

## Weighted Feature Based Imperialist Competitive Algorithm With Ensemble Learning For Imbalanced Data Classification

D. Kavitha<sup>1</sup> and Dr. R. Ramkumar<sup>2</sup>

<sup>1</sup>Research Scholar, Computer Science, Nandha Arts and Science College, Erode.

E-mail: kavitha.d@nandhaarts.org

<sup>2</sup>School of Computer Science, VET Institute of Arts and Science (Co-education) College, Erode.

**Article History:** Received: 11 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 16 April 2021

**Abstract :** In recent trends, discovering classification knowledge from imbalanced data received has grabbed much interest by many researchers. Data set imbalancing might occur if any one class comprises considerably smaller number of examples than remaining classes. Various application greatly necessitates minority class which is regarded as quite interesting aspect. The imbalanced classes' distribution set up a challenge for standard learning algorithms like k-Nearest Neighbor (KNN), Support Vector Machine (SVM) and Neural Network (NN), subsequently biasing is done towards majority classes. An Improved Coefficient vector based Grey Wolf Optimization (ICGWO) Algorithm with ensemble classifier is deployed in previous approaches for classification. Nonetheless, adequate outcomes in terms of accuracy and execution time cannot be achieved. A Weighted Feature based Imperialist Competitive Algorithm (WFICA) with Ensemble Learning (EL) for imbalanced data classification is chiefly suggested for mitigating this issue. Primarily, normalization scheme is exploited for data transformation from different scales to an identical scale through Z-score normalization technique. Synthetic Minority Oversampling TEchnique (SMOTE) with Locally Linear Embedding (LLE) algorithm is deployed for oversampling process. Weighted Feature based Imperialist Competitive Algorithm (WFICA) is utilized for Optimal features selection which is done for classification accuracy enhancement. Ensemble Learning (EL) incorporated with Improved Bidirectional Long Short Term Memory (IBi-LSTM), Enhanced Weighted Support Vector Machine (EWSVM) and k-Nearest Neighbour (k-NN) classifiers is employed on selected features basis for performing classification. The suggested methodology is validated through experimental result and improved performance is attained when contrasted with prevailing system pertaining to accuracy, precision, recall and f-measure.

**Keywords:** Imbalanced data classification, oversampling, Ensemble Learning (EL) and Weighted Feature based Imperialist Competitive Algorithm (WFICA)

### 1. INTRODUCTION

The learning process usually takes place with class imbalance distribution in various applications which is positive or minority class and inadequately characterized [1-3]. Basically, it refers to number of examples from positive class (minority) is considerably lesser than number of negative class examples (majority). It is probably predicted as rare occurrences undiscovered or ignored, or assumed as noise or outliers when rare examples are rarely exist ensuing in further positive class (minority) misclassifications in contradiction to predominant class.

Most significantly lesser training examples in one class associated to other class leads to class imbalance generally. The class imbalance distribution might happen in binary circumstances; 1) when class imbalance is an intrinsic issue or it ensues unsurprisingly. A naturally imbalanced class distribution ensues during credit card fraud or in infrequent disease recognition [4]. One more circumstances is 2) when data is not certainly imbalanced, as a substitute it is also costly for acquiring such data for minority class learning caused by cost, confidentiality and tremendous strength for obtaining well-represented data set, similar to a very infrequent failure occurrence of a space-shuttle. Class imbalance encompasses enormous complications in learning, together with imbalanced class distribution, training sample size, class overlapping and small disjuncts.

The class imbalance classification has been emphasized recently through various studies [5]. There are several domain applications like fault diagnosis [6], anomaly detection, medical diagnosis [7-8], detection of oil spillage in satellite images, face recognition [9], text classification, protein sequence detection and many others in class imbalance distribution. The class imbalance problem substantial dares along with its recurrent occurrence in pattern recognition and data mining real-world applications have immersed a lot of researchers. Due to which two workshops committed to researches in addressing class imbalance concerns were presented at AAAI 2000 and ICML 2003 [10] respectively.

Classification is regarded as a substantial task for imbalanced dataset. A classification learning algorithms range such as Decision Tree (DT), Back propagation Neural Network (BPNN), Bayesian network, nearest neighbor, Support Vector Machines (SVM) [11], besides recently described associative classification, have been well-built and

effectively useful to several application domains. Nevertheless, data set imbalanced class distribution has chance upon a severe challenge to most classifier learning algorithms supposing a relatively balanced distribution.

## 2. LITERATURE SURVEY

Ohsaki et al (2017) performed classification using Confusion-Matrix-based Kernel LOGistic Regression (CM-KLOGR) through which several evaluation criteria harmonic mean like sensitivity, positive predictive value, and others negatives need to be derived from confusion matrix. On basis of KLOGR, along with Minimum Classification Error and Generalized Probabilistic Descent (MCE/GPD) learning are greatly utilized for optimization in consistent manner. The harmonic mean, KLOGR, and MCE/GPD, CM-KLOGR benefits chiefly enhances multifaceted performances in a well-proportioned approach. The suggested system attains improved effectiveness which is validated by experiments using benchmark imbalanced datasets [12].

Lu et al (2019) suggested an approach for imbalanced data classification using Improved Weighted- Extreme Learning Machine (IW-ELM). The voting technique integration into weighted Extreme Learning Machine (weighted ELM), mainly comprises three key steps namely training weighted ELM classifiers, eradicating unusable classifiers for obtaining proper classifiers to accomplish voting, and lastly obtaining classification outcome on basis of majority voting. The suggested system is validated through simulations on various real world imbalanced datasets with a number of imbalance ratios which outclasses weighted ELM and other correlated classification algorithms [13].

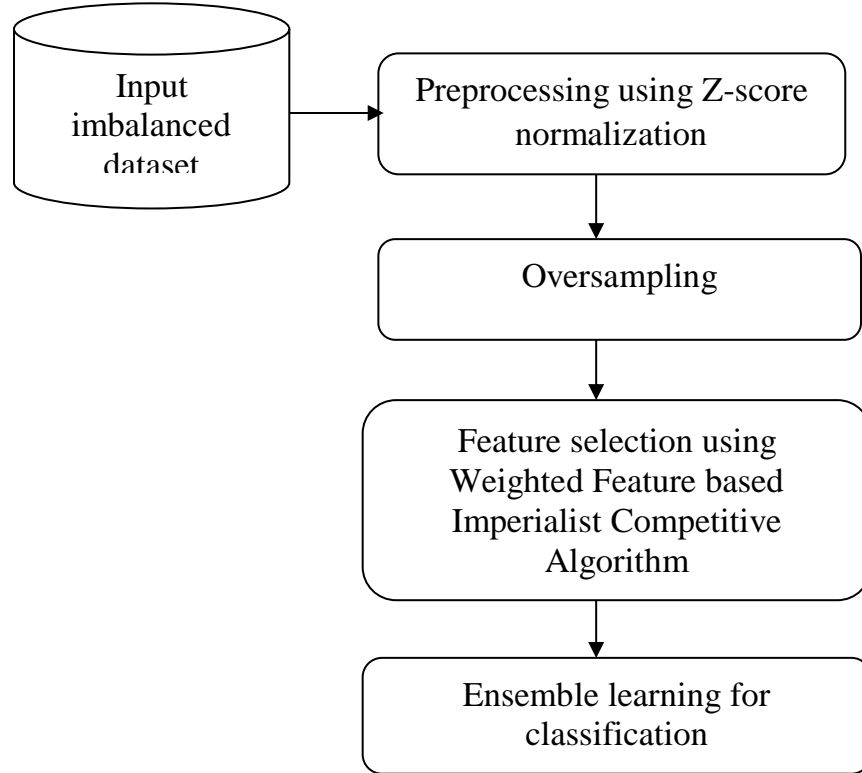
Liu et al (2019) utilized evolutionary under-sampling and feature selection combination for designing an ensemble classification system. Bootstrap technique is deployed for many sample subsets generation in original data. V-statistic is established for imbalanced data distribution measurement besides it is considered as genetic algorithm optimization objective for under-sampling sample subsets. Furthermore,  $F_1$  and G mean indicators are considered as two optimization objectives as well multi-objective ant colony optimization algorithm is exploited for resampled data feature selection for an ensemble system construction. Ten low-dimensional and four high-dimensional typical imbalanced datasets are utilized for experimentation and six conventional algorithms and four measures are engaged for a reasonable comparison. It is thereby revealed that improved classification performance is attained comparatively particularly for the high-dimensional imbalanced data [14].

Thaher et al (2019) established a proficient Software Fault Prediction (SFP) model on basis of wrapper feature selection method in combination with Synthetic Minority Oversampling Technique (SMOTE) with intention of exploiting learning model prediction accuracy. The search approach in wrapper FS technique is done by recent optimization algorithm, Queuing Search Algorithm (QSA) binary variant. The designed model performance assessment is done on 14 real-world benchmarks from PROMISE repository pertaining to three assessment methods; sensitivity, specificity, and Area under Curve (AUC). Investigational outcomes expose SMOTE technique a positive influence in enhancing prediction accomplishment in a extremely imbalanced data. Furthermore, binary QSA (BQSA) expresses a superior efficiency on datasets 64.28% in managing FS issues. The BQSA and SMOTE combination attained acceptable AUC outcomes (66.47-87.12%) [15].

Xu (2016) attained for imbalanced data classification by designing a Maximum Margin of Twin Spheres Support Vector Machine (MMTSSVM) which necessitates for finding homocentric spheres. The small sphere captures as several samples in majority class as probable; instead, large sphere pushes out most samples in minority class through increasing margin amid two homocentric spheres. MMTSSVM encompasses a QPP and a linear programming issue as opposed to a QPPs pair as in traditional TSVM or an enormous QPP in SVM, thus it significantly upsurges computational speed. More prominently, MMTSSVM evades matrix inverse operation. Experimental outcomes on nine benchmark datasets validate the suggested MMTSSVM approaches effectiveness. Lastly, MMTSSVM is utilized into Alzheimer's disease medical experiment and likewise obtaining superior experimental outcome [16].

## 3. PROPOSED METHODOLOGY

A Weighted Feature based Imperialist Competitive Algorithm (WFICA) with Ensemble Learning (EL) is greatly suggested for imbalanced data classification. The following are the steps involved in suggested approach: preprocessing, oversampling, feature selection and classification steps. The suggested research flow diagram is presented in figure 1.



**Figure 1: Flow diagram of the proposed work**

### 3.1 Dataset

Two benchmark datasets such as statlog (ladsat satellite) and letter recognition datasets obtained from UCI machine learning repository are greatly utilized for testing suggested approach performance. The statlog dataset mainly comprises multi-spectral pixels values in  $3 \times 3$  neighborhoods in a satellite image, besides classification is related with central pixel in every neighborhood. There are 6435 instances with 36 features in dataset and these features are numeric values ranging from 0 to 255 which are considered as 6-class classification problem. For multi-class dataset, least frequent class label are selected as minority and remaining as majority. Therefore, class 4 is treated as minority class and left over classes are pooled into one class as majority one.

The letter recognition dataset comprises 20 000 instances and 16 features and instances are integer values. The dataset depends on 16 features for categorizing 26 capital English alphabets.

### 3.2 Preprocessing

Data preprocessing is one amid furthest data mining phases for enhancing machine learning algorithms accuracy. Normalization usually occurs on all data before training and testing which is used for ensuring data avoidance overwhelmed with each other. The normalization scheme is exploited for data transformation from different scales to an identical scale. In Z-score normalization method, normalizing values are done based on feature A mean and standard deviation. The formula is given by:

$$v' = \frac{v - \bar{A}}{\sigma_A} \quad (1)$$

Where,

$v'$ ,  $v$  - new and old of each entry in data respectively

$\sigma_A$ ,  $\bar{A}$  - standard deviation and mean of A respectively

### 3.3 Oversampling Technique

Oversampling is performed on normalized datasets which are considered as input for which Synthetic Minority Oversampling Technique (SMOTE) with Locally Linear Embedding (LLE) algorithm is greatly utilized. The operational issue is mitigated by using SMOTE which is a potent methodology. A novel methodology is exploited for enhancing traditional SMOTE algorithm via integrating Locally Linear Embedding algorithm (LLE). Initially high-dimensional data mapping into low-dimensional space is done via LLE algorithm, where input data is further separable, besides oversampling is achieved through SMOTE. For a specified dataset  $X = \{x_1, x_2, \dots, x_N\}$  in a d-

dimensional space  $R^d$ , LLE algorithm finds an l-dimensional dataset Y in  $R^l$ . Practically LLE algorithm implementation can be done in three steps: :

- (1) Construct a k-NN graph  $G_{kNN}(X)$  for X: for every  $x_i \in X$ , its k nearest neighbor is signified as  $X_{kNN}(x_i)$
- (2) Evaluate weight matrix W such that  $x_i$  is best linearly spanned via  $X_{kNN}(x_i)$
- (3) Extract low-dimensional data Y

Synthetic Minority Oversampling TEchnique (SMOTE) is performed subsequently which comprises of two phases.

In first stage, k nearest neighbors are obtained using Euclidean distance computation from every minority data with respect to all other minority data besides storing in ascending order, and then k lowest distance data are considered as nearest neighbors (kNN). Euclidean distance amid one minority data (x) and another minority data (y) from first attribute to n (maximum number of attributes) is stated in Formula (2)

$$d(x,y)=\sqrt{\sum_{a=1}^n(x_a - y_a)^2} \quad (2)$$

In second stage, synthetic data generation is done via interpolation technique amid two minority data. One of its kNN randomization is done for candidates in synthetic data generation process. Subsequently, original minor data (x) and one selected candidate (y) will be used for new synthetic data generation amid x and y. Synthetic data formula amid x and y for a-th attribute is stated in formula (3)

$$\text{SyntheticData}_a(x, y) = x_a + r \cdot (x_a - y_a) \text{ for } 0 \leq r \leq 1 \quad (3)$$

Where,

r- random number amid 0 and 1

The above formula is useful for n features. The process repetition is done till desired synthetic data amount is attained.

### 3.4 Weighted Feature based Imperialist Competitive Algorithm (WFICA)

Selection of optimal features is carried out through Weighted Feature based Imperialist Competitive Algorithm (WFICA), through which the classifier's accuracy level is enhanced. Being known as a new evolutionary algorithm and inspired by imperialist competitive, the Imperialist Competitive Algorithm (ICA) proves its significance in the optimizations. ICA can be considered as a solid approach, since it is highly influenced by the imperialism concept that tends to expand the rule and supremacy of a governance out of its boundaries. According to this algorithm, the process begins with a population that is defined as number of features. Among population, a few optimal features are chosen and taken as the imperialist features. Remaining of population is split as colonies (features) across considered imperialists. Subsequently, across all empires, the imperialistic competition starts, during which the inadequate empire who fails to augment its supremacy and defeated in this competition will be excluded from the competition. Consequently, each colony (feature) moves toward its corresponding imperialists (features) accompanied by the competition across empires. Eventually, all the features are possibly converge to a state of sole empire for the whole world by collapse system, and remaining countries turn out to be a part of that empire. Here, the expected solution is indicated by the solid empire.

#### 3.4.1. Generating initial empires

Optimization tends to determine best features. In this study, number of features are taken as population (country). As represented in following equation, number of features is notated as  $1 \times N$  array in an N-dimensional problem.

$$\text{Number of features} = (x_1, x_2, \dots, x_n), x_i \in \mathbb{R}, 1 \leq i \leq N \quad (4)$$

Features' classification accuracy refers to the cost value of features. Besides, utilization of multiple classifiers is allowed to perform the classification. For classification, this study prefers to utilize ensemble learning. The following equation expresses the cost function.

$$\text{Cost} = \frac{\text{Number of correctly classified features}}{\text{Total number of features}} \quad (5)$$

For them,  $N_{pop}$  needs to be generated. At the variables  $(x_1, x_2, \dots, x_n)$ , the  $f(x)$  is taken as the cost of each feature. Following that,

$$\text{cost} = f(\text{features}) = f(x_1, x_2, \dots, x_n). \quad (6)$$

To form the empires (imperialist),  $N_{imp}$  of the most powerful features is selected. Apparently, the remaining  $N_{col}$  of the population can be considered as colonies. At this point, the calculation of normalized cost of an imperialist is formulated as follows

$$C_n = c_n - \max_i \{c_i\} \quad (7)$$

Here, the cost of  $n^{\text{th}}$  imperialist is represented by  $c_n$  and its corresponding normalized cost is denoted by  $C_n$ . Then, each imperialist's normalized power is given by

$$P_n = \left\{ \frac{C_n}{\sum_{i=1}^{N_{imp}} C_i} \right\} \quad (8)$$

Therefore, the initial number of empire colonies can be expressed as follows,

$$\text{No. } C_n = \text{round}(p_n \cdot N_{col}) \quad (9)$$

In which, the initial number of colonies of n<sup>th</sup> empire is signified by No.  $C_n$ ; the total number of all colonies is indicated by  $N_{col}$ . The colonies for imperialists can be divided by arbitrarily selecting the No.  $C_n$  of colonies and giving them to the n<sup>th</sup> empire.

### 3.4.2. Moving the colonies of an empire toward the imperialist

Every colony (features) moves toward the imperialist (imperialist features) in the direction of x-units can be taken as the vector from colony to imperialist, where x is the random variable that accompanies uniform distribution as given by

$$x \sim U(0, \beta \times D), \quad \beta > 1 \quad (10)$$

Here, the distance amid features and imperialist is denoted by d. The colony gets closer to the imperialist through  $\beta$ .

### 3.4.3 Revolution

In an empire, numerous features are switched by the same quantity of new countries generated, in each iteration step. The process of generating some new countries and switching them by some colonies belongs to that empire is randomly performed. For switching the number of colonies of the empire with the same number of newly generated features, the following expression is applied.

$$N.R.C = \text{round}\{\text{Revolution Rate} \times \text{No.}(\text{The colonies of } empire_n)\} \quad (11)$$

Here, the number of revolutionary colonies is signified by N.R.C. Besides, it helps enhancing the global convergence of the ICA and prohibits from trapped by local minimum.

### 3.4.4 Exchanging positions of the imperialist and a colony

When a colony moves, it has a possibility of accessing to an optimal position that is better than the imperialist. Hence, the imperialist makes a move towards the position of that colony and vice-versa.

### 3.4.5. Total power of an empire

As represented by the below equation, the empire's total power relies on all colonies owned by the empire.

$$T \cdot C_n = \text{cost}(imperialist_n) + \xi \cdot \text{mean}(\text{cost}(\text{colonies of } empire_n)) \quad (12)$$

Here, a position coefficient is signified by  $\xi$ .

For enhancing the accuracy of the classification task, the weight value of the feature is added in this proposed work. Hence, the equation (14) is rewritten as follows

$$T \cdot C_n = \text{cost}(imperialist_n) + \xi \cdot W_F \cdot \text{mean}(\text{cost}(\text{colonies of } empire_n)) \quad (13)$$

### 3.4.6. Imperialistic competition

Each of the empires are in competition, where they compete each other for taking control of colonies of others. Consequently, the power of inadequate empires starts decreasing and the efficient empires' power gradually rises. For attaining this objective, the each empire's possession probability is identified on the basis of its total power. Then, estimation of normalized total cost is given by,

$$N \cdot T \cdot C_n = T \cdot C_n - \max\{T \cdot C_i\} \quad (14)$$

Here, the total cost is represented by  $T \cdot C_n$ ; the normalized total cost of n<sup>th</sup> empire is denoted by  $N \cdot T \cdot C_n$ . At this point, the computation of each empire's possession probability is given by

$$P_n = \left\{ \frac{N \cdot T \cdot C_n}{\sum_{i=1}^{N_{imp}} N \cdot T \cdot C_i} \right\} \quad (15)$$

In accordance with empires' possession probability, the mentioned colonies among empires can be divided. The formulation of vector P is given by

$$P = [p_{p1}, p_{p2}, p_{p3}, \dots, p_{p_{N_{imp}}}] \quad (16)$$

Moreover, the subsequent equation expresses the vector R accompanied by uniformly distributed elements

$$R = [r_1, r_2, r_3, \dots, r_{N_{imp}}] \quad r_1, r_2, r_3, \dots, r_{N_{imp}} \sim U(0,1) \quad (17)$$

Eventually, vector D can be obtained through,

$$D = P - R = [p_{p1} - r_1, p_{p2} - r_2, p_{p3} - r_3, \dots, p_{p_{N_{imp}}} - r_{N_{imp}}] \quad (18)$$

The mentioned colonies are handed to an empire by the elements of D, and their corresponding index in D is higher.

### 3.4.7. The eliminated empire

If all the colonies of an empire is lost by that empire, it will collapse and turn out to be one among the remaining colonies.

### 3.4.8. Convergence

Ultimately, highly robust empire will be obtained regardless of competitors, besides each colony falls under the rule of this solo empire. As such, the same costs exist in each colony which is similar to the one possessed by the unique empire. That is to say, all the colonies and their corresponding unique empire have no difference. By attaining this ideal world, the proposed algorithm is terminated.

#### Algorithm 1: Imperialist Competitive Algorithm (ICA)

**Input:** Number of features in dataset

**Output:** Best features

**Step 1:** Assign classification accuracy as objective function:  $f(x)$ ,  $x=(x_1, x_2, \dots, x_d)$

**Step 2:** Initialize the number of features

**Step 3:** Create a few random solution in the search space and generate initial empires.

**Step 3:** Assimilation: features move towards imperialist states in different directions.

**Step 4:** Revolution: Random changes befall in the features of some countries.

**Step 5:** Exchange of position within a features and Imperialist features. A features that accompanies optimal position than the imperialist features, can easily take the control of empire by switching the existing imperialist features.

**Step 6:** Imperialistic competition: All imperialists compete to take possession of colonies of each other.

**Step 7:** Eradicate the powerless empires. Inadequate empires slowly lose their supremacy and get abolished.

**Step 8:** If the stop condition is fulfilled, terminate, else go to 2.

**Step 9:** Selected features

**Step 10:** Terminate

### 3.5 Ensemble Learning (EL)

Based on the selected features the classification is performed with the help of Ensemble Learning (EL) which includes Improved Bidirectional Long Short Term Memory (IBi-LSTM), Enhanced Weighted Support Vector Machine (EWSVM) and k-Nearest Neighbour (k-NN) classifiers.

#### 3.5.1 Improved Bidirectional Long Short Term Memory (IBi-LSTM)

The LSTM network is a special type of improved RNN. The LSTM cell unit undergoes a subtle combination of forgetting gates, input gates, and output gates, and introduces cell-state connections in the RNN network to resolve the problems of gradient disappearance or explosion during deep propagation. It is often used to deal with long-term dependent time series. The structure of a basic LSTM cell unit is shown in Figure 2.

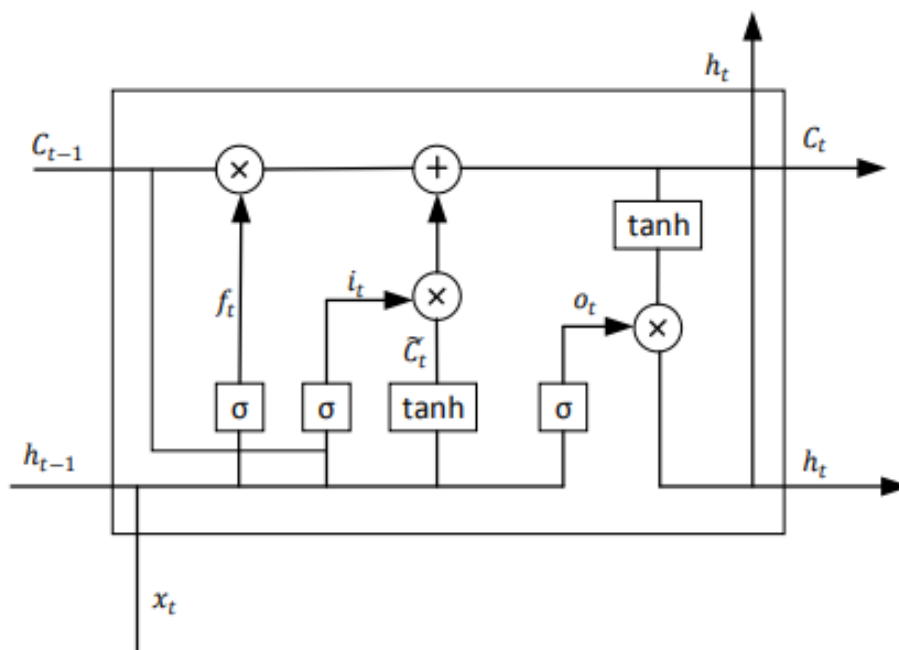


Figure 2. LSTM cell structure

Unlike the simplistic approach of RNN, post-processing of training and learning done through the forget gate  $f_t$ , the input gate  $i_t$ , and the output gate  $o_t$ , the LSTM includes flexible options, besides it does not prefer to solely utilize the cell state  $c_t$  at the previous moment as the input of the current cell like RNN does. The subsequent steps are followed at the time of processing the information inside LSTM cell unit.

As expressed by the following equation, the hidden state  $h_{t-1}$  at the previous moment, the current input  $x_t$ , and the state information  $C_{t-1}$  of the cell's internal memory unit are received by each gate of the cell, from which they start performing various operations, and ensure the possibility of activation through logic function. By using a non-linear transformation, the forgetting gate delivers the output that sets within the range of 0 and 1, and identifies the impact of the information possessed by the last memory cell  $C_{t-1}$  on the current memory state. The updation of State information  $C_t$  includes two parts, i.e. i) determining through forgetting gate; ii) determining through input gate and inputs at time  $t$ . In other words, the state that requires to get updated is identified through a part of the input gate, whereas the other part helps the updated information to reach the cell state. Ultimately, the non-linearly activated  $\tanh(\cdot)$  and the output gate information helps to identify the state  $h_t$  of the unit, the output at time  $t$  and the input hidden state at time  $t + 1$ .

$$f_t = \sigma(W_{fx}x_t + W_{fh}h_{t-1} + W_{fc}C_{t-1} + b_f) \quad (19)$$

$$i_t = \sigma(W_{ix}x_t + W_{ih}h_{t-1} + W_{ic}C_{t-1} + b_i) \quad (20)$$

$$C_t = f_t * C_{t-1} + i_t * \tanh(W_{cx}x_t + W_{ch}h_{t-1} + b_c) \quad (21)$$

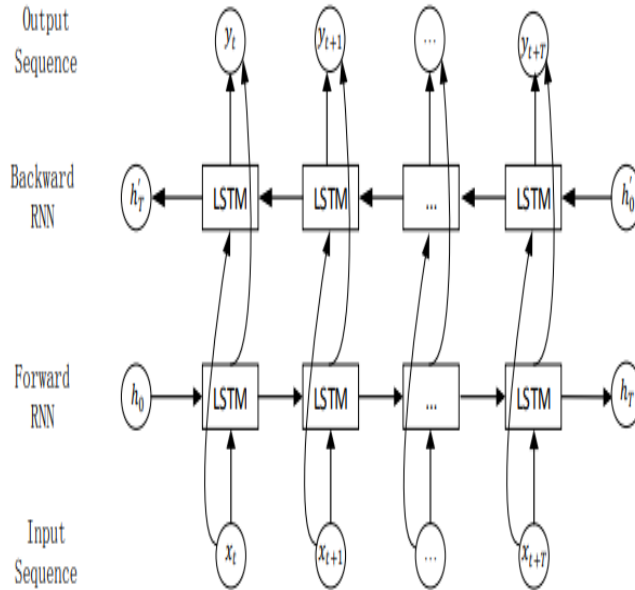
$$o_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1} + W_{oc}C_t + b_o) \quad (22)$$

$$h_t = o_t * \tanh(C_t) \quad (23)$$

In the aforementioned expression, the sigmoid activation function is denoted by  $\sigma(\cdot)$ ; the tanh activation function is signified by  $\tanh(\cdot)$ ; the connection weight matrix and the bias vector within the forgetting gate, input gate, output gate and different input quantities are represented by  $W$  and  $b$ . Generally, these are unknown parameters and need to be learned in the deep learning network.

### IBi-LSTM

By joining to a backward propagation LSTM network, the Bi-LSTM network gets enabled to handle the past information in forward propagation and take the future sequence information in a reverse recursive process. Consequently, the concurrent learning of forward and backward sequence rules can be done. In IBi-LSTM, the standard sigmoid and hyperbolic tangent (tanh) functions are replaced by softsign function due to its flatter curve and a slower descending derivative, and it is deployed as an internal function of IBi-LSTM in this proposed work. Besides, in the gate activation function, the efficient error reduction is attained, since the gate gets enabled to delete information as the standard sigmoid function is replaced by the softsign function.



**Figure. 3. IBi-LSTM neural network structure deployed in time direction**





**3.5.3K-nearest neighbor (k-NN)**

For classifying imbalanced data, the dataset are fed as an input. In accordance with the identicalness with neighbors, the dataset is efficiently classified by the simple clustering algorithm called K-Nearest Neighbor Classifier KNN. The number of data which are taken for classification process is signified by K. In the provided dataset, the k nearest samples is selected from the classified training data, and the class that takes highly representative samples for separation is determined. By employing Euclidean Distances (ED), the distances within database samples and test samples are identified in this model. The following equation expresses the Euclidean distance amid  $X = (x_1, x_2, x_3, \dots \cdot x_n)$  and  $Y = (y_1, y_2, y_3, \dots \cdot y_n)$ .

$$ED(x,y) = \sqrt{\sum_{j=1}^k (X_i - Y_i)^2} \quad (29)$$

Before performing the classification task, for the provided input, the output probabilities are averaged, which are from each classifier, i.e. Improved Bidirectional Long Short Term Memory (IBi-LSTM), Enhanced Weighted Support Vector Machine (EWSVM) and k-Nearest Neighbour (k-NN). As expressed below, the output  $S_i$  is averaged for output i.

$$S_i = \frac{1}{n} \sum_{j=1}^n r_j(i) \quad (30)$$

For a given selected features, the output i of classifier j is represented by  $r_j(i)$  above.

For each network, this method assigns a different weight. At the time of merging the result, the networks that accompanies minimum classification error is expected to possess a larger weight in the validation set. Prior to performing the prediction process, from each classifier, the output probabilities are multiplied by a weight  $\alpha$  for some given input pattern.

$$S_i = \sum_{j=1}^n \alpha_j r_j(i) \quad (31)$$

**4. EXPERIMENTAL RESULTS**

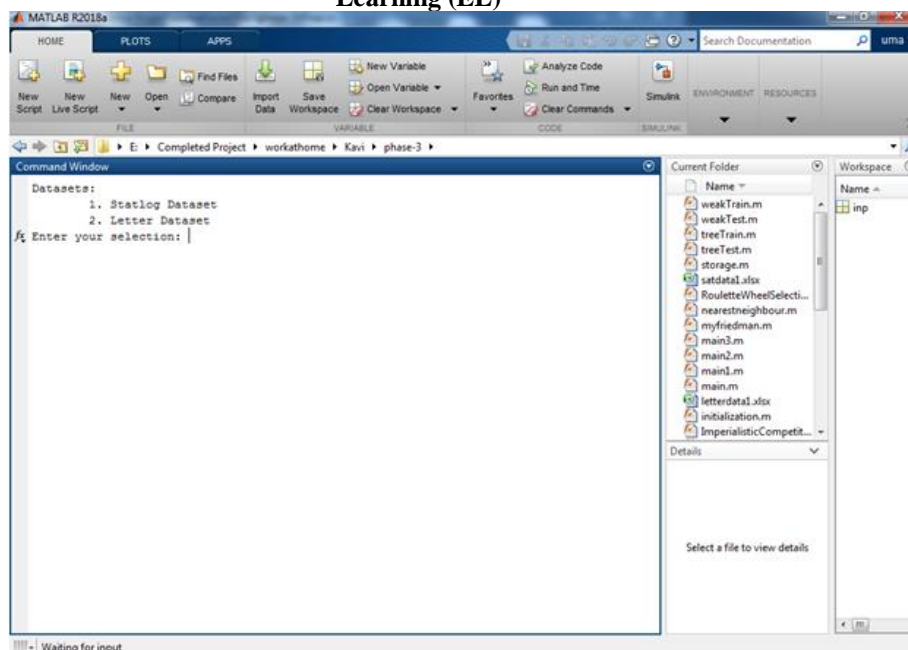
The overall research work is evaluated under MATLAB simulator, during which the performance of the proposed and existing methodologies have been assessed. From the UCI machine learning repository, the letter recognition and Statlog (ladsat satellite) datasets are gathered, and they are considered as a benchmark datasets for carrying out the performance evaluation for the proposed approach. Accordingly, the efficiency of the proposed method, namely Weighted Feature based Imperialist Competitive Algorithm (WFICA) with Ensemble Learning (EL) is obtained by comparing it with the performance of the prevailing methods, like Gini-Index Feature Selection (GI-FS), Weighted Gini Index –Feature Selection (WGI-FS), Modified Step size based Glowworm Swarm Optimization algorithm (MSGSO) and Improved Coefficient vector-based Grey Wolf Optimization (ICGWO) Algorithm. During the comparison, Accuracy, Precision, Recall, and F-measure are taken as the performance metrics.

**Table 1: Summary of Benchmark Datasets**

Dataset	Instances	Features	Imbalanced ratio
Statlog	6435	36	9.3
Letter	20 000	16	24.3–27.25



**Figure 5: Weighted Feature based Imperialist Competitive Algorithm (WFICA) with Ensemble Learning (EL)**



**Figure 6: Dataset selection**

In Figure. 5, Weighted Feature based Imperialist Competitive Algorithm (WFICA) with Ensemble Learning (EL) is depicted, which is the proposed method in this study. Figure. 6 illustrates the process of dataset selection as regards the classification of imbalanced data.

**Table 2: Performance analysis**

Methods	Statlog dataset				Letter dataset			
	Accuracy	Precision	Recall	F-measure	Accuracy	Precision	Recall	F-measure
GI-FS	79.89	80.56	78.23	79.41	79.89	80.56	78.23	79.41
WGI-FS	86.23	84.12	86.37	85.23	82.90	82.90	82.82	82.86
MSGSO	87.93	86.06	87.32	86.69	85.16	85.20	85.19	85.20
ICGWO	90.75	89.32	90.48	88.31	88.39	88.37	88.32	86.75
WFICA with EL	95.22	93.15	95.05	92.09	93.63	92.43	93.60	90.91

### 1. Accuracy

As a highly intuitive metric for measuring the performance, Accuracy defines the ratio between appropriately predicted observation and the overall observations.

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN} \quad (32)$$

Here,

*TP* - True Positive

*FN* - False Negative

*FP* - False Positive

*TN*- True Negative

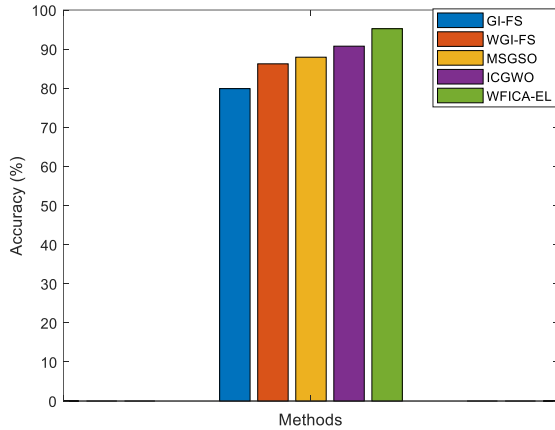


Figure 7 (a): Accuracy comparison for Statlog dataset

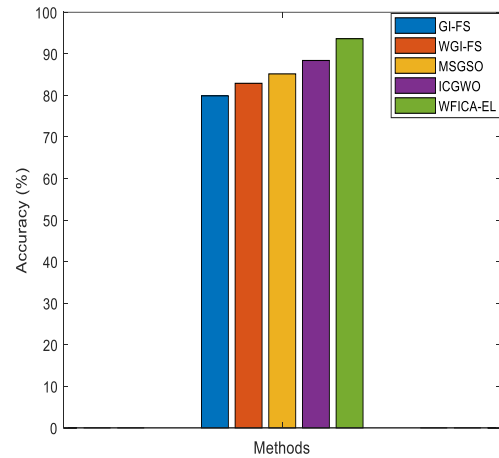


Figure 7 (b): Accuracy comparison for Letter Recognition dataset

Figure 7: Accuracy comparison

In Figure. 7, the Accuracy values obtained by the proposed and existing methods are individually compared, as regards Statlog and letter datasets. In the figure, the evaluated methods lie on X-axis; Y-axis stands for corresponding Accuracy rates, and the graphs represent that the proposed WFICA with EL algorithm is efficient among all methods compared. For Statlog dataset, the proposed algorithm obtains the accuracy rate of 95.22%, but the GI-FS, WGI-FS, MSGSO, and ICGWO solely obtain 79.89%, 86.23%, 87.93%, and 90.75%, correspondingly. For letter dataset, the proposed approach delivers the accuracy rate of 93.63%, whereas the present methods, such as GI-FS, WGI-FS, MSGSO and ICGWO can only obtain 79.89%, 82.90%, 85.16% and 88.39%, correspondingly.

**2. Precision**

Precision refers to the ratio between the number of observations that are appropriately predicted as positive, and the overall observations that are predicted as positive.

$$\text{Precision} = \frac{TP}{TP+FP} \tag{33}$$

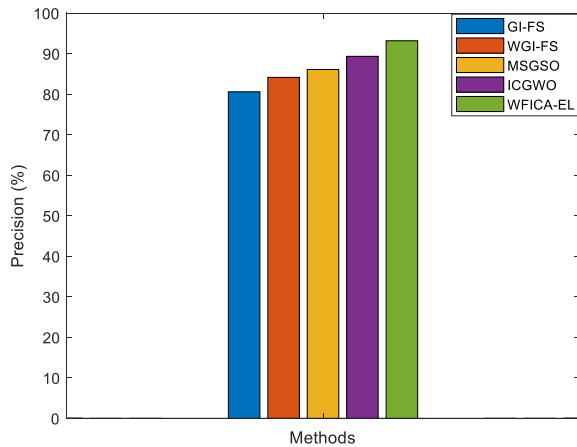


Figure 8 (a): Precision comparison for Statlog dataset

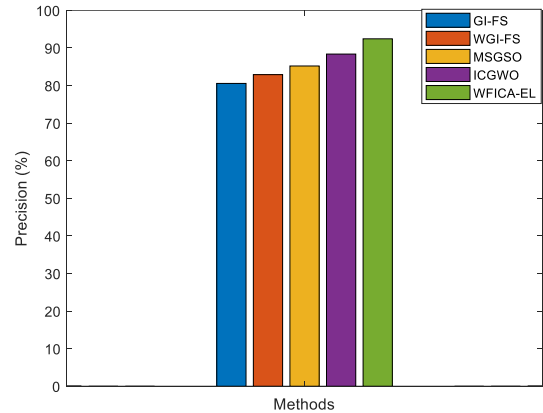


Figure 8 (b): Precision comparison for Letter Recognition dataset

Figure 8: Precision comparison

Figure. 8 individually compares the Precision rates of the proposed and existing methods as regards Statlog and letter datasets. In the figure, X-axis represents the methods evaluated; Y-axis stands for corresponding Precision rates. For Statlog dataset, the proposed WFICA with EL algorithm obtains 93.15% precision rate, but the GI-FS, WGI-FS, MSGSO, and ICGWO solely obtain 80.56%, 84.12%, 86.06%, and 89.32%, correspondingly. For letter dataset,

the proposed approach delivers the 92.43% precision, whereas the existing approaches, such as GI-FS, WGI-FS, MSGSO and ICGWO can only obtain 80.56%, 82.90%, 85.20% and 88.37%, correspondingly.

### 3. Recall

Recall defines the ratio between the number of observations that are appropriately predicted as positive, and the total number of observations in actual class- yes.

$$\text{Recall} = \frac{TP}{TP+FN} \quad (34)$$

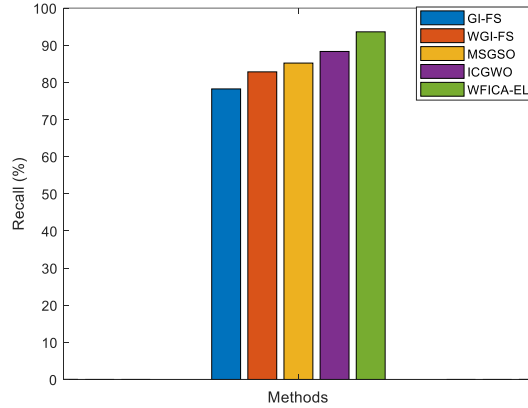
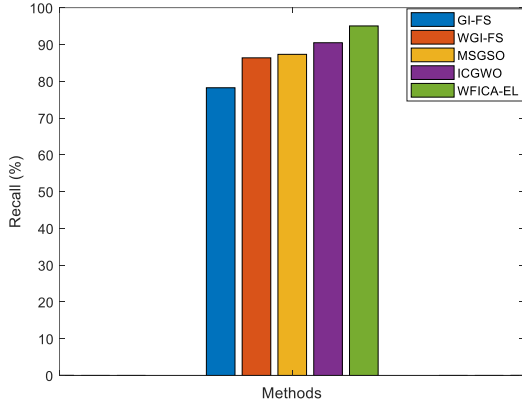


Figure 9 (a): Recall comparison for Statlog dataset

Figure 9(b): Recall comparison for Letter Recognition dataset

### Figure 9: Recall comparison

Figure. 9 individually compares the Recall rates of the proposed and existing methods for Statlog and letter datasets. In the figure, X-axis represents the methods evaluated; Y-axis stands for corresponding Recall rates. For Statlog dataset, the proposed WFICA with EL algorithm obtains 95.05% recall rate, whereas the GI-FS, WGI-FS, MSGSO, and ICGWO solely obtain 78.23%, 86.37%, 87.32%, and 90.48%, correspondingly. For letter dataset, the proposed approach delivers the 93.60% recall, whereas the existing approaches, such as GI-FS, WGI-FS, MSGSO and ICGWO obtain only 78.23%, 82.82%, 85.19%, and 88.32%, correspondingly.

### 4. F-measure

It refers to as the weighted average of Recall and Precision. Thus, both FPs (false positives) and FNs (false negatives) are taken in F-measure.

$$\text{F-measure} = 2 * \frac{(\text{Recall} * \text{Precision})}{(\text{Recall} + \text{Precision})} \quad (35)s$$

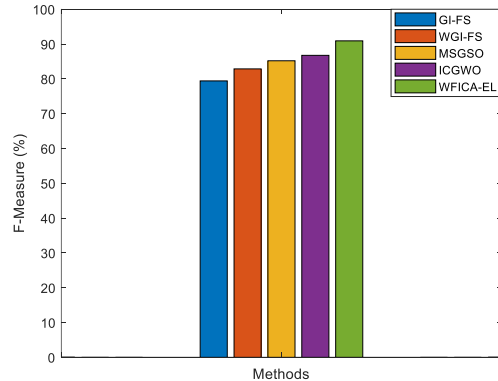
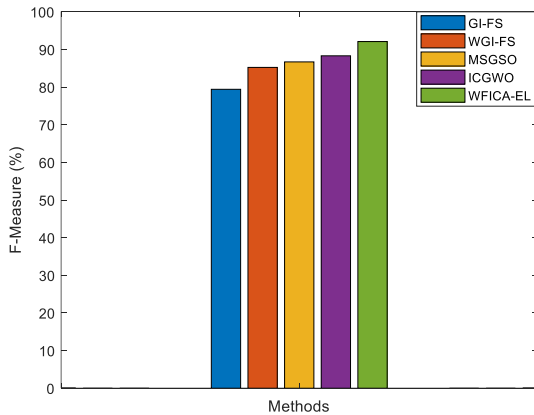


Figure 10 (a): F-measure comparison for Statlog dataset

Figure 10(b): F-measure comparison for Letter Recognition dataset

### Figure 10: F-measure comparison

In Figure. 10, the F-measure values obtained by the proposed and existing methods are individually compared, as regards Statlog and letter datasets. In the figure, the evaluated methods lie on X-axis; Y-axis stands for corresponding F-measure rates, and the graphs represent that the proposed WFICA with EL algorithm is efficient among all methods compared. For Statlog dataset, the proposed algorithm obtains the F-measure rate of 92.09%, whereas the GI-FS, WGI-FS, MSGSO, and ICGWO solely obtain 79.41%, 85.23%, 86.69%, and 88.31%, correspondingly. For letter dataset, the proposed approach delivers the F-measure rate of 90.91%, whereas the present methods, such as GI-FS, WGI-FS, MSGSO and ICGWO acquire only 79.41%, 82.86%, 85.20% and 86.75%, correspondingly.

## 5. CONCLUSION

The proposed system designed a Weighted Feature based Imperialist Competitive Algorithm (WFICA) with Ensemble Learning (EL) for effective imbalanced data classification. Initially z-score based normalization is performed to improve the classification accuracy. Then the Synthetic Minority Oversampling TEchnique (SMOTE) with Locally Linear Embedding (LLE) algorithm is applied on imbalanced data. In order to improve the accuracy, optimal features are selected by using Weighted Feature based Imperialist Competitive Algorithm (WFICA). According to the selected features, the classification is performed with the help of Ensemble Learning (EL) which includes Improved Bidirectional Long Short Term Memory (IBi-LSTM) , Enhanced Weighted Support Vector Machine (EWSVM) and k-Nearest Neighbour (k-NN) classifiers. The experimental results shows that the proposed system achieves better performance compared with the existing system in terms of accuracy, precision, recall and f-measure.

## References

1. Ali, H., Salleh, M. N. M., Saedudin, R., Hussain, K., & Mushtaq, M. F. (2019). Imbalance class problems in data mining: A review. *Indonesian Journal of Electrical Engineering and Computer Science*, 14(10.11591).
2. Hassib, E. M., El-Desouky, A. I., El-Kenawy, E. S. M., & El-Ghamrawy, S. M. (2019). An imbalanced big data mining framework for improving optimization algorithms performance. *IEEE Access*, 7, pp. 170774-170795.
3. Li, J., Fong, S., Wong, R. K., & Chu, V. W. (2018). Adaptive multi-objective swarm fusion for imbalanced data classification. *Information Fusion*, 39, pp.1-24.
4. Aida Ali, Siti Mariyam Shamsuddin and Anca Ralescu, "Classification with class imbalance problem: A review", *International journal of Advance Soft Computing . Application*, Vol. 5, No. 3, November 2013.
5. Kotsiantis, S., D. Kanellopoulos, and P. Pintelas, Handling imbalanced datasets: a review. *GESTS International Transactions on Computer Science and Engineering*, 2006. Vol 30(No 1): pp. 25-36.
6. Zhu, Z.-B. and Z.-H. Song, Fault diagnosis based on imbalance modified kernel Fisher discriminant analysis. *Chemical Engineering Research and Design*, 2010. 88(8): p. 936-951.
7. Jain, A., Ratnoo, S., & Kumar, D. (2017, August). Addressing class imbalance problem in medical diagnosis: A genetic algorithm approach. In *2017 International Conference on Information, Communication, Instrumentation and Control (ICICIC)* (pp. 1-8). IEEE.
8. Zhu, M., Xia, J., Jin, X., Yan, M., Cai, G., Yan, J., & Ning, G. (2018). Class weights random forest algorithm for processing class imbalanced medical data. *IEEE Access*, 6, 4641-4652.
9. Yi-Hung, L. and C. Yen-Ting. Total margin based adaptive fuzzy support vector machines for multiview face recognition. in *Systems, Man and Cybernetics, 2005 IEEE International Conference on*. 2005.
10. Chawla, N.V., N. Japkowicz, and A. Kotcz. in *Proc ICML 2003 Workshop on Learning from Imbalanced Data Sets*. 2003.
11. Richhariya, B., & Tanveer, M. (2018). A robust fuzzy least squares twin support vector machine for class imbalance learning. *Applied Soft Computing*, 71, 418-432.
12. Ohsaki, M., Wang, P., Matsuda, K., Katagiri, S., Watanabe, H., & Ralescu, A. (2017). Confusion-matrix-based kernel logistic regression for imbalanced data classification. *IEEE Transactions on Knowledge and Data Engineering*, 29(9), 1806-1819.
13. Lu, C., Ke, H., Zhang, G., Mei, Y., & Xu, H. (2019). An improved weighted extreme learning machine for imbalanced data classification. *Memetic Computing*, 11(1), 27-34.
14. Liu, Y., Wang, Y., Ren, X., Zhou, H., & Diao, X. (2019). A classification method based on feature selection for imbalanced data. *IEEE Access*, 7, 81794-81807.
15. Thaher, T., Mafarja, M., Abdalhaq, B., & Chantar, H. (2019, October). Wrapper-based feature selection for imbalanced data using binary queuing search algorithm. In *2019 2nd International Conference on new Trends in Computing Sciences (ICTCS)* (pp. 1-6). IEEE.
16. Xu, Y. (2016). Maximum margin of twin spheres support vector machine for imbalanced data classification. *IEEE transactions on cybernetics*, 47(6), 1540-1550.

17. Atashpaz-Gargari, E., & Lucas, C. (2007, September). Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition. In *2007 IEEE congress on evolutionary computation* (pp. 4661-4667). Ieee.
18. Aliniya, Z., & Mirroshandel, S. A. (2019). A novel combinatorial merge-split approach for automatic clustering using imperialist competitive algorithm. *Expert Systems with Applications*, *117*, 243-266.