An Optimized Discretization Approach using k-Means Bat Algorithm

Rozlini Mohamed[*1], Noor Azah Samsudin[2]

[1,2]Univesiti Tun Hussein Onn Malaysia, 86400 Parit Raja, Johor, Malaysia
rozlini@uthm.edu.my[1*]

Article History: Received: 10 November 2020; Revised: 12 January 2021; Accepted: 27 January 2021; Published online: 05 April 2021

Abstract: This study has proposed a relatively new discretization approach using k-means and Bat algorithm in preparation phase of classification problem. In essence, bat algorithm is applied to find the best search space solution. Eventually, the best search space solution is utilized to produce cluster centroid. The cluster centroid is very useful to determine appropriate breakpoint for discretization. The proposed discretization approach is applied in the experiments with continuous datasets. Decision Tree, k-Nearest Neighbours and Naïve Bayes classifiers are used in the experiments. The proposed discretization approach is evaluated against other existing approaches: K-Means algorithm, hybrid K-Means with Particle Swarm Optimization (PSO) and hybrid K-Means with Whale Optimization Algorithm (WOA). The classification performance is evaluated in terms of accuracy, recall, f-measure and receiver operating characteristic curve (ROC). To test the performance of the proposed algorithm, nine benchmark continuous datasets are used. The proposed algorithm show the better results compare to other approaches. The proposed algorithm performs better in discretization to solve classification problems.

Keywords: Discretization, Bat algorithm, classification, K-Means

1. Introduction

Discrete values is necessary in representation of knowledge for data mining application. This is because the characteristics of discrete values that are very close to the representation of knowledge make these discrete values easier to handle compared to continuous values. From Madhu et al. (2014), the conversion process from continuous value into discrete data is a major step in data preparation. Thus, the continuous attribute is need to convert into discrete value before the data mining process. Where, the continuous values is composed with a range that called as breakpoint. For example the distance attribute can be transformed in discrete values representing by intervals: from 0 to 10km, over than 10km into 100km and over 100km. The task to determine continuous value into these range is known as discretization, and become an essential task of the data preparation in classification (Cano et al., 2016).

Choose the correct data processing method has significant impact on dataset classification (AlMuhaideb & Menai, 2016). The major challenge in classification problem is to obtain the better result in classification performance. There are many ways to improve classification performance such as feature selection (Uçar, 2020), fuzzy clustering (Xu et al., 2020), enhancement of Random Forest Classification (More & Rana, 2020), and discretization (Zhou et al., 2021).

The optimization approach is intensively developing since it is widely used to solve problems in the real life (Slowik & Kwasnicka, 2017). Recently, the data mining field have adjust the algorithms and method with advanced optimization, theory graph and matrix computations. Based on the methods, matrix representation is used to present the data. While, optimization problem is used to formulate the data mining problems with matrix variables (Azham Hussain, et al, 2019). The task of data mining is a process to find the goal of optimization problem, depending on minimizing or maximizing objective function.

Data preparation is an important process in classification. Meanwhile, discretization process is important in classification. However, most of the research in discretization lack in optimization approach (Hacibeyoğlu & Ibrahim, 2016; Lavangnananda & Chattanachot, 2017). In this paper, a better discretization scheme is obtain through optimization algorithm. The objective of discretization is to find the best solution in many optimization problems. Thus, searching in a whole space is needed to find the best solution. A new hybrid optimized discretization approach in data preparation phase is proposed in this research. To avoid loss of information and to maintain the accuracy of the classification algorithm are the challenging issues of discretization process. Discretization of continuous value for feature can be used to solve that problem. The feature value is divided into discrete range where each range present a category. This research proposes a new discretization approach based on hybrid K-Means with Bat algorithm discretization approach for single-class single- label.

This paper is organized as follows: Literature Review on K-Means as discretization approach and optimization algorithm are presented in Section 2. Section 3 discusses The Proposed Discretization Approach based on K-Means and Bat algorithm and the description of the data sets. Section 4 discusses the description of

the discretization methods used for comparison, experiment results the followed by discussion. Conclusion of this paper is presented in Section 5.

2. Literature Review

a. K-Means as Discretization Approach

Various discretization approaches can be used in many problems. Discretization can involve one method or more than method. For example, the research from (Fikri et al., 2020) uses fuzzy logic and Random Forest classifier as discretization approach to improve classification accuracy. Also employ multivariate discretization (Zamudio-Reyes et al., 2017), and K-Means (MacQueen, 1967) as discretization approach. In 1967, J. MacQueen was proposed as an iterative algorithms. At the beginning, k data points are randomly select as reference point called as centroid.

K-Means can be used as discretization approach. In (Maryono et al., 2018) K-Means act as discretization on mixed attribute dataset. In another research, K-Means is combined with discretization technique and Naïve Bayes classifier (Tahir et al., 2016) applied in network intrusion detection system. Moreover, K-Means can be implemented as discretization approach without combination with another approach such as in network intrusion detection research (Zhao et al., 2018) and graph optimal graph clustering (Han et al., 2020)

b. Optimization Algorithm

Recently, the real-life problems can be solve by using optimization algorithms. The right choice of an optimization algorithm is needed to solve the optimization problem. There are many way to classify the optimization algorithms which are depending on the characteristics and focus. One of the commonly used algorithms is swarm intelligence-based. This section present three prominent swarm-based optimization algorithm; Bat Algorithm (BA), Particle Swarm Optimization (PSO) and Whale Optimization Algorithm (WOA).

The population-based metaheuristic optimization algorithm (Nguyen et al., 2020) known as Particle Swarm Optimization (PSO) algorithm is proposed by Kennedy and Eberhart (1995). PSO simulates the movement of birds that are randomly looking for food in search space. According to PSO, every bird is considered as a solution or particle. PSO was used to resolve many kind of optimization problems such as scheduling (Marichelvam et al., 2020), multi-objective optimization (Qu et al., 2020), and clustering (Li et al., 2019).

Mirjalili & Lewis in 2016 (2016) was present Whales Optimization Algorithm (WOA) (Gharehchopogh & Gholizadeh, 2019). WOA mimic the activities of humpback whales and this algorithm also from the nature-inspired meta-heuristic. WOA was used to solve problems such as engineering design problems (Chen et al., 2019), multiobjective optimization problem (Got et al., 2020) and clustering problem (Nagarajan & Dhinesh Babu, 2019).

Xin-She (2010) was present Bat Algorithm (BA) (Nguyen et al., 2020). BA was developed that mimic the behavior of bat where according to echo to find the pray. Meanwhile, the algorithm of BA is changing pulse rates of loudness and emission to find the best solution. BA have been employed in various applications. In (Aboubi et al., 2016; Kaur et al., 2018) BA was used for classification in medical data. Moreover, BA was used to solve problems in engineering application, such as in fault diagnosis (X. Yang et al., 2019), seismic safety (Bekdaş et al., 2018), and searching problems for robotic sectors (Tang et al., 2020).

BA is used in hair analysis for vitamin D content prediction (Hassanien et al., 2017). In another research, BA was able to handle the emotional controller problem, where this approach outperforms the PSO algorithm (Khooban & Javidan, 2016). Furthermore, Gao was employed BA in visual tracking (Gao et al., 2016) and the experiment results show BA is good in track the target in during image tracking process compare to PSO.

3. The Proposed Discretization Approach

This research proposed a new discretization approach, called hBA for discretizing the continuous values of a datasets. To evaluate the effectiveness of the proposed approach, the rest of experiments have been conducted.

a. Data Acquisition

Nine continuous datasets are obtained from UCI Machine Learning Repository (http://archive.ics.uci.edu/ml) and used. UCI was created in 1987 by David Aha(Imran et al., 2013) and fellow graduate students at UC Irvine, where more than 500 datasets were provided to public for research purposes. The 9 continuous datasets that are used in this research are listed as follows:

i. Credit Approval, DS1
ii. Hill Valley, DS2
iii. Image Segmentation, DS3
iv. Libras Movement, DS4
v. Plant Species, DS5
vi. Steel Plates Faults, DS6
vii. Urban Land, DS7
viii. Automobile, DS8
ix. Yeast, DS9

The scale of instances are from 159 to 5000 and attributes in range 8 to 100 features. These 9 datasets are from various domains, consisting of different number of instances and attributes. The information about datasets are present in Table 1 including the dataset name, instances, attributes and dimension. These datasets are in the format of Comma-Separated Values (CSV) which is the delimited text file that uses a comma to separate values for machine learning using WEKA (Waikato Environment for Knowledge Analysis).

Table 1. Continuous dataset information

Dataset	No of Instances	No of Attributes	Dimension
Credit Approval	690	15	10,350
Hill Valley	606	100	60,600
Image Segmentation	210	19	3,990
Libras Movement	360	90	32,400
Plant Species	1600	64	102,400
Steel Plates Faults	1941	27	52,407
Urban Land	507	147	74,529
Automobile	159	25	3,975
Yeast	1484	8	11,872

b. Discretization with hybrid K-Means with BA

A. K-Means Algorithm

K-Means is an iterative algorithm. At the beginning, k data points are randomly selected as reference points, also known as centroids. Data are divided into k clusters. Let assume cluster k –th consist of x data point that nearest to center point, kc. Location of center point and the data point are repetition process and repeated until meet the optimum solution. The definition of K-Means are represented using equation (1).

$$\sum_{k=1}^{K} \sum_{i \in S_k} \|x_j - k_k c\|^2 \quad (1)$$

B. Bat Algorithm

Bat algorithm (BA) mimics the bat behavior where a group of bats in a population will fly randomly to find the prey. Each bat will detect the nearest prey to them and will update the position and speed. The bat that is closest to the prey becomes the best bat in the population. In BA the speed is known as the velocity and a set of bats is known as the solution. According to BA the fitness function must be computed for each bat and the best fitness function for each bat is known as $pbest$. Then, the highest $pbest$ will be the $gbest$. The bat with the $gbest$ becomes the best bat in population.

This study is follow the following rules of BA:

(i) First, distance detection. The entire of bats in population will used echo to detect their position with pray.

(ii) Second, the bats randomness flies to search the pray at position x_i and velocity v_i with a fixed frequency $f_{\min}$. During the searching, loudness A_0 and wavelength λ is changing iteratively. When bats emitted their pulses, automatically the wavelength and the pulse rate are adjusted, $r \in [0,1]$, subject to nearest on the goal.

(iii) Third, a loudness differs from a maximum A_0 to a minimum value $A_{\min}$. Thus, from the above rules, to update the velocities v_i^t and location x_i^t are using the equations (2) to (4);

$$f = f_{\min} + (f_{\max} - f_{\min})\beta, \quad (2)$$

$$x_i^t = x_i^{t-1} + v_i^t, \quad (3)$$

$$v_i^t = v_i^{t-1} + (v_i^{t-1} - v_*)f_i, \quad (4)$$

where $\beta \in [0,1]$ is a random vector.

C. Hybrid Discretization K-Means with BA Algorithm

In discretization, the vital role is to determine the breakpoints of the integer values. The continuous value can be assigned according to breakpoints, as integer values such as 1,2 or 3. In hBA approach, the cluster centroid of each cluster, k-th is determined by BA. The format of the dataset is presented in Table 2. In hBa, each bat position consists of the number of features denote by N in dataset. The information regarding the solution is given by $S = \{s_1, s_2, \cdots, s_n\}$ where n is the number of solutions. Each solution is $S_i = \{a_{i1}, a_{i2}, \cdots, a_{ik}\}$, where k is the number of attributes for the S_i-th solution in the DS-th dataset.

Table 2. Dataset Format

Attribute / Instance	a_1	a_2	a_3	$\cdots$	a_N
S_1	0.23	1.33	0.56	2.33	3.33
$\vdots$	:	:	:	:	:
S_n	…	…	…	…	…

For example, it is assumed that dataset DS, has 10 features, 15 instances and 20 generations or repetitions. After 20 repetitions, the 10-th bat or instance number 10th represented by S_{10} is considered the best in the population. The position for 10th instance is $S_{10} = \{a_{10,1}, a_{10,2}, \cdots, a_{10,10}\}$. Thus the initial centroid for cluster k in K-Means algorithm, $kc = \{a_{10,1}, a_{10,2}, \cdots, a_{10,10}\}$.

Let the set of data points in dataset $D = \{x_1, x_2, \cdots, x_m\}$, where $x_i = \{x_{i1}, x_{i2}, \cdots, x_{ir}\}$

$$\sum_{m=1}^{m} \sum_{j=1}^{t} \|x_m - kc_j\|^2 \quad (2)$$

where, $\|x_k - v_j\|^2$ is the Eucledian distance between a point, x_m, and a centroid, k_{cj} , iterated over all D point in the m-th cluster, for all k, cluster.

c. Classifiers Performance

In optimize discretization approach process at the end the results can be evaluated through classifier. Classifier is a learning algorithm that learn the model from training data. There are four classifier use in this research, which are Decision Tree, k-Nearest Neighbours and Naïve Bayes. These classifiers are usually used in classification (Shafiq et al., 2020).

To compare classifiers, four classification evaluation criteria are used; Accuracy, Precision, Recall and ROC. These performance criteria are used to evaluate the effectiveness of optimize in discretization and feature selection in order to improve classification accuracy through six experiments.

4. Results and Discussion

The algorithms used in this experiment are executed using MATLAB. Validation of the algorithm using four classifiers (Tree, k-Nearest Neighbours and Naïve Bayes) from WEKA. The goal of this experiment is to validate that the discrete data can improve classification performance in terms of accuracy, recall, f-measure and ROC.

The experiment is conducted by converting all continuous dataset and generating new discrete datasets. The comparison have been made between proposed approach, hBA between continuous dataset denote as $cont$ and discrete dataset that convert by K-Means classifiers denote as dk, hybrid K-Means with PSO, $hPSO$ and hybrid K-Means with WOA, $hWOA$.

a. Accuracy of Discrete Datasets

The results of performance measure accuracy for Naïve Bayes classifier shows in Table 3. The accuracy of eight datasets out of nine datasets achieve better results after discretization process. Six datasets out of eight datasets are using hybrid discretization, where four datasets are improved by hBA.

Table 4 shows the accuracy of six out of nine datasets which are improved after discretization process. hBa and dK are able to improve three datasets out of six datasets, respectively. By using Decision Tree, five datasets out of nine datasets achieve better results after discretization process. As shown in Table 5, dBA improved the accuracy of three datasets out of five datasets and dk improved the accuracy of two datasets out of five datasets.

Table 3. Accuracy of Naïve Bayes for Discrete Datasets

	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9
$cont$	0.781	0.519	0.773	0.644	0.859	0.669	0.791	0.519	0.585
dk	0.765	0.519	0.786	0.630	0.842	0.693	0.821	0.566	0.579
$hPSO$	0.843	0.527	0.691	0.308	0.613	0.594	0.648	0.478	0.560
$hWOA$	0.849	0.488	0.668	0.426	0.315	0.585	0.664	0.474	0.410
hBA	0.849	0.526	0.821	0.686	0.867	0.692	0.817	0.532	0.598

Table 4. Accuracy of k-Nearest Neighbors for Discrete Datasets

	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9
$cont$	0.807	0.584	0.869	0.864	0.754	0.718	0.785	0.788	0.524
dk	0.828	0.584	0.895	0.861	0.739	0.718	0.804	0.836	0.522
$hPSO$	0.811	0.529	0.618	0.556	0.257	0.557	0.480	0.677	0.421
$hWOA$	0.823	0.444	0.755	0.630	0.336	0.610	0.626	0.673	0.408
hBA	0.810	0.484	0.864	0.837	0.829	0.683	0.809	0.700	0.551

Table 5. Accuracy of Decision Tree for Discrete Datasets

	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9
$cont$	0.848	0.500	0.890	0.701	0.488	0.751	0.800	0.814	0.548
dk	0.842	0.500	0.890	0.706	0.493	0.751	0.804	0.805	0.575
$hPSO$	0.853	0.480	0.882	0.652	0.514	0.734	0.776	0.724	0.595
$hWOA$	0.848	0.440	0.831	0.286	0.294	0.682	0.606	0.762	0.480
hBA	0.833	0.467	0.834	0.411	0.132	0.667	0.669	0.673	0.462

b. Recall of Discrete Datasets

The performance measure results in terms of recall for Naïve Bayes classifier are shown in Table 6. All datasets obtain good results after discretization process by using hybrid discretization which is 8 datasets from 9 datasets using hBA.

Table 7 shows that six out of nine datasets are improved after discretization process. hBA, $hPSO$, $hWOA$ discretization approach improved 2, 1 and 3 from 6 datasets, respectively. By using Decision Tree, six datasets out of nine datasets are improved after discretization process as shown in Table 8. hBA improved two datasets, dK improved one dataset and $hWOA$ improved three datasets from six datasets.

Table 6. Recall of Naïve Bayes for Discrete Datasets

	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9
$cont$	0.762	0.507	0.776	0.628	0.851	0.603	0.786	0.522	0.576
dk	0.722	0.502	0.590	0.264	0.489	0.417	0.256	0.314	0.358
$hPSO$	0.843	0.518	0.652	0.267	0.579	0.584	0.637	0.478	0.396
$hWOA$	0.762	0.507	0.776	0.628	0.851	0.603	0.786	0.522	0.576
hBA	0.848	0.515	0.811	0.678	0.860	0.639	0.810	0.553	0.567

Table 7. Recall of k-Nearest Neighbors for Discrete Datasets

	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9
$cont$	0.807	0.584	0.871	0.858	0.740	0.719	0.768	0.774	0.523
dk	0.719	0.498	0.624	0.278	0.383	0.485	0.363	0.270	0.500
$hPSO$	0.810	0.520	0.619	0.250	0.239	0.557	0.446	0.673	0.431
$hWOA$	0.807	0.588	0.871	0.859	0.740	0.719	0.768	0.774	0.523
hBA	0.809	0.487	0.867	0.833	0.821	0.676	0.802	0.698	0.532

Table 8. Recall of Decision Tree for Discrete Datasets

	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9
$cont$	0.848	0.503	0.890	0.697	0.479	0.752	0.792	0.811	0.599
dk	0.745	0.503	0.681	0.219	0.378	0.539	0.363	0.258	0.537
$hPSO$	0.848	0.503	0.824	0.281	0.292	0.681	0.595	0.755	0.484
$hWOA$	0.848	0.503	0.890	0.699	0.479	0.752	0.792	0.811	0.61
hBA	0.854	0.483	0.883	0.648	0.507	0.730	0.767	0.742	0.576

c. F-Measure of Discrete Datasets

The performance measure results in terms of f-measure for Naïve Bayes classifier are shown in Table 9. All datasets obtained good results after discretization process using hybrid discretization. Where, 7 datasets using hBA as discretization approach.

Table 10 shows the F-Measure of five datasets out of nine datasets which are improved after discretization process. hBA, $hWOA$ and dk are able to improve two datasets, one dataset, and one dataset, respectively. By using Decision Tree, five datasets out of nine datasets are improved after discretization process as shown in Table 11. dBA technique improved three datasets out of five datasets. Both $hPSO$ and $hWOA$ techniques improved one dataset out of five datasets.

Table 9. F-Measure of Naïve Bayes for Discrete Datasets

	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9
$cont$	0.753	0.433	0.766	0.631	0.852	0.591	0.787	0.505	0.566
dk	0.722	0.343	0.542	0.275	0.474	0.370	0.255	0.500	0.556
$hPSO$	0.843	0.483	0.638	0.250	0.585	0.553	0.627	0.454	0.540
$hWOA$	0.858	0.477	0.666	0.407	0.286	0.531	0.649	0.465	0.410
hBA	0.848	0.469	0.808	0.671	0.861	0.630	0.812	0.530	0.567

Table 10. F-Measure of k-Nearest Neighbors for Discrete Datasets

	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9
$cont$	0.807	0.584	0.870	0.856	0.741	0.718	0.772	0.778	0.522
dk	0.718	0.405	0.613	0.290	0.330	0.482	0.357	0.779	0.489
$hPSO$	0.810	0.486	0.617	0.230	0.231	0.557	0.445	0.669	0.423
$hWOA$	0.833	0.440	0.754	0.456	0.336	0.611	0.608	0.672	0.407
hBA	0.809	0.470	0.860	0.831	0.819	0.677	0.803	0.697	0.532

Table 11. F-Measure of Decision Tree for Discrete Datasets

	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9
$cont$	0.848	0.450	0.889	0.694	0.478	0.751	0.294	0.811	0.552
dk	0.741	0.440	0.674	0.215	0.366	0.539	0.357	0.809	0.523
$hPSO$	0.848	0.530	0.826	0.278	0.288	0.681	0.596	0.754	0.535
$hWOA$	0.862	0.467	0.830	0.408	0.127	0.667	0.664	0.672	0.460
hBA	0.853	0.466	0.879	0.647	0.505	0.731	0.767	0.735	0.574

d. ROC of Discrete Datasets

The performance measure results in term of ROC for Naïve Bayes classifier are shown in Table 12. ROC of five datasets out of nine datasets achieved good results after discretization process. From these five datasets, four datasets are using hBA approach and one dataset is using $hWOA$ approach.

Table 13 shows the ROC of seven datasets out of nine datasets which are improved after discretization process. hBa technique is able to improve five datasets out of seven datasets. Both techniques, $hWOA$ and dk are able to improve one dataset. By using Decision Tree, seven datasets out of nine datasets are improved after discretization process as shown in Table 14. hBA improved six out of seven datasets, dK and $hWOA$ improved one out of six datasets. In this experiment, dBA and $hWOA$ obtained the same result for DS1.

Table 12. ROC of Naïve Bayes for Discrete Datasets

	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9
$cont$	0.878	0.492	0.945	0.943	0.990	0.875	0.963	0.786	0.816
dk	0.768	0.494	0.839	0.679	0.960	0.775	0.773	0.473	0.709
$hPSO$	0.843	0.481	0.912	0.766	0.967	0.832	0.894	0.760	0.669
$hWOA$	0.878	0.492	0.945	0.943	0.959	0.875	0.963	0.786	0.816
hBA	0.910	0.493	0.947	0.934	0.955	0.861	0.959	0.788	0.803

Table 13. ROC of k-Nearest Neighbors for Discrete Datasets

	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9
$cont$	0.804	0.584	0.925	0.919	0.869	0.816	0.857	0.862	0.685
dk	0.751	0.498	0.802	0.610	0.665	0.696	0.662	0.482	0.731
$hPSO$	0.823	0.491	0.782	0.598	0.616	0.737	0.665	0.795	0.670
$hWOA$	0.804	0.584	0.925	0.919	0.869	0.816	0.857	0.862	0.685
hBA	0.818	0.476	0.944	0.914	0.910	0.821	0.882	0.826	0.767

Table 14. ROC of Decision Tree for Discrete Datasets

	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9
$cont$	0.858	0.494	0.939	0.870	0.759	0.858	0.901	0.901	0.733
dk	0.755	0.500	0.855	0.604	0.374	0.733	0.662	0.488	0.760
$hPSO$	0.881	0.494	0.925	0.661	0.667	0.820	0.801	0.870	0.740
$hWOA$	0.858	0.494	0.939	0.870	0.759	0.858	0.901	0.901	0.733
hBA	0.881	0.470	0.954	0.844	0.779	0.865	0.987	0.868	0.770

5. Conclusion

In this paper, one new optimize discretization approach was proposed. The experiment was done to compare effectiveness of the proposed approach, hBA to improve classification performance over discrete datasets that were generated with continuous dataset, also discrete dataset that using another approach where; dk, $hPSO$ and $hWOA$ approach. From the experiment, its proof that optimization algorithm employ during data preparation step able to solve classification problem. Also, the results show the optimization algorithm was able to improve the classification performance in terms of accuracy, recall, f-measure and ROC.

This research shows that hBA outperforms almost all datasets compared to continuous dataset and discrete dataset which uses another approach. Thus, BA is a good discretization approach, where it is able to maintain the accuracy of the classification algorithm and avoid loss of information. However, the proposed approach still have room for improvement in future research, since hBA was not able to improve classification performance in

all datasets. In the future, this research will be conducted on feature selection by using optimization algorithm especially Bat Algorithm. Optimization algorithm may examine with mix type of attributes and imbalance datasets.

6. Acknowledgements

The authors would like to thank Ministry of Higher Education, Malaysia for supporting this research under Fundamental Research Grant Scheme Vot K213 (FRGS/1/2019/ICT02/UTHM/02/2) and Universiti Tun Hussein Onn Malaysia for Multidisciplinary Research, Vot H511.

References

1. Aboubi, Y., Drias, H., & Kamel, N. (2016). BAT-CLARA: BAT-inspired algorithm for Clustering LARge Applications. *IFAC-Papers OnLine*, *49*(12), 243–248. https://doi.org/https://doi.org/10.1016/j.ifacol.2016.07.607
2. AlMuhaideb, S., & Menai, M. E. B. (2016). An individualized preprocessing for medical data classification. *Procedia Computer Science*, *82*, 35–42.
3. Azham Hussain, S.V Manikanthan, Padmapriya T. and Mahendran Nagalingam. "Genetic algorithm based adaptive offloading for improving IoT device communication efficiency". Wireless Network, August, 2019. DOI: 10.1007/s11276-019-02121-4.
4. Bekdaş, G., Nigdeli, S. M., & Yang, X.-S. (2018). A novel bat algorithm based optimum tuning of mass dampers for improving the seismic safety of structures. *Engineering Structures*, *159*, 89–98. https://doi.org/https://doi.org/10.1016/j.engstruct.2017.12.037
5. Cano, A., Luna, J. M., Gibaja, E. L., & Ventura, S. (2016). LAIM discretization for multi-label data. *Information Sciences*, *330*, 370–384. https://doi.org/https://doi.org/10.1016/j.ins.2015.10.032
6. Chen, H., Xu, Y., Wang, M., & Zhao, X. (2019). A balanced whale optimization algorithm for constrained engineering design problems. *Applied Mathematical Modelling*, *71*, 45–59. https://doi.org/https://doi.org/10.1016/j.apm.2019.02.004
7. Eberhart, R., & Kennedy, J. (1995). A new optimizer using particle swarm theory. *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, 39–43. https://doi.org/10.1109/MHS.1995.494215
8. Fikri, M. N., Hassan, M. F., & Tran, D. C. (2020). The impact of fuzzy discretization's output on classification accuracy of random forest classifier. *International Journal of Advanced Trends in Computer Science and Engineering*, *9*(3), 3950–3956. https://doi.org/10.30534/ijatcse/2020/218932020
9. Gao, M.-L., Shen, J., Yin, L.-J., Liu, W., Zou, G.-F., Li, H.-T., & Fu, G.-X. (2016). A novel visual tracking method using bat algorithm. *Neurocomputing*, *177*, 612–619. https://doi.org/https://doi.org/10.1016/j.neucom.2015.11.072
10. Gharehchopogh, F. S., & Gholizadeh, H. (2019). A comprehensive survey: Whale Optimization Algorithm and its applications. *Swarm and Evolutionary Computation*, *48*, 1–24. https://doi.org/https://doi.org/10.1016/j.swevo.2019.03.004
11. Got, A., Moussaoui, A., & Zouache, D. (2020). A guided population archive whale optimization algorithm for solving multiobjective optimization problems. *Expert Systems with Applications*, *141*, 112972. https://doi.org/https://doi.org/10.1016/j.eswa.2019.112972
12. HACIBEYOĞLU, M., & IBRAHIM, M. (2016). Comparison of the effect of unsupervised and supervised discretization methods on classification process. *International Journal of Intelligent Systems and Applications in Engineering*, *0*(0 SE-Research Article). https://doi.org/10.18201/ijisae.267490
13. Han, Y., Zhu, L., Cheng, Z., Li, J., & Liu, X. (2020). Discrete Optimal Graph Clustering. *IEEE Transactions on Cybernetics*, *50*(4), 1697–1710. https://doi.org/10.1109/TCYB.2018.2881539
14. Hassanien, A. E., Tharwat, A., & Own, H. S. (2017). Computational model for vitamin D deficiency using hair mineral analysis. *Computational Biology and Chemistry*, *70*, 198–210. https://doi.org/10.1016/j.compbiolchem.2017.08.015
15. Imran, M., Hashim, R., & Khalid, N. E. A. (2013). An Overview of Particle Swarm Optimization Variants. *Procedia Engineering*, *53*, 491–496. https://doi.org/10.1016/j.proeng.2013.02.063
16. Kaur, T., Saini, B. S., & Gupta, S. (2018). An optimal spectroscopic feature fusion strategy for MR brain tumor classification using Fisher Criteria and Parameter-Free BAT optimization algorithm. *Biocybernetics and Biomedical Engineering*, *38*(2), 409–424. https://doi.org/https://doi.org/10.1016/j.bbe.2018.02.008
17. Khooban, M.-H., & Javidan, R. (2016). A novel control strategy for DVR: Optimal bi-objective

structure emotional learning. *International Journal of Electrical Power & Energy Systems*, *83*, 259–269. https://doi.org/https://doi.org/10.1016/j.ijepes.2016.04.014

18. Lavangnananda, K., & Chattanachot, S. (2017). Study of discretization methods in classification. *2017 9th International Conference on Knowledge and Smart Technology: Crunching Information of Everything, KST 2017*, *February*, 50–55. https://doi.org/10.1109/KST.2017.7886082
19. Li, X., Wu, X., Xu, S., Qing, S., & Chang, P.-C. (2019). A novel complex network community detection approach using discrete particle swarm optimization with particle diversity and mutation. *Applied Soft Computing*, *81*, 105476. https://doi.org/https://doi.org/10.1016/j.asoc.2019.05.003
20. MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*, 281–297. https://projecteuclid.org/euclid.bsmsp/1200512992
21. Madhu, G., Rajinikanth, T. V, & Govardhan, A. (2014). Improve the Classifier Accuracy for Continuous Attributes in Biomedical Datasets Using a New Discretization Method. *Procedia Computer Science*, *31*, 671–679. https://doi.org/https://doi.org/10.1016/j.procs.2014.05.315
22. Marichelvam, M. K., Geetha, M., & Tosun, Ö. (2020). An improved particle swarm optimization algorithm to solve hybrid flowshop scheduling problems with the effect of human factors – A case study. *Computers & Operations Research*, *114*, 104812. https://doi.org/https://doi.org/10.1016/j.cor.2019.104812
23. Maryono, D., Hatta, P., & Ariyuana, R. (2018). Implementation of numerical attribute discretization for outlier detection on mixed attribute dataset. *2018 International Conference on Information and Communications Technology (ICOIACT)*, 715–718. https://doi.org/10.1109/ICOIACT.2018.8350795
24. Mirjalili, S., & Lewis, A. (2016). The Whale Optimization Algorithm. *Advances in Engineering Software*, *95*, 51–67. https://doi.org/10.1016/j.advengsoft.2016.01.008
25. More, A. S., & Rana, D. P. (2020). An Experimental Assessment of Random Forest Classification Performance Improvisation with Sampling and Stage Wise Success Rate Calculation. *Procedia Computer Science*, *167*, 1711–1721.
26. Nagarajan, G., & Dhinesh Babu, L. D. (2019). A hybrid of whale optimization and late acceptance hill climbing based imputation to enhance classification performance in electronic health records. *Journal of Biomedical Informatics*, *94*, 103190. https://doi.org/https://doi.org/10.1016/j.jbi.2019.103190
27. Nguyen, B. H., Xue, B., & Zhang, M. (2020). A survey on swarm intelligence approaches to feature selection in data mining. *Swarm and Evolutionary Computation*, *54*, 100663. https://doi.org/https://doi.org/10.1016/j.swevo.2020.100663
28. Qu, B., Li, C., Liang, J., Yan, L., Yu, K., & Zhu, Y. (2020). A self-organized speciation based multi-objective particle swarm optimizer for multimodal multi-objective problems. *Applied Soft Computing*, *86*, 105886. https://doi.org/https://doi.org/10.1016/j.asoc.2019.105886
29. Shafiq, M., Tian, Z., Bashir, A. K., Jolfaei, A., & Yu, X. (2020). Data mining and machine learning methods for sustainable smart cities traffic classification: A survey. *Sustainable Cities and Society*, *60*(February), 102177. https://doi.org/10.1016/j.scs.2020.102177
30. Slowik, A., & Kwasnicka, H. (2017). Nature inspired methods and their industry applications—swarm intelligence algorithms. *IEEE Transactions on Industrial Informatics*, *14*(3), 1004–1015.
31. Tahir, H. M., Said, A. M., Osman, N. H., Zakaria, N. H., Sabri, P. N. '. M., & Katuk, N. (2016). Oving K-Means Clustering using discretization technique in Network Intrusion Detection System. *2016 3rd International Conference on Computer and Information Sciences (ICCOINS)*, 248–252. https://doi.org/10.1109/ICCOINS.2016.7783222
32. Tang, H., Sun, W., Yu, H., Lin, A., & Xue, M. (2020). A multirobot target searching method based on bat algorithm in unknown environments. *Expert Systems with Applications*, *141*, 112945. https://doi.org/https://doi.org/10.1016/j.eswa.2019.112945
33. Uçar, M. K. (2020). Classification Performance-Based Feature Selection Algorithm for Machine Learning: P-Score. *IRBM*.
34. Xu, K., Pedrycz, W., Li, Z., & Nie, W. (2020). Optimizing the prototypes with a novel data weighting algorithm for enhancing the classification performance of fuzzy clustering. *Fuzzy Sets and Systems*. https://doi.org/https://doi.org/10.1016/j.fss.2020.05.009
35. Yang, X., Chen, W., Li, A., Yang, C., Xie, Z., & Dong, H. (2019). BA-PNN-based methods for power transformer fault diagnosis. *Advanced Engineering Informatics*, *39*, 178–185. https://doi.org/https://doi.org/10.1016/j.aei.2019.01.001
36. Yang, X. S. (2010). A new metaheuristic Bat-inspired Algorithm. *Studies in Computational Intelligence*, *284*, 65–74. https://doi.org/10.1007/978-3-642-12538-6_6

37. Zamudio-Reyes, R., Cruz-Ramírez, N., & Mezura-Montes, E. (2017). A Multivariate Discretization Algorithm Based on Multiobjective Optimization. *2017 International Conference on Computational Science and Computational Intelligence (CSCI)*, 375–380. https://doi.org/10.1109/CSCI.2017.62
38. Zhao, R., Qu, Y., Deng, A., & Zwiggelaar, R. (2018). Inconsistency Measure Associated Discretization Methods to Network-based Intrusion Detection. *2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 1–6. https://doi.org/10.1109/FUZZ-IEEE.2018.8491570
39. Zhou, Y., Kang, J., Kwong, S., Wang, X., & Zhang, Q. (2021). An evolutionary multi-objective optimization framework of discretization-based feature selection for classification. *Swarm and Evolutionary Computation*, *60*, 100770. https://doi.org/https://doi.org/10.1016/j.swevo.2020.100770