

Paper 198-2011

The Final Frontier: A SAS® Approach to Data Envelopment Analysis

Sabah Sadiq, Institute for Advanced Analytics, North Carolina State University, Raleigh, N.C.

ABSTRACT

The purpose of this paper is to discuss the methodology for using PROC OPTMODEL and macros to rank real estate properties using Data Envelopment Analysis (DEA) and Super Efficiency Models. DEA was used to categorize apartments as efficient or inefficient, in terms of costs and benefits as seen through the buyer's perspective. The Super Efficiency Model was then used to comparatively rank efficient apartments in order of cost effectiveness.

During this process, macros using OPTMODEL index sets and arrays were developed that will compute DEA and Super Efficiency values given data sets containing input and output values. These macros allow SAS/OR® users to seamlessly implement DEA while greatly reducing modeling time and coding errors. PROC APPEND was also included in the macro in order to accumulate efficiency values and produce a graphical comparison of efficiencies.

INTRODUCTION

Ten apartments must be compared in terms of apartment features and costs. Inputs were defined as apartment costs and outputs were defined as apartment benefits. Therefore, the objective is to find apartments with maximized outputs and minimized inputs. Table 1 outlines the data used for this problem:

Table 1: Apartment Features

APARTMENT	DMU	Outputs			Inputs	
		Total Area (ft ²)	Furnished (% included)	Utilities (% included)	Morning Drive (min)	Rent (\$/month)
Campus Edge	1	641	100	100	5	869
Clarion Crossing	2	652	25	30	5	644
Lake Johnson Mews	3	748	30	20	5	725
Legends Cary Towne	4	722	25	10	12	720
Raleigh Town Apartments	5	1200	0	10	9	950
Remington Place	6	1005	0	10	7	765
Renewal Place	7	1800	0	100	11	650
Walnut Creek	8	720	25	50	5	591
Wedgewood Apartments	9	420	100	100	10	620
Westgrove Tower	10	580	100	50	7	600

TERMINOLOGY

DEA is a linear programming model used to evaluate the performance of decision making units (DMU's). A DMU is any unit of interest with similar inputs and outputs that can be compared. DEA has typically been used to evaluate productivity performance of DMU's such as banks, hospitals, and universities. The objective of the linear program is to maximize efficiency, where efficiency is the ratio of weighted outputs to weighted inputs and restricted to a range of 0 to 1. The input and output weights are allowed to vary between models. Therefore, each DMU is allowed to pick weights that will maximize their efficiency score.

When redefining efficiency to represent the costs versus the benefits, as seen through the buyer's perspective, DEA can be used to determine what apartments are worth considering. In order to evaluate the efficiency of the 10 apartments, the DEA linear program must be solved 10 times (to create an efficiency score for each apartment). This analysis requires the following constraints:

1. The denominator must be removed from the objective function to eliminate non-linearity. This will be done by forcing the weighted efficiency denominator to 1.

- The outputs for each DMU cannot exceed the inputs. This will be done by ensuring the weighted outputs minus the weighted inputs is less than or equal to zero.

With these conditions and maximized objective function, the following linear model is used:

$$\text{Maximize } z = \sum_{i=1}^n u_i y_i \quad (1)$$

$$\text{Subject to: } \sum_{j=1}^m v_j x_j = 1 \quad (2)$$

$$\sum_{i=1}^n u_i y_i - \sum_{j=1}^m v_j x_j \leq 0 \quad (3)$$

$$u_i \geq 0, v_j \geq 0 \quad (4-5)$$

Where:

z = Efficiency of DMU under consideration

u_i = n output coefficients of DMU under consideration

y_i = n output weighting coefficients for DMU under consideration

v_j = m input coefficients for DMU under consideration

x_j = m input weighting coefficients for DMU under consideration

Equation (1) creates the efficiency score for DMU i . Equation (2) forces the weighted inputs of DMU i (originally in the objective function denominator) equal to 1. Equation (3) ensures that the weighted outputs cannot exceed the weighted inputs for all DMU's (capping efficiency to 1).

THEORY

DATA ENVELOPMENT ANALYSIS

DEA is a non-parametric approach to analyze efficiency. The advantage of this approach over parametric methods (i.e. regression) is that no assumptions are needed regarding the relationship between the inputs and outputs. Thus, complex or ambiguous relationships can still be modeled. Another advantage is that DEA compares DMU's to the most efficient producer, whereas, traditional regression compares DMU's to the average producer.

The DEA model searches for the points with the lowest input needed for a given set of outputs. Collectively, these points make up the efficiency frontier. If a DMU lies on the efficiency frontier, it achieves a maximum output given its inputs and reaches an objective function of one. Any objective value less than one is deemed inefficient and can be compared to efficient DMU's using shadow prices. This allows us to evaluate what other DMU's are more efficient given the same weights as the inefficient DMU. Although this is useful in productivity performance, many situations arise in which DMU's need to be ranked.

SUPER EFFICIENCY MODEL

Although the DEA model is useful in determining which units are efficient, it does not provide a way to rank which units are more efficient. One proposed solution to this problem is to remove the restriction that caps efficiency to one. This Super Efficiency Model allows us to examine the degree to which DMU's surpass the efficient frontier. This allows efficient DMU's to be ranked. Note that the super efficiency ratings for inefficient units are less than one and thus remain unchanged from their DEA values. In order to perform this analysis, the constraint forcing the inputs to exceed the outputs (Eq. 3) must be relaxed for the DMU under consideration.

THE MACRO

In order to simplify the process of creating multiple linear models for each DMU, a macro was used (see References). The macro performs all data handling, DEA modeling, report generation, and graphical output. All that is needed are two data sets containing the DMU's and their respective inputs (input data set) and outputs (output data set). Note that the data sets must contain a column containing the DMU number and remaining columns must be named as "INPUTX" or "OUTPUTX", respectively. Figure 1 represents the general structure needed for this program:

Figure 1: Input Data Set

	DMU	INPUT1	INPUT2
1	1	5	869
2	2	5	644
3	3	5	725
4	4	12	720
5	5	9	950
6	6	7	765
7	7	11	650
8	8	5	591
9	9	10	620
10	10	7	600

After passing these data sets to the macro, the data was sorted and analyzed to create counting variables (quantity of DMU's, quantity of inputs, quantity of outputs). A do loop was then modeled to generate a DEA efficiency for every DMU. Within this loop, the OPTMODEL language was used to create the following sets and parameters:

Table 1: Sets and Parameters

Set/Parameter	Description
Inputs	Quantity of Inputs
Outputs	Quantity of Outputs
DMU	Quantity of DMU's
n	Current DMU
Scale	Input data set, indexed by DMU
Objective	Output data set, indexed by DMU

After creating the above sets and parameters, the input and output data sets were read into the Scale and Objective parameters:

```
/*Read in Data*/
read data &PassedInputs
  into DMU = [DMU]
  {d in Inputs} < scale[DMU,d]=col("INPUT"||d) >;

read data &PassedOutputs
  into DMU = [DMU]
  {e in Outputs} < objective[DMU,e]=col("OUTPUT"||e) >;
```

The OPTMODEL language was then used to create input and output weighting variables x and y , respectively. These weights along with the sets and parameters were used to solve the DEA model. The objective function, Efficiency, sums the product of output coefficients and weights, for the current DMU in set n . The Scaling constraint forces the product of input coefficients and weights equal to one, for the current DMU in set n . The CapToOne constraint ensures that the product of the output coefficients and weights minus the product of input coefficients and weights is less than or equal to zero, for all DMU's. The following is the OPTMODEL language used for these equations:

```

/*Declare Variables and System of Equations*/
var x{n,Inputs}>=0, y{n,Outputs}>=0;
Max Efficiency = sum{i in n}(sum{j in Outputs}(Objective[i,j]*y[i,j]));
con Scaling: sum{i in n}(sum{j in Inputs}(Scale[i,j]*x[i,j])) = 1;
con CapToOne {i in DMU}: sum{a in n}(sum{b in Outputs}
    (Objective[i,b]*y[a,b])) - sum{c in n}(sum{d in
    Inputs}(Scale[i,d]*x[c,d])) <= 0;

```

After solving the above system of equations, two data sets were outputted containing the efficiency value and benchmarking units:

```

/*Create Benchmarking Dataset*/
create data DualOutput from DMU = &LOOP_COUNT {h in DMU} <
col("BENCHMARK"||h)=CapToOne[h].dual >;

```

```

/*Create Efficiency DataSet*/
create data OptOutput from DMU = &LOOP_COUNT Efficiency=Efficiency;

```

The OPTMODEL procedure was then ended and PROC APPEND was used to add the temporary data sets to summarized tables of benchmarks and efficiencies, each time the macro looped through a DMU.

```

/*Append Report Set*/
proc append base=Benchmarks data=DualOutput;
run;

```

```

/*Append Plotting Set*/
proc append base=Efficiencies data=OptOutput;
run;

```

After looping through all DMU's, the summarized efficiency table was manipulated to create two columns, one containing efficient DMU's and one containing inefficient DMU's. PROC SGPLOT was then used to create a bar chart containing RGB transparency levels for efficient and inefficient DMU's. PROC PRINT was also used to output the exact efficiency values and benchmarking units.

In order to calculate the super efficiencies, a new macro was created that altered the CapToOne constraint so that it looped through all DMU's except the current DMU being considered (set n). The DIFF operator was used to return an index set containing all DMU's except for n. This ensured that the DMU being considered was allowed to surpass one, while the other DMU's were still capped at one.

```

/*Altered Super Efficiency CapToOne Constraint*/
con CapToOne {i in DMU diff n}: sum{a in n}(sum{b in Outputs}
    (Objective[i,b]*y[a,b])) - sum{c in n}
    (sum{d in Inputs}(Scale[i,d]*x[c,d])) <= 0;

```

THE SOLUTION

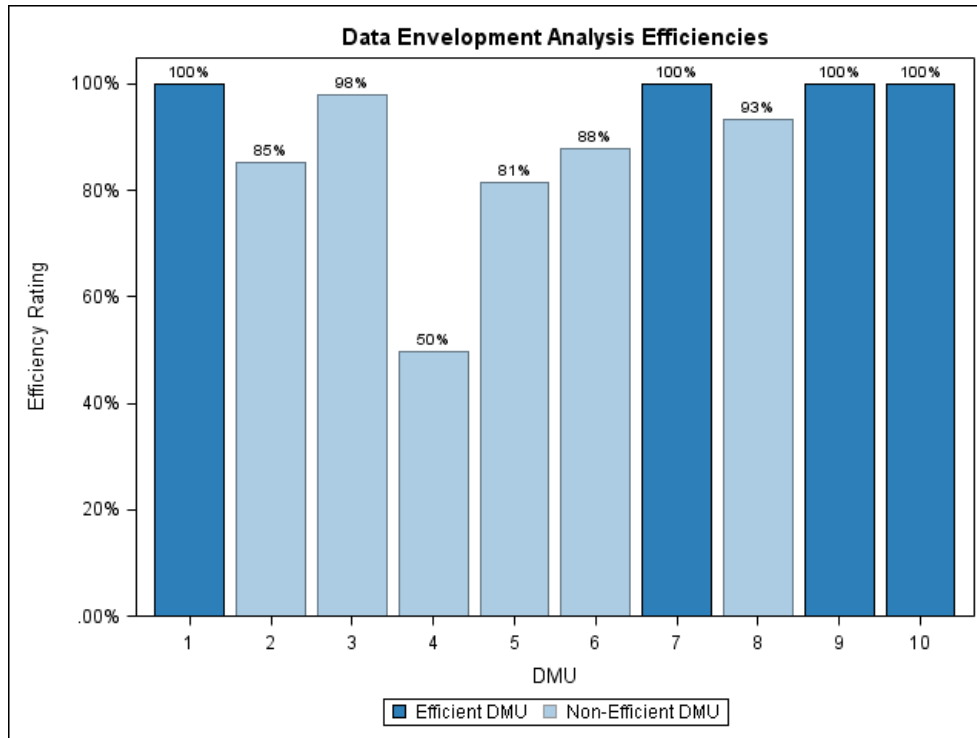
After manipulating column headings, input and output data sets for the apartment problem were passed to the DEA and SE macros:

```

%DEA_Macro(PassedInputs=inputs, PassedOutputs=outputs);
%SE_Macro(PassedInputs=inputs, PassedOutputs=outputs);

```

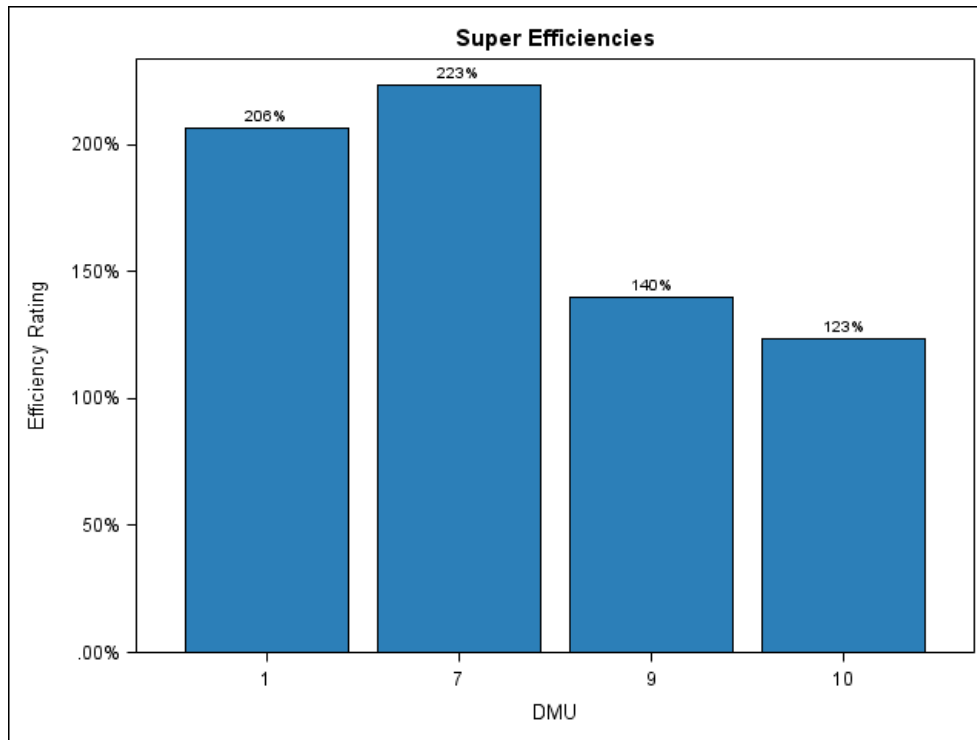
The DEA macro produced the following graphical output of apartment efficiencies:

Figure 2: Apartment DEA Efficiency Ratings

Based on the above graph, DMU 1 (Campus Edge), DMU 7 (Renewal Place), DMU 9 (Wedgewood Apartments), and DMU 10 (Westgrove Tower) are all efficient apartments.

In order to rank apartments, the super efficiency output was analyzed:

Figure 3: Apartment Super Efficiency Ratings



Based on the above super efficiencies, apartments were ranked in the following order:

Table 3: Efficient Apartment Rankings

Rank	DMU	Apartment
1	7	Renewal Place
2	1	Campus Edge
3	9	Wedgewood Apartments
4	10	Westgrove Tower

CONCLUSION

Although DEA has its traditional applications in productivity analysis, it is also a competitive tool for many cost benefit problems. When transforming the traditional DEA outputs and inputs to a cost benefit analysis, the applications become much more widespread. PROC OPTMODEL provides an ideal way to implement this type of analysis to reach valid conclusions. PROC APPEND can be used to output data sets that contain all objective function values for plotting and report generation. The macros presented in this paper take advantage of these facts and allow SAS/OR® users to perform DEA/SE analyses while eliminating repetitive linear programming coding.

REFERENCES

- Adler, N (2002). "Review of ranking methods in the data envelopment analysis context". *European journal of operational research* (0377-2217), 140 (2), p. 249.
- Estellita, Lins, Marcos Pereira (2005). "Real Estate Appraisal: A Double Perspective Data Envelopment Analysis Approach". *Annals of operations research* (0254-5330), 138 (1), p. 79.
- Kuosmanen, Timo (2007). *Ecological economics* (0921-8009), 62 (1), p. 56.
- Sadiq, S (2011). *DEA and Super Efficiency Macros*. Retrieved February 24, 2011, from <https://sites.google.com/site/deamacro/>
- SAS Publishing (2010). *SAS/OR® 9.22 User's Guide: Mathematical Programming*. Cary, NC: SAS Institute

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Name: Sabah Sadiq
Enterprise: Institute for Advanced Analytics
Address: 920 Main Campus Drive, Suite 530
City, State ZIP: Raleigh, NC 27606
E-mail: ssadiq07@gmail.com
Web: <http://www.linkedin.com/pub/sabah-sadiq/8/11/31b>

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.