Electromyogram (EMG) and Signal Decomposition

Shreesh Karjagi, Project Phase I, Neuroengineering M260 Fall 2022

1. Introduction

Scientists have divided the human body into several distinct systems. One of these systems is the neuromuscular system. This system is made up of two subsystems: the nervous system and the muscular system (Begg et al., 2007). Through nerve signals, the central nervous system (CNS) controls the muscles. Movement and arrangement of limbs are controlled by electrical impulses propagating between muscles, the PNS, and the CNS (Bronzino, 1999). The nerves carry electrical impulses from the central nervous system to the muscles. The skeletal-muscular system consists of muscle groups attached to bones by tendons, and a motion occurs when nerve impulses cause muscle contractions or relaxations that either attract or repel the bone (Begg et al., 2007). Neuromuscular diseases can result in an impaired muscle-nerve connection, leading to pathological non-communication (Lepore et al., 2019). Pathological non-communication at the spinal cord, motor neurons, muscle, or neuromuscular junction modifies the muscle's electrical impulses (Subasi., 2019). Decomposition of an Electromyography (EMG) signal can be characterized as a technique to treat neuromuscular disorders by recording and understanding the action potential of muscles as shown in **Figure 1** and **Figure 2**. EMGs are used to evaluate the health of muscle and nerve cells. EMGs were first recorded mainly for diagnostic reasons (Subasi., 2019). However, as bioelectric stimulation technology has advanced, EMGs have become a critical tool for artificial control of limb movements, such as functional electrical stimulation (FES) and rehabilitation (Bronzino, 1999). This report examines and discusses the key steps and methods (biomedical signal processing techniques, dimension reduction, and classification methods) required in the effective decomposition of an EMG signal. These will include the underlying assumptions and techniques of decomposition, as well as the limitations of every decomposition strategy. The role of EMG signal decomposition in diagnosis of Neuromuscular Disorders will also be discussed.

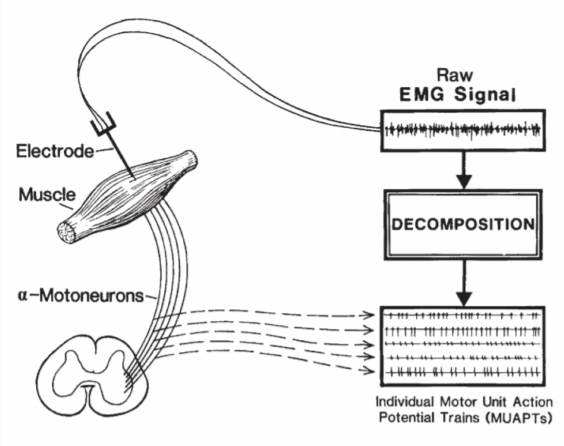

Figure 1: The fundamental structure and decomposition of an EMG signal (from DeLuca et al., 1982).

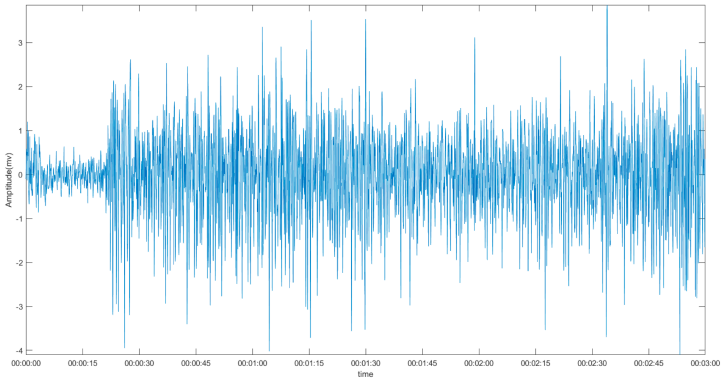

Figure 2: Depicts a raw Amplitude (mV) – Time, EMG signal (from Zhang et al., 2022).

2. Instrumentation and the EMG

Electromyography's most common types are needle or fine wire EMG and surface EMG (sEMG). Electrodes, a signal collection system with signal filters, amplifiers, and visual displays are required to capture EMG signals (Subasi., 2019). The EMG must be sensitive to noise and artifacts in order to obtain an usable signal (Subasi., 2019). As a result, the devices used to produce an EMG signal are built to meet specific parameters, such as filter bandwidth, gain, and input impedance (Preedy & Peters, 2002; Begg et al., 2008).

3. The Different EMG Electrodes

With more precision than sEMG, the wire electrode or needle electrode can reach a specific motor unit and acquire the action potential (Subasi., 2019). A needle electrode consists of a wire attached to an electrode composed of stainless steel as shown in **Figure 3** (Subasi., 2019). A concentric ring needle with a single monopolar electrode is the most frequent form. Single fiber EMG (sFEMG) and macro EMG are two other forms of needle electrodes (Togawa, Tamura, & Oberg, 1997). A wire electrode, used in kinesiology research, is another type of electrode composed of silver, nickel, platinum, or chromium that is more flexible than a needle electrode (Subasi., 2019). The cannula is utilized to insert the electrode wire into the muscle and is afterwards withdrawn (Stålberg., 1980). A ground electrode is utilized because the EMG detects and records voltage relative to the ground (Stålberg., 1980).

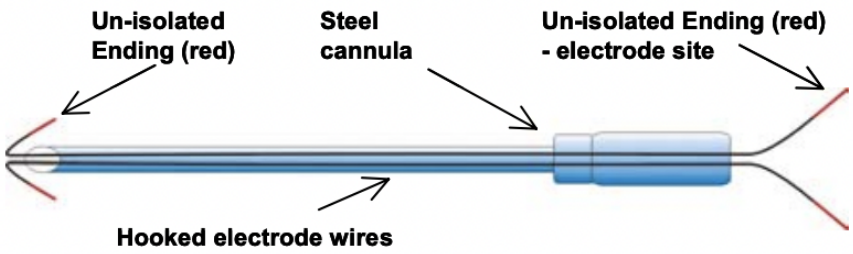

Figure 3: Schematic of a steel cannula holds two unisolated tiny wires for a fine wire electrode. (from Konrad. P, 2005).

4. Signal Acquisition

All sEMG or needle EMG recordings demonstrate voltage potential changes resulting from intracellular nerve or muscle activation (Preston & Shapiro, 2012). The transmission of intracellular electric potentials over extracellular fluid and tissue is referred to as volume conduction. Skin or fat layers are capable of conducting near-field or far-field potentials (Subasi., 2019). Skeletal muscle EMG includes four elements:

1. Insertional activity: caused by needle electrode injection and action in the muscle.
2. Spontaneous activity: recorded with a relaxed, stationary muscle/ relaxed muscle.
3. Motor Unit Action Potential (MUAP) morphology and stability: after determining the starting action, the MUAP waveform and shape can be processed. If their physiological generation is accurately interpreted, the shape of individual MUAPs may reveal the abnormality's cause (Subasi., 2019).
4. MUAP recruitment: The amount of motor units recruited by the CNS can be deduced from EMG waveforms shown in **Figure 4**. These can evaluate physical strength and help discover neuromuscular diseases (Begg et al., 2007).

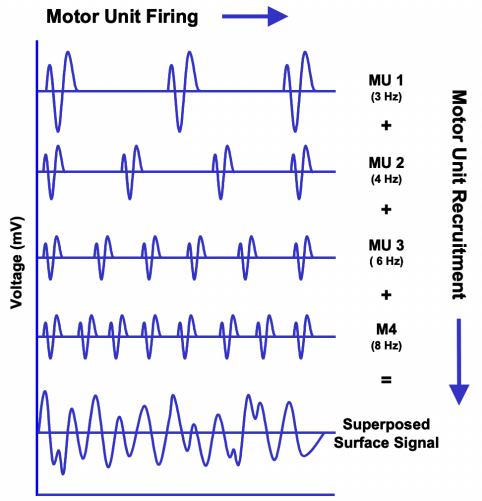

Figure 4: Recruitment and firing frequency of motor units modulates force output and is reflected in the superposed EMG signal. (Adopted & modified from Konrad. P, 2005).

5. Signal Amplification and Filtering

Scientific recommendations for research investigations, Surface Electromyography for the Non-Invasive Assessment of Muscles (SENIAM) and International Society of Electrophysiology and Kinesiology (ISEK), prohibit the use of a smaller band setting, and the goal is to assess the EMG throughout the whole 10-to-500-Hz frequency range (Konrad. P, 2005). According to Konrad (2015), any form of notch filter (to cancel out 50 or 60 Hz noise) is unacceptable since it removes too much EMG signal. Biofeedback devices that employ significantly preprocessed signals should not be utilized in scientific research (Konrad. P, 2005). The signal is amplified, band-pass filtered, sampled, and low-pass filtered (Subasi., 2019). Butterworth or Chebyshev filters are commonly used, and cutoff or band-pass frequencies are rarely needed (Subasi., 2019). This is due to a larger allocation of EMG signal power in the 5- to 500-Hz power density spectra (Subasi., 2019). Intramuscular recordings should be made with a high-frequency cut-off of **450 Hz or a 10–450 Hz band-pass filter** (Subasi., 2019). The previously described levels are perfect filter settings that can be fine-tuned (Begg et al., 2007).

6. Signal Digitization

Analog signals are collected, amplified, and band-pass filtered. Before a signal can be displayed and analyzed on a computer, it must first be converted from analog to digital (A/D) (Subasi., 2019;Konrad. P, 2005). Another critical technical consideration is sampling frequency. zTo accurately "translate" a signal's entire frequency spectrum, the sampling rate of the A/D board must be double the maximum expected frequency of the signal (Subasi., 2019). This relationship is described by Nyquist's Sampling Theorem: when a signal is sampled too low, it becomes indistinguishable or aliases (**Figure 5**). During digitization, the Nyquist rate is double the highest frequency cutoff of the band-pass filter (Subasi., 2019;Konrad. P, 2005). For example, nearly all of the signal power in EMGs is located between 10 and 250 Hz, and scientific recommendations (SENIAM, ISEK) call for an amplifier band setting of 10 to 500 Hz. To avoid signal loss, a sampling frequency of at least 1000 Hz (double band of EMG) or even 1500 Hz would be required (Konrad. P, 2005). A low-pass filter is used in some cases to primary smooth the EMG signal so that a lower sampling rate (50-100 Hz) can be used for digitization (Konrad. P, 2005). The raw unfiltered EMG signal can also be saved in the computer at a sampling rate greater than 2.5 kHz (Subasi., 2019). It should be noted that, as previously explained, the signal should be influenced by high-frequency noise and band-pass filtered (Subasi., 2019). The analog EMG signal can be converted to a digital signal using an A/D converter with a resolution of 8 to 12 bits (Subasi., 2019). Digitized signals can also be normalized, as in force/torque experiments where normalization is performed with respect to the maximum voluntary contraction (MVC) (**Figure 6**). This procedure requires that the subjects be in good order and trained to avoid variability and

experimental incompatibilities (Subasi., 2019). Normalization can also be accomplished in terms of joint angle or muscle length, reduction or extension velocity, or applied load (Begg et al., 2007).

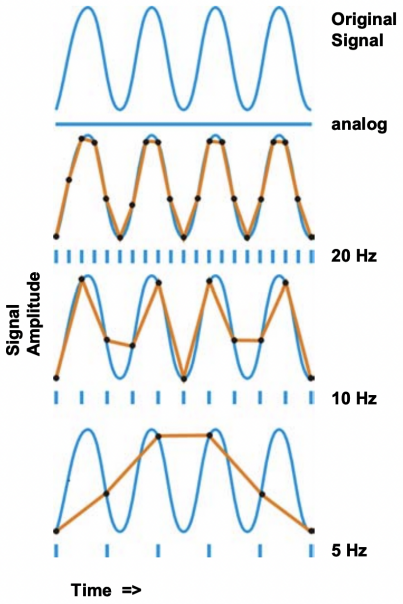

Figure 5: A/D sampling frequency and a digitized signal. Signal information is lost at low frequencies (low traces). (Adopted from Konrad. P, 2005).

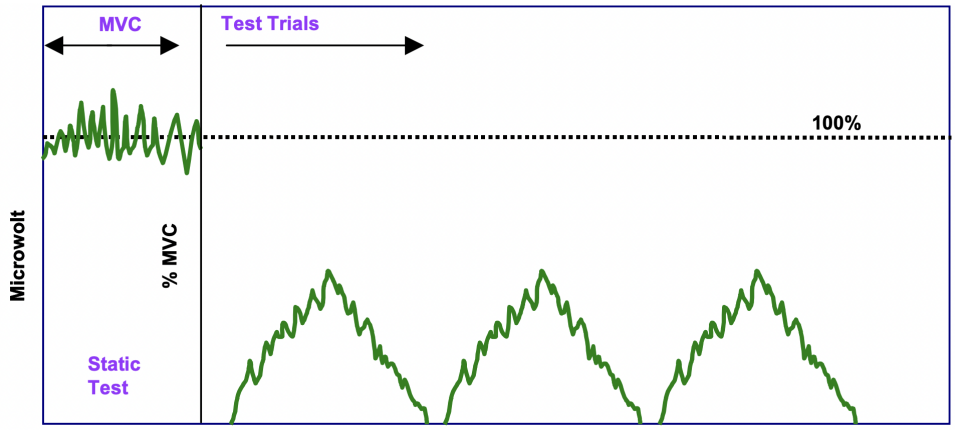

Figure 6: MVC Normalization. Each muscle is static MVC contracted before the test/exercise. This MVC innervation level is the reference for future trials (100%) (Adopted from Konrad. P, 2005).

7. Spike Detection and Alignment

According to Konrad (2005), the EMG interference pattern is random because the actual set of recruited motor units varies continuously within the diameter of available motor units and the motor unit action potentials superpose in an unpredictable manner. As a result, the precise shape of a raw EMG burst cannot be replicated. To address this issue, the non-reproducible portion of the signal is reduced using digital smoothing techniques

that highlight the average development trend of the signal (Konrad. P, 2005). The steep amplitude spikes are removed, and the signal is given a "linear envelope." There are two established algorithms: The moving average (MovAg) The sliding window method averages a given amount of data over a user-defined time window. When applied to rectified signals, it is known as the Average Rectified Value (ARV), and it serves as a "estimator of amplitude behavior" (SENIAM) (Konrad. P, 2005). It relates to information about the area under the selected signal epoch (**Figure 7**). Root Mean Square (RMS) Based on the square root calculation, the RMS reflects the mean power of the signal (also called RMS EMG) and is the preferred recommendation for smoothing (Basmajian, J. V., , & De Luca, C. J., (1962); De Luca, C. J., & Knaflitz, M. (1992)). The root mean square value (RMSV) of the entire EMG signal will be computed using the "**rms**" **MATLAB® function**, with a 25-ms averaging window, in accordance with what De Luca (1992) has reported. To properly compute the threshold value, the acquisition of the EMG signal should begin prior to the subject's first movement, so that the window between the beginning of signal acquisition and the subject's first movement is ideally characterized only by electric noise (Mengarelli et., 2016). Consequently, a default interval of 4 s after the beginning of the signal recording will be selected for the calculation of the threshold, which is obtained as the mean value of the RMSV noise amplitude plus two standard deviations (De Luca, C. J., & Knaflitz, M. (1992)).

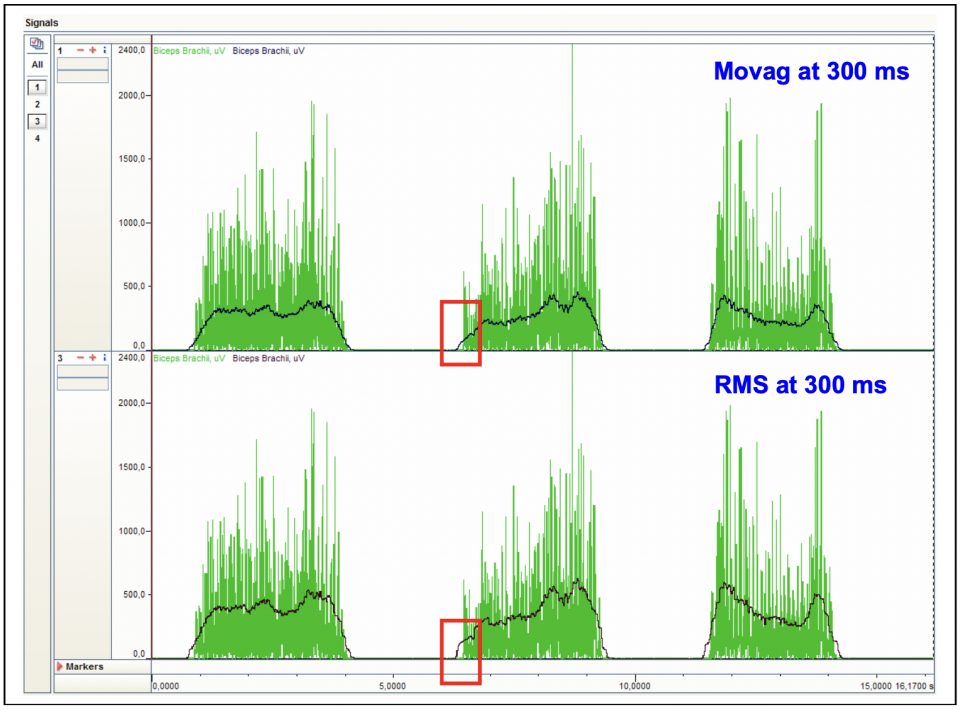

Figure 7: Using the same window width, compare two smoothing algorithms. RMS algorithm (lower trace) shows higher EMG amplitude than MovAg. (Adopted from Konrad. P, 2005).

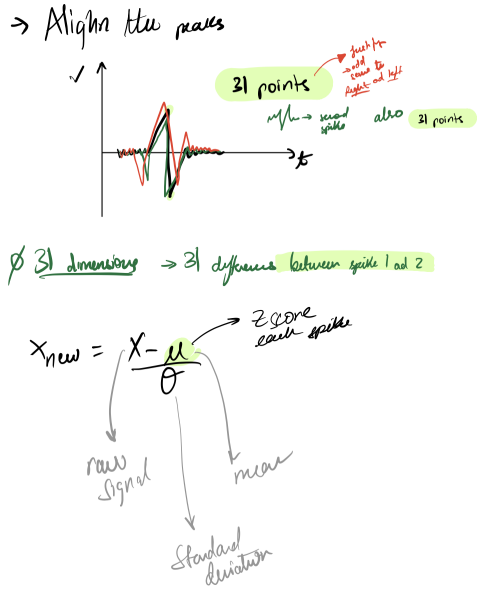

Figure 8: General approach to aligning the peaks (from Jonathan's discussion)

9. Extracting Features and Principle Component Analysis (PCA)

Here, a PCA, or Principal Component Analysis, is required. Using PCA, we may reduce the enormous dataset to a more manageable size while retaining the original data. This can be divided into 6 steps. (1) data standardization. This is done so that data can be distributed around a mean of zero and a standard deviation of one, allowing all covariances of the data to be compared on the same scale. Manually, we would utilize the formula shown in **Equation 1**. (2) Involves the construction of a covariance matrix, in which each row and column represent a feature. Therefore, each entry represents the covariance between the respective row and column. If the covariance for a given pair of features is positive, then both of these features tend to increase simultaneously. When the covariance is negative, one characteristic tends to increase as the other characteristic decreases. If the covariance is zero, then the two characteristics have no relationship. The matrix form is primarily used to visualize these values, allowing us to identify patterns and validate predictions based on data trends. Manually, we would combine the covariance and standardization equations as shown in **Equation 2**. (3) eigenvectors and eigenvalues are extracted from the covariance data to identify the factors that generate the most covariance and are therefore the most significant. The significant formula is Av=v, where v is the eigenvector and lambda is the scalar or eigenvalue. To determine the eigenvectors, we must identify the vector that points exactly in the direction in which a transformation matrix pushes things. (4) Arrange the eigenvalues in descending order. This allows us to distinguish high variance principal components with more information from those with low variance, and to prepare for step five using a bar graph or scatter plot. (5) Picking our essential components. The primary

objective of this phase is to lower the number of dimensions and eliminate the principal components with little variance, which is reliant on the results of (4). For instance, if 95% of the information is contained in the first two major components, we would like to retain only those two components. (6) concludes with the transformation and reduction of the dataset to new dimensions. Here, we only need to arrange the eigenvectors representing the principal components we wish to use as the columns of a matrix, which will result in a new projection matrix: XW=X'. X' represents the transformed dataset. This is possible with the scikit-learn function. Despite the fact that the code is provided as a reference in this article, it is important to emphasize the primary steps for this function. First, we would load in the data, normalize using a scalar, import the PCA transformer, fit the pca transformer to the dataset, and label the data for this plot.

$$\frac{x-\mu}{\sigma}$$

Equation 1: Standardize the data (from Jonathan's discussion).

$$\text{Covariance of the data: } \frac{\sum (x_i-\bar{x})\cdot(y_i-\bar{y})}{N-1}$$

$X_1, X_2 \cdots X_{31}$

	Y_1	X_2	X_{31}
Y_1	Var(x1)	Cov(x1,x2)	
X_2		Var(x2)	
X_{31}			Var(x31)

Equation 2: Covariance of the data (from Jonathan's discussion).

<u>10. K-Nearest Neighbor (Clustering and Classification)</u>

The k-NN is an effective nonparametric method in many cases (Hand, Mannila, & Smyth, 2001). According to Subasi (2019) Data record *t* is classified by its k nearest neighbors. In general, majority voting is used to classify *t* with or without distance-based weighing (Subasi., 2019). To use k-NN, a proper value for k is needed, and classifier success depends on this value (Subasi., 2019). The k-NN classifier is k-biased (Subasi., 2019). There are several ways to select the k value, but one is to run the classifier multiple times with different k values and pick the one with the highest accuracy (Subasi., 2019). k-NN has a high cost for classifying new instances because most computation occurs at classification time, not with the first training examples (Subasi., 2019). Methods such as indexing training examples (Mitchell, 1997; Bishop, 2007) are used to reduce query time computation. k-NN uses all training data to classify cases (Subasi., 2019). Because k-NN is a sluggish learning method, it can't be used with a huge repository (Subasi., 2019). To improve its efficiency, find samples that represent the full training data for classification,

then develop an inductive learning model from the training dataset using this model for classification (Subasi., 2019). Because k-NN is a basic yet successful classification approach, it encourages the building of a k-NN model to improve its competence and classification accuracy. If Euclidean distance is employed to quantify similarity, multiple data points with the same class label are close (Subasi., 2019). If these samples are used to model the entire training dataset, it will reduce the number of data points for classification, improving efficiency (Subasi., 2019). If a sample covers a new data point, its label classifies it. If not, the distance between the new data point and each sample's nearest border must be determined, and the nearest boundary should be used (Subasi., 2019). New data can be categorized using k-NN (Guo et al., 2003). In model creation as shown in **Figure 9**, each data point has its greatest local neighborhood covering the most similar data points (Subasi., 2019). Based on these local neighborhoods, every cycle may have the largest local neighborhood. This greatest global neighborhood can be used to sample its data points (Subasi., 2019). Repeat the previous step until all data points are covered by samples (Guo et al., 2003).

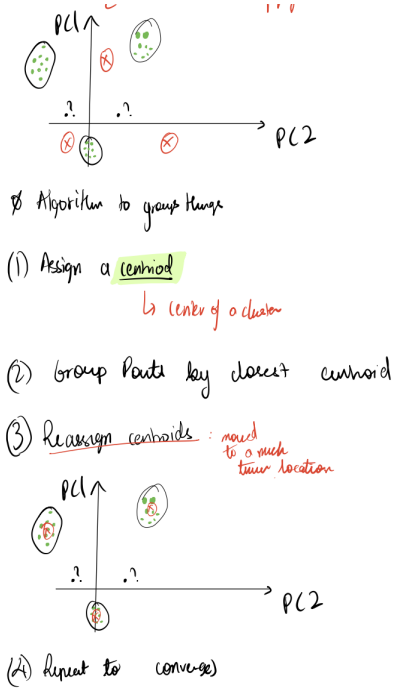

Figure 9: k-NN model creation schematic (from Jonathan's discussion).

11. Uses of EMG Signals in Diagnosis of Neuromuscular Disorders

Needle EMG diagnosis neuromuscular pathology. Doctors capture needle EMG during muscle contraction in patients with muscular weakness (Subasi., 2019). Single MUAP waveforms show muscular CNS responsiveness (Subasi., 2019). This data may identify muscular discomfort, nerve injury, pinched nerves, and muscular dystrophy (MD) (Subasi., 2019). According to Subasi (2019) by monitoring motor unit accomplishment over a given time period, needle EMG is also used to evaluate whether nerve wounds

heal and return to normal with complete muscle response. Diagnostic EMG detects abnormal muscle activity during rest (Subasi., 2019). Normal muscle relaxation is electrically quiet, but spontaneous muscular activation and seizures may cause irregular waveforms (Sornmo & Laguna, 2005).

13. Discussion

Classifying EMG data using machine learning may assist neurologists diagnose neuromuscular problems. To create an automated diagnostic system, the EMG signal is pre-processed and features are retrieved using Fourier, wavelet, autoregressive (AR), or other signal-processing methods. After EMG feature extraction, the data is sent into a classifier such as ANNs, k-NN, SVM, or decision tree to detect illness (Begg et al., 2008). Developing an automated method to diagnose neuromuscular illnesses is difficult. Patients' EMG signal characteristics vary widely. Age affects loudness and duration of EMG signal. To solve this challenge, we need to design a signal-processing technology that can store or locate distinct EMG data.

12. References

Begg, R., Lai, D. T., & Palaniswami, M. (2007). Computational intelli-gence in biomedical engineering. CRC Press.

Bronzino, J. D. (1999). Biomedical engineering handbook. Vol. 2. CRC Press.

Lepore, E.; Casola, I.; Dobrowolny, G.; Musarò, A. (2019) Neuromuscular Junction as an Entity of Nerve-Muscle Communication. Cells, 8, 906. https://doi.org/10.3390/cells8080906

De Luca CJ, Le Fever RS, McCue MP, Xenakis AP. (1982) Behavior of human motor units in different muscles during linear-varying contractions. J Physiol (London);329:113–28.

Subasi, A. (2019). Practical guide for biomedical signals analysis using machine learning techniques: A MATLAB based approach. Academic Press.

Zhang, Chizhou, and Tao Sun. (2022). "Discussion of the Influence of Multiscale PCA Denoising Methods with Three Different Features" Sensors 22, no. 4: 1604. https://doi.org/10.3390/s22041604

Preedy, V. R., & Peters, T. J. (Eds.). (2002). Skeletal muscle: Pathology, diagnosis and management of disease. Cambridge: Cambridge University Press.

Begg, R., Lai, D. T., & Palaniswami, M. (2008). Computational intelligence in biomedical engineering. Boca Raton, FL: CRC Press.

Togawa, T., Tamura, T., & Oberg, P. A. (1997). Biomedical transducers and instruments. Boca Raton, FL: CRC Press.

Stålberg, E. (1980). Macro EMG, a new recording technique. Journal of Neurology, Neurosurgery & Psychiatry, 43(6), 475-482.

Konrad, P. (2005). The abc of emg. A practical introduction to kinesiological electromyography, 1(2005), 30-5.
Preston, D. C., & Shapiro, B. E. (2012). Electromyography and neuromuscular disorders: Clinical-electrophysiologic correlations. London: Elsevier Health Sciences.
Basmajian, J. V., , & De Luca, C. J., (1962). Muscles alive. Their functions revealed by electromyography. Academic Medicine, 37(8), 802.
De Luca, C. J., & Knaflitz, M. (1992). Surface electromyography: What's new? (pp. 1-22). Torino: CLUT.
Mengarelli, A., Cardarelli, S., Verdini, F., Burattini, L., Fioretti, S., & Di Nardo, F. (2016, August). A MATLAB-based graphical user interface for the identification of muscular activations from surface electromyography signals. In 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC) (pp. 3646-3649). IEEE.
Deutsch, H. P., & Beinker, M. W. (2019). Principal component analysis. In Derivatives and internal models (pp. 793-804). Palgrave Macmillan, Cham.
Hand, D. J., Mannila, H., & Smyth, P. (2001). Principles of data mining (adaptive computation and machine learning). Cambridge, MA: MIT Press
Bishop, C. M. (2007). Pattern recognition and machine learning (information science and statistics). New York: Springer.
Mitchell, T. M. (1997). Machine learning. Vol. 45(37) (pp. 870–877). Burr Ridge, IL: McGraw Hill.
Guo, G., Wang, H., Bell, D., Bi, Y., & Greer, K. (2003). KNN model-based approach in classification. In Presented at the OTM confederated inter- national conferences “On the Move to Meaningful Internet Systems” (pp. 986–996). Springer.

Resources so far:

Matlab code to find linear envelope of a signal and compute and plot spectra

I will be using (a modified version) of the following code from William Rose’s (2019) “Electromyogram analysis.”

```
% Low Pass filter frequency = 50 Hz. Sampling rate = 10000 Hz.
When setting up digital filters in Matlab, the Nyquist frequency should be used
to normalize the frequencies.
x=load ('EMG_example_20s_2000Hz-2022.csv');
Fs=10000;
Fnyq=Fs/2
% Line above is Nyquist frequency. Next line rectifies the signal after
removing the mean value.
y=abs(x-mean(x));
% Next line defines the desired final cutoff frequency, in Hz.
fco=20
% Next line creates a 2nd order Butterworth low pass filter. The cutoff
frequency is adjusted upward by 25% because the filter will be applied twice
```

```
(forward and backward). The adjustment assures that the actual -3dB frequency
after two passes will be the desired fco specified above. This 25% adjustment
factor is correct for a 2nd order Butterworth; for a 4th order Butterworth used
twice, multiply by 1.116.
[b,a]=butter(2,fco*1.25/fnyq);
% Next line filters the rectified data forward and backward.
z=filtfilt(b,a,y);
% Next lines create a vector of time values for plotting, plot the data versus
time: raw data in blue, offset vertically for visibility, rectified data in
green, and linear envelope in red. Axis labels and a legend are also added.
plot(t,x-mean(x)-4,'b',t,y,'g',t,z,'r');
xlabel('Time(s)');ylabel('EMG(V)');
legend('Raw (offset)','Rectified','Linear envelope');
%Next lines create a vector of frequencies present in the spectra, up to the
Nyquist frequency.
N=length(x);
freqs=0:SR/N:fnyq;
%Next: compute Fast Fourier Transform (FFT) and plot the amplitude spectrum, up
to the Nyquist frequency.
xfft = fft(x-mean(x));
figure; >> plot(freqs,abs(xfft(1:N/2+1)));
%Next: compute and plot the power spectrum, up to the Nyquist frequency.
Pxx = xfft.*conj(xfft);
figure;
plot(freqs,abs(Pxx(1:N/2+1)));
```