Utilizing Monte Carlo Method for Ranking Extreme Efficient Units in Data Envelopment Analysis

Gh-R.Jahanshahloo\textsuperscript{a}, M.Zahedi-Seresht\textsuperscript{b}\textsuperscript{*}

\textsuperscript{(a,b) Department of Mathematics, Kharazmi University, Tehran, Iran}

Received Spring 2015, Accepted Summer 2015

Abstract
Data envelopment analysis (DEA) is a mathematical programming method for calculating efficiency of decision making units (DMU). In calculating the efficiency score of units through DEA we may come up with some efficient units. But the question is among these efficient units which of them is better. As we know, it is possible to rank inefficient units through efficiency score; however, for ranking efficient units it is not helpful and other methods should be developed in these regards. To obviate this problem there have been so many attempts in the literature which have their pros and cons. Cross-efficiency method was first introduced by Sexon et al. for ranking efficient units. The major problem of this method is alternative optimal solutions in each model which must be solved for each DMU. Another problem of this method is dependency of obtained solutions on the solution obtained by other units. Another method which has widely been used is super efficiency, presented by Anderson and Petersen. There are several flaws in their suggested method. Infeasibility, instability, dependency of the model on the input and output orientation and non-zero slack variables are the weaknesses of this method which may occur in specific problems. This article is an attempt to present a method which does not have the aforementioned problems and can be utilized to calculate the rank of extreme efficient units through using the Hit or Miss Monte Carlo method. At the end of the article some examples are made in order to show the efficiency of the presented method.

Keywords: Data Envelopment Analysis, Efficiency, Ranking, Monte Carlo Simulation, Cross-Efficiency, Supper-Efficiency.

* Corresponding author: Zahedi_s@yahoo.com
1- Introduction

Data Envelopment Analysis (DEA) was first introduced by Charnes et al. [1] through presenting CCR model and it was developed further by Banker et al. [2] through introducing BCC model. DEA is an efficient method for calculating peer decision making units with some input and outputs based on the observed data. The importance of DEA lies in this fact that it can be utilized for evaluating decision making units such as banks, universities, public units, etc. Consider \( n \) number of DMUs with input vectors of \( x_j = (x_{1j}, x_{2j}, \ldots, x_{mj}) \) and output vectors of \( y_j = (y_{1j}, y_{2j}, \ldots, y_{sj}) \) which \( j = 1, \ldots, n \) such that \( x_{ij} \) and \( y_{ij} \) represent \( i \) input and \( i \) output for \( DMU_j \) \( j = 1, \ldots, n \) respectively which all of them are nonnegative. The efficient score of \( DMU_d \) is calculated by the equation 

\[
e_d(u, v) = y_d u^t / x_d v^t \]

in which \( u \) and \( v \) are the weight vectors of inputs and outputs. \( E_d(u, v) \) is called the relative efficiency of \( DMU_d \) and is defined as follows:

\[
E_d(u, v) = e_d(u, v) / \max \{ e_j (u, v) \} \\
j = 1, \ldots, n
\]

(1)

It is clear that \( E_d^* \) is not beyond 1 and each unit whose optimal objective value equals 1 is considered as efficient. For each \( n \) units, the model (1) should be solved once and the optimum solution of model (1) introduces input and output weights which is different from optimum solutions of other DMUs. As we know, it is possible the more than one DMU be efficient and their efficient scores equal to 1. It could be intriguing to study DEA in order to find how efficient units could be discriminated. There has been so much research in this area, each of them presents a method for ranking efficient units. Abello et al. [3] discussed special qualities and properties. Sexton et al. [4] proposed cross-efficiency method. This method uses weights obtained by solving each of \( n \) linear programming problems. The cross-efficiency method calculates the efficiency score of each DMU \( n \) times and save the data in a matrix. The cross-efficiency score of each DMU is included in each row of this matrix. Then the average of these rows should be calculated. This number is the ranking score of DMU. There are some problems in this method. The major one appears when the DEA models give alternative solutions. Sungmook Lim [5] replaced the secondary goal through minimizing (or maximizing) the best (or worst) cross-efficiency of peer DMUs. Another important model for ranking extreme efficient units has been proposed
by Andersen and Petersen [6]. This method removes the extreme DMU from observed set and rank ranks it based on the effect of this unit on the production possibility set. Instability and infeasibility in some cases are the main problems of this model that will be completely explained in the next section. Then, Mehrabian et al. (MAJ) [7] have improved the AP model. However in some circumstances, this model is infeasible. Saati et al. [8] have modified MAJ model and solved its infeasibility and Jahanshahloo et al. [9] have changed the type of data normalization to achieve better result. Jahanshahloo et al. [10] proposed a ranking method based on a full-inefficient frontier. This method can be used for ranking DMUs to get information about the system, and also for ranking only efficient DMUs.

Some papers based on specific norms have proposed some models to remove the difficulties of these models. For instance, Jahanshahloo et al. [11] have used L1 norm and Amirteimori et al. [12] have proposed L2 norm for ranking DMUs. Jahanshahloo et al. [13] proposed Gradient line to rank efficient units. The most advantage of this method is that it is always feasible. The L\(\infty\) norm (Tchebycheff norm) for ranking extreme efficient units was proposed by Rezai Balf et al. [14]. This method was always feasible as well. Moreover, Jahanshahloo et al. [15] proposed two ranking methods. First, they defined an ideal line and determined a common set of weights for efficient DMUs to obtain a new efficiency score and then efficient units were ranked by this score. Then they defined a special line and compared all efficient DMUs with this line for ranking. These studies continued. Chen, Y. [16] suggested new super-efficiency model to recognize the super-efficiency score when infeasibility occurs. This model can be used for ranking the efficient DMUs even if infeasibility occurs. Adler, N. et al., [17] divided the all ranking methods into six groups, somewhat overlapping areas. Abri et al. [18] proposed convex combination of extreme DMUs for ranking non-extreme DMUs.

In this article the effect of DMU on the product possibility set is to be calculated through the Hit or Miss Monte Carlo method in order to rank the intended unit. Like super-efficiency, in this paper a method is presented which is based on omitting extreme efficient DMU from observed set. The presented method obtains the rank of this unit through simple calculation without any problems
experienced in other methods, especially in super-efficiency. In part 2, we will state two widely used methods and define their uses. In part 3, production possibility set (PPS) will be introduced. Also, in this part a method will be presented can check whether the newly created artificial DMU belonging PPS created by n DMU or not. Also in this section, through examples with 2 inputs and 1 output, the rank of units is investigated by the super-efficiency method, cross-efficiency method and our presented method. In part 4, the efficiency of our method is compared with other methods through a practical example with 3 inputs and 3 outputs. Like super-efficiency method, our method can also obtain the rank of extreme efficient units without any drawbacks of other methods, specifically the super-efficiency method. At the end, a synopsis of the conducted procedure in the study is presented along with some suggestions for further studies.

2- Background

In this section, CCR model and its efficiency and the two widely used methods of super-efficiency and cross-efficiency are to be explained.

2-1- The CCR model and efficiency

The CCR model which was presented by Charnes et al. [1] is an input oriented standard model for n DMUs. Each DMU is specified by some inputs \(x_{ij} = (x_{1j}, x_{2j}, \ldots, x_{mj})\) and outputs \(y_j = (y_{1j}, y_{2j}, \ldots, y_{sj})\) in which all inputs and outputs are nonnegative and all DMUs are independent and homogenous. The efficient score of DMU \(d\) is obtained by the following model known as CCR model:

\[
\begin{align*}
\text{Min} & \quad \theta_d \\
\text{s.t.} & \quad \sum_{i=1}^{m} \lambda_i x_{ij} \leq \theta x_{id} \quad i=1,\ldots,m \\
& \quad \sum_{j=1}^{n} \lambda_j y_{ij} \geq y_{rd} \quad r=1,\ldots,s \\
& \quad \lambda_j \geq 0 \quad j=1,\ldots,n \\
\end{align*}
\]

\(\theta=1, \lambda_j = 0 \ (j \neq d)\) and \(\lambda_d = 1\) is feasible for model (2). So, the optimal solution for this model which is presented by \(\theta^*_d\) is less than 1 or equals to 1. It is clear that if \(\theta^*_d\) equals to 1, DMU \(d\) will be efficient and if \(\theta^*_d\)less than 1, DMU \(d\) will be inefficient. Dual of the model (2) is as follows:

\[
\begin{align*}
\text{Max} & \quad E_{dd} = \sum_{r=1}^{s} u_{rd} y_{rd} \\
\text{s.t.} & \quad \sum_{i=1}^{m} v_{id} x_{id} = 1 \\
& \quad \sum_{r=1}^{s} u_{rj} y_{rj} - \sum_{i=1}^{m} v_{ij} x_{ij} \leq 0 \quad j=1,\ldots,n \\
& \quad u_{rd} \geq 0, v_{id} \geq 0 \quad r=1,\ldots,s \quad i=1,\ldots,m \\
\end{align*}
\]

In this model \(E_{dd}\) equals to \(\theta^*_d\) in CCR model for DMU \(d\).

2-2- Cross-Efficiency and Supper-Efficiency
As it was mentioned, usually among evaluating units we may encounter some efficient units in which it is not possible to rank them according to the efficiency scores. Therefore, it was deemed necessary to devise new models for ranking efficient units. Scholars in this field have proposed various models for ranking efficient units which some of them will be stated below.

Sexton et al. [4] presented cross-efficiency method based on optimum weights of each unit. It should be mentioned that model (3) is equal to the following model:

\[
E_{dd}^* = \text{Max } E_{dd} = \frac{\sum_{i=1}^{m} v_{id} x_{id}}{\sum_{i=1}^{m} v_{id} x_{id}} \tag{4}
\]

s.t.
\[
E_{dj} = \frac{\sum_{i=1}^{m} u_{rd} y_{rd}}{\sum_{i=1}^{m} v_{id} x_{id}} \leq 1 \quad j=1,\ldots,n
\]
\[
u_{id} \geq 0 \quad i=1,\ldots,m
\]
\[u_{rd} \geq 0 \quad r=1,\ldots,s\]

In which \(v_{id}\) are the \(i\) input weight and \(u_{rd}\) of the \(r\) output weight of DMU \(d\). Then, the cross-efficiency for DMU \(j\) of the DMU \(d\) weights which is obtained from model (4) is as follows:

\[
E_{dj} = \frac{\sum_{i=1}^{m} u_{rd} y_{rd}}{\sum_{i=1}^{m} v_{id} x_{id}} \quad d,j=1,\ldots,n \tag{5}
\]

In which \(u_{rd}^*\) and \(v_{id}^*\) are the optimum weights of the model (4).

For DMU \(j\) (\(j = 1,2,\ldots,n\)) the average of all \(E_{dj}\) (\(d=1,\ldots,n\)) that is \(E_j\) is considered as cross-efficiency score as follows:

\[
E_j = \frac{1}{n} \sum_{d=1}^{n} E_{dj} \tag{6}
\]

As it was mentioned above, this method has some weaknesses. The major weakness of this method is alternative optimum solutions for each DMU which is obtained through model (4). Also, the solution of equation (5) directly depends on the optimal solution of model (4).

Another widely used method for efficient units is super-efficiency presented by Anderson and Petersen [6]. This method was created based on model (3) or CCR model in which we omit a DMU from the set of observations and rank it between the remained DMUs. Super-efficiency model is as follows:

\[
\text{Min } \theta \tag{7}
\]

s.t.
\[
\sum_{j=1}^{n} \lambda_j x_{ij} \leq \theta x_{io} \quad i=1,\ldots,m
\]
\[
\sum_{j=1}^{n} \lambda_j y_{rij} \geq y_{ro} \quad r=1,\ldots,s
\]
\[
\lambda_j \geq 0 \quad j=1,\ldots,n, \quad j \neq o
\]

Although this model is useful to rank extreme efficient DMUs, Thrall [19] pointed out that it may be infeasible and unstable in CCR model. However, it is not yet clear under which conditions the model is inefficient or instable. Another weakness of this model is that it just can be utilized for ranking extreme efficient DMUs, though most of the efficient units in real problems are extreme DMU.
3- Proposed Method
Consider n DMUs and assume some of them are efficient and extreme units. As mentioned before, super-efficiency and cross-efficiency methods have some drawbacks for ranking.

In this section we propose a method for ranking them through generating random DMUs in the production of possibility set (PPS).

The set of feasible activities is called PPS which is denoted by $T_c$. We postulate the PPS of the proposition of $T_c$

P1-The observed activities $(x_j,y_j)$ $(j=1,...,n)$ belong to $T_c$.

P2-If an activity $(x,y) \in T_c$ then $(\lambda x, \lambda y) \in T_c$ for every positive scalar $\lambda$.

P3-For an activity $(x,y) \in T_c$, any nonnegative activity $(\bar{x}, \bar{y}) \in T_c$ with $\bar{x} \geq x$ and $\bar{y} \leq y$.

P4-Any nonnegative linear combination of activities in $T_c$ belong to $T_c$.

Using P1, P2, P3 and P4 for n DMUs we can conclude as follows:

$$T_c = \left\{ (x,y) \left| \begin{array}{l}
\sum_{j=1}^{n} \lambda_j x_j \leq x_i \\
\sum_{j=1}^{n} \lambda_j y_j \geq y_r \\
\lambda_j \geq 0 \text{ for } j = 1, ..., n
\end{array} \right. \right\}$$

Where $\lambda$ is a nonnegative vector in $\mathbb{R}^n$.

Fig.(1) represents PPS for n unit with one input and output which was created through using 4 propositions mentioned above. As you can see in the Fig.(1), all observed DMUs are present and in addition there are other artificial DMUs.

3-1 calculating the ratio of the volume of old PPS and new PPS resulting from omitting the extreme efficient DMU

Here the question is whether one new DMU belongs to old PPS or not. How can we determine whether this DMU belongs to the PPS of the old DMUs or not?

Consider model (8):

$$\min \theta_{new}$$
$$\text{s.t. } \sum_{i=1}^{n} \lambda_i x_{i,j} \leq \theta x_{i,new} \quad i=1,\ldots,m$$
$$\sum_{j=1}^{n} \lambda_j y_{r,j} \geq y_{r,new} \quad r=1,\ldots,s$$
$$\lambda_i \geq 0 \quad j=1,\ldots,n$$

In which $(x_{new}, y_{new})$ is new DMU. In calculating model (8), if $\theta_{new}^* > 1$ or model (8) is infeasible, then new DMU does not belong to PPS and if $\theta_{new}^* \leq 1$, new DMU belongs to PPS which was created by n DMUs.

For instance, consider 3 DMUs with their PPS in Fig. (2) below:
Utilizing Monte Carlo Method for Ranking Extreme Efficient Units in Data Envelopment Analysis

Set of observations includes DMU1, DMU2 and DMU3. Now through using model (8), we want to see whether \( DMU_{\text{new}} \) belongs to the PPS created by the 3 DMUs or not. Consider Fig. (3) and model (9) for this question:

\[
\begin{align*}
\text{Min} & \quad \theta_{\text{new}} \\
\text{s.t.} & \quad \lambda_1 + 2\lambda_2 + 2\lambda_3 \leq 1/2 \theta \\
& \quad \lambda_1 + \lambda_2 + 2\lambda_3 \geq 1 \\
& \quad \lambda_1, \lambda_2, \lambda_3 \geq 0
\end{align*}
\]

In model (9), \( \theta_{\text{new}}^* = 2 \) which represents \( DMU_{\text{new}} \) doesn’t belong to the PPS created by 3 DMUs.

Now consider Fig. (4) and model (10) for new DMU:

\[
\begin{align*}
\text{Min} & \quad \theta_{\text{new}} \\
\text{s.t.} & \quad \lambda_1 + 2\lambda_2 + 2\lambda_3 \leq 3/2 \theta \\
& \quad \lambda_1 + \lambda_2 + 2\lambda_3 \geq 1 \\
& \quad \lambda_1, \lambda_2, \lambda_3 \geq 0
\end{align*}
\]
In model (16), $\theta_{new}^* = 2/3$ represents that $DMU_{new}$ belongs to the PPS created by 3 DMUs.

Now consider a PPS which is made by $n$ DMU. Through omitting extreme efficient $DMU_d$, from set of observation it is clear that $T_\epsilon$ changes to $T'_\epsilon$. It is clear that $T'_\epsilon$ is feasible as yet $n-1$ DMUs remain.

$$
T'_\epsilon = \left\{ (x,y) \mid \begin{array}{l}
\sum_{j=1}^{n} \lambda_j x_j \leq x, \\
\sum_{j=1}^{n} \lambda_j y_j \geq y,
\end{array} \right\}
\lambda_j \geq 0 \text{ for } j = 1, \ldots, n, j \neq d
$$

Consider $T = T_\epsilon - T'_\epsilon$ is the effect of $DMU_d$ on PPS and the result of $n$ DMUs. $T$ equals to the share of $DMU_d$ in making PPS and it is possible to utilize it for ranking extreme efficient units. Consider Fig.(5):

It is obvious that $DMU_1$ is the extreme efficiency. After omitting it from set of observation, $T'_\epsilon$ are made from two other DMUs. See Fig.(6):

Fig. 4. The PPS of 3DMUs and a DMU into the PPS.

Fig. 5. The PPS for 3 DMUs.
Fig. 6. The PPS before and after removing a DMU.

As shown, $DMU_1$ has a contribution in making half of the PPS before the change.

In order to calculate such a change after omitting, the super-efficiency method should solve model (7) which may cause some problems.

Infeasibility and instability are the major problems which model (7) represents. In this section, we will present a method which does not have any of aforementioned problems and can be used for ranking extreme efficient units. Then, a method will be presented which calculate the contribution of extreme efficient DMU in making PPS and we can use it for ranking extreme efficient units. In the next section the presented method ranks units without any problems mentioned for other methods.

Consider model (2) for n DMU. Assume $\bar{x}_i = \max \{x_{ij}, j = 1, \ldots, n\}$ for i=1,\ldots, m and $\bar{y}_r = \max \{y_{rj}, j = 1, \ldots, n\}$ for r=1,\ldots, s. The first category of these constraints is divided by $\bar{x}_{ij}$ (i=1,\ldots,m) and the second one is divided by $\bar{y}_r = \max \{y_{rj}, j = 1, \ldots, n\}$ for r=1,\ldots, s. Through this, data are normalized without any problem for the overall structure of the problem and data.

Min $\theta$ \hspace{1cm} (11)

s.t. $\sum_{j=1}^{m} \lambda_j \frac{x_{ij}}{\bar{x}_i} \leq \theta \frac{x_{id}}{\bar{x}_i}$ \hspace{1cm} i=1,\ldots,m

$\sum_{j=1}^{m} \lambda_j \frac{y_{rj}}{\bar{y}_r} \geq \frac{y_{rd}}{\bar{y}_r}$ \hspace{1cm} r=1,\ldots,s

$\lambda_j \geq 0$ \hspace{1cm} j=1,\ldots,n

This change does not affect CCR and PPS models. Now model (11) is rewritten into model (12):

Min $\theta$ \hspace{1cm} (12)

s.t. $\sum_{j=1}^{m} \lambda_j \bar{x}_{ij} \leq \theta \bar{x}_{io}$ \hspace{1cm} i=1,\ldots,m

$\sum_{j=1}^{m} \lambda_j \bar{y}_{rj} \geq \bar{y}_{ro}$ \hspace{1cm} r=1,\ldots,s

$\lambda_j \geq 0$ \hspace{1cm} j=1,\ldots,n
For all r,s. for all i,j and \(0 \leq \hat{y}_{rij} \leq 1\) In model (12), \(0 \leq \hat{x}_{ij} \leq 1\)
Consider model (13) and (14) as follows:

\[
\text{Min } \theta_{r'c'} \quad (13)
\]
\[
\text{s.t. } \sum_{i=1}^{n} \lambda_j x_{ij} \leq \theta x_{i_{\text{new}}} \quad i=1,\ldots,m
\]
\[
\sum_{j=1}^{n} \lambda_j y_{rj} \geq y_{r_{\text{new}}} \quad r=1,\ldots,s
\]
\[
\lambda_j \geq 0 \quad j=1,\ldots,n
\]

\[
\text{Min } \theta_{r'c'} \quad (14)
\]
\[
\text{s.t. } \sum_{j \neq d}^{n} \lambda_j x_{ij} \leq \theta x_{i_{\text{new}}} \quad i=1,\ldots,m
\]
\[
\sum_{j \neq d}^{n} \lambda_j y_{rj} \geq y_{r_{\text{new}}} \quad r=1,\ldots,s
\]
\[
\lambda_j \geq 0 \quad j=1,\ldots,n \quad j \neq d
\]

In which model (14) is obtained as the result of omitting DMU\(_d\) from model (13).
Generate new DMU with new inputs and outputs between zero and one randomly and make vectors of input and output \((x_{\text{new}},y_{\text{new}})\). Then follow these steps:
1-Consider R=1 and N’=1.
2-Solve model (14).
3-If \(\theta_{r'c'} \leq 1\) , then \((x_{\text{new}},y_{\text{new}}) \in T'_c\) ,Change N’ to N’+1 and again generate new random DMU and go to step2 otherwise go to step6.
4-If \(\theta_{r'c'} > 1\) , then solve model (13) and go to step5.
5-If \(\theta_{r'c'} \leq 1\) , then \((x_{\text{new}},y_{\text{new}}) \in T=T_c - T'_c\) ,change R to R+1 and N’ to N’+1 and again generate new random DMU and go to step2 otherwise go to step6.
6-If \(\theta^* > 1\) , then \((x_{\text{new}},y_{\text{new}}) \in T_c\) .
Again generate new random DMU and go to step2.

Where R equals to the number of random generated artificial input and output vectors in \(T=T_c - T'_c\) and it represents the contribution of DMU\(_d\) in making PPS. Also consider N’ as generated artificial input and output vectors which are in \(T\). Consider \(r_d = N'/(N' - R) \geq 1\) as scores for raking Extreme efficient DMUs. R is the effect and the contribution of DMU\(_d\) in making PPS and \(r_d\) is the efficiency score of this extreme unit. It is obvious that if R equals to zero, then \(r_d = 1\) and DMU\(_d\) is not extreme.

As it was mentioned, due to the weaknesses of cross-efficiency and super-efficiency methods, we proposed a new method. (a) In super-efficiency method through omitting an extreme efficient unit it is possible that infeasibility may occur in model (7). This problem is absent in our proposed method. (b) The second problem in this method was instability which is not present in proposed method (c) Our method is not dependent on the input and output orientation while super-efficiency method depends on the orientation. (d) Our
Utilizing Monte Carlo Method for Ranking Extreme Efficient Units in Data Envelopment Analysis

Method does not have to do with nonzero slack variables while in super-efficiency method it is one of the problems (e) alternative optimum solution is one of the drawbacks of the cross-efficiency method which does not exist in our presented method since a unique value of objective function, i.e. $\theta$, is used. (f) The efficiency score of cross-efficiency is related to the optimum solutions of other units while our proposed method is not related to other decision making units.

Consider 3 DMUs with 2 inputs and one output in table 1.

As it can be seen in table 2, all of them are efficient.

According to the obtained results in cross-efficiency and super-efficiency methods, it is seen that the rank of DMUs is different from each other. This indicates that these methods are not reliable for ranking.

Since the rank of each efficient unit is directly related to its effect on the PPS, our suggested method of ranking can calculate the rank of extreme efficient units through considering their effects on the PPS. To rank available efficient units in this question we generate 1000 random artificial DMUs with inputs and outputs between zero and one.

Through solving model (13) and (14) and the aforementioned steps, R and $N'$ are achieved for ranking. Then, we calculate $r_j (j=1,\ldots,n)$.

<table>
<thead>
<tr>
<th>DMUs</th>
<th>$DMU_A$</th>
<th>$DMU_B$</th>
<th>$DMU_C$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output y</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Input $x_1$</td>
<td>1</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>Input $x_2$</td>
<td>6</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 1. Inputs and outputs of 3 DMUs.

<table>
<thead>
<tr>
<th>DMU</th>
<th>Efficiency</th>
<th>Super Efficiency score</th>
<th>Ranking by Super Efficiency</th>
<th>Cross-Efficiency score</th>
<th>Ranking by Cross-Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>$DMU_A$</td>
<td>1.00</td>
<td>4</td>
<td>1</td>
<td>0.55</td>
<td>3</td>
</tr>
<tr>
<td>$DMU_B$</td>
<td>1.00</td>
<td>1.3</td>
<td>2</td>
<td>0.75</td>
<td>1</td>
</tr>
<tr>
<td>$DMU_C$</td>
<td>1.00</td>
<td>1</td>
<td>3</td>
<td>0.72</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 2. Ranking by super-efficiency and cross-efficiency.
From the information of table 3, A has the first rank, next is B and then C between 3 efficient DMUs.

As shown in table (4), if we increase random generated input and output vectors to 2000, the rank does not change. For 2000 generated vectors in table (4).

It can be seen from tables (3) and (4) that among efficient units A has the first rank;

B has second rank and C is the last. As R=0 for C, it can be concluded that it is not extreme DMU.

Now consider 3 new DMUs in table (5).

The result of ranking by the supper-efficiency and the cross efficiency are shown in table (6)

Table 3. The value of R and \( r_j \) for 1000 artificial generated random DMUs.

<table>
<thead>
<tr>
<th>DMUs</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>152</td>
<td>40</td>
<td>0</td>
</tr>
<tr>
<td>( N' )</td>
<td>677</td>
<td>718</td>
<td>723</td>
</tr>
<tr>
<td>( r_j )</td>
<td>1.29</td>
<td>1.06</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 4. The value of R and \( r_j \) for 2000 artificial generated random DMUs.

<table>
<thead>
<tr>
<th>DMUs</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>274</td>
<td>70</td>
<td>0</td>
</tr>
<tr>
<td>( N' )</td>
<td>1395</td>
<td>1430</td>
<td>1425</td>
</tr>
<tr>
<td>( r_j )</td>
<td>1.24</td>
<td>1.05</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 5. Inputs and outputs of 3 DMUs.

<table>
<thead>
<tr>
<th>DMUs</th>
<th>Output y</th>
<th>( DMU_A )</th>
<th>( DMU_B )</th>
<th>( DMU_C )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input ( x_1 )</td>
<td>1</td>
<td>1</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Input ( x_2 )</td>
<td>6</td>
<td>4</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

Table 6. Ranking by super-efficiency and cross-efficiency.

<table>
<thead>
<tr>
<th>DMUs</th>
<th>Efficiency</th>
<th>Super Efficiency score</th>
<th>Ranking by Super Efficiency score</th>
<th>Cross-Efficiency score</th>
<th>Ranking by Cross-Efficiency score</th>
</tr>
</thead>
<tbody>
<tr>
<td>( DMU_A )</td>
<td>1.00</td>
<td>Infeasible</td>
<td>1</td>
<td>0.65</td>
<td>3</td>
</tr>
<tr>
<td>( DMU_B )</td>
<td>1.00</td>
<td>1.3</td>
<td>1</td>
<td>0.75</td>
<td>1</td>
</tr>
<tr>
<td>( DMU_C )</td>
<td>1.00</td>
<td>1</td>
<td>2</td>
<td>0.72</td>
<td>2</td>
</tr>
</tbody>
</table>
As can be seen in table (6), model (7) is infeasible for first DMU. Now, see table (7) and table (8) which is ranked all 3 DMUs by our proposed method.
As can be seen, infeasibility does not occurred in our method.

4- Numerical example
Jahanshahlo et al., [20] used empirical example to rank 20 DMUs. This example has three inputs and outputs as it is shown in Table (9) below:
In table (10) for all inefficient DMUs it is possible to use their efficient scores and there is no problem to rank. For ranking efficient units in table (9) the new method presented in this article has been utilized which their ranks are shown in table (10).
Since R was not zero for all of them, all of the efficient units were extreme. In table (10) the ranks of all units are calculated by supper-efficiency method and cross-efficiency method. It can be seen that ranks are different in two methods while in our method there is no change in ranks with different random generated vectors.
To assess the correlation between the new methods presented in this study with those widely used and discussed in the relevant literatures, we have used the Spearman rank-order correlation test, the results of which are presented in Table 11.

Table 7. The value $r_j$ for 1000 artificial generated random DMUs.

<table>
<thead>
<tr>
<th>DMUs</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>205</td>
<td>46</td>
<td>0</td>
</tr>
<tr>
<td>$N'$</td>
<td>730</td>
<td>775</td>
<td>767</td>
</tr>
<tr>
<td>$r_j$</td>
<td>1.28</td>
<td>1.05</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 8. The value $r_j$ for 2000 artificial generated random DMUs.

<table>
<thead>
<tr>
<th>DMUs</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>384</td>
<td>63</td>
<td>0</td>
</tr>
<tr>
<td>$N'$</td>
<td>1505</td>
<td>1524</td>
<td>1509</td>
</tr>
<tr>
<td>$r_j$</td>
<td>1.34</td>
<td>1.04</td>
<td>1</td>
</tr>
</tbody>
</table>
Table 9. Data of 20 Iranian bank branches (Empirical Example).

<table>
<thead>
<tr>
<th>DMUs</th>
<th>Staff</th>
<th>Computer terminals</th>
<th>Space ($m^2$)</th>
<th>Deposits</th>
<th>Loans</th>
<th>Charges</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.950</td>
<td>0.700</td>
<td>0.155</td>
<td>0.190</td>
<td>0.521</td>
<td>0.293</td>
</tr>
<tr>
<td>2</td>
<td>0.7096</td>
<td>0.600</td>
<td>1.000</td>
<td>0.277</td>
<td>0.627</td>
<td>0.462</td>
</tr>
<tr>
<td>3</td>
<td>0.798</td>
<td>0.750</td>
<td>0.513</td>
<td>0.228</td>
<td>0.970</td>
<td>0.261</td>
</tr>
<tr>
<td>4</td>
<td>0.865</td>
<td>0.550</td>
<td>0.210</td>
<td>0.193</td>
<td>0.632</td>
<td>1.000</td>
</tr>
<tr>
<td>5</td>
<td>0.815</td>
<td>0.850</td>
<td>0.268</td>
<td>0.233</td>
<td>0.722</td>
<td>0.246</td>
</tr>
<tr>
<td>6</td>
<td>0.842</td>
<td>0.650</td>
<td>0.207</td>
<td>0.603</td>
<td>0.569</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>0.719</td>
<td>0.600</td>
<td>0.182</td>
<td>0.900</td>
<td>0.716</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>0.785</td>
<td>0.750</td>
<td>0.125</td>
<td>0.234</td>
<td>0.298</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>0.476</td>
<td>0.600</td>
<td>0.080</td>
<td>0.364</td>
<td>0.244</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0.678</td>
<td>0.550</td>
<td>0.082</td>
<td>0.184</td>
<td>0.049</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>0.711</td>
<td>1.000</td>
<td>0.318</td>
<td>0.403</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>0.811</td>
<td>0.650</td>
<td>0.255</td>
<td>0.923</td>
<td>0.628</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>0.659</td>
<td>0.850</td>
<td>0.340</td>
<td>0.645</td>
<td>0.261</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>0.976</td>
<td>0.800</td>
<td>0.144</td>
<td>0.514</td>
<td>0.243</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>0.685</td>
<td>0.950</td>
<td>1.000</td>
<td>0.262</td>
<td>0.098</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>0.613</td>
<td>0.900</td>
<td>0.115</td>
<td>0.402</td>
<td>0.464</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>1.000</td>
<td>0.600</td>
<td>0.090</td>
<td>1.000</td>
<td>0.161</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>0.634</td>
<td>0.650</td>
<td>0.059</td>
<td>0.349</td>
<td>0.068</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>0.372</td>
<td>0.700</td>
<td>0.039</td>
<td>0.190</td>
<td>0.111</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>0.583</td>
<td>0.550</td>
<td>0.110</td>
<td>0.615</td>
<td>0.764</td>
<td></td>
</tr>
</tbody>
</table>

Table 10. Ranking By our new method, super-efficiency and cross-efficiency.

<table>
<thead>
<tr>
<th>DMUs</th>
<th>Efficiency</th>
<th>Ranking With 1000 random DMUs</th>
<th>Ranking With 2000 random DMUs</th>
<th>Super-efficiency ranking</th>
<th>Cross-efficiency ranking</th>
</tr>
</thead>
<tbody>
<tr>
<td>DMU1</td>
<td>1.000</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>9</td>
</tr>
<tr>
<td>DMU2</td>
<td>0.901</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>12</td>
</tr>
<tr>
<td>DMU3</td>
<td>0.991</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>DMU4</td>
<td>1.000</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>DMU5</td>
<td>0.897</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>8</td>
</tr>
<tr>
<td>DMU6</td>
<td>0.748</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>11</td>
</tr>
<tr>
<td>DMU7</td>
<td>1.000</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>DMU8</td>
<td>0.797</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>16</td>
</tr>
<tr>
<td>DMU9</td>
<td>0.787</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td>DMU10</td>
<td>0.289</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>DMU11</td>
<td>0.604</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>14</td>
</tr>
<tr>
<td>DMU12</td>
<td>1.000</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>DMU13</td>
<td>0.816</td>
<td>11</td>
<td>11</td>
<td>11</td>
<td>10</td>
</tr>
<tr>
<td>DMU14</td>
<td>0.469</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>17</td>
</tr>
<tr>
<td>DMU15</td>
<td>1.000</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>DMU16</td>
<td>0.639</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>DMU17</td>
<td>1.000</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>DMU18</td>
<td>0.472</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>18</td>
</tr>
<tr>
<td>DMU19</td>
<td>0.408</td>
<td>19</td>
<td>19</td>
<td>19</td>
<td>19</td>
</tr>
<tr>
<td>DMU20</td>
<td>1.000</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>7</td>
</tr>
</tbody>
</table>
Table 11. Spearman rho for proposed method and other methods.

<table>
<thead>
<tr>
<th>methods</th>
<th>Spearman rank-order correlation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our method &amp; Super-efficiency</td>
<td>1</td>
</tr>
<tr>
<td>Our method &amp; Cross-efficiency</td>
<td>0.925</td>
</tr>
</tbody>
</table>

As can be seen in Table 11, there is a strong correlation between this new method and other widely used ones and scatterplots are as follow:

As the scatter plots Fig (7) and Fig (8) and Table 11 suggest, there is a strong correlation between the new proposed method for ranking and other widely used methods. However, our method can do without such complications as infeasibility, alternative optimal solutions, and complex computations, associated with the conventional ones.

Fig. 7. Scatterplot of Our new method vs Super-efficiency.

Fig. 8. Scatterplot of Our new method vs Cross-efficiency.
However, some questions might arise regarding this new method: how many random numbers should be generated for computing? Or at what point can one be assured that the ranking of the DMUs would not change anymore? To assure that the ranking would not change anymore when using the our new method, go through the following steps:
1- Let C=1000.
2- Repeat the algorithmic steps presented in Section3.
3- Let C=C+a (where a can be defined by the user)
4- Repeat the algorithmic steps presented in Section3.
5- If the ranking with C random numbers is the same as the ranking with C-a, proceed to step 6; otherwise, return to step 3.
6- Now, for measuring validity, use the Spearman rank-order correlation test for ranking with our new method and other conventional methods.
As in Table 10, we can increase the number of random numbers until the ranking offered by this method turns out to be the same in two consecutive steps.

**5- Conclusion**

In this article 2 well-known and highly used methods, i.e. supper-efficiency and cross-efficiency were presented and their drawbacks were mentioned. Then, a new method was presented which through utilizing Monte Carlo method the effect of each of extreme efficient DMUs on PPS is analyzed without any of the drawbacks of previous methods and the rank of each DMU is obtained. An important question which was raised here is that how many random vectors should be built for input and output in order to calculate the rank of each unit. The answer is up to the point that ranks do not change anymore. This number can be called as the reliability point for the generated ranks.
Utilizing Monte Carlo Method for Ranking Extreme Efficient Units in Data Envelopment Analysis

References


