IMPLEMENTATION AND PERFORMANCE ANALYSIS OF NATURE INSPIRED ALGORITHM

Neha Kathuria, Aayushi Bansal

Department of Computer Science & Engineering
Guru Jambeshwar University of Science &Technology, Hissar, Haryana -125001.
Email: kathurianeha90@gmail.com, aayushib2@gmail.com

Abstract: The feature relevancy or usefulness would be based on a predictive model which is trained on the training data. An extreme number of features carry the problem of memory usage in order to represent the dataset. In this research work, the feature selection using Bacterial Foraging optimization algorithm is performed on two datasets i.e., Iris and Diabetic datasets and finding their accuracy by applying the two classification algorithms called Naïve Bayes and KNN. The studied method consists of two steps, firstly the features are selected using the BFO algorithm and accuracy is computed with the help of classifiers. The results of the experiment are compared with accuracy of original datasets without feature selection in WEKA. The feature selection using BFO yields better results. The accuracy is increased after selecting the relevant features.

INTRODUCTION

Due to the presence of a large amount of data and need for turning such a huge data into knowledge and useful information, Data Mining has secured an attention in this field. A huge amount of raw data is present in the information industry and this raw data has to be converted into useful information the absence of which will make it less useful, or not useful at all. “Necessity is the mother of invention.” Necessity to uncover the hidden patterns and make the data ‘information rich’ attracted the attention towards the Data Mining. Data Mining is a promising field. Data Mining is a task of extracting and discovering the hidden interesting patterns from a huge amount of data. Data Mining is an essential step in the process of decision making and adding the information to our knowledge base. [1]

Data Mining: Data Mining is the process of solving the problems of evaluating the useful information/data already present in a large amount of database. Data Mining is used to uncover concealed patterns for evaluation. The data mining is a step in the knowledge discovery process through which the user can interact. The interesting patterns evaluated are presented to the user. [1] As the number of dimensions and the size of data increases, the data analysis needs to be performed. The process of extracting the knowledge from the dataset is referred to as KDD (Knowledge Discovery from data).

Feature Selection: Feature selection is a process of selecting the useful and relevant features in the data set. The feature relevancy or usefulness would be based on a predictive model which is trained on the training data. Feature selection is important as it helps in reducing the size of the data and complexity of the model and makes it simpler and easily understandable. The feature selection aims to minimize the cost and improve the performance of the model. The selection of attributes would be determined based on some evaluation measure i.e., information gain, gain ratio, PCA etc. [8]

NATURE INSPIRED ALGORITHM

Nature has a rich source of inspiration. Nature tends to favor the animals with successful or good foraging strategies and eliminate the ones with poor strategies. The activity of foraging by animals is known as an optimization process. In the foraging process, animals maximize their energy by taking actions per unit time. They have the tendency to make good decisions and finding the best solution in the changing environment. As the name suggests, these algorithms have been developed by drawing inspiration from nature. These algorithms are constantly inspiring the developers and scientist. They can be used to solve the real-world optimization problems and find out the global optimal solution within the search space. [5]

Bacteria Foraging Optimization: Bacterial Foraging Optimization Algorithm is based on the behavior of E Coli bacteria which is present in the human intestine. The E Coli bacteria try to search for food and avoid other substances because of its control system. The bacteria search for the nutrients and move in a direction of increasing nutrients by taking small steps and also communicate with each other by sending signals. The two basic operations performed by the bacteria at the time of foraging are swim and tumble. [4] The Bacterial Foraging Optimization Algorithm has the advantage that it takes less computational time, has the less computational burden, number of objective functions can be handled and had global convergence. [6]

LITERATURE REVIEW

Iztok Fister *et al.* [3] represented the various nature-inspired algorithms. They classified the existing algorithms into four main categories. These are Swarm Intelligence based, Bio-inspired but not SI based, Physics and Chemistry based, and others. This classification is not unique because it largely depends on the focus, perspective, and emphasis maybe. The emphasis or focus is about search path, the interaction of multiple agents, updating equations, and source of inspiration.

Agarwal and Mehta [6] presented the review of 12 nature inspired algorithms and highlight the features based on their input parameters, applications, and mechanisms. The paper said that nature inspired algorithms to simulate the behavior of nonliving and living things and inspired from nature's ecosystem. It gains attention of the researcher towards various toolboxes available and also studied the efficiency of nature inspired algorithms over benchmark test problems in order to solve the "curse of dimensionality" problem.

Passino [4] proposed a new evolutionary computation technique known as Bacterial Foraging Optimization Algorithm (BFOA) in 2002. In BFOA, the foraging behavior, i.e., methods for locating, handling, and ingesting food, of E. coli bacteria is mimicked. In the process of foraging, E. coli bacteria undergo four stages, namely, chemo-taxis, swarming, reproduction, and elimination and dispersal. The bacteria can move in two different directions, i.e., swim (unit movement in the same direction) and tumble (unit movement in a different direction). The idea of the BFO is based on the fact that the animals with poor foraging strategies will eliminate and favor the propagation of genes of those animals that have successful foraging strategies.

WJ Tang [7] in 2006 studied the Bacterial Foraging Algorithm for the optimization in dynamic environments called DBFA. The searching and convergence ability of dynamic environments is desired. The existing BFO uses the artificial reproduction process for the convergence speed but it is not capable for dynamic environments due to lack of diversity. The bacteria adapt the changing environment in DBFA due to the selection scheme that DBFA adopts. The DBFA was compared with the BFA in many aspects and DBFA shows the satisfactory performance.

The modification in the individual steps may improve the model's performance. A.Abraham and A.Biswas [8] provided a simple analysis of the single step that is used in BFOA, i.e., reproduction. The analysis is focused on the reproduction in a simple two-bacterial system working on a one-dimensional fitness landscape. The analysis shows that the contribution of reproduction event leads the quick convergence of bacteria to the near-optimum solution.

Xiaohui Yan *et al.* [9] presents the improved BFO to overcome the shortcoming of classical BFO that the optimization ability is not so good in the later. The comparison of classical BFO, GA, and PSO are done. The proposed BFOLS algorithm is a powerful algorithm for optimization. In the new algorithm, a lifecycle model is found in which bacteria could split, die, and migrate in the foraging process dynamically. It offers improvements over classical BFO and shows competitive performances compared with other algorithms on higher-dimensional problems.

Jun Li *et al.* [10] analyzed the BFO on its various operations like elimination and dispersal to avoid the escape in local minima and chemotaxis to adjust the step length. To improve the accuracy and efficiency of the algorithm, an improved BFO was designed. The results indicate that the improved BFO algorithm is superior to the basic BFO algorithm and it improves the precision and convergence speed.

IMPLEMENTATION

For implementation, we are using the MATLAB R2013a software. The implementation of feature selection using Bacterial Foraging Optimization is done based on the fitness function used i.e. Information gain. The two datasets are taken and apply BFO algorithm on them for selecting the relevant features. The accuracy of the model is defined by the classification algorithm i.e., Naïve Bayes and KNN. The graph in Fig 4.1 shows the Indexes of minimum Vs Minimum Value i.e. no. of selected features with their minimum optimal value Vs no. of trails to select a feature.

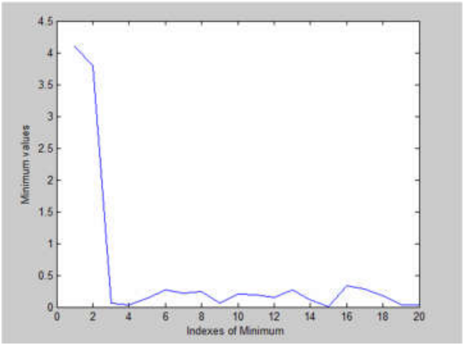

Fig 1: Selected Features using BFO

Variation in the Values

If we vary the two variables of BFO i.e., Swim length and Chemo-tactic value, the accuracy of the system, according to the values, may vary. Here we have taken 20 as Chemo-tactic value of bacteria and vary the Swim length as 5, 10, and 15. The graphs of the selected features with different values are shown in the fig. below:

When we vary the swim length value, the minimum value of the feature decreases. In Fig 4.2, the graph shows the random values taken using BFO. Due to the random values of indexes, the minimum value varies. It randomly increases and decreases and finally decreases with the increasing indexes.

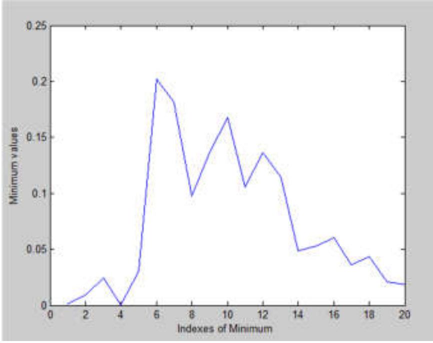

Fig 2: Swim Length 5, Chemo-tactic value 20

When we are taking the swim length as 10, the graph shows the random values of bactria as in the previous case. The minimum value increases and decreases according to the indexes. At the end, it goes increasing at the maximum value of indexes.

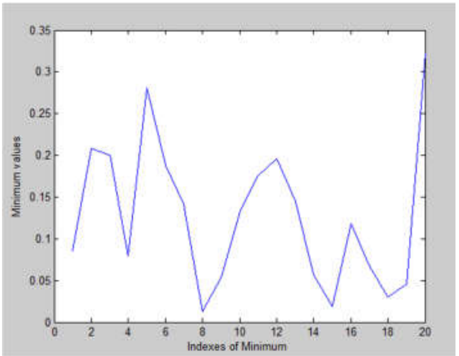

Fig 3: Swim Length 10, Chemo-tactic value 20

In fig 4.4, the graph show the drastically decrease in the minimum value as the indexes increases. And it contains the minimum variation in the values till the last index.

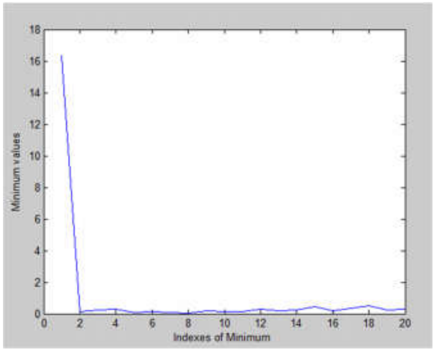

Fig 4: Swim length 15, Chemo-tactic value 20

After varying the Swim length and Chemo-tactic values of the bacteria, the minimum value of optimal solution vary and it is concluded that the wim length with value 15 shows the better solution than the previous two.

EXPERIMENTAL RESULTS

The experiments were done on the two datasets. These are Iris dataset and Diabetic dataset both have 4 features. The accuracy, precision, recall, and error rate of both the datasets are estimated by applying Naïve Bayes and KNN classification algorithm on both of them individually. The confusion matrices are shown below in tables.

IRIS DATASET

Firstly the whole work on Iris dataset is concluded in the form of confusion matrix which is shown in the table given below:

	Naïve Bayes		KNN	
Confusion Matrix	50	0	50	50
	0	50	0	0

Table 1: Confusion Matrix for both classifiers on Iris dataset

The accuracy of both the classifiers is shown in the graph. From the graph, it is clear that the accuracy of Naïve Bayes classification algorithm is more than KNN.

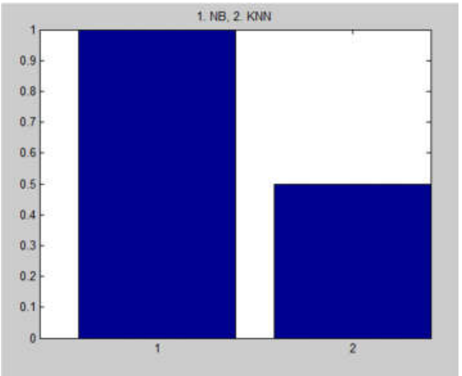

Fig 5: Comparison of NB and KNN based on their Accuracy

ROC analysis of KNN and Naïve Bayes on Iris dataset is shown in the given graphs. To draw a ROC curve, only the true positive rate (TPR) and false positive rate (FPR) are needed. The TPR defines how many correct positive results occur among all positive samples available during the test. FPR, on the other hand, defines how many incorrect positive results occur among all negative samples available during the test. The best possible prediction method would yield a point in the upper left corner or coordinate (0,1) of the ROC space, representing 100% sensitivity (no false negatives) and 100% specificity (no false positives). The (0,1) point is also called a *perfect classification*. A random guess would give a point along a diagonal line from the left bottom to the top right corners. Therefore the closer the ROC curve is to the upper left corner, the higher the overall accuracy of the test.

The area under the curve is a measure of text accuracy. The closer the curve follows the left-hand border and then the top border of the ROC space, the more accurate the test. The closer the curve comes to the 45-degree diagonal of the ROC space, the less accurate the test.

The plot shows a diagonal line where for every true positive of a model; we are just as likely to encounter a false positive. This diagonal line also shows the random guessing of the values. The upper left corner of the curve gives the best solution i.e., it gives the less false positive values.

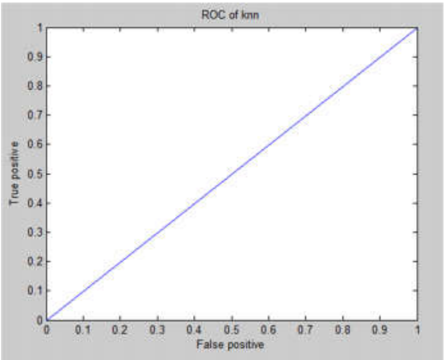

Fig 6: ROC of KNN on Iris dataset

ROC of Naïve Bayes on Iris dataset is given with graph. It shows the reverse nature than the actual ROC curve. This means that the Naïve Bayes is not as good when compared with KNN. it is concluded that the more FP values. The FP values increases on a constant TP value. And at the end, TP value goes on inceasing at some constant FP value.

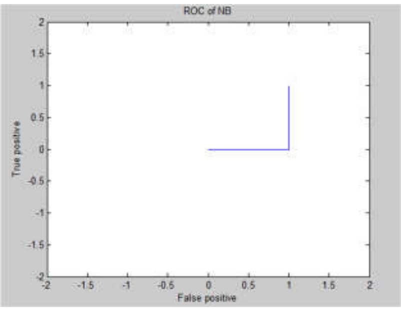

Fig 7: ROC of Naïve Bayes on Iris dataset

<u>DIABETIC DATASET</u>

After the first dataset, we are using the second dataset i.e., Diabetic dataset and concluded it in the form of confusion matrix as shown below:

	Naïve Bayes		KNN	
Confusion Matrix	118	3	118	32
	0	29	0	0

Table 2: Confusion Matrix for both classifiers on Diabetic dataset

The confusion matrix of Diabetic dataset for both the classifiers is combined in one table so that we can easily compare the two. The confusion matrix is the base to find the accuracy, precision, recall, and error rate for both the classifiers.

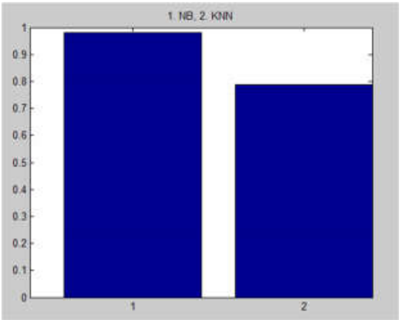

Fig 8: Comparison of NB and KNN based on their accuracy

The comparison between the Naïve Bayes and KNN based on their accuracy is given in the graph. As the previous one, the accuracy of Naïve Bayes overcome the same of KNN.

The ROC of KNN for both the datasets is same. The diagonal line shows that every positive rate encounter the same negative rate. As the previous dataset, the diagonal line shows the random guessing of the values. The more upper left corner of curve is, the more the accuracy of the classifier on particular dataset.

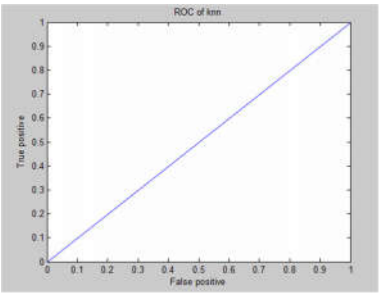

Fig 9: ROC of KNN on diabetic dataset

The ROC of Naïve Bayes on Diabetic dataset is showing the perfect shape of ROC curve. This means that the curve moves steeply up from zero and more horizontal. The model is less accurate if the line goes near the diagonal. The graph shows the most accurate values as FP values are less and it will predict the values truly and correctly.

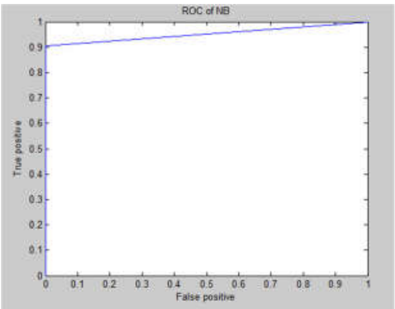

Fig 10: ROC of Naïve Bayes on diabetic dataset

RESULTS ANALYSIS

The accuracy, precision, recall, and error rate of the model is calculated based on the confusion matrix.

The figures and graphs of each of them is shown accordingly.

ACCURACY

The accuracy matrix table is given here for both the datasets and the classifiers.

	Naïve Bayes	KNN
Iris	1	0.5
Diabetic	0.975	0.7867

Table 3: Accuracy matrix

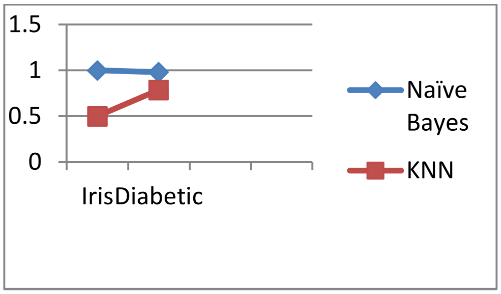

Fig 11: Accuracy of both datasets on different classifier

We are comparing the individual classifier values on both the datasets. The graph show that the accuracy of Naïve Bayes is better on Iris dataset but the accuracy of KNN is less on Iris than on Diabetic dataset. .

PRECISION

The precision matrix table is given here for both the datasets and the classifiers.

	Naïve Bayes	KNN
Iris	1	0.5
Diabetic	0.98	0.7867

Table 4: Precision matrix

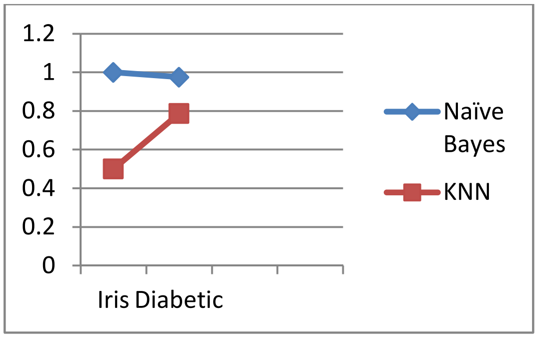

Fig 12: Precision of both datasets on different classifier

Naïve Bayes is compared on the datasets based on the precision and it shows better results on Iris dataset same as the accuracy factor. Similarly in KNN, values are less in Iris dataset as the previous accuracy factor.

RECALL

The recall matrix table is given here for both the datasets and the classifiers

	Naïve Bayes	KNN
Iris	1	1
Diabetic	1	1

Table 5: Recall matrix

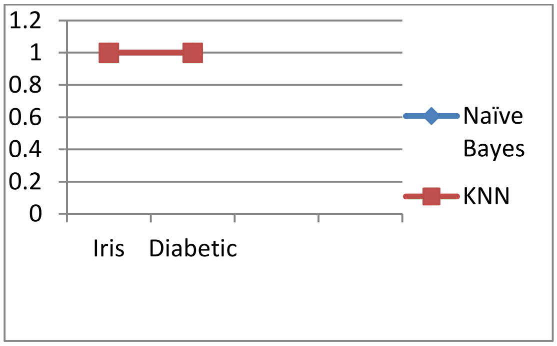

Fig 13: Recall of both datasets on different classifier

The graph show that the recall of Naïve Bayes and KNN is same in both the datasets. There is no variation in the values of recall on both the Naïve Bayes and KNN classification algorithm.

ERROR RATE

The error rate matrix table is given here for both the datasets and the classifiers.

	Naïve Bayes	KNN
Iris	0	0.5
Diabetic	0.02	0.213

Table 6: Error Rate matrix

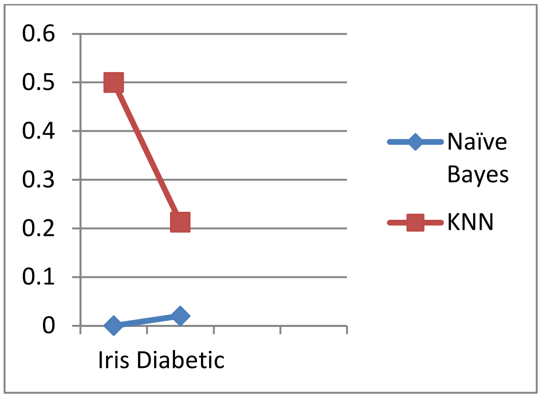

Fig 14: Error rate of both datasets on different classifier

The graph shows that the error rate of Naïve Bayes is less in iris dataset. In case of KNN, error rate is less in Diabetic dataset.

The results show that the Naïve Bayes classifier gives better performance in case of Iris dataset. Similarly, KNN gives gives better performance on Diabetic dataset.

COMPARISON OF RESULTS

The accuracy of two datasets with both the classifiers in MATLAB is compared with accuracy of same in WEKA. Firstly we are showing the separate accuracy of both the classifiers on different datasets. The accuracy of datasets in WEKA is obtained without using feature selection prior to the classifier.

ACCURACY IN WEKA

	Naïve Bayes	KNN
Iris	95.53	94.67
Diabetic	75.75	70.19

Table 7: Accuracy matrix in WEKA

Note: the values are in percentage.

We have to compare the accuracies of both the classifiers on individual datasets. The accuracies of these classifiers have been obtained using BFO for feature selection and without using BFO. The two tables of iris and diabetic datasets are shown below with their comparison graphs.

IRIS DATASET

	Naïve Bayes	KNN
Using BFO	100	50
Without using BFO	95.53	94.67

Table 8: Comparison of accuracy on Iris dataset

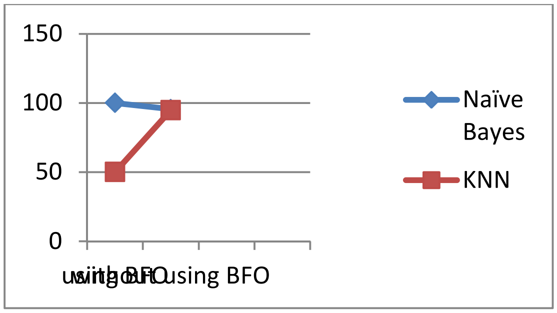

Fig 15: Comparison of accuracy on Iris dataset

The above graph shows the better accuracy using BFO than without BFO on Naïve Bayes classifier. The comparison results are on the Iris dataset. But it is opposite in case of KNN. It shows better results without using BFO.

DIABETIC DATASET

	Naïve Bayes	KNN
Using BFO	98	78.67
Without using BFO	75.75	70.19

Table 9: Comparison of accuracy on Diabetic dataset

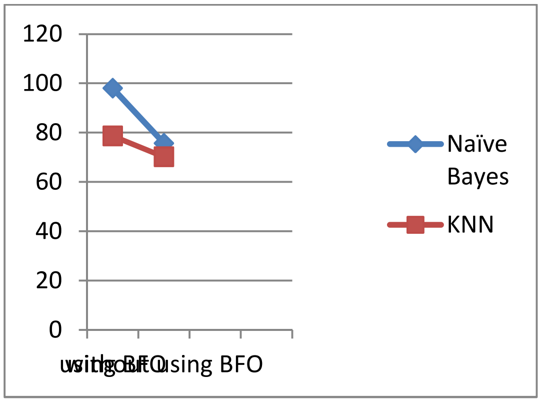

Fig 16: Comparison of accuracy on Diabetic dataset

The accuracies are compared on Diabetic dataset using BFO and without using BFO on both the classifiers. The above graph shows the better accuracy using BFO than without BFO on both Naïve Bayes and KNN classifier. The comparison results are on the Diabetic dataset.

When we compare the accuracies using BFO and without using BFO individually on different datasets, the comparison result goes in the favor of BFO. The BFO gives better accuracy on Iris dataset in case of Naïve Bayes classification algorithm i.e., it gives 100% accuracy but it does not perform as better as Naïve Bayes in case of KNN. The diabetic dataset shows better accuracy of Using BFO in both classifier cases. So, the feature selection using BFO increases the accuracy of the dataset than without using feature selection when compared in WEKA. The performance of the classifiers shows that the Naïve Bayes perform better on Iris dataset and KNN gives better results on Diabetic dataset.

The overall result shows that the classification algorithm perform better and increase the accuracy of datasets when the feature selection is performed using BFO.

CONCLUSION

In this research work, the Bacterial Foraging Optimization based features selection on the dataset has been studied and compared it with the dataset in WEKA. The accuracy of both is studied based on the classification algorithms. The results are compared with their accuracy. The individual accuracy of Naïve bayes and KNN is also shown in the work. The result demonstrates that one of the two classification algorithms perform better than the other i.e., Naïve Bayes shows better accuracy on the datasets than KNN algorithm. The results are calculated individually on datasets with the graphs and compare them with each other. When we compare the accuracies using BFO and without using BFO individually on different datasets, the comparison result goes in the favor of BFO. The BFO gives better accuracy on Iris dataset in case of Naïve Bayes classification algorithm i.e., it gives 100% accuracy. The final result is obtained by combining the results and plot graphs to find the better accuracy of the dataset. It is observed that the Naïve Bayes demonstrate the accuracy of KNN on both the datasets The overall result shows that the classification algorithm perform better and increase the accuracy of datasets when the feature selection is performed using BFO.

REFERENCES

[1] J. Han and M. Kamber, *Data Mining: concepts and techniques, 2nd edition*, Morgan Kaufmann, 2009.

[2] Sonu Rani, Dharminder kumar, Sunita Beniwal, “Improving Medical Diagnosis using filter and Wrapper Techniques”, *International Journal of Advanced Research in Computer And Communication Engineering*, vol. 5, pp 438-440,August 2016.

[3] IztokFister Jr., Xin-She Yang, IztokFister, Janez Brest, and DusanFister, “A Brief Review of Nature-

Inspired Algorithms for Optimization", *ELEKTROTEHNISKI VESTNIK 80(3*), pp. 1–7*,* 2013, English Edition.

[4] KM Passino, "Biomimicry of Bacterial Foraging for Distributed Optimization and Control", *IEEE Control Systems Magazine*, pp. 52–67, 2002.

[5] Chapter 5 Bacterial Foraging Optimization, pp. 62-73 http://shodhganga.inflibnet.ac.in/bitstream/10603/24455/10/10_chapter%205.pdf

[6] P. Agarwal and S. Mehta, "Nature-Inspired Algorithms: State-of-Art, Problems and Prospects", *International Journal of Computer Applications,* vol. 100, no.14, pp. 14-21, August 2014.

[7] W. J. Tang, Q. H. Wu, and J. R. Saunders, "Bacterial Foraging Algorithm For Dynamic Environments", *IEEE Congress on Evolutionary Computation*, pp. 3124-3130, July 16-21, 2006.

[8] A. Abraham, A. Biswas, and S. Dasgupta et al. "Analysis of reproduction operator in bacterial foraging optimization algorithm", *IEEE World Congress on Computational Intelligence*, pp. 1476–1483, June 2008.

[9] Xiaohui Yan, Yunlong Zhu, Hao Zhang, Hanning Chen, and Ben Niu, "An Adaptive Bacterial Foraging Optimization Algorithm with Lifecycle and Social Learning", *Hindawi Publishing Corporation Discrete Dynamics in Nature and Society*, pp. 1-22, 2012.

[10] Jun Li, Jianwu Dang, Feng Bu, and Jiansheng Wang, "Analysis and Improvement of the Bacterial Foraging Optimization Algortithm", *Journal of Computing Science and Engineering*, vol. 8, no. 1, pp. 1-10, March 2014,