Scientific Journal of Mechanical and Industrial Engineering

Sci. J. Mech. Ind. Eng., 1(1): 36-39 (2012)

The Measuring Efficiency in Data Envelopment Analysis with Genetic Algorithm

R. Taghaodi[1], M. J. Esfahani[2], S. Molla-Alizadeh-Zavardehi[3], A. Mahmoodi Rad[4,*]

[1] *Department of Mathematics, Kashan Branch, Islamic Azad University, Kashan, Iran*
[2]*Department of Industrial engineering, Islamic Azad University, Naragh Branch, Naragh, Iran*
[3] *Department of Industrial Engineering, Masjed Soleiman Branch, Islamic Azad University, Masjed Soleiman, Iran*
[4] *Department of Mathematics, Masjed Soleiman Branch, Islamic Azad University, Masjed Soleiman, Iran*

*Corresponding author's E-mail: alimahmoodirad@yahoo.com

Abstract

An important objective of Data Envelopment Analysis (DEA) is to evaluate the efficiency of decision making units (DMUs). In DEA models, To evaluate the relative efficiency of DMUs, for a large dataset with many inputs/outputs would need to have a long time with a huge computer. This paper considers Genetic Algorithm (GA) for measuring efficiency of DMUs in DEA. GA requirements for computer memory and CPU time are far less than that needed by conventional DEA methods and can therefore be a useful tool in measuring the efficiency of large datasets. Since the operators have important roles on the fitness of the algorithms, all the operators and parameters are calibrated by means of the Taguchi experimental design in order to improve their performances.

Original Article

Received 09 Nov. 2012
Accepted 17 Nov. 2012

Keywords:
Data Envelopment analysis,
Genetic Algorithm,
Taguchi experimental

INTRODUCTION

Data envelopment analysis (DEA), which initially proposed by Charnes et al. [3], is a non-parametric method for evaluating the relative efficiency of decision making units (DMUs) on the basis of multiple inputs and outputs. Since DEA was proposed in 1978, it has been got comprehensive attention both in theory and application. Now DEA becomes the important analysis tool and research way in management science, operational research, system engineering, decision analysis and so on.

Over the last decade DEA has gained considerable attention as a managerial tool for measuring performance. It has been used widely for assessing efficiency, in the public and private sectors, of organizations such as banks, airlines, hospitals, universities and manufacturers [7].

DEA for a large dataset with many input/output variables and/or DMUs would require huge computer resources in terms of memory and CPU time and take a long time even though with a very fast computer [7]. Furthermore, in order to obtain the results, it must be solved as a separated mathematical programming problem for each DMU. The related works of this area are as follows:

Udhayakumar et al. [15] developed a GA that employs one point crossover and perturbation mutation operators for solving the P-model of chance constrained technique. Azadeh et al. [1] presented a hybrid GA-DEA for assessment and optimization of critical inputs from two different viewpoints of efficiency and cost in electricity transmission units. They used a specific measure and cost allocation super-efficiency DEA models for sensitivity analysis and to determine the critical inputs based on efficiency and cost. In this paper, the GA approach is employed to estimate the efficiency of DMUs in large datasets.

This paper is organized as follows: Section 2 briefly describes the DEA technique. The proposed GA for estimating the efficiency is explained in sections 3. The experimental design and comparisons are presented in section 4. Finally, in section 5, conclusion is provided.

Model DEA

Assume we have a set of observed DMUs $\{DMU_j : j=1,2,...,n\}$ with input and output vectors (x_j, y_j), $x_j=(x_{1j},...,x_{mj})^T>0$ and $y_j=(y_{1j},...,y_{sj})^T>0$ for *j=1, 2,…,n*. In the model proposed by Thompson et al. [14],

SJMIE To cite this paper: Taghaodi R., Esfahani M.J., Molla-Alizadeh-Zavardehi S., Mahmoodi Rad A.. 2012. The Measuring Efficiency in Data Envelopment Analysis with Genetic Algorithm. *Sci. J. Mech. Ind. Eng.*, 1 (1): 36-39.
Journal homepages: http://sjmie.science-line.com/

(often referred to as the TDT model) the efficiency of DMU_o is obtained by the following model.

$$Max_{(u,v)} \frac{\dfrac{u^T y_o}{v^T x_o}}{\max_{1\le j\le n}\{\dfrac{u^T y_j}{v^T x_j}\}} \qquad (1)$$

$$s.t \qquad u,v \ge 0.$$

Where $u \in R^{m\times 1}$ and $v \in R^{s\times 1}$ are the column vectors of input and output weights, respectively.

The model (1) is a non-linear programming model that for the convert it to the linear programming model, we suppose $\max_{1\le j\le n}\{\frac{u^T y_j}{v^T x_j}\} = \frac{1}{t}$, and use the Charnes and Cooper's linear transformation technique [4], so, we obtain

$$\begin{aligned} Max \quad & \theta = u^T y_o \\ s.t \quad & v^T x_o = 1, \\ & u^T y_j - v^T x_j \le 0 \quad j = 1,2,\ldots,n, \qquad (2) \\ & u \ge 0, \ v \ge 0. \end{aligned}$$

Where is the CCR model to obtain the relative efficiency score of DMU_o. If θ^* is the optimum value of θ, then DMU_o is said to be efficient if $\theta^* = 1$ and DMU_o is inefficient, if $\theta^* < 1$, where θ^* is the optimal objective function value of model (2).

In the evaluation of large organization (about millions) by using DEA, even if we employ a high-speed computer, many calculations are needed. Also, it may take a long time to estimate the efficiency of DMUs in these kinds of applications, and because of estimating the efficiency; a linear program must be solved for each DMU. In other words, the conventional DEA models aren't able to be solved via this number of DMUs [7, 11].

To get rid of this problem (relative efficiency of each number of DMUs), we proposed GA which be detailed in the following section.

The Proposed GA

GA created by Holland in the early 1970s, as a stochastic global search method based on principles of evolution theory. Its original idea comes from Darwinian's evolution theory. It is based on the mechanics of natural selection (dependent on the evolution principle ``Survival of the fittest'') and natural genetics. GA has been developed quickly as a simple and effective optimization technique.

In GA there are some chromosomes (The solution to a problem is called a chromosome) which play the role of a set of values for independent variables as a solution for the problem. In each iteration (called generation), there are three basic genetic operations, such as selection, mutation and crossover, then applied one after another to obtain a new generation of chromosomes in which the expected quality over all the chromosomes is better than that of the previous generation. This process is repeated until the termination criterion is met, and the best chromosome of the last generation is reported as the final solution.

Representation

One of the most important decisions in designing a metaheuristic is how to represent solutions in an efficient way to the searching space. Solution representation should be easy to decode to reduce the cost of the algorithm. The problem is a DEA, we consider a population size as the number of chromosomes and weights of inputs and outputs as the genes of each chromosome, respectively. The length of chromosome is equal to the numbers of inputs plus outputs (m + s), and then initial generation are produced with these weights in range (0, positive number), randomly.

Selection mechanism

Here in DEA problem, since our objective is maximization of relative efficiency in model (1), better solutions are those results in an upper objective function, so the fitness value considered as objective function. Using the Roulette wheel selection mechanism, a solution with higher fitness value (i.e. better efficiency) has the more chance to be selected for the new generation.

Genetic operators

1. Reproduction: With more probability, better parents can generate better offspring, so it would be necessary to transfer the best solutions of each generation to the next one. Therefore, the pr% of chromosomes with the better fitness values are copied to the next generation. This is called reproduction.

2. Crossover: The main purpose of crossover is to search the parameter space and hence is considered as the most important operator in GA. The crossover operator takes two chromosomes (parents) from the old population and exchanges the next generation of their structures to produced new offspring. There are wide varieties of proposed crossover operations. The commonly used crossover operators are one-point crossover, Two-point crossover, Uniform crossover and Arithmetic crossover.

3. Mutation: The mutation operator can be considered as a simple form of a local search. The main purpose of applying mutation is to avoid convergence to a local optimum and diversify the population. The used mutation operators in the literature are Swap Mutation, Big Swap Mutation, Inversion Mutation, Displacement Mutation and Perturbation mutation.

MATERIALS AND METHODS

Experimental Design

Test problems: In this subsection Instances generation are conducted to set the parameters and evaluate the performances of GA. First, we generated random problem instances for n = 50, 100, 150, 200, 400, 600, 800 and 1000 DMUs, respectively [11]. After

SJMIE To cite this paper: Taghaodi R., Esfahani M.J., Molla-Alizadeh-Zavardehi S., Mahmoodi Rad A.. 2012. The Measuring Efficiency in Data Envelopment Analysis with Genetic Algorithm. *Sci. J. Mech. Ind. Eng.*, 1 (1): 36-39.
Journal homepages: http://sjmie.science-line.com/

specifying the number of DMUs in a given instance, for each DMU, four problem types A, B, C, and D of inputs and outputs numbers (m, s) were generated from discrete uniform distribution [10, 50]. The problem details are shown in Table 1.

Table 6. Test problems characteristics.

Problem size	Problem type (*m*, *s*)				
	DMUs	A	B	C	D
1	50	(4, 4)	(4, 8)	(8, 4)	(8, 8)
2	100	(5, 5)	(5, 10)	(10, 5)	(10, 10)
3	150	(5, 5)	(5, 10)	(10, 5)	(10, 10)
4	200	(10, 10)	(10, 20)	(20, 10)	(20, 20)
5	400	(10, 10)	(10, 20)	(20, 10)	(20, 20)
6	600	(15, 15)	(15, 30)	(30, 15)	(30, 30)
7	800	(15, 15)	(15, 30)	(30, 15)	(30, 30)
8	1000	(20, 20)	(20, 40)	(40, 20)	(40, 40)

DMUs: Decision making units

Parameter setting: The performance of the GA is generally sensitive to the parameter tuning which affects the search ability and the convergence quality. Choosing proper parameters is time-consuming and sometimes depends on particular instances.

In the related works, to be economic, several experimental designs have been proposed to decrease the number of experiments. Among several experimental design techniques, the Taguchi experimental design method has been successfully employed for a systematic approach for optimization.

Taguchi has created a transformation of the repetition data to another value which is the measure of variation. The transformation is the signal-to-noise (S/N) ratio which explains why this type of parameter design is called robust design. Here, the term "signal" denotes the desirable value (mean response variable) and "noise" denotes the undesirable value (standard deviation). So the S/N ratio indicates the amount of variation present in the response variable. The aim is to maximize the signal-to-noise ratio. In the Taguchi method, the S/N ratio of the minimization objectives is as such [11, 13]:

S/N ratio= $-10 \log_{10} (\text{objective function})^2$

The S/N ratios are averaged in each level, and its value is plotted against each control factor in Fig. 1.

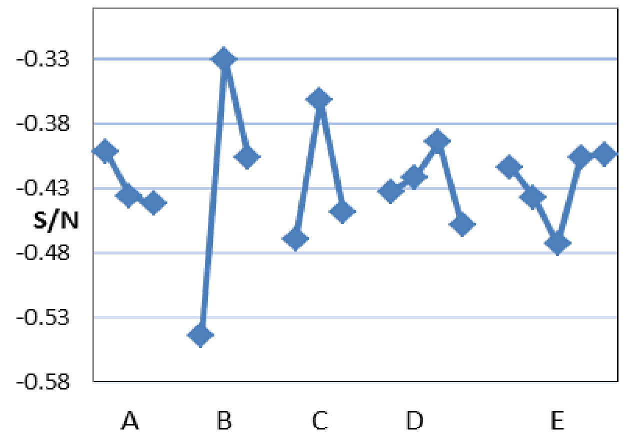

Fig. 1. Mean S/N ratio plot for each level of the factors in GA.

RESULTS

Experimental results

A computational study was conducted to evaluate the efficiency and effectiveness of the proposed algorithm, which was coded in MATLAB and run on a PC with 2.8 GHz Intel Core 2 Duo and 4 GB of RAM memory. For this purpose, we present and compare the results of GA with the SA algorithm as an effective algorithm in the literature.

We use searching time as stopping criterion to be equal for both algorithms which is equal to 1.5 × (n + m + s) milliseconds. Therefore, CPU time is affected by all the problem characteristic n, m and s. The more the number of DMUs, inputs and outputs, the more the rise of CPU time increases. Each instance is run five times. The performance measure that we will be using is the Relative Percentage Deviation (RPD) is used for each instance:

$$RPD = \frac{Max_{sol} - Alg_{sol}}{Max_{sol}} \times 100$$

Where Alg_{sol} is the obtained objective value for a given instance and Max_{sol} is the maximum or the best known solution for each instance. The problems have been run ten times and the averages of RPDs for each algorithm and each problem size are showed in Fig. 2. From this figures, it is concluded that GA has a better convergence than SA on this problems.

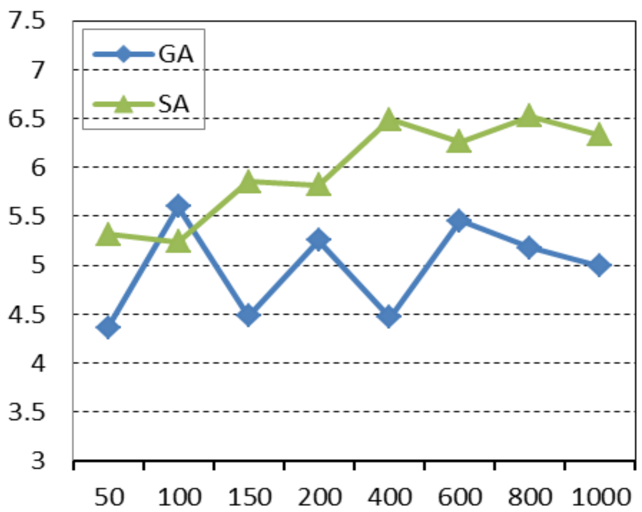

Fig. 11. Means plot for the interaction between GA, SA and problem size

CONCLUSION

DEA is a non-parametric method that is widely used for measuring the efficiency of DMUs. DEA for a large dataset with many input/output variables and/or many DMUs would require huge computer resources in terms of memory and CPU time. In this paper, we have proposed and developed the metaheuristic algorithm, GA, to obtain the relative efficiency of DMUs in large datasets.

SJMIE To cite this paper: Taghaodi R., Esfahani M.J., Molla-Alizadeh-Zavardehi S., Mahmoodi Rad A.. 2012. The Measuring Efficiency in Data Envelopment Analysis with Genetic Algorithm. *Sci. J. Mech. Ind. Eng.*, 1 (1): 36-39.
Journal homepages: http://sjmie.science-line.com/

Acknowledgements

This study was supported by Islamic Azad University, Kashan Branch. The authors are grateful for this financial support.

REFERENCES

1. Azadeh, A., Ghaderi, S.F., Tarverdian, S., Saberi, M., Forecasting electrical consumption by integration of neural network, time series and ANOVA. Applied Mathematics and Computation, 2007. 186: 1753–1761.
2. Celebi, D., Bayraktar, D., An integrated neural network and data envelopment analysis for supplier evaluation under incomplete information, Expert Systems with Applications, 2008. 35: 1698–1710.
3. Charnes, A., Cooper, W.W., & Rhodes, E, Measuring the efficiency of decision making units. European Journal of Operation Research, 1978. 2: 429–444.
4. Charnes, A., Cooper, W. W, Programming with linear fractional functionals, Naval Res. Logist. Quart., 1962: 9 (3,4): 181-185.
5. Cooper, W. W., Seiford, L. M., Tone, K, Introduction to Data Envelopment Analysis, 2006.
6. Costa, A., Markellos, R.N., Evaluating public transport efficiency with neural network models. Transportation Research Part C: Emerging Technologies, 1997. 5: 301–312.
7. Emrouznejad, A., Shale, E, A combined neural network and DEA for measuring efficiency of large scale datasets. Computers & Industrial Engineering, 2009. 56: 249–254.
8. Goldberg, D.E., Genetic Algorithms in Search, Optimization & Machine Learning, Addison-Wesley, Reading, MA , 1989.
9. Holland, J. Adaptation in natural and artificial systems. Ann Arbor: University of Michigan Press, 1975.
10. Pendharkar, P., Rodger, J., Technical efficiency-based selection of learning cases to improve forecasting accuracy of neural networks under monotonicity assumption, Decision Support Systems, 2003. 36: 117–136.
11. Rahimian, M, Mahmoodirad, A, Molla-Alizadeh-Zavardehi, S, Differential evolution Algorithm for Measuring Efficiency in DEA, Scientific Journal of Mechanical and Industrial Engineering, in press.
12. Samoilenko, S., Osei-Bryson, K.M, Determining sources of relative inefficiency in heterogeneous samples: Methodology using Cluster Analysis, DEA and Neural Networks. European Journal of Operational Research, 2010. 206: 479–487.
13. Taguchi, G, Introduction to quality engineering. White Plains: Asian Productivity Organization/UNIPUB 1986.
14. Thompson, R.G., P.S. Dharmapala and R.M. Thrall, Polyhedral Assurance Regions with Linked Constraints in W.W. Cooper and A. Whinston, eds. New Directions in Computational Economics, 1994.
15. Udhayakumar, A., Charles, V., Kumar, M. Stochastic simulation based genetic algorithm for chance constrained data envelopment analysis problems Original Research Article. Omega, 2012. 39: 387-397.
16. Wang, S., Adaptive non-parametric efficiency frontier analysis: A neural network- based model. Computers and Operations Researchm, 2003. 30: 279–295.