Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 192 (2021) 4178–4185

Procedia Computer Science

www.elsevier.com/locate/procedia

25th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems

Epigenetic Modification of Genetic Algorithm for Outlier Detection

Kornel Chromiński[a,*], Magdalena A. Tkacz[a]

[a]*Institute of Computer Science, University of Silesia in Katowice, ul. Bedzińska 39, Sosnowiec, Poland*

Abstract

The article presents a new operator in the genetic algorithm. The proposed operator mimics the epigenetic process of prion inheritance. For living organisms, epigenetic processes have a large impact on the differentiation of the population, hence the idea to imitate these processes in genetic algorithms. For the purposes of the experiments, a genetic algorithm to detect outliers was used. The proposed operator mimics the epigenetic process was added to the basic genetic algorithm and the impact of the operator on the effectiveness of the algorithm was assessed. The impact of the proposed modification on the efficiency of the genetic algorithm was tested on six data sets. The article also presents an experimental analysis of the proposed operator.

Keywords: Genetic algorithms; Outlier detection; Epigenetics

1. Introduction

Genetic algorithms are one of the ways of solving optimization problems. Genetic algorithms mimic processes derived from observation of nature (based on inheritance), trying to obtain the optimal result for the problem which we are trying to solve with the genetic algorithm. Apart from the processes mimicking genetic operations, in genetic algorithms we deal with the classification of possible solutions and subjecting them to evaluation based on some specific evaluation function. However, genetic algorithms were developed quite a long time ago and do not take into account new knowledge about the inheritance process and the processes influencing gene expression. One of such processes, discovered relatively recently, are epigenetic processes, referred to as out-of-gene inheritance. The article presents a new operator in the genetic algorithm that mimics the epigenetic process - prion inheritance. To research the effectiveness of the proposed operator, a genetic algorithm to detect outliers was used.

Detecting outliers in data sets is one of the tasks of data analysis. The presence of outliers in the data set may cause the data analysis to be misrepresented. Outliers can also be what we are looking for in data analysis, they are

* Corresponding author. Tel.: +48-32-368-97-63
E-mail address: kornel.chrominski@us.edu.pl

1877-0509
10.1016/j.procs.2021.09.193

an anomaly that appears in the data set, so it is important to detect their presence at an early stage of data analysis. Outlier detection methods are extremely useful in areas such as:

- medicine – disease detection
- finance – embezzlement detection
- industry – failure detection
- computer networks – detection of unusual activity in the network

Detecting outliers is also extremely important for classifiers and machine learning. Most classifier-based methods are sensitive to the presence of outliers. Outliers can also be treated as a kind of classifier. As part of outlier detection, we try to decide on the basis of which criterion to decide whether the data in the data set is correct or the outlier. As a result of the methods for detecting outliers, we obtain a division of the data set into two subsets: a subset of valid data and a subset of outliers

The formal definition of the outlier detection problem can be formulated as follows:
For a given set D consisting of N points $\{x_1, x_2, ..., x_N\}$, where each point is a multidimensional vector m of A attributes the search for the subset $O \subseteq D$ of size K, in such a way as to minimize the entropy of $E(D - O)$ (i.e. the average amount of information):

$$\min_{O \subseteq D} E(D - O) \tag{1}$$

where $|O| = K$

The proposed modification mimics the epigenetic process of prion inheritance. In nature, epigenetics prosesses have a significant impact on the inheritance processes and phenotypic features of individuals. Modification that mimic epigenetic processes was introduced as new operator to the genetic algorithm. Its introduction to the genetic algorithm is aimed at reducing the number of iterations needed to obtain the optimal result. Reducing the number of iterations also affects the total runtime of the algorithm.

2. Genetic Algorithm for Outlier Detection

The experiments used a genetic algorithm to detect outliers presented in [1]. The algorithm is based on Classic Genetic Algorithm, and contains all the basic operators found in the genetic algorithm (crossover operations, selection, mutation e.t.c.)[2, 3, 4].

In the case of the algorithm for detecting outliers, the principles of creating the initial population, the method of calculating the value of the fitness function and the basic operators in the algorithm are as follows:

a) **Individual encoding:** the encoding of individuals in the algorithm used is binary, where 0 is a non-outlier and 1 is an outlier. The length of an individual corresponds to the number of elements in the data set.
b) **Adaptation function:** the Akaike [5] information criterion was used to calculate the value of the adaptation function, and the penalty for outliers in the data set. The formula for calculating the fitness function of an individual in a population is as follows:

$$fit_k = AIC(k) + \kappa m_d log(n) \tag{2}$$

where:
fit_k – value of adaptation function of the k-th individual,
$AIC(k)$ – the value of the Akaike information criterion for the k-th individual,
κ – penalty factor for outlier, experimentally determined value $\kappa = 5$
m_d – the number of outliers in the data set,
n – the number of individuals.

c) **Individual selection**: in the algorithm the tournament method of selecting individuals was used [6].
d) **Mutation of individuals**: A point mutation was used in the algorithm [7].

3. Modification of Genetic Algorithms Mimic Epigenetics Process

The article presents modification mimicking epigenetic process - inheritance by prion. The proposed modification is implemented as a new operator, in addition to standard operators.

3.1. Epigenetics

Epigenetics [8, 9, 10] is a science that studies the processes of non-gene inheritance, as well as the influence of external factors on the level of gene expression. Gene expression determines the phenotypic features of an individual, i.e. its adaptation to the environment, and behavior, appearance, etc. For a long time, scientists have wondered, for example, why there is a difference in the appearance and behavior of monozygotic twins, or why the cloned individuals, despite identical gene sequences, had different coloration, displayed different behavior. Some of these phenomena could be explained thanks to the discovery of mechanisms that are the subject of epigenetics research. Concepts related to epigenetics appeared at the time of the discovery that some changes in the genotype of living organisms are not directly related to the DNA structure, its changes and inheritance processes. Then researches began to wonder what could cause these changes. The result of the research was the discovery of numerous molecules that influence the processes taking place in living organisms, which affect how the genetic code will be read. In other words, how will an individual's phenotype change without changing his DNA. It turned out that epigenetic processes play a significant role in population differentiation and adaptation to new conditions. Epigenetic modifications are also the source of part of the disease, and also have an impact on intra-individual characteristics, such as the perception of the world, or personality traits. It can be concluded that the genotype of living organisms is a place for storing relatively static genetic information, and epigenetic processes are specific dynamic controllers (inhibitors or catalysts) responsible for the activation of certain information.

3.2. Modifications mimic Prion inheritance

Modification presented in the paper is a modification that mimics the epigenetic process of prion inheritance. The proposed modification is an additional operator performed on the population in the genetic algorithm. Prions are protein molecules found in every living organism [11]. Under normal conditions, prions are inactive (non-infectious prions) and have no influence on processes in the living organism. However, it is possible for an external or internal impulse to occur for the prion to move from a neutral state to an invasive state (infectious prion). In the case of an external factor, very often there is a transition to an invasive state of the prion in all (or most) members of a given population exposed to the same external factor. Invasive prions can affect the expression level of our genes.

In the proposed modification, the inheritance process with the prion within a population affected by the same external factor was mapped. The base algorithm was an outlier detection algorithm to which an additional operator was introduced - simulating the occurrence of an external factor. In the modified genetic algorithm, apart from the standard operators, such as crossover and mutation, another process has been introduced: individuals from a given population are, with a certain probability, exposed to the influence of some external factor. The external factor is a small fragment of the genotype (simulation of the prion molecule), which in terms of coding corresponds to the coding of individuals in the genetic algorithm. In simple terms, the action of this factor causes a change to occur in the code of the individuals, the change is the same in all individuals and appears at the same place in the DNA sequence. This changes a fragment of the individual's genotype into a prion sequence.

The prion sequence is generated with an equal probability (50% each) from the genotype of the individual with the highest fitness function value or at random. The prion does not attack (does not build into) the individual with the best value of fitness function - this is to prevent the algorithm from losing the best solution (it is in a sense the strongest and should survive environmental changes). The figure 1 shows the prion inheritance process.

The length of the generated prion simulating sequence is random, but it cannot be longer than 10% of the genotype length of the individuals (the maximum length of the prion of up to 10% has been established so that the prion does not change too large fragment of the individual's sequence, which could lead to the disappearance of population diversity). The general notation of an algorithm implementing a modification simulating the inheritance process with a prion is presented as Algorithm 1.

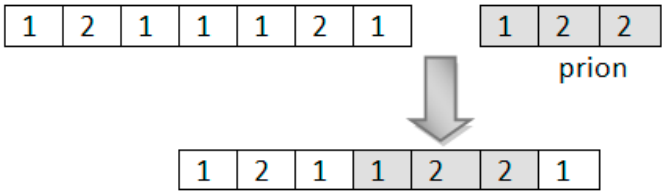

Fig. 1. The operator mimic prion inheritance

Algorithm 1: Algorithm of the operator mimic prion inheritance

Data: population of individuals, the probability of the prion (P_p)
Result: population of individuals treated with prion

```
1  begin
2      Randomize the prion length from the range ⟨0; 10%⟩ the length of the individual;
3      Select the place where the prion is to be installed;
4      Pick a binary number;
5      if number = 0 then
6          Generate a prion from an individual with the best value of fitness function;
7      end
8      else
9          create a prion at random
10     end
11     With P_p select individuals treated by the prion operator;
12     foreach selected individual do
13         prion embeds in an individual's genotype;
14     end
15 end
```

Based on the 1 algorithm, the operation is performed on individuals selected from the population with a certain probability. In the performed operation, a prion genotype is generated, corresponding in terms of coding to the genotype of individuals. The change in the genotype of an individual is accomplished by incorporating the prion sequence into the genotype of the individual at a predetermined location. The prion genotype is generated from the individual with the best fitness function value, or at random. The prion genotype is inserted into the genotype of individuals selected for the proposed operator, replacing the genotype of the individual.
Algorithm 1 shows only the proposed operator. The other operators of the genetic algorithm remain unchanged.

4. Experiments

The experiments were aimed at checking whether supplementing the genetic algorithm for detecting outliers (OGA) with additional operators mimicking epigenetic processes (EpiOGA) would improve the efficiency of this algorithm. The first step was to select the optimal modification probability. In the next step, the obtained values of the fitness function for the optimal probability of modification occurrence were compared with the values of the fitness function for the base algorithm. The algorithms were implemented entirely by the authors using the scripting language R, available from The R project [12]. The `Microbenchmark` [13] package was used to measure the times of the conducted experiments. The R-Studio [14] tool was used as the development environment. All experiments were carried out on a PC with the following parameters:

- **Operating System:** Windows 10 64 bit;
- **Processor:** dual-core Intel Core I7-6500U, CPU clock speed 2.5 GHz;

- **RAM:** 8 GB;
- **R version :** 3.4.1 (64 bit);
- **R-Studio version:** 1.1.463.

4.1. Data Sets used in experiments and parameters of Genetic Algorithm

Six data sets were used in the experiments on the possibility of using operators that mimic epigenetic processes in genetic algorithm for outliers detection. DataSet 1 is an artificial data set created for the experiments, the rest of data sets are publicly available sets used to test methods for detecting outliers. Table 1 presents the data sets used in the experiments with their short characteristics.

Table 1. Data sets used in the experiments

Data Set	Name	Number of elements	Dimensional	The number of outliers
dataSet 1	artificial date set	2000	3	190
dataSet 2	Thyroid Disease [15]	3772	6	93
dataSet 3	Breast Cancer Wisconsin [16]	570	30	212
dataSet 4	Pima Indians Diabetes [17]	768	8	268
dataSet 5	Glass Identification [18]	214	7	9
dataSet 6	Pen-Based Recognition of Handwritten [18]	6870	16	156

The number of outliers in the data sets and which values should be considered outliers were known for all the data sets used in the experiments. The collections varied in size and number of columns taken into account in the outlier detection process.

4.2. Algorithm parameters

For the genetic algorithm for the purposes of the conducted experiments, it was necessary to determine the parameters of the algorithm. The necessary parameters to determine the probabilities of the occurrence of crossover and mutation operators, the selection method and the stop condition. The parameter values for genetic algorithms are presented in Table 2. Parameter values are literature values and were taken from the author's publication [1].

Table 2. Algorithm parameters used in the experiments

Parameter	Value
Number of individuals in the population	10% of the number of items in the dataset
Probability of crossover	100%
Probability of mutation	1%
Selection	tournament
Stop condition	number conditioned by the length of the set generation without improving the value of the adaptation function

The presented parameters were used in experiments for both the base algorithm and the algorithm with added proposed operators mimicking epigenetic processes.

4.3. Empirical selection of the optimal probability of epigenetic modifications

The first step was to select the optimal modification probability. For the six test data sets, the number of generations (depending on the probability of epigenetic modifications) needed to detect all outliers in the data sets was compared. Table 3 shows a summary of the number of generations needed to detect all outliers in the six data sets used in experiments in basic genetic algorithm and genetic algorithm with modifications mimic epigenetic processes with

Table 3. The number of generations needed to obtain the best result for individual test sets depending on the probability of epigenetic operators (The minimum number of iterations for a given test set is marked in bold)

probability	dataSet 1	dataSet 2	dataSet 3	dataSet 4	dataSet 5	dataSet 6
0% (without modification)	2675	2175	661	740	1809	1479
5%	2627	**1883**	592	887	1813	1468
10%	**2543**	1934	577	855	1788	1411
20%	2589	2032	**545**	721	1655	1325
30%	2633	2066	587	**585**	1564	1237
40%	2649	1851	594	753	**1419**	**1195**
50%	2703	2221	633	798	1498	1264
60%	2729	2553	664	911	1789	1406
70%	2781	2354	682	935	1877	1522
80%	2845	2574	701	943	1892	1689
90%	2897	2765	702	971	1906	1755
100%	3015	3100	746	990	1923	1769

different probabilities of epigenetic modification. The presented results are the average value of 20 runs of a given genetic algorithm.

Table 3 shows that if the probability of the occurrence of modifications mimicking epigenetic processes in the genetic algorithm is below 60%, the genetic algorithm performance is improved by reducing the number of generations needed to detect outliers in the data set. The largest decrease in the number of generations can be observed for the data set 4, for the probability of a modification of 30%, and for the set 5 and 6, for the probability of occurrence of modifications equal to 40%. The smallest difference in the number of generations needed to detect getting values can be observed in the case of set 1. In the case of a high probability of epigenetic modification (above 60%), a deterioration of the performance of the genetic algorithm can be observed by increasing the number of generations needed to detect all outliers in relation to the algorithm without modification. The probability of the occurrence of modification is relatively high for operators taking place in the genetic algorithm. Based on empirical research, the authors suggest the probability of the occurrence of a modification mimic the epigenetic process at the level of 20%.

4.4. Comparison of the efficiency of the base algorithm and algorithm with epigenetic modifications

For the probability of the occurrence of modifications imitating epigenetic processes, at which the largest decrease in the number of generations needed to detect all outliers in the test sets was obtained, a comparison was made of the change in the value of the fitness function in the algorithm with modifications in relation to the base algorithm. The values of the fitness function for subsequent generations of the algorithm with epigenetic operator and the algorithm without modification are shown in the graphs in Figure 2.

On the basis of Figure 2 it can be observed that the greatest differences in the dynamics of the change in the value of the fitness function for the genetic algorithm with the operator imitating the epigenetic process in relation to the basic genetic algorithm occur in the sets 1 and 5. On the other hand, the smallest differences in the dynamics of changing the value of the fitness function can be observed for sets 2, 4 and 6.

Table 4 presents the average values of times and the value of the standard deviation of carrying out all operators of the genetic algorithm to detect outliers. The run of the genetic algorithm was repeated 20 times to calculate the mean time. The table also shows the value of the percentage change in the time value of the algorithm with the operator mimicking the epigenetic process in relation to the base genetic algorithm.

Based on table 4 it can be seen that despite the addition of an additional operator to the genetic algorithm, it was possible to reduce the time of the genetic algorithm to detect outliers with the operator imitating the epigenetic process in relation to the basic genetic algorithm. The reduction in uptime was achieved by a significant reduction in the number of generations needed to obtain the optimal result by the genetic algorithm. The mean value of the reduction in operating time was 11.63%, the highest mean reduction was obtained for the set of 5 - it was 18.50%. Table 4 in the last column also shows the value of the level of statistical significance of the obtained time difference, for each set the value of the level of statistical significance indicates that there is a statistically significant difference

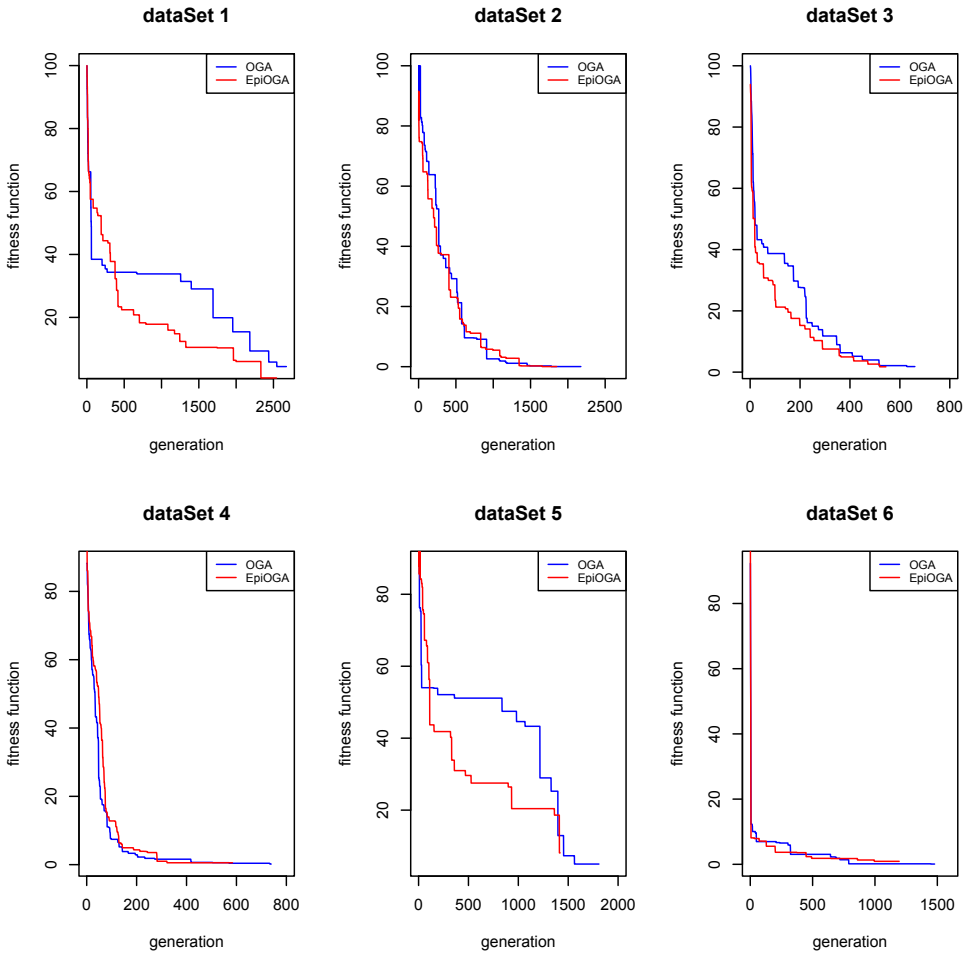

Fig. 2. Change in the value of the fitness function in subsequent generations for the base genetic algorithm for detecting outliers (OGA), and the algorithm with epigenetic modification (EpiOGA) for data sets used in the experiments.

Table 4. Comparison of avarege time in algorithm with modifications and without it

data set	AVG t[min] EpiOGA	sd t EpiOGA	AVG t[min] OGA	sd t OGA	% change	p-value
dataSet 1	26.235	1.780	28.279	6.063	↓ 7.20%	0.002
dataSet 2	11.361	0.785	12.331	1.269	↓ 7.90%	0.001
dataSet 3	2.307	0.173	2.479	0.297	↓ 6.90%	0.001
dataSet 4	1.648	0.185	1.986	0.222	↓ 17.00%	0.001
dataSet 5	27.883	1.608	34.220	2.322	↓ 18.50%	0.001
dataSet 6	7.369	0.677	8.406	0.345	↓ 12.30%	0.001

in the algorithms' operation times. The presented results show that the inclusion of newly discovered mechanisms in genetics and inheritance in genetic algorithms may have a positive impact on the operation of these algorithms.

5. Conclusion and Future Work

The article presents the results of research on the assessment of the effectiveness of the proposed new operator that mimics epigenetic processes in the genetic algorithm. As part of the research, the optimal probabilities of occurrence of the additional genetic algorithm operator proposed in the work were selected. The empirically determined optimal probability of the occurrence of an operator imitating the inheritance process with the use of a prion in the genetic algorithm. Research on the effectiveness of the proposed operator was carried out on the genetic algorithm for detecting outliers. The obtained results allow to conclude that the proposed modification improves the efficiency of the genetic algorithm by reducing the number of generations needed to find the optimal solution (in the algorithm used in the experiments - detection of all outliers). The reduction in the number of generations also influenced into a reduction in the total runtime of the genetic algorithm.

As part of further research, it is planned to test the proposed operator in other genetic algorithms, as well as to check the possibility of using operators based on other epigenetic processes. In the future, the authors plan to research the development of a new epigenetic algorithm based on epigenetic operators.

References

[1] Ö. G. Alma. *Outlier Detection Methods: Genetic Algorithms Based Outlier Detection using Information Criteria*. LAP LAMBERT Academic Publishing, 2010.
[2] W. Banzhaf, P. Nordin, and R. Keller. *Genetic Programming: An Introduction*. MORGAN KAUFMANN PUBL INC, 1997.
[3] D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Professional, 1989.
[4] R. R. Sharapov. Genetic algorithms: Basic ideas, variants and analysis. *Vision Systems: Segmentation and Pattern Recognition*, pages 407–422, 2007.
[5] H. Akaike. Information theory and an extension of the maximum likelihood principle. *Second International Symposium on Information Theory*, 1973.
[6] B. Miller and D. Goldberg. Genetic algorithms, tournament selection, and the effects of noise. *Complex Systems*, 9:193–212, 1995.
[7] S. N. Sivanandam and S. N. Deepa. *Introduction to Genetic Algorithms*. Springer-Verlag GmbH, 2007.
[8] R. Al-Haddad and et al. Epigenetic changes in diabetes. *Neuroscience Letters*, 625:64–69, jun 2016.
[9] C. Dupont, D. Armant, and C. Brenner. Epigenetics: Definition, mechanisms and clinical perspective. *Seminars in Reproductive Medicine*, 27(05):351–357, aug 2009.
[10] D. Moore and S. David. *The Developing Genome: An Introduction to Behavioral Epigenetics*. OXFORD UNIV PR, 2015.
[11] J. Manjrekar. Epigenetic inheritance, prions and evolution. *Journal of Genetics*, 96(3):445–456, jul 2017.
[12] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2008.
[13] O. Mersmann and et al. *Accurate Timing Functions*, October 2018. wersja 1.4-6.
[14] RStudio Team. *RStudio: Integrated Development Environment for R*. RStudio, Inc., Boston, MA, 2015.
[15] J.R. Quinlan, P.J. Compton, K.A. Horn, and L. Lazurus. Inductive knowledge acquisition: A case study. *Proceedings of the Second Australian Conference on Applications of Expert Systems*, 1686.
[16] W.H. Wolberg and O.L. Mangasarian. Multisurface method of pattern separation for medical diagnosis applied to breast cytology. *Proceedings of the National Academy of Sciences*, 87:9193–9196, 1990.
[17] F. T. Liu, K. Ming Ting, and Zhi-Hua Zhou. Isolation forest. *Eighth IEEE International Conference on Data Mining*, 2008.
[18] F. Keller, , E. Muller, and K. Bohm. Hics: High-contrast subspaces for density-based outlier ranking. *ICDE*, 2012.