DOI: 10.15439/2017F91
ISSN 2300-5963 ACSIS, Vol. 11

Optimising SVM to classify imbalanced data using Dispersive Flies Optimisation

Haya Abdullah Alhakbani
Department of Computing
Goldsmiths College
University of London
London SE14 6NW,UK
Email: halha001@gold.ac.uk

Mohammad Majid al-Rifaie
Department of Computing
Goldsmiths College
University of London
London SE14 6NW,UK
Email: m.majid@gold.ac.uk

***Abstract*—Finding efficient solutions for search and optimisation problems has inspired many researchers to utilise nature informed algorithms, where the interactions in swarm could lead to promising solutions for challenging problems. One problem in machine learning is class imbalance, which occurs in real-world applications such as medical diagnosis. This problem can bias the classification or make it entirely out of context where the algorithms being applied to classify the data can potentially ignore the important minority class instances. In this paper, a parameters optimisation algorithm is proposed, which uses a swarm intelligence technique, Dispersive Flies Optimisation (DFO), to optimise the support vector machine kernel's parameters and perform cost sensitive learning to improve the classifier's performance on imbalanced data. The use of the swarming behaviour of the flies and their diversity in the search space in conducting cost sensitive learning are investigated on eight real-world datasets. The proposed algorithm has been compared with other techniques to optimise the classifier's parameters, that includes the well-known particle swarm optimisation, the frequently used grid search as well as random search, which is used as a control algorithm. The results demonstrate the statistically significant outperformance of the proposed optimisation technique over other techniques on the same datasets.**

I. Introduction

OVER the last decade, there has been a rapid increase in datasets worldwide due to the unparalleled growth in globalisation, as well as global markets. However, datasets are rendered useless unless there is a way to analyse them in a meaningful way. Data mining technologies have been adopted by various businesses like banking, retailing and telecommunication as the upcoming technology to help in converting large amounts of data which have been stored on a database into actionable knowledge and useful information. Nevertheless, dealing with large datasets present its own challenges, such as the issue of class imbalance that occur in real-world applications: fraud detection, medical diagnosis, direct marketing campaign and many other predictive models. This problem occurs when the number of instances in one class (i.e. majority class) extremely outnumber the number of instances in the other class (i.e.minority class). This is often due to the limitations of a data collection process such as high cost or privacy problems; for instance, biomedical data, which is derived from a rare disease and an abnormal condition, or some data that is often obtained via expensive experiments.

Numerous research have applied data mining techniques in solving imbalanced data issue at both data and algorithmic levels [1]. In this paper, a swarm intelligence model is proposed to optimise the support vector machine (SVM) parameters: two important parameters for the radial basis function (RBF) kernel are c and γ, as the choice of their values affects the classification accuracy. The model uses Dispersive Flies Optimisation (DFO) to tune the classifier's parameters and improve its performance on an imbalanced dataset without changing the dataset distribution.

II. Swarm intelligence and data mining

Swarm intelligence and evolutionary computation have been widely used to solve challenging problems in data mining such as feature selection and class imbalance [2]. When it comes to class imbalance and cost sensitive learning, choosing the kernel's parameters values is a challenging problem. Various swarm intelligence techniques have been used for parameters tuning or optimisation [3], [4]. Despite the rapid development in using swarm intelligence techniques to solve the class imbalance problem at the algorithmic level by optimising the kernel's parameters, these techniques face the challenge of the slow convergence rate, the trap to local optima and the number of tunable parameters. Al-Rifaie (2014) proposed a new meta heuristic, Dispersive Flies Optimisation, derived from the swarming behaviour of flies, which they use to locate the food source and the way it is communicated to other flies so that they can access the food source with minimal attempt to locate it [5]. In this paper, DFO will be used to perform SVM cost sensitive learning on various benchmarks data and compare the proposed method with both evolutionary and non evolutionary search based techniques from the literature on the same datasets. In the next section, DFO is described and its main components are explained.

A. Dispersive Flies Optimisation

DFO, first introduced in [5], is an algorithm inspired by the swarming behaviour of flies hovering over food sources. The swarming behaviour of flies is determined by several factors including the presence of threat which disturbs their convergence on the marker (or the optimum value). Therefore,

IEEE Catalog Number: CFP1785N-ART ©2017, PTI

having considered the formation of the swarms over the marker, the breaking or weakening of the swarms is also noted in the proposed algorithm. Algorithm 1 summarises the DFO algorithm.

The algorithm is characterised by two main components: a dynamic rule for updating flies position (assisted by a social neighbouring network that informs this update), and communication of the results of the best found fly to other flies. As stated earlier, the swarm is disturbed for various reasons; one of the impacts of such disturbances is the displacement of flies which may lead to discovering better positions. To consider this eventuality, a stochastic element is introduced to the update process. Based on this, individual components of flies' position vectors are reset if a random number, r, generated from a uniform distribution on the unit interval $U(0,1)$ is less than the *disturbance threshold* or *dt*. This guarantees a disturbance to the otherwise permanent stagnation over a likely local minima[1].

In summary, DFO is a simple numerical optimiser over continuous search spaces. DFO is a population based stochastic algorithm, originally proposed to search for an optimum value in the feasible solution space. The simplicity of the algorithm has been compared against several other swarm and evolutionary computation techniques in [6] where the elegance of the algorithm in having only one tunable parameter (the disturbance threshold), is explored. It has also been shown that DFO outperforms the standard versions of the well-known Particle Swarm Optimisation, Genetic Algorithm (GA) as well as Differential Evolution (DE) algorithms on an extended set of benchmarks over three performance measures of error, efficiency and reliability [5]. It is demonstrated that DFO is more efficient in 84.62% and more reliable in 90% of the 28 standard optimisation benchmarks used; furthermore, when there exists a statistically significant difference, DFO converges to better solutions in 71.05% of problem sets. Further analysis is also conducted to explore the diversity of the algorithm throughout the optimisation process, a measure that potentially provide more understanding on algorithm's ability to escape local minima. In addition to theoretical research on this algorithm, DFO has recently been applied to medical imaging [7]; furthermore, ongoing and current research are being conducted in the fields of image analysis, simulation and gaming [8], computational aesthetic measurements [9], (digital) arts [10], [11], protein folding, etc.

III. Experiments

In this paper, DFO is used to search for the optimal kernel parameters: c and γ. In this model, F-measure is deployed as an evaluation metric and the performance of DFO is compared against other parameters optimisation techniques to find the optimal kernel values over a set of benchmark datasets.

Algorithm 1 Dispersive Flies Optimisation

```
while Function Evalutions < Evaluations Allowed do
    for $i = 1 \to N$ do
        $\vec{x}_i$.fitness $\leftarrow f(\vec{x}_i)$
    end for
    $\vec{x}_s = \arg^* \min [f(\vec{x}_i)]$
    $\vec{x}_{i_n} = \arg^* \min [f(\vec{x}_{i_{\text{left}}}), f(\vec{x}_{i_{\text{right}}})]*$
    for $i = 1 \to N$ do
        for $d = 1 \to \mathrm{D}$ do
            $\tau_d \leftarrow x_{i_{nd}}^{t-1} + \mathrm{U}(0,1) \times (x_{sd}^{t-1} - x_{id}^{t-1})$
            if $(r < dt)$ then
                $\tau_d \leftarrow x_{\min,d} + r\,(x_{\max,d} - x_{\min,d})$
            end if
        end for
        $\vec{x}_i \leftarrow \vec{\tau}$
    end for
end while
```

$*\ \vec{x}_{i_{\text{left}}} = \vec{x}_{i-1}$ and $\vec{x}_{i_{\text{right}}} = \vec{x}_{i+1}$

TABLE I
Dataset list

Dataset	Minority Class	Majority Class	Attributes
Vehicle	199	647	18
Sonar	97	111	60
Ionosphere	34	126	34
WDBC	212	357	32
Abalone	42	689	8
Hepatitis	32	123	19
German credit	300	700	20
Breast Cancer	241	458	9

In order to evaluate the performance of the proposed technique, eight real-world datasets are used and available from the the University of California, Irvine (UCI) machine repository[2]. These datasets are imbalanced and they vary in size and class distribution. Moreover, they have been widely used as benchmarks to compare the performance of various methods in the literature. Table I provides a description of the datasets used. In this experiment, the authors have applied the proposed method on the Abalone datasets for the class '9' versus '18' and for the Vehicle dataset, the model is applied on the class 'Van' vs the others. Moreover, normalisation was applied on the datasets to scale each feature values to a [0,1] range, and instances with missing values are removed. Furthermore, to make predictions on new data valid, a train/test split is used, in which 80% of the dataset is used for training and 20% is used for testing. The advantages of train/test split are that the optimised c and γ are evaluated on unseen dataset. As the datasets are imbalanced, F-measure is used as a fitness value for SVM, in which the goal is to find the c and γ that will give the maximum F-measure.

A. Experiment set up

Fifty flies are set to optimise the SVM's parameters, in which the range for c that has been defined as [2^{-5},2^{15}] and the range of γ has been defined as [2^{-15},2^{3}] based on [12]. The iterations allowed is equal to 10. At the *initialisation phase*, each fly is assigned randomly to two values, with the first value being for c and the second for γ; using these values the fitness

[1] The source code of the original DFO algorithm can be found in the following web page: http://doc.gold.ac.uk/mohammad/DFO

[2] http://archive.ics.uci.edu/ml/

value, the F-measure, is generated. The fitness value is stored for each fly, to find the best neighbouring fly and the best fly in the whole swarm. At every iteration, the components of the position vector are independently updated at the *update phase*, considering the components vector for the best neighbouring fly and the components vector for the best fly in the whole swarm. It also considers if the random number, r, that is generated from the uniform distribution on the range [0,1], is less than the disturbance threshold dt. In the experiment, the dt is empirically equal to 0.5, which means 50% of the flies' components are randomly initialised to new positions in the search space. This will enhance the diversity of the algorithm and will provide a balance between exploration and exploitation. In order to ensure that the performance of the algorithm is not solely due to the disturbance mechanism, a control algorithm (random algorithm) is also applied to the problem and the results are reported.

IV. Results and discussion

Table II summarises the results of applying DFO as optimisation algorithm and compares them with other methods on the same datasets. This include PSO, grid search and random search. As shown in the table, the use of DFO was found to improve the F-measure for all datasets and the proposed model outperforms other techniques on the same datasets. For example, for the Ionosphere dataset, the F-measure increased from 94.52%, as obtained by the PSO, to 98.59%. Similar improvements in the F-measure can be seen in the rest of the datasets. As a result, the proposed model which uses DFO to optimise the SVM kernel's parameters c and γ, demonstrates the ability to improve the classifier performance on imbalanced datasets. As Fig. 1 illustrate, while the other technique exhibit varying performance over different datasets, DFO is shown to provide a consistent outperformance over all datasets. Given the importance of conducting a statistical analysis measuring the presence of any significant difference in the performance of the proposed model and the other techniques including PSO, grid and random search, t-test is applied. This statistical significant test is applied using the outcome of the entire trials (30 runs) on each experiment. Based on the results, the F-measure difference is significant at 5% level. The result of this test indicates that the proposed optimisation technique offers a statistically significant improvement in the classifier's performance on the imbalanced datasets when compared to the other techniques.

A. Impact of Disturbance Threshold

The disturbance mechanism in DFO provides a stable independent convergence throughout the optimisation process. It also maintains a balance between exploration and exploitation. At the update phase, the dt is the only adjustable parameter to set that controls the diversity of the algorithm. A suitable value for this parameter depends on the size of the swarm, the number of iterations and the size of the search space. Therefore, further work needs to be done to find a theoretically suitable value for this parameter. In this experiment, dt is empirically set to 0.5, which allows for an enhanced diversity of the population in covering the search space, as well as the ability to escape local optima.

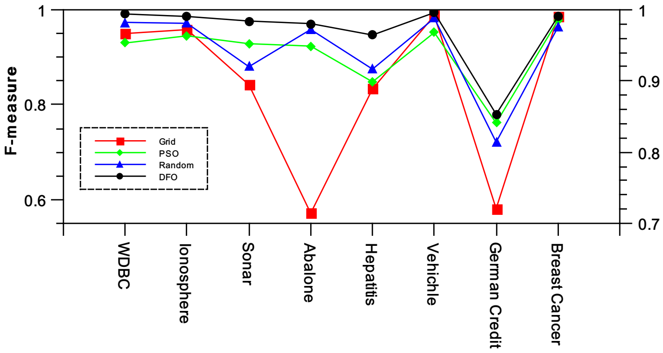

Fig. 1. Comparison of F-measure on all datasets

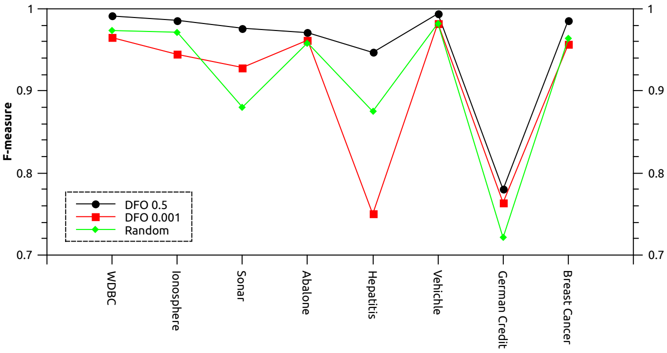

Fig. 2. Negative impact of reducing the disturbance threshold to $dt = 0.001$

As stated previously, random algorithm is included in the comparison as a control algorithm to ensure the DFO's performance is not solely attributable to its disturbance mechanism and that the coupled mechanisms of forming and breaking of the swarm, together, give rise to the performance of the algorithm. Equally, in order to demonstrate the impact of the absence or reduction of diversity (induced through the disturbance mechanism), another control algorithm with small disturbance threshold ($dt = 0.001$) is proposed. Fig. 2) illustrates that the sole presence of diversity or the lack of it, negatively impacts the performance of the algorithm.

V. Conclusion

Class imbalance is a major problem in machine learning. This work investigated the use of DFO to optimise the RBF kernel's parameters to improve the classifier performance without changing the distribution of the dataset by applying data level solutions such as oversampling or undersampling the dataset. The proposed method has performed *statistically significantly* better when compared to other techniques on all datasets. Moreover, the simplicity of this swarm intelligence algorithm adds to its appeal when applied to complex search

TABLE II
PERFORMANCE MEASUREMENTS COMPARISON OF DFO-SVM AND OTHER TECHNIQUES

Dataset	Method	Accuracy	Sensitivity	Specificity	F-measure	AUC
WDBC	PSO	92.98%	93.33%	92.75%	93.00%	0.93
	Grid	94.73%	90.69%	97.18%	95.00%	0.93
	Random	97.36%	93.87%	100%	97.35%	0.96
	DFO	99.12%	98%	100%	**99.12%**	0.99
Sonar	PSO	92.85%	90.90%	100%	92.86%	0.92
	Grid	87.71%	76.19%	95.14%	84.21%	0.823
	Random	88.90%	92.30%	81.25%	88.00%	0.86
	DFO	97.61%	96.42%	100%	**97.63%**	0.98
Ionosphere	PSO	94.36%	92.30%	100%	94.52%	0.96
	Grid	97.14%	95.83%	97.83%	95.83%	0.95
	Random	97.18%	100%	91.30%	97.15%	0.95
	DFO	98.59%	97.87%	100%	**98.59%**	0.98
Abalone	PSO	93.87%	30.76%	100%	92.35%	0.65
	Grid	97.95%	40.00%	100%	57.14%	0.88
	Random	96.59%	37.50%	100%	95.85%	0.68
	DFO	97.27%	62.50%	99.28%	**97.09%**	0.80
Hepatitis	PSO	87.50%	33.33%	100%	84.82%	0.66
	Grid	87.50%	83.33%	90.00%	83.33%	0.83
	Random	87.50%	50.00%	92.85%	87.50%	0.71
	DFO	93.75%	100%	93.33%	**94.68%**	0.96
Vehicle	PSO	95.29%	86.36%	98.41%	95.21%	0.92
	Grid	98.22%	98.43%	97.62%	98.81%	0.99
	Random	98.24%	95.35%	99.21%	98.23%	0.97
	DFO	99.41%	97.50%	100%	**99.40%**	0.98
German Credit	PSO	76.00%	54.90%	83.22%	76.14%	0.69
	Grid	79.33%	45.74%	94.66%	58.11%	0.83
	Random	73.50%	48.28%	82.39%	72.11%	0.65
	DFO	78.00%	65.07%	83.94%	**78.00%**	0.74
Breast Cancer	PSO	97.81%	97.77%	97.82%	97.81%	0.97
	Grid	99.25%	97.94%	100%	98.56%	0.96
	Random	96.34%	94.00%	97.70%	96.34%	0.95
	DFO	98.54%	98.70%	98.88%	**98.59%**	0.98

and optimisation problems with only one parameter to tune as opposed to the presence of more parameters in several other swarm and evolutionary computation techniques. Amongst the future work is the comparison of the performance of DFO against other swarm and evolutionary computation techniques over larger datasets.

REFERENCES

[1] N. V. Chawla, *Data Mining for Imbalanced Datasets: An Overview*. Boston, MA: Springer US, 2005, pp. 853–867. [Online]. Available: http://dx.doi.org/10.1007/0-387-25465-X_40

[2] S. Dara, H. Banka, and C. S. R. Annavarapu, "A rough based hybrid binary pso algorithm for flat feature selection and classification in gene expression data," *Annals of Data Science*, pp. 1–20, 2017. [Online]. Available: http://dx.doi.org/10.1007/s40745-017-0106-3

[3] P. Cao, D. Zhao, and O. R. Zaïane, "A pso-based cost-sensitive neural network for imbalanced data classification," in *Revised Selected Papers of PAKDD 2013 International Workshops on Trends and Applications in Knowledge Discovery and Data Mining - Volume 7867*. New York, NY, USA: Springer-Verlag New York, Inc., 2013, pp. 452–463. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-40319-4_39

[4] J. Li and B. Li, *Parameters Selection for Support Vector Machine Based on Particle Swarm Optimization*. Cham: Springer International Publishing, 2014, pp. 41–47. [Online]. Available: https://doi.org/10.1007/978-3-319-09333-8_5

[5] M. M. al-Rifaie, "Dispersive flies optimisation," in *Proceedings of the 2014 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, M. P. M. Ganzha, L. Maciaszek, Ed., vol. 2. IEEE, 2014, pp. pages 529–538. [Online]. Available: http://dx.doi.org/10.15439/2014F142

[6] M. M. al - Rifaie, "Perceived simplicity and complexity in nature," in *AISB 2017: Computational Architectures for Animal Cognition*, University of Bath, Bath, U.K., 2017, pp. 299–305.

[7] M. M. al Rifaie and A. Aber, *Dispersive Flies Optimisation and Medical Imaging*. Cham: Springer International Publishing, 2016, pp. 183–203. [Online]. Available: https://doi.org/10.1007/978-3-319-21133-6_11

[8] M. King and M. M. al-Rifaie, "Building simple non-identical organic structures with dispersive flies optimisation and a* path-finding," in *AISB 2017: Games and AI*, University of Bath, Bath, U.K., 2017, pp. 336–340.

[9] M. M. al Rifaie, A. Ursyn, R. Zimmer, and M. A. J. Javid, *On Symmetry, Aesthetics and Quantifying Symmetrical Complexity*. Cham: Springer International Publishing, 2017, pp. 17–32. [Online]. Available: https://doi.org/10.1007/978-3-319-55750-2_2

[10] M. M. al Rifaie, F. F. Leymarie, W. Latham, and M. Bishop, "Swarmic autopoiesis and computational creativity," *Connection Science*, pp. 1–19, 2017. [Online]. Available: http://dx.doi.org/10.1080/09540091.2016.1274960

[11] J. M. Bishop and M. M. al Rifaie, "Autopoiesis, creativity and dance," *Connection Science*, vol. 29, no. 1, pp. 21–35, 2017. [Online]. Available: http://dx.doi.org/10.1080/09540091.2016.1271399

[12] C.-W. Hsu, C.-C. Chang, C.-J. Lin *et al.*, "A practical guide to support vector classification," 2003.