Multi Objective Feature Selection using Non-dominated Sorting based Evolutionary Algorithm

By

Ved Prakash

(Roll No.: 1903016)

Under the Guidance of

Dr. Sumit Mishra

Report submitted of 2nd progress seminar

Department of Computer Science and Engineering

Indian Institute of Information Technology Guwahati

Guwahati – 781015, INDIA.

January, 2024

Contents

Nomenclature

DE Differential Evolution

$MOFS-BDE$ Multi-objective Feature Selection Binary Differential Evolution

List of Figures

Abstract

Feature subset selection problem is one of the critical data pre-processing step to build statistical or machine learning model. Here, in this work, we have approached feature subset selection problem as multi-objective optimization problem and analyzed a non-dominated sorting based evolutionary algorithm known as binary differential evolution algorithm to solve the feature subset selection problem. We also have proposed a modified binary differential evolution algorithm for feature subset selection.

1 Introduction

Feature Selection is one of the important and critical data pre-processing step for building the statistical or machine learning models. The goal of feature selection is to get optimal subset of relevant features which represents original feature set with increased classification accuracy. Feature Selection Algorithms reduces the dimensionality of the data by removing the redundant and irrelevant features. It also results into lesser learning time, improved model interpretability, reduced overfitting and improved classification performance. Among all the feature subset selection methods, which have been proposed in the literature, broadly they can be categorized as filter, wrapper and embedded methods based on the way how method evaluate the feature. In flter methods, feature are evaluated according to their information value or statistical measures. In wrapper methods, feature subset as a whole is evaluated using some learning algorithm. Embedded methods are part of the algorithm for example Lasso Regreession.

Feature Subset Selection is computationally an NP Hard Problem [1]. For n number of features there might be $2^n - 1$ possible feature subset and theoretically a feature subset selection algorithm should evaluate all the feature subset in order to find the best feature subset. The computational cost of the problem increases exponentially with increase in number of feature. So, instead of the best feature subset, a optimal or near optimal feature subset with accuracy near to best one might be an accepted solution. Heuristc and random search methods are used for this purpose of finding the near-optimal feature subset. Due to strong exploration capability of meta heuristic search methods, many researcher have shown their interest in these methods including genetic algorithm [2], [3], ant colony optimization [4], particle swarm optimization [5], firefly algorithm [6], memetic algorithm [7], artificial bee colony [8], grasshopper optimization algorithm [9], evolutionary gravitational search [10], etc.

Feature Subset Selection methods have goal of minimizing the classification error by selecting less number of features. less number of feature is important constraint as it helps to reduce the curse of dimensionality and hence reduces the chance of over fitting by removing the redundant and irrelevant features and improve generalization on new data, it also helps to reduce the cost of acquiring the data. Hence, feature subset selection problem can be formulated as multi-objective optimization problem with two objectives, *i.e.*, to reduce the classification error and to reduce the number of attributes in the feature subset. In general, these two goals are conflicting to each other, hence the the objective is to balance the trade-off between these two conflicting goals. In recent years, a lot of research has been done and researchers have progressed towards the development of feature selection based multi-objective methods to solve these issues. The aforementioned evolutionary based algorithms have been broadly used in solving the feature subset selection problems.

Differential Evolutionary algorithm is one of the popular evolutionary algorithm used for the purpose of feature subset selection.

2 Literature Review

Evolutionary algorithms have recently been used by many researchers to solve feature subset selection problem, such as Genetic Algorithm [11, 12] , Particle swarm optimization [13, 14], Artificial Bee Colony algorithm [15], Differential Evolution [16–20] etc.

Zhu *et. al* [12] proposed a Feature Selection method incorporating Genetic Algorithm with local search (*i.e.*, forms a memetic algorithm). This algorithm combines filter ranking measure into a wrapper framework to take advantage of both filter and wrapper approaches.

Amoozegar *et. al* [13] proposed a multi-objective PSO based method named RFPSOFS that ranks the features based on their frequencies in the archive set. Then, these ranks are used to refine the archive set and guide the particles. Khushaba *et. al* [16] presents a novel feature selection method utilizing a combination of differential evolution (DE) optimization method and a proposed repair mechanism based on feature distribution measures.

Different approaches have different pros and cons [21], for example, Multi Objective Genetic Algorith based approach has slow convergence that prevents from finding the optimum pareto front. Genetic algorithm and Particle Swarm optimization based feature selection algorithms does not perform well on very high dimensional feature space.

There are many more research which has been done to solve feature selection problem using Muti-Objective evolutionary computation algorithm. A detailed survey can be found in [21]. Here, we have reviews some typical DE-based feature selection and multi-objective feature selection methods.

2.1 Differential Evolution Based Feature Selection Algorithm

This section talks about some typical Differential Evolutions based algorithms. Khushaba *et al.* [16] proposed differential evolution based feature selection method utilizing a combination of differential evolution (DE) optimization method and a repair mechanism based on feature distribution measures. However,this algorithms do not effectively reduce the dimensionality as it select the feature subset with predefined number of cardinality. Ahmed Al-Ani *et al.* [22] proposed method aims to reduce the search space using a simple and powerful, procedure that involves distributing the features among a set of wheels. Two versions of the method were presented. In the first one, the desired feature subset size is predefined by the user, while in the second the user only needs to set an upper limit to the feature subset size. Kumar *et al.* [17] introduced a DE-based feature selection algorithm so as to solve the anaphora resolution in a resource-poor language. These wroks indeeed has shown the effectiveness of the Differential Evolution algorithms for the purpose of feature subset selection problem but all of these algorithms were limited to the single objective, *i.e.*, accuracy.

Oliveira *et al.* [23] proposed a feature selection approach based on a hierarchical multi-objective genetic algorithm. The underpinning paradigm was the “overproduce and choose”. The algorithm operates in two levels. Firstly, it performs feature selection in order to generate a set of classifiers and then it chooses the best team of classifiers. Hamdani *et al.* [24] introduced the NSGA-II into feature selection, but the performance of their method has not been compared with any other

EA-based algorithms.

During recent years, researchers have shown great interest in using evolutionary algorithms for the purpose of feature subset selection problems. Xue *et al.* [25] presents first study on multi-objective particle swarm optimization (PSO) for feature selection. they investigate two PSO-based multi-objective feature selection algorithms. The first algorithm introduces the idea of nondominated sorting into PSO to address feature selection problems. The second algorithm applies the ideas of crowding, mutation, and dominance to PSO to search for the Pareto front solutions. Hancer *et al.* [15] proposed a feature selection approach based on a multi-objective artificial bee colony algorithm integrated with non-dominated sorting procedure and genetic operators. Two different implementations of the proposed approach were developed: ABC with binary representation and ABC with continuous representation.

In past decade, differential evolution algorithm have been greatly used by many researcher to solve the multiobjective optimization problem. Extending the work of [16], Xue *et al.* [18] used Differential evolution (DE) for multi-objective feature selection in classification and they proposed the algorithm DEMOFS. Sikdar *et al.* [19] propose a multiobjective differential evolution (MODE)-based feature selection and ensemble learning approaches for entity extraction in biomedical texts. However, these approaches generally suffer from the disadvantage of stagnating in the local optima, because they use the traditional DE operators, *e.g.*, the DE/rand/1/bin strategy, and generally are lack of the problem-oriented operators.

Zhang *et al.* [20] proposed a Binary differential evolution algorithm with self-learning for multi-objective feature selection. The author proposed a novel binary mutation operator based upon the probability difference and one bit purifying search to make algorithm more efficient.

2.2 MOFS-BDE Algorithm

The MOFS-BDE algorithm was prposed by Zhang *et. al.* [20] in their work "Binary differential evolution with self-learning for multi-objective feature selection".

The algorithm uses the mutation strategy such that, it selects the best one among the three random vectors as the base vector, and employs the difference between the remaining two vectors as a mutation probability to be used on the base vector to generate a mutation vector for the next crossover operator. It is illustrated in Equations (1), (2).

$$C_i = \begin{cases} \sigma & if X_{best}(t) \prec X_i(t) \\ min(1, F.(X_{r1}(t) \oplus X_{r2}(t)) + \sigma) & \text{Otherwise} \end{cases} \tag{1}$$

$$v_{i,j}t = \begin{cases} X_{best}(t) & C_{i,j} < rand \\ 1 - X_{best}(t) & \text{Otherwise} \end{cases} \tag{2}$$

The cross-over strategy is similar to standard differential evolution algorithm and it can be illustrated as follows:

$$u_{i,j}(t) = \begin{cases} v_{i,j}(t) & if U(0,1) \leq CR or j = j_{rand} \\ x_{i,j}(t) & \text{Otherwise} \end{cases} \tag{3}$$

Algorithm 1 ONE-BIT PURIFYING SEARCH (OPS)

Input: The population P_t and the set of non-dominated solution S_t.

Output: The new population P_t

1: Randomly select a solution from S_t , and set it as the reference solution, $X_{ref} = \left(x_{ref_1}, x_{ref_2}, \ldots, x_{ref_D}\right)$

2: Randomly select two feature bits, u_1 and u_2, from X_{ref}, satisfying $x_{ref_{u1}} = 1$ and $x_{ref_{u2}} = 0$

3: Judge the relative importance between the two feature bits u_1 and u_2

4: **for** an optimal solution $X_h \in S_t$ **do**

5: Generate a new individual X_h by checking the following four cases: (Without loss of generality, we suppose $u_1 \prec u_2$)

6: **Initialize** $X_h\prime = X_h$

7: **if** $X_{h,u_1} = X_{h,u_2} = 1$ **then**

8: Set $x_{h,u_2}\prime = 0$

9: **else if** $X_{h,u_1} = X_{h,u_2} = 0$ **then**

10: Set $x_{h,u_1}\prime = 1$

11: **else if** $X_{h,u_1} = 1$ **AND** $X_{h,u_2} = 0$ **then**

12: Set $x_{h,u_1}\prime = 0$

13: **else**$x_{h,u_1}\prime = 1$ and $x_{h,u_2}\prime = 0$

14: If $X_h\prime$ dominates X_h , population P_t saves $X_h\prime$ to replace X_h ; if $X_h\prime$ is dominated by X_h , the population keeps X_h unchanged; otherwise, it saves both $X_h\prime$ and X_h into P_t.

15: **if** If the size of P_t is larger than N **then**, remove $|P_t| - N$ individuals with high ranks, and reduce the crowding distances from P_t using the **Non-Dominated Sorting Algorithm** and Crowding Distance Methods.

16: Output population P_t

Algorithm 2 MOFS-BDE

Parameters: The maximal iteration times T_{max} , the population size N, the frequency of implementing OPS T_{loc} , the scale F, and the crossover probability CR.

Input: The dataset for classification.

Output: The Pareto-optimal solutions with each corresponding to a feature subset.

Initialize a number of individuals, $P_0 = \{X_1, X_2, \ldots, X_N\}$
Let $t = 0$ ▷ Iteration steps
Iteration: Set the set $P_{t+1} = \phi$
for $i = 1, 2, \ldots N$ **do**
    Randomly select three vectors from the population P_t , denoted as $X_{r1}(t)$, $X_{r2}(t)$ and $X_{r3}(t)$, $r_1 \neq r_2 \neq r_3 \neq i$
    Select the best one from the three vectors as the base vector, $X_{best}(t)$
    Generate a new mutation vector $V_{i(t)}$ for the i_{th} individual according to Eq. (1) and (2)
    Generate a trial vector $U_{i(t)}$ for the i_{th} individual according to Eq. (3):
    Evaluate the fitness of the trial vector $U_i(t)$
    Compare the i_{th} individual $X_{i(t)}$ with $U_{i(t)}$.If $X_{i(t)}$ dominates $U_{i(t)}$, save $X_i(t)$ into P_{t+1} ;if $U_{i(t)}$ dominates $X_{i(t)}$, save $U_{i(t)}$ into P_{t+1} ;otherwise, save both $U_{i(t)}$ and $X_{i(t)}$ into P_{t+1}
If the size of P_{t+1} is larger than N, remove $|P_{t+1}| - N$ individuals with higher ranks and shorter crowding distances from P_{t+1} using the method of non-dominated sorting and crowding distance ;
Step If $t/T_{loc} = t/T_{loc}$, run the problem-specific local search (refer to Algorithm 1 for details);
If $t < T_{max}$, let t ++ , and return back to Step 3; otherwise, terminate the algorithm, and output the Pareto-optimal solutions.

3 Research objectives

The goal of this work is to study and apply binary differential evolution algorithm to the banking data sets. The main objective of this work are as follow –

- A modified binary mutation operator based on probability difference and roulette-wheel based correlation correction has been designed.
- Application of multi-objective feature subset selection methods on banking datasets.

4 Research Methodology

4.1 Algorithm Selection

The MOFS-BDE 2 algorithm which was prposed by Zhang *et. al.* [20] in their work "Binary differential evolution with self-learning for multi-objective feature selection", has been selected as the foundation for this research due to its demonstrated efficacy in tackling optimization challenges involving multiple objectives and it is tailored to address the feature subset selection problem.

4.2 Enhancement Focus

The work primarily investigates and implement enhancements to the mutation strategy within the MOFS-BDE algorithm.

4.3 Enhancement Strategy

The mutation strategy has been adjusted such that, during initial generations, it focuses more on global exploration and during later generation it focuses more on local exploitation. Hence, during the initial generations it tries to search within the global search spaces so avoid getting stuck into local optima and during the later generation, though it still searching in global search space but also keeps focus around the space where best solutions till current generation exists.

4.4 Application on Banking Dataset for Credit Risk Prediction

Credit Risk assessment is one of the very key thing in banking industry. To assess the credit risk for each loans, banks calculates some of the credit risks kpi (key performance indicator) including probability of default, exposure at default, loss given default etc. These prediction are done using various modeling techniques. One of the very critical data pre-processing techniques to build the credit risk models are, feature subset selection.

During past decade, a lot of research have been done to solve feature subset selection for credit risk modeling. In this section, we briefly talk about some of the research which have been done to solve feature subset selection problem in banking, especially in credit risk area using genetic algorithm.

Oreski *et. al* [26] proposed Genetic algorithm-based heuristic for feature selection in credit risk assessment. They propose the hybrid genetic algorithm with neural networks (HGA-NN), which is used to identify an optimum feature subset. As a pre-processing step to the genetic algorithm, they have used the fast algorithms for feature ranking and earlier experience (*i.e.,* domain knowledge). Lappas *et. al* [27] proposed a machine learning approach combining expert knowledge with genetic algorithms in feature selection for credit risk assessment. Krishna *et. al* [28] proposed feature subset selection using adaptive differential evolution for banking application.

In our knowledge, the most of the research. which have been done for solving feature subset selection problem for banking and credit risk assessment using evolutionary computation are focused on one objective, *i.e,* model accuracy. *For*

example, [26] has single objective to optimize the model performance, [27] also has single objective to optimize the model performance with some constraint which are imposed based upon expert knowledge.

In this work, we have formulated feature subset selection for banking and credit risk assessment, as multi objective optimization problem. The two objectives which have been considered are, to minimize the error rate and to minimize the cardinality of the feature subset to be used for creating the model.

4.5 Dataset description

We have used German credit data [29] which is one of the well known credit risk dataset and many researchers have tested their algorithm [26] on this dataset. The dataset includes 1,000 observations and 20 attributes, such as age, gender, credit amount, duration of credit, and credit history. There are 7 numerical and 13 categorical attributes. Numerical attributes are age, credit amount, duration, installment rate, present residence since, number of existing credits at this bank, and number of people being liable to provide maintenance. Categorical attributes are gender, job, housing, savings account/bonds, checking account, credit history, purpose, personal status and sex, other debtors, property, other installment plans, telephone, and foreign worker. Each observation is labeled as either "good" or "bad" credit risk, based on whether the applicant paid back the loan in full or defaulted on it.

4.6 Experimental Setup

We have performed one hot encoding on the categorical features of the German credit dataset [29]. While doing so, to address the dummy variable trap, we have created one less dummy feature than the total number of categories available for that particular feature. The total number of features in the final dataset was 27. We split the dataset into two parts, we trained model on one and tested the performance on other.

We have used multi objective feature selection binary differential evolution (MOFS-BDE) 2 algorithm and executed it for 50 generation. The fitness function that we have used for the MOFS-BDE algorithm is, XGBoost algorithm, and the two objectives to the MOFS-BDE algorithms are "to minimize the errror rate" and "to minimize the cardinality" of the feature subset.

We also have used our proposed mutation strategy, i.e., MOFS-BDE with modified mutation strategy 10 and executed it for 50 generations. The 'average error' and 'average number of features used' during each generations have been compared, and MOFS-BDE with modified mutation strategy 10 has shown better performance.

5 Objectives on Focus in the Current Progress Review Seminar

5.1 Enhancement in Mutation Strategy

The mutation strategy of standard differential evolution algorithm is more suitable to the continuous valued search space. The same mutation strategy might not yield very good result on binary valued search space. In recent past, some of the researcher have proposed the mutation strategy for the binary differential evolution algorithm [20, 30], but there is still scope of improvement in the mutation strategy, such that it should have balance between global exploration and local exploitation.

Zorarpaci *et al.* [30], in their work "a hybrid approach of differential evolution and artificial bee colony for feature selection", uses mutation strategy mentionend in Equation (4).

$$\hat{X}_i = X_{r1} + F(X_{r2} - X_{r3}) \tag{4}$$

where F is the scaling factor predefined within the range of $[0,1]$ and X_{r1}, X_{r2} and X_{r3} are randomly chosen solution vectors which must satisfy $X_{r1} \neq X_{r2} \neq X_{r3} \neq i$, where i is the current solution vector. This indicates that population size must be chosen at least 4. The $\hat{X}_i$ is termed as the mutated vector. If it is found that the value of each parameter of the mutant vector $\hat{X}_i \geq 0.5$ then we set the parameter value to 1, otherwise the parameter value is set to 0 [19].

Zhang *et al.* [20], in their work, "Binary differential evolution with self-learning for multi-objective feature selection" proposed a mutation strategy such that, out of three randomly selected vector, the two vectors other than best vector among three, have been used to calculate the mutation probability C_i as mentioned in Equation (5). This mutation probability C_i has been used to create the mutation vector v_i as illustrated in equation 6.

$$C_i = \begin{cases} \sigma & if X_{best}(t) \prec X_i(t) \\ min(1, F.(X_{r1}(t) \oplus X_{r2}(t)) + \sigma) & \text{Otherwise} \end{cases} \tag{5}$$

$$v_{i,j}t = \begin{cases} X_{best}(t) & C_{i,j} < rand \\ 1 - X_{best}(t) & \text{Otherwise} \end{cases} \tag{6}$$

Though, some research have been done around mutation strategy specific to **binary** differential evolution algorithm, but we believe that, this area of Binary Differential Evolution still needs more attention as there are still opportunitis to improve the mutation strategy suitable to **binary** differential evolution algorithm.

In forthcoming section, we have done the analysis of mutation strategy in MOFS-BDE algorithm 2 and observe the scope to improve the same.

5.2 Analysis of Mutation Strategy in MOFS-BDE Algorithm

Equation (4) is sensitive to the order of selection as it uses subtraction operator, but Equation (5) is independant of order of selection as it uses $\oplus$ operator. Since here, we are dealing with binary values vector, *i.e.*, bit is either 0 or 1, so the value -1 does not make much sense as it does not represent anything in the binary valued feature space. In this section we have analyzed output of the mutation strategy 5 on some of the braodly possible scenarios.

- If both bits are same then $X_{r1}(t) \oplus X_{r2}(t)$ will become zero and Equation (5) will become

$$C_i = \begin{cases} \sigma & \text{if } X_{best}(t) \prec X_i(t) \\ min(1, F.(0) + \sigma) & \text{Otherwise} \end{cases}$$

  i.e.,

$$C_i = \begin{cases} \sigma & \text{if } X_{best}(t) \prec X_i(t) \\ min(1, \sigma) & \text{Otherwise} \end{cases}$$

  Since, σ is a very small turbulance coefficinet and much much less than 1 so $\min(1, \sigma) = \sigma$

$$C_i = \begin{cases} \sigma & \text{if } X_{best}(t) \prec X_i(t) \\ \sigma & \text{Otherwise} \end{cases}$$

  So, if both bits are same then c_i will be σ. In this case, the condition $C_{i,j} < rand$ will most of the time be true. as random number generator theoretically generates values between 0 and 1 with equal distribution. and since σ is closer to zero then it is more chance that, rand will be greater than σ and hence more chance that $v_{i,j}t$ in Equation (6) will have same bit value as $X_{best}(t)$.

- **If both bits are different** If both bits are different then $X_{r1}(t) \oplus X_{r2}(t)$ will become 1. then (5) will become

$$C_i = \begin{cases} \sigma & \text{if } X_{best}(t) \prec X_i(t) \\ min(1, F.(1) + \sigma) & \text{Otherwise} \end{cases}$$

  i.e.,

$$C_i = \begin{cases} \sigma & \text{if } X_{best}(t) \prec X_i(t) \\ min(1, F + \sigma) & \text{Otherwise} \end{cases}$$

  So, if both bits are different then C_i depends on scaling factor F and turbulance co-efficient σ. Higher the value of F and σ (as σ is very small turbulance co-efficient so it mainly depends on F), higher will be the value of C_i. and

hence more chance that $v_{i,j}t$ in Equation (6) will have bit value opposite to $X_{best}(t)$.

So, the issue with approach in Equation (5) are following –

- If X_{r1} and X_{r2} have same bits then it is very very high chance that the mutant vector will borrow bit from the $X_{best}t$.
- If X_{r1} and X_{r2} have different bits and learning rate F is small, then also it is very high chance that, the bits of mutant vector will be borrowed from $X_{best}t$.
- Hence the true spirit of differential evolution, i..e, to utilize the difference in population is not being utilized in these scenarios and hence it might impact the global exploration capability of the algorithm.

5.3 Modified Mutation Approach

5.3.1 Literature Survey on Mutation Approach

He and Han [31] first use logical operators to replace the original operations of the mutation mechanism in DE, and thus the individual can be evolved directly using binary string.

$$\hat{X}_i = X_{r1} \otimes F \odot (X_{r2} \oplus X_{r3}) \tag{7}$$

In mutation equation by He and Han [31], $\otimes$ indicated AND operation, $\odot$ indicates OR operation and $\oplus$ indicates XOR operations. This approach can evolve the bit string individual directly. However, by using the binary strings, there are only two different codes "0" and "1" in the population; the differential of two individuals on the same location is too negligible to operate the complicated mutation like the real coding individuals.

Peng *et al.* [32] proposed novel binary mutation strategy based on XOR logical operation mainly. As we all know, the bit coded as "0" after the XOR operation represents the common between the two selected bits; otherwise the "1" represents difference. According to the common and difference feature patterns of two randomly selected individuals, dichotomous mutation executes difference operations and the new mutation equation in is equation 8:

$$\hat{X}_i = ((X_{r1} \oplus X_{r2}) \otimes rand) \odot (!(X_{r1} \oplus X_{r2}) \otimes X_{r1}) \tag{8}$$

In Equation (8), ! denotes the NOT operator. In dichotomous mutation, the scale factor disappeared and we no longer have to worry about its value. If the bits between X_{r1} and X_{r2} are difference, then the mutation value is randomly chosen from "0" or "1"; otherwise, if the bits between X_{r1} and X_{r2} are common, then the mutation value is determined by the value of X_{r1}.

Wang *et. al* [33] proposes an weighted mutation strategy for the differential evolution algorithm. The mutation strategy is mentioned in Equation (9).

$$\hat{X}_i = \mu\left[x_{r1} + F.(x_{r2} - x_{r3})\right] + (1-\mu)\left[x_{r1} + F.(rand.x_{best} - x_{r1})\right] \tag{9}$$

In Equation (9), $\mu = \exp(-\sqrt{g})$ is a monotonically decreasing function with increase in generation g. $\mu \in [0,1]$, X_{r1}, X_{r2}, X_{r3} are three random solution other than X_i and $r_1 \neq r_2 \neq r_3 \neq i$, rand is random number between $[0,1]$ and x_{best} is optimal solution of the current generation. However, the problem is that, the mutation strategy mentioned in Equation (9) is better suited to the standard differential evolution algorithm in continuous valued search space and it may not work equally well in binary valued search space.

5.3.2 Our Approach for Mutation Strategy

In this section, we have proposed a weighted mutation strategy specially suitable for the ***binary*** differential evolution algorithm. The proposed mutation strategy is mentioned in Equation (10) and Equation (11).

$$\begin{aligned}\hat{X}_i = & \underbrace{\mu(floor((X_{r1} + X_{r2} + X_{r3})/1.5)}_{initial} + \\ & \underbrace{(1-\mu)(X_{best\_among\_3}.int((X_{r2} \odot X_{r3})) + int(\neg X_{best\_among\_3}).int((X_{r2} \otimes X_{r3}))}_{later}\end{aligned} \tag{10}$$

$$\hat{X}_i(t) = \begin{cases} 0 & \text{if } X_i(t) < 0.5 \\ 1 & \text{Otherwise} \end{cases} \tag{11}$$

In Equation (10), $\odot$ represents OR operation and $\otimes$ represents AND operation, the int function is used to convert boolean value to integer value. Equation (11) is for converting values to either zero or one. In Equation (10), $\mu = \exp(-\sqrt{g})$ and μ is a monotonically decreasing function with increase in g, *i.e.*, generation. So, during the initial generations, value of μ will be higher and later it will decrease. Hence, during the initial generation, first part of the equation will have more say in the mutation which is more focused on global exploration and during the later generations, second part of the equation will have more say on the mutation strategy, which is focused on exploitation and also tries to considered differences in the population. Below mentioned is the detailed analysis about the algorithm during initial and later generations.

- **During Initial Generation:** During initial generation, when μ will be closer to 1 then $(1-\mu)$ will be closer to 0 and hence second part of the equation will have negligible impact on the outcome and hence it can be said that, during initial generation, outcome will be guided by first part of the equation, *i.e.*, $\mu(floor((X_{r1} + X_{r2} + X_{r3})/1.5)$.
  - If none of the feature subset X_{r1}, X_{r2}, X_{r3} consist this feature then chance is less that mutation vector consist this feature.

– But, if any of X_{r1}, X_{r2}, X_{r3} consists this feature, then chances will be more that, during initial generation, mutation vector will also have this feature.

– If all of the feature subset X_{r1}, X_{r2}, X_{r3} consists this feature then chances are even stronger that mutation vector consist this feature.

- **During Later Generations:** During later generations, when μ will be closer to 0 then $(1-\mu)$ will be closer to 1 and hence first part of the equation will have negligible impact on the outcome and hence it can be said that, during later generations, outcome will be guided by second part of the equation, $(1-\mu)(X_{best\_among\_3}.int((X_{r2} \odot X_{r3})) + int(\neg X_{best\_among\_3}).int((X_{r2} \otimes X_{r3}))$

$$\hat{X}_i = \begin{cases} (1-\mu)(X_{best\_among\_3}.int((X_{r2} \odot X_{r3}))) & \text{if } X_{\text{best\_among\_3}} = 1 \\ (1-\mu)\left(int(\neg X_{best\_among\_3})).int(X_{r2} \otimes X_{r3})\right) & \text{Otherwise} \end{cases}$$

i.e.,

$$\hat{X}_i = \begin{cases} (1-\mu)(int((X_{r2} \odot X_{r3}))) & \text{if } X_{\text{best\_among\_3}} = 1 \\ (1-\mu)\left(int(X_{r2} \otimes X_{r3})\right) & \text{Otherwise} \end{cases}$$

So, during later generations, if $X_{\text{best\_among\_3}}$ consists this feature and any of the other two vectors selected for generating mutation vector consists this feature then chances are more that mutation vector will have this feature. Also, if $X_{\text{best\_among\_3}}$ doesn't consists this feature but other two vectors selected for generating mutation vector consists this feature then there is chance that mutation vector will have this feature.

Hence, it can be said that, the mutation operator purposed in Equation (10) and Equation (11) is designed in such a way that, it tries to make balance between global exploration and local exploitation.

6 Results Obtained/Goals achieved

Figure 1 anf 2 shows average error and average number of features used in each generation till 50th generation of the algorithm. In the figure 1 it can be seen that, proposed mutation strategy is better minimizing the error in comparison with mutation strategy in existing MOFS-BDE algorithm. Similarly, in Figure 2 it can be seen that, generation by generation, proposed mutation strategy is better performing.

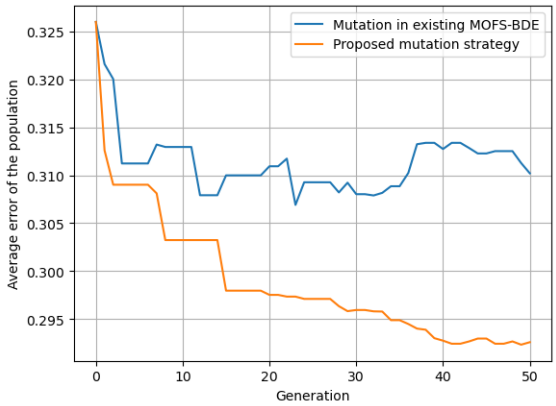

Figure 1: Average error for each generation.

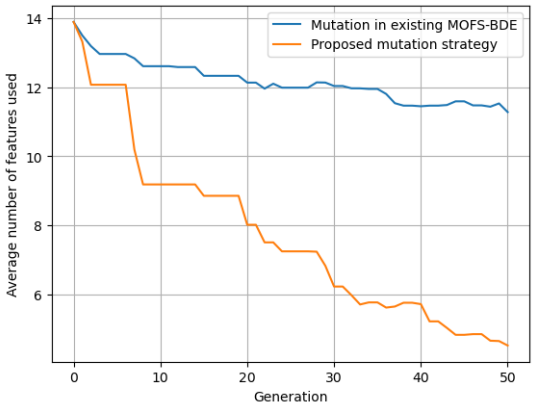

Figure 2: Average number of features for each generation

Figure 3: Average error and average number of features for each generation.

7 Summary

The results mentioned in section 6 shows that, modified MOFS-BDE algorithm with our proposed mutation strategy performs better than the existing MOFS-BDE algorithm. The modified algorithm is not only minizing the error but it is also fastly reducing the feature alnogwith geneation and hence this algorithm will be efficient in datasets with more number of features.

8 Limitations

The limitation of any evolutionary based algorithm including MOFS-BDE algorithm is that, they take more time to converge and are computationally slow.

9 Objectives on Focus in the Next Progress Review Seminar

During the next progress seminar, the comparison of old and proposed mutation strategy will be done using **Hypervolume**. Hypervolume is a metric used in multi-objective optimization problems to assess the quality of a set of solutions.

10 Tentative Timeline of Ph.D.

References

[1] C. Bin, H. Jiarong, and W. Yadong, "The minimum feature subset selection problem," *Journal of Computer Science and Technology*, vol. 12, no. 2, pp. 145–153, 1997. [Online]. Available: https://jcst.ict.ac.cn/en/article/id/437

[2] A. K. Das, S. Das, and A. Ghosh, "Ensemble Feature Selection using Bi-objective Genetic Algorithm," *Knowledge-Based Systems*, vol. 123, pp. 116–127, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705117300801

[3] D. Yilmaz Eroglu and K. Kilic, "A Novel Hybrid Genetic Local Search Algorithm for Feature Selection and Weighting with an Application in Strategic Decision making in Innovation Management," *Information Sciences*, vol. 405, pp. 18–32, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020025517306497

[4] S. Tabakhi and P. Moradi, "Relevance Redundancy Feature Selection based on Ant Colony Optimization," *Pattern Recognition*, vol. 48, no. 9, pp. 2798–2811, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0031320315001211

[5] K. Chen, F.-Y. Zhou, and X.-F. Yuan, "Hybrid Particle Swarm Optimization with Spiral-shaped Mechanism for Feature Selection," *Expert Systems with Applications*, vol. 128, pp. 140–156, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417419302015

[6] Y. Zhang, X. fang Song, and D. wei Gong, "A Return-cost-based Binary Firefly Algorithm for Feature Selection," *Information Sciences*, vol. 418-419, pp. 561–574, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020025516314098

[7] N. García-Pedrajas, A. de Haro-García, and J. Pérez-Rodríguez, "A Scalable Memetic Algorithm for Simultaneous Instance and Feature Selection," *Evolutionary Computation*, vol. 22, no. 1, pp. 1–45, 03 2014. [Online]. Available: https://doi.org/10.1162/EVCO_a_00102

[8] Y. Zhang, S. Cheng, Y. Shi, D. wei Gong, and X. Zhao, "Cost-sensitive Feature Selection using Two-archive Multi-Objective Artificial Bee Colony Algorithm," *Expert Systems with Applications*, vol. 137, pp. 46–58, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417419304440

[9] M. Mafarja, I. Aljarah, A. A. Heidari, A. I. Hammouri, H. Faris, A.-Z. Ala'M, and S. Mirjalili, "Evolutionary Population Dynamics and Grasshopper Optimization Approaches for Feature Selection Problems," *Knowledge-Based Systems*, vol. 145, pp. 25–45, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705117306159

[10] M. Taradeh, M. Mafarja, A. A. Heidari, H. Faris, I. Aljarah, S. Mirjalili, and H. Fujita, "An Evolutionary Gravitational Search-based Feature Selection," *Information Sciences*, vol. 497, pp. 219–239, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020025519304414

[11] N. Kozodoi, S. Lessmann, K. Papakonstantinou, Y. Gatsoulis, and B. Baesens, "A Multi-objective Approach for Profit-driven Feature Selection in Credit Scoring," *Decision Support Systems*, vol. 120, pp. 106–117, 05 2019.

[12] Z. Zhu, Y.-S. Ong, and M. Dash, "Wrapper–Filter Feature Selection Algorithm Using a Memetic Framework," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 37, no. 1, pp. 70–76, 2007.

[13] M. Amoozegar and B. Minaei, "Optimizing Multi-objective PSO based Feature Selection Method using a Feature Elitism Mechanism," *Expert Systems with Applications*, vol. 113, 07 2018.

[14] A. Peimankar, S. J. Weddell, T. Jalal, and A. C. Lapthorn, "Evolutionary Multi-objective Fault Diagnosis of Power Transformers," *Swarm and Evolutionary Computation*, vol. 36, pp. 62–75, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2210650216301699

[15] E. Hancer, B. Xue, M. Zhang, D. Karaboga, and B. Akay, "Pareto Front Feature Selection based on Artificial Bee Colony Optimization," *Information Sciences*, vol. 422, pp. 462–479, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020025516312609

[16] R. N. Khushaba, A. Al-Ani, and A. Al-Jumaily, "Feature Subset Selection using Differential Evolution and a Statistical Repair Mechanism," *Expert Systems with Applications*, vol. 38, no. 9, pp. 11 515–11 526, 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417411004362

[17] U. Sikdar, A. Ekbal, S. Saha, O. Uryupina, and M. Poesio, "Differential Evolution-based Feature Selection Technique for Anaphora Resolution," *Soft Computing*, vol. 19, 08 2014.

[18] B. Xue, W. Fu, and M. Zhang, "Differential Evolution (DE) for Multi-objective Feature Selection in Classification," *GECCO 2014 - Companion Publication of the 2014 Genetic and Evolutionary Computation Conference*, 07 2014.

[19] U. K. Sikdar, A. Ekbal, and S. Saha, "MODE: Multiobjective Differential Evolution for Feature Selection and Classifier Ensemble," *Soft Computing*, vol. 19, no. 12, pp. 3529–3549, 2015.

[20] Y. Zhang, D. wei Gong, X. zhi Gao, T. Tian, and X. yan Sun, "Binary Differential Evolution with Self-learning for Multi-objective Feature Selection," *Information Sciences*, vol. 507, pp. 67–85, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020025519307819

[21] Q. Al-Tashi, S. J. Abdulkadir, H. M. Rais, S. Mirjalili, and H. Alhussian, "Approaches to Multi-Objective Feature Selection: A Systematic Literature Review," *IEEE Access*, vol. 8, pp. 125 076–125 096, 2020.

[22] A. Al-Ani, A. Alsukker, and R. N. Khushaba, "Feature Subset Selection using Differential Evolution and a Wheel based Search Strategy," *Swarm and Evolutionary Computation*, vol. 9, pp. 15–26, 2013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S221065021200065X

[23] L. S. Oliveira, M. Morita, and R. Sabourin, *Feature Selection for Ensembles Using the Multi-Objective Optimization Approach*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 49–74. [Online]. Available: https://doi.org/10.1007/3-540-33019-4_3

[24] T. M. Hamdani, J.-M. Won, A. M. Alimi, and F. Karray, "Multi-objective Feature Selection with NSGA II," in *Adaptive and Natural Computing Algorithms*, B. Beliczynski, A. Dzielinski, M. Iwanowski, and B. Ribeiro, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 240–247.

[25] B. Xue, M. Zhang, and W. N. Browne, "Particle Swarm Optimization for Feature Selection in Classification: A Multi-Objective Approach," *IEEE Transactions on Cybernetics*, vol. 43, no. 6, pp. 1656–1671, 2013.

[26] S. Oreski and G. Oreški, "Genetic Algorithm-based Heuristic for Feature Selection in Credit Risk Assessment," *Expert Systems with Applications: An International Journal*, vol. 41, pp. 2052–2064, 03 2014.

[27] P. Z. Lappas and A. N. Yannacopoulos, "A Machine Learning Approach Combining Expert Knowledge with Genetic Algorithms in Feature Selection for Credit Risk Assessment," *Applied Soft Computing*, vol. 107, p. 107391, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1568494621003148

[28] G. Krishna and R. Vadlamani, "Feature Subset Selection using Adaptive Differential Evolution: An Application to Banking," 01 2019, pp. 157–163.

[29] H. Hofmann, "Statlog (German Credit Data)," UCI Machine Learning Repository, 1994, DOI: 10.24432/C5NC77.

[30] E. Zorarpacı and S. A. Özel, "A Hybrid Approach of Differential Evolution and Artificial Bee Colony for Feature Selection," *Expert Systems with Applications*, vol. 62, pp. 91–103, 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417416302858

[31] X. He and L. Han, "A Novel Binary Differential Evolution Algorithm Based on Artificial Immune System," in *2007 IEEE Congress on Evolutionary Computation*, 2007, pp. 2267–2272.

[32] H. Peng, Z. Wu, P. Shao, and C. Deng, "Dichotomous Binary Differential Evolution for Knapsack Problems," *Mathematical Problems in Engineering*, vol. 2016, pp. 1–12, 01 2016.

[33] T. Wang, K. Wu, T. Du, and X. Cheng, "Adaptive Dynamic Disturbance Strategy for Differential Evolution Algorithm," *Applied Sciences*, vol. 10, no. 6, 2020. [Online]. Available: https://www.mdpi.com/2076-3417/10/6/1972