# RANKING ALTERNATIVES USING DATA ENVELOPMENT ANALYSIS: A DUAL APPROACH

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF SOCIAL SCIENCES

OF

IZMIR UNIVERSITY OF ECONOMICS

BY

ONUR CAN YILMAZ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

THE GRADUATE SCHOOL OF SOCIAL SCIENCES

JUNE 2014

Approval of the Graduate School of Social Sciences

Prof. Dr. Cengiz Erol
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree for the degree of Master of Science.

~~Assoc.~~ Prof. Dr. Burcu Özçam Adıvar
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Asst. Prof. Dr. Özgür Özpeynirci
Supervisor

Examining Committee Members

Assoc. Prof. Dr. Burcu Özçam Adıvar

Asst. Prof. Dr. Ahmet Camcı

Asst. Prof. Dr. Özgür Özpeynirci

# ABSTRACT

RANKING ALTERNATIVES USING DATA ENVELOPMENT ANALYSIS:
A DUAL APPROACH

Yılmaz, Onur Can

M.S, Department of Logistics Management

Supervisor : Asst. Prof. Dr. Özgür ÖZPEYNİRCİ

June 2014, 69 pages

This thesis addresses the problem of ranking alternatives evaluated with multiple criteria. Köksalan and Tuncer (2009) propose a mathematical programming approach based on Data Envelopment Analysis (DEA). Their approach requires binary and continuous decision variables and removes alternatives until the alternative under consideration becomes efficient. In this study, we propose a dual DEA based approach that myopically removes alternatives, but does not require integer or binary decision variables. We test the performance of our method and its competitor on randomly generated instances. We show that our method is significantly faster and obtains nearly the same rankings. Finally, we use both methods to rank 160 countries based on Logistics Performance Index data.

Keywords : ranking alternatives, data envelopment analysis, dual approach

# ÖZ

VERİ ZARFLAMA ANALİZİ KULLANARAK ALTERNATİFLERİ SIRALAMA: DUAL YAKLAŞIM

Yılmaz, Onur Can

Yüksek Lisans, Lojistik Yönetimi Bölümü

Tez Yöneticisi : Yrd. Doç. Dr. Özgür ÖZPEYNİRCİ

June 2014, 69 pages

Bu tez çalışması, çok kriterli alternatiflerin sıralanması üzerinedir. Köksalan ve Tuncer (2009) tarafından veri zarflama analizi tabanlı bir matematiksel programlama modeli önerilmiştir. Önerilen model ikili ve sürekli karar değişkenleri gerektirmektedir ve değerlendirilen alternatif etkin oluncaya kadar diğer alternatifleri göz ardı eder. Bu çalışmada, alternatifleri miyopik olarak göz ardı eden ancak tamsayılı veya ikili karar değişkenlerine ihtiyaç duymayan eşlenik (dual) veri zarflama analizi tabanlı bir yaklaşım önerilmiştir. Önerdiğimiz metot ile rakibinin performansı rassal üretilmiş örnekler üzerinde test edilmiştir. Önerdiğimiz metodun daha hızlı olduğu ve seçeneklerin sıralanmasında diğer yönteme yakın sonuçlar verdiği gösterilmiştir. Son olarak, her iki yöntem 160 ülkeyi Lojistik Performans İndeksi verisi ile sıralamak için kullanılmıştır.

Anahtar Kelimeler: alternatifleri sıralama, veri zarflama analizi, eşlenik yaklaşım

***to my family…***

# *ACKNOWLEDGEMENTS*

*I express my gratitude to my supervisor Assist. Prof. Dr. Özgür Özpeynirci who has guided, encouraged and supported me with his valuable opinion in every step occurrence of this academic study.*

*I thank the other academic member in thesis jury because of their valuable contribution, support and evaluation.*

*I thank a lot to Logistics Management Departments' valuable academic members and assistants who have not withheld their support since the day I was registered in Logistics Management Master Program.*

*And I thank a lot to Social Science Institute Directorship's personnel who has provided me with necessary convenience during master program.*

# TABLE OF CONTENTS

## LIST OF FIGURES

# LIST OF TABLES

## LIST OF ALGORTIHMS

# LIST OF ABBREVIATIONS

| | | |
|---|---|---|
| AES | : | Area of the Efficiency Score |
| CCR | : | Charnes Cooper Rhodes |
| DEA | : | Data Envelopment Analysis |
| DM | : | Decision Maker |
| DMU | : | Decision Making Units |
| KT | : | Koksalan Tuncer (method) |
| LPI | : | Logistics Performance Index |
| MCDM | : | Multi - Criteria Decision Making |
| OY | : | Özpeynirci Yılmaz (method) |

# CHAPTER 1

# INTRODUCTION

The ranking alternatives (or units) evaluated with multiple criteria is a problem faced by the decision makers (DM) in real life applications. The existence of multiple criteria not only affects the decision making process, but also increases the complexity of the ranking problem. There are several methodologies and techniques available in the literature for ranking alternatives in the presence of multiple criteria.

Data Envelopment Analysis (DEA) is a technique that has been used widely in the recent years for evaluating the efficiency of alternatives and benchmarking between independent decision making units. In this study, we propose a model that ranks the alternatives based on the data envelopment analysis.

Researchers have started to categorize units with their own efficiency values through DEA. The DEA is a method which receives a growing interest in the problem area of evaluating the so called "Decision Making Units" (DMUs) which convert multiple inputs into multiple outputs.

The DEA method enables the researcher to rank performance/efficiency benchmarking of DMUs with multiple inputs and outputs. DEA considers each DMU separately and finds the maximum efficiency value of the DMU under consideration by comparing it with the other DMUs.

Köksalan and Tuncer (2009) propose a DEA based method for ranking alternatives. Their method considers each DMU one by one. For a given DMU, it solves a DEA model with additional binary decision variables. If the DMU under consideration is not efficient, then they allow the model to remove one or more DMUs from the DMU list. This procedure is complete when the DMU under consideration becomes efficient. We propose a dual approach for ranking alternatives. Our approach also considers removing DMUs until the DMU under consideration becomes efficient. However, our approach does not require any binary or integer decision variables.

After this introductory chapter, we present background information and review DEA literature in the next chapter. In Chapter 3, we mention a DEA based approach to ranking multiple criteria alternatives proposed by Köksalan and Tuncer (2009). In Chapter 4, we introduce our dual approach to ranking multi-criteria alternatives. In Chapter 5, we test the performance of Köksalan and Tuncer's (2009) method and our method on randomly generated instances. This chapter also includes a comparison of the methods on a data set obtained from Logistic Performance Index of 160 countries. We conclude the thesis and discuss further research directions in the final chapter.

# CHAPTER 2

# LITERATURE REVIEW

In this part, we talk about the Data Envelopment Analysis literature and ranking methods.

## 2.1 Background Information

Data Envelopment Analysis (DEA) was known as "program follow through" in the early years. DEA was originated by Farrell in 1957 and it was called "The measurement of productive efficiency". The point of origin of DEA is relevant to productivity which was defined in his classical paper by Farrell (1957) as,

"*The problem of measuring the productive efficiency of an industry is important to both the economic theorist and the economic theorist maker. If the theoretical arguments as to the relative efficiency of different economic systems are to be subjected to empirical testing, it is essential to be able to make some actual measurements of efficiency. Equally, if economic planning is to concern itself with particular industries, it is important to know how far a given industry can be expected to increase its output by simple increasing its efficiency, without absorbing further resources.*"

The focus is to evaluate the performance of DMUs while consuming some varying amount of its inputs for producing some varying amount of its outputs. Charnes (1978) uses linear programming models for assessing the productivity of DMUs which have multiple inputs and outputs. The initial framework was developed by

Charnes (1978), and implemented and defined as Data Envelopment Analysis. Charnes (1978) described DEA as a;

*"Mathematical programming model applied to observational data which provides a new way of obtaining empirical estimates of relations - such as the production functions and/or efficient production possibility surfaces that are cornerstone of modern economics."*

Many theoretical extensions are developed based on the DEA method, which is designed a method to measure the efficiency of similar Decision Making Units using the ratio of its weighted outputs to its weighted inputs that provides for ranking/sorting alternatives to Decision Makers (DM).

The scope of DMU is wide, flexible and generic. DMUs may represent several decision making units, varying from countries to cities, from universities to schools, from firms to academic departments, from agriculture to the industry, from restaurants to retail stores, besides the real life applications for modeling operational processes for the performance evaluation. This wide concept makes DEA a very active research tool used for efficiency and productivity analysis.

Besides the first model by Charnes, Cooper and Rhodes (1978), which is called the CCR model, there are several related DEA models in the literature. One can see Cooper, Seiford, Tone (2000) for different DEA models including CCR, Additive DEA models and Slack-Based Measurement models. Seiford and Thrall (1990), Li and Reeves (1999), Cooper and Seiford (2004), Ahn, Charnes and Cooper (2006), Cook and Seiford (2009) and also provide a review on CCR models.

## 2.2 Data Envelopment Analysis

There are several statistical methods for ranking alternatives in the literature. In this section, we focus on the main basic DEA models. The classical method is shown below, that assigns an efficiency score to each DMU between 0 (worst) and 1 (best). If the score is equal to 1, DMU is said to be efficient, and inefficient if otherwise. The non-linear classical DEA is simply modeled as the ratio of value of the multiple outputs to that of multiple inputs. Before explaining this model, we look at the indices, parameters and the decision variables;

Indices:

*n* = the number of DMU's in the process, $k$=1,…,$n$

*m* = the number of inputs considered, $j$=1,…,$m$

*s* = the number of outputs considered, $i$=1,…,$s$

Parameters:

$Y_{ik}$ = *the value of output i for DMU k (i=1,...,s ; k = 1,...,n)*

$X_{jk}$ = the value of input *j* for DMU *k (j=1,...,m ; k = 1,...,n)*

Decision Variables:

$U_i$ = the weight for output *i (i = 1,....,s)*

$V_j$ = the weight for input *j (j = 1,...,m)*

Objective function :

$$\max \mathrm{Z} = \frac{\sum_{i=1}^{s} U_i \ Y_{io}}{\sum_{j=1}^{m} V_j \ X_{jo}}$$

This non-linear model's aim is to maximize the ratio of weighted of its inputs to weighted of its outputs for a DMU *0*. In the study, the DMU in consideration is referred to as DMU *0*.

subject to :

$$\frac{\sum_{i=1}^{s} U_i \ Y_{ik}}{\sum_{j=1}^{m} V_j \ X_{jk}} \leq 1 \ \forall k$$

Obviously, without constraints, the objective function value is unbounded, so the solution will be infeasible. With these constraints, the ratio of virtual output to that virtual input is expected to be smaller than 1 for each DMU, so the efficiency score of each DMU is equal to 1 or below. The efficiency ratio is ranged from 0 to 1, thus each unit will choose weights so as to maximize self efficiency.

for $j = 1, 2, \ldots, n,$

$$u_i, v_j \geq 0 \text{ for all } i = 1, \ldots, s \,;\, j = 1, \ldots, m$$

Below we show the linear version of the above model. This model is also known as the CCR (Charnes, Cooper and Rhodes) model.

objective function:

$$Max\ Z = \sum_{i=1}^{s} U_i \ Y_{io}$$

subject to:

$$\sum_{J=1}^{m} V_j \, X_{j0} = 1$$

$$\sum_{i=1}^{s} U_i \, Y_{ik} \leq \sum_{J=1}^{m} V_i \, X_{jk}$$

for $k = 1, \ldots\ldots, n$

$$U_i, V_j \geq 0, \ i = 1, \ldots, s; \ j = 1, \ldots, m.$$

This model determines the efficiency of each DMU and is widely used in the literature.

These models can be developed as input or output oriented. Output oriented puts the outputs values to consume minimum inputs. Input oriented puts the inputs values to produce maximum outputs. We will employ input oriented CCR model to evaluate the performance of DMUs because of example set.

We will explain the duality of the CCR model in the following part. The dual method's aim is to produce a hypothetical DMU consuming the minimum amount of inputs, while producing no less than the DMU under consideration. If the DMU is inefficient, then a combination of other DMUs can produce the same amount of output with less amount of inputs.

Before explaining this model, we first provide the indices, parameters and the decision variables;

Indices:

$n$ = the number of DMU's in the process,

$m$ = the number of inputs considered,

$s$ = the number of outputs considered,

Parameters:

$Y_{ik}$= the value of output i for DMU k *(i = 1,…,n) ; k = 1,…,n)*

$X_{jk}$ = the value of input j for DMU k *(j=1,…,m ; k = 1,…,n)*

Decision Variables:

$\lambda_k$ = is for constituting virtually DMU k, hybrid k of the DMUs *(i=1,…,n)*

$\Theta$ = rate weight $(0 \leq \Theta \leq 1)$

objective function:

min $\Theta$

subject to:

$$\sum_{k=1}^{n} \lambda_k X_{jk} \leq \Theta X_{j0} \quad \forall j$$

Input *j* of the hybrid DMU must be smaller than or equal to the input j of DMU under consideration.

$$\sum_{k=1}^{n} \lambda_k Y_{ik} \geq Y_{i0} \ \forall i$$

Output *i* of hybrid DMU must greater than or equal to output i of DMU under consideration.

$\lambda_k \geq 0$ for all *k*,

If a DMU is detected to be inefficient, the combination of the other efficient units can produce the same amount of output using fewer inputs. In this dual model, the hybrid of DMU is constituted as virtually using $\lambda_k$ and $\Theta$. We will utilize and develop duality of CCR model in our method in the forth section.

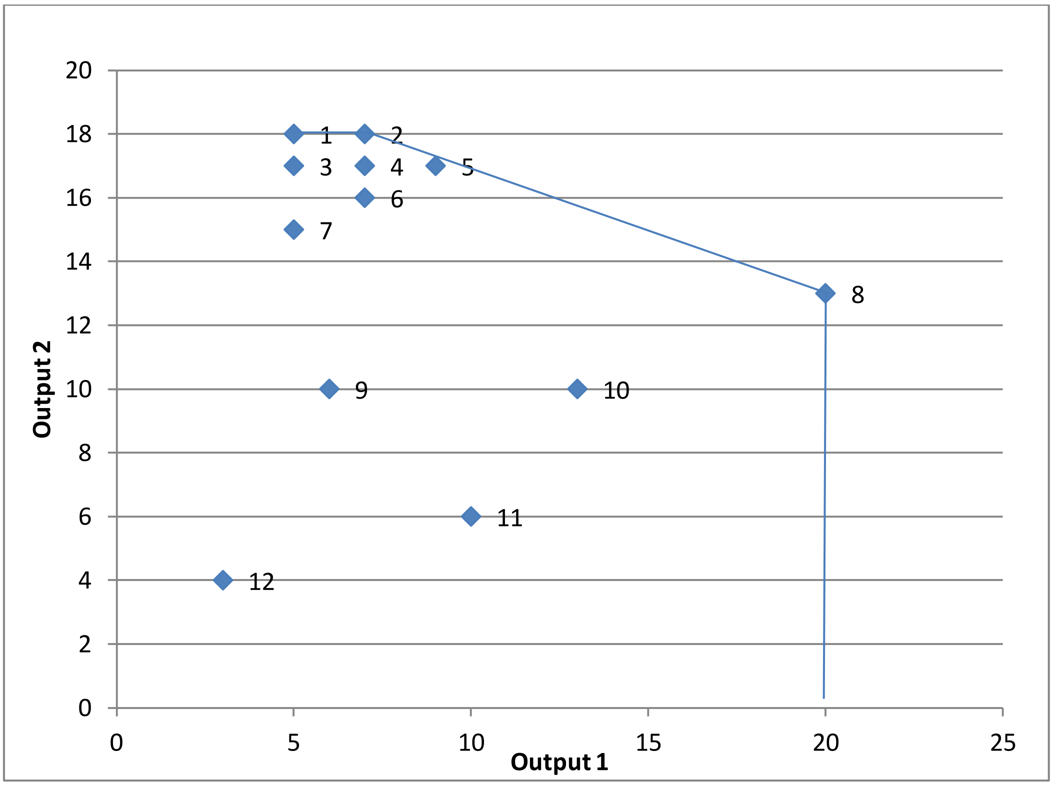
## 2.3 Ranking Methods

Several approaches (such as Cook and Kress, 1990; Sinuary-Stern, 1994; Landsdowne, 1996; Obata and Ishii, 2003; Foroughi and Tamiz, 2005; Büyükbaşaran, 2005; Köksalan et al., 2010) have adapted DEA models in practice. The determination of the in/efficient units is constituted with Pareto frontier. The classical DEA model (CCR) determines the in/efficient units for ranking alternatives. The DM can compute and determine the DMUs score for each unit's valuable or not. The efficiency or inefficiency score of the DMU's misguide the DM to eliminate alternatives according to the criteria in different situations. Köksalan and Tuncer (2009) illustrate the frontier and DMUs of a set graphically, using a small size data set example, which has 12 DMUs, one input and two outputs. The values of input and outputs are shown in Table 2.1.

Table 2.1 Inputs/outputs values for first example set
(Source: Köksalan and Tuncer, 2009)

| DMU # | INPUT | OUTPUT1 | OUTPUT2 |
|---|---|---|---|
| 1 | 1 | 5 | 18 |
| 2 | 1 | 7 | 18 |
| 3 | 1 | 5 | 17 |
| 4 | 1 | 7 | 17 |
| 5 | 1 | 9 | 17 |
| 6 | 1 | 7 | 16 |
| 7 | 1 | 5 | 15 |
| 8 | 1 | 20 | 13 |
| 9 | 1 | 6 | 10 |
| 10 | 1 | 13 | 10 |
| 11 | 1 | 10 | 6 |
| 12 | 1 | 3 | 4 |

Figure 2.1 Graphical representation of the first example
(Source : Köksalan and Tuncer, 2009)




Köksalan and Tuncer (2009) show the output 1 in the x-axis and output 2 in the y-axis. The frontier is formed by 1, 2 and 8 which has efficiency score of 1. We compute the efficiency scores of the DMU's using GAMS and show the results in Table 2.2.

Table 2.2 Efficiency scores for the first set.

| # of DMUs | Efficiency Score |
|---|---|
| 1 | 1 |
| 2 | 1 |
| 3 | 0.9444 |
| 4 | 0.9516 |
| 5 | 0.9888 |
| 6 | 0.9033 |
| 7 | 0.8333 |
| 8 | 1 |
| 9 | 0.5947 |
| 10 | 0.7249 |
| 11 | 0.5000 |
| 12 | 0.2490 |

Köksalan and Tuncer (2009) mention the ranking methods briefly. If we remove any DMUs from the set, the efficiency score of DMUs will change. If DMU 8 is removed from the set, the efficiency scores of several DMUs will increase, moreover some inefficient DMUs will become efficient.

Sinuary-Stern (1994) offer a measure to rank inefficient DMUs which counts the minimum number of inefficient units that should leave the set for each DMU in the set to become efficient. Köksalan and Tuncer (2009) propose a model uses the efficiency values for each number of DMUs (0 to $n$-1) leaving the reference set.

Below we present the model of Sinuary-Stern (1994)

objective function :

$$D_k = Min\ Z = \sum_{t=1}^{n} N_t$$

subject to :

$$\sum_{J=1}^{m} V_j X_{jk} = 1$$

$$\sum_{i=1}^{s} U_i Y_{ik} = 1$$

$$\sum_{i=1}^{s} U_i Y_{it} \le \sum_{J=1}^{m} V_j X_{jt} + N_t * M$$

for all $t \neq k$

$$N_t = \begin{cases} 1, if\ DMU\ t\ is\ deleted \\ 0, otherwise \end{cases}$$

$$U_{i,}\ V_j \ge 0,\ i = 1,\dots,s;\ j = 1,\dots,m$$

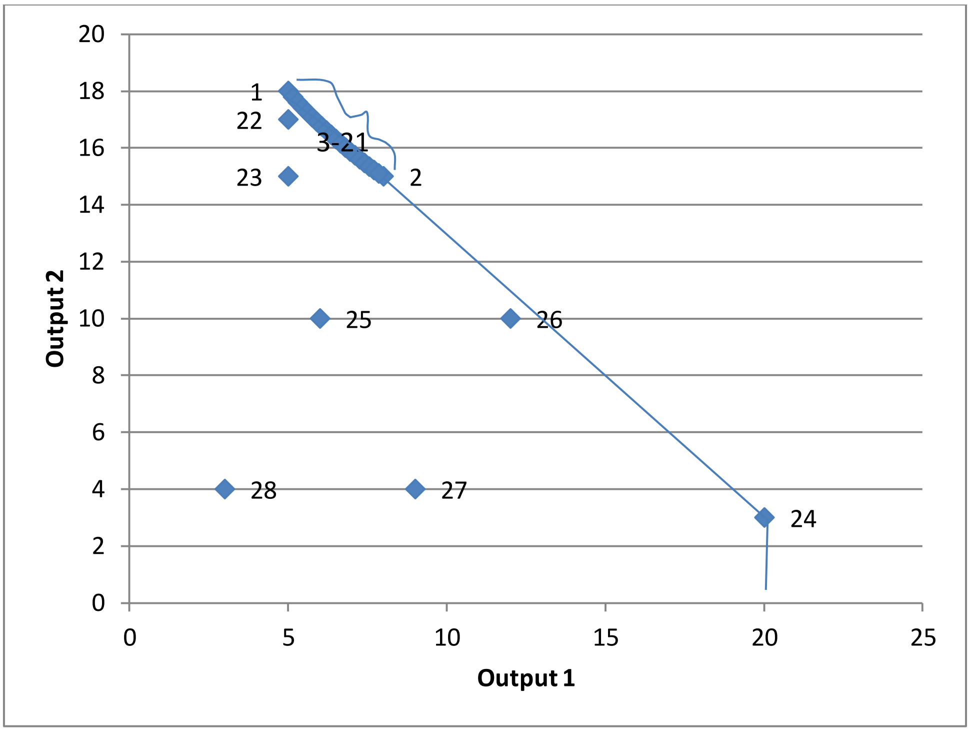
The objective function minimizes the numbers of DMUs that should be removed from the set so that the DMU under consideration (DMU k) becomes efficient. $M$ is a sufficiently large constant that allows the weighted outputs of DMU $t$ to be not larger than that of inputs if DMU $t$ is not removed. On the other hand, if DMU $t$ is removed from the set, the constraint becomes redundant.

Köksalan and Tuncer (2009) discuss that the model may have some problems in some special cases. They provide an example, of which the values of inputs/outputs are shown in Table 2.3, and the graphical representation in Figure 2.2.

Table 2.3 Input/Outputs Values, Efficiency scores and $D_k$ for the second example (Source: Köksalan and Tuncer, 2009).

| # of DMUs | INPUT | OUTPUT1 | OUTPUT2 | Efficiency Score | $D_k$ |
|---|---|---|---|---|---|
| 1 | 1 | 5.00 | 18.00 | 1.0000 | 0 |
| 2 | 1 | 8.00 | 15.00 | 1.0000 | 0 |
| 3 | 1 | 5.15 | 17.79 | 0.9974 | 1 |
| 4 | 1 | 5.30 | 17.60 | 0.9957 | 2 |
| 5 | 1 | 5.45 | 17.42 | 0.9943 | 3 |
| 6 | 1 | 5.60 | 17.25 | 0.9935 | 4 |
| 7 | 1 | 5.75 | 17.08 | 0.9926 | 5 |
| 8 | 1 | 5.90 | 16.92 | 0.9922 | 6 |
| 9 | 1 | 6.05 | 16.76 | 0.9917 | 7 |
| 10 | 1 | 6.20 | 16.61 | 0.9917 | 8 |
| 11 | 1 | 6.35 | 16.46 | 0.9917 | 9 |
| 12 | 1 | 6.50 | 16.31 | 0.9917 | 10 |
| 13 | 1 | 6.65 | 16.16 | 0.9917 | 10 |
| 14 | 1 | 6.80 | 16.01 | 0.9917 | 9 |
| 15 | 1 | 6.95 | 15.87 | 0.9922 | 8 |
| 16 | 1 | 7.10 | 15.74 | 0.9930 | 7 |
| 17 | 1 | 7.25 | 15.60 | 0.9935 | 6 |
| 18 | 1 | 7.40 | 15.47 | 0.9943 | 5 |
| 19 | 1 | 7.55 | 15.34 | 0.9952 | 4 |
| 20 | 1 | 7.70 | 15.22 | 0.9965 | 3 |
| 21 | 1 | 7.85 | 15.10 | 0.9978 | 2 |
| 22 | 1 | 5.00 | 17.00 | 0.9565 | 6 |
| 23 | 1 | 5.00 | 15.00 | 0.8696 | 21 |
| 24 | 1 | 20.00 | 3.00 | 1.0000 | 0 |
| 25 | 1 | 6.00 | 10.00 | 0.6957 | 17 |
| 26 | 1 | 12.00 | 10.00 | 0.9565 | 1 |
| 27 | 1 | 9.00 | 4.00 | 0.5652 | 2 |
| 28 | 1 | 3.00 | 4.00 | 0.3043 | 25 |

Figure 2.2 Graphical Representation of Second Example
(Source: Köksalan and Tuncer, 2009)




Köksalan and Tuncer (2009) show the output 1 in the x-axis and output 2 in the y-axis in Figure 2.2. The frontier is formed by DMUs 1, 2 and 24, which has the efficiency score of 1. According to the graphic, DMUs 3-21 are located close to each other because of their output values. This is a very extreme situation for DM to be ranking/sorting DMUs in such a crowded region, because of problematic results. When values of input and outputs of DMU's are close to each alternative, it is not sufficient for DM for ranking/sorting or eliminating DMUs for a decision.

## 2.4 Other Methods

We can see many DEA based ranking methods in the literature. Adler, Friman and Sinuary-Stern (2002), Büyükbaşaran (2005) mention the detailed literature review of ranking methods. Many sorting/ranking methods are proposed to be used with DEA for the Decision Maker (DM) including efficiency method, cross efficiency method, super efficiency method, multivariate statistical techniques, and multiple criteria decision concepts.

Classical DEA methods compute the efficiency of the DMUs. However, when the aim is to rank these DMUs, the efficiency values may not be enough since there may be several efficient DMUs. In order to increase the differentiation among DMUs (especially among efficient ones), researchers propose different approaches. Super efficiency and cross efficiency are two such approaches.

In classical DEA, the efficiency scores of all DMUs are bounded by 100%. However, in super efficiency approach, the DMU under consideration is not used in efficient frontier construction. Hence it may be efficient more than 100% if it does not lie inside the efficient frontier. We refer to Zhu (2001) for a detailed discussion of super efficiency.

The next approach is cross efficiency. In classical DEA, each DMU selects the best input and output weights in order to maximize its efficiency and the rankings are based on these efficiency scores. The cross efficiency approach, on the other hand, uses the efficiency scores of DMUs when they are not under consideration. For each DMU, a model is solved and the efficiency scores of all DMUs (including that for the DMU under consideration) are recorded, hence for each DMU we collect $n$ efficiency scores; one score when it is under consideration, and $n$-1 efficiency scores when other DMUs are under consideration. A function (for example average) of these $n$ scores is used to compute the cross efficiency score of the DMU. We refer to Doyle and Green (1994) for a review of cross efficiency, including several derivations.

# CHAPTER 3

# DEA BASED APPROACH TO RANKING MULTI-CRITERIA ALTERNATİVES

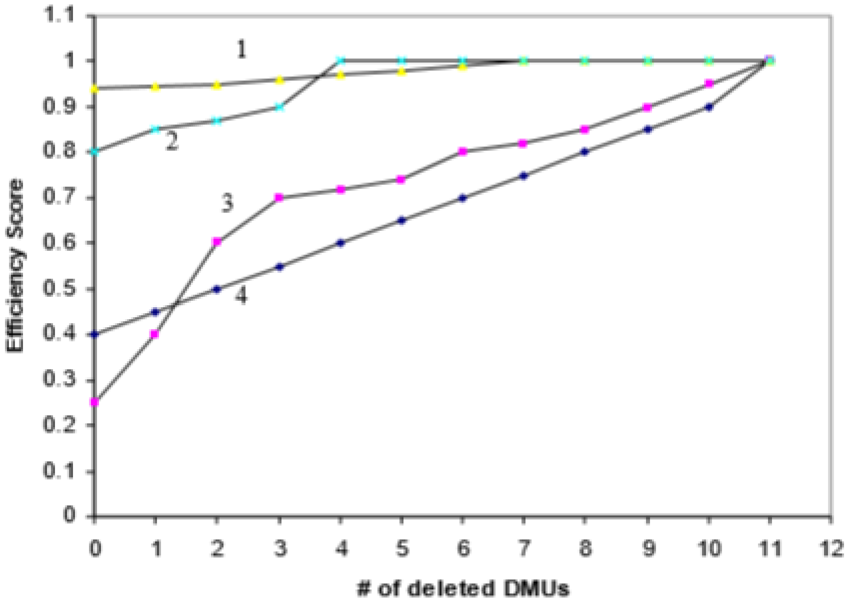
## 3.1 Motivation of Method

Köksalan and Tuncer (2009) develop a CCR based model Area of Efficiency Score (AES) for ranking alternatives. A detailed analysis of the method can be found in Tuncer (2006). It is a nonparametric method and measures the efficiency of units. We call this model proposed by Köksalan and Tuncer (2009) the KT model referring to the initials of the authors.

## 3.2. Area of the Efficiency Score

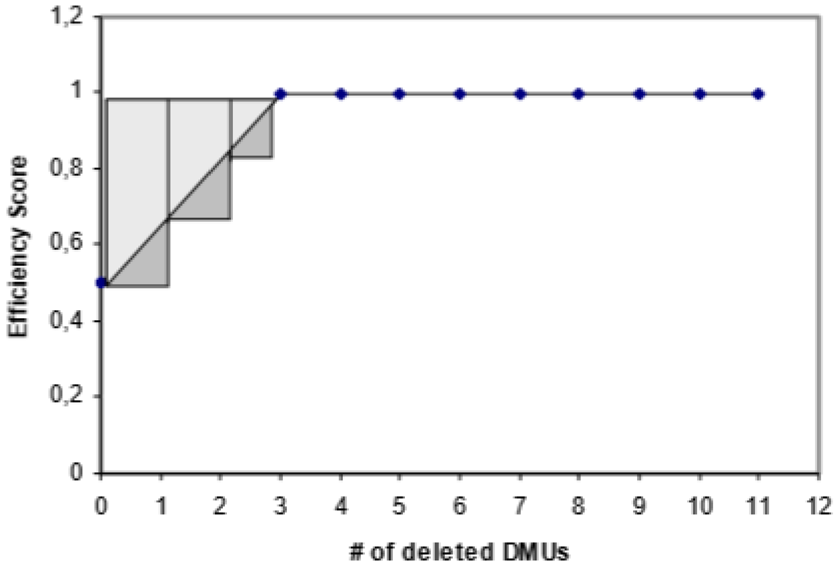
Köksalan and Tuncer (2009) use the area under the efficiency score curve. This curve is generated for each DMU separately. The curve shows the efficiency score of the DMU for each number of DMUs deleted from the DMU set. The efficient DMUs start at efficiency level while the inefficient DMUs start at their' beginning scores. The inefficient DMUs make their way up to (100% efficiency) from their initial efficiency scores by removing DMUs. This technique is called the Area of the Efficiency Score (AES) by Köksalan and Tuncer (2009). They illustrate this with four inefficient DMUs in Figure 3.1. They show the efficiency score of DMUs in the y-axis and number of deleted DMUs in the x-axis.

Figure 3.1 Efficiency score- # of deleted DMUs
(Source : Köksalan and Tuncer, 2009)




This graph illustrates how the efficiency scores of DMUs change step by step as the number of deleted DMUs increase. DMU 1 starts with a high efficiency score but it can only reach the 1- level (100% efficient level) when 7 DMUs are deleted. DMU 2 reaches the efficiency line after deleting 4 DMUs. Although DEA score of DMU 3 is lower than that of DMU 4, DMU 3 improves quickly with the number of two deleted DMUs. The relationship between DMU 3 and DMU 4 shows the superiority of DMU 3 until the efficiency line. Köksalan and Tuncer (2009) investigate the area which is under the score curve for each DMUs. They offer two ways to measure the area. The first one is to measure the area between the score curves and the efficiency level line. The second approach measures the areas in discrete histogram-like bars. In Figure 3.2, the two approaches are shown, efficiency score of DMUs in y-axis and number of deleted DMUs in the x-axis.

Figure 3.2 The areas according to two measurement approaches
(Source: Köksalan and Tuncer, 2009)




The calculations of area of efficiency method is showed as a mathematically below. They proposed which they used second approach because of discrete measurement.

$E_{kj}$ = the efficiency score of DMU k with deletion of j DMUs, $j = 0, \ldots\ldots, D_k$

$A_{kl}$ = area for DMU k computed with approach $l = 1, 2$

$A_{k1} = \sum_{j=0}^{D_k - 1} (1 - (E_{kj} + E_{kj+1})/2)/n - 1$ is showed as a triangle in Figure 3.2. Since we compute the average of preceding $E_{kj}$ pairs, there are n-1 terms in the above summation. By diving this term to $n$-1, we compute the average area under the efficiency scores curve.

$A_{k2} = \sum_{j=0}^{D_k} (1 - E_{kj})\ /n$ is showed three bars in Figure 3.2

They also offered third approach the efficiency measurement instead of inefficiency. The model is showed below.

$$A_{k3} = \sum_{j=0}^{n-1} E_{kj} \quad /n$$

For instance, Table 3.1 show efficiency score of DMU k for each number of DMUs deleted from the five DMUs set.

Table 3.1 Efficiency scores of DMU k.

| # of Deleted DMUs | Efficiency Score of DMU k |
|---|---|
| 0 | 0.7 |
| 1 | 0.8 |
| 2 | 1 |

$$A_{k1} = \sum_{j=0}^{Dk-1} (1 - (E_{kj} + E_{kj+1}) / 2) \quad /4$$ = (1- (0.7+0.8)/2) = 0.25

= (1- (0.8+1) / 2) = 0.10

$A_{k1}$ = 0.0875

$$A_{k2} = \sum_{j=0}^{D_k} (1 - E_{kj})/5$$ = (0.3+0.2+0+0+0) = 0.50/5 $A_{k2}$ = 0.10

$$A_{k3} = \sum_{j=0}^{n-1} E_{kj}/5$$ = 0.7 + 0.8 + 1 + 1 + 1 = 4.5/5 $A_{k3}$ = 0.90

As is seen, $A_{k1}$ and $A_{k2}$ methods measure the area above the efficiency score curves of the DMUs. The third approach measures the area efficiency score with each deletion. Koksalan and Tuncer (2009) use the second method for the area of efficiency score.

## 3.3. Mathematical Model (KT)

This model is solved for each DMU $k$ separately. We first introduce its indices, parameters and decision variables below;

Model KT($k$):

Indices :

$n$ = the number of DMUs in the process,

$m$ = the number of inputs considered,

$s$ = the number of outputs considered,

Parameters :

$Y_{ik}$ = the value of output i for DMU $k$ $(i=1,\ldots,s\ ;\ k = 1,\ldots,n)$

$X_{jk}$ = the value of input j for DMU $k$ $(j=1,\ldots,m\ ;\ k = 1,\ldots,n)$

$M$= A large number

Decision Variables :

$U_i$ = the weight for output $i$ $(i = 1,\ldots.,s)$

$V_j$ = the weight for input $j$ $(j = 1,\ldots,m)$

$$N_t = \begin{cases} 1, if\ DMU\ t\ is\ deleted \\ 0, otherwise \end{cases}$$

objective function :

$$Max\ Z\ = \sum_{i=1}^{s} U_i\ Y_{ik}$$

This model maximizes the efficiency of DMU by limiting the number of deleted DMUs by consecutive numbers between zero and $D_k$.

subject to :

$$\sum_{J=1}^{m} V_j\, X_{jk}\ = 1$$

This constraint sets the input value of DMU k as 1.

$$\sum_{i=1}^{s} U_i Y_{it} \leq \sum_{J=1}^{m} V_j X_{jt} + N_t * M$$

*for all t ≠ k*

The constraint ensures that the total output value of DMU t is less than or equal to its input value. However, if DMU t is deleted, then this constraint becomes redundant due to the *M* value on the right hand side.

$$\sum_{i=1}^{s} U_i Y_{ik} \leq 1$$

The total output value of DMU k (the efficiency of DMU k) should be at most 1.

$$\sum_{t=1}^{n} N_t \leq R_k$$

The above constraint ensures that at most $R_k$ DMUs are deleted.

$U_{i,}\ V_j \geq 0,\ i = 1,\dots,s;\ j = 1,\dots,m$

Consequently the model gives the highest AES score to $DMU_k$ when DM decide the how many DMUs are allowed to be removed as referred $R_k$.

### 3.4 KT Model Algorithm

Here we present the algorithm of Köksalan and Tuncer (2009). This algorithm considers each DMU separately, changes $R_k$ values systematically and reports the AES at the end.

For each DMU k, the algorithm sets the $R_k$ value as 0 and solves KT model. If the optimal objective function value is 1 (the DMU is efficient) then the algorithm computes the AES value and proceeds to next DMU. Otherwise, $R_k$ value is increased by one until DMU k becomes efficient. The AES value is initially 0 for each DMU and it is increased by Z until DMU is efficient. Then AES value is increased by ($n$-$R_k$-1) units . Note that ($n$-$R_k$-1) is the remaining number DMUs and the current DMU will be efficient if we delete these DMUs. Finally AES value is divided by $n$ and we obtain a final score between 0 and 1.

KT model has $n$ binary variables. The above algorithm requires to solve the KT model for at least $n$ times one for each DMU. In case of inefficient DMUs, total number of models solved increases.

For each DMU k

Set $AES_k$=0, $R_k$=0

Solve KT model and let the optimal objective function value Z

Set $AES_k$ = Z

While Z < 1

Set $R_k = R_k + 1$

Solve KT model and let the optimal objective function value Z

Set $AES_k = AES_k + Z$

End of while

Set $AES_k = AES_k + (n-R_k-1)$

Set $AES_k = AES_k / n$

next *k*

report *AES* and *R* vectors

# CHAPTER 4

# RANKING ALTERNATIVES USING DATA ENVELOPMENT ANALYSIS: A DUAL APPROACH

We develop a ranking method based on the CCR dual model, which is similar to the KT model. Originally, the dual technique yields the same efficiency score as the primal model. It provides another way of looking at the same problem. The dual model tries to construct a hypothetical composite unit out of the existing units which may outperform the given unit efficient or not. We refer to this method which is proposed by us, OY model.

## 4.1. Mathematical Model (OY)

Indices, parameters and the decision variables are mentioned at the second chapter. We add just one more constraint to this dual CCR to complete the same task KT model does. We know $\Theta$ that represents the efficiency score of unit, changes from 0 to 1 ( $0 < \Theta \leq 1$ ). And $\lambda_k$ represents the dual variables which identify benchmarks for inefficient units.

objective function :

min $\Theta$

subject to :

$$\sum_{k=1}^{K} \lambda_k X_{jk} \leq \Theta X_{j0} \ \forall j$$

$$\sum_{k=1}^{K} \lambda_k Y_{ik} \geq Y_{i0} \ \forall i$$

$$\lambda_k \geq 0$$

$V_j$ : $j = 1,\ldots,m$ (for each input j)

$U_i$ : $i = 1,\ldots,s$ (for each output i)

We add just one constraint CCR dual model to generate our model that handles the removed DMUs.

$$\sum_{k=1}^{n} W_k \ \lambda_k = 0$$

$W_k$ is a parameter for the model and initially $W_k = 0$ for all n. When DMU $k$ is removed by the model, we update the $W_k = 1$ hence forcing the corresponding $\lambda_k = 0$ since $\lambda_k$ are nonnegative decision variables.

## 4.2 OY Model Algorithm

We provide our algorithm below. This algorithm computes the OY score for each DMU. This algorithm uses OY model, which is a linear programming model.

The number of basic variables (that can be nonzero) of a linear programming model is equal to the number of constraints. In the OY model, there are ($I$+$J$+1) constraints and at most ($I$+$J$+1) out of $n$ $\lambda_k$ values can be positive at an optimal solution.

Consider an inefficient DMU 0 and let DMUs 1 and 2 are reference for this DMU, ie $\lambda_1$ and $\lambda_2$ are positive and other $\lambda_k$ values are 0. In this case, DMU 0's efficiency value may increase only if DMU 1 or DMU 2 are removed from them DMU set. Removing any other DMU will not affect the efficiency of DMU 0. Based on this observation, we consider the reference DMUs and find the best reference DMU such that its removal will increase the efficiency score of DMU 0 most.

OY algorithm runs for each DMU k. If DMU k is efficient then it computes the $OY_k$ score and proceeds to the next DMU. If it is inefficient, then it considers each reference DMU j with $\lambda_j > 0$. Temporarily sets $w_j$ values to 1, solves OY model and sets $w_j$ to 0 again. We identify the best DMU to be removed with this strategy. The removal of DMUs continue until DMU k becomes efficient.

For each DMU $k$

Set $OY_k$=0, $R_k$=0 and $w_j$=0 for all $j$

Solve OY model and let the optimal objective function value be Z

Set $OY_k = OY_k + Z$

while $Z < 1$ then

Solve OY model and let the optimal $\lambda_j$ values be $\lambda_j^*$

Set $Z^* = 0$

For each reference DMU $j$ (such that $\lambda_j^* > 0$)

Set $w_j = 1$

Solve OY model and let Z* be the optimal obj. func. value

If Z > Z*

Set $Z^* = Z$ and $j^* = j$

end

Set $w_j = 0$

next $j$

$OY_k = OY_k + Z^*$

Set $w_{j^*} = 1$

Set $R_k = R_k + 1$

end (of while)

Set $OY_k = OY_k + (n - R_k - 1)$

Set $OY_k = OY_k / n$

next $k$

report $OY$ and $R$ vectors

Note that this algorithm is myopic and detects the best DMU to be removed iteratively. Moreover, for a given $R_k$ value, it sticks to the previous removal decisions and finds the DMU to be removed given the previous $R_k$-1 removals. However, KT model considers the best subset (with cardinality of $R_k$) of DMUs to be removed.

Obviously, the removal decisions of KT and OY models differ. The AES and OY scores and total number of DMUs to be deleted may vary between the two models. We compute the correlations between these values in the next chapter.

# CHAPTER 5

# EXPERIMENTS

In this chapter, we present the computational experiments made for both OY and KT models. The differences of between the models are showed in this section.

We performed the computational experiments on computer with Intel R Xenon X5482 @ 3.20 GHz 2 processors, 10 GB RAM and Windows Server 2008 R2 Enterprise-64 Bit. We solve the mathematical models using GAMS 22.8.

## 5.1 Random Data Generation

We also benefit from data generation the following model, which is proposed by Özpeynirci and Köksalan (2007). The model is producing some random outputs with random weighted inputs for seeing differences between the methods.

$$Y_{rj} = H_j * \left( \sum_{i=1}^{m} W_{ir} * X_{ij} \right) + e_{rj}$$

$W_{ir}$ = the weight of input *i* consumed for output *r*

$e_{rj}$ = error term on output *r* of DMU *j*

$H_j$ = the efficiency value of DMU *j*

$X_{ij}$ = the amount of input *i* consumed by DMU *j*

$Y_{rj}$ = the amount of output *r* produced by DMU *j*

$X_{ij}$ ~ uniform (10, 50)

$W_{ir}$ ~uniform (0.10, 0.50) and sum ( $W_{ir}$ ) = 1

$e_{rj}$ ~ uniform (0, 20)

$H_j$ ~ uniform (60%, 100%)

We can generate multiple output/input data set using this model for evaluating performance of DMUs. We produce data set which has 250 DMUs, 2 inputs and 6 outputs using this model, that is mentioned above. This set is used for evaluating the performance of the two different models. In Table 5.1, the results are shown only for 63 DMUs because of total time of KT model. Although our method found all the efficiency scores of DMUs in 1 hour 17 min, in contrast, the KT model took more than 50 hours to get same information. KT model finds the efficiency scores of only the first 63 DMUs at the end of a 50 hours run time. We present the results of the different methods for comparison two methods using data generation. We provide the details of Table 5.1 in Appendix A. Note that CPU and Total Time are given as in seconds.

Table 5.1 Comparison of  between OY and KT Models with $R_0$, $EFF_0$, $CPU_0$ , $Duration\ TIME_O$ with using Data Generation

| DMU | $EFF_{OY}$ | $R_{0Y}$ | $CPU_{OY}$ | $TIME_{OY}$ | $EFF_{KT}$ | $R_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0.007 | 0.48 | 1 | 0 | 0.105 | 0.4 | 0 | 0.000000 |
| 2 | 1 | 0 | 0.015 | 0.58 | 1 | 0 | 0.164 | 0.5 | 0 | 0.000000 |
| 3 | 0.999764 | 4 | 0.751 | 2.34 | 0.999764 | 4 | 0.736 | 1.44 | 0 | 0.000000 |
| 4 | 0.861277 | 162 | 23.235 | 48.57 | 0.87837 | 156 | 16529.29 | 16543.09 | 6 | -0.017093 |
| 5 | 0.986564 | 32 | 27.67 | 57.59 | 0.98743 | 32 | 16538.32 | 16554.29 | 0 | -0.000866 |
| 6 | 0.993149 | 44 | 35.356 | 73.13 | 0.99424 | 38 | 16619.37 | 16638.02 | 6 | -0.001091 |
| 7 | 0.958575 | 108 | 50.853 | 104.47 | 0.961286 | 102 | 24587.08 | 24613.29 | 6 | -0.002711 |
| 8 | 0.878762 | 184 | 80.101 | 163.34 | 0.893367 | 147 | 48042.12 | 48080.14 | 37 | -0.014605 |
| 9 | 0.998061 | 14 | 81.888 | 166.94 | 0.998146 | 13 | 48044.3 | 48083.19 | 1 | -0.000085 |
| 10 | 0.947778 | 107 | 94.239 | 191.98 | 0.953126 | 97 | 49383.16 | 49430.21 | 10 | -0.005348 |
| 11 | 0.981087 | 37 | 99.574 | 202.41 | 0.983066 | 33 | 49390.46 | 49439.94 | 4 | -0.001979 |
| 12 | 0.984608 | 60 | 110.072 | 223.64 | 0.986621 | 58 | 50442.65 | 50496.57 | 2 | -0.002013 |
| 13 | 0.993308 | 24 | 114.338 | 232.14 | 0.994079 | 22 | 50451.68 | 50507.26 | 2 | -0.000771 |
| 14 | 0.996631 | 33 | 122.061 | 247.77 | 0.997118 | 27 | 50518.7 | 50576.54 | 6 | -0.000487 |
| 15 | 0.99716 | 19 | 125.271 | 254.18 | 0.997557 | 16 | 50523.18 | 50582.48 | 3 | -0.000397 |
| 16 | 1 | 0 | 125.314 | 254.28 | 1 | 0 | 50523.23 | 50582.59 | 0 | 0.000000 |
| 17 | 0.932982 | 130 | 150.83 | 305.79 | 0.944223 | 96 | 52786.13 | 52852.8 | 34 | -0.011241 |
| 18 | 0.996288 | 24 | 154.726 | 313.59 | 0.997068 | 20 | 52790.7 | 52859.08 | 4 | -0.000780 |
| 19 | 0.984088 | 35 | 159.047 | 322.2 | 0.986037 | 32 | 52799.81 | 52870.48 | 3 | -0.001949 |
| 20 | 0.970658 | 61 | 168.05 | 340.43 | 0.975551 | 51 | 52831.84 | 52906.33 | 10 | -0.004893 |
| 21 | 0.999735 | 5 | 168.754 | 341.83 | 0.999765 | 5 | 52832.7 | 52907.55 | 0 | -0.000030 |

| DMU | $EFF_{OY}$ | $R_{0Y}$ | $CPU_{OY}$ | $TIME_{OY}$ | $EFF_{KT}$ | $R_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 22 | 0.990573 | 19 | 171.397 | 347.33 | 0.991088 | 17 | 52835.23 | 52911.11 | 2 | -0.000515 |
| 23 | 0.95134 | 76 | 182.17 | 369.47 | 0.955652 | 73 | 52965.7 | 53046.97 | 3 | -0.004312 |
| 24 | 0.944513 | 119 | 195.424 | 396.40 | 0.94958 | 105 | 56751.26 | 56840.48 | 14 | -0.005067 |
| 25 | 0.914119 | 119 | 210.943 | 427.52 | 0.921669 | 116 | 57690.69 | 57788.81 | 3 | -0.007550 |
| 26 | 0.941818 | 123 | 231.714 | 469.7 | 0.948426 | 102 | 61766.36 | 61872.59 | 21 | -0.006608 |
| 27 | 0.856687 | 203 | 256.86 | 520.73 | 0.872245 | 156 | 108044.9 | 108165.5 | 47 | -0.015558 |
| 28 | 0.999761 | 4 | 257.71 | 522.44 | 0.999761 | 4 | 108045.6 | 108166.5 | 0 | 0.000000 |
| 29 | 0.999467 | 7 | 258.7 | 524.44 | 0.999467 | 7 | 108046.7 | 108168.2 | 0 | 0.000000 |
| 30 | 0.999626 | 4 | 259.279 | 525.64 | 0.999626 | 4 | 108047.4 | 108169.1 | 0 | 0.000000 |
| 31 | 0.981389 | 65 | 266.917 | 541.05 | 0.982448 | 62 | 108349.4 | 108476.2 | 3 | -0.001059 |
| 32 | 0.989139 | 49 | 275.829 | 558.77 | 0.991273 | 39 | 108456.6 | 108586.3 | 10 | -0.002134 |
| 33 | 0.974169 | 44 | 281.858 | 571.19 | 0.975283 | 43 | 108471.4 | 108604.1 | 1 | -0.001114 |
| 34 | 0.839711 | 179 | 309.922 | 628.46 | 0.852353 | 177 | 132692.2 | 132841.1 | 2 | -0.012642 |
| 35 | 0.930042 | 136 | 332.412 | 674.76 | 0.937163 | 114 | 146987.8 | 147145.9 | 22 | -0.007121 |
| 36 | 0.928549 | 142 | 353.925 | 718.72 | 0.932467 | 133 | 169458.1 | 169626.8 | 9 | -0.003918 |
| 37 | 0.999726 | 5 | 354.96 | 720.73 | 0.999726 | 5 | 169458.9 | 169628 | 0 | 0.000000 |
| 38 | 0.999039 | 15 | 357.184 | 725.43 | 0.999162 | 11 | 169461.2 | 169631 | 4 | -0.000123 |
| 39 | 0.901807 | 113 | 374.49 | 760.9 | 0.913648 | 106 | 169713.7 | 169891.7 | 7 | -0.011841 |
| 40 | 0.979057 | 66 | 383.054 | 778.02 | 0.982118 | 60 | 169973.7 | 170156.4 | 6 | -0.003061 |
| 41 | 0.996021 | 29 | 388.055 | 788.03 | 0.996943 | 24 | 169988.1 | 170172.4 | 5 | -0.000922 |
| 42 | 1 | 0 | 388.101 | 788.12 | 1 | 0 | 169988.1 | 170172.5 | 0 | 0.000000 |
| 43 | 0.912783 | 139 | 413.959 | 840.6 | 0.935093 | 98 | 170489.5 | 170681.5 | 41 | -0.022310 |
| 44 | 0.992423 | 13 | 415.458 | 843.6 | 0.993061 | 13 | 170491.6 | 170684.4 | 0 | -0.000638 |

| DMU | $EFF_{OY}$ | $R_{0Y}$ | $CPU_{OY}$ | $TIME_{OY}$ | $EFF_{KT}$ | $R_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 45 | 0.872303 | 130 | 431.132 | 875.14 | 0.895833 | 117 | 170963.2 | 171165 | 13 | -0.023530 |
| 46 | 0.986534 | 61 | 440.551 | 894.45 | 0.988809 | 51 | 171259.2 | 171464.7 | 10 | -0.002275 |
| 47 | 0.980521 | 55 | 448.278 | 910.45 | 0.982997 | 52 | 171324.9 | 171534.7 | 3 | -0.002476 |
| 48 | 0.960871 | 70 | 460.193 | 935.11 | 0.964834 | 57 | 171369.8 | 171584.2 | 13 | -0.003963 |
| 49 | 0.975413 | 37 | 464.818 | 944.33 | 0.976778 | 37 | 171377.8 | 171595.1 | 0 | -0.001365 |
| 50 | 0.901363 | 105 | 478.868 | 972.75 | 0.919213 | 89 | 171472 | 171696.3 | 16 | -0.017850 |
| 51 | 0.959136 | 60 | 487 | 989.95 | 0.962488 | 55 | 171502 | 171730.8 | 5 | -0.003352 |
| 52 | 0.988997 | 38 | 493.184 | 1002.67 | 0.990443 | 31 | 171519.5 | 171750.6 | 7 | -0.001446 |
| 53 | 0.996917 | 26 | 497.305 | 1011.08 | 0.997291 | 22 | 171529.3 | 171761.9 | 4 | -0.000374 |
| 54 | 0.96815 | 56 | 503.692 | 1024.29 | 0.972621 | 47 | 171546.9 | 171783 | 9 | -0.004471 |
| 55 | 0.998947 | 6 | 504.147 | 1025.19 | 0.998947 | 6 | 171547.4 | 171784.2 | 0 | 0.000000 |
| 56 | 0.982897 | 25 | 507.02 | 1031 | 0.983069 | 25 | 171551.2 | 171789.9 | 0 | -0.000172 |
| 57 | 0.952961 | 75 | 514.733 | 1046.51 | 0.962186 | 54 | 171574.2 | 171817.2 | 21 | -0.009225 |
| 58 | 0.979836 | 41 | 521.243 | 1059.54 | 0.980925 | 39 | 171590.3 | 171836.3 | 2 | -0.001089 |
| 59 | 0.99971 | 8 | 522.657 | 1062.35 | 0.999716 | 7 | 171591.6 | 171838.1 | 1 | -0.000006 |
| 60 | 0.948718 | 88 | 535.22 | 1088.18 | 0.954973 | 82 | 172447.9 | 172700.4 | 6 | -0.006255 |
| 61 | 0.988085 | 54 | 545.476 | 1109.11 | 0.98961 | 51 | 172934.6 | 173191.3 | 3 | -0.001525 |
| 62 | 0.986128 | 29 | 549.504 | 1117.32 | 0.988492 | 26 | 172941 | 173199.6 | 3 | -0.002364 |
| 63 | 0.98489 | 34 | 555.296 | 1128.52 | 0.987541 | 29 | 172949.9 | 173210.9 | 5 | -0.002651 |

When we compare the two models which have 250 DMUs, 2 inputs and 6 outputs, we make the following observations:

1. CPU Time of OY model is much lower than that of KT model.
2. Duration Time of OY model is much lower than that of KT model.
3. DMU that is under consideration, $EFF_{OY} \leq EFF_{KT}$.
4. DMU that is under consideration, $R_{OY} \geq R_{KT}$.

We can observe the differences between $R_{0Y}$- $R_{KT}$ and $EFF_{OY}$ - $EFF_{KT}$ under consideration DMUs which the models make the same tasks. And we demonstrate our methods performance with examples.

We also make a correlation analysis between $R_{OY}$ and $R_{KT}$ values only for 63 DMUs. The correlation value is 0.988417. Number of DMUs deleted by OY model and KT model presented by graphically in Figure 5.1.

The correlation of efficiency scores ( $EFF_{OY}$ and $EFF_{KT}$ )is 0.997647. The result of positive correlation analysis confirms that the results of our method is very similar to that of KT. The ratio of $EFF_{OY}$ to $EFF_{KT}$ is found nearly one, and we present it graphically in Figure 5.2.

In a detailed comparison between methods, each of DMUs choose their weights to become efficient in both models. DMUs which is are under consideration try to virtually delete. Consequently, KT model considers each DMU separately, needs n binary decision variables and searches all possible removal decisions for each DMU from the set to remove best R DMUs. This procedure continues until DMU becomes efficient. According to our method, OY model runs for each DMU and continues to compute efficiency score of DMUs with deleted DMUs (previously removed *R-1* DMUs). The KT model aims is to remove best R DMUs, in contrast, our myopic model that does not need any binary or integer decision variables, removes the best DMU given the previously removed R-1 DMUs. This is the reason that the two methods are separated from each other. Although, the removal decisions are different in both models, our method finishes the process earlier for large number of DMUs.

Figure 5.1 Number of DMUs deleted by OY model and KT model

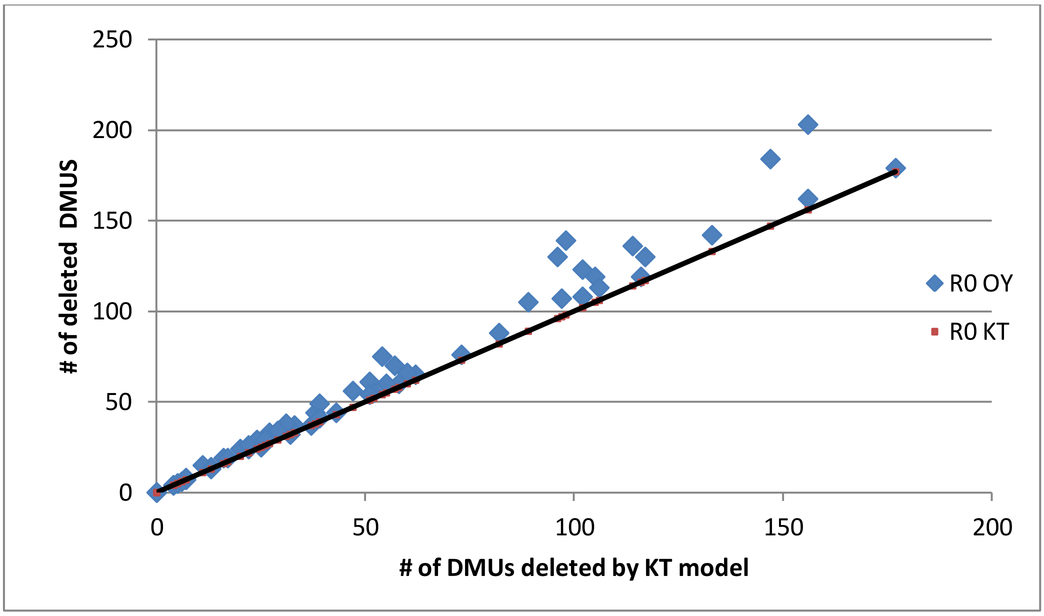
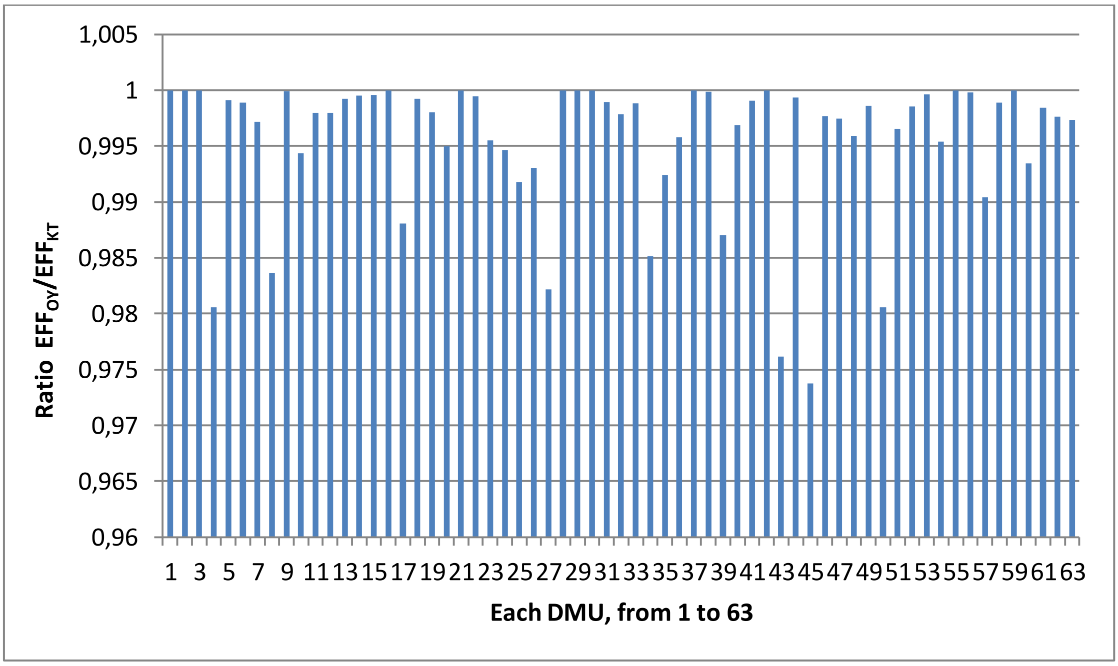



Figure 5.2 The Ratio of $EFF_{OY}$ to $EFF_{KT}$



## 5.2 Logistics Performance Index Application

In this section, we apply KT and OY methods on Logistics Performance Index data set. Logistics Performance Index (LPI) is defined as "an interactive benchmarking tool created to help countries identify the challenges and opportunities they face in their performance on trade logistics and what they can do to improve their performance." The LPI data set for 2014 (LPI 2014) allows comparisons across 160 countries, calculated from two different perspectives: international and domestic on the six dimensions; the efficiency of customs and border management clearance "Customs", the quality of trade and transport infrastructure "Infrastructure", the ease of arranging competitively priced shipments (Shipments), the competence and quality of logistics services-trucking, forwarding, and customs brokerage (Quality of logistics services), the ability to track and trace consignments (Tracking and tracing), the frequency with which shipments reach consignees within scheduled or expected delivery times (Timeliness). We use the six dimensions above as outputs of the countries and one constant input which is value 1 in our method and compare the results with KT method.

The data which is presented in Table B.1 in Appendix B , is received from LPI for the comparison of two approaches.

We make a test for ranking countries as a logistics application. The results are in the Table C.1 in Appendix C.

Correlation Analysis is utilized efficiency score of between $EFF_{OY}$ and $EFF_{KT}$ models, the result found as 0.999778. The result of positive correlation analysis also confirms the sensitivity to another method. The ratio of $EFF_{OY}$ to $EFF_{KT}$ is analyzed and found to be nearly one in this application, that is presented by graphically in Figure 5.3.

On the other hand, we also make an correlation analysis between $R_{OY}$ and $R_{KT}$ for our application, which was found to be 0.999317. The number of DMUs deleted by OY model and KT model presented by graphically for LPI data set in Figure 5.4.

Figure 5.3 The Ratio of $EFF_{OY}$ to $EFF_{KT}$

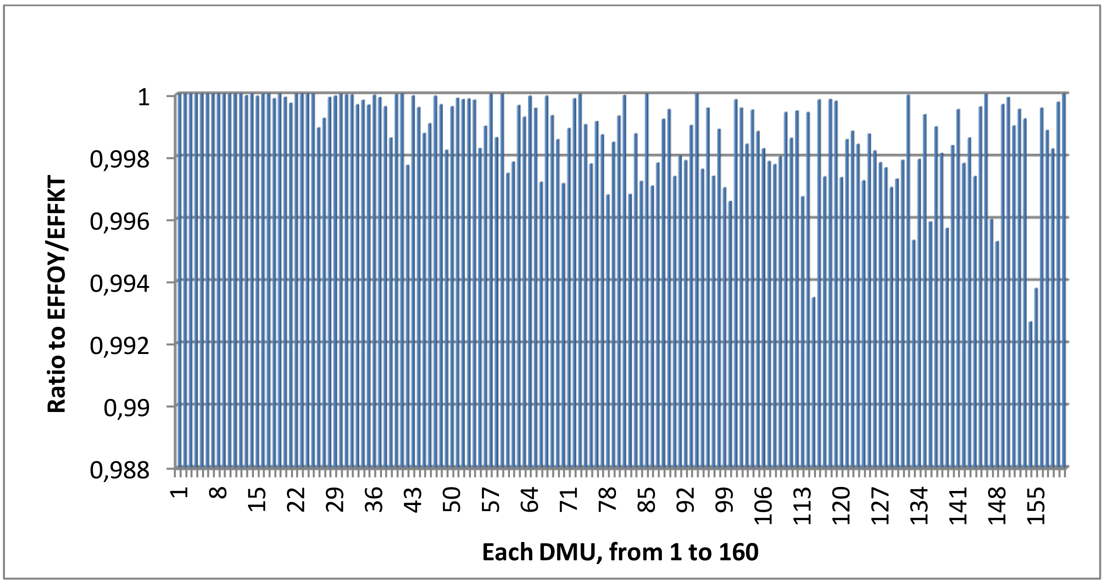
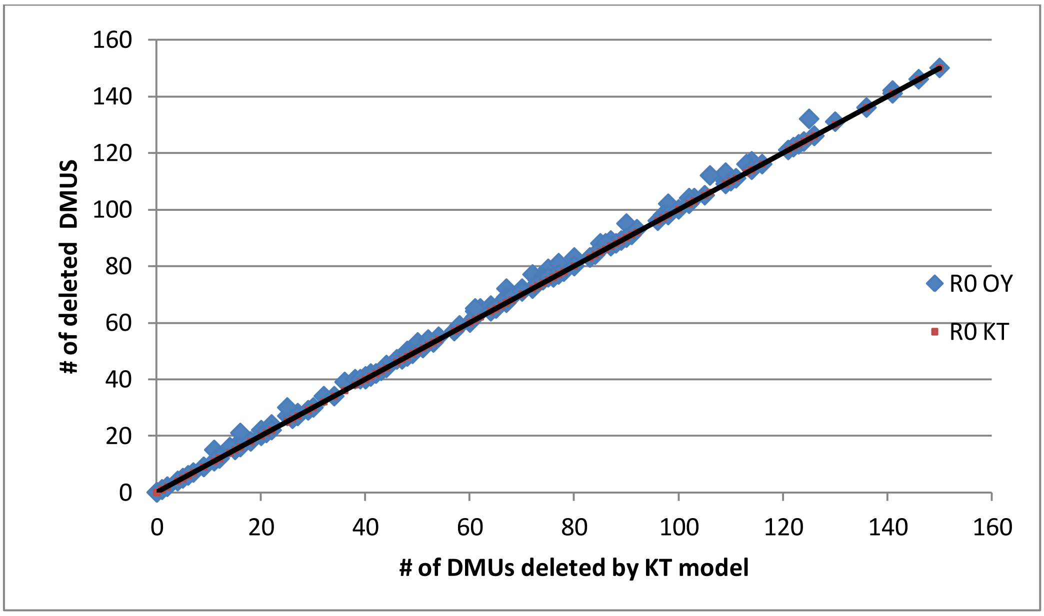



Figure 5.4 Number of DMUs deleted by OY model and KT model



## 5.3 Observations

In this section, we report our observations of the performances of the models on randomly generated and LPI data sets.

Table 5.2. Total Time (second) of Models related to # of DMUs, # of Inputs/Outputs

| # of DMUs | 2 inputs 3 outputs | | 2 inputs 6 outputs | |
|---|---|---|---|---|
| | OY Model | KT model | OY Model | KT model |
| 25 | 34.54 | 26.69 | 35.13 | 25.84 |
| 50 | 146.79 | 129.54 | 161.68 | 116.82 |
| 75 | 327.17 | 333.82 | 333.57 | 345.55 |
| 100 | 602.87 | 929.94 | 684.66 | 993.87 |
| 150 | 1405.16 | 7304.9 | 1579 | 9857 |
| 250 | 4116.25 | ---- | 4657.5 | ---- |

The dashed line in Table 5.2 shows that the KT model could not find any solution in time, when we increase the number of DMUs, specifically 250 DMUs. We present precise total time measurement performance of models in Table 5.3.

Table 5.3 Total Time of the models related to numbers of DMUs and # of Inputs/Outputs

| | # of DMUs (by OY Method) | | | # of DMUs (by KT Method) | | |
|---|---|---|---|---|---|---|
| # of inputs and outputs | 100 | 160 (LPI ) | 250 | 100 | 160 (LPI) | 250 |
| 1 input+ 2 output | 541.13 | 1491.44 | 3797.03 | 1152.72 | 2881.79 | 8890.09 |
| 1 input+ 4 output) | 660.26 | 1713.08 | 4392.32 | 1048.51 | 2860.49 | 12482.24 |
| 1 input+ 6 output | 719.13 | 1905.09 | 5001.68 | 1040.08 | 3122.10 | 11596.98 |

Table 5.4. Correlation Analysis between $CPU_{OY}$ and $CPU_{KT}$, $\text{EFF}_{OY}$ and $\text{EFF}_{KT}$

| # of DMUs | $CPU_{OY}$ | $CPU_{KT}$ | $CPU_0$ *Correlation Analysis* | $EFF_0$ *Correlation Analysis* |
|---|---|---|---|---|
| 25 | 14.578 | 15.185 | 0.996423 | 0.998249 |
| 50 | 87.874 | 74.863 | 0.999417 | 0.998515 |
| 75 | 178.734 | 261.500 | 0.995705 | 0.997141 |
| 100 | 350.310 | 836.230 | 0.993853 | 0.997684 |
| 150 | 870.135 | 9506.865 | 0.987685 | 0.997130 |
| 250 | 2244.771 | --- | ---- | ---- |

On the other hand, we can be aware of differences between $CPU_{OY}$ and $CPU_{KT}$ in Table 5.4. From 25 DMUs to 150 DMUs, the degree of positive correlation decreases. In any case, the correlation of efficiency scores are very close and using OY method seems to be a reasonable sacrifice from the optimality in order to obtain a good result in less computation time.

If DM has many alternatives for ranking, the dual approach save time in the computational experiments. If the DM has a large set as numbers of alternatives/units, the dual form is more efficient computation compared to the primal model KT. This is the reason that the primal form KT model changes related to the number of DMUs when the OY model entails the number of inputs/outputs values.

Table 5.5 Average of $R_O$ the models related to numbers of DMUs and # of Inputs/Outputs

| | Average of $R_{OY}$ | | | Average of $R_{KT}$ | | |
|---|---|---|---|---|---|---|
| # of inputs and outputs | 100 DMUs | 160 DMUs (LPI) | 250 DMUs | 100 DMUs | 160 DMUs (LPI) | 250 DMUs |
| 1 input+ 2 output | 21.53 | 71.58 | 121.62 | 19.84 | 71.29 | 119.59 |
| 1 input+ 4 output) | 47.04 | 64.16 | 118.79 | 46.94 | 63.56 | 118.46 |
| 1 input+ 6 output | 46.56 | 59.88 | 117.70 | 46.48 | 58.94 | 117.30 |

As is seen, the average of $R_{OY}$ and $R_{KT}$ are very close to each other in the same data set in Table 5.5.

# CHAPTER 6

# CONCLUSION

Data Envelopment Analysis (DEA) is a non-parametric technique that has been used for several purposes including the ranking alternatives in the literature. DEA is used for assessing the relative performance of units and benchmarking between the independent decision making units as we mentioned before. Especially, DEA does not limit the number of the inputs and outputs and not require to weight restrictions.

In the study, we propose a DEA-based CCR dual approach for ranking alternatives. Our new model considers the efficiency score of alternatives while virtually deleting DMUs in the creation of a ranking list. Mathematically, the dual approach seems more complex, but in fact it is much faster to solve, as it has only as many constraints as there are factors, as we have seen.

We use the data generation model and a data from 2014 published in the Logistics Performance Index (LPI). Using the different set in both models is showed in the computational experiments part. For comparison between KT and OY models, we observe that there are the approximately same results under different assumptions for ranking.

The KT and OY models are differed by the removal decisions. KT model considers all possibilities for each Decision Making Unit separately to remove best R DMUs. On the other hand, our model runs for each DMU to remove throughout previous removed DMUs in the process. Although the KT model is more effective in contrast to OY model, KT method could not give any solution in time for large number of

DMUs.  Our OY model that does not need any binary or integer decision variables, review myopically on the DMUs and gives the results early.

In further research, If there would be many criteria with multiple inputs and outputs, KT model is useful for ranking in this situation. In other cases, our method is effective and beneficial for large number of DMUs for Decision Makers.

# REFERENCES


Adler, N., Friedman, L., Sinuany-Stern, Z.,(2002) , "Review of Ranking Methods in Data Envelopment Analysis", European Journal of Operational Research, 140, 249-265

Ahn, T., Charnes, A., Cooper, W.W., (2006). “Using Data Envelopment Analysis to Measure the Efficiency of Not-for-Profit Organizations: A Critical Evaluation – a Comment.” Managerial and Decision Economics 9: 251-253

Anderson, P., Peterson, N.C., (1993), "A Procedure For Ranking Efficient Units in Data Envelopment Analysis", Management Science, 39, 1261-1264

Büyükbaşaran, T., (2005), "Ranking Units by Target-Direction-Set Value Efficiency Analysis and Mixed Integer Programming", MS Thesis, Middle East Technical University

Charnes, A., Cooper, W., Rhodes, E., (1978), "Measuring Efficiency of Decision Making Units", European Journal of Operational Research, 2, 429-444

Cook, W., Seiford, L., (2009), "Data Envelopment Analysis (DEA) - Thirty Years On", European Journal of Operational Research, 192, 1-17

Cook, W.D., Kress, M., (1990), "A Data Envelopment Model For Aggreagating Preference Rankings", Management Science, 36, 1302-1310

Cooper, W.W., Seiford, L., Zhu, J. (2004), "Data Envelopment Analysis History, Models and Interpretations", Cooper W.W, Seiford L.M., Zhu, J. (Ed), *Handbook of Data Envelopment Analysis(2nd ed.),* 1-39, Boston: Kluwer Academic Publishers

Cooper, W.,W., Seiford, L., Tone, K., (2000), “Data Envelopment Analysis: A Comprehensive Text with Models, Applications, References, and DEA-Solver Software”, Kluwer Academic Publishers, USA

Doyle, J., Green, R., (1994), "Efficiency and Cross-Efficiency in DEA: Derivations, Meanings and Uses", The Journal of the Operational Research Society, 45, 567-578

Farrell, J., (1957), "The Measurement of The Productive Efficiency", Journal of the Royal Statistical Society, 120, 253-281

Foroughi, A.A., Tamiz, M., (2005), "An Effective Total Ranking Model for Ranked Voting System", The International Journal of Management Science, 33, 491-496

Köksalan, M., Tuncer, C., (2009), "A DEA Based Approach to Ranking Multi-Criteria Alternatives", International Journal of Information Technology & Decision Making, 1, 29-54

Köksalan, M., Büyükbaşaran, T., Özpeynirci, Ö., Wallenius, J., (2010), "A Flexible Approach to Ranking With An Application to MBA Programs", European Journal of Operational Research, 201, 471-476

Landsdowne, Z.F., (1996), "Ordinal Ranking Methods for Multi criterion Decision Making", Naval Research Logistics, 43, 613-627

Li, X. B.,Reeves, G., (1999), "A Multiple Criteria Approach to Data Envelopment Analysis", European Journal of Operational Research, 115, 507-517

Logistic Performance Index, (2014), "http://lpi.worldbank.org / International Global Ranking/ 2014", last date accessed: 06.06.2014

Obata, T., Ishii, H., (2003), "A Method for Discriminating Efficient Candidates with Ranked Voting Data", European Journal of Operational Research, 151, 233-237

Özpeynirci, Ö., Köksalan, M., (2007), "Performance Evaluation Using The Data Envelopment Analysis In The Presence of Time Lags", Journal of Productivity Analysis, 27, 221-229

Tuncer, C., (2006), "A DEA Based Approach to Ranking Multi-Criteria Alternatives", MS Thesis, Middle East Technical University

Seiford, L., Thrall, R.M., (1990), “Recent Developments in DEA: The Mathematical Programming Approach to Frontier Analysis”, Journal of Econometrics, 47, 7-38

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

APPENDIX A.

Table A.1 Comparison of OY and KT Model with using Data Generation (Continued)

| DMUs | $EFF_{OY}$ | $R_{0Y}$ | $CPU_{OY}$ | $TIME_{OY}$ | $EFF_{KT}$ | $R_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0.007 | 0.48 | 1 | 0 | 0.105 | 0.4 | 0 | 0.000000 |
| 2 | 1 | 0 | 0.015 | 0.58 | 1 | 0 | 0.164 | 0.5 | 0 | 0.000000 |
| 3 | 0.999764 | 4 | 0.751 | 2.34 | 0.999764 | 4 | 0.736 | 1.44 | 0 | 0.000000 |
| 4 | 0.861277 | 162 | 23.235 | 48.57 | 0.87837 | 156 | 16529.29 | 16543.09 | 6 | -0.017093 |
| 5 | 0.986564 | 32 | 27.67 | 57.59 | 0.98743 | 32 | 16538.32 | 16554.29 | 0 | -0.000866 |
| 6 | 0.993149 | 44 | 35.356 | 73.13 | 0.99424 | 38 | 16619.37 | 16638.02 | 6 | -0.001091 |
| 7 | 0.958575 | 108 | 50.853 | 104.47 | 0.961286 | 102 | 24587.08 | 24613.29 | 6 | -0.002711 |
| 8 | 0.878762 | 184 | 80.101 | 163.34 | 0.893367 | 147 | 48042.12 | 48080.14 | 37 | -0.014605 |
| 9 | 0.998061 | 14 | 81.888 | 166.94 | 0.998146 | 13 | 48044.3 | 48083.19 | 1 | -0.000085 |
| 10 | 0.947778 | 107 | 94.239 | 191.98 | 0.953126 | 97 | 49383.16 | 49430.21 | 10 | -0.005348 |
| 11 | 0.981087 | 37 | 99.574 | 202.41 | 0.983066 | 33 | 49390.46 | 49439.94 | 4 | -0.001979 |
| 12 | 0.984608 | 60 | 110.072 | 223.64 | 0.986621 | 58 | 50442.65 | 50496.57 | 2 | -0.002013 |
| 13 | 0.993308 | 24 | 114.338 | 232.14 | 0.994079 | 22 | 50451.68 | 50507.26 | 2 | -0.000771 |
| 14 | 0.996631 | 33 | 122.061 | 247.77 | 0.997118 | 27 | 50518.7 | 50576.54 | 6 | -0.000487 |
| 15 | 0.99716 | 19 | 125.271 | 254.18 | 0.997557 | 16 | 50523.18 | 50582.48 | 3 | -0.000397 |
| 16 | 1 | 0 | 125.314 | 254.28 | 1 | 0 | 50523.23 | 50582.59 | 0 | 0.000000 |
| 17 | 0.932982 | 130 | 150.83 | 305.79 | 0.944223 | 96 | 52786.13 | 52852.8 | 34 | -0.011241 |
| 18 | 0.996288 | 24 | 154.726 | 313.59 | 0.997068 | 20 | 52790.7 | 52859.08 | 4 | -0.000780 |

| DMUs | $EFF_{OY}$ | $R_{0Y}$ | $CPU_{OY}$ | $TIME_{OY}$ | $EFF_{KT}$ | $R_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 19 | 0.984088 | 35 | 159.047 | 322.2 | 0.986037 | 32 | 52799.81 | 52870.48 | 3 | -0.001949 |
| 20 | 0.970658 | 61 | 168.05 | 340.43 | 0.975551 | 51 | 52831.84 | 52906.33 | 10 | -0.004893 |
| 21 | 0.999735 | 5 | 168.754 | 341.83 | 0.999765 | 5 | 52832.7 | 52907.55 | 0 | -0.000030 |
| 22 | 0.990573 | 19 | 171.397 | 347.33 | 0.991088 | 17 | 52835.23 | 52911.11 | 2 | -0.000515 |
| 23 | 0.95134 | 76 | 182.17 | 369.47 | 0.955652 | 73 | 52965.7 | 53046.97 | 3 | -0.004312 |
| 24 | 0.944513 | 119 | 195.424 | 396.4 | 0.94958 | 105 | 56751.26 | 56840.48 | 14 | -0.005067 |
| 25 | 0.914119 | 119 | 210.943 | 427.52 | 0.921669 | 116 | 57690.69 | 57788.81 | 3 | -0.007550 |
| 26 | 0.941818 | 123 | 231.714 | 469.7 | 0.948426 | 102 | 61766.36 | 61872.59 | 21 | -0.006608 |
| 27 | 0.856687 | 203 | 256.86 | 520.73 | 0.872245 | 156 | 108044.9 | 108165.5 | 47 | -0.015558 |
| 28 | 0.999761 | 4 | 257.71 | 522.44 | 0.999761 | 4 | 108045.6 | 108166.5 | 0 | 0.000000 |
| 29 | 0.999467 | 7 | 258.7 | 524.44 | 0.999467 | 7 | 108046.7 | 108168.2 | 0 | 0.000000 |
| 30 | 0.999626 | 4 | 259.279 | 525.64 | 0.999626 | 4 | 108047.4 | 108169.1 | 0 | 0.000000 |
| 31 | 0.981389 | 65 | 266.917 | 541.05 | 0.982448 | 62 | 108349.4 | 108476.2 | 3 | -0.001059 |
| 32 | 0.989139 | 49 | 275.829 | 558.77 | 0.991273 | 39 | 108456.6 | 108586.3 | 10 | -0.002134 |
| 33 | 0.974169 | 44 | 281.858 | 571.19 | 0.975283 | 43 | 108471.4 | 108604.1 | 1 | -0.001114 |
| 34 | 0.839711 | 179 | 309.922 | 628.46 | 0.852353 | 177 | 132692.2 | 132841.1 | 2 | -0.012642 |
| 35 | 0.930042 | 136 | 332.412 | 674.76 | 0.937163 | 114 | 146987.8 | 147145.9 | 22 | -0.007121 |
| 36 | 0.928549 | 142 | 353.925 | 718.72 | 0.932467 | 133 | 169458.1 | 169626.8 | 9 | -0.003918 |
| 37 | 0.999726 | 5 | 354.96 | 720.73 | 0.999726 | 5 | 169458.9 | 169628 | 0 | 0.000000 |
| 38 | 0.999039 | 15 | 357.184 | 725.43 | 0.999162 | 11 | 169461.2 | 169631 | 4 | -0.000123 |
| 39 | 0.901807 | 113 | 374.49 | 760.9 | 0.913648 | 106 | 169713.7 | 169891.7 | 7 | -0.011841 |
| 40 | 0.979057 | 66 | 383.054 | 778.02 | 0.982118 | 60 | 169973.7 | 170156.4 | 6 | -0.003061 |
| 41 | 0.996021 | 29 | 388.055 | 788.03 | 0.996943 | 24 | 169988.1 | 170172.4 | 5 | -0.000922 |

| DMUs | $EFF_{OY}$ | $R_{0Y}$ | $CPU_{OY}$ | $TIME_{OY}$ | $EFF_{KT}$ | $R_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 42 | 1 | 0 | 388.101 | 788.12 | 1 | 0 | 169988.1 | 170172.5 | 0 | 0.000000 |
| 43 | 0.912783 | 139 | 413.959 | 840.6 | 0.935093 | 98 | 170489.5 | 170681.5 | 41 | -0.022310 |
| 44 | 0.992423 | 13 | 415.458 | 843.6 | 0.993061 | 13 | 170491.6 | 170684.4 | 0 | -0.000638 |
| 45 | 0.872303 | 130 | 431.132 | 875.14 | 0.895833 | 117 | 170963.2 | 171165 | 13 | -0.023530 |
| 46 | 0.986534 | 61 | 440.551 | 894.45 | 0.988809 | 51 | 171259.2 | 171464.7 | 10 | -0.002275 |
| 47 | 0.980521 | 55 | 448.278 | 910.45 | 0.982997 | 52 | 171324.9 | 171534.7 | 3 | -0.002476 |
| 48 | 0.960871 | 70 | 460.193 | 935.11 | 0.964834 | 57 | 171369.8 | 171584.2 | 13 | -0.003963 |
| 49 | 0.975413 | 37 | 464.818 | 944.33 | 0.976778 | 37 | 171377.8 | 171595.1 | 0 | -0.001365 |
| 50 | 0.901363 | 105 | 478.868 | 972.75 | 0.919213 | 89 | 171472 | 171696.3 | 16 | -0.017850 |
| 51 | 0.959136 | 60 | 487 | 989.95 | 0.962488 | 55 | 171502 | 171730.8 | 5 | -0.003352 |
| 52 | 0.988997 | 38 | 493.184 | 1002.67 | 0.990443 | 31 | 171519.5 | 171750.6 | 7 | -0.001446 |
| 53 | 0.996917 | 26 | 497.305 | 1011.08 | 0.997291 | 22 | 171529.3 | 171761.9 | 4 | -0.000374 |
| 54 | 0.96815 | 56 | 503.692 | 1024.29 | 0.972621 | 47 | 171546.9 | 171783 | 9 | -0.004471 |
| 55 | 0.998947 | 6 | 504.147 | 1025.19 | 0.998947 | 6 | 171547.4 | 171784.2 | 0 | 0.000000 |
| 56 | 0.982897 | 25 | 507.02 | 1031 | 0.983069 | 25 | 171551.2 | 171789.9 | 0 | -0.000172 |
| 57 | 0.952961 | 75 | 514.733 | 1046.51 | 0.962186 | 54 | 171574.2 | 171817.2 | 21 | -0.009225 |
| 58 | 0.979836 | 41 | 521.243 | 1059.54 | 0.980925 | 39 | 171590.3 | 171836.3 | 2 | -0.001089 |
| 59 | 0.99971 | 8 | 522.657 | 1062.35 | 0.999716 | 7 | 171591.6 | 171838.1 | 1 | -0.000006 |
| 60 | 0.948718 | 88 | 535.22 | 1088.18 | 0.954973 | 82 | 172447.9 | 172700.4 | 6 | -0.006255 |
| 61 | 0.988085 | 54 | 545.476 | 1109.11 | 0.98961 | 51 | 172934.6 | 173191.3 | 3 | -0.001525 |
| 62 | 0.986128 | 29 | 549.504 | 1117.32 | 0.988492 | 26 | 172941 | 173199.6 | 3 | -0.002364 |
| 63 | 0.98489 | 34 | 555.296 | 1128.52 | 0.987541 | 29 | 172949.9 | 173210.9 | 5 | -0.002651 |
| 64 | 0.865057 | 159 | 577.193 | 1174.18 | - | - | - | - | - | - |

| DMUs | $EFF_{OY}$ | $R_{0Y}$ | $CPU_{OY}$ | $TIME_{OY}$ | $EFF_{KT}$ | $R_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 65 | 1 | 0 | 577.247 | 1174.29 | - | - | - | - | - | - |
| 66 | 0.963132 | 129 | 590.311 | 1201.31 | - | - | - | - | - | - |
| 67 | 0.954029 | 78 | 601.68 | 1224.72 | - | - | - | - | - | - |
| 68 | 0.994841 | 24 | 604.7 | 1231.13 | - | - | - | - | - | - |
| 69 | 0.993896 | 30 | 608.634 | 1239.23 | - | - | - | - | - | - |
| 70 | 0.963778 | 60 | 617.085 | 1256.44 | - | - | - | - | - | - |
| 71 | 0.997526 | 17 | 620.119 | 1263.05 | - | - | - | - | - | - |
| 72 | 0.990137 | 15 | 621.847 | 1266.55 | - | - | - | - | - | - |
| 73 | 0.982609 | 24 | 624.267 | 1271.55 | - | - | - | - | - | - |
| 74 | 0.950798 | 62 | 633.458 | 1290.57 | - | - | - | - | - | - |
| 75 | 0.980994 | 60 | 643.606 | 1311 | - | - | - | - | - | - |
| 76 | 0.999891 | 2 | 644.098 | 1312.01 | - | - | - | - | - | - |
| 77 | 0.914999 | 128 | 656.589 | 1337.21 | - | - | - | - | - | - |
| 78 | 0.881182 | 179 | 683.152 | 1392.28 | - | - | - | - | - | - |
| 79 | 0.975495 | 45 | 688.926 | 1404.2 | - | - | - | - | - | - |
| 80 | 0.979386 | 73 | 702.34 | 1431.75 | - | - | - | - | - | - |
| 81 | 0.912821 | 148 | 727.382 | 1482.81 | - | - | - | - | - | - |
| 82 | 0.907548 | 118 | 744.592 | 1518.07 | - | - | - | - | - | - |
| 83 | 0.986047 | 42 | 751.89 | 1533.09 | - | - | - | - | - | - |
| 84 | 0.971401 | 66 | 760.572 | 1550.9 | - | - | - | - | - | - |
| 85 | 0.991822 | 13 | 761.65 | 1553.1 | - | - | - | - | - | - |
| 86 | 0.974947 | 68 | 774.363 | 1579.03 | - | - | - | - | - | - |
| 87 | 0.959159 | 104 | 792.783 | 1617.09 | - | - | - | - | - | - |

| DMUs | $EFF_{OY}$ | $R_{0Y}$ | $CPU_{OY}$ | $TIME_{OY}$ | $EFF_{KT}$ | $R_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 88 | 0.977421 | 82 | 807.091 | 1646.43 | - | - | - | - | - | - |
| 89 | 0.985384 | 28 | 810.562 | 1653.53 | - | - | - | - | - | - |
| 90 | 0.986638 | 19 | 812.594 | 1658.03 | - | - | - | - | - | - |
| 91 | 0.999947 | 1 | 812.806 | 1658.44 | - | - | - | - | - | - |
| 92 | 0.995432 | 28 | 818.452 | 1670.05 | - | - | - | - | - | - |
| 93 | 0.994121 | 40 | 822.556 | 1678.67 | - | - | - | - | - | - |
| 94 | 0.998924 | 13 | 826.177 | 1685.57 | - | - | - | - | - | - |
| 95 | 0.990273 | 33 | 831.19 | 1695.8 | - | - | - | - | - | - |
| 96 | 0.983104 | 57 | 840.903 | 1715.82 | - | - | - | - | - | - |
| 97 | 0.980348 | 64 | 851.257 | 1737.04 | - | - | - | - | - | - |
| 98 | 0.912269 | 155 | 872.829 | 1782.27 | - | - | - | - | - | - |
| 99 | 0.98959 | 48 | 881.316 | 1799.42 | - | - | - | - | - | - |
| 100 | 0.988034 | 55 | 889.553 | 1816.95 | - | - | - | - | - | - |
| 101 | 0.960204 | 99 | 906.126 | 1851.58 | - | - | - | - | - | - |
| 102 | 0.928248 | 118 | 922.188 | 1885.12 | - | - | - | - | - | - |
| 103 | 0.945635 | 108 | 936.266 | 1914.15 | - | - | - | - | - | - |
| 104 | 0.888911 | 138 | 956.731 | 1956.74 | - | - | - | - | - | - |
| 105 | 0.876183 | 188 | 993.023 | 2031.7 | - | - | - | - | - | - |
| 106 | 0.994765 | 40 | 999.027 | 2044.23 | - | - | - | - | - | - |
| 107 | 1 | 0 | 999.081 | 2044.33 | - | - | - | - | - | - |
| 108 | 1 | 0 | 999.131 | 2044.43 | - | - | - | - | - | - |
| 109 | 1 | 0 | 999.182 | 2044.53 | - | - | - | - | - | - |
| 110 | 0.871202 | 165 | 1029.732 | 2107.37 | - | - | - | - | - | - |

| DMUs | $EFF_{OY}$ | $R_{0Y}$ | $CPU_{OY}$ | $TIME_{OY}$ | $EFF_{KT}$ | $R_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 111 | 0.986988 | 36 | 1034.025 | 2116.28 | - | - | - | - | - | - |
| 112 | 0.945888 | 113 | 1050.478 | 2150.64 | - | - | - | - | - | - |
| 113 | 0.901535 | 101 | 1063.935 | 2179.06 | - | - | - | - | - | - |
| 114 | 0.936948 | 87 | 1074.418 | 2201.08 | - | - | - | - | - | - |
| 115 | 0.86934 | 196 | 1094.261 | 2241.85 | - | - | - | - | - | - |
| 116 | 0.999897 | 4 | 1094.941 | 2243.25 | - | - | - | - | - | - |
| 117 | 0.950936 | 109 | 1111.949 | 2278.52 | - | - | - | - | - | - |
| 118 | 0.876646 | 186 | 1139.78 | 2337.19 | - | - | - | - | - | - |
| 119 | 0.947506 | 113 | 1159.273 | 2377.04 | - | - | - | - | - | - |
| 120 | 1 | 0 | 1159.309 | 2377.14 | - | - | - | - | - | - |
| 121 | 0.991258 | 22 | 1162.917 | 2384.87 | - | - | - | - | - | - |
| 122 | 0.967565 | 78 | 1173.92 | 2407.41 | - | - | - | - | - | - |
| 123 | 0.953888 | 58 | 1180.094 | 2420.33 | - | - | - | - | - | - |
| 124 | 0.999232 | 10 | 1181.987 | 2424.13 | - | - | - | - | - | - |
| 125 | 0.915601 | 146 | 1204.508 | 2471.09 | - | - | - | - | - | - |
| 126 | 0.911968 | 158 | 1252.224 | 2542.83 | - | - | - | - | - | - |
| 127 | 0.995413 | 8 | 1253.074 | 2544.53 | - | - | - | - | - | - |
| 128 | 0.977737 | 39 | 1256.954 | 2552.44 | - | - | - | - | - | - |
| 129 | 0.958073 | 87 | 1270.211 | 2579.85 | - | - | - | - | - | - |
| 130 | 0.999903 | 2 | 1270.509 | 2580.46 | - | - | - | - | - | - |
| 131 | 0.98825 | 30 | 1275.316 | 2590.27 | - | - | - | - | - | - |
| 132 | 1 | 0 | 1275.368 | 2590.39 | - | - | - | - | - | - |
| 133 | 0.890191 | 149 | 1301.497 | 2644.94 | - | - | - | - | - | - |

| DMUs | $EFF_{OY}$ | $R_{0Y}$ | $CPU_{OY}$ | $TIME_{OY}$ | $EFF_{KT}$ | $R_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 134 | 0.990766 | 17 | 1303.654 | 2649.45 | - | - | - | - | - | - |
| 135 | 0.996266 | 19 | 1306.505 | 2655.47 | - | - | - | - | - | - |
| 136 | 0.987696 | 47 | 1315.241 | 2673.78 | - | - | - | - | - | - |
| 137 | 0.987602 | 41 | 1322.218 | 2688 | - | - | - | - | - | - |
| 138 | 0.998777 | 16 | 1324.758 | 2693.31 | - | - | - | - | - | - |
| 139 | 0.999843 | 3 | 1325.249 | 2694.41 | - | - | - | - | - | - |
| 140 | 0.997614 | 27 | 1329.299 | 2702.61 | - | - | - | - | - | - |
| 141 | 0.916641 | 99 | 1346.575 | 2739.38 | - | - | - | - | - | - |
| 142 | 0.998794 | 15 | 1349.957 | 2746.37 | - | - | - | - | - | - |
| 143 | 0.900023 | 161 | 1379.876 | 2808.25 | - | - | - | - | - | - |
| 144 | 0.995184 | 24 | 1383.779 | 2816.45 | - | - | - | - | - | - |
| 145 | 0.974139 | 83 | 1399.342 | 2848.92 | - | - | - | - | - | - |
| 146 | 0.949488 | 69 | 1407.938 | 2867.06 | - | - | - | - | - | - |
| 147 | 0.985914 | 22 | 1410.404 | 2872.26 | - | - | - | - | - | - |
| 148 | 1 | 0 | 1410.449 | 2872.36 | - | - | - | - | - | - |
| 149 | 0.958691 | 105 | 1425.392 | 2903.61 | - | - | - | - | - | - |
| 150 | 0.947598 | 79 | 1434.072 | 2921.62 | - | - | - | - | - | - |
| 151 | 0.838672 | 186 | 1458.823 | 2974.61 | - | - | - | - | - | - |
| 152 | 0.996878 | 23 | 1462.315 | 2982.12 | - | - | - | - | - | - |
| 153 | 0.917601 | 113 | 1474.192 | 3006.85 | - | - | - | - | - | - |
| 154 | 1 | 0 | 1474.235 | 3006.94 | - | - | - | - | - | - |
| 155 | 0.934613 | 74 | 1484.105 | 3027.66 | - | - | - | - | - | - |
| 156 | 0.999675 | 6 | 1485.372 | 3030.17 | - | - | - | - | - | - |

| DMUs | $EFF_{OY}$ | $R_{0Y}$ | $CPU_{OY}$ | $TIME_{OY}$ | $EFF_{KT}$ | $R_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 157 | 0.967794 | 91 | 1501.979 | 3064.93 | - | - | - | - | - | - |
| 158 | 0.997139 | 9 | 1502.546 | 3066.13 | - | - | - | - | - | - |
| 159 | 1 | 0 | 1502.592 | 3066.22 | - | - | - | - | - | - |
| 160 | 0.998148 | 15 | 1504.55 | 3070.43 | - | - | - | - | - | - |
| 161 | 0.959671 | 73 | 1515.536 | 3093.25 | - | - | - | - | - | - |
| 162 | 0.954737 | 108 | 1531.366 | 3126.81 | - | - | - | - | - | - |
| 163 | 0.861109 | 173 | 1553.006 | 3172.96 | - | - | - | - | - | - |
| 164 | 0.941785 | 94 | 1566.948 | 3202.39 | - | - | - | - | - | - |
| 165 | 0.997583 | 17 | 1569.873 | 3208.92 | - | - | - | - | - | - |
| 166 | 1 | 0 | 1569.906 | 3209.01 | - | - | - | - | - | - |
| 167 | 0.968351 | 64 | 1579.17 | 3228.64 | - | - | - | - | - | - |
| 168 | 0.961415 | 95 | 1589.351 | 3250.47 | - | - | - | - | - | - |
| 169 | 0.864282 | 193 | 1611.309 | 3296.94 | - | - | - | - | - | - |
| 170 | 0.998646 | 4 | 1611.756 | 3297.84 | - | - | - | - | - | - |
| 171 | 0.806721 | 204 | 1639.785 | 3358.1 | - | - | - | - | - | - |
| 172 | 0.997775 | 8 | 1640.269 | 3359.29 | - | - | - | - | - | - |
| 173 | 0.998172 | 12 | 1641.936 | 3362.7 | - | - | - | - | - | - |
| 174 | 0.820052 | 189 | 1666.765 | 3415.87 | - | - | - | - | - | - |
| 175 | 0.983764 | 42 | 1673.066 | 3429.48 | - | - | - | - | - | - |
| 176 | 0.998323 | 18 | 1675.579 | 3434.98 | - | - | - | - | - | - |
| 177 | 0.993106 | 12 | 1676.881 | 3437.69 | - | - | - | - | - | - |
| 178 | 0.868512 | 149 | 1696.092 | 3478.73 | - | - | - | - | - | - |
| 179 | 1 | 0 | 1696.141 | 3478.83 | - | - | - | - | - | - |

| DMUs | $EFF_{OY}$ | $R_{0Y}$ | $CPU_{OY}$ | $TIME_{OY}$ | $EFF_{KT}$ | $R_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 180 | 0.924775 | 106 | 1707.515 | 3503.56 | - | - | - | - | - | - |
| 181 | 1 | 0 | 1707.561 | 3503.66 | - | - | - | - | - | - |
| 182 | 0.870543 | 157 | 1729.834 | 3550.11 | - | - | - | - | - | - |
| 183 | 0.920277 | 141 | 1748.913 | 3590.97 | - | - | - | - | - | - |
| 184 | 0.998842 | 16 | 1751.73 | 3596.98 | - | - | - | - | - | - |
| 185 | 0.983545 | 35 | 1756.819 | 3608.1 | - | - | - | - | - | - |
| 186 | 0.964577 | 84 | 1765.661 | 3626.61 | - | - | - | - | - | - |
| 187 | 0.929623 | 117 | 1783.312 | 3664.36 | - | - | - | - | - | - |
| 188 | 0.992565 | 48 | 1789.333 | 3677.17 | - | - | - | - | - | - |
| 189 | 0.999947 | 2 | 1789.72 | 3677.97 | - | - | - | - | - | - |
| 190 | 0.887808 | 170 | 1813.564 | 3728.58 | - | - | - | - | - | - |
| 191 | 1 | 0 | 1813.617 | 3728.68 | - | - | - | - | - | - |
| 192 | 0.903573 | 126 | 1829.799 | 3763.2 | - | - | - | - | - | - |
| 193 | 0.886158 | 173 | 1857.174 | 3822.28 | - | - | - | - | - | - |
| 194 | 0.96376 | 92 | 1873.756 | 3857.04 | - | - | - | - | - | - |
| 195 | 0.9983 | 5 | 1874.199 | 3858.04 | - | - | - | - | - | - |
| 196 | 0.985739 | 21 | 1876.466 | 3862.95 | - | - | - | - | - | - |
| 197 | 0.994137 | 19 | 1879.523 | 3869.55 | - | - | - | - | - | - |
| 198 | 0.936277 | 130 | 1894.738 | 3903.08 | - | - | - | - | - | - |
| 199 | 0.847558 | 170 | 1914.995 | 3946.99 | - | - | - | - | - | - |
| 200 | 0.999835 | 3 | 1915.589 | 3948.31 | - | - | - | - | - | - |
| 201 | 0.999453 | 4 | 1916.169 | 3949.51 | - | - | - | - | - | - |
| 202 | 0.932849 | 101 | 1932.37 | 3983.67 | - | - | - | - | - | - |

| DMUs | $EFF_{OY}$ | $R_{0Y}$ | $CPU_{OY}$ | $TIME_{OY}$ | $EFF_{KT}$ | $R_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 203 | 0.999212 | 9 | 1934.16 | 3987.48 | - | - | - | - | - | - |
| 204 | 0.988549 | 17 | 1936.077 | 3991.28 | - | - | - | - | - | - |
| 205 | 0.810501 | 199 | 1956.251 | 4035.3 | - | - | - | - | - | - |
| 206 | 0.929839 | 138 | 1972.52 | 4070.33 | - | - | - | - | - | - |
| 207 | 0.991491 | 20 | 1974.891 | 4075.54 | - | - | - | - | - | - |
| 208 | 0.998558 | 11 | 1976.527 | 4078.94 | - | - | - | - | - | - |
| 209 | 0.969795 | 80 | 1990.038 | 4107.79 | - | - | - | - | - | - |
| 210 | 0.899955 | 152 | 2013.785 | 4158.55 | - | - | - | - | - | - |
| 211 | 0.985101 | 48 | 2020.66 | 4173.66 | - | - | - | - | - | - |
| 212 | 0.99993 | 1 | 2020.838 | 4174.06 | - | - | - | - | - | - |
| 213 | 0.984026 | 66 | 2033.578 | 4201.41 | - | - | - | - | - | - |
| 214 | 0.960586 | 102 | 2043.062 | 4222.12 | - | - | - | - | - | - |
| 215 | 0.924234 | 134 | 2068.803 | 4277.26 | - | - | - | - | - | - |
| 216 | 0.987339 | 43 | 2077.221 | 4295.21 | - | - | - | - | - | - |
| 217 | 0.998299 | 13 | 2079.011 | 4299.21 | - | - | - | - | - | - |
| 218 | 0.985712 | 42 | 2084.419 | 4310.32 | - | - | - | - | - | - |
| 219 | 0.996188 | 17 | 2086.679 | 4315.22 | - | - | - | - | - | - |
| 220 | 0.979412 | 73 | 2100.549 | 4344.89 | - | - | - | - | - | - |
| 221 | 0.990308 | 25 | 2104.023 | 4352.1 | - | - | - | - | - | - |
| 222 | 0.954025 | 105 | 2118.681 | 4383.54 | - | - | - | - | - | - |
| 223 | 0.999403 | 5 | 2119.084 | 4384.44 | - | - | - | - | - | - |
| 224 | 0.998837 | 12 | 2120.779 | 4388.24 | - | - | - | - | - | - |
| 225 | 0.995477 | 13 | 2122.328 | 4391.85 | - | - | - | - | - | - |

| DMUs | $EFF_{OY}$ | $R_{0Y}$ | $CPU_{OY}$ | $TIME_{OY}$ | $EFF_{KT}$ | $R_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 226 | 0.976308 | 56 | 2129.656 | 4408.07 | - | - | - | - | - | - |
| 227 | 0.983953 | 48 | 2136.609 | 4423.11 | - | - | - | - | - | - |
| 228 | 0.99126 | 13 | 2137.618 | 4425.61 | - | - | - | - | - | - |
| 229 | 0.979222 | 38 | 2144.143 | 4439.64 | - | - | - | - | - | - |
| 230 | 0.987384 | 19 | 2145.876 | 4443.45 | - | - | - | - | - | - |
| 231 | 0.968666 | 57 | 2153.372 | 4460.17 | - | - | - | - | - | - |
| 232 | 1 | 0 | 2153.426 | 4460.27 | - | - | - | - | - | - |
| 233 | 0.980638 | 31 | 2157.199 | 4468.68 | - | - | - | - | - | - |
| 234 | 0.991835 | 21 | 2159.622 | 4473.88 | - | - | - | - | - | - |
| 235 | 0.997684 | 23 | 2163.346 | 4481.7 | - | - | - | - | - | - |
| 236 | 0.996537 | 16 | 2166.016 | 4487.4 | - | - | - | - | - | - |
| 237 | 1 | 0 | 2166.07 | 4487.5 | - | - | - | - | - | - |
| 238 | 0.965618 | 86 | 2177.183 | 4511.24 | - | - | - | - | - | - |
| 239 | 0.999341 | 8 | 2178.534 | 4514.04 | - | - | - | - | - | - |
| 240 | 0.997727 | 16 | 2181.711 | 4520.75 | - | - | - | - | - | - |
| 241 | 0.994767 | 15 | 2183.466 | 4524.56 | - | - | - | - | - | - |
| 242 | 0.991386 | 42 | 2189.594 | 4537.87 | - | - | - | - | - | - |
| 243 | 0.988089 | 31 | 2193.359 | 4545.88 | - | - | - | - | - | - |
| 244 | 0.987227 | 55 | 2202.125 | 4564.6 | - | - | - | - | - | - |
| 245 | 0.98674 | 22 | 2204.9 | 4570.71 | - | - | - | - | - | - |
| 246 | 0.958046 | 91 | 2217.411 | 4597.35 | - | - | - | - | - | - |
| 247 | 0.964166 | 99 | 2230.24 | 4625.66 | - | - | - | - | - | - |
| 248 | 0.990876 | 28 | 2233.681 | 4633.16 | - | - | - | - | - | - |

| DMUs | $EFF_{OY}$ | $R_{0Y}$ | $CPU_{OY}$ | $TIME_{OY}$ | $EFF_{KT}$ | $R_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 249 | 0.971676 | 50 | 2240.652 | 4648.59 | - | - | - | - | - | - |
| 250 | 0.997424 | 25 | 2244.771 | 4657.5 | - | - | - | - | - | - |

APPENDIX B.

Table B.1 Logistics Performance Index Data

| SEQ | COUNTRY | CODE | CUSTOMS | INFRASTRUCTURE | INTERNATIONAL SHIPMENTS | LOGISTICS QUALITY AND COMPETENCE | TRACKING AND TRACING | TIMELINESS |
|---|---|---|---|---|---|---|---|---|
| 1 | Germany | CODE1 | 4.10 | 4.32 | 3.74 | 4.12 | 4.17 | 4.36 |
| 2 | Netherlands | CODE2 | 3.96 | 4.23 | 3.64 | 4.13 | 4.07 | 4.34 |
| 3 | Belgium | CODE3 | 3.80 | 4.10 | 3.80 | 4.11 | 4.11 | 4.39 |
| 4 | United Kingdom | CODE4 | 3.94 | 4.16 | 3.63 | 4.03 | 4.08 | 4.33 |
| 5 | Singapore | CODE5 | 4.01 | 4.28 | 3.70 | 3.97 | 3.90 | 4.25 |
| 6 | Sweden | CODE6 | 3.75 | 4.09 | 3.76 | 3.98 | 3.98 | 4.26 |
| 7 | Norway | CODE7 | 4.21 | 4.19 | 3.42 | 4.19 | 3.50 | 4.36 |
| 8 | Luxembourg | CODE8 | 3.82 | 3.91 | 3.82 | 3.78 | 3.68 | 4.71 |
| 9 | United States | CODE9 | 3.73 | 4.18 | 3.45 | 3.97 | 4.14 | 4.14 |
| 10 | Japan | CODE10 | 3.78 | 4.16 | 3.52 | 3.93 | 3.95 | 4.24 |
| 11 | Ireland | CODE11 | 3.80 | 3.84 | 3.44 | 3.94 | 4.13 | 4.13 |
| 12 | Canada | CODE12 | 3.61 | 4.05 | 3.46 | 3.94 | 3.97 | 4.18 |
| 13 | France | CODE13 | 3.65 | 3.98 | 3.68 | 3.75 | 3.89 | 4.17 |
| 14 | Switzerland | CODE14 | 3.92 | 4.04 | 3.58 | 3.75 | 3.79 | 4.06 |
| 15 | Hong Kong SAR, China | CODE15 | 3.72 | 3.97 | 3.58 | 3.81 | 3.87 | 4.06 |
| 16 | Australia | CODE16 | 3.85 | 4.00 | 3.52 | 3.75 | 3.81 | 4.00 |
| 17 | Denmark | CODE17 | 3.79 | 3.82 | 3.65 | 3.74 | 3.36 | 4.39 |
| 18 | Spain | CODE18 | 3.63 | 3.77 | 3.51 | 3.83 | 3.54 | 4.07 |
| 19 | Taiwan | CODE19 | 3.55 | 3.64 | 3.71 | 3.60 | 3.79 | 4.02 |

| SEQ | COUNTRY | CODE | CUSTOMS | INFRASTRUCTURE | INTERNATIONAL SHIPMENTS | LOGISTICS QUALITY AND COMPETENCE | TRACKING AND TRACING | TIMELINESS |
|---|---|---|---|---|---|---|---|---|
| 20 | Italy | CODE20 | 3.36 | 3.78 | 3.54 | 3.62 | 3.84 | 4.05 |
| 21 | Korea, Rep. | CODE21 | 3.47 | 3.79 | 3.44 | 3.66 | 3.69 | 4.00 |
| 22 | Austria | CODE22 | 3.53 | 3.64 | 3.26 | 3.56 | 3.93 | 4.04 |
| 23 | New Zealand | CODE23 | 3.92 | 3.67 | 3.67 | 3.56 | 3.33 | 3.72 |
| 24 | Finland | CODE24 | 3.89 | 3.52 | 3.52 | 3.72 | 3.31 | 3.80 |
| 25 | Malaysia | CODE25 | 3.37 | 3.56 | 3.64 | 3.47 | 3.58 | 3.92 |
| 26 | Portugal | CODE26 | 3.26 | 3.37 | 3.43 | 3.71 | 3.71 | 3.87 |
| 27 | United Arab Emirates | CODE27 | 3.42 | 3.70 | 3.20 | 3.50 | 3.57 | 3.92 |
| 28 | China | CODE28 | 3.21 | 3.67 | 3.50 | 3.46 | 3.50 | 3.87 |
| 29 | Qatar | CODE29 | 3.21 | 3.44 | 3.55 | 3.55 | 3.47 | 3.87 |
| 30 | Turkey | CODE30 | 3.23 | 3.53 | 3.18 | 3.64 | 3.77 | 3.68 |
| 31 | Poland | CODE31 | 3.26 | 3.08 | 3.46 | 3.47 | 3.54 | 4.13 |
| 32 | Czech Republic | CODE32 | 3.24 | 3.29 | 3.59 | 3.51 | 3.56 | 3.73 |
| 33 | Hungary | CODE33 | 2.97 | 3.18 | 3.40 | 3.33 | 3.82 | 4.06 |
| 34 | South Africa | CODE34 | 3.11 | 3.20 | 3.45 | 3.62 | 3.30 | 3.88 |
| 35 | Thailand | CODE35 | 3.21 | 3.40 | 3.30 | 3.29 | 3.45 | 3.96 |
| 36 | Latvia | CODE36 | 3.22 | 3.03 | 3.38 | 3.21 | 3.50 | 4.06 |
| 37 | Iceland | CODE37 | 3.54 | 3.34 | 3.15 | 3.46 | 3.38 | 3.51 |
| 38 | Slovenia | CODE38 | 3.11 | 3.35 | 3.05 | 3.51 | 3.51 | 3.82 |
| 39 | Estonia | CODE39 | 3.40 | 3.34 | 3.34 | 3.27 | 3.20 | 3.55 |
| 40 | Romania | CODE40 | 2.83 | 2.77 | 3.32 | 3.20 | 3.39 | 4.00 |
| 41 | Israel | CODE41 | 3.10 | 3.11 | 2.71 | 3.35 | 3.20 | 4.18 |
| 42 | Chile | CODE42 | 3.17 | 3.17 | 3.12 | 3.19 | 3.30 | 3.59 |
| 43 | Slovak Republic | CODE43 | 2.89 | 3.22 | 3.30 | 3.16 | 3.02 | 3.94 |

| SEQ | COUNTRY | CODE | CUSTOMS | INFRASTRUCTURE | INTERNATIONAL SHIPMENTS | LOGISTICS QUALITY AND COMPETENCE | TRACKING AND TRACING | TIMELINESS |
|---|---|---|---|---|---|---|---|---|
| 44 | Greece | CODE44 | 3.36 | 3.17 | 2.97 | 3.23 | 3.03 | 3.50 |
| 45 | Panama | CODE45 | 3.15 | 3.00 | 3.18 | 2.87 | 3.34 | 3.63 |
| 46 | Lithuania | CODE46 | 3.04 | 3.18 | 3.10 | 2.99 | 3.17 | 3.60 |
| 47 | Bulgaria | CODE47 | 2.75 | 2.94 | 3.31 | 3.00 | 2.88 | 4.04 |
| 48 | Vietnam | CODE48 | 2.81 | 3.11 | 3.22 | 3.09 | 3.19 | 3.49 |
| 49 | Saudi Arabia | CODE49 | 2.86 | 3.34 | 2.93 | 3.11 | 3.15 | 3.55 |
| 50 | Mexico | CODE50 | 2.69 | 3.04 | 3.19 | 3.12 | 3.14 | 3.57 |
| 51 | Malta | CODE51 | 3.00 | 3.08 | 3.23 | 3.00 | 3.15 | 3.15 |
| 52 | Bahrain | CODE52 | 3.29 | 3.04 | 3.04 | 3.04 | 3.29 | 2.80 |
| 53 | Indonesia | CODE53 | 2.87 | 2.92 | 2.87 | 3.21 | 3.11 | 3.53 |
| 54 | India | CODE54 | 2.72 | 2.88 | 3.20 | 3.03 | 3.11 | 3.51 |
| 55 | Croatia | CODE55 | 2.95 | 2.92 | 2.98 | 3.00 | 3.11 | 3.37 |
| 56 | Kuwait | CODE56 | 2.69 | 3.16 | 2.76 | 2.96 | 3.16 | 3.39 |
| 57 | Philippines | CODE57 | 3.00 | 2.60 | 3.33 | 2.93 | 3.00 | 3.07 |
| 58 | Cyprus | CODE58 | 2.88 | 2.87 | 3.01 | 2.92 | 3.00 | 3.31 |
| 59 | Oman | CODE59 | 2.63 | 2.88 | 3.41 | 2.84 | 2.84 | 3.29 |
| 60 | Argentina | CODE60 | 2.55 | 2.83 | 2.96 | 2.93 | 3.15 | 3.49 |
| 61 | Ukraine | CODE61 | 2.69 | 2.65 | 2.95 | 2.84 | 3.20 | 3.51 |
| 62 | Egypt, Arab Rep. | CODE62 | 2.85 | 2.86 | 2.87 | 2.99 | 3.23 | 2.99 |
| 63 | Serbia | CODE63 | 2.37 | 2.73 | 3.12 | 3.02 | 2.94 | 3.55 |
| 64 | El Salvador | CODE64 | 2.93 | 2.63 | 3.20 | 3.16 | 3.00 | 2.75 |
| 65 | Brazil | CODE65 | 2.48 | 2.93 | 2.80 | 3.05 | 3.03 | 3.39 |
| 66 | Bahamas, The | CODE66 | 3.00 | 2.74 | 2.96 | 2.92 | 2.64 | 3.19 |
| 67 | Montenegro | CODE67 | 2.83 | 2.84 | 3.15 | 2.45 | 2.76 | 3.19 |

| SEQ | COUNTRY | CODE | CUSTOMS | INFRASTRUCTURE | INTERNATIONAL SHIPMENTS | LOGISTICS QUALITY AND COMPETENCE | TRACKING AND TRACING | TIMELINESS |
|---|---|---|---|---|---|---|---|---|
| 68 | Jordan | CODE68 | 2.60 | 2.59 | 2.96 | 2.94 | 2.67 | 3.46 |
| 69 | Dominican Republic | CODE69 | 2.58 | 2.61 | 2.93 | 2.91 | 2.91 | 3.18 |
| 70 | Jamaica | CODE70 | 2.88 | 2.84 | 2.79 | 2.72 | 2.72 | 3.14 |
| 71 | Peru | CODE71 | 2.47 | 2.72 | 2.94 | 2.78 | 2.81 | 3.30 |
| 72 | Pakistan | CODE72 | 2.84 | 2.67 | 3.08 | 2.79 | 2.73 | 2.79 |
| 73 | Malawi | CODE73 | 2.79 | 3.04 | 2.63 | 2.86 | 2.63 | 2.99 |
| 74 | Kenya | CODE74 | 1.96 | 2.40 | 3.15 | 2.65 | 3.03 | 3.58 |
| 75 | Nigeria | CODE75 | 2.35 | 2.56 | 2.63 | 2.70 | 3.16 | 3.46 |
| 76 | Venezuela, RB | CODE76 | 2.39 | 2.61 | 2.94 | 2.76 | 2.92 | 3.18 |
| 77 | Guatemala | CODE77 | 2.75 | 2.54 | 2.87 | 2.68 | 2.68 | 3.24 |
| 78 | Paraguay | CODE78 | 2.49 | 2.46 | 2.83 | 2.76 | 2.89 | 3.22 |
| 79 | Côte d'Ivoire | CODE79 | 2.33 | 2.41 | 2.87 | 2.62 | 2.97 | 3.31 |
| 80 | Rwanda | CODE80 | 2.50 | 2.32 | 2.78 | 2.64 | 2.94 | 3.34 |
| 81 | Bosnia and Herzegovina | CODE81 | 2.41 | 2.55 | 2.78 | 2.73 | 2.55 | 3.44 |
| 82 | Maldives | CODE82 | 2.95 | 2.56 | 2.92 | 2.79 | 2.70 | 2.51 |
| 83 | Cambodia | CODE83 | 2.67 | 2.58 | 2.83 | 2.67 | 2.92 | 2.75 |
| 84 | Sco Tomi and Principe | CODE84 | 2.42 | 2.59 | 2.95 | 2.50 | 3.13 | 2.77 |
| 85 | Lebanon | CODE85 | 2.29 | 2.53 | 2.53 | 2.89 | 3.22 | 2.89 |
| 86 | Ecuador | CODE86 | 2.49 | 2.50 | 2.79 | 2.61 | 2.67 | 3.18 |
| 87 | Costa Rica | CODE87 | 2.39 | 2.43 | 2.63 | 2.86 | 2.83 | 3.04 |
| 88 | Kazakhstan | CODE88 | 2.33 | 2.38 | 2.68 | 2.72 | 2.83 | 3.24 |
| 89 | Sri Lanka | CODE89 | 2.56 | 2.23 | 2.56 | 2.91 | 2.76 | 3.12 |
| 90 | Russian Federation | CODE90 | 2.20 | 2.59 | 2.64 | 2.74 | 2.85 | 3.14 |
| 91 | Uruguay | CODE91 | 2.39 | 2.51 | 2.64 | 2.58 | 2.89 | 3.06 |

| SEQ | COUNTRY | CODE | CUSTOMS | INFRASTRUCTURE | INTERNATIONAL SHIPMENTS | LOGISTICS QUALITY AND COMPETENCE | TRACKING AND TRACING | TIMELINESS |
|---|---|---|---|---|---|---|---|---|
| 92 | Armenia | CODE92 | 2.63 | 2.38 | 2.75 | 2.75 | 2.50 | 3.00 |
| 93 | Namibia | CODE93 | 2.27 | 2.57 | 2.70 | 2.69 | 2.56 | 3.15 |
| 94 | Moldova | CODE94 | 2.46 | 2.55 | 3.14 | 2.44 | 2.35 | 2.89 |
| 95 | Nicaragua | CODE95 | 2.66 | 2.20 | 2.69 | 2.58 | 2.58 | 3.17 |
| 96 | Algeria | CODE96 | 2.71 | 2.54 | 2.54 | 2.54 | 2.54 | 3.04 |
| 97 | Colombia | CODE97 | 2.59 | 2.44 | 2.72 | 2.64 | 2.55 | 2.87 |
| 98 | Burkina Faso | CODE98 | 2.50 | 2.35 | 2.63 | 2.63 | 2.49 | 3.21 |
| 99 | Belarus | CODE99 | 2.50 | 2.55 | 2.74 | 2.46 | 2.51 | 3.05 |
| 100 | Ghana | CODE100 | 2.22 | 2.67 | 2.73 | 2.37 | 2.90 | 2.86 |
| 101 | Senegal | CODE101 | 2.61 | 2.30 | 3.03 | 2.53 | 2.65 | 2.53 |
| 102 | Liberia | CODE102 | 2.57 | 2.57 | 2.57 | 2.86 | 2.57 | 2.57 |
| 103 | Honduras | CODE103 | 2.70 | 2.24 | 2.79 | 2.47 | 2.61 | 2.79 |
| 104 | Ethiopia | CODE104 | 2.42 | 2.17 | 2.50 | 2.62 | 2.67 | 3.17 |
| 105 | Nepal | CODE105 | 2.31 | 2.26 | 2.64 | 2.50 | 2.72 | 3.06 |
| 106 | Solomon Islands | CODE106 | 2.49 | 2.46 | 2.22 | 2.72 | 2.72 | 2.96 |
| 107 | Burundi | CODE107 | 2.60 | 2.40 | 2.60 | 2.51 | 2.51 | 2.76 |
| 108 | Bangladesh | CODE108 | 2.09 | 2.11 | 2.82 | 2.64 | 2.45 | 3.18 |
| 109 | Benin | CODE109 | 2.64 | 2.35 | 2.69 | 2.35 | 2.45 | 2.85 |
| 110 | Tunisia | CODE110 | 2.02 | 2.30 | 2.91 | 2.42 | 2.42 | 3.16 |
| 111 | Fiji | CODE111 | 2.40 | 2.47 | 2.72 | 2.22 | 2.47 | 2.97 |
| 112 | Angola | CODE112 | 2.37 | 2.11 | 2.79 | 2.31 | 2.59 | 3.02 |
| 113 | Chad | CODE113 | 2.46 | 2.33 | 2.33 | 2.34 | 2.71 | 3.02 |
| 114 | Tajikistan | CODE114 | 2.35 | 2.36 | 2.73 | 2.47 | 2.47 | 2.74 |
| 115 | Mauritius | CODE115 | 2.25 | 2.50 | 2.63 | 2.48 | 2.34 | 2.88 |

| SEQ | COUNTRY | CODE | CUSTOMS | INFRASTRUCTURE | INTERNATIONAL SHIPMENTS | LOGISTICS QUALITY AND COMPETENCE | TRACKING AND TRACING | TIMELINESS |
|---|---|---|---|---|---|---|---|---|
| 116 | Georgia | CODE116 | 2.21 | 2.42 | 2.32 | 2.44 | 2.59 | 3.09 |
| 117 | Macedonia, FYR | CODE117 | 2.35 | 2.50 | 2.38 | 2.51 | 2.46 | 2.81 |
| 118 | Libya | CODE118 | 2.41 | 2.29 | 2.29 | 2.29 | 2.85 | 2.85 |
| 119 | Mali | CODE119 | 2.08 | 2.20 | 2.80 | 2.20 | 2.70 | 2.90 |
| 120 | Botswana | CODE120 | 2.38 | 2.23 | 2.42 | 2.58 | 2.40 | 2.94 |
| 121 | Bolivia | CODE121 | 2.40 | 2.17 | 2.35 | 2.68 | 2.68 | 2.60 |
| 122 | Guinea | CODE122 | 2.34 | 2.10 | 2.47 | 2.35 | 2.41 | 3.10 |
| 123 | Zambia | CODE123 | 2.54 | 2.31 | 2.13 | 2.47 | 2.47 | 2.91 |
| 124 | Guyana | CODE124 | 2.46 | 2.40 | 2.43 | 2.27 | 2.47 | 2.74 |
| 125 | Azerbaijan | CODE125 | 2.57 | 2.71 | 2.57 | 2.14 | 2.14 | 2.57 |
| 126 | Papua New Guinea | CODE126 | 2.40 | 2.23 | 2.47 | 2.47 | 2.27 | 2.73 |
| 127 | Guinea-Bissau | CODE127 | 2.43 | 2.29 | 2.29 | 2.57 | 2.29 | 2.71 |
| 128 | Comoros | CODE128 | 2.58 | 2.30 | 2.51 | 2.26 | 2.37 | 2.37 |
| 129 | Uzbekistan | CODE129 | 1.80 | 2.01 | 2.23 | 2.37 | 2.87 | 3.08 |
| 130 | Niger | CODE130 | 2.49 | 2.08 | 2.38 | 2.28 | 2.36 | 2.76 |
| 131 | Lao PDR | CODE131 | 2.45 | 2.21 | 2.50 | 2.31 | 2.20 | 2.65 |
| 132 | Madagascar | CODE132 | 2.06 | 2.15 | 2.38 | 2.33 | 2.29 | 3.07 |
| 133 | Lesotho | CODE133 | 2.22 | 2.35 | 2.48 | 2.23 | 2.35 | 2.60 |
| 134 | Central African Republic | CODE134 | 2.47 | 2.50 | 2.16 | 2.31 | 2.31 | 2.47 |
| 135 | Mongolia | CODE135 | 2.20 | 2.29 | 2.62 | 2.33 | 2.13 | 2.51 |
| 136 | Equatorial Guinea | CODE136 | 2.35 | 2.11 | 2.11 | 2.20 | 2.53 | 2.86 |
| 137 | Zimbabwe | CODE137 | 1.89 | 2.25 | 2.25 | 2.50 | 2.22 | 2.93 |
| 138 | Tanzania | CODE138 | 2.19 | 2.32 | 2.32 | 2.18 | 2.11 | 2.89 |
| 139 | Togo | CODE139 | 2.09 | 2.07 | 2.47 | 2.14 | 2.49 | 2.60 |

| SEQ | COUNTRY | CODE | CUSTOMS | INFRASTRUCTURE | INTERNATIONAL SHIPMENTS | LOGISTICS QUALITY AND COMPETENCE | TRACKING AND TRACING | TIMELINESS |
|---|---|---|---|---|---|---|---|---|
| 140 | Turkmenistan | CODE140 | 2.31 | 2.06 | 2.56 | 2.07 | 2.32 | 2.45 |
| 141 | Iraq | CODE141 | 1.98 | 2.18 | 2.31 | 2.15 | 2.31 | 2.85 |
| 142 | Cameroon | CODE142 | 1.86 | 1.85 | 2.20 | 2.52 | 2.52 | 2.80 |
| 143 | Bhutan | CODE143 | 2.09 | 2.18 | 2.38 | 2.48 | 2.28 | 2.28 |
| 144 | Haiti | CODE144 | 2.25 | 2.00 | 2.27 | 2.14 | 2.32 | 2.63 |
| 145 | Myanmar | CODE145 | 1.97 | 2.14 | 2.14 | 2.07 | 2.36 | 2.83 |
| 146 | Gambia, The | CODE146 | 2.06 | 2.00 | 2.67 | 2.22 | 2.00 | 2.46 |
| 147 | Mozambique | CODE147 | 2.26 | 2.15 | 2.08 | 2.10 | 2.08 | 2.74 |
| 148 | Mauritania | CODE148 | 1.93 | 2.40 | 2.07 | 2.06 | 2.23 | 2.75 |
| 149 | Kyrgyz Republic | CODE149 | 2.03 | 2.05 | 2.43 | 2.13 | 2.20 | 2.36 |
| 150 | Gabon | CODE150 | 2.00 | 2.08 | 2.58 | 2.25 | 1.92 | 2.31 |
| 151 | Yemen, Rep. | CODE151 | 1.63 | 1.87 | 2.35 | 2.21 | 2.21 | 2.78 |
| 152 | Cuba | CODE152 | 2.17 | 1.84 | 2.47 | 2.08 | 1.99 | 2.45 |
| 153 | Sudan | CODE153 | 1.87 | 1.90 | 2.23 | 2.18 | 2.42 | 2.33 |
| 154 | Djibouti | CODE154 | 2.20 | 2.00 | 1.80 | 2.21 | 2.00 | 2.74 |
| 155 | Syrian Arab Republic | CODE155 | 2.07 | 2.08 | 2.15 | 1.82 | 1.90 | 2.53 |
| 156 | Eritrea | CODE156 | 1.90 | 1.68 | 1.90 | 2.23 | 2.01 | 2.79 |
| 157 | Congo, Rep. | CODE157 | 1.50 | 1.83 | 2.17 | 2.17 | 2.17 | 2.58 |
| 158 | Afghanistan | CODE158 | 2.16 | 1.82 | 1.99 | 2.12 | 1.85 | 2.48 |
| 159 | Congo, Dem. Rep. | CODE159 | 1.78 | 1.83 | 1.70 | 1.84 | 2.10 | 2.04 |
| 160 | Somalia | CODE160 | 2.00 | 1.50 | 1.75 | 1.75 | 1.75 | 1.88 |

APPENDIX C.

Table C.1 Comparison of between OY and KT Models with $R_0$, $EFF_0$, $CPU_{0,}$ $TIME_O$ using LPI set

| SEQ | COUNTRY | DMU | $R_{OYS}$ | $EFF_{OY}$ | $CPU_{OY}$ | $TIME_{OY}$ | $R_{KT}$ | $EFF_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Germany | CODE1 | 0 | 1.000000 | 0.052 | 0.30 | 0 | 1.000000 | 0.163 | 0.40 | 0 | 0.000000 |
| 2 | Netherlands | CODE2 | 1 | 0.999993 | 0.136 | 0.55 | 1 | 0.999993 | 0.396 | 0.81 | 0 | 0.000000 |
| 3 | Belgium | CODE3 | 0 | 1.000000 | 0.143 | 0.70 | 0 | 1.000000 | 0.556 | 1.01 | 0 | 0.000000 |
| 4 | United Kingdom | CODE4 | 1 | 0.999922 | 0.334 | 1.03 | 1 | 0.999922 | 0.682 | 1.22 | 0 | 0.000000 |
| 5 | Singapore | CODE5 | 1 | 0.999935 | 0.389 | 1.21 | 1 | 0.999935 | 0.819 | 1.43 | 0 | 0.000000 |
| 6 | Sweden | CODE6 | 2 | 0.999919 | 0.548 | 1.64 | 2 | 0.999919 | 1.217 | 1.95 | 0 | 0.000000 |
| 7 | Norway | CODE7 | 0 | 1.000000 | 0.604 | 1.75 | 0 | 1.000000 | 1.267 | 2.15 | 0 | 0.000000 |
| 8 | Luxembourg | CODE8 | 0 | 1.000000 | 0.611 | 1.84 | 0 | 1.000000 | 1.33 | 2.26 | 0 | 0.000000 |
| 9 | United States | CODE9 | 1 | 0.999955 | 0.624 | 1.99 | 1 | 0.999955 | 1.557 | 2.58 | 0 | 0.000000 |
| 10 | Japan | CODE10 | 5 | 0.999552 | 1.318 | 3.32 | 5 | 0.999552 | 2.168 | 3.44 | 0 | 0.000000 |
| 11 | Ireland | CODE11 | 1 | 0.999936 | 1.333 | 3.46 | 1 | 0.999936 | 2.399 | 3.76 | 0 | 0.000000 |
| 12 | Canada | CODE12 | 7 | 0.999107 | 1.981 | 5.49 | 7 | 0.999107 | 3.127 | 4.92 | 0 | 0.000000 |
| 13 | France | CODE13 | 5 | 0.999402 | 2.421 | 6.39 | 5 | 0.999456 | 3.738 | 5.96 | 0 | -0.000054 |
| 14 | Switzerland | CODE14 | 5 | 0.999364 | 3.025 | 7.51 | 5 | 0.999364 | 4.137 | 6.62 | 0 | 0.000000 |
| 15 | Hong Kong SAR, China | CODE15 | 9 | 0.998541 | 4.488 | 10.61 | 9 | 0.998608 | 5.241 | 8.32 | 0 | -0.000068 |
| 16 | Australia | CODE16 | 6 | 0.998827 | 5.257 | 12.02 | 6 | 0.998827 | 6.022 | 9.47 | 0 | 0.000000 |
| 17 | Denmark | CODE17 | 2 | 0.999765 | 5.466 | 12.42 | 2 | 0.999765 | 6.335 | 9.99 | 0 | 0.000000 |
| 18 | Spain | CODE18 | 15 | 0.997314 | 6.987 | 16.24 | 11 | 0.997467 | 7.575 | 12.08 | 4 | -0.000153 |
| 19 | Taiwan | CODE19 | 4 | 0.999553 | 7.111 | 16.95 | 4 | 0.999553 | 7.929 | 12.74 | 0 | 0.000000 |
| 20 | Italy | CODE20 | 13 | 0.997631 | 7.964 | 19.91 | 12 | 0.997736 | 9.229 | 14.76 | 1 | -0.000105 |

| SEQ | COUNTRY | DMU | $R_{OYS}$ | $EFF_{OY}$ | $CPU_{OY}$ | $TIME_{OY}$ | $R_{KT}$ | $EFF_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 21 | Korea, Rep. | CODE21 | 18 | 0.994869 | 11.156 | 25.77 | 16 | 0.995168 | 11.394 | 17.81 | 2 | -0.000298 |
| 22 | Austria | CODE22 | 9 | 0.998232 | 11.268 | 27.07 | 9 | 0.998232 | 12.439 | 19.79 | 0 | 0.000000 |
| 23 | New Zealand | CODE23 | 4 | 0.999713 | 11.393 | 27.97 | 4 | 0.999713 | 12.772 | 20.51 | 0 | 0.000000 |
| 24 | Finland | CODE24 | 6 | 0.999027 | 11.577 | 28.87 | 6 | 0.999027 | 13.729 | 21.86 | 0 | 0.000000 |
| 25 | Malaysia | CODE25 | 9 | 0.998814 | 12.237 | 30.18 | 9 | 0.998814 | 14.825 | 23.61 | 0 | 0.000000 |
| 26 | Portugal | CODE26 | 21 | 0.994352 | 14.894 | 35.60 | 16 | 0.995433 | 17.67 | 27.49 | 5 | -0.001081 |
| 27 | United Arab Emirates | CODE27 | 22 | 0.991218 | 17.574 | 41.03 | 20 | 0.991992 | 20.743 | 32.20 | 2 | -0.000773 |
| 28 | China | CODE28 | 20 | 0.995501 | 19.123 | 44.57 | 20 | 0.995609 | 23.815 | 36.93 | 0 | -0.000108 |
| 29 | Qatar | CODE29 | 15 | 0.996826 | 19.684 | 47.12 | 15 | 0.996895 | 26.184 | 40.47 | 0 | -0.000069 |
| 30 | Turkey | CODE30 | 16 | 0.995079 | 20.119 | 49.67 | 16 | 0.995079 | 28.092 | 43.80 | 0 | 0.000000 |
| 31 | Poland | CODE31 | 12 | 0.996956 | 20.644 | 52.00 | 12 | 0.996982 | 30.086 | 46.58 | 0 | -0.000026 |
| 32 | Czech Republic | CODE32 | 12 | 0.997710 | 21.085 | 54.25 | 12 | 0.997738 | 31.761 | 49.14 | 0 | -0.000028 |
| 33 | Hungary | CODE33 | 16 | 0.996321 | 23.173 | 58.52 | 14 | 0.996667 | 33.957 | 52.50 | 2 | -0.000346 |
| 34 | South Africa | CODE34 | 24 | 0.993838 | 25.304 | 63.64 | 22 | 0.994036 | 37.536 | 57.30 | 2 | -0.000198 |
| 35 | Thailand | CODE35 | 27 | 0.991573 | 27.116 | 69.22 | 25 | 0.991925 | 41.554 | 62.81 | 2 | -0.000352 |
| 36 | Latvia | CODE36 | 18 | 0.995246 | 28.28 | 72.45 | 18 | 0.995286 | 44.32 | 66.98 | 0 | -0.000040 |
| 37 | Iceland | CODE37 | 21 | 0.990471 | 29.347 | 76.03 | 21 | 0.990587 | 47.688 | 71.96 | 0 | -0.000116 |
| 38 | Slovenia | CODE38 | 28 | 0.987953 | 32.231 | 82.54 | 27 | 0.988357 | 51.77 | 77.72 | 1 | -0.000404 |
| 39 | Estonia | CODE39 | 30 | 0.987878 | 34.025 | 88.21 | 25 | 0.989277 | 55.701 | 83.42 | 5 | -0.001399 |
| 40 | Romania | CODE40 | 26 | 0.993107 | 35.295 | 92.84 | 26 | 0.993129 | 60.094 | 89.63 | 0 | -0.000022 |
| 41 | Israel | CODE41 | 11 | 0.997393 | 35.602 | 94.00 | 11 | 0.997393 | 61.755 | 92.39 | 0 | 0.000000 |
| 42 | Chile | CODE42 | 39 | 0.976615 | 41.924 | 106.61 | 36 | 0.978859 | 67.739 | 100.88 | 3 | -0.002243 |
| 43 | Slovak Republic | CODE43 | 29 | 0.990874 | 44.179 | 112.84 | 29 | 0.990935 | 72.61 | 107.97 | 0 | -0.000061 |
| 44 | Greece | CODE44 | 27 | 0.982956 | 46.129 | 117.57 | 27 | 0.983379 | 76.907 | 114.10 | 0 | -0.000422 |
| 45 | Panama | CODE45 | 39 | 0.979419 | 51.321 | 128.15 | 36 | 0.980660 | 83.806 | 123.30 | 3 | -0.001241 |

| SEQ | COUNTRY | DMU | $R_{OYS}$ | $EFF_{OY}$ | $CPU_{OY}$ | $TIME_{OY}$ | $R_{KT}$ | $EFF_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 46 | Lithuania | CODE46 | 40 | 0.975241 | 56.607 | 139.58 | 38 | 0.976172 | 90.23 | 132.19 | 2 | -0.000931 |
| 47 | Bulgaria | CODE47 | 22 | 0.994265 | 57.237 | 141.91 | 22 | 0.994329 | 93.851 | 137.39 | 0 | -0.000064 |
| 48 | Vietnam | CODE48 | 41 | 0.979552 | 60.996 | 149.87 | 40 | 0.979889 | 100.895 | 146.89 | 1 | -0.000337 |
| 49 | Saudi Arabia | CODE49 | 34 | 0.975042 | 63.451 | 155.81 | 32 | 0.976803 | 106.008 | 154.21 | 2 | -0.001761 |
| 50 | Mexico | CODE50 | 42 | 0.977702 | 66.509 | 163.99 | 41 | 0.978098 | 113.607 | 164.64 | 1 | -0.000396 |
| 51 | Malta | CODE51 | 40 | 0.980490 | 68.905 | 171.12 | 39 | 0.980623 | 120.372 | 173.93 | 1 | -0.000132 |
| 52 | Bahrain | CODE52 | 29 | 0.980530 | 71.114 | 177.37 | 29 | 0.980703 | 125.293 | 180.97 | 0 | -0.000172 |
| 53 | Indonesia | CODE53 | 40 | 0.970585 | 75.022 | 185.74 | 40 | 0.970742 | 132.266 | 190.74 | 0 | -0.000156 |
| 54 | India | CODE54 | 42 | 0.977863 | 77.479 | 193.03 | 42 | 0.978059 | 139.252 | 200.36 | 0 | -0.000196 |
| 55 | Croatia | CODE55 | 47 | 0.962954 | 83.409 | 205.81 | 46 | 0.964645 | 147.799 | 211.76 | 1 | -0.001691 |
| 56 | Kuwait | CODE56 | 41 | 0.966860 | 86.735 | 213.98 | 41 | 0.967859 | 154.834 | 221.28 | 0 | -0.001000 |
| 57 | Philippines | CODE57 | 34 | 0.986998 | 87.497 | 217.56 | 34 | 0.986998 | 160.361 | 229.09 | 0 | 0.000000 |
| 58 | Cyprus | CODE58 | 53 | 0.962272 | 94.556 | 232.65 | 50 | 0.963622 | 169.417 | 241.37 | 3 | -0.001350 |
| 59 | Oman | CODE59 | 30 | 0.991257 | 95.158 | 235.37 | 30 | 0.991257 | 174.427 | 248.63 | 0 | 0.000000 |
| 60 | Argentina | CODE60 | 50 | 0.964515 | 100.877 | 247.74 | 48 | 0.966978 | 182.865 | 260.49 | 2 | -0.002464 |
| 61 | Ukraine | CODE61 | 43 | 0.967465 | 105.398 | 257.77 | 43 | 0.969579 | 190.432 | 270.83 | 0 | -0.002114 |
| 62 | Egypt, Arab Rep. | CODE62 | 42 | 0.968902 | 107.914 | 264.49 | 42 | 0.969259 | 197.6 | 280.69 | 0 | -0.000356 |
| 63 | Serbia | CODE63 | 47 | 0.971978 | 111.086 | 274.65 | 46 | 0.972695 | 206.507 | 292.67 | 1 | -0.000716 |
| 64 | El Salvador | CODE64 | 41 | 0.978200 | 113.119 | 281.37 | 41 | 0.978262 | 213.525 | 302.62 | 0 | -0.000062 |
| 65 | Brazil | CODE65 | 49 | 0.959443 | 118.348 | 294.07 | 49 | 0.959885 | 222.743 | 315.20 | 0 | -0.000443 |
| 66 | Bahamas, The | CODE66 | 44 | 0.962033 | 122.286 | 302.29 | 44 | 0.964766 | 230.994 | 326.50 | 0 | -0.002733 |
| 67 | Montenegro | CODE67 | 49 | 0.973333 | 125.67 | 309.14 | 49 | 0.973395 | 239.857 | 338.46 | 0 | -0.000062 |
| 68 | Jordan | CODE68 | 55 | 0.960889 | 130.387 | 321.46 | 54 | 0.961558 | 250.474 | 351.87 | 1 | -0.000669 |
| 69 | Dominican Republic | CODE69 | 64 | 0.952338 | 137.659 | 336.51 | 61 | 0.953737 | 261.571 | 366.74 | 3 | -0.001399 |
| 70 | Jamaica | CODE70 | 53 | 0.949545 | 143.211 | 348.54 | 53 | 0.952285 | 270.635 | 379.20 | 0 | -0.002740 |

| SEQ | COUNTRY | DMU | $R_{OYS}$ | $EFF_{OY}$ | $CPU_{OY}$ | $TIME_{OY}$ | $R_{KT}$ | $EFF_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 71 | Peru | CODE71 | 65 | 0.952488 | 147.589 | 362.40 | 62 | 0.953546 | 282.905 | 395.69 | 3 | -0.001058 |
| 72 | Pakistan | CODE72 | 53 | 0.967337 | 150.185 | 370.11 | 53 | 0.967486 | 292.462 | 408.98 | 0 | -0.000149 |
| 73 | Malawi | CODE73 | 47 | 0.954741 | 151.848 | 376.80 | 47 | 0.954754 | 299.642 | 419.49 | 0 | -0.000012 |
| 74 | Kenya | CODE74 | 45 | 0.974174 | 154.058 | 385.63 | 44 | 0.975134 | 307.926 | 430.81 | 1 | -0.000960 |
| 75 | Nigeria | CODE75 | 49 | 0.963886 | 157.57 | 394.88 | 48 | 0.966053 | 316.963 | 443.51 | 1 | -0.002167 |
| 76 | Venezuela, RB | CODE76 | 66 | 0.952091 | 163.871 | 407.78 | 64 | 0.952932 | 330.649 | 461.84 | 2 | -0.000841 |
| 77 | Guatemala | CODE77 | 62 | 0.947845 | 169.583 | 422.01 | 61 | 0.949088 | 341.143 | 476.24 | 1 | -0.001243 |
| 78 | Paraguay | CODE78 | 72 | 0.940033 | 179.592 | 444.18 | 67 | 0.943087 | 355.377 | 494.42 | 5 | -0.003054 |
| 79 | Côte d'Ivoire | CODE79 | 65 | 0.949576 | 188.107 | 461.52 | 61 | 0.951050 | 369.179 | 511.97 | 4 | -0.001474 |
| 80 | Rwanda | CODE80 | 60 | 0.949186 | 191.895 | 472.96 | 60 | 0.949859 | 381.513 | 528.39 | 0 | -0.000672 |
| 81 | Bosnia and Herzegovina | CODE81 | 57 | 0.958398 | 193.533 | 479.58 | 57 | 0.958439 | 391.987 | 542.72 | 0 | -0.000041 |
| 82 | Maldives | CODE82 | 48 | 0.957737 | 196.631 | 488.49 | 48 | 0.960834 | 400.415 | 554.06 | 0 | -0.003097 |
| 83 | Cambodia | CODE83 | 64 | 0.943523 | 202.226 | 503.36 | 64 | 0.944731 | 413.008 | 570.78 | 0 | -0.001208 |
| 84 | Sco Tomi and Principe | CODE84 | 54 | 0.960726 | 204.024 | 513.76 | 52 | 0.963428 | 423.345 | 585.55 | 2 | -0.002702 |
| 85 | Lebanon | CODE85 | 43 | 0.968156 | 205.113 | 518.05 | 43 | 0.968156 | 430.707 | 595.73 | 0 | 0.000000 |
| 86 | Ecuador | CODE86 | 79 | 0.933695 | 212.919 | 538.39 | 75 | 0.936462 | 447.26 | 616.87 | 4 | -0.002767 |
| 87 | Costa Rica | CODE87 | 68 | 0.935010 | 220.174 | 552.90 | 67 | 0.937087 | 458.781 | 632.76 | 1 | -0.002078 |
| 88 | Kazakhstan | CODE88 | 67 | 0.937896 | 224.686 | 565.96 | 67 | 0.938657 | 472.256 | 650.22 | 0 | -0.000761 |
| 89 | Sri Lanka | CODE89 | 65 | 0.940952 | 230.798 | 577.97 | 65 | 0.941420 | 483.055 | 664.89 | 0 | -0.000468 |
| 90 | Russian Federation | CODE90 | 75 | 0.931136 | 241.309 | 598.59 | 73 | 0.933604 | 499.529 | 686.16 | 2 | -0.002468 |
| 91 | Uruguay | CODE91 | 74 | 0.932242 | 249.503 | 615.43 | 73 | 0.934097 | 514.953 | 705.73 | 1 | -0.001856 |
| 92 | Armenia | CODE92 | 77 | 0.931386 | 256.968 | 633.67 | 72 | 0.933382 | 528.376 | 724.07 | 5 | -0.001996 |
| 93 | Namibia | CODE93 | 76 | 0.928602 | 264.23 | 651.32 | 74 | 0.929538 | 545.264 | 745.74 | 2 | -0.000936 |
| 94 | Moldova | CODE94 | 51 | 0.972299 | 265.18 | 656.33 | 51 | 0.972299 | 553.487 | 756.91 | 0 | 0.000000 |
| 95 | Nicaragua | CODE95 | 69 | 0.933477 | 272.436 | 672.31 | 67 | 0.935738 | 566.889 | 774.93 | 2 | -0.002260 |

| SEQ | COUNTRY | DMU | $R_{OYS}$ | $EFF_{OY}$ | $CPU_{OY}$ | $TIME_{OY}$ | $R_{KT}$ | $EFF_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 96 | Algeria | CODE96 | 65 | 0.932849 | 278.828 | 685.95 | 65 | 0.933269 | 579.149 | 791.13 | 0 | -0.000420 |
| 97 | Colombia | CODE97 | 78 | 0.924172 | 287.434 | 704.27 | 78 | 0.926620 | 595.792 | 812.53 | 0 | -0.002448 |
| 98 | Burkina Faso | CODE98 | 72 | 0.933845 | 291.648 | 717.12 | 70 | 0.934907 | 607.623 | 829.00 | 2 | -0.001063 |
| 99 | Belarus | CODE99 | 81 | 0.924971 | 300.772 | 738.15 | 77 | 0.927766 | 625.173 | 851.68 | 4 | -0.002794 |
| 100 | Ghana | CODE100 | 68 | 0.935161 | 305.151 | 752.50 | 67 | 0.938398 | 637.415 | 868.13 | 1 | -0.003238 |
| 101 | Senegal | CODE101 | 59 | 0.960896 | 306.954 | 759.92 | 58 | 0.961069 | 648.151 | 882.29 | 1 | -0.000173 |
| 102 | Liberia | CODE102 | 69 | 0.934820 | 309.418 | 770.77 | 68 | 0.935244 | 659.849 | 898.15 | 1 | -0.000424 |
| 103 | Honduras | CODE103 | 67 | 0.937264 | 315.538 | 783.54 | 66 | 0.938779 | 672.558 | 915.50 | 1 | -0.001515 |
| 104 | Ethiopia | CODE104 | 77 | 0.929431 | 320.459 | 796.82 | 77 | 0.929910 | 688.439 | 936.43 | 0 | -0.000479 |
| 105 | Nepal | CODE105 | 88 | 0.917457 | 332.919 | 821.24 | 85 | 0.918564 | 713.887 | 967.51 | 3 | -0.001107 |
| 106 | Solomon Islands | CODE106 | 83 | 0.917198 | 344.478 | 845.73 | 80 | 0.918814 | 741.426 | 1000.24 | 3 | -0.001616 |
| 107 | Burundi | CODE107 | 76 | 0.918424 | 351.228 | 860.69 | 76 | 0.920417 | 755.726 | 1019.42 | 0 | -0.001993 |
| 108 | Bangladesh | CODE108 | 75 | 0.936420 | 356.584 | 874.98 | 74 | 0.938548 | 771.02 | 1039.87 | 1 | -0.002128 |
| 109 | Benin | CODE109 | 72 | 0.927080 | 362.871 | 888.17 | 72 | 0.928959 | 783.618 | 1056.46 | 0 | -0.001879 |
| 110 | Tunisia | CODE110 | 71 | 0.947526 | 364.937 | 898.17 | 70 | 0.948085 | 796.352 | 1073.85 | 1 | -0.000560 |
| 111 | Fiji | CODE111 | 89 | 0.918979 | 373.679 | 918.28 | 89 | 0.920282 | 832.154 | 1115.73 | 0 | -0.001304 |
| 112 | Angola | CODE112 | 83 | 0.930203 | 378.853 | 932.74 | 83 | 0.930711 | 854.298 | 1143.51 | 0 | -0.000508 |
| 113 | Chad | CODE113 | 89 | 0.912208 | 388.701 | 954.02 | 87 | 0.915228 | 894.651 | 1189.71 | 2 | -0.003019 |
| 114 | Tajikistan | CODE114 | 91 | 0.918910 | 395.516 | 968.79 | 91 | 0.919456 | 916.898 | 1217.33 | 0 | -0.000546 |
| 115 | Mauritius | CODE115 | 95 | 0.903622 | 404.388 | 989.95 | 90 | 0.909571 | 940.129 | 1246.83 | 5 | -0.005949 |
| 116 | Georgia | CODE116 | 84 | 0.919230 | 408.534 | 1003.24 | 84 | 0.919400 | 957.076 | 1268.97 | 0 | -0.000169 |
| 117 | Macedonia, FYR | CODE117 | 90 | 0.894628 | 421.454 | 1028.21 | 90 | 0.897017 | 977.485 | 1295.32 | 0 | -0.002389 |
| 118 | Libya | CODE118 | 76 | 0.928182 | 426.21 | 1039.15 | 76 | 0.928347 | 990.662 | 1313.08 | 0 | -0.000165 |
| 119 | Mali | CODE119 | 80 | 0.932164 | 431.659 | 1052.04 | 80 | 0.932381 | 1007.247 | 1335.11 | 0 | -0.000217 |
| 120 | Botswana | CODE120 | 102 | 0.897946 | 442.833 | 1077.68 | 98 | 0.900362 | 1046.275 | 1380.76 | 4 | -0.002416 |

| SEQ | COUNTRY | DMU | $R_{OYS}$ | $EFF_{OY}$ | $CPU_{OY}$ | $TIME_{OY}$ | $R_{KT}$ | $EFF_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 121 | Bolivia | CODE121 | 87 | 0.909641 | 450.237 | 1096.36 | 87 | 0.910973 | 1067.14 | 1406.68 | 0 | -0.001332 |
| 122 | Guinea | CODE122 | 86 | 0.918609 | 453.98 | 1109.03 | 85 | 0.919707 | 1084.067 | 1428.99 | 1 | -0.001098 |
| 123 | Zambia | CODE123 | 83 | 0.908597 | 460.903 | 1124.96 | 83 | 0.910065 | 1101.575 | 1451.95 | 0 | -0.001468 |
| 124 | Guyana | CODE124 | 96 | 0.893278 | 473.711 | 1150.36 | 96 | 0.895775 | 1133.697 | 1490.11 | 0 | -0.002497 |
| 125 | Azerbaijan | CODE125 | 65 | 0.924221 | 479.186 | 1163.30 | 65 | 0.925409 | 1144.536 | 1504.47 | 0 | -0.001188 |
| 126 | Papua New Guinea | CODE126 | 105 | 0.886391 | 491.646 | 1190.31 | 105 | 0.888014 | 1186.07 | 1552.83 | 0 | -0.001623 |
| 127 | Guinea-Bissau | CODE127 | 98 | 0.891637 | 501.925 | 1211.67 | 97 | 0.893606 | 1215.227 | 1587.72 | 1 | -0.001968 |
| 128 | Comoros | CODE128 | 80 | 0.912091 | 505.032 | 1224.03 | 80 | 0.914253 | 1229.189 | 1607.00 | 0 | -0.002162 |
| 129 | Uzbekistan | CODE129 | 76 | 0.928525 | 510.144 | 1236.05 | 75 | 0.931317 | 1242.631 | 1625.01 | 1 | -0.002792 |
| 130 | Niger | CODE130 | 93 | 0.895555 | 519.342 | 1255.52 | 92 | 0.898009 | 1262.435 | 1650.17 | 1 | -0.002454 |
| 131 | Lao PDR | CODE131 | 98 | 0.893819 | 529.458 | 1274.79 | 98 | 0.895724 | 1283.725 | 1677.72 | 0 | -0.001905 |
| 132 | Madagascar | CODE132 | 88 | 0.915597 | 531.709 | 1284.64 | 88 | 0.915628 | 1300.45 | 1699.83 | 0 | -0.000031 |
| 133 | Lesotho | CODE133 | 110 | 0.874045 | 543.371 | 1309.20 | 109 | 0.878173 | 1331.141 | 1737.24 | 1 | -0.004128 |
| 134 | Central African Republic | CODE134 | 88 | 0.895392 | 549.399 | 1324.92 | 86 | 0.897275 | 1347.114 | 1758.75 | 2 | -0.001884 |
| 135 | Mongolia | CODE135 | 109 | 0.897419 | 554.211 | 1341.56 | 109 | 0.898014 | 1380.076 | 1798.17 | 0 | -0.000595 |
| 136 | Equatorial Guinea | CODE136 | 113 | 0.881382 | 567.394 | 1367.86 | 109 | 0.885015 | 1458.84 | 1883.91 | 4 | -0.003633 |
| 137 | Zimbabwe | CODE137 | 104 | 0.893052 | 575.028 | 1387.57 | 103 | 0.893999 | 1487.044 | 1918.36 | 1 | -0.000946 |
| 138 | Tanzania | CODE138 | 112 | 0.884761 | 583.527 | 1410.03 | 106 | 0.886450 | 1526.74 | 1964.56 | 6 | -0.001689 |
| 139 | Togo | CODE139 | 117 | 0.868414 | 594.463 | 1433.67 | 114 | 0.872177 | 1567.132 | 2012.75 | 3 | -0.003763 |
| 140 | Turkmenistan | CODE140 | 116 | 0.883879 | 603.749 | 1454.81 | 113 | 0.885347 | 1604.415 | 2056.52 | 3 | -0.001468 |
| 141 | Iraq | CODE141 | 114 | 0.876111 | 609.727 | 1473.84 | 114 | 0.876548 | 1633.218 | 2092.43 | 0 | -0.000437 |
| 142 | Cameroon | CODE142 | 104 | 0.881313 | 620.407 | 1497.48 | 102 | 0.883284 | 1667.829 | 2133.68 | 2 | -0.001971 |
| 143 | Bhutan | CODE143 | 110 | 0.872401 | 628.06 | 1517.78 | 110 | 0.873635 | 1691.936 | 2164.41 | 0 | -0.001234 |
| 144 | Haiti | CODE144 | 126 | 0.844820 | 644.05 | 1549.27 | 126 | 0.847065 | 1748.928 | 2230.34 | 0 | -0.002244 |
| 145 | Myanmar | CODE145 | 116 | 0.872463 | 650.18 | 1567.31 | 116 | 0.872824 | 1776.369 | 2264.78 | 0 | -0.000361 |

| SEQ | COUNTRY | DMU | $R_{0YS}$ | $EFF_{OY}$ | $CPU_{OY}$ | $TIME_{OY}$ | $R_{KT}$ | $EFF_{KT}$ | $CPU_{KT}$ | $TIME_{KT}$ | $R_{OY}$- $R_{KT}$ | $EFF_{OY}$- $EFF_{KT}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 146 | Gambia, The | CODE146 | 100 | 0.907080 | 652.699 | 1578.03 | 100 | 0.907083 | 1801.824 | 2295.48 | 0 | -0.000003 |
| 147 | Mozambique | CODE147 | 132 | 0.854403 | 664.896 | 1605.37 | 125 | 0.857857 | 1874.007 | 2376.15 | 7 | -0.003454 |
| 148 | Mauritania | CODE148 | 102 | 0.874104 | 670.343 | 1622.54 | 102 | 0.878269 | 1903.399 | 2411.98 | 0 | -0.004165 |
| 149 | Kyrgyz Republic | CODE149 | 126 | 0.852366 | 675.114 | 1640.60 | 126 | 0.852654 | 1953.754 | 2470.52 | 0 | -0.000288 |
| 150 | Gabon | CODE150 | 111 | 0.889138 | 678.596 | 1653.61 | 111 | 0.889237 | 1982.932 | 2508.59 | 0 | -0.000099 |
| 151 | Yemen, Rep. | CODE151 | 123 | 0.860637 | 687.996 | 1676.08 | 123 | 0.861522 | 2021.076 | 2555.08 | 0 | -0.000885 |
| 152 | Cuba | CODE152 | 122 | 0.862961 | 691.77 | 1693.49 | 122 | 0.863383 | 2066.405 | 2608.18 | 0 | -0.000422 |
| 153 | Sudan | CODE153 | 124 | 0.843323 | 696.523 | 1713.04 | 124 | 0.844000 | 2104.249 | 2653.86 | 0 | -0.000676 |
| 154 | Djibouti | CODE154 | 131 | 0.848388 | 706.103 | 1738.23 | 130 | 0.854639 | 2170.735 | 2728.68 | 1 | -0.006251 |
| 155 | Syrian Arab Republic | CODE155 | 142 | 0.802288 | 727.621 | 1778.41 | 141 | 0.807327 | 2265.532 | 2832.24 | 1 | -0.005039 |
| 156 | Eritrea | CODE156 | 121 | 0.861959 | 731.409 | 1794.19 | 121 | 0.862350 | 2293.058 | 2867.35 | 0 | -0.000392 |
| 157 | Congo, Rep. | CODE157 | 141 | 0.812186 | 745.109 | 1825.74 | 141 | 0.813138 | 2366.618 | 2951.08 | 0 | -0.000952 |
| 158 | Afghanistan | CODE158 | 136 | 0.816476 | 756.351 | 1850.09 | 136 | 0.817920 | 2419.001 | 3013.17 | 0 | -0.001443 |
| 159 | Congo, Dem. Rep. | CODE159 | 150 | 0.751090 | 763.705 | 1871.65 | 150 | 0.751285 | 2468.662 | 3073.32 | 0 | -0.000196 |
| 160 | Somalia | CODE160 | 146 | 0.760860 | 767.052 | 1886.59 | 146 | 0.760860 | 2502.858 | 3116.91 | 0 | 0.000000 |