SVM and ANN based Classification of EMG signals by using PCA and LDA

Md Mobasser Alam Midya (001910801057), Aditya Mishra (001910801159),
Rabin Chalak (001910801062), Rukhshanda Hussain (001910801068)
B.E.E. IV, Department of Electrical Engineering, Jadavpur University, Kolkata, India

***Abstract*—In the following project the experiments have been conducted employing Support vector machines (SVMs) and Artificial Neural networks (ANNs) for classification of Electromyography (EMG) signals. The classifiers aim to classify ten individual and combined finger motions present in ENG wave information to one of the predefined set of movements. We have explored the effectiveness of both the methods in two settings namely, with dimensionality reduction and without dimensionality reduction prior to classification. For dimensionality reduction the two commonly know methods, Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) are utilized to extract the features with the highest energy content. These extracted features are the input to the classification networks. Additionally, we have also explored several types of kernels, number of features considered and other relevant hyperparameters both in preprocessing and classification modules to find the most effective setting for this task. https://github.com/Ruxie189/Foreground_SVM.git**

***Index Terms*—Support vector machines, Artificial Neural networks, Electromyography, Principal Component Analysis, Linear Discriminant Analysis**

I. Introduction

Recently, biomedical signals have been used for communication in Human-Computer Interfaces (HCI) for medical applications. For instance, the myoelectric signals (MES) generated in human muscles have unidimensional patterns. Consequently, upsampling these signals to get Electromyography (EMG) signals followed by analysis of it can give relevant insights to electrical activity of muscles. EMG signals can be determined by placing electrodes (both needle and surface type) on the skin. Electrical activity of a motor unit can provide information for identification of underlying muscular disorder [1]. It can be extended to several other applications such as, human machine interaction, motor control diseases, rehabilitation engineering [2] etc. Therefore, classification of EMG signals have gathered immense recognition in medical sciences and biomedical engineering. Over the decades several pattern recognition algorithms have been developed for studying prosthetic control, utilizing EMG signal classification by logging a set of MES in a proper range of frequencies to classify the corresponding EMG signals. Typically a pattern recognition system could be used to classify the acquired EMG signals into one of the predefined set of movements [3].

In the recent years there has been analysis on the relative performance of support vector machines (SVMs) and neural networks for classification of EMG signals from normal, myopathy, and neuropathy subjects [4]. Authors in [1] proposed a novel PSO- SVM model combining particle swarm optimization with SVMs for diagnosis neuromuscular disorder via EMG signal classification. Identification of EMG signals also enables designing of myoelectric control systems. It involves two main steps, Classification of EMG signals followed by estimation of operator's joint angles. To classify a multichannel surface electromyography signals with the aim of controlling myoelectric prostheses, a support vector machine (SVM) approach has been applied it being a suitable for real-time application [5]. Authors in [6] used 16 channels for classifying the individual finger movements consisting of 9 classes. [7] in their works, employ feature extraction in time domain followed by classification of a 10 classes of finger movements via SVM. Bayesian vote was adopted for post processing.

Feature extraction and classification have been considered as two key issues to design prosthetic/assistive devices or any useful application based on EMG [8]. Feature extraction is done to define a feature vector from the original EMG signals, whereas a classifier is used to discriminate these feature vectors and group them into different classes. A significant issue in feature extraction from EMG signals is the low quality of signals. It is not always strictly repeatable, and may sometimes even be contradictory since it may be modified by many factors such as muscular fatigue, electrode shift, sweat, changing in thickness of skins, tissues. Furthermore, feature extraction of features from residual muscles of an amputee or disabled is somewhat difficult. Another challenge is to develop an EMG based myoelectric control system that can simultaneously allow movement of multiple degrees of freedom (DOFs). Despite several EMG recognition methods, proposed over the decades, more proficiency is required for controlling multiple DOFs. Patter recognition systems have shown greater efficiency in achieving high accuracy values being able to discriminate amongst multiple patterns.

The performance of better discrimination between individual movements largely depends on the proper presentation of the EMG signals. It is done mainly in the pre-processing stage. Features can be extracted both in time domain and time-frequency domain. Many time domain features such as root mean square (RMS) value, slope sign change (SSC), zero crossing (ZC), waveform length (WL), willison amplitude, auto regressive (AR) model etc. and time-frequency domain features such as wavelet transform (WT), fast fourier transform (FFT) show good performance [9]. Supervised learning methods includes support vector machine (SVM), artificial neural networks (ANN) [10], k-nearest neighbor (KNN) etc. and unsupervised learning methods are K-means Clustering, Hidden Markov Models, Self Organizing Map etc. In this work, experiments have been conducted to estimate the best performing model among SVM [11] and ANN [10], with or without dimensionality reduction using Principal component analysis (PCA) [12] and Linear discriminant analysis (LDA). So, by summarizing, this project has the following two-fold contribution:

1) We have used two different data analysis methods named Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) for dimensionality reduction of the feature set for the sake of reducing the computational cost.
2) We have used two different classifiers SVM and MLP for the classification of different action classes both with and without LDA and PCA as feature extractors.

II. Dataset Description

The EMG signal database for this work consists of 10 classes having both individual and combined finger movements. The dataset is collected from the work presented by Dr. Rami Khushaba and his group of University of Technology Sydney [7].

Eight subjects, six males and two females, aged between 20 and 35 years were recruited to perform the required fingers movements. The subjects were all normally limbed with no neurological or muscular disorders. Ten classes of individual and combined fingers movements were implemented including: the flexion of each of the individual fingers, i.e., Thumb (T), Index (I), Middle (M), Ring (R), Little (L) and the pinching of combined Thumb–Index (T–I), Thumb–Middle (T–M), Thumb–Ring (T–R), Thumb–Little (T–L), and finally the hand close (HC) as shown in the Figure 1.

In total each of the 10 classes comprise of 6 files containing 20,000 EMG signal data having 2 EMG channels within it. Therefore the dimesion of the collected data becomes $[(8x10x6)x(20000x2)]$ where $(20,000x2)$ is featured for rows and column of the matrix respectively. Additionally, the amplitude of an EMG signal is randomly stochastic with a Gaussian distribution ranging from 0 to 10 mV, peak to peak. Measurement of amplitude involves two parameters, **the root-mean-square (RMS)** value and **the mean absolute (MA)** value. There are tow main concerns influencing the fidelity of an EMG signal:

1) **Signal to noise ratio** - the ratio of the energy in the EMG signal to the energy in the noise signal.
2) **Distortion of the signal** - the relative contribution of any frequency component in the EMG signal should not be altered.

III. Methodology

The first stage involves dimensional reduction and feature extraction using two commonly used methods namely, Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA). For the classification task we explore the effectiveness two different classifiers, Support Vector Machines (SVMs) and ANNs with Multilayer Perceptron (MLP).

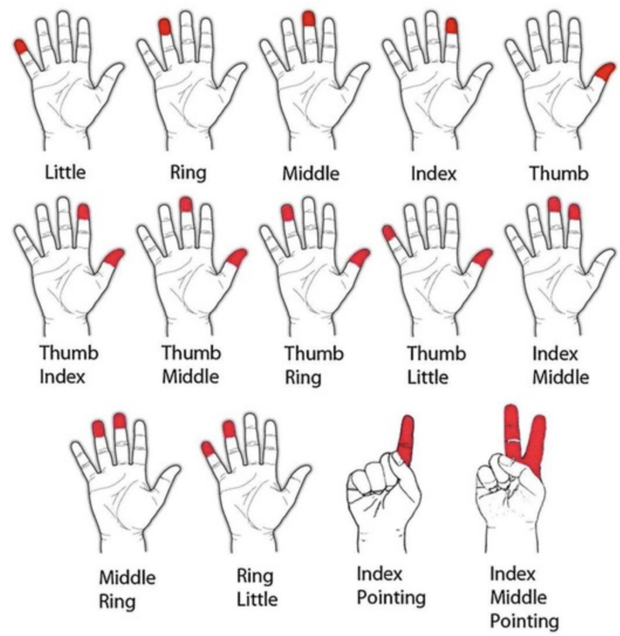

Fig. 1: Ten classes comprising of various finger movements in the EMG signal dataset.

A. Dimensionality Reduction for feature extraction

*1) **Principal Component Analysis (PCA)**:* Principal component analysis (PCA) [12] is a popular technique for analyzing large datasets containing a high number of dimensions/features per observation, increasing the interpretability of data while preserving the maximum amount of information, and enabling the visualization of multidimensional data. Formally, PCA is a statistical technique for reducing the dimensionality of a dataset. This is accomplished by linearly transforming the data into a new coordinate system where (most of) the variation in the data can be described with fewer dimensions than the initial data. Many studies use the first two principal components in order to plot the data in two dimensions and to visually identify clusters of closely related data points. The process of obtaining principal components from a large raw dataset is explained in the following steps:

1) Organize a data set as an m × n matrix, where m is the number of measurement types and n is the number of trials.
2) The mean of the entire data should be calculated for each of the data dimensions in the whole data set. Then the data is normalized so that PCA works properly. This is done by subtracting the respective means from the numbers in the respective column.
3) Then we have to calculate the Covariance Matrix corresponding to our raw dataset using the following formula in Equation 1 where x = arithmetic mean of data X, y = arithmetic mean of data Y, n = number of observations.

$$cov(x,y) = \frac{1}{n-1}\sum_{i=1}^{n}[(x_i - \bar{x})(y_i - \bar{y})] \qquad (1)$$

4) Calculate the Eigenvectors and Eigenvalues of the Co-

variance Matrix.

5) The next step is to choose the components (i.e. the eigenvectors) and forming the feature vector. In general, once eigenvectors are found from the covariance matrix, the next step is to order them by eigenvalue, highest to lowest. This gives the components in order of significance. After that, the components with lesser significance can be ignored as per the requirement of the researcher. The feature vector is constructed by taking the eigenvectors that we want to keep from the list of eigenvectors and forming a matrix with these eigenvectors in the columns.
6) Once the components (eigenvectors) are chosen to keep in the data and a feature vector has been formed, then the transpose of the vector should be calculated and multiplied on the leftover of the scaled original data set, transposed.

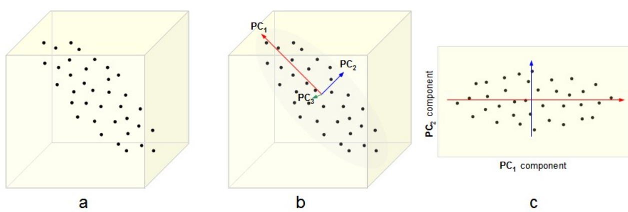

Fig. 2: Dimensionality reduction by PCA. [13]

So, thus PCA gives us multiple principal components for multidimensional data (PCs $<=$ Dimension of the data), but it tries to squeeze most of the information within the initial variables into the first components, hen maximum remaining information in the second component, and so on. But the new variables being a mixture of initial variables are combined in such a way that they are uncorrelated.

*2) **Linear Discriminant Analysis (LDA):*** Linear discriminant analysis (LDA) is a method used to find a linear combination of features that characterizes or separates two or more classes of objects or events. The resulting combination may be used as a linear classifier, or, more commonly, for dimensionality reduction before later classification. While PCA chooses new axes for dimensions such that variance (and hence the 'shape') of the data is preserved, LDA chooses new axes such that the separability between two classes is optimized, and hence is a supervised technique. Both, however, represent the newly formed dimensions in terms of linear combinations of dimensions in the dataset.

Linear Discriminant Analysis is a three-step process:

1) Calculate the 'separability' between the classes. It is defined as the distance between the mean of different classes and allows for the algorithm to put a quantitative measure on 'how difficult' the problem is (closer means = harder problem). This separability is kept in a 'between-class scatter matrix'.
2) Compute the within-class variance or the distance between the mean and the sample of every class. This is another factor in the difficulty of separation, higher variance within a class makes the clean separation more difficult.
3) Construct a lower-dimensional space that maximizes the between-class variance ('separability') and minimizes the within-class variance. LDA can be computed using singular value decomposition, eigenvalues, or using the least-squares method.

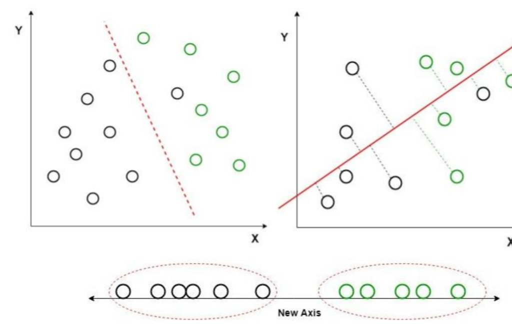

Fig. 3: Three steps in Linear Discriminant analysis [14]

B. Classifiers

*1) **Support Vector Machines (SVM):*** The SVM proposed by Cortes et al. [11] is a fully supervised machine learning algorithm widely used for several classification tasks. The objective of the SVM is to find a hyperplane in an N-dimensional space where N is the number of features in the dataset, that distinctly classifies the data points. In order to separate the two classes of data points, there are many possible hyperplanes or the decision boundaries that could be chosen. The objective involves finding a plane that has the maximum margin, i.e the maximum distance between data points of both classes. This is shown in Figure 4. Maximizing the margin distance provides some reinforcement so that future data points can be classified with more confidence. Data points falling on either side of the hyperplane can be attributed to different classes.

For non-linearly separable data often a kernelized SVM is used. A kernel is a measure of similarity between data points which is utilised in kernelized SVM to extract information about the similarity of two data points in original and transformed feature space. Several types of kernel functions include RBF, Polynomial, sigmoid. In this work we have used a Gaussian kernel that can be mathematically expressed as,

$$G_{ND}(x, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp \frac{-|x|^2}{2\sigma^2} \tag{2}$$

where x is a vector and σ is the standard deviation.

*2) **Artificial Neural Networks (ANN):*** Artificial Neural Networks (ANN) [10] are multi-layer fully-connected neural nets consisting of an input layer, multiple hidden layers, and an output layer as shown in Figure 5. Every node in one layer is connected to every other node in the next layer. The network can be made deeper by increasing the number of hidden layers in them. In our methodology we have adopted

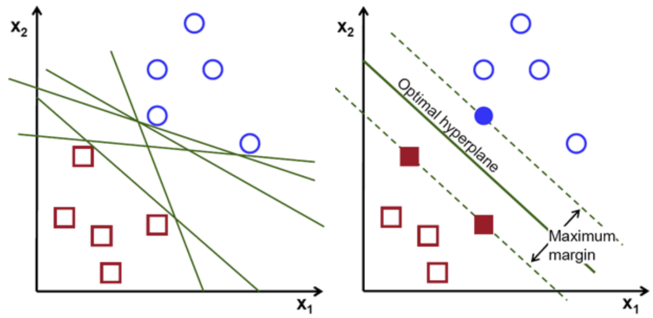

Fig. 4: Possible hyperplanes vs. optimal hyperplane [15]

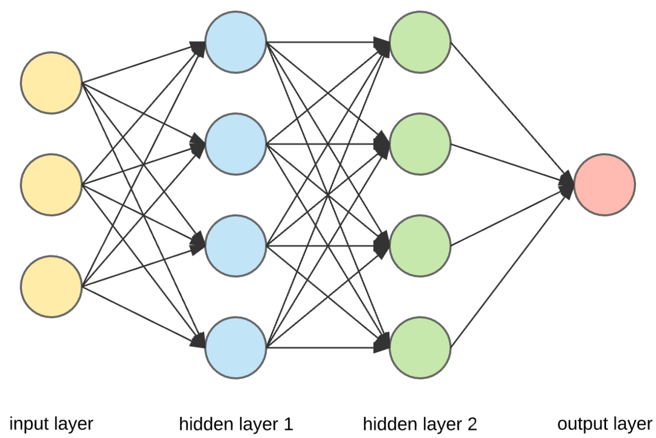

Fig. 5: Basic structure of an Artificial Neural Network with three layers mainly, an input, 2 hidden and an output layer. [16]

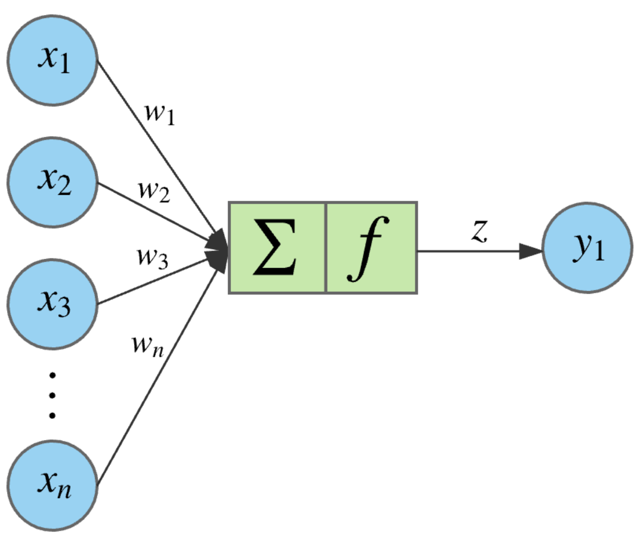

Fig. 6: Multiplication of weights for each connections to the inputs [16]

multilayer perceptrons (MLPs) with back propagation. The input values to a processing element, i_n, are multiplied by a connection weight, $w_{n,m}$ as shown in Figure 6. It is through the adjustment of the connection strengths or weights that learning is emulated in ANNs. All of the weight-adjusted input values to a processing element are then aggregated using a vector to scalar function such as summation averaging, input maximum, or mode value to produce a single input value to the node. Once the input value is calculated, the processing element then uses a transfer function to produce its output (and consequently the input signals for the next processing layer). The transfer function transforms the node's input value. Typically this transformation involves the use of a sigmoid, hyperbolic-tangent, or other nonlinear function. In this way, the output of every node is calculated as shown in Equation 3 where, w is the weight matrix and b is the bias vector. This is followed by computation of error in between the desired output and output obtained.

$$\hat{y} = \sum_{i=1}^{n} (w_i \cdot x_i + b) \tag{3}$$

The next stet involves travelling back from the output layer to the hidden layer to adjust the weights such that the error is decreased, also called as back propagation. The Back propagation algorithm in neural network computes the gradient of the loss function for a single weight by the chain rule. It efficiently computes one layer at a time, unlike a native direct computation. It computes the gradient, but it does not define how the gradient is used. It generalizes the computation in the delta rule.

IV. Experimental Results

In this section we analyse the classification results obtained on the dataset described in the previous section. The section has been divided into two subsections, one each for the SVM and ANN classifiers. We evaluate each of these classifiers on their performances on the dataset individually, as well as by using PCA and LDA as feature extraction schemes. For LDA, we use 5 features and for PCA, we reduce the features to 100 from 20000 features of the original dataset. We use three different splits to test the models, the train and test set sized being in the ratio $1:1$, $2:1$ and $5:1$, respectively. Results have been reported on the test splits only.

A. Evaluation metrics

The following metrics are used for the above classification task. For a batch size of n, TP, TN, FP, and FN represent the true positive, true negative, false positive and false negative respectively per class.

1) **Accuracy:** This metric is used to describe the performance of the model across all classes. Therefore, the accuracy can be define as the ratio between the number of correct predictions to the total number of predictions. Mathematically,

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

2) **Precision:** Precision can be calculated as the ratio between the number of Positive samples correctly classified to the total number of samples classified as Positive, either correctly or incorrectly. It measures the ability of

the model to classify a sample as positive that can be mathematically expressed as,

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

3) **Recall:** It is calculated as the ratio between the number of Positive samples correctly classified as Positive to the total number of Positive samples. The recall measures the model's ability to detect Positive samples as,

$$Recall = \frac{TP}{TP + FN} \quad (6)$$

B. SVM

Table I shows the performance of the SVM classifier on the dataset. The confusion matrices obtained for each case has also been reported in Figure 8.

TABLE I: Classification results on various Train-Test Splits using various feature extraction techniques and SVM Classifier

Split	Method	Subjects	Accuracy	Precision	Recall
1:1	**SVM**	300	0.17	0.10	0.17
	PCA + SVM	300	0.17	0.06	0.17
	LDA + SVM	300	**0.33**	**0.27**	**0.33**
2:1	**SVM**	200	0.17	0.07	0.17
	PCA + SVM	200	0.17	**0.13**	0.17
	LDA + SVM	200	**0.18**	**0.13**	**0.18**
5:1	**SVM**	100	**0.19**	0.07	**0.19**
	PCA + SVM	100	0.15	**0.09**	0.15
	LDA + SVM	100	0.12	**0.09**	0.12

C. ANN

The performance of the ANN classifier has been recorded in Table II. The confusion matrices obtained for each case has also been reported in Figure 9. Emboldened values in the Table II denote the best results obtained in a certain split.

TABLE II: Classification results on various Train-Test Splits using various feature extraction techniques and ANN Classifier

Split	Method	Subjects	Accuracy	Precision	Recall
1:1	**ANN**	300	0.11	0.13	0.11
	PCA + ANN	300	0.14	0.16	0.14
	LDA + ANN	300	**0.30**	**0.28**	**0.30**
2:1	**ANN**	200	0.12	0.14	**0.13**
	PCA + ANN	200	**0.14**	0.10	**0.13**
	LDA + ANN	200	0.12	**0.17**	0.12
5:1	**ANN**	100	**0.11**	0.10	**0.11**
	PCA + ANN	100	0.06	0.05	0.06
	LDA + ANN	100	**0.11**	**0.11**	**0.11**

D. Comparison

Figure 7 compares of the performances of SVM and ANN. We can observe that the SVM classifiers generally perform better than the ANN models, which might be attributed to the lack of sufficient training data.

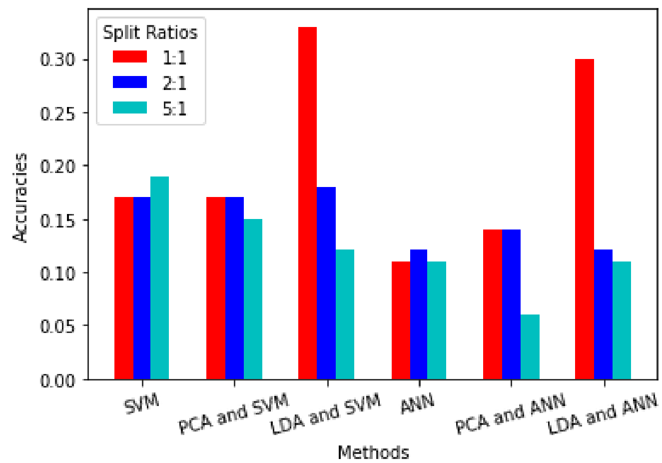

Fig. 7: Confusion Matrices for ANN Trainings.

V. Conclusion

Data plays a key role in the performance of these models even though immense research efforts are being employed. Uncertainty of datasets will inevitably lead to models performing poorly if it is too prevalent in the dataset. Our experimental results show that the results obtained from the original feature map and the feature map obtained after dimensionality reduction using PCA are generally comparable. However, classification accuracy improves significantly with the use of the reduced feature map from LDA. Moreover, the classification results of SVM and ANN are also comparable, yet SVM performing marginally better than ANNs due to the big data constraint of ANNs.

However, we must conclude that the classification results are not sufficiently good enough for deployment in commercial purposes or as a diagnosis tool in a medical environment, and therefore, requires further work.

References

[1] A. Subasi, "Classification of emg signals using pso optimized svm for diagnosis of neuromuscular disorders," *Computers in biology and medicine*, vol. 43, no. 5, pp. 576–586, 2013.

[2] M. R. Ahsan, M. I. Ibrahimy, O. O. Khalifa, *et al.*, "Emg signal classification for human computer interaction: a review," *European Journal of Scientific Research*, vol. 33, no. 3, pp. 480–501, 2009.

[3] K. Englehart and B. Hudgins, "A robust, real-time control scheme for multifunction myoelectric control," *IEEE transactions on biomedical engineering*, vol. 50, no. 7, pp. 848–854, 2003.

[4] N. F. Güler and S. Koçer, "Use of support vector machines and neural network in diagnosis of neuromuscular disorders," *Journal of medical systems*, vol. 29, no. 3, pp. 271–284, 2005.

[5] M.-F. Lucas, A. Gaufriau, S. Pascual, C. Doncarli, and D. Farina, "Multi-channel surface emg classification using support vector machines and signal-based wavelet optimization," *Biomedical Signal Processing and Control*, vol. 3, no. 2, pp. 169–174, 2008.

[6] A. Al-Timemy, G. Bugmann, N. Outram, J. Escudero, and H. Li, "Finger movements classification for the dexterous control of upper limb prosthesis using emg signals," in *Conference Towards Autonomous Robotic Systems*. Springer, 2012, pp. 434–435.

[7] R. N. Khushaba, S. Kodagoda, M. Takruri, and G. Dissanayake, "Toward improved control of prosthetic fingers using surface electromyogram (emg) signals," *Expert Systems with Applications*, vol. 39, no. 12, pp. 10 731–10 738, 2012.

[8] A. Phinyomark, P. Phukpattaranont, and C. Limsakul, "Feature reduction and selection for emg signal classification," *Expert systems with applications*, vol. 39, no. 8, pp. 7420–7431, 2012.

[9] B. Hudgins, P. Parker, and R. N. Scott, "A new strategy for multifunction myoelectric control," *IEEE transactions on biomedical engineering*, vol. 40, no. 1, pp. 82–94, 1993.

[10] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[11] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[12] K. Pearson, "Liii. on lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, vol. 2, no. 11, pp. 559–572, 1901.

[13] E. Hossain, "Principle component analysis: Dimension reduction," Oct 2020. [Online]. Available: https://medium.com/analytics-vidhya/principle-component-analysis-dimension-reduction-dea987f9d38

[14] Y. Xiaozhou, "Linear discriminant analysis, explained," Jan 2022. [Online]. Available: https://towardsdatascience.com/linear-discriminant-analysis-explained-f88be6c1e00b

[15] R. Gandhi, "Support vector machine - introduction to machine learning algorithms," Jul 2018. [Online]. Available: https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms\-934a444fca47

[16] A. Dertat, "Applied deep learning - part 1: Artificial neural networks," Oct 2017. [Online]. Available: https://towardsdatascience.com/applied-deep-learning-part-1-artificial-neural-networks-d7834f67a4f6

VI. Confusion Matrices

We present the confusion matrices for the experiments in this section as an appendix the report.

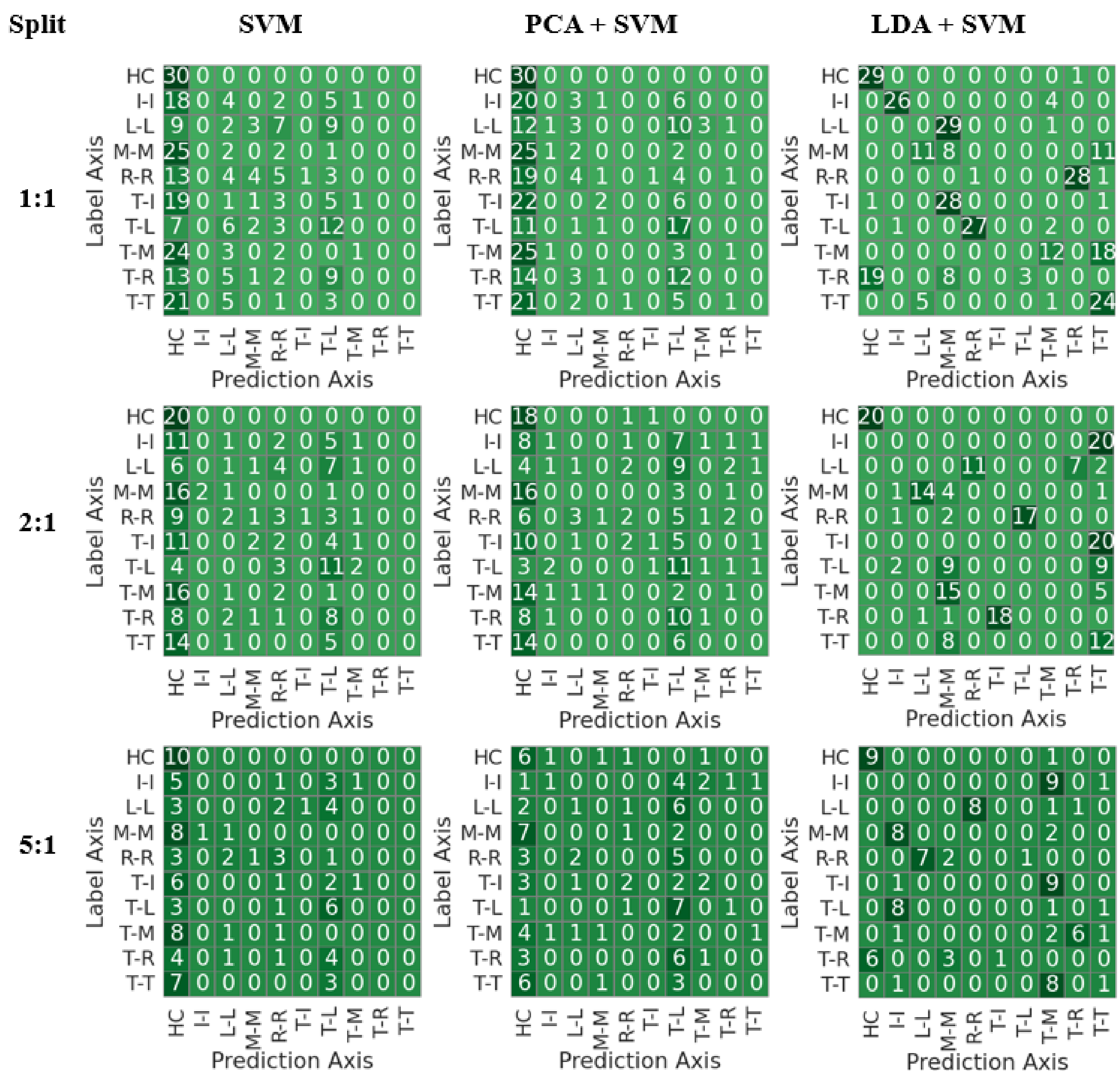

Fig. 8: Confusion Matrices for SVM Trainings.

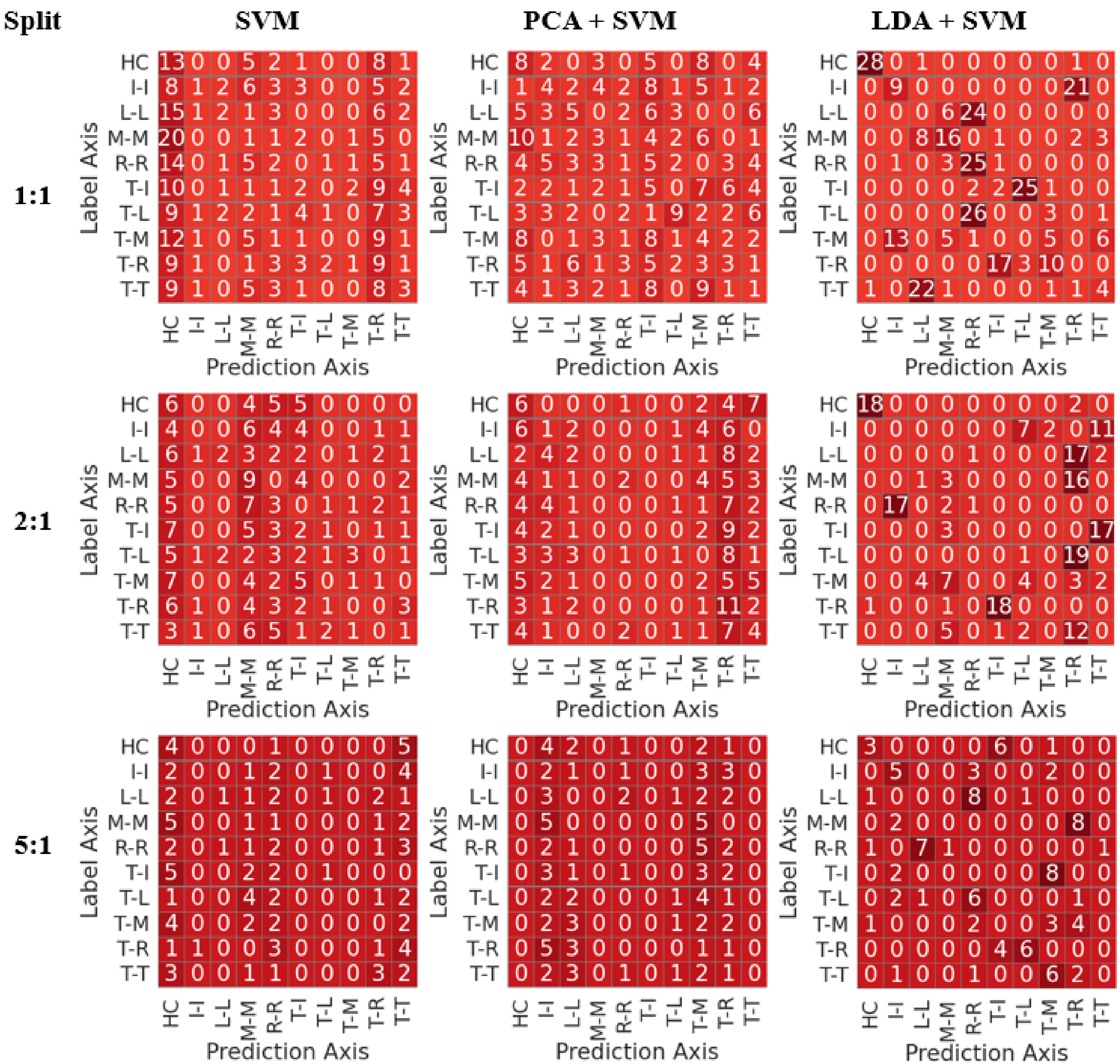

Fig. 9: Confusion Matrices for ANN Trainings.