Automated machine learning for the classification of normal and abnormal electromyography data

Kefalas, M.; Koch, M.; Geraedts, V.J.; Wang, H.; Tannemaat, M.; Bäck, T.H.W.

Automated Machine Learning for the Classification of Normal and Abnormal Electromyography Data

Marios Kefalas
Leiden University
Leiden, The Netherlands
m.kefalas@liacs.leidenuniv.nl

Milan Koch
Leiden University
Leiden, The Netherlands
m.koch@liacs.leidenuniv.nl

Victor Geraedts
Leiden University
Leiden, The Netherlands
v.j.geraedts@lumc.nl

Hao Wang
Sorbonne Université
Paris, France
hao.wang@lip6.fr

Martijn Tannemaat
Leiden University
Leiden, The Netherlands
m.r.tannemaat@lumc.nl

Thomas Bäck
Leiden University
Leiden, The Netherlands
t.h.w.baeck@liacs.leidenuniv.nl

2020 IEEE International Conference on Big Data (Big Data) | 978-1-7281-6251-5/20/$31.00 ©2020 IEEE | DOI: 10.1109/BigData50022.2020.9377780

***Abstract*—Needle electromyography (EMG) is a common technique used in clinical neurophysiology to record the electrical activity of muscles at different levels of activation. It can be used to diagnose various neurological/muscular disorders, as the EMG signals of patients with both nerve diseases (neuropathies) and muscle diseases (myopathies) differ from the signal in healthy controls. A major drawback of this examination is that it relies on visual inspection and as such, it is highly subjective and prone to errors. Based on EMG time series of 65 individuals (40 with ALS/IBM and 25 healthy), we aim to develop an automated machine-learning pipeline for the classification of EMG recordings of muscles in either disease or healthy (muscle-level). The automated pipeline consists of feature extraction, feature selection, modelling algorithm, and optimization, in which the most significant features are automatically selected from the feature space and the hyperparameters of the model are optimized by a Bayesian technique as part of the automated approach. Aside from the muscle-level approach, we also explore a patient-level approach, which uses the output of the muscle-level automated pipeline in a post-processing manner to classify patients in being either disease or healthy, based on their muscle recordings. The resulting two approaches yield an AUC score of 81.7% (muscle-level) and 81.5% (patient-level), indicating that such approaches can assist clinicians in diagnosing if a patient has a neuropathy/myopathy or is healthy.**

***Index Terms*—Automated Machine Learning, EMG, ALS, IBM, Neuromuscular, Time Series Classification**

I. Introduction

Needle or intramuscular electromyography (EMG) is a common technique used in clinical neurophysiology to record the electrical activity of muscles at different levels of activation [1]. As the EMG signals of patients with both nerve diseases (neuropathies) and muscle diseases (myopathies) differ from the signal in healthy controls, EMG can be used to diagnose various neurological disorders. The most commonly used method to interpret the EMG is qualitative, based on visual inspection of the signal in real time by an experienced examiner. A major drawback of this method is that it is highly subjective and prone to errors. In particular for the diagnosis of myopathies, EMG has been called one of the most difficult areas in electrodiagnostic medicine [1]. In theory, a neuropathic EMG, with fibrillation potentials, positive sharp waves, high-amplitude and long duration motor unit potentials (MUPs) and a reduced interference pattern should be clearly distinguishable from a myopathic EMG containing smaller, short-duration polyphasic MUPs and a full interference pattern. In practice, however, the diagnostic yield of qualitative EMG analysis, for the distinction between both abnormal/myopathic and between neuropathic/myopathic is disappointingly low. In the past decades, several quantitative EMG (qEMG) methods such as turns-amplitude analysis have been developed in an attempt to increase the diagnostic yield of the EMG, but so far sensitivity and specificity of various qEMG techniques has remained similar to visual inspection [2], [3]. Similarly, another quantitative technique called the clustering index method yielded a sensitivity of 92% for neurogenic and 61% for myopathic patients [4]. Interpretation of the EMG in patients with Inclusion Body Myositis (IBM) (a myopathy) is particularly challenging, as it may contain both myopathic and neurogenic features [5]. As IBM may also mimic motor neuron disease clinically, inappropriate interpretation of the EMG can lead to an incorrect diagnosis. A retrospective study of mislabeled IBM patients found that routine EMG commonly pointed to a neurogenic disorder: it showed fibrillations and positive sharp waves, as well as excessive amounts of polyphasic long-duration neurogenic MUPs in the majority of mislabeled patients [6]. This is highly unfortunate as Amyotrophic Lateral Sclerosis (ALS), a neuropathy, is a progressive fatal disease, whereas life expectancy is not significantly affected in IBM [7]. Most qEMG methods have been published several decades ago and are based on assumptions with regards to MUP morphology and physiology. Recent advances in computer processing power and machine learning techniques enable a big data approach that processes a large number of features without any underlying assumptions about the nature of the signal. We have previously shown that such an approach, developed for the automotive industry but applied to electroencephalography (EEG) signals, could

classify Parkinson Disease patients with good cognition from those with poor cognition with an accuracy of 91% [8].

A first approach towards automatic classification of specific diseases, either myopathic or neuropathic, is the differentiation between a normal EMG assessment from a healthy individual, and an abnormal EMG assessment from a patient with a myopathic or neuropathic disease. Here, we aimed to evaluate an automated time series classification algorithm for usage in differentiating EMG time series from healthy individuals and EMG time series from patients with either neuropathic or myopathic diseases. Our approach is automated and limits as much as possible arbitrary choices, providing at the same time valuable diagnostic information without having to rely heavily on clinical expertise.

The rest of the paper is organized as follows. In section II we give an overview of the related work in the field and limitations thereof. In section III we present our dataset used in this study and in section IV we show the pre-processing to transform the data. In section V we give a detailed explanation of the automated machine-learning muscle-level pipeline and in section VI we define and present our patient-level algorithm. In section VII we discuss the performance evaluation metrics. Finally, in section VIII we show the experimental results of our methods and we conclude in section IX.

II. Related Work

Electromyography (EMG) is the study of the electric activity of the muscle, and assists in the diagnosis of neuromuscular disorders. EMGs are used to detect and describe different disease processes affecting the motor unit (MU), the smallest functional unit of the muscle. During an EMG the motor unit action potentials (MUPs) are recorded using a needle electrode at slight voluntary contraction. The MUP reflects the electrical activity of a single anatomical motor unit. It represents the compound action potential of those muscle fibers within the recording range of the electrode. EMGs can detect neuromuscular disorders due to the structural reorganization of the MU, because of disorders affecting peripheral nerve and muscle [9]. Current clinical practice is based on expert visual inspection of MUP traces and simultaneous assessment of their audio characteristics in real time. This subjective assessment, even if satisfactory, may not be sufficient to describe less apparent deviations or mixed patterns of abnormalities [10]. Therefore, for an automated EMG signal classification to be effective, a systematic and thorough treatment of EMG signals must be carried out. Because of this, a number of computer-based quantitative EMG analysis algorithms have been developed [11].

In this view, [12] developed an EMG-based classifier for neuromuscular disorders using a Multi-Layer Perceptron (MLP). The authors compared the performance of five different feature extraction techniques from the EMG signals (autoregressive, root mean square, mean absolute value, zero crossing and waveform length) across five different classification tasks: healthy/unhealthy, healthy/myopathy, healthy/neuropathy, myopathy/neuropathy, healthy/myopathy/neuropathy. Their results showed that the autoregressive feature extraction from the EMG signal returned the best results in four out of five groups and they achieved the highest accuracy (86.3%) when classifying healthy/myopathy/neuropathy. In [13], a dataset of 50 healthy, 50 neurogenic, and 50 myopathic subjects is generated using an EMG simulation software, while the feature set consists of 8 features regarding signal amplitude and phase alongside with statistical metrics, such as mean and variance. The classification utilizes four different algorithms with a 97.78% classification accuracy using Support Vector Machines (SVM). In [14] the authors use an openly available clinical database consisting of recordings of ten healthy subjects, seven myopathic and eight patients with ALS. They use five feature extraction techniques (waveform length, zero crossings, slope sign changes, Willison amplitude, and root mean square). The study reports a 100% accuracy rate for normal subjects, 94% for myopathies and 96% for patients with ALS using the Linear Discriminant Analysis (LDA) classifier. In [15] the authors introduce a novel method for an automatic classification of subjects with or without neuromuscular disorders. This method is based on multiscale entropy of recorded surface electromyograms (sEMG) and Support Vector Classification. They achieved a diagnostic yield of 81.5% for healthy/patient classification and 70.4% for healthy/myopathy/neuropathy classification. In [16] the authors describe a method for the classification of neuromuscular disorders. The approach involves isolating single motor unit action potentials (MUPs), computing their scalograms, taking the maximum values of the scalograms in five selected scales, and averaging across MUPs to give a single 5-dimensional feature vector per subject. The SVM analysis reduces the vector to a single decision parameter, called the Wavelet Index, allowing the subject to be assigned to one of three groups: myogenic, neurogenic or normal. In [14] Naik et. al present an ensemble empirical mode decomposition algorithm that decomposes a single-channel EMG into a set of noise-canceled intrinsic mode functions, which are then linearly separated by the FastICA algorithm. Five time-domain features extracted from the separated components are then classified using the LDA, and the classification results are fine-tuned with a majority voting scheme. The authors achieved a diagnostic yield of 98% on a clinical EMG database, to discriminate between the normal, myopathic, and ALS subjects. More recently, Subasi et. al [17] present a bagging ensemble classifier for the automated classification of EMG signals. They use statistical values of the discrete wavelet transform coefficients and use those as features in a bagging ensemble of SVM, achieving a 99% accuracy for the diagnosis of neuromuscular disorders.

The work presented above is by no means exhaustive. To the best of our knowledge though, there has not been much research in hyperparameter tuning in the selected algorithms in this context. The use of hyperparameter optimization techniques would, for example, enhance the model performance further [18]. What is more, it is evident that most of the studies only consider a limited number of features as input

to the classifiers (i.e., Hudgin's set of features [19]). An automatic approach to find relevant time series representations would create and give insights to new features, or rather biomarkers [8], and would assist in avoiding time-consuming feature engineering processes. In addition most studies have been done on a specific muscle (i.e., biceps brachii) and not on an arbitrary set of muscles. This could affect the generalization capability of the classification task if, for example, a different muscle is put to the test.

In this study, we address such shortcomings by using a fully automated pipeline to limit arbitrary choices. The pipeline contains units for feature extraction, feature selection, a machine learning model and hyperparameter optimization. Furthermore, the data used are collected from routine clinical practice, rather than an artificial research setting. Finally, we focus on presenting the machine learning approach in detail.

III. Data Set

The EMG data contain 380 muscle recordings from 65 muscles (at rest or at maximum contraction) based on 65 patients with IBM ($n = 20$), ALS ($n = 20$) and healthy (control group) ($n = 25$). As IBM is relatively rare, we used all available consecutive recordings from 2004-2019. As multiple muscles were examined per patient, we have the EMG of 122 muscles of healthy subjects and 258 muscles of ALS/IBM patients. All recordings were age-matched. These recordings were made within routine clinical care.

The data were collected by the department of clinical neurophysiology of the Leiden University Medical Center (LUMC), a tertiary referral center for neuromuscular diseases[1]. The EMGs were performed with concentric needle electrodes and recorded using Medelec Synergy electromyography equipment[2]. In general, the assessment takes place in three phases: with the muscle at rest, during slight activation and during (near-) maximal activation. Recording at maximal muscle activation is commonly avoided when the EMG signal appears to be normal at near-maximal activation levels, as the EMG becomes increasingly painful when the muscle is fully activated. The EMG machine routinely stores the last 40 seconds of the examination as 200 consecutive segments of $0.2s$ (we shall refer it as a trace hereafter). From every muscle recording the longest artefact-free series of consecutive $0.2s$ segments was selected rigorously by clinicians for this study, through visual inspection. This means that for all pairs of patient and muscle the number of traces varies and is at most 200.

The diagnosis was based on established clinical criteria; in brief: criteria for IBM were the presence of both typical clinical features and muscle biopsy showing atrophy, inflammation and rimmed vacuoles, criteria for ALS were typical clinical features, EMG abnormalities and progressive neurological decline, and criteria for healthy subjects were defined as subjects with atypical complaints of muscles cramps, pain, or fear of a neuromuscular disease without clinical weakness upon neurological examination and no signs of muscle weakness during a follow-up period of at least two years.

For all the patients and muscles, the data were recorded with two sampling rates; namely 4800Hz and 5000Hz comprising of 16642 and 14279 traces, respectively.
Formally, let $p \in \{1, 2, .., 65\}$ denote the patient, $m \in \{1, 2, .., 65\}$ the muscle, and $t \in \{1, 2, .., Tr_{(p,m)}\}$ the trace. Here, $Tr_{(p,m)}$ stands for the number of traces for each patient and muscle, which depends on the longest artefact-free segment of the muscle recording.
An EMG trace can then be denoted as,

$$\mathbf{s}_t^{(p,m)} := (s_1^t, s_2^t, \ldots, s_{l_t}^t)^\intercal \in \mathbb{R}^{l_t} \quad \forall(p, m, t), \tag{1}$$

where l_t is variable and depends on the sampling rate and duration of the trace. We can also denote the muscle recording for the tuple (patient, muscle) ($\forall(p, m)$) as

$$\mathbf{S}^{(p,m)} := [\mathbf{s}_1^{(p,m)}, \mathbf{s}_2^{(p,m)}, \ldots, \mathbf{s}_{Tr_{(p,m)}}^{(p,m)}]^\intercal \in \mathbb{R}^N, \tag{2}$$

where $N = l_1 + \cdots + l_{Tr_{(p,m)}}$.

As stated in section I, our approach is a binary classification task. It aims to differentiate between a normal EMG assessment from a healthy individual, and an abnormal EMG assessment from a patient with a myopathic (IBM) or neuropathic (ALS) disease. In this view, the classification targets, labeled by experts, are for each patient $p : T^p = \{\text{DISEASE}, \text{CTRL}\}$, where DISEASE includes *both* ALS and IBM and CTRL represents healthy controls. It goes without saying that a muscle recording of a patient belonging to a particular class, receives the same class label. In the following Section IV the data preprocessing is described.

IV. Data Preprocessing

For data preprocessing, we first downsampled all 5000Hz traces to 4800Hz[3]. This was done for consistency as well as for computational purposes. In addition, we renamed certain muscle groups for consistency between recordings (genioglossus $\rightarrow$ tongue). These preprocessing steps can be considered on a trace level and they transform equations (1) and (2) from before to (3) and (4), respectively, as:

$$\mathbf{s}_t^{(p,m)} := (s_1^t, s_2^t, \ldots, s_l^t)^\intercal \in \mathbb{R}^l \quad \forall(p, m, t), \tag{3}$$

where $l = 960$ at a trace duration of $0.2s$ and sampling rate of 4800Hz, and $\forall(p, m)$,

$$\mathbf{S}^{(p,m)} := [\mathbf{s}_1^{(p,m)}, \mathbf{s}_2^{(p,m)}, \ldots, \mathbf{s}_{Tr_{(p,m)}}^{(p,m)}]^\intercal \in \mathbb{R}^{l \cdot Tr_{(p,m)}}, \tag{4}$$

In the next steps we move from the trace level to the muscle level. For this we designed a unique ID which takes into account the patient identifier, the muscle examined, and the side examined ($\{\text{Left}, \text{Right}\}$). With this unique ID we

[1]https://www.spierziektencentrum.nl/location/lumc/

[2]Oxford Instruments, Abingdon, Oxfordshire, UK

[3]We used the resample function of the signal module of the scipy package https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.resample.html

grouped together traces belonging to the same patient identifier, the muscle examined, and the side examined. We then reconstructed a 5-second time series by stitching together consecutive 0.2s segments of each unique ID, which at 4800Hz results in 24000 data points per examined muscle. By creating time series of equal length, we aimed to avoid bias caused by differences in the sample length and reduce the amount of processing time required. We used the last 5s available from each recording, under the assumption that the part of the recording from the muscle at near-maximal contraction is the most likely to contain information useful for classification. Nine (9) recordings had fewer than 24000 data points, in which case the entire recording was used. Finally, we discarded 98 recordings with 960 data points in total, which correspond to a duration of $0.2s$ (with 4800Hz).

Taking Eq. (3) and (4) into account, we denote EMG traces for each patient p, muscle m, and examination side $s \in \{\text{Left}, \text{Right}\}$ as follows:

$$\mathbf{s}_t^{(p,m,s)} := (s_1^t, s_2^t, \ldots, s_l^t)^\intercal \in \mathbb{R}^l. \tag{5}$$

And the concatenation of all traces for each patient and muscle is:

$$\mathbf{S}^{(p,m,s)} := [\mathbf{s}_1^{(p,m,s)}, \ldots, \mathbf{s}_{Tr_{(p,m,s)}}^{(p,m,s)}]^\intercal \in \mathbb{R}^N, \tag{6}$$

where $N = l \cdot Tr_{(p,m,s)}$ and $l = 960$ is the trace length of $0.2s$ duration and 4800Hz sampling rate.

V. Machine Learning Pipeline

The pipeline used in this paper was originally developed for applications in the automotive industry for time series classification problems with vehicle on-board data [20], [21]. Later it has been applied to EEG (electroencephalogram) data to predict cognitive function in Parkinson's disease patients potentially eligible for DBS (deep brain stimulation) [8]. The (automated) pipeline has been continuously developed further and consists of the following steps:

1) Feature Extraction from Time Series,
2) Feature Selection,
3) Modeling, and
4) Hyperparameter Optimization of the Classifier.

The input of this fully automated pipeline are labeled time series (here: EMG). The output are performance measures after optimizing the hyperparameters.

A. Time Series Feature Extraction

The pipeline aims at being comprehensible, computationally efficient, and applicable to different time series problems. To ensure this, our pipeline uses features computed from the time series. Such features are computationally efficient to use and relatively easy to interpret.

In this paper, we propose to extract an excessive number of features from the time series and subsequently select the most significant ones for the problem at hand, based on some pre-defined feature selection criterion. Since those numerous features covers a broad range of time series characteristics, this procedure allows the application of this pipeline to various problems with very different relevant features.

In this study, the feature extraction $\boldsymbol{\mathcal{F}}$ uses the EMG recordings of each patient and muscle of each side (see section IV) as input and constructs a k-dimensional (k is the number of features) real-valued feature vector, $\boldsymbol{\mathcal{F}}\colon \mathbb{R}^N \to \mathbb{R}^k$:

$$\forall (p,m,s), \quad \mathbf{S}^{(p,m,s)} \mapsto \boldsymbol{\mathcal{F}}\left(\mathbf{S}^{(p,m,s)}\right).$$

Thus, each tuple (p,m,s) results in a feature vector which can be denoted as $\boldsymbol{\mathcal{F}}^{(p,m,s)}$. This feature vector represents the input for the feature selection procedure.

Within the feature extraction phase, for each time series (p,m,s) 63 time series characterization methods are utilized, from which by default 794 features are computed by using multiple parametrizations[4]. These features are pre-defined in the *tsfresh* package [22], [23]. In this work, *tsfresh* has been applied with its default settings. In the next step, from this generated feature space the most significant features are selected.

B. Feature Selection

The feature selection phase describes the selection of relevant features from the massive number of extracted features (from *tsfresh*) for the classification task. For each tuple (p,m,s) of patient and muscle, we use $\boldsymbol{\mathcal{F}}_{sel}^{(p,m,s)} \in \mathbb{R}^{k'}$ to represent the vector resulting from feature selection (sel stands for "selected" and k' is the number of selected features). Numerous feature selection methods have been proposed like the forward or backward selection. To even distinguish between relevant and non-relevant features the so-called feature importance can be used as a measure. Feature importance describes the mean decrease of accuracy or also the mean decrease of impurity when modeling with random forests. When in a forward selection features are added iteratively until the feature importance stagnates or deteriorates, backward elimination uses all features in the beginning and removes less important features gradually.

In our pipeline, another feature selection algorithm called *boruta* [24] is used since it has shown best performances when compared to other methods [20]. The *boruta* algorithm includes a random forest model which is build on real features and shadow features. Shadow features are generated by randomly shuffling the values of each real feature vector. As soon as a real feature exposes a higher feature importance than the maximal feature importance over all shadow features, it is considered for selection. This procedure is repeated to guarantee that the selected features have a statistically significant meaning.

C. Modeling

In the phase of modeling, a random forest model is trained with the selected features of the previous phase. We have implemented a random forest model due to its simplicity

[4]Please see https://tsfresh.readthedocs.io/en/latest/text/list_of_features.html for the detailed list of features

TABLE I
OVERVIEW OF THE APPROACHES FOR AUTOMATED EMG ASSESSMENTS WITH MACHINE LEARNING.

Approach	Description	EMG cases	Class 0 (healthy)	Class 1 (disease)	Length
1	DISEASE vs. CTRL, muscle-level	380	122	258	≤ 24000
2	DISEASE vs. CTRL, over- and under-sampling, muscle-level	380	122	258	≤ 24000

and its efficiency. Furthermore, random forests are known to achieve good performances in different domains. However, any other classifier can be implemented here. A random forest is an ensemble learning method. It is the conglomeration of several decision trees with the resulting decision being the average outcome of all those decision trees [25] in the case of regression or by taking the majority vote in case of classification.

In this EMG study, we can summarize the input to the random forest model as $\{(\boldsymbol{\mathcal{F}}_{sel}^{(p,m,s)}, \mathcal{T}^{(p)})\}$, where $p \in \{1,\ldots,65\}$, $m \in \{1,\ldots,65\}$, $s \in \{\text{Left}, \text{Right}\}$.

We have 380 intramuscular EMG recordings, of which 258 belong to patients with a neuromuscular disorder and the remaining 122 to healthy individuals. Evidently, this dataset is not balanced. Thus, in addition to the previous modeling approach we also performed a balanced approach. In detail, we used a combination of over-sampling the minority class (healthy) and under-sampling the majority class (disease), by allowing the two classes to "meet" halfway (rounded down). In other words, if the difference is 20 data-points (EMG recordings), we under-sample the majority class by 10 and over-sample the minority class by another 10. The under-sampling of the majority class happens randomly, whereas the oversampling of the minority class takes place using the well known *Synthetic Minority Over-Sampling Technique* (SMOTE) [26]. We should note here that the balancing is applied *only* to the training set in each fold of the 10-fold CV. The two modelling approaches will be called henceforth *approach 1* and *approach 2*. In addition, Table I shows an overview of the modeling approaches.

D. Hyperparameter Optimization

The optimization of hyperparameters enhances the performance of a machine learning algorithm. Table II shows the search space of the hyperparameter optimization conducted in this study. It is notable that the search space contains not only integer variables but also categorical ones. Various methods and algorithms are available for hyperparameter optimization like Grid Search, Evolutionary Algorithms and Bayesian Optimization [27]. In this study, a state-of-the-art Bayesian Optimization algorithm, namely *Mixed-integer Parallel Efficient Global Optimization* (MIP-EGO) [28], [29], is chosen due to its efficiency for optimizing expensive problems. It can handle mixed-integer categorical variables in an efficient way. MIP-EGO suggests in each iteration a candidate hyperparameter setting which is evaluated by measuring the performance of the model on a test data set. We execute MIP-EGO for 200 iterations and we use the F1-score macro as our optimization criterion, in order to take into account the class imbalance during training.

TABLE II
HYPERPARAMETER SEARCH SPACE FOR OPTIMIZING THE RANDOM FOREST CLASSIFIER.

Parameter	**Range**
Max depth of each tree	$\{None, 2, 4, 6, \ldots, 100\}$
Number of trees	$\{1, 2, \ldots, 100\}$
Max number of features when splitting a node	{`auto`, `sqrt`, `log2`}
Min number of samples required to split a node	$\{2, 3, \ldots, 20\}$
Min number of samples required in the leaf node	$\{1, 2, \ldots, 10\}$
Use bootstrap training samples?	{`True`, `False`}

VI. PATIENT-LEVEL APPROACH

The pipeline we have proposed so far operates on *the level of muscles*, meaning it predicts, for each muscle recording (constructed from the same patient and the same side), the probability of this muscle falling into the disease category. In addition, we would like to give the same prediction on the *patient-level*, which takes all prediction probabilities on the muscles from the same patient and then aggregates them to make an overall predictive decision for this patient. We will call this approach *patient-level approach*.

Four different aggregation methods are proposed for the patient-level prediction, which utilizes prediction probabilities of the recorded muscles of all the patients:

1) **Majority method**: classify the patient as being in the disease class if more than half of his examined muscles have a score greater than 0.5. Otherwise, classify him as being healthy.
2) **Median method**: classify the patient as being in the disease class if the median of the scores of his examined muscles is greater than 0.5. Otherwise, classify him as being healthy.
3) **Two-muscles method**: classify the patient as being in the disease class if at least two of his examined muscles have a score larger than 0.5. Otherwise, classify him as being healthy. The reason for using more than one muscle in this approach is that by using two muscles we reduce the impact of a potential outlier.
4) **Two-muscles average method**: classify the patient as being in the disease class if the average of two of his examined muscles with the highest score is larger than 0.5. Otherwise, classify him as being healthy.

The difference between methods 3 and 4 above can be made clear with an example. If a patient has 0.80 and 0.49 as

the highest two scores, then the two-muscles method would classify him as healthy, whereas the two-muscles-average method would classify him as being in the disease class. Thus, this seems like an interesting alternative method.

VII. Performance Evaluation

As previously mentioned, the data set used in this paper contains data of 40 patients with neuromuscular disorders and 25 healthy patients. In detail we have 380 intramuscular EMG recordings, of which 258 have a neuromuscular disorder and the other 122 are healthy. Evidently, this dataset is not balanced, and thus classification accuracy is not an appropriate performance measure, as it will overestimate the performance. We report it for approach 2, as the dataset is balanced there, and for completeness we also report it for approach 1. In this view, we have also included some other commonly employed performance measures, namely, precision, recall, F_1-score, sensitivity, specificity, ROC (Receiver operating characteristic) curve, and the Area Under the ROC (AUC). We explain these performance measures briefly as follows:

- **accuracy**: the number of correct classifications divided by the number of data points.
- **positive class**: *DISEASE* (i.e., the disease class).
- **negative class**: *CTRL* (i.e., the healthy class).
- **true positive**: correct classifications to class *DISEASE*.
- **false positive**: incorrect classifications to class *DISEASE*.
- **precision**: the number of true positive classifications divided by the total number of positive classifications.
- **recall/sensitivity**: the number of true positive classifications divided by the total number of true positives (i.e., true positive rate).
- **Specificity**: the number of true negative classifications divided by the total number of true negatives (i.e., true negative rate).
- $F_1 = 2 \times \text{precision} \times \text{recall}/(\text{precision} + \text{recall})$.
- The **ROC curve** describes the trade-off between true positive rate and false positive rate while the **area under the curve** (AUC) quantifies such a trade-off.

We calculate the $F1$-score, the recall and precision with two schemes, namely, *macro* and *weighted*. The former calculates metrics for each label (DISEASE, CTRL), and finds their unweighted mean. This does not take label imbalance into account. The latter calculates metrics for each label (DISEASE, CTRL), and finds their average weighted by the class's support (the number of true instances for each label). This alters "macro" to account for label imbalance.

Furthermore, confusion matrices or visualization methods such as ROC can provide deeper performance insights. A confusion matrix describes the frequency of cases that are correctly or incorrectly classified [30] and is considered as a useful illustration of the classification quality. Depending on the data, the ROC additionally helps understanding the performance of the model [27].

We clarify the two types of results presented in section VIII: the ones obtained from the muscle-level approach and from the patient-level approach. The former means that the results underline the performance of the automatic machine learning pipeline on the EMG recordings classification task (henceforth known as muscle-level). The latter quantifies the performance of the post-processing task which aims to classify the patients (henceforth known as patient-level), using the output of the muscle-level pipeline.

A. Muscle-Level

The resulting performance scores are based on a 10-fold cross validation (CV). In a cross validation, the data set is randomly split into K folds (here $K = 10$) and trained on $K-1$ folds and tested on the remaining Kth fold. This process is repeated until each fold has served as test set. The average of performance scores from all K folds represents the final score. In contrast to CV, when trained on all data, the models with optimized hyperparameter settings for EMG assessment achieve a final classification accuracy of 100%. This is a clear indication of model overfitting, i.e., such models would not generalize well for new patients.

We would also like to emphasize here that during the CV in the pipeline, the folds are generated in a patient level way. This means that the EMG recordings belonging to one patient are **all** included in the training or testing fold and are **never** separated between the training data and test data. This is important in order to prevent data leakage, as two different EMG recordings of one patient carry similar information about the underlying process that generated them (i.e., same pathophysiology). Each resulting performance score represents the average of 5 independent runs of the automatic machine learning pipeline.

B. Patient-Level

The resulting performance scores are based on the post-processing of the scores returned by the pipeline. For the patient-level approach we follow the procedure explained in detail in section VI. Each resulting performance score of the patient-level approach represents the average of the post-processing of the 5 independent runs of the automatic machine learning pipeline (muscle-level).

VIII. Results

In this section, the results of the muscle-level and patient-level classification tasks are presented.

A. Muscle-level results

The muscle-level approach aims at classifying intramuscular EMG recordings as either disease (ALS/IBM) or healthy. In Table III, we present the results for the muscle-level approaches 1 and 2. For clarity, approach 1 refers to the unbalanced muscle-level pipeline and approach 2 refers to the balanced muscle-level pipeline (see Table I). Furthermore, Figures 1 and 2 show the confusion matrices of both modeling approaches 1 and 2 for the training and the test set, respectively.

First of all, the achieved results indicate that a task like this can be carried out by machine learning techniques. Comparing

between approaches 1 and 2, Table III shows that approach 1 (AUC $= 0.817$) is generally better suited for this task than approach 2 (AUC $= 0.795$), although the difference between the two is minimal. Here, we take the AUC as the major performance value since it quantifies the best potential performance for both approaches while the other scores only compare them with a fixed decision threshold (0.5 in this paper). From Figures 1 and 2 we can see that the sensitivity of approach 1 is greater than that of approach 2, however the specificity of approach 2 is greater than that of approach 1. This can also be backed-up from Table III where the sensitivity of approach 1 and 2 is 0.896 and 0.816, respectively, whereas the specificities are 0.546 for approach 1 and 0.604 for approach 2. A reason for this behavior could be partially due to the fact that for approach 2 we reduce in every fold the training data of our positive class and increase the training data of our negative class in order to balance the data-points between the two labels.

Finally, in Table IV we can see the common features[5] selected in every fold of the 10-fold CV and in every single of the 5 independent runs. We show their aggregated impurity-based importance values (averaged over a 10-fold cross validation, and then averaged over all 5 repeated runs of the 10-fold CV) and the standard deviation of the means over the 5 runs. The standard deviation shows that the average importance of these features has been consistent throughout the runs and their ranking is quite reliable. These features should be further investigated for their predictive power and clinical relevance and interpretability.

B. Patient-level results

The patient-level approach aims at classifying patients as either disease (ALS/IBM) or healthy, based on the prediction scores of their intramuscular EMG recordings, from muscle-level approaches 1 and 2. In Table VI we show the performance scores of all the methods of the patient-level post-processing on approach 1 and approach 2.

The achieved results indicate again that a task like this can be carried out by machine learning techniques. Comparing the methods and approaches within Table VI, we see that the post-processing of approach 1 has higher diagnostic yield than the patient-level post-processing of approach 2. This is also backed up when comparing the AUC between the two approaches. In more details, we see that the AUC of the median and two-muscles average of the patient-level post processing of approach 1 is 0.815 and 0.798, respectively, compared to 0.786 and 0.777 of approach 2. A closer look at Table VI suggests that generally for approach 1 the majority method allows for the best results in terms of the F1 score (for both "macro" and "weighted" averages), with the two-muscles coming in the second rank, then the median method for the "macro" average and the two-muscles average for the "weighted" average. The two-muscles average method comes last in the "macro" average and the median method for the "weighted" average. For approach 2 the two-muscles come in the first place, then the two-muscles average, then the majority method, and last the median method. Note that, the AUC score is not used to compare all methods since it is not defined for the majority and two-muscles methods. Figures 3 and 4 show the ROC curves from all 5 repetitions of the median and two-muscles average methods of the patient-level post-processing of approach 1.

Finally, on Table V we see the average percentage of improvement for each patient-level's method, when using hyperparameter optimization vs not using hyperparameter optimization in both approaches. We averaged the percentages of improvement overall the performance metrics of each method. The last row shows the average improvement overall these methods. From the table we can see an average improvement of 2.94% on the patient-level when using hyperparameter optimization on approach 1, compared to using the default values of the random forest algorithm[6] (no hyperparameter optimization) and 0.75% for the patient-level of approach 2. These results directed us to apply hyperparameter optimization on both approaches 1 and 2. We can also see that hyperparameter optimization can have a positive or negative impact based on the experimental setup (approach 1 vs approach 2).

TABLE III
PERFORMANCE SCORES FOR THE MUSCLE-LEVEL **APPROACH 1** AND **APPROACH 2**. THE SCORES ARE CALCULATED ON THE TEST SET AND AVERAGED OVER A 10-FOLD CROSS VALIDATION. THE MEAN AND STANDARD DEVIATION ARE AGGREGATED FROM 5 REPEATED RUNS OF THE 10-FOLD CV.

Score	Approach 1	Approach 2
Accuracy	0.778±0.021	0.747±0.009
F1 (macro)	0.708±0.027	0.692±0.012
F1 (weighted)	0.759±0.021	0.740±0.008
Precision (macro)	0.767±0.032	0.723±0.013
Recall (macro)	0.721±0.025	0.710±0.011
Precision (weighted)	0.792±0.029	0.773±0.005
Recall (weighted)	0.778±0.021	0.747±0.009
Sensitivity	0.896±0.015	0.816±0.006
Specificity	0.546±0.037	0.604±0.025
AUC	0.817±0.023	0.795±0.031

		Predicted	
		CTRL	DIS
Actual	CTRL	85.15	24.65
	DIS	2.19	230.01

		Predicted	
		CTRL	DIS
Actual	CTRL	6.52	5.69
	DIS	2.79	23.01

Fig. 1. Confusion matrix of modeling **approach 1** for the training data (left) and test data (right). *CTRL* is the CTRL class, referring to healthy recordings and *DIS* is the DISEASE class, referring to the disease recordings. The scores are calculated and averaged over all folds of the 10-fold cross validation. The values are averaged over 5 repetitions of the 10-fold CV.

IX. CONCLUSIONS AND OUTLOOK

This paper presents an automated method for classifying electromyography (EMG) data on a muscle-level and a patient-

[5]Please see https://tsfresh.readthedocs.io/en/latest/text/list_of_features.html

[6]See here for the default values https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html#sklearn.ensemble.RandomForestClassifier

		Predicted	
		CTRL	DIS
Actual	CTRL	169.41	1.84
	DIS	0.2	171.062

		Predicted	
		CTRL	DIS
Actual	CTRL	7.38	4.82
	DIS	5.1	20.75

Fig. 2. Confusion matrix of modeling **approach 2** for the training data (left) and test data (right). *CTRL* is the CTRL class, referring to healthy recordings and *DIS* is the DISEASE class, referring to the disease recordings. The scores are calculated and averaged over all folds of the 10-fold cross validation. The values are averaged over 5 repetitions of the 10-fold CV.

TABLE IV
IMPURITY-BASED IMPORTANCE SCORES FOR THE MUSCLE-LEVEL **APPROACH 1**. THESE ARE THE COMMON FEATURES SELECTED BY BORUTA IN EVERY FOLD OF THE 10-FOLD CV AND IN EVERY REPETITION OF THE 10-FOLD CV. THE IMPORTANCE SCORES ARE CALCULATED AND AVERAGED OVER ALL FOLDS OF THR 10-FOLD CV. THE MEAN AND STANDARD DEVIATION ARE AGGREGATED FROM 5 REPEATED RUNS OF THE 10-FOLD CV.

Feature	Importance Score
percentage_of_reoccurring_values_to_all_values	4.6 ± 0.12
fft_coefficient__coeff_34__attr_"abs"	4.43 ± 0.1
fft_coefficient__coeff_31__attr_"abs"	3.53 ± 0.13
ratio_value_number_to_time_series_length'	3.48 ± 0.06
fft_coefficient__coeff_40__attr_"abs"	3.46 ± 0.12
percentage_of_reoccurring_datapoints_to_all_datapoints	2.91 ± 0.05

level method for classifying patients. Both tasks aim at classifying between healthy and not healthy. Our data set contains 65 patients and 65 muscles. As multiple muscles were examined per patient, we have the EMG of 122 muscles of healthy subjects and 258 muscles of ALS/IBM patients. The data were collected from routine clinical practice, rather than an artificial research setting.

For the muscle-level classification task our method extracts and selects the most significant features from the time series, trains a random forest model and optimizes its hyperparameters in an automated approach. For this classification task we develop two approaches; one where the data labels are kept imbalanced (approach 1) and one where we balance the labels (approach 2). The achieved results indicate that a task like this can be carried out by machine learning techniques. Comparing between approaches 1 and 2, shows that approach 1 (AUC $= 0.817$) is generally better suited for this task than approach 2 (AUC $= 0.795$), although the difference between

TABLE V
PERCENTAGE OF IMPROVEMENT OF THE PATIENT-LEVEL POST-PROCESSING OF APPROACH 1 AND APPROACH 2, USING HYPERPARAMETER OPTIMIZATION VS NO HYPERPARAMETER OPTIMIZATION. EACH ROW SHOWS THE AVERAGE IMPROVEMENT FOR THAT PATIENT-LEVEL'S METHOD PERFORMANCE METRICS. THE LAST ROW SHOWS THE AVERAGE IMPROVEMENT OVERALL THESE METHODS.

	Approach 1	Approach 2
Majority	4.73%	0.14%
Median	1.87%	−1.13%
Two Muscles	2.55%	2.59%
Two Muscles Average	2.61%	1.41%
Average Improvement	2.94%	0.75%

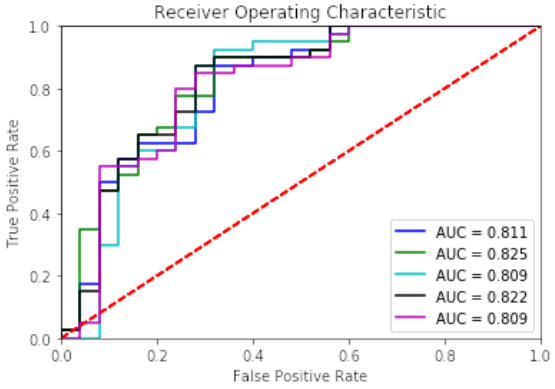

Fig. 3. ROC curves of all 5 repetitions of the **median** method on the **patient-level** post-processing of modeling **approach 1**.

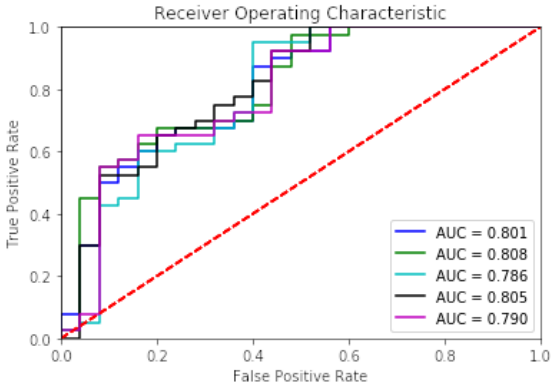

Fig. 4. ROC curves of all 5 repetitions of the **two-muscles average** method on the **patient-level** post-processing of modeling **approach 1**.

the two is minimal. Taking into consideration Figure 1 for **approach 1**, we see that the test error is slightly higher than the training error. The reason for this can be attributed to the small sample size used in this study. For **approach 2** (see Figure 2) we argue that the testing result can not be compared directly to that on the train set since the class-balancing procedure is only applied on the training set. We also see that in both approaches, sensitivity outweighs the specificity. As a screening algorithm, a high sensitivity is preferable to limit the amount of false-negatives. From a clinical point of view, sensitivity is the more important metric in this algorithm. What we should also emphasize here is that the automatically computed features, allow for a high diagnostic yield. Since EMG classification is routinely performed qualitatively, this method allows for the identification of new EMG biomarkers.

For the patient-level classification task, the achieved results indicate again that a task like this can be carried out by machine learning techniques. We see that the post-processing of approach 1 has higher diagnostic yield than the patient-level post-processing of approach 2. This is also backed up when comparing the AUC between the two approaches. In more detail, we see that the AUC of the median and two-muscles average of the patient-level post processing of approach 1 is 0.815 and 0.798, respectively, compared to 0.786 and 0.777

TABLE VI

PERFORMANCE SCORES OF ALL THE METHODS OF THE PATIENT-LEVEL POST-PROCESSING ON MODELLING APPROACHES 1 AND 2, TESTED IN THIS PAPER. THE SCORE ARE CALCULATED ON THE TEST SET AND AVERAGED IN A 10-FOLD CROSS VALIDATION. THE MEAN AND STANDARD DEVIATION ARE AGGREGATED FROM 5 REPEATED RUNS OF THE 10-FOLD CV. NOTE THAT FOR THE MAJORITY AND TWO-MUSCLE METHODS THE AUC SCORES ARE NOT APPLICABLE. THE REASON BEHIND THAT IS THAT WE DECIDED TO USE A FIXED SCORE THRESHOLD.

Approach	Method	Accuracy	F1 macro	F1 weighted	Precision macro	Recall macro	Precision weighted	Recall weighted	Sensitivity	Specificity	AUC
Approach 1	Majority	0.782±0.028	0.753±0.032	0.772±0.029	0.789±0.035	0.743±0.03	0.786±0.031	0.782±0.028	0.91±0.034	0.576±0.054	—
	Median	0.757±0.033	0.718±0.04	0.742±0.036	0.768±0.041	0.710±0.037	0.763±0.037	0.757±0.033	0.915±0.03	0.504±0.061	0.815±0.008
	Two-Muscles	0.769±0.022	0.73±0.024	0.753±0.022	0.794±0.038	0.72±0.022	0.784±0.031	0.769±0.022	0.935±0.038	0.504±0.046	—
	Two-Muscles Average	0.766±0.02	0.716±0.021	0.743±0.019	0.815±0.044	0.707±0.018	0.797±0.036	0.766±0.02	0.965±0.029	0.448±0.018	0.798±0.01
Approach 2	Majority	0.72±0.02	0.701±0.024	0.718±0.021	0.705±0.021	0.701±0.025	0.719±0.021	0.72±0.02	0.785±0.034	0.616±0.061	—
	Median	0.717±0.018	0.696±0.023	0.714±0.02	0.7±0.019	0.694±0.025	0.714±0.021	0.717±0.018	0.795±0.011	0.592±0.059	0.786±0.021
	Two-Muscles	0.738±0.022	0.707±0.021	0.729±0.021	0.732±0.03	0.701±0.019	0.736±0.025	0.738±0.022	0.865±0.034	0.536±0.022	—
	Two-Muscles Average	0.742±0.013	0.704±0.018	0.728±0.015	0.742±0.015	0.697±0.016	0.742±0.013	0.742±0.013	0.890±0.014	0.504±0.036	0.777±0.02

(a)

Actual \ Predicted	CTRL	DIS
CTRL	12.6	12.4
DIS	3.4	36.6

(b)

Actual \ Predicted	CTRL	DIS
CTRL	14.4	10.6
DIS	3.6	36.4

(c)

Actual \ Predicted	CTRL	DIS
CTRL	12.6	12.4
DIS	2.6	37.4

(d)

Actual \ Predicted	CTRL	DIS
CTRL	11.2	13.8
DIS	1.4	38.6

Fig. 5. Confusion matrices of all the methods of the **patient-level** post-processing of modelling **approach 1**. *CTRL* is the CTRL class, referring to the healthy controls and *DIS* is the DISEASE class, referring to the disease patients. (a): Median method, (b): Majority method, (c): Two-muscles method, (d): Two-muscles-average method. The entries are averaged over all 5 repetitions.

of approach 2. The results further show that the majority method yields the best results in terms of the F1 score (for both "macro" and "weighted" averages), with the two-muscles coming in the second rank, then the median method for the "macro" average and the two-muscles average for the "weighted" average. The two-muscles average method comes last in the "macro" average and the median method for the "weighted" average. Similarly, for approach 2 the two-muscles come in the first place, then the two-muscles average, then the majority method, and last the median method. Finally, we saw an average improvement of 2.94% on the patient-level when using hyperparameter optimization on approach 1, compared to using the default values of the random forest algorithm (no hyperparameter optimization) and 0.75% for the patient-level of approach 2. These results directed us to apply hyperparameter optimization on both approaches. It also indicated to us, that hyperparameter optimization can have a positive or negative impact based on the experimental setup (approach 1 vs approach 2). This, however, is still to be investigated.

To conclude, we see that the algorithms presented can assist clinicians in diagnosing if a patient has a neuropathy/myopathy or is healthy. In fact, the EMG in ALS patients is likely to show neurogenic changes (e.g., increased MUP amplitudes compared to healthy subjects), whereas the EMG of IBM patients is more likely to show myopathic changes (e.g., decreased MUP amplitudes). The fact that our approach reaches a relatively high performance in spite of the heterogeneity of the diseased group shows its potential. Indeed, performance may be higher when a similar approach is used to distinguish healthy controls from ALS- or IBM-patients as separate groups. In addition, both ALS and IBM can be patchy diseases, meaning that only a proportion of muscles may be affected at the time of the EMG recording. As the EMG signal of non-affected other muscles is expected to be similar to that of healthy controls, at least when using current qualitative assessment, it is remarkable that the performance of the muscle-level approach was relatively high. This suggests that the EMG signal of these apparently normal muscles may contain information that is used by the ML-based approach but not during routine clinical assessment.

A major limitation of this study lies in the relatively small dataset. This is unavoidable given the rarity of IBM in particular, which has a current population of less than 100 patients in the Netherlands [31]. We specifically investigated IBM and ALS patients because of the well-known clinical difficulties in interpreting the EMG of these diseases. Whether our approach works equally well for other myopathies/neuropathies remains to be established. However, as these are usually easier to classify using current clinical assessment, we would expect the performance of our ML approach to be higher, rather than lower, as well. An additional limitation of the current approach is the random selection of the 5-second EMG segment for each muscle. This selection was based on the absence of artefacts, without using any information on the level of muscle activation, although we aimed to use the last 5s available, assuming that this segment was more likely to contain information of the muscle at (near-) maximal contraction. Longer recordings, in which the clinical level of muscle activation is clearly marked, may lead to further improvements in performance, as muscle

activity at rest is different in both IBM and ALS patients compared to healthy subjects.

As next steps, a more detailed analysis of the nature of these features could point towards useful biomarkers for disease progression. Furthermore, we intend to evaluate the pipeline for IBM vs. ALS vs. healthy controls. Finally, we will explore an integrated patient-level pipeline to directly classify patients to a class.

REFERENCES

[1] M. Z. Daniel Dumitru and, Anthony A. Amato and, *Electrodiagnostic Medicine*, 2nd ed. Philadelphia: Hanley & Belfus, Sep. 2001.

[2] J. M. Gilchrist, S. D. Nandedkar, C. S. Stewart, J. M. Massey, D. B. Sanders, and P. E. Barkhaus, "Automatic analysis of the electromyographic interference pattern using the turns: amplitude ratio," *Electroencephalography and Clinical Neurophysiology*, vol. 70, no. 6, pp. 534–540, Dec. 1988. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/0013469488901514

[3] C. R. Stewart, S. D. Nandedkar, J. M. Massey, J. M. Gilchrist, P. E. Barkhaus, and D. B. Sanders, "Evaluation of an automatic method of measuring features of motor unit action potentials," *Muscle & Nerve*, vol. 12, no. 2, pp. 141–148, Feb. 1989.

[4] H. Uesugi, M. Sonoo, E. Stlberg, K. Matsumoto, M. Higashihara, H. Murashima, Y. Ugawa, Y. Nagashima, T. Shimizu, H. Saito, and I. Kanazawa, ""Clustering Index method": a new technique for differentiation between neurogenic and myopathic changes using surface EMG," *Clinical Neurophysiology: Official Journal of the International Federation of Clinical Neurophysiology*, vol. 122, no. 5, pp. 1032–1041, May 2011.

[5] J. L. Joy, S. J. Oh, and A. I. Baysal, "Electrophysiological spectrum of inclusion body myositis," *Muscle & Nerve*, vol. 13, no. 10, pp. 949–951, Oct. 1990.

[6] R. Dabby, D. J. Lange, W. Trojaborg, A. P. Hays, R. E. Lovelace, T. H. Brannagan, and L. P. Rowland, "Inclusion body myositis mimicking motor neuron disease," *Archives of Neurology*, vol. 58, no. 8, pp. 1253–1256, Aug. 2001.

[7] F. M. Cox, M. J. Titulaer, J. K. Sont, A. R. Wintzen, J. J. G. M. Verschuuren, and U. A. Badrising, "A 12-year follow-up in sporadic inclusion body myositis: an end stage with major disabilities," *Brain: A Journal of Neurology*, vol. 134, no. Pt 11, pp. 3167–3175, Nov. 2011.

[8] M. Koch, V. Geraedts, H. Wang, M. Tannemaat, and T. Bck, "Automated Machine Learning for EEG-Based Classification of Parkinsons Disease Patients," in *2019 IEEE International Conference on Big Data (Big Data)*, Dec. 2019, pp. 4845–4852.

[9] C. Pattichis and M. Pattichis, "Time-scale analysis of motor unit action potentials," *IEEE Transactions on Biomedical Engineering*, vol. 46, no. 11, pp. 1320–1329, Nov. 1999. [Online]. Available: http://ieeexplore.ieee.org/document/797992/

[10] E. K. Richfield, B. A. Cohen, and J. W. Albers, "Review of Quantitative and Automated Needle Electromyographic Analyses," *IEEE Transactions on Biomedical Engineering*, vol. BME-28, no. 7, pp. 506–514, Jul. 1981, conference Name: IEEE Transactions on Biomedical Engineering.

[11] A. Subasi, "Medical decision support system for diagnosis of neuromuscular disorders using DWT and fuzzy support vector machines," *Computers in Biology and Medicine*, vol. 42, no. 8, pp. 806–815, Aug. 2012. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0010482512000960

[12] "Electromyography (EMG) based Classification of Neuromuscular Disorders using Multi-Layer Perceptron | Elsevier Enhanced Reader," library Catalog: reader.elsevier.com.

[13] N. T. Artug, I. Goker, B. Bolat, G. Tulum, O. Osman, and M. B. Baslo, "Feature extraction and classification of neuromuscular diseases using scanning EMG," in *2014 IEEE International Symposium on Innovations in Intelligent Systems and Applications (INISTA) Proceedings*. Alberobello, Italy: IEEE, Jun. 2014, pp. 262–265. [Online]. Available: http://ieeexplore.ieee.org/document/6873628/

[14] G. R. Naik, S. E. Selvan, and H. T. Nguyen, "Single-Channel EMG Classification With Ensemble-Empirical-Mode-Decomposition-Based ICA for Diagnosing Neuromuscular Disorders," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 24, no. 7, pp. 734–743, Jul. 2016. [Online]. Available: http://ieeexplore.ieee.org/document/7153535/

[15] R. Isteni, P. A. Kaplanis, C. S. Pattichis, and D. Zazula, "Multiscale entropy-based approach to automated surface EMG classification of neuromuscular disorders," *Medical & Biological Engineering & Computing*, vol. 48, no. 8, pp. 773–781, Aug. 2010. [Online]. Available: http://link.springer.com/10.1007/s11517-010-0629-7

[16] A. P. Dobrowolski, M. Wierzbowski, and K. Tomczykiewicz, "Multiresolution MUAPs decomposition and SVM-based analysis in the classification of neuromuscular disorders," *Computer Methods and Programs in Biomedicine*, vol. 107, no. 3, pp. 393–403, Sep. 2012. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0169260710002981

[17] A. Subasi, E. Yaman, Y. Somaily, H. A. Alynabawi, F. Alobaidi, and S. Altheibani, "Automated EMG Signal Classification for Diagnosis of Neuromuscular Disorders Using DWT and Bagging," *Procedia Computer Science*, vol. 140, pp. 230–237, 2018. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1877050918320064

[18] M. Claesen and B. De Moor, "Hyperparameter Search in Machine Learning," *arXiv:1502.02127 [cs, stat]*, Apr. 2015, arXiv: 1502.02127. [Online]. Available: http://arxiv.org/abs/1502.02127

[19] B. Hudgins, P. Parker, and R. Scott, "A new strategy for multifunction myoelectric control," *IEEE Transactions on Biomedical Engineering*, vol. 40, no. 1, pp. 82–94, Jan. 1993. [Online]. Available: http://ieeexplore.ieee.org/document/204774/

[20] M. Koch, H. Wang, and T. Bäck, "Machine learning for predicting the damaged parts of a low speed vehicle crash," in *13th International Conference on Digital Information Management*, 2018, pp. 179–184.

[21] M. Koch and T. Bäck, "Machine learning for predicting the impact point of a low speed vehicle crash," in *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Dec 2018, pp. 1432–1437.

[22] M. Christ, A. W. Kempa-Liehr, and M. Feindt, "Distributed and parallel time series feature extraction for industrial big data applications," *CoRR*, vol. abs/1610.07717, 2016. [Online]. Available: http://arxiv.org/abs/1610.07717

[23] M. Christ, N. Braun, J. Neuffer, and A. W. Kempa-Liehr, "Time series feature extraction on basis of scalable hypothesis tests (tsfresh – a python package)," *Neurocomputing*, vol. 307, pp. 72–77, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231218304843

[24] M. Kursa and W. Rudnicki, "Feature selection with the boruta package," *Journal of Statistical Software, Articles*, vol. 36, no. 11, pp. 1–13, 2010. [Online]. Available: https://www.jstatsoft.org/v036/i11

[25] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning, Second Edition: Data Mining, Inference, and Prediction*, 2nd ed., ser. Springer Series in Statistics. Berlin: Springer New York and Springer Berlin, 2009.

[26] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, Jun. 2002. [Online]. Available: https://www.jair.org/index.php/jair/article/view/10302

[27] A. Geron, *Hands-on machine learning with scikit-learn & tensorflow*, 1st ed. O'Reilly Media, Inc., Sebastopol, CA, USA, 2017.

[28] H. Wang, B. van Stein, M. Emmerich, and T. Bäck, "A New Acquisition Function for Bayesian Optimization Based on the Moment-Generating Function," in *Systems, Man, and Cybernetics (SMC), 2017 IEEE International Conference on*. IEEE, 2017, pp. 507–512.

[29] H. Wang, M. Emmerich, and T. Bäck, "Cooling Strategies for the Moment-Generating Function in Bayesian Global Optimization," to appear in Evolutionary Computation (CEC), 2018 IEEE Congress on. IEEE, 2018, p. to appear.

[30] M. Hofmann and A. Chisholm, *Text Mining and Visualization: Case Studies Using Open-Source Tools*. Taylor & Francis Group, USA, 2016.

[31] U. A. Badrising, M. Maat-Schieman, S. G. van Duinen, F. Breedveld, P. van Doorn, B. van Engelen, F. van den Hoogen, J. Hoogendijk, C. Hweler, A. de Jager, F. Jennekens, P. Koehler, H. van der Leeuw, M. de Visser, J. J. Verschuuren, and A. R. Wintzen, "Epidemiology of inclusion body myositis in the Netherlands: a nationwide study," *Neurology*, vol. 55, no. 9, pp. 1385–1387, Nov. 2000.