Neurocomputing 199 (2016) 66–76

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Classification of epilepsy seizure phase using interval type-2 fuzzy support vector machines

Udeme Ekong [a], H.K. Lam [a], Bo Xiao [a], Gaoxiang Ouyang [b], Hongbin Liu [a], Kit Yan Chan [c], Sai Ho Ling [d]

[a] *Department of Informatics, King's College London, Strand, London WC2R 2LS, United Kingdom*
[b] *State Key Laboratory of Cognitive Neuroscience and Learning, School of Brain and Cognitive Sciences, Beijing Normal University, No. 19, XinJieKoWai St., HaiDian District, Beijing 100875, PR China*
[c] *Department of Electrical and Computer Engineering, Curtin University, Perth, Australia*
[d] *Centre for Health Technologies, Faculty of Engineering and Information Technology, University of Technology, Sydney, NSW, Australia*

ARTICLE INFO

Article history:
Received 14 October 2015
Received in revised form
11 January 2016
Accepted 14 March 2016
Available online 25 March 2016

Keywords:
Classification
Epilepsy
Fuzzy support vector machine
Interval type-2 fuzzy sets

ABSTRACT

An interval type-2 fuzzy support vector machine (IT2FSVM) is proposed to solve a classification problem which aims to classify three epileptic seizure phases (seizure-free, pre-seizure and seizure) from the electroencephalogram (EEG) captured from patients with neurological disorder symptoms. The effectiveness of the IT2FSVM classifier is evaluated based on a set of EEG samples which are collected from 10 patients at Peking university hospital. The EEG samples for the three seizure phases were captured by the 112 2-s 19 channel EEG epochs, where each patient was extracted for each sample. Feature extraction was used to reduce the feature vector of the EEG samples to 45 elements and the EEG samples with the reduced features are used for training the IT2FSVM classifier. The classification results obtained by the IT2FSVM are compared with three traditional classifiers namely Support Vector Machine, k-Nearest Neighbor and naive Bayes. The experimental results show that the IT2FSVM classifier is able to achieve superior learning capabilities with respect to the uncontaminated samples when compared with the three classifiers. In order to validate the level of robustness of the IT2FSVM, the original EEG samples are contaminated with Gaussian white noise at levels of 0.05, 0.1, 0.2 and 0.5. The simulation results show that the IT2FSVM classifier outperforms the traditional classifiers under the original dataset and also shows a high level of robustness when compared to the traditional classifiers with white Gaussian noise applied to it.

1. Introduction

A classification problem can be best illustrated when an object or group of objects have to be assigned into a pre-defined group or class where the assignment is made based on a number of observed features/attributes pertaining to that particular object. Classification is a very important field of research due to the advantageous nature that a classifier with high generalization ability would benefit the economical, industrial and medical fields [1]. As a result of this, extensive research has been carried out over the years and this has resulted in a large number of applications such as risk classification of loan clients [2], hand-writing recognition [3], image classification [4] and speech recognition [5].

Literature review shows that classification methods can be categorized by four types namely logic based approach (e.g. decision trees) [2], statistical approach (e.g. Bayesian classification) [6], instance-based approach (e.g. nearest neighbor algorithm [7]) and machine learning (e.g. single layer perceptrons, neural networks [8,9] and support vector machine (SVM) [10]).

The decision tree method is carried out by categorizing the inputs based on the feature values in the input [7]. A drawback of this method is that once the splitting rule makes a wrong decision, it is impossible to produce the correct path and this would therefore generate an accumulation of errors. Bayesian classifier is based on the assumption that equal prior probabilities exist for all classes [6]. The main limitation of the Bayesian classifier is that the posterior probabilities cannot be determined directly [8]. An example of the instance-based method is the k-Nearest Neighbor

E-mail addresses: udeme.ekong@kcl.ac.uk (U. Ekong), hak-keung.lam@kcl.ac.uk (H.K. Lam), bo.xiao@kcl.ac.uk (B. Xiao), ouyang@bnu.edu.cn (G. Ouyang), hongbin.liu@kcl.ac.uk (H. Liu), kit.chan@curtin.edu.au (K.Y. Chan), steve.Ling@uts.edu.au (S.H. Ling).

http://dx.doi.org/10.1016/j.neucom.2016.03.033
0925-2312/© 2016 Elsevier B.V. All rights reserved.

(kNN) [7] technique which is based on the principle that objects in a dataset generally exist in the neighborhood of other objects with similar properties. The technique finds the k nearest objects to the particular input and determines its class by identifying the most frequent class label.

The single layer perceptron can be simply described as a component that computes the sum of weighted inputs and then feeds to the system outputs. A major limitation of the single layer perceptron is that it can only learn linearly separable problems and is therefore incompatible when considering non-linear problems [9]. This problem is solved by the introduction of the Neural Network (NN). The Neural Network can be divided into 3 distinct segments: the input units which have the primary responsibility of receiving information; the hidden units which contain neurons to carry out the input–output mapping and the output units which store the processed results [7]. When the optimal connection weights and transfer functions are determined, the NN can be used as a universal approximator [11] which is able to approximate any continuous functions (e.g., hyperplanes) to any arbitrary precision in a compact domain.

The Support Vector Machine (SVM) was first proposed by Vapnik in 1995 [7] as a machine learning model which can be applied to various supervised and unsupervised learning applications [12–14]. The SVM approach can be redeveloped as Support Vector Classification (SVC) which are used for task such as pattern recognition and Support Vector Regression (SVR) which is mainly applicable to time-series applications [12]. The SVM uses the hyperplane to separate two data classes. The SVM attempts to maximize the margin between the hyperplane and the input samples which is being separated by it thereby reducing the generalization error. Data that is difficult to separate on the input space is mapped into a higher dimensional feature space for the ease of separation. The higher dimensional feature space computations are done with the use of a kernel function [7]. This feature illustrates a very important trait of the SVM which is its ability to perform well in a high dimensional feature space [15,16].

The SVM performs structural risk minimization (SRM) in order to find a trade-off between model complexity and generalization capability [7]. Therefore the SVM can achieve good generalization ability for classification problems as it can simultaneously minimize the empirical risk [10]. The SRM principle is grounded on the fact that the generalization error of the model is bounded by the sum of the empirical error and a confidence interval which is based on the Vapnik–Chervonenkis (VC) dimension [7], a higher classification performance is achieved by minimizing this bound. The SVM also provides a global optimization solution to the problem at hand and therefore provides a more credible output when compared to the neural network which provides a local optimization solution [10]. One of the drawbacks of the SVM method is its sensitivity to outliers, this stems from the fact that the same penalty weight is assigned to each sample and an outlier would significantly distort the representation of the input and therefore affect the classification performance. Another drawback is that when the SVM is applied to a classification problem with imbalanced dataset (i.e. negative samples significantly outweighs the positive samples) the optimum separating hyperplane is skewed towards the positive with the consequence that the SVM could be very ineffective in identifying targets that should be mapped to the positive class [12,15].

A relatively recent classification method is based on fuzzy logic [17] which is the theory of fuzzy sets used to handle fuzziness or imprecision in datasets. The approach attempts to assign each variable with membership functions with respect to its relative distance to the class [17,4]. There are two main types namely type-1 and type-2 fuzzy sets [18–20]. In type-1 fuzzy sets, the membership values are precise numbers in the range between 0 and 1 whilst the membership grade of a type-2 fuzzy set is a type-1 fuzzy set due to the imprecision in assigning a membership grade. As a result, type-2 fuzzy sets are effective in modeling higher level uncertainty in the human decision making process when compared to the type-1 fuzzy set, where the membership grade is distinct. In fuzzy logic, classification rules are specified by the user instead of being inherently decided upon by the machine learning method like in the SVM or NN. Therefore fuzzy logic is not a black-box method and the decision rules are clearly visible. Incorporating the mechanisms of fuzzy logic, NN and SVM, two hybrid machine learning methods namely Neural Fuzzy Network (NFN) and Fuzzy Support Vector Machine (FSVM) [13] were developed. The NFN works effectively when the amount of sample data provided is sufficient but suffers from a significantly reduced generalization performance when the amount of sample data is not enough. The FSVM however works effectively even when the amount of sample data is limited and is proven to provide higher generalization performance [13]. There are many complex systems used in industry that are prone to abrupt changes such as the random failure of components or sudden environmental disturbances. Markov jump systems (MJS) can be used to represent these systems [21–23]. In Markovian jump systems, each event governed by a Markov process corresponds to the jump in finite operation modes of practical systems. This method is used to estimate the probability of an object moving from one state to another. This is done by using observations in the historical data to estimate the probability of transition [24]. In the literature we also see fuzzy logic being applied Markovian jump systems [3–5]. Interval type-2 fuzzy logic systems can also be applied to deal with complex non-linear MJS [25].

When considering a real world application of the SVM, it is important to account for the difficulty in obtaining a precise measurement of the input data. A main deficiency of the SVM technique was its sensitivity to outliers and sample noise. This SVM deficiency is caused by the same penalty cost setting to each sample. The FSVM attempts to resolve this deficiency by assigning membership to each sample with respect to the relative importance of this sample. Hence, it reduces the impact of outliers in the input dataset [26].

The application being considered in this paper is the classification of the phases involved in the onset of an epileptic seizure, where the epilepsy signals obtained from the Electroencephalograph (EEG) using real clinical data is subjected to the novel classification technique [27,28]. This is a very challenging classification problem as the EEG has multiple features and is also contaminated with noise and distortion [29,30]. The classification technique is designed to differentiate between the 3 seizure phases namely seizure-free, pre-seizure and seizure. The early detection of seizure phases is a potentially life-saving application/research field and this is a major motivation for this research. The accurate classification/differentiation between the 3 seizure phases would give doctors and other healthcare professionals ample time to be able to prepare for the oncoming seizure. Therefore the main objective of the research carried out in this paper is to propose an adequate classifier to deal with this problem. As a result of this, an interval type-2 fuzzy support vector machine (IT2FSVM) is being proposed to deal with this problem. The IT2FSVM will be utilized to differentiate between the 3 seizure phases. The IT2FSVM is proposed due to its superior ability at dealing with uncertainties and unbalanced data [26]. This therefore provides a higher level of classification accuracy than the traditional SVM and forms the basis for the implementation of this classifier. The classification performance of the IT2FSVM technique will be compared to some traditional classifiers including the kNN technique [7], SVM [12] and naive Bayes classifier [6].

This paper is organized as follows. Section 2 reviews the SVM theory. Section 3 reviews the interval type-2 fuzzy inference system (IT2FIS). Section 5 proposes the IT2FSVM structure with a detailed schematic to illustrate how it functions. Section 6 introduces epilepsy, data collection and feature extraction. Section 7 presents the classification method to deal with the epilepsy seizure phase classification problem. Section 8 presents and analyzes the experimental results obtained from the application of the IT2FSVM method to the epilepsy seizure phase classification problem with a comparison to other existing methods followed by a discussion of the results obtained. Section 9 draws a conclusion.

2. Support vector machines

The SVM theory is reviewed in this section, this provides the theoretical background to the development of IT2FSVM. The main objective of the SVM is to create a separating hyperplane such that the distance between the hyperplane and the nearest data point in each class is maximized.

Given a dataset S containing labelled training points

$$(y_1, x_1), \dots, (y_N, x_N) \quad i = 1, 2, \dots, N \tag{1}$$

where vector x_i represents the training point, y_i represents the label and N represents the total number of samples. The vector x_i is assigned to either of the two classes and is represented by the class label $y_i \in \{-1, 1\}$. The hyperplane is ideally placed in the middle of the margin between the two classes being separated. The data points that are in close proximity to the margin are the basis of its definition and are known as support vectors (SVs) [7]. In a non-linear function, searching for the optimum hyperplane in the input space is difficult. Hence, the input space is mapped onto a higher dimensional feature space. Let $z = \varphi(x)$ represent the feature vector where x is an input vector and $\varphi(x)$ is a transformation function. The hyperplane can then be defined as

$$\omega \cdot z + b = 0 \tag{2}$$

where z is the feature space vector, ω is the weight vector and b is the scalar threshold (bias). The set S is linearly separable if there exists a combination of $\boldsymbol{\omega}$ and b that satisfies the following inequalities for all elements of the set S:

$$\begin{cases} \omega \cdot z_i + b \geq 1, & \text{if } y_i = 1 \\ \omega \cdot z_i + b \leq -1, & \text{if } y_i = -1, \ i = 1, 2, \dots, N \end{cases} \tag{3}$$

where $z_i = \varphi(x_i)$.

As the set S is not linearly separable for all of its elements, a leeway for some classification violations must be allowed in order to accommodate the elements of the set that are not linearly separable. This deficiency can be resolved by introducing nonnegative slack variables $\xi_i \geq 0$ for the samples x_i which do not satisfy (3). Hence, (3) is then modified to

$$\begin{cases} \omega \cdot z_i + b \geq 1 - \xi_i, & \text{if } y_i = 1 \\ \omega \cdot z_i + b \leq -1 - \xi_i, & \text{if } y_i = -1, \ i = 1, 2, \dots, N \end{cases} \tag{4}$$

The optimal hyperplane can be obtained as a solution to the constrained optimization problem

$$\min \frac{1}{2} \| \omega \|^2 + C \sum_{i=1}^{N} \xi_i \tag{5}$$

subject to

$$y_i(\omega \cdot z_i + b) \geq 1 - \xi_i, \quad i = 1, 2, \dots, N \tag{6}$$

$$\xi_i \geq 0 \quad i = 1, 2, \dots, N \tag{7}$$

where (5) is the convex cost function, (6) and (7) are the constraints, $\| \cdot \|$ denotes the l^2 norm (i.e. Euclidean norm), and C is known as the regularization constant which is the only free parameter in the SVM formulation and can be tuned to find a balance between margin maximization and classification violation. The optimal hyperplane can be found by constructing a Lagrangian multiplier and obtaining the dual formation:

$$\min Q(\alpha) = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} y_i y_j \alpha_i \alpha_j z_i \cdot z_j - \sum_{i=1}^{N} \alpha_i \tag{8}$$

subject to

$$\sum_{i=1}^{N} y_i \alpha_i = 0, \quad 0 \leq \alpha_i \leq C, \ i = 1, 2, \dots, N \tag{9}$$

where $\alpha = (\alpha_1, \dots, \alpha_N)$ represents the vector of the nonnegative Lagrange multipliers which satisfy the constraints in (5).

Karush–Kuhn–Tucker theorem [31] is important to the development of the SVM. The theorem states that the solution α_i to (9) satisfies the following conditions:

$$\alpha_i(y_i(\omega \cdot z_i + b) - 1 + \xi_i) = 0, \quad i = 1, 2, \dots, N \tag{10}$$

$$(C - \alpha_i)\xi_i = 0, \quad i = 1, 2, \dots, N \tag{11}$$

The equalities (10) and (11) suggest that it is only the nonzero values α_i in (8) that satisfy the constraints in (6). The values of x_i that correspond with the solution α_i are known as support vectors. The instance is correctly classified when x_i corresponds with $\alpha_i = 0$ and is a significant distance away from the decision margin.

For the construction of the optimal hyperplane $\omega \cdot z + b$, we would require that

$$\omega = \sum_{i=1}^{N} \alpha_i y_i z_i \tag{12}$$

and the scalar bias b should be determined via the Karush–Kuhn–Tucker conditions in (10).

The decision function can then be derived from (3) and (12) as

$$f(x) = \text{sgn}(\omega \cdot z + b) = \text{sgn}\left(\sum_{i=1}^{N} \alpha_i y_i z_i \cdot z + b \right) \tag{13}$$

where $\text{sgn}(\cdot)$ represents the sign function which extracts the sign (positive or negative) of a real number. As we have no knowledge of the higher dimensional feature space $\varphi(\cdot)$, carrying out the computation in (8) and (13) would be rendered impossible due to its complicated nature. An advantageous characteristic of the SVM is that it is not necessary to determine $\varphi(\cdot)$. The problem is alleviated with the aid of a kernel function which has the ability to compute the dot product of the data points in the feature space of z. It is however obligatory for these functions to satisfy Mercer's theorem [32] before they can be used for computing the dot product [26]:

$$z_i \cdot z_j = \varphi(x_i) \cdot \varphi(x_j) = K(x_i, x_j) \tag{14}$$

where $K(x_i, x_j) = \varphi(x_i) \cdot \varphi(x_j)$ is the kernel function which is used for the mapping onto a higher dimensional feature space. The kernel functions can be linear or nonlinear. The nonlinear separating hyperplane can be determined by solving the following equation:

$$\min Q(\alpha) = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} y_i y_j \alpha_i \alpha_j K(x_i, x_j) - \sum_{i=1}^{N} \alpha_i \tag{15}$$

subject to

$$\sum_{i=1}^{N} y_i \alpha_i = 0, \quad 0 \leq \alpha_i \leq C, \ i = 1, 2, \dots, N. \tag{16}$$

The decision function can then be described as follows:

$$f(x) = \text{sgn}(\omega \cdot z + b) = \text{sgn}\left(\sum_{i=1}^{N} \alpha_i y_i K(x, x_i) + b\right) \tag{17}$$

3. Interval type-2 fuzzy inference system (IT2FIS)

Fuzzy inference systems are mainly used to represent the relationship between the input and output variables in systems. Fuzzy inference systems are governed by selecting IF–THEN rules which utilize linguistic labels for the expression of rules and facts. A type-2 fuzzy inference system (T2FIS) is a fuzzy logic system where the uncertainty of the membership functions are incorporated into fuzzy set theory. In the circumstance where no uncertainty exists, a type-2 fuzzy set would reduce to a type-1 fuzzy set, this is identical to the concept of probability reducing to the determinism when the unpredictability is eradicated [33]. In order to distinguish between a type-1 and type-2 fuzzy set, a tilde symbol is placed above the symbol for the fuzzy set, in this case, A represents a type-1 fuzzy set and $\tilde{A}$ represents a type-2 fuzzy set [34]. In the research that is carried out in this paper, the IT2FIS is used instead of the T2FIS because the mathematics that is required for the IT2FIS is much simpler than the mathematics that is required for the T2FIS.

An example of triangular IT2FIS membership function is shown in Fig. 1. The dashed lines represent the lower membership function (LMF) and the dotted line represents the upper membership function (UMF). For the research that is being carried out, different kinds of membership function can be applied. However, the triangular membership function was used due to the ease of implementation. The type-1 fuzzy logic is a universal approximator in the sense that it can approximate any non-linear function in a compact domain to an arbitrary level of accuracy. This characteristic is extended to the type-2 case so we would expect a similar level of ability. With this in mind, the IT2FIS should have a high level of performance irregardless of the shape of the membership function chosen as the performance is also affected by other factors such as the number of fuzzy rules chosen. The membership function can either be predefined by the users or designed with the aid of optimization methods such as the genetic algorithm (GA). The membership function for each input is represented by seven points (p_1 to p_7) which can be optimized by the GA. Unlike in the type-1 case where the membership grade is a crisp value, the membership grade in an IT2FIS is an interval. The IT2FIS is then bounded at the two extremes of this interval to give us the LMF and UMF which are both type-1 fuzzy sets. The area between the UMF and LMF is known as the footprint of uncertainty (FOU) which is shown as the gray area in Fig. 1.

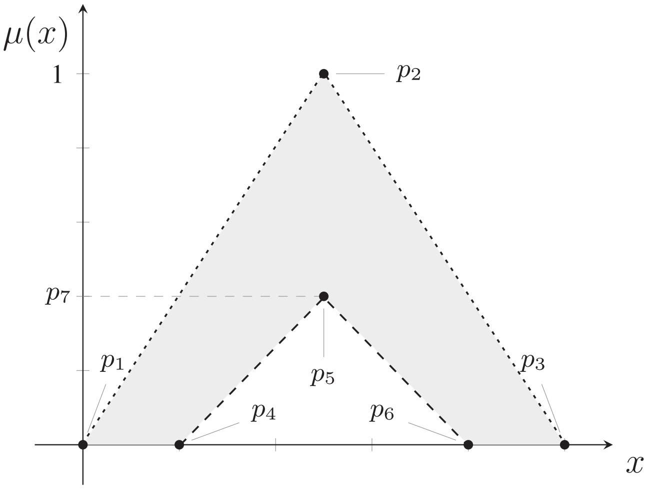


Fig. 1. An example of IT2 membership functions. Dashed line: lower membership function. Dotted line: Upper membership function. Gray area: footprint of uncertainty.

Type-2 fuzzy sets are more prevalent than type-1 fuzzy sets in rule-based fuzzy logic systems as they have a higher level of nonlinearity and therefore type-2 fuzzy sets have the ability to model uncertainties better than the type-1 fuzzy sets with less number of rules. The structure of the IT2FIS detailing the input–output relationship is shown in Fig. 2. The IT2FIS consists of 5 major components [35]: fuzzifier, fuzzy rules, inference engine, type-reducer and defuzzifier. The crisp input is first transformed into fuzzy sets in the fuzzifier block as the rule base is activated by fuzzy sets and not numbers. In the fuzzification stage, when the measurements are perfect the input is modelled as a crisp dataset, when the measurements are noisy but stationary it is modelled as an interval type-2 fuzzy set. After the input is fuzzified, the fuzzy input set is then mapped onto the fuzzy output set with the aid of the inference block. This is achieved by quantifying each rule using fuzzy set theory and then using the mathematics behind fuzzy set theory to obtain an output for each rule. The output of the fuzzy inference block would then contain one or more fuzzy output sets. The fuzzy output sets are then converted into a crisp output with the aid of the output processing unit. In an IT2FIS the output processing unit consists of two blocks: the type-reducer and the defuzzifier blocks. In the first step, the IT2 fuzzy output set is reduced to an interval-valued type-1 fuzzy set through type-reduction.

Given an IT2FIS with n inputs $x_i \in X_i, \ldots, x_n \in X_n$ to give a singular output $y \in Y$. The rule base for this IT2FIS consists of K IT2 fuzzy rules expressed in the following form [19]:

$$R^k : \text{If } x_1 \text{ is } \tilde{F}_1^k \text{ and} \cdots \text{and } x_n \text{ is } \tilde{F}_n^k \text{ THEN } y \text{ is } \tilde{G}^k \tag{18}$$

where $k = 1, \ldots, K$, $\tilde{F}_n^k$ and $\tilde{G}^k$ represent type-2 fuzzy sets.

The rules are responsible for the mapping of an input domain X to an output domain Y. Experimentation has shown that the general T2FIS model has high computational costs and complexity. This has resulted in the development of the IT2FIS which makes the computation simplified. The membership grades for interval fuzzy sets can be portrayed by their lower and upper membership grades of the FOU. The output of the firing strength for an IT2FIS ω_i is represented by a lower and upper bound i.e., $\omega_i = [\underline{\omega}_i, \overline{\omega}_i]$. The defuzzified output is obtained by type reduction which is implemented using the KM algorithm [35] given in Section 4:

4. KM algorithm

The first step of defuzzification is type reduction, where a type-2 fuzzy set is reduced to a type-1 fuzzy set. The KM algorithm which was developed by Karnik and Mendel [35] is an example of such a method. The KM algorithm is iterative and has fast convergence rates, hence its suitability for the research conducted in this thesis. The iterative procedure produces an upper and lower bound of the output. The second step of output processing occurs after type-reduction. In the case of the KM algorithm being used as

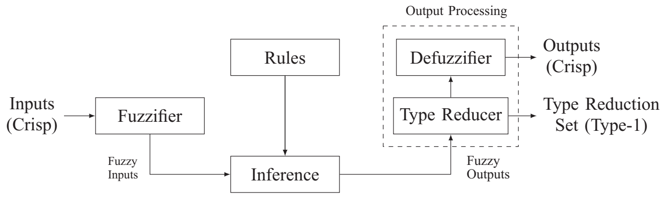


Fig. 2. Block diagram showing the IT2FIS.

a type-reducer, the type-reduced set is confined to a finite interval of numbers, the defuzzifier then obtains the defuzzified value (which is a crisp output) by calculating the average of the upper and lower bounds of this interval. A detailed description of the KM algorithm is shown below in Sections 4.1 and 4.2.

4.1. Lower bound

(1) Determine the lower bound of the output $\underline{x}_i$ $(i=1,\ldots,n)$ in ascending order and then assign the same labels to them such that $\underline{x}_1 \le \underline{x}_2 \le \cdots \le \underline{x}_n$.
(2) Match the weights ω_i with the corresponding $\underline{x}_i$ and reassign the labels to match with the new $\underline{x}_i$ which are now in ascending order.
(3) Initialize ω_i, i.e.,

$$\omega_i = \frac{\underline{\omega}_i + \overline{\omega}_i}{2} \quad \text{where } i=1,\ldots,n \tag{19}$$

then calculate

$$y = \frac{\sum_{i=1}^{n} \underline{x}_i \omega_i}{\sum_{i=1}^{n} \omega_i} \tag{20}$$

(4) Determine the pivot point p where $(1 \le p \le N-1)$ such that

$$\underline{x}_p \le y \le \underline{x}_{p+1} \tag{21}$$

(5) Assign the firing strength as

$$\begin{cases} \overline{\omega}_i, & i \le p \\ \underline{\omega}_i, & i > p \end{cases} \tag{22}$$

then calculate

$$y' = \frac{\sum_{i=1}^{n} \underline{x}_i \omega_i}{\sum_{i=1}^{n} \omega_i} \tag{23}$$

(6) Check if $y' = y$; If yes, stop and set $\underline{y} = y$, if no, go to step 7
(7) Set $y = y'$ and go to step 3

4.2. Upper bound

(1) Define the upper bound of the output $\overline{x}_i$ $(i=1,\ldots,n)$ in ascending order and then assign the same labels to them such that $\overline{x}_1 \le \overline{x}_2 \le \cdots \le \overline{x}_n$.
(2) Match the weights ω_i with the corresponding $\overline{x}_i$ and reassign the labels to match with the new $\overline{x}_i$ which are now in ascending order.
(3) Initialize ω_i i.e.

$$\omega_i = \frac{\underline{\omega}_i + \overline{\omega}_i}{2} \quad \text{where } i=1,\ldots,n \tag{24}$$

then calculate

$$y = \frac{\sum_{i=1}^{n} \overline{x}_i \omega_i}{\sum_{i=1}^{n} \omega_i} \tag{25}$$

(4) Determine the pivot point p where $(1 \le p \le N-1)$ such that

$$\overline{x}_p \le y \le \overline{x}_{p+1} \tag{26}$$

(5) Assign the firing strength as

$$\begin{cases} \underline{\omega}_i, & i \le p \\ \overline{\omega}_i, & i > p \end{cases} \tag{27}$$

then calculate

$$y' = \frac{\sum_{i=1}^{n} \overline{x}_i \omega_i}{\sum_{i=1}^{n} \omega_i} \tag{28}$$

(6) Check if $y' = y$; If yes, stop and set $\overline{y} = y$, if no, go to step 7
(7) Set $y = y'$ and go to step 3

The defuzzified output of the IT2FIS is given as:

$$y = \frac{\overline{y} + \underline{y}}{2} \tag{29}$$

5. Interval type-2 fuzzy support vector machines (IT2FSVMs)

In this section, the mechanism of the IT2FSVM classifier is discussed. The standard SVM classifier is used for this hybrid classification mechanism which involves the merging of an IT2FIS with an SVM to form the IT2FSVM. The IT2FSVM can be characterized as a multiple-input–single-output classifier. The ability of the IT2FIS to handle uncertainty makes it very complementary to the SVM in solving difficult non-linear problems.

The overall IT2FSVM architecture is shown in Fig. 3. The feature vector input is obtained after feature extraction has been carried out on the EEG input data to extract the relevant features. Details of this feature extraction method can be found in Section 6.1. As the hyperplane can only separate 2 classes, multiple SVMs are required as there are more than 2 classes in a classification problem. For the application in this chapter which is to differentiate between the epileptic seizure stages, multiple SVMs are required as there are three classes (seizure-free, pre-seizure and seizure). There are three IT2 SVM blocks in the diagram which are used to individually separate between the seizure phases. IT2 SVM 1 separates between the seizure-free and pre-seizure phases with the label "-1" indicating the input data belongs to the seizure-free class and label "1" indicating the input data belongs to the pre-seizure class. IT2 SVM 2 separates between the seizure-free and seizure phase with the label "-1" indicating the input data belongs to the seizure-free class and label "1" indicating the input data belongs to the seizure class. Finally, IT2 SVM 3 separates between the pre-seizure and seizure phase with the label "-1" indicating the input data belongs to the pre-seizure class and label "1" indicating the input data belongs to the seizure class. The output labels of the three IT2 SVM blocks are presented in $Output_1$ to $Output_3$ which are then subjected to a rule-based class determiner in order to determine what the final classification would be.

The rule based class determiner system for selecting the final classification output for the IT2FSVM is shown in Table 1. The final class is a whole number between 1 and 3 where "1" represents the seizure-free phase, "2" represents the pre-seizure phase and "3" represents the seizure phase.

The IT2FSVM block consists of a feature vector input, 3 fuzzy rules each consisting of two SVMs associated with the lower and

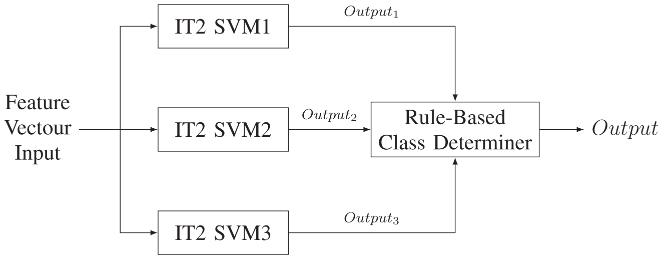


Fig. 3. Block diagram of IT2FSVM.

Table 1
Table showing the if–then rules used by the rule based class determiner system. Table showing the if–then rules. Class 1: seizure-free, class 2: pre-seizure, class 3: seizure.

Case	$Output_1$	$Output_2$	$Output_3$	Final class ($Output$)
1	−1	−1	−1 or 1	1
2	1	−1 or 1	−1	2
3	−1 or 1	1	1	3
4	1	−1	1	3
5	−1	1	−1	3

upper membership functions and a defuzzification block which is used to produce the final crisp output. The original EEG input data had a 19×100 vector input and feature extraction is used to reduce it to a 45-input feature vector which is used as the input of the IT2SVM.

The final output of the SVM block is obtained by combining the outputs of the SVMs with the aid of a membership function where membership grades or weights are assigned to each output to depict the impact that it would have on the final output. The number of fuzzy rules can be defined by any integer value but an increase in the number of fuzzy rules would lead to a slower convergence of training and also a higher computational cost of the system. In this chapter, there are 3 fuzzy rules employed to implement the IT2FSVM. The membership grade is obtained from the membership function which is defined by the user and the shape of the membership function is a triangle as shown in Fig. 1. The shape of the membership function is represented by the points p_1 to p_7 which are optimized by the GA.

Referring to Fig. 3, we have three IT2 SVMs. Each IT2 SVM is governed by the following rules:

$$R^j: \text{ If } \|x\| \text{ is } \tilde{F}^j \text{ THEN } y \text{ is } \tilde{G}^j, \quad j=1,2,3 \tag{30}$$

where $\|x\|$ is the normalized input which is described further in Section 7. $\tilde{F}^j$ is defined as an IT2 triangular membership function as shown in Fig. 1 and $\tilde{G}^j$ is a singleton with $\underline{\mathrm{SVM}}_{jk}$ as LMF and $\overline{\mathrm{SVM}}_{jk}$ as UMF, $k=1$, 2, 3, denoting the number of IT2FSVMs in Fig. 2. $\underline{\mathrm{SVM}}_{jk}$ and $\overline{\mathrm{SVM}}j_k$ are two SVMs with the output $\underline{Out}_{jk}$ and $\overline{Out}_{jk}$ defined by the following hyperplanes:

$$\underline{Out}_{jk} = \mathrm{sgn}(\underline{\omega}_{jk}\cdot z+\underline{b}_{jk}) = \mathrm{sgn}\left(\sum_{i=1}^{N}\alpha_{ijk}y_iK(x_i,x)+\underline{b}_{jk}\right) \tag{31}$$

$$\overline{Out}_{jk} = \mathrm{sgn}(\overline{\omega}_{jk}\cdot z+\overline{b}_{jk}) = \mathrm{sgn}\left(\sum_{i=1}^{N}\alpha_{ijk}y_iK(x_i,x)+\overline{b}_{jk}\right) \tag{32}$$

where $j=1$, 2, 3 denotes the j-th (lower or upper) SVM in Fig. 2 and $k=1$, 2, 3 denotes the k-th IT2 SVM in Fig. 2. The $Output_k$ of the IT2 SVM k can then be obtained by the defuzzification process outlined in Section 4. The rule-based class determiner would then make the final class decision.

6. Absence epilepsy

Epilepsy, which is characterized with its ability to instantiate recurrent seizures (an interruption of normal brain functions) which are unforeseen in nature is a very common and significant neurological disorder caused by a sudden discharge of cortical neurons [27,28]. Epileptic seizures are classified as either partial (involving focal brain regions) or generalized (where it involves a widespread region of the brain across both hemispheres) [36]. The length of time for the seizure occurrence varies from a few seconds up to a minute with some of the effects including momentary lapse of consciousness for the sufferer of the seizure [36]. A complete loss of consciousness occurs when the epileptic activity involves both the cortical and subcortical structures of the brain and this occurrence is known as an absence seizure.

The unexpected nature of these seizures has proven to have an adverse effect on the quality of life for those who are suffering from them. The impact is most prevalent in the formative stages of a child's life as we see an increase in the requirements for special education and also a higher incidence of below-average school performance [28,37]. It also proves life-threatening in situations where the sufferer is isolated at the time of its occurrence and there is no experienced or medical help on hand to alleviate the situation. Therefore having an accurate understanding or predictive model for the pre-seizure phase (the transition towards an absence seizure occurrence) is a very vital task as it would provide the sufferers and their carers enough notice of the upcoming seizure so they could prepare themselves and dampen the impact of the seizure occurrence.

Absence seizures can be best characterized by the spike-and-wave discharges (SWDs) which are as a result of synchronized oscillations in the thalamocortical networks of the brain [38,39]. The classification process of EEG signals consists of two main parts which are feature extraction and classification. In the literature, there are a wide range of available feature extraction methods which range from the traditional methods to the non-linear methods. Traditional methods include the Fourier transform and also spectral analysis whilst the non-linear methods include Lyapunov exponents [28,40], correlation dimension [28] and similarity [41]. After feature extraction has been implemented to the raw data, the extracted features are then used and applied to the pre-determined classification technique. There are a wide range of classification techniques for EEG classification in the literature, examples of these include the artificial neural network [42,43] and also the neuro-fuzzy systems [44–46].

For this particular problem of accurately classifying and thereby predicting the onset of an epileptic seizure, the extracted features are applied to various classifiers (kNN, naive Bayes, SVM and FSVM) with the main aim of being able to recognize and distinguish between the 3 seizure phases (seizure-free, pre-seizure and seizure phase). The raw data obtained for the simulations being carried out were obtained from the Peking University by the aid of 10 patients who were suffering from absence epilepsy, their ages ranging from 6 to 21 years old. The study has been approved by the ethics committee of Peking University People's Hospital and the patients all signed documents in consent of their clinical data being used for research purposes. The EEG data (which was sampled at a frequency of 256 Hz with the aid of a 16-bit analogue-to-digital converter and then filtered within a frequency band of 0.5–35 Hz) was recorded by the Neurofile NT digital video EEG system using a standard international 10-20 electrode placement (Fp1, Fp2, F3, F4, C3, C4, P3, P4, O1, O2, F7, F8, T3, T4, T5, T6, Fz, Cz and Pz).

There are 3 sets of EEG signals which are extracted from the 3 seizure phases (seizure-free, pre-seizure and seizure) to obtain 112 2-s 19-channel EEG epochs from 10 patients for each dataset. The timing of the onset and offset in the SWDs was identified by a neurologist and these SWDs were identified to be large amplitude 3–4 Hz discharges with a spike-wave morphology typically lasting above a second in duration. The criteria for determining the different seizure phases are that there is an interval between the seizure-free phase and beginning of the seizure phase which is greater than 15 s, the interval is between 0 and 2 s before the occurrence of the seizure and that the interval occurs during the first 2 s of the absence seizure. A more detailed description of the procedure for data collection can be found in [47,48].

6.1. Feature extraction

The feature extraction procedure is very vital in the classification process as it obtains the relevant characteristics and information from a large dataset (EEG signals in this instance). This procedure has the knock-on effect of simplifying the dataset and also reducing the effect of redundant data points that have little or no effect in the classification of the dataset. This procedure is important in improving the performance of the classifier as classification is more effective when the classifier is subject to fewer data points.

For the EEG case being undertaken, there are 19 columns (19 channels) of signal output. The 19 columns represent signals that were drawn from 19 EEG sensors with each column containing 100 samples. The purpose of the feature extraction being carried out here is to extract the relevant features from the 19×100 dataset and thereby reducing the dimensionality.

Research into the existing literature provides evidence to suggest that the 19 channels of the EEG data vary in importance with regard to classification. It was observed that some of the channels have a lesser impact on the classification of the EEG and the exclusion of these channels has been investigated in [29,30]. Both studies have discovered that some of the channels (F3, Fz, F4, C3 and Cz) are the most significant ones for the classification between the seizure-free and seizure patients and the remaining electrodes are found to have relevant information for the classification between the different seizure phases.

In the research carried out in this paper, the most relevant channels were selected by considering different combinations. The research showed that the 1st, 2nd, 3rd, 4th, 5th, 6th, 11th, 12th, 13th, 14th channels out of the 19 channels contain the most significant information for classification, which is in agreement with the results in [29,30] that channels F3, Fz, F4, C3 and Cz contain the most important information. For each of the channels, a feature vector containing the time-domain and frequency-domain components of the dataset is created [43]. The first part of the feature vector composed of computations in the time-domain such as the standard deviation, second order norm, third order norm, fourth order norm, absolute sum, maximum value and minimum value of the 100 sample points from each channel. The second part is composed of computations in the frequency domain such as the mean frequency, maximum frequency, minimum frequency, standard deviation of frequency, windowing filtered mean frequency and windowing filtered maximum frequency of each chosen channel will form the second part of the feature vector.

Since the computations would result in a large vector which would be difficult to classify, the principal component analysis (PCA) is used to reduce the number of dimensions in the feature vector. Given that each channel has its own particular characteristics, we choose different principal components for each channel. After the number of dimensions is reduced, we finally have 45 points which form the feature vector. This feature vector is then applied to the pre-determined classifiers.

7. Method

A classifier based on the proposed IT2FSVM structure has been implemented for the classification of the 3 seizure phases with the aid of the feature vectors obtained from the feature extraction method as shown in Section 6.1. The structure of the IT2FSVM consists of 3 IT2 SVM blocks that are used to distinguish between the 3 seizure phases. Fig. 3 shows the overall structure of the FSVM classifier which consists of 18 45-input–single-output SVMs (6 for each of the IT2 SVM blocks). The 3 sets of SVMs attempt to distinguish between 3 classes of data stems from the fact that the SVM can only separate between 2 classes at any given time.

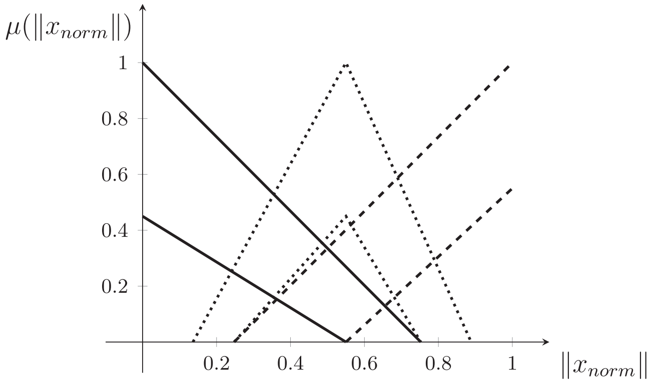


Fig. 4. Membership functions for SVM Block 1. Dotted line: membership function for rule 1, straight line: membership function for rule 2, dashed line: membership function for rule 3.

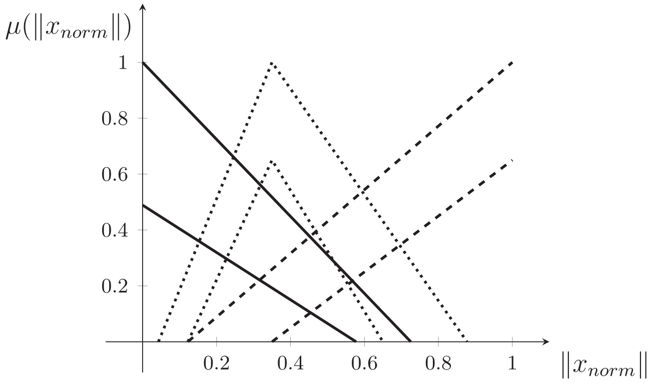


Fig. 5. Membership functions for SVM Block 2. Dotted line: membership function for rule 1, straight line: membership function for rule 2, dashed line: membership function for rule 3.

There are 3 fuzzy rules for each of the IT2 SVM blocks. The parameters of the triangular membership functions, i.e., p_1 to p_7, as shown in Fig. 1 are optimized by the GA in order to influence the shape of the membership functions. The GA optimization is performed to maximize the classification accuracy using 70% of dataset as the training samples. The rest 30% of dataset are used as the test samples. The lower and upper membership functions for SVM Blocks 1–3 after training are shown in Figs. 4–6. The membership grade is represented on the y-axis and the normalized inputs are represented on the x-axis. The normalized input denoted as x_{norm} is calculated as follows:

$$x_{norm} = \overline{x}_1^2 + \overline{x}_2^2 + \cdots + \overline{x}_N^2 \tag{33}$$

where

$$\overline{x}_i = \frac{x_i}{\max(x) - \min(x)}, \quad i = 1, 2, \ldots, N, \tag{34}$$

x_i is the i-th element of the vector x, min(x) and max(x) denote the minimum and maximum value of the elements in x, respectively.

The simulations that have been conducted with MATLAB. The control parameters of the GA are shown in Table 2. Different combinations of kernel functions are utilized in the SVMs. The optimal combination was chosen based on its ability to maximize the classification accuracy of the classifier. The value for the regularization constant C was chosen via trial and error. Different values were implemented and the value for C which produced the best results was retained. The parameters used for the SVM are as follows: in the IT2 SVM1, there are 6 SVMs used, with all utilizing

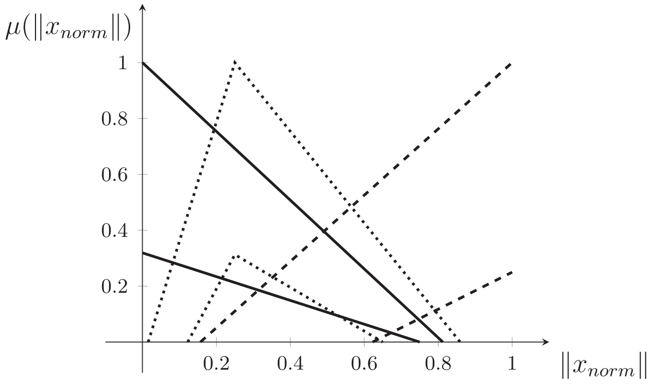


Fig. 6. Membership functions for SVM Block 3. Dotted line: membership function for rule 1, straight line: membership function for rule 2, dashed line: membership function for rule 3.

Table 2
GA parameters.

Parameter	Value
Number of iterations	10
Population size	20
Selection	Stochastic uniform selection function
Elitism	Yes (Best two chromosomes are passed onto the next generation)
Crossover	Scattered crossover
Crossover fraction	0.8
Mutation	Gaussian mutation
Stopping criterion	It stops when the weighted average relative change in the best fitness function value over 100 generations is less than or equal to 10^{-6}

Table 3
Summary of training samples classification performance for EEG signal with original dataset. Classifier 1: FSVM classifier, classifier 2: traditional SVM classifier, classifier 3: k-Nearest Neighbor classifier, 4: naive Bayes classifier.

Classifier	Classification accuracy (%)			
	Average	Class 1	Class 2	Class 3
1	99.0510	100.000	97.1400	100.0000
2	86.6667	100.0000	90.0000	70.0000
3	100.0000	100.0000	100.0000	100.0000
4	77.1400	90.0000	41.4333	100.0000

Table 4
Summary of testing samples classification performance for EEG signal with original dataset. Classifier 1: FSVM classifier, classifier 2: traditional SVM classifier, classifier 3: k-Nearest Neighbor classifier, 4: naive Bayes classifier.

Classifier	Classification accuracy (%)			
	Average	Class 1	Class 2	Class 3
1	87.7800	100.000	70.0000	93.3300
2	71.1100	90.0000	70.0000	53.333
3	56.6667	96.6700	23.3333	50.0000
4	77.7778	100.0000	33.3333	100.0000

the RBF kernel function with the width of the RBFs for all 6 of them set to $\sqrt{1/200}$, and the regularization constant $C=500$; in IT2 SVM2 6 SVMs are used, with the polynomial kernel function applied in all cases and the degree of polynomial set to 2, and $C=5000$; in IT2 SVM3 the kernel function utilized for all SVMs is the quadratic kernel function with $C=500$.

In order to obtain an appreciation of the robustness of the proposed classifier, white Gaussian noise with the levels of 0.05, 0.1, 0.2 and 0.5 have been added to the original test dataset. Under these noisy conditions, the simulations were carried out 10 times for each of the noise levels and four statistical factors namely worst, average, best and standard deviation of classification accuracy were calculated. We take these four statistical factors into account since the noisy data is random in nature and drawing conclusions from a single simulation would not accurately evaluate the robustness of the classifier to noise.

8. Experimental results/discussion

The proposed IT2FSVM classifier is used to classify between the 3 epilepsy seizure phases using the feature vector that has been obtained by the method detailed in Section 6.1. For comparison purposes, 3 traditional classifiers (kNN, naive Bayes and SVM classifiers) are considered. When traditional SVM classifier is considered, they are connected in the classifier structure as shown in Fig. 2, i.e., replacing the IT2SVM with the traditional SVM. For the design of the hyperplane, all three traditional SVMs take the RBF kernel with the width of $\sqrt{1/1400}$ and regularization constant $C=500$. The computational process for the classifiers used is identical to the standard algorithms in the literature. Readers are referred to SVM [49], IT2FIS [50], GA [51], naive Bayes [52] and kNN [53] for further examples, information and tutorials.

The classification accuracy with respect to the training dataset for all classifiers is given in Tables 3 and 4. The tables show the training and testing classification accuracy from the best performed classifiers during the design. They tabulate the worst (among the three classes), best (among the three classes), average (over the three classes) and individual class classification accuracy for both training and testing dataset.

It can be seen from Table 3 that the kNN classifier performs the best in terms of average classification accuracy of 100%. The IT2FSVM classifier comes in the second place with 99.0510% (less than 1% compared with the kNN classifier). This however is not an indication of the kNN being a superior classifier as we see that it suffers from a significant reduction in its average classification performance when exposed to unseen test data with and without noise as seen in column 3 of Table 4 and column 2 of Tables 5–7. The 100% average training accuracy seen in Table 3 is reduced to 56.6667% in Table 4 when the classifier is subjected to the test data. Another significant impact of this is that the kNN has an individual testing classification accuracy of 23.3333% as seen in the 5th column of Table 4 when classifying the pre-seizure phase (class 2), this is significant because the accurate classification of the pre-seizure phase is a core objective in addressing the problem of epilepsy seizure phase classification. This would give the patients the advance warning and therefore sufficient time to prepare for the onset of the seizure. The SVM and naive Bayes are ranked third and fourth. Table 4 shows that the IT2FSVM classifier outperforms other classifiers in terms of average classification accuracy for testing dataset. It also shows that the IT2FSVM demonstrates an outstanding generalization ability dealing with unseen data. Compared with other classifiers, the average testing classification accuracies are 10–21% higher. The results show that the naive Bayes classifier performs the worst to the testing data and its generalization capability is the poorest. Referring to the worst individual class testing classification accuracy, IT2FSVM can still achieve 70% while other degrade around 23–50%.

Tables 5–7 show the testing classification accuracy for the testing data subject to Gaussian noise with the levels of 0.05, 0.1, 0.2 and 0.5. The experiments were repeated 10 times for each classifiers. The “Worst” and the “Best” columns show the worst

Table 5
Summary of testing classification performance for EEG signal under dataset subject to noise level of 0.05. Classifier 1: FSVM classifier, classifier 2: traditional SVM classifier, classifier 3: k-Nearest Neighbor classifier, 4: naive Bayes classifier.

Classifier	Classification accuracy (%)						
	Worst	Mean	Best	Std	Class 1	Class 2	Class 3
1	62.2200	66.1100	68.8900	0.0211	8.3000	93.0000	97.0000
2	61.1100	66.2200	68.8900	0.0235	11.1333	96.0000	97.3333
3	56.6700	57.8900	58.8900	0.0176	96.0000	25.0000	52.6700
4	77.7778	78.3333	80.0000	0.7857	99.0000	37.8900	100.0000

Table 6
Summary of testing classification performance for EEG signal under dataset subject to noise level of 0.1. Classifier 1: FSVM classifier, classifier 2: traditional SVM classifier, classifier 3: k-Nearest Neighbor classifier, 4: naive Bayes classifier.

Classifier	Classification accuracy (%)						
	Worst	Mean	Best	Std	Class 1	Class 2	Class 3
1	74.4400	79.4400	85.5600	0.0034	55.3333	66.3333	99.0000
2	66.6700	68.6700	70.0000	0.0126	15.0000	89.0000	98.3333
3	54.4400	56.2200	57.8800	0.0228	92.6700	22.0000	54.0000
4	76.6667	78.8889	82.2222	1.8251	100.0000	33.6667	100.0000

Table 7
Summary of testing classification performance for EEG signal under dataset subject to noise level of 0.5. Classifier 1: FSVM classifier, classifier 2: traditional SVM classifier, classifier 3: k-Nearest Neighbor classifier, 4: naive Bayes classifier.

Classifier	Classification accuracy (%)						
	Worst	Mean	Best	Std	Class 1	Class 2	Class 3
1	73.3300	78.0000	83.3300	0.0384	94.0000	40.6667	99.3300
2	72.2200	74.6700	80.0000	0.0250	27.6667	79.3333	99.3333
3	50.0000	53.3333	55.7800	0.0207	91.6667	21.6667	54.0000
4	76.6667	79.0000	82.2222	1.8898	99.3333	34.3333	100.0000

Table 8
Summary of testing samples classification performance for EEG signal under dataset subject to noise level of 0.2. Classifier 1: FSVM classifier, classifier 2: traditional SVM classifier, classifier 3: k-Nearest Neighbor classifier, 4: naive Bayes classifier.

Classifier	Classification accuracy (%)						
	Worst	Mean	Best	Std	Class 1	Class 2	Class 3
1	73.3300	78.0000	82.2200	0.0295	86.0000	48.0000	100.0000
2	67.7800	68.6700	70.0000	0.0126	36.3333	76.3333	98.3333
3	54.4444	56.6700	58.8900	0.0236	92.6700	23.0000	54.3333
4	75.5556	78.2222	80.0000	1.5585	99.6667	33.6667	100.0000

and best testing individual class classification accuracies among the 10 experiments. The “Mean” and “Std” columns show the mean and standard deviation of the average testing accuracies of the three classes of the 10 experiments. The columns for “Class 1”, “Class 2” and “Class 3” show average testing classification accuracy for classes 1 to 3, respectively, of the 10 experiments (Table 8).

In general, the classification accuracies decrease for all classifiers when the noise level increases. In most of the cases, the average testing classification of IT2FSVM and naive Bayes classifiers achieve the best result. However, when it is down to the individual class classification accuracy, especially for higher noise levels (0.1, 0.2 and 0.5), the IT2FSVM performs more robustly with the lowest class classification accuracy of 40% while other classifiers obtain lower class classification accuracies ranging from 15% to 36%. Similar to the comment concerning the kNN and its poor performance in accurately classifying the pre-seizure phase (Class 2), it is important to also note that the naive Bayes classifier exhibits a relatively poor ability to classify the pre-seizure phase as we see that the SVM and IT2FSVM provide superior class classification for the pre-seizure phase in the training, noise-free testing and noise testing of both classifiers. This is a critical difference between these classifiers. The IT2FSVM however achieves a superior overall/average classification accuracy when compared to the SVM and this result shows the superiority and suitability of the IT2FSVM for classifying the three epilepsy seizure phases.

9. Conclusion

In this paper, a novel classification method, IT2FSVM, was proposed to use EEG to classify the epileptic seizure from patients with neurological disorder symptoms, where the three epileptic seizure phases seizure-free, pre-seizure and seizure were taken into account. The IT2FSVM merges the SVM and IT2FIS to create a hybrid classifier which attempts to achieve more accurate classification when compared to the traditional classifiers. The simulation results show that the IT2FSVM can achieve more accurate classifications than the traditional kNN, naive Bayes and SVM method do when the classifier is subjected to the original and uncontaminated input data. The input data was then contaminated with noise in order to evaluate the robustness of the proposed IT2FSVM. The validation results show that the proposed IT2FSVM achieves a more significant level of robustness to noisy data when compared to other classification methods.

9.1. Future work

In this section, some ideas are discussed with the purpose of utilizing them for future work on the research carried out in this paper.

- Investigating different parameters for the IT2FSVM classifier to evaluate their effect on performance. This refers to the IT2FIS (e.g. membership function shape, type-reduction method), genetic algorithm (GA) parameters (e.g. mutation, crossover), and also using the GA to optimize the SVM parameters (e.g. kernel method, regularization constant).
- Expose the IT2FSVM proposed in this paper to a wider range of problems including non-classification problems like time-series prediction in order to test its viability.
- For the epilepsy seizure phase classification, we notice that the greatest difficulty is in being able to differentiate between the seizure-free and pre-seizure classes. Research could be conducted in other signal processing and feature extraction techniques that could be better suited to extracting the distinct features in both classes.
- Further application of fuzzy logic into the SVM by proposing a fuzzy kernel method and applying to a classification or time-series problem.

Future research direction will aim to optimize the membership function and the IT2FSVM architectures in order to further improve the overall classification accuracy.

Acknowledgment

The work described in this paper was partly supported by King's College London and China Scholar Council.

References

[1] P.L. Lanzi, W. Stolzmann, S.W. Wilson, Learning Classifier Systems: From Foundations to Applications, No. 1813, Springer Science & Business Media, New York (2000).
[2] J. Zurada, Could decision trees improve the classification accuracy and interpretability of loan granting decisions? in: 2010 43rd Hawaii International Conference on System Sciences (HICSS), Hawaii, U. S., IEEE, 2010, pp. 1–9.
[3] J. Pradeep, E. Srinivasan, S. Himavathi, Neural network based handwritten character recognition system without feature extraction, in: 2011 International Conference on Computer, Communication and Electrical Technology (ICCCET), Tirunelveli, India, IEEE, 2011, pp. 40–44.
[4] H. Zhang, W. Shi, K. Liu, Fuzzy-topology-integrated support vector machine for remotely sensed image classification, IEEE Trans. Geosci. Remote Sens. 50 (3) (2012) 850–862.
[5] C. Jen-Tzung, Linear regression based Bayesian predictive classification for speech recognition, IEEE Trans. Speech Audio Process. 1 (January (11)) (2003) 70–79.
[6] U. Kumar, S.K. Raja, C. Mukhopadhyay, T.V. Ramachandra, Hybrid Bayesian classifier for improved classification accuracy, IEEE Geosci. Remote Sens. Lett. 8 (November (3)) (2010) 474–477.
[7] S.B. Kotsiantis, Supervised machine learning: a review of classification techniques, Informatica 31 (2007) 249–268.
[8] G.P. Zhang, Neural networks for classification: a survey, IEEE Trans. Syst. Man Cybern. Part C: Appl. Rev. 4 (November (30)) (2000) 451–462.
[9] A. Santillana Fernandez, C. Delgado-Mata, R. Velazquez, Training a single-layer perceptron for an approximate edge detection on a digital image, in: 2011 International Conference on Technologies and Applications of Artificial Intelligence (TAAI), Taoyuan, Taiwan, IEEE, 2011, pp. 189–193.
[10] C.-T. Lin, C.-M. Yeh, S.-F. Liang, J.-F. Chung, N. Kumar, Support-vector-based fuzzy neural network for pattern classification, IEEE Trans. Fuzzy Syst. 14 (1) (2006) 31–41.
[11] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, Neural Netw. 2 (5) (1989) 359–366.
[12] Y.-H. Liu, Y.-T. Chen, Face recognition using total margin-based adaptive fuzzy support vector machines, IEEE Trans. Neural Netw. 18 (1) (2007) 178–192.
[13] H.-S. Yan, D. Xu, An approach to estimating product design time based on fuzzy-support vector machine, IEEE Trans. Neural Netw. 18 (3) (2007) 721–731.
[14] H. Lam, J. Prada, Interpretation of handwritten single-stroke Graffiti using support vector machines, Int. J. Comput. Intell. Appl. 8 (04) (2009) 369–393.
[15] Y. Chen, J.Z. Wang, Support vector learning for fuzzy rule-based classification systems, IEEE Trans. Fuzzy Syst. 11 (6) (2003) 716–728.
[16] X. Yang, G. Zhang, J. Lu, J. Ma, A kernel fuzzy C-means clustering-based fuzzy support vector machine algorithm for classification problems with outliers or noises, IEEE Trans. Fuzzy Syst. 19 (1) (2011) 105–115.
[17] Y. Wang, Type-2 fuzzy probabilistic system, in: 2012 9th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), Chongqing, China, IEEE, 2012, pp. 482–486.
[18] S. Greenfield, F. Chiclana, Fuzzy, in: 3-D: Contrasting complex fuzzy sets with type-2 fuzzy sets, in: 2013 Joint IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS), Edmonton, Canada, IEEE, 2013, pp. 1237–1242.
[19] O. Uncu, I. Turksen, Discrete interval type 2 fuzzy system models using uncertainty in learning parameters, IEEE Trans. Fuzzy Syst. 15 (1) (2007) 90–106.
[20] C.-F. Juang, R.-B. Huang, W.-Y. Cheng, An interval type-2 fuzzy-neural network with support-vector regression for noisy regression problems, IEEE Trans. Fuzzy Syst. 18 (4) (2010) 686–699.
[21] Y. Wei, J. Qiu, H.R. Karimi, M. Wang, A new design of h8 filtering for continuous-time Markovian jump systems with time-varying delay and partially accessible mode information, Signal Process. 93 (9) (2013) 2392–2407.
[22] J. Qiu, Y. Wei, H.R. Karimi, New approach to delay-dependent h8 control for continuous-time Markovian jump systems with time-varying delay and deficient transition descriptions, J. Frankl. Inst. 352 (1) (2015) 189–215.
[23] J. Chen, F. Liu, New design approach of robust h8 control for uncertain fuzzy Markovian jump systems, in: Fourth International Conference on Fuzzy Systems and Knowledge Discovery, 2007. FSKD 2007, Haikou, China, vol. 1. IEEE, 2007, pp. 326–330.
[24] Y. Zhang, S. Xu, B. Zhang, Robust output feedback stabilization for uncertain discrete-time fuzzy Markovian jump systems with time-varying delays, IEEE Trans. Fuzzy Syst. 17 (2) (2009) 411–420.
[25] Y. Fenghua, S. Hao, G. LuSheng, C. Zongxiang, Robust resilient controller design for uncertain discrete-time fuzzy Markovian jump delayed systems, in: 2011 30th Chinese Control Conference (CCC), Yantai, China, IEEE, 2011, pp. 2285–2290.
[26] C.-F. Lin, S.-D. Wang, Fuzzy support vector machines, IEEE Trans. Neural Netw. 13 (2) (2002) 464–471.
[27] R.S. Fisher, W.v.E. Boas, W. Blume, C. Elger, P. Genton, P. Lee, J. Engel, Epileptic seizures and epilepsy: definitions proposed by the international league against epilepsy (ILAE) and the international bureau for epilepsy (IBE), Epilepsia 46 (4) (2005) 470–472.
[28] B. Litt, J. Echauz, Prediction of epileptic seizures, Lancet Neurol. 1(May) (2002) 22–30.
[29] O.A. Rosso, A. Mendes, R. Berretta, J.A. Rostas, M. Hunter, P. Moscato, Distinguishing childhood absence epilepsy patients from controls by the analysis of their background brain electrical activity (ii): a combinatorial optimization approach for electrode selection, J. Neurosci. Methods 181 (2) (2009) 257–267.
[30] F. Amor, S. Baillet, V. Navarro, C. Adam, J. Martinerie, M. Le Van Quyen, Cortical local and long-range synchronization interplay in human absence seizure initiation, Neuroimage 45 (3) (2009) 950–962.
[31] C. Wu, X. Wang, D. Bai, H. Zhang, Fast incremental learning algorithm of SVM on KKT conditions, in: Sixth International Conference on Fuzzy Systems and Knowledge Discovery, Yantai, China, vol. 1, IEEE 2009, pp. 551–554.
[32] C.-T. Lin, C.-M. Yeh, S.-F. Liang, J.-F. Chung, N. Kumar, Support-vector-based fuzzy neural network for pattern classification, IEEE Trans. Fuzzy Syst. 14 (1) (2006) 31–41.
[33] J.M. Mendel, Type-2 fuzzy sets and systems: an overview, IEEE Comput. Intell. Mag. 2 (1) (2007) 20–29.
[34] N.N. Karnik, J.M. Mendel, Operations on type-2 fuzzy sets, Fuzzy Sets Syst. 122 (2) (2001) 327–348.
[35] D. Wu, J.M. Mendel, Enhanced Karnik–Mendel algorithms, IEEE Trans. Fuzzy Syst. 17 (4) (2009) 923–934.
[36] A.E. Cavanna, F. Monaco, Brain mechanisms of altered conscious states during epileptic seizures, Nat. Rev. Neurol. 5 (May) (2009) 267–276.
[37] B.D. Killory, X. Bai, M. Negishi, C. Vega, M.N. Spann, M. Vestal, J. Guo, R. Berman, N. Danielson, J. Trejo, et al., Impaired attention and network connectivity in childhood absence epilepsy, Neuroimage 56 (4) (2011) 2209–2217.
[38] A. Gorji, C. Mittag, P. Shahabi, T. Seidenbecher, H.-C. Pape, Seizure-related activity of intralaminar thalamic neurons in a genetic model of absence epilepsy, Neurobiol. Disease 43 (1) (2011) 266–274.
[39] H.K. Meeren, J.P. Pijn, E.L.V. Luijtelaar, A.M. Coenen, F.H.L. da Silva, Cortical focus drives widespread corticothalamic networks during spontaneous absence seizures in rats, J. Neurosci. 22 (February) (2002) 1480–1495.

[40] L.D. Iasemidis, J.C. Sackellares, H.P. Zaveri, W.J. Williams, Phase space topography and the Lyapunov exponent of electrocorticograms in partial seizures, Brain Topogr. 2 (3) (1990) 187–201.

[41] M. Niknazar, S.R. Mousavi, S. Motaghi, A. Dehghani, B.V. Vahdat, M. B. Shamsollahi, M. Sayyah, S.M. Noorbakhsh, A unified approach for detection of induced epileptic seizures in rats using ECoG signals, Epilepsy Behav. 27 (2) (2013) 355–364.

[42] H.K. Lam, U. Ekong, H. Liu, B. Xiao, H. Araujo, S.H. Ling, K.Y. Chan, A study of neural-network-based classifiers for material classification, Neurocomputing 144 (2014) 367–377.

[43] H.K. Lam, U. Ekong, B. Xiao, G. Ouyang, H. Liu, K.Y. Chan, S.H. Ling, Variable weight neural networks and their applications on material surface and epilepsy seizure phase classifications, Neurocomputing 149 (2015) 1177–1187.

[44] I. Güler, E.D. Übeyli, Adaptive neuro-fuzzy inference system for classification of EEG signals using wavelet coefficients, J. Neurosci. Methods 148 (2) (2005) 113–121.

[45] N. Kannathal, M.L. Choo, U.R. Acharya, P.K. Sadasivan, Entropies for detection of epilepsy in EEG, Comput. Methods Programs Biomed. 80 (3) (2005) 187–194.

[46] E.D. Ubeyli, Automatic detection of electroencephalographic changes using adaptive neuro-fuzzy inference system employing Lyapunov exponents, Expert Syst. Appl. 36 (July) (2009) 9031–9038.

[47] M. Thulasidas, C. Guan, J. Wu, Robust classification of eeg signal for brain-computer interface, IEEE Trans. Neural Syst. Rehabil. Eng. 14 (1) (2006) 24–29.

[48] J. O'Keefe, M.L. Recce, Phase relationship between hippocampal place units and the EEG theta rhythm, Hippocampus 3 (3) (1993) 317–330.

[49] V. Jakkula, Tutorial on Support Vector Machine (SVM), School of EECS, Washington State University, Washington, US, 2006.

[50] J.M. Mendel, R.I.B. John, Type-2 fuzzy sets made simple, IEEE Trans. Fuzzy Syst. 10 (2) (2002) 117–127.

[51] D. Whitley, A genetic algorithm tutorial, Stat. Comput. 4 (2) (1994) 65–85.

[52] N. Friedman, D. Geiger, M. Goldszmidt, Bayesian network classifiers, Mach. Learn. 29 (2–3) (1997) 131–163.

[53] I. Tetko, I. Baskin, A. Varnek, Tutorial on Machine Learning. Part 2. Descriptor Selection Bias, 2013. http://dx.doi.org/http://infochim.u-strasbg.fr/CS3/program/Tutorials/Tutorial2b.pdf.

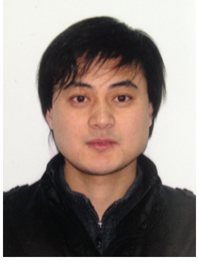
Udeme Essien Ekong received his B.Eng. degree in Electronic Engineering from King's College London in 2011 and is currently in the Penultimate year of his Ph. D. at the Department of Informatics, King's College London. His research interests include neural networks, support vector machines and fuzzy logic.

H.K. Lam received the B.Eng. (Hons.) and Ph.D. degrees from the Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong, in 1995 and 2000, respectively. During the period of 2000 and 2005, he worked with the Department of Electronic and Information Engineering at The Hong Kong Polytechnic University as Post-Doctoral Fellow and Research Fellow respectively. He joined as a Lecturer at King's College London in 2005 and is currently a Reader.

His current research interests include intelligent control systems and computational intelligence. He has served as a program committee member and international advisory board member for various international conferences and a reviewer for various books, international journals and international conferences. He is an associate editor for IEEE Transactions on Fuzzy Systems, IEEE Transactions on Circuits and Systems II: Express Briefs, IET Control Theory and Applications, International Journal of Fuzzy Systems and Neurocomputing; and guest editor for a number of international journals. He is in the editorial board of Journal of Intelligent Learning Systems and Applications, Journal of Applied Mathematics, Mathematical Problems in Engineering, Modelling and Simulation in Engineering, Annual Review of Chaos Theory, Bifurcations and Dynamical System, The Open Cybernetics and Systemics Journal, Cogent Engineering and International Journal of Sensors, Wireless Communications and Control. He is an IEEE senior member.

He is the coeditor for two edited volumes: Control of Chaotic Nonlinear Circuits (World Scientific, 2009) and Computational Intelligence and its Applications (World Scientific, 2012), and the coauthor of the monograph: Stability Analysis of Fuzzy-Model-Based Control Systems (Springer, 2011).

Bo Xiao received bachelor and master (Hons.) degrees in Engineering from the College of Communication Engineering, Chongqing University, PR China, in 2010 and 2013, respectively. He is currently pursuing his Ph. D. degree of Robotics at King's College London. His current research interests include neural networks, support vector machine, fuzzy logic and fuzzy model based control.



Gaoxiang Ouyang received the B.S. in automation and the M.S. in control theory and control engineering both from Yanshan University, China, in 2002 and 2004 respectively, and the Ph.D. degree at the Department of Manufacturing Engineering, City University of Hong Kong in 2010. Dr. Ouyang is an Associate Professor in the School of Brain and Cognitive Sciences at Beijing Normal University, Beijing, China. His research interests are bio-signal analysis, neural engineering, and dynamics system. He has published numerous refereed international journals and conference papers.

Hongbin Liu is currently a lecturer (Assistant Professor) in the Centre for Robotics Research, Department of Informatics, King's College London, UK. He received the B.S. degree in 2005 from the Northwestern Polytechnique University, Xi'an, China, awarded the M.Sc. degree from King's College London in 2006. He received the Ph.D. degree in 2010 from King's College London, UK. He received the Chinese Government Award for Outstanding Students Abroad in 2009. He is a member of IEEE. His research interests include the tactile/force perception based robotic cognition, the modelling of dynamic interaction, soft robotics and medical robotics.

Kit Yan Chan received his Ph.D. degree in Computing from London South Bank University, United Kingdom, in 2006. After his Ph.D. study, he worked as a Postdoctoral Research Fellow in the Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hong Kong, until 2009. Currently, he is a Senior Lecturer in the Department of Electrical and Computer Engineering, Curtin University of Technology, Australia. His research interests include new product development and manufacturing process design.

S.H. Ling received the B.Eng. degree from the Department of Electrical Engineering, M.Phil. and Ph.D. degrees from the Department of Electronic and Information Engineering in the Hong Kong Polytechnic University in 1999, 2002 and 2007 respectively. Currently, he works in the University of Technology, Sydney, Australia as Lecturer. He has authored and coauthored over 130 books, international journal and conference papers on computational Intelligence and its industrial applications. His current research interests include evolution computations, fuzzy logics, neural networks, hybrid systems and biomedical applications. Currently, he serves as Co-Editors-in-Chief for Journal of Intelligent Learning Systems and Applications.