Available online at http://ijim.srbiau.ac.ir

Int. J. Industrial Mathematics Vol. 3, No. 1 (2011) 41-53

Ranking Efficient Units by Regular Polygon Area (RPA) in DEA

F. Rezai Balf *

Department of Mathematics, Qaemshahr Branch, Islamic Azad University, Qaemshahr, Iran.

Received 29 September 2010; revised 29 January 2011; accepted 9 February 2011.

Abstract

In this paper, we address the problem of assessing the rank of the set of efficient units in Data Envelopment Analysis (DEA). DEA measures the efficiencies of decision-making units (DMUs) within the range of less than or equal to one. The corresponding efficiencies are referred to as the best relative efficiencies, which measure the best performance of DMUs and determine an efficiency frontier. This research proposes a methodology to use an common set of weights for the performance indices of only DEA efficient DMUs. Then DMUs are ranked according to the efficiency score weighted by the common set of weights in two dimensional space.

Keywords : Data Envelopment Analysis (DEA); Weights Analysis; Efficiency; Ranking.

1 Introduction

Data Envelopment Analysis (DEA) is a nonparametric method of measuring the efficiency of a decision-making unit (DMU) such as a firm, first introduced by Charnes, Cooper, and Rhodes (CCR) (1978). To measure the technical efficiency of any observed input-output bundle, one needs to know the maximum quantity of output that can be produced from the relevant input bundle. However, in DEA we construct a benchmark technology from the observed input-output bundles of the firms in the sample, such as a production frontier. Justifying each unit on frontier is interpreted as efficiency and any deviation from this frontier is interpreted as inefficiency. Firms that are found to be technically inefficient can be ranked aspect of their measured levels of efficiency. Firms that are found to be efficient are, also, all ranked equally by a criterion. Andersen and Petersen (1993) suggested a criterion that permits one to rank order firms that have all been found to be at 100It is worthwhile that noted AP model can be infeasible, sometimes. A potential

*Email address: frb_balf@yahoo.com, Tel:+98-123-2244111

problem of feasibility with these supper efficiency models has been noted by Dula and Hickman (1997), Seiford and Zhu (1999), Harker and Xue (2002), and Lovel and Rouse (2003). For some efficient observations, there may not exist any input-oriented or output-oriented projection onto a frontier that is constructed from the remaining observations in the data set. In this study, we introduce a different approach for ranking efficient units. In our approach, we aim to obtain one common set of weights (CSW) to create the identity critical of projection the efficient units in two dimensional space. Then, we calculate the area of the regular polygon constructer of all efficient units. However, we rank the projected efficient units. The ranking that adopts the one common set of weights generated from our methodology makes sense because a decision maker objectively chooses the one common set of weights for the purpose of maximizing the group efficiency. The second section of this paper represents discussion about regular polygon area (RPA). In Section 3, a method for finding CSW is briefly discussed. Section 4, present a brief discussion about supper-efficiency ranking techniques. Section 5, gives a complete ranking of DMUs by RPA method. Numerical example and conclusion of the methods are presented in the last sections.

2 Regular polygon area (RPA)

In this section, we present a rule for calculating of regular polygon area (RPA). For this purpose, we first consider a triangle as simplest regular polygon and then introduce general case for finding regular polygon area. Consider the Cartesian coordinates system in two dimensional spaces. Four possible cases exist for appointing origin, when we depict a triangle in this system. First case, the origin is one of vertexes, second case the origin lie inside triangle, in third case the origin lie outside triangle and finally the last case origin lie on one of triangle edges. However, we will show the area of a triangle can be written as determinant form for each of above quaternion cases. Also, it can be shown that if points O, A and B be triangle vertexes in anti-clock wise sense respectively, then determinant value is positive and it is negative when these points are clock wise. It should be note the value of area is positive.

Theorem 2.1. *If points $O(0,0), A(x_1, y_1)$ and $B(x_2, y_2)$ be coordinates of triangle vertexes then the area of triangle $\triangle OAB$ determine as follows:*

$$S_{\triangle OAB} = \frac{1}{2}\begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix}$$

Proof: Consider $\triangle OAB$, according to Fig. 1.

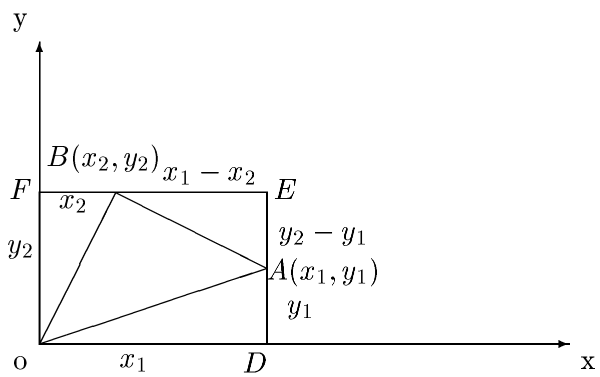

Fig. 1. The graph of triangle OAB

Obviously, finding the area $\triangle OAB$ equal to subtracting the area three triangles $\triangle OAD$,$\triangle OFB$ and $\triangle EAB$ of the rectangular area $\square ODEF$. Hence, it can be shown that

$$\begin{aligned} S_{\triangle OAB} &= S_{\square ODEF} - S_{\triangle OFB} - S_{\triangle OAD} - S_{\triangle EAB} \\ &= x_1y_2 - \tfrac{x_2y_2}{2} - \tfrac{x_1y_1}{2} - \tfrac{(x_1-x_2)(y_2-y_1)}{2} \\ &= x_1y_2 - \tfrac{x_2y_2}{2} - \tfrac{x_1y_1}{2} - \tfrac{x_1y_2-x_1y_1-x_2y_2+x_2y_1}{2} \\ &= \tfrac{x_1y_2-x_2y_1}{2} \\ &= \tfrac{1}{2}\begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} \end{aligned}$$

The proof is complete.

In continue we extend this topic when origin is not triangle vertex (See Fig. 2 and Fig. 3).

Theorem 2.2. *If points $A(x_1,y_1), B(x_2,y_2)$ and $C(x_3,y_3)$ be arbitrarily coordinates of triangle vertexes in anti-clock wise sense then the area of triangle $\triangle ABC$ is determined as follows:*

$$S_{\triangle ABC} = \frac{1}{2}\left\{\begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} + \begin{vmatrix} x_2 & y_2 \\ x_3 & y_3 \end{vmatrix} + \begin{vmatrix} x_3 & y_3 \\ x_1 & y_1 \end{vmatrix}\right\}$$

Proof: For proof consider two below cases:

1. The origin is outside triangle (see Fig. 2)
2. The origin is inside triangle (see Fig. 3)

In first case according to Fig. 2, we obtain:

$$S_{\triangle ABC} = S_{\triangle OAB} + S_{\triangle OBC} - S_{\triangle OAC}$$

and with respect to Theorem (2.1) we have:

$$S_{\triangle ABC} = \frac{1}{2}\begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} + \frac{1}{2}\begin{vmatrix} x_2 & y_2 \\ x_3 & y_3 \end{vmatrix} - \frac{1}{2}\begin{vmatrix} x_1 & y_1 \\ x_3 & y_3 \end{vmatrix}$$

According to determinant property, Substitute;

$$-\frac{1}{2}\begin{vmatrix} x_1 & y_1 \\ x_3 & y_3 \end{vmatrix} = \frac{1}{2}\begin{vmatrix} x_3 & y_3 \\ x_1 & y_1 \end{vmatrix}$$

then, we obtain

$$S_{\triangle ABC} = \frac{1}{2}\{\begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} + \begin{vmatrix} x_2 & y_2 \\ x_3 & y_3 \end{vmatrix} + \begin{vmatrix} x_3 & y_3 \\ x_1 & y_1 \end{vmatrix}\}$$

The proof is complete.
The proof of second part of Theorem (2.2) is similar to the first part according to Fig. 3. We ignore the proof of it.

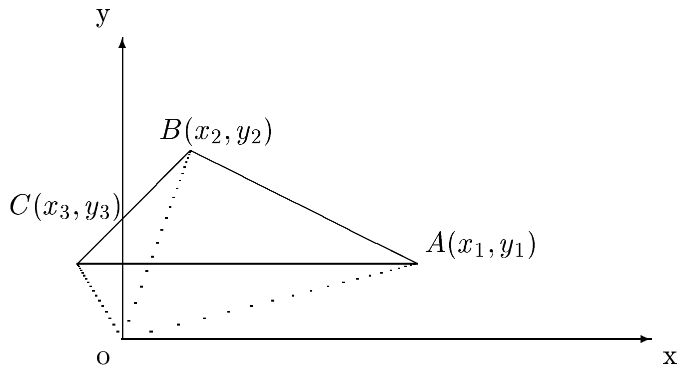

Fig. 2. The graph of triangle $\triangle ABC$ that isn't contain origin.

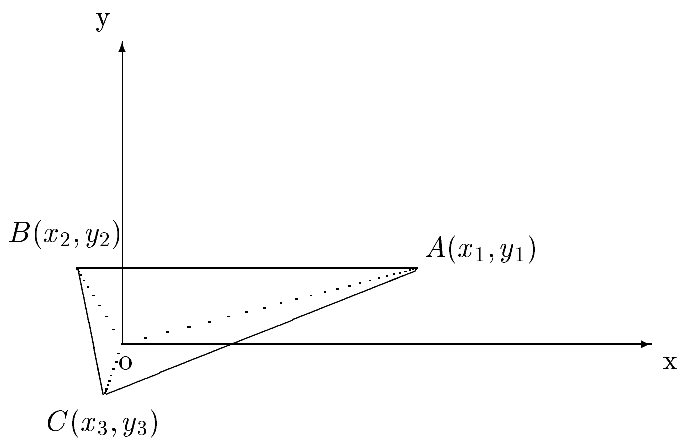

Fig. 3. The graph of triangle $\triangle ABC$ contain origin.

At this point, we interest to extend the above method for finding the regular polygon area when the coordinate vertexes are available. It is mentionable that every regular polygon with n vertex can be partitioned to $n-2$ triangular.

Theorem 2.3. *The area of any regular polygon with $p_j(x_j, y_j)$, $j = 1, ..., n$ vertex in anti-clock wise sense is as follows*

$$S_{p_1p_2...p_n} = \frac{1}{2}\{\begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} + \begin{vmatrix} x_2 & y_2 \\ x_3 & y_3 \end{vmatrix} + \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix} + ... + \begin{vmatrix} x_n & y_n \\ x_1 & y_1 \end{vmatrix}\}$$

Proof: The proof is by induction over : Consider the following relation,

$$S_{p_1p_2...p_n} = S_{\triangle p_1p_2p_3} + S_{\triangle p_1p_2p_4} + ... + S_{\triangle p_1p_kp_{k+1}} + ... + S_{\triangle p_1p_{n-1}p_n}$$

where it can be written as

$$S_{p_1p_2...p_n} = \frac{1}{2}\left\{\begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} + \begin{vmatrix} x_2 & y_2 \\ x_3 & y_3 \end{vmatrix} + \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix} + ... + \begin{vmatrix} x_n & y_n \\ x_1 & y_1 \end{vmatrix}\right\}$$

This relation has been proved for $n = 3$ already. Suppose that it satisfy in case $n = k$ as hypothesis induction:

$$S_{p_1p_2\ldots p_k} = \frac{1}{2}\left\{\begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} + \begin{vmatrix} x_2 & y_2 \\ x_3 & y_3 \end{vmatrix} + \ldots + \begin{vmatrix} x_k & y_k \\ x_1 & y_1 \end{vmatrix}\right\}$$

We show the above relation due for $n = k + 1$ (assertion induction) as

$$S_{p_1p_2\ldots p_{k+1}} = \frac{1}{2}\left\{\begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} + \begin{vmatrix} x_2 & y_2 \\ x_3 & y_3 \end{vmatrix} + \ldots + \begin{vmatrix} x_k & y_k \\ x_{k+1} & y_{k+1} \end{vmatrix} + \begin{vmatrix} x_{k+1} & y_{k+1} \\ x_1 & y_1 \end{vmatrix}\right\}$$

Hence, for this purpose it is sufficient we add $S_{\triangle p_1p_kp_{k+1}}$ to the two parts of hypothesis induction, we obtain

$$S_{p_1p_2\ldots p_k} + S_{\triangle p_1p_kp_{k+1}} = \frac{1}{2}\left\{\begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} + \begin{vmatrix} x_2 & y_2 \\ x_3 & y_3 \end{vmatrix} + \ldots + \begin{vmatrix} x_k & y_k \\ x_1 & y_1 \end{vmatrix}\right\} + S_{\triangle p_1p_kp_{k+1}}$$

With substituting

$$S_{\triangle p_1p_kp_{k+1}} = \frac{1}{2}\left\{\begin{vmatrix} x_1 & y_1 \\ x_k & y_k \end{vmatrix} + \begin{vmatrix} x_k & y_k \\ x_{k+1} & y_{k+1} \end{vmatrix} + \begin{vmatrix} x_{k+1} & y_{k+1} \\ x_1 & y_1 \end{vmatrix}\right\}$$

to above relation, the assertion is satisfied

$$S_{p_1p_2\ldots p_n} = \frac{1}{2}\left\{\begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} + \begin{vmatrix} x_2 & y_2 \\ x_3 & y_3 \end{vmatrix} + \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix} + \ldots + \begin{vmatrix} x_n & y_n \\ x_1 & y_1 \end{vmatrix}\right\}$$

For further description consider the following example.

Example 2.1. *Find the area of following* 5*- regular polygon, according to Fig. 4.*

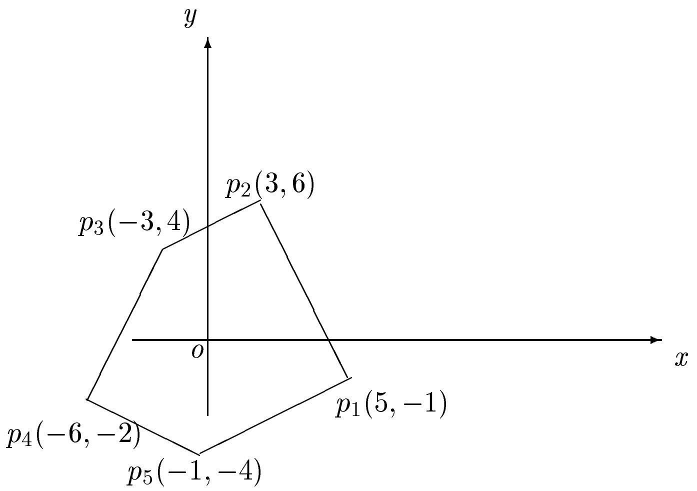

Fig.4. The graph of 5*- regular polygon.*

$$\begin{aligned} S_{p_1p_2p_3p_4p_5} &= S_{\triangle p_1p_2p_3} + S_{\triangle p_1p_3p_4} + S_{\triangle p_1p_4p_5} \\ &= \tfrac{1}{2}\left\{\begin{vmatrix} 5 & -1 \\ 3 & 6 \end{vmatrix} + \begin{vmatrix} 3 & 6 \\ -3 & 4 \end{vmatrix} + \begin{vmatrix} -3 & 4 \\ 5 & -1 \end{vmatrix} + \begin{vmatrix} 5 & -1 \\ -3 & 4 \end{vmatrix} + \begin{vmatrix} -3 & 4 \\ -6 & -2 \end{vmatrix}\right. \\ &\quad \left. + \begin{vmatrix} -6 & -2 \\ 5 & -1 \end{vmatrix} + \begin{vmatrix} 5 & -1 \\ -6 & -2 \end{vmatrix} + \begin{vmatrix} -6 & -2 \\ -1 & -4 \end{vmatrix} + \begin{vmatrix} -1 & -4 \\ 5 & -1 \end{vmatrix}\right\} = 68 \end{aligned}$$

3 A method for finding common set of weights (CSW)

The flexibility in the choice of weights is both a weakness and strength. It is a weakness because the judicious choice of weights by a unit possibly unrelated to the value of any input or output may allow a unit to appear efficient but there may be concern that this is more to do with the choice of weights than any inherent efficiency. This flexibility is also strength, however, if a unit turns out to be inefficient even when the most favourable weights have been incorporated in its efficiency measure then this is a strong statement.
This section presents a multiple objective programming procedure for finding a common set of weights in DEA [9]. An important outcome of such an analysis is a set of virtual multipliers or weights accorded to each factor taken into account. These sets of weights are, typically, different for each of the participating DMUs. In this section, by means of solving only one problem, we can determine a common set of weights.
In DEA for calculating the efficiency of different DMUs, different set of weights are obtained, which seems to be unacceptable in reality. So the following model is used to find common set of weights which has some advantages that will be discussed later on. This idea is formulated as maximizing the ratio of outputs and inputs simultaneously for all DMUs. So we presents the following multiple objective functional programming problem.

$$
\begin{array}{lll}
max & \left\{\frac{\sum_{r=1}^{s} u_r y_{r1}}{\sum_{i=1}^{m} v_i x_{i1}}, \ldots, \frac{\sum_{r=1}^{s} u_r y_{rn}}{\sum_{i=1}^{m} v_i x_{in}}\right\} & \\
S.t. & \frac{\sum_{r=1}^{s} u_r y_{rj}}{\sum_{i=1}^{m} v_i x_{ij}} \leq 1, & j=1,\ldots,n \\
 & u_r \geq \epsilon, & r=1,\ldots,s \\
 & v_i \geq \epsilon, & i=1,\ldots,m
\end{array}
\tag{3.1}
$$

where $U=(u_1,\ldots,u_s)^T$ and $V=(v_1,\ldots,v_m)^T$ are the weights of outputs and inputs, respectively, and ϵ is a positive non-Archimedean infinitesimal.
For solving this problem, the following procedure is suggested. Here we consider the infinite norm, so it tends the maximization of the objective function pertaining the DMU will minimum ratio of outputs to inputs.

$$
\begin{array}{lll}
max & \left\{min\left\{\frac{\sum_{r=1}^{s} u_r y_{r1}}{\sum_{i=1}^{m} v_i x_{i1}}, \ldots, \frac{\sum_{r=1}^{s} u_r y_{rn}}{\sum_{i=1}^{m} v_i x_{in}}\right\}\right\} & \\
S.t. & \frac{\sum_{r=1}^{s} u_r y_{rj}}{\sum_{i=1}^{m} v_i x_{ij}} \leq 1, & j=1,\ldots,n \\
 & u_r \geq \epsilon, & r=1,\ldots,s \\
 & v_i \geq \epsilon, & i=1,\ldots,m
\end{array}
\tag{3.2}
$$

By introducing non-negative variable z , problem (3.2), can be converted to the following problem:

$$
\begin{aligned}
max \quad & z \\
S.t. \quad & \textstyle\sum_{r=1}^{s} u_r y_{rj} - z \sum_{i=1}^{m} v_i x_{ij} \geq 0, \quad && j = 1, \ldots, n \\
& \textstyle\sum_{r=1}^{s} u_r y_{rj} - \sum_{i=1}^{m} v_i x_{ij} \leq 0, && j = 1, \ldots, n \\
& z \geq 0 \\
& u_r \geq \epsilon, && r = 1, \ldots, s \\
& v_i \geq \epsilon, && i = 1, \ldots, m
\end{aligned}
\tag{3.3}
$$

Note that instead of solving n linear programming DEA models, only one non-linear programming problem is solved and the efficiency for all DMUs are obtained.

4 Supper-efficiency ranking techniques

Suppose we have a set of n (productive units), DMUs. Each DMU_j, $j = 1, ..., n$ consumes m different inputs to produce s different outputs. Two types of orientation DEA models are often used to evaluate DMUs' relative efficiency: CRS models, such as CCR model, and VRS models, such as BCC model. For example, CCR model in multiplier form is defined as a linear programming model as follows:

$$
\begin{aligned}
max \quad & \textstyle\sum_{r=1}^{s} u_r y_{ro} \\
S.t. \quad & \textstyle\sum_{r=1}^{s} u_r y_{rj} - \sum_{i=1}^{m} v_i x_{ij} \leq 0, \quad && j = 1, \ldots, n \\
& \textstyle\sum_{i=1}^{m} v_i x_{io} = 1, \\
& u_r \geq \epsilon, && r = 1, \ldots, s \\
& v_i \geq \epsilon, && i = 1, \ldots, m
\end{aligned}
\tag{4.4}
$$

Where ϵ is a non-Archimedean element defined to be smaller than any positive real number. The BCC (Banker et al., 1984) model adds an additional constant variable, u_o , in order to permit variable return-to-scale:

$$
\begin{aligned}
max \quad & \textstyle\sum_{r=1}^{s} u_r y_{ro} + u_o \\
S.t. \quad & \textstyle\sum_{r=1}^{s} u_r y_{rj} - \sum_{i=1}^{m} v_i x_{ij} + u_o \leq 0, \quad && j = 1, \ldots, n \\
& \textstyle\sum_{i=1}^{m} v_i x_{io} = 1, \\
& u_r \geq \epsilon, && r = 1, \ldots, s \\
& v_i \geq \epsilon, && i = 1, \ldots, m
\end{aligned}
\tag{4.5}
$$

In most models of Data Envelopment Analysis (DEA) (Charnes et al., 1978 ; Banker et al., 1984; Cooper et al., 2000) [3, 2, 4], the best performers have efficiency score unity, and these units lie on frontier efficiency. Several authors have proposed methods for ranking these efficient units. See Andersen and Petersen (1993) [1], Doyle and Green (1993, 1994) [6, 7], Stewart (1994) [13], Tofallis (1996)[10], Seiford and Zhu (1999)[11], Mehrabian (1999)[10], Zhu (2001)[15] and Jahanshahloo (2005)[8], among others. It is mentioned that only Jahanshahloo method is able to rank all kind of efficient DMUs (extreme and non-extreme efficient DMUs), while the other methods are not able to rank non-extreme efficient DMUs. For example, Andersen and Petersen (1993) developed a new procedure for ranking efficient units. The methodology enables an extreme efficient unit "o" to obtain an efficiency score greater than one by eliminating the o-th constraint in the model (4.4), as shown in model (4.6).

$$\begin{array}{lll} max & \sum_{r=1}^{s} u_r y_{ro} & \\ S.t. & \sum_{r=1}^{s} u_r y_{rj} - \sum_{i=1}^{m} v_i x_{ij} \leq 0, & j = 1, \dots, n, j \neq o \\ & \sum_{i=1}^{m} v_i x_{io} = 1, & \\ & u_r \geq \epsilon & r = 1, \dots, s \\ & v_i \geq \epsilon & i = 1, \dots, m \end{array} \tag{4.6}$$

The next section develops the new method. Our goal is to translate the basic idea of combining RPA and DEA in order to determining ranking of efficient units.

5 Ranking of DMUs by RPA method

This section deals with ranking of DMUs by RPA method. Suppose that we have n DMUs each with m inputs and s outputs. The vectors v and u are the weight vectors for input and output, respectively. We have to somehow solve for each DMU a linear programming (LP) which can lead to a different optimal solution. The index "o" selects the DMU for which the optimization should be evaluated, as shown in models (4.4) or (4.5), according to constant return to scale or variable return to scale. Throughout this section we assume that efficiency is evaluated by means of the CCR model (4.4) of technical efficiency. This measure was introduced by Charnes et al. (1978)[3]. We assume that in evaluating DMUs with model (4.4), k DMUs ($k \leq n$) , are efficient. The set $E_o = \{j \mid DMU_j \ \ is \ \ efficient\}$ and (v^*, u^*) be optimal common set of weights by model (3.3). Let us define f function as:

$$f : R^{m+s} \rightarrow R^2$$

$$f(x, y) = (v^* x, u^* y)$$

Define the set B as follows:

$$B = \{z_j \mid z_j = (v^* x_j, u^* y_j), \ j \in E_o, \ x_j \in R^m, \ y_j \in R^s\}$$

It is obviously, $z_j \geq 0 = (0, 0), \ j \in E_o$ and $B \subseteq R^2_+$. In fact the set B is projection of A, in two dimension plane under common set of weights. However, we rank the members of the set B instead of $DMU_j, \ j \in E_o$.

5.1 The ranking method for members in the set B

In this section, we describe the ranking approach for the members of the set B. Suppose that T be convex hall of $\{\ z_j \mid\ z_j \in B\}$. Set $T = convex(B)$. It is trivial that T is a regular polygonal in R^2_+ . Suppose that S be area of T. Set, $S = RPA(T)$. For ranking z_p in B, we first remove it from the set of T then we obtain (another) convex hall $T_p = T - \{z_p\}$. Let $S_p = RPA(T_p)$. It is obvious $T_p \subseteq T$ and $S_p \leq S$, (see Fig. 5), and set $\theta_p = S - S_p \geq 0$. This value θ_p is defined the rank of z_p .

As already mentioned, z_j is as a point of $T = convex(B)$. If z_j be extreme point in T , then $\theta_p > 0$, meanwhile if z_j be non-extreme point of T , then $\theta_p = 0$. For more description define $B^+ = \{\ z_j \mid \theta_j > 0,\ j \in B\}$ and $B^0 = \{\ z_j \mid \theta_j = 0,\ j \in B\}$. Let $Card(B^0) = k_0$ and $Card(B^+) = k_1$, then it is obvious that, $Card(B^0) + Card(B^+) = Card(B)$, that is, $k_0 + k_1 = k$. Then, we present the below definition:

Definition 5.1. *Suppose that $z_p,\ z_q \in B^+$, then z_p has higher rank of z_q if and only if $\theta_p > \theta_q$.*

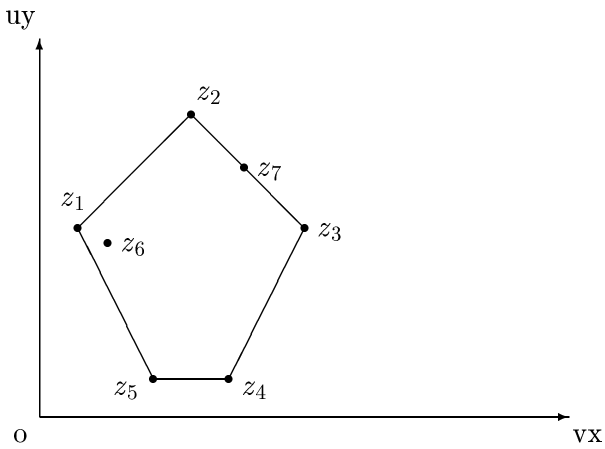

Fig. 5. a. Convex hall of $\{z_j\}$, $\jmath = 1, \ldots, 7$.

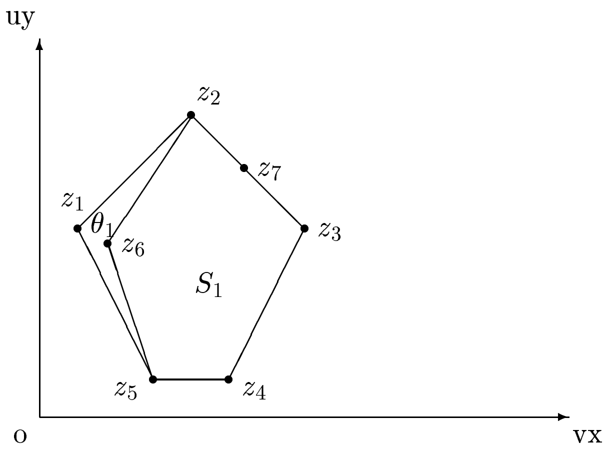

Fig. 5. b. Convex hall of $\{z_j\}$, $\jmath = 2, \ldots, 7$.

Fig. 5. a shows a convex hall of 7 point together with its area S , meanwhile, Fig. 5. b shows a convex hall after removing z_1 . We gain θ_1 as its rank , that is, $\theta_1 = S - S_1 > 0$.

Now if we verify points z_6 or z_7 we obtain $\theta_6 = \theta_7 = 0$. In last case we should be rank the units z_6 and z_7, in the convex set $T_0 = convex\{z_6, z_7\}$.

Some notes are worthwhile:

Note 1. Suppose that $z_p,\ z_q \in B^+$, then the probability $\theta_p = \theta_q$ tend to vanish. It is trivial that points in B^+ have higher rank than the points in B^0 . Nevertheless, we interest to rank points belong to B^0 in second order, according to mentioned method based on the new convex set $T_0 = convex(B - B^+)$.

Note 2. If $T = convex(B)$ be a segment line, that is,$S = 0$, then we suppose that this probability is zero. Otherwise this method encounter with a problem.

Note 3. For non-extreme points which have the zero rank value, it is suggested the distance of origin be as a criterion for ranking.

6 Numerical example

Example 6.1. *To illustrate the application of this method, we consider 19 with two inputs and two outputs (Table 1).*

Table 1
The value of inputs and outputs

DMUs	Input1	Input2	Output1	Output2
1	81	87.6	5191	205
2	85	12.8	3629	0
3	56.7	55.2	3302	0
4	91	78.8	3379	0
5	216	72	5368	639
6	58	25.6	1674	0
7	112.2	8.8	2350	0
8	293.2	52	6315	414
9	186.6	0	2865	0
10	143.4	105.2	7689	66
11	108.7	127	2165	266
12	105.7	134.4	3963	315
13	235	236.8	6643	236
14	146.3	124	4611	128
15	57	203	4869	540
16	118.7	48.2	3313	16
17	58	47.4	1853	230
18	146	50.8	4578	217
19	0	91.3	0	508

Table 2
The results of using different models for ranking of DMUs

DMUs	1	2	5	9	15	19
AP rank	4	1	3	-	2	-
MAJ rank	5	3	2	6	4	1
Tch. Norm rank	5	2	3	6	4	1
RPA rank	6	4	3	1	2	5

DMUs 1, 2, 5, 9, 15 and 19 are CCR efficient. The results of ranking using RPA method are compared with Tchebycheff norm, AP and MAJ methods in Table 2. As shown in Table 2, DMU19 and DMU9 have highest and lowest rank in MAJ and Tchebycheff models, respectively. Meanwhile, both of them (DMU19 and DMU9) are infeasible in AP model. Also, notice that, all DMUs are ranked very close to each other in MAJ and Tchebycheff models, while this is not in AP model. In model AP, DMU2 and DMU1 have first and last rank respectively. The operations for ranking efficient units 1, 2, 5, 9, 15 and 19 according to model (3.3), the common set of weights have obtained as follows:

$$u_1^* = 0.01000, \quad u_2^* = 0.089560, \quad v_1^* = 0.351901, \quad v_2^* = 0.498316, \quad z^* = 0.44$$

Therefore we acquire:

$$z_1 = (72.16, 70.27), \quad z_2 = (36.29, 36.29), \quad z_5 = (111.89, 110.91), \quad z_9 = (65.66, 28.65),$$
$$z_{15} = (121.22, 97.05), \quad z_{19} = (45.5, 45.5)$$

Now we rank the units of the set $B = \{z_1, z_2, z_5, z_9, z_{15}, z_{19}\}$

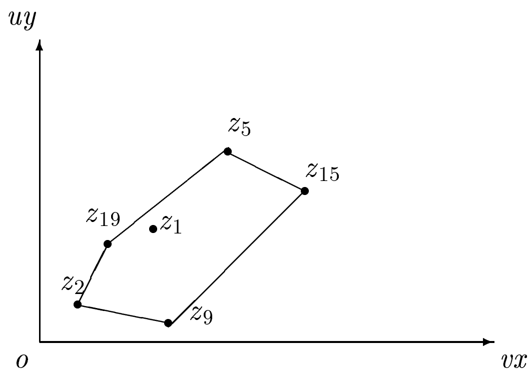

Fig.6. Convex hall of members of B

For ranking units, first we compute $RPA(T) = S$ *. According to the Fig. 6, consider that* z_1 *don't effective in constructive* $RPA(T) = S$ *. Hence; we obtain:*

$$S = \frac{1}{2}\left\{ \begin{vmatrix} 65.66 & 28.65 \\ 121.22 & 97.05 \end{vmatrix} + \begin{vmatrix} 121.22 & 97.05 \\ 111.89 & 110.91 \end{vmatrix} + \begin{vmatrix} 111.89 & 110.91 \\ 45.5 & 45.5 \end{vmatrix} + \begin{vmatrix} 45.5 & 45.5 \\ 36.29 & 36.29 \end{vmatrix} \right.$$
$$\left. + \begin{vmatrix} 36.29 & 36.29 \\ 65.66 & 28.65 \end{vmatrix} \right\} = 4186.43$$

where $\theta_i = S - S_i$ *have obtained follows:*

$$\theta_1 = 0, \quad \theta_2 = 340.87, \quad \theta_5 = 1029.17, \quad \theta_9 = 2433.39, \quad \theta_{15} = 1408.23, \quad \theta_{19} = 9.03$$

7 Conclusion

In this paper, we have addressed a different rankingof efficient units which is called regular polygon area (RPA) method. In our approach, first efficient units are transformed into two dimension space by a common set of weights. Then the area from the regular polygon constructer of all projected efficient units is considered calculatal. However, we ranked the projected efficient units according to the difference between regular polygon areas before and after removed. We also suggested that one can be work on this method over imprecise data [5], and focus on Stability regions for keeping efficiency [15]. Reader should attend that this method has a drawback, because the projection function of $f : R^{m+s} \to R^2, \;\; f(x, y) = (vx, uy)$ is not injective map.

References

[1] P. Andersen, N.C. Petersen, A procedure for ranking efficient units in data envelopment analysis, Management Science 39 (1993) 1261-1264.

[2] R.D. Banker, A. Charnes, W.W. Cooper, Some model for estimating technical and scale inefficiencies in data envelopment analysis, Manegement Science 30 (1984) 1078-1092.

[3] A. Charnes, W.W. Cooper, E. Rhodes, Measuring the efficiency of decision making units, European Journal of Operating Research 2 (6) (1978) 429-444.

[4] W.W. Cooper, L.M. Seiford, K. Tone, Data Envelopment Analysis - A Comprehensive Text with Models, Application, References and DEA-Solver Software. Kluwer Academic Publishers, Dordrecht 2000.

[5] D.K. Despotis, Y.G. Smirlis, data envelopment analysis with imprecise data, European Journal of Operational Research 140 (2002) 243-246.

[6] J. Doyle, R. Green, Data envelopment analysis and multiple criteria decision making, Omega 21 (1993) 713-713.

[7] J. Doyle, R. Green, Efficiency and cross-efficiency in DEA; Derivations, meanings and uses, Journal of the Operational Research Society 45 (1994) 567- 578.

[8] G.R. Jahanshahloo, F. Hosseinzadeh Lotfi, H. Zhiani Rezai, F. Rezai Balf, Using Monte Carlo method for ranking efficient DMUs, Applied Mathematics and Computation 162 (2005) 371-379.

[9] G.R. Jahanshahloo, M.R. Alirezai, S. Saati, S. Mehrabian, The Role of Multipliers in DEA, with an empirical study, J. Economic Mgmt. Quarterly, J. Islamic Azad University, 6 331-347.

[10] S. Mehrabian, M.R. Alirezaee, G.R. Jahanshahloo, A complete efficiency ranking of decision making units in data envelopment analysis, Computational Optimisation and Application 14 (1999) 261-266.

[11] L.M. Seiford, J. Zhu, Infeasibility of super-efficiency data envelopment analysis Models. INFORS 37 (1999) 174-187.

[12] L.M. Seiford, J. Zhu, Stability regions for maintaining efficiency in data envelopment analysis, European Journal of Operational Research 108 (1998) 127-139.

[13] T.J. Stewart, Data envelopment analysis and multiple criteria decision making - A Response, Omega 22 (1994) 205-206.

[14] C. Tofallis, Improving discernment in DEA using profiling, Omega 24 (1996) 361-364.

[15] J. Zhu, Super-efficiency and DEA sensitivity analysis. European Journal of Operational Research 129 (2001) 443-455.