

Paper 274-25

An Extension to SAS/OR® for Decision System Support

Ali Emrouznejad

Higher Education Funding Council for England,
Northavon house, Coldharbour Lane,
Bristol, BS16 1QD
U.K.

ABSTRACT

This paper explores the use of the optimisation procedures in SAS/OR software with application to the decision system support using Data Envelopment Analysis (DEA) techniques.

Data Envelopment Analysis is a linear programming method for assessing the efficiency and productivity of Decision Making Units. Over the last decade DEA has gained considerable attention as a managerial tool for measuring performance of organisations and it has been used widely for assessing the efficiency of public and private sectors such as banks, airlines, hospitals, universities and manufactures. As a result, new applications with more variables and more complicated models are being introduced.

On the other hand, the SAS® System has numerous optimisation routines that handle the standard optimisation problems such as linear programming with all types of constraints. To facilitate the use of DEA by SAS users, a SAS/DEA code was implemented in the SAS programming language.

The SAS macro developed in this paper selects the chosen variables from a SAS data file and constructs a Linear-programming model based on the selected DEA.

An example is given to illustrate how could one use the code to measure the efficiency of organisations.

KEYWORDS:

SAS, Data Envelopment Analysis, Linear Programming

INTRODUCTION

Data Envelopment Analysis (DEA)¹ is a linear programming method for assessing the efficiency and productivity of Decision Making Units. DEA continues to grow in importance as managerial tools become more reliable and handle performance measurement of organisations. As a result, new applications with more variables and more complicated models are being introduced. On the other hand, the SAS System has numerous optimisation routines that handle the standard optimisation problems such as linear programming with all types of constraints².

The main purpose of this paper is to implement a SAS/DEA System. This is followed by an illustration of how the user could use this system.

The paper unfolds as follows. Next section describes the procedures of the SAS System and SAS/OR products. In section 2 the DEA models and method of calculations in DEA are explained. The remainder of this paper illustrates features that have been added to the SAS System for the purpose of DEA modeling, call it SAS/DEA. These include procedures for data handling, model building and reporting the results of DEA model. Finally, this paper uses a small data set to show some of the features of the SAS/DEA system.

ABOUT SAS

Today SAS is recognised as a one of the lead packages for statistical analysis and as a powerful tool for data base systems in many organisations, both in public and private sectors. SAS users come from every major industry (banking to

pharmaceuticals, manufacturing to telecommunications, ...) all with the same basic needs to make better strategic decisions and to gain a competitive edge.

There are many applications in SAS that the users recognised as powerful tools in organisational management. For example the SAS/OR System has numerous optimisation procedures which handle the standard problems such as linear and non-linear programming with all types of constraints. These capabilities are exposed to the SAS users in a variety of places such as in procedures LP and NLP³.

In addition to the standard procedures available in SAS System applications such as neural network, simulation⁴ and control project management⁵ are introduced. The absence of Data Envelopment Analysis code in SAS force the users to use the other packages while the SAS/OR is very powerful and has capability of handling all types of DEA models. Therefore this paper aims to introduce a new application in SAS System for performance measurement of Decision Making Units using Data Envelopment Analysis (DEA) technique.

ABOUT DEA

Decision Making Units (DMUs) are units of organisations such as banks, universities, and hospitals, which typically perform the same function. A DMU usually uses a set of inputs (resources) to secure a set of outputs (products).

DEA is a method for measuring efficiency of DMUs using linear programming techniques to “envelop” observed input - output vectors as tightly as possible⁶. One main advantage of DEA is that it allows several inputs and several outputs to be considered at the same time. In this case, efficiency is measured in terms of inputs or outputs along a ray from the origin.

Assume a set of observed DMUs, {DMU *j*; *j*=1,...,*n*}, associated with *m* inputs, {*x*_{*ij*}; *i*=1,...,*m*}, and *s* outputs, {*y*_{*rj*}; *r*=1,...,*s*}. In the method originally proposed by Charnes, Cooper, Rhodes (1978)¹ the efficiency of the DMU *j*₀ is defined as follows.

Model 1. Output oriented - CRS model

Max *h*

s.t.

$$\sum \lambda_j x_{ij} + S_i^+ = x_{ij_0} \quad \forall i$$

$$\sum \lambda_j y_{rj} - S_r^- = h y_{rj_0} \quad \forall r$$

$$S_i^+, S_r^- \geq 0 \quad \forall i, \forall r$$

$$\lambda_j \geq 0 \quad \forall j.$$

Where;

*x*_{*ij*} = the amount of the *i*th input at DMU *j*,

*y*_{*rj*} = the amount of the *r*th output from DMU *j* and

*j*₀ = the DMU to be assessed.

If *h*^{*} is the optimum value of *h*, then DMU *j*₀ is said to be Pareto

efficient iff $h^+ = 1$ and the optimal values of S_i^+ & S_r^+ are zero for all i & r . This means no other DMU or combination of DMUs exist which can produce at least the same amount of output as DMU j_0 , with less for some resources and/or no more for any other resources.

In Model (1), S_i and S_r represent slack variables. Thus a slack in an input i , i.e. $S_i^+ > 0$, represents an additional inefficiency use of input i . A slack in an output r , i.e. $S_r^- > 0$, represents an additional inefficiency in the production of output r .

The DEA Model (1) is known as output - oriented model because it expands output of DMU j_0 within the production space. It should be solved n times once for each DMU being evaluated to generate n optimal values of (h^+, λ^+) .

For DMU j_0

If radial expansion is possible Model (1) will yield $h_{j_0}^+ > 1$,

If radial expansion is not possible Model (1) will yield $h_{j_0}^+ = 1$.

The positive elements of the optimal values in λ identify the set of dominating DMUs located on the constructed production frontier, against which the DMU j_0 is evaluated. This subset of DMUs is called "peers" to DMU j_0 . From Model (1) it is clear that the model defines the relative efficiency of a DMU in terms of output maximisation. The input oriented model of DEA can be defined in a similar way.

Model 2. Input oriented - CRS model

Min ϕ

s. t.

$$\sum_j \lambda_j x_{ij} + S_i^+ = \phi x_{ij_0} \quad \forall i$$

$$\sum_j \lambda_j y_{rj} - S_r^- = y_{rj_0} \quad \forall r$$

$$S_i^+, S_r^- \geq 0 \quad \forall i, \forall r$$

$$\lambda_j \geq 0 \quad \forall j.$$

Where;

x_{ij} = the amount of the i^{th} input at DMU j ,

y_{rj} = the amount of the r^{th} output from DMU j and

j_0 = the DMU to be assessed.

Assume that ϕ^* is the optimum value of ϕ . DMU j_0 is said to be Pareto efficient iff $\phi^* = 1$ and the optimal value of S_i^+ and S_r^- are zero ($\forall i, r$).

Besides developing DEA in theory, practitioners in a number of fields have quickly recognised that DEA is a useful methodology for measuring productivity and efficiency. Therefore a number of software was developed. Some of the well known packages which researchers frequently use is: Warwick DEA Software, Frontier Analysis, IDEAS and GAMAS DEA⁷. The first three are commercial and they allow some DEA analysis to be undertaken, there are only few options to change in details of the model used^{8,9}. Warwick DEA software has ability to calculate the effect of weight restriction and target setting¹⁰. The advantage of GAMS DEA is that it has the capability of building other DEA models. However GAMS is not popular software in organisational management. Many of the organisations such as Banks, Universities, Hospitals, Telecommunications, Airlines and so on use SAS software as a data base system or statistical analysis. The SAS/DEA introduced in this paper provides a powerful management tool for assessing the efficiency of organisations in SAS system.

SAS/DEA

The SAS/ DEA software developed in this paper solves DEA models. The program can handle both input minimisation and output maximisation. To enhance model there are several parameters. The user can select the desired parameters according to the particular model that is required. Users familiar

with SAS can add their own features to build other DEA models. Users not familiar with SAS need only to run the program with their model specification prior running the system.

The SAS/DEA requires two initial data sets that contain the input and output variables for observed units. The data describing inputs/outputs must be presented in the format that variables appear in columns and units in rows and saved as .txt file (Tab delimited). The program has the ability to accommodate unlimited number of inputs/outputs with unlimited number of DMUs. The only limitation is the memory of computer used to run the SAS/DEA.

The SAS/DEA software then converts data sets to a DEA model selected. Based on the data and parameters specified in the SAS/DEA, the code first create the usual linear program then it use "Proc LP" to solve the model. Results then will be transferred to report files.

The SAS/DEA produces a table of efficiencies of DMUs. It also supplies much other valuable information including lambdas in primal and weights in dual DEA models. These information are very useful for analysing the inefficient units, where the source of inefficiency comes from and how could improve an inefficient unit to the desired level.

Figure (1) illustrates the data flow in the SAS/DEA. It shows

- the input and output variables data sets that are required for SAS/DEA.
- the parameters that SAS/DEA needs to define the model.
- The macros; (%data, %model and %report) and
- the result data sets that are available for print or can be used for further analysis.

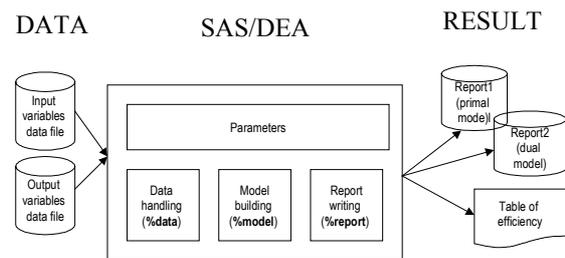


Figure 1. Data flow in SAS/DEA

In the rest of this paper the procedure of implementation of SAS/DEA together with an example are explained.

DEFINITION OF TERMS AND TYPOGRAPHICAL CONVENTIONS

In the rest of this paper and particularly in the SAS/DEA code we will see several type styles used. Style conventions are summarised below:

- Courier font** is used to show example of SAS statements. In most cases, this paper uses lowercase type for SAS code. The user can enter own SAS code in lowercase, uppercase or a mixture of the two. Enter any titles and footnotes exactly as you want them to appear on the printout.
- _Underscore_** Variable name that are surround by “_” are specifically used as parameters to the SAS. In all case these variables must be used without any change.
- _Underscore** Variable name that are started with “_” are specifically used as parameters to the SAS/DEA. In all case these variables can be redundant if default values are adopted.

The terms in Table 1 are used for defining parameters needed in SAS/DEA. All these parameters are arbitrary and in absence of them the default value will be assigned.

Table 1. The terms used for defining parameters in SAS/DEA

Parameter	Meaning
<code>_InpData</code>	Specifies the file name of input variables
<code>_NoInut</code>	Specifies the number of the input variables in the optimisation DEA model
<code>_OutData</code>	Specifies the file name of output variables
<code>_NoOutput</code>	Specifies the number of the output variables in the optimisation DEA model
<code>_Orienta</code>	This parameter is used to specify the orientation of the DEA model to be solved. "InputMin" for input minimisation model and "OutputMax" for output maximisation model

The SAS/DEA runs three macros for data handling (%data), model building (%model) and report writing (%report).

DATA HANDLING (%DATA)

This part of SAS/DEA reformats the data to a suitable format that can be used in SAS/OR. The SAS/DEA requires two data sets including input and output variables. Both data sets need to be ".txt" files saved as "text tab delimited". Variables must be presented in columns and units in rows. The unit names must start with a letter and may contain up to 50 characters and must be listed in the first column of the data file. The other columns are including numeric values of variables. These variables can be entered into the file in any order. The first row of the data file represents the name of the variables. An example of data file for input variables is:

Unit	Input1	Input2
DMU1	10	20
DMU2	30	15
DMU3	15	40

The data file for output variable has similar format.

There are four parameters prior to calling the data macro:

`_InpData`, `_NoInput`, `_OutData`, `_NoOutput`.

The purpose of each parameter is presented in Table 1.

SAS procedure for data handling is as follows.

```
%macro data;
  data NY;
    infile &_OutData delimiter='09'X missover
    dsd ;
    array NYarray(&_nOutput) $ NY1-
    NY&_nOutput ;
    length unitname $ 50 ;
    input unitname $
      NYarray(*) $ ;
    if _n_ eq 1 then output ;
    drop unitname;
  run;

  data AY DMUs;
    infile &_OutData delimiter='09'X missover
    dsd;
    array AYarray(&_nOutput) AY1-AY&_nOutput;
    length unitname $ 50 ;
    input unitname $
      AYarray(*) ;
    if _n_ gt 1 then output ;
  run;

  data NX;
    infile &_InpData delimiter='09'X missover
    dsd ;
    array NXarray(&_nInput)$ NX1-NX&_NInput ;
```

```
length unitname $ 50 ;
input unitname $
  NXarray(*) $ ;
if _n_ eq 1 then output;
drop unitname;
run;

data AX;
  infile &_InData delimiter='09'X missover
  dsd ;
  array AXarray(&_nInput) AX1-AX&_nInput;
  length unitname $ 50 ;
  input unitname $
    AXarray(*) ;
  if _n_ gt 1 then output;
  call symput ('_nDmu',_n_-1);
run;

data DMUs;
  set AY (keep=UnitName);
  Uj0=_n_;
run;
%mend data;
```

MODEL BUILDING (%MODEL)

This part of SAS/DEA builds the requested DEA model in the format suitable for SAS/OR and for purpose of calling "Proc LP". There is one parameter prior calling the procedures:

`_Orienta`

For example, for solving a standard input minimisation model the user should set the parameters to:

`_Orienta=InputMin;`

and for solving output maximisation model the user should set the parameters to:

`_Orienta=OutputMax;`

SAS procedure for model building is as follows.

```
%macro model;
%local i j j0;
data MODEL1;
  %let _nUnit=&_nDmu;
  array NYarray(&_nOutput)$ NY1-NY&_nOutput ;
  array AYarray(&_nOutput) AY1-AY&_nOutput ;
  array NXarray(&_nInput) $ NX1-NX&_NInput ;
  array AXarray(&_nInput) AX1-AX&_nInput;
  array Jarray(&_nUnit) U1-U&_nUnit ;

  length _row_ $ 50 _col_ $ 50 _type_ $ 8 ;
  keep _row_ _col_ _type_
  _coef_ ;

  J0=&jj; /* Jo is unit under assessment*/
  if &_Orienta='InputMin' then do;
    _type_='MIN';
    _row_='OBJ';
    _col_='.';
    _coef_='.';
    output;
    _type_='.';
    _row_='OBJ';
    _col_='FI';
    _coef_=1;
    output;
    _type_='.';
    _row_='OBJ';
    _col_='rhs_';
    _coef_=0;
    output;
  end;

  link ReadNX;
  do j= 1 to &_nUnit;
    link ReadAX;
```



```

data=EffJ0; run;
  data lp2; set lp2; Uj0=&jj;run;
  proc datasets nolist; append base=Report1
data=lp2;run;
  data lp3; set lp3; Uj0=&jj;run;
  proc datasets nolist; append base=Report2
data=lp3;run;
%end;
%report;
%mend sasdea;

```

In the above code the "%sasdea macro" is used to manage all previously explained codes including data handling, model building and report writing. To get the result, user needs to set up the parameters and run only one statement:

```
%sasdea;
```

AN EXAMPLE OF SAS/DEA

In this section I present a simple example of two inputs two outputs and 6 DMUs for illustration of SAS/DEA. This example were taken from Sexton¹¹, page 19, and will allow us to compare the SAS/DEA results with that of reported by Sexton. He used this example for assessing six nursing homes. A DEA model with 2 inputs and 2 outputs were used.

The inputs are

- Staff hours per day (StHr), including nurses, physician, therapists, and so on.
- Suppliers per day (Supp) which are measured in thousands of dollars.

The outputs are

- Total Medicare plus Medicaid-reimbursed patient days (MCPD)
- Total privately paid patient days (PPPD).

The inputs are measured on a daily basis, while the outputs are annual totals. It is possible to express all inputs and outputs in the same units, but this is not necessary, since the basic DEA model assumes constant returns to scale. For more information see pages 19-23 of Sexton¹¹.

The input and output variables are saved in files "InVar.txt" and "OutVar.txt" respectively and are as follows.

"InVar.txt" file		
DMU	StHr	Supp
A	150	0.2
B	400	0.7
C	320	1.2
D	520	2.0
E	350	1.2
F	320	0.7

"OutVar.txt" file		
DMU	MCPD	PPPD
A	14000	3500
B	14000	21000
C	42000	10500
D	28000	42000
E	19000	25000
F	14000	15000

A user needs to set the parameters as required and run the following code for input orientation model.

```

%let _InData='C:\sasdea\InVar.TXT';
%let _nInput=2;
%let _OutData='C:\sasdea\Outvar.TXT' ;
%let _nOutput=2;
%let _Orienta='InputMin';
%sasdea;

```

The above code manages to get the result based on the specified parameters and data files.

SAMPLE RESULTS FROM SAS/DEA

The result of running above code is presented below. The third column shows the efficiency score.

The SAS System		
OBS	UNITNAME	EFF
1	A	1.00000
2	B	1.00000
3	C	1.00000
4	D	1.00000
5	E	0.97750
6	F	0.86745

As this table shows units E and F are inefficient units, all other units are efficient. This is exactly matched with the results reported by Sexton in Table 5 (page 22). Unit F is less efficient than unit E. The best that unit F can do is 0.867, while the maximum of unit E is 0.977.

SAS/DEA also creates a wealth of information and saved them in two files, report1 and report2. By using these in SAS system we can specify a management strategy indicating by how much each unit should reduce each input and/ or increase each output in order to become perfectly efficient. To see this we need the value of slacks and lambdas in primal models (1 or 2) and value of weights in dual to each. These all are saved in files report1 and report2 respectively. It is important to use these reports effectively to derive the measure required for managerial purpose.

For example the lambdas' value for unit E can be obtained from report1 by using fields _var_ and _value_. It can be seen that the value of lambdas for unit E are as follows

The SAS System			
OBS	_VAR_		_VALUE_
3	LA A		0.20000
4	LA B		0.08048
5	LA C		0.00000
6	LA D		0.53833
7	LA E		0.00000
8	LA F		0.00000

What these values suggest is that the target for unit E has an input output levels that are 0.2 times those for unit A plus 0.8048 times those for unit B plus 0.53833 those for unit D. Therefore the optimum value of input output variables for unit E are:

StHr	Supp	MCPD	PPPD
342	1.17	19000	25000

Thus unit E should try to reduce its input levels by 2.3% without reducing either output levels. (Note that, 342 equals 97.7% of 350 and 1.17 equals 97.7% of 1.2).

CONCLUSION

Today many organisations recognised SAS as a one of the lead packages for data base system and statistical analysis. In

particular, optimisation procedures in SAS/OR are exposed to the user in a variety of places such as "Proc LP" and "Proc NLP". Therefore many applications such as neural network and control project management are introduced. SAS/DEA as introduced in this paper is a new application in SAS/OR that is a powerful managerial tool for measuring the efficiency and productivity of Decision Making Units.

The SAS/DEA application has several advantages including:

- It can be used by SAS users with no extra cost to buy any DEA software.
- It is very powerful with no limitation on the input and output variables or number of DMUS. The only limitation is the memory and disk space of the computer uses.
- It is flexible to add other DEA model.
- A SAS/DEA for end user is as ease as to run one statement:

```
%sasdea.
```
- And importantly is SAS has powerful managerial tools for report writing and statistical analysis as well as optimisation procedures. The SAS/DEA report's files can directly feed to other SAS routines for further analysis. The results can dynamically be used in conjunction with cost reduction or profit maximisation in decision system support.

REFERENCES

- ¹ Charnes, A., W. W. Cooper, and E. Rhodes. (1978), "Measuring the Efficiency of Decision Making Units," *European Journal of Operations Research* 2(6) pp 429-44.
- ² SAS Institute Inc. (1989), *SAS/OR User's Guide, Version 6*, First Edition, Cary, NC: SAS Institute Inc.
- ³ Trevor Kearney (1999), "Advances in Mathematical Programming and Optimisation in the SAS System," *SUGI24 Proceedings*, Cary, NC: SAS Institute Inc.
- ⁴ Cohen, Marc-david and Phil Meanor (1995), *QSIM Application Discrete Event Queuing Simulation, Release 6.11*, Cary, NC: SAS Institute Inc.
- ⁵ SAS Institute Inc. (1993), *SAS/OR User's Guide, Project Management, Version 6*, First Edition, Cary, NC: SAS Institute Inc.
- ⁶ Boussofiane, A, R. Dyson and E Thanassoulis, (1991), "Applied data envelopment analysis," *European Journal of Operational Research* vol. 52, pp. 1-15.
- ⁷ Olesen and Petersen (1995), "A presentation of GAMS for DEA," *Computers & Operations Research* 23(4) pp 323-339.
- ⁸ Hollingsworth, B. (1997), "A review of data envelopment analysis software," *The Economic Journal* 107(443) pp 1268-70.
- ⁹ Hollingsworth, B. (1999). "Data Envelopment Analysis and Productivity Analysis: a review of the options," *The Economic Journal* 109(456) pp 458-62.
- ¹⁰ Thanassoulis E. and A. Emrouznejad (1998), *Warwick DEA Software, User's Guide*, ISBN: 0902610635, Warwick Business school, Warwick University, UK.
- ¹¹ Sexton, Thomas R. (1986), "The Methodology of Data Envelopment Analysis," in Silkman, Richaed H. (Editor) *Measuring Efficiency: An Assessment of Data Envelopment Analysis. New Directions for Program Evaluation*, no 32, San Francisco: Jossey-Bass, 1986.

ACKNOWLEDGMENTS

The author wishes to express his thanks to John Thompson and

Judy Akinbolu for their valuable comments and suggestions. Acknowledgement is also due to "Higher Education Funding Council for England" for their financial support.

CONTACT INFORMATION

Author:

Ali Emrouznejad
 Higher Education Funding Council for England,
 Northavon house, Coldharbour Lane,
 Bristol, BS16 1QD
 U.K.
 Email: A.Emrouznejad@warwick.ac.uk
 Web: <http://www.warwick.ac.uk/~bsrlu/>