Ranking DMUs in the DEA context using super and cross efficiency

Byungho Jeong, Jae-won Ha
The Industrial and Information System Engineering
Chonbuk University
Chonju, Chonbuk, 561-756, Korea
e-mail: jeong@chonbuk.ac.kr, jwha@chonbuk.ac.kr

Chang-Soo Ok
Department of Industrial Engineering
Hongik University
Seoul, 121-791, Korea
E-mail: okcs@hongik.ac.kr

Abstract—This work proposes a revised cross evaluation matrix which can be utilized to obtain a full rank of DMUs. The revised matrix contains super-efficiency values for diagonal elements and cross-efficiency values for non-diagonal elements of the matrix. This matrix enables better the difference of efficiency or performance of DMUs than the original cross evaluation matrix.

Keywords—Data Envelopment Analysis, Super efficiency, Cross efficiency

I. INTRODUCTION

Data Envelopment Analysis (DEA), developed by Charnes et al.[1, 2], is a linear programming procedure for a frontier analysis of decision making units with multiple inputs and outputs. Generally, the DEA model classifies DMUs (Decision Making Units) into two groups which are efficient and inefficient. While the DMUs in inefficient group have efficiency measure of less than one, the ones from the efficient group form the Pareto frontier and have same efficiency measure of ‘1’. Thus, this approach is only able to provide a dichotomized classification of DMUs. It is clear that a complete ranking list is more useful to evaluate DMUs. Therefore, considerable research efforts have advanced the DEA procedure to rank DMUs finely.

Cook and Kress proposed a DEA structure to deal with an ordinal preference ranking[3]. Cooper and Tone argued that original efficiency value would be determined from different facets. In other words, efficiency values are being derived from comparisons involving performances of different sets of DMUs[2]. However, both model only ranked inefficient DMUs according to a scalar measure of inefficiency in DEA based on slack variables. Torgersen et al. developed a complete ranking method by measuring importance of DMUs as a benchmark for inefficient DMUs[4]. Anderson and Peterson suggested a super-efficiency model to rank efficient DMUs[5]. The approach enables an extreme efficient unit $k$ to achieve an efficiency score greater than one by removing the $k^{th}$ constraint limiting the score under 0-1 in the primal formulation.

While, a cross efficiency approach was firstly introduced by Sexton et al. and dealt with the subject of ranking in DEA[6]. Doyle and Green recommended the cross evaluation matrix for ranking all DMUs[7]. Once the best set of weights has been chosen for a particular DMU, the cross-efficiency for other DMUs can be calculated using the weight set. This procedure is repeated for all DMUs to get efficiency and cross-efficiency values of all DMUs. This approach provided an intuitive interpretation of cross efficiency and simple efficiency as peer appraisal and self appraisal respectively. The cross evaluation matrix consists of simple efficiency values(diagonal elements) and cross-efficiency values(non-diagonal elements). This matrix can be used to rank all DMUs like a pairwise comparison matrix.

Sinuany-Stern et al. suggested an analytical hierarchical process (AHP/DEA) ranking model[8]. In the first stage, a DEA model is applied for every pair of DMUs, only two DMUs at a time, ignoring all others. Using the results of the first stage, a pairwise comparison matrix can be created and this matrix can be used to provide a full scale ranking of all DMUs. However, the first stage can result in a matrix of many 1’s because each DEA model is run for only two DMUs. That would make difficult fully ranking of all DMUs.

In this paper we suggest a revised cross evaluation matrix to get a full rank of DMUs. The revised matrix is made up of super-efficiency values for diagonal elements and cross-efficiency values for non-diagonal elements of the matrix. This matrix would reflect better the difference of efficiency or performance of DMUs than the original cross evaluation matrix. To illustrate the proposed ranking method, we present a case study which evaluates 22 project groups financially supported by the Korean government research program.

II. A MODEL FOR SUPER-EFFICIENCY AND CROSS-EFFICIENCY

A weakness of the original DEA model is that a considerable number of DMUs typically is characterized as efficient and these efficient DMUs have same efficiency value of ‘1’[5]. Thus, the model is not able to discriminate efficiency or performance among efficient DMUs and provide any preferential difference to efficient DMUs. Also, the cross evaluation matrix has a similar drawback[7]. Simple efficiency values, constituting diagonal elements of the cross evaluation matrix, do not reflect on the difference of efficiency or performance of efficient DMUs. To tackle
this problem, this work proposes a revised cross evaluation matrix by replacing simple efficiency with super efficiency in the diagonal elements of the matrix. The proposed method in this work combines super and cross efficiency model. The proposed approach can be described as (1). The mathematical model derives super efficiency and cross efficiency with a two-Phase procedure.

\[
\text{max } \sum_{i} u_{i} y_{i0} \quad \text{(goal 1)}
\]

\[
\text{max or min } \sum_{i} u_{i} y_{i} \quad \text{(goal 2)}
\]

\[
\begin{align*}
\text{s.t.} & \quad \sum_{q} v_{q} x_{q0} = 1 \\
& \quad -\sum_{q} v_{q} x_{qj} + \sum_{i} u_{i} y_{ij} \leq 0, \quad \forall j, j \neq 0 \\
& \quad v_{q}, u_{i} \geq 0
\end{align*}
\]

where \(x_{i} = [x_{ij}]\) is an input vector in which \(x_{ij}\) is the \(q^{th}\) input value of DMU \(j\). \(y_{i} = [y_{ij}]\) is an output vector in which \(y_{ij}\) denotes the \(j^{th}\) output value of each DMU \(j\). \(v_{q}\) is a weight of the \(q^{th}\) input and \(u_{i}\) is a weight of the \(i^{th}\) output variable for DMU \(0\).

By excluding the DMU under evaluation from the constraint set, the first objective function for obtaining the super-efficiency value of the DMU, \(i\), is allowed to have larger value than ‘1’. Obtaining this super-efficiency for DMU \(0\) can be thought of as another process of self-appraisal. The super-efficiency can make up for the inaccuracy of differences between DMUs than simple efficiency used in [7]. The second objective function is to find cross efficiency values of other DMUs. Let DMU \(0\) choose its own weights \(v_{q}\) and \(u_{i}\) respectively for input and output of super-efficiency. The cross-efficiency of other DMUs can be calculated simply using these weights that \(0\) has chosen if the optimum solution of the super-efficiency model for DMU \(0\) is unique. In the case of multiple solutions, DMU \(0\)’s evaluation of the other DMUs may depend on which of the alternative solutions to the LP solution process find first. Thus, a secondary objective function is introduced to resolve this ambiguity. Some choice of weights may lead to a lower(or higher) cross-efficiency for any other DMU being judged on DMU \(0\)’s weights. This leads to the possibility that there might be a particular choice of weights which not only obtains the maximum super-efficiency for DMU \(0\) as primary goal, but as a secondary goal, minimizes(or maximizes) the other DMU’s cross-efficiency in some way.

Now, we can construct a revised cross evaluation matrix for ranking DMUs using super-efficiency values and cross-efficiency values obtained by executing the above model for each DMU. Figure 1 shows a revised cross evaluation matrix constructed by replacing simple efficiency of the diagonal elements with super-efficiency. By using the super-efficiency in the diagonal elements instead of simple efficiency, the revised matrix can explain the differences of performance between efficient DMUs more precisely than the existing cross evaluation matrix. Therefore, we can get a more accurate ranking of DMUs from this revised matrix.

### III. ALGORITHM DETAIL

The model described in Section II consists of two phases. The Phase I provides super-efficiency for DMUs. This phase can be considered a self appraisal process for each DMU. Phase II finds cross-efficiencies, i.e., peer appraisal, in the case that the model of Phase I has multiple optimal solutions.

#### A. Phase I: Maximize goal 1

This phase can be considered a self appraisal process for each DMU \(i\).

\[
\text{max } \sum_{i} u_{i} y_{i0}
\]

\[
\begin{align*}
\text{s.t.} & \quad \sum_{q} v_{q} x_{q0} = 1 \\
& \quad -\sum_{q} v_{q} x_{qj} + \sum_{i} u_{i} y_{ij} \leq 0, \quad \forall j, j \neq 0 \\
& \quad v_{q}, u_{i} \geq 0
\end{align*}
\]

Let the optimal solution and optimal objective value \((v_{q}^{*}, u_{i}^{*})\) and \(\theta^{*} = \sum u_{i}^{*} y_{i0}\) respectively. If the optimal solution \((v_{q}^{*}, u_{i}^{*})\) is unique, the algorithm determines cross efficiencies of other \((N-1)\) DMUs by Equation (3) and stops. Otherwise, go to Phase II to find cross efficiencies.

\[
CE_{0h} = \sum_{i} u_{i}^{*} y_{ih} / \sum_{q} v_{q}^{*} x_{qh}
\]

#### B. Phase II: Maximize or Minimize goal 2

Firstly, the optimal solution of Phase I is added as a constraint to a new LP model for Phase II

\[
\text{max or min } \sum_{i} u_{i} y_{i}
\]

\[
\begin{align*}
\text{s.t.} & \quad \sum_{q} v_{q} x_{q0} = 1 \\
& \quad -\sum_{q} v_{q} x_{qj} + \sum_{i} u_{i} y_{ij} \leq 0, \quad \forall j, j \neq 0 \\
& \quad v_{q}, u_{i} \geq 0
\end{align*}
\]
\[- \sum q v^*_q x^*_q + \sum i u^*_i y^*_i \leq 0, \quad \forall j, j \neq 0 \quad (4)\]

For the optimal solution \((v^*_q, u^*_i)\) of Phase II, the cross-efficiency of DMU \(h\) is calculated by Equation (3). This cross efficiency is an efficiency evaluated by DMU 0’s weight, that is, in viewpoint of DMU 0. To construct a \(N \times N\) revised cross evaluation matrix for \(N\) DMUs using the above model, it requires to run a LP model at most \(N^2\) times in the case that all DMUs have multiple optimum solutions for the super-efficiency model.

IV. A NUMERICAL EXAMPLE

A. Data Set

To validate our approach a numerical analysis is provided in this chapter with a real application. This data set includes 23 projects funded by Korean government. The 21st century frontier R&D program is a long-term national program of Korea, for selective and intensive development of strategic technologies to enhance national scientific competitiveness to the level of advanced countries by 2010. The objective of the program is to develop the national economy through improving national competitiveness, public welfare, quality of life to the level of advanced countries and creating new industries through the development of future technologies. Table 1 summarizes the number of subprojects in each project center available in the data set which was used in DEA analysis of this paper. The number of subprojects supported by these 22 centers is totally 2,670 subprojects. 652 subprojects among them had or have been supported for one year and 315 subprojects for two years, 390 subprojects for three years, and 51 subprojects for 4 years.

The data set includes input and output factors which are related with the performance of the subprojects. The input variables are R&D investment(Korean million won), the number of involved researchers classified by academic degree such as doctoral(Ph.D), master(MSc), and bachelor (BSc). While, the output variables include the number of journal papers published, the number of patent, amount of royalty income(Korean million won), and human resource development. The number of papers was counted in the number of papers published in SCI indexed journal and non SCI indexed journal separately. The number of patents was also gathered separately in the number of applied patents and registered patents. The human resource development was counted by the number of earned Ph.D, MSc, and BSc degrees who were trained or educated in the process of these subprojects.

B. Numerical Results
For the experiments, we built a program to create LP (Linear Programming) Problems in C programming language while solving the LP problems using LINDO.

We have generated and solved 23 LP models to obtain super and cross efficiency for 23 DMUs. In this experiment, all models have a single optimal solution in the phase I. It seems because those are super efficiency models simply by Equation (3). Table 2 shows the cross evaluation matrix for the given data set. This matrix can be used to get priority values of 23 DMUs and to get a final rank.

V. CONCLUSIONS

In this work, we proposed a modified cross evaluation matrix to acquire a full rank of DMUs. The matrix created by the proposed approach consists of super-efficiency values for diagonal elements and cross-efficiency values for non-diagonal elements of the matrix. With this matrix we are able to differentiate DMUs precisely than the original cross evaluation matrix.

The relationship between input variables and output variables of each subproject is not clear in yearly basis. It may not be right that the number of journal papers and patents is considered as the output of the same year in which the R&D resources such as R&D investment and researchers were input. These output measures seem to be relatively long term measures. However, the temporal distance existing between the researches carried out and the outcome produced by the research makes it difficult to capture a formalized assessment of the long term effect of the research. This limitation should be considered in our future work.

ACKNOWLEDGMENT

This work was supported in part by 2009 Hongik University Research Fund.

REFERENCES


Table 2. A cross evaluation matrix with the example data set

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>...</th>
<th>23</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.816</td>
<td>0.569</td>
<td>0.928</td>
<td>0.172</td>
<td>0.341</td>
<td>1.000</td>
<td>0.785</td>
<td>0.649</td>
<td>1.000</td>
<td>0.770</td>
<td>0.792</td>
<td>...</td>
<td>0.351</td>
</tr>
<tr>
<td>2</td>
<td>0.492</td>
<td>0.838</td>
<td>0.824</td>
<td>0.213</td>
<td>0.248</td>
<td>0.982</td>
<td>0.481</td>
<td>0.591</td>
<td>0.684</td>
<td>0.930</td>
<td>0.841</td>
<td>...</td>
<td>0.342</td>
</tr>
<tr>
<td>3</td>
<td>0.608</td>
<td>0.330</td>
<td>1.318</td>
<td>0.155</td>
<td>0.237</td>
<td>0.949</td>
<td>0.534</td>
<td>0.690</td>
<td>1.000</td>
<td>0.336</td>
<td>0.901</td>
<td>...</td>
<td>0.241</td>
</tr>
<tr>
<td>4</td>
<td>0.264</td>
<td>0.370</td>
<td>0.404</td>
<td>0.292</td>
<td>0.331</td>
<td>1.000</td>
<td>0.224</td>
<td>0.252</td>
<td>0.793</td>
<td>0.404</td>
<td>0.418</td>
<td>0.671</td>
<td>...</td>
</tr>
<tr>
<td>5</td>
<td>0.304</td>
<td>0.431</td>
<td>0.291</td>
<td>0.192</td>
<td>0.407</td>
<td>1.000</td>
<td>0.525</td>
<td>0.300</td>
<td>0.787</td>
<td>0.452</td>
<td>0.349</td>
<td>0.475</td>
<td>...</td>
</tr>
<tr>
<td>6</td>
<td>0.397</td>
<td>0.546</td>
<td>0.725</td>
<td>0.069</td>
<td>0.282</td>
<td>2.176</td>
<td>0.627</td>
<td>0.394</td>
<td>1.000</td>
<td>0.650</td>
<td>0.727</td>
<td>0.762</td>
<td>...</td>
</tr>
<tr>
<td>7</td>
<td>0.745</td>
<td>0.457</td>
<td>0.792</td>
<td>0.146</td>
<td>0.354</td>
<td>1.000</td>
<td>0.837</td>
<td>0.632</td>
<td>1.000</td>
<td>0.725</td>
<td>0.671</td>
<td>0.910</td>
<td>...</td>
</tr>
<tr>
<td>8</td>
<td>0.647</td>
<td>0.339</td>
<td>1.000</td>
<td>0.178</td>
<td>0.283</td>
<td>0.801</td>
<td>0.472</td>
<td>0.754</td>
<td>1.000</td>
<td>0.479</td>
<td>0.506</td>
<td>0.897</td>
<td>...</td>
</tr>
<tr>
<td>9</td>
<td>0.702</td>
<td>0.524</td>
<td>1.000</td>
<td>0.196</td>
<td>0.311</td>
<td>1.000</td>
<td>0.783</td>
<td>0.639</td>
<td>1.459</td>
<td>0.667</td>
<td>0.760</td>
<td>0.890</td>
<td>...</td>
</tr>
<tr>
<td>10</td>
<td>0.585</td>
<td>0.602</td>
<td>0.392</td>
<td>0.176</td>
<td>0.261</td>
<td>0.890</td>
<td>0.580</td>
<td>0.483</td>
<td>1.000</td>
<td>1.072</td>
<td>0.462</td>
<td>1.000</td>
<td>...</td>
</tr>
<tr>
<td>11</td>
<td>0.492</td>
<td>0.694</td>
<td>1.000</td>
<td>0.207</td>
<td>0.268</td>
<td>1.000</td>
<td>0.405</td>
<td>0.590</td>
<td>0.526</td>
<td>0.640</td>
<td>0.928</td>
<td>0.914</td>
<td>...</td>
</tr>
<tr>
<td>12</td>
<td>0.372</td>
<td>0.214</td>
<td>0.639</td>
<td>0.224</td>
<td>0.224</td>
<td>1.000</td>
<td>0.332</td>
<td>0.597</td>
<td>1.000</td>
<td>0.671</td>
<td>0.386</td>
<td>1.202</td>
<td>...</td>
</tr>
</tbody>
</table>
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ...
| 23 | 0.441 | 0.501 | 0.298 | 0.185 | 0.378 | 1.000 | 0.541 | 0.336 | 0.804 | 0.593 | 0.432 | 0.607 | ... | 0.769 |