Chapter 6
Statistical Analysis in DEA

6.1 Introduction

DEA is often classified as a non-statistical or deterministic approach that does not easily allow genuine hypothesis testing. Although DEA has not historically emphasized the use of traditional statistical tests, considerable progress has been made in this respect over the last 15 years. We will cover some important results in this chapter.

Initially, however, let us note that the background for DEA is operations research and management science. Management science is concerned with use of scientific, mostly mathematical, methods to solve real problems. This means that DEA studies have emphasized model-building as emphatically as they have model testing. That is, a DEA model developed for evaluation purposes is not to be evaluated solely based on its ability to explain and predict data in the best possible way. Basic properties of production economics like free disposability, economies of scale and convexity, the logic of the production structure from an engineering perspective, the relevance of the identified peers to industry representatives, etc., serves to validate the model just as statistical tests serve to validate a statistical model developed to replicate some underlying data generation process as closely as possible. Therefore, we maintain that interesting insights can arise from the use of DEA models without in the heavy use of statistical testing.

There are, of course, particular situations for which we are interested in performing hypothesis tests and constructing confidence intervals based on DEA models. Thus for example, we might wish to

- Test model-building assumptions like the returns to scale assumption
- Test for relevant and irrelevant inputs and outputs
- Test for differences between different groups of firms in terms of efficiency
- Test allocative and scale efficiency of a group of firms
- Test whether efficiency depends on external factors

In general, there are three ways to conduct such tests.

P. Bogetoft and L. Otto, *Benchmarking with DEA, SFA, and R*, International Series in Operations Research & Management Science 157, DOI 10.1007/978-1-4419-7961-2_6,

One is to rely on general *non-parametric tests*, i.e. tests used when the underlying distribution is unknown. We discuss some of these, including Kolmogorov–Smirnov tests and Kruskal–Wallis tests.

Another way is to rely on *parametric tests*, making assumptions regarding the underlying distribution of inefficiency and noise in the data. We will cover a series of such tests based on *asymptotic statistical theory*. Relying on asymptotic theory means that the theoretical properties are only established for large samples. However, simulation studies based on samples of moderate size, those including 50 firms and above, do suggest that they can be used quite generally.

The third approach, and one that has become popular with the development of effective computer programs, is the use of the *bootstrap*. The bootstrap is a computer-based method that can answer many statistical questions. The approach replicates sampling uncertainty by creating repeated samples of the original sample. We will spend most of this chapter covering bootstrap-based inference in DEA models.

In the appendix, we discuss the use of statistical methods in second-stage analyses, i.e. analyses performed after the development of a benchmarking model, to validate the model and to explore the possible causes of the variations in efficiencies. A common approach in such studies is tobit regression, and such analyses are not only relevant for DEA based benchmarking.

6.2 Asymptotic tests

In this section, we will assume that firm's efficiency is the realization of a random variable and that this is the sole reason why observed performance deviates from the underlying production possibility frontier; i.e. all deviations are efficiency-related, and there is no noise in the data.

Specifically, let us consider a DEA setting and assume that the true Farrell output efficiency ϕ, i.e.

$$\phi = \max\{ F \mid (x, Fy) \in T \}$$

is a random variable with values in $[1, \infty[$ and a density function g. Also, we assume that there is a non-zero likelihood of nearly efficient performance; i.e. $\int_1^{1+\delta} g(\phi)\, d\phi > 0$ for all $\delta > 0$.

In the following, it is importing to note that we distinguish between the true but unknown and unobservable technology T and a DEA estimate T^* of T. Now, it is clear that the estimated efficiency F in any finite sample of firms

$$F = \max\{ F \mid (x, Fy) \in T_\gamma^* \}$$

where

$$T_\gamma^* = \Big\{ (x,y) \in \mathbb{R}_+^m \times \mathbb{R}_+^n \mid x \geq \sum_{k=1}^{K} \lambda^k x^k,\ y \leq \sum_{k=1}^{K} \lambda^k y^k,\ (\lambda^1,\ldots,\lambda^K) \in \Lambda^K(\gamma) \Big\}$$

is biased downwards; i.e. it is always weakly smaller than true (in)efficiency ϕ, $F \leq \phi$. Recall here that $\Lambda^K(\gamma)$ is the restrictions on λ that depends on the returns to scale assumptions, i.e. fdh, vrs, or crs, as discussed in Sect. 4.4. The reason is that we have only observed a subset of practices, not necessarily the best practices, and the estimate of T^* of T is therefore an inner approximation, $T^* \subseteq T$, meaning that F measured against T^* is less than ϕ measured against T. Thus, estimated efficiency values never make a firm look less efficient than it really is, only more so. DEA-based estimates in this setting are cautious and puts the firms in a positive light.

However, asymptotically (with the number of firms going to infinity), this bias reduces to zero; that is, *the DEA estimators are consistent*. This holds as soon as the probability of observing nearly efficient firms is strictly positive, as we assumed above. Consistency is a nice statistical property because it means that for large samples, our evaluation is correct.

Additionally, one can show that if the density function g is monotonously declining (i.e. $f' > f \Rightarrow g(f') \leq g(f)$), then the DEA estimator F is the *maximum likelihood estimator* for ϕ.

The consistency results indicate that for large samples of firms, the distribution of F is similar to the distribution of ϕ. Therefore, in a large sample, the distribution of a test statistic $t(F)$ will be similar to the distribution of $t(\phi)$, and the distribution of $t(\phi)$ can be found from the density g of ϕ. This technique can be used to construct a series of tests as we do in the subsections that follow.

6.2.1 Test for group differences

If the set of K firms is divided into two groups with K_1 and K_2 firms, $K = K_1 + K_2$, we may be interested in testing whether there are significant differences between the efficiencies of the two groups—note that we use K, K_1 and K_2 as both the number of firms and the set of firms. This procedure may be relevant if we aim to test whether one special ownership structure is more efficient than another, whether one particular treatment is more effective than another, whether a specific region offers more favorable conditions for firms than another, etc.

Letting the density of the distributions of the efficiencies in the different groups be g_1 and g_2, respectively, we seek to test

$$H_0 : g_1 = g_2 \text{ against } H_A : g_1 \neq g_2.$$

As mentioned before, the distributions of $t(F)$ and $t(\phi)$ are asymptotically the same. If $t(\phi)$ is exponentially distributed, a chi-square distribution with 2 degrees of freedom, then $\sum_{k=1}^{K} t(F^k)$ is asymptotically χ^2–distributed with $2K$ degrees of freedom.

Under the null hypothesis, the two groups have the same distribution of efficiency, and the ratio

$$T_{EX} = \frac{\sum_{k \in K_1} t(F^k)/K_1}{\sum_{k \in K_2} t(F^k)/K_2}$$

is the ratio of two asymptotically χ^2–distributions and is therefore asymptotically distributed as a Fisher distribution with $2K_1$ and $2K_2$ degrees of freedom, $T_{EX} \overset{a}{\sim} F(2K_1, 2K_2)$. Note that T_{EX} might be greater or less than 1 such that the test is two-sided.

If we assume that true efficiency is $\phi = 1 + \epsilon$ where ϵ is exponential distributed, then we should simply use $t(F) = F - 1$ such that

$$T_{EX} = \frac{\sum_{k \in K_1} (F^k - 1)/K_1}{\sum_{k \in K_2} (F^k - 1)/K_2}$$

and reject the hypothesis if T_{EX} is greater than the 95% quantile in the distribution $F(2K_1, 2K_2)$.

Likewise, if $t(\phi)$ has a half-normal distribution, then $t(\phi)^2$ is χ^2 distributed, and therefore, $\sum_{k=1}^{K} t(F_k)_2$ is asymptotically χ^2–distributed with K degrees of freedom. The test statistic

$$T_{HN} = \frac{\sum_{k \in K_1} t(F^k)^2/K_1}{\sum_{k \in K_2} t(F^k)^2/K_2}$$

is therefore distributed as $F(K_1, K_2)$. This will be the case if, for example, $\phi - 1$ has a half-normal distribution, and in this case, we should again use $t(F) = F - 1$.

Lastly, if we have no a priori assumptions about the distribution of ϕ_1 and ϕ_2, we may use the non-parametric Kolmogorov– Smirnov test statistic

$$T_{KS} = \max_{k=1,\ldots,K} \left\{ |G_1(F^k) - G_2(F^k)| \right\}$$

where G_1 and G_2 are the empirical cumulative distributions in the two subsets such that T_{KS} is the largest vertical distance between the cumulative distributions. Large values of T_{KS} as evaluated via the Kolmogorov-Smirnov test as an indication that H_0 is false. Note that this test depends on the rank (i.e. the order) of F^k only and not on the individual values of F^k.

Another non-parametric test based on ranks is the Kruskal–Wallis test used to test groups of data. We will not show how to run this test but would like to note that the test only depends on the rank of the observations. This test is helpful because it can be used to test the hypothesis that several groups have the same distribution.

Numerical example in R: Milk producers

We want to test data from a group of milk producers to determine if efficiency depends on the breed of cow. The inputs are cost categories, and the output is milk. Group 1 is comprised of farmers without jersey cows, whereas group 2 is comprised of farmers with jersey cows.

Implementing the T_{EX} and T_{HN} tests in R is easy; these tests are simply a matter of summing the efficiencies with 1 subtracted. The commands `qf` and `pf` calculate the quantile (.95 for 95% or 5% tail probability) and the probability in the Fisher distribution. The calculated output efficiencies are split into two groups `F1` and `F2` based on the value of the two-level factor `race`, and the test evaluates whether the efficiency of the two groups is identical.

The Kolmogorov– Smirnov and the Kruskal–Wallis tests are more complicated, but R already contains special methods for those tests; therefore, it is easy to use them in R.

The code and output for the tests are shown here:

```
> library(Benchmarking)
> cattle = read.csv("projekt.csv")
> attach(cattle)
> kgMilk <- milkPerCow * cows
> x <- cbind(unitCost, capCost, fixedCost)
> y <- matrix(kgMilk)
> FF <- eff(dea(x,y,ORIENTATION="out"))
> TEX <- sum(F1-1)/length(F1) / (sum(F2-1)/length(F2))
> TEX
[1] 1.989044
> qf(.025, 2*length(F1), 2*length(F2))
[1] 0.6369572
> qf(.975, 2*length(F1), 2*length(F2))
[1] 1.682756
> pf(TEX, 2*length(F1), 2*length(F2))
[1] 0.9947547
> THN <- sum((F1-1)^2)/length(F1) / (sum((F2-1)^2)/length(F2))
> THN
[1] 2.000593
> qf(.025, length(F1), length(F2))
[1] 0.5357977
> qf(.975, length(F1), length(F2))
[1] 2.148472
> pf(THN, length(F1), length(F2))
[1] 0.9628421
> # Kolmogorov-Smirnov test
> ks.test(F1, F2)

        Two-sample Kolmogorov-Smirnov test
data:  F1 and F2
D = 0.4893, p-value = 0.0006954
alternative hypothesis: two-sided

> # Kruskal--Wallis, 2 groups
```

```
> kruskal.test(FF, race=="jersey")

        Kruskal-Wallis rank sum test
data:  FF and race == "jersey"
Kruskal-Wallis chi-squared = 11.6309, df = 1, p-value = 0.0006487
```

The value of the T_{EX} is calculated to be 1.989044, and as the 97.5% upper critical value (the size of the test is 5%) in the F distribution with 80 (the number of firms in group 1) and 21 (the number of firms in group 2; breed "jersey") is 1.68, we reject the hypothesis that the distribution of efficiency in the groups is identical. The T_{HN}, on the other hand, is 2.00, and the upper critical value is 2.148. Thus, we do not reject the hypothesis that they are identical; rather, the groups could be identical. The results of both the Kolmogorow–Smirnov test and the Kruskal–Wallis test lead us to reject the null hypothesis. Based on the boxplot and densities in Fig. 6.1, it

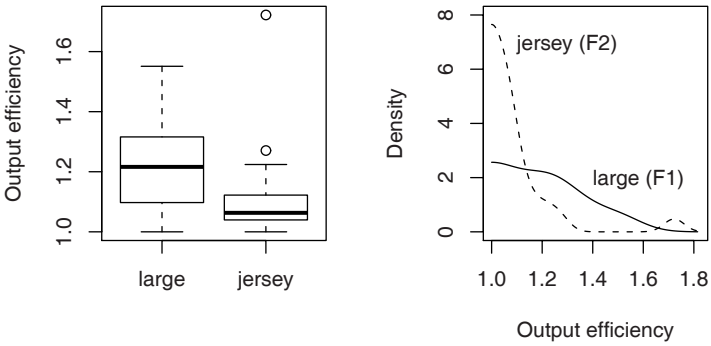

Fig. 6.1 Boxplot and densities for output efficiency of the two subgroups

does look as if group 2 (the "jersey" breed) has steeper density and mass closer to 1 than group 1. Most of our tests also show that the difference is significant, and what we see in the figure is therefore most likely not a matter of chance. One result that emerges is that the for group F2 ("jersey"), the average output efficiency is lower than that for group F1 ("large"); i.e. F2 is more efficient than F1 on average. Note that there is an outlier in group F2, indicated both at the top of the boxplot as a circle and in the density illustration as a blip to the far right.

6.2.2 Test of model assumptions

In model development and model validation, we may want to test if an alternative model specification better represents firm performances. We might, for example, be interested in testing whether we can assume variable return to scale or whether some outputs can be eliminated from the model specification.

Here we will distinguish not between two groups of observations, but rather distinguish between two sets of model assumptions, or what accounts to the same, distinguish between two technology sets. In Sect. 4.3, we argued that the estimated technology set should be the smallest set containing the data and fulfilling certain assumptions (the minimal extrapolation principle). The question we ask here is therefore whether an estimated technology set can be made even smaller by adding further restrictions and still be in agreement with data. Let the technology set be T_1, and let the smaller technology set be T_2. For example, we could have the same assumptions regarding the technology sets but assume CRS in T_1 and VRS in T_2, with the additional restriction $\sum_{k=1}^{K} \lambda^k = 1$. Likewise, the technology set T_1 could include n outputs, and the technology set T_2 could include $n' > n$ outputs; the greater number of outputs would result in the existence of more restrictions and therefore yield a smaller technology set.

In both examples, and in general, a smaller technology set (i.e. one with more restrictions) results in a better (or unchanged) efficiency level; for input efficiency, we obtain $E_1 \leq E_2 \leq 1$, and for output efficiency, we obtain $F_1 \geq F_2 \geq 1$ where the subscript of the efficiencies E and F is a product of the corresponding technology set T_1 and the smaller set T_2.

In statistical language, technology set T_1 represents the null hypothesis and the smaller technology set T_2 the alternative. We test technology hypothesis T_1 against alternative T_2.

If the efficiencies calculated under T_1 are very different from the efficiencies calculated under T_2, the two technologies are not at all similar, and we should reject the null-hypothesis technology T_1 and opt for the alternative technology T_2; the extra restrictions in T_2 are of real importance. If the efficiencies are more or less the same, then the extra restrictions are of no importance, and we opt for the null-hypothesis technology T_1. Therefore, we can test the technology assumptions by testing whether efficiency is the same under the two technologies.

Now, let the distribution of the efficiency scores for K firms under the two technology assumptions T_1 and T_2 be g_1 and g_2, respectively. We will then test the hypothesis

$$H_0 : g_1 = g_2 \text{ against } H_A : g_1 \neq g_2$$

using the same ideas as above, except that we now sum the figures for all firms in both the numerator and the denominator. If we accept the hypothesis H_0, we use technology T_1, whereas if we reject the hypothesis, we use technology M_2. More specifically, if $t(\phi_1)$ and $t(\phi_2)$ are exponentially distributed for some monotone transformation $t(\cdot)$, then just as before, the test statistic

$$T_{EX} = \frac{\sum_{k=1}^{K} t(F_1^k)}{\sum_{k=1}^{K} t(F_2^k)},$$

where F_1^k and F_2^k are the output efficiency of firm k based on technologies T_1 and T_2, respectively, will follow a F–distribution under H_0 with $2K$ and $2K$ degrees of freedom, $F(2K, 2K)$.

The test is one-sided as $T_{EX} \geq 1$, and therefore, the critical value for a test of size 5% is the 95% quantile in the F–distribution with $2K$ and $2K$ degrees of freedom, $F(2K, 2K)$; i.e. for large values of T_{EX}, we reject the null hypothesis H_0 that model M^1 is true.

Likewise, if $t(\phi_1)$ and $t(\phi_2)$ have a half-normal distribution for some monotone transformation $t(\cdot)$, then we can use the test statistic

$$T_{HN} = \frac{\sum_{k=1}^{K} t(F_1^k)^2}{\sum_{k=1}^{K} t(F_2^k)^2}$$

with large values in a $F(K, K)$ distribution as critical values for the test of H_0.

Lastly, if we have no a priori assumptions about the distribution of ϕ_1 and ϕ_2, we can use the non-parametric Kolmogorov– Smirnov test statistic

$$T_{KS} = \max_{k=1,\ldots,K} \{ |G_1(F^k) - G_2(F^k)| \}$$

where G_1 and G_2 are the empirical cumulative distributions in the two models such that T_{KS} is the largest vertical distance between the cumulative distributions. Large values for T_{KS} indicate that the distributions differ and therefore that H_0 is false; the null hypothesis H_0 is rejected.

Numerical example in R: Milk producers

Implementing the tests for model assumptions is just as easy as implementing the tests of group differences. However, we present an example anyway to introduce yet another example of a hypothesis.

So far, we have used two examples to test our model assumptions. Here, we use a third example to test whether to include fewer inputs. The null hypothesis is technology T_1 with m inputs, whereas the alternative is technology T_2 with $m' > m$ inputs. Again, the alternative includes more restrictions and specifically more input restrictions in the LP formulation. The test statistics are as previously described.

We use the same data set that we used to test group differences. We want to test whether we really need capacity costs when we already include the number of cows and whether veterinary expenses are important on their own even though they are part of unit costs. Thus, the alternative technology set T_2 includes among its inputs the number of cows and veterinary expenses, whereas technology T_2, the null hypothesis, excludes these two inputs.

The input matrix `x1` in the example below excludes the variables in question, whereas the input matrix `x2` includes them. The following code reads data, calculates efficiency and makes graphs as shown in Fig. 6.2. The graphs are slightly different from the ones we presented in the test for group differences.

```
library(Benchmarking)
cattle = read.csv("projekt.csv")
kgMilk <- with(cattle, milkPerCow * cows )
```

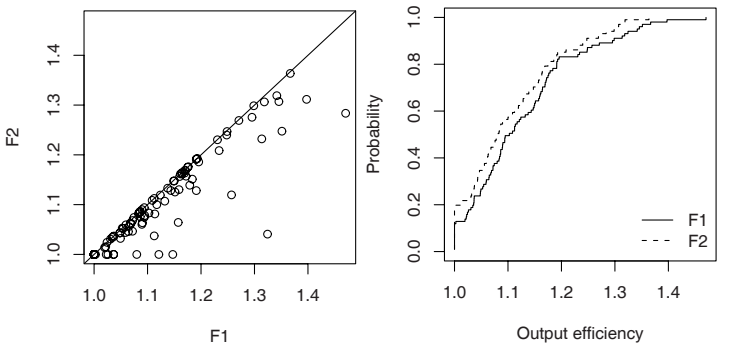

Fig. 6.2 Efficiency when capacity cost and veterinary costs are excluded (F_1) and included (F_2) in the inputs for milk production: comparing efficiencies and the empirical distribution of efficiencies.

```
x1 <- with(cattle, cbind(unitCost,          fixedCost,     cows))
x2 <- with(cattle, cbind(unitCost, capCost, fixedCost, vet, cows))
y <- matrix(kgMilk)
F1 <- eff(dea(x1,y,ORIENTATION="out"))
F2 <- eff(dea(x2,y,ORIENTATION="out"))

plot(F1,F2, xlim=range(F1,F2), ylim=range(F1,F2))
abline(0,1)

K <- length(F1)
plot(sort(F1), (1:K)/K, type="s", ylim=c(0,1),
      ylab="Probability", xlab="Output␣efficiency")
lines(sort(F2), (1:K)/K, type="s", lty="dashed")
legend("bottomright",c("F1","F2"),
      lty=c("solid","dashed"),bty="n")
```

The box plot shows that the two technologies T_1 and T_2 are only slightly different in terms of efficiency; the spread is slightly greater for `F1` that for `F2`. The same pattern is seen in the top right plot, where some of the efficiencies are identical (i.e. on the diagonal line) and some for `F1` are larger than those for `F2` (below the diagonal line). This is no surprise given that the number of inputs is smaller in `F1`; firms will have unchanged or greater output efficiency, as discussed in Sect. 4.6 on page 93. The bottom figure shows the empirical distribution. The distribution of `F2` is above that of `F1`; for every level of efficiency, the proportion of firms at that level or lower is larger for technology T_2 than for technology T_1.

The problem is whether the difference that we see is statistically significant. This is where the test statistics come into play. Based on the above calculations for the two efficiencies, the test statistics are calculated below.

```
> TEX <- sum(F1-1)/length(F1) / (sum(F2-1)/length(F2))
> TEX
[1] 1.211835
> qf(.95, 2*length(F1), 2*length(F2))
[1] 1.261131
> pf(TEX, 2*length(F1), 2*length(F2))
```

```
[1] 0.9135035
>
> THN <- sum((F1-1)^2)/length(F1) / (sum((F2-1)^2)/length(F2))
> THN
[1] 1.381849
> qf(.95, length(F1), length(F2))
[1] 1.389417
> pf(THN, length(F1), length(F2))
[1] 0.9471316
>
> # Kolmogorov-Smirnov test
> ks.test(F1, F2, alternative = "greater")

        Two-sample Kolmogorov-Smirnov test

data:  F1 and F2
D^+ = 0, p-value = 1
alternative hypothesis: the CDF of x lies above that of y

Warning message:
In ks.test(F1, F2, alternative = "greater") :
  cannot compute correct p-values with ties
> # Kruskal--Wallis
> kruskal.test(list(F1, F2))

        Kruskal-Wallis rank sum test

data:  list(F1, F2)
Kruskal-Wallis chi-squared = 2.519, df = 1, p-value = 0.1125
```

The T_{EX} and T_{HN} are estimated to be 1.21 and 1.38, and both fall below the critical value, the 95%–quantile. The results of the Kolmogorof–Smirnof test and the Kruskal–Wallis test both support the same conclusion. Note that the probabilities for these tests are tail probabilities. Therefore, we do not reject the null hypothesis that we need to include capacity cost and veterinary costs among the inputs, and for all uses of the technology, we should be using T^1 with the fewest input variables.

Practical application: DSO regulation

In the regulation of German electricity distribution operators, DSOs, a series of tests were undertaken to ensure that models did not unintentionally favor or disadvantage specific types of companies. We will discuss regulation in greater detail in Chap. 10. The tests for the DSO technologies was conducted as second-stage tests of the best of four scores that the regulation prescribed using non-parametric Kruskal–Wallis tests, cf. also Chap. 10. However, we could also have used tests like those above to directly evaluate the individual DEA models and test for the impact of such factors as 1) whether the DSO is located in what was formerly West or East Germany or 2) whether the DSO is also involved in gas distribution, water distribution etc.

The same regulations also stipulate that no single DSO can have too large an impact on average efficiency in the DEA models. This requirement was tested using the test statistic

$$\frac{\sum_{h\in K\setminus k}\left(E(h,K\setminus k)-1\right)^2}{\sum_{h\in K\setminus k}\left(E(h,K)-1\right)^2}.$$

Here, K is both the set and the number of DSOs in the data set, and k is a potential outlier. Also, $E(h,K)$ is the efficiency of h when all DSOs are used to estimate the technology, and $E(h,K\setminus k)$ is the efficiency when DSO k does not enter into the estimation. The test therefore compares the average efficiency of the other operators when DSO k cannot affect the technology with the average efficiency of the other DSOs when DSO k is part of the evaluation process. Because $E(h,K\setminus k)\geq E(h,K)$, this ratio is always less than or equal to 1, and the smaller the ratio, the larger the impact of k; i.e. small values will be an indication that k is an outlier. We see that this line of thought resembles the model specification test problems above, which suggests that we can evaluate the test statistic in a $F(K-1,K-1)$ distribution.

6.3 The bootstrap method

Bootstrap is a general computer-based statistical method for calculating the accuracy of statistical estimates. Generally, "pulling oneself up by one's bootstraps" means to succeed based on one's own efforts despite very difficult circumstances and without help from anyone. The statistical bootstrap method has some of this flavor and recalls the story of Baron von Munchausen, who pulled himself and his horse out of a swamp by pulling on his own hair while holding on to the horse with his legs. In the following pages, we first give a short introduction to bootstrap as a general method and then explore the details of bootstrap DEA models.

The basic idea of bootstrap is to sample observations with replacements from one's data set and thereby create a new "random" data set of the same size as the original. Using this dataset, one can calculate the necessary statistics, called replicates. This process is repeated to create a *sample of replicates*. Based on this sample, we can draw conclusions about the distribution of the statistics in which we are interested.

Let us consider a very simple example, a sample of n observations $x_1, x_2, \ldots, x_n$. Imagine that we have observed 7 numbers 94, 197, 16, 38, 99, 141, and 23. The mean is $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i = 86.86$, and the (unbiased) standard error is $s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\bar{x})^2} = 66.77$. The estimate of the standard error of the mean is $\frac{s}{\sqrt{n}} = 25.24$. The standard error is very easy to estimate when we simply wish to determine the variance of the mean because we can use an explicit formula. Unfortunately, we do not always have an explicit formula for the standard error or for variance.

Table 6.1 The bootstrap algorithm for estimating standard errors

1. Select B independent bootstrap samples $x^1, x^2, \ldots, x^B$, i.e. a sample drawn with replacement from our data set.
2. Calculate the estimate for each bootstrap sample:

$$t(x^b) \quad (b = 1, \ldots, B) .$$

3. Estimate the standard error using the sample standard error of the B replications

$$\hat{s}_B = \sqrt{\frac{1}{B-1} \sum_{b=1}^{B} \left(t(x^b) - \bar{t}\right)^2}$$

where $\bar{t} = \frac{1}{B} \sum_{b=1}^{B} t(x^b)$.

If instead of investigating the mean we wish to find the median and the variance of the median, we must undertake a much more complicated process because the formula for calculating the variance of the median is not easily determined. This is where the bootstrap method becomes key.

A bootstrap sample in this case is a random sample obtained by sampling 7 (the number of elements in the sample) elements or data points *with* replacements from our original sample. Hence, the bootstrap sample could be $x^b = (x_6, x_1, x_4, x_1, x_3, x_3, x_5)$, i.e. 141, 94, 38, 94, 16, 16, and 99. Based on this bootstrap sample, we estimate the statistic $t(x^b)$ we are interested in: here, the median. Now, instead of trying to calculate the standard deviation of the estimated median, we make B bootstrap replications. For each bootstrap replication b, we calculate $t(x^b)$, the median. As the bootstrap estimate of the standard error of $t(x)$ with B replications, we use $\hat{s}_B = \sqrt{\frac{1}{B-1} \sum_{b=1}^{B} (t(x^b) - \bar{t}\,)^2}$ where $\bar{t} = \frac{1}{B} \sum_{b=1}^{B} t(x^b)$ is the mean over the replications of the statistic we are interested in.

The idea of the bootstrap method is that if the empirical distribution of x^b corresponds more or less to the true distribution of x, then the empirical distribution of $t(x^b)$ will correspond more or less to the true distribution of $t(x)$. This means that we can use the empirical distribution of $t(x^b)$ as the true but unknown distribution of $t(x)$. Thus, when we are interested in the variance of the median, $t(x)$, which is difficult or impossible to determine, we can simply use the empirical variance of the median of the bootstrap, $t(x^b)$, which is much easier to obtain.

The bootstrap method can be described as the algorithm in Table 6.1. The limit of $\hat{s}_B$ as B goes to infinity is the ideal bootstrap estimate.

Luckily, we do not have to program the algorithm in Table 6.1 ourselves; it is part of the package boot in R, and now we show how to use it in the small numerical example we have just seen.

Table 6.2 Bootstraping the variance of the median in a sample with 7 numbers

```
library(boot)
treat <- c(94, 197, 16, 38, 99, 141, 23)
func <- function(d,i) { median(d[i]) }
B <- 200
boo <- boot(treat, func, B)
sqrt(var(boo$t))
mean(boo$t)
hist(boo$t,main=NULL)
```

Numerical example in R

Bootstrap is easy in R because the package boot contains the function `boot`, which organizes the resampling and calculation of a statistic (function) we provide; this is just an implementation of the algorithm in Table 6.1. In our example in which we investigate the variance of the median, we use the R script in Table 6.2. The first line is the command to load the library `boot` that contains the commands and methods for bootstrap in R. The second line defines our data set, our original sample, as the variable `treat`. To use the R function `boot`, we must define a function that calculates the statistic of interest. In our case the function must calculate the median, and it must be defined with two arguments, the first the original data and the second a vector of indices, frequencies or weights that define the bootstrap sample. Here, the function is called `func`, and the two arguments are `d` for data and `i` for the indices, such that `d[i]` is a bootstrap sample and the return of the function is the median of the bootstrap sample `d[i]`. Next, we define variable `B` as the number of bootstrap replicates; in this case, we use 200 replicates. To actually generate the bootstrap replicates, we use the R function `boot`. This function takes 3 arguments: the original sample, the function we have defined to calculate the statistics of interest, and the number of replicates (bootstrap iterations) we seek, here the defined by the variable `B`.

The function `boot` can take many more arguments than we use here; see the manual, `>?boot`, for others.

The output from the bootstrap function is put into the variable `boo`, a boot object. Hereafter, we can gain access to the replicates of the 200 calculated statistics (medians in our case) in the component `t` in the object/variable `boo`, i.e. the variable `boo$t`. Now we can easily calculate the variance of the median as `boo$t`, and if we want to determine the standard error, we can simply take the square root. The resulting standard error of the median of our sample `treat` is

```
> sqrt(var(boo$t))
  38.00217
```

showing that the standard error of the median of our 7 numbers is 38. A histogram of the bootstrap replicas is shown in Fig. 6.3. The figure indicates that the most common median is between 90 and 99, and based on the data set, we can see that

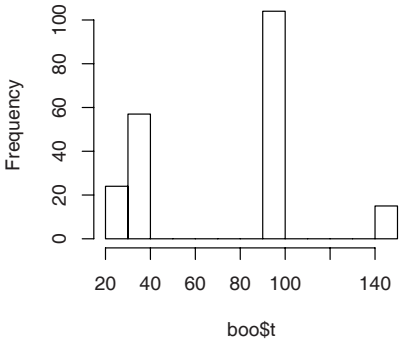

Fig. 6.3 Histogram of bootstrap replicas for the median of the 7 numbers

Table 6.3 Bootstrap the median of numbers with different replications in R

```
library(boot)
func <- function(d,i) { median(d[i]) }
treat <- c(94, 197, 16, 38, 99, 141, 23)
Ber <- c(10,50,100,250,500,1000,5000,10000,1000000)
res <- NULL
for(B in Ber) {
   boo <- boot(treat, func, B)
   res <- c(res, format(sqrt(var(boo$t)), digits=3))
}
Ber     # print Ber
res     # print res, the results
rbind(Ber, res)
```

it must be 94 or 99—the median in the original data set `treat` is 94. The second most common median is just below 50 and the actual number is 38.

If we make the same calculations again, we may obtain a figure for variance that is somewhat different because we obtain another series of replications. However, if the number of replications is very large, then each time we repeat the bootstrap series of replications, the variance will be almost the same. The question is then how many replications we should conduct to develop a stable estimate of the variance?

The calculated standard errors of the median from several bootstraps when the number of bootstrap replicates B is ranging from 10 to 1 000 000 is calculated using the R program in Table 6.3. The results achieved by running this code are shown in Table 6.4; we have run the program several times and show the different standard errors in the different rows. When the number of bootstrap replications is larger than 1000, there is hardly any difference between the levels of variance for the different runs. Thus, the desired level of precision of the estimated variance determines the number of replications.

For a bootstrap sample of size 10, one of the standard errors differs substantially from the other bootstrap samples, as can be seen in Table 6.4. Based on considerations like this one, it is suggested in the literature that bootstrap samples, B, ranging

Table 6.4 Bootstrap estimates of standard error of the median

B:	10	50	100	250	500	1 000	5 000	10 000	1 000 000
Run 1: Std.err:	32.7	38.5	37.5	38.9	36.7	38.3	37.9	38.1	37.8
Run 2: Std.err:	38.7	44.5	40.4	39.7	38.4	37.9	37.8	37.4	37.8
Run 3: Std.err:	2.58	43.3	33.2	37.0	36.8	38.2	37.7	37.8	37.8
Run 4: Std.err:	35.9	37.8	41	37.3	38.6	38.7	37.6	38.0	37.9

from 50 to 200 usually make the bootstrap a good standard error estimator. As we shall see later, however, these suggested numbers of bootstrap replications are to small for DEA models.

If we want to find the variance of another function or statistic instead of the median of our sample, we can simply redefine the function `func` to calculate the new statistic, which may include very complicated calculations (as is the cace, for instance, with DEA efficiency). If we want to consider another sample, we can just change the contents of `treat`.

6.3.1 Confidence interval

Using the bootstrap sample, we can also directly determine the confidence intervals for the statistic. This approach yields more precise results than do efforts to construct the confidence intervals based on the estimated standard deviation because the latter technique rests on the assumption that the distribution in question is symmetric and can be reasonably approximated using a normal distribution. This is not the case for the aforementioned example intended to determine the median of the 7 numbers.

To find a 50% confidence interval for the sample, we can use the command quantile in R, as shown in Table 6.5. The results are shown in Fig. 6.4. In the figure, the

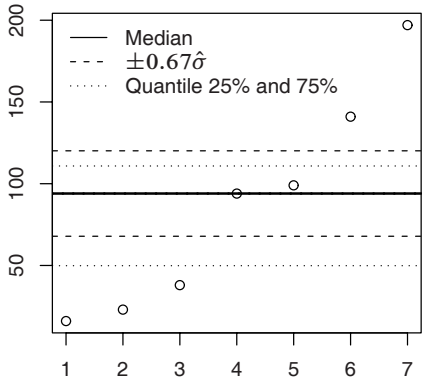

Fig. 6.4 Confidence interval for median of 7 numbers based on 200 replicats

Table 6.5 Calculating a 50% confidence interval for the median of 7 numbers

```
library(boot)
treat <- c(94, 197, 16, 38, 99, 141, 23)
func <- function(d,i) { median(d[i]) }
B <- 200
boo <- boot(treat, func, B)
sqrt(var(boo$t))
mean(boo$t)

quant <- .50 # 50% confidence interval
ci <- boot.ci(boo, conf=quant)
m <- mean(boo$t)
b <- m - median(treat)    # bias
mu <- m -b                # bias corrected median
sd <- sqrt(var(boo$t))    # std.error
quantile(boo$t,c((1-quant)/2, 1-(1-quant)/2) ) -b
```

7 numbers are shown in sorted order, and the median is marked with the solid line through the point at 94. The 50% confidence interval based on a normal approximation is shown as a dashed line, and of course, it is symmetric around the median. The dotted line is based on the command `quantile`, and this confidence interval is not symmetric around the median. The upper line is a little lower that the normal line, and the lower line is much lower than the normal line. This corresponds to the histogram in Fig. 6.3, where the distribution does not seem to be symmetric. Based on the actual numbers in the sample `treat`, the 50% interval for the median 94 is from 38 to 99. This corresponds to the histogram in which one can see that the median in half of the replicas is between 35 and 100.

6.4 Bootstrapping in DEA

We will now discuss how to estimate the variance of efficiency measures for a sample of firms using the bootstrap method. Let the observations be $(x^1, y^1), \ldots, (x^K, y^K)$ and the corresponding Farrell input efficiency measures be $E^1, \ldots, E^K$, i.e. $E^k = \min\{\, \theta \in \mathbb{R}_+ \mid (\theta x^k, y^k) \in T \,\}$. None of what follows would change if we considered Farrell ouput efficiency instead.

It does not make sense to compute variance as $\frac{1}{n-1}\sum_{k=1}^{K}(E_k - \bar{E})^2$ because then we would be assuming that all the firms have efficiencies based on a distribution with the same mean and therefore that all differences in efficiency are purely random and not systematic; firms with high efficiency would then be highly efficient by chance and because they are good at what they do.

Instead, we use our observations as a sample $\mathfrak{X} = \{(x^1, y^1), \ldots, (x^K, y^K)\}$ of inputs and outputs from K firms that we can use to estimate the technology set T via DEA assuming variable returns to scale (vrs)

$$\widehat{T} = \{\,(x, y) \mid x \geq \sum_{k=1}^{K} \lambda^k x^k, y \leq \sum_{k=1}^{K} \lambda^k y^k, \lambda^k \geq 0, \sum_{k=1}^{K} \lambda^k = 1\,\}.$$

The DEA estimated efficiency scores are then

$$\widehat{E}^k = \min\{\,\theta \in \mathbb{R} \mid (\theta x^k, y^k) \in \widehat{T}\,\} \qquad (k = 1, \ldots, n)$$

where we have used the estimated technology set $\widehat{T}$ for the technology set T.

We use this procedure to consider the sample $\mathcal{X} = \{(x^1, y^1), \ldots, (x^K, y^K)\}$ as a realization of identically and independently distributed random variables (X, Y) with a probability distribution $\mathcal{P}$ with support in T; i.e. we assume that there is no observational uncertainty in the sense that $(x^k, y^k) \in T$ with probability 1. In Chap. 7, we introduce a parametric method that allows for this form of observational uncertainty.

The distribution of $\widehat{E}^k$ and $\widehat{T}$ depends on the distribution of the sample of observations $\mathcal{X}$. However, this relationship is complex; the sample $\mathcal{X}$ is generated by the probability distribution $\mathcal{P}$, of which we have no direct knowledge. To derive a reasonable estimate $\mathcal{P}^*$ of $\mathcal{P}$, we can use the bootstrap, i.e. a sample with replacements from the original set of observations. Using this bootstrap estimate $\mathcal{P}^*$ of $\mathcal{P}$, we can generate a sample $\mathcal{X}^*$ from the distribution $\mathcal{P}^*$, then calculate a DEA estimate T^* for the technology and estimate efficiency as $E^{k*} = \min\{\,\theta \in \mathbb{R} \mid (\theta x^k, y^k) \in T^*\,\}$. When we repeat this sample generation process many times, we obtain many estimates of E^{k*} and can then calculate the empirical variance of E^k $(k = 1, \ldots, n)$.

6.4.1 Naive bootstrap

There are two ways to perform an ordinary bootstrap for the DEA model. Unfortunately as we will see, neither of them is satisfactory, and we will therefore present a better alternative.

The two simple but unsatisfactory methods are as follows:

1. Bootstrap the set directly $\{E^1, \ldots, E^K\}$ as we did in Sect. 6.3 on the variable `treat`. In using this method, we assume that all the E's are independent and identically distributed with a probability distribution $\mathcal{P}_E$. This implies that any differences in efficiency are purely random because they all come from the same distribution $\mathcal{P}_E$. On that basis, firm inefficiency appears to be related neither to x^k nor to y^k. This outcome is not satisfactory.
2. We bootstrap the set $\mathcal{X} = \{(x^1, y^1), \ldots, (x^K, y^K)\}$, and for each bootstrap sample, b, we estimate the technology T^b and the efficiency E^{kb} for firm k. When we make B bootstrap samples, B replicas, we can calculate the mean and variance of the efficiency of firm k using $\bar{E}^{k*} = \frac{1}{B}\sum_{b=1}^{B} E^{kb}$ and $\frac{1}{B}\sum_{b=1}^{B}(E^{kb} - \bar{E}^{k*})^2$.

One problem is that for some firm k, (x^k, y^k) may not be in a bootstrap sample, a replica b, and (x^k, y^k) may not be in the technology set generated by the bootstrap sample, $(x^k, y^k) \notin T^{*b}$. This implies that we have a firm outside the technology set, but one of our assumptions was that all observations are inside the technology set with probability 1. If we calculate the efficiency anyway, we find in this case that $E^{kb} > 1$.

This could easily happen for firms where $E^k = 1$ as a bootstrapped technology set T^* will always be a subset of the technology set $\widehat{T}$ estimated on all observations, $T^* \subset \widehat{T}$, and therefore $E^{k*} \geq E^k$. Essentially, we could in many bootstrap samples find firms where $E^{kb} > 1$.

We could disregard the requirement that all observations be inside the technology set and just use $E^{kb} = 1$ if we obtained $E^{kb} > 1$. One problem with this technique is that the probability of E near 1 will be underestimated because the method puts a positive probability mass at $E = 1$ and the estimated distribution is therefore not a good estimate of the empirical distribution near $E = 1$.

6.4.2 Smoothing

The bootstrap sample will nearly always contain repeated values, and if n is small, then it will even contain values repeated several times. To avoid spikes in the distribution like those that we saw in Fig. 6.3, it is advisable to use a *smoothed bootstrap* method to smoothe the distribution. As before, we want to bootstrap the sample $(x^1, \ldots, x^K)$. Here, the sample is constructed in the following way: For $r = 1, \ldots, K$

1. choose k at random with a replacement from $\{1, \ldots, K\}$,
2. generate ϵ from a standard normal distribution,
3. set $z^r = x^k + h\epsilon$ and call h the window or band width.

Our bootstrap sample is then $(z^1, \ldots, z^K)$, not a real sample from the original sample $(x^1, \ldots, x^K)$, but a smoothed sample. In this way, we smoothen the fixed number of points to imitate a continuous distribution function of the inputs x. The distribution for these smoothed points is a normal distribution with variance h^2 and is therefore symmetric around the observation points. When we use the bootstrap sample to calculate the efficiencies E, there might be a problem for efficiencies near the boundary at 1 because they must be equal to or below 1. To handle problems related to E near 1, we can use a reflection method, augmenting the dataset by adding reflections of all the points in the bootstrap; i.e. whenever we have efficiency E, we augment the dataset with the reflection on 1, $2 - E$, such that E and $2 - E$ are symmetric around 1. Then, we simply use the value below or equal to 1.

6.4.3 Bias and bias correction

In what follows we shall use the following terms:

θ^k	The true efficiency based on the true but unknown technology T
$\hat{\theta}^k$	DEA-estimated efficiency and $\widehat{T}$ the estimated DEA technology
θ^{kb}	The bootstrap replica b estimate based on the replica technology T^b
θ^{k*}	The bootstrap estimate of θ^k
$\tilde{\theta}^k$	The bias-corrected estimate of θ^k

The DEA estimate is upward biased: if there are no measurement errors, then all of the observations in the sample are from the technology set $\widehat{T} \subset T$. Then in $\hat{E}^k \geq E^k$, because we are minimizing over a smaller set (i.e. the estimated efficiency is an upward-biased estimate of E^k,), the estimated efficiency may be larger than the real efficiency. The size of $\widehat{T}$ depends on the sample, and therefore, E^k is sensitive to sampling variations in the obtained frontier. If there are measurement errors, then there is no direct subset relation between $\widehat{T}$ and T.

To eliminate the bias, we first estimate the bias and obtain a bias-corrected estimate. We can estimate the bias as

$$\text{bias}^k = \text{EV}(\hat{\theta}^k) - \theta^k.$$

Unfortunately, we do not know the distribution of θ^k, so we cannot calculate $\text{EV}(\hat{\theta}^k)$. This is where the bootstrap enters in. When θ^{kb} is a bootstrap replica estimate of θ^k, the bootstrap estimate of the bias is

$$\text{bias}^{k*} = \frac{1}{B}\sum_{b=1}^{B} \theta^{kb} - \hat{\theta}^k = \bar{\theta}^{k*} - \hat{\theta}^k.$$

A bias-corrected estimator of θ^k is then

$$\tilde{\theta}^k = \hat{\theta}^k - \text{bias}^{k*} = \hat{\theta}^k - \bar{\theta}^* + \hat{\theta}^k = 2\hat{\theta}^k - \bar{\theta}^{k*}.$$

The precision of the estimates can be determined based on the variance of the bootstrap estimate

$$\hat{\sigma}^2 = \frac{1}{B}\sum_{b=1}^{B} (\theta^{kb} - \bar{\theta}^{k*})^2.$$

6.5 Algorithm to bootstrap DEA

We have argued that the naive use of standard bootstrap methods is not satisfactory for DEA models, and we have discussed how to improve by smoothing and bias cor-

Table 6.6 Description of simplified version of `boot.sw98`

(1) Compute $\hat{\theta}^k$ as solutions to $\min\{\,\theta \mid (\theta x^k, y^k) \in \hat{T}\,\}$ for $k = 1, \ldots, n$.

(2) Use bootstrap via smooth sampling from $\hat{\theta}^1, \ldots, \hat{\theta}^K$ to obtain a bootstrap replica $\theta^{1*}, \ldots, \theta^{K*}$. This is done as follows

(2.1) Bootstrap, sample with replacement from $\hat{\theta}^1, \ldots, \hat{\theta}^K$, and call the results $\beta^1, \ldots, \beta^K$.

(2.2) Simulate standard normal independent random variables $\epsilon^1, \ldots \epsilon^K$.

(2.3) Calculate

$$\tilde{\theta}^k = \begin{cases} \beta^k + h\epsilon^k & \text{if } \beta^k + h\epsilon^k \le 1 \\ 2 - \beta^k - h\epsilon^k & \text{otherwise} \end{cases} \qquad \begin{array}{l}\text{(Smoothing and reflection}\\ \text{cf. page 172)}\end{array}$$

Note that by construction, $\tilde{\theta}^k \le 1$.

(2.4) Adjust $\tilde{\theta}^k$ to obtain parameters with asymptotically correct variance, and then estimate the variance $\hat{\sigma}^2 = \frac{1}{n}\sum_{k=1}^{K}(\hat{\theta}^k - \bar{\hat{\theta}})^2$ and calculate

$$\theta^{k*} = \bar{\beta} + \frac{1}{\sqrt{1 + h^2/\hat{\sigma}^2}}\,(\tilde{\theta}^k - \bar{\beta})$$

where $\bar{\beta} = \frac{1}{n}\sum_{k=1}^{K}\beta^k$.

(3) Calculate bootstrapped input based on bootstrap efficiency $x^{kb} = \frac{\hat{\theta}^k}{\theta^{k*}} x^k$.[1]

(4) Solve the DEA program to estimate θ^{kb} as

$$\theta^{kb} = \min\{\,\theta \ge 0 \mid y^k \le \sum_{j=1}^{K}\lambda_j y_j,\ \theta x^k \ge \sum_{j=1}^{K}\lambda_j x_j^{kb},$$
$$\lambda_j \ge 0,\ \sum_{j=1}^{K}\lambda_j = 1\,\} \qquad (k = 1, \ldots, n)$$

(5) Repeat the steps from (21) to obtain the bootstrap estimates

$$(\theta^{1b}, \ldots, \theta^{Kb}) \qquad (b = 1, \ldots, B)$$

(6) Calculate the mean and variance of $(\theta^{1b}, \ldots, \theta^{Kb})$ to get the bootstrap estimate θ^{k*}, the bias-corrected estimate $\tilde{\theta}^{k*}$, and the variance.

rection. We now present present a simplified method with smoothing of the method used in the R function `boot.sw98`.

It has been suggested that $B = 1000$ is suitable for calculating confidence intervals.

Table 6.7 Simplified version of `boot.sw98` in R

```
y <- cbind(1,2,3,4,5)
x <- cbind(2,4,3,5,6)

B <- 1000
thetaboot <- matrix(nrow=B, ncol=dim(x)[2])
thetati <- matrix(nrow=B, ncol=dim(x)[2])
# (1)
theta <- 1/dea(x,y,RTS=1,ORIENTATION=1)
N <- length(theta)
h <- 0.014  # bandwidth

# (2.1)
for ( b in 1:B) {
  beta <- sample(theta, N, replace=TRUE)
# (2.2)
  eps <- rnorm(N)
  thetatilde <- rep(0,N)
# (2.3)
  for (i in 1:N) {
    if ( beta[i]+h*eps[i] <= 1.0 ) {
      thetatilde[i] <- beta[i]+h*eps[i]
    } else {
      thetatilde[i] <- 2.0 -beta[i] -h*eps[i]
    }
  }
  thetati[b,] <- thetatilde
# (2.4)
  v = var(theta)
  thetastar = mean(beta) + (thetatilde-mean(beta))/(sqrt(1.+h^2/v))
# (3)
  xstar = theta/thetastar * x
  xstar = matrix(1,dim(x)[1],1) %*% theta/thetastar * x
# (4)
  thetaboot[b,] <-  1/dea(xstar,y,RTS=1,ORIENTATION=1)
} # for b
# done, now let's see the results
# (6)
print(colMeans(thetaboot),digits=3)
print(colMeans(thetati),digits=3)
bias <- colMeans(thetaboot) - colMeans(thetati)
print(bias,digits=3)
print(sd(thetaboot),digits=3)
boxplot(data.frame(thetaboot),boxwex=.5,ylim=c(min(thetaboot)-.1,1.05))
```

The DEA efficiency measures the radial distance in the input space from the observation point to the boundary of the technology set. We make a premature bootstrap of the efficiencies and use them to calculate the input vectors with this bootstrapped efficiency; this is done in step 3 in the above description. These bootstrapped input vectors are the inputs that determine the bootstrapped technology set in step 4 from which the final bootstrapped efficiency estimates are calculated. Note that x^{kb} is on the same ray as x^k. We could change this by also making the ray a random variable in the form of angles to be bootstrapped —i.e. by using polar coordinates to express x^k instead of the usual rectangular coordinates.

Please note that `boot.sw98` in FEAR bootstraps the Shaphard efficiency, and not Farrell efficiency as the R program does in Table 6.7. This is not a problem because the user has access to the individual bootstrap replica estimates in the component `boot` and then can just use `1/boot` for the Farrell bootstrap estimates.

6.5.1 Confidence intervals

As mentioned in Sect. 6.3.1, it is not advisable to calculate 95% confidence intervals because $\tilde{\theta}^k \pm 1.96\sigma_\theta$ as the distribution might not be a normal or symmetric; rather, it could be a skewed distribution or could have larger or smaller tails than the normal distribution. Instead, it is advisable to use the R function `quantile`. That is, to calculate a 95% confidence interval for firm 3, use

```
quantile(thetaboot[,3],probs=c(.025, .975),type=8)
```

If we do not include the firm index, here `3`, then the interval is based on all firms. This does not make any sense because the different firms have different efficiency levels, and we must determine the confidence interval for one firm at the time. For a 90% confidence interval, we just use `probs=c(.05,0.95)`. To determine the intervals for all firms, we can use

```
apply(thetaboot, 2, function(x) {
     quantile(x,probs=c(.025, .975), type=8, na.rm=TRUE) })
```

In the R function `boot.sw98` as part of the FEAR package, the confidence interval is estimated for the bias-corrected distance function values.

6.6 Numerical example in R

We will use the small examples from Table 6.8 to estimate the standard errors of the efficiency estimates and the confidence intervals for the input distance functions with a variable return technology. The R program including the data using the function `boot.sw98` is shown in Table 6.9.

The output is shown in Table 6.10. Note that if the aim is to obtain estimates of variance, the number of replicates, the value of the parameter `NREP`, must be at

Table 6.8 1 input og 1 output example

Firm	x	y
1	2	1
2	4	2
3	3	3
4	5	4
5	6	5

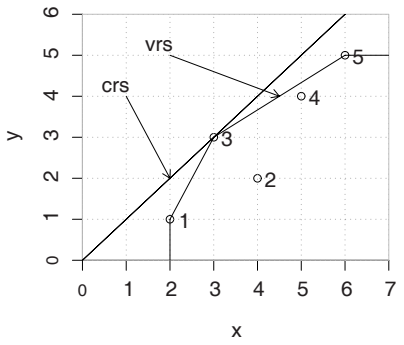

Table 6.9 Bootstrap DEA, R program

```
library(FEAR)
# Data
y <- cbind(1,2,3,4,5)
x <- cbind(2,4,3,5,6)

# DEA, Shephard input distance function,
d <- FEAR::dea(x,y, RTS=1, ORIENTATION=1)
# Efficiencies
print(1/d,digits=3)
print(mean(1/d),digits=3)

#  Bootstrap
b <- boot.sw98(x,y, RTS=1, ORIENTATION=1, NREP=2000)
print(b,digits=3)
print(sqrt(b$var),digits=3)
```

least 50; correspondingly, to obtain confidence intervals, at least 100 are required. It might also be necessary for the number of replicates to be much larger to obtain stable results for larger datasets; however, that relation has not been tested as of this writing. Part of the output is the individual replications, returned as item `boot`. All of the output items are described in the help file for `boot.sw98` in the FEAR package; from inside R, we use the command `?boot.sw98`. In the last line, we have calculated the standard error of the input distance, the square root of the variance.

The above method is very simple to use in practice. However, it does have a pedagogical drawback: everything is hidden in the function `boot.sw98`. To make up for this, we mimicked the function in R statements to see the inner working of bootstrap in DEA, just as we did for the traditional bootstrap procedure in section 6.3 on page 165.

The bias-corrected estimate is in item `dhat.bc` and can also be found by subtracting the bias from the DEA estimate of the distance function value, item `dhat`; i.e. `b$dhat − b$bias`. The confidence interval is estimated around the bias-

Table 6.10 Output from bootstrap

```
> # Efficiens
> print(1/d,digits=3)
[1] 1.000 0.625 1.000 0.900 1.000
> print(mean(1/d),digits=3)
[1] 0.905
> #  Bootstrap
> print(b,digits=3)
$bias
[1] -0.143 -0.151 -0.130 -0.101 -0.150

$var
[1] 0.00914 0.01061 0.00707 0.00538 0.01358

$conf.int
     [,1] [,2]
[1,] 1.01 1.35
[2,] 1.61 1.99
[3,] 1.01 1.30
[4,] 1.12 1.42
[5,] 1.00 1.41

$dhat
[1] 1.00 1.60 1.00 1.11 1.00

$dhat.bc
[1] 1.14 1.75 1.13 1.21 1.15

$boot
      [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  ...
[1,] 0.560 0.565 0.565 0.572 0.572 0.576 0.590 0.591  ...
[2,] 0.983 0.986 1.027 1.037 1.086 1.092 1.107 1.119  ...
[3,] 0.569 0.602 0.611 0.614 0.622 0.626 0.628 0.631  ...
[4,] 0.662 0.701 0.702 0.708 0.713 0.713 0.723 0.727  ...
[5,] 0.533 0.537 0.541 0.543 0.544 0.545 0.550 0.551  ...
...
     [,1997] [,1998] [,1999] [,2000]
[1,]    1.00    1.00    1.00    1.00
[2,]    1.60    1.60    1.60    1.60
[3,]    1.00    1.00    1.00    1.00
[4,]    1.11    1.11    1.11    1.11
[5,]    1.00    1.00    1.00    1.00

> print(sqrt(b$var),digits=2)
[1] 0.096 0.103 0.084 0.073 0.117
```

corrected estimate. The default confidence interval is 95% but can be changed using the option `alpha`. Either a scalar option or a vector option is available, indicating the statistical sizes of the confidence intervals to be estimated. Thus, `alpha=.1` will calculate limits corresponding to a $1.0 - 0.1 = 90\%$ interval.

To explain the confidence interval further, let us recalculate the Shephard input values to Farrell input values by calculating the reciprocal. This is done below where the output from the R commands is also shown.

```
> 1/b$dhat
[1] 1.000000 0.625000 1.000000 0.900009 1.000000
> 1/b$dhat.bc
[1] 0.8764797 0.5707759 0.8855686 0.8228137 0.8705459
> 1/b$conf.int[,c(2,1)]
          [,1]      [,2]
[1,] 0.7439961 0.9932824
[2,] 0.5030548 0.6218341
[3,] 0.7764515 0.9935884
[4,] 0.7085692 0.8951720
[5,] 0.7082100 0.9940264
```

Because of the reciprocal property, the upper limit becomes the lower limit and vice versa, and that is why the index in `$conf.int` is reversed. These numbers indicate that the upper limit of the confidence interval `1/b$conf.int` is very close to the estimated efficiency `1/b$dhat`, whereas the lower limit is far below. The closeness of the upper limits and the efficiencies means that the frontier corresponding to the upper limit coincides with the DEA-estimated frontier. The lower limit in the confidence interval for the efficiencies corresponds to a frontier to the left of the DEA frontier; if we measure the efficiency of the observations against this frontier, we get the lower limits of efficiency; this frontier is shown in Figure 6.5 on the next page as a dotted frontier. This frontier corresponding to the lower limit of the efficiencies is far from the efficiency estimates because a variation in inputs during the bootstrap procedure in which the input gets smaller will enlarge the technology set and move it to the left (as the new input can be outside the frontier) and will therefore create a new frontier. A larger input, on the other hand, will mostly leave the frontier unchanged because it will be below the already existing frontier. Note that bias-corrected efficiency is more likely to be in the middle of the confidence interval because bias correction is intended to correct for the derived bias or skewness in the DEA estimation.

6.7 Interpretation of the bootstrap results

To further example how to interpret the DEA bootstrap results, let us investigate two special cases. The first contains just one input and one output, whereas the second contains two inputs and one output.

6.7.1 One input, one output

Let us take a closer look at the output in Table 6.10 on page 178 from the R commands in Table 6.9. This is a small problem involving 5 firms, 1 input, and 1 output. Bootstrap is conducted using the method `boot.sw98`, and the output includes several components or items. The item named `$dhat` is the estimated Shephard input distance Function, which is equal to the reciprocal of technical input efficiency, Farrell efficiency; i.e. $TE = \frac{1}{\texttt{b\$dhat}}$. We can see this by comparing lines 8, 13, and 34. The bias-corrected Shephard input distance function is found to be `$dhat.bc` in line 37. The bias-corrected Shephard input distance functions can also be found by subtracting the bias from the DEA estimates; i.e. as `$dhat` − `$bias` ; cf. our discussion of this idea in Sect. 6.4.3 on page 173.

If the bias-estimated distance input function value is $\tilde{\theta}$, then a point on the bias-corrected frontier is $\frac{1}{\tilde{\theta}}x$ where x is the observation of the input. Because we are looking at input functions and input efficiency, the output y remain the same.

We can plot the observations and the input corresponding to the bias-corrected Sheppard input distance function by

```
dea.plot.frontier(x,y,txt=1:N)
dea.plot.frontier(x/b$dhat.bc,y,lty="dashed",add=T)
dea.plot.frontier(x/b$conf.int[,2],y,lty="dotted",add=T)
```

The options `lty` specify the line type; the default is `solid`.

The resulting figure is shown in Fig. 6.5. If we were to draw a random sample

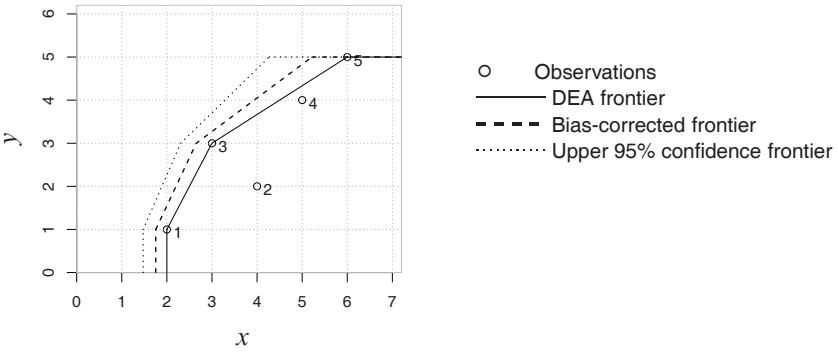

Fig. 6.5 Bias-corrected frontier, input direction

to estimate the frontier, it would to the right of the 95% confidence frontier with a probability of 95%.

Another way to demonstrate efficiency and confidence intervals is As shown in Fig. 6.6, constructed using the following R commands:

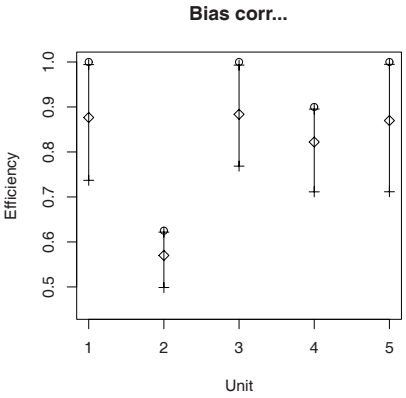

Fig. 6.6 Bias-corrected efficiency estimates (◇), DEA estimates (○) and 95% confidence limits—one input, one output

```
plot(1/b$dhat,ylim=c(.45,1),main="Bias corr...",
     xlab="Firm", ylab="Efficiency")
points(1/b$dhat.bc,pch=5)
for ( i in 1:5 )lines(rep(i,2),1/b$conf.int[i,],type="o",pch=3)
```

6.7.2 Two inputs

The isoquants for the two inputs are calculated using the following R program, which is similar to the program for one input and one output in Table 6.9 on page 177 except that the isoquant is plotted instead of the frontier. To plot the isoquant, we have normalized the inputs with the output and then used an output of 1 for all firms because then all firms have the same isoquant and can be compared. Thus, implicitly, we are assuming constant returns to scale.

```
# The data
y <- t(matrix(c(1,2,3,1,2)))
x <- t(matrix(c(2,2,6,3,6,  5,4,6,2,2), ncol=2))
N <- dim(x)[2]
x1 = x[1,]/y
x2=x[2,]/y
# The frontier for the technologies
dea.plot.isoquant(x1,x2,txt=1:N)
# The observations have dotted lines from origo
for ( i in 1:length(y) ) {
   lines(c(0,x1[i]), c(0,x2[i]),lty="dotted")
}
# bootstrap
b <-boot.sw98(rbind(x1,x2),matrix(rep(1,N),nrow=1),NREP=2000,RTS=3)
dea.plot.isoquant(x1/b$dhat.bc,x2/b$dhat.bc,lty="dashed",add=T)
```

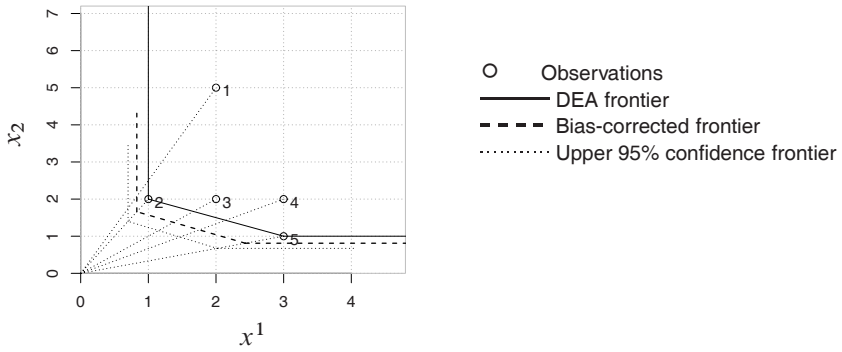

Fig. 6.7 Bias-corrected frontier, input direction, 2 inputs

```
dea.plot.isoquant(x1/b$conf.int[,2],x2/b$conf.int[,2],lty="dotted",add=T)
```

The graphs are in Fig. 6.7. Again, we can see that the bias-corrected frontier is below the Isoquant, making the technology set larger, and that the upper confidence limit is increasing it even further.

The graph in Fig. 6.8 is made using the R program lines

```
plot(b$dhat,ylim=c(1,3),main="Bias␣corr...",
       xlab="Firm",ylab="Distance␣function")
points(b$dhat.bc,pch=5)
for ( i in 1:5 )lines(rep(i,2),b$conf.int[i,],type="o",pch=3)
```

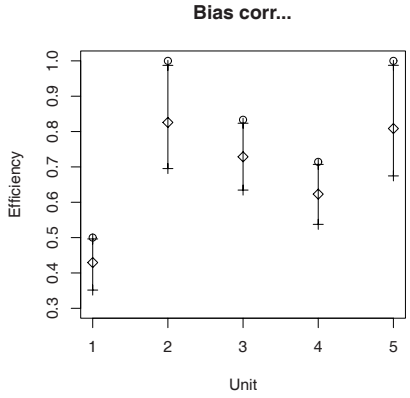

Fig. 6.8 Bias-corrected efficiency estimates (◇), DEA estimates (○) and 95% confidence limits – two inputs

6.8 Statistical tests using bootstrapping

Let us finally illustrate how to use bootstrapping to test hypotheses. Specifically, we will show how to tests a returns to scale hypothesis, but other tests can be developed along the same lines.

We wish to test whether the technology set T from which our observations are sampled exhibits constant returns to scale. Formally, we wish to test the hypothesis that the technology exhibits constant returns to scale against the alternative, that it is VRS:

$$H_0 : T \text{ is CRS}$$
$$H_A : T \text{ is VRS}$$

If we reject H_0, then we can test if the technology set is DRS, but we will leave that project to the reader.

If the hypothesis is true, then the efficiencies calculated from the VRS technology are the same as the efficiencies calculated from the CTS technology. If there is not CRS, then at least one of the efficiencies will be different; i.e. CRS efficiency will be smaller than VRS efficiency. One way to examine this is to see whether the scale efficiency, cf. page 99,

$$SE^k = \frac{E^k_{\text{CRS}}}{E^k_{\text{VRS}}} \qquad (k = 1, \ldots, K)$$

is equal to 1 for all firms, meaning that the technology is CRS, or whether there is at least one firm where it is less than 1, meaning that the technology is VRS. For a given set of observations of K firms, we must therefore reject the hypothesis if at least one of the estimated SE has a value less than 1. However, as the connection between the technology set and the scale efficiencies is an uncertain or stochastic connection, we must reject the hypothesis if at least one of the estimated SE has a value significantly less than 1, i.e. if one of the estimated SE is less than a critical value. The problem is then to compute this critical value.

Instead of looking at the scale efficiencies individually, we could look at the test statistic

$$S^1 = \frac{1}{K} \sum_{k=1}^{K} \frac{E^k_{\text{CRS}}}{E^k_{\text{VRS}}}$$

or the one that we are going to use in the following:

$$S = \frac{\sum_{k=1}^{K} E^k_{\text{CRS}}}{\sum_{k=1}^{K} E^k_{\text{VRS}}}. \tag{6.1}$$

If the H_0 is true, then S will be close to 1, and if the alternative is true, then $S < 1$. As $S \leq 1$ by construction, we will reject H_0 if S is significantly smaller than 1. We therefore seek a critical threshold for the statistic S; if it is smaller than this

value, then we will reject the hypothesis. Thus, we seek a critical value c_α that will determine whether we reject H_0, the hypothesis of constant returns to scale, if $S < c_\alpha$ and $\Pr(S < c_\alpha \mid H_0) = \alpha$ where α is the size of the test, typically 5% ($\alpha = 0.05$). The size of the test, α is the probability of rejecting the hypothesis even though it is true. (This is a type I error.)

Unfortunately, we do not know the distribution of S under H_0, and therefore, we cannot calculate c_α directly. One way to address this lack of distributional knowledge is to use a bootstrap method, and we will now show that one can bootstrap the distribution of S under H_0. We show how this can be done using a very small example: the data from Table 6.8 on page 177. First, we enter the data and calculate the statistic S and its quantile using the following commands in R:

```
library(FEAR)
y <- cbind(1,2,3,4,5)
x <- cbind(2,4,3,5,6)
e <- 1/dea(x,y,RTS=3)
ev <- 1/dea(x,y,RTS=1)
sum(e)/sum(ev)
nrep <- 2000
Bc <- boot.sw98(x,y,NREP=nrep,RTS=3)
Bv <- boot.sw98(x,y,NREP=nrep,RTS=1,XREF=x,YREF=y,DREF=1/e)
s <- colSums(1/Bc$boot)/colSums(1/Bv$boot)
quantile(s,c(1,2,5,10,15,30,50)/100.0)
```

We calculate the CRS efficiency (`RTS=3`), the VRS efficiency (`RTS=1`), and the test statistic S from (6.1). The following lines is the bootstrap. First, the variable `nrep` is set to the number of bootstrap replications that we will use. Then, we bootstrap under the null-hypothesis. Thereafter, we bootstrap under the alternative while assuming that H_0 is in fact true by using the option `DREF=1/e` where `1/e` is efficiency calculated under the CRS technology.

The output is shown in Table 6.11. The estimate of S is 0.802945, which seems to

Table 6.11 Output for test of constant returns to scale

```
> y <- cbind(1,2,3,4,5)
> x <- cbind(2,4,3,5,6)
> nrep <- 2000
> e <- 1/dea(x,y,RTS=3)
> ev <- 1/dea(x,y,RTS=1)
> sum(e)/sum(ev)
[1] 0.802945
> Bc <- boot.sw98(x,y,NREP=nrep,RTS=3)
> Bv <- boot.sw98(x,y,NREP=nrep,RTS=1,XREF=x,YREF=y,DREF=1/e)
> s <- colSums(1/Bc$boot)/colSums(1/Bv$boot)
> quantile(s,c(1,2,5,10,15,30,50)/100.0)
       1%        2%        5%       10%       15%
30%       50%
0.7409859 0.7431850 0.7472870 0.7531393 0.7585869 0.7940538 0.8561436
```

be far less than 1, but we only have 5 firms, and the output from `quantile` shows that .80 corresponds to a little more than 30%. Therefore, there is a 30% probability of observing a lower value of S than the one we obtained, and therefore, we do not reject H_0; i.e. we do not reject that there exist constant returns to scale. If we were to make further calculations under this model, we would therefore assume constant returns to scale and use a CRS technology.

Earlier, we introduced the idea of the critical value, which can be calculated using the function `critValue`, which takes the bootstrapped statistics and the size of the test as input. We also have at our disposal the function `typeIerror`, which calculates the probability of type I error: the probability of rejecting the hypothesis if it is true.

```
critValue <- function(s,alfa)  {
  ss <- sort(s)
  mean( ss[floor(alfa*length(s))], ss[ceiling(alfa*length(s))] )
}

typeIerror <- function(shat,s) {
  reject <- function(alfa)  {
    quantile(s,alfa,names=F) - shat
  }
  uniroot(reject,c(0,1))$root
}
```

Both functions are part of the Benchmarking package. Using the two functions with the data above yields the output

```
> shat <- sum(e)/sum(ev)
> shat
[1] 0.802945
> critValue(s,0.05)
[1] 0.7418619
> typeIerror(shat,s)
[1] 0.3337649
```

Thus, if the estimated value of S is less than the critical value 0.7418619, we reject the hypothesis. Correspondingly, because the estimate of S, `shat`, is 0.802945, we do not reject the hypothesis. The results obtained using `typeIerro` show that there is a probability of 0.3337649 that one will obtain a lower estimate of S than the one we found, or in other words, that we will be making a mistake if we reject the hypothesis on the basis of our estimate.

6.9 Summary

DEA originates in the operations research and management science, and this means that the evaluation of DEA models is not a purely statistical exercise. Indeed, historically the use of traditional statistical tests has not been emphasized. Considerable progress has however been made in this respect over the last 15 years, and we introduced some important contributions in this chapter.

One possibility is to use general non-parametric tests, i.e. tests used when the underlying distribution is unknown, like Kolmogorov–Smirnov tests and Kruskal–Wallis tests. Such tests can be used to evaluate a series of different assumptions and hypothesis but as always they may suffer from limited power.

Another possibility is to rely on parametric tests. If we can make reasonable assumptions regarding the underlying distribution of inefficiency and noise in the data, a series of tests are possible. We discussed tests for group differences and tests for model assumptions. To justify the distributional assumptions in a parametric approach, we may rely on asymptotic theory, i.e. theoretical properties that can only be established for large samples. Simulation studies based on samples of moderate size suggests that such assumptions may well be justified in many applications.

A third approach, and one that has become particularly popular with the development of effective computer programs, is the use bootstrapping. The bootstrap is a computer-based method that can answer many statistical questions. The approach replicates sampling uncertainty by creating repeated samples of the original sample. We spend most of this chapter covering bootstrap-based inference in DEA models. In particular, we showed how to make bias corrections and construct bias corrected confidence intervals for the individual efficiencies. One advantage of R is that effective bootstrapping methods for DEA models have been made easily available, not the least via the FEAR package.

In the appendix, we discuss the use of statistical methods in second-stage analyses, i.e. analyses performed after the development of a benchmarking model, to validate the model and to explore the possible causes of the variations in efficiencies. A common approach in such studies is tobit regression, and we discuss how to perform and interpret such an analyses.

6.10 Bibliographic notes

Consistency of DEA estimates and asymptotic tests are based on Banker (1993) and Banker (1996).

The bootstrap method was invented in 1979 and it is now a well established statistical method. A good reference to the statistical theory of bootstrap with lot of examples is Efron and Tibshirani (1993); the mathematical level of the book is moderate. Our description of the bootstrap, and in particular Table 6.1 is taken from that book. A more advanced text assuming a grounding in statistics is Davison and Hinkley (1997). The reflection method is described in (Silverman, 1986, 30).

R is based on S, a language and an environment for data analysis. Bbootstrap methods have been in S almost since the beginning (Chambers and Hastie, 1992).

Bootstrap of DEA model have a winding history, the first attempt was done around 1992. The bootstrap method for DEA described in this book is from Simar and Wilson (1998) and Simar and Wilson (2000). Their approch is implemented in R as `boot.sw98` as part of the FEAR library (Wilson, 2008). The simplified

description of `boot.sw98` in Table 6.6 by and large follows Simar and Wilson (1998).

The tobit model covered in the Appendix was first used by Tobin in 1958 (Tobin, 1958), is discussed in many textbooks, including Greene (2008) and Maddala (1983). The tobit model is traditionally used with point of truncation at 0, which makes the marginal impact relatively easy to calculate. Because efficiency scores are truncated at 1, we have derived the marginal impact for this case. An important critical paper on the tobit approach in benchmarking, Simar and Wilson (2007), instead proposes the use of bootstrapping. Hoff (2007) also identifies a number of theoretical issues associated with current practice, but she concludes after analyzing an actual dataset that the tobit procedure does produce reasonable estimates and, moreover, can be substituted for by a regular OLS approach under some conditions. McDonald (2009) questions whether the DEA scores should be seen as a censored distribution, arguing for the use of a "fractional" model, but he also concludes that theoretical niceties are of little concern to "instrumentalists", and that hundreds of two-stage DEA studies have proven very useful in providing insight into real-world production processes.

6.11 Appendix: Second stage analysis

When we have estimated the efficiencies of the firms in an industry, we often become interested in understanding why some firms are more efficient than others. Is their efficiency related to firm size, CEO age, the fraction of highly educated employees at the firm, the use of ICT, the business environment in different regions, and/or other factors?

We may also wonder if the variations in estimated efficiency really reflect variations in performance or if we may have left out important inputs or output (i.e. we might be interested in validating the model). Should we have included a measure of soil quality in a farming model, a measure of socio-economic status of the model examining students in a school, or a measure of quality in a hospital model? In developing a benchmarking model for German DSO regulation, cf. Sect. 10.3, we did, for example, make a final evaluation of several hundreds of omitted candidate variables.

Both aims are often pursued using what is commonly called second-stage analysis, i.e. post-efficiency analysis that aims to explain the variations and validate the model. In this appendix, we discuss the use of statistical methods in second-stage analyses. The relevance of such analyses and the corresponding methods is not restricted to DEA studies. Other best-practice results can be analyzed using the same methods.

To investigate if *categorial variables* like high/low, east/west, and low/medium/high may explain some of the variation, we can use a number of non-parametric tests: e.g. the Mann-Whitney-Wilcoxon rank-sum test. This is a non-parametric test used to assess whether two independent samples of observations have equally

large values. This process is largely equivalent to performing an ordinary parametric two-sample *t*-test on the data after ranking the combined sample. We can also use other non-parametric tests like the Kolmogorov-Smirnov and Kruskal-Wallis tests, as demonstrated in Sect. 6.2.1. All tests can easily be undertaken in R.

The most common approach used to investigate if a set of *continues variables* variables may explain the variations in efficiency is to conduct a tobit regression. Tobit regression is similar to ordinary regression analysis except that the noise term is truncated. The use of this method in a benchmarking context is the focus of some debate in the literature (cf. below), but it is widely applied and is generally considered to be useful.

Let E be the Farrell input efficiency calculated in a DEA model, an SFA model or some combination of models (cf. e.g. the combined use of several models in regulatory benchmarking as explained in Chap. 10). We will return to models of output efficiency later. We are now interested in modeling how E depends on other variables $z = (z_1, z_2, \ldots, z_q)$. That, is we would like to estimate a model

$$E = g(z, a).$$

whereby efficiency E is explained by the variables z and parameters a.

6.11.1 Ordinary linear regressions OLS

A model is a linear regression ,model

$$E = a_1 1 + a_2 z_2 + \cdots + a_q z_q + \varepsilon = az + \varepsilon$$

where ε is a random error that reflects that the model does not completely explain the efficiency levels. It is easy to estimate this model using OLS. In R, this can be done using the function `lm`.

One advantage of this approach is that it is easy to find the marginal effect on efficiency based on a marginal change in z_j:

$$\frac{\partial E}{\partial z_j} = a_j,$$

Because this effect is independent of the value of all the variables, it is also easy to interpret—it shows how much the efficiency tends to increase if a_j is increased by one unit.

Although ordinary regressions are widely used in practice, they suffer from a theoretical problem in a benchmarking setting. They do not take into account that efficiencies are greater than 0 and less than or equal to 1 and that many efficiencies are typically at the upper boundary of 1. There is nothing in the method that ensures that the fitted value, the expected value, or the mean will be less than or equal to 1. The tobit model for censored regression can be used to solve this problem.

6.11.2 Tobit regression

When the dependent variable is censored, we do not observe the underlying values of this variable in all cases. Values in a specific range are reported as a single value. In the case of E, we can see the underlying efficiencies as a stochastic variable and the observation of efficiency E as a a censored version hereof where values below 0 are reported as 0 and values above 1 are reported as one. Therefore, the model becomes

$$E = \begin{cases} 0, & \text{if } az + \varepsilon \leq 0 \\ az + \varepsilon & \text{if } 0 < az + \varepsilon < 1 \\ 1 & \text{if } az + \varepsilon \geq 1 \end{cases}$$

Our challenge is to estimate a on the basis of the observed efficiencies E^k from K firms $k = 1, \ldots, K$.

In general, we do not have any firms with reported efficiency of 0. Therefore, let K_1 be the number of firms for which $E = 1$ (i.e. the number of efficient firms) and K_0 be the number of firms for which $E < 1$. We then have $K = K_0 + K_1$.

The probability that $E = 1$ is the probability that $az + \varepsilon >= 1$. Let F be the probability distribution function for ε and f the corresponding density function. Then the probability of $E = 1$ is

$$\begin{aligned} \Pr(E = 1) = \Pr(az + \varepsilon \geq 1) &= 1 - \Pr(az + \varepsilon < 1) \\ &= 1 - \Pr(\varepsilon < 1 - az) = 1 - F(1 - az), \end{aligned}$$

and the probability that $E = 0$ is

$$\Pr(E = 0) = \Pr(az + \varepsilon \leq 0) = \Pr(\varepsilon < -az) = F(-az).$$

The case where in which $0 < E < 1$ corresponds to $E = az + \varepsilon$ or $\varepsilon = E - az$ such that the density is this case [ED21]is $f(E - az)$.

The likelihood function for K observations of efficiencies is then given as the product of the K individual terms for the cases mentioned above.

$$\begin{aligned} L &= \prod_{k:E^k=1} \Pr(E^k = 1) \prod_{k:0<E^k<1} f(E^k - az^k) \\ &= \prod_{k:E^k=1} \left(1 - F(1 - az^k)\right) \prod_{k:0<E^k<1} f(E^k - az^k). \end{aligned}$$

We have here not taken into account that E in the theory could be equal to 0. Because the number of such observations is 0, the corresponding likelihood factor is 1 irrespective of the value of $\Pr(E = 0)$.

To estimate the above model, we also need to choose a probability distribution F. The most commonly used distribution is the normal distribution, and in this case, the model is called the *tobit regression* model. We will not formulate the likelihood

function in this particular case but will instead refer the reader to the literature mentioned in the bibliographic notes. The actual optimization process is conducted using standard iterative optimization routines that are also available in R. As part of the estimation process using standard programs, the variance of the estimated parameters is also calculated such that statistical inference is possible.

Now, in benchmarking applications, we are typically interested in knowing the marginal effect of a marginal change in one of the explanatory variables z. In the OLS framework, these effects are readily available as the parameter estimates a. In the tobit framework, they are more difficult to determine, and we will provide them here.

In the rest of this section, we use EV for the mean or expectation of a random variable to be able to distinguish the mean EV from efficiency E. We are interested in knowing how $\mathrm{EV}(E|z)$ varies with z, i.e. how a change in z influences efficiency E on average. The conditional expectation consists of three parts corresponding to the three parts of the model for E.

$$\begin{aligned}\mathrm{EV}(E|z) &= \int E\, d\Pr(E|z) \\ &= \int 0\, d\Pr(E=0|z) + \int E\, d\Pr(0<E<1|z) + \int 1\, d\Pr(E=1|z) \\ &= \int_{-az}^{1-az} \varepsilon\, d\Pr(\varepsilon|z) + 1 - \Pr(\varepsilon < 1-az|z).\end{aligned}$$

where there last equality can be verified by inserting the definition of E and making a few reformulations.

We now calculate the two probability terms separately. The last is simple to calculate when we assume that the error term is normally distributed, i.e. $\varepsilon \sim N(0,\sigma^2)$. The first term is slightly more complicated because it involves real integration. The final result is that

$$\begin{aligned}\mathrm{EV}(E|z) = az\left(\Phi\Big(\frac{1-az}{\sigma}\Big) - \Phi\Big(\frac{-az}{\sigma}\Big)\right) \\ + \sigma\left(\varphi\Big(\frac{-az}{\sigma}\Big) - \varphi\Big(\frac{1-az}{\sigma}\Big)\right) + 1 - \Phi\Big(\frac{1-az}{\sigma}\Big).\end{aligned}$$

Although this process looks complicated, the terms can interpreted simply based on the defining equation. The last two terms, $1-\Phi$, correspond to the effect of the firms where $E=1$ multiplied by the probability of this event. The first term is the linear effect az multiplied by the probability that $0<E<1$. The second term is the effect of the error term ε. In the linear model, the OLS model, this effect is zero because the expected value of ε is 0, but here, the mean of ε is conditioned to the interval where $0<az+\varepsilon<1$, i.e. $-az<\varepsilon<1-az$.

Based on the above, we can also find

$$\mathrm{EV}(E|0 < E < 1, z) = az + \sigma \frac{\varphi\left(\frac{-az}{\sigma}\right) - \varphi\left(\frac{1-az}{\sigma}\right)}{\Phi\left(\frac{1-az}{\sigma}\right) - \Phi\left(\frac{-az}{\sigma}\right)} = az + \sigma\, M(az)$$

where the function $M(\cdot)$ is called the *inverse Mills ratio*.

Now we can determine how $\mathrm{EV}(E|z)$ varies with z by finding the derivative of $\mathrm{EV}(E|z)$ w.r.t. z. To do so, we must find the derivatives of the individual terms in $\mathrm{EV}(E|z)$. We will not present the details here, but we should note that they make use of the chain rule and the fact that Φ is the antiderivative of ϕ such that $\Phi' = \varphi$ and $\Phi(t) = \int_{-\infty}^{t} \varphi(\varepsilon)\, d\varepsilon$. By collecting terms and canceling out where possible, we get

$$\frac{\partial\, \mathrm{EV}(E|z)}{\partial z_h} = a_h \left(\Phi\left(\frac{1-az}{\sigma}\right) - \Phi\left(\frac{-az}{\sigma}\right) \right). \tag{6.2}$$

Again, the results are easy to interpret: the term a_h corresponds to the linear term that we also found for the OLS model in Sect. 6.11.1, but here, it is corrected for the probability that $0 < E < 1$. If $E = 0$ or $E = 1$, then a marginal change in z will not change E.

All of the above calculations can be easily done numerically; both Φ and φ are available as functions in R, as we shall see in the numerical example.

Output efficiency and tobit

For output efficiency F, we have $F \geq 1$; therefore, the model is

$$F = \begin{cases} az + \varepsilon & \text{for } az + \varepsilon > 1, \\ 1 & \text{otherwise,} \end{cases}$$

where there is no upper bound; the bound that was an upper bound for input efficiency E is here a lower bound. To determine the expectation of F, we use some of the same terms as before. However, we use them a little differently and derive

$$\mathrm{EV}(F|z) = \Phi\left(\frac{1-az}{\sigma}\right) + az\left(1 - \Phi\left(\frac{1-az}{\sigma}\right)\right) + \sigma\, \varphi\left(\frac{1-az}{\sigma}\right),$$

and the derivative w.r.t. z_h becomes

$$\frac{\partial\, \mathrm{EV}(E|z)}{\partial z_h} = a_h \left(1 - \Phi\left(\frac{1-az}{\sigma}\right)\right).$$

Again, this corresponds to the derivative of the expected figure for input efficiency where the upper bound is now the lower bound and the upper bound is infinity. The interpretation is also as before; the linear effect a_h is multiplied by the probability that $F > 1$, i.e. 1 minus the probability that $F = 1$.

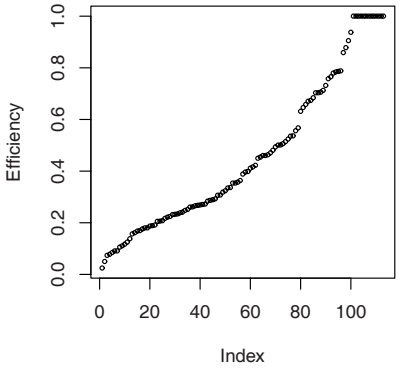

Fig. 6.9 Efficiency in Norwegian forestry

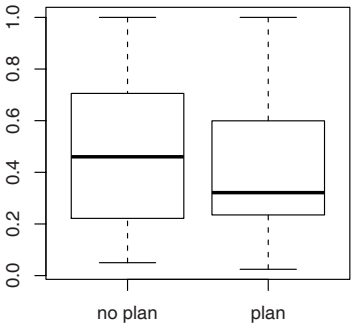

Fig. 6.10 Explaining efficiency by the absence or presence of a forest plan ($z_6 = \mathbf{0}, \mathbf{1}$)

6.11.3 Numerical example in R

We use a data set for 113 farmers in forestry in Norway. The basic DEA model is quite simple; it includes just two inputs and one output. The input variables are the value of the woodland and variable cost, and the output is earned profit. The variables that we will later use to explain efficiency, are secondary income from ordinary farming (z_1), owner age (z_3), and whether there is a long-term plan (z_6).

The input efficiencies in a variable-returns-to-scale DEA technology are shown in sorted order in Fig. 6.9. We see that there is tremendous variation in efficiency levels and that only a few firms are fully efficient. We may therefore ask what might explain this variation and what additional variables we should perhaps have included in the DEA model.

The efficiencies were calculated using the R script in Table 6.12 on the facing page, where we have also included the second step: an OLS regression and a tobit regression. The function `tobit` used to conduct tobit regressions is part of the AER package. The tobit regression is the R method `tobit` called with an input formula just like `lm` for linear regression. Numerical differences may affect the convergence, and we therefore ended up rescaling the z_1 variable by dividing the original values by 10^6; this process yielded a maximal value of 2,49.

In Fig. 6.10, the empirical box plot indicates that firms without a plan are more efficient than firms with a plan. However, the tendency is only vague, and in the OLS regression, the parameter for the plan factor, z_6 is estimated at $-.016$, which indicates that a firm with a forest plan has an efficiency level that is 1.6 percentage points lower. The standard error of the estimate is relative large, and the t-value of 0.76 shows that the parameter is not at all significantly different from zero. The same

Table 6.12 Two-stage DEA in R

```
> library(Benchmarking)
> library(AER)
> d <- read.csv("norWood2004.csv", header=T, comment.char = "#")
> x <- cbind(d$x,d$m)
> y <- d$y
> e <- dea(x,y)
> E <- eff(e)
> eOls <- lm(E ~ z1+z3+z6, data=d)
> summary(eOls)

Call:
lm(formula = E ~ z1 + z3 + z6, data = d)

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.850e-01  1.503e-01   1.231   0.2210
z1          -1.023e-07  6.062e-08  -1.688   0.0943 .
z3           7.425e-03  2.962e-03   2.507   0.0137 *
z6          -1.635e-02  5.479e-02  -0.298   0.7659
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> zz1 <- d$z1/1e6
> eTob <- tobit(E ~ zz1+z3+z6, left=-Inf, right=1, data=d)
> summary(eTob)

Call:
tobit(formula = E ~ zz1 + z3 + z6, left = -Inf, right = 1, data = d)

Observations:
         Total  Left-censored     Uncensored Right-censored
           113              0            100             13

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.165955   0.165135   1.005   0.3149
zz1         -0.125615   0.066745  -1.882   0.0598 .
z3           0.008456   0.003265   2.590   0.0096 **
z6          -0.010403   0.060475  -0.172   0.8634
Log(scale)  -1.171818   0.073290 -15.989   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

estimated parameter in the tobit model is -0.01 with a t-value of -0.17 that is also not significantly different from zero. Therefore, the tendency we see in the numbers is probably purely incidental; it is likely that having a plan does not influence efficiency.

In Fig. 6.11, the efficiencies are plotted against the variable z_1, secondary income from ordinary farming. The tendency in the figure is that the larger the secondary

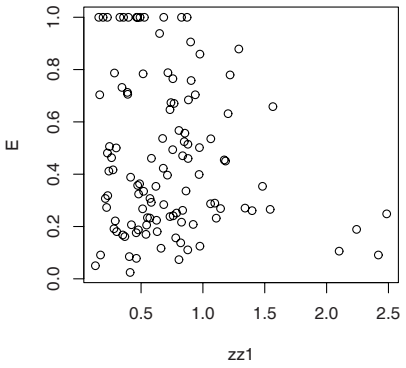

Fig. 6.11 Explaining efficiency by secondary income z_1 (rescaled to `zz1`)

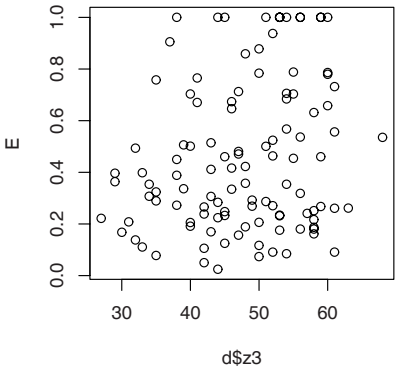

Fig. 6.12 Explaining efficiency by the age of owner z_3

Table 6.13 Tobit model with continuous age and age below 37 as an explanation

Model		Intercept	z_1	Age	z_6
Age continuous	Estimate	0.166	-0.126	0.008	-0.010
	z value	1.0	-1.9	2.6	-0.2
Age < 37	Estimate	0.594	-0.127	-0.202	-0.000
	z value	9.8	-1.9	-2.3	-0.0

income, the lower the efficiency level. This may be because farmers spend more time on secondary work and therefore neglect wood farming to some degree, which will lead to lower efficiency. The estimated parameter in the OLS regression for variable z1, determined using method `lm` as indicated in Table 6.12, is negative. This supports the impression, based on the figure, that higher secondary income is associated with lower efficiency. The parameter is only significantly different from zero at a 10% level; the t-value is only 1.77.

In Fig. 6.12, the age of the owner z_3 is plotted against efficiency, and it emerges that the effect of age is positive and significantly different from zero. The older the owner, the more efficient the firm. This may indicate that forestry farming is learned during the practice of forestry. From the figure, we can see that the increase only occurs below the age of 37. Instead of using age `z3` as a continuous variable, we can also use it as a factor with levels under 37 and over 37. The command used to estimate a tobit model, where age is this two-level factor, is

```
tobit(E~zz1+as.factor(d$z3<37)+z6, left=-Inf, right=1, data=d)
```

and the results are shown in Table 6.13, where the estimates achieved using age z_3 as a continuous variable are also shown. The difference between the two tobit models

Table 6.14 Comparing marginal effects in the Norwegian forest model

	$z_1 \cdot 10^{-6}$	z_3	z_6
OLS	-0.102	0.00742	-0.01635
Mean of effect for all firms	-0.114	0.00769	-0.00946
Effect at mean value of z	-0.116	0.00782	-0.00962
Effect at min	-0.100	0.00673	-0.00828
Effect at max	-0.111	0.00748	-0.00920

is minimal. The conclusion is that age matters, but only in the early years, and that young owner are less effective than older ones.

Let us now turn to the effect of a change in a variable. What would be the effect on efficiency if the secondary income from ordinary farming increased? As we can see from the formula, (6.2) the marginal effect of a marginal change in z depends on the value of the explanatory variables z. To calculate a marginal effect, it must therefore be for a specific value of z. The value could correspond to a specific firm or the mean firm. We could also calculate the effect for all firms and then take the mean. We will show how to do this and then compare the results with those achieved using the OLS model. In R, the value of the distribution function for a standardized normal distribution at the point x results from the function `pnorm(x)`, and the calculations corresponding to (6.2) are shown below:

```
# The tobit model
eTob <- tobit(E ~ zz1+z3+z6, left=-Inf, right=1, data=d)
# the standard error, needed for the use of standard normal dist.
s <- sqrt(var(residuals(eTob)))
# The mean at the effect for all firms
az <- fitted(eTob)
mean(coef(eTob)[2] * (pnorm((1-az)/s) - pnorm(-az/s)))
mean(coef(eTob)[3] * (pnorm((1-az)/s) - pnorm(-az/s)))
mean(coef(eTob)[4] * (pnorm((1-az)/s) - pnorm(-az/s)))
# the effect at the mean of az
az <- mean(fitted(eTob))
coef(eTob) * (pnorm((1-az)/s) - pnorm(-az/s))
# the effect at the min value of az
az <- min(fitted(eTob))
coef(eTob) * (pnorm((1-az)/s) - pnorm(-az/s))
# the effect at the max value of az
az <- max(fitted(eTob))
coef(eTob) * (pnorm((1-az)/s) - pnorm(-az/s))
# the OLS model
lm(E ~ zz1+z3+z6, data=d)
```

The results are collected in the Table 6.14.

If we increase z_1 with 1 000 000 and increase `zz1` by 1, then efficiency E in the OLS model will decrease by .102. In the tobit model for the firm with the lowest expected efficiency level, the minimum az, the effect on E is $-.100$, whereas for the firm with the highest efficiency level, the effect is $-.111$. If owner age increases by 10 years, efficiency increases $10 \times 0.0067 = 0.067$ for the youngest owners and

$10 \times 0.0078 = 0.0784$ for an owner of average age. Therefore, if the efficiency of a young owner is 60%, then after 10 years, it will be 66.7% ceteris paribus.

Note that the effect of the mean firm is -0.116, whereas the mean of the effect is -0.114. This is just a small difference, but it is sufficient to show that the change in efficiency is not linear in the tobit model.

6.11.4 Problems with the two-step method

The tobit model has been used in hundreds of studies of efficiency and productivity analysis but is also the focus of some recent debates.

An assumption in the model above is that z and ε are independently distributed. If that is not the case, the likelihood function might not factorize as the conditional likelihood function given z. If z and u are not independent, then we may have $\mathrm{EV}(u|z) \neq \mathrm{EV}(u)$, and many of our results above will not hold. For instance, the estimates based on the above-proposed second-stage methods might be biased and not inconsistent. An alternative is to use bootstrapping methods. Another option is to use stochastic frontier analysis (SFA), in which the relationship of dependence between efficiency and the other variables can be integrated into the model formulation by letting the mean and possibly the variance of the half-normal inefficiency term ε depend on z.

Still, theoretical niceties are of little concern to "instrumentalists", and there is considerable evidence of the success of two-stage studies in which scores are treated as descriptive measures.