A coherent approach to Bayesian Data Envelopment Analysis

Mike G. Tsionas[*]

Abstract

Mitropoulos et al. (Mitropoulos P. , M. A. Talias, and I. Mitropoulos, 2015, Combining stochastic DEA with Bayesian analysis to obtain statistical properties of the efficiency scores: An application to Greek public hospitals. European Journal of Operational Research 243, 302-311) suggested the use of a Bayesian approach in Data Envelopment Analysis (DEA) which can be used to obtain posterior distributions of efficiency scores. In this paper, we avoid their assumption that alternative data sets are simulated from the predictive distribution obtained from their simple data generating process of a normal distribution for the data. The new approach has two significant advantages. First, the posterior proposed in this paper is coherent or principled in the sense that it is consistent with the DEA formulation. Second, and perhaps surprisingly, it is not necessary to solve linear programming problems for each observation in the sample. Bayesian inference is organized around Markov Chain Monte Carlo techniques that can be implemented quite easily. We conduct extensive Monte Carlo experiments to investigate the finite-sample properties of the new approach. We also provide an application to a large U.S banking data set. The sample is an unbalanced panel of US banks with 2,397 bank–year observations for 285 banks. The main purpose of the analysis is to compare distributions of efficiency scores. Relative to DEA, Bayes DEA provides different efficiency scores and their sample distribution has significantly less probability concentration around unity. The comparison with bootstrap-DEA shows that results from Bayes DEA are in broad agreement.

Key Words: OR in service industries; Data Envelopment Analysis; Efficiency; Bayesian Analysis; Markov Chain Monte Carlo.

Acknowledgments: The author is grateful to the Editor, Professor Dyson, and two anonymous reviewers for their useful comments on an earlier version of the paper.

[*] Lancaster University Management School, LA1 4YX, U.K, email: m.tsionas@lancaster.ac.uk

1 Introduction

Mitropoulos et al. (2015) suggested the use of a Bayesian approach in DEA which can be used to obtain posterior distributions of efficiency scores. Mitropoulos et al. (2015), first, use Bayesian techniques to generate from the predictive distribution of the model. Secondly, they solve a chance-constrained DEA problem for every alternative sample. The model assumes that inputs and outputs are jointly normally distributed with unknown mean μ and covariance matrix Σ. Then the authors place a normal prior on μ and a Wishart prior on Σ. The use of Bayesian methods in this context is inconsequential as alternative data sets might have been generated from a normal distribution, $\mathcal{N}(\bar{Z}, S)$ where $\bar{Z}$ is the sample mean of the data and S is the sample covariance matrix. The reason is that the posterior predictive distribution converges fast (as the sample size grows larger) to the sampling distribution. Tsionas and Papadakis (2010) also proposed a Bayesian approach to stochastic or chance-constrained DEA (CCDEA) as in Mitropoulos et al. (2015). However, in their paper the data is *fixed* and *no alternative data sets are drawn*. Instead, parameters like μ and Σ are drawn from their posterior conditional distributions and the CCDEA problem is solved for each parameter draw. Tsionas and Papadakis (2010) note that the same procedure cannot be applied to non-stochastic DEA: "[S]ince the DEA problem defines an operator $\mathbf{T}$ that does not depend on the parameter vector, θ, it follows that, conditional on the data, there is no variability on $\mathbf{r}$ [the vector of efficiency scores] and, therefore, no statistical inference questions arise. In other words, the distribution of $\mathbf{r}$ is singular with respect to the posterior measure. In stochastic DEA, θ enters the problem through the dependence of T on θ, so there is scope to explore aspects of the posterior distribution of $\mathbf{r}$." (p. 311). However, as pointed out in Simar and Wilson (2000) this does not work and special care must be exercised in the application of the bootstrap, viz. in the generation of alternative data sets. Moreover, the bootstrap in Simar and Wilson (2000) does not require a normality assumption. The criticism in Simar and Wilson (2000) applies equally to the naive bootstrap as well as to a Bayesian approach that does not take into account the special features of DEA in this context. The procedure in Mitropoulos et al. (2015) is, essentially, a version of the naive bootstrap which has been heavily criticized by Simar and Wilson (2000).

Mitropoulos et al. (2015) is part of a long search for a data generating process (DGP) that can be used to perform Bayesian inferences on efficiency scores, including the bootstrap approach[1]. **The problem with assuming outright that the data are normally distributed (and independent across DMUs) is that it is not consistent with DEA itself**: First, a DGP is assumed and then DEA is applied to compute efficiency scores. In statistics, we usually proceed from a different angle. For example, suppose a certain (perhaps complicated) technique yields a given estimator, e.g. in the context of the method of moments. If we wish to perform Bayesian inferences, we need to find a likelihood function whose posterior mean or mode would be close to this estimator. [2]

The rest of the paper is organized as follows. We review the DEA problem in Section 2. We develop a stochastic

[1] For earlier work see Banker (1993, 1996), Simar and Wilson (1998, 2007) and Tsionas (2003).

[2] Although the data is given in Bayesian analysis, statistical uncertainty is allowed for as the sampling model is, for example, $\mathcal{N}(\mu, \sigma^2)$ for normal data with mean μ and standard deviation σ. The problem, in turn, is to perform inferences with respect to μ and σ *conditionally on the data.*

model for Bayesian DEA in Section 3. In Section 4 we present results from Monte Carlo experiments. An empirical application to U.S banking is presented in Section 5. We conclude by offering a summary of the paper, its main contributions and directions for future research.

2 DEA

Suppose we have a vector of inputs $x \in \mathbb{R}_{+}^{m}$ and a vector of outputs $y \in \mathbb{R}_{+}^{q}$. Suppose we have the observed data $\mathcal{D} = \{(x_i, y_i)\,\forall i = 1, ..., n\}$. Given a production set $\mathcal{P} = \left\{(x, y) \in \mathbb{R}_{+}^{m+q} | (x \text{ can produce } y\right\}$ the DEA approach uses the convex and free-disposal hulls:

$$\mathcal{T} = \left\{(x, y) \in \mathbb{R}_{+}^{m+q} | y \leq \sum_{i=1}^{n} \gamma_i y_i,\ x \geq \sum_{i=1}^{n} \gamma_i x_i,\ \gamma_i \geq 0\,\forall i = 1, ..., n, \sum_{i=1}^{n} \gamma_i = 1\right\}. \tag{1}$$

For a given firm with input - output vector (x, y) Farrell's input efficiency is given by by the Shephard input distance function defined as follows:

$$\delta(x, y) \equiv \sup\left\{\lambda | \lambda^{-1} x \in \mathcal{X}(y)\right\}, \tag{2}$$

where $\mathcal{X}(y) = \left\{x \in \mathbb{R}_{+}^{m} | (x, y) \in \mathcal{T}\right\}$ is the input requirement set to produce a given y. Equivalently, we have the following linear programming problem due to Charnes et al. (1978):

$$\tilde{\delta}(x, y)^{-1} = \min\left\{\alpha > 0 | y \leq \sum_{i=1}^{n} \gamma_i y_i, \alpha x \geq \sum_{i=1}^{n} \gamma_i x_i, \sum_{i=1}^{n} \gamma_i = 1, \gamma_i \geq 0\,\forall i = 1, ..., n\right\}. \tag{3}$$

Moreover, $\tilde{\delta}(x, y) \geq 1,\ \forall x \in \mathcal{X}(y)$. If $\tilde{\delta}(x, y) = 1$ then $(x, y) \in \partial \mathcal{X}(y)$ and the point (x, y) is input-efficient (the boundary of a set is denoted by ∂). In this case we seek the maximum contraction in inputs that still produces a feasible input-output combination. The output-oriented case can be treated in a similar way. Of course, only rarely if ever we are interested in obtaining efficiency for a single input - output vector (x, y). Usually, we are interested in obtaining efficiency scores for all DMUs in a data set. In this case we have to solve a linear programming problem for each DMU. This problem is the following:

$$\begin{gathered} \min\nolimits_{\alpha,\gamma} \alpha \\ \text{subject to:} \\ y_o \leq \textstyle\sum_{i=1}^{n} \gamma_i y_i, \\ \alpha x_o \geq \textstyle\sum_{i=1}^{n} \gamma_i x_i, \\ \textstyle\sum_{i=1}^{n} \gamma_i = 1, \gamma_i \geq 0\,\forall i = 1, ..., n, \end{gathered} \tag{4}$$

for every $o \in \{1, ..., n\}$. For large data sets, solving a linear programming problem n times is computationally intensive and time consuming. For this reason, empirical applications of DEA have been limited to data sets with a relatively manageable number of DMUs. An additional reason for doing so is that, for purposes of statistical inference, the bootstrap is often applied and, therefore, the computational burden increases. In the next section, we provide a Bayesian formulation of the problem where it is not necessary to solve linear programming problems at all.

3 Bayesian analysis

3.1 Motivation

A very general result has been presented by Pincus (1970) who showed that any optimization problem can be considered equivalently from the Bayesian perspective for an appropriately defined posterior. For example, suppose we seek: $x^* = \arg\min_{x \in \mathcal{X}} f(x)$, where $f : \mathcal{X} \subseteq \Re^d \to \Re$. Pincus (1970) showed that one can use the following posterior to recover the solution of the problem as the posterior mean: $p_h(x) \propto \frac{\exp\{-hf(x)\}}{\int_{\mathcal{X}} \exp\{-hf(z)\}dz}$ where h is a constant that tends to infinity, provided that the integral in the denominator converges. This is always guaranteed if $\mathcal{X}$ is compact. In that case we have $\lim_{h\to\infty} \int_{\mathcal{X}} xp_h(x)dx = x^*$. Pincus (1970) has also proposed the use of a Metropolis algorithm to set up a Markov chain to produce draws $\left\{x^{(s)}, s = 1, ..., S\right\}$ which converge to the posterior distribution $p_h(x)$ (as $S \to \infty$). Therefore, the solution can be estimated accurately as: $x^* = S^{-1} \sum_{s=1}^{S} x^{(s)}$ for a given large value of h.

In more familiar terms, suppose we have a linear regression model: $y_i = x_i'\beta + e_i \, \forall i = 1, ..., n$, where $x_i \in \mathbb{R}^d$ is a vector of regressors, y_i is a dependent variable, $\beta \in \mathbb{R}^d$ is a vector of regression parameters and e_i is an error term. Often, the parameter vector β is estimated using the method of least squares (LS), viz.: $\min_\beta \sum_{i=1}^n (y_i - x_i'\beta)^2$ or least absolute deviations (LAD): $\min_\beta \sum_{i=1}^n |y_i - x_i'\beta|$. Sometimes the objective function is: $\min_\beta \max_{i=1,...,n} |y_i - x_i'\beta|$ also known as minimax or L_∞-regression.

The estimators of β are identical with the maximum likelihood estimator under the assumption of normality for e_i in the first case or the assumption of a Laplace distribution in the second case[3]. For example, in the first case, the Pincus (1970) likelihood would be: $\mathscr{L}(\beta, h) \propto \exp\left\{-h \cdot \sum_{i=1}^n (y_i - x_i'\beta)^2\right\}$, while in the second case it would be: $\mathscr{L}(\beta, h) \propto \exp\left\{-h \cdot \sum_{i=1}^n |y_i - x_i'\beta|\right\}$. Clearly, it is not necessary to let h diverge to infinity and we can consider it as a parameter. However, in the first case we must include a term $h^{n/2}$ (apart from an irrelevant constant not depending on β or h) which comes from the integrating constant while in the second case the relevant term is h^n. All three objective functions are special cases of: $\min_\beta \left\{\sum_{i=1}^n |y_i - x_i'\beta|^q\right\}^{1/q}$, where $q \geq 1$ is a constant. These estimators minimize the L_q-norm of the residuals and is also known as L_q-regression. Minimax regression is equivalent to $q = \infty$ while for $q = 1$ and $q = 2$ we obtain the LAD and LS cases, respectively. From Box and Tiao (1973, p. 157) regression in the L_q-norm is equivalent to ML when the errors have the density $f(e) = c(h) \cdot \exp\{-h \cdot |e|^q\}$, where $c(h) = h^q$. Therefore, for L_∞-regression, the integrating constant in the likelihood is h^{nq}.

[3] The standard forms of the densities are: $f_{\text{Normal}}(e) \propto \exp\left\{-z^2/2\right\}$ and $f_{\text{Laplace}}(e) \propto \exp\left\{-|z|\right\}$.

More generally, when the objective function is of the form: $\min_\beta \sum_{i=1}^n \varrho\left(y_i - x_i'\beta\right)$, for a certain "penalty" function $\varrho(\cdot)$, the likelihood consistent with the problem is: $\mathscr{L}(\beta,h) = c(h)\cdot\exp\left\{-h\cdot\sum_{i=1}^n \varrho\left(y_i - x_i'\beta\right)\right\}$ where $c(h)^{-1} = \int_{-\infty}^{\infty}\exp\left\{-h\cdot\varrho(e)\right\}de$, provided that the integral converges. This technique offers a principled way to address the problem of finding a posterior distribution *coherent* with DEA. By "coherent" we mean that properties of the function $f(x)$ to be optimized are *explicitly* incorporated into the Bayesian posterior. On another matter, we can treat h as a parameter (with a proper or improper prior, $p(h)$) and let the data speak for themselves regarding the joint distribution: $p(x,h) \propto p_h(x)p(h)$. In this manner we also obtain a marginal posterior distribution for parameter h which, effectively, *allows for statistical uncertainty in the data.*

3.2 Data Generating Process and Likelihood / Posterior

In this section we provide a Bayesian interpretation of DEA, that is we provide a posterior distribution consistent or coherent with DEA. First, we define the parameter vector:

$$\theta_i = [\alpha_i, \boldsymbol{\gamma}_i]\ \forall i = 1,...,n,\ \theta = [\{\theta_i,\, i = 1,...,n\}], \tag{5}$$

where h is not unlike the Pincus constant and $\boldsymbol{\gamma}_i = [\gamma_{i1},...,\gamma_{in}]$. Notice that γ_{ij} is the weight for unit j when unit i is being evaluated $(i,j = 1,...,n)$.[4]

The restrictions that must be imposed *a priori* are:

$$\alpha_i > 0, \boldsymbol{\gamma}_i \in \mathbb{R}_+^n, \sum_{j=1}^n \gamma_{ij} = 1\ \forall i = 1,...,n. \tag{6}$$

Given the data, we have the additional restrictions:

$$y_i \le \sum_{j=1}^n \gamma_{ij}y_j,\ \alpha_i x_i \ge \sum_{j=1}^n \gamma_{ij}x_j\ \forall i = 1,...,n. \tag{7}$$

Notice that $\boldsymbol{\gamma}_i = [\gamma_{i1},...,\gamma_{ip}]$, is allowed to be different for each observation as the solution of (3) depends on the

[4] One may argue that γs are sample statistics which are not of particular interest, unlike Farrell inefficiencies. However, the intensity variables (γs) are also unknown in DEA and they are always obtained from (that is given) the data, So, there is no fundamental distinction between Farrell efficiencies and γs. Since, in Bayesian analysis, all inference is conditional on the given data and no alternative data sets are considered, as in the frequentist or sampling-theory approach, there can be no distinction between population and "sampling parameters". However, the fact remains that when the DEA problem is formulated, the γs are unknown so, they must be treated as parameters. Moreover, on the issue of statistical coherence, it is known that coherence depends on whether all evidence (data and prior) are used in a way that "makes sense". Effectively, this means that all inferences must depend on the likelihood and the prior, i.e. via the posterior obtained using Bayes' theorem.

point (x, y). In turn, we can define the following posterior density:

$$\begin{gathered} p(\theta, h|\mathcal{D}) \propto h^{n-1} \exp\left(-h \textstyle\sum_{i=1}^{n} \alpha_i\right) \cdot p(\theta), \text{ with:} \\ \alpha_i > 0, \boldsymbol{\gamma}_i \in \mathbb{R}_{+}^{m}, \textstyle\sum_{j=1}^{n} \gamma_{ij} = 1 \; \forall i = 1, ..., n, \\ y_i \leq \textstyle\sum_{j=1}^{n} \gamma_{ij} y_j, \; \alpha_i x_i \geq \textstyle\sum_{j=1}^{n} \gamma_{ij} x_j \; \forall i = 1, ..., n, \end{gathered} \tag{8}$$

where $\mathcal{D} = \{(x_i, y_i) \, \forall i = 1, ..., n\}$ is the data, $p(\theta)$ is a prior density on $\boldsymbol{\alpha} = [\alpha_1, ..., \alpha_n]'$ and $\boldsymbol{\gamma} = [\boldsymbol{\gamma}_i, i = 1, ..., n] \in \mathbb{R}^{n^2}$. The use of the term $\exp\{-h \sum_{i=1}^{n} \alpha_i\}$ for given $h > 0$ comes from Pincus (1970), as in (3) we want to minimize with respect to α_i. The use of the restrictions is due to (6) and (7). Finally, the term h^{n-1} comes from the integrating constant of the exponential distribution: $p(\alpha) = he^{-h\alpha}$, $\alpha \geq 0$ when the prior of h has the agnostic or Jeffreys form: $p(h) \propto h^{-1}$.[5] Integrating with respect to h using properties of the *gamma* distribution[6] yields the problem:

$$\begin{gathered} p(\theta|\mathcal{D}) \propto \left(\textstyle\sum_{i=1}^{n} \alpha_i\right)^{-n} \cdot p(\theta), \text{ with:} \\ \alpha_i > 0, \boldsymbol{\gamma}_i \in \mathbb{R}_{+}^{m}, \textstyle\sum_{j=1}^{n} \gamma_{ij} = 1 \; \forall i = 1, ..., n, \\ y_i \leq \textstyle\sum_{j=1}^{n} \gamma_{ij} y_j, \; \alpha_i x_i \geq \textstyle\sum_{j=1}^{n} \gamma_{ij} x_j \; \forall i = 1, ..., n. \end{gathered} \tag{9}$$

We denote the space of the restrictions in (8) or (9) by Θ. The dimensionality of the parameter vector is $d = n(n+1)$.

The specification of the prior $p(\theta)$ can be performed by assuming that:

$$\theta \text{ is i) uniform over the space } \Theta. \tag{10}$$

or

$$\text{ii) } \theta \sim \mathcal{N}(\mu_\theta, \Sigma_\theta) \text{ truncated in the space } \Theta \text{ of the restrictions, where } \mu_\theta \text{ and } \Sigma_\theta \text{ are unknown.} \tag{11}$$

Although (8) is complicated, Markov Chain Monte Carlo (MCMC) methods can be used to explore it and perform Bayesian inference. These methods require the conditional posterior distributions $p(\alpha_i|\boldsymbol{\gamma}, \mathcal{D})$ and $p(\boldsymbol{\gamma}|\boldsymbol{\alpha}, \mathcal{D})$.

Suppose first that θ is uniform over the space Θ. The conditional posterior distribution of α_i is given by:

$$p(\alpha_i|\boldsymbol{\gamma}, \mathcal{D}) \propto \left(\sum_{i=1}^{n} \alpha_i\right)^{-n} \cdot \mathbf{1}\left(\alpha_i > 0, \alpha_i x_i \geq \sum_{j=1}^{n} \gamma_{ij} x_j\right) \forall i = 1, ..., n, \tag{12}$$

[5] The prior belongs to the family of *gamma* distributions for h: $f_G(h) = \frac{A^N}{\Gamma(N)} h^{N-1} \exp(-Ah)$ when $A = N = 0$, and $\Gamma(N)$ is the *gamma* function. It is, of course, improper as its integral does not converge. However, it is associated with the useful notion that $\ln h$ is flat in the entire real line thus representing, in a convincing way, the fact that we do not have any prior knowledge about h whatsoever. Other priors for h can be constructed by varying the parameters A and N. For example, a proper but uninformative prior would have $N = 1$ and A equal to a small positive number, say 10^{-3}.

[6] The shape parameter is n and the scale parameter is $\sum_{i=1}^{n} \alpha_i$.

where $\mathbf{1}(\cdot)$ is the indicator function. This distribution is defined over the set of the m constraints: $\alpha_i x_i \geq \sum_{j=1}^{n} \gamma_{ij} x_j$. Next, we consider the conditional posterior distribution of $\boldsymbol{\gamma}_i$ which is given by:

$$p(\gamma_i|\boldsymbol{\alpha}, \boldsymbol{\gamma}_{-i}, \mathcal{D}) \propto \mathbf{1}\left(\boldsymbol{\gamma}_i \in \mathbb{R}^n_+, \sum_{j=1}^{n} \gamma_{ij} = 1, y_i \leq \sum_{j=1}^{n} \gamma_{ij} y_j,\ \alpha_i x_i \geq \sum_{j=1}^{n} \gamma_{ij} x_j\right) \forall i = 1, ..., n, \tag{13}$$

where $\boldsymbol{\gamma}_{-i}$ denotes all elements of $\boldsymbol{\gamma}$ except the i$^{\text{th}}$ element. Each$\boldsymbol{\gamma}_i$ must also satisfy the restrictions given by:

$$\gamma_{ij} \geq 0\, \forall i, j = 1, ..., n, \sum_{j=1}^{n} \gamma_{ij} = 1, \sum_{j=1}^{n} \gamma_{ij} y_j \geq y_i, \sum_{j=1}^{n} \gamma_{ij} x_j \leq \alpha_i x_i\, \forall i = 1, ..., n. \tag{14}$$

Therefore, to draw from the conditional posterior distribution of γ_i in (13), we have to draw uniform random numbers from the set defined by (14). To draw from (12) and (13) we use a Hit-and-Run algorithm. Suppose $\chi \in \mathcal{S} \subset \mathbb{R}^d$ and we want to draw a sample from a multivariate distribution with density $f(\chi)$, known up to a normalizing constant, where $\mathcal{S}$ is an arbitrary set. Using the Hit-and-Run algorithm (Smith, 1984, see also Belisle et al., 1993) we proceed as follows:[7]

1. Given χ_t:

2. Generate a direction $\mathscr{D}_{t+1}$ and a point: $\chi_{t+1} = \chi_t + \omega_t \mathscr{D}_{t+1}$,where ω_t has density: $g(\omega) \propto f(\chi_t + \omega \mathscr{D}_{t+1})$ and $\chi_{t+1} \in \mathcal{S}$.

When f is uniform the hit-and-run algorithm is simpler as ω is generated from a uniform distribution. In more general cases the algorithm works for arbitrary distributions such as the exponential in (12) or the normal case for $p(\theta)$. The sample $\{x_t,\, \forall t = 1, ..., S\}$, converges to the distribution whose density is f.

When θ has a normal prior, $\theta \sim N(\mu_\theta, \Sigma_\theta)$, we assume that $\theta_i \equiv [\alpha_i, \boldsymbol{\gamma}_i']' \sim i.i.d.\mathcal{N}_{q+1}(\mu_\theta, \Sigma_\theta)$, for all $i = 1, ..., n$. For the parameters we assume a benchmark flat prior:

$$p(\mu_\theta, \Sigma_\theta) \propto |\Sigma_\theta|^{-(d+1)/2}. \tag{15}$$

The conditional posteriors of α_i and $\boldsymbol{\gamma}_i$ are now normal with means the relevant elements of μ_θ and Σ_θ subject to the same restrictions as in (12) and (13). Application of the Hit-and-Run algorithm is, again, simple in this case. The conditional posterior distributions of μ and Σ are standard once we have simulated values $\theta_i^{(s)}$ from the posterior (where

[7] It is possible to use other algorithms. Prominent among them is, of course, the Metropolis-Hastings algorithm in its various formulations (for a useful discussion, see Chib and Greenberg, 1995). Another alternative is the slice sampler (Neal, 2003 and Bishop, 2006). The performance of these algorithms is an interesting topic for future research although we may guess that to satisfy the restrictions, a large number of candidate draws may be required.

$s = 1, ..., S$ indicates the draw index and S is the number of draws). These conditional posteriors are as follows:

$$\begin{aligned} \mu_\theta | \theta, \mathcal{D} &\sim \mathcal{N}_d\left(\hat{\mu}_\theta,\, n^{-1}\Sigma_\theta\right), \\ \Sigma_\theta &\sim \mathcal{W}\left(\nu, \mathbf{S}\right), \end{aligned} \tag{16}$$

where $\hat{\mu}_\theta = S^{-1}\sum_{s=1}^{S}\theta_i^{(s)}$, $\nu = n + 1$, $\mathbf{S} = S^{-1}\sum_{s=1}^{S}\left(\theta_i^{(s)} - \hat{\mu}\right)\left(\theta_i^{(s)} - \hat{\mu}\right)'$ and $\mathcal{W}$ denotes the Wishart distribution. Given draws from the posterior conditional distributions the procedure is repeated S times to generate a MCMC sample of parameter draws that converges in distribution to (9). **It should be noticed that, in the posterior, the data is considered given and, therefore, there is no need to generate alternative data sets that might have been observed but were not actually observed**. Therefore, since the data is given, statistical uncertainty arises from uncertainty with respect to the parameters of the DEA problem.[8]

In what follows, we use the prior in (10). Starting values for MCMC were obtained from a standard implementation of DEA although convergence of MCMC does not depend on initial conditions. The number of iterations and the length of the burn-in phase in MCMC are described in the context. It is known that under quite general conditions, convergence of MCMC to the DEA posterior is guaranteed (Roberts and Smith, 1994). In the next section, we provide Monte Carlo simulation evidence on the performance of the new techniques before embarking on an actual empirical application as in section 5.

4 Monte Carlo evidence

The use of Monte Carlo simulation is necessary in our context to validate the proposed data generating process and the proposed likelihood / posterior in the context of artificial data. In such synthetic data we know inefficiency in advance which can be compared with results obtained from the novel Bayesian DEA proposed here. For example, the Bayesian posterior mean converges to population means as the sample gets larger under very general conditions (Shen and Wasserman, 2001). The similarity between the bootstrap and Bayesian analysis has been pointed out before by Efron (2012) which helps to understand why bootstrap-based and Bayes-based DEA should deliver comparable results. Liang et al. (2016) provide further connections and show how MCMC and the bootstrap can work together in the analysis of big data. In the context of DEA, the bootstrap is computationally expensive and cannot be applied in huge data sets although it can be applied more easily in subsets of the data and parallel computing without too much effort. Therefore, in relatively large samples, inefficiency estimates should be close to population inefficiencies. The purpose of Monte Carlo simulation is to i) validate this result in the context of DEA and perhaps more importantly to ii) examine how large the sample must be, depending on the dimensionality of the DEA problem, so that inefficiency estimates become approximately consistent.

[8] As Friedman et al. (2009, p. 272) put it: "[. . .] [T]he bootstrap distribution represents an (approximate) nonparametric, noninformative posterior distribution for our parameter. But this bootstrap distribution is obtained painlessly — without having to formally specify a prior and without having to sample from the posterior distribution. Hence we might think of the bootstrap distribution as a "poor man's" Bayes posterior. By perturbing the data, the bootstrap approximates the Bayesian effect of perturbing the parameters, and is typically much simpler to carry out."

4.1 Monte Carlo experiment 1

In this section we report computational experience and results from a Monte Carlo experiment. We have $p = 3$ inputs, x_{i1}, x_{i2} and x_{i3} and a single output. Output y_i and x_{i2} and x_{i3} are generated as random uniform deviates in the interval $[1, 10]$. We have: $y_i = x_{i1}^{\alpha} x_{i2}^{\beta} x_{i3}^{1-\alpha-\beta} \exp(-u_i)$ where $u_i \sim \text{iid}\, |\mathscr{N}(0, \sigma^2)|$, for each $i = 1, ..., n$. The u_is represent true technical inefficiency scores on which our main interest focuses in this study. Solving in terms of the first input we generate: $x_{i1} = y_i x_{i2}^{-\beta/\alpha} x_{i3}^{(\alpha+\beta-1)/\alpha} \exp(u_i/\alpha)$.

We consider various values of α, β, σ and the sample size n. We use 120,000 iterations the first 20,000 of which are discarded to mitigate possible start up effects.[9] The results are summarized in Tables 1 and 2. Overall, MCMC is quite useful in the sense that biases are negligible even in small samples although negatively biased for very small samples (up to 20). From the results in Table 2, The autocorrelation inherent in MCMC is moderate at lag 50 and the frequency of Geweke's (1992) convergence t-statistic diagnostics in excess of 1.96 in absolute values decreases with the sample size. Relative numerical efficiency (RNE) is high, reaching almost 45% in some cases, relative to *i.i.d* sampling from the posterior if that were possible at all (in which case RNE would have been equal to one).

4.2 Monte Carlo experiment 2

In the second Monte Carlo experiment we take up the case of three inputs and two outputs. We have $p = 3$ inputs, y_{i1}, y_{i2} are generated as random uniform deviates in the interval $[1, 10]$. The three inputs are generated using a transformation function as follows:

$$x_{i1}^{\beta} x_{i2}^{\gamma} x_{i3}^{\delta} = y_{i1}^{\alpha} y_{i2}^{1-\alpha} \exp(v_i - u_i). \tag{17}$$

We fix $\alpha = \frac{1}{3}$ and $\beta = \gamma = \delta = \frac{1}{3}$. Inputs x_{i2} and x_{i3} are generated from a log-linear reduced form:

$$\log x_{i2} = \pi_{11} \log y_{i1} + \pi_{12} \log y_{i2} + \xi_{i1},$$

$$\log x_{i3} = \pi_{21} \log y_{i1} + \pi_{22} \log y_{i2} + \xi_{i2},$$

and x_{i1} is generated using (17). The new element, besides the fact that we have multiple outputs, is that we also introduce noise, $v_i \sim iid\mathcal{N}(0, \sigma_v^2)$. For the reduced form $\xi_{i1}, \xi_{i2} \sim iid\mathcal{N}(0,\, 0.2^2)$ and $\pi_{ij} = \frac{1}{4}$. We have varied the parameters π_{ij} without noticing any effect on the qualitative behavior of our results. We denote $\tilde{\lambda} = \sigma/\sigma_v$, and we keep fixed $\sigma_v = 0.15$. From the results it turns out that the behavior of MCMC schemes is acceptable and performs well in

[9] Several random number generators have been tried and the results remained virtually the same. We use the random number generators G05SQF and G05DDF available in NAG library. Additionally we have tried ZUFALL from NETLIB using randomly generated integer seeds. For documentation, see `http://www.nag.com/numeric/fl/nagdoc_fl24/pdf/G05/g05conts.pdf`. It is important to emphasize that autocorrelation in MCMC draws is inherent in the procedure and is not due to the quality of the random number generator sequence. That is, MCMC delivers a sample that converges in distribution to the posterior but the sample is not i.i.d as the draws are dependent. The amount of dependence can be measured using autocorrelation functions for draws of individual parameters as well as relative numerical efficiency (RNE), see Geweke (1992).

terms of RNE, convergence, as well as autocorrelation at lags 10 and 50. Despite the presence of statistical noise, in samples of reasonable size the Bayesian procedures deliver results which are essentially unbiased in samples of size in excess of 20 or 50 and signal-to-noise ratio between $\tilde{\lambda} = \frac{1}{2}$ and $\tilde{\lambda} = 2$. Autocorrelation is not destructively large: With autocorrelation coefficients around 0.2 at lag 10, MCMC can explore the posterior in a thorough way. At lag 50 the autocorrelation coefficients are much lower, RNEs are high and exploration of the posterior in (9) is thorough.

4.3 Prior sensitivity analysis

Although the prior in (10) is non-informative it is clearly of interest to examine how sensitive are the results from small to moderate or even large deviations from the baseline prior. Although implementation of prior sensitivity analysis in the context of Monte Carlo simulation with alternative data sets is computationally expensive, we believe it is important to examine it for the following reason. For a particular data set, the prior assumptions may or may not make a significant difference in terms of final results. Examining prior robustness in terms of Monte Carlo with alternative data sets, provides a more or less "final" answer as to whether deviations from baseline prior assumptions affect the results in essential ways. If they do, then it is an empirical matter to examine prior sensitivity in the context of a given empirical application. If not, then we can be more or less certain that, in actual applications, the prior does *not* have an excessive effect on the results.

The more general form of (10) is the following, where we adopt informative priors for both μ and Σ, keeping the flat prior on λ which is likely to be found useful to most applied researchers:

$$\begin{array}{c} p(\theta) \propto \left(\sum_{i=1}^n \alpha_i\right)^{-n} \cdot \prod_{i=1}^n \left\{ \mathbf{1}(\alpha_i > 0) \cdot p(\gamma_i) \cdot \mathbf{1}(\gamma_i \in \mathbb{R}^n_+,\ \sum_{j=1}^n \gamma_{ij} = 1) \right\} \\ \cdot f_W\left(\Sigma_\theta; \underline{\nu}, \underline{A}\right) \cdot f_N\left(\mu_\theta; \underline{\mu}, \underline{V}\right), \end{array} \tag{18}$$

where and $f_N\left(\mu; \underline{\mu}, \underline{V}\right)$ denotes the multivariate normal distribution with mean vector $\underline{\mu}$ and covariance matrix $\underline{V}$, and $f_W\left(\Sigma_\theta; \underline{\nu}, \underline{A}\right)$ denotes the density of the Wishart distribution with parameters $\underline{\nu}, \underline{A}$ (Zellner, 1971, p. 395):

$$f_W\left(\Sigma_\theta^{-1}\right) \propto |\Sigma_\theta|^{-(\underline{\nu}+d+1)/2} \exp\left(-\tfrac{1}{2}\mathrm{tr}\Sigma_\theta^{-1}\underline{A}\right), \tag{19}$$

The problem is how to allow for changes in the prior that cover a wide range of beliefs, in order to perform sensitivity analysis of the inefficiency estimates. We proceed as follows:

(i) Alternative values of each element of $\underline{\mu}$ are allowed in the interval $[-100,\ 100]$.

(ii) $\underline{V} = \kappa I$ and $A = gI$, where the constants κ and g vary between 0.01 and 100 (I denotes the identity matrix).

(iii) Parameters a and $\underline{\nu}$ vary in the interval 0.01 to 100.

We generate simultaneously 1,000 different parameters from (i) through (iii) using uniform distributions. For each generated set of parameters, a new prior is formulated as in (18). Based on the new prior we perform MCMC in the context of the Monte Carlo experiment in section 4.2. Efficiencies are estimated using the posterior means of δ_i^{-1}s in (3). The posterior means are recorded and then they are compared with the inefficiency estimates delivered by the baseline prior in (10). The results are reported in Table 3.

In addition we examine sensitivity of our results under (11) where the parameter vector follows a general multivariate normal distribution $\mathcal{N}(\mu, \Sigma)$. This is important as the uniform specification in (10) is only one of many possible specifications. From these results the impact of the prior is not excessively large and, indeed, it can be ignored for reasonable sample sizes and signal-to-noise ratio up to 2. As the signal-to-noise increases the impact of the prior is smaller **These results are reassuring in the sense that prior assumptions about fundamental parameters of the problem do not lead to substantially different efficiency scores.** Other statistics (like rank correlation coefficients between the 1,000 alternative priors and the baseline prior) have been computed and are available upon request but are not reported here in the interest of space. From the results of Table 3, efficiencies are very close to the specification in (10) when we use (11) instead, for different values of $\tilde{\lambda}$.

5 Application

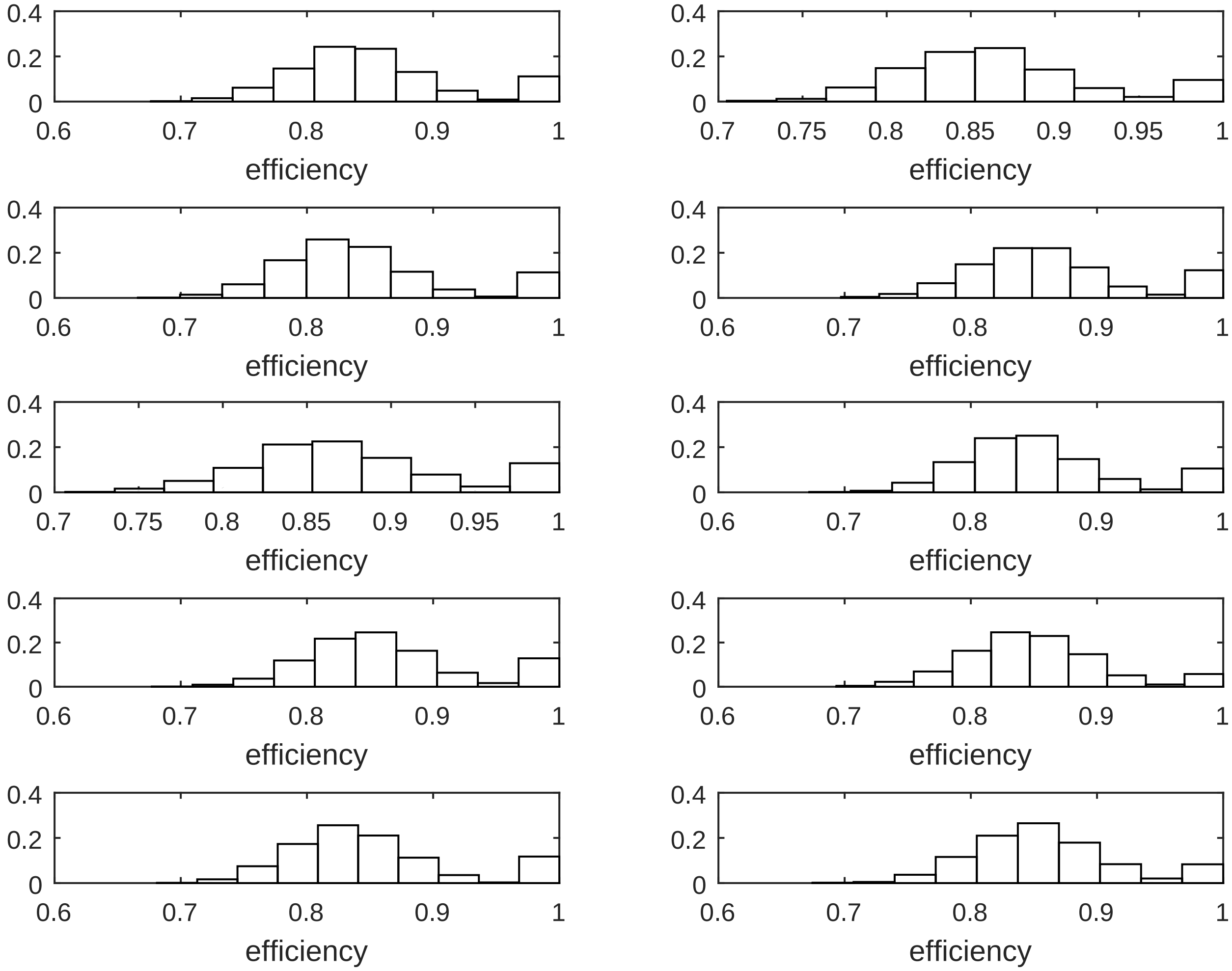
We provide an empirical application to a data set previously analysed by Malikov, Kumbhakar and Tsionas (2015). The sample is an unbalanced panel of US banks with 2,397 bank–year observations for 285 banks. We have the following outputs: consumer loans (y_1), real estate loans (y_2), commercial and industrial loans (y_3) and securities (y_4). The inputs are labour (number of full-time equivalent employees, x_1), physical capital (x_2), purchased funds (x_3), interest-bearing transaction accounts (x_4) and non-transaction accounts (x_5). We use 150,000 iterations the first 50,000 of which are discarded to mitigate possible start-up effects. Convergence was diagnosed successfully using Geweke's (1992) t-statistics for the θ_is. In this application, the MCMC scheme converged to our satisfaction since the fraction of t-statistics in excess of 1.96 was less than 2%.

Our main objective is to estimate efficiency scores and subsequently provide statistical inferences.[10] With a large number of decision-making-units this can be done in a variety of ways. In Figures 1 through 4 we report posterior histograms from MCMC for various groups of DMUs: Fully efficient scores according to DEA (Figure 1), between 90% and 100% (Figure 2), between 70% and 90% (Figure 3) and, finally, between 50% and 70% in Figure 4. We limit attention to ten banks only since results for other banks were quite similar.[11] These posteriors can provide an aid to appreciating the range of posterior uncertainty involved in DEA. For fully efficient DMUs according to DEA scores, the posteriors have

[10] Of course, statistical inference subsumes much more. For example, formal hypothesis testing and other types of inference can be easily performed once we have access to the posterior via numerical means, such as MCMC. Moreover, Bayes factors can be used based on numerical approximations to the marginal or integrated likelihood. In this paper, we abstract from exploration of such and related issues as we are mainly interested in marginal posterior distributions of efficiency scores.

[11] Posterior histograms were obtained for all banks and then ten from each group were taken randomly. The prior of λ_is was the uniform in (10).

Figure 1: Posteriors of efficiency scores for DMUs with DEA scores = 1



of course considerable point mass at unity but they also contain significant mass between roughly 80% and 95% or 90% depending on the case. From Figure 3, DEA scores in excess of 90% are still compatible with full efficiency. Ranking is, of course, formally possible based on these results thus solving a long-standing problem in the relevant literature (viz. Andersen and Petersen, 1993).

Lower scores are not compatible with 100% efficiency as evidenced in Figures 4 and 5 although some units can be 100% efficient even though their DEA scores are only between 70% and 90% (Figure 3). Only a few could be considered as having efficiency greater than about 70% conditional on DEA scores between 50% and 70% (Figure 4) but the range of their efficiency scores, depending on the case, is roughly between 40% and 75%. Overall, it is evident that DEA does a good job of efficiency estimation in the intermediate cases but it does not always do so well with the boundary cases. Since statistical uncertainty is involved it is not expected to do so even in the intermediate cases and *this statistical uncertainty must be taken into account seriously* when performing efficiency measurement.[12]

[12] For the empirical application relative numerical efficiency (RNE) was 47% on the average. Median autocorrelation at lag 50 for efficiency scores was 0.13.

Figure 2: Posteriors of efficiency scores for DMUs with DEA scores between 0.9 and <1

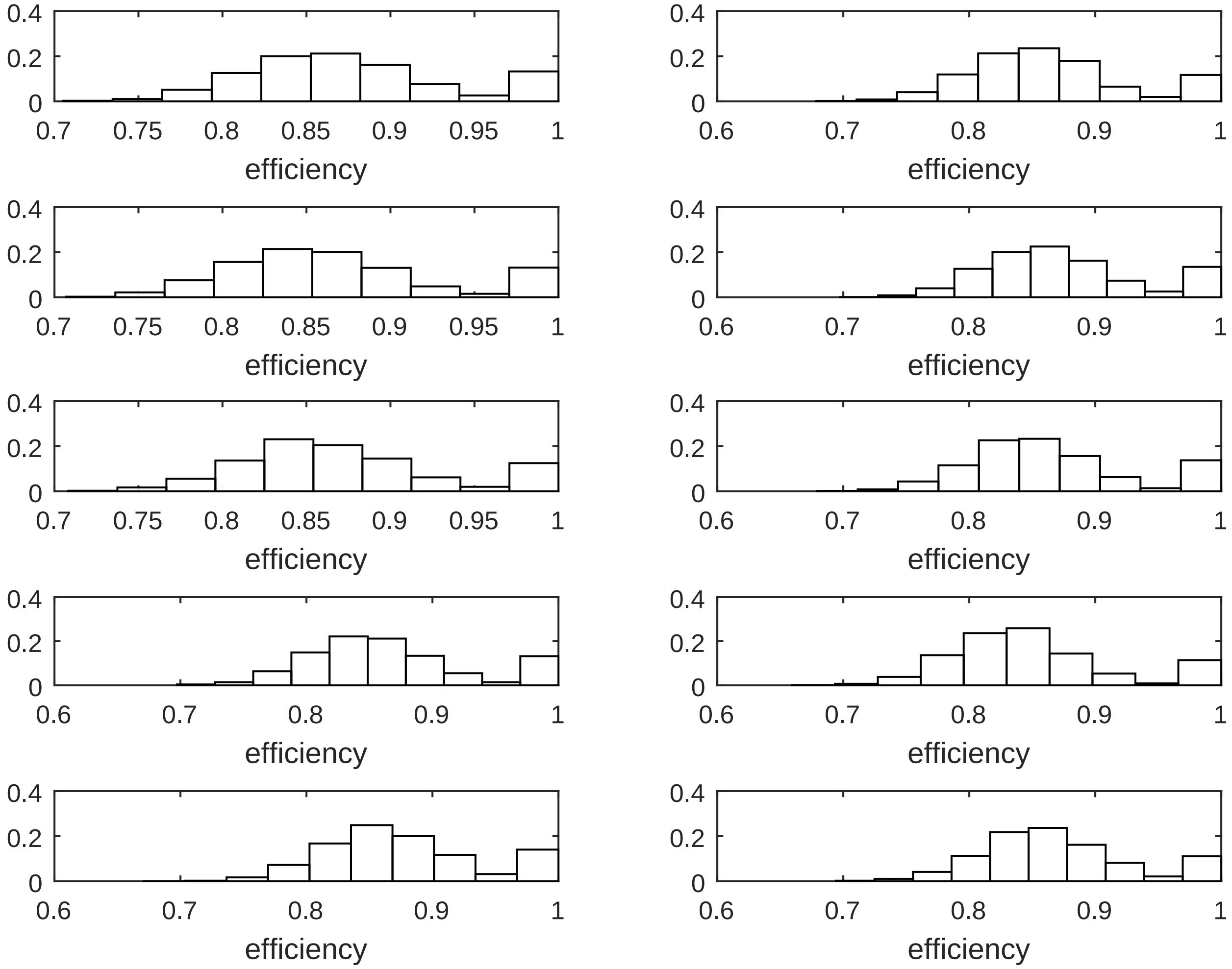


Figure 3: Posteriors of efficiency scores for DMUs with DEA scores between 0.7 and 0.9

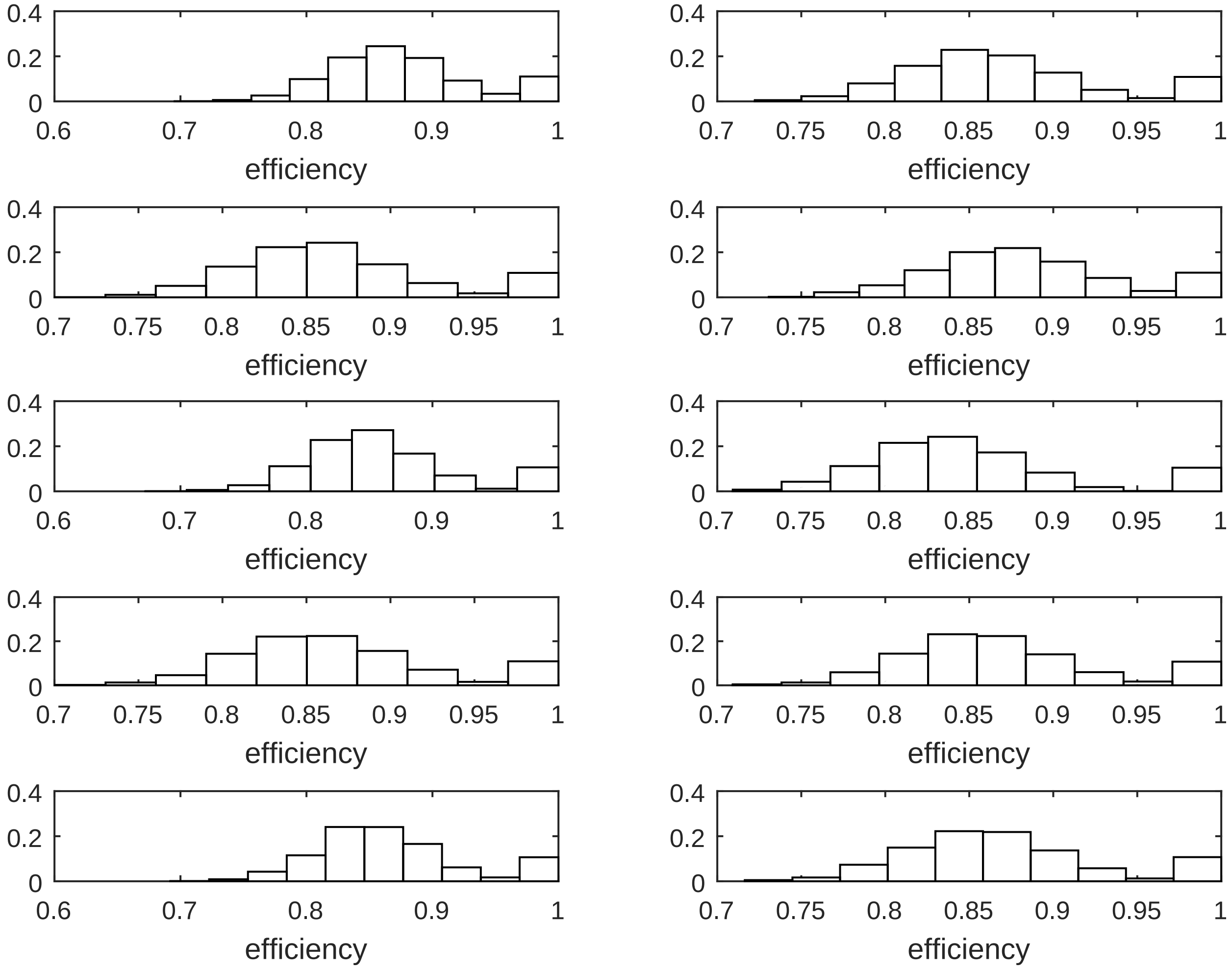


Figure 4: Posteriors of efficiency scores for DMUs with DEA score between 0.5 and 0.7

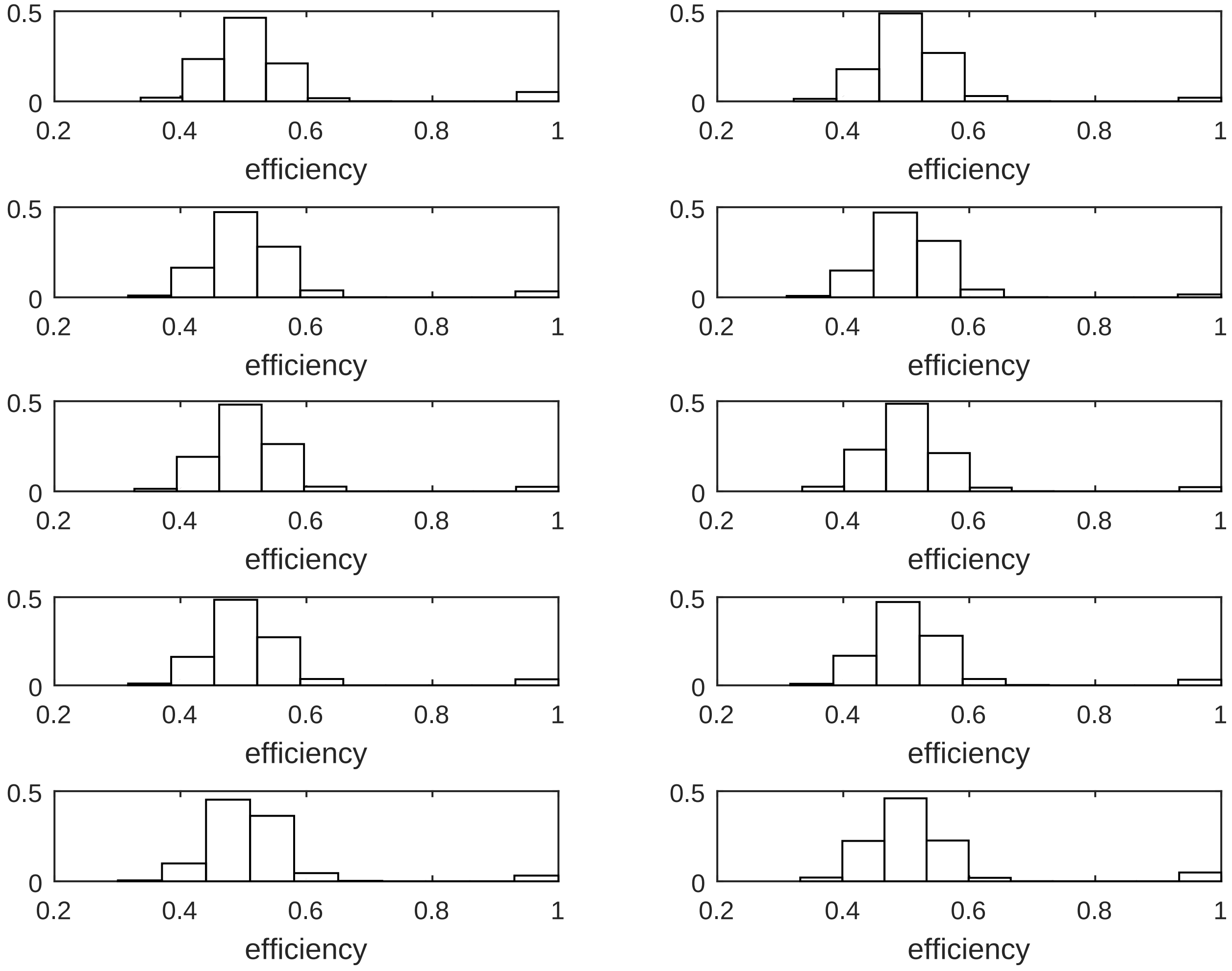
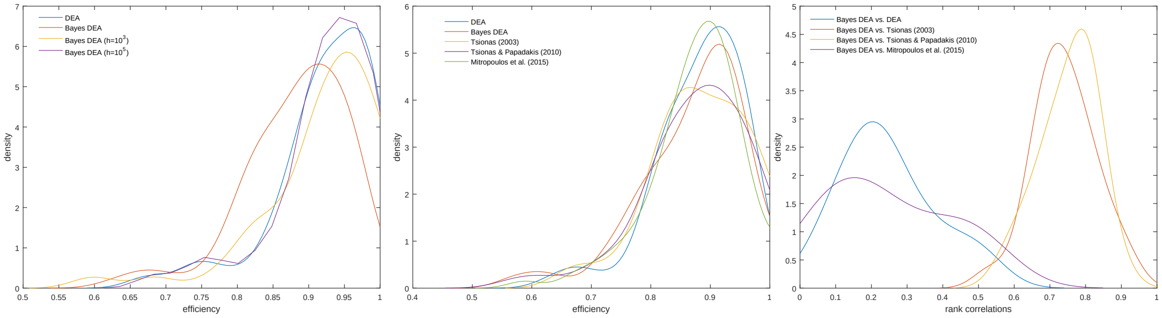


Figure 5: Densities of efficiency scores



Next, we take up a comparison with DEA. Since the sample is quite large we need specialized solvers of the linear programming problem. We have used the HOPDM algorithm in Gondzio (1995) and Gondzio and Kouwenberg (2006). In Figure 5 (upper left panel) we present kernel densities of efficiency scores for DEA, Bayes DEA (when h is treated as a parameter) and two Bayes DEA approached when h is fixed at 10^3 and 10^5. Interestingly, with a fixed h, as h increases, the results are quite similar to DEA as expected from Pincus' (1970) result on optimization. Relative to DEA, Bayes DEA provides different efficiency scores and their sample distribution has significantly less probability concentration around unity. This indicates that DEA will classify some banks as efficient when, in fact, this is not so once we allow for statistical uncertainty.

We focus on comparing with other Bayesian techniques, namely Tsionas (2003), Tsionas and Papadakis (2010) and Mitropoulos et al. (2015). These approaches are Bayesian in nature but rest upon different foundations. Tsionas (2003) uses DEA inefficiency estimates to craft a prior that is, in turn, used in stochastic frontier models to obtain more precise estimates. Tsionas and Papadakis (2010) focus on a Bayesian interpretation of chance-constraint DEA. Finally, Mitropoulos et al. (2015) assume that the data generating process is a multivariate normal distribution for inputs and outputs. The mean vector and the covariance matrix of this process are estimated using a conjugate prior and, in turn, samples are drawn from the predictive distribution, DEA is applied, and inefficiency posterior means are computed. Therefore, the focus in Tsionas (2003) is on improving inefficiency estimates from stochastic frontier models, the focus in Tsionas and Papadakis (2010) is on chance-constrained DEA while the emphasis in Mitropoulos et al. (2015) is on the "straightforward" interpretation of DEA in a Bayesian context.

[13]

The new efficiency scores are compared with plain DEA, and the Bayes procedures of Tsionas (2003), Tsionas and Papadakis (2010) and Mitropoulos et al. (2015) in the middle panel of Figure 5. As the densities seem to be fairly similar we compare rank correlation coefficients of efficiency scores for each MCMC iteration. We report the density of rank correlation coefficients in the right panel of Figure 5. Correlations with DEA are rather low averaging 0.25 and ranging

[13] All computations were performed in Fortran 77 at the High End Computing Cluster (HEC) of Lancaster University. The combined facility offers over 6,500 cores, 28 TB of aggregate memory, 70TB of high performance filestore and 1.5PB of medium performance filestore. A number of nodes offer Nvidia GPU cards, which support CUDA and OpenCL applications. The cluster operating system is Scientific Linux, with job submission handled by Son of Grid Engine (SoGE). For comparison with bootstrap-DEA on a desktop computer see the paragraphs before Concluding Remarks.

from nearly zero to 0.70. Correlations with the Bayes DEA methods of Tsionas (2003) and Tsionas and Papadakis (2010) are much higher ranging from 0.50 to almost one with an average value of 0.8. This suggests that the bootstrap and the Bayesian DEA method proposed in this paper are likely to deliver results that are in broad agreement. The advantage of Bayesian DEA is that computational burden is lower as we do not have to solve repeatedly expensive linear programming problems. Rank correlations with Mitropoulos et al. (2015) are low, and not very much different from those between Bayes DEA and plain DEA.

Finally, we present CPU times of execution in Table 1. To obtain the timings we implement all computations on a desktop PC running Intel ®i9 @3.4 GHz. We use S=150,000 MCMC iterations to compare CPU timings. It is comforting that with the larger number of MCMC iterations, the results are nearly identical. CPU times are reported in Table 4.

Concluding remarks

In this paper we provide an alternative Bayesian interpretation of DEA relative to Mitropoulos et al. (2015). Our interpretation is coherent in the sense that the Bayesian posterior follows directly from the DEA optimization problem. The fundamental principle is based on a result due to Pincus (1970). The Bayesian interpretation of DEA relies on the fact that every optimization problem can be considered equivalently as a problem of finding the posterior mean in a suitably defined posterior or likelihood. Importantly, it is not necessary to solve expensive linear programming problems for each observation in the sample so expensive optimizations can be avoided. Moreover, statistical uncertainty is formally taken into account by formulating a posterior to explain the data. Bayesian inference is organized around Markov Chain Monte Carlo techniques that can be implemented quite easily with the Hit-and-Run algorithm. We conduct extensive Monte Carlo experiments to investigate the finite-sample properties of the new approach and we provide an application to a large U.S banking data set with 2,397 bank–year observations for 285 banks.

As this work comes close to the connection with Stochastic Frontier Analysis (SFA) it is perhaps worthwhile to mention some possible future extensions. First, the Bayesian approach provides another way to introduce uncertainty or a data-generating-process in DEA. In this sense, it does not appear impossible to construct priors from a set-aside sample and use them in the remaining sample. Although this approach wastes a number of observations, at the same time it provides a way to calibrate more precisely a reasonable efficiency distribution to be used in SFA.

Second, there may exist other approaches to recover a data-generating-process that is compatible with DEA and allows one-sided error terms as in SFA. This issue is under investigation and we hope to report results in the near future. Third, there is another problem that suggests itself. Flexible production frontiers (flexible in terms of functional form and distribution of the one-sided error term) may be able to approximate results from DEA. A standard bootstrap works in the first case, so a comparison in terms of efficiency scores and their distribution, is worthwhile, with an eye towards finding approximations of DEA results in large samples with a large number of inputs and outputs.

References

Andersen, P. and Petersen, N. C. (1993). Ranking efficient units in data envelopment analysis. Management Science 39 (10), 1261—1264.

Banker, R. D. (1993), Maximum likelihood, consistency and DEA: a statistical foundation, Management Science 39 (10), 1265—1273.

Banker, R.D., 1996. Hypothesis Tests Using Data Envelopment Analysis. Journal of Productivity Analysis 7, 139—159.

Belisle, C. J. P., H. E. Romeijn, and R. L. Smith (1993). Hit-and-Run algorithms for generating multivariate distributions. Mathematics of Operations Research 18 (2), 255—266.

Bishop, C. (2006). 11.4: *Slice sampling*. Pattern Recognition and Machine Learning. Springer. ISBN 0387310738.

Box, G. E. P., and G. C. Tiao (1973). *Bayesian inference in statistical analysis*. Wiley, New York.

Charles, V. Tsolas, I.E., and Gherman, T. (2018). Satisficing data envelopment analysis: a Bayesian approach for peer mining in the banking sector. Annals of Operations Research 269 (1 - 2), 81-102.

Charnes, A., Cooper, W.W., and Rhodes, E. (1978). Measuring the efficiency of decision making units. European Journal of Operational Research 2, 429–444.

Chib, S. and E. Greenberg (1995). Understanding the Metropolis-Hastings Algorithm. The American Statistician 49 (4), 327—335.

Efron, B. (2012). Bayesian inference and the parametric bootstrap. Annals of Applied Statistics 6 (4), 1971—1997.

Friedman, J., Hastie, T., & Tibshirani, R. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer Series in Statistics. Freely available at http://www-stat.stanford.edu/~tibs/ElemStatLearn/ .

Geweke, J. (1992). Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments, in J.O. Berger, et al. (eds.), *Bayesian Statistics*, Vol. 4, pp. 169-194. Oxford: Oxford University Press.

Gondzio, J. (1995). HOPDM (version 2.12) – A fast LP solver based on a primal-dual interior point method. European Journal of Operational Research 85 (1), 221–225.

Gondzio, J. and Kouwenberg, R. (2006). High Performance Computing for Asset Liability Management. Operations Research 49 (6), 879–891.

Koop, G., Steel, M. F. J. , and Osiewalski, J., (1995). Posterior analysis of stochastic frontier models using Gibbs sampling. Computational Statistics. 10, 353–373.

Griffin, J. E., and Steel, M. F. J. (2007). Bayesian stochastic frontier analysis using WinBUGS. Journal of Productivity Analysis, 27 (3), 163–176.

Liang, F., J. Kim, and Q. Song (2016). A Bootstrap Metropolis–Hastings algorithm for Bayesian analysis of big data. Technometrics 58 (3), 304—318.

Malikov, E., S.C. Kumbhakar, and M. G. Tsionas (2015). A cost system approach to the stochastic directional technology distance function with undesirable outputs: The case of US banks in 2001–2010. Journal of Applied Econometrics

31 (7), 1407—1429.

Mitropoulos P. , M. A. Talias, and I. Mitropoulos (2015). Combining stochastic DEA with Bayesian analysis to obtain statistical properties of the efficiency scores: An application to Greek public hospitals. European Journal of Operational Research 243, 302—311.

Neal, R. M. (2003). Slice Sampling. Annals of Statistics 31 (3), 705—767.

Pincus, M. (1970). A Monte Carlo method for the approximate solution of certain types of constrained optimization problems. Operations Research 18 (6), 1225—1228.

Roberts, G.O., and A. F. M. Smith (1994). Simple conditions for the convergence of the Gibbs sampler and Metropolis-Hastings algorithms. Stochastic Processes and their Applications 49 , 207—216.

Senel, T., and Cengiz, M.A. (2016). A Bayesian Approach for Evaluation of Determinants of Health System Efficiency Using Stochastic Frontier Analysis and Beta Regression. Computational and Mathematical Methods in Medicine, volume 2016, Article ID 2801081, es http://dx.doi.org/10.1155/2016/2801081.

Shen,X., and Larry Wasserman (2001). Rates of convergence of posterior distributions, Annals of Statistics 29, 687–714.

Smith, R. (1984). Efficient Monte Carlo procedures for generating points uniformly distributed over bounded regions. Operations Research 32 (6), 1296—1308.

Simar, L. and Wilson, P.W. (2000). A general methodology for bootstrapping in nonparametric frontier models. Journal of Applied Statistics 27, 779—802.

Tsionas, E. G. (2003). Combining DEA and stochastic frontier models: An empirical Bayes approach. European Journal of Operational Research 3 (16), 499—510.

Tsionas, E. G. and E. N. Papadakis (2010). A Bayesian approach to statistical inference in stochastic DEA. Omega 38, 309—314.

Zellner, A. (1971). *An introduction to Bayesian inference in econometrics*. Wiley, New York.

Table 1: Monte Carlo results, I

"Bias" is the median bias in efficiency scores from MCMC scheme. GCD provides the percentage of Geweke's (1992) t-statistic convergence diagnostic in excess of 1.96. RNE denotes relative numerical efficiency (Geweke, 1992). "acf (10)" and "acf (50)" denote median autocorrelation in θ at lags 10 and 50 respectively.

	Bias	GCD, $\|t\| > 1.96$	RNE	acf (10)	acf (50)
	$\alpha = \beta = \frac{1}{3}, \sigma = \frac{1}{10}$				
$n = 10$	-0.11	26.2%	0.21	0.54	0.13
$n = 20$	-0.07	15.3%	0.29	0.62	0.14
$n = 50$	-0.02	7.2%	0.33	0.41	0.16
$n = 100$	0.0013	3.3%	0.47	0.43	0.14
	$\alpha = \beta = \frac{1}{3}, \sigma = \frac{1}{2}$				
$n = 10$	-0.13	26.5%	0.32	0.21	0.12
$n = 20$	-0.11	16.1%	0.37	0.34	0.14
$n = 50$	-0.01	4.4%	0.41	0.40	0.16
$n = 100$	0.0023	2.3%	0.45	0.45	0.15
	$\alpha = \frac{2}{3}, \beta = \frac{1}{6}, \sigma = \frac{1}{10}$				
$n = 10$	-0.13	6.1%	0.30	0.33	0.16
$n = 20$	-0.07	3.3%	0.38	0.41	0.13
$n = 50$	-0.03	2.2%	0.44	0.40	0.11
$n = 100$	0.001	30.2%	0.45	0.44	0.14
	$\alpha = \frac{2}{3}, \beta = \frac{1}{6}, \sigma = \frac{1}{2}$				
$n = 10$	-0.10	2.4%	0.37	0.24	0.18
$n = 20$	-0.10	1.1%	0.41	0.35	0.12
$n = 50$	-0.01	4.1%	0.45	0.45	0.17
$n = 100$	0.0020	3.3%	0.45	0.43	0.12

With $m = q = 5$ the CPU timings we obtained in the empirical application, were as follows: Bootstrap-DEA with 5,000 replications takes 280 seconds while MCMC takes 7.28. With S=10,000 MCMC iterations, CPU time was much lower (1.2 seconds). Of course, MCMC is inherently serial and cannot be parallelized easily. We also ran bootstrap DEA and Bayesian MCMC on synthetic data sets consisting of randomly generated input - output data, from standard lognormal distributions and 2,500 observations (which is nearly the same number of observations as in the empirical application). We do so to examine the effect of varying m and q. The results are reported in Table 4. As the number of inputs and outputs decreases, CPU times for MCMC are lower as there are less constraints in the parameter space.

Table 2: Monte Carlo results II

"Bias" is the median bias in efficiency scores from MCMC. GCD provides the percentage of Geweke's (1992) t-statistic convergence diagnostic in excess of 1.96. RNE denotes relative numerical efficiency (Geweke, 1992). "acf (10)" and "acf (50)" denote median autocorrelation in θ at lags 10 and 50 respectively.

	Bias	GCD, $\|t\| > 1.96$	RNE	acf (10)	acf (50)
	$\tilde{\lambda} = \frac{1}{2}$				
$n = 10$	-0.12	2.1%	0.37	0.21	0.13
$n = 20$	-0.05	2.2%	0.42	0.32	0.14
$n = 50$	-0.01	31.5%	0.45	0.41	0.16
$n = 100$	0.0013	3.4%	0.45	0.53	0.14
	$\tilde{\lambda} = 1$				
$n = 10$	-0.12	3.2%	0.40	0.31	0.12
$n = 20$	-0.10	3.4%	0.42	0.44	0.14
$n = 50$	-0.01	3.1%	0.44	0.45	0.16
$n = 100$	0.0012	3.3%	0.45	0.44	0.15
	$\tilde{\lambda} = 2$				
$n = 10$	-0.11	3.3%	0.38	0.43	0.16
$n = 20$	-0.05	3.5%	0.43	0.41	0.13
$n = 50$	-0.03	3.1%	0.45	0.42	0.11
$n = 100$	0.001	3.1%	0.45	0.42	0.14

Table 3: Prior sensitivity analysis

The table reports: i) In the first column, the median (across the sample and all 10,000 alternative priors) of the absolute difference between efficiency scores from a new prior and efficiency scores from the baseline prior. ii) In the second column the median (across the sample and all 10,000 alternative priors) of the absolute difference between efficiency scores from (11) relative to the specification in (10).

	Difference of efficiency scores	Difference of efficiency scores between (11) and (10)
	$\tilde{\lambda} = 2$	
$n = 10$	0.11%	0.072%
$n = 20$	0.0021%	0.0044%
$n = 50$	0.0017%	0.0032%
$n = 100$	0.0011%	0.0028%
$\tilde{\lambda} = 1$		
$n = 10$	0.13%	0.34%
$n = 20$	0.0032%	0.0041%
$n = 50$	0.0021%	0.0037%
$n = 100$	0.0015%	0.0032%
$\tilde{\lambda} = \frac{1}{2}$		
$n = 10$	0.32%	0.25%
$n = 20$	0.15%	0.013%
$n = 50$	0.047%	0.0082%
$n = 100$	0.023%	0.0055%

Table 4. CPU times (in seconds)

Notes: m in the number of inputs, q is the number of outputs, B is the number of bootstrap replications in DEA and S is the total number of MCMC simulations. CPU timing in bootstrap DEA with bias correction includes only the computation of linear programs. For MCMC it includes CPU time for obtaining posterior moments and densities / histograms of functions of interest (efficiency scores). In all cases the number of observations is 2,400 (which is quite close to the actual number of observations in the empirical application). Computations were performed on an Intel ®i9 @ 3.4 GHz.

	DEA	Bayesian MCMC (S=150,000)
m=5,q=5, B=500	27.23	7.40
m=5,q=5, B=5,000	281.89	—
m=2,q=2, B=500	18.31	5.12
m=1,q=1, B=500	10.18	3.47
m=1,q=1, B=5,000	101.25	—