**National Institute of Technology, Raipur**



**Department of Information Technology**

**A**

**Project Report**

**On**

**“Feature selection using (AOA) Arithmetic optimization Algorithm”**

**Under Supervision of**

**Dr. Tirath Prasad Sahu Sir**

**Submitted by**

| **Anurag Choudhury** | **17118012** |
| --- | --- |
| **Srijan Agrawal** | **17118082** |
| **Nikhil Chandrakar** | **17118903** |

**ACKNOWLEDGEMENT**



The pleasure, the achievement, the glory, the satisfaction, the reward appreciation and construction of our project cannot be thought of without the few, who apart from their regular schedule spent their valuable time. A number of persons contributed either directly or indirectly in shaping the desired outcome. We owe a debt of gratitude to **Dr. Satya Prakash Sahu, Head of Department, National Institute of Technology, Raipur, Chhattisgarh** for providing this opportunity to develop this project.

We express our sincere thanks to our guide **Dr.Tirath Prasad Sahu, Assistant Professor, National Institute of Technology, Raipur, Chhattisgarh** for his valuable suggestions and guidance which helped in successful completion of our project. Without his valuable time it would have been impossible for us to complete this project.

At last but not the least we would like to thank our parents for always encouraging us in our studies and also our friends who directly or indirectly helped us in our work.

**Abstract**

Feature Selection aims to locate the main data from a given arrangement of features. As this task can be viewed as an optimization problem, the combinatorial development of the potential arrangements might be in-practical for a comprehensive search. Feature selection can productively improve the precision of classification and diminish the estimation, storage and calculation demands, and hence it has been applied in research progressively.

Metaheuristic algorithms are among the most impressive algorithms for feature selection. There are numerous points of interest that make these algorithms more helpful to take care of a wide scope of issues today. One of the significant favorable circumstances of metaheuristic algorithms is giving fast and effective solutions for huge and complicated problems.

Metaheuristic algorithms are fundamentally used to explore search space to discover ideal solutions. In this work, an Arithmetic optimization algorithm is proposed for Feature selection which improves the classification accuracy and guarantees ideal size of feature subset.

**Keywords:** AOA(Arithmetic optimization algorithm), Feature selection, metaheuristic algorithm

**Table of Contents**

| **Sno.** | **Topic** |
| --- | --- |
| 1 | Introduction  1.1 Objective  1.2 Motivation  1.3 Structure of Project |
| 2 | Table of Literature Survey |
| 3 | Feature selection  3.1 What is Feature selection?  3.2 Why feature selection?  3.3 Applications of Feature selection |
| 4 | Metaheuristic Algorithm  4.1 What is a Metaheuristic Algorithm?  4.2 Nature inspired Metaheuristic Algorithm  4.3 Why Nature inspired Metaheuristic Algorithm? |
| 5 | AOA Algorithm  5.1 AOA Algorithm  5.2 Foundation of AOA Algorithm  5.3 Working of AOA Algorithm |
| 6 | Methodology |
| 7 | Result and Discussion |
| 8 | Conclusion |
| 9 | References |

**Introduction**

**1.1 Objective**

The main objective of this project is Feature selection using a new meta-heuristic method called Arithmetic Optimization Algorithm (AOA) and applying it to some benchmark datasets, so that the number of irrelevant features are diminished and the performance of classification is improved.

**1.2 Motivation**

The primary motivation driving this project is that the majority of the features in a dataset don't add to better outcomes. Consequently, these unessential and immaterial features increment the complexity of the system as well as the processing time. Hence, feature selection is significant for better performance in classification.

**1.3 Structure of Project**

The structure of the project is divided into two parts, which involves feature selection using AOA algorithm for 5 benchmark datasets, and at last a comparative analysis is performed with other algorithms.

**Table of Literature Survey**

| Sno. | Research Paper | Authors | Year | Application |
| --- | --- | --- | --- | --- |
| 1 | Parameter adaptive harmony search algorithm for unimodal and multimodal optimization problems [1] | Vijay Kumar, [Jitender Kumar Chhabra, DineshKumar](https://www.sciencedirect.com/science/article/abs/pii/S1877750313001403#!) | 2014 | The proposed technique uses the power of global and local searching from HMRC and PAR parameters. |
| 2 | Material and shape optimization of bi-directional functionally graded plates by GIGA and an improved multi-objective particle swarm optimization algorithm [2] | Chao Wang, Jin Ming Koh, Tiantang Yu, Neng Gang Xie, Kang Hao Cheong | 2020 | In this study, a multi-objective optimization framework is developed for the design of bi-directional functionally graded plates (2D-FGPs) with variable thickness. |
| 3 | Queuing search algorithm: A novel metaheuristic algorithm for solving engineering optimization problems [3]. | Jinhao Zhang Li Xiao, Jiang Gao, Quanke Pan | 2018 | A novel metaheuristic algorithm called queuing search (QS) is proposed, which is inspired from human activities in the queuing process. |
| 4 | An adaptive multiscale approach for identifying multiple flaws based on XFEM and a discrete artificial fish swarm algorithm[4]. | Wenhu Zhao, Chengbin Du, Shouyan Jiang | 2018 | This paper presents an adaptive multiscale approach for identifying multiple flaws or damage regions without any prior knowledge of their quantity. |
| 5   |  | | --- | | Drone Squadron Optimization: a novel self-adaptive algorithm for global numerical optimization[5]. | Wolfgang Banzhaf, Vinícius Veloso de Melo | 2017 | This paper proposes Drone Squadron Optimization (DSO), a new self-adaptive metaheuristic for global numerical optimization which is updated online by a hyper-heuristic. |
| 6 | A Comprehensive Survey of the Harmony Search Algorithm in Clustering Applications[6]. | Laith Abualigah, Ali Diabat, Zong Woo Geem | 2020 | The Harmony Search Algorithm (HSA) is a swarm intelligence optimization algorithm which has been successfully applied to a broad range of clustering applications, including data clustering, text clustering, fuzzy clustering, image processing, and wireless sensor networks. |
| 7 | The Whale Optimization Algorithm[7]. | Seyedali Mirjalili, Andrew Lewis | 2016 | This paper proposes a novel nature-inspired meta-heuristic optimization algorithm, called Whale Optimization Algorithm (WOA), which mimics the social behavior of humpback whales. The algorithm is inspired by the bubble-net hunting strategy. |
| 8 | Equilibrium optimizer: A novel optimization algorithm[8]. | Afshin Faramarzi, Mohammad Heidarinejad, Brent Stephens, Seyedali Mirjalili | 2020 | Developed a novel optimization algorithm inspired by mass balance models. Demonstrated effectiveness and superiority of the proposed method. |
| 9 | Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces[9]. | Rainer Storn & Kenneth Price. | 1997 | A new heuristic approach for minimizing possibly nonlinear and non-differentiable continuous space functions is presented. By means of an extensive testbed it is demonstrated that the new method converges faster and with more certainty than many other acclaimed global optimization methods. |
| 10 | Krill herd: A new bio-inspired optimization algorithm[10]. | Amir Hossein Gandomi, Amir Hossein Alavi | 2012 | In this paper, a novel biologically-inspired algorithm, namely krill herd (KH) is proposed for solving optimization tasks. The KH algorithm is based on the simulation of the herding behavior of krill individuals. |

**Feature selection**

**3.1 What is Feature selection?**

In machine learning and statistics, feature selection, also termed as variable selection, attribute selection or variable subset selection, is the method of selecting a subset of relevant features for use in model development.

**3.2 Why Feature selection?**

In the field of machine learning, feature selection is quite possibly the most basic problem which can diminish the quantity of features to expand the exhibition of characterization. The majority of the features in a dataset don't add to better outcomes. Hence, these insignificant and irrelevant features increment the complexity of the system as well as the handling time. In this way, feature selection is significant for better performance in classification.

**3.3 Applications of Feature selection**

Feature selection can be utilized in clinical finding tasks where the undertaking is to choose a subset of clinical tests. Huge Scale information mining application is another model where the feature selection problem is utilized. Power System control ,text classification, choosing a subset of sensors in self-sufficient robot configuration are broadly utilized applications for feature selection problems.

**Metaheuristic Algorithm**

**4.1 What is a Metaheuristic Algorithm?**

Metaheuristic algorithms are computational insight models particularly utilized for modern optimization problems .To take care of optimization problems effectively, metaheuristic algorithms can be utilized which are sufficiently powerful to manage such issues.

**4.2 Nature inspired Metaheuristic Algorithm**

Nature inspired Metaheuristic algorithms are found through the conduct of organic nature and have its own points of interest and inconveniences. Nonetheless, the conventional optimization algorithm is used to take care of the issue in a smaller dimension. Therefore, many researchers started to focus on nature which provided nature-inspired models to solve many difficult problems, such as optimization problems.

**4.3 Why Nature inspired Metaheuristic Algorithm ?**

The major pre-processing task in information discovery databases is Feature selection. If an original dataset contains n number of features, there are 2n possible searches. Getting the ideal feature subset from all the searches is a comprehensive task. As of now, nature inspired metaheuristic algorithms prove their efficiency in solving optimization problems.

**Feature selection using AOA algorithm**

**5.1 AOA Algorithm:**

Arithmetic Optimization Algorithm (AOA) utilizes the distribution behavior of the main arithmetic operators in mathematics including (Multiplication (M), Division (D), Subtraction (S), and Addition (A)). AOA is mathematically modeled and implemented to perform the optimization processes in a wide range of search spaces.

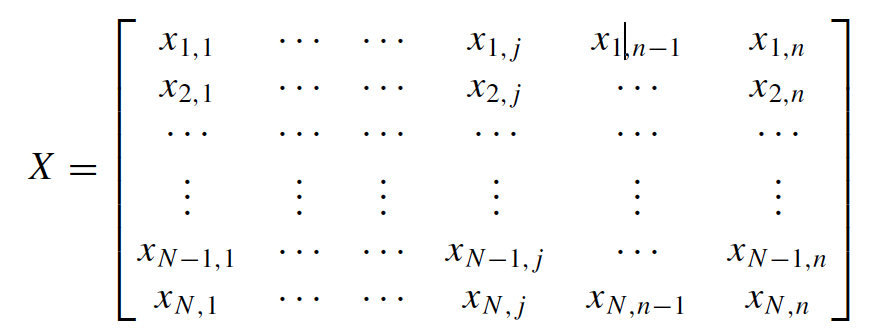
**5.2 Foundation of AOA Algorithm:**

Arithmetic is a fundamental component of number theory, and it is one of the important parts of modern mathematics, along with geometry, algebra, and analysis. Arithmetic operators (i.e., Multiplication, Division, Subtraction, and Addition) are the traditional calculation measures used usually to study the numbers. We use these simple operators as a mathematical optimization to determine the best element subjected to specific criteria from some set of candidate alternatives (solutions). Optimization problems occur in all quantitative disciplines from engineering, economics, and computer sciences to operations research and industry, and the improvement of solution techniques has attracted the interest of mathematics for eras.

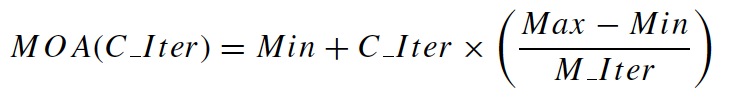
**5.3 Working of AOA Algorithm:**

*Initialization Phase*

In AOA, the optimization process begins with a set of candidate solutions (X) as shown in below Matrix, which is generated randomly, and the best candidate solution in each iteration is considered as the best-obtained solution or nearly the optimum so far.

****

Before the AOA starts working, it should select the search phase (i.e., exploration or exploitation). So, the Math Optimizer Accelerated (MOA) function is a coefficient calculated used in the following search phases.

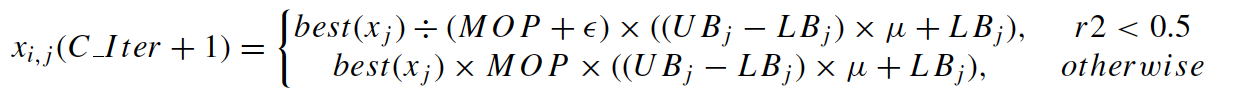
****

where MOA(C Iter) denotes the function value at the tth iteration, which is calculated by the above equation. (C I ter) denotes the current iteration, which is between 1 and the maximum number of iterations (M I ter ). Min and Max denote the minimum and maximum values of the accelerated function, respectively.

*Exploration phase*

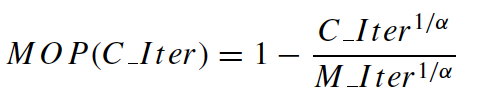
A function is employed based on using four mathematical operations to show the effect of the different operator’s distribution values. Hence, the exploration search detects the near-optimal solution that may be deduced after several endeavours (iterations). In addition, the exploration operators (D and M) were operated at this stage of optimization to support the other stage (exploitation) in the search process through enhanced communication between them.

The exploration operators of AOA explore the search area randomly on several regions and approach to find a better solution based on two main search strategies (Division (D) search strategy and Multiplication search strategy), which are modeled in below eq. This phase of searching (exploration search by executing D or M) is conditioned by the Math Optimizer accelerated (MOA) function for the condition of r1 > MOA (r1 is a random number). Below figure shows how the used operators converge toward the optimal area. The first operator (D), in this phase, is conditioned by r2 < 0.5 and the other operator (M) will be neglected until this operator finishes its current task. Otherwise, the second operator (M) will be engaged to perform the current task instead of the D (r2 is a random number). Note, a stochastic scaling coefficient is considered for the element to produce more diversification courses and explore diverse regions of the search space. We employed the simplest rule, which is able to simulate the behaviors of Arithmetic operators.



where xi(C Iter+1) denotes the ith solution in the next iteration, xi, j(C Iter) denotes the jth position of the ith solution at the current iteration, and best(x j ) is the jth position in the best-obtained solution so far. ϵ is a small

integer number, U Bj and L Bj denote to the upper bound value and lower bound value of the jth position, respectively. μ is a control parameter to adjust the search process, which is fixed equal to 0.5.

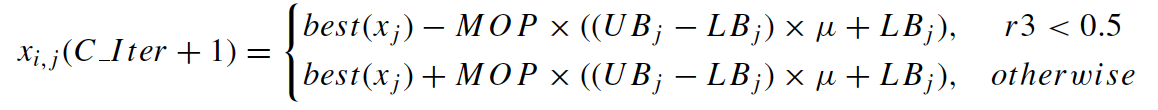


where Math Optimizer probability (MOP) is a coefficient, MOP(C Iter) denotes the function value at the t th iteration, and (C I ter) denotes the current iteration and (M I ter ) denotes the maximum number of iterations. α is a sensitive parameter and defines the exploitation accuracy over the iterations, which is fixed equal to 5.

*Exploitation phase*

The exploitation search detects the near-optimal solution that may be deduced after several endeavours (iterations). In addition, the exploitation operators (S and A) were operated at this stage of the optimization to support the exploitation stage through enhanced communication between them.

This phase of searching (exploitation search by executing S or A) is conditioned by the MOA function value for the condition of r1 is not greater than the current MOA(C Iter) value. In AOA, the exploitation operators (Subtraction (S) and Addition (A)) of AOA explore the search area deeply on several dense regions and approach to find a better solution based on two main search strategies (i.e., Subtraction (S) search strategy and Addition (A) search strategy).



This phase exploits the search space by conducting a deep search, which is very clear. The first operator (S), in this phase, is conditioned by r3 <0.5 and the other operator (A) will be neglected until this operator finishes its current task. Otherwise, the second operator (A) will be engaged to perform the current task instead of the S. These procedures in this phase are similar to the partitions of the previous phase. However, exploitation search operators (S and A) often attempt to avoid getting stuck in the local search area. This procedure assists the exploration search strategies in finding the optimal solution and keeping the diversity of the candidate solutions. We carefully designed μ parameters to produce a stochastic value at each iteration to maintain exploration not only during first iterations but also last iterations. This part of searching is very helpful in the situation of local optima stagnation, particularly in the last iterations.

The below figure explains how a search solution updates its variables (positions) according to D, M, S, and A in a 2- Dimensional search space. It can be seen that the final-obtained position can be in a stochastic position within a range which is determined by the positions of D, M, S, and A in the search scope. In other concepts, D, M, S, and A estimate the position of the near-optimal solution, and other solutions update their positions stochastically around the area of the near-optimal solution.

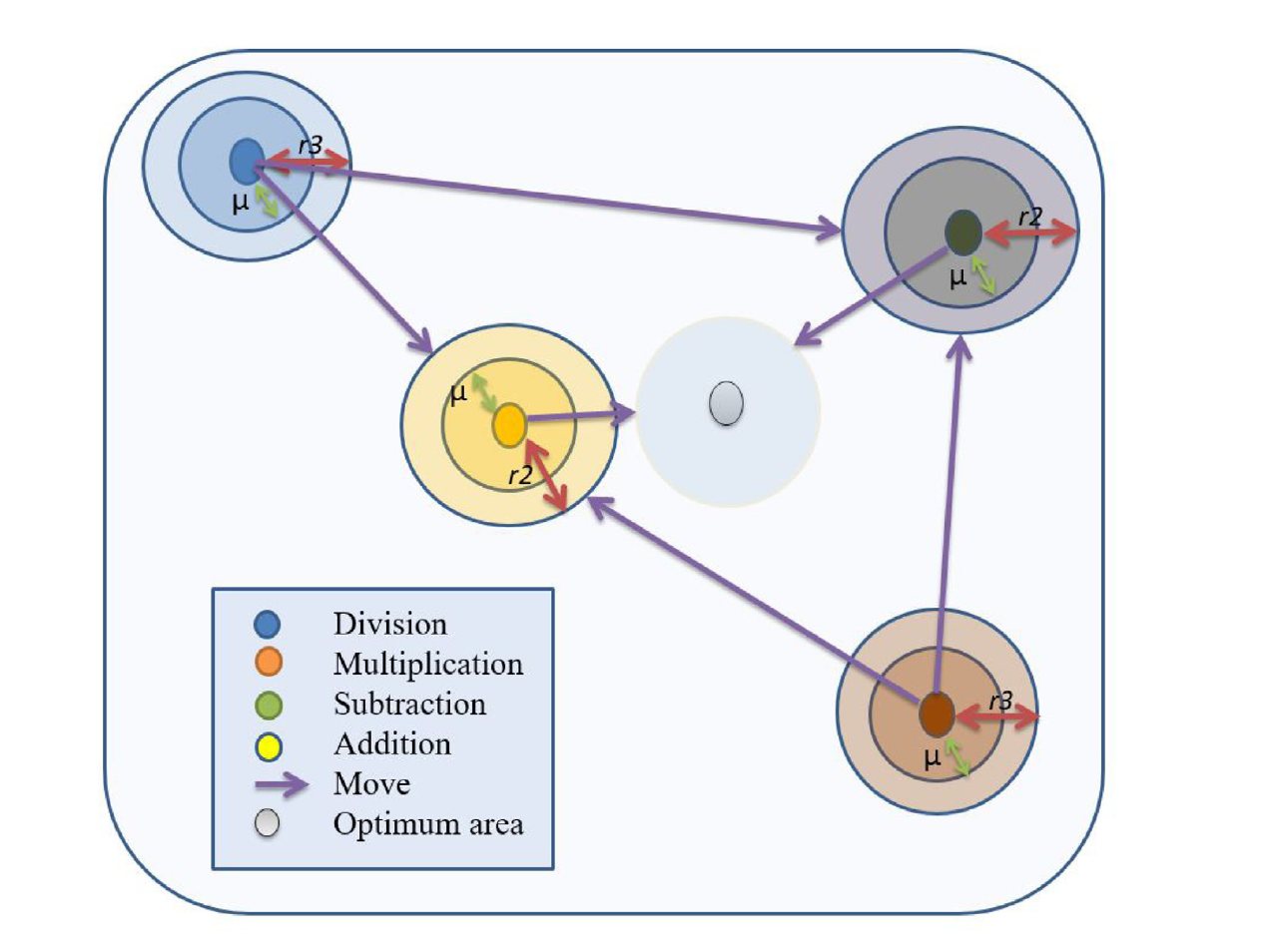
****

Figure 1: Model of updating the position of math operators in AOA toward the optimum area.

**5.3 Pseudo-code of AOA Algorithm:**

1: Initialize the Arithmetic Optimization Algorithm parameters α, μ.

2: Initialize the solutions’ positions randomly. (Solutions: i=1, ..., N.)

3: while (C Iter < M Iter) do

4: Calculate the Fitness Function (FF) for the given solutions

5: Find the best solution (Determined best so far).

6: Update the MOA value using Eq. (2).

7: Update the MOP value using Eq. (4).

8: for (i=1 to Solutions) do

9: for ( j=1 to Positions) do

10: Generate a random values between [0, 1] (r 1, r 2, and r3)

11: if r1 >MOA then

12: Exploration phase

13: if r2 >0.5 then

14: (1) Apply the Division math operator (D “ °¿ ”).

15: Update the ith solutions’ positions using the first rule

16: else

17: (2) Apply the Multiplication math operator (M “ °ø ”).

18: Update the ith solutions’ positions using the second rule

19: end if

20: else

21: Exploitation phase

22: if r3 >0.5 then

23: (1) Apply the Subtraction math operator (S “ − ”).

24: Update the ith solutions’ positions using the first rule

25: else

26: (2) Apply the Addition math operator (A “ + ”).

27: Update the ith solutions’ positions using the second rule

28: end if

29: end if

30: end for

31: end for

32: C Iter=C Iter+1

33: end while

34: Return the best solution (x).

*The computational complexity of AOA*

The computational complexity of the proposed AOA essentially relies on three factors: initialization processes, fitness function evaluation, and updating of solutions. The complexity of the initialization process is of

O(N) where N shows the population size. The complexity of the fitness function is dependent on the problem, so we do not discuss it here. Finally, the complexity of updating solutions is O(M×N) + O(M×N×L) where M indicates iterations and L is the number of parameters in the problem (dimension). Therefore, the computational complexity of the proposed AOA is of O(N × (ML + 1)).

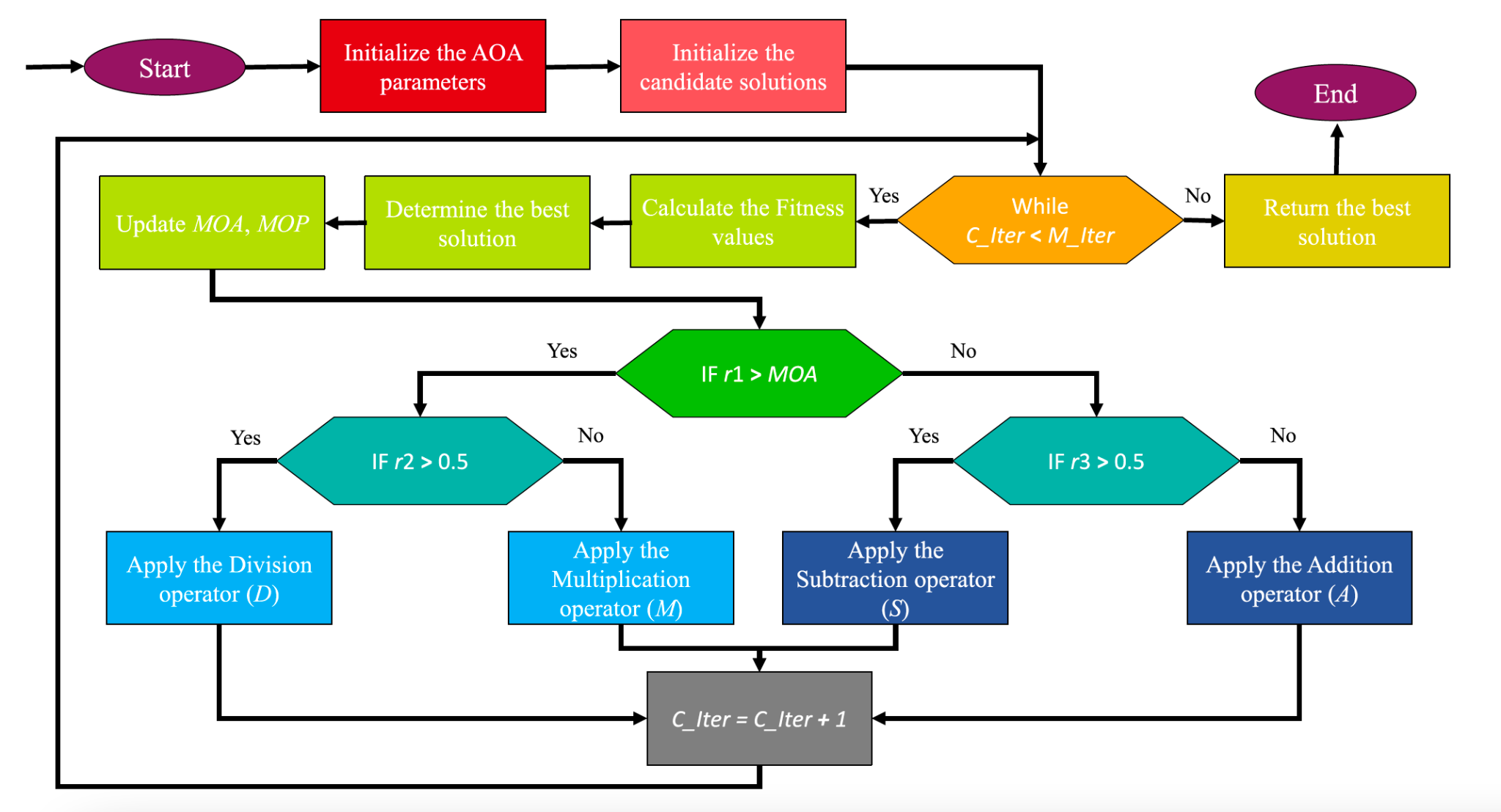
****

Figure 2: Flowchart of the proposed AOA

**Methodology**

This section explains the methodology of feature selection using Arithmetic optimization algorithm:

**1: Parameter initialization**

The first step is to initialize certain parameters such as:

* Population size n = 10
* Maximum number of iterations = 100
* Lower bound (lb) = 0
* Upper bound (ub) = 1
* Threshold = 0.5
* MOP\_MAX = 1
* MOP\_MIN = 0.2
* Alpha = 5
* Mu = 0.499

**2: Trimming of dataset**

After applying AOA(Arithmetic optimization algorithm) on benchmark dataset, the original dataset is trimmed into trimmed dataset , which accommodates the values of only selected features i.e. values corresponding to the selected features are denoted by ‘1’ and values corresponding to other features are denoted by ‘0’.

**3: Performance evaluation**

**KNN classification** algorithm is used to assess classification performance of the trimmed dataset. The K-NN algorithm stores the training data and waits for the testing data and classification is conducted using the most related training data. This method classifies instances based on the relationship among the variables.

**Results and Discussion**

**1: Ionosphere dataset**

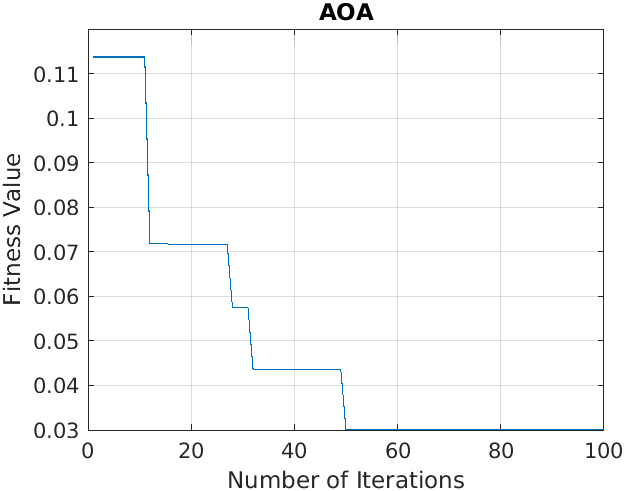
The original ionosphere dataset from UCI machine learning repository is a binary classification dataset with dimensionality 34. There is one attribute having values of all zeros, which is discarded. So the total number of dimensions is 33. The ‘bad’ class is considered an outlier class and the ‘good’ class as inliers."Good" radar returns are those showing evidence of some type of structure in the ionosphere. "Bad" returns are those that do not; their signals pass through the ionosphere.

**Input:**

* Feature vector matrix: (351 X 34) where, number of features is 34 and number of instances is 351
* Label matrix: (351 X 1)
* Parameter settings:
  + Population size = 10
  + Maximum number of iterations = 100
  + K-value in K nearest neighbour = 5

**Output:**

* Accuracy of validation model = 97.14%
* Number of selected features= 8
* Computational time = 5.40 seconds
* Convergence curve :



**2: Arrhythmia dataset**

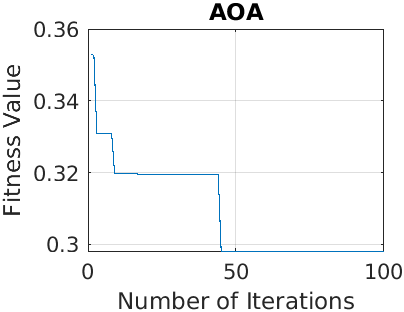
This dataset contains 279 attributes, 206 of which are linear valued and the rest are nominal. Concerning the study of H. Altay Guvenir: "The aim is to distinguish between the presence and absence of cardiac arrhythmia and to classify it in one of the 16 groups. Class 01 refers to 'normal' ECG classes 02 to 15 refers to different classes of arrhythmia and class 16 refers to the rest of unclassified ones. For the time being, there exists a computer program that makes such a classification. However there are differences between the cardiology and the programs classification. Taking the cardiologist as a gold standard we aim to minimise this difference by means of machine learning tools."

**Input:**

* Feature vector matrix: (452 X 279) where, number of features is 279 and number of instances is 452
* Label matrix: (452 X 1)
* Parameter settings:
  + Population size = 10
  + Maximum number of iterations = 100
  + K-value in K nearest neighbour = 5

**Output:**

* Accuracy of validation model = 67.88%
* Number of selected features= 81
* Computational time = 5.9478 seconds
* Convergence curve :



**3: Human activity dataset**

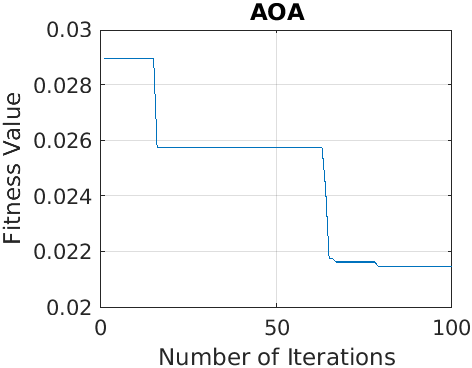
Human activity sensor data contains observations derived from sensor measurements taken from smartphones worn by people while doing different activities (walking, lying, sitting etc). The dataset consists of accelerometer and gyroscope data captured at 50Hz. The raw sensor data contain fixed-width sliding windows of 2.56 sec (128 readings/window). The activities performed by the subject include: 'Walking', 'ClimbingStairs', 'Sitting', 'Standing',and 'Laying'.

**Input:**

* Feature vector matrix: (24075 X 60) where, number of features is 60 and number of instances is 24075
* Label matrix: (24075 X 1)
* Parameter settings:
  + Population size = 10
  + Maximum number of iterations = 100
  + K-value in K nearest neighbour = 5

**Output:**

* Accuracy of validation model = 98.04%
* Number of selected features= 16
* Computational time = 124.9433 seconds
* Convergence curve :

****

**4: Cities dataset :**

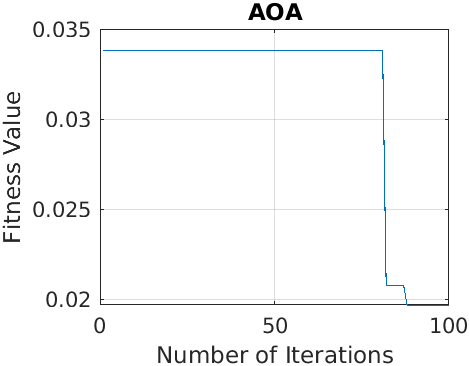
This dataset contains quality of life ratings for U.S. metropolitan cities.This dataset contains 9 attributes which are climate, housing, health, crime transportation, education, arts, recreation, economics and on the basis of these attributes classification is done.

**Input:**

* Feature vector matrix: (329 X 9) where, number of features is 9 and number of instances is 329
* Label matrix: (329 X 1)
* Parameter settings:
  + Population size = 10
  + Maximum number of iterations = 100
  + K-value in K nearest neighbour = 5

**Output:**

* Accuracy of validation model = 98.46%
* Number of selected features= 4
* Computational time = 3.83 seconds
* Convergence curve :



**5: Breast cancer Wisconsin (diagnostic) dataset**

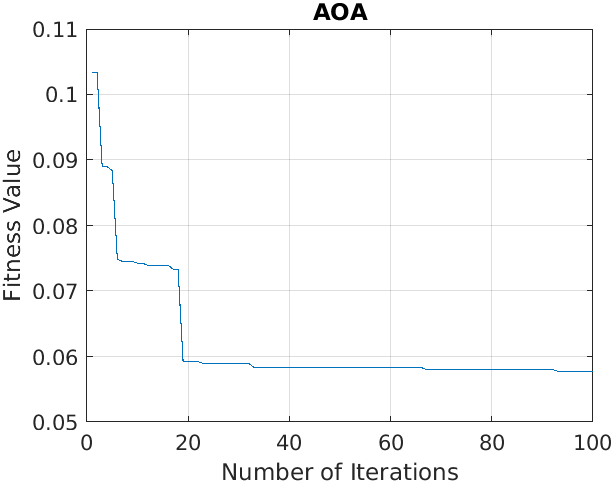
This dataset contains attributes that are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. They describe characteristics of the cell nuclei present in the image.

**Input:**

* Feature vector matrix: (569 X 32) where, number of features is 32 and number of instances 569
* Label matrix: (32 X 1)
* Parameter settings:
  + Population size = 10
  + Maximum number of iterations = 100
  + K-value in K nearest neighbour = 5

**Output:**

* Accuracy of validation model = 97.26%
* Number of selected features= 26
* Computational time = 6.23 seconds
* Convergence curve :



**6: Wine dataset**

These data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines. The attributes are (1) Alcohol (2) Malic acid (3) Ash (4) Alkalinity of ash (5) Magnesium (6) Total phenols (7) Flavonoids (8) Non Flavonoid phenols (9) Proanthocyanidins (10)Color intensity (11)Hue (12)OD280/OD315 of diluted wines (13)Proline

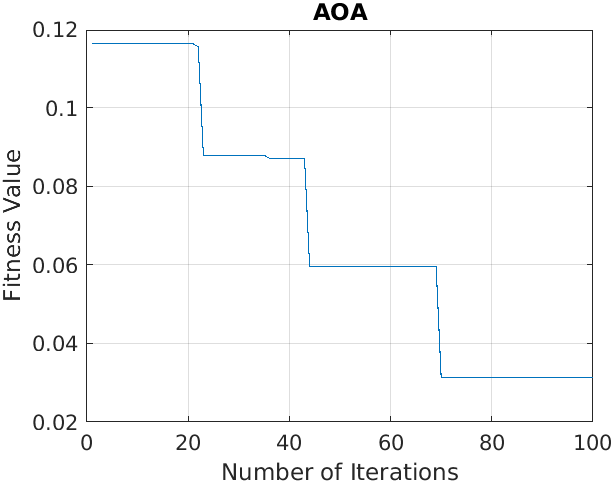
In a classification context, this is a well posed problem with "well behaved" class structures.

**Input:**

* Feature vector matrix: (178 X 13) where, number of features is 13 and number of instances 178
* Label matrix: (178 X 1)
* Parameter settings:
  + Population size = 10
  + Maximum number of iterations = 100
  + K-value in K nearest neighbour = 5

**Output:**

* Accuracy of validation model = 94.28%
* Number of selected features= 5
* Computational time = 4.76 seconds
* Convergence curve :



**7: Congressional Voting record dataset**

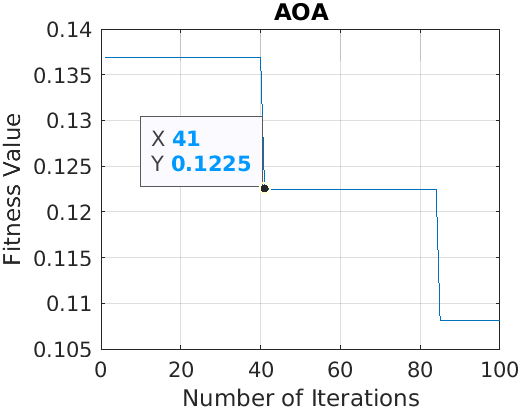
This data set includes votes for each of the U.S. House of Representatives Congressmen on the 16 key votes identified by the CQA. The CQA lists nine different types of votes: voted for, paired for, and announced for (these three simplified to yea), voted against, paired against, and announced against (these three simplified to nay), voted present, voted present to avoid conflict of interest, and did not vote or otherwise make a position known (these three simplified to an unknown disposition).

**Input:**

* Feature vector matrix: (435 X 16) where, number of features is 16 and number of instances 435
* Label matrix: (435 X 1)
* Parameter settings:
  + Population size = 10
  + Maximum number of iterations = 100
  + K-value in K nearest neighbour = 5

**Output:**

* Accuracy of validation model = 89.39%
* Number of selected features= 5
* Computational time = 5.90 seconds
* Convergence curve :



**8: SPECT Heart dataset**

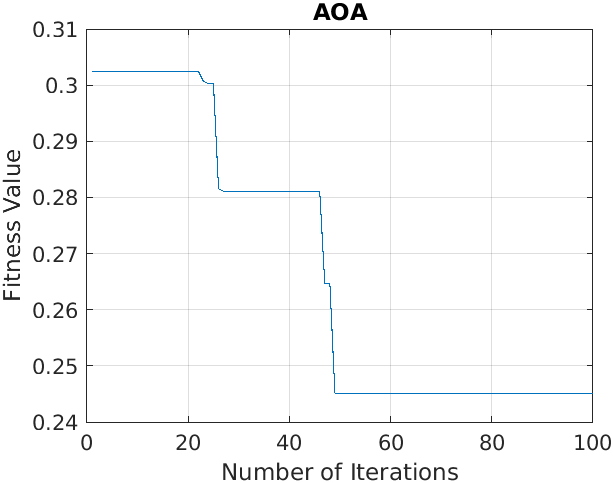
The dataset describes diagnosing of cardiac Single Proton Emission Computed Tomography (SPECT) images. Each of the patients is classified into two categories: normal and abnormal. The database of 267 SPECT image sets (patients) was processed to extract features that summarize the original SPECT images. As a result, 44 continuous feature patterns were created for each patient. The pattern was further processed to obtain 22 binary feature patterns.

**Input:**

* Feature vector matrix: (267 X 22) where, number of features is 22 and number of instances 267
* Label matrix: (267 X 1)
* Parameter settings:
  + Population size = 10
  + Maximum number of iterations = 100
  + K-value in K nearest neighbour = 5

**Output:**

* Accuracy of validation model = 71.69%
* Number of selected features= 10
* Computational time = 5.58 seconds
* Convergence curve :



**9: ILPD (Indian Liver Patient Dataset) Data Set**

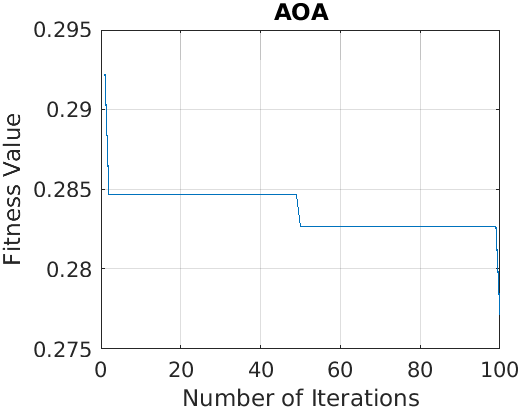
This data set contains 416 liver patient records and 167 non liver patient records.The data set was collected from north east of Andhra Pradesh, India. Selector is a class label used to divide into groups(liver patient or not). This data set contains 441 male patient records and 142 female patient records. Any patient whose age exceeded 89 is listed as being of age "90".

**Input:**

* Feature vector matrix: (583 X 10) where, number of features is 10 and number of instances 583
* Label matrix: (583 X 1)
* Parameter settings:
  + Population size = 10
  + Maximum number of iterations = 100
  + K-value in K nearest neighbour = 5

**Output:**

* Accuracy of validation model = 72.41%
* Number of selected features= 4
* Computational time = 4.84 seconds
* Convergence curve :



**10: BreastEW dataset**

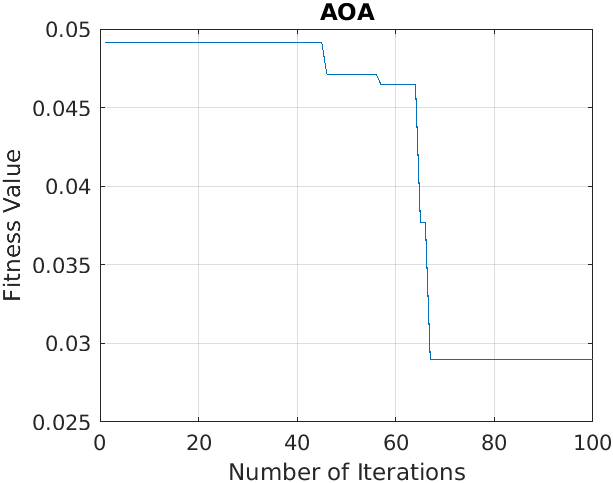
This dataset contains a record of 569 patients . The dataset includes 29 attributes and on the basis of these attributes a patient is classified as a patient with cancer or a patient without cancer.

**Input:**

* Feature vector matrix: (569 X 30) where, number of features is 30 and number of instances 569
* Label matrix: (569 X 1)
* Parameter settings:
  + Population size = 10
  + Maximum number of iterations = 100
  + K-value in K nearest neighbour = 5

**Output:**

* Accuracy of validation model = 96.46%
* Number of selected features= 9
* Computational time = 6.06 seconds
* Convergence curve :



**11: Zoo Animal Classification Dataset**

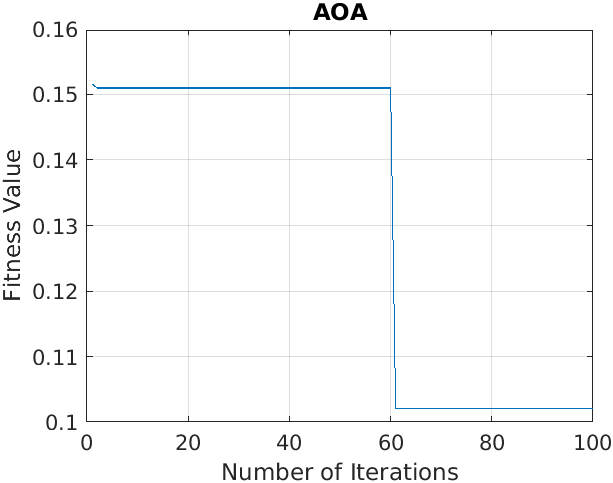
This dataset consists of 101 animals from a zoo. There are 16 variables with various traits to describe the animals. The 7 Class Types are: Mammal, Bird, Reptile, Fish, Amphibian, Bug and Invertebrate. The purpose for this dataset is to be able to predict the classification of the animals, based upon the variables.

**Input:**

* Feature vector matrix: (101 X 16) where, number of features is 16 and number of instances 101
* Label matrix: (101 X 1)
* Parameter settings:
  + Population size = 10
  + Maximum number of iterations = 100
  + K-value in K nearest neighbour = 5

**Output:**

* Accuracy of validation model = 80%
* Number of selected features= 6
* Computational time = 5.63 seconds
* Convergence curve :



**12: Semeion Handwritten Digit Data Set**

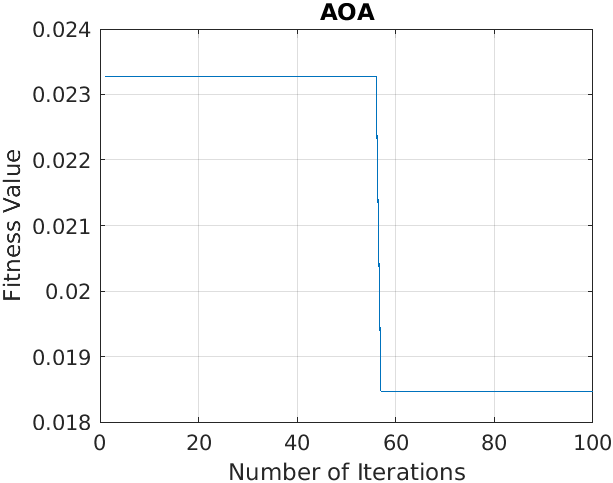
1593 handwritten digits from around 80 persons were scanned, stretched in a rectangular box 16x16 in a gray scale of 256 values.Then each pixel of each image was scaled into a boolean (1/0) value using a fixed threshold. Each person wrote on a paper all the digits from 0 to 9, twice. The commitment was to write the digit the first time in the normal way (trying to write each digit accurately) and the second time in a fast way (with no accuracy). This dataset consists of 1593 records (rows) and 256 attributes (columns). Each record represents a handwritten digit, originally scanned with a resolution of 256 grayscale (28). Each pixel of the original scanned image was first stretched, and after scaled between 0 and 1 (setting to 0 every pixel whose value was under the value 127 of the grey scale (127 included) and setting to 1 each pixel whose original value in the grey scale was over 127). Finally, each binary image was scaled again into a 16x16 square box (the final 256 binary attributes).

**Input:**

* Feature vector matrix: (1593 X 265) where, number of features is 265 and number of instances 1593
* Label matrix: (1593 X 1)
* Parameter settings:
  + Population size = 10
  + Maximum number of iterations = 100
  + K-value in K nearest neighbour = 5

**Output:**

* Accuracy of validation model = 97.16%
* Number of selected features= 124
* Computational time = 11.78 seconds
* Convergence curve :



**13: Glass Identification Dataset**

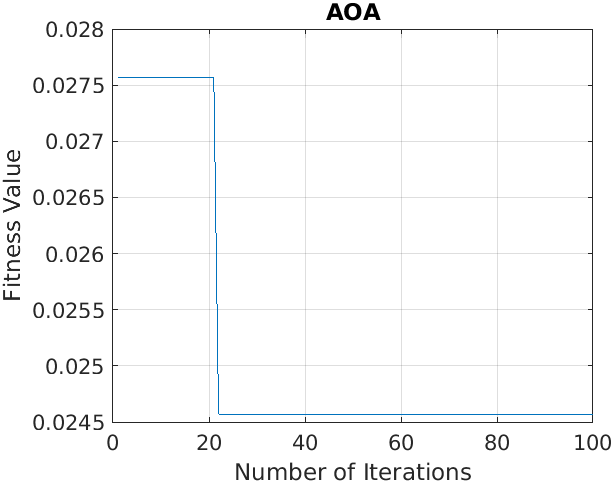
This is a Glass Identification Data Set. It contains 10 attributes including id. The response is glass type(discrete 7 values).

**Input:**

* Feature vector matrix: (214 X 10) where, number of features is 10 and number of instances 214
* Label matrix: (214 X 1)
* Parameter settings:
  + Population size = 10
  + Maximum number of iterations = 100
  + K-value in K nearest neighbour = 5

**Output:**

* Accuracy of validation model = 97.61%
* Number of selected features= 5
* Computational time = 3.75 seconds
* Convergence curve :



**14: ISOLET Dataset**

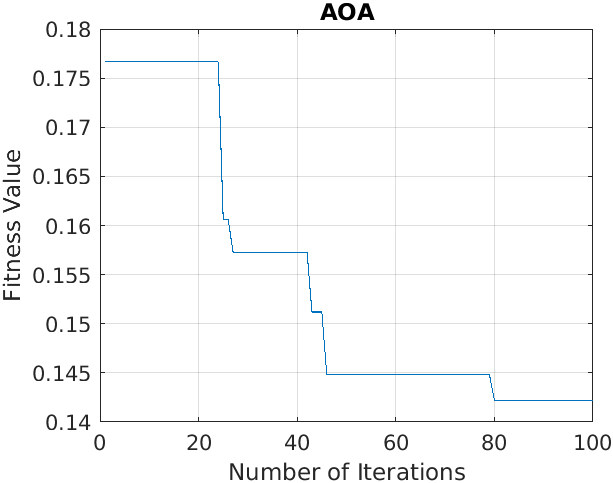
This data set was generated as follows. 150 subjects spoke the name of each letter of the alphabet twice. Hence, we have 52 training examples from each speaker. The speakers are grouped into sets of 30 speakers each, and are referred to as isolet1, isolet2, isolet3, isolet4, and isolet5. The data appears in isolet1+2+3+4.data in sequential order, first the speakers from isolet1, then isolet2, and so on. The test set, isolet5, is a separate file.

**Input:**

* Feature vector matrix: (1559 X 617) where, number of features is 617 and number of instances 1559
* Label matrix: (214 X 1)
* Parameter settings:
  + Population size = 10
  + Maximum number of iterations = 100
  + K-value in K nearest neighbour = 5

**Output:**

* Accuracy of validation model = 81.99%
* Number of selected features= 183
* Computational time = 14.38 seconds
* Convergence curve :



**15: Clean1 dataset**

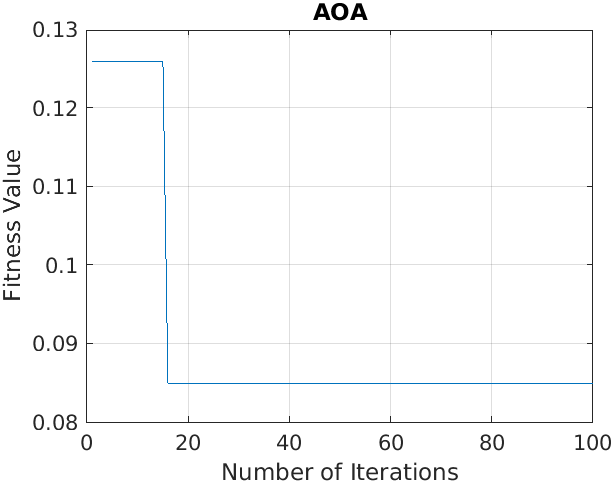
The clean1 dataset is derived from the MUSK dataset. This dataset describes a set of 92 molecules of which 47 are judged by human experts to be musks and the remaining 45 molecules are judged to be non-musks. The goal is to learn to predict whether new molecules will be musks or non-musks. However, the 167 features that describe these molecules depend upon the exact shape, or conformation, of the molecule. Because bonds can rotate, a single molecule can adopt many different shapes. To generate this data set, the low-energy conformations of the molecules were generated and then filtered to remove highly similar conformations. This left 476 conformations. Then, a feature vector was extracted that describes each conformation.

**Input:**

* Feature vector matrix: (476 X 167) where, number of features is 167 and number of instances 476
* Label matrix: (476 X 1)
* Parameter settings:
  + Population size = 10
  + Maximum number of iterations = 100
  + K-value in K nearest neighbour = 5

**Output:**

* Accuracy of validation model = 82.10%
* Number of selected features= 58
* Computational time = 5.26 seconds
* Convergence curve :



**Comparative Analysis**

**Ionosphere dataset:**

| **Algorithm** | **Classification Accuracy(%)** | **Number of selected features** | **Convergence curve** | **Computational time (seconds)** |
| --- | --- | --- | --- | --- |
| (AOA) Arithmetic optimization algorithm | 97.14 | 8 |  | 5.40 |
| (PSO) Particle Swarm optimization | 91.42 | 9 |  | 7.82 |
| (SMA) Slime Mould Algorithm | 88.57 | 3 |  | 2.29 |
| (WOA) Whale optimization Algorithm | 90 | 7 |  | 5.63 |

**Arrhythmia dataset**

| **Algorithm** | **Classification Accuracy(%)** | **Number of selected features** | **Convergence curve** | **Computational time (seconds)** |
| --- | --- | --- | --- | --- |
| (AOA) Arithmetic optimization algorithm | 67.88 | 81 |  | 5.947 |
| (PSO) Particle Swarm optimization | 67.77 | 130 |  | 7.7554 |
| (SMA) Slime Mould Algorithm | 68.88 | 127 |  | 6.64 |
| (WOA) Whale optimization Algorithm | 64.44 | 59 |  | 6.23 |

**Human Activity dataset**

| **Algorithm** | **Classification Accuracy(%)** | **Number of selected features** | **Convergence curve** | **Computational time (seconds)** |
| --- | --- | --- | --- | --- |
| (AOA) Arithmetic optimization algorithm | 98.04 | 16 |  | 124.94 |
| (PSO) Particle Swarm optimization | 98.00 | 25 |  | 441.34 |
| (SMA) Slime Mould Algorithm | 98.027 | 6 |  | 53.45 |
| (WOA) Whale optimization Algorithm | 97.86 | 18 |  | 322.20 |

**Cities dataset**

| **Algorithm** | **Classification Accuracy(%)** | **Number of selected features** | **Convergence curve** | **Computational time (seconds)** |
| --- | --- | --- | --- | --- |
| (AOA) Arithmetic optimization algorithm | 98.46 | 4 |  | 3.83 |
| (PSO) Particle Swarm optimization | 98.46 | 3 |  | 7.05 |
| (SMA) Slime Mould Algorithm | 95.38 | 2 |  | 3.79 |
| (WOA) Whale optimization Algorithm | 98.42 | 6 |  | 5.66 |

**Breast cancer Wisconsin (diagnostic) dataset**

| **Algorithm** | **Classification Accuracy(%)** | **Number of selected features** | **Convergence curve** | **Computational time** |
| --- | --- | --- | --- | --- |
| (AOA) Arithmetic optimization algorithm | 97.26 | 26 |  | 6.23 |
| (PSO) Particle Swarm optimization | 96.27 | 29 |  | 7.59 |
| (SMA) Slime Mould Algorithm | 94.21 | 23 |  | 4.98 |
| (WOA) Whale optimization Algorithm | 93.35 | 27 |  | 5.92 |

**Conclusion**

The primary inspiration behind this project is that the majority of the features in a dataset don't add to better results. Thus, these irrelevant and unimportant features increment the complexity of the framework as well as the handling time. Thus, feature selection is important for better performance in classification.

We applied AOA algorithm on five benchmark datasets and then compared the results with other metaheuristic algorithms like Particle Swarm Optimization (PSO), Slime Mould Algorithm (SMO),Whale Optimization Algorithm(WOA). The results obtained show the performance of AOA (arithmetic optimization algorithm) to prove it’s ability not only for low-dimensional problems but also for high dimensional problems.

**References**

[1] V. Kumar, J.K. Chhabra, D. Kumar, Parameter adaptive harmony search algorithm for unimodal and multimodal optimization problems, J. Comput. Sci. 5 (2) (2014) 144–155.

[2] W. Chao, Y. Jin Ming Koh, G.X. Neng, H.C. Kang, Material and shape optimization of bi-directional functionally graded plates by giga and an improved multi-objective particle swarm optimization algorithm, Comput. Methods Appl. Mech. Engrg. 366 (2020).

[3] J. Zhang, M. Xiao, L. Gao, Q. Pan, Queuing search algorithm: A novel metaheuristic algorithm for solving engineering optimization problems, Appl. Math. Model. 63 (2018) 464–490.

[4] W. Zhao, C. Du, S. Jiang, An adaptive multiscale approach for identifying multiple flaws based on xfem and a discrete artificial fish swarm algorithm, Comput. Methods Appl. Mech. Engrg. 339 (2018) 341–357.

[5] V.V. de Melo, W. Banzhaf, Drone squadron optimization: a novel self-adaptive algorithm for global numerical optimization, Neural Comput. Appl. 30 (10) (2018) 3117–3144.

[6] L. Abualigah, A. Diabat, Z.W. Geem, A comprehensive survey of the harmony search algorithm in clustering applications, Appl. Sci. 10 (11) (2020) 3827.

[7] S. Mirjalili, A. Lewis, The whale optimization algorithm, Adv. Eng. Softw. 95 (2016) 51–67.

[8] A. Faramarzi, M. Heidarinejad, B. Stephens, S. Mirjalili, Equilibrium optimizer: A novel optimization algorithm, Knowl.-Based Syst. 191 (2020) 105190.

[9] R. Storn, K. Price, Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces, J. Glob. Optim. 11 (4) (1997) 341–359.

[10] A.H. Gandomi, A.H. Alavi, Krill herd: a new bio-inspired optimization algorithm, Commun. Nonlinear Sci. Numer. Simul. 17 (12) (2012) 4831–4845.