IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 2 August 2022, accepted 8 September 2022, date of publication 16 September 2022, date of current version 29 September 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3207480

RESEARCH ARTICLE

Fuzzy and SVM Based Classification Model to Classify Spectral Objects in Sloan Digital Sky

ARODH LAL KARN[1], CARLOS ANDRÉS TAVERA ROMERO[2], (Member, IEEE), SUDHAKAR SENGAN[3], (Member, IEEE), ABOLFAZL MEHBODNIYA[4], (Senior Member, IEEE), JULIAN L. WEBBER[4], (Senior Member, IEEE), DENIS A. PUSTOKHIN[5], AND FRANK-DETLEF WENDE[6]

[1]Department of Financial and Actuarial Mathematics, School of Mathematics and Physics, Xian Jiaotong-Liverpool University, Suzhou 215123, China
[2]COMBA R&D Laboratory, Faculty of Engineering, Universidad Santiago de Cali, Cali 76001, Colombia
[3]Department of Computer Science and Engineering, PSN College of Engineering and Technology, Tirunelveli 627152, India
[4]Department of Electronics and Communications Engineering, Kuwait College of Science and Technology, Kuwait City, Kuwait
[5]Department of Logistics, State University of Management, 109542 Moscow, Russia
[6]Department of Logistics and Marketing, Faculty of Economics and Business, Financial University under the Government of the Russian Federation, 125993 Moscow, Russia

Corresponding author: Sudhakar Sengan (sudhasengan@gmail.com)

This work was supported by the Dirección General de Investigaciones at Universidad Santiago de Cali under Grant 01-2022.

ABSTRACT The Sloan Digital Sky Survey (SDSS) comprises about one billion objects classified spectrometrically. Because astronomical datasets are so enormous, manually classifying them is nearly impossible—a huge dataset results in class imbalance and overfitting. We recommend a framework in this research study that overcomes these constraints. The framework uses a hybrid Synthetic Minority Oversampling Technique + Edited Nearest Neighbor (SMOTE + ENN) balancer. The balanced dataset is then used to extract features via a non-linear algorithm using Kernel Principal Component Analysis (KPCA). The features are then passed into the proposed Int-T2-Fuzzy Support Vector Machine classifier, which uses a modified type reducer and inference engine to achieve more precise categorization. Using the Sloan Digital Sky Survey dataset and a number of evaluation metrics, the SMOTE+ENN model's performance is measured. The research shows that the model does a good job.

INDEX TERMS Sloan digital sky, fuzzy logic, fuzzy control, support vector machine, nearest neighbor, machine learning, astronomical, kernel principal component analysis.

I. INTRODUCTION

Astronomy has recently seen significant advances in detectors, instruments, telescopes, and even probes launched into outer space and distant planets to collect data for sky surveys to map our universe. Data-oriented astronomy refers to the organization of acquired data into very large datasets. These astronomical datasets are in distinct forms, including light curves, optical and infrared spectra, image data, and photometric redshifts, representing a wide range of astronomical data and objects. Astronomers begin the categorization process by carefully scanning the dataset and categorising them into likely quasars, stars, and galaxies. Transients like asteroids, gamma rays, and supernovae that appear for a concise volume of time in space can also be found in imaging data. Many challenges arise when processing these data with a large number of bands, including image calibration noises, spatial distortion, and restricted or unbalanced labelled training samples, i.e., Hughes phenomenon and dimensionality reduction-related artefacts such as overfitting, redundancy, spectral variability, loss of significant features between the channels, and so on.

Significant efforts are invested in investigating the idea of applying Machine Learning (ML) techniques that automate the knowledge discovery process and astronomical information extraction within these massive unprocessed datasets,

The associate editor coordinating the review of this manuscript and approving it for publication was Mouloud Denai.

which could probably be a definite solution to the challenges mentioned earlier. Astro-informatics, a new study in astronomical data science, helps determine the astronomical knowledge and information from these vast raw databases. These tools help automate astronomical data's scanning process using cutting-edge data mining techniques, data science tools [1], and statistical methods. Initial efforts were made in 2010 by the National Research Council-USA. It laid the groundwork for future scientific contributions [2], [3], [4] to the field in which it was centered. It enriched it by leveraging vast, globally dispersed digital astronomy database collections such as the United States Naval Observatory Astrometric (USNO-A2), Digitized Palomar Observatory Sky Survey (DPOSS), Square Kilometer Array Observatory (SKAO), Sloan Digital Sky Survey (SDSS) and new initiatives such as Large Synoptic Survey Telescope (LSST) and Visible and Infrared Survey Telescope (VISTA). However, the area is immobile in its early phases. More case studies are needed to develop new accurate methodologies, particularly in ML and, in particular, Deep Learning (DL), in addition to the availability of large datasets from astronomy that are now freely accessible to the public. The classification problem must be solved in order to map our universe, better understand it correctly, and support existing and emerging cosmological theories. DL is the ideal candidate technique, as it has demonstrated its ability to work with large image databases like the one in our domain.

When certain classes have considerably more examples than others, the problem of class imbalance arises. Classifiers perform poorly on unbalanced datasets because they extrapolate from sample data and produce an essential hypothesis that best matches the facts [5]. By practice, in a binary classification problem, the minority examples' <Class Label> is positive, and the majority <Class Label> is negative. Yet, when dealing with unbalanced data sets, the most straightforward theory frequently categorizes nearly all occurrences as negative. Hence, biased classifiers have a high level of predictability for the negative class, but their prediction accuracy is weak for the positive class. Minority case classification is usually a challenge in various disciplines, including fraud detection, the discovery of network intrusion, web-based research, medical evaluation, text classification, and the categorization of astronomical objects. Various methods have been offered to address the problem of class imbalance. Such methods are grouped as internal [6], [7] and external [8], [9]. In the internal approach, the imbalance problem is addressed by changing/developing new algorithms for learning. In an external system, resample the original data collection by over-sampling or under-sampling the minority or majority class, respectively, to create a balanced set of data that allows those classifiers to work better on the minority class.

Oversampling techniques are preferred over undersampling methods in most circumstances. It is because, during undersampling, we likely exclude occurrences that may contain crucial information. Researchers in recent years have suggested many different classification approaches. Two methods for automatic classification of star spectra, such as $\chi 2$-minimization and Artificial Neural Network (ANN), are proposed [10]. Singh *et al.* [11] describes a rapid and reliable method for identifying an optical stellar spectrum library ranging from O-to M-type stars. To automate the classification process, the technique uses two tools: (a) Principle Component Analysis (PCA) to reduce the data dimensionality; and (b) a Multilayer Back Propagation Network (MBPN) that relies on ANN for automation of the classification process.

The ANN method using a backpropagation based supervised learning algorithm was used to categorise Calgary's Infrared Astronomical Satellite (IRAS) spectra in the area of 8 μm to 23 $\mu m,$ which contains 2000 bright sources [12]. Bora *et al.* [13] uses an ANN for star classification. The training set used is synthetic spectra in the ultraviolet (UV) area range of 1250–3220Å and the International Ultraviolet Explorer (IUE) of the low-resolution test set. Bazarghan and Gupta [14] proposed a Probabilistic Neural Network (PNN), which automatically classified approximately 5000 SDSS spectrum into nearly 158 reference library spectral types ranging from O- to M-type stars.

The Support Vector Machine (SVM) is the most common classification method used in ML and data mining. Scientists focus more attention on SVM and suggest several new improvements. For multitask learning, the proximal SVM is used [15]. Datta and Das [16] presents the Near-Bayesian Support Vector Machine (NBSVM) to handle the problem of unbalanced classification. Liu *et al.* [17] proposed Ramploss Non-Parallel SVM (RNPSVM), a nonparallel hyperplane that is sparse and resilient [18]. Nonparallel SVM [NPSVM], a new non-parallel classifier, is proposed. SVM can also be widely used in astronomical research, particularly in the field of automatic spectral categorization. SVM is employed to categorise spectra using the dimension reduction approach PCA [19]. In another method, ISOMAP was used to reduce the number of dimensions, and SVM was used to classify star spectra [20], [21].

ML is used extensively in cosmology and also in astrophysics [22]. A non-exhaustive list of applications includes (i) supernova photometric classification [23], [24], (ii) gravitational wave research [25], (iii) photometrical redshift [26], (iv) galaxy morphology [27], and (v) atmospheric parameter determination for stellar sources [28]. Many surveys have successfully used ML applications to separate stars and galaxies. For instance, they classified the Sloan Digital Sky Survey's SDSS sources using multiple tree approaches. Whitten *et al.* [29] used data from the Canada-France-Hawaii Telescope Lensing Survey (CFHTLenS) to train classifiers that combined supervised and unsupervised ML methods. Convolutional Neural Networks (CNN) have recently been used with images as input to attain an Area Under the Curve (AUC) $>$ 0.99 for Canada-France-Hawaii Telescope Lensing Survey data and SDSS [30]. And others have published numerous ML methods in the context of star and Galaxy categorization [31], [32].

At the close of the past decade, a Type-2 Fuzzy Logic System (T2FLS) was suggested that employs nearly two fuzzy Membership Functions (MFs), which increases the capability to handle language uncertainty representation [33], [34]. T2FS and T2FLS are used in the following applications: (i) control and system modelling [35], [36], (ii) robots and motion control [37], [38], and (iii) image processing [39], [40]. When the systems are subjected to various uncertainties, T2FLS outperforms standard Type-1 fuzzy systems. Later, an interval T2FLS was developed to reduce computational complexity, and the notion of an Interval Type-2 Fuzzy Kernel, known as IT2FK, was not employed during the prior SVM approach. Such things make it more likely that IT2FK-SVM will be used in this research to put astronomical objects into groups.

Int-type-2 fuzzy sets, a subset of type-2 fuzzy sets, are widely used in practice due to their low computing cost and ease of implementation. The researchers demonstrated that Int-type-2 fuzzy ideas handle uncertainties better than type-1 fuzzy techniques. When determining the exact membership functions of the fuzzy sets utilised, interval type-2 fuzzy sets are beneficial since they provide more robust generalisations. This is why we used the Interval type-2 fuzzy model for our study.

In this research paper, Int-T2-FSVM, an interval type 2 Fuzzy SVM model, is proposed to classify astronomical objects such as stars, quasars, and galaxies. The work employs the SDSS dataset to study the model's efficacy. To avoid the class imbalance issue in the dataset, the model employs a SMOTE + ENN hybrid balancer. Then the features are extracted from the balanced dataset using Kernel Principal Component Analysis (KPCA) non-linear feature extraction. The parts are then fed into the proposed Int-T2-FSVM classifier, and the model employs a modified type reducer and inference engine to generate a more accurate classification. The performance of the model is tested using the SDSS dataset and different evaluation metrics. The results show that the performance of the model is good.

The paper is structured in such a way that Section 2 discusses fundamental technologies utilised to build the categorization framework suggested throughout this work, Section 3 discusses the methods involved, Section 4 discusses the proposed model's performance evaluation, and Section 5 ends the work.

II. PRELIMINARIES

A. SYNTHETIC MINORITY OVERSAMPLING TECHNIQUE (SMOTE)

SMOTE is an oversampling method that generates synthetic samples on behalf of the minority class. The approach aids in overcoming the "overfit" issue caused by random oversampling. The method focuses on all of the features to make examples by combining positive and closer examples (see Fig. 1).

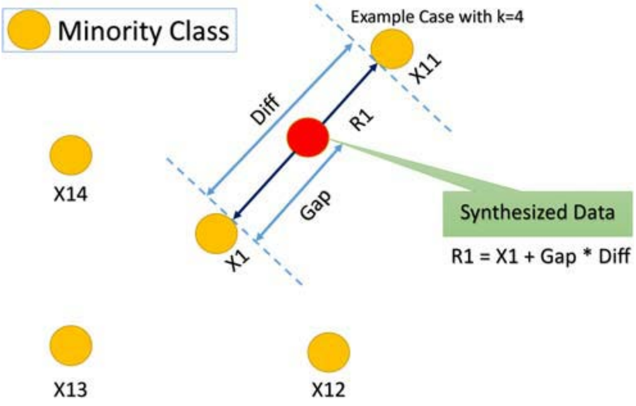

FIGURE 1. Synthetic minority oversampling technique.

1) PROCEDURE OF SMOTE

First, the total number of observations for oversampling N is determined. In general, it is selected so that the distribution of binary classes is 1:1; however, it can scale back. Iteration starts by choosing a random class instance that is positive. Then, the k-Nearest Neighbor (KNN)'s value (by default 5) is obtained for such cases. Lastly, N of those K occurrences were selected to create synthetic models through interpolation. The difference between the feature vector and its neighbours is calculated using a distance metric. Such variation is then multiplied by any number in the range (0, 1), and summed up with an earlier feature vector. It can be depicted graphically as below:

Although the above method is beneficial, it does have a few limitations.

a) The generated synthetic instances point in a similar direction and are linked with artificial lines connecting diagonal models. As a result, the generated decision surface by some classifier algorithms becomes more complicated.
b) SMOTE generates a massive number of noisy data points in feature space.

B. HYBRIDIZED SMOTE

Undersampling, as well as oversampling techniques, are combined in hybridization approaches. This was done to improve the performance of the classifier model for samples made with these techniques.

C. SMOTE+ENN

The alternative hybrid approach employed in this work is SMOTE+ENN, which eliminates many observations from the test space. In this case, Edited Nearest Neighbor (ENN) is an alternative under-sampling approach that estimates the nearest neighbours of the majority class. It is eliminated if the nearest neighbours incorrectly label a specific majority class instance.

1) EDITED NEAREST NEIGHBOR

The deployed ENN approach operates by defining each observation's KNN and then determining if the observation's KNN's majority class is matched with a class of observation. In ENN, the number of nearest neighbours is K = 3 by default. The following explains the ENN algorithm:

- It provided a dataset of N observations and calculated K, the number of nearest neighbours. If K cannot be calculated, assume it is 3.
- Use the rest of the observations in the dataset to figure out KNN for the class of the observation, and then use KNN to find the majority class.
- If a class of observation and the majority class KNN differ, the observation and KNN are forbidden in a dataset.
- Step 2 and step 3 are iterated until the necessary parts of every class are matched.

This approach has much potential compared to Tomek Links [41] because ENN eliminates the observation and its KNN when the observation's class and K-NN's majority class differ, rather than simply removing the statement and its nearest neighbour with different classes. So, it is expected that ENN will clean up more data related to Tomek Links.

Incorporating the above approach with SMOTE's oversampled data substantially cleans the data. NN's samples' misclassifications are deleted in the above two classes. Hence, the separation of the classes is more evident and briefer.

The following explains the SMOTE-ENN process:

Step 1: From the minority, the class selects random data.

Step 2: Find the distance between the randomly generated data and also its KNN.

Step 3: The difference is multiplied by the random values 0 and 1, and the result is added to the synthetic sample of the minority class.

Step 4: Repeat until the appropriate proportion of the minority class is reached (Step 2– Step 3).

Step 5: The nearest neighbours are determined as K. Assume K as Step 3 if K cannot be estimated.

Step 6: Calculate KNN for the class of observation from the remaining dataset observations, and then, from KNN, return the majority class.

Step 7: If a class of observation and the majority class KNN differ, the statement and KNN are eliminated in a dataset.

Step 8: It is repeated until the necessary proportion of every class is met (Step 2 and Step 3).

D. FEATURE EXTRACTION MODE

1) KPCA-THEORY OF NON-LINEAR FEATURE EXTRACTION

After the dataset is balanced, the next step is to extract features from the dataset. PCA is feature extraction and linear dimensional reduction technique for High-Dimensional (HD) data. It converts the primary HD space's input data to the typical subspace, extracts the input data's primary feature vector, and achieves the goal of exploring the data. Most of the time, PCA can be done quickly and effectively on a set of data represented by 2$^{\text{nd}}$ -order correlations that change linearly or come from a Gaussian distribution. The variations of the accurate data, on the other hand, are widely known to be non-linear as well as highly non-Gaussian; correlations of 2$^{\text{nd}}$-order could not represent the majority of the data. As a result, if PCA is used, it would give a bad performance. Here, for our work, we propose "KPCA", a modified PCA approach that is non-linear and depends on functions of the kernel by inherently constructing a mapping from input space to feature space F, which is non-linear via non-linear transformation (Φ) as well as achieves PCA that is linear in feature space F. Among the two input samples, say (x, y), that is in the primary universe, it is possible to avoid non-linear mappings, and by using the Kernel Function (KF) given below, we can calculate the dot products in feature space:

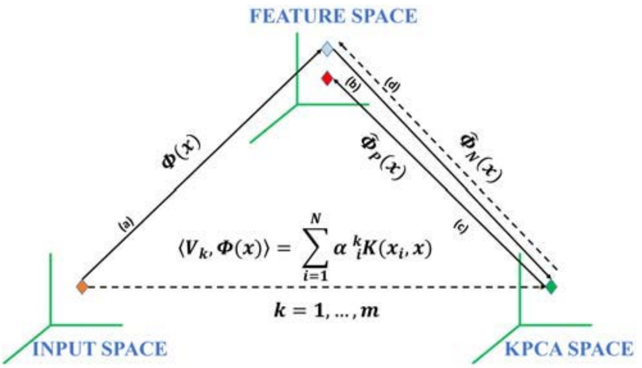

FIGURE 2. Kernel principal component analysis framework.

$$k(x, y) = \Phi(x) \cdot \Phi(y) \tag{1}$$

The KPCA approach conceptual structure is depicted schematically in Fig. 2. There are numerous KF forms in Equation 1. If KF is a positive integral operator's continuous kernel, a mapping exists (Φ) into a dot product space (F), so that method holds, according to Mercer's functional analysis theorem. It can reduce dimensionality more effectively if the KF requirement meets Mercer's theorem [42] and a good KF is chosen.

Few examples of KFs, EQU (2), EQU (3) and EQU (4):

$$\text{Polynomial kernel,} \quad k(x, y) = \langle x, y\rangle^d \tag{2}$$

$$\text{Sigmoid kernel,} \quad k(x, y) = \tanh\left(\beta_0\langle x, y\rangle + \beta_1\right) \tag{3}$$

$$\text{Radial basis kernel,} \quad k(xy) = \exp\left(\frac{\| x - y \|^2}{c}\right) \tag{4}$$

where d, β_0, β_1, and c are prior defined by the user

Mercer's theorem constantly satisfies the polynomial and the radial-based kernels, but the sigmoid kernel only helps it for specific β_0, β_1 values (Equation 2). For its better performance, the radial basis function is frequently used as a KF in KPCA; hence, the radial-based kernel is used as the KPCA-KF in this research (Equation 3). The radial basis function is often used as a KF in KPCA because it works better. Because of this, the radial-based kernel is used as the KPCA-KF in this research (Equation 4).

Providing an input data set (with a '0' mean X $(x_1, \ldots, x_N)$ $\in$ R_m where N is the number of samples and m is the measurement variables dimension) and the covariance matrix calculated by the PCA and KPCA algorithms, such as (i) PCA covariance and (ii) KPCA covariance, Equation 5 within a linear feature space F rather than a non-linear input space:

$$C = \frac{1}{N} x_i x_i^T = \frac{1}{N} XX^T \tag{5}$$

and Equation 6

$$C^F = \frac{1}{N} \sum\nolimits_{j=1}^{N} \Phi_j(x)\Phi_j(x)^T \tag{6}$$

where it is assumed that Equation 7

$$\sum\nolimits_{k=1}^{N} \Phi\left(x_k\right) = 0, \tag{7}$$

$\Phi(\cdot)$ is a non-linear mapping function that maps input vectors from input space to F.

It is noted that the feature space's dimension can be immense, probably infinite. For the covariance matrix to be calculated, the eigenvalue problem in feature space must be fixed: Equation 8

$$\lambda v = C^F v \tag{8}$$

Here,

Eigenvalues ≥ 0;

Eigenvector $v \in F$,

Equation 9 linearly expresses eigenvector 'v' for any $\Phi(x_i)$:

$$v = \sum\nolimits_{i=1}^{N} a(i)\Phi\left(x_i\right) \tag{9}$$

Equation 10 can be rewritten as the kernel eigenvalue problem:

$$N\lambda a = Ka, \tag{10}$$

where a $N * N$ matrix 'K' is a kernel matrix, $K = k_{ij} = \left(\Phi\left(x_i\right) \cdot \Phi\left(x_j\right)\right) = k\left(x_i, x_j\right)$ and 'α' is the feature vector of the kernel matrix. When reconstructing input data from feature space, we use Equation 11.

$$y_k = \langle v_k, \Phi(x) \rangle = \sum\nolimits_{i}^{N} a_i^k \left\langle \Phi\left(x_i\right), \Phi(x) \right\rangle \tag{11}$$

E. SUPPORT VECTOR MACHINES

This section discusses SVM theory, the theoretical foundation for the proposed Int-T2-FSVM framework. The SVM algorithm is a type of supervised learning algorithm. This indicates that for the training phase, data that is already labelled is used. Thus, it is feasible to develop a classifier that is used on a sample of items with an unknown assignment (this part is referred to as generalization). It is challenging to differentiate classes in the input space, even with labelled data. The SVM algorithm's primary principle is mapping data into HD feature space, where hyperplanes can be separated. The location of categorised items within the separated hyperplanes determines the output of a classifier.

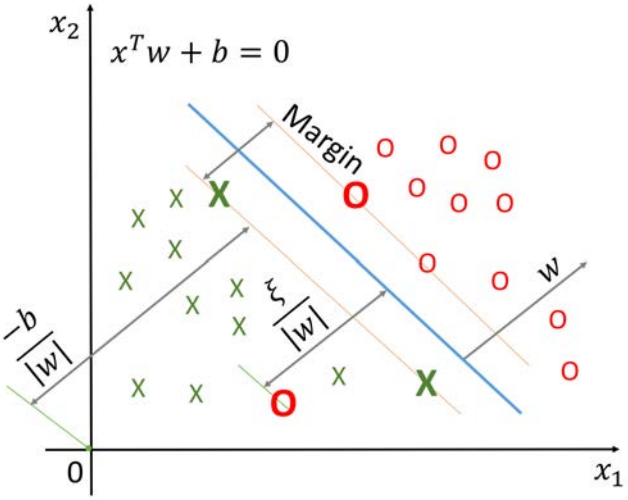

FIGURE 3. Linearly non-separable hyperplane and margin.

We provided a dataset S with labelled training points, Equation 12

$$(y_1, x_1), \ldots, (y_N, x_N) \quad i = 1, 2, \ldots, \mathrm{N} \tag{12}$$

where the training point is denoted by a vector x_i, the label is also denoted by a vector y_i, and the number of samples is denoted by 'N'.

Vector x_i is allotted to any of the two classes, which are denoted with $<$*Class Label*$>$ $yi \in \{-1, 1\}$. *A hyperplane* can be optimally positioned in the middle, separating the two classes. Data points closest to the margin serve as the foundation for such a definition and are referred to as "Support Vectors" (SV).

Fig. 3 shows a linearly non-separable case. Slack variables ξ_i refer to violations of strict separation.

Misclassification penalization, $\xi_i \geq 0$, is proportional to the distance between the misclassified point of x_i and canonical hyperplane restricting its class. Objective functions associated with margin maximization are denoted by Equation 13 and Equation 14:

$$\frac{1}{2} \| \mathbf{w} \|^2 + C \sum\nolimits_{j=1}^{2} \xi_i \tag{13}$$

$$\text{subject to: } y_i \left(x_i^T \mathbf{w} + b\right) \geq 1 - \xi_i$$

$$\xi_i \geq 0 \quad \mathrm{i} = 1, 2, \ldots, \mathrm{N} \tag{14}$$

C is weighted to account for classification errors. During classification errors that are unavoidable due to the linearity of the separating hyperplane, minimization of the objective function (1) with constraint (2) offers the maximum possible margin. By arranging the Lagrange function, the optimal hyperplane is initiated. The Lagrange function for the primary problem is as follows: Equation 15:

$$L_p(w, b, \xi) = \frac{1}{2} \| \mathbf{w} \|^2 + C \sum\nolimits_{i=1}^{n} \xi_i - \sum\nolimits_{i=1}^{n} \alpha_i \left\{ y_i \left(x_i^T \mathbf{w} + b \right) - 1 + \xi_i \right\}$$

$$-\sum_{i=1}^{n}\mu_i\xi_i \tag{15}$$

where $\alpha_i \geq 0$ and $\mu_i \geq 0$ are Lagrange multipliers. The primaeval problem is expressed as Equation 16:

$$\min_{v,b.\xi} L_p(\mathbf{w},b,\xi) \tag{16}$$

In this situation, first-order conditions are revealed in Equation 17

$$\begin{aligned}
&\frac{\partial L_p}{\partial \mathbf{w}} = 0 : \mathbf{w} - \sum_{i=1}^{n}\alpha_i y_i x_i = 0\\
&\frac{\partial L_p}{\partial b} = 0 : \sum_{i=1}^{n}\alpha_i y_i = 0\\
&\frac{\partial L_p}{\partial \xi_i} = 0 : \mathrm{C}{-}\alpha_i - \mu_i = 0
\end{aligned} \tag{17}$$

With particular reference to the Lagrange multipliers' scenarios, Equation 18:

$$\begin{aligned}
\alpha_i &\geq 0\\
\mu_i &\geq 0\\
\alpha_i\left\{y_i\left(x_i^T\mathbf{w}+b\right)-1+\xi_i\right\} &= 0\\
\mu_1\xi_i &= 0
\end{aligned} \tag{18}$$

Therefore $\sum_{i=1}^{n}\alpha_i y_i b = 0$, the primal problem translates into Equation 19:

$$\begin{aligned}
L_D(\alpha) = {} & \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j x_i^T x_j\\
& - \sum_{i=1}^{n}\alpha_i y_i x_i^T \sum_{j=1}^{n}\alpha_i y_i x_j + C\sum_{i=1}^{n}\xi_i\\
& + \sum_{j=1}^{n}\alpha_i - \sum_{j=1}^{n}\alpha_i\xi_i - \sum_{j=1}^{n}\mu_i\xi_i\\
= {} & \sum_{j=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j x_i^T x_j\\
& + \sum_{i=1}^{n}\xi_i\left(C-\alpha_i-\mu_i\right)
\end{aligned} \tag{19}$$

As the last term is 0, the first level dual problem results in Equation 20:

$$L_D(\alpha) = \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j x_i^T x_j \tag{20}$$

The following is the initial decision function: Equation 21:

$$D\left(x\right) = \sum_{i\in S}\alpha_i y_i x_i^T x + b \tag{21}$$

S is a collection of SV indices. Since α_i It is non-zero for SV sum-up in Equation 22; it is applied only for SV. Forever and ever α_i,

$$b = y_i - \mathbf{w}^T x_i \tag{22}$$

is fulfilled. To assure the value of precision, an average of b computed for unbounded SV is assumed; equation 23:

$$b = \frac{1}{|U|}\sum_{i=U}\left(y_i - \mathbf{w}^T x_i\right) \tag{23}$$

where U denotes a collection of unbounded SV indices.

SVM solves the classification problem by mapping the inputs 'x'' into a HD space by mapping non-linear features $\phi(x)$ separated by complicated decision boundaries in the input space. Because of this, the problem becomes a situation in the feature space that can be separated in a linear way.

$x_i^T x_j$ substituting just a scalar product by KF, $K(x_i, x_j)\phi^T(x_j)\phi(x_j)$ supposed to be symmetric and positive-definite, subject to constraints, the dual problem is reformulated as follows: Equation 24 and Equation 25

$$Q(\alpha) = -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j K\left(x_i, x_j\right) + \sum_{i=1}^{n}\alpha_i \tag{24}$$

$$\begin{gathered}
\text{Subject to } \sum_{j=1}^{n} y_i\alpha_i = 0, 0 \leq \alpha_i \leq C\\
i = 1, 2, \ldots, N
\end{gathered} \tag{25}$$

$K\left(x, x_i\right) = x^T x_i$ in (8) is a linear kernel.

The Gaussian kernel is the most common, and its most common definition is Equation 26

$$K\left(x, x_i\right) = \exp\left(-\frac{\|x - x_i\|^2}{2\sigma^2}\right) \tag{26}$$

1) INFLUENCE OF KARUSH-KUHN-TUCKER THEOREM (KKTT)

KKTT is significant to the SVM's development. According to the theorem, the answer must meet the following requirements:

$$\alpha_i\left(y_i\left(\omega\cdot z_i + b\right) - 1 + \xi_i\right) = 0, \quad i = 1, 2, \ldots, \mathrm{N} \tag{27}$$

$$\left(C - \alpha_i\right)\xi_i = 0, \quad i = 1, 2, \ldots, \mathrm{N} \tag{28}$$

Equation 27 and Equation 28 imply that only non-zero values 'α_i', meet the requirements. SVs are the values 'x_i' that corresponds to the solution 'α_i'. When 'x_i', it corresponds to $\alpha_i = 0$ and a sufficient distance from the decision margin, the instance is appropriately classified.

In order to build the best possible hyperplane $\omega\cdot z + b$, we would require that Equation 29

$$\omega = \sum_{i=1}^{N}\alpha_i y_i z_i \tag{29}$$

The scalar bias b should be calculated using the KKTT conditions. The decision function can hence be obtained from Equation 30 and Equation 31 as follows:

$$\mathrm{f}(x) = \mathrm{sgn}(\omega\cdot z + b) = \mathrm{sgn}\left(\sum_{i=1}^{N}\alpha_i y_i z_i\cdot z + b\right) \tag{30}$$

where $sgn(\cdot)$ is the sign function that determines the sign (+/−) of a real value. Since we lack data for feature space of higher dimension $\varphi(\cdot)$, the calculations in EQU (31) are impractical because of their complexity. A beneficial feature of the SVM is that it does not require determining'$\varphi(\cdot)$. Complexity is resolved using a KF that can compute data points as dot products in the 'z' feature space. Before these functions can be used to figure out the dot products, they must prove Mercer's theorem.

$$z_i\cdot z_j = \varphi\left(x_i\right)\cdot\varphi\left(x_j\right) = K\left(x_i, x_j\right) \tag{31}$$

Here, $K\left(x_i, x_j\right) = \varphi\left(x_i\right)\cdot\varphi\left(x_j\right)$ KF is used for mapping onto a feature space of a higher dimension. KFs can be either

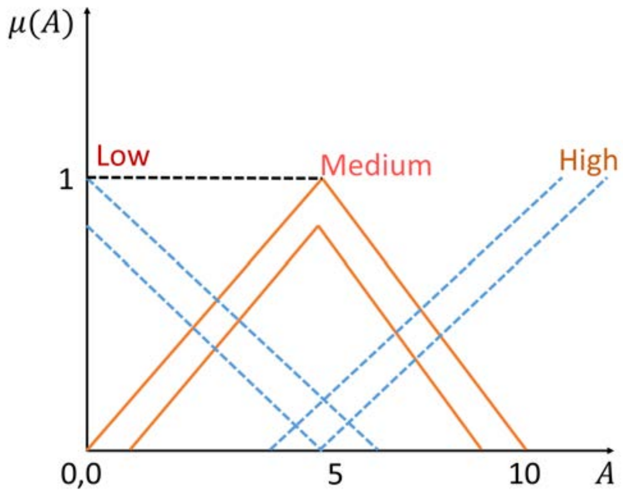

FIGURE 4. Membership function of type-2.

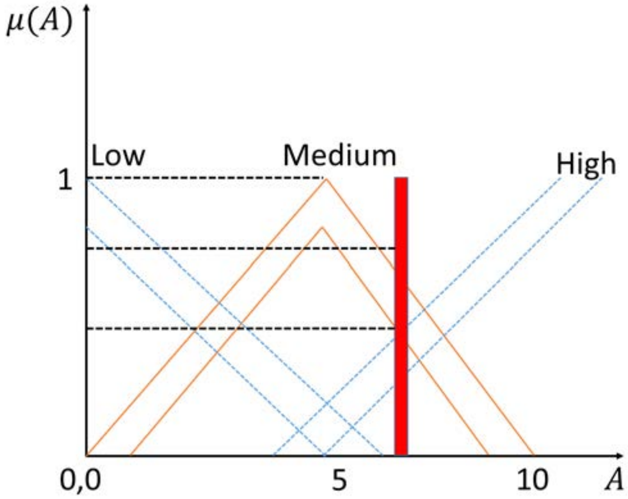

FIGURE 5. Membership function showing grading.

linear or non-linear. Solving the following Equation 32 and EQU (33) yields the non-linear separating hyperplane:

$$\min Q(\alpha) = \frac{1}{2}\sum\nolimits_{i=1}^{N}\sum\nolimits_{j=1}^{N} y_i y_j \alpha_i \alpha_j K\left(x_i, x_j\right) - \sum\nolimits_{i=1}^{N}\alpha_i \tag{32}$$

subject to Equation 33

$$\sum\nolimits_{i=1}^{N} y_i\alpha_i = 0, \quad 0 \leq \alpha_i \leq C, \quad i = 1, 2, \ldots, N \tag{33}$$

The Decision function can finally be illustrated as Equation 34:

$$f(x) = \operatorname{sgn}(\omega \cdot z + b) = \operatorname{sgn}\left(\sum\nolimits_{i=1}^{N} \alpha_i y_i K\left(x, x_i\right) + b\right) \tag{34}$$

F. FUZZY SETS OF TYPE-2 AND INTERVAL TYPE-2 FLS

In various scientific and technical implementations, especially in control systems, Fuzzy Logic (FL) is vital. In 1965, Zadeh developed fuzzy sets to analyse unprobabilistic uncertainty in information. In the Fuzzy Logic System (FLS), the information utilised to generate the rules is uncertain [43]. Uncertainty about the antecedent and consequent regenerates into uncertainty about the antecedent and consequent Membership Functions (MF). Antecedent and consequent MFs in Type-2 FLSs are Type-2 fuzzy sets that simply manage rule uncertainty. So an expansion of the standard fuzzy set concept, *i.e.,* a Type-2 fuzzy set, into the concept of Type-1 fuzzy sets was commenced. In type-2 fuzzy, the grades of MF are also fuzzy. Membership of the Type-2 grade is any subset of (0, 1), which is known as primary membership [44]. A secondary membership (0,1) corresponds to each primary membership and describes the primary membership probability. Type-2 fuzzy defines a subset of Type-1 fuzzy, which is represented by a MF, as shown in Fig. 4, by a triangle MF [45].

The output processor, which contains a de-fuzzifier and a type reducer, produces a Type-1 output of a fuzzy set or a definite number. If Type-2 logic MF is expressed as an interval, FL Type-2 is transformed into an interval of Type-2 FL. Even though If-Then rules were commonly active for FL Type-2 characterization, antecedent/consequential sets are Type-2 now.

type-2 fuzzy set, represented by $\tilde{A}$, classified by type-2 MF $\mu_{\tilde{A}}(x, u)$, $x \in X$ and $u \in j_x \subseteq [0, 1]$, denoted by Equation 35:

$$\tilde{A} = \int_{x\in X}\int_{u\in J_x} \frac{\mu_{\tilde{A}}(x, u)}{x, u}, \quad J_x \subseteq [0, 1] \tag{35}$$

where $\int\int$ implies the union's overall maximum allowable *x, u,* and $J_x \subseteq [0, 1]$, J, which is termed as *x*'s basic membership.

A secondary membership value for each primary membership value describes the possibility of a primary membership value. However, secondary MF has values in the [0, 1] range illustrated in Fig. 5. The most critical task in the Type-2 FLS design is the MF's specification. The choice of MF style (Gaussian, Triangular) and thus selecting their specific parameters directly impacts performance. IT2FLSs are being studied for a variety of mitigation techniques. Most of the time, these strategies are built on the knowledge of experts, Genetic Algorithms (GA), Neural Networks (NN), and other similar methods.

Nonetheless, there is a need to simplify and optimise the classification of unambiguous MFs in this space. IT2FLS practices are used in a wide range of science and engineering fields due to the increased practicability within the computations. If the MF position cannot be determined precisely, the degree of membership cannot be taken as a fixed range of (0, 1), and Type-2 fuzzy sets are the best option. If all A are assigned to their distribution, the Type-2 3-D FL-MF specifies the formation of Type-2 fuzzy set features. The Footprint Of Uncertainty (FOU) is defined as a union of primary memberships bounded by the upper and lower Type-1 MF, referred to as upper MF $\bar{\mu}_{\tilde{A}}(x)$ and lower MF $\underline{\mu}_{\tilde{A}}(x)$.

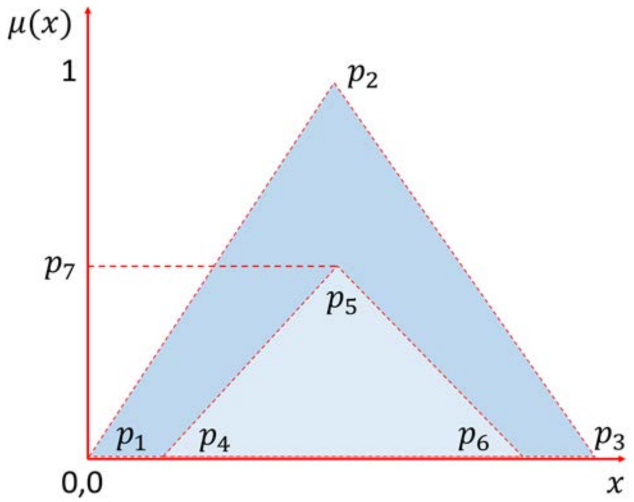

FIGURE 6. IT2 MFs examples; dashed line: LMF; dotted line: UMF; blue area: FOU.

When the unknowns in the MFs are taken out, type-2 fuzzy sets are reduced to type-1 fuzzy sets that can be identified exactly.

FLS of Type-2, like FLS of Type-1, has 4 general modules: (1) Fuzzifier; (2) Fuzzy rule base; (3) Fuzzy Inference Engine (FIE); and (4) Output processor. A notable distinction between FLS of Type-1 and FLS of Type-2 is that Type-2 FLS processor output requires an added step: This type-reducer directly before the defuzzifier is needed to lower the fuzzy output sets of Type-2 to fuzzy output sets of Type-1. After type reduction, the defuzzifier takes the fuzzy output sets of Type-1 and turns them into clear values.

G. INTERVAL TYPE-2 FUZZY INFERENCE SYSTEM (INT-T2-FIS)

Int-T2-FIS is being employed as an alternative to T2FIS since the arithmetic needed for Int-T2- FIS is significantly more accessible than the arithmetic needed for T2FIS.

Different types of MF can be used for the research being directed. Fig. 6 shows the triangular Int-T2-FIS MF. The dashed lines denote the lower MF named LMF, while the dotted line denotes the upper MF called UMF. Yet, due to its ease of implementation, the triangle MF was utilized. In the perception that every non-linear process can be imprecise to an arbitrary level of precision in a confined domain, FL Type-1 is a global approximator. This trait is prolonged to the Type-2 scenario; thus, we can assume a comparable level of competence. Keeping this point in mind, Int-T2-FIS must perform well regardless of the MF shape, as other factors influence performance, such as the number of fuzzy rules used. Users can predefine the MF or design it using optimization approaches like the GA. The GA can optimise MF for each input, denoted by nearly seven points: *p1, p2, p3, p7.*

FOU is defined as the space between UMF and LMF, which is seen in Fig. 7 as a blue area. A FOU is a union of the entire Type-2 FS fuzzy membership grades, representing uncertainties in a fuzzy system. Because a type-2 FS's FOU adds a dimension of mathematics, type-2 FSs are likely to outdo their counterparts of type-1. Contradict to the Type-1 example, in which the grade of membership is a single value, the membership grade of Int-T2-FIS is a range. Int-T2-FIS is limited at the two extremities of the period to yield LMF and UMF, both of which are fuzzy sets of Type-1.

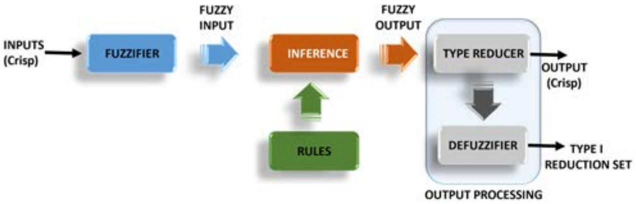

FIGURE 7. Int-T2-FIS block diagram.

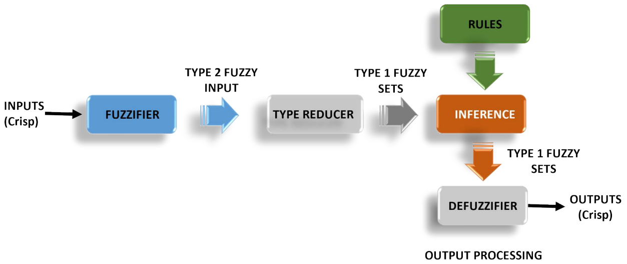

FIGURE 8. Type-2 FLS with reducer.

The construction of the Int-T2-FIS details the relationship between input and output. The Int-T2-FIS is made up of five primary modules: (1) Fuzzifier; (2) Fuzzy Rules; (3) Inference Engine; (4) Type Reducer; and (5) Defuzzifier. The output unit of an Int-T2-FIS is made up of 2 blocks: (a) type-reducer and (b) defuzzifier. Because fuzzy settings activate the rule basis, rather than numbers, in the fuzzifier block, crisp inputs are initially converted to FS. Once measurements are excellent, input is preserved as a crisp data set in the fuzzification step; once the measurements are chaotic but stable, input is represented as a Type-2 fuzzy interval set. A set of fuzzy inputs is mapped onto fuzzy outputs with the help of a fuzzy inference engine after the input has been fuzzified. This is accomplished by quantifying every rule using the fuzzy set theory and then applying the mathematics underlying the theory of the fuzzy set to produce an output favouring every rule. The fuzzy inference block's result now has one of many sets from fuzzy production. With the help of output processing units, the fuzzy output collections are turned into crisp output.

Provided an Int-T2- FIS with n inputs $x_i \in X_i, \ldots, x_n \in X_n$ to produce a single output $\in Y$. This Int-T2- FIS 's rule base is made up of K IT2 fuzzy rules, written as follows: Equation 36

$$R^k : \text{If } x_1 \text{ is } \tilde{F}_1^k \text{ and} \cdots \text{and } x_n \text{ is } \tilde{F}_n^k \text{ THEN } y \text{ is } \tilde{G}^k \quad (36)$$

$k = 1, \ldots, K, \tilde{F}_n^k$ *and* $\tilde{G}^k$, epitomizes Type-2 fuzzy sets.

1) COMPUTATIONALLY EFFICIENT TYPE-REDUCER

The Karnik-Mendel (KM) iterative approach using the center of sets is a prominent type-reducer. Unfortunately, such a type-reduction approach is mathematically demanding, especially when many MFs have a considerable rule base. Fig. 8, the schematic construction of a Type-2 FIS, demonstrates that type-reduction is conducted on the FIE's output. As a result, the inference engine and the type-reducer must deal with intermission firing strength. These raise the mathematical load and make Type-2 FLSs inappropriate for some real-time functions. Equivalent Type-1 fuzzy systems, termed ET1FSs, allow Type-2 fuzzy systems, which are considered Type-1 fuzzy systems collections [46]. This notion can reduce type-reduction to identify an equivalent Type-1 fuzzy system corresponding to a specific input. The type-reducer must identify the equivalent type-1 membership grade (ET1MG) for every interval fuzzy set. When the ET1MG is determined, the type-2 fuzzy set FS is reduced to a crisp value, and the type-2 FLS output may be determined using a defuzzifier and FIE of Type 1. In brief, the type reduction technique can be used before the inference engine to select the best ET1MGs based on inputs. In this case, the inference engine only keeps track of crisp computing integers instead of sets of intervals. This means that the computational overhead is lower, and the calculations may be done faster than with a FLS of Type-2 used with the KM iterative algorithm.

Even though the novel technique alters the processing order, the type-reducer proposed should not modify features of FLS of Type-2. Type-2 FLC must meet the below constraints:

1. Once the uncertainty footprint is taken away, the Type-2 FLS reduces to its Type-1 equivalent. This means that the type reducer should produce a Type-1 FLS that is equivalent to the Type-1 FLS that was used as a baseline.
2. ET1F alters as input changes. As a result, the type-reducer must fit all the input variables.
3. According to research on using FLS of Type-2 for control, the control surface of FLC is often smooth when compared to FLC of Type-1, particularly near the origin ($e = 0, \dot{e} = 0$). One feature that makes a FLC of Type-2 more robust than a FLC of Type-1 is the Type-2's smoother control surface. As a result, the type-reducer should result in softer control surfaces.

Considering the above constraints, the type-reducer built with GA to reduce the interval fuzzy set $[f_l, f_u]$ to an ET1MG, f_{eq}, can be defined as the following Equation 37:

$$f_{eq} = f_u - \sum_{i=1}^{N} \alpha_i \frac{2\,|x_i|}{P_{x_i 2} - P_{x_i 1}} \times (f_u - f_l) \tag{37}$$

N denotes the number of inputs

α_i denotes weight evolved by GA

x_i denotes i^{th} input

$Px_i2 - Px_i1$ (is the support of a baseline Type-1 fuzzy system while the footprint of uncertainty disappears).

2) INFERENCE ENGINE

The inference engine is responsible for applying the inference rules to the fuzzy input and producing the output. The inference rules, in particular, are engaged in assessing linguistic values and mapping them to fuzzy sets, which then need defuzzification to be transformed into crisp values. Inference rules that give the system's calculation functionality are one of the primary principles of the Mamdani method [47]. These guidelines can be founded on prior experiences, observations, and expert knowledge. Every fuzzy inference rule comprises two concepts: (1) *If-Then* statements and (2) the variables of linguistic expression. Antecedents and consequences are contained in the *If-Then* rules. When creating an inference rule, "AND," "OR," and, occasionally, "NOT" operators are utilised [49]. The combination of operators is known as t-norms. The following defines the fuzzy "&" operator:

$$\mu A \cap B(x) = \min[\mu A(x), \mu B(x)] \tag{38}$$

μA represents class A membership

μB represents the class B membership.

This rule obtains the least number of fuzzy set membership values necessary to compute the "AND" operation. The fuzzy "OR" operator is described as:

$$\mu A \cup B(x) = \max[\mu A(x), \mu B(x)] \tag{39}$$

Equation 38 and Equation 39, x represent the corresponding fuzzy sets' degrees of MF. For example, $A(x)$ denotes fuzzy set A membership degrees. The "OR" operation is calculated by obtaining the most outstanding value of membership values of the fuzzy sets. We utilized the "AND" operator to create the inference rules because the evaluation factors are interdependent. The "OR" operator is typically used for separate, non-closely connected components. The rule strength allows the fuzzy outputs to be aggregated into a distribution [48].

3) DEFUZZIFICATION

The inference engine's fuzzy output is mapped to a crisp value that gives the exact fuzzy set representation during defuzzification. In this proposed fuzzy methodology, the crisp production is generated by employing the centroid method, which is defined below, Equation 40:

$$z = \frac{\sum_{j=1}^{n} z_j \mu_c(z_j)}{\sum_{j=1}^{n} \mu_c(z_j)} \tag{40}$$

The centroid approach determines a single scalar value by using the centre of mass, denoted as z, in the distribution of fuzzy output. The fuzzy set membership is represented by u_c, while the membership value is presented by z_j.

H. DATA

The SDSS DR14 data collection is used in this study. The SDSS is one of the largest spectroscopic surveys, having begun observations in 1998 and completing three phases. SDSS-IV, the fourth phase, is already in progress [49]. The camera for the telescope was made up of 30 Charge-Coupled Devices (CCD) chips, each with a resolution of 2048 x 2048 pixels. The chips were stacked in five rows, each with six chips. Each row looks at the space via different optical filters (u', g', r', i, z') with different wavelengths: u' = 354 nm, g' = 475 nm, r'= 622 nm, i' = 763 nm, and z' = 905 nm [50]. SDSS DR14 is the SDSS-second IV's release. More than 2.54 million spectra have been given, comprising 928859 stellar spectra. The raw spectra contain 3850 points within the range of wavelength specified by the device, which is $\lambda = 3950\ to\ 9350$Å. In terms of resolution, the interval is uniform ($\frac{\delta\lambda}{\lambda} = \frac{1}{4342}$). When the redshift is taken into consideration, the range shared by all spectra is 3806 to 7371 wavelengths. The spectra were then corrected for redshift using the Shannon criterion to preserve the form of the spectral lines, as described by [51]. We increased the sample of the spectra earlier for this purpose, resulting in 5748 points for each spectrum. After that, each spectrum was normalized by dividing it by its average value between 4250 *and* 5150Å. To minimize the dimensionality of the data array, we used wavelet filtering accompanied by offloading by a factor of four to create spectra with 1443 wavelengths. We save most of the information in this procedure, including the forms of the lines, as well as complete neutrality.

I. METHODOLOGY

1) DATA PROCESSING

To begin with, because of the enormous number of sources in a spectroscopic catalogue, we divided the entire dataset into 2 parts, utilizing one part for the initial training set and the other for the introductory test set. For unbiased comparison, we divided the dataset into 25% for the test set and 75% for the training set (30% is used for cross-validation). In training, we employ the SMOTE preprocessing model to avoid the fit being influenced by an imbalance between the several classes, which is mainly produced by galaxies' excess. We train such models to forecast a source's <Class Label> in stable test datasets and analyse how the number of facts in the training set impacts model efficiency by introducing the classification model using escalating percentages of the whole training set.

2) IMPLEMENTATION: INTERVAL TYPE-2 FUZZY SVMS (INT-T2-FSVM) FOR CLASSIFYING ASTRO PHYSICAL OBJECTS

The methodology of the Int-T2-FSVM classifier used in the selected dataset is discussed. This hybrid classification method combines Int-T2-FIS and SVM, which generates Int-T2-FSVM and employs a standard classifier from SVM. Int-T2-FSVM is a classifier with several inputs and a single output. Int-T2-FIS's capacity to manage insecurity makes it an excellent companion to SVM in addressing challenging non-linear situations. Fig. 9 depicts the overall architecture of Int-T2-FSVM. The input of the feature vector is acquired after the K-PCA component has extracted the required features from the SMOTE+ENN balanced input data of SDSS. Multiple Int-T2-FSVMs are required in the application in this study, which is to differentiate between astrophysical objects because there are three types (stars, galaxies and QSO). Since the hyperplane can only tell the difference between two classes, more SVMs are needed if there are more than two classes.

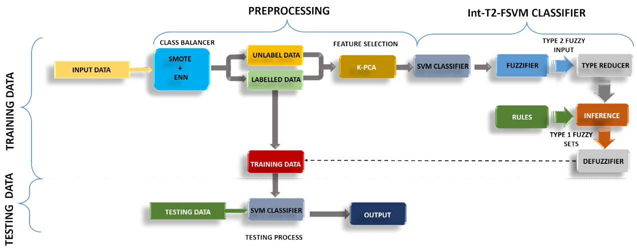

FIGURE 9. General architecture of the proposed Int-T2-FSVM.

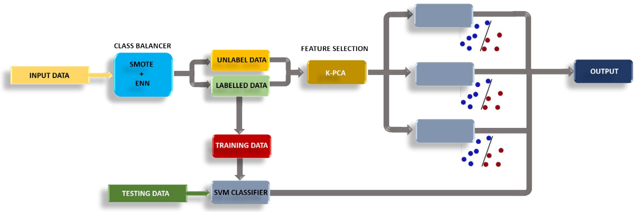

FIGURE 10. Int-T2-FSVM architecture for the astrophysical object classification.

TABLE 1. Outline the if-then rules that were applied.

Test Case	Output			End Class Output
	I	II	III	
1	-1	-1	-1/1	1
2	1	-1/1	-1	2
3	-1/1	1	1	3
4	1	-1	1	3
5	-1	1	-1	3

As in Fig. 10, the block of Int-T2-FSVM can be reproduced and utilised to segregate the unique objects separately. We can recommend three Int-T2-FSVM blocks for identifying three classes [52].

1. Int-T2-FSVM1 can tell the difference between the phases of a star and a galaxy. A label of "-1" means that the data is from the star class, and a label of "1" means that it is from the galaxy class.
2. Int-T2-FSVM2 can tell the difference between the Star and Quasar classes. An input data label of "-1" means that the data fits the Star class, and an input data label of "1" means that the data fits the Quasar class.
3. Int-T2-FSVM3 can tell the difference between Galaxy and Quasar classes. A label of "-1" means that the data fits the Galaxy class, and a label of "1" means that the data fits the Quasar class.

Outputs 1 through 3 show the labels of the outputs of 3 Int-T2-FSVM blocks, which are then run through a classifier based on rules to decide the final classification (Tab. 1).

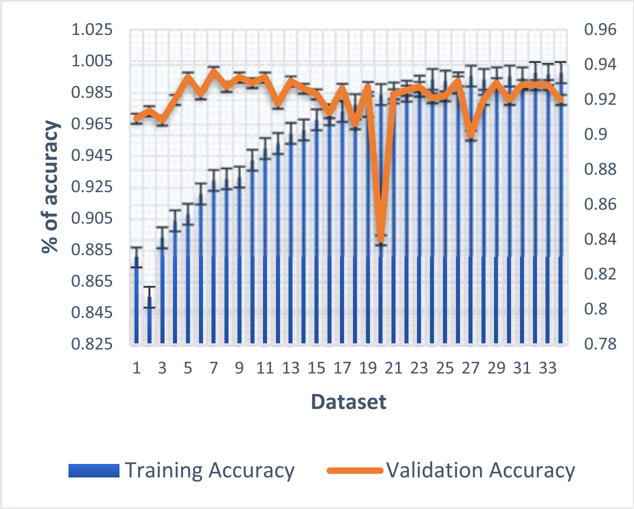

FIGURE 11. Training and validation accuracy.

Table 1 depicts a class determiner system based on rules that pick the output of Int-T2-FSVM 's final classification. The whole number 1-3 stand for the last class. "1" means Star class, "2" means Galaxy class, and "3" means Quasar class.

The int-T2-FSVM block comprises three fuzzy rules linked with LMF and UMF and a defuzzification block that produces final crisp outputs [53]. The SVM block's final result is created via merging SVM outputs with MF, where MG is applied to each output to show the effect on the end output. Some integer values may represent fuzzy rule values. However, increasing the number of rules will result in delayed training convergence and a higher system computing cost. Three fuzzy rules are used in this study to execute Int-T2-FSVM. MG is derived from MFs, which a user outlines and has a triangle shape, as illustrated in Fig. 1. The GA optimises the point's p1 to p7 to represent the membership function shape.

As represented in Fig. 11, there are 3 IT2 SVMs where every IT2 SVM is regulated via the rules below:

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

$\| x \|$ is normalized input

$\tilde{F}^j$ is IT2 triangular MF as denoted in Equation 41

$\tilde{G}^j$ is a singleton by output $\underline{Out}_{jk}$ as well as $\bar{O}ut_{jk}$ by definition in the below hyperplanes:

$$\begin{aligned}\underline{Out}_{jk} &= \operatorname{sgn}\left(\underline{\omega}_{jk} \cdot z + \underline{b}_{jk}\right)\\ &= \operatorname{sgn}\left(\sum\nolimits_{i=1}^{N} \alpha_{ijk} y_i K\left(x_i, x\right) + \underline{b}_{jk}\right)\end{aligned} \tag{41}$$

$$\begin{aligned}\bar{O}ut_{jk} &= \operatorname{sgn}\left(\overline{\omega}_{jk} \cdot z + \bar{b}_{jk}\right)\\ &= \operatorname{sgn}\left(\sum\nolimits_{i=1}^{N} \alpha_{ijk} y_i K\left(x_i, x\right) + \bar{b}_{jk}\right)\end{aligned} \tag{42}$$

j = 1 to 3 refers to jth (lower/upper) SVM

k = 1 to 3 refers to kth Int-T2-FSVM

TABLE 2. Features involved in SDSS dataset.

Data		Columns		
#	Column	Count	Non-Null	Data Type
1	Obj$_{id}$	30000	Non-Null	Int 64
2	RA	30000	Non-Null	Float 64
3	DEC	30000	Non-Null	Float 64
4	U	30000	Non-Null	Float 64
6	I	30000	Non-Null	Float 64
7	Z	30000	Non-Null	Float 64
8	Run	30000	Non-Null	Int 64
9	Rerun	30000	Non-Null	Int 64
10	Camcol	30000	Non-Null	Int 64
11	Field	30000	Non-Null	Int 64
12	Specobj$_{id}$	30000	Non-Null	UInt 64
13	Class	30000	Non-Null	Object
14	Redshift	30000	Non-Null	Float 64
15	Plate	30000	Non-Null	Int 64
16	Mjd	30000	Non-Null	Int 64
17	Fiber$_{id}$	30000	Non-Null	Int 64

A defuzzification technique may then be used to obtain Int-T2-FSVM k's output k. A rule-based class determiner would make the final class selection.

J. FEATURES INVOLVED IN SDSS DATASET

There are various features in the SDSS dataset (Tab. 2). The following are the features required to make a classification in our work [54].

- RED SHIFT: Redshift is the essential attribute that distinguishes quasars. Quasar's distance is calculated by its redshift, a measurement by which the universe's expansion stretches the wavelength of its light before reaching Earth. The greater the redshift, the greater the distance; the further back in time, astronomers view the object.
- RIGHT ASCENSION: The eastward angular distance of a particular location is measured along the celestial equator from the sun at the March equinox to the (hour circle of the) place in the question above the earth. This attribute can be derived from the image table.
- When combined with right ascension, declination is an astronomical coordinate system that indicates the point location on the celestial sphere in an equatorial coordinate system.

III. PERFORMANCE METRICS

The measures we use to evaluate the performance of the classifiers are discussed now.

A. CONFUSION MATRIX (CM)

CM holds counts of all probable model forecast results; for each categorization, there are nearly four probable results. If the model successfully predicts "real" things, it is referred to as a "True Positive (TP) (t_p)", and if it mistakenly predicts "Not Real" objects, it is referred to as a "False Negative (FN)" (f_n). If, on the other hand, the model correctly predicts that an object is "not-real," this is a True Negative (TN) (t_n). It is, however, a "False Positive (FP)" if it is classified as "real" when it is not In a nutshell, it contains the overall number of TP/FP and TN/FN.

We measured a probabilistic classifier, which means that the classification of the sources into stars/galaxies is based on the probability that the class threshold is set. In our scenario, all objects with <*Class Label*>=1 are galaxies, all objects with <*Class Label*>=2 are stars, and all objects with <*Class Label*>=3 are quasars. The requirements for completeness and purity determine the class. When a <*Class Label*> is provided, the classification performance may be summarised using a CM to comprehensively compare predicted and true values.

B. ACCURACY

The number of predictions in a given model indicates the model's accuracy. Our model's accuracy is the initial measure because the dataset's size is similar to ours. The accuracy of the model is computed as given below:

$$A = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} \tag{43}$$

$$A = \frac{t_p + t_n}{t_p + t_n + f_p + f_n} \tag{44}$$

In the above, Equation 42 and Equation 43 $\mathrm{t_p}$ are the TP, $\mathrm{t_n}$ which denotes the TN, $\mathrm{f_p}$ represents the FP, and $\mathrm{f_n}$ signifies the FN.

Sensitivity-specificity and precision-recall are two categories of metrics that may be helpful for imbalanced classification because they are class-specific.

C. SENSITIVITY-SPECIFICITY METRICS

Sensitivity is a measure of how accurately the positive class was predicted and referred to as the True Positive Rate (TPR). The complement to sensitivity, or True Negative Rate (TNR), is sensitivity Specificity, which summarises how accurately the negative class was predicted. Equation 44 provides the following measurement for the sensitivity (Sn):

$$(Sn) = (TP)/((TP + FN)) \tag{45}$$

D. SPECIFICITY

The sensitivity for imbalanced classification may be more intriguing than the specificity. Equation (45) is presented as the following:

$$Sp = (TN)/((FP + TN)) \tag{46}$$

E. PRECISION

Precision is a metric that measures the proportion of the TP in the given samples. The precision can be calculated using the formula given below, Equation 6

$$Precision = \frac{TP}{TP + FP} \tag{47}$$

F. RECALL

A recall is a metric that measures how many real positive tuples are correctly classified. Equation (47)

$$\text{Recall} = \frac{TP}{TP + FN} \tag{48}$$

G. F1-SCORE

It is a model that combines recall and precision, and it is done by calculating the Harmonic Mean between precision and recall. The following is how it was calculated: Equation 48

$$F_1 = 2 * \frac{1}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}} / F_1 = \frac{2PR}{P + R} \tag{49}$$

H. FALSE-POSITIVE RATE (FPR)

FPR is the number of positive values that were mistakenly counted as negatives divided by the number of negatives that should have been counted, Equation 49

$$FPR = \frac{f_p}{f_p + t_n} \tag{50}$$

The Receiver-Operating Characteristic (ROC) Curve is a systematic technique for summarising a classifier's performance. TPR and FPR are plotted as a function of p_{cut} in a parametric plot, Equation 50.

$$\text{TPR}\ (p_{\text{cut}}) = \frac{t_p}{t_p + f_n} \text{FPR}\ (p_{\text{cut}}) = \frac{f_p}{f_p + t_n} \tag{51}$$

In conjunction, "Recall" is represented as TPR, indicating completeness. An AUC can be used to summarise the performance of a classifier. It takes a value between 0 and 1. 1 is the value an ideal classifier brings, and an average classifier takes the value of 0.5.

We present the results of the unrefined proposed model in Tab. 3. The results are compared with and without the use of SMOTE + ENN for all the metrics; the results show that the model performance to correctly predict the class label is getting better by using SMOTE + ENN to balance the data. The results are comparable with other existing models in terms of all the metrics. The adoption of KPCA as the feature extraction scheme reflects greater efficiency as the adopted model proves its credibility by effectively reducing the dimension of the dataset. The SDSS dataset we chose proves to be a difficult platform for our proposed classification model [55]. The proposed model's training and validation accuracy is displayed (Tab. 4).

It is common for many classification models to generate poor representations of the labelled data for datasets that provide a thinner training set than the generalisation task requirement. But the "SMOTE + ENN" effective balancing model proposed in this research work helps solve this problem, as shown by its ROC in Fig. 12.

Following the training and testing of the proposed model and observing the accuracy of training and loss, we can conclude that the model performed well since the training

TABLE 3. Performance of proposed model's.

	Accuracy	Sn	Sp	Recall	Precision	F1-Score
With Smote + ENN	96.21%	88.42%	87.63%	97.19%	96.21%	97.42%
Without Smote + ENN	84.37%	82.74%	80.11%	89.23%	82.1%	90.33%

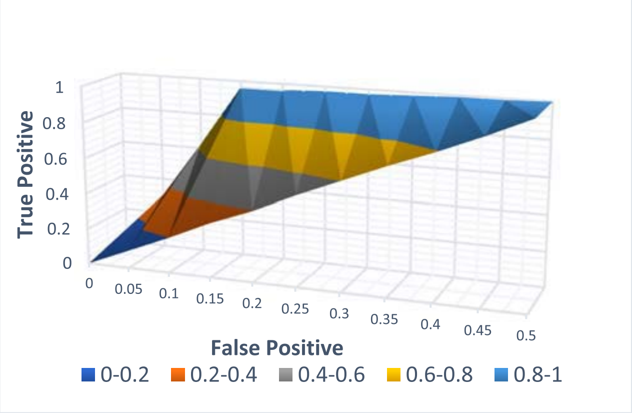

FIGURE 12. ROC curve.

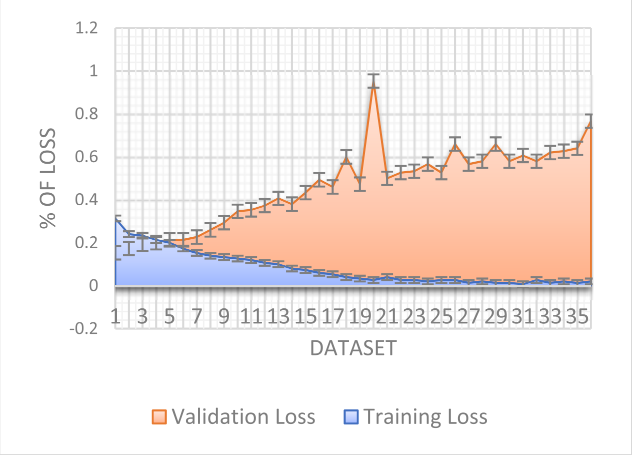

FIGURE 13. Training and validation loss.

accuracy is more than 97% after 30 epochs and the training loss is relatively low, as shown in Fig. 13. A high generalisation model prevents overfitting and gives useful results when dividing astronomical image data into real and fake objects [52], [53], [54], [55].

Because the two major classes in our data (real and non-real objects) are similar in size, we considered accuracy and recall to be the most important performance metrics in our solution and benchmark model (Fig. 14). Accuracy is a good measure of quality. In this case, losing true items (FNR) is more important than contaminating our collection of predicted objects with FP, which humans can quickly wholly eliminate. These findings indicate that it can play a valuable role in future astronomical surveys. Fuzzy-based approaches seem to be as good as, if not better than, human scanners in this sector. However, unlike astronomers, they can categorise thousands of transients in a single second. Unlike traditional ML algorithms, Int-T2-FSVM does not involve the creation of sophisticated and case-specific features. Fuzzy SVMs use simple data augmentation during training to come up with abstract features for categorising on their own.

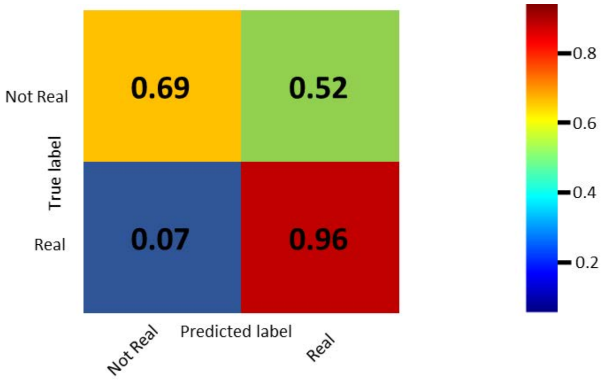

FIGURE 14. Confusion matrix.

DL models, particularly the proposed Int-T2-FSVM, are critical for future astronomical sky surveys like the SDSS. In contrast to human scanners, deep models can produce continuous-valued classification certainty ratings that can be tweaked for maximum recall and precision. Furthermore, they can handle the enormous data throughput generated by the different sky surveys.

I. COMPARISON WITH OTHER EXISTING MODELS

Most previous research work related to this paper uses standard supervised learning techniques to achieve the goal of automatic classification. The ML categorization of SDSS transient survey images is a baseline model for the proposed work. The same dataset was used in this research study, but several learning techniques were used, including (i) Random Forest (RF), (ii) k-Nearest Neighbors (k-NN), (iii) Adaboost, (iv) Support Vector Machine (SVM), (v) Easy Ensemble and (vi) Naïve Bayes (NB). The same dataset was used in this research study, but several learning techniques were used, including (1) RF, (2) KNN, (3) NB, and (4) SVM. And then match their performance using the same measures using DL-CNN and compare the proposed work to the past work. In the very different image data (g, r, I, z, u), they should also use the PCA algorithm to pull out features like shape, location, FWHM, and objects near a local object.

Our proposed model uses KPCA as the feature extraction model and the recommended Int-T2-FSVM classifier. The benchmark model achieved the results shown in Fig. 15, and it is evident that none of the other models improved more than our proposed model.

IV. THREATS TO VALIDATE

In this section, we go over potential threats to our experiment and how we mitigated them. Validity assesses whether

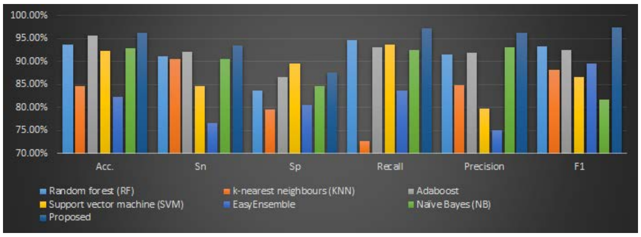

FIGURE 15. Comparison with existing models.

experiment results adhere to the specifications provided in the study procedure.

A. THREATS TO INTERNET VALIDITY

If an experimental condition has an effect or not, and if there is adequate data to back the assertion, then it is said to be having internal validity. The primary threat to internal validity in our case is the SMOTE+ENN that we utilised, which may be reasonable for our dataset. However, there are many more effective models available, such as Weighted SVM and Deep SMOTE, and using such a model might have produced considerably better outcomes.

B. THREATS TO EXTERNAL VALIDITY

The applicability of the results of the experiment is referred to as external validity. We used the Int-T2-FSVM to classify astrophysical objects by using multiple Int-T2-FSVMs. This was necessary for the application of this study, which was to tell the difference between three types of astrophysical objects (stars, galaxies and QSO). A significant barrier to the experiment's success was the lack of processing power, which prevented the model from being trained from scratch to more effectively learn the dataset's astronomical labels. It's possible that the experiment's findings won't translate accurately from experimental categories to real ones.

C. CONSTRUCT VALIDITY

If an experimental variable's operational definition reflects its theoretical meaning, then it is considered to have construct validity. The SDSS dataset was used in our experiment to evaluate the effectiveness of the suggested model. The entire dataset was split into two sections, with one serving as the first training set and the other as the initial test set. We separated the dataset into a 25% test set and a 75% training set in order to conduct fair comparisons (30% is used for cross-validation). However, for the classification model with SMOTE + ENN, we only achieved a sensitivity and specificity performance of 88.42% and 87.63%, respectively. Without SMOTE + ENN, the results were even worse.

V. CONCLUSION

Classifying stellar has always been challenging, given the enormous volume of data. The existing classifiers run into issues like class imbalance and overfitting. In this paper, a framework to classify stellar objects such as "stars", "quasars", and "galaxies" from the SDSS dataset was presented. The model avoids the class imbalance by employing "SMOTE+ENN". The balanced dataset is subjected to "K-PCA" for feature extraction. The extracted features are fed to the proposed classifier "Int-T2-FSVM". The model employs an enhanced type reducer and inference engine to get better accuracy in classification. The experiment results show that the proposed model produces better accuracy and precision for the SDSS dataset when compared to other existing models.

REFERENCES

[1] *Astro 2020 Astronomy and Astrophysics Decadal Survey*. Accessed: Nov. 4, 2021. [Online]. Available: https://www.nap.edu/read/12951/chapter/1

[2] N. M. Ball, R. J. Brunner, A. D. Myers, and D. Tcheng, "Robust machine learning applied to astronomical data sets. I. Star–galaxy classification of the Sloan digital sky survey DR3 using decision trees," *Astrophys. J.*, vol. 650, no. 1, pp. 497–509, Oct. 2006.

[3] R. J. Brunner, S. G. Djorgovski, T. A. Prince, and A. S. Szalay, "Massive datasets in astronomy," in *Handbook of Massive Data Sets*. Boston, MA, USA: Springer, 2001.

[4] A. D'Isanto and K. L. Polsterer, "Photometric redshift estimation via deep learning-generalized and pre-classification-less, image based, fully probabilistic redshifts," *Astron. Astrophys.*, vol. 609, pp. 1–16, Jan. 2017.

[5] R. Akbani, S. Kwek, and N. Japkowicz, "Applying support vector machines to imbalanced datasets," in *Proc. Eur. Conf. Mach. Learn. (ECML)*, 2004, pp. 39–50.

[6] N. Japkowicz, "The class imbalance problem: Significance and strategies," in *Proc. Int. Conf. Artif. Intell.*, 2000, pp. 111–117.

[7] Y. Zhang and Y. Zhao, "Automated clustering algorithms for classification of astronomical objects," *Astrophys. J.*, vol. 422, pp. 1113–1121, Aug. 2004.

[8] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, no. 28, pp. 321–357, Jun. 2006.

[9] H. Han, W. Wang, and B. Mao, "Borderline-smote: A new over-sampling method in imbalanced data sets learning," in *Proc. ICIC*, 2005, pp. 878–887.

[10] R. K. Gulati, R. Gupta, P. Gothoskar, and S. Khobragade, "Stellar spectral classification using automated schemes," *Astrophys. J.*, vol. 426, p. 340, May 1994.

[11] H. P. Singh, R. K. Gulati, and R. Gupta, "Stellar spectral classification using principal component analysis and artificial neural networks," *Monthly Notices Roy. Astronomical Soc.*, vol. 295, pp. 312–318, Apr. 2002.

[12] R. Gupta, H. P. Singh, K. Volk, and S. Kwok, "Automated classification of 2000 bright *IRAS* sources," *Astrophys. J. Suppl. Ser.*, vol. 152, no. 2, p. 201, 2004.

[13] A. Bora, R. Gupta, H. P. Singh, and K. Duorah, "Automated star–galaxy segregation using spectral and integrated band data for TAUVEX/ASTROSAT satellite data pipeline," *New Astron.*, vol. 14, no. 8, pp. 649–653, Nov. 2009.

[14] M. Bazarghan and R. Gupta, "Automated classification of Sloan digital sky survey (SDSS) stellar spectra using artificial neural networks," *Astrophys. Space Sci.*, vol. 315, nos. 1–4, pp. 201–210, Jun. 2008.

[15] H. Li, F.-L. Chung, and S. Wang, "A SVM based classification method for homogeneous data," *Appl. Soft Comput.*, vol. 36, pp. 228–235, Nov. 2015.

[16] S. Datta and S. Das, "Near-Bayesian support vector machines for imbalanced data classification with equal or unequal misclassification costs," *Neural Netw.*, vol. 70, pp. 39–52, Oct. 2015.

[17] H.-Y. Liu, W. Wang, R. Wang, C. Tung, P. Wang, and I. Chang, "Image recognition and force measurement application in the humanoid robot imitation," *IEEE Trans. Instrum. Meas.*, vol. 61, no. 1, pp. 149–161, Jan. 2012.

[18] Y. Tian, Z. Qi, X. Ju, Y. Shi, and X. Liu, "Nonparallel support vector machines for pattern classification," *IEEE Trans. Cybern.*, vol. 44, no. 7, pp. 1067–1079, Jul. 2014.

[19] P. R. Fiorentin, C. A. L. Bailer-Jones, Y. S. Lee, T. C. Beers, T. Sivarani, R. Wilhelm, C. A. Prieto, and J. E. Norris, "Estimation of stellar atmospheric parameters from SDSS/SEGUE spectra," *Astronomy Astrophys.*, vol. 467, no. 3, pp. 1373–1387, Jun. 2007, doi: 10.1051/0004-6361:20077334.

[20] D.-W. Kim, P. Protopapas, Y.-I. Byun, C. Alcock, R. Khardon, and M. Trichas, "Quasi-stellar object selection algorithm using time variability and machine learning: Selection of 1620 quasi-stellar object candidates from MACHO Large magellanic cloud database," *Astrophys. J.*, vol. 735, no. 2, p. 68, Jul. 2011.

[21] Y. Bu, F. Chen, and J. Pan, "Stellar spectral subclasses classification based on isomap and SVM," *New Astron.*, vol. 28, pp. 35–43, Apr. 2014.

[22] B. Ishak, "A review of electromagnetic waves for thermonuclear fusion research, by Ernesto Mazzucato: Scope: Monograph. Level: Specialist," *Contemp. Phys.*, vol. 58, no. 3, pp. 265–267, Jul. 2017.

[23] M. Lochner, J. D. McEwen, H. V. Peiris, O. Lahav, and M. K. Winter, "Photometric supernova classification with machine learning," 2016, *arXiv:1603.00882*.

[24] T. Charnock and A. Moss, "supernovae: Photometric classification of supernovae," *Astrophys. Source Code Library*, vol. 2017, p. ascl:1705.017, May 2017.

[25] M. V. D. Santos, M. Quartin, and R. R. R. Reis, "On the cosmological performance of photometrically classified supernovae with machine learning," *Monthly Notices Roy. Astronomical Soc.*, vol. 497, no. 3, pp. 2974–2991, Sep. 2020.

[26] M. Carrillo, J. A. González, M. Gracia-Linares, and F. S. Guzmán, "Time series analysis of gravitational wave signals using neural networks," *J. Phys., Conf. Ser.*, vol. 654, Nov. 2015, Art. no. 012001.

[27] M. Bilicki *et al.*, "Photometric redshifts for the kilo-degree survey: Machine-learning analysis with artificial neural networks," *Astron. Astrophys.*, vol. 616, p. A69, Aug. 2018.

[28] A. Gauci, K. Adami, and J. Abela, "Machine learning for galaxy morphology classification," 2010, *arXiv:1005.0390*.

[29] D. D. Whitten *et al.*, "J-PLUS: Identification of low-metallicity stars with artificial neural networks using SPHINX," 2018, *arXiv:1811.02279*.

[30] E. C. Vasconcellos, R. R. Carvalho, R. R. Gal, F. L. LaBarbera, H. V. Capelato, H. F. Velho, M. Trevisan, and R. S. Ruiz, "Decision tree classifiers for star/galaxy separation," *Astronomical J.*, vol. 141, no. 6, p. 189, 2011.

[31] S. Alam, "The eleventh and twelfth data releases of the Sloan digital sky survey: Final data from SDSS-III," *Astrophys. J. Suppl. Ser.*, vol. 219, no. 1, p. 12, Jul. 2015.

[32] E. J. Kim and R. J. Brunner, "Star–galaxy classification using deep convolutional neural networks," *Monthly Notices Roy. Astronomical Soc.*, vol. 464, Oct. 2016, Art. no. stw2672.

[33] I. Sevilla-Noarbe, "Star–galaxy classification in the dark energy survey Y1 data set," *Monthly Notices Roy. Astronomical Soc.*, vol. 481, pp. 5451–5469, Dec. 2018.

[34] O. Castillo, *Type-2 Fuzzy Logic: Theory and Applications*. Berlin, Germany: Springer, 2008.

[35] C. Wu, X. Wang, D. Bai, and H. Zhang, "Fast incremental learning algorithm of SVM on KKT conditions," in *Proc. 6th Int. Conf. Fuzzy Syst. Knowl. Discovery*, 2009, pp. 551–554.

[36] R. Martínez, O. Castillo, and L. T. Aguilar, "Optimization of interval type-2 fuzzy logic controllers for a perturbed autonomous wheeled mobile robot using genetic algorithms," *Inf. Sci.*, vol. 179, no. 13, pp. 2158–2174, Jun. 2009.

[37] K. H. Cheng, "Hybrid learning-based neuro-fuzzy inference system: A new approach for system modeling," *Int. J. Syst. Sci.*, vol. 39, no. 6, pp. 583–600, 2008.

[38] O. Castillo, "Optimization of an interval type 2 fuzzy controller for an autonomous mobile robot using the particle swarm optimization algorithm," *Stud. Fuzziness Soft Comput.*, vol. 27, no. 2, pp. 173–180, 2012.

[39] M. H. F. Zarandi and R. Gamasaee, "Type-2 fuzzy hybrid expert system for prediction of tardiness in scheduling of steel continuous casting process," *Soft Comput.*, vol. 16, no. 2, pp. 1–16, 2012.

[40] M. Hanmandlu, O. P. Verma, N. K. Kumar, and M. Kulkarni, "A novel optimal fuzzy system for color image enhancement using bacterial foraging," *IEEE Trans. Instrum. Meas.*, vol. 58, no. 8, pp. 2867–2879, Aug. 2009.

[41] Y. Yan, G. Mauris, E. Trouve, and V. Pinel, "Fuzzy uncertainty representations of coseismic displacement measurements issued from SAR imagery," *IEEE Trans. Instrum. Meas.*, vol. 61, no. 5, pp. 1278–1286, May 2012.

[42] I. Tomek, "Two modifications of CNN," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-6, no. 11, pp. 769–772, Nov. 1976.

[43] 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.

[44] S. Mishra, C. N. Bhende, and B. K. Panigrahi, "Detection and classification of power quality disturbances using S-transform and probabilistic neural network," *IEEE Trans. Power Del.*, vol. 23, no. 1, pp. 280–287, Jan. 2008.

[45] Q. Liang and J. M. Mendel, "Interval type-2 fuzzy logic systems: Theory and design," *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 5, pp. 535–550, Oct. 2000.

[46] L.-X. Wang, "A new look at type-2 fuzzy sets and type-2 fuzzy logic systems," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 3, pp. 693–706, Jun. 2017.

[47] H.-J. Wu, Y.-L. Su, and S.-J. Lee, "A fast method for computing the centroid of a type-2 fuzzy set," *IEEE Trans. Syst., Man, Cybern., B, Cybern.*, vol. 42, no. 3, pp. 764–777, Jun. 2012.

[48] F. Topaloglu and H. Pehlivan, "Comparison of Mamdani type and Sugeno type fuzzy inference systems in wind power plant installations," in *Proc. 6th Int. Symp. Digit. Forensic Secur. (ISDFS)*, Mar. 2018, pp. 1–4.

[49] I. Salhi, A. Belattar, and S. Doubabi, "Takagi–Sugeno fuzzy modeling for three-phase micro-hydropower plant prototype," *Int. J. Hydrogen Energy*, vol. 42, no. 28, pp. 17782–17792, 2017.

[50] M. R. Blanton *et al.*, "Sloan digital sky survey IV: Mapping the Milky Way, nearby galaxies, and the distant universe," *Astronomical J.*, vol. 154, no. 1, p. 28, Jun. 2017.

[51] T. De, D. F. Burnet, and A. K. Chattopadhyay, "Clustering large number of extragalactic spectra of galaxies and quasars through canopies," *Commun. Statist.-Theory Methods*, vol. 45, no. 9, pp. 2638–2653, 2016, doi: 10.1080/03610926.2013.848286.

[52] D. Dablain, B. Krawczyk, and N. V. Chawla, "DeepSMOTE: Fusing deep learning and SMOTE for imbalanced data," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jan. 27, 2022, doi: 10.1109/TNNLS.2021.3136503.

[53] C. Huang, Y. Li, C. C. Loy, and X. Tang, "Learning deep representation for imbalanced classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5375–5384, doi: 10.1109/CVPR.2016.580.

[54] J. Zhai, J. Qi, and S. Zhang, "Binary imbalanced data classification based on modified D2GAN oversampling and classifier fusion," *IEEE Access*, vol. 8, pp. 169456–169469, 2020, doi: 10.1109/ACCESS.2020.3023949.

[55] V. A. Fajardo, D. Findlay, C. Jaiswal, X. Yin, R. Houmanfar, H. Xie, J. Liang, X. She, and D. B. Emerson, "On oversampling imbalanced data with deep conditional generative models," *Expert Syst. Appl.*, vol. 169, May 2021, Art. no. 114463.

ARODH LAL KARN received the M.S. degree in corporate management and the Ph.D. degree from the Department of Management Science and Engineering specializing in the interdisciplinary area of MIS and financial engineering, more specifically in high-frequency trading, from the Harbin Institute of Technology, China. He is currently working as an Assistant Professor with the Department of Financial and Actuarial Mathematics, School of Mathematics and Physics, Xi'an Jiaotong-Liverpool University (XJTLU), Suzhou, China. He has ten years of experience in the teaching/research/industry. He has published papers in more than 30 indexed journals and proceedings. His research interests include economics of information systems, big data, cloud architecture, electronic business, and the IoT. He guided several UG and PG students in IS stream. He is a member of editorial board in many reputed journals, like *Physica A: Statistical Mechanics and its Applications* (Elsevier), a Certified Member of AEIC, which is strategic partner of world-leading publishing houses, like Springer, SAGE, and ATLANTIS, and Wiley. He received the award for Huang Tiyun Innovation Research at the Department of MIS, Harbin Institute of Technology. He delivered guest lectures at various reputed institutions and universities inside and outside China. He has published two monographs and several book chapters relating to financial engineering and IS.

CARLOS ANDRÉS TAVERA ROMERO (Member, IEEE) received the degree in system engineering and the Ph.D. degree in computer science engineering from the Universidad del Valle, Cali. Since 1998, he has been a Teacher and a Project Tutor for undergraduate students and a Tutor for master's and Ph.D. students' projects. He is currently a full-time Professor at the Universidad Santiago de Cali, Cali, Colombia, and an Information Systems Developer with various registered products. He is also the Leader of information systems development research line at the COMBA R&D Laboratory, Universidad Santiago de Cali.

SUDHAKAR SENGAN (Member, IEEE) received the M.E. degree from the Faculty of Computer Science and Engineering, Anna University, Chennai, Tamil Nadu, India, in 2007, and the Ph.D. degree in information and communication engineering from Anna University. He is currently working as a Professor and the Director of International Relations, Department of Computer Science and Engineering, PSN College of Engineering and Technology (Autonomous), Tirunelveli, Tamil Nadu. He has 20 years of experience in teaching/research/industry. He has published papers in 140 international journals, 20 international conferences, and ten national conferences. His research interests include security, MANET, the IoT, cloud computing, and machine learning. He has filed 20 Indian and three international patents in various fields of interest. He guided more than 100 Projects for UG and PG students in engineering streams. He is the Recognized Research Supervisor at the Faculty of Information and Communication Engineering, Anna University. He received the award of Honorary Doctorate (Doctor of Letters-D.LITT.) from International Economics University; SAARC Countries in Education and Students Empowerment, in April 2017. He has published three textbooks for Anna University, Chennai Syllabus. He is a member of various professional bodies, like MISTE, MIEEE, MIAENG, MIACSIT, MICST, MIE, and MIEDRC.

ABOLFAZL MEHBODNIYA (Senior Member, IEEE) received the Ph.D. degree from INRS-EMT, University of Quebec, Montreal, Canada, in 2010. He is currently an Associate Professor and the Head of ECE Department, Kuwait College of Science and Technology. Before coming to KCST, he worked as a Marie-Curie Senior Research Fellow with University College Dublin, Ireland, and prior to that, he worked as an Assistant Professor with Tohoku University, Japan, and as a Research Scientist in Advanced Telecommunication Research (ATR) International, Kyoto, Japan. His research interests are in the field of communications engineering, the IoT, and artificial intelligence in wireless networks and real-world applications. He is a Senior Member of IEICE. He was a recipient of numerous awards, including the JSPS Young Faculty Startup Grant, the KDDI Foundation Grant, the Japan Radio Communications Society (RCS) Active Researcher Award, the European Commission Marie Skłodowska-Curie Fellowship, and the NSERC Visiting Fellowships in Canadian Government Laboratories.

JULIAN L. WEBBER (Senior Member, IEEE) received the Ph.D. degree from Bristol University, in 2004. Following postdoctoral research on wireless communications at Hokkaido University, in 2007, he joined the Advanced Telecommunications Research Institute International, Kyoto, in 2012. He has been a Visiting Researcher and Research Assistant Professor with Osaka University, since 2018. He is currently an Associate Professor with the Department of Electronics and Communications Engineering, Kuwait College of Science and Technology. His research interests include communications engineering, machine learning, signal and image processing, and emphasizing real-time implementation. He is a member of the IEICE.

DENIS A. PUSTOKHIN received the Ph.D. degree in logistics and supply chain management from the State University of Management, Moscow, Russia. He is currently an Associate Professor with the State University of Management. He has published over 40 conferences and journal papers. His research interests include enterprise logistics planning, artificial intelligence, big data, the Internet of Things, and reverse logistics network design.

FRANK-DETLEF WENDE has over 20 years of combined industry and teaching experience. He is currently a Full Professor and the Head of the Department of Logistics and Marketing, Faculty of Economics and Business, Financial University under the Government of the Russian Federation. He has published over 30 conferences and journal papers. His research interests include logistics, big data technology and applications, information management, and the Internet of Things. He has completed various consultancy projects from various funding agencies and industries.