G.U. Journal of Science
20(4): 117-125 (2007)

www.gujs.org

A Goal Programming Approach to Weight Dispersion in Data Envelopment Analysis

Hasan BAL, H.Hasan ÖRKCÜ♠

Gazi University Arts and Science Faculty, Department of Statistics, Ankara, Turkey

Received: 15.04.2005 Revised: 12.06.2007 Accepted: 13.08.2007

ABSTRACT

In this study, a multi-criteria data envelopment analysis (MCDEA) model, used in the literature to moderate the homogeneity of weights dispersion, is solved using pre-emptive goal programming. The MCDEA model solved using pre-emptive goal programming gives the same relative efficiency as the classical DEA model while it improves the homogeneity of input-output weights. This conclusion is confirmed by the computational results obtained when the two models are applied to a real data set relative to the socio-economic performances of European countries and to randomly generated instances with various numbers of decision making units, inputs and outputs.

Key Words: Data envelopment analysis, weight dispersion, pre-emptive goal programming, coefficient of variation.

1. INTRODUCTION

Data envelopment analysis (DEA) is a fractional mathematical programming technique that was developed by Charnes et al [1]. It is used to measure the productive efficiency of decision making units (DMUs) and evaluate their relative efficiency. It determines the productivities of DMUs, specified as the ratio of the weighted sum of outputs to the weighted sum of inputs, compares them to each other and determines the most efficient DMU. DEA obtains the optimal weights for all inputs and outputs of each unit without imposing any constraint on these weights. While it is an advantage of DEA that these weights are free, the assigned weights are sometimes unrealistic. The issue of unrealistic weights has been investigated by the techniques of weight~~s~~ restriction. However, these techniques may give infeasible solutions for weights [2–14]. This paper addresses the problem of unrealistic weights- not by using weights restrictions, but by using pre-emptive goal programming. The proposed method gives the same relative efficiency values as the classical DEA model while improving the homogeneity of input-output weights, as will be illustrated by the computational results.

This paper is organized as follows. In Section 2, the basic classical DEA model is given. In Section 3, the multi criteria data envelopment analysis (MCDEA) model is presented and its formulation as a goal program (GPMCDEA) is explained. In Section 4, both the classical DEA and the GPMCDEA are applied to a real data set relative to the European countries and their solutions are compared. In Section 5, the simulation data performances of approaches are compared. Lastly, in Section 6, a summary of this research and its results is provided.

2. DATA ENVELOPMENT ANALYSIS

DEA evaluates the relative efficiency of homogeneous units by considering multiple inputs and outputs. Inputs can be resources used by a DMU and outputs can be products produced and/or performance measures of the DMU. The efficiency is defined as a ratio of the weighted sum of outputs to the weighted sum of inputs. DEA has been extensively used to compare the efficiencies of non-profit and profit organizations such as schools, hospitals, shops, bank branches and other environments where there are relatively homogeneous DMUs [15].

Assuming that there are n DMUs, each with m inputs and s outputs, the relative efficiency, w_o, of a particular DMU o is obtained by solving the following fractional programming problem:

♠Corresponding author, e-mail: hhorkcu@*gazi.edu.tr*

$$w_o = \text{Max } \frac{\sum_{r=1}^{s} u_r y_{ro}}{\sum_{i=1}^{m} v_i x_{io}}$$

$$\frac{\sum_{r=1}^{s} u_r y_{rj}}{\sum_{i=1}^{m} v_i x_{ij}} \le 1 \quad j = 1, 2, \ldots, n \tag{1}$$

$$u_r \ge 0, \quad r = 1, 2, \ldots, s$$

$$v_i \ge 0, \quad i = 1, 2, \ldots, m$$

where j is the DMU index, $j = 1, \ldots, n$; r the output index, $r = 1, \ldots, s$; i the input index, $i = 1, \ldots, m$; y_{rj} the value of the r^{th} output for the j^{th} DMU, x_{ij} the value of the i^{th} input for the j^{th} DMU, u_r the weight given to the r^{th} output; and v_i the weight given to the i^{th} input. In this model, DMU_o is efficient if and only if $w_o = 1$.

A DMU is considered individually in determining its relative efficiency. This DMU is referred to as the target DMU. The target DMU effectively selects weights that maximize its output to input ratio, subject to the constraints that the output to input ratios of all the n DMUs with these weights are ≤ 1. A relative efficiency score of 1 indicates that the DMU under consideration is efficient, whereas a score less than 1 implies that it is inefficient.

The proposed fractional program can be converted into a linear programming problem where the optimal value of the objective function indicates the relative efficiency of DMU o. The reformulated linear programming problem is as follows:

$$w_o = \text{Max } \sum_{r=1}^{s} u_r y_{ro}$$

$$\sum_{i=1}^{m} v_i x_{io} = 1$$

$$\sum_{r=1}^{s} u_r y_{rj} - \sum_{i=1}^{m} v_i x_{ij} \le 0 \quad j = 1, 2, \ldots, n \tag{2}$$

$$u_r \ge 0, \quad r = 1, 2, \ldots, s$$

$$v_i \ge 0, \quad i = 1, 2, \ldots, m$$

In model (2), the weighted sum of the inputs for the target DMU is forced to 1, thus allowing for the conversion of the fractional programming problem into a linear programming problem which can be solved by using a commercial linear programming software.

Model (2) can be expressed equivalently in the form given by Li and Reeves [16]:

$$\min d_o \left(\text{or max } \sum_{r=1}^{s} u_r y_{ro} \right)$$

$$\sum_{i=1}^{m} v_i x_{io} = 1 \tag{3}$$

$$\sum_{r=1}^{s} u_r y_{rj} - \sum_{i=1}^{m} v_i x_{ij} + d_j = 0 \quad j = 1, 2, \ldots, n$$

$$u_r \ge 0, \quad r = 1, 2, \ldots, s$$

$$v_i \ge 0, \quad i = 1, 2, \ldots, m$$

$$d_j \ge 0, \quad j = 1, 2, \ldots, n$$

where d_o is the deviation variable for DMU_o and d_j the deviation variable of DMU_j. The quantity d_o, which is bounded by the interval $(0,1]$, can be regarded as a measure of inefficiency. Under model (3), DMU_o is efficient if and only if $d_o = 0$ or $\sum_{r=1}^{s} u_r y_{r_o} = 1$. If DMU_o is not efficient, its efficiency score is $1 - d_o$. The smaller the value of d_o, the less inefficient (thus the more efficient) DMU_o is. We shall call model (2) or (3) the classical DEA model. We say that the classical DEA method minimizes DMU_o's inefficiency, as measured by d_o, under the constraint that the weighted sum of the outputs is less than or equal to the weighted sum of the inputs for each DMU.

3. MULTIPLE CRITERIA DEA MODEL

The form of the multiple criteria data envelopment analysis (MCDEA) model is not unique; it depends upon the efficiency criteria used. A MCDEA problem that has the three criteria: minimizing d_o, the deviation of the DMUo, minimizing M, the maximum deviation, and minimizing $\sum_{j=1}^{n} d_j$, the sum of the deviations, can be modeled as in Li and Reeves [16]:

$$\begin{aligned}
&\min\ d_o \ \left(\text{or}\ \max\ \sum_{r=1}^{s} u_r y_{ro}\right)\\
&\min\ M\\
&\min\ \sum_{j=1}^{n} d_j\\
&\qquad \sum_{i=1}^{m} v_i x_{io} = 1\\
&\qquad \sum_{r=1}^{s} u_r y_{rj} - \sum_{i=1}^{m} v_i x_{ij} + d_j = 0\ ,\ j = 1,\ 2,\ \ldots,\ n\\
&\qquad M - d_j \geq 0\ ,\ j = 1,\ 2,\ \ldots,\ n\\
&\qquad u_r \geq 0,\quad r = 1, 2,\ \ldots, s\\
&\qquad v_i \geq 0,\quad i = 1, 2,\ \ldots, m\\
&\qquad d_j \geq 0,\quad j = 1,\ 2,\ \ldots,\ n
\end{aligned} \tag{4}$$

The first objective of model (4) is identical to the objective of model (3). The variable M in the second objective represents the maximum quantity among all deviation variables $d_j, j = 1,\ 2,\ \ldots,\ n$. The third objective function is a straightforward representation of the sum of the deviations. The feasible region for the decision variables u_r and v_i in model (4) is the same as that in model (3). The constraints $M - d_j \geq 0\ , j = 1,\ 2, \ldots n$, that define the maximum deviation M do not change the decision feasible region of decision variables as discussed in [16]. There are several solution methods for the multiple objective linear programming model (4), such as the Steuers's narrowing cone, the multiple criteria simplex method, compromise programming and goal programming [17].

Goal programming is a branch of multiple objective programming, also known as multiple-criteria decision making (MCDM). It can be thought of as an extension of linear programming to handle multiple, normally conflicting objective measures. Each of these measures is given a goal or target value to be achieved. Unwanted deviations from this set of target values are then minimised in an achievement function. This function can be a vector or a weighted sum dependent on the goal programming variant used. As satisfaction of the target is deemed to satisfy the decision maker(s), an underlying satisficing philosophy is assumed [18].

The initial goal programming formulations ordered the unwanted deviations into a number of priority levels, with the minimisation of a deviation in a higher priority level being of infinitely more importance than any deviations in lower priority levels. This is known as lexicographic or pre-emptive goal programming. Ignizio [19] gives an algorithm showing how a pre-emptive goal program can be solved as a series of linear programmes. Pre-emptive goal programming should be used when there exists a clear priority ordering amongst the goals to be achieved.

Thus, pre-emptive goal programming is used in solving the multi objectives and assigning priority to objectives in model (4). The assignment of priorities to these objectives is generally decided by the decision maker [18, 19]. The measure of relative efficiency, d_o, is the main objective (since it is an efficiency measure of the unit considered); thus the top priority is assigned to it. It must be satisfied first, prior to the other two objectives, the second priority minimizes the maximum deviation and the third priority minimizes the sum of deviations (the order of the second and the third priorities may be changed). Therefore, for any DMU, the above mentioned MCDEA model can be formulated as a goal program (Goal Programming Multiple Criteria Data Envelopment Analysis-GPMCDEA) as follows:

$$\begin{aligned}
&\min\ a = \left\{ n_1 + p_1 + p_2,\ \sum_j n_{3j},\ \sum_j d_j \right\}\\
&\qquad \sum_{i=1}^{m} v_i x_{io} + n_1 - p_1 = 1\\
&\qquad \sum_{r=1}^{s} u_r y_{ro} + n_2 - p_2 = 1\\
&\qquad \sum_{r=1}^{s} u_r y_{rj} - \sum_{i=1}^{m} v_i x_{ij} + d_j = 0\ ,\ j = 1,\ 2,\ \ldots,\ n\\
&\qquad M - d_j + n_{3j} - p_{3j} = 0\ ,\ j = 1,\ 2,\ \ldots,\ n\\
&\qquad u_r \geq 0,\quad r = 1, 2,\ \ldots, s\\
&\qquad v_i \geq 0,\quad i = 1, 2,\ \ldots, m\\
&\qquad d_j \geq 0,\quad j = 1,\ 2,\ \ldots,\ n\\
&\qquad n_1, p_1, n_2, p_2 \geq 0\\
&\qquad n_{3j}, p_{3j} \geq 0\ ,\ j = 1,\ 2,\ \ldots,\ n
\end{aligned} \tag{5}$$

where for the DMU under evaluation, n_1 and p_1 are the unwanted deviation variables for the goal which makes the weighted sum of inputs to unity, n_2 is the wanted deviation variable for the goal which makes the weighted sum of outputs less than or equal to unity, p_2 is the unwanted deviation variable for the goal which makes the weighted sum of outputs less than or equal to unity, n_{3j} 's are the unwanted deviation variables for the goal (i.e., $M - d_j \geq 0\ ,\ j = 1,\ 2,\ \ldots,\ n$) which realizes M as the maximum deviation, and p_{3j} 's are the wanted deviation variables for the same goal (i.e., $M - d_j \geq 0\ ,\ j = 1,\ 2,\ \ldots,\ n$).

Whereof our aim is to minimize the sum of unwanted deviations n_1, p_1 (i.e, $\sum_{i=1}^{m} v_i x_{io} = 1$) and p_2 (equivalently minimizing d_o or maximizing $\sum_{r=1}^{s} u_r y_{ro}$) in the first priority, under this first priority our second priority is to minimize $\sum_j n_{3j}$, and under the first two priorities our third priority is to minimize the sum of deviations (i.e, $\sum_j d_j$).

4. AN APPLICATION

The efficiency and weights dispersion of the classical and the goal programming DEA are evaluated using a real data set relative to the efficiency of 27 European countries. The set extracted from [20] characterizes each country by four inputs and four outputs as illustrated in Table 1. The input variables are: x_{1j}, the unemployment rate, x_{2j}, the inflation rate, x_{3j}, the infant mortality rate (per 1000 new borns), and x_{4j}, the population (in millions) whereas the output variables are: y_{1j}, the gross national product by purchasing power parity per capita (in 1000 Euros), y_{2j}, the health expenditure per capita (in Euros), y_{3j}, the percent of gross national product spent on education, and y_{4j}, the percent of gross national product spent on research and development.

Table 1. The input and output levels for 27 European countries as extracted from [20]

Country	j	x_{1j}	x_{2j}	x_{3j}	x_{4j}	y_{1j}	y_{2j}	y_{3j}	y_{4j}
Belgium	1	8.6	2.4	0.147	10.47	29.0	2081	3.1	1.6
Germany	2	8.9	1.8	0.123	82.46	27.0	3402	5.1	2.3
Greece	3	9.3	3.3	0.156	11.08	20.8	1106	3.1	0.5
Spain	4	8.1	3.6	0.135	43.30	24.0	1215	5.0	0.8
France	5	9.3	2.0	0.128	62.70	26.3	2957	6.0	2.2
Ireland	6	4.3	2.9	0.155	4.17	34.1	1430	6.0	1.5
Italy	7	7.1	2.3	0.138	58.60	24.4	1788	4.9	1.1
Luxembourg	8	4.6	3.2	0.120	0.45	63.0	2217	4.0	1.6
Holland	9	3.9	1.6	0.134	16.31	30.9	2271	5.1	1.9
Austria	10	5.1	1.8	0.121	8.23	30.2	1968	5.4	1.6
Portugal	11	7.6	2.9	0.154	15.56	17.1	1238	5.8	0.6
Slovenia	12	6.1	2.5	0.133	2.01	20.5	1054	2.8	0.6
Finland	13	7.7	1.3	0.101	5.24	27.7	1508	7.5	1.6
Czech. Rep.	14	7.4	2.5	0.146	10.23	18.7	934	5.1	1.3
Denmark	15	3.8	2.0	0.220	5.41	30.0	2131	8.1	1.9
Estonia	16	5.4	4.4	0.263	1.34	15.9	512	2.7	0.6
Latvia	17	7.4	6.7	0.396	2.30	12.8	487	2.5	0.5
Lithuania	18	5.9	3.8	0.271	3.41	13.5	687	2.4	0.5
Hungary	19	7.3	3.9	0.628	10.08	15.6	705	4.6	0.7
Malta	20	7.0	3.0	0.557	0.40	17.1	878	4.1	0.7
Poland	21	13.9	1.4	0.378	38.16	12.5	498	7.5	0.6
Slovakia	22	14.3	4.5	0.627	5.40	14.6	938	4.7	0.8
Sweden	23	7.3	1.5	0.087	9.03	28.5	1748	8.3	3.8
England	24	5.3	2.4	0.151	60.21	28.7	3619	5.3	2.4
Bulgaria	25	8.9	7.0	0.409	7.74	8.4	756	3.1	0.7
Romania	26	7.6	6.8	0.563	21.62	8.8	678	2.9	0.6
Turkey	27	9.8	10.2	0.887	72.06	6.9	457	3.2	0.7

Tables 2 and 3 summarize respectively the results of the classical and of the preemptive goal programming models. Column 3 displays the efficiency of the DMUp, w_j. Columns 4-7 report the input weights v_{ij}, $i = 1, \ldots, 4$ whereas columns 8-11 tally the output weights u_{ij}, $i = 1, \ldots, 4$. Finally, column 12 indicates the coefficient of variation c_j of the weights. c_j, the ratio of sample standard deviation to the sample mean, measures the variability of the weights relative to their mean (or average). It compares the relative dispersion in one type of data with the relative dispersion in another type of data.

Table 2. Results of the classical DEA model for the data set relative to 27 European countries

Country	j	w_j	v_{1j}	v_{2j}	v_{3j}	v_{4j}	u_{1j}	u_{2j}	u_{3j}	u_{4j}	c_j
Belgium	1	0,812	0	0.201	1.739	0	0	0.0004	0	0	2.35
Germany	2	1	0.0411	0.352	0	0	0.0044	0.0003	0	0	2.32
Greece	3	0,365	0	0.081	3.399	0	0	0.0003	0	0	2.59
Spain	4	0,533	0.086	0	2.215	0	0.0054	0	0.0807	0	2.42
France	5	0,915	0.0005	0	5.749	0.0041	0	0.0003	0	0	2.64
Ireland	6	0,938	0.0731	0	3.089	0.0496	0.0026	0	0.1418	0	2.40
Italy	7	0,628	0.0332	0.1579	2.9091	0	0.0084	0.0002	0.0225	0	2.43
Luxembourg	8	1	0.2174	0	0	0	0.0065	0.0003	0	0	2.56
Holland	9	1	0.2360	0.0134	0	0.0036	0.0047	0.0004	0	0	2.40
Austria	10	1	0	0.3212	1.034	0.0259	0.0039	0.0004	0	0	1.96
Portugal	11	0,579	0.0878	0	2.1604	0	0	0	0.0999	0	2.40
Slovenia	12	0,553	0	0.2295	0	0.2120	0	0.0001	0.1768	0	1.30
Finland	13	1	0.0557	0.4264	0	0.0032	0.0226	0	0.0500	0	1.95
Czech. Rep.	14	0,532	0.0898	0	2.2996	0	0.0056	0	0.0838	0	2.42
Denmark	15	1	0	0.111	0	0.1317	0	0	0.1235	0	1.29
Estonia	16	0,493	0.1420	0	0	0.1741	0	0	0.1829	0	1.30
Latvia	17	0,315	0.0979	0	0	0.1200	0	0	0.1260	0	1.30
Lithuania	18	0,306	0.0992	0	0	0.1216	0	0	0.1278	0	1.30
Hungary	19	0,301	0.0509	0	0	0.0624	0	0	0.0655	0	1.30
Malta	20	1	0.0193	0.2540	0	0.2571	0.0009	0	0.2401	0	1.23
Poland	21	0,928	0	0.7143	0	0	0	0	0.1238	0	2.23
Slovakia	22	0,408	0.0068	0	0.0847	0.0929	0	0	0.0870	0	2.00
Sweden	23	1	0.0059	0.1201	0	0.0860	0	0	0.0577	0.1371	1.06
England	24	1	0.1615	0	0	0.0024	0.0029	0.0003	0	0	2.26
Bulgaria	25	0,217	0.0544	0	0	0.0667	0	0	0.0700	0	1.30
Romania	26	0,179	0.1316	0	0	0	0	0	0.0617	0	1.87
Turkey	27	0,153	0.1020	0	0	0	0	0	0.0479	0	1.87

Table 2 is characterized by a large number of zero weights. It infers that the second and third input as well as the second and fourth outuput were not used to evaluate the efficiency of the DMUs. These resulting models are counter-intuitive. They are invalid from a socio-economic context where the inflation rate (x_{2j}) and infant mortality rate (x_{3j}), as well as the health expenditure per capita (y_{2j}) and percent of gross national product spent on research and development (y_{4j}) are fundamental indices of the development of a country. Table 3, on the other hand, has few zero weights. Some of the weights might be small in magnitude, but their inputs and outputs are accounted for in the model. The dispersion of weights given in Table 3 is more homogeneous than the dispersion of weights given by the classical model and reported in Table 2. Indeed, the comparison of the last column of Tables 2 and 3 indicates a smaller coefficient of variations for the weights induced by the GPMCDEA model than for the weights produced by the classical model. This conclusion is further supported by the comparative bar graph displayed in Figure 1, which shows that the coefficient of variation of GPMCDEA is consistently smaller for all 27 countries than its classical model counterpart. On the other hand, the two models yield similar efficiency values for all 27 countries, as can be deduced from Figure 2.

5. A COMPARISON OF THE METHODS BY SIMULATION

In the preceding section, the results obtained undoubtedly applied to one sample. In this section, the performance of the multi-criteria and the classical DEA models are compared using randomly generated instances, where the performance is measured in terms of the homogeneity of weight dispersion; which in turn is evaluated by the coefficient of variation of the input and output weights. If the coefficient of variation of the weights of every DMU_j, $j = 1, \ldots, n$, of GPMCDEA is smaller than its counterpart for the classical DEA model, then GPMCDEA yields more homogeneous weight dispersion~~s;~~ otherwise, the classical DEA model yields more homogeneous weight dispersions. The computational investigation considers randomly generated instances with eight levels of n, and five levels for each pair of (m, r). For each combination of n, and (m, r), 10000 random instances are generated with the x_{ij} and y_{rj}, $i = 1, \ldots, m$, $r = 1, \ldots, s$, and $j = 1, \ldots, n$, randomly generated from the Uniform(0,1000). The instances are solved using MATLAB 7. The results are summarized in Table 4, which indicates the number of DMUs, n, the number of inputs, m, the number of outputs, s, the number of times GPMCDEA yields better results than the classical DEA, k, and the CPU time(in seconds) of the classical DEA, cpu_{DEA}, and of GPMCDEA, $cpu_{GPMCDEA}$. For example, when $n=15$, $m=1$, and $s=1$, the weight dispersion GPMCDEA gives more homogeneous solutions than the classical DEA model $k=9264$ times out of 10,000 instances; which means that the classical DEA model outperforms GPMCDEA only on 736 occasions out of the 10,000 tested instances.

Table 3. Results of the preemptive goal programming DEA model for the data set relative to 27 European countries

Country	j	w_j	v_1	v_2	v_3	v_4	u_1	u_2	u_3	u_4	c
Belgium	1	0,810	0.0285	0.171	1.175	0.0167	0.0102	0.0002	0.0154	0.0318	2.09
Germany	2	1	0.0311	0.119	0.0101	0.0062	0.0082	0.0002	0.0065	0.0301	1.38
Greece	3	0,359	0.0027	0.081	2.137	0	0.0037	0.0002	0.0234	0.0001	2.49
Spain	4	0,531	0.0168	0.0198	2.115	0.0117	0.0068	0	0.0707	0.0378	2.42
France	5	0,911	0.0069	0.0891	2.951	0.0062	0	0.0002	0.0507	0.0151	2.48
Ireland	6	0,937	0.0937	0	1.876	0.0756	0.0028	0.0001	0.1026	0.0584	2.19
Italy	7	0,625	0.0332	0.1519	1.504	0.0036	0.0055	0.0002	0.0198	0.0357	2.22
Luxembourg	8	1	0.1379	0.0858	0.0878	0.1898	0.0014	0.0003	0.0204	0.098	0.81
Holland	9	1	0.2169	0.0184	0.0083	0.0076	0.0038	0.0003	0.0387	0	1.88
Austria	10	1	0.0087	0.3212	0.7031	0.0359	0.0048	0.0004	0	0.0398	1.69
Portugal	11	0,571	0.0981	0.0052	0.9013	0.0064	0.0128	0.0002	0.01047	0.0871	2.06
Slovenia	12	0,553	0.01089	0.2095	0.0098	0.2012	0.0068	0.0001	0.1092	0	1.25
Finland	13	1	0.0601	0.3062	0	0.0268	0.0223	0	0.05	0	1.65
Czech. Rep.	14	0,531	0.0838	0.0387	0.8221	0.0161	0.0039	0.0002	0.0534	0	2.07
Denmark	15	1	0	0.1058	0.0616	0.1435	0.0039	0.0001	0.0791	0.0188	0.99
Estonia	16	0,492	0.1021	0.0695	0.0794	0.0865	0.0103	0	0.1231	0.0209	0.69
Latvia	17	0,315	0.0612	0.0564	0	0.0734	0.0107	0.0001	0.0487	0.0065	0.89
Lithuania	18	0,305	0.1102	0	0.0854	0.0979	0.0008	0.0002	0.0623	0.0098	0.98
Hungary	19	0,301	0.0319	0.0412	0.0897	0.0539	0.0078	0.0001	0.0218	0.0055	0.89
Malta	20	1	0.0193	0.254	0	0.2571	0.0046	0.0001	0.2001	0.0203	1.18
Poland	21	0,921	0	0.5143	0.7607	0	0.0204	0	0.0802	0.0942	1.47
Slovakia	22	0,408	0.0141	0	0.2104	0.1235	0.0045	0.0001	0.0523	0	1.43
Sweden	23	1	0.0019	0.1034	0	0.0926	0.0057	0.0002	0.0134	0.1021	1.16
England	24	1	0.1015	0.0058	0.0046	0.0075	0.0056	0.0002	0.0067	0.0347	1.53
Bulgaria	25	0,217	0.0671	0	0.0304	0.0513	0.0187	0	0.0234	0	0.98
Romania	26	0,173	0.0941	0.0105	0.0069	0.0097	0.0127	0	0.0218	0.0137	1.32
Turkey	27	0,158	0.0322	0.0178	0.0026	0.007	0.0036	0.0001	0.0244	0.0091	0.89

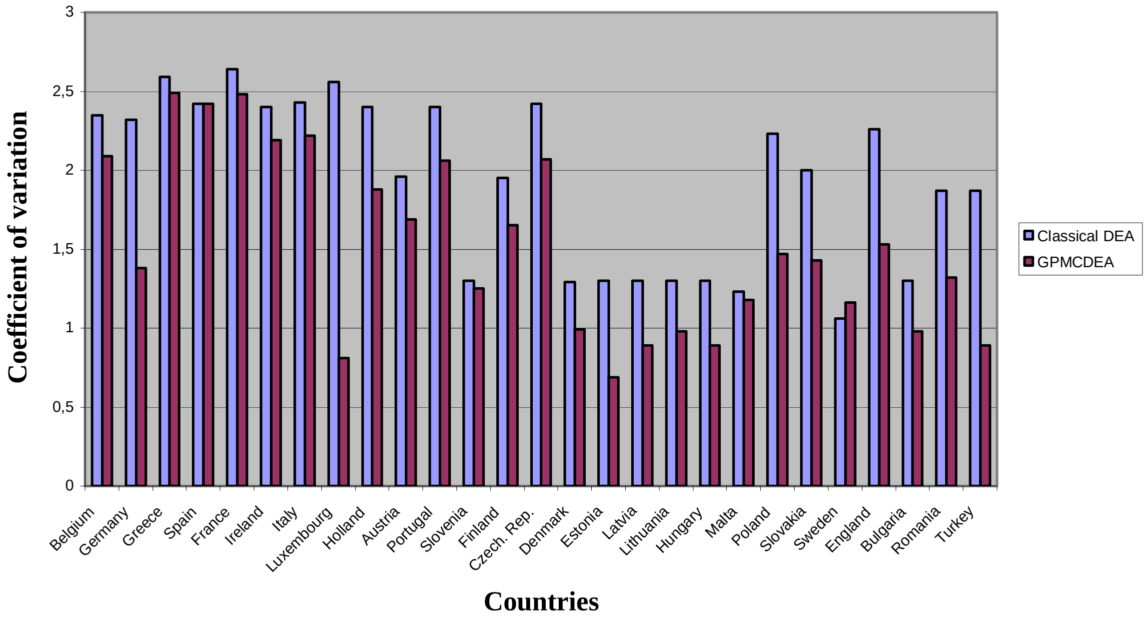

Figure 1.Comparison of the coefficient of variation of the weights for the classical DEA and GPMCDEA models for the 27 European country data set.

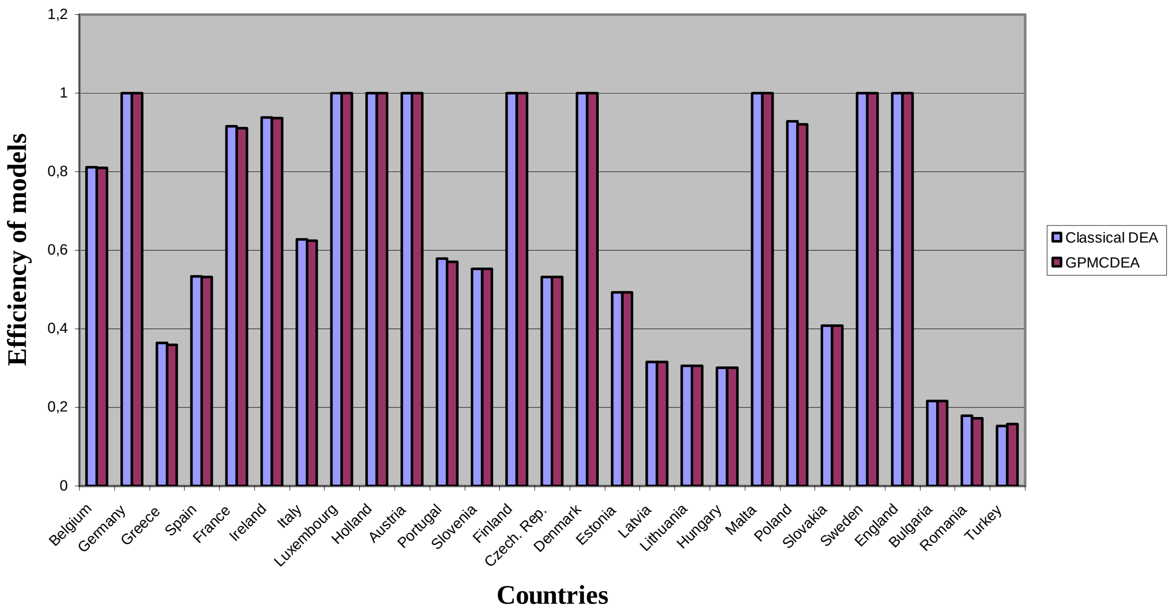

Figure 2. The Comparison of the relative efficiency resulting from the classical DEA and GPMCDEA models for the 27 European country data set.

Table 4 shows *k*, the number of times GPMCDEA produces more homogeneous weights than the classical DEA models in 10,000 replications of each problem instance, and the corresponding average CPU times: cpu_{DEA} and $cpu_{GPMCDEA}$. Each instance is characterized by the number of DMUs, *n*, the number of inputs, *s*, and the number of outputs, *m*. This table shows that GPMCDEA performs better than the classical DEA model in more than 88% of the cases and reaches 92.64% of the cases for *n*=15, *s*=1, and =1.

Table 4. Comparison of the GPMCDEA and the classical models in terms of weighted dispersion and computation time on 10000 replications of randomly generated instances.

n	s	m	k	cpu_{DEA}	$cpu_{GPMCDEA}$
15	1	1	9264	2.87	4.02
	1	2	9128	2.90	4.03
	2	1	9058	2.89	4.10
	2	2	9067	2.91	4.15
	2	3	9214	2.95	4.25
30	2	4	9098	3.17	5.68
	3	3	8976	3.12	5.73
	3	4	9087	3.25	5.87
	4	2	9001	3.24	5.71
	4	3	9201	3.30	5.90
45	3	2	9082	3.82	6.02
	3	3	9000	3.87	6.14
	3	5	8899	4.12	6.21
	4	4	9025	4.15	6.17
	4	5	8945	4.28	6.20
60	3	2	9148	4.78	7.61
	3	5	9008	4.87	7.58
	4	5	9178	4.76	7.63
	5	5	8938	4.86	7.91
	5	6	9002	4.91	7.88
80	4	4	9128	4.77	8.05
	2	5	9098	4.81	8.02
	6	5	8888	4.98	8.14
	6	7	9024	4.95	8.26
	3	8	9149	5.03	8.17
100	4	6	9100	5.10	8.88
	7	5	8983	5.15	8.57
	4	7	8825	5.17	8.64
	5	8	9069	5.21	8.84
	7	8	9164	5.19	8.90
150	5	7	9000	6.02	10.05
	8	5	8869	5.98	10.35
	8	8	9043	6.24	10.24
	7	10	9029	6.32	10.47
	10	9	8814	6.45	10.68
200	7	9	9101	7.87	12.68
	8	6	9133	7.91	12.95
	10	10	8992	8.26	12.88
	12	12	8968	8.68	13.04
	15	13	9007	8.51	13.51

This better performance is reached at the cost of a higher computational time. However, the increase of CPU time is of the order of only few seconds; which undoubtedly does not hinder the usefulness of GPMCDEA. Indeed, the largest observed average run time is 13.51 seconds, whereas the corresponding average run time for the classical model is 8.51 seconds.

6. CONCLUSIONS

In many instances, classical DEA models yield non-homogeneous weight dispersion of input and output parameters. Indeed, they yield several input-output weights that are zero or that have extreme values which imply that the corresponding parameters are not taken into account to interpret the efficiency of the decision making units. This paper overcomes this problem by solving a multi-criteria data envelopment analysis model using pre-emptive goal programming. The obtained solution improves the dispersion of weights as demonstrated by a real case data set and randomly generated instances. The results can be further improved if weighted linear goal programming is used to balance the weights and reduce the number of efficient DMUs.

ACKNOWLEDGMENT

We thank the anonymous referees for their helpful and constructive comments on the previous version of the manuscript which improved the presentation of this article.

REFERENCES

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

[2] Meza, L.A., Lins, M.P.E., "Review of Methods for Increasing Discrimination in Data **Envelopment Analysis",** ***Annals of Operations Research***, 116, 225-242, (2002).

[3] Thompson, R.G., Singleton, F.G., Thrall, R.M., Smith, B.A., "Comparative Site Evaluations for Locating a High-Energy Physics Lab in Texas", ***Interfaces***, 16: 35-49, (1986).

[4] Thompson, R.G., Langemeier, L.N., Lee, C.T., Thrall, R.M., "The Role of Multiplier Bounds in Efficiency Analysis with Application to Kansas Farming", ***Journal of Econometrics***, 46: 93-108 (1990).

[5] Charnes, A., Cooper, W.W, Huang, Z.M., Sun, D.B., "Polyhedral Cone-Ratio Models with an Illustrative Application to Large Commercial Banks", ***Journal of Econometrics***, 46: 73-91 (1990).

[6] Wong, Y.H.B, Beasley, J.E., "Restricting Weight Flexibility in Data Envelopment Analysis", ***Journal of the Operational Research Society***, 41(9): 829-835 (1990).

[7] Zhu, J., "Data Envelopment Analysis with Preference Structure", ***Journal of the Operational Research Society***, 47, 136-150 (1996).

[8] Halme, M., Joro, T., Korhonen, P., Salo, S., Wallenius, J., "A Value Efficiency Approach to Incorporating Preference Information in Data Envelopment Analysis", ***Management Science***, 45, 103-115, (1999).

[9] Jahanshahloo, G.R., Memariani, A., Hosseinzadeh, F., Shoja, N., "A Feasible Interval for Weights in Data Envelopment Analysis", ***Applied Mathematics and Computation***, 160, 155-168, (2005).

[10] Allen, R., Athanassopoulos, A., Dyson, R.G., Thanassoulis, E., "Weight Restrictions and Value Judgement in Data Envelopment Analysis: Evolution, Development and Future Directions", ***Annals of Operations Research***, 73, 13-34, (1997).

[11] Podinovski, V.V., Athanassopoulos, A., "Assessing the Relative Efficiency of Decision Making Units in DEA Models with Weight Restrictions", ***Journal of the Operational Research Society***, 49, 500-508 (1998).

[12] Podinovski, V.V., "Computation Efficient Targets in DEA Models with Production Trade-offs and Weight Restrictions", ***European Journal of Operational Research***, 181, 586-591, (2007).

[13] Cooper, W.W., Ruiz, J.L., Sirvent I., "Choosing Weights from Alternative Optimal Solutions of Dual Multiplier Models in DEA", ***European Journal of Operational Research***, 180, 443-458, (2007).

[14] Kousmanen, T., Cherchye, L., Sipilainen, T., "The Law of One Price in Data Envelopment Analysis: Restricting Weight Flexibility Across Firms", ***European Journal of Operational Research***, 170, 735-757 (2006).

[15] Cooper, W.W., Seiford, L.M., Tone, K., "Data Envelopment Analysis", ***Kluwer Academic Publishers***, Boston, USA (2000).

[16] Li, X.B., Reeves, G.R., "A multiple criteria approach to data envelopment analysis", ***European Journal of Operational Research***, 115: 507-517 (1999).

[17] Zeleny, M., "Multiple Criteria Decision Making", ***McGraw-Hill***, New York, USA, 215-280 (1982).

[18] Ignizio, J.P., "Goal Programming and Extension", ***Lexington Books***, Lexington, 20-55 (1976).

[19] Bal, H., Örkcü, H.H., Çelebioğlu, S., "An Experimental Comparison of the new Goal Programming and the Linear Programming Approaches in the Two-Group Discriminant Problems", ***Computers and Industrial Engineering***, 50 (3): 296-311 (2006).

[20] DPT(Devlet Planlama Teşkilatı- Turkish Republic Prime Ministry State Planning Organization) , *AB* ***Üyesi Ülkelerde ve Aday Ülkelerde Ekonomik Gelişmeler***, AB ile İlişkiler Genel Müdürlüğü, Şubat 2007, http://www.dpt.gov.tr/abigm/abmeg/2007ekonomik gelisme.pdf.