Chaotic Salp Swarm Optimization Using SVM for Class Imbalance Problems

Gillala Rekha[1(✉)], V. Krishna Reddy[2(✉)], and Amit Kumar Tyagi[3(✉)]

[1] Koneru Lakshmaiah Educational Foundation, Hyderabad, India
gillala.rekha@klh.edu.in
[2] Koneru Lakshmaiah Educational Foundation, Guntur, India
vkrishnareddy@kluniversity.in
[3] Vellore Institute of Technology, Chennai, India
amitkrtyagi025@gmail.com

Abstract. In most of the real world applications, misclassification cost of minority class samples can be very high. For high dimensional data, it will be a challenging problem as it may increase in overfitting and degradation of performance of the model. Selecting the most discriminate features is popular and recently used to address this problem. To solve class imbalance problems may optimization algorithms have been proposed in the literature. One among them is bio-inspired optimization algorithm. These algorithms are used to optimize the feature or instance selection. In this paper, a new bio-inspired algorithm called Chaotic Salp Swarm Algorithms (CSSA) were used to find the most discriminating features/attributes from the dataset. We employed 10 chaotic maps functions to assess the main parameters of salp movements. The proposed algorithm selects the important features from the dataset and it is mainly comprised of features selection phase, and classification phase. In the former phase, the most important features were selected using CSSA. Finally, the selected features from CSSA were used to train Support Vector Machine (SVM) classifier in the classification phase. Experimental results proved the ability of selecting optimal feature subset using CSSA, with accurate classification performance. Our observation on different data sets using Accuracy, F-measure, G-Mean, AUC and weighted as indicative metric provide better solution.

Keywords: Salp swarm algorithm · Support vector machine · Chaotic mapping · Feature selection · Optimization algorithm

1 Introduction

The imbalanced datasets occur most often in many application domains [7]. The datasets are said to be skewed in nature, when one class is sufficiently represented called majority/negative class while other very important class has fewer

Supported by KL University.

A. Abraham et al. (Eds.): HIS 2019, AISC 1179, pp. 220–229, 2021.
https://doi.org/10.1007/978-3-030-49336-3_22

samples called minority/positive class. In simple terms, the important/positive samples will be very less in number compared to that of majority or negative samples. When the classifier is learned on such data distribution may result in misclassification of positive samples.

To deal with skewed distribution problem, different approaches have been proposed in literature, broadly including data-level techniques, algorithm-level techniques and ensemble techniques [10]. The data-level technique works as pre-processing technique by resampling the data. The most common techniques used in resampling are Random OverSampling (ROS) and Random UnderSampling (RUS). In the former technique, the random samples are generated for minority samples, to balance the distribution. While in the latter, the samples of majority class are discarded. The main drawback for oversampling is increase in the classifier training time and for undersampling loss of important information [6]. One of the most simple and most popular oversampling technique is Synthetic Minority Over-sampling TEchnique (SMOTE) proposed by Chawla [3]. SMOTE generates synthetic samples from current minority samples by interpolation. But it may not be possible for critical applications like medical diagnosis which depends much on real data. Some of the methods proposed in extension with issues of SMOTE are Borderline-SMOTE [11], Safe-level SMOTE [14], Cluster Based Oversampling [12]. For Random Undersampling the proposed methods are Edited Nearest Neighbor (ENN) [8] and Tomek Links [9] are purely based on data cleaning.

The algorithm-level techniques surrounded with modification of existing learning algorithms to avoid the bias towards minority class or incorporate different misclassification cost while learning. Few examples of former are Hellinger Distance Decision Trees (HDDT) [4], Class Confidence Proportion Decision Tree (CCPDT) [13] and other insensitive class-size decision trees and for latter, the author [5] presents Naive Bayesian and Support Vector Machine learning methods with equal and unequal costs. In the literature, the ensemble methods often give better results than individual classifiers. Ensemble algorithms [7] like bagging and boosting with pre-processing techniques have been successfully designed to work with imbalanced data.

Recently, feature selection has gained interest by researchers in addressing the imbalance learning problem [16]. Previous techniques like sampling techniques, algorithm techniques, and ensemble methods have focused on training data samples. On the other hand, feature selection focus on identifying important features from the training data. Feature selection is an important pre-processing technique to achieve better performance of the classification algorithm. The main goal of feature selection is to eliminate redundant features in the dataset. Feature selection has been classified into filter method and wrapper method. Filter method works by using data properties without depending on learning algorithms whereas, wrapper methods will use the learning algorithms to generate important features. The latter methods are more exact but computationally expensive than former methods. The feature selection problem is defined as multi-objective optimization problem and the aim is to select the important features to maximize the performance of the classifier. However, exhaustive search

for optimal subset features is almost practically impossible. Swarm-based and evolutionary algorithms have been proposed in the literature to search for the most important features from the given dataset. Many Bio-inspired optimization algorithms exist in literature namely Particle Swarm Optimization (PSO) [17], Aritificial Bee Colony (ABC) [2], Dragon Fly Algorithm (DFA) [18], Salp Swarm Algorithm (SSA) [15].

In this work, the important features are selected using Salp Swarm Algorithm (SSA). As any other optimization algorithms, SSA falls into local optima and does not find the global optima. Hence, in this paper we combined various chaos function with SSA and proposed Chaotic Salp Swarm Algorithm (CSSA) to select the most important features. In this work, a support vector machine model was proposed to evaluate the imbalanced dataset. The proposed model consists of two phases. The former is the attribute selection phase, wherein the most distinction attributes were selected using the proposed CSSA algorithm. The resultant optimal attributes were then trained on SVM classifier in the next phase, i.e. the classification phase. The rest of this paper is organized as follows: Section 2 presents a brief description of the SSA that are used in our proposed model. The brief introduction to chaotic map functions were presented in Sect. 3. Section 4 deals with assessment methods used for measuring the performance of the classifier on imbalanced data. The proposed model is been presented in Sect. 5. The experimental results along with its analysis and conclusion were presented in Sect. 6 and 7 respectively.

2 Salp Swarm Optimization (SSA)

Recently, evolutionary and swarm-based algorithms have been widely used for feature selection problem. These algorithms adaptively search the feature space by applying agents to reach optimal solution.

The Working of Salp Swarm Algorithms (SSA). In this section, the basic representation of SSA is proposed. Salp swarm algorithm is a nature based meta-heuristic algorithms proposed by Mirjalili [15] in 2017. SSA came from swarming behaviour of salp in the heavy oceans. They form a swarm known as salp chain for optimal locomotion based on foraging in oceans Mirjalili [15].

Mathematical Model of SSA: In SSA, the entire population is divided into two groups, leader and the follower. The front position of swarm taken up by the leader followed by the remaining swarms as followers. The salp positions are represented in an n-dimensional search space, where 'n' is the number of dimensions or features. The positions of all the salps are denoted on two-dimensional matrix called x. 'F' denotes the food source in the target search space.

The mathematical model to update the positions of the leaders is as follows:

$$x_i^1 = \begin{cases} F_i + r_1((up_n - lw_n)r_2 + lw_n)\, r_3 \geq 0 \\ \\ F_i - r_1((up_n - lw_n)r_2 + lw_n)\, r_3 < 0 \end{cases} \tag{1}$$

Where x_i^1 represents the salp position known as leaders in $i-1^{th}$ dimension, up_n and lw_n denotes the upper and lower boundaries at i-1th dimension respectively. F_i is the position of food at $i-1^{th}$ dimension and r_1, r_2, r_3 are the random numbers. The mathematical definition of r_1 is represented as follows:

$$2e^{-\left(\frac{4k}{K}\right)^2} \tag{2}$$

Where K is the maximum number of iteration and k represents the current iteration. The random number r_2, r_3 are generated uniformly between the range of [0, 1]. The position of the followers is updated using the Eq. (3):

$$x_i^j = \frac{1}{2}\alpha t^2 + \beta_0 t \tag{3}$$

Where x_i^j shows the position of j^{th} follower, $j \geq 2$, β_0 is the initial speed, $\alpha = \frac{\beta_{final}}{\beta_0}$ and $\beta = \frac{x-x_0}{k}$. Because time is represented as optimization within each iteration process and discrepancy within the iterations is equal to 1 and $\beta_0 = 0$, the equation for updating the followers position in $i-1^{th}$ dimension is represented as follows:

$$x_i^j = \frac{1}{2}\left(x_i^j + x_i^{j-1}\right) \tag{4}$$

3 Chaotic Map

Chaos play an vital role to address the behavior of swarms at each iteration. A negligible change in its underlying state of swarm may prompt non-linear change in the future behavior. Chaos optimization algorithms are popular search algorithms recently applied to evolutionary algorithms. The main course of action is to search for global optima based on chaotic properties like stochastic, regularity and ergodicity. Its main attention is to avoid evolutionary algorithms to fall into local optima. In this work, we used ten chaotic mapping techniques to increase the performance of SSA. Figure 1 shows the different chaos mapping functions.

4 Performance Metrics for Skewed Data Distribution

Accuracy (Acc) is a notable performance metric used in classification. It is defined as the quantitative relation between the classified data samples (correctly) to the total number of data samples (5). In the imbalanced datasets, accuracy shows bias towards majority class and lead to wrong decisions. Therefore, different performance metrics are need to assess the performance of the classifier when trained on imbalanced datasets. The suitable metrics used are precision, recall, AUC to measure the performance of classifier when trained on imbalanced datasets. Precision is the proportion of true positive to the total

Map No.	Name	Definition	Range
M1	Chebyshev	$p_{q+1} = cos(q \cos^{-1}(p_q))$	(−1,1)
M2	Circle	$p_{q+1} = mod(p_q + r - (\frac{l}{2\pi})sin(2\pi p_q), 1)$, $l = 0.5$ and $r = 0.2$	(0,1)
M3	Gauss/mouse	$p_{q+1} = \begin{cases} 1, & p_q = 0 \\ \frac{1}{mod(p_q, 1)}, & \text{otherwise} \end{cases}$	(0,1)
M4	Iterative	$p_{q+1} = \sin(\frac{l\pi}{p_q})$, $l = 0.7$	(-1,1)
M5	Logistic	$p_{q+1} = l p_q(1 - p_q)$, l=4	(0,1)
M6	Piecewise	$p_{q+1} = \begin{cases} \frac{p_q}{l}, & 0 \le p_q < l \\ \frac{p_q - l}{0.5 - l}, & l \le p_q < 0.5 \\ \frac{1 - l - p_q}{0.5 - l}, & 0.5 \le p_q < 1 - l \\ \frac{1 - p_q}{l}, & 1 - l \le p_q < 1 \end{cases}$, $l = 0.4$	(0,1)
M7	Sine	$p_{q+1} = \frac{l}{4}\sin(\pi p_q), l = 4$	(0,1)
M8	Singer	$p_{q+1} = \mu(7.86p_q - 23.31p_q^2 + 28.75p_q^3 - 13.302875p_q^4)$, $\mu = 1.07$	(0,1)
M9	Sinusoidal	$p_{q+1} = l p_q^2 sin(\pi p_q)$, $l = 2.3$	(0,1)
M10	Tent	$p_{q+1} = \begin{cases} \frac{p_q}{0.7}, & p_q < 0.7 \\ \frac{10}{3}(1 - p_q), & p_q \ge 0.7 \end{cases}$	(0,1)

Fig. 1. Different choatic mapping functions

number of true positive and false positive samples (6). Recall/sensitivity represents how well the model detects the true positive samples (7). The F-measure combines both recall and precision and defined as (8). Therefore, F-measure is suitable when the data is skewed in nature than any other metric.

$$Acc = (TruePositive + TrueNegative) / (TruePositive + TrueNegative + FalsePositive + FalseNegative) \tag{5}$$

$$Precision = (TruePositive) / (TruePositive + FalsePositive) \tag{6}$$

$$Recall = (TruePositive) / (TruePositive + FalseNegative) \tag{7}$$

$$F - Score = 2 * (Precision * Recall) / (Precision + Recall) \tag{8}$$

In this paper, the various performance metrics used are Accuracy, AUC, F-Score, G-Mean and Weighted measure. We have used weighted metric of F-Score, G-Mean and AUC and it is defined as (9)

$$Weightedmetric = (F - Score + G - Mean + AUC) /3 \tag{9}$$

5 CSSA: The Working Model

In this section a detail description of the proposed model will be illustrated. The proposed model consists of feature selection and classification phases. The proposed architecture is shown in Fig. 2. Each phase is been explained in detail as below.

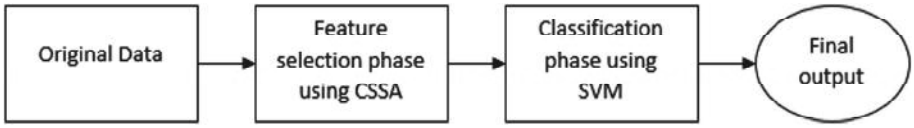

Fig. 2. The architecture of proposed model

5.1 Feature Selection Phase Using Chaotic Salp Swarm Algorithm (CSSA)

To select the appropriate attributes/features, Chaotic Salp Swarm algorithm (CSSA) was used. As discussion before, there are three main parameters chosen randomly and may affect the salp positions. The randomness in exploitation and exploration may badly affect the performance of the algorithms. To overcome this situation, chaotic maps are used to replace the random parameters using in the SSA. So, the combination of chaotic maps with salp swarm algorithm can be defined as Chaotic Salp Swarm Algorithms (CSSA). In this paper, ten chaotic mapping techniques were used to select the features for classification. The CSSA feature selection is based on wrapper method to find the optimal features at each iteration and result in improvement of classification performance.

Fitness Function. The position of the salp at each iteration is evaluated by using predefined fitness function Fi. The main objective criteria in evaluating the position of salp were to select the minimum features with maximum classification accuracy. The fitness function is defined with combination of accuracy and weight factor with value between [0, 1] as below.

$$Fi = max(Accuracy + WF(1 - FS/FC)) \tag{10}$$

Where FS stands for feature subset and FC stands for total count of attributes/features. The weight factor (WF) is used to improve the accuracy of the classifier and usually set near to 1. In our experiments, WF is set to 0.9. We employed 10-fold cross validation with dataset partitioned into training and testing set. The training set is used to learn the SVM classifier and the test set is used to evaluate the classifier and select the optimal features. Additionally, to select the discriminate features K-Nearest Neighbor(K-NN) algorithm was been used, where k represents the number of nearest neighbors. The best solution is the one, with optimal number of attributes/features and with maximum classification accuracy.

5.2 Classification Phase

In classification phase, the selected features are trained on Support Vector Machine (SVM). We applied different kernel function such as Linear, Radial Basis Function, and Polynomial. The kernel functions are used to transform the

non-linear data into high dimensional space, to make it linearly separable. The selected features are given to SVM classifier as input to check the robustness of the selected features.

6 Result and Analysis

In this section, the proposed model is evaluated on different datasets using 10 choatic mapping function. In the first experiment aims we evaluated the performance of Chaotic SSA using different SVM kernel techniques. In the second experiment, our proposed model is compared with SSA used to deal with class imbalance problems.

6.1 Data Set

We evaluate the proposed algorithm using 11 datasets from Keel repository with different imbalance ratio (IR) [1]. Table 1 shows the details of the imbalanced datasets with number of features and imbalance ratio.

Table 1. Datasets used

Dataset	No. of features	No. of samples	IR
Breast Cancer	9	286	2.36
Elico	7	220	1.86
Glass	9	214	1.82
Harberman	3	306	2.78
Pageblock	10	5472	8.79
Parkinsons	22	195	3
Pima	8	768	1.87
Thyroid	5	215	5.14
Vehicle	18	846	2.88
Wisconsin	9	683	1.86
Yeast	8	1484	2.46

6.2 Results

In this section, we compared the results (Fig. 3, 4 and 5) using different kernel functions to determine the best kernel function. We also aim at identifying the best chaotic mapping function for imbalanced datasets. From the experiments it is been observed that SVM with linear kernel performed well on most of the datasets, next polynomial and last is the radial basis kernel. Out of 11 datasets, 5 datasets had better performance using linear kernel compared with polynomial and radial basis kernel. We also observed that all choatic mapping functions

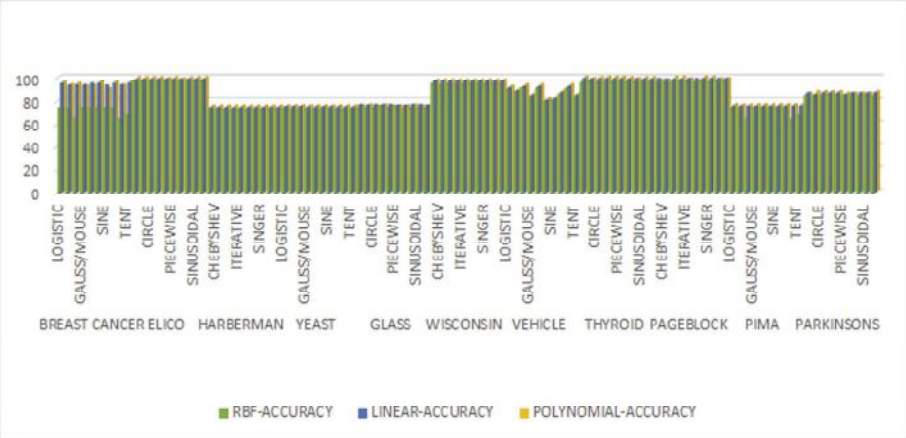

Fig. 3. Performance of accuracy on different choatic mapping functions

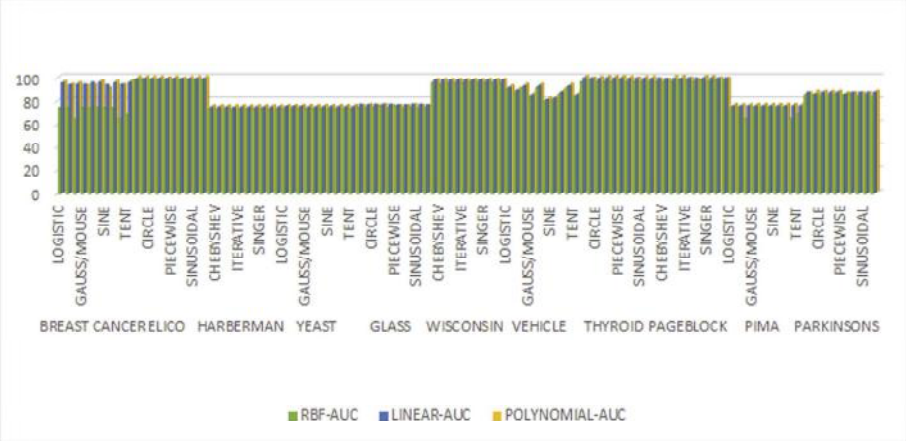

Fig. 4. Performance of AUC on different choatic mapping functions

worked well when classified using SVM (RBF, Linear, Polynomial kernel) on Breast cancer, Elico, Wisconsin, Thyroid and Pageblock datasets. In the experiment, the datasets are evaluated based on SSA and CSSA and better result shown when experimented on CSSA.

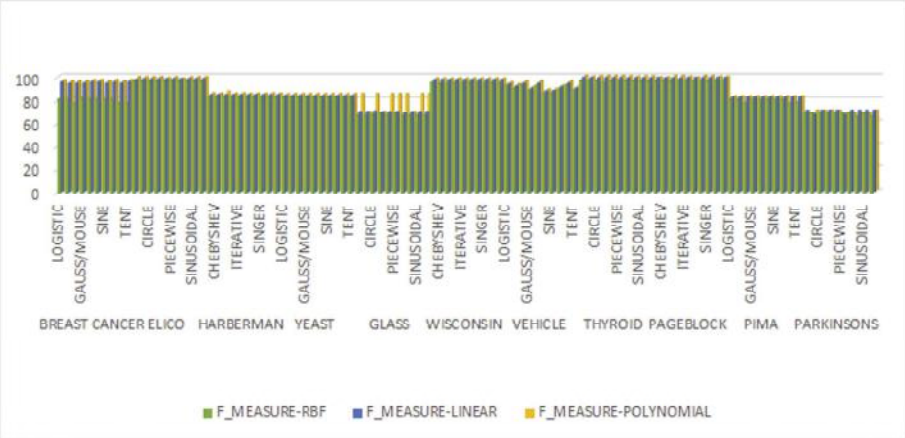

Fig. 5. Performance of F-Measure on different choatic mapping functions

7 Conclusion

In this paper, a novel hybrid chaos with salp swarm algorithm (CSSA) was proposed. To enhance the performance of SSA ten chaotic mapping functions were used. The proposed CSSA is applied on feature selection to select the most discriminate features from imbalanced dataset. The results shows that the CSSA algorithm outperformed over SSA. In terms of classification, Linear SVM performed well on features selected from CSSA.

References

1. Alcalá-Fdez, J., et al.: Keel data-mining software tool: data setrepository, integration of algorithms and experimental analysis framework. J. Multiple Valued Logic Soft Comput. **17** (2011)
2. Braytee, A., Hussain, F.K., Anaissi, A., Kennedy, P.J.: ABC-sampling for balancing imbalanced datasets based on artificial bee colony algorithm. In: 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA), pp. 594–599. IEEE (2015)
3. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: synthetic minority over-sampling technique. J. Artif. Intell. Res. **16**, 321–357 (2002)
4. Cieslak, D.A., Chawla, N.V.: Learning decision trees for unbalanced data. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 241–256. Springer (2008)
5. Datta, S., Das, S.: Near-bayesian support vector machines for imbalanced data classification with equal or unequal misclassification costs. Neural Netw. **70**, 39–52 (2015)
6. Fernández, A., del Río, S., Chawla, N.V., Herrera, F.: An insight into imbalanced big data classification: outcomes and challenges. Complex Intell. Syst. **3**(2), 105–120 (2017)

7. Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., Herrera, F.: A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.) **42**(4), 463–484 (2011)
8. García-Pedrajas, N., del Castillo, J.A.R., Cerruela-Garcia, G.: A proposal for local k values for k-nearest neighbor rule. IEEE Trans. Neural Netw. Learn. Syst. **28**(2), 470–475 (2015)
9. Gu, Q., Cai, Z., Zhu, L., Huang, B.: Data mining on imbalanced data sets. In: 2008 International Conference on Advanced Computer Theory and Engineering, pp. 1020–1024. IEEE (2008)
10. Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., Bing, G.: Learning from class-imbalanced data: review of methods and applications. Expert Syst. Appl. **73**, 220–239 (2017)
11. Lemaître, G., Nogueira, F., Aridas, C.K.: Imbalanced-learn: a python toolbox to tackle the curse of imbalanced datasets in machine learning. J. Mach. Learn. Res. **18**(1), 559–563 (2017)
12. Lim, P., Goh, C.K., Tan, K.C.: Evolutionary cluster-based synthetic oversampling ensemble (eco-ensemble) for imbalance learning. IEEE Trans. Cybern. **47**(9), 2850–2861 (2016)
13. Liu, W., Chawla, S., Cieslak, D.A., Chawla, N.V.: A robust decision tree algorithm for imbalanced data sets. In: Proceedings of the 2010 SIAM International Conference on Data Mining, pp. 766–777. SIAM (2010)
14. Ma, J., Afolabi, D.O., Ren, J., Zhen, A.: Predicting seminal quality via imbalanced learning with evolutionary safe-level synthetic minority over-sampling technique. Cogn. Comput., 1–12 (2019)
15. Mirjalili, S., Gandomi, A.H., Mirjalili, S.Z., Saremi, S., Faris, H., Mirjalili, S.M.: Salp swarm algorithm: a bio-inspired optimizer for engineering design problems. Adv. Eng. Softw. **114**, 163–191 (2017)
16. Moayedikia, A., Ong, K.L., Boo, Y.L., Yeoh, W.G., Jensen, R.: Feature selection for high dimensional imbalanced class data using harmony search. Eng. Appl. Artif. Intell. **57**, 38–49 (2017)
17. Wahono, R.S., Suryana, N.: Combining particle swarm optimization based feature selection and bagging technique for software defect prediction. Int. J. Softw. Eng. Its Appl. **7**(5), 153–166 (2013)
18. Zhang, L., Srisukkham, W., Neoh, S.C., Lim, C.P., Pandit, D.: Classifier ensemble reduction using a modified firefly algorithm: an empirical evaluation. Expert Syst. Appl. **93**, 395–422 (2018)