Logistic-weighted Regression Improves Decoding of Finger Flexion from Electrocorticographic Signals

Weixuan Chen*-*IEEE Student Member*, Xilin Liu-*IEEE Student Member*, and Brian Litt-*IEEE Senior Member*

Abstract— **One of the most interesting applications of brain computer interfaces (BCIs) is movement prediction. With the development of invasive recording techniques and decoding algorithms in the past ten years, many single neuron-based and electrocorticography (ECoG)-based studies have been able to decode trajectories of limb movements. As the output variables are continuous in these studies, a regression model is commonly used. However, the decoding of limb movements is not a pure regression problem, because the trajectories can be apparently classified into a motion state and a resting state, which result in a binary property overlooked by previous studies. In this paper, we propose an algorithm called logistic-weighted regression to make use of the property, and apply the algorithm to a BCI system decoding flexion of human fingers from ECoG signals. Our results show that the application of logistic-weighted regression improves decoding performance compared to the application of linear regression or pace regression. The proposed algorithm is also immensely valuable in the other BCIs decoding continuous movements.**

I. Introduction

Brain computer interfaces (BCIs) decode brain signals enabling people to control devices without muscular movement. Because BCI systems provide a direct communication pathway between the brain and an external device, they are of great help to people with severe paralysis. One of the most important applications of BCIs is to assist people who have disrupted neuromuscular channels through which the brain communicates with and controls its external environment by reproducing their motor functions with a cursor or a robotic arm [1]. Thanks to the progress of invasive recording techniques and decoding algorithms in the past ten years, many single neuron-based and electrocorticography (ECoG)-based studies have been able to decode continuous trajectories of limb movements. Unlike traditional BCIs reviewed in [2–4] which classify discrete brain states, these studies require prediction of continuous variables. In other words, they belong to regression problems rather than classification problems.

The simplest and most robust solution to the regression problem is to linearly model the relationship between brain signals and limb movements. This linear relationship can be established using linear regression or its variants, including pace regression [5], ridge regression [6], or time-embedded linear Wiener filter [7–10]. To take into account the physiological, physical, and mechanical constraints that affect the flexion of limbs, some studies applied switching models [11] or Bayesian models [12,13] to the results of linear regressions above. Other studies have explored the utility of non-linear methods, including neural networks [14–17], multilinear perceptrons [18], and support vector machines [18], but they tend to have difficulty with high dimensional features and limited training data [13].

Nevertheless, the studies of limb movement translation are in fact not pure regression problems, because the limbs are not always under the motion state. Whether it is during an experiment or in the daily life, the resting state of the limbs is usually as long as their motion state, if not longer. In this case, the recorded movement data will exhibit a binary property, which was not made the best of in the studies introduced above. Given the binary property, the limbs may obey different models under the two different states, and it is unreasonable to estimate the limb trajectories without distinguishing between the two states. A possible negative effect was shown in "an interesting observation" of [15], in which the estimated hand trajectory showed high correlation with the actual one when a primate performed a movement, while the correlation diminished when its arm was in the resting state.

In this paper, we propose a novel algorithm named logistic-weighted regression to synthesize the binary information and the continuous information of the movement data. First, with real data, we illustrate the significance of exploiting the above-mentioned binary property. Second, we prove the statistical principles of the proposed algorithm on the basis of the law of total expectation. Finally, we compare the results of the algorithm with those of two existing popular methods.

II. Methods

A. Data Collection

The BCI Competition IV dataset collected by Kubanek et al. [5] is employed for this study. Three patients with intractable epilepsy had electrode arrays on the surface of the brain for the purpose of localization of seizure foci prior to surgical resection. Each subject had a 48- or 64-electrode array placed over the frontal-parietal-temporal region including parts of sensorimotor cortex. During the experiment, the patients were asked to move specific individual fingers in response to visual cues. The subjects typically flexed the indicated finger 3-5 times during a 1.5-3 s time period, and then rest for 2 s. The ECoG signals from the electrodes were amplified, band-pass filtered between 0.15 and 200 Hz, digitized at 1 kHz and recorded in a general purpose BCI2000 system [19]. The flexion of each finger was

Weixuan Chen is with the Department of Bioengineering, University of Pennsylvania, Philadelphia, PA 19104 USA (phone: 616-666-6361; fax: 215-349-5733; e-mail: wchen@seas.upenn.edu).

Xilin Liu is with the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104 USA (e-mail: xilinliu@seas.upenn.edu).

Brian Litt is with the Department of Bioengineering and the Department of Neurology, University of Pennsylvania, Philadelphia, PA 19104 USA (e-mail: littb@mail.med.upenn.edu).

measured and digitized in 12 bit and 25 Hz by a data glove (5DT Data Glove 5 Ultra, Fifth Dimension Technologies, Inc.).

B. Feature Extraction

First, bad channels (channels with abnormal spikes) were manually omitted. As a result, Channel 55 of Subject 1 and Channel 21, 38 of Subject 2 were removed. Second, a common average reference (CAR) montage was performed on all channels [5]. Then the features below were extracted from every channel to compose the feature vector $\boldsymbol{\Phi}$:

- Average frequency-domain feature: As proposed by Sanchez et al., sensorimotor ECoG dynamics are shown in sub-bands (1-60Hz), gamma band (60-100Hz), fast gamma band (100-300Hz) and ensemble depolarization (300-6000Hz) [20]. We extracted the spectral amplitudes in particular frequency ranges: 5-15 Hz, 20-25 Hz, 75-115 Hz, 125-160 Hz, and 160-175 Hz. The 35-70 Hz frequency range was abandoned, because it had been demonstrated to reflect conflicting spectral phenomena [21].
- Local motor potential (LMP) feature: the mean value of the raw unrectified time-domain signal.
- Variance feature: the variance of the raw unrectified time-domain signal.

A moving window of 80 ms in length with 40 ms overlapping was used when extracting all the features above. Since the sampling rate of the data-glove is 25 Hz, the features have the same length of the data-glove's position measurements.

C. Logistic-weighted Regression

The main characteristic of the finger flexion data is that it's approximately binary, as is shown in Fig. 1. Since the features described above don't have a similar binary property, a traditional linear regression cannot fit the finger flexion data very well. As is shown in Fig. 2, the huge fluctuation of the target variable make the linear regression result full of oscillations.

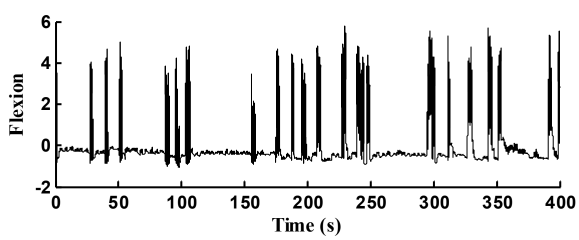

Figure 1. Real finger flexion time course of Subject 1's thumb.

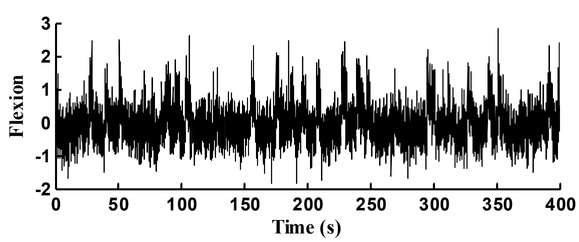

Figure 2. Finger flexion time course of Subject 1's thumb predicted by linear regression.

On the other hand, since the finger flexion data is approximately binary, a threshold can be manually set to easily classify it into two states (motion and non-motion, Fig. 3). Because the natural flexion or extension of fingers are limited to certain ranges due to the physical constraints of our hands [12], the threshold is very robust. After the classification the target variable becomes entirely binary, as a result of which a logistic regression can be used to estimate the probability that a time bin belongs to the motion state. The result of the logistic regression exhibits much more binary information than that of the linear regression (Fig. 4), but it loses all the information within each state.

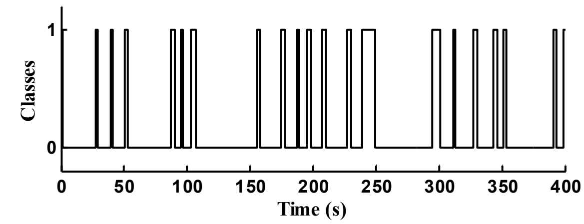

Figure 3. Classification result of Subject 1's thumb with a threshold of 1.4. Class 1 represents the motion state, while class 0 represents the non-motion state. Neighbouring labels of class 1 were combined within every 4 s to cancel noise.

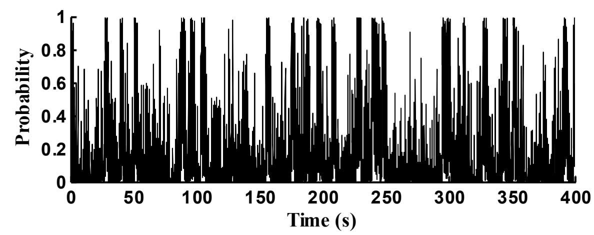

Figure 4. Logistic regression result of Subject 1's thumb using binary labels as the target variable.

Therefore, a better strategy is to combine the two methods. In the algorithm called logistic-weighted regression, we weight the linear regression results with the logistic regression results on the basis of the law of total expectation.

Based on the two states of a certain finger, a pair of mutual exclusive hypotheses for each time point can be established:

$$\begin{aligned} &H_0\text{: The finger is under the non-motion state} \\ &H_1\text{: The finger is under the motion state} \end{aligned} \tag{1}$$

If H_0 is true, the optimal estimate of the finger flexion y (the conditional expectation $E(y|H_0)$) will be 0; If H_1 is true, the optimal estimate of the finger flexion y (the conditional expectation $E(y|H_1)$) will be the linear regression result $\hat{y}$.

In logistic regression, the regression result $\hat{y}_l$ is the posterior probability of accepting H_1, which can be written as a logistic sigmoid acting on a linear function of the feature vector $\boldsymbol{\Phi}$ [22]:

$$\hat{y}_l = p(H_1|\boldsymbol{\Phi}) = \sigma(\boldsymbol{w}^T\boldsymbol{\Phi}) \tag{2}$$

where $\boldsymbol{w}$ is the optimal weight vector and $\sigma(\cdot)$ is the logistic sigmoid function. Since H_0 and H_1 are mutual exclusive, $p(H_0|\boldsymbol{\Phi}) = 1 - p(H_1|\boldsymbol{\Phi})$.

Combining the estimates under the two hypotheses and their corresponding probabilities, the result of the logistic-weighted regression $\hat{y}_{lw}$ is the total expectation of the finger flexion. According to the law of total expectation, it is

$$\begin{aligned}\hat{y}_{lw} &= E(y) = E(y|H_1)p(H_1|\boldsymbol{\Phi}) + E(y|H_0)p(H_0|\boldsymbol{\Phi}) \\ &= \hat{y}\hat{y}_l + 0 \times (1 - \hat{y}_l) = \hat{y}\hat{y}_l\end{aligned} \tag{3}$$

which is the linear regression result weighted by the logistic regression result.

In the linear regression, the target variable is the real finger flexion data, for which the measurement error of the data-glove can be neglected. However, in the logistic regression, the target variable is the binary labels $t \in \{0,1\}$ classified with manually set thresholds, as a result of which the classification error has to be taken into consideration. Within the process of the logistic regression, the error of the target variable t is propagated to the weight vector $\boldsymbol{w}$, and finally the regression result $\hat{y}_l$. In tradition, the regression result with error should be represented as $\hat{y}_l(1 \pm \delta)$. However, since $\hat{y}_l$ is a posterior probability satisfying the restriction $0 \le \hat{y}_l \le 1$ in the case of logistic regression, $\hat{y}_l(1 \pm \delta)$ is no more a proper form for its potential to exceed the restriction. Therefore, the error-including result is alternatively written as ${\hat{y}_l}^{1\pm\delta}$.

Assuming $\alpha = 1 \pm \delta$, the final expression of the logistic-weighted regression result will be

$$\hat{y}_{lw} = \hat{y}{\hat{y}_l}^{\alpha}, \alpha > 0 \tag{4}$$

where the error-correction coefficient α is optimized via cross-validation.

The predicted flexion of the same finger as the former figures given by the logistic-weighted regression is shown in Fig. 5. In this case, α is set to be 1 (default value) without optimization. It is obvious that even without error correction the logistic-weighted regression result has already been much better than the linear regression result, and very close to the real flexion data.

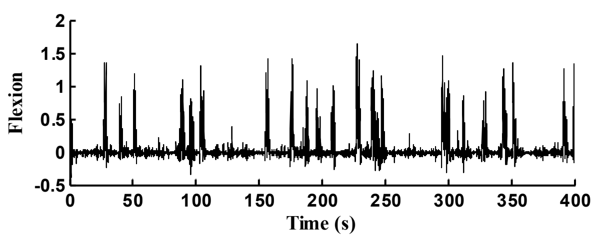

Figure 5. Finger flexion time course of Subject 1's thumb predicted by logistic-weighted regression.

D. Least Mean Square (LMS) Adaptive Filtering

In order for the regression results to further imitate the time-domain characteristics of the finger flexion, an FIR adaptive filter was trained on the training data with the LMS algorithm [23]. The resulting filters were used to filter the testing regression results. The adaptive filter lengths and the LMS step sizes were optimized via cross-validation.

III. RESULTS

The original dataset consists in a 10-min recording per subject. We set apart the first 33.3% of each recording (200,000 samples) as the testing set to evaluate the algorithms, and the remaining 66.7% (400,000 samples) functioned as the training set. The error-correction coefficient of the logistic regression and the parameters of the adaptive filtering were optimized via 5-fold cross-validation within the training set.

To compare the algorithm with the existing methods, we also implemented linear regression and pace regression (part of the Java-based Weka package [24]), with which logistic-weighted regression was respectively replaced in the full procedure. The performances of these methods were evaluated based on the correlation between the real finger flexions within the testing set and the corresponding flexions predicted by different methods. As is shown in Table I, the decoding performance of logistic-weighted regression is better than those of the other methods, whether it is for every single finger, every single subject or the overall average.

TABLE I. CORRELATION COEFFICIENTS BETWEEN REAL FINGER FLEXIONS AND FINGER FLEXIONS PREDICTED BY LINEAR REGRESSION (A), PACE REGRESSION (B) AND LOGISTIC-WEIGHTED REGRESSION (C).

Subjects	Thumb	Index	Middle	Ring	Little	Avg.
(A) Linear regression						
Sub. 1	0.5058	0.7091	0.2597	0.5077	0.3768	0.4718
Sub. 2	0.5696	0.3053	0.3307	0.4541	0.3481	0.4016
Sub. 3	0.6868	0.6582	0.5948	0.6005	0.6521	0.6385
Avg.	0.5874	0.5575	0.3951	0.5208	0.4590	0.5040
(B) Pace regression						
Sub. 1	0.5235	0.7231	0.2861	0.5194	0.3924	0.4889
Sub. 2	0.5565	0.3055	0.3413	0.4677	0.3610	0.4064
Sub. 3	0.6952	0.6782	0.6159	0.6109	0.6422	0.6485
Avg.	0.5917	0.5689	0.4144	0.5327	0.4652	0.5146
(C) Logistic-weighted regression						
Sub. 1	0.5968	0.7621	0.2958	0.4982	0.4901	0.5286
Sub. 2	0.7012	0.4185	0.4438	0.5054	0.5150	0.5168
Sub. 3	0.8086	0.7005	0.6341	0.6170	0.7760	0.7072
Avg.	0.7022	0.6270	0.4579	0.5402	0.5937	0.5842

In order to show the advantage of the algorithm in details, an excerpt of real finger flexions and predicted finger flexions within the testing set is shown in Fig. 6. It is obvious in the figure that, compared with the other two methods, logistic-weighted regression gives a much better estimate during the non-motion state of the finger without losing the sensitivity of detecting movements. This is because logistic regression generates a small probability of movement when a finger is resting in reality, and by weighting the linear regression result with the probability, the oscillations appearing in the non-motion phases of Fig. 6(a) and Fig. 6(b) are greatly suppressed.

IV. DISCUSSION & CONCLUSION

This article proposed a new method of decoding ECoG signals for the prediction of finger flexion in human beings. The decoder, based on logistic-weighted regression, has been evaluated in the BCI Competition IV dataset, and showed advantages over two typical methods both qualitatively and quantitatively. In particular, the combination of linear regression and logistic regression based on the law of total expectation enabled a much better estimate of the finger flexion during its non-motion state.

Besides, another advantage of the proposed method is its simple algorithm structure and low computational burden compared with models assembling prior knowledge and non-linear methods, which make it more tractable for real-time applications. Last but not least, the logistic-weighted regression has the potential to be cooperative with and improved by a lot of existing methods. As was introduced in the Introduction section, modifications like [11–13] are directly based on the results of a linear regression. Since the logistic-weighted regression is able to replace a linear regression in any decoder, it is likely to further improve the performance of these modifications.

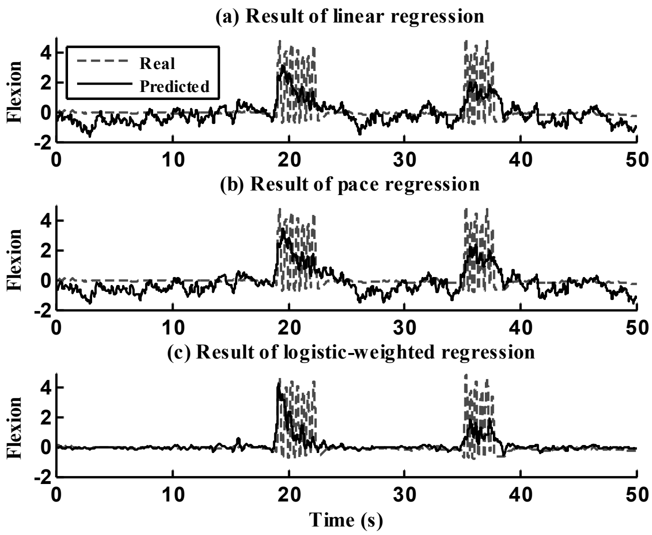

Figure 6. Real finger flexions and finger flexions predicted by linear regression (a), pace regression (b) and logistic-weighted regression (c). The excerpt is from the testing set of Subject 1's thumb between 120s and 170s.

There are some directions in which this work could be further improved. On one hand, some physiological, physical, and mechanical constraints can be introduced to make the algorithm more grounded in reality. For example, in the experiments of Kubanek et al. [5], subjects were dictated to move only one finger at a time, in which case the sum of the motion probabilities of the five fingers predicted by logistic regressions should be restricted to not exceed one. On the other hand, it is worthwhile to explore the potential for combining logistic regression with the variants of linear regression, such as pace regression, ridge regression, and time-embedded linear Wiener filter.

ACKNOWLEDGMENT

The authors would like to thank Gerwin Schalk who provided the dataset.

REFERENCES

[1] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, “Brain–computer interfaces for communication and control,” Clin. Neurophysiol., vol. 113, no. 6, pp. 767–791, Jun. 2002.

[2] F. Lotte, M. Congedo, a Lécuyer, F. Lamarche, and B. Arnaldi, “A review of classification algorithms for EEG-based brain-computer interfaces.,” J. Neural Eng., vol. 4, no. 2, pp. R1–R13, Jun. 2007.

[3] A. Bashashati, M. Fatourechi, R. K. Ward, and G. E. Birch, “A survey of signal processing algorithms in brain-computer interfaces based on electrical brain signals.,” J. Neural Eng., vol. 4, no. 2, pp. R32–57, Jun. 2007.

[4] K. R. Müller, C. W. Anderson, G. E. Birch, and G. E. Muller, KR and Anderson, Charles W and Birch, “Linear and nonlinear methods for brain-computer interfaces,” Neural Syst. Rehabil. Eng. IEEE Trans., vol. 11, no. 2, pp. 165–169, 2003.

[5] J. Kubánek, K. J. Miller, J. G. Ojemann, J. R. Wolpaw, and G. Schalk, “Decoding flexion of individual fingers using electrocorticographic signals in humans,” J. Neural Eng., vol. 6, no. 6, p. 66001, Dec. 2009.

[6] G. H. Mulliken, S. Musallam, and R. a Andersen, “Decoding trajectories from posterior parietal cortex ensembles.,” J. Neurosci., vol. 28, no. 48, pp. 12913–26, Nov. 2008.

[7] N. Liang, L. Bougrain, and others, “Decoding Finger Flexion using amplitude modulation from band-specific ECoG,” in European Symposium on Artificial Neural Networks-ESANN 2009, 2009, pp. 467–472.

[8] T. J. Bradberry, R. J. Gentili, and J. L. Contreras-Vidal, “Reconstructing three-dimensional hand movements from noninvasive electroencephalographic signals.,” J. Neurosci., vol. 30, no. 9, pp. 3432–7, Mar. 2010.

[9] T. J. Bradberry, R. J. Gentili, and J. L. Contreras-Vidal, “Fast attainment of computer cursor control with noninvasively acquired brain signals.,” J. Neural Eng., vol. 8, no. 3, p. 036010, Jun. 2011.

[10] A. Presacco, R. Goodman, L. Forrester, and J. L. Contreras-Vidal, “Neural decoding of treadmill walking from noninvasive electroencephalographic signals.,” J. Neurophysiol., vol. 106, no. 4, pp. 1875–87, Oct. 2011.

[11] R. Flamary and A. Rakotomamonjy, “Decoding finger movements from ECoG signals using switching linear models,” Front. Neurosci., vol. 6, p. 29, Mar. 2012.

[12] Z. Wang, Q. Ji, K. J. Miller, and G. Schalk, “Prior knowledge improves decoding of finger flexion from electrocorticographic signals.,” Front. Neurosci., vol. 5, p. 127, Nov. 2011.

[13] Z. Wang, Q. Ji, K. J. Miller, and G. Schalk, “Decoding finger flexion from electrocorticographic signals using a sparse Gaussian process,” in Proceedings of the 2010 20th International Conference on Pattern Recognition, 2010, pp. 3756–3759.

[14] J. K. Chapin, K. A. Moxon, R. S. Markowitz, and M. A. Nicolelis, “Real-time control of a robot arm using simultaneously recorded neurons in the motor cortex.,” Nat. Neurosci., vol. 2, no. 7, pp. 664–70, Jul. 1999.

[15] J. C. Sanchez, D. Erdogmus, J. C. Principe, J. Wessberg, and M. Nicolelis, “A comparison between nonlinear mappings and linear state estimation to model the relation from motor cortical neuronal firing to hand movements,” in Proceedings of SAB Workshop on Motor Control in Humans and Robots: On the Interplay of Real Brains and Artificial Devices, 2002, pp. 59–65.

[16] J. C. Sanchez, D. Erdogmus, Y. Rao, J. C. Principe, M. Nicolelis, and J. Wessberg, “Learning the contributions of the motor, premotor, and posterior parietal cortices for hand trajectory reconstruction in a brain machine interface,” in First International IEEE EMBS Conference on Neural Engineering, 2003. Conference Proceedings., 2003, pp. 59–62.

[17] S.-P. Kim, Y. N. Rao, D. Erdogmus, J. C. Sanchez, M. a. L. Nicolelis, and J. C. Principe, “Determining Patterns in Neural Activity for Reaching Movements Using Nonnegative Matrix Factorization,” EURASIP J. Adv. Signal Process., vol. 2005, no. 19, pp. 3113–3121, 2005.

[18] K. H. Kim, S. S. Kim, and S. J. Kim, “Superiority of nonlinear mapping in decoding multiple single-unit neuronal spike trains: a simulation study.,” J. Neurosci. Methods, vol. 150, no. 2, pp. 202–11, Jan. 2006.

[19] G. Schalk, D. J. McFarland, T. Hinterberger, N. Birbaumer, and J. R. Wolpaw, “BCI2000: a general-purpose brain-computer interface (BCI) system.,” IEEE Trans. Biomed. Eng., vol. 51, no. 6, pp. 1034–43, Jun. 2004.

[20] J. J. C. Sanchez, A. Gunduz, P. R. Carney, and J. C. Principe, “Extraction and localization of mesoscopic motor control signals for human ECoG neuroprosthetics.,” J. Neurosci. Methods, vol. 167, no. 1, pp. 63–81, Jan. 2008.

[21] K. J. Miller, P. Shenoy, M. Den Nijs, L. B. Sorensen, R. P. N. Rao, and J. G. Ojemann, “Beyond the gamma band: the role of high-frequency features in movement classification,” Biomed. Eng. IEEE Trans., vol. 55, no. 5, pp. 1634–1637, 2008.

[22] C. M. Bishop and N. M. Nasrabadi, Pattern recognition and machine learning, vol. 1. Springer, 2006.

[23] J. J. Shynk, “Frequency-domain and multirate adaptive filtering,” IEEE Signal Process. Mag., vol. 9, no. 1, pp. 14–37, 1992.

[24] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, “The WEKA data mining software: an update,” ACM SIGKDD Explor. Newsl., vol. 11, no. 1, p. 10, Nov. 2009.