometimes low frequencies associated with the EMG signal are indicative of any DC signals caused by experimental conditions like movement and perspiration. We can choose to cut off frequencies below 5 to 20 Hz and thus force the mean of the signal to be near 0. At the higher frequencies, we cut off around 200 Hz to about 1 kHz. This should be high so we do not remove the rapid

bursts seen in the raw EMG signal but also prevent misinformation due to high sampling and aliasing.

In general, we can use a Butterworth low-pass filter as an infinite impulse response filter. Another approach would be to take the mean of a moving window of the signal as a finite impulse response.

3.2 Detecting Spikes

After filtering the raw signal, the next step is to create a spike train. Usually this is done by amplitude thresh-holding. Thresh-holding interestingly involves compromising between obtaining false-negatives, in the sense that we can miss spikes by setting a high threshold, and obtaining false-positives, in the sense that we can count additional spikes by setting a low threshold. Ideally, a threshold should be set based on the level of noise inherent in the signal estimation. This can be based off of the standard deviation of the signal. However, including the spikes in the calculation of the standard deviation can lead to errors and several corrections exist. Once this step is completed, some post-processing steps can be taken to align the signal and cut off excess spaces at the beginning and the end of the signal.

3.3 Feature Extraction

The signal can be thought of as a high energy high variance data-set, and one way to simplify this data is to perform dimensionality reduction. A common way to approach this is to use Principal Components Analysis (PCA) which finds the axes of high variance inherent in the data. Usually, this is reduced to a small number of principal components not only for visualization, with two or three principal components, but also captures a large substantial amount of the variance. Other methods of dimensionality reduction exist like t-SNE, which is an example of a nonlinear manifold learning technique but proves not only to be a good visualization tool, but also a high dimensional clustering machine learning technique.

3.4 Clustering

The final step apart from validation is the clustering of the signals obtained in the previous step. As mentioned before, t-SNE works as a way to perform low dimensional clustering. Another method is to use something like Gaussian mixtures. What this method assumes is that the data is distributed into clusters of multidimensional Gaussian distributions. Using expectation maximization, one can fit some number of Gaussian distributions to the signals clustered by the previous step. Another method would be to use k-means, which is a nearest-neighborhood algorithm that calculates the distance between various points and assign points to centroids based on the distance of points between the means of the centroids. Usually, the number of clusters determines the number of spikes being sorted so this is an imperative step in determining the correct number of spikes to be clustered.

4 Conclusion

Eventually, we can validate our results using k-fold cross validation and see how much variance we can capture in our signal. We can also manually evaluate this by superimposing our signal with our collected signal. We can also see what each spike from each neuron looks like to see if there's any spikes that are similar and may have been double counted.

References

[1] Stashuk, Dan."EMG signal decomposition: how can it be accomplished and used?." Journal of Electromyography and Kinesiology 11.3 (2001): 151-173.

[2] "T-SNE visualization of large-scale neural recordings." George Dimitriadis, Joana Neto, Adam Kampff. bioRxiv 087395; 10.1101/087395

[3] "A fully automated spike sorting algorithm using t-distributed neighbor embedding and density based clustering." Mohammad Hossein Nadian, Saeed Karimimehr, Jafar Doostmohammadi, Ali Ghazizadeh, Reza Lashgari. bioRxiv 418913; 10.1101/418913

[4] Quiroga, R. Quian, Zoltan Nadasdy, and Yoram Ben-Shaul. "Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering." Neural computation 16.8 (2004): 1661-1687.