Multi-layer perceptron training algorithms for pattern recognition of myoelectric signals

Le M.D. Khong, Timothy J. Gale, Danchi Jiang, Jan C. Olivier
School of Engineering
University of Tasmania
Hobart, Australia
Le.Khong@utas.edu.au, T.Gale@utas.edu.au

Max Ortiz-Catalan
Department of Signals and Systems,
Biomedical Engineering Division,
Chalmers University of Technology
Gothenburg, Sweden
maxo@chalmers.se

***Abstract*—A challenge in using myoelectric signals in control of motorised prostheses is achieving effective signal pattern recognition and robust classification of intended motions. In this paper, the performance of Matlab's Multi-layer Perceptron (MLP) backpropogation training algorithms in motion classification were assessed. The test and evaluation platform used was "BioPatRec", a Matlab-based open-source prosthetic control development environment, together with algorithms sourced from Matlab's neural network toolbox. The algorithms were used to interpret multielectrode myoelectric signals for motion classification, with the aim of finding the best performing algorithm and network model. The results showed that Matlab's trainlm and trainrp algorithms could achieve a higher accuracy than other tested MLP training algorithms (94.13 ± 0.037% and 91.09 ± 0.047%, respectively). Discussion of these results investigates significant features to obtain the highest performance.**

***Index Terms*—prosthetic control; pattern recognition; myoelectric signals; neural network.**

I. INTRODUCTION

Neural control of prosthetic limbs may be achieved through applying pattern recognition and motion categorisation algorithms to signals acquired from sEMG or implanted electrodes. Typically four or more bipolar electrodes [1] are sufficient for intended motions to be resolved and categorised [2, 3] and then used as input to a prosthetic control system. However, challenges remain, including improving pattern recognition and accuracy of motion classification [4, 5]. Here we investigate pattern recognition of surface electromyography (sEMG) signals as assessed by the accuracy of motion categorisation. We utilise "BioPatRec" [6] [7], an open-source Matlab-based research platform for neural-control of prosthetics that has modules for communications, signal recording, signal treatment, signal features, pattern recognition and control. In BioPatRec, movement classification can be assessed offline (using pre-recorded data) or real-time (where the accuracy of the classifier is evaluated while the subject is requested to execute different motions at random), with performance defined in terms of success in making a correct prediction of category of motion and maintaining this during the full range of a motion [1].

BioPatRec pattern recognition algorithms include Linear Discriminant Analysis (LDA), Multi-layer Perceptron (MLP) and Regulatory Feedback Network (RFN). Reported average offline accuracies for LDA, MLP and RFN are up to 92.1±0.04%, 91.2 ±0.05%, and 83.5 ±0.09% with corresponding real-time accuracies of 67.1 ±10%, 60.9 ±8.8%, and 67.4 ±10%, respectively [6]. Although offline accuracy is relatively high, there is potential for substantial improvement in the relatively low real-time accuracy, particularly with MLP.

The standard BioPatRec MLP implementation is a feedforward artificial neural network that uses backpropagation as a supervised training technique and gradient descent as the training algorithm and is implemented in stand-alone Matlab code. This paper explores potential improvements in BioPatRec's off-line MLP classification performance with alternative training techniques provided in Matlab's neural network toolbox. This is a pre-curser to the planned real-time assessment of the most promising techniques.

II. METHOD

The BioPatRec platform was used in this study with MLP training algorithms implemented in Matlab's neural network toolbox. To maintain consistency of implementation, all algorithms (including the gradient descent algorithm) used in this paper were the Matlab neural network toolbox versions. i.e. The BioPatRec implementation of the gradient descent algorithm was not used. The BioPatRec platform was modified to access toolbox MLP training algorithms, then investigations were conducted into the classification performance of MLP training algorithms and techniques together with the influence of network architecture. Training techniques were compared using the same data sets and data parameters.

The process of offline pattern recognition in BioPatRec is summarized in Fig. 1. The main parts of the process are Data Treatment and Motion Classification. The following sections detail aspects of this process.

978-1-4799-1467-8/13/$31.00

A. Data sources

Pre-recorded data was used in the classification, as described by Ortiz-Catalan *et al*. [6]. The data included recordings of 10 movements recorded by 4 pairs (4 channels) of bipolar electrodes placed around the forearm, collected from 20 subjects of both genders, non-amputee. The first pair of electrodes was placed at the extensor carpi ulnaris, the other pairs were placed equally spaced around the forearm. Subjects had an average age of 30.1 ± 10.5, average height of 1.77 ± 0.08 m, average weight of 68.38 ± 11 kg. This data is available online as part of BioPatRec data repository [7].

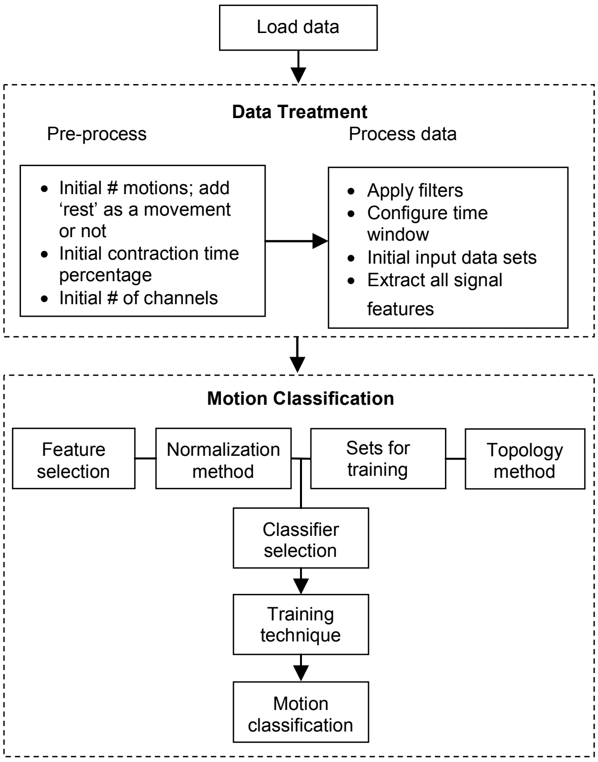


Fig. 1. BioPatRec offline pattern recognition flowchart.

B. Network architecture

The network architecture comprised an input vector, one or two layers of hidden neurons (with the number of neurons matching the number of inputs in each layer) and a layer of output neurons corresponding to the number of classes. The activation function used was either the logistic sigmoid function (*logsig*) or the hyperbolic tangent sigmoid transfer function (*tansig*), with a 0 to 1 output range.

C. MLP training algorithms

Twelve MLP training techniques were used. These were Levenberg-Marquardt backpropagation (trainlm), BFGS quasi-Newton backpropagation (trainbfg), conjugate gradient backpropagation with Powell-Beale restarts (traincgb), conjugate gradient backpropagation with Fletcher-Reeves update (traincgf), conjugate gradient backpropagation with Polak-Ribiere updates (traincgp), gradient descent backpropagation (traingd), gradient descent with adaptive learning rate backpropagation (traingda), gradient descent with momentum (traingdm), gradient descent with momentum & adaptive learning rate backpropagation (traingdx), one step secant backpropagation (trainoss), resilient backpropagation (trainrp) and scaled conjugate gradient backpropagation (trainscg).

The data was separated into training, validation and test sets. The resulting classification performance using two different data set ratios were compared. The proportion of data in the training, validation and test sets was either 40:20:40 or 70:15:15 (training:validation:test).

D. Data treatment

The data was treated to enable the pattern recognition algorithms to be fed with concurrent data with signal features extracted from fixed time windows [6]. A set of 4 signal features in the time domain (mean absolute value, waveform length, zero crossing and slope sign change) was extracted which were known to provide useful information of the intact arm [1, 3, 6]. Inputs comprised a feature vector, a product of signal features and the number of channels. Each feature vector had a corresponding output movement vector where movements were marked by "1" on the corresponding column and "0" on all others [6]. These were the inputs and target outputs for the movement classifiers.

Data was selected and treated through the BioPatRec GUI. Offline pre-processing then comprised selecting movements, channels, frequency and spatial filters (none applied), addition of "rest" as a movement, selection of the contraction time percentage (cTp=0.7) and extracting the data. The data were cut in time windows (200 ms, time increment 50 ms [8, 9]) and divided into training set, validation set, testing set; then signal features were extracted.

E. Motion classification

A total of 11 labels were classified, which included 10 hand and wrist motions, plus the resting position. The movements were hand open/close, wrist flex/extend, pronation/supination, fine/side grip, pointer and thumb up. The MLP parameters were the 4 signal features mentioned in *Data treatment* and min-max normalization (0 to 1) (a range of (-1, +1) was used in [6]). The statistical test was performed with 20 subjects and results were reported for offline accuracy and standard deviation of training functions. Wilcoxon Sign-Rank test was used to report the behaviour of the alternative data set ratio and network architecture for each training function [9].

III. Results

A. Performance of training algorithms

The "traimlm" and "trainrp" training algorithms were observed to be most accurate. For both the 40:20:40 (Fig. 2(a)) and 70:15:15 (Fig. 2(b)) ratios between training, validation and testing sets, accuracy was over 90% ($p < 0.05$).

With both 1 and 2 hidden layer architecture, the "trainlm" and "trainrp" also both obtained an accuracy of over 90%

while the accuracy of the other techniques lay between 50% and 80%, Figs. 3(a) and 3(b) ($p < 0.05$).

With both the *tansig* and *logsig* activation functions, "trainlm" and "trainrp" consistently achieved the highest accuracy (over 90%) (Figs. 4(a) and 4(b)), obtaining accuracies of approximately 95% and 91% respectively. Methods "traingdx" and "trainscg" achieved the second highest accuracies (approximately 80%).

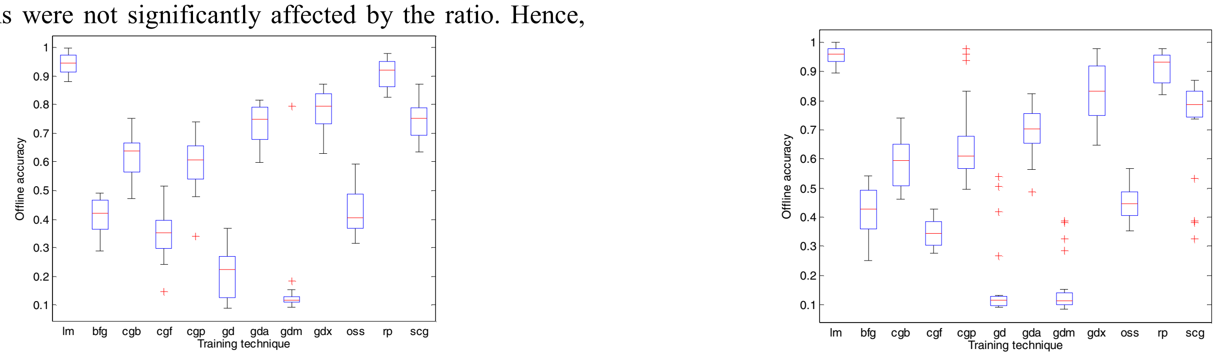
B. Comparison between parameters

The effect of input data sets, network complexity and activation function with respect to individual training technique was explored. The statistical test reported is the p-value of Wilcoxon Sign-Rank test of each training function over their range of accuracies.

Results for the two training set ratios (40:20:40 and 70:15:15) generally showed that accuracy was higher when the proportion of training data was larger. Both "trainlm" and "trainrp" were observed to be successful but with slightly different accuracies between the two set ratios, 94.13 ±0.037% vs. 95.61 ±0.028% for "trainlm" and 91.09 ±0.047% vs. 91.69 ±0.049% for "trainrp" for 40:20:40 and 70:15:15, respectively (Table 1). The p-value showed that the other activation functions were not significantly affected by the ratio. Hence, there was no significant role of these ratios on the effect of offline accuracy.

The network complexity was tested. One hidden layer of neurons was compared with two hidden layers of neurons. Again, high accuracies were observed in "trainlm" and "trainrp". The p-value showed that only "trainrp" benefited the increase in the number of layers but the rise was not significant (from 91.69 ±0.049% to 92.31 ±0.045%). The accuracy was slightly increased with the increase of the number of hidden neurons, Table 2. There was an alternative impact on the other training functions with the vast majority of p-values greater than 0.05.

It was interesting to consider the effect of the activation function on the behaviour of the training function since many of the training functions were tools for function approximation problems such as "trainlm" and "trainbfg". The network performance was successful with "trainlm" and "trainrp". The test showed that "tansig" and "logsig" affected the accuracies of many training functions but the "trainlm", "traincgf", "trainrp" and "trainscg" did not ($p > 0.05$). The accuracies resulting from training by "tansig" and "logsig" were 95.61 ±0.028% vs. 95.21 ±0.031% for "trainlm" and 91.69 ±0.049% vs. 91.81 ±0.046% for "trainrp", Table 3.

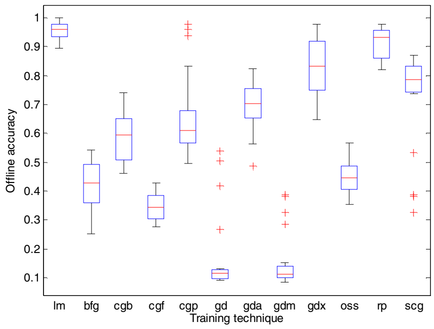


(a) (b)

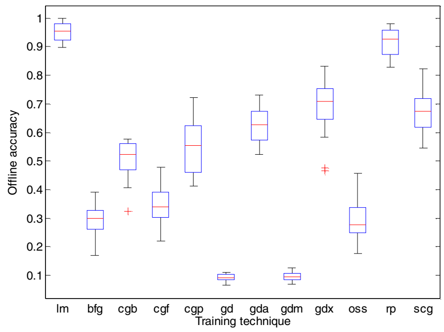
Fig. 2. Offline accuracy of the different MLP training algorithms for two sets of ratios between training, validation and testing sets: (a) 40:20:40 and (b) 70:15:15, with lm=trainlm, bfg=trainbfg, cgb=traincgb, cgf=traincgf, cgp=traincgp, gd=traingd,gda=traingda,gdm=traingdm, gdx=traingdx, oss=trainoss, rp=trainrp, scg=trainscg. The central mark of each box is the median, the edges of the box are the 25th (lower edge) and 75th percentile (upper edge), outliers are shown by "+" and the whiskers indicate the maximum and minimum data distribution (excluding outliers).

(a) (b)

Fig. 3. Offline accuracy of the different MLP training algorithms for (a) 1 and (b) 2 hidden layers of neurons, with lm=trainlm, bfg=trainbfg, cgb=traincgb, cgf=traincgf, cgp=traincgp, gd=traingd,gda=traingda,gdm=traingdm, gdx=traingdx, oss=trainoss, rp=trainrp, scg=trainscg. The central mark of each box is the median, the edges of the box are the 25th (lower edge) and 75th percentile (upper edge), outliers are shown by "+" and the whiskers indicate the maximum and minimum data distribution (excluding outliers).



(a)



(b)

Fig. 4. Offline accuracy of the different MLP training algorithms for two types of activation function: (a) *tansig* and (b) *logsig*, with lm=trainlm, bfg=trainbfg, cgb=traincgb, cgf=traincgf, cgp=traincgp, gd=traingd,gda=traingda,gdm=traingdm, gdx=traingdx, oss=trainoss, rp=trainrp, scg=trainscg. The central mark of each box is the median, the edges of the box are the 25th (lower edge) and 75th percentile (upper edge), outliers are shown by "+" and the whiskers indicate the maximum and minimum data distribution (excluding outliers).

TABLE I. COMPARISON OF INPUT RATIO OF 40:20:40 AND 70:15:15

	40:20:40		**70:15:15**		
	Acc	***Std***	***Acc***	***Std***	***Wilcoxon p-value***
trainlm	94.13%	0.037	95.61%	0.028	0.0004
trainbfg	40.95%	0.062	42.86%	0.079	0.4330
traincgb	61.56%	0.078	58.91%	0.081	0.5016
traincgf	34.38%	0.083	34.71%	0.046	p > 0.05
traincgp	58.89%	0.088	66.15%	0.143	0.2471
traingd	21.46%	0.081	17.42%	0.138	0.1790
traingda	73.18%	0.068	70.01%	0.082	0.0438
traingdm	15.36%	0.149	15.64%	0.097	0.6813
traingdx	78.25%	0.067	82.52%	0.101	0.1259
trainoss	42.58%	0.079	44.83%	0.059	0.4115
trainrp	91.09%	0.047	91.69%	0.049	0.0090
trainscg	74.82%	0.063	72.44%	0.167	0.5755

TABLE II. COMPARISON OF ONE HIDDEN LAYER AND TWO HIDDEN LAYERS

	1 hidden layer		**2 hidden layers**		
	Acc	***Std***	***Acc***	***Std***	***Wilcoxon p-value***
trainlm	95.61%	0.028	95.74%	0.028	0.5257
trainbfg	42.86%	0.079	54.60%	0.081	0.0010
traincgb	58.91%	0.081	61.31%	0.070	0.2322
traincgf	34.71%	0.046	32.32%	0.054	0.1790
traincgp	66.15%	0.143	64.11%	0.074	0.8228
traingd	17.42%	0.138	13.21%	0.021	0.6542
traingda	70.01%	0.082	68.13%	0.078	0.1672
traingdm	15.64%	0.097	12.64%	0.022	0.6012
traingdx	82.52%	0.101	75.74%	0.066	0.0124
trainoss	44.83%	0.059	58.63%	0.078	0.0001
trainrp	91.69%	0.049	92.31%	0.045	0.0304
trainscg	72.44%	0.167	80.84%	0.051	0.3135

TABLE III. COMPARISON OF ACTIVATION FUNCTION TANSIG AND LOGSIG

	tansig		**logsig**		
	Acc	***Std***	***Acc***	***Std***	***Wilcoxon p-value***
trainlm	95.61%	0.028	95.21%	0.031	0.3317
trainbfg	42.86%	0.079	29.22%	0.055	0.0002
traincgb	58.91%	0.081	50.64%	0.062	0.0051
traincgf	34.71%	0.046	34.48%	0.061	p > 0.05
traincgp	66.15%	0.143	55.54%	0.100	0.0187
traingd	17.42%	0.138	9.21%	0.011	0.0002
traingda	70.01%	0.082	62.49%	0.058	0.0017
traingdm	15.64%	0.097	9.64%	0.014	0.0004
traingdx	82.52%	0.101	69.26%	0.098	0.0002
trainoss	44.83%	0.059	29.65%	0.065	0.0001
trainrp	91.69%	0.049	91.81%	0.046	0.5775
trainscg	72.44%	0.167	68.16%	0.082	0.2471

IV. DISCUSSION

The "trainlm" and the "trainrp" methods consistently achieved an accuracy of over 90%. This reached similar accuracies reported from using the gradient decent technique as in BioPatrec without Matlab's Neural Network Toolbox algorithms. The other training algorithms in contrast were not so successful. The advantages of "trainlm" in solving data fitting problems was indicated during the test by the achievement of the highest accuracy between techniques.

It was noticed that the backpropagation with gradient descent algorithm was not as successful as it was in the implementation of BioPatRec [6]. This may have been because of the change of normalization method. Its accuracy was 91.2 ±0.05% for (-1, +1) normalisation whilst it was below 50% with (0, 1) normalisation. The gradient descent with adaptive learning algorithms ("traingda" and "traingdx") achieved higher accuracy than the pure technique, "traingd". This suggested consideration of normalisation over the range (-1, +1) for future work.

A limitation of the study is that it has not considered the behavior of these training techniques with respect to simultaneous movements. Only single isolated movements of hand and wrist were studied. Future work could explore this aspect together with the different topologies of classifiers.

V. Conclusion

BioPatRec is a free open source platform for prosthetic control which provides an environment for research groups to share and compare their various algorithms in a common framework. In this work, we report alternative MLP training algorithms that resulted in good classification accuracies. These will be further evaluated to verify if the benefits hold in the real-time classification of limb motions.

References

[1] G. Li, A. E. Schultz, and T. A. Kuiken, "Quantifying pattern recognition-based myoelectric control of multifunctional transradial prostheses," *IEEE transactions on neural systems and rehabilitation engineering,* vol. 18, pp. 185-192, April 2010.

[2] M. D. John B. Hijjawi, M. D. Todd A. Kuiken, M. D. Robert D. Lipschutz, L. A. Miller, K. A. Stubblefield, and M. D. Gregory A. Dumanian, "Improved Myoelectric Prosthesis Control Accomplished Using Multiple Nerve Transfers," *HAND/PERIPHERAL NERVE,* vol. 118, pp. 1573-1578, Decemder 2006.

[3] B. Hudgins, P. Parker, and S. Robert N, "A New Strategy for Multifunction Myoelectric Control," *IEEE transactions on bio-medical engineering,* vol. 40, pp. 82-94, January 1993.

[4] G. Li, Y. Li, L. Yu, and Y. Geng, "Conditioning and Sampling Issues of EMG Signals in Motion Recognition of Multifunctional Myoelectric Prostheses," *Annals of Biomedical Engineering,* vol. 39, pp. 1779 -1787, June 2011.

[5] M. D. Chul Ho Jang, M. D. Hee Seung Yang, M. D. Hea Eun Yang, and M. D. Seon Yeong Lee, "A Survey on Activities of Daily Living and Occupations of Upper Extremity Amputees," *Annals of Rehabilitation Medicine,* vol. 35(6), pp. 907-921, December 2011.

[6] M. Ortiz-Catalan, R. Brånemark, and B. Håkansson, "BioPatRec: A modular research platform for the control of artificial limbs based on pattern recognition algorithms," *Source Code for Biology and Medicine,* April 2013.

[7] M. Ortiz-Catalan. (2013). *BioPatRec*. Available: https://code.google.com/p/biopatrec/

[8] F. FD, P. JC, and F. CJ, "Evaluation of EMG processing techniques using information theory," *Biomed Eng Online 2010,* vol. 9, 2010.

[9] J. Demˇsar, "Statistical Comparisons of Classifiers over Multiple Data Sets," *Journal of Machine Learning Research,* pp. 1-30, 2006.