ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 10, October 2015

FUZZY MODEL WITH LEARNING APPROACHES FOR CLASSIFICATION PROBLEM

L.Jegatha Deborah, P.Vijayakumar

University College of Engineering Tindivanam, A Constituent College of Anna University
Chennai,Melpakkam-604 001, India

Abstract: The new Incremental learning approach to endow a Takagi-Sugeno-type Fuzzy classification Model has been done using the learning approaches. Such as Online Incremental linear Support Vector (ISVM) Machine and Margin-selective Gradient Descent learning Algorithm (MGDA). In this learning approach the Fuzzy classification rules is learned by Online Incremental Support Vector Machine and Margin-selective Gradient Descent learning Algorithm. This model initially does not contain any rule set. A one-pass clustering algorithm is used to determine the number of rules and initial fuzzy set in rule antecedent part. An Incremental Support Vector Machine is proposed to tune the consequent part in the fuzzy classification rule with high generalization ability. The use of Online ISVM instead of batch SVM enables this learning model to handle online training problems with only one new sample available at a time. The Margin-selective Gradient Descent learning Algorithm is proposed to tune the rule consequent part. The MGDA is used to prevent overtraining in then learning process. Compare to other learning model this model reduce the test error rate.

Keywords: Fuzzy classifiers (FCs), Incremental learning, Incremental support vector machines(ISVMs), Margin-selective Gradient Descent learning Algorithm.

1. Introduction

In Machine learning the training dates are given to train the machine. The learning scenario can be divided into two modes such as batch learning mode and incremental learning mode. In batch learning mode all data are collected in advanced, and training is performed in one batch. The problem in this mode is that memory may not sufficient to handle all training problem at a time and when all training data cannot be obtained at once, i.e., they are sequentially obtained. This model is computationally waste because it retrained the model from scratch. But in incremental mode learning is obtained sequentially in an online fashion. This learning mode reduce the memory consumption for storing the training data. In this advantage most of the learning models using this mode. Such as neural networks fuzzy systems [1]-[14] and support vector machines (SVMs) [15]-[18].

Several fuzzy rule based classification models are available. These models helps to reduce only the empirical risk (training error). But the error not only occur in training it also depend on the testing process. SVMs are designed via the formation of structural risk minimization and help to find an optimal separating hyperplane that maximizes the margin between two classes in the feature space [19]. Compare to other classifier SVM have the high generalization ability. Depending upon their learning mode SVM is classified in to two types such as Batch SVM and Online

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 10, October 2015

Incremental SVM. In Batch SVM [19], [20] the data are processed sequentially in a batch. This type needs more memory to store the training data. And the process are done from the scratch. In incremental SVM [15]-[18] single data is trained at a time. Two types of incremental SVMs are available such as Online ISVM and Chunk ISVM. In chunk ISVMs, an SVM is trained using one chunk (subset) of training data at a time, and a solution is found using the traditional batch SVMs. Chunk ISVMs[20] improve learning efficiency by discarding some past/new training samples according to their locations relative to the classification hyper plane. However, they are unsuitable for problems where new training samples arrive one by one. In Online ISVMs [18], an SVM is trained with one new training sample at a time. In contrast with traditional batch SVMs, a solution in an online ISVM is constructed recursively. This method shows the advantage of incremental mode so we use this method for learning. One disadvantage of SVMs is that the number of Support Vectors in an SVM is usually large. So the model built whether batch or incremental learning the result of the SVM model is large.

In a fuzzy classification model using SVM for high generalization ability [21]-[28]. For FC design, one common approach is regarding a support vector (SV) [21]as a fuzzy rule, where the number of fuzzy rules is equal to the number of SVs. The number of SVs in an SVM is usually very large, especially for complex classification problems, the designed FC is equally large. To solve this problem, different methods were proposed to remove relatively irrelevant fuzzy rules in the original rule set. This reduced FC may degrade the performance. Another approach using cluster algorithm to generate the fuzzy rule [24]-[27] and small rule set is obtained. A linear batch SVM is then employed to determine the consequent parameters. For the SVM-trained FCs above, the consequent parameters are all learned through batch SVMs, which do not take advantage of ISVMs. A chunk ISVM-trained FC (ISVM-FC) was proposed in [28] for consequent parameter learning. Because of the use of chunk ISVMs, the chunk ISVM-FC is unsuitable for incremental training problems where only one sample is available at a time. In addition, the chunk ISVM-FC and the batch SVM trained FC described earlier do not tune the antecedent parameters to an optimal extent. For these problems, this paper proposes the idea of tuning the FC consequent and antecedent parameters using an online ISVM and margin selective gradient descent algorithm (MGDA), respectively.

Contributions of the Fuzzy Model are twofold. The first one is the introduction of an online linear ISVM to optimize the consequent parameters of FC. The specific use of the online linear ISVM not only improves model generalization performance but makes it possible to train the Fuzzy Model sample by sample as well, which avoids huge data access at a time. The second one is the proposal of the MGDA to tune the antecedent parameters of an FC. The linear ISVM finds a separating hyper plane in the consequent space. The MGDA tunes antecedent parameters using the training samples in the two SV located hyper planes. This operation moving positions of mapped training samples in the consequent space to proper region so that the slack variable values are minimized and hyper plane change direction is considered. It should be emphasized that in addition to the introduction of the MGDA into antecedent parameter learning, this Fuzzy Model differs from the chunk ISVM-FC [43] in using different ISVM for consequent parameter learning. The chunk ISVM-FC using chunk ISVM to update the consequent parameters in a batch model. When a new stream of chunk data is available, the consequent parameters in the chunk ISVM-FC must be retrained from scratch. The use of chunk ISVM is to reduce the total number of data selected from old and new training sample in model retraining. This Fuzzy Model uses online ISVM to update the consequent parameters in an incremental model instead of batch model. As a result, this Fuzzy Model can be online trained with one sample at a

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 10, October 2015

time, while the chunk ISVM-FC is not characterized with this online training ability.

2.Literature Review

Many learning models have been introduced for classification. G. Cauwenberghs and T. Poggio, "Incremental and decremental support vector machine learning," [15]. An on-line recursive algorithm for training support vector machines, one vector at a time, is presented. Adiabatic increments retain the Kuhn- Tucker conditions on all previously seen training data, in a number of steps each computed analytically. The incremental procedure is reversible, and decremental "unlearning" offers an efficient method to exactly evaluate leave-one-out generalization performance. It is computationally efficient scheme for on-line SVM training. Unlearning offers an efficient method to exactly evaluate leave-one-out generalization performance.

Y. Chen and J. Z. Wang, "Support vector learning for fuzzy rule-based classification systems," [21].This paper investigates the connection between fuzzy classifiers and kernel machines, establishes a link between fuzzy rules and kernels, and proposes a learning algorithm for fuzzy classifiers. Fuzzy inference on the IF-part of a fuzzy rule can be viewed as evaluating the kernel function. The kernel function is then proven to be a Mercer kernel if the reference functions meet certain spectral requirement. The corresponding fuzzy classifier is named positive definite fuzzy classifier (PDFC). A PDFC can be built from the given training samples based on a support vector learning approach with the IF-part fuzzy rules given by the support vectors. Since the learning process minimizes an upper bound on the expected risk (expected prediction error) instead of the empirical risk (training error), the resulting PDFC usually has good generalization ability in a high dimensional feature space .Powerful machine learning approach for pattern recognition problems. But the number of Support Vectors (SVs) in SVMs and thereby the number of rules in PDFC is usually high which leads us to a classifier with high test time and the consequent part parameters of fuzzy if-then rules are learned by solving SVMs model which is time.

C. T. Lin, C. M. Yeh, S. F. Liang, J. F. Chung, and N. Kumar, "Support vector based fuzzy neural network for pattern classification," [22]. Support-vector-based fuzzy neural network (SVFNN) is proposed for pattern classification in this paper. The SVFNN combines the superior classification power of support vector machine (SVM) in high dimensional data spaces and the efficient human-like reasoning of FNN in handling uncertainty information. A learning algorithm consisting of three learning phases is developed to construct the SVFNN and train its parameters. In the first phase, the fuzzy rules and membership functions are automatically determined by the clustering principle. In the second phase, the parameters of SVFFNN are calculated by the SVM with the proposed adaptive fuzzy kernel function. In the third phase, the relevant fuzzy rules are selected by the proposed reducing fuzzy rule method. It minimize the empirical risk and expected risk and it reach a good classification performance in the testing phase. This model reduce the rules so it degrades the performance of the original classification model.

C. F. Juang, S. H. Chiu, and S. W. Chang, "A self-organizing TS-type fuzzy network with support vector learning and its application to classification problems," [25]. SOTFN-SV is a T-S-type fuzzy system constructed by hybridizing fuzzy clustering and support vector machine. The proposed face detection method consists of three stages. In the first stage, SOTFN-SV is applied to skin color segmentation. In the second stage, face size and shape filters are employed to exclude some face candidates to reduce the number of false alarms. In the final stage, colors of the eyes, mouth, and face skin regions of the remaining face candidates are used as detection features. An SOTFN-SV colour filter uses these features as

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 10, October 2015

inputs to make a final detection decision. The proposed method has a fast detection speed and detects not only the face, but also its size and orientation. Experimental results verify the efficiency and effectiveness of the proposed face detection method. It is very efficient and effective method for handling a face detection problem.

W. Y. Cheng and C. F. Juang, "An incremental support vector machine trained TS type fuzzy system for on-line classification problems"[28]. The ISVM-FC is a fuzzy system that consists of Takagi–Sugeno (TS)-type fuzzy rules. Structure and parameters in the ISVM-FC are trained incrementally from one subset of training data at a time. It generates all rules according to the distribution of the training data. An incremental linear support vector machine (SVM) is used to tune the resulting rule parameters to give the classifier better generalization performance. Three simulations are conducted to verify the performance of the ISVM-FC. Comparisons with fuzzy classifiers and Gaussian-kernel SVM with batch and incremental learning modes demonstrate that the ISVM-FC improves training and test times, and reduces memory consumption for classifier storage without deteriorating the generalization ability. It avoids the use of large amounts of memory required for storing training data and reduces training time. But they are unsuitable for new training samples arrive one by one.

However, most of the existing systems using Batch SVM for learning process which handling only batch of training samples not handle one training sample at a time. In this paper, we define a solution of handling one sample at a time by using online Incremental SVM.

3. Problem Statement

In the existing Fuzzy Classifier using Batch SVM [19], [20] for learning process. This processes needs more memory space to store the training data. The problem in this mode is that memory may not sufficient to handle all training problem at a time and when all training data cannot be obtained at once. The data are obtained sequentially in batch. This model is computationally waste because it retrained the model from scratch and the model size is large. Because number of support vector equal to number of fuzzy rules. So the complex classification problem the model size is very large.

4.System Frame Work

In proposed work incremental learning mode is used for learning. This type of learning reduces memory consumption for training data storage. Clustering to generate the fuzzy rules. Since a small rule set is obtained. The proposed work aim is fuzzy classification model with high generalization ability. The aim is achieve by using the online incremental linear SVM and Margin-Selective gradient descent learning algorithm is tune the parameter.

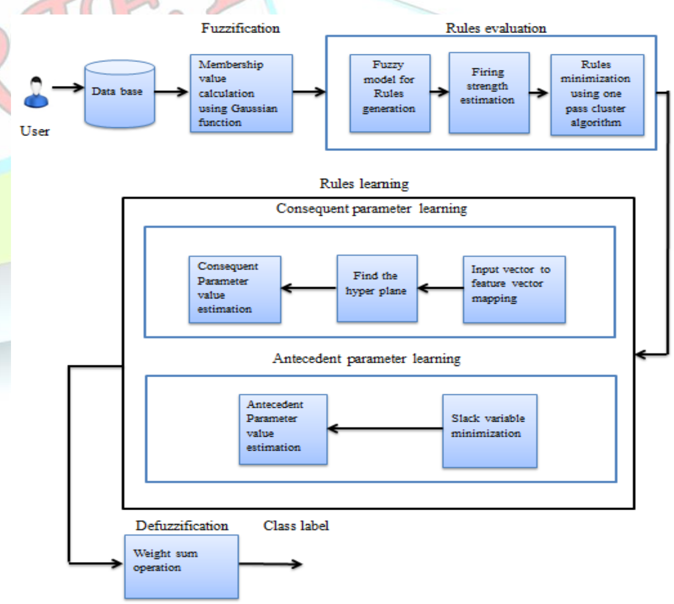

Fig.1 System Architecture

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 10, October 2015

4.1 Fuzzification

Crisp input values are transferred into fuzzy values in the stage of fuzzification. The first step is to take the inputs and determine the membership functions. A membership function (MF) is a curve that defines how each point in the input space is mapped to a membership value (or degree of membership) between 0 and 1. Gaussian functions are popular methods for specifying fuzzy set(A_{ij}). Because curves have the advantage of being smooth and nonzero at all points. The input is always a crisp numerical value limited to the universe of discourse of the input variable and the output is a fuzzy degree of membership in the qualifying linguistic set.

The first step is to take the inputs and determine the membership functions. A membership function (MF) is a curve that defines how each point in the input space is mapped to a membership value (or degree of membership) between 0 and 1. Gaussian functions are popular methods for specifying fuzzy set(A_{ij}). The input is always a crisp numerical value limited to the universe of discourse of the input variable and the output is a fuzzy degree of membership in the qualifying linguistic set.

$$M_{ij}(x_j)=\exp\left\{-\left(x_j-m_{ij}\right)^2 \middle| 2\sigma^2{}_{ji}\right\} \quad (1)$$

where m_{ij} and σ_{ij} denote the center and width of the fuzzy set, respectively. Unlike other SVM-trained FCs [21]–[28], where a common width is used in all fuzzy rules or in all of the fuzzy sets in the same rule, the FM3 uses different and tunable widths, σ_{ij} , for different fuzzy sets to improve classification performance.

4.2 Rules Evaluation

4.2.1 Rules generation

In this stage rules are generated with the help of fuzzy set. The rules are in the form of first order Takagi Sugeno fuzzy rules. The performance of a fuzzy system have first order TS type consequent. A linear combination of input variables plus a constant, has been shown to outperform that with zero-order TS-type consequent. Classification rule is the following form.

If $x_1$is A_{i1} and ….. and x_n is A_{in} then $\hat{y}' = a_{i0} + \sum_{j=1}^{n} a_{ij}\, x_j$ (2)

where $x_1, \ldots, x_n$ are inputs, A_{ij} is a fuzzy set, and a_{ij} is a real number. Fuzzy set A_{ij} is employed with Gaussian membership function.

4.2.2 Firing strength estimation

Each rule have the degree value. That rule value is a strength of that corresponding rules. Dates are passed into the rules and which have the maximum firing values that rule is a corresponding rule to that data. The t-norm operation is performed using the algebraic product operation, and the rule firing strength $\mu_i(x)$ is determined by

$$\mu_i(x)=\prod_{j=1}^{n} M_{ij}(x_j)=\exp\left\{-\sum_{j=1}^{n}\left(\left(x_j-m_{ij}\right)^2 \middle| 2\sigma^2{}_{ji}\right)\right\} \quad (3)$$

where x= [x_1,.., x_n]. Suppose that there are *r* rules. For the output calculation, the simple weight sum operation is employed, and the output can be written as

$$y' = \sum_{i=1}^{r} \mu_i\left(+\sum_{j=o}^{n} a_{ij}\, x_j\right) + b \quad (4)$$

The bias term *b* is added to the output to account for the learning in a linear SVM.

4.2.3 Rule minimization

Rule minimization is done by using the one pass cluster algorithm. Depending upon the rule firing strength number of rules is minimized. In this algorithm have predefined threshold value. This threshold value is compare with rule firing strength .If the rule firing strength is smaller than a predefined threshold then new cluster (rule) generated. If the rule firing strength is greater than a predefined threshold then no new cluster (rule) is generated.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 10, October 2015

4.3 Rules Learning

Two types of learning methods are used to learn the rules such as

- Consequent parameter learning (ISVM)
- Antecedent parameter learning (MGDA)

4.3.1Consequent parameter learning

Using Online Incremental Support Vector Machine(ISVM) learning algorithm for consequent parameter learning. This learning method train the system only one sample available at a time. In SVM the data in a input space is mapped into feature space for high dimension. The separate function is used to calculate the hyper plane which correctly separate the data into a class. And the Consequent of the rule is updated.

Consequent Parameter Learning by Linear Incremental Support Vector Machine

This Fuzzy Model uses the online linear ISVM to tune all of these free parameters $\boldsymbol{a_{ij}}$ in the consequent part. The output y´ in (4) can be written as follows:

$$y' = a^T \emptyset + b \quad (5)$$

Where

$$a = [a_{10} \dots \dots a_{1n} \dots\dots a_{r0} \dots . a_{rn}]^T \in R^{r(n+1)} \quad (6)$$

and

$$\emptyset = [\mu_1 x_0 \dots \dots \mu_1 x_n \dots\dots \mu_r x_0 \dots . \mu_r x_n]^T \in R^{r(n+1)} \quad (7)$$

Equation (5) shows that the consequent vector a in the T–S type fuzzy model functions as a linear combination coefficient vector for the feature vector. The online linear ISVM determines only the consequent parameter vector a and does not change the structure of the T–S-type fuzzy model. The function of the model after learning is still the same as in (5), i.e., remains a T–S-type fuzzy model. This novel learning technique is different from previous nonlinear Mercer kernel SVM-trained fuzzy models [21]–[23], where both model structure and parameters are determined by SVMs. In the online linear ISVM-trained T–S-fuzzy model, each input datum $\boldsymbol{x_k}$= [$\boldsymbol{x_{k0}}$,...., $\boldsymbol{x_{kn}}$] is transformed to the feature Vector $\emptyset_k$ via (7), where

$$\emptyset_k = [\mu_1 x_{k0} \dots \dots \mu_1 x_{kn} \dots\dots \mu_r x_{k0} \dots . \mu_r x_{kn}]^T \in R^{r(n+1)} \quad (8)$$

The softly constrained optimization problem in the feature space is given as follows:

$$Min_{a,b,\varepsilon} \ \frac{1}{2} a^T a + C \sum_{i=1}^{N} \xi_i$$

$$subject\ to\ y_i(a^T \phi_i + b) \geq 1 - \xi_i, \xi_i \geq 0, i = 1 \dots . N. \quad (9)$$

where the cost function 0.5$a^T a$+ $C \sum_{i=1}^{N} \xi_i$ i=1$\xi_i$is called a soft margin, C is a user-defined positive parameter, $\xi_i$is a slack variable, and ξ_i is an upper bound on the number of training errors. In an online ISVM, the dual form is expressed as follows [15][18].

$$L = \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i Q_{ij} \alpha_j - \sum_{i=1}^{N} \alpha_i + b \sum_{i=1}^{N} \alpha_i y_i \quad (10)$$

Where $Q_{ij} = y_i y_j < x_i, x_j >$ is a symmetric positive-definite kernel matrix. The Karush–Kuhn–Tucker (KKT) conditions in (10) lead to the following necessary conditions:

$$g_i = \frac{\partial L}{\partial \alpha_i} = \sum_{j=1}^{N} Q_{ij} \alpha_j + y_i b - 1$$

$$x \begin{cases} \geq 0, & if\ \alpha_i = 0 \\ = 0, & if\ 0 < \alpha_i < C \\ \leq 0, & if\ \alpha_i = C \end{cases} \quad (11)$$

$$h = \frac{\partial L}{\partial b} = \sum_{j=1}^{N} y_j \alpha_j \quad (12)$$

According to (11), the training dataset D^Ncan be divided into three categories: The set S that denotes unbounded SVs (0<$\alpha_i < C$), the set E that denotes bounded SVs (α_i= C), and the set O that denotes non-SVs (α_i= 0).

The online ISVM is trained with one sample at a time, and a new solution is built by using an

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 10, October 2015

existing optimal solution. The coefficient solution $\{\alpha_i$, b}, i = 1, . . . , N, has been found for the old dataset D^N. The old solution is updated when a new training sample (x_{N+1}, y_{N+1}) is available. The principle of coefficient update is that the new coefficients $\{\alpha_i+\Delta\alpha_i, \Delta\alpha_{N+1}$,b}, i= 1,…,N at the incremental step should also satisfy the KTT conditions in (11) and (12) for the enlarged dataset $D^{(N+1)} = D^{(N)} U\{x_{N+1}\}$. The difference of the KKT conditions for the old and new datasets $D^{(N)}$ and $D^{(N+1)}$ is

$\Delta g_i = Q_{jN+1}\Delta$

$$\alpha_{N+1} + \sum_{j\in s} Q_{ij}\,\Delta\alpha_j + y_i\Delta b \;\; \forall i \in D^{(N+1)} \qquad (13)$$

$$0= y_{N+1}\Delta\alpha_{N+1} + \sum_{j=1}^{N} y_j\,\Delta\alpha_j. \qquad (14)$$

Equations (13) and (14) show that any change in $\Delta\alpha N + 1$ accompanies updates in $\Delta\alpha_j, j \in S$, and Δb to hold the equilibrium.

$$\begin{pmatrix} \Delta b \\ \Delta\alpha_s \end{pmatrix} = -\begin{pmatrix} 0 & y_s^T \\ y_s & Q_{SS} \end{pmatrix}^{-1} \begin{pmatrix} y_{N+1} \\ Q_{SN+1} \end{pmatrix} \Delta\alpha_{N+1} \qquad (15)$$

$$\Delta\alpha_{N+1} = -\frac{\sum_{j\in s} Q_{cj}\alpha_i + y_c b - 1}{Q_{cc} + \sum_{j\in s} Q_{cj}\,\beta_j + y_c\,\beta} \qquad (16)$$

By substituting the $\Delta\alpha_{N+1}$ found into (16), the coefficients $\Delta\alpha_i, i = 1, \ldots, N$ and Δb are obtained, and the new coefficients $\alpha_i^{(N+1)}$ and $b^{(N+1)}$ after the incremental learning step are

$$\alpha_i^{(N+1)} = \alpha_i + \Delta\alpha_i, \quad i=1,\ldots,N+1 \qquad (17)$$

$$b^{(N+1)} = b + \Delta b \qquad (18)$$

The use of new coefficient gives the new consequent parameter as follows

$$a_{ij}^{N+1} = a_{ij} + \sum_{k=1}^{N+1} y_k \Delta\alpha_k \mu_i x_{kj} \qquad (19)$$

Where

$$a_{ij} = \sum_{k\in SV} y_k \alpha_k \mu_i x_{kj} \qquad (20)$$

The recursive update in (19) makes it feasible to train the Fuzzy Model with one sample at a time to minimize the soft margin in (9). The new bias value $b^{(N+1)}$ in (4) is updated as in (18). Minimization of the soft margin in (9) helps to reduce the bound on the generalization error in the feature space and, therefore, improves generalization ability.

4.3.2 Antecedent parameter learning

In antecedent parameter learning we are using Margin-Selective Gradient Descent Learning Algorithm (MGDA).This algorithm calculate the error rate for each data. Depending upon the error rate the margin will be adjusted. If the error rate is minimized then the width of the margin is maximized. The MGDA update the antecedent parameter of the rule.

Margin-Selective Gradient Descent Algorithm for Antecedent Parameter Learning

This section introduces the MGDA for tuning antecedent parameters $\boldsymbol{m_{ij}}$ and $\boldsymbol{\sigma_{ij}}$. The typical gradient descent (GD) algorithm used in neural FCs minimizes output errors of all training samples, which may cause overtraining and degrade the generalization performance. The motivation of using the MGDA is to avoid overtraining and increase the separation margin width by learning only with the training samples located in the margin-selective regions. The slack variable ξ_k of the soft margin in (7) after the online linear ISVM learning is given as follows:

$$\xi_k = \begin{cases} 0, & y_k y`_k \geq 1 \\ 1 - y_k y`_k & otherwise \end{cases} \qquad (21)$$

The objective of the MGDA is to minimize ξ_k of the input samples located in margin-selective regions in the feature space using the following objective function:

$$e_k = \frac{1}{2}\xi^2{}_k = \frac{1}{2}(y`_k - y_k)^2, \quad y_k y`_k < 1 \qquad (22)$$

MGDA is only applied to training samples located in the margin-selective region satisfying the constraint $y_k y`_k < 1$. For an input sample x_k in

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 10, October 2015

the margin-selective region, parameters m_{ij} and σ_{ij} are updated as follows:

$$m_{ij}(k+1)= m_{ij}(k)-\eta\frac{\partial\varepsilon_k}{\partial m_{ij}}, \qquad i=1,...,r,\ j=1,...,n \qquad (23)$$

Where

$$\frac{\partial\varepsilon_k}{\partial m_{ij}} =(y`_k - y_k)\left(\sum_{j=0}^{n} a_{ij}\, x_j\right)\mu_i \frac{2(x_j-m_{ij})}{\sigma^2{}_{ji}} \qquad (24)$$

and

$$\sigma_{ij}(k+1)= m_{ij}(k)-\eta\frac{\partial\varepsilon_k}{\partial \sigma_{ij}}, \qquad i=1,...,r,\ j=1,...,n \qquad (25)$$

Where

$$\frac{\partial\varepsilon_k}{\partial \sigma_{ij}} =(y`_k - y_k)\left(\sum_{j=0}^{n} a_{ij}\, x_j\right)\mu_i \frac{2(x_j-m_{ij})^2}{\sigma^3{}_{ji}} \qquad (26)$$

The linear online ISVM is first applied to incrementally learn the optimal FM3 consequent parameters for the whole dataset. The MGDA is then applied to incrementally learn the antecedent parameters for the whole dataset. After the MGDA, locations of all training samples in the feature space change. A new optimal separating hyper plane using the linear online ISVM is found for better classification performance.

4.4 Defuzzification

The process of deffuzzification is to convert the fuzzy values into the crisp values. The output calculation, the simple weight sum operation is employed, and the output can be written as

$$y' = \sum_{i=1}^{r} \mu_i \left(+\sum_{j=o}^{n} a_{ij}\, x_j\right) + b \qquad (27)$$

The bias term *b* is added to the output to account for the learning in a linear SVM. The final decision function is sign (y´) to determine if a given pattern belongs to class “1” or “ - 1.”

5. Performance Evaluation and Result Discussions

5.1 Experiment Methodology

The proposed frame work is tested for various classification problems available in the UCI repository [http://archive.ics.uci.edu/mll]. The frame work is tested for liver disorder dataset. In the repository the underlying numeric format are fuzzified to convert the numeric into fuzzy format. The fuzzy format are then transformed into rules format using the rule generation. Then minimize the rule using the rule minimization. Learning the consequent and antecedent parameter. The outcome of the defuzzification is crisp format.

5.2 Performance Analysis

In this study, we have taken liver disorder as our web application. This dataset analyzes some liver disorders using four attributes. The task is to select if a given male individual suffers from alcoholism.

Table.1 Error rate minimization

Threshold	SONFIN Error rate	Batch SVM Error rate	ISVM-FC Error rate	**FM^3** Error rate
0.2	0.379	0.33	0.270	0.274
0.4	0.372	0.322	0.278	0.268
0.5	0.368	0.313	0.269	0.257
0.7	0.357	0.303	0.258	0.245
0.8	0.346	0.299	0.240	0.230
1	0.339	0.287	0.235	0.225

The proposed method is implemented in JAVA. we have analyzed the cost function, learning type, structure learning, antecedent parameter, consequent parameter, characteristic with existing approaches is show in the following section.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 10, October 2015

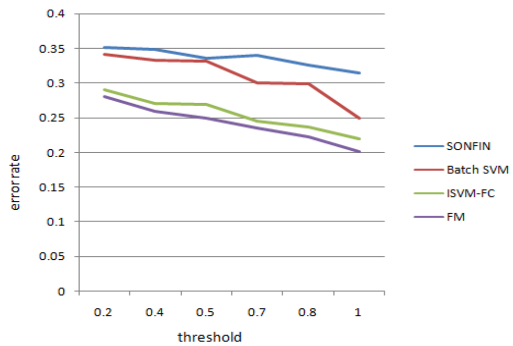

Fig.2 Error rate minimization

The above graph represent the error rate and threshold of the learning based on the result computed using SONFIN, MDSOFN, ISVM-FC. From the graphical results it is evident that fuzzy model provides the minimum error rate based on the threshold.

5.3 Comparison with other models

The proposed method is implemented in JAVA. we have analyzed the cost function, learning type, structure learning, antecedent parameter, consequent parameter, characteristic with existing approaches is show in the following section.

Table .2 Comparison with other models - SONFIN, MDSOFN, ISVM-FC, and FM^3

Models	SONFIN	MDSOFN	ISVM-FC	**FM^3**
Cost function	Empirical risk minimization	Empirical risk minimization	Structural risk minimization in (25)	Structural risk minimization in(25)
Learning	Online incremental	Online incremental	Chunk incremental	Online incremental
Structure learning	Rule growing	Rule growing	Rule growing	Rule growing
Antecedent parameter	Learning via GD algorithm	Learning via GD algorithm	Fixed	Learning via MGDA
Consequent parameter	Learning via GD algorithm	Learning via GD algorithm	Learning via Chunk linear ISVM	Learning via online linear ISVM
Characteristic	Consequent and antecedent parameter learning to minimize training error	Consideration between and within class distances in consequent parameter learning	Consequent parameter learning to minimize test error bound	Consequent and antecedent parameter learning to minimize test error bound

Conclusion

This paper proposes an FM3 to design a T–S-type fuzzy model for classification problems. The rule number in the FM3 is determined by clusters instead of SVs so that a small model is achieved. Parameter tuning using the ISVM minimizes a soft margin instead of training errors, which helps to increase the generalization ability. In addition, the use of online ISVM enables to handle the online classification problems where only one sample is available at a time. This online learning ability also avoids the preload of the huge amount of training data required for batch training

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 10, October 2015

algorithms. The incorporation of MDGA for antecedent parameters tuning moves selected training samples in the feature mapped space to a desired location. Experimental results have verified the effectiveness of the MDGA. The advantages of the FM3 were verified through comparisons with different fuzzy classification models and Gaussian-kernel-based online ISVM. In addition to the classification performance improvement studied in this paper, rule interpretability improvement in the FM3 will be studied in the future.

References

[1] C. F. Juang and C. T. Lin, "An on-line self-constructing neural fuzzy inference network and its applications," IEEE Trans. Fuzzy Syst., vol. 6, no. 1, pp. 12–32, Feb. 1998.
[2] N. E. Mitrakis, J. B. Theocharis, and V. Petridis, "A multi layered neuro fuzzy classifier with self-organizing properties," Fuzzy Sets Syst., vol. 159, no. 23, pp. 3132–3159, 2008.
[3] E. D. Lughofer, "FLEXFIS: A robust incremental learning approach for evolving Takagi–Sugeno fuzzy models," IEEE Trans. Fuzzy Syst., vol. 16, no. 6, pp. 1393–1410, Dec. 2008.
[4] C. F. Juang and Y. W. Tsao, "A self-evolving interval type-2 fuzzy neural network with on-line structure and parameter learning," IEEE Trans. Fuzzy Syst., vol. 16, no. 6, pp. 1411–1424, Dec. 2008.
[5] P. P. Angelov and X. Zhou, "Evolving fuzzy-rule-based classifiers from data streams," IEEE Trans. Fuzzy Syst., vol. 16, no. 6, pp. 1462–1475, Dec. 2008.
[6] J. C. Huhn and E. Hullermeier, "FR3: A fuzzy rule learner for inducing reliable classifiers," IEEE Trans. Fuzzy Syst., vol. 17, no. 1, pp. 138–149, Feb. 2009.
[7] G. D. Wu and P. H. Huang, "A maximizing-discriminability-based self-organizing fuzzy network for classification problems," IEEE Trans. Fuzzy Syst., vol. 18, no. 2, pp. 362–373, Apr. 2010.
[8] A. Lemos, W. Caminhas, and F. Gomide, "Multivariable Gaussian evolving fuzzy modeling system," IEEE Trans. Fuzzy Syst., vol. 19, no. 1, pp. 91–104, Feb. 2011.
[9] H. J. Song, C. Y. Miao, R. Wuyts, Z. Q. Shen, M. D'Hondt, and F. Catthoor, "An extension to fuzzy cognitive maps for classification and prediction," IEEE Trans. Fuzzy Syst., vol. 19, no. 1, pp. 116–135, Feb. 2011.
[10] C. S. Chen, "Robust self-organizing neural-fuzzy control with uncertainty observer for MIMO nonlinear systems," IEEE Trans. Fuzzy Syst., vol. 19, no. 4, pp. 694–706, Aug. 2011.
[11] Z. Chen, S. Aghakhani, J. Man, and S. Dick, "ANCFIS: A neuro fuzzy architecture employing complex fuzzy sets," IEEE Trans. Fuzzy Syst., vol. 19, no. 2, pp. 305–322, Apr. 2011.
[12] G. D. Wu, Z. W. Zhu, and P. H. Huang, "A TS-Type maximizing discriminability- based recurrent fuzzy network for classification problems," IEEE Trans. Fuzzy Syst., vol. 19, no. 2, pp. 339–352, Apr. 2011.
[13] C. F. Juang, T. C. Chen, and W. Y. Cheng, "Speedup of implementing fuzzy neural networks with high-dimensional inputs through parallel processing on graphic processing units," IEEE Trans. Fuzzy Syst., vol. 19, no. 4, pp. 717–728, Aug. 2011.
[14] Y. C. Chen, N. R. Pal, and I. F. Chung, "An integrated mechanism for feature selection and fuzzy rule extraction for classification," IEEE Trans. Fuzzy Syst., vol. 20, no. 4, pp. 683–698, Aug. 2012.
[15] G. Cauwenberghs and T. Poggio, "Incremental and decremental support vector machine learning," in Advances in Neural Information Processing Systems, T. K. Leen, T. G. Dietterich, and V. Tresp, Eds. Cambridge, MA, USA: MIT Press, 2000, vol. 13, pp. 409–415.
[16] C. P. Diehl and G. Cauwenberghs, "SVM incremental learning, adaptation and optimization," in Proc. Int. Joint Conf. Neural Netw., 2003, pp. 2685– 2690.
[17] A. Shilton, M. Palaniswami, D. Ralph, and A. C. Tsoi, "Incremental training of support vector machines," IEEE Trans. Neural Netw., vol. 16, no. 1, pp. 114 131, Jan. 2005.

[18] P. Laskov, C. Gehl, S. Kruger, and K. R. Muller, "Incremental support vector learning: Analysis, implementation and applications," Mach. Learn. Res., vol. 7, pp. 1909–1936, 2006.

[19] N. Cristianini and J. S.-Taylor, An Introduction to Support Vector Machines And Other Kernel-based Learning Methods. Cambridge, U.K.: Cambridge Univ. Press, 2000.

[20] P. Mitra, C. A. Murthy, and S. K. Pal, "Data condensation in large databases by incremental learning with support vector machines," in Proc. Int. Conf. Pattern Recog., 2000, pp. 708–711.

[21] Y. Chen and J. Z. Wang, "Support vector learning for fuzzy rule-based classification systems," IEEE Trans. Fuzzy Syst., vol. 11, no. 6, pp. 716– 728, Dec. 2003.

[22] C. T. Lin, C. M. Yeh, S. F. Liang, J. F. Chung, and N. Kumar, "Support vector- based fuzzy neural network for pattern classification," IEEE Trans. Fuzzy Syst., vol. 14, no. 1, pp. 31–41, Feb. 2006.

[23] S. M. Zhou and J. Q. Gan, "Constructing L2-SVM-based fuzzy classifiers in high-dimensional space with automatic model selection and fuzzy rule ranking," IEEE Trans. Fuzzy Syst., vol. 15, no. 3, pp. 398–409, Jun. 2007.

[24] C. F. Juang, S. H. Shiu, and S. J. Shiu, "Fuzzy system learned through fuzzy clustering and support vector machine for human skin color segmentation," IEEE Trans. Syst., Man, Cybern. A, Syst., Humans, vol. 37, no. 6, pp. 1077–1087, Nov. 2007.

[25] C. F. Juang, S. H. Chiu, and S. W. Chang, "A self-organizing TS-type fuzzy network with support vector learning and its application to classification problems," IEEE Trans. Fuzzy Syst., vol. 15, no. 5, pp. 998–1008, Oct. 2007.

[26] C. F. Juang and G. C. Chen, "A TS fuzzy system learned through a support vector machine in principal component space for real-time object detection," IEEE Trans. Ind. Electron., vol. 59, no. 8, pp. 3309–3320, Aug. 2012.

[27] G. C. Chen and C. F. Juang, "Object detection using color entropies and a fuzzy classifier," IEEE Comput. Intell. Mag., vol. 8, no. 1, pp. 33–45, Feb. 2013.

[28] W. Y. Cheng and C. F. Juang, "An incremental support vector machine trained TS-type fuzzy system for on-line classification problems," Fuzzy Sets Syst., vol. 163, no. 1, pp. 24–44, Jan. 2011.

[29] Wei-Yuan Cheng and Chia-Feng Juang," A Fuzzy model with Online Incremental SVM and Margin-Selective Gradient Descent Learning for Classification problems" Fuzzy Sets Syst., vol. 22, issue no. 2, year April 2014.