Dealing with Desirable Inputs in Data Envelopment Analysis: A Slacks-based Measure Approach

Kaoru Tone
National Graduate Institute for Policy Studies, Tokyo, Japan

Email address: tone@grips.ac.jp

To cite this article:

Received: October 6, 2021; Accepted: October 21, 2021; Published: October 30, 2021

Abstract: In Data Envelopment Analysis (DEA) the situation of inputs vs. outputs is positioned as cause and effect. Effects include desirable (ordinary) outputs and undesirable outputs, e.g. pollutants. This situation is well studied and many applications have been published. In this paper, we introduce a new type of inputs, called Good (Desirable) Inputs. As explained in Introduction, we find several examples of such inputs, e.g. Electric car, Women in office and Test takers of vaccine. We handle this by means of SBM (Slacks-based Measure). Usually, efficiency values of DEA models are in the range $0,1$, while in this model a negative efficiency value may be assigned to inefficient DMUs (decision making units). This is caused by shortages of Good Input values. As an example, we refer to “Women’s Rights Movements” in a country where women’s right is not fully guaranteed. Suppose local governments where men and women are serving as officers. They are inputs to office, while Women are Desirable input and Men are Ordinary input. As outputs, we assume Service as Ordinary output and Claim as Undesirable output. Several extensions of this model are introduced. (a) Variable returns to scale, (b) Weight restrictions, (c) Super-efficiency issue and (d) SBM_Max model.

Keywords: DEA, SBM, Desirable Inputs

1. Introduction

Generally speaking, in DEA, inputs (I) indicate input resources and the smaller the better, while outputs (O) correspond to productions induced by inputs (I), which are the larger the better. Among outputs, there exist undesirable outputs (OBad) incidental to outputs, e.g. CO$_2$, which are the smaller the better. Tone [8] formulated these situations in the framework of SBM (slacks-based measure) and this model is utilized all over the world. See Bai et al. [3], Ding et al. [4], Chen et al. [2], Liu et al. [5], and Wu et al. [14] for recent applications.

In this paper, a new input style, called Desirable inputs (IGood), is proposed, which are the larger the better. We show four potential examples of (IGood).

(a) In accordance with the increasing concern on environmental pollution, spread of Electric cars is worldwide required. We compare the energy consumption issue of countries. We consider “Number of Electric Cars” as a Good Input (IGood). Other IO items are (I) Total consumption of Energy, (O) GDP and (OBad) Pollutants (e.g. CO$_2$). DMUs are Country 1 to Country N.

(b) Consider the case that there are many new projects in a company and the manager wishes to evaluate the efficiency of projects. For this purpose, a certain number of evaluators is assigned to each project and they report Cost (Input), Return (Output), Risk (Undesirable Output) for the assigned project. In this case, we consider the number of evaluators is a Desirable input (IGood) and the larger the better.

(c) “Women’s Rights Movement” is a matter of urgency in a country where women’s right is not fully guaranteed. In such a country, suppose local governments where men and women are serving as officers. They are inputs to office, while women are (IGood) and men are (I). As outputs, we assume (O) Service and (OBad) Claims.

(d) Suppose that, inspired by COVID-19, many medical enterprises are eager to develop vaccine. In this case, (IGood) Number of test takers, (I) Cost, (O) Recovery, (OBad) Failure. As DMUs, we assume potential Vaccine 1 to Vaccine N.

We reasoned that such situations occur in many fields of enterprise.
We formulate this situation in the framework of SBM.

2. An SBM Model with Desirable Inputs

Suppose that there are \( n \) DMUs (decision making units) each having four factors: inputs, good (desirable) inputs, outputs, and bad (undesirable) outputs, as represented by four vectors \( \mathbf{x} \in \mathbb{R}^{m_1}, x^g \in \mathbb{R}^{m_2}, y \in \mathbb{R}^{s_1}, y^b \in \mathbb{R}^{s_2} \), respectively. We define the matrices \( X, X^g, Y \) and \( Y^b \) as follows.

\[ X = [x_1, \ldots, x_n] \in \mathbb{R}^{m_1 \times n}, x^g = [x^g_1, \ldots, x^g_n] \in \mathbb{R}^{m_2 \times n}, Y = [y_1, \ldots, y_n] \in \mathbb{R}^{s_1 \times n} \text{ and } Y^b = [y^b_1, \ldots, y^b_n] \in \mathbb{R}^{s_2 \times n}. \]

We assume \( X > 0, X^g > 0, Y > 0 \) and \( Y^b > 0 \).

The production possibility set \( (P) \) is defined by

\[
P = \{(x, x^g, y, y^b) | x \geq X \lambda, x^g \leq X^g \lambda, y \leq Y \lambda, y^b \geq Y^b \lambda, \lambda \geq 0\},
\]

where \( \lambda \in \mathbb{R}^n \) is the intensity vector. Notice that the above definition corresponds to the constant returns to scale technology. We discuss the other return to scale cases in Section 4.

Definition 1 (Efficient DMU) A DMU \( u \) \((x_u, x^g_u, y_u, y^b_u)\) is efficient in the presence of desirable inputs and undesirable outputs if there is no vector \((x, x^g, y, y^b) \in P\) such that \( x_0 \geq x, x^g_0 \leq x^g, y_0 \leq y \) and \( y^b_0 \geq y^b \) with at least one strict inequality.

In accordance with this definition, we modify the SBM in Tone [6] as follows.

\[
[SBM] \rho^* = \min \left\{ \frac{1 - \frac{1}{m_1 + m_2} \sum_{i=1}^{m_1} x^g_i}{1 + \frac{1}{s_1 + s_2} \sum_{g=1}^{s_1} x^g_i} \right\}
\]

Subject to \( x_0 = X \lambda + s^c \)

\[
x^g_0 = X^g \lambda - s^g - \]

\[
y_0 = Y \lambda - s \]

\[
y^b_0 = Y^b \lambda + s^b \]

\( s^c \geq 0, s^g \geq 0, s \geq 0, s^b \geq 0, \lambda \geq 0 \).

The vectors \( s^c \in \mathbb{R}^{m_1} \) and \( s^b \in \mathbb{R}^{s_2} \) correspond to excesses in inputs and bad outputs, respectively, while \( s^g \in \mathbb{R}^{m_2} \) represents shortages in good inputs and \( s \in \mathbb{R}^{s_1} \) expresses shortages in good outputs. The objective function (2) strictly decreases with respect to \( s^c (\forall i), s^g (\forall i), s^b (\forall r) \) and \( s (\forall r) \). Let an optimal solution of the above program be \((\lambda^*, s^c*, s^g*, s^b*, s^b*)\). Then, we have:

Theorem 1 The DM U0 is efficient in the presence of desirable inputs and undesirable outputs if and only if \( \rho^* = 1 \), i.e., \( s^c* = 0, s^g* = 0, s^b* = 0 \).

If the DMUs is inefficient, i.e., \( \rho^* < 1 \), it can be improved

\[
\rho^* = \tau^*, \lambda^* = \lambda / t^*, s^c* = s^c / t^*, s^g* = s^g / t^*, s^b* = s^b / t^*, \quad (8)\]

(See Tone [2001] for detail). The existence of \((\tau^*, \lambda^*, s^c*, s^g*, s^b*, s^b*)\) with \( t^* > 0 \) is guaranteed by [LP].

3. Why We Employ (IGood) and (I) as Inputs and (O) and (OBad) as Outputs

The situation of inputs vs. outputs is positioned as cause and effect. Use of (I) and (OBad) as inputs and (O) and (IGood) as outputs seems plausible, because the former is the less the better and the latter is the larger the better. However, this standpoint is just a pretense and neglects the cause and effect relationship. Our objective function (2) expresses orthodoxy cause and effect relationship.

4. An Illustrative Example

Table 1 shows a fictional data on 14 cities in which inputs are Women and Men with resulting outputs Service and Claim. From the point of view of “Women’s Right Movements”, we assume Women as (IGood) while Men as an ordinary input. As outputs we take Service as an ordinary output while Claim as an Undesirable Output (OBad).
We solved this data using the program (2) to (6) under the constant returns to scale (CRS) assumption. Table 2 exhibits the efficiency score. It is remarkable that City 13 has a negative efficiency score (-0.3505). This indicates that City 13 has a large slack (59) in (IGood) Women compared with its current value (25). See Table 3. Hence, its numerator of the objective function (2) comes to be negative. City 13 is the worst in efficiency.

### Table 2. Efficiency Score.

<table>
<thead>
<tr>
<th>DMU</th>
<th>Score</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>City1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>City2</td>
<td>0.5</td>
<td>6</td>
</tr>
<tr>
<td>City3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>City4</td>
<td>0.8571</td>
<td>4</td>
</tr>
<tr>
<td>City5</td>
<td>0.5625</td>
<td>5</td>
</tr>
<tr>
<td>City6</td>
<td>0.2727</td>
<td>11</td>
</tr>
<tr>
<td>City7</td>
<td>0.2483</td>
<td>12</td>
</tr>
<tr>
<td>City8</td>
<td>0.297</td>
<td>10</td>
</tr>
<tr>
<td>City9</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>City10</td>
<td>0.4324</td>
<td>8</td>
</tr>
<tr>
<td>City11</td>
<td>0.3926</td>
<td>9</td>
</tr>
<tr>
<td>City12</td>
<td>0.1446</td>
<td>13</td>
</tr>
<tr>
<td>City13</td>
<td>-0.3505</td>
<td>14</td>
</tr>
<tr>
<td>City14</td>
<td>0.4461</td>
<td>7</td>
</tr>
</tbody>
</table>

### Table 3. Slacks.

<table>
<thead>
<tr>
<th>DMU</th>
<th>Score</th>
<th>Rank</th>
<th>Slack</th>
<th>Slack</th>
</tr>
</thead>
<tbody>
<tr>
<td>(IGood)Women</td>
<td>(I)Men</td>
<td>(O)Service</td>
<td>(OBad)Claim</td>
<td></td>
</tr>
<tr>
<td>City1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>City2</td>
<td>0.5</td>
<td>6</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>City3</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>City4</td>
<td>0.8571</td>
<td>4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>City5</td>
<td>0.5625</td>
<td>10</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>City6</td>
<td>0.2727</td>
<td>11</td>
<td>35</td>
<td>0</td>
</tr>
<tr>
<td>City7</td>
<td>0.2483</td>
<td>12</td>
<td>40</td>
<td>33.333</td>
</tr>
<tr>
<td>City8</td>
<td>0.297</td>
<td>10</td>
<td>50</td>
<td>0</td>
</tr>
<tr>
<td>City9</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>City10</td>
<td>0.4324</td>
<td>8</td>
<td>30</td>
<td>266.667</td>
</tr>
<tr>
<td>City11</td>
<td>0.3926</td>
<td>9</td>
<td>35</td>
<td>400</td>
</tr>
<tr>
<td>City12</td>
<td>0.1446</td>
<td>13</td>
<td>90</td>
<td>100</td>
</tr>
<tr>
<td>City13</td>
<td>-0.3505</td>
<td>14</td>
<td>59</td>
<td>590</td>
</tr>
<tr>
<td>City14</td>
<td>0.4461</td>
<td>7</td>
<td>50</td>
<td>350</td>
</tr>
<tr>
<td>Average</td>
<td>0.4859</td>
<td>28.8571</td>
<td>124.2857</td>
<td>385.7143</td>
</tr>
<tr>
<td>Max</td>
<td>1</td>
<td>90</td>
<td>590</td>
<td>3400</td>
</tr>
<tr>
<td>Min</td>
<td>-0.3505</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>St Dev</td>
<td>0.3827</td>
<td>27.7318</td>
<td>195.4414</td>
<td>918.91</td>
</tr>
</tbody>
</table>

Table 4 exhibits projections to efficient status.
5. Extensions

This section discusses extensions to the following issues:

(a) The variable returns to scale issue

We can extend the model in (2) to the variable returns to scale by adding the constraint,

$$ e^\lambda = 1 $$

(b) Weight restrictions to (IGood) vs (I) and (O) vs (OBad)

If putting preference (or importance) on input/output items is required, we can impose weights to the objective function in (2) as follows:

$$ \rho^* = \min \frac{1}{1 + \sum_{r=1}^{s_2} w^r_f \frac{y^r_f}{y^r_f}} \left[ \left( \sum_{i=1}^{m} w^i_l \frac{x^i_l}{x^i_l} + \sum_{i=1}^{m} w^i_l \frac{x^i_l}{x^i_l} \right) \right] \frac{1}{1 + \sum_{r=1}^{s_2} w^r_f \frac{y^r_f}{y^r_f}} $$

where $$ w^i_l, w^i_l, w^r_f $$ and $$ w^r_f $$ are the weights to the input $$ i $$, the desirable input $$ i $$, the desirable output $$ r $$, and the undesirable output $$ r $$, respectively, and $$ \sum_{i=1}^{m} w^i_l + \sum_{i=1}^{m} w^i_l = m + m_2, w^i_l \geq 0 (\forall i), w^i_l \geq 0 (\forall i), \sum_{i=1}^{m} w^i_l + \sum_{i=1}^{m} w^i_l = m + m_2, w^i_l \geq 0 (\forall i), \sum_{i=1}^{m} w^i_l + \sum_{i=1}^{m} w^i_l = s_1 $$ and $$ S_2, w^r_f \geq 0 (\forall r), w^r_f \geq 0 (\forall r) $$.

(c) Super efficiency issue

For efficient DMUs, we can apply the super efficiency model developed in Tone [7] by adding (IGood) factors in its formulation. Table 5 reports the results.
(d) Application of SBM_Max model

For an inefficient DMU, the SBM_Max model attempts to find nearly closest reference point on the efficient frontiers so that slacks are minimized, while the scores are maximized. Inefficient DMUs can be improved to the efficient status with less input-reductions and less output-enlargement. See Appendix A for a brief introduction and Tone [9] for details. We added (IGood) factors to the formulation in Tone [10].
6. Conclusion

Probably, this is the first trial to incorporate desirable input (IGood) into DEA research. We believe this model has a reasonable position for complying with the demand of modern society. Especially, the outcome of SBM\_Max model is more practically applicable. Future research subjects include applications of this model to Network SBM (Tone and Tsutsui [11]), Dynamic SBM (Tone and Tsutsui [12]) and Dynamic and Network SBM (Tone and Tsutsui [13]).

Appendix

In this appendix, we introduce the non-oriented SBM\_Max model briefly.

Step 1. Solve SBM-Min

First, we solve the ordinary SBM (SBM-Min) model as represented by the program (2) for DMU \((x_o, y_o) (o = 1, \ldots, n)\). Let an optimal solution be \((\lambda^*, s^*, s^+)\).

Step 2. Define efficient DMUs

We define the set \(R^e\) of all efficient DMUs as

\[
R^e = \{ j \mid \rho^*_{ij} = 1, j = 1, \ldots, n \}. \tag{A1}
\]

We denote these efficient DMUs as \((x^e_{o1}, y^e_{o1}), (x^e_{o2}, y^e_{o2}), \ldots, (x^e_{oNeff}, y^e_{oNeff})\), where Neff is the number of efficient DMUs.

Step 3. Local reference set

For an inefficient DMU \((x_o, y_o)\), we define the local reference set \(R^e_{o}\), i.e., efficient DMUs set for DMU \((x_o, y_o)\), by (A2).

\[
R^e_{o} = \{ j \mid \rho^*_{oj} > 0, j = 1, \ldots, n \}. \tag{A2}
\]

Step 4. Pseudo-Max score

For each inefficient DMU, i.e., \(\rho^*_{o1} < 1\), we solve the following program.

---

<table>
<thead>
<tr>
<th>City7</th>
<th>City8</th>
<th>City9</th>
<th>City10</th>
<th>City11</th>
<th>City12</th>
<th>City13</th>
<th>City14</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.455696</td>
<td>0.661765</td>
<td>0.728155</td>
<td>0.705727</td>
<td>0.665584</td>
<td>0.409577</td>
<td>0.791557</td>
<td></td>
</tr>
<tr>
<td>0.13</td>
<td>0.40</td>
<td>0.50</td>
<td>0.55</td>
<td>0.60</td>
<td>0.25</td>
<td>0.70</td>
<td></td>
</tr>
<tr>
<td>56.25</td>
<td>52.5</td>
<td>45.0</td>
<td>50.0</td>
<td>55.0</td>
<td>60.0</td>
<td>70.0</td>
<td></td>
</tr>
</tbody>
</table>

---

Table 8. Continue.

<table>
<thead>
<tr>
<th>DMU</th>
<th>Score</th>
<th>Rank</th>
<th>(O)Service</th>
<th>Projection</th>
<th>Diff.(%)</th>
<th>(I)Men</th>
<th>Projection</th>
<th>Diff.(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>City1</td>
<td>1</td>
<td>1</td>
<td>1000</td>
<td>1000</td>
<td>0</td>
<td>20</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>City2</td>
<td>0.5</td>
<td>12</td>
<td>1000</td>
<td>1000</td>
<td>0</td>
<td>20</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>City3</td>
<td>1</td>
<td>1</td>
<td>1500</td>
<td>1500</td>
<td>0</td>
<td>10</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>City4</td>
<td>0.857143</td>
<td>4</td>
<td>1500</td>
<td>1500</td>
<td>0</td>
<td>15</td>
<td>10</td>
<td>-33.333</td>
</tr>
<tr>
<td>City5</td>
<td>0.825</td>
<td>6</td>
<td>1500</td>
<td>1833.333</td>
<td>22.222</td>
<td>30</td>
<td>30</td>
<td>0</td>
</tr>
<tr>
<td>City6</td>
<td>0.846262</td>
<td>5</td>
<td>2500</td>
<td>2533.333</td>
<td>1.333</td>
<td>20</td>
<td>13</td>
<td>-35</td>
</tr>
<tr>
<td>City7</td>
<td>0.455696</td>
<td>13</td>
<td>3500</td>
<td>3500</td>
<td>0</td>
<td>40</td>
<td>21.25</td>
<td>-46.875</td>
</tr>
<tr>
<td>City8</td>
<td>0.661765</td>
<td>11</td>
<td>4000</td>
<td>4000</td>
<td>0</td>
<td>50</td>
<td>22.5</td>
<td>-55</td>
</tr>
<tr>
<td>City9</td>
<td>1</td>
<td>1</td>
<td>5000</td>
<td>5000</td>
<td>0</td>
<td>25</td>
<td>25</td>
<td>0</td>
</tr>
<tr>
<td>City10</td>
<td>0.728155</td>
<td>8</td>
<td>4000</td>
<td>5066.667</td>
<td>26.667</td>
<td>50</td>
<td>26</td>
<td>-48</td>
</tr>
<tr>
<td>City11</td>
<td>0.705727</td>
<td>9</td>
<td>4500</td>
<td>6111.111</td>
<td>35.802</td>
<td>55</td>
<td>50</td>
<td>-44.444</td>
</tr>
<tr>
<td>City12</td>
<td>0.665584</td>
<td>10</td>
<td>4100</td>
<td>6666.667</td>
<td>62.602</td>
<td>50</td>
<td>33.3333</td>
<td>-33.333</td>
</tr>
<tr>
<td>City13</td>
<td>0.409577</td>
<td>14</td>
<td>4200</td>
<td>4200</td>
<td>0</td>
<td>55</td>
<td>21</td>
<td>-61.818</td>
</tr>
<tr>
<td>City14</td>
<td>0.791557</td>
<td>7</td>
<td>5000</td>
<td>7258.333</td>
<td>45.167</td>
<td>40</td>
<td>37</td>
<td>-7.5</td>
</tr>
</tbody>
</table>

---

Table 8. Continue.
Let an optimal slacks be \((s^+, s^*)\). We solve the following program with variables \((\lambda, s^+, s^*)\).

\[
\begin{align*}
\text{[Pseudo-2]} & \quad \min & & -\frac{1}{m} \sum_{i=1}^{m} \frac{s_{io}}{x_{io}} - s^+ - s^* \\
& & & + \frac{1}{s} \sum_{r=1}^{s} \frac{s^*}{y_{ro}} \\
& \text{subject to} & & \sum_{j \in R^{eff}} x_j \lambda_j + s^+ = x_o \\
& & & \sum_{j \in R^{eff}} y_j \lambda_j - s^* = y_o \\
& & & s^+, s^+, \lambda \geq 0.
\end{align*}
\]

We define the set \(R_h\) by

\[
R_h = \{1, \ldots, h\} \quad (h = 1, \ldots, \text{Neff}). \tag{A8}
\]

**Step 5.2. Find slacks and max-score for the set \(R_h\)**

We evaluate the efficiency score of the inefficient DMU \((x_o, y_o)\) referring to the set \(R_h\) by solving the following program.

\[
\begin{align*}
\text{[Max-1]} & \quad \max & & -\frac{1}{m} \sum_{i=1}^{m} \frac{s_{io}}{x_{io}} \\
& & & + \frac{1}{s} \sum_{r=1}^{s} \frac{s^*}{y_{ro}} \\
& \text{subject to} & & \sum_{j \in R_h} x_j \lambda_j + s^+ = x_o \\
& & & \sum_{j \in R_h} y_j \lambda_j - s^* = y_o \\
& & & s^+, s^+, \lambda \geq 0.
\end{align*}
\]

If this program is infeasible, we define \(\rho_{oh}^* = 0\).

Otherwise, let an optimal slacks be \((s^+, s^+)\).

(a) If the optimal objective value is 1, i.e., \(s^* = 0\) and \(s^+ = 0\), we define \(\rho_{oh}^* = 0\). This indicates that DMU \((x_o, y_o)\) can be expressed as a non-negative combination of DMUs in \(R_h\) and hence, in view of \(\rho_{oh}^{\min} < 1\), it is inside the production possibility set.

(b) If the optimal objective value is less than 1, we again solve the following program with the variables \((\lambda, s^+, s^+)\).

\[
\begin{align*}
\text{[Max-2]} & \quad \min & & -\frac{1}{m} \sum_{i=1}^{m} \frac{s_{io}}{x_{io}} \\
& & & + \frac{1}{s} \sum_{r=1}^{s} \frac{s^*}{y_{ro}} \\
& \text{subject to} & & \sum_{j \in R_h} x_j \lambda_j + s^+ = x_o \\
& & & \sum_{j \in R_h} y_j \lambda_j - s^* = y_o \\
& & & s^+, s^+, \lambda \geq 0.
\end{align*}
\]

Let the optimal slacks be \((s^+, s^+)\). We define \(\rho_{oh}^*\) by

\[
\rho_{oh}^* = \frac{1}{m} \sum_{i=1}^{m} \frac{s_{io}^* + s_{io}^{**}}{x_{io}^*} \\
\frac{1}{s} \sum_{r=1}^{s} \frac{s_{ro}^* + s_{ro}^{**}}{y_{ro}^*}. \tag{A11}
\]

This distance is units-invariant.

**Step 5.1. Reorder the distance**

We renumber the efficient DMUs in the ascending order of \(d_h\), so that

\[
d_1 \leq d_2 \leq \ldots \leq d_{\text{Neff}}. \tag{A7}
\]
We assign $\rho_{o}^{*}$ as the max-score referring to the set $R_{o}$.  

**Step 5.3. SBM_Max and projection**

Finally, we define the max-score $\rho_{o}^{\text{max}}$ of inefficient DMU $(x_{o}, y_{o})$ by

$$[\text{SBM-Max}] \quad \rho_{o}^{\text{max}} = \max \{\rho_{o}^{\text{pseudomax}}, \rho_{o}^{*}, \rho_{o}^{\text{Neff}}\}.$$  \hspace{1cm} (A12)

We also hold the slacks $(s^{*}, s^{**})$ corresponding to the maximum $\rho_{o}^{\text{max}}$. The projection of DMU $(x_{o}, y_{o})$ onto efficient frontiers is given by

$$[\text{Projection}] \quad x_{o}^{*} = x_{o} - s^{*} - s^{**}, y_{o}^{*} = y_{o} + s^{*} + s^{**}.$$  \hspace{1cm} (A13)

The projected point $(x_{o}^{*}, y_{o}^{*})$ is efficient with respect to the efficient DMU set $R_{\text{eff}}$. However, it does not always satisfy Pareto-Koopmans efficiency condition. This model belongs to polynomial time as for the computational complexity.

**References**


