FEAR 1.0: A Software Package for Frontier Efficiency Analysis with R

Paul W. Wilson*

November 2005

Abstract

This paper describes a software package for computing nonparametric efficiency estimates, making inference, and testing hypotheses in frontier models. Commands are provided for bootstrapping as well as computation of some new, robust estimators of efficiency, etc.

JEL Classification nos. C14, C15, C44

Keywords: nonparametric, DEA, efficiency, software

*Department of Economics, University of Texas at Austin, 1 University Station C3100, Austin, Texas 78712 USA; email wilson@eco.utexas.edu.
1 Introduction

An extensive literature on measurement of efficiency in production has evolved from the pioneering work of Debreu (1951) and Farrell (1957). A large part of this literature uses linear-programming based measures of efficiency along the lines of Charnes et al. (1978) and Färe et al. (1985); these methods have been termed Data Envelopment Analysis (DEA).

Gattoufi et al. (2004) cite more than 1,800 articles on DEA published in more than 490 refereed journals. DEA and similar non-parametric estimators offer numerous advantages, the most obvious being that one need not specify a (potentially erroneous) functional relationship between production inputs and outputs. Until very recently the nonparametric efficiency literature has ignored statistical issues such as inference, hypothesis testing, etc. However, statistical inference and hypothesis testing are now possible with DEA and other nonparametric efficiency estimators due to results surveyed by Simar and Wilson (2000b, 2005b).

Standard software packages (e.g., LIMDEP, STATA, TSP) used by econometricians do not include procedures for DEA or other nonparametric efficiency estimators. Several specialized, commercial software packages are available, as well as a small number of non-commercial, free-ware programs; these have been reviewed by Hollingsworth (1997, 1999, 2004) and Barr (2004). Academic versions of the commercial packages range in price from about US$280 to US$1500 at current exchange rates.

To varying degrees, the existing packages perform well the tasks they were designed for. Each includes facilities for reading data into the program, in some cases in a variety of formats, and procedures for estimating models that the authors have programmed into their software. A common complaint heard among practitioners, however, runs along the lines of “package XYZ will not let me estimate the model I want!” The existing packages are designed for ease of use (again, with varying degrees of success), but at a cost inflexibility, limiting the user to models and procedures the authors have explicitly made available. Moreover, none of the existing packages include procedures for statistical inference. Although the asymptotic distribution of DEA estimators is now known (see Kneip et al., 2003, for details) for the general case with \( p \) inputs and \( q \) outputs, bootstrap methods remain the only useful approach for inference. Yet, none of the existing packages include procedures for bootstrapping in
frontier models. Similarly, none of the existing packages provide procedures for the newly-developed robust alternatives to DEA surveyed in Simar and Wilson (2005b).

FEAR 1.0 consists of a software library that can be linked to the general-purpose statistical package R. The routines included in FEAR allow the user to compute DEA estimates of technical, allocative, and overall efficiency while assuming either variable, non-increasing, or constant returns to scale. The routines are highly flexible, allowing measurement of efficiency of one group of observations relative to a technology defined by a second, reference group of observations. Consequently, the routines can be used to compute estimates of Malmquist indices and their components in any of the decompositions that have been proposed, scale efficiency measures, super-efficiency scores along the lines of Andersen and Petersen (1993), and other measures that might be of interest.

Commands are also included to facilitate implementation of the bootstrap methods described by Simar and Wilson (1998, 2000a). These features can be further used to implement methods of inference for Malmquist indices as in Simar and Wilson (1999), statistical tests for irrelevant inputs and outputs or aggregation possibilities as described in Simar and Wilson (2001b), as well as statistical tests of constant returns to scale versus non-increasing or varying returns to scale as described in Simar and Wilson (2001a). A routine for maximum likelihood estimation of a truncated regression model is included for regressing DEA efficiency estimates on environmental variables as described in Simar and Wilson (2005a). In addition, FEAR includes commands to compute free-disposal hull (FDH) efficiency estimates (Deprins et al., 1984), to perform outlier analysis using the methods of Wilson (1993), and to compute the new, robust, root-\(n\) consistent order-\(m\) efficiency estimators described by Cazals et al. (2002). Most of these features are unavailable in existing software packages.

2 Where to Get FEAR and R

R is a language and environment for statistical computing graphics. It is an implementation of the S language developed at Bell Laboratories, but unlike the commercial version of S marketed as S-Plus by Lucent Technologies, R is freely available under the Free Software Foundation’s GNU General Public License. According to the R project’s web pages (http://www.r-project.org),
“R provides a wide variety of statistical... and graphical techniques, and is highly extensible.... One of R’s strengths is the ease with which well-designed publication-quality plots can be produced, including mathematical symbols and formulae where needed.... R is an integrated suite of software facilities for data manipulation, calculation and graphical display. It includes (i) an effective data handling and storage facility; (ii) a suite of operators for calculations on arrays, in particular matrices; (iii) a large, coherent, integrated collection of intermediate tools for data analysis; (iv) graphical facilities for data analysis and display either on-screen or on hardcopy; and (v) a well-developed, simple and effective programming language which includes conditionals, loops, user-defined recursive functions and input and output facilities.”

R includes an online help facility and extensive documentation in the form of manuals included with the package. In addition, several books describing uses of R are available (e.g., Dalgaard, 2002; Venables, 2002; and Verzani, 2004); see also the recent review by Racine and Hyndman (2002). The current version of R can be downloaded from http://lib.stat.cmu.edu/R/CRAN. Pre-compiled binary versions are available for a variety of platforms; at present, however, FEAR is being made available only for the Microsoft Windows (NT, 95 and later) operating systems running on Intel (and clone) machines.

After downloading and installing R, FEAR can be downloaded from http://www.eco.utexas.edu/faculty/Wilson/Software/FEAR/fear.html

A Command Reference and User’s Guide can also be downloaded from this site; the User’s Guide gives instructions for installing FEAR into R, as well as a number of examples illustrating some of the capabilities of FEAR. The web site also contains licensing information; use of FEAR is free for academic purposes.

3 Using FEAR and R

Once R and FEAR have been downloaded and installed, R’s graphical user interface (GUI) can be started by clicking on the desktop R icon; commands are typed at a prompt in the “RConsole” window contained within the RGui window. After installing FEAR, R’s online help facility can be used to find documentation on the commands implemented in FEAR, as
shown in Figure 1. Clicking on a command name displays a page giving details on use of the command, including arguments that must be passed, optional arguments and any defaults, etc., as well as a detailed description of what is returned by the command.

FEAR includes data that can be used to illustrate the library’s capabilities. First, the command `library(FEAR)` must be typed in the RConsole window to load the FEAR library, as shown in Figure 2. One might then type `help(ccr)` to learn that the Charnes et al. (1981) data contain 5 inputs used to produce 3 outputs among 70 schools. The following commands load these data and then organize the input vectors in a (5 \times 70) matrix and the output vectors in a (3 \times 70) matrix:

```r
data(ccr)
x=t(matrix(c(ccr$x1,ccr$x2,ccr$x3,ccr$x4,ccr$x5),
nrow=70,ncol=5))
y=t(matrix(c(ccr$y1,ccr$y2,ccr$y3),nrow=70,ncol=5))
```

The following commands reproduce the outlier analysis in Wilson (1993):

```r
tmp=ap(x,y,NDEL=12)
windows()
ap.plot(RATIO=tmp$ratio)
```

The resulting plot is shown in Figure 4.

Next, Shephard (1970) output-distance function estimates can be computed:

```r
d.out=dea(x,y,ORIENTATION=2)
```

R’s `summary` command

```r
summary(d.out)
```

gives summary statistics on the estimates. Alternatively, the `summary` command can be combined with the `ifelse` statement to obtain summary statistics on the distance function estimates that are different from unity:

```r
summary(ifelse(d.out==1,NA,d.out))
```

as shown in Figure 3. FEAR’s `show.dens` command can be used to produce plots of kernel density estimates for the distance function estimates contained in `d.in` and `d.out`; the necessary bandwidth parameters can be obtained with FEAR’s `eff.bw` command:

```r
h=eff.bw(d.out)
show.dens(dist=d.out,bw=h,
           XLAB="output distance function estimates")
```

The resulting plots are shown in Figure 5; here, they have been drawn on the user’s screen, but alternatively, postscript code for the plots can be written to a file for inclusion in a manuscript, etc. The `show.dens` command uses a reflection method to avoid inconsistency
problems for the kernel density estimator and the lower (upper) boundary 1 for the input (output) efficiency estimates.

The command `boot.sw98` implements the bootstrap algorithm described by Simar and Wilson (1998); this command can be used to estimate 95-percent confidence intervals for the input distance functions corresponding to each observation in the Charnes et al. (1981) data by typing

```r
result = boot.sw98(XOBS = x, YOBS = y, DHAT = d.out, ORIENTATION = 2)
```

The following command will produce TeX code for a table displaying results from the last command; the first column contains the observation number, the second column contains the input distance function estimates, while the third and fourth columns contain the lower and upper bounds of the estimated confidence intervals:

```r
tmp = paste(c(1:70), "n &", d.in, "n &",
result$conf.int[,1], "n &",
result$conf.int[,2], "\cr", sep = "")
```

4 Concluding Remarks

*FEAR* is a very flexible, extensible package unlike any currently available for estimation of productivity and efficiency. The cost of this flexibility is that the user must type commands at a command-line prompt; for some, this may be unsatisfactory. For others, however, *FEAR* will be a useful tool, allowing one to estimate quantities that have not been explicitly programmed into other packages, and that have perhaps not been anticipated by this or other authors.
References


Figure 1: List of commands in the FEAR package
Figure 2: The R graphical user interface
Figure 3: Output from R’s `summary` command
Figure 4: Plots produced by the `ap.plot` command
Figure 5: Kernel density plots produced by the `show.dens` command