Data Envelopment Analysis: A Primer for Novice Users and Students at all Levels

R. Samuel Sale
Lamar University

Martha Lair Sale
Florida Institute of Technology

In the three decades since the publication of Charnes, Cooper, and Rhodes' (1978) classic article on the topic, data envelopment analysis, or DEA, has become very popular. From 1978 to 2001, over 3,000 publications by over 2,000 authors have used DEA (Tavares, 2002). The purpose of this paper is to present DEA in a way that is appropriate for new users at all levels. This paper is written with a minimum of technical language and assumptions as to the readers' prior knowledge. In this way, it represents a stand-alone lesson in DEA that is appropriate for undergraduate students. On the other hand, rather than completely omitting advanced topics appropriate for more demanding users; these topics are touched on briefly and interested readers are directed to other works that address such advanced topics in more detail.

Keywords: data envelopment analysis, performance measurement techniques, analysis techniques

Introduction

Three decades ago, Charnes, Cooper, and Rhodes (1978) introduced a non-parametric analysis method known as data envelopment analysis, or DEA. Since that classic article, DEA has become very popular with over 3,000 publications by over 2,000 authors between 1978 and 2001 (Tavares, 2002). DEA is a very flexible method of comparing the efficiency performance of various decision-making units, or DMUs. The definition of the DMU is quite flexible. DMUs can be individuals, branches of an organization, or entire organizations. What is important is not the scale, but that all DMUs exist in the same basic environment and convert the same set inputs into the same set of outputs. DEA is concerned with measuring the relative efficiency of the various DMUs as they convert their inputs into outputs. As a non-parametric method, DEA does not require or assume any functional relationship between the inputs and outputs. Instead, a weighting scheme is used. This will be discussed in greater detail below.

The purpose of this paper is to present DEA in a way that is both simple enough to be appropriate for undergraduates and rigorous enough to be appropriate for doctoral students. This is accomplished by first deciding which potential DEA related topics are of general importance to all analysts and which are of importance to some subset of analysts. Topics that are of general importance are presented with a minimum of technical language and mathematics. That being said, it is assumed that readers have a basic conceptual understanding of linear programming and its uses. However, it is not assumed that they are experts in linear programming, or that they are at all familiar with DEA prior to reading this paper. Topics that may be important to some analysts, but not others, are mentioned but not covered in depth. Instead interested readers are directed to sources of specific relevance to these topics.

This paper is divided into several sections. In this first section, DEA is introduced and the purpose and framework of the paper is established. The second section concerns the selection of appropriate DMUs, inputs, and outputs for use in a DEA. The third section concerns incorporating collected data into the DEA model. The fourth section concerns solving the DEA model and analyzing the results. The fifth section summarizes and concludes the paper.

Data Selection

There are a few important issues to consider prior to beginning a DEA. These issues center on defining the DMUs to be analyzed and the inputs and outputs used to analyze them. First and foremost, all DMUs should convert the same set of inputs into the same set of outputs. If this is not the case, the analyst must take corrective action. For example, if one DMU is unique in its production of an output that is included in the analysis, then that DMU is guaranteed a relative efficiency rating of 1.0 regardless of its performance relative to the inputs or any of the other outputs. This situation severely limits the interpretive power of the DEA.

Ideally, all important inputs are used and outputs are produced by all DMUs; however, sometimes this is not the case. It may be that a DMU produces some valuable output that the other DMUs do not. In this case, omission of the output from analysis would unfairly penalize the DMU engaged in its production, while inclusion of the output would unfairly aide the DMU, unless the analyst takes steps to prevent this from happening. Fortunately, there are relatively simple methods by which the analyst can limit the weight assigned to a given output when determining a DMU's relative efficiency. The allowable weights can be limited in any DEA, and maximums or minimums can be placed on input or output weights. The method by which this is done will be discussed in the next section.

Another important thing to consider prior to data collection is what inputs and outputs will be used to measure the performance of each DMU. Often the selection of appropriate inputs and outputs is rather straightforward; but at other times, the selection is not so clear. In these cases, it is important to remember that for a measure to be an appropriate input it should be something that, all else being equal, the DMUs would like to minimize. One not entirely obvious potential input is floor space. Even though this is something that cannot easily be adjusted by the DMU after operations have begun, space in which to operate is an input and producing the same outputs in a smaller space is preferred. So this may be a valid input.

Sometimes an analyst may be tempted to add a very large number of inputs and outputs to a DEA. While there is nothing inherently wrong with this, some care must be taken. Although DEA is very different than techniques such as regression analysis, the analyst must consider some of the same things when selecting inputs and outputs. Namely, a particular concept should not be "double-counted." That is to say, each input and output should measure something that is distinct from all of the other inputs and outputs or confounding effects may make the interpretation of results very difficult. Also similar to when using regression, the analyst must be aware of degrees of freedom. One rule of thumb is that there should be at least twice as many DMUs as there are inputs and outputs combined. If this is not the case then the likelihood of most or all DMUs receiving efficiency scores at or near 1.0 is great. Again, this limits the interpretive power of the DEA.

Many DEA models are static in nature; that is to say, they contain data from a single time period. Often, studies that do contain data for multiple time periods perform separate DEAs for each period. However, it is possible and sometimes beneficial to treat each DMU-time period combination as a distinct DMU in a single DEA. For more on this technique, called window analysis, the interested reader is directed to Charnes, Clark, and Cooper (1985).

After the data has been collected but before the DEA is formulated, the analyst should consider whether the data exhibits constant or variable returns to scale. If a process exhibits constant returns to scale, then doubling all inputs should allow all outputs to be doubled. If a process exhibits constant returns to scale, then multiplying all inputs by any positive number should impact all outputs linearly based on that number. If on the other hand a process exhibits variable returns to scale then this relationship is not necessarily linear. The interested reader is directed to Banker, Charnes, and Cooper (1984) and Banker and Thrall (1992) for more information on estimating returns to scale and modeling without assuming constant returns to scale.

Formulating the Problem

There are multiple ways of formulating a DEA model. These formulations are equivalent. This means that there is no "right" or "wrong" but instead that the decision of which to use is based on convenience and interpretive power. Regardless of which formulation is used, a separate optimization is performed for each decision-making unit, or DMU. In other words, if there are n DMUs then n different optimizations must be performed in order to complete a DEA regardless of whether a fractional or linear formulation is used.

A DEA model can be formulated as a fractional or linear program. The main reason to formulate a DEA model as a fractional program is to explain what the analyst is trying to accomplish through DEA. In this respect, the fractional formulation is of more use as a teaching tool than as a practical tool. As stated above, each DMU takes a turn as the focal DMU while separate optimizations are performed. The objective of these optimizations is to select the weights used when calculating the DMUs relative efficiencies. A DMU's efficiency is defined as the sum of weighted outputs divided by the sum of weighted inputs. Each optimization selects the set of weights that results in the highest possible efficiency for the focal DMU associated with that optimization. These separate optimizations share a common set of constraints: when the set of weights are applied to any DMU, it must not result in an efficiency rating greater than one. A fractional formulation for the case of s outputs, m inputs, and n DMUs where the y terms represent output levels, the x terms represent input levels, and the u and v terms represent the weights associated with outputs and inputs respectively, is shown below as Formulation 1.

$$\begin{aligned} \text{Maximize} \quad & \frac{\sum_{r=1}^{s} u_r y_{r1}}{\sum_{i=1}^{m} v_i x_{i1}} \\ \text{subject to} \quad & \frac{\sum_{r=1}^{s} u_r y_{rj}}{\sum_{i=1}^{m} v_i x_{ij}} \leq 1 \text{ for } j = 1, \ldots, n, \text{ and} \\ & u_r, v_i \geq 0, \text{ for } r = 1, \ldots, s \text{ and } r = 1, \ldots, m. \end{aligned} \tag{1}$$

The above formulation should be relatively intuitive. However, the formulation is not linear and thus cannot be solved by linear optimization methods. Charnes and Cooper (1962) demonstrate that this particular type of non-linear problem can be converted to linearity by algebraic manipulation. This modification allows the DEA to be solved using linear methods. A linear version of the above formulation is shown below as Formulation 2.

$$\begin{aligned} \text{Maximize} \quad & \sum_{r=1}^{s} u_r y_{r1} - \sum_{i=1}^{m} v_i x_{i1} \\ \text{subject to} \quad & -\sum_{i=1}^{m} v_i x_{i1} \leq -1, \\ & \sum_{r=1}^{s} u_r y_{rj} - \sum_{i=1}^{m} v_i x_{ij} \leq 0 \text{ for } j = 1, \ldots, n, \text{ and} \\ & u_r, v_i \geq 0, \text{ for } r = 1, \ldots, s \text{ and } r = 1, \ldots, m. \end{aligned} \tag{2}$$

Note that the first constraint has been changed from a greater than or equal constraint to a less than or equal constraint by multiplying both sides by negative one. The reason to do this is so that all of the constraints are of the same type. While this is not strictly necessary, it does make certain aspects of model formulation easier if it is the case. In order to use Formulation 2, the analyst must arrange the data in a very specific, yet relatively simple way. This can easily be done using a spreadsheet such as Microsoft Excel. It should also be noted after installing Excel's Solver add-in, the DEA model can also be solved using Excel. This allows the analyst to perform a DEA without the need for any special software.

For illustrative purposes, assume the analyst wants to examine the efficiency of six companies (DMUs), all of which produce the same two products (output 1 and output 2) using the same raw material (input). Figure 1 contains fictitious data concerning these six DMUs and is the raw data that will be used in examples below.

	Input	Output 1	Output 2
DMU 1	64.1	102	683
DMU 2	60.8	171	545
DMU 3	50.0	175	250
DMU 4	60.0	210	192
DMU 5	59.9	126	456
DMU 6	70.8	46	604

Figure 1: Raw data for example problem

Figure 2 shows a screen capture of this data entered into Microsoft Excel and formulated as described below.

	A	B	C	D	E	F
1		Output 1	Output 2	Input		Obj
2	Objective Coefficients	102	683	-64.1		0
3	Decision Variables	0	0	0		RHS
4	Input Constraint	0	0	-64.1	0	-1
5	DMU 1 Constraint	102	683	-64.1	0	0
6	DMU 2 Constraint	171	545	-60.8	0	0
7	DMU 3 Constraint	175	250	-50	0	0
8	DMU 4 Constraint	210	192	-60	0	0
9	DMU 5 Constraint	126	456	-59.9	0	0
10	DMU 6 Constraint	46	604	-70.8	0	0

Figure 2: Data entered into Excel in primal form

One simple way to arrange the data in accordance with Formulation 2 is so that each column is associated with a different input or output and each row is associated with a different DMU (cells A5 through D10). This represents most of the *A* matrix. In linear programming, the *A* matrix is consists of the constraint coefficients where each column is associated with a different decision variable and each row is associated with a different constraint. In order to complete the *A* matrix, an additional row must be added. This additional row is associated with the inputs constraint shown above. The new row should have zeros in each column associated with an output and the input values for the focal DMU in each column associated with an input (row 4). Now, every value in columns associated with inputs (column D), including the values in this new row, should be multiplied by negative one. The result is the completed *A* matrix. In linear programming, the column vector *b* represents the "right hand sides" of the constraints. As can be seen in Figure 2 (column F), all but one element of the *b* vector is equal to zero. The element in the "new" row just described has a value of negative one. In linear programming, the row vector *c* (row 2) represents the objective function coefficients. When using Formulation 2, the elements of the *c* vector will be equal to the elements from the row of the *A* matrix associated with the focal DMU. As stated in the previous section, it is possible to set the maximum possible weights for an input or output. This is done rather simply by adding the appropriate constraint row to the *A* matrix and *b* vector.

To recap, cells B2 through D2 contain the row vector *c*, which represents the objective coefficients for the linear program. Note that this row is identical to one of the rows of the *A* matrix. This is always the case when solving this formulation. In this case, the objective coefficients are identical to row 5, which is the row associated with DMU 1. This indicates that DMU 1 is currently the focal DMU. Cells B3 through D3 represent the decision variables, which will contain the optimal combination of input and output weights after the linear program is solved. Cells B5 through C10 contain the output performance for the various DMUs. Cells D5 through D10 contain negative one times the input usage for the various DMUs. Cells B4 and C4 are equal to zero because the input constraint is not associated with outputs. This is always the case when solving this formulation. Cell D2 is identical to cell D4. This is always the case when solving this formulation. Cells B4 through D10 represent the *A* matrix. Cell F2 is equal to the products of rows 2 and 3 summed from column B to D. This is the objective cell that is to be maximized by manipulating cells B3 through D3. Cells E4 through E10 are equal to the sum of products of row 3 and rows 4 through 10 respectively, each summed from column B to D. These cells represent the left hand side of the constraints. Changes to the decision variables will impact the objective cell and also these cells. Cells F4 through F10 represent column vector *b*, or the right hand sides of the constraints. Note that the cell associated with the input constraint is equal to negative one and the remaining cells are equal to zero. This is always the case when solving this formulation.

In the above example, there are 6 DMUs, 2 outputs, and 1 input. Therefore, the *A* matrix is 7 by 3. It is seven rows high because there is a row for each DMU plus one associated with the input constraint. It is three columns wide because there are a total of three inputs and outputs combined. Additional inputs, outputs, or DMUs would be treated exactly as described above. This means that if there were more than one input then more than one row of the *A* matrix would be generated by multiplying raw data by negative one. When the linear programming formulation shown in Figure 2 is solved, the resulting optimal solution is the optimal set of input and output weights; however, the optimal value is not the efficiency because of the changes involved in converting from Formulation 1 to Formulation 2. That being said, it is not difficult to determine the efficiency rating once the optimal weights have been generated.

In order to complete the DEA, the formulation should be modified slightly so that each DMU has a turn as the focal DMU. This involves changing the values in the *c* vector (row 2) as well as the non-zero value from the input constraint and then re-solving the linear program. Although the modification is somewhat labor intensive, once the analyst becomes familiar with the process, it proceeds quickly.

Every linear program has a dual. The solution to this dual formulation is intimately related to the solution to the original, or primal, formulation. Some information that may be of interest to the analyst is more readily available after solving the primal problem while other information is more readily available after solving the dual problem. Some authors use the dual of Formulation 2 to solve DEAs. This approach will not be taken here, but it is a common enough approach that it should be mentioned so that inexperienced readers will not be surprised if they see such a formulation. When the dual of a linear program is taken, several changes occur. Chief among these is that a maximization problem becomes a minimization problem and that the *A* matrix is transposed so that rows become columns and vice versa.

This section describes in detail the essential subjects a novice should master prior to attempting to formulate a DEA. However, it is by no means an exhaustive list of what some analyst may wish to know. For more information on extensions to the basic DEA models described above, the interested reader is directed to Cooper (2001, p186-189).

Analyzing the Results

This section concerns the information that the analyst can obtain after solving the DEA. Figure 3 is a graphical representation of the performance of the various DMUs where the horizontal axis is units of output 1 per unit of input and the vertical axis is units of output 2 per unit of input. The line segments are the efficient frontier suggested by the data. Dividing both of the outputs by the input values for the various DMUs allows the three dimensional data from Figure 1 to be represented by a two-dimensional graph. If there had been two inputs and one output a figure similar to Figure 3 could have been created where the horizontal axis is units of input one divided by units of output and the vertical axis is units of input two divided by units of output. These two techniques are often used in DEA analyses that contain a total of three inputs and outputs.

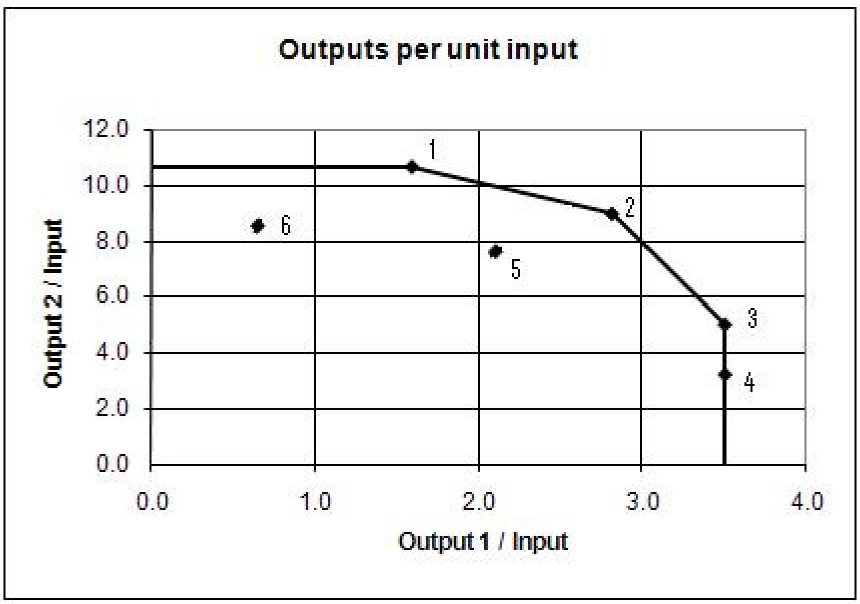

Figure 3: Outputs per unit input

Figures such as Figure 3 serve multiple purposes. They allow the analyst to get a rough idea of the DMUs relative efficiencies. The segmented line that passes through DMUs 1, 2, 3, and 4 represents the efficient frontier implied by the data. This means that any data points between the frontier and the origin, such as DMUs 5 and 6 are inefficient but possible. On the other hand, the data suggests that a data point beyond the frontier may represent impossibly high levels of output for the amount of input being used. Any data point that falls on the efficient frontier between DMU 1 and DMU 3 is relatively efficient and Pareto optimal. Any data point that falls on the vertical or horizontal sections of the efficient frontier are relatively efficient but not Pareto optimal. A DMU is Pareto optimal if improvement in performance along one axis (for example, increased volume of one output) can only be accomplished by degrading performance along another axis (for example, decreasing volume of the other output or increasing volume of the input). The distinction between Pareto optimal and relatively efficient but not Pareto optimal can best be explained by examining DMUs 3, and 4. DMU 3 is Pareto optimal because no other DMU produces both more of output 1 and more of output 2 per unit input. DMU 4 is not Pareto optimal because DMU 3 is superior with respect to output 2 while not inferior with respect to output 1. Despite this, DMU 4 is relatively efficient. This will be discussed in greater detail below. Also, figures such as Figure 3 are a good way of displaying large amounts of data. Finally, comparison of such figures to the efficiency ratings yielded by a DEA should either instill novice analysts with confidence that the analysis has been performed correctly or indicate that mistakes have been made.

As stated above, relatively efficient DMUs exist on the frontier; however, Figure 3 can be used to determine the efficiency of relatively inefficient DMUs as well. This is done by taking the ratio of the lengths of two partially overlapping line segments. The first line segment connects the origin with the DMU. The second line segment passes through the DMU while connecting the origin and the efficient frontier. If Figure 3 had been input oriented instead of output oriented, the efficient frontier would exist between the origin and all inefficient DMUs. In this case, the efficiency would instead be reciprocal of this ratio. It should be noted that Figure 3 implicitly assumes constant returns to scale. Although the analyst probably cannot determine the efficiency of an inefficient DMU with much precision by examining Figure 3, the analyst probably can determine whether an efficiency calculated after performing the DEA is plausibly by looking at Figure 3.

Figure 3 can also provide information concerning output mix issues for inefficient DMUs as well as the reference set of efficient DMUs appropriate for comparison to an inefficient DMU. This is done by examining where the line segment described above intersects the efficient frontier. Figure 4 is the same as Figure 3 except that it includes the line segments described above for the two relatively inefficient DMUs and labels the DMUs.

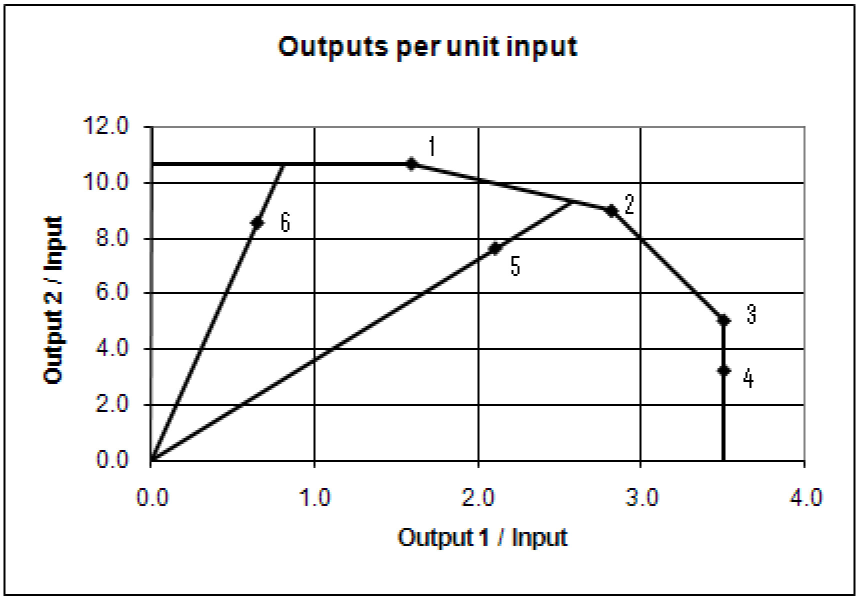

Figure 4: Outputs per unit input showing line segments for inefficient DMUs

The line segment passing through DMU 5 in Figure 4 intersects the efficient frontier between DMU 1 and DMU 2. This indicates that DMUs 1 and 2 are the efficient reference DMUs, as opposed to DMUs 3 or 4, to which DMU 5 should be compared. Note that the line segment passing through DMU 6 intersects the horizontal portion of the efficient frontier. This is significant. First it means that DMU 6 has only one reference DMU. It also means that if a new DMU were added at this intersection point, it would have a relative efficiency of one but would not be Pareto optimal. This seeming contradiction is possible because DEA measures relative efficiency using a "radial measure." Essentially this means that DEA assumes a fixed proportion mixture of inputs and a fixed proportion mixture of outputs. Graphically, this means that the two line segments used to determine a DMU's efficiency are partially overlapping instead of extending from the origin in distinct directions. Look for example at DMU 4. As indicated by DMU 3, DMU 4 could increase its performance with respect to output 2 without degrading its performance with respect to output 1; therefore, it is not Pareto optimal. On the other hand, DMU 4 exists on the efficient frontier; therefore, it is relatively efficient. This reliance on a radial measure of efficiency is why relative efficiency is a necessary condition but not sufficient to indicate Pareto optimality.

When there are four or more inputs and outputs combined, it is not possible to generate a graphical representation such as Figure 4. The rest of this section describes how to analyze the results of a DEA without relying on a graphical representation. Some of the first things an analyst may be interested in obtaining are the optimal weights and relative efficiency scores. The optimal weights are readily available because they are the decision variables; however, the objective value is not the efficiency score. Figure 5 is a screen capture of the same cells shown in Figure 2 after solving the linear program associated with DMU 5.

	A	B	C	D	E	F
1		Output 1	Output 2	Input		Obj
2	Objective Coefficients	126	456	-59.9		-0.18
3	Decision Variables	0.0018	0.0013	0.0167		RHS
4	Input Constraint	0	0	-59.9	-1	-1
5	DMU 1 Constraint	102	683	-64.1	0	0
6	DMU 2 Constraint	171	545	-60.8	0	0
7	DMU 3 Constraint	175	250	-50	-0.2	0
8	DMU 4 Constraint	210	192	-60	-0.4	0
9	DMU 5 Constraint	126	456	-59.9	-0.2	0
10	DMU 6 Constraint	46	604	-70.8	-0.3	0

Figure 5: Solution for DMU 5

DMU 5's efficiency can be calculated based on the values of cells B2 through D4. It is equal to $\frac{\sum(weight * output)}{\sum(weight * input)}$, or $\frac{0.0018 * 126 + 0.0013 * 456}{0.0167 * 59.9}$, which is in turn equal to 0.82. Recall that cell D2 is equal to negative one times the input. Note that if the objective value is equal to zero then the DMU's relative efficiency is always equal to one so the above calculation need not be performed for efficient DMUs.

If the analyst is interested in determining the efficient reference DMUs for an inefficient focal DMU, this can also be determined by examining Figure 5. Note that the constraints associated with DMU 1 and DMU 2 are both binding. This is because the left and right sides (columns E and F) of these two constraints are equal. This indicates that these are the reference DMUs appropriate for the focal DMU, DMU 5.

Figure 6 is a screen capture of the formulation after solving the linear program associated with DMU 6. Note that one of the decision variables, cell B3, has a value of zero and compare this to Figure 5.

	A	B	C	D	E	F
1		Output 1	Output 2	Input		Obj
2	Objective Coefficients	46	604	-70.8		-0.20
3	Decision Variables	0.0000	0.0013	0.0141		RHS
4	Input Constraint	0	0	-70.8	-1	-1
5	DMU 1 Constraint	102	683	-64.1	0	0
6	DMU 2 Constraint	171	545	-60.8	-0.1	0
7	DMU 3 Constraint	175	250	-50	-0.4	0
8	DMU 4 Constraint	210	192	-60	-0.6	0
9	DMU 5 Constraint	126	456	-59.9	-0.2	0
10	DMU 6 Constraint	46	604	-70.8	-0.2	0

Figure 6: Solution for DMU 6

The fact that the decision variable associated with output 1 is equal to zero is significant and must be addressed. Note that in Figure 4, the line segment that passes through DMU 6 and intersects the efficient frontier does so at a point that is not Pareto optimal. If the "virtual DMU" that exists at the point where a DMU's radial line intersects the efficient frontier is not Pareto optimal then the decision variable associated with the output that can be improved will always be equal to zero as is the case in Figure 6. Based on recent discussion, the reader should not be surprised that the solution for DMU 4 is associated with zero values for the decision variable associated with output 2 and a zero value for the objective value. This is confirmed by Figure 7.

	A	B	C	D	E	F
1		Output 1	Output 2	Input		Obj
2	Objective Coefficients	210	192	-60		0
3	Decision Variables	0.0048	0.0000	0.0167		RHS
4	Input Constraint	0	0	-60	-1	-1
5	DMU 1 Constraint	102	683	-64.1	-0.6	0
6	DMU 2 Constraint	171	545	-60.8	-0.2	0
7	DMU 3 Constraint	175	250	-50	0	0
8	DMU 4 Constraint	210	192	-60	0	0
9	DMU 5 Constraint	126	456	-59.9	-0.4	0
10	DMU 6 Constraint	46	604	-70.8	-1	0

Figure 7: Solution for DMU 4

As stated previously, DMU 4 is relatively efficient but not Pareto optimal. How these facts are related to the zero values mentioned above can be highlighted by examining the optimal weights and efficiency scores of DMU 3 and DMU 4. With the current set of weights, the efficiency of DMU 3 is $\frac{0.0048*175+0.0000*250}{0.0167*50}$, or 1. The efficiency of DMU 4 is $\frac{0.0048*210+0.0000*192}{0.0167*60}$, which is also equal to 1. DMU 4 is non-Pareto optimal because its value of output 2 per unit of input is less than the value associated with DMU 3. As can be estimated graphically in Figure 3 or calculated after examining the two expressions above, this value is equal to 3.2 for DMU 4 and equal to 5 for DMU 3. Because the constraints associated with DMU 3 and DMU 4 are both already binding, any deviation from 0 for the decision variable associated with output 2 will exceed the bounds imposed by these constraints. Because DMU 3 has a greater value of output 2 per unit input than DMU 4, any deviation from zero will increase the value of the left hand side of the constraint associated with DMU 3 faster than it will increase the left hand side of the constraint associated with DMU 4. To bring the constraints back to binding but not violated would require changing one of the other two decision variables. However, these DMUs have identical values for output 1 per unit input. This means that deviations to the other two decision variables will impact them identically and cause the constraint associated with DMU 3 to become binding prior to the constraint associated with DMU 4. When the constraint associated with the focal DMU is non-binding, the focal DMU does not have a relative efficiency of one. Therefore, for DMU 4, any non-zero value of the decision variable associated with output two is not optimal. This is why DMUs that are relatively efficient but not Pareto optimal will always have a decision variable equal to zero.

Readers who are already familiar with DEA may have noticed that "non-Archimedean infinitesimals" have not yet been mentioned. Similar to infinity, non-Archimedean infinitesimals are concepts rather than an actual numbers. They represent positive numbers that are smaller in value than any other positive number. Some authors use these infinitesimals in a second stage optimization to distinguish between DMUs that are Pareto optimal and DMUs that are relatively efficient but not Pareto optimal. This is an important distinction that has already been discussed, but it is a distinction that can often be made without recourse to infinitesimals. While DMUs that are relatively efficient but not Pareto optimal always have zero value decision variables, zero value decision variables do not always indicate that the DMU is not Pareto optimal. One way to determine whether the DMU is Pareto optimal or not is to use non-Archimedean infinitesimals. Another way is to examine the sensitivity analysis of the linear program

As can be seen from examining Figure 4, DMU 1 is relatively efficient and Pareto optimal. Figure 8 is a screen capture of the formulation after solving the linear program associated with DMU 1. Note that the decision variable associated with output one is equal to zero.

	A	B	C	D	E	F
1		Output 1	Output 2	Input		Obj
2	Objective Coefficients	102	683	-64.1		-0
3	Decision Variables	0.0000	0.0015	0.0156		RHS
4	Input Constraint	0	0	-64.1	-1	-1
5	DMU 1 Constraint	102	683	-64.1	-0	0
6	DMU 2 Constraint	171	545	-60.8	-0.2	0
7	DMU 3 Constraint	175	250	-50	-0.4	0
8	DMU 4 Constraint	210	192	-60	-0.7	0
9	DMU 5 Constraint	126	456	-59.9	-0.3	0
10	DMU 6 Constraint	46	604	-70.8	-0.2	0

Figure 8: Solution for DMU 1

As stated above, a value of zero for a decision variable at optimality may or may not indicate that the DMU under consideration is not Pareto optimal. In the current example, the analyst can simply plot the data and determine from observation whether or not a particular DMU is Pareto optimal. This is not so easily done with larger problems. The difficulty in interpreting the meaning of a decision variable equal to zero is due to the fact that the linear programs used to solve a DEA often result in multiple optimal solutions. The objective function always has the same set of coefficients as the constraint associated with the focal DMU. This means the two are parallel. If the focal DMU is relatively efficient, this means that this constraint is binding. When a binding constraint is parallel to the objective function, there are usually an infinite number of equally optimal solutions. Either this is the case or the optimal solution occurs at a degenerate point. For the purposes of this paper, it can be assumed that the former is the case rather than the latter.

The difficulty in interpreting the meaning of a decision variable equal to zero is that there may be an equally optimal solution that does not have any decision variables equal to zero. A decision variable with a value of zero only indicates that a relatively efficient DMU is not Pareto optimal if there is no alternate optimal solution available in which the decision variables are non-zero. The purpose of the non-Archimedean infinitesimals is to make certain solutions that would otherwise be equally optimal preferable to others and thus avoid solutions that have decision variables equal to zero. Although this is certainly one option, in practice it requires a second optimization to occur after the first. Instead of performing this second optimization, the analyst can instead note that a decision variable is equal to zero and consult the sensitivity analysis prior to deciding whether this indicates that the DMU is not Pareto optimal. Figure 9 is a screen capture of the sensitivity analysis generated during the solution shown in Figure 8.

Adjustable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
B3	Decision Variables Output 1	0.0000	0.0000	102	0	1E+30
C3	Decision Variables Output 2	0.0015	0.0000	683	0	0
D3	Decision Variables Input	0.0156	0.0000	-64.1	0	1E+30

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
E4	Input Constraint	-1	0	-1	1	1E+30
E5	DMU 1 Constraint	-1.11022E-16	1	0	0.188693126	1
E6	DMU 2 Constraint	-0.150567721	0	0	1E+30	0.150567721
E7	DMU 3 Constraint	-0.41399899	0	0	1E+30	0.41399899
E8	DMU 4 Constraint	-0.654924704	0	0	1E+30	0.654924704
E9	DMU 5 Constraint	-0.266834627	0	0	1E+30	0.266834627
E10	DMU 6 Constraint	-0.22019036	0	0	1E+30	0.22019036

Figure 9: Sensitivity analysis for the primal solution for DMU 1

As can be seen by examining Figure 9, cells G5 and H5 are both zero. The values in cells G4 through H6 indicate by how much the objective coefficients may change before it influences the objective basis. In linear programming, the basis defines which corner of the feasible region is being examined. The fact that the allowable increase and allowable decrease associated with the same decision variable are both equal to zero indicates that any change in the value of the objective coefficient, output 2 in this case, will result in a change in the optimal basis. Stated another way, there are multiple optimal solutions.

When an analyst is interested in distinguishing between those relatively efficient DMUs that are Pareto optimal and those that are not, the first step is determine whether a decision variable is zero at optimality. The analyst should examine the sensitivity analyses of any such DMUs as described above. Although the absence of multiple optimal solutions proves that the DMU is not Pareto optimal, the presence of multiple optimal solutions does not prove that the DMU is Pareto optimal. For this, the use of non-Archimedean infinitesimals is probably the best solution. The interested reader is directed to Cooper (2001) for more information on the use of non-Archimedean infinitesimals.

Conclusions

The stated purpose of this paper is to present DEA in a way that appropriate for both undergraduates and doctoral students. The paper represents a catalog of DEA related topics and issues that are important for any DEA analyst to know which is presented in language that is simple enough for most undergraduate students to understand. The authors believe that this paper represents a good first resource that includes all, or nearly all, of the material required for educators preparing to teach DEA to undergraduate or masters level students. Much care was given in considering which topics should be addressed in depth and which should be alluded to. If the sole purpose of this paper had been undergraduate education, several topics mentioned in passing would have been left out entirely. The reason for mentioning these topics is to increase the rigor of the paper and make it appropriate for doctoral level studies. At the doctoral level, the educator is less interested in obtaining a digest that can be passed along to the students. Instead, the educator is interested in finding a primer that provides general information and citations where more specific information can be located. The authors believe that the paper and associated list of works cited succeed in this regard. In sum, this paper represents a stand alone lesson appropriate for lower level students and a starting point for more advanced learner.

References

Banker, R., Charnes, A., and Cooper, C. “Some Models for Estimating Technical and Scale Inefficiencies in Data Envelopment Analysis.” Management Science, 30.9 (1984): 1078-1092.

Banker, R. and Thrall, R. “Estimation of Returns to Scale Using Data Envelopment Analysis.” European Journal of Operational Research, 62.1 (1992): 74-84.

Charnes, A., Clark, C., and Cooper, C. “A Developmental Study of Data Envelopment Analysis in Measuring the Efficiency of Maintenance Units in the U.S. Air Forces.” Annals of Operations Research, 2.1 (1985): 95-112

Charnes, A. and Cooper, W. “Programming with Linear Fractional Functionals.” Naval Research Logistics Quarterly, 9.3 (1962): 181-186.

Charnes, A., Cooper, W., and Rhodes, E. “Measuring the Efficiency of Decision Making Units.” European Journal of Operational Research, 2.6 (1978): 429-444.

Cooper, C. “Data Envelopment Analysis.” Encyclopedia of Operations Research and Management Science. Gass, S. and Harris, C. (Eds.). Norwell, Massachusetts: Kluwer Academic Publishers (2001): 183-191.

Tavares, G. “A Bibliography of Data Envelopment Analysis (1978-2001).” Rutcor Research Report Piscataway, New Jersey: Rutgers University (2002).