Omega 128 (2024) 103130

Contents lists available at ScienceDirect

Omega

journal homepage: www.elsevier.com/locate/omega

Support Vector Frontiers with kernel splines

Nadia M. Guerrero [a], Raul Moragues [a], Juan Aparicio [a,b,*], Daniel Valero-Carreras [a]

[a] Center of Operations Research (CIO), Miguel Hernandez University of Elche (UMH), Av. de la Universidad s/n, Elche 03202 Spain
[b] Joint Research Unit, Valencian Graduate School and Research Network of Artificial Intelligence (valgrAI), Camino de Vera s/n, Valencia 46022 Spain

ARTICLE INFO

Keywords:
Data envelopment analysis
Support vector regression
Linear splines
Convexity

ABSTRACT

Among recent methodological proposals for efficiency measurement, machine learning methods are playing an important role, particularly in the reduction of overfitting in classical statistical methods. In particular, Support Vector Frontiers (SVF) is a method which adapts Support Vector Regression (SVR) to the estimation of production technologies through stepwise frontiers. The SVF estimator is convexified in a second stage to deal with convex technologies. In this paper, we propose SVF-Splines, an extension of SVF for the estimation of efficiency in multi-input multi-output production processes which uses a transformation function generating linear splines to directly estimate convex production technologies. The proposed methodology reduces the computational complexity of the original SVF and does not require a two-step estimation process to obtain convex production technologies. A simulated experiment comparing SVF-Splines with standard DEA and (convexified) SVF indicates better performance of the proposed methodology, with improvements of up to 95 % in mean squared error when compared with DEA. The computational advantages of SVF-Splines are also observed, with runtime over 70 times faster than SVF in certain scenarios, with better scaling as the size of the problem increases. Finally, an empirical illustration is provided where SVF-Splines is calculated with respect to various typical technical efficiency measures of the literature.

1. Introduction

When faced with a group of companies or other entities which an analyst wants to evaluate and compare from a benchmarking point of view, an important line of research is the determination of the underlying production process that is behind the observed data. Many of the existing approaches in the literature can be split into two families, parametric and non-parametric methods. Among the most widely used parametric approaches, we encounter Stochastic Frontier Analysis (SFA) [1,26] while among the non-parametric perspectives, Data Envelopment Analysis (DEA) [5,8] has received enough attention to develop into its own research topic.

Among the advantages of non-parametric approaches, their flexibility, the mild conditions required for their use, and the natural way in which they deal with multi-input multi-output production processes have been pointed out [11]. In particular, DEA is characterised by its estimation of the production technology as the smallest set which satisfies envelopment of the data from above, free disposability of inputs and outputs, and convexity. The smallest set is achieved via the principle of minimal extrapolation. Within this context, various types of assumptions are possible, yielding different estimators. For example, a related estimator is Free Disposal Hull (FDH) [12], which removes the convexity postulate. This results in stepwise frontiers in FDH as opposed to the piecewise linear frontiers estimated by DEA.

As one of the most well-known non-parametric models, many properties of DEA have been considered. In particular, the postulate of minimal extrapolation has led to criticisms being leveraged that it is a conservative estimator, sometimes even labelling it as a pure descriptive approach [14]. This results in a set which fits too closely to the observed data and may not correctly estimate the underlying production process. Various authors have attempted to overcome this issue and endow DEA with inferential capabilities from the statistical point of view, such as [11], who propose a characterization of the Data Generating Process (DGP) that is behind the observations. They assume that the observations are a sample of identically and independently distributed random variables with an unknown joint distribution. The task of the estimation of the production technology can then be identified with estimating the support of the underlying DGP. They used this setting to perform

Area: Data-Driven Analytics This manuscript was processed by Associate Editor Stanko Dimitrov.
* Corresponding author at: Center of Operations Research (CIO). Miguel Hernandez University of Elche (UMH), Av. de la Universidad s/n, Elche 03202 Spain.
E-mail address: j.aparicio@umh.es (J. Aparicio).

https://doi.org/10.1016/j.omega.2024.103130
Received 26 August 2023; Accepted 4 June 2024
Available online 5 June 2024

inference tasks such as proving consistency and performing bootstrapping to estimate confidence intervals on DEA models.

Recently, another approach which has become more important makes use of some of the similarities between nonparametric methods and the machine learning literature. Among machine learning models, Support Vector Machines (SVM) [38,39] are an interesting family of machine learning algorithms, since the approach is based on solid statistical learning theory. Support Vector algorithms use the principle of Structural Risk Minimization to aim to obtain models with good generalization capabilities via bounds on estimates of the out-of-sample generalization error (prediction error) of models. Some recent contributions in this line of research ([36], 2022) proposed an adaptation of Support Vector Regression (SVR), called Support Vector Frontiers (SVF) to the estimation of stepwise production frontiers, i.e., comparable with FDH, and a Convexified SVF (CSVF), which is comparable to DEA.

Other methodology works which use machine learning principles for measuring efficiency can be seen in the Corrected Convex Nonparametric Least Squares (CCNLS) proposal by [21], while [31] proposed a smooth nonparametric kernel frontier estimator. [10] introduced an estimator based on quadratic and cubic splines with shape constraints. Decision trees have been adapted in various ways, such as [14,35]. The Structural Risk Minimization was used to construct a technology estimator by [16]. A representation of production frontiers using hinging hyperplanes was introduced by [29]. Boosting methods have been adapted by [17,18]. Additive models based on splines have been proposed by [13]. In addition to the regression-based approaches, based on supervised learning methods, a recent contribution has proposed an unsupervised learning-based generalization of DEA [27,28], among others.

One of the tools which allow SVMs to be very flexible is the use of transformation functions with associated kernels. These map the original space of predictors into a higher-dimensional space, where the classification/regression task is performed via a hyperplane which, when transformed back to the original space, can have different and flexible shapes. Usual kernels can be linear, polynomial, splines, gaussian, RBF kernels, among others. Within the SVM family, we encounter the Support Vector Regression (SVR) algorithm, which applies the SVM approach to regression problems. The flexibility of SVM using kernels allows SVR to estimate functions satisfying a variety of properties. An important family of kernels is given by the splines generating kernels, which allow the flexibility of splines interpolation to be used in conjunction with Support Vector Machines [39].

In the context of efficiency measurement, SVF [36,37] resorts to SVR that partitions the input space into a grid of cells, and associates to each cell values of 0 or 1 according to the location of data points on the grid. The use of constant values results in the use of step functions, which yields a stepwise estimation of the production frontier, in line with FDH, which is at a second stage convexified to obtain a production technology along the lines of DEA. However, this choice of transformation function causes the method to have a large computational expense. We remark that the Boolean grid of values used by SVF can be seen as a kernel generating splines of order 0, ([39], p. 464).

In this paper, we propose SVF-Splines, an extension of SVF which uses a transformation function involving splines of order 1. This results in piecewise linear estimators which can be directly compared with DEA while, at the same time, reducing the computational complexity of SVF, as we will show. We consider the restrictions which ensure that the estimator satisfies the microeconomic postulates of convexity, free disposability in inputs and outputs and data envelopment. We then compare this estimator in a computational experiment with traditional DEA and with CSVF, that is, the convexified version of SVF. We observe better results with lower computation times, particularly as the number of observations and dimensions increase. We also adapt a variety of classical measures of efficiency to the SVF-Splines estimator and illustrate with an empirical example the efficiencies obtained by DEA and the new approach.

The rest of the paper is structured as follows. Section 2 describes background concepts about Data Envelopment Analysis, Support Vector Regression and (Convex) Support Vector Frontiers. Section 3 adapts the linear splines kernel to SVF to introduce the SVF-Splines algorithm, proves that it satisfies the microeconomic postulates, characterizes the estimated technology as a DEA-type technology, and shows how to calculate a variety of measures of efficiency with the SVF-Splines estimator. Section 4 results from a computational experiment comparing the proposed SVF-Splines algorithm to DEA and Convex Support Vector Frontiers, as well as a discussion about their computational characteristics. Section 5 then illustrates the results obtained by SVF-Splines in an empirical example. Finally, Section 6 presents the conclusions obtained and outlines further possible lines of research.

2. Background

2.1. Data Envelopment Analysis

Data Envelopment Analysis (DEA) is one of the most well-known techniques for measuring the efficiency of a set of units which use a variety of inputs to produce a variety of outputs. It is a nonparametric technique which estimates technical efficiency as the "distance" (along some permissible direction) to the efficient frontier of a production technology. The DEA production technology consists of the unique smallest set which envelops the observations, while satisfying convexity and free disposability of inputs and output [5]. In a production process with n DMUs (Decision Making Units) which use m inputs and produce s outputs, we denote inputs as $\mathbf{x} \in \mathbb{R}^m_+$ and outputs as $\mathbf{y} \in \mathbb{R}^s_+$. Let$X \in \mathbb{R}^{m\times n}_+$ ($Y \in \mathbb{R}^{s\times n}_+$) be the matrix containing all the inputs (outputs) of the DMUs in the dataset, with each DMU as a column. The DEA estimate of the technology under Variable Returns to Scale (VRS) is [5]:

$$\widehat{T}_{DEA} = \left\{(\boldsymbol{x},\boldsymbol{y}) \in \mathbb{R}^{m+s}_+ : \boldsymbol{x} \geq X\lambda, \boldsymbol{y} \leq Y\lambda, \lambda \geq \mathbf{0}, \lambda\mathbf{1} = 1\right\} \tag{1}$$

DEA assumes convexity of its production technology, which is a polyhedral set. When the convexity assumption is relaxed, we obtain the Free Disposal Hull (FDH) estimator, which envelops the data and satisfies free disposability of inputs and outputs and minimal extrapolation but does not satisfy convexity. The production technology estimated by FDH is stepwise, and the convexification of this technology is the technology estimated by DEA on the same data.

A region of the production technology of particular importance for the measurement of the efficiency of DMUs is the efficient frontier. There are various possible characterizations of this subset, such as the weakly efficient frontier $\delta^W(T)$ and strongly efficient frontier $\delta^S(T)$:

$$\delta^W(T) = \{(\boldsymbol{x},\boldsymbol{y}) \in T : \widehat{\boldsymbol{x}} < \boldsymbol{x}, \widehat{\boldsymbol{y}} > \boldsymbol{y} \Rightarrow (\widehat{\boldsymbol{x}},\widehat{\boldsymbol{y}}) \in T\} \tag{2}$$

$$\delta^S(T) = \{(\boldsymbol{x},\boldsymbol{y}) \in T : \widehat{\boldsymbol{x}} \leq \boldsymbol{x}, \widehat{\boldsymbol{y}} \geq \boldsymbol{y}, (\boldsymbol{x},\boldsymbol{y}) \neq (\widehat{\boldsymbol{x}},\widehat{\boldsymbol{y}}) \Rightarrow (\widehat{\boldsymbol{x}},\widehat{\boldsymbol{y}}) \notin T\} \tag{3}$$

Elements of the strongly efficient frontier do not admit any improvement along any variable (input or output) without worsening along some other component (input or output) while remaining feasible. The weakly efficient frontier, however, consists of those elements that are not strictly Pareto dominated by any other feasible bundle, i.e., it also contains those elements which allow for improvement along one dimension while keeping the remaining variables constant. These elements do not belong to the strongly efficient frontier. Hence, the strongly efficient frontier is a subset of the weakly efficient frontier, though they do not necessarily coincide. In the case of DEA, the efficient frontiers are both piecewise linear sets. Various measures of efficiency project to either of the two efficient frontiers.

In this paper, as technical efficiency measures, we consider the radial measures, both input and output oriented, which were the first introduced in [15,5]. We also consider the Directional Distance Function [7], and the Weighted Additive Measure [9].

With respect to an arbitrary production technology T, the output-

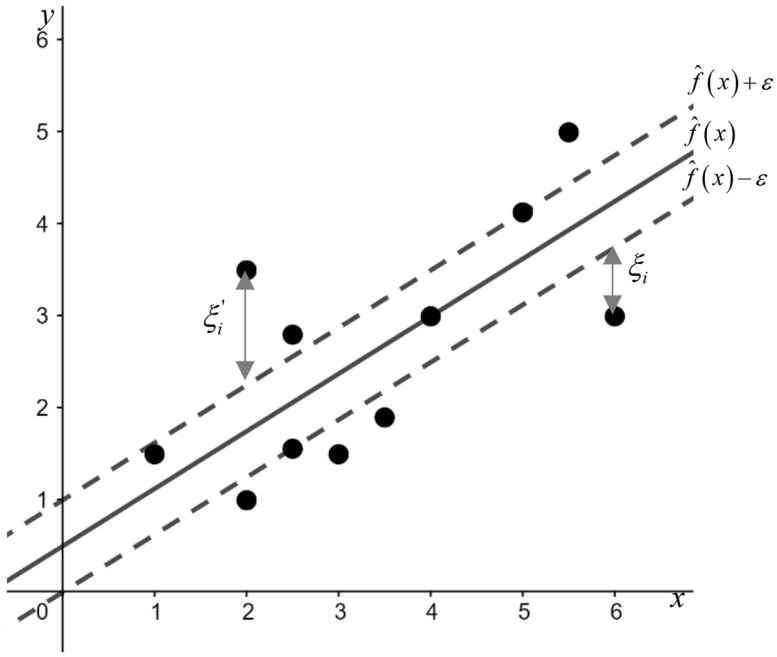

Fig. 1. Linear Support Vector Regression estimation.

oriented radial measure measures how much the outputs can be increased by the same proportion while remaining feasible. It can be calculated using the following model:

$$\psi(\boldsymbol{x},\boldsymbol{y}) = \max\{\psi \in \mathbb{R} : (\boldsymbol{x}, \psi\boldsymbol{y}) \in T\} \tag{4}$$

Similarly, the input-oriented radial measure describes how much every input can be reduced by the same amount while the DMU does not become infeasible, and is calculated by solving the following model:

$$\theta(\boldsymbol{x},\boldsymbol{y}) = \min\{\theta \in \mathbb{R} : (\theta\boldsymbol{x}, \boldsymbol{y}) \in T\} \tag{5}$$

The directional distance function (DDF) projects the given bundle of inputs and outputs along a pre-specified direction, given by a nonzero directional vector $\boldsymbol{g} = (\boldsymbol{g}^-, \boldsymbol{g}^+)$. It is a graph type measure, since it seeks to improve both inputs and outputs simultaneously. It was introduced by [7]:

$$\beta(\boldsymbol{x},\boldsymbol{y}) = \max\{\beta \in \mathbb{R} : (\boldsymbol{x} - \beta\boldsymbol{g}^-, \boldsymbol{y} + \beta\boldsymbol{g}^+) \in T\} \tag{6}$$

In this paper, we choose the directional vector $\boldsymbol{g} = (\boldsymbol{x}, \boldsymbol{y})$ given by the values of the input-output bundle itself. This choice results in a units-invariant measure. The DDF and both radial measures project DMUs to the weakly efficient frontier, so there may be additional potential improvements (slacks) along some directions.

The Weighted Additive (WA) measure which we consider ensures that DMUs are projected to the strongly efficient frontier, as it detects slacks along any input or output. It takes as its basis a slightly different DEA model, introduced in [23]. Given input-output weights $(\boldsymbol{\rho}^-, \boldsymbol{\rho}^+) \in \mathbb{R}^{m+s}_{++}$, the Weighted Additive Model is calculated as follows:

$$WA(\boldsymbol{x},\boldsymbol{y}) = \max\left\{\boldsymbol{\rho}^-\boldsymbol{s}^- + \boldsymbol{\rho}^+\boldsymbol{s}^+ : (\boldsymbol{x} - \boldsymbol{s}^-, \boldsymbol{y} + \boldsymbol{s}^+) \in T, (\boldsymbol{s}^-, \boldsymbol{s}^+) \in \mathbb{R}^{m+s}_+\right\} \tag{7}$$

In particular, we use weights corresponding to the Range Adjusted Measure [9]. These weights are given by $\rho^{-(j)} = \frac{1}{(m+s)R_j^-}$ and $\rho^{+(r)} = \frac{1}{(m+s)R_r^+}$, where R_j^- is the range of input j and R_r^+ is the range of values of output r. This choice results in a graph measure which is invariant to units of measurement.

The radial measures both determine as efficient those DMUs with efficiency 1. However, in the case of the output orientation, every DMU attains values larger than unity, while in the input oriented measure the efficiencies attain values between 0 and 1. Meanwhile, the DDF and WA can be considered measures of inefficiency, as efficient DMUs attain values of 0, and larger values indicate less efficient units. With the choices of weights and directional vector above, they are bounded above by 1.

2.2. Support Vector Regression and splines kernel

We now describe the Support Vector Regression (SVR) algorithm and the splines kernel that we will use. SVR is a regression algorithm that adapts the Structural Risk Minimization problem to estimate a regression on a variable while not overfitting too much to the data. Originally introduced with the Euclidean or l_2 norm, it has been extended to deal with other norms, such as the l_1 norm, which results in a linear objective function, other l_p or the l_∞ norms (see e.g. [6,32], and [41]). The base SVR model with respect to such a norm (with margin ε) is given by:

$$\begin{aligned} \underset{\boldsymbol{w},b,\xi',\xi_i}{Min} \quad & \| \boldsymbol{w} \| + C\sum_{i=1}^{n}\left(\xi_i'^2 + \xi_i^2\right) \\ & y_i - (\boldsymbol{w}\cdot\boldsymbol{x}_i + b + \varepsilon) \le \xi_i', \quad i = 1,\dots,n \\ & (\boldsymbol{w}\cdot\boldsymbol{x}_i + b - \varepsilon) - y_i \le \xi_i, \quad i = 1,\dots,n \\ & \xi_i', \xi_i \ge 0, \quad i = 1,\dots,n \end{aligned} \tag{8}$$

The objective function of this model consists of two parts, a regularization term $\| \boldsymbol{w} \|$ and an empirical error term $\sum_{i=1}^{n}(\xi_i'^2 + \xi_i^2)$, which are combined via a weight C. This C is a hyperparameter which, together with the margin hyperparameter ε, is obtained via a cross-validation process. The SVR model estimates a decision function given by $\widehat{f}(\boldsymbol{x}) = \boldsymbol{w}^*\cdot\boldsymbol{x} + b^*$ with errors of ξ, ξ' for the observed data outside of an ε-insensitive region. Thus, the observations within an ε margin of the estimated function $\widehat{f}(\boldsymbol{x})$ attain an error of 0, and the empirical errors are measured to this ε margin of the decision function. A graphical illustration of a function estimated by the model can be found in Fig. 1.

The SVR method can be adapted to estimate nonlinear functions via the use of a transformation function ϕ which maps the space of predictor variables into a higher-dimensional space in such a way that the obtained estimation function is linear in the transformed space but not in the original space. This is sometimes called the "kernel trick" in the

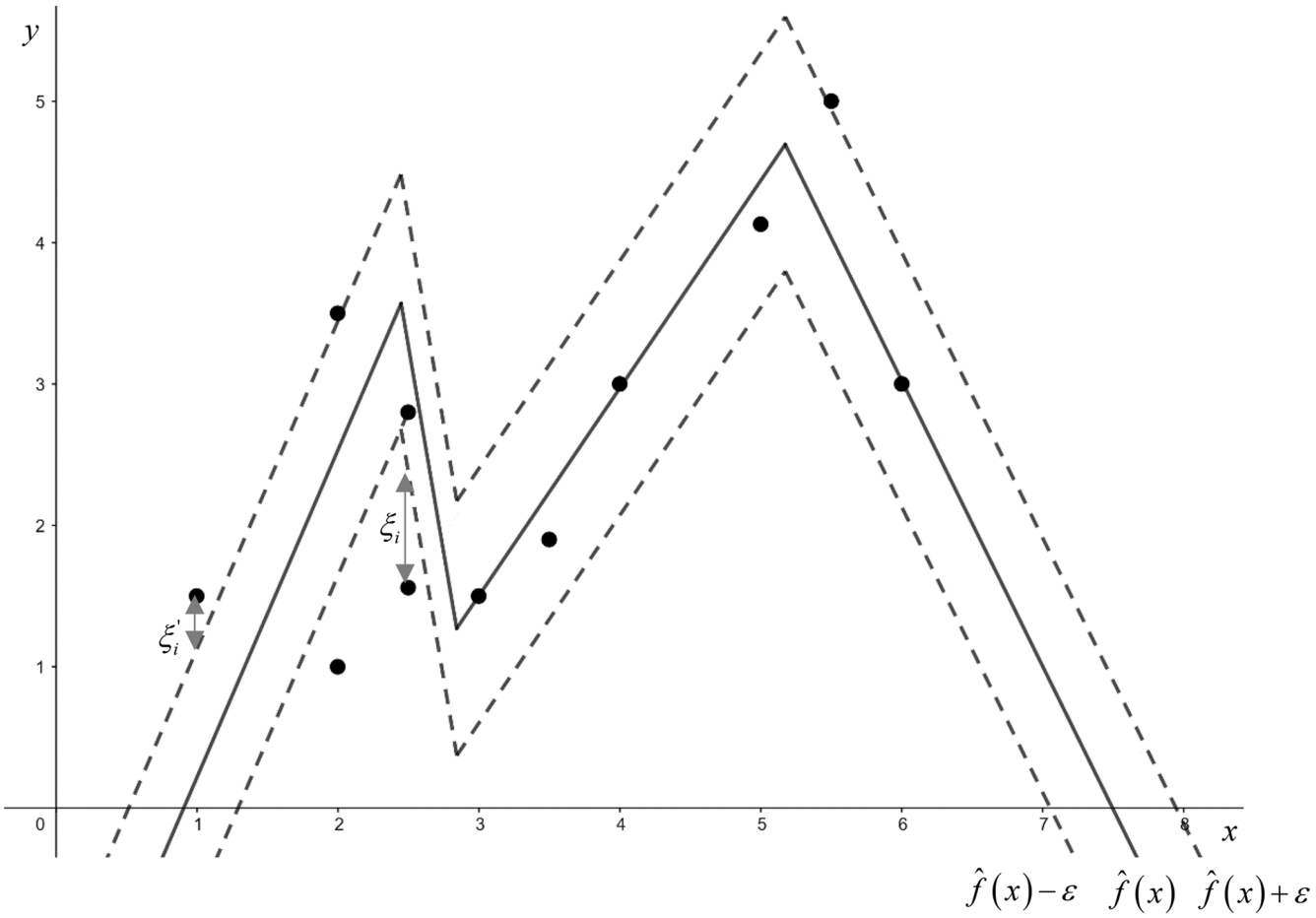

Fig. 2. Support Vector Regression with linear splines.

literature. The kernel SVR model with transformation function ϕ is the following:

$$\begin{aligned} \underset{\boldsymbol{w},b,\xi',\xi_i}{Min} \quad & \| \boldsymbol{w} \| + C\sum_{i=1}^{n}\left(\xi'^{2}_i + \xi_i^2\right) \\ & y_i - (\boldsymbol{w}\cdot\phi(\boldsymbol{x}_i) + b + \varepsilon) \le \xi'_i, \quad i = 1,...,n \\ & (\boldsymbol{w}\cdot\phi(\boldsymbol{x}_i) + b - \varepsilon) - y_i \le \xi_i, \quad i = 1,...,n \\ & \xi'_i, \xi_i \ge 0, \qquad\qquad i = 1,...,n \end{aligned} \tag{9}$$

When a transformation function ϕ is used, the function estimated by the model is $\widehat{f}(\boldsymbol{x}) = \boldsymbol{w}^*\cdot\phi(\boldsymbol{x}) + b^*$, and its shape depends on the characteristics of the transformation function ϕused. There is a wide variety of possible kernels to use in this context, such as linear, polynomial, Gaussian, grid-like and splines kernels.

In this paper, we focus our attention on the splines transformation function proposed by Vapnik in ([39], p. 464). This transformation uses a finite number of knots to construct splines of order q by splitting each input dimension into a finite number of knots. Splines are flexible functions defined piecewise by polynomials of degree q, whose formulation is given by:

$$\begin{aligned} \phi(\boldsymbol{x}_i) = \quad & \left(1, x_i^{(1)}, \left(x_i^{(1)}\right)^2, ..., \left(x_i^{(1)}\right)^q, \left(x_i^{(1)} - t_1^{(1)}\right)_+^q, ..., \left(x_i^{(1)} - t_{k_1}^{(1)}\right)_+^q, \right. \\ & 1, x_i^{(2)}, \left(x_i^{(2)}\right)^2, ..., \left(x_i^{(2)}\right)^q, \left(x_i^{(2)} - t_1^{(2)}\right)_+^q, ..., \left(x_i^{(2)} - t_{k_2}^{(2)}\right)_+^q, \\ & \vdots \\ & \left. 1, x_i^{(m)}, \left(x_i^{(m)}\right)^2, ..., \left(x_i^{(m)}\right)^q, \left(x_i^{(m)} - t_1^{(m)}\right)_+^q, ..., \left(x_i^{(m)} - t_{k_m}^{(m)}\right)_+^q \right) \end{aligned} \tag{10}$$

In this transformation function, the jth component of $\boldsymbol{x}$ is transformed into a $(1 + q + k_j)$-dimensional vector, where k_j is the number of knots along dimension j. The first component of such vector is a constant value of 1, the next q components are powers of the original component of $\boldsymbol{x}$, and the final k_j elements are defined by:

$$\left(x_i^{(j)} - t_{l_j}^{(j)}\right)_+^q = \begin{cases} \left(x_i^{(j)} - t_{l_j}^{(j)}\right)^q & if \quad x_i^{(j)} > t_{l_j}^{(j)} \\ 0 & if \quad x_i^{(j)} \le t_{l_j}^{(j)} \end{cases}, l_j = 1,...,k_j, j = 1,...,m \tag{11}$$

In particular, splines of order $q = 0$ yield step functions as estimators, while splines of order $q = 1$, also called linear splines, produce piecewise linear estimators, see Fig. 2 for an example. Higher values of q provide piecewise approximations using polynomials of degree q. In this paper, we focus on splines of order $q = 1$, inspired by the nature of DEA, which estimates a piecewise linear production function.

2.3. Support Vector Frontiers

A recent contribution [36] shows how to adapt Support Vector Regression to the estimation of production functions in the single-output case. The authors adapt the SVR model to satisfy the properties of production functions of envelopment from above and monotonicity and develop Support Vector Frontiers (SVF). They propose a transformation function $\phi^G_{SVF}(\boldsymbol{x})$ of the space of inputs which consists of binary values on a grid with its components taking values of 0 or 1 whenever they are dominated by $\boldsymbol{x}$. As a consequence, SVF estimates step production functions, in other words, comparable with the Free Disposal Hull estimator. The authors also show how to convexify this technology in order to estimate DEA-like production functions, via Convexified Support Vector Frontiers (CSVF). Additionally, SVF has later been extended to the multi-output context in [37]. The multi-output SVF model is given by:

$$
\begin{array}{lll}
\underset{\boldsymbol{w}_r,\xi_{ri}}{Min} & \sum_{r=1}^{s}\left\|\boldsymbol{w}^{(r)}\right\| + C\sum_{r=1}^{s}\sum_{i=1}^{n}\xi_i^{(r)} & \\
s.t. & \boldsymbol{w}^{(r)}\cdot\phi^G_{SVF}\left(\boldsymbol{x}_i\right) - y_i^{(r)} \geq 0, & i=1,\ldots,n \quad r=1,\ldots,s \\
 & \boldsymbol{w}^{(r)}\cdot\phi^G_{SVF}\left(\boldsymbol{x}_i\right) - y_i^{(r)} \leq \varepsilon + \xi_i^{(r)}, & i=1,\ldots,n \quad r=1,\ldots,s \\
 & W^{(r)}_{l_1\ldots s_j\ldots l_m} \leq W^{(r)}_{l_1\ldots l_m}, & l_1=1,\ldots,k_1,\ldots,l_m=1,\ldots,k_m \quad s_j=l_j-1 \\
 & & l_j=1,\ldots,k_j, j=1,\ldots,m \quad r=1,\ldots,s \\
 & \xi_i^{(r)} \geq 0, & i=1,\ldots,n
\end{array}
\tag{12}
$$

We remark that this model, when $s=1$, is equivalent to the single-output model introduced in [36]. It involves a transformation function $\phi^G_{SVF}(\boldsymbol{x})$ associated to a grid G common to all outputs, which partitions the input space into a grid of binary values, as well as a different set of weights $\boldsymbol{w}^{(r)}$ for each output dimension. Each vector of weights has one component for each cell of the grid G. In (12), $\phi^G_{SVF}(\boldsymbol{x})$ can be considered a splines transformation of order 0. It results in an estimation for each output $\widehat{y}^{(r)}_{SVF}(\boldsymbol{x}_i) := \boldsymbol{w}^{(r)}\cdot\phi^G_{SVF}(\boldsymbol{x}_i)$, $r=1,\ldots,s$, which is a stepwise production frontier in line with Free Disposal Hull.

The SVF model has an empirical error $\xi_i^{(r)}$ associated with each $y_i^{(r)}$, the output dimension r of DMU i, and model (12) minimizes the weighted sum $\xi_i^{(r)}$ of the regularization term $\sum_{r=1}^{s}\|\boldsymbol{w}^{(r)}\|$ and the empirical error $\sum_{r=1}^{s}\sum_{i=1}^{n}\xi_i^{(r)}$, weighted by a parameter C. The first constraint in (12) ensures that the estimator envelops the observed data from above, while the second constraint will penalize the error committed beyond a margin parameter ε via the empirical error $\xi_i^{(r)}$, which will be nonnegative by the final restriction. Finally, the third restriction, which involves variables $W^{(r)}_{l_1\ldots l_m}$ defined by $W^{(r)}_{l_1\ldots l_m} := \sum_{\substack{s_1=1,\ldots,l_1 \\ \vdots \\ s_m=1,\ldots,l_m}} w^{(r)}_{s_1\ldots s_m}$, for each output dimension $r=1,\ldots,s$, will ensure that the estimated output $\widehat{y}^{(r)}_{SVF}(\boldsymbol{x})$ is a monotonic non-decreasing function. Once model (12) is solved, optimal values $\boldsymbol{w}^{*(r)}$ are obtained, and the estimated output is given by $\widehat{y}^{(r)}_{SVF}(\boldsymbol{x}) = \boldsymbol{w}^{*(r)}\cdot\phi^G_{SVF}(\boldsymbol{x})$ for each $r=1,\ldots,s$. The associated production technology is given by:

$$
\widehat{T}_{SVF} := \left\{(\boldsymbol{x},\boldsymbol{y}) \in \mathbb{R}^{m+s}_{+} : \boldsymbol{y} \leq \widehat{\boldsymbol{y}}_{SVF}(\boldsymbol{x})\right\} \tag{13}
$$

By ([37], Section 3.1), this technology satisfies envelopment of the data, free disposability, and it is an FDH-style production technology. In particular, it can be characterised as an FDH-type estimator on a set of "virtual points" defined by the grid involved in the transformation function $\phi^G_{SVF}(\boldsymbol{x})$. These points are virtual in the sense that they are not necessarily in the original data. Instead, they are defined using the predictions of the model on extreme points of the grid cells. In a second stage, this production technology is convexified to obtain a DEA-style technology with respect to these virtual points, yielding the Convexified SVF (CSVF) estimator. This results in a two-stage estimation of convex production technologies. In particular, to simplify computation, each split input dimension is split in the same number of nodes, which leads to a single hyperparameter d to be tuned together with C and ε via a five-fold cross-validation procedure.

The margin hyperparameter ε can be used to define a more robust notion of technical efficiency, which the authors name ε-insensitive technical efficiency. This notion considers any DMU within a margin ε of the technical efficient frontier to be ε-insensitive technically efficient, and allows for the identification of those units which are not considered ε-insensitive technically efficient as units which are far from being technically efficient.

As recognized by its authors, the SVF estimator has some limitations, particularly of computational complexity. The proposed model involves a large amount of variables and restrictions, given that there is one w associated to each cell of the grid, which results in an exponential ($\approx md^m$) number of restrictions involving the W.

In this paper, we propose a way to overcome these limitations by using a model based on linear splines in order to reduce the computational complexity and directly estimate convex production technologies in a single stage.

3. Support Vector Frontiers with kernel splines

3.1. The single-output case

In this section, we show how to adapt a linear splines kernel (with $q=1$) to measure efficiency via the modification of the Support Vector Frontiers estimation with this transformation function. This adaptation reduces the exponential complexity of the restrictions of model (12) to a set of a linear number of restrictions via the choice of nodes along each of the input components. We begin by presenting the single-output model and proving its properties, before generalizing it to the multi-output case:

$$
\begin{aligned}
&\underset{\boldsymbol{w},\xi_i}{Min} && \| \boldsymbol{w} \| + C\sum_{i=1}^{n}\xi_i \\
&s.t. \\
&\quad \boldsymbol{w}\cdot\phi^{G}_{SVF-SP}(\boldsymbol{x}_i) - y_i \le \varepsilon + \xi_i && i = 1,\ldots,n && (14.1)\\
&\quad y_i - \boldsymbol{w}\cdot\phi^{G}_{SVF-SP}(\boldsymbol{x}_i) \le 0 && i = 1,\ldots,n && (14.2)\\
&\quad W^{(j)}_{l_j} \ge 0 && l_j = 0,\ldots,k_j \quad j = 1,\ldots,m && (14.3)\\
&\quad w^{(j)}_0, w^{(j)}_{-1} \ge 0 && && j = 1,\ldots,m \quad (14.4)\\
&\quad w^{(j)}_k \le 0 && k = 1,\ldots,k_j \quad j = 1,\ldots,m && (14.5)\\
&\quad \xi_i \ge 0 && i = 1,\ldots,n && (14.6)
\end{aligned}
\tag{14}
$$

where

$$
\phi^{G}_{SVF-SP}(\boldsymbol{x}_i) = \left(1, x^{(1)}_i, \left(x^{(1)}_i - t^{(1)}_1\right)_+, \ldots, \left(x^{(1)}_i - t^{(1)}_{k_1}\right)_+, \quad \ldots \quad 1, x^{(m)}_i, \left(x^{(m)}_i - t^{(m)}_1\right)_+, \ldots, \left(x^{(m)}_i - t^{(m)}_{k_m}\right)_+\right) \tag{15}
$$

with

$$
\left(x^{(j)}_i - t^{(j)}_{l_j}\right)_+ = \begin{cases} x^{(j)}_i - t^{(j)}_{l_j} & if \quad x^{(j)}_i > t^{(j)}_{l_j} \\ 0 & if \quad x^{(j)}_i \le t^{(j)}_{l_j} \end{cases}, l_j = 1,\ldots,k_j, j = 1,\ldots,m \tag{16}
$$

Thus, $\phi^{G}_{SVF-SP}(\boldsymbol{x}_i)$ is a transformation from $\mathbb{R}^m \to \mathbb{R}^{2m+\sum_{j=1}^{m}k_j}$ (i.e., a $\sum_{j=1}^{m}(k_j+2)$-dimensional space). We denote this dimension by h. The corresponding weights vector is:

$$
\boldsymbol{w} = \left(w^{(1)}_{-1}, w^{(1)}_0, w^{(1)}_1, \ldots, w^{(1)}_{k_1}, \quad \ldots \quad, w^{(m)}_{-1}, w^{(m)}_0, w^{(m)}_1, \ldots, w^{(m)}_{k_m}\right) \tag{17}
$$

We remark that weights $w^{(j)}_1, \ldots, w^{(j)}_{k_j}$ correspond to the nodes $t^{(j)}_1, \ldots, t^{(j)}_{k_j}$ in (15). Hence, we denote the first two components along each input dimension j, which do not correspond to any such nodes, by $w^{(j)}_{-1}$ and $w^{(j)}_0$. For ease of notation, we combine those weights which get used in each interval between two consecutive nodes of the splines transformation:

$$
W^{(j)}_{l_j} = \sum_{k=0}^{l_j} w^{(j)}_k \quad l_j = 0,\ldots,k_j, \quad j = 1,\ldots,m \tag{18}
$$

This transformation function and associated kernel involves various parameters, which must be estimated or chosen appropriately. Each input dimension j is split into a number k_j of nodes, and defines $k_j + 2$ components of both $\boldsymbol{w}$ and ϕ. We choose to divide each input dimension into a number d of nodes between the minimum and maximum values observed along each dimension of the same width $(\max - \min)/d$. This defines $d+1$ nodes for each input, so that the dimension of the transformation function, as well as that of $\boldsymbol{w}$, is $h = m(d+3)$. This value d is a hyperparameter of the algorithm which will be estimated via a five-fold cross-validation process together with C and ε. The input space is thus divided into $(d+2)^m$ grid cells.

Solving problem (14) yields optimal values $\boldsymbol{w}^*$ and $\boldsymbol{\xi}^*$ and the corresponding piecewise linear production function is given by $\widehat{f}(\boldsymbol{x}) = \widehat{y}_{SVF-SP}(\boldsymbol{x}) = \boldsymbol{w}^*\cdot\phi^{G}_{SVF-SP}(\boldsymbol{x})$. We will denote the corresponding $W^{(j)}_{l_j}$ at optimum by $W^{(j)*}_{l_j}$ as well. We now prove that $\widehat{f}(\boldsymbol{x})$ satisfies data envelopment, monotonicity and concavity. Note that the properties of free disposability and convexity for the multi-input multi-output framework are translated into (non-decreasing) monotonicity and concavity of the corresponding production function for the multi-input single-output case, respectively.

Proposition 3.1. *For each $i = 1,\ldots,n$, $\widehat{f}(\boldsymbol{x})$ satisfies envelopment of the data. That is, for each $i = 1,\ldots,n$, we have $y_i \le \widehat{f}(\boldsymbol{x}_i)$.*

Proof. Holds by constraint (14.2) and definition of $\widehat{f}(\boldsymbol{x})$, which forces $y_i - \boldsymbol{w}^*\cdot\phi^{G}_{SVF-SP}(\boldsymbol{x}_i) = y_i - \widehat{f}(\boldsymbol{x}_i) \le 0$ for each i.■

We now prove that $\widehat{f}(\boldsymbol{x})$ is monotonic non-decreasing.

Proposition 3.2. *If $\boldsymbol{x} \in \mathbb{R}^m$ and $\mathbf{z} \ge \mathbf{x}$, then $\widehat{f}(\mathbf{z}) \ge \widehat{f}(\mathbf{x})$.*

Proof. Assume that $\mathbf{z} \ge \mathbf{x}$. We construct a series of inequalities where at each step only one component changes. That is,

$$
\begin{aligned}
\mathbf{x} = \boldsymbol{\alpha}_0 &= (x(1),\ldots,x(m)) \le \boldsymbol{\alpha}_1 = (z(1),\ldots,x(m)) \le \ldots \le \boldsymbol{\alpha}_m \\
&= (z(1),\ldots,z(m)) = \mathbf{z}.
\end{aligned}
$$

We will prove the inequality between $\boldsymbol{\alpha}_{j-1}$ and $\boldsymbol{\alpha}_j$ for each $j = 1,\ldots,m$. Hence, we consider $\widehat{f}(\boldsymbol{\alpha}_j) - \widehat{f}(\boldsymbol{\alpha}_{j-1}) = \boldsymbol{w}^*\cdot\phi^{G}_{SVF-SP}(\boldsymbol{\alpha}_j) - \boldsymbol{w}^*\cdot\phi^{G}_{SVF-SP}(\boldsymbol{\alpha}_{j-1}) = \boldsymbol{w}^*\cdot\left(\phi^{G}_{SVF-SP}(\boldsymbol{\alpha}_j) - \phi^{G}_{SVF-SP}(\boldsymbol{\alpha}_{j-1})\right)$. Since the only component that changes between $\boldsymbol{\alpha}_{j-1}$ and $\boldsymbol{\alpha}_j$ is the jth component and $\boldsymbol{w}^*$ is fixed, the only components of ϕ^{G}_{SVF-SP} that change are those involving the jth component. Furthermore, $\beta^{(j)} = \alpha^{(j)}_j - \alpha^{(j)}_{j-1} \ge 0$. The other terms cancel out, and we have:

$$
\begin{aligned}
\widehat{f}(\boldsymbol{\alpha}_j) - \widehat{f}(\boldsymbol{\alpha}_{j-1}) &= w^{(j)*}_{-1}(1-1) + w^{(j)*}_0\beta^{(j)} \\
&\quad + \sum_{k=1}^{k_j} w^{(j)*}_k\left(\left(\alpha^{(j)}_j - t^{(j)}_k\right)_+ - \left(\alpha^{(j)}_{j-1} - t^{(j)}_k\right)_+\right).
\end{aligned}
$$

For each k, the terms involving $t^{(j)}_k$ within the summatory can either both be active, i.e., $\alpha^{(j)}_j \ge \alpha^{(j)}_{j-1} \ge t^{(j)}_k$, both inactive, that is, $t^{(j)}_k \ge \alpha^{(j)}_j \ge \alpha^{(j)}_{j-1}$, or only the $\alpha^{(j)}_j$ active, which happens when $\alpha^{(j)}_j \ge t^{(j)}_k \ge \alpha^{(j)}_{j-1}$. The term $\left(\alpha^{(j)}_j - t^{(j)}_k\right)_+ - \left(\alpha^{(j)}_{j-1} - t^{(j)}_k\right)_+$ is then $\left(\alpha^{(j)}_j - t^{(j)}_k\right)_+ - \left(\alpha^{(j)}_{j-1} - t^{(j)}_k\right)_+ = \alpha^{(j)}_j - \alpha^{(j)}_{j-1} = \beta^{(j)}$ (both active), $0 - 0 = 0 \le \beta^{(j)}$ if both are inactive, or $\left(\alpha^{(j)}_j - t^{(j)}_k\right) - 0 = \alpha^{(j)}_j - t^{(j)}_k \le \alpha^{(j)}_j - \alpha^{(j)}_{j-1} = \beta^{(j)}$. Thus, in any case, these terms are bounded above by $\beta^{(j)}$. Then, each such term is multiplied by $w^{(j)*}_k$, which is non-positive by restriction (14.5). Thus, we have $\widehat{f}(\boldsymbol{\alpha}_j) - \widehat{f}(\boldsymbol{\alpha}_{j-1}) \ge w^{(j)*}_0\beta^{(j)} + \sum_{k=1}^{k_j} w^{(j)*}_k\beta^{(j)} = W^{(j)*}_{k_j}\beta^{(j)} \ge 0$.

We remark that this final inequality holds as $W^{(j)*}_{k_j} \ge 0$, by constraint (14.3). Therefore, we have $\widehat{f}(\boldsymbol{\alpha}_j) \ge \widehat{f}(\boldsymbol{\alpha}_{j-1})$ for each j. By applying this argument repeatedly, we obtain that $\widehat{f}(\mathbf{z}) - \widehat{f}(\mathbf{x})$, as claimed. Thus, $\widehat{f}(\mathbf{x})$ is monotonic non-decreasing. ■

Next, to prove concavity of the estimated production function, we consider the nature of the components of the transformations, and the signs of the components of $\mathbf{w}^*$. Intuitively, as each of the individual w (except the first ones) are negative, so the slope of the estimated frontier only decreases as more terms activate, resulting in a concave production function.

Proposition 3.3. *The estimated production function $\widehat{f}(\mathbf{x})$ is concave.*

Proof. We consider each component of $\phi^{G}_{SVF-SP}(\boldsymbol{x})$. The first two terms along each input dimension are linear, and their corresponding weights in $\mathbf{w}^*$, that is, w^*_{-1} and w^*_0, are non-negative by constraint (14.4). Thus, the corresponding summands in the expression for $\widehat{f}(\mathbf{x})$ are linear, hence both convex and concave. The remaining terms are defined as the

maximum between two linear functions, the constant 0 and $x^{(j)} - t_{l_j}^{(j)}$. Since both terms are linear, they are convex, i.e., the area above their respective curves is convex. The area of the maximum between two functions corresponds to the intersection of both regions, so each term in $\phi^G_{SVF-SP}(\boldsymbol{x})$ involving a maximum is convex. These terms are then multiplied by the corresponding weights $w_{l_j}^{(j)*}$, which are non-positive by restriction (14.5), so the corresponding products are concave functions. Thus, the expression for $\widehat{f}(\mathbf{x})$ is a sum of concave functions, and is thus concave. ■

Having established these properties of the estimated production function, we now move on to the multi-output case, and consider the corresponding production technology.

3.2. The multi-output case

We now present the multioutput SVF-Splines model, which extends the above single-output model to the multi-output case following the approach by [40]. The idea is that a multi-output model can be obtained by using a transformation of the outputs and estimating weights for each of the outputs separately, so that with the same grid in the input space (which a common structure of knots for all the outputs), different estimations are obtained for each of the multiple outputs. The model is as follows:

$$
\begin{aligned}
\underset{\mathbf{w}_r,\xi_{ri}}{Min} \quad & \sum_{r=1}^{s} \| \boldsymbol{w}^{(r)} \| + C \sum_{r=1}^{s} \sum_{i=1}^{n} \xi_i^{(r)} \\
s.t. \quad & \\
& \boldsymbol{w}^{(r)} \cdot \phi^G_{SVF-SP}(\boldsymbol{x}_i) - y_i^{(r)} \leq \varepsilon + \xi_i^{(r)} \quad i = 1,\ldots,n \quad r = 1,\ldots,s \quad (19.1) \\
& y_i^{(r)} - \boldsymbol{w}^{(r)} \cdot \phi^G_{SVF-SP}(\boldsymbol{x}_i) \leq 0 \quad i = 1,\ldots,n \quad r = 1,\ldots,s \quad (19.2) \\
& W_{l_j}^{(j)(r)} \geq 0 \quad l_j = 0,\ldots,k_j \quad j = 1,\ldots,m \quad r = 1,\ldots,s \quad (19.3) \\
& w_0^{(j)(r)}, w_{-1}^{(j)(r)} \geq 0 \quad j = 1,\ldots,m \quad r = 1,\ldots,s \quad (19.4) \\
& w_k^{(j)(r)} \leq 0 \quad k = 1,\ldots,k_j \quad j = 1,\ldots,m \quad r = 1,\ldots,s \quad (19.5) \\
& \xi_i^{(r)} \geq 0 \quad i = 1,\ldots,n \quad r = 1,\ldots,s \quad (19.6)
\end{aligned}
\tag{19}
$$

Here, $\phi^G_{SVF-SP}(\boldsymbol{x})$ is defined as in the single-output model by (15), whereas there is a $\mathbf{w}^{(r)}$ associated with each output. Correspondingly, the $W$are defined by:

$$
W_{l_j}^{(j)(r)} = \sum_{k=0}^{l_j} w_k^{(j)(r)}, \quad l_j = 0,\ldots,k_j, \quad j = 1,\ldots,m, \quad r = 1,\ldots,s. \tag{20}
$$

In this setting, once model (19) is solved yielding optimal values $\mathbf{w}^*$ and $\boldsymbol{\xi}^*$, we define the rth component of the output vector corresponding to input profile$\boldsymbol{x}$ by:

$$
\widehat{f}^{(r)}(\boldsymbol{x}) = \widehat{y}^{(r)}{}_{SVF-SP}(\boldsymbol{x}) = \boldsymbol{w}^{(r)*} \cdot \phi^G_{SVF-SP}(\boldsymbol{x}). \tag{21}
$$

This defines the multi-output production frontier $\widehat{\mathbf{f}}(\mathbf{x})$. The corresponding production possibility set or technology is defined as the set of those collections of inputs and outputs whose outputs lie below the production frontier:

$$
\widehat{T}_{SVF-SP} := \left\{ (\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{m+s}_{+} : \boldsymbol{y} \leq \widehat{\boldsymbol{f}}(\boldsymbol{x}) \right\} \tag{22}
$$

We now prove that $\widehat{T}_{SVF-SP}$ satisfies data envelopment, free disposability in inputs and outputs and convexity. With these definitions, we can extend the results above to the multi-output estimator. The proofs are very similar to the ones in [37] Lemma 1, Propositions 1, 2, 3, 4. As in the single-output model, the hyperparameters C, ε and d are estimated by five-fold cross-validation.

Proposition 3.4. *For all $i = 1,\ldots,n$, we have $(\mathbf{x}_i, \mathbf{y}_i) \in \widehat{T}_{SVF-SP}$.*

Proof. Follows by constraint (19.2) and the definition of the technology. ■

Proposition 3.5. *$\widehat{T}_{SVF-SP}$ satisfies free disposability in inputs and outputs.*

Proof. Let $(\mathbf{x}, \mathbf{y}_x) \in \widehat{T}_{SVF-SP}$ and $(\mathbf{z}, \mathbf{y}_z)$ satisfy $\mathbf{z} \geq \mathbf{x}$ and $\mathbf{y}_z \leq \mathbf{y}_x$. Then, by Proposition 3.2 we have $\widehat{f}^{(r)}(\mathbf{z}) \geq \widehat{f}^{(r)}(\mathbf{x})$ for each component of $\widehat{\mathbf{f}}(\mathbf{x})$. Since $\mathbf{y}_z \leq \mathbf{y}_x$, we have that $\mathbf{y}_z \leq \mathbf{y}_x \leq \widehat{\mathbf{f}}(\mathbf{x}) \leq \widehat{\mathbf{f}}(\mathbf{z})$, so that $(\mathbf{z}, \mathbf{y}_z) \in \widehat{T}_{SVF-SP}$ and $\widehat{T}_{SVF-SP}$ satisfies free disposability. ■

Proposition 3.6. *$\widehat{T}_{SVF-SP}$ is convex.*

Proof. By Proposition 3.3, each component of $\widehat{\mathbf{f}}(\mathbf{x})$ is concave. Thus, so is $\widehat{\mathbf{f}}(\mathbf{x})$. Now, ([24], p. 81) shows that a function $\widehat{\mathbf{f}}(\mathbf{x})$ is concave if and only if its hypograph is a convex set, where the hypograph is $HG_{\widehat{\mathbf{f}}} = \{(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{m+s} : \mathbf{y} \leq \widehat{\mathbf{f}}(\mathbf{x})\}$. In this context, the production technology $\widehat{T}_{SVF-SP} = \left\{ (\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{m+s}_{+} : \boldsymbol{y} \leq \widehat{\boldsymbol{f}}(\boldsymbol{x}) \right\} = HG_{\widehat{\boldsymbol{f}}} \cap \mathbb{R}^{m+s}_{+}$ is the intersection of the hypograph and the non-negative quadrant of $\mathbb{R}^{m+s}$, and both these sets are convex. Therefore, since the intersection of convex sets is a convex set, the production technology $\widehat{T}_{SVF-SP}$ is a convex set.

Proposition 3.7. *If $s = 1$, then the multi-output model coincides with the single-output model.*

Proof. Clear from the formulation of the models. ■

Corollary. $\widehat{T}_{DEA} \subseteq \widehat{T}_{SVF-SP}$.

Proof. The set $\widehat{T}_{DEA}$ is, by the principle of minimal extrapolation, the intersection of all sets satisfying envelopment, free disposability and convexity. By Propositions 3.4, 3.5 and 3.6, $\widehat{T}_{SVF-SP}$ satisfies envelopment, free disposability and convexity. Thus, $\widehat{T}_{DEA}$ is a subset of $\widehat{T}_{SVF-SP}$. ■

3.3. Characterization of the estimated technology as a DEA-type technology

We now proceed to characterise the estimated technology as a DEA-type technology with respect to a set of "virtual points", which are not

observed in the data but rather constructed from the SVF-Splines estimations and the extreme points of the grid cells involved in the splines transformation function. In other words, we prove that the estimated technology consists of the smallest convex set enveloping these virtual points which satisfies free disposability, i.e., a DEA-type technology with these virtual points as observations.

A grid consists of cells $C_{l_1 l_2 \ldots l_m}$, where $l_j = 0, \ldots, k_j$ for each input $j = 1, \ldots, m$, with corresponding lower extreme knot-point $\boldsymbol{a}_{l_1 l_2 \ldots l_m} = \left(t_{l_1}^{(1)}, t_{l_2}^{(2)}, \ldots, t_{l_m}^{(m)}\right)$. Each grid cell has 2^m extreme points and is the convex closure of its extreme points. The output values estimated by SVF-Splines at each extreme point of a grid cell is given by $\widehat{\boldsymbol{f}}(\boldsymbol{a}_{l_1 l_2 \ldots l_m}) = \widehat{\boldsymbol{y}}_{SVF-SP}(\boldsymbol{a}_{l_1 l_2 \ldots l_m})$, and the set of all pairs $(\boldsymbol{a}_{l_1 l_2 \ldots l_m}, \widehat{\boldsymbol{f}}(\boldsymbol{a}_{l_1 l_2 \ldots l_m}))$ forms the virtual data of $\widehat{T}_{SVF-SP}$. In other words, if we let A be the matrix containing the inputs $\boldsymbol{a}_{l_1 l_2 \ldots l_m}$ as columns and $\widehat{\boldsymbol{f}}(A)$ be the matrix of corresponding estimated outputs, we have:

$$\widehat{T}_{SVF-SP-DEA} = \left\{(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}_+^{m+s} : \boldsymbol{x} \geq A\lambda, \boldsymbol{y} \leq \widehat{\boldsymbol{f}}(A)\lambda, \lambda \geq \boldsymbol{0}, \lambda \boldsymbol{1} = 1\right\} \tag{23}$$

By the properties of DEA, this set is the smallest set satisfying envelopment of the "virtual points" $(\boldsymbol{a}_{l_1 l_2 \ldots l_m}, \widehat{\boldsymbol{f}}(\boldsymbol{a}_{l_1 l_2 \ldots l_m}))$ determined by SVF-Splines, free disposability in inputs and outputs, and convexity. We now prove the following equality:

Proposition 4. *$\widehat{T}_{SVF-SP-DEA} = \widehat{T}_{SVF-SP}$. In other words, $\widehat{T}_{SVF-SP}$ is a DEA-type production technology with respect to the virtual points $(\boldsymbol{a}_{l_1 l_2 \ldots l_m}, \widehat{\boldsymbol{f}}(\boldsymbol{a}_{l_1 l_2 \ldots l_m}))$ for each $l_1 = 0, \ldots, k_1; \ldots; l_m = 0, \ldots, k_m$.*

Proof. Recall the definition of the technology estimated by SVF-Splines (22):

$$\widehat{T}_{SVF-SP} := \left\{(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}_+^{m+s} : \boldsymbol{y} \leq \widehat{\boldsymbol{f}}(\boldsymbol{x})\right\}.$$

We now prove that these two characterizations (22) and (23) of the production technology coincide. By Propositions 3.5 and 3.6, we have that $\widehat{T}_{SVF-SP}$ is a production possibility set satisfying free disposability of inputs and outputs and convexity. Regarding envelopment of the virtual points, we have proved envelopment of the original data in Proposition 3.4, but now the data defining the technology are not the original data but instead the virtual data given by $\widehat{\boldsymbol{f}}(\boldsymbol{x})$. These points are also contained in $\widehat{T}_{SVF-SP}$ by the definition of the technology, thus $\widehat{T}_{SVF-SP}$ also envelops the virtual data $(\boldsymbol{a}_{l_1 l_2 \ldots l_m}, \widehat{\boldsymbol{f}}(\boldsymbol{a}_{l_1 l_2 \ldots l_m}))$ for each extreme of the grid. Therefore, by the principle of minimal extrapolation, we have $\widehat{T}_{SVF-SP-DEA} \subseteq \widehat{T}_{SVF-SP}$.

For the reverse inclusion, we consider an arbitrary $(\boldsymbol{x}, \boldsymbol{y}) \in \widehat{T}_{SVF-SP}$, and we will prove that $(\boldsymbol{x}, \boldsymbol{y}) \in \widehat{T}_{SVF-SP-DEA}$. The input profile $\boldsymbol{x}$ belongs to a cell of the grid, which we denote by C, and is defined by a set of extreme points $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_{2^m}$. In particular, C is the convex closure of these points. As $\boldsymbol{x} \in C$, it can be written as a convex linear combination of its corner points, that is, there exists $\lambda \geq \boldsymbol{0}$ with $\sum_{\nu=1}^{2^m} \lambda_\nu = 1$ such that $\boldsymbol{x} = \sum_{\nu=1}^{2^m} \boldsymbol{\alpha}_\nu \lambda_\nu$ We now show that $\boldsymbol{y} \leq \widehat{\boldsymbol{f}}(A)\lambda$. Now, $\boldsymbol{f}(\boldsymbol{x})$ is a piecewise linear function, which is linear within the confines of each cell of the grid. Thus, since $\boldsymbol{x} \in C$, we have $\widehat{\boldsymbol{f}}(\boldsymbol{x}) = \widehat{\boldsymbol{f}}\left(\sum_{\nu=1}^{2^m} \boldsymbol{\alpha}_\nu \lambda_\nu\right) = \sum_{\nu=1}^{2^m} \widehat{\boldsymbol{f}}(\boldsymbol{\alpha}_\nu)\lambda_\nu = \widehat{\boldsymbol{f}}(A)\lambda$. As, by assumption, $(\boldsymbol{x}, \boldsymbol{y}) \in \widehat{T}_{SVF-SP}$, we have that $\boldsymbol{y} \leq \widehat{\boldsymbol{y}}_{SVF-SP}(\boldsymbol{x}) = \widehat{\boldsymbol{f}}(\boldsymbol{x}) = \widehat{\boldsymbol{f}}(A)\lambda$. Thus, $\widehat{T}_{SVF-SP} \subseteq \widehat{T}_{SVF-SP-DEA}$, and equality follows. ■

3.4. Measures of efficiency using SVF-Splines

We will use this characterization of the estimated technology $\widehat{T}_{SVF-SP}$ as a DEA-like technology with respect to a set of virtual points to adapt some of the measures of efficiency available in the literature to this context. We adapt the radial measures, both input and output oriented, as well as the models defining the Directional Distance Function and the Weighted Additive Measure. With respect to the virtual points $(\boldsymbol{a}_{l_1 l_2 \ldots l_m}, \widehat{\boldsymbol{f}}(\boldsymbol{a}_{l_1 l_2 \ldots l_m}))$ defined above, the output-oriented radial measure can be calculated using the following model [5]:

$$\begin{aligned}\psi(\boldsymbol{x}, \boldsymbol{y}) = \max\{\psi \in \mathbb{R} : (\boldsymbol{x}, \psi\boldsymbol{y}) \in \widehat{T}_{SVF-SP}\} = \max\{\psi : \boldsymbol{x} \geq A\lambda, \psi\boldsymbol{y} \\ \leq \widehat{\boldsymbol{f}}(A)\lambda, \lambda \geq \boldsymbol{0}, \lambda\boldsymbol{1} = 1\}\end{aligned} \tag{24}$$

Similarly, the input-oriented radial measure is calculated by solving model [5]:

$$\begin{aligned}\theta(\boldsymbol{x}, \boldsymbol{y}) = \min\{\theta \in \mathbb{R} : (\theta\boldsymbol{x}, \boldsymbol{y}) \in \widehat{T}_{SVF-SP}\} = \min\{\theta : \theta\boldsymbol{x} \geq A\lambda, \boldsymbol{y} \leq \widehat{\boldsymbol{f}}(A)\lambda, \lambda \\ \geq \boldsymbol{0}, \lambda\boldsymbol{1} = 1\}\end{aligned} \tag{25}$$

The directional distance function (DDF), with directional vector $\mathbf{g} = (\mathbf{g}^-, \mathbf{g}^+)$ [7] is the solution to the following model:

$$\begin{aligned}\beta(\boldsymbol{x}, \boldsymbol{y}) &= \max\{\beta \in \mathbb{R} : (\boldsymbol{x} - \beta\boldsymbol{g}^-, \boldsymbol{y} + \beta\boldsymbol{g}^+) \in \widehat{T}_{SVF-SP}\} \\ &= \max\{\beta : \boldsymbol{x} - \beta\boldsymbol{g}^- \geq A\lambda, \boldsymbol{y} + \beta\boldsymbol{g}^+ \leq \widehat{\boldsymbol{f}}(A)\lambda, \lambda \geq \boldsymbol{0}, \lambda\boldsymbol{1} = 1\}\end{aligned} \tag{26}$$

The weighted additive (WA) model [23] is calculated in the case of resorting to the new approach as:

$$\begin{aligned}WA(\boldsymbol{x}, \boldsymbol{y}) &= \max\{\boldsymbol{\rho}^-\boldsymbol{s}^- + \boldsymbol{\rho}^+\boldsymbol{s}^+ \in \mathbb{R} : (\boldsymbol{x} - \boldsymbol{s}^-, \boldsymbol{y} + \boldsymbol{s}^+) \in \widehat{T}_{SVF-SP}, (\boldsymbol{s}^-, \boldsymbol{s}^+) \in \mathbb{R}_+^{m+s}\} \\ &= \max\{\boldsymbol{\rho}^-\boldsymbol{s}^- + \boldsymbol{\rho}^+\boldsymbol{s}^+ : \boldsymbol{x} - \boldsymbol{s}^- \geq A\lambda, \boldsymbol{y} + \boldsymbol{s}^+ \leq \widehat{\boldsymbol{f}}(A)\lambda, \lambda \geq \boldsymbol{0}, \lambda\boldsymbol{1} = 1, \boldsymbol{s}^-, \boldsymbol{s}^+ \geq \boldsymbol{0}\}\end{aligned} \tag{27}$$

These linear models allow for the estimation of the efficiency of an input-output bundle with respect to the SVF-Splines estimated production technology. They can further be extended to calculate ε-insensitive technical efficiency, as in [37], by substituting the terms $\widehat{\boldsymbol{f}}(A)$ by $\widehat{\boldsymbol{f}}(A) - \varepsilon$ in the second restriction of each of the models above. For example, the ε-insensitive radial output efficiency would be calculated using:

$$\psi(\boldsymbol{x}, \boldsymbol{y}) = \max\{\psi : \boldsymbol{x} \geq A\lambda, \psi\boldsymbol{y} \leq (\widehat{\boldsymbol{f}}(A) - \varepsilon)\lambda, \lambda \geq \boldsymbol{0}, \lambda\boldsymbol{1} = 1\} \tag{28}$$

The rest of the models are analogous.

These models, however, involve many virtual points ($(d+2)^m$). In practice, we often substitute $\boldsymbol{a}_{l_1 \ldots l_m}$ by the original data to obtain a simpler model, based only on the set of points $(\boldsymbol{x}_i, \widehat{\boldsymbol{f}}(\boldsymbol{x}_i))$, $i = 1, \ldots, n$, which is computationally less expensive. The radial output-oriented model would then be calculated using the following model, with the remaining measures being analogous:

$$\psi(\boldsymbol{x}, \boldsymbol{y}) = \max\{\psi : \boldsymbol{x} \geq X\lambda, \psi\boldsymbol{y} \leq f(X)\lambda, \lambda \geq \boldsymbol{0}, \lambda\boldsymbol{1} = 1\} \tag{29}$$

4. Computational experience

In this section, we assess the quality of the SVF-Splines method by comparing it with other techniques. In particular, we perform a simulated experiment where we compare the performance of SVF-Splines

with first DEA and then CSVF using different production technologies with multiple inputs and a single output, within a simulated experience extracted from [27].

The details of the simulated production functions appear in Table 1. The simulated production functions are Cobb-Douglas functions, which are widely known in the economic literature. Each scenario uses a different number of inputs, and the exponent associated with each input indicates the level of theoretical marginal contribution of each variable to the output. The exponents of the inputs add up to 0.5 in every case, which is related to non-increasing returns to scale. The input values were obtained independently and identically distributed from $Uni[1,10]$. The observed output value is then calculated by multiplying by an extra inefficiency term e^{-u}, with $u \sim \exp(1/3)$.

Each scenario was simulated with sample sizes $n = 30, 50, 70, 100$. Each combination of scenario and sample size was then replicated 50 times. The performance of each algorithm was measured using the standard Mean Squared Error (MSE) between the real, unobserved production frontier and the estimated production frontier over the 50 trials, which is calculated as $\sum_{t=1}^{50}\sum_{i=1}^{n}\left(f(\boldsymbol{x}_i^t) - \widehat{f}(\boldsymbol{x}_i^t)\right)^2/50n$, as well as the corresponding Bias, with formulation $\sum_{t=1}^{50}\sum_{i=1}^{n}|f(\boldsymbol{x}_i^t) - \widehat{f}(\boldsymbol{x}_i^t)|/50n$, where the superscript t corresponds to the trial considered.

The experiments were performed in the Scientific Computation Cluster at the Miguel Hernandez University of Elche, which has a Supermicro SYS-1029GQ-TRT node, with two Intel(R) Xeon(R) Gold 6242R CPU @ 3.10 GHz processors, 80 cores and 768 GB of RAM. The algorithm was programmed in Python and CPLEX v20.1.0 was used for solving the optimization problems.

4.1. Comparison between DEA and SVF-Splines

We first compare the SVF-Splines methodology with standard DEA via the MSE and Bias scores obtained in the simulated scenarios. Table 2 reports the comparison between these two methods with respect to both MSE and Bias for each combination of scenario and sample size. The first two columns indicate the scenario (which corresponds to the number of inputs) and the sample size. The next three columns refer to the MSE associated with the DEA and SVF-Splines estimators, and the relative difference between the two methods. The final three columns indicate the Bias of the techniques and their relative difference.

Regarding the hyperparameters C, ε and d involved in SVF-Splines, we tune them using a five-fold cross validation procedure, which evaluates which combination of values yields a better estimator, as measured by the out-of-sample MSE at each fold. The sets of possible values which we consider for the hyperparameters are $C \in \{0.001, 0.01, 0.05, 0.1, 0.5, 1, 2, 5, 10, 100\}$, $\varepsilon \in \{0, 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1\}$, whereas the number of partitions d along each input dimension depends on the number n of DMUs via $d = 0.1{\cdot}h{\cdot}n$, rounded to the nearest integer, with $h = 1, \ldots, 10$.

We observe that SVF-Splines outperforms DEA in every scenario, with improvements in MSE of up to 95 % and up to 77 % in the case of Bias. The relative improvements increase as the sample size increases (except in the scenario with a single input). As the dimensionality of the problem increases, the MSE and Bias of SVF-Splines grows slower than those of DEA. These results indicate that SVF-Splines is capable of estimating production functions closer to the unobserved theoretical production function than DEA, so that we can conclude that SVF-Splines seems not overfit to the data as much as DEA. In the case of DEA, this overfitting could be attributed to the principle of minimal extrapolation (see, for example, [14]).

Finally, we point out that the new approach (SVF-splines) has demonstrated superior performance over the standard DEA approach in terms of MSE and bias within a finite-sample analysis (with $n \leq 200$). However, claiming complete superiority would be premature. It is important to consider various properties, such as consistency, from a statistical perspective. While consistency has been extensively studied for the DEA estimator (see, for example, [20]), a similar analysis is lacking for SVF-splines. Consequently, our comparison with the standard DEA model is confined to finite-sample analysis and our findings may be particularly relevant in scenarios with limited data sample sizes. Notably, a detailed analysis involving a greater number of DMUs and variables exceeds the scope of this paper but presents a potential avenue for future research.

4.2. Comparison between SVF-Splines and CSVF

In addition to standard DEA, we compare SVF-Splines with Convexified Support Vector Frontiers (CSVF). Table 3 shows those comparisons which could be performed, which were not as extensive as in the previous comparison due to the high computational cost with CSVF, as described in Table 4. The structure of Table 3 is analogous to that of Table 2. The same set of potential hyperparameter values described in the previous section was used in the Cross-Validation process of both CSVF and SVF-Splines.

We can observe in Table 3 that, while in the single-input case, CSVF performs slightly better than SVF-Splines, as soon as there are at least two inputs, SVF-Splines commits a smaller MSE and Bias than CSVF. This could be due to the fact that SVF-Splines directly estimates a convex production function while CSVF first estimates a stepwise function which is later convexified.

We now discuss the computational workload associated with CSVF and SVF-Splines. Table 4 reports the average time spent by both SVF-Splines and CSVF in the overall Cross-Validation process (CV), as well as the time used to create the Best Model (BM) and the time spent to Solve the Best Model (SBM) on average in each configuration. We can observe how CSVF is a very computationally expensive method. In particular, CSVF requires a long time to execute as soon as there are at least 2 inputs, and it scales much slower as the number of inputs and/or DMUs increases.

In particular, we can see how, already with only two inputs and 70 DMUs, the computational time needed by CSVF is around 6.5 times that required by the new approach, and this ratio only increases as the number of DMUs and inputs increase. With the largest setting solved by CSVF, SVF-Splines can solve the problem over 70 times faster than CSVF.

Table 1
Data generating processes used.

Scenario/ Num. Inputs	Data Generating Process ($f(\boldsymbol{x})\cdot e^{-u}$)
1	$x_1^{0.5}\cdot e^{-u}$
2	$x_1^{0.4}\cdot x_2^{0.1}\cdot e^{-u}$
3	$x_1^{0.3}\cdot x_2^{0.1}\cdot x_3^{0.1}\cdot e^{-u}$
4	$x_1^{0.3}\cdot x_2^{0.1}\cdot x_3^{0.08}\cdot x_4^{0.02}\cdot e^{-u}$
5	$x_1^{0.3}\cdot x_2^{0.1}\cdot x_3^{0.08}\cdot x_4^{0.01}\cdot x_5^{0.01}\cdot e^{-u}$
6	$x_1^{0.3}\cdot x_2^{0.1}\cdot x_3^{0.08}\cdot x_4^{0.01}\cdot x_5^{0.006}\cdot x_6^{0.004}\cdot e^{-u}$
9	$x_1^{0.3}\cdot x_2^{0.1}\cdot x_3^{0.08}\cdot x_4^{0.005}\cdot x_5^{0.004}\cdot x_6^{0.001}\cdot x_7^{0.005}\cdot x_8^{0.004}\cdot x_9^{0.001}\cdot e^{-u}$
12	$x_1^{0.2}\cdot x_2^{0.075}\cdot x_3^{0.025}\cdot x_4^{0.05}\cdot x_5^{0.05}\cdot x_6^{0.08}\cdot x_7^{0.005}\cdot x_8^{0.004}\cdot x_9^{0.001}\cdot x_{10}^{0.005}\cdot x_{11}^{0.004}\cdot x_{12}^{0.001}\cdot e^{-u}$
15	$x_1^{0.15}\cdot x_2^{0.025}\cdot x_3^{0.025}\cdot x_4^{0.05}\cdot x_5^{0.025}\cdot x_6^{0.025}\cdot x_7^{0.05}\cdot x_8^{0.05}\cdot x_9^{0.08}\cdot x_{10}^{0.005}\cdot x_{11}^{0.004}\cdot x_{12}^{0.001}\cdot x_{13}^{0.005}\cdot x_{14}^{0.004}\cdot x_{15}^{0.001}\cdot e^{-u}$

Table 2
Comparison between DEA and SVF-Splines according to MSE and Bias.

Scenario	Sample Size	Mean Squared Error			Bias		
		DEA	SVF-SP	Improvement	DEA	SVF-SP	Improvement
1	**30**	0.02	0.016	17 %	0.104	0.097	7 %
	50	0.011	0.01	13 %	0.076	0.072	5 %
	70	0.007	0.006	11 %	0.059	0.057	4 %
	100	0.009	0.008	12 %	0.069	0.067	4 %
	200	0.000	0.000	3 %	0.01	0.009	2 %
2	**30**	0.074	0.035	52 %	0.198	0.135	32 %
	50	0.048	0.019	60 %	0.16	0.102	37 %
	70	0.031	0.012	60 %	0.127	0.08	38 %
	100	0.024	0.01	61 %	0.109	0.069	38 %
	200	0.001	0.001	34 %	0.026	0.018	29 %
3	**30**	0.132	0.05	62 %	0.285	0.174	40 %
	50	0.098	0.034	65 %	0.242	0.139	43 %
	70	0.07	0.022	70 %	0.203	0.105	49 %
	100	0.056	0.013	76 %	0.177	0.083	54 %
	200	0.003	0.001	63 %	0.045	0.023	49 %
4	**30**	0.2	0.064	67 %	0.342	0.191	44 %
	50	0.156	0.042	74 %	0.3	0.149	51 %
	70	0.114	0.025	78 %	0.252	0.119	53 %
	100	0.095	0.018	81 %	0.231	0.096	59 %
	200	0.006	0.001	79 %	0.060	0.024	60 %
5	**30**	0.312	0.121	61 %	0.468	0.292	38 %
	50	0.239	0.063	74 %	0.397	0.209	48 %
	70	0.225	0.058	74 %	0.387	0.2	49 %
	100	0.176	0.037	78 %	0.343	0.163	52 %
	200	0.009	0.000	82 %	0.059	0.022	62 %
6	**30**	0.313	0.098	68 %	0.465	0.256	45 %
	50	0.272	0.073	73 %	0.424	0.211	51 %
	70	0.227	0.05	78 %	0.385	0.17	56 %
	100	0.2	0.037	82 %	0.359	0.143	60 %
	200	0.012	0.001	90 %	0.084	0.026	69 %
9	**30**	0.392	0.129	68 %	0.52	0.286	45 %
	50	0.368	0.095	74 %	0.501	0.24	52 %
	70	0.337	0.067	80 %	0.475	0.203	57 %
	100	0.318	0.052	84 %	0.46	0.173	62 %
	200	0.023	0.001	94 %	0.114	0.028	76 %
12	**30**	0.413	0.135	68 %	0.534	0.296	45 %
	50	0.416	0.107	74 %	0.539	0.257	52 %
	70	0.396	0.084	79 %	0.526	0.229	57 %
	100	0.378	0.064	83 %	0.515	0.2	61 %
	200	0.030	0.002	94 %	0.135	0.031	77 %
15	**30**	0.439	0.165	63 %	0.566	0.34	40 %
	50	0.421	0.131	69 %	0.543	0.295	46 %
	70	0.427	0.093	78 %	0.55	0.245	56 %
	100	0.408	0.078	81 %	0.537	0.221	59 %
	200	0.034	0.002	95 %	0.147	0.033	77 %

Table 3
Comparison between SVF-Splines and CSVF according to MSE and Bias.

Scenario	Sample Size	Mean Squared Error			Bias		
		SVF-SP	CSVF	Improvement	SVF-SP	CSVF	Improvement
1	**30**	0.016	0.016	0 %	0.097	0.09	−8 %
	50	0.01	0.008	−25 %	0.072	0.066	−9 %
	70	0.006	0.005	−20 %	0.057	0.053	−8 %
	100	0.008	0.008	0 %	0.067	0.064	−5 %
2	**30**	0.035	0.05	30 %	0.135	0.164	18 %
	50	0.019	0.03	37 %	0.102	0.125	18 %
	70	0.012	0.021	43 %	0.08	0.105	24 %
	100	0.01	0.017	41 %	0.069	0.087	21 %
3	**30**	0.005	0.009	44 %	0.089	0.077	13 %
	50	0.034	0.058	41 %	0.139	0.192	28 %

Furthermore, the estimated MSE and Bias obtained are lower for the SVF-Splines than for the CSVF algorithm.

Computational times used by SVF-Splines with more inputs and DMUs are reported in Fig. 3. It can be seen that, even in the most expensive case of 100 DMUs with 15 inputs, the computational load is lower than that of CSVF with 3 inputs and 30 DMUs.

The computational burden can be observed to be more in that the creation of models, and in particular the Best Model (BM), which is much more computationally expensive for CSVF than for SVF-Splines. The Solution of the Best Model (SBM), despite the size of the models and how expensive they are to construct, is very fast in both cases. We can attribute this improvement of the new approach to the grid associated to the transformation function, which for CSVF requires a number of parameters which is exponential in the number of inputs m ($\approx md^m$),

Table 4
Execution time of the SVF and SVF-Splines algorithms.

Scenario	Sample Size	SVF-SP			CSVF		
		CV	BM	SBM	CV	BM	SBM
1	30	161.02	0.02	0.0	137.68	0.04	0.0
	50	196.22	0.0	0.02	157.02	0.06	0.0
	70	242.26	0.12	0.04	195.46	0.2	0.0
	100	325.68	0.06	0.02	262.3	0.24	0.0
2	30	188.68	0.08	0.0	324.18	0.4	0.0
	50	251.7	0.06	0.02	925.4	2.12	0.0
	70	335.52	0.1	0.06	2166.32	5.82	0.04
	100	489.0	0.14	0.02	5401.16	13.74	0.06
3	30	208.66	0.04	0.0	4697.6	7.64	0.06
	50	299.28	0.06	0.0	21,764.54	46.02	0.18

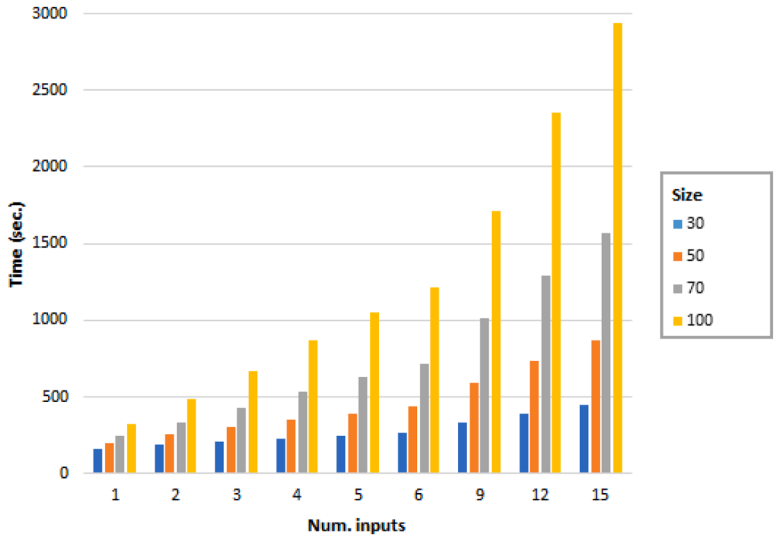

Fig. 3. SVF-Splines runtime according to sample size and number of inputs.

as well as corresponding restrictions. On the other hand, the transformation function related to SVF-Splines has a linear number of parameters in the inputs ($\approx m(d+3)$).

From this computational experiment, we can conclude that SVF-Splines is a method which reduces the overfitting present in DEA, obtaining estimations closer to the unobserved efficient frontier. It furthermore improves upon the results obtained by CSVF, with much lower computational time required and lower MSE and Bias as soon as the number of inputs is at least 2.

5. Empirical illustration

In this section, we present an illustration of the results that can be obtained by SVF-Splines in an empirical database from the literature.

Table 5
Descriptive statistics of the Spanish tax offices dataset.

	x1	x2	x3	y1	y2
Average	32	91	57.6	10.45	8.91
Median	23	70	22.8	7.39	6.59
Std. Dev.	30	74	88.3	9.49	9.60
Max.	191	368	420.2	49.27	60.02
90 %	60	194	176.7	20.65	18.97
75 %	39	109	63.7	10.71	10.18
25 %	17	43	8.9	5.30	3.62
10 %	11	32	5.0	3.88	2.14
Min.	7	21	0.8	2.51	1.48

This database consists of 44 regional tax offices in Spain, which oversee tax collection in almost all Spanish provinces. The data is from 2011, and the database was used in [3]. We calculate the efficiency scores using the efficiency measures introduced above (Section 3.4), that is, the output and input-oriented radial models (24), (25), the directional distance function (26) with directional vector $\boldsymbol{g} = (\boldsymbol{x},\boldsymbol{y})$, and the Weighted Additive (27) with the weights corresponding to the Range Adjusted Measure, that is, $\rho^{-(j)} = \frac{1}{(m+s)R_j^-}$ and $\rho^{+(r)} = \frac{1}{(m+s)R_r^+}$, where R_j^- is the range of input j and R_r^+ is the range of values of output r.

The dataset used contains three variables considered as inputs, and two outputs. There are two inputs related to labour: the number of tax inspectors and specialists (x_1) and the number of workers in the rest of the workforce (x_2). We also consider as an input the number of successful complaints by taxpayers against the tax authority (x_3). This variable indicates an output that a manager should desire to minimize, sometimes called a bad output in the literature. Thus, it is considered as an input in the model, as in [19,25]. The two outputs considered correspond to the two main taxes collected by the tax offices: the inheritance and gift tax (y_1) and the real estate transfer tax settlements processed (y_2).

The data was normalized so that the values of the variables lie between 0 and 1 by dividing the values of each variable by the maximum attained value. This does not change the values of the measures of efficiency, which are all units-invariant. Table 5 shows descriptive statistics of the dataset before normalization.

The sets of potential hyperparameter values considered in the five-

Table 6
Estimated efficiencies in the Spanish tax offices dataset.

DMU	Radial Output			Radial Input			Weighted Additive (RAM)			Directional Distance Function		
	DEA	SVF-SP	ε-insensitive efficient	DEA	SVF-SP	ε-insensitive efficient	DEA	SVF-SP	ε-insensitive efficient	DEA	SVF-SP	ε-insensitive efficient
ALMERIA	1.4075	1.5825	No	0.6966	0.5892	No	0.1046	0.1118	No	0.4075	0.4512	No
CADIZ	1.1414	1.2755	No	0.8866	0.7504	No	0.0649	0.0824	No	0.1297	0.2305	No
CORDOBA	1.5738	1.9151	No	0.4969	0.4468	No	0.0692	0.0745	No	0.3691	0.5684	No
GRANADA	1.2926	1.4025	No	0.7888	0.6659	No	0.0619	0.0716	No	0.2899	0.3494	No
HUELVA	1.4185	1.5963	No	0.7937	0.5286	No	0.0355	0.0416	No	0.4185	0.5416	No
JAEN	1.3502	1.6406	No	0.7441	0.5945	No	0.0480	0.0540	No	0.3217	0.4803	No
MALAGA	1.0000	1.0935	Yes	1.0000	0.9022	Yes	0.0000	0.0625	Yes	0.0000	0.0892	Yes
SEVILLA	1.1930	1.2155	No	0.8076	0.8012	No	0.0523	0.0808	No	0.1402	0.1947	No
HUESCA	1.0000	1.6517	Yes	1.0000	0.9511	Yes	0.0000	0.0202	Yes	0.0000	0.4394	Yes
TERUEL	1.0000	1.2668	Yes	1.0000	1.0000	Yes	0.0000	0.0141	Yes	0.0000	0.0000	Yes
ZARAGOZA	1.0000	1.0666	Yes	1.0000	0.9341	Yes	0.0000	0.0233	Yes	0.0000	0.0401	Yes
OVIEDO	1.0000	1.0000	Yes	1.0000	1.0000	Yes	0.0000	0.0247	Yes	0.0000	0.0000	Yes
BALEARES	1.2670	1.8781	No	0.8114	0.5058	No	0.0581	0.0625	No	0.2285	0.3861	No
CANTABRIA	1.2319	1.3858	Yes	0.8058	0.6540	Yes	0.0243	0.0293	Yes	0.2049	0.3531	Yes
ALBACETE	1.2181	1.3919	Yes	0.8598	0.6805	Yes	0.0135	0.0218	Yes	0.1867	0.3651	Yes
CIUDAD REAL	1.0000	1.0000	Yes	1.0000	0.9996	Yes	0.0000	0.0032	Yes	0.0000	0.0000	Yes
CUENCA	1.2651	1.9234	No	0.8621	0.6563	No	0.0178	0.0307	No	0.2398	0.6550	No
GUADALAJARA	1.0364	1.7132	Yes	0.9778	0.6335	Yes	0.0083	0.0233	Yes	0.0345	0.5594	Yes
TOLEDO	1.1788	1.2682	Yes	0.7632	0.7387	Yes	0.0211	0.0260	Yes	0.1512	0.2588	Yes
AVILA	1.0000	1.2552	Yes	1.0000	1.0000	Yes	0.0000	0.0152	Yes	0.0000	0.0000	Yes
BURGOS	1.1882	1.2954	Yes	0.8444	0.7433	Yes	0.0172	0.0225	Yes	0.1672	0.2772	Yes
LEÓN	1.0000	1.0067	Yes	1.0000	0.9897	Yes	0.0000	0.0151	Yes	0.0000	0.0058	Yes
PALENCIA	1.1641	1.3862	Yes	0.8632	0.8000	Yes	0.0084	0.0205	Yes	0.1544	0.3670	Yes
SALAMANCA	1.0139	1.0910	Yes	0.9875	0.9035	Yes	0.0014	0.0142	Yes	0.0135	0.0841	Yes
SEGOVIA	1.2139	1.4089	Yes	0.8711	0.7071	Yes	0.0093	0.0228	Yes	0.1682	0.3802	Yes
SORIA	1.0000	1.8165	Yes	1.0000	1.0000	Yes	0.0000	0.0203	Yes	0.0000	0.0000	Yes
VALLADOLID	1.3574	1.5670	No	0.7649	0.6038	No	0.0305	0.0360	No	0.3362	0.4688	No
ZAMORA	1.0000	1.0805	Yes	1.0000	0.9086	Yes	0.0000	0.0162	Yes	0.0000	0.0746	Yes
BARCELONA	1.0000	1.0000	Yes	1.0000	1.0000	Yes	0.0000	0.1021	Yes	0.0000	0.0000	Yes
GIRONA	1.0000	1.1410	Yes	1.0000	0.8714	Yes	0.0000	0.0286	Yes	0.0000	0.0802	Yes
LLEIDA	1.1634	1.2723	Yes	0.8622	0.7531	Yes	0.0160	0.0240	Yes	0.1457	0.2472	Yes
TARRAGONA	1.0000	1.1972	Yes	1.0000	0.7872	Yes	0.0000	0.0238	Yes	0.0000	0.1921	Yes
BADAJOZ	1.6460	1.7975	No	0.6317	0.4785	No	0.0699	0.0750	No	0.5002	0.7027	No
CACERES	1.4216	1.6921	No	0.7855	0.5000	No	0.0245	0.0321	No	0.3863	0.5989	No
A CORUÑA	1.0000	1.0000	Yes	1.0000	1.0000	Yes	0.0000	0.0046	Yes	0.0000	0.0000	Yes
LUGO	1.0000	1.0236	Yes	1.0000	0.9710	Yes	0.0000	0.0147	Yes	0.0000	0.0206	Yes
OURENSE	2.0479	2.1611	No	0.5134	0.4196	No	0.0469	0.0514	No	0.6079	0.6995	No
PONTEVEDRA	1.2544	1.3647	No	0.7844	0.7178	No	0.0545	0.0672	No	0.2519	0.2821	No
LA RIOJA	1.0000	1.3953	Yes	1.0000	0.7593	Yes	0.0000	0.0269	Yes	0.0000	0.3182	Yes
MADRID	1.0000	1.0000	Yes	1.0000	1.0000	Yes	0.0000	0.0105	Yes	0.0000	0.0000	Yes
MURCIA	1.2786	1.3135	No	0.7771	0.7422	No	0.1426	0.1808	No	0.2496	0.2610	No
ALICANTE	1.0000	1.2054	No	1.0000	0.8316	No	0.0000	0.1235	No	0.0000	0.1379	No
CASTELLÓN	1.6437	1.8878	No	0.6331	0.5140	No	0.0584	0.0637	No	0.5191	0.6429	No
VALENCIA	1.0000	1.0000	Yes	1.0000	0.9993	Yes	0.0000	0.0285	Yes	0.0000	0.0000	Yes
Average	1.1811	1.3779		0.8776	0.7735		0.0241	0.0432		0.1505	0.2683	
Std. dev.	0.2304	0.3143		0.1398	0.1829		0.0326	0.0362		0.1735	0.2279	
# eff. Units	19	6	27	19	7	27	19	0	27	19	9	27

fold cross-validation process were: $C \in \{0.001, 0.01, 0.05, 0.1, 0.5, 1, 2, 5, 10\}$, $\varepsilon \in \{0, 0.001, 0.005, 0.01, 0.05, 0.1, 0.2\}$. We remark that, since the values of each variable were normalized to lie between 0 and 1, a margin of 0.1 corresponds 10 % of the maximum value of each output variable, i.e., already quite a large margin.[1] The value of d was chosen within $d = 0.1 \cdot h \cdot n$ for $h = 1, \ldots, 20$, rounded to the closest integer. In other words, $d = \{4, 9, 13, 18, \ldots, 70, 75, 79, 84, 88\}$. The best hyperparameter combination was chosen as $C = 1.0$, $\varepsilon = 0.05$ and $d = 4$. This indicates an ε-insensitive margin of 5 % of the maximum along each output, with the input space being divided into $(d+2)^m = 6^3 = 216$ cells via $m(d+3) = 3 \cdot 7 = 21$ weight hyperparameters.

Table 6 then presents the efficiencies estimated by DEA and SVF-Splines with respect to each of the measures of efficiency, as well as whether each DMU is considered ε-insensitive efficient with respect to the estimated production technology. The final three rows of Table 6 indicate the average efficiency measured by each method, the corresponding standard deviation, and the number of DMUs considered efficient with respect to each algorithm and measure. We observe that SVF-Splines consistently estimates higher inefficiencies, indicating a frontier which fits the data less closely than that of DEA, which suffers from overfitting.[2] We observe that, while DEA always considers 19 units as efficient regardless of the measure, SVF-Splines considers 7 DMUs as efficient in the case of the Radial Input measure and 9 units when the Directional Distance Function is applied. Moreover, under the SVF-Splines model, 6 DMUs are considered efficient with respect to the Radial Output measure, and no DMUs are considered fully efficient with

[1] Support Vector Regression (SVR), foundational to the SVF-splines approach, depends on hyperparameters, including the margin. This margin defines the error-free region around the regression line and is crucial in SVR models. Tuning it involves considering various values, a non-trivial task. Normalizing data allows reinterpretation of margin values, with a margin of z corresponding to 100·z% of the maximum output value. For the empirical case, we chose a margin of 0.2, which seems large enough.

[2] In all cases, a higher value of the measure indicates a greater level of inefficiency, with the exception of the Radial Input measure, where the interpretation is reversed: higher values denote better technical efficiency.

respect to the Range Adjusted Measure (WA), which is capable of detecting additional sources of inefficiencies along any variables, and projects to the strongly efficient frontier.

Regarding ε-insensitive technical efficiency, 27 DMUs are considered efficient with this margin of $\varepsilon = 0.05$. This can be seen as an indication that the remaining 17 DMUs can be considered very far from efficient, whereas those 27 which are within the margin can be considered to be close to efficient with respect to this more robust notion of efficiency, even if they are not completely efficient.

Additionally, we use the Li test adapted to the production context [33], based on the Li test [22], as a tool to compare the vectors of efficiency scores estimated by each method with respect to the same measures of efficiency. This is a nonparametric statistical test for the similarity of two distributions of technical efficiency scores. We remark that the efficiencies cannot be compared between different measures, since each of them has different properties, whether in orientation, range of potential values, and other properties. Hence, we compare the efficiencies obtained, always with respect to the same measure, by both DEA and SVF-Splines.

The version of the Li test that we use requires that the efficiencies are in the output orientation. That is, that efficient units attain the value 1, and larger values indicate higher inefficiencies, without an upper bound, which is the same orientation as the output-oriented radial measure. Therefore, we transform the other measures appropriately. For the input-oriented measure, we transform it to its reciprocal $1/x$. For the DDF and RAM, which are bounded measures of inefficiency, the corresponding transformation is $1/(1-x)$. This is because, with the choices of weights used, they are inefficiency measures bounded between 0 and 1, with efficient units attaining 0. The corresponding measures of inefficiency are $1-x$, but these are oriented as an input measure. In order to orient them in the output sense as required for the Li Test, we therefore calculate their reciprocal.

We present in Fig. 4 the kernel density distributions comparing the efficiency scores estimated by DEA and SVF-Splines. We again observe that SVF-Splines estimates higher average inefficiencies. The Li test yields evidence that the distributions are indeed statistically significantly different according to this measure in the radial measures and RAM. Regarding the DDF, the p-value in question was 0.079, which does not allow for this conclusion at a significance level of 0.05, but would do so at the significance value of 0.10 also recommended by Simar and Zelenyuk.

Therefore, we can observe in this empirical illustration that SVF-Splines shows higher discriminatory power than DEA by considering fewer units as fully efficient, estimating higher average inefficiencies which show statistically significant differences compared to DEA. Furthermore, SVF-Splines adds an additional layer of classification in the difference between the estimated efficiencies and the ε-insensitive efficient units. This notion of ε-insensitivity can be seen as a more robust region of efficiency, which allows to label some DMUs as being highly inefficient.

6. Conclusions and future work

Traditional non-parametric methods for the measurement of

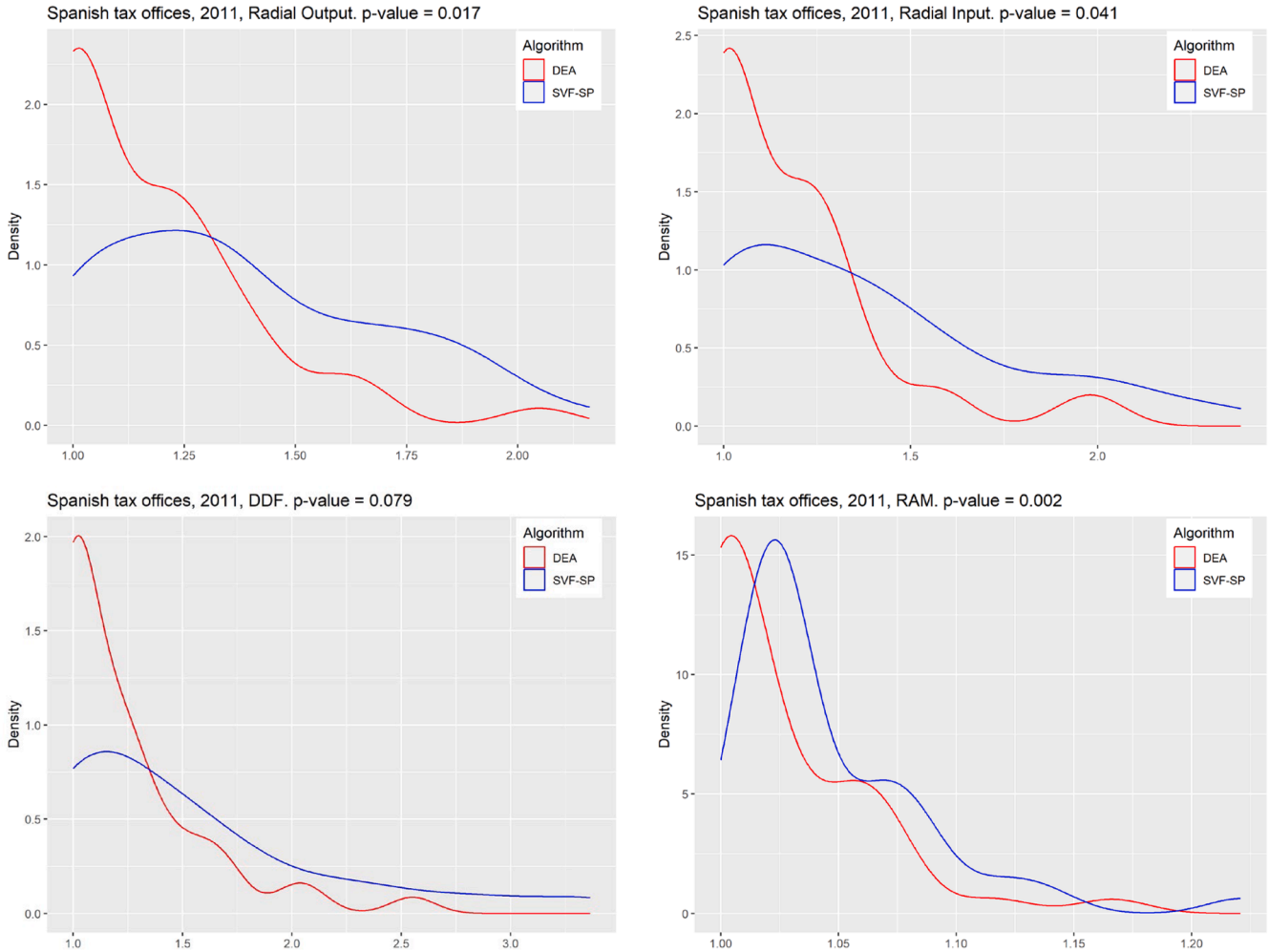

Fig. 4. Kernel density plots for different measures comparing efficiencies estimated by DEA and SVF-Splines.

efficiency such as FDH and DEA have many interesting properties, but have received criticism regarding their performance, such as that they suffer from a problem of overfitting to the observed data (see, for example, [14] or [35]), which can be attributed to the principle of minimal extrapolation and may result in lack of generalization capability (low inferential power). An area of recent interest in the measurement of efficiency is the use of machine-learning methodologies to improve the estimators by overcoming these overfitting issues and provide them with generalization capability and robustness.

One approach of particular interest is Support Vector Frontiers (SVF) ([36], 2022), which adapts Support Vector Regression to estimate stepwise production technologies satisfying the axioms of Free Disposal Hull except for minimal extrapolation. These production technologies can then be convexified to obtain convex production technologies, along the lines of DEA. However, the SVF approach presents some limitations regarding its computational time, as well as its two-stage estimation, where first a stepwise production frontier is calculated, which is later convexified.

In this paper, we have proposed an extension of Support Vector Frontiers by using a transformation function involving linear splines, that is, with a transformation function involving kernels which generate splines of order 1. The transformation function in SVF could be considered as a splines transformation of order 0. This extension, which we denote by SVF-Splines, allows for the direct estimation of convex, piecewise linear, DEA-type, production technologies, in both a single and multi-output context, with lower computational costs. This paper defines the SVF-Splines model and characterizes its corresponding production technology. The properties satisfied by DEA, except for minimal extrapolation, are established for the production possibility set estimated by SVF-Splines in a multi-input multi-output context. These properties are envelopment of the observed data, free disposability of inputs and outputs and convexity. Furthermore, the estimated set is identified with a DEA estimator defined on a set of virtual points. Subsequently, we have shown how to estimate technical inefficiency in the SVF-Splines context for a variety of standard measures of efficiency from the literature.

This contribution therefore proposes a significant improvement to the SVF estimator by using a different transformation function under the assumption of convexity. The change of kernel results in a different set of restrictions being used to guarantee the satisfaction of the microeconomic postulates.

The validity of the proposed approach is evaluated by a simulated experiment where it was compared with DEA and CSVF (the convexified version of the original SVF by Valero-Carreras, 2021, 2022). The results from the simulation show that SVF-Splines can estimate production technologies closer to the unobserved theoretical production frontier than DEA and SVF-Splines, as measured using MSE and Bias statistics. Furthermore, the computational time required by SVF-Splines is much lower than that of CSVF, which must create and solve models with a number of variables and restrictions which is exponential in the number of inputs, whereas SVF-Splines only has linearly many such variables and restrictions. Regarding the results, SVF-Splines obtained a similar performance to DEA and CSVF in the single-input single-output scenario. However, when the number of inputs is at least 2, SVF-Splines obtained improvements between 52 % and 95 % in MSE and of 32 % and 77 % in Bias with respect to DEA, and between 30 % and 44 % in MSE and between 18 % and 28 % in Bias for SVF-Splines when compared to CSVF. Regarding the comparison with CSVF, only those scenarios with at most 3 inputs and 50 DMUs were solved, since in larger scenarios the computation time required by CVSF becomes impracticable.

Furthermore, an empirical example is provided to illustrate the results that can be obtained using SVF-Splines. The database under study consists of 44 regional tax offices in Spain. From this example, it can be observed how SVF-Splines classifies a fewer DMUs as efficient than DEA and how it estimates overall smaller average efficiencies, thus identifying more sources of inefficiency. Efficiency scores are calculated with respect to the radial output and input measures, the Directional Distance Function, and the Weighted Additive model, to illustrate the variety of measures available in the literature of nonparametric efficiency estimation. In addition, comparisons between the estimated vectors of efficiencies with respect to the same measure show significant differences with respect to efficiency measures between DEA and SVF-Splines. Additionally, when the more robust concept of epsilon-insensitive efficiency is considered, the method classifies a larger number of DMUs as efficient, which can be interpreted as evidence pointing towards that those that are still labelled as inefficient through the new approach can be considered to be very far from technical efficiency.

Regarding the possible utilization by a decision-maker, while he may still be interested in a quick estimation and does not care about overfitting to the available data, DEA is a valid option. However, if the goal is to obtain more robust results and to perform inference, SVF-Splines is an option to consider. Regarding the limitations of the method, SVF-Splines requires a cross-validation procedure which involves large grid of values to establish three hyperparameters, which results in some computational load. It is still slower than DEA, which involves only linear programming. This could be further improved by narrowing down the potential ranges of values. The determination of knots may also be problematic and could be performed in other ways (for example, a non equi-distance split of each input dimension).

We finally mention various potential lines of future work, such as the use of estimators involving higher order splines (quadratic, cubic, etc.). Other types of transformation functions such as Radial Basis Functions or Gaussian kernels could also be used. We observe that the restrictions on the parameters which ensure that the production axioms hold are not the same as in SVF, and more information about which properties of SVR allow the satisfaction of the properties may be interesting, regarding other potential kernels and their associated constraints. In this paper we have focused our attention on measuring the efficiency of units in a single dataset, and the proposed algorithm could be adapted to work with panel data, and the study of whether the differences in efficiency arise from its various sources (efficiency change, scale efficiency change, technical change). Further validation using more real databases in different contexts would always be useful. Other robustness increasing methods such as relaxed support vector regression [30] or based on the directional distance function [4] could also be considered. Additional future research could entail comparative analyses among DEA-related methods, such as Supper-Efficiency DEA [2] and SBM (Slack-Based Measure) (see [34]), and the new approach (SVF-Splines) to elucidate their relative advantages and limitations in different contexts. Additionally, further research is warranted to delve into the 'black-box' mechanism underlying support vector-based approaches for constructing production frontiers. By addressing these areas, we aim to enhance our understanding of efficiency measurement techniques and contribute to the advancement of knowledge in this field.

CRediT authorship contribution statement

Nadia M. Guerrero: Writing – review & editing, Writing – original draft, Methodology, Conceptualization. **Raul Moragues:** Writing – original draft, Visualization, Formal analysis. **Juan Aparicio:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Funding acquisition. **Daniel Valero-Carreras:** Validation, Software, Data curation.

Declaration of competing interest

The authors notify that there's no financial/personal interest or belief that could affect their objectivity.

Data availability

Data will be made available on request.

Acknowledgments

The authors thank the grant PID2022–136383NB-I00 funded by MICIU/AEI/ 10.13039/501100011033 and by ERDF/EU. D. Valero thanks the grant ACIF/2020/155 (*Generalitat Valenciana*). Additionally, J. Aparicio thanks the grant PROMETEO/2021/063 funded by the Valencian Community (Spain).

References

[1] Aigner D, Lovell CAK, Schmidt P. Formulation and estimation of stochastic frontier production function models. J Econom 1977;6(1):21–37. https://doi.org/10.1016/0304-4076(77)90052-5.
[2] Andersen P, Petersen NC. A procedure for ranking efficient units in data envelopment analysis. Manage Sci 1993;39(10):1261–4. https://doi.org/10.1287/mnsc.39.10.1261.
[3] Aparicio J, Cordero JM, Díaz-Caro C. Efficiency and productivity change of regional tax offices in Spain: an empirical study using Malmquist–Luenberger and Luenberger indices. Empir Econ 2020;59(3):1403–34. https://doi.org/10.1007/s00181-019-01667-8.
[4] Arabmaldar A, Sahoo BK, Ghiyasi M. A generalized robust data envelopment analysis model based on directional distance function. Eur J Oper Res 2023;311(2):617–32. https://doi.org/10.1016/j.ejor.2023.05.005.
[5] Banker RD, Charnes A, Cooper WW. Some models for estimating technical and scale inefficiencies in data envelopment analysis. Manage Sci 1984;30(9):1078–92. https://doi.org/10.1287/mnsc.30.9.1078.
[6] Blanco V, Puerto J, Rodriguez-Chia AM. On lp-support vector machines and multidimensional kernels. J Mach Learn Res 2020;21(14):1–29. http://jmlr.org/papers/v21/18-601.html.
[7] Chambers RG, Chung Y, Färe R. Profit, directional distance functions, and nerlovian efficiency. J Optim Theory Appl 1998;98(2):351–64. https://doi.org/10.1023/A:1022637501082.
[8] Charnes A, Cooper WW, Rhodes E. Measuring the efficiency of decision making units. Eur J Oper Res 1978;2(6):429–44. https://doi.org/10.1016/0377-2217(78)90138-8.
[9] Cooper WW, Park KS, Pastor JT. RAM: a range adjusted measure of inefficiency for use with additive models, and relations to other models and measures in DEA. J Productivity Anal 1999;11(1):5–42. https://doi.org/10.1023/A:1007701304281.
[10] Daouia A, Noh H, Park BU. Data envelope fitting with constrained polynomial splines. J R Stat Soc 2016;78(1):3–30. https://doi.org/10.1111/RSSB.12098. *Series B: Statistical Methodology*.
[11] Daraio C, Simar L. Advanced robust and nonparametric methods in efficiency analysis, Vol. 4. Springer US; 2007. https://doi.org/10.1007/978-0-387-35231-2.
[12] Deprins D, Simar L, Tulkens H. Measuring labor-efficiency in post offices. Université catholique de Louvain, Center for Operations Research and Econometrics (CORE); 1984. p. 243–67. https://doi.org/10.1007/978-0-387-25534-7_16. M. Marchand, P. Pestieau, & H. Tulkens, Eds.LIDAM Reprints CORE 571.
[13] España VJ, Aparicio J, Barber X, Esteve M. Estimating production functions through additive models based on regression splines. Eur J Oper Res 2023. https://doi.org/10.1016/j.ejor.2023.06.035.
[14] Esteve M, Aparicio J, Rabasa A, Rodriguez-Sala JJ. Efficiency analysis trees: a new methodology for estimating production frontiers through decision trees. Expert Syst Appl 2020;162:113783. https://doi.org/10.1016/j.eswa.2020.113783.
[15] Farrell MJ. The measurement of productive efficiency. J R Stat Soc Ser A 1957;120(3):253. https://doi.org/10.2307/2343100.
[16] Guerrero NM, Aparicio J, Valero-Carreras D. Combining data envelopment analysis and machine learning. Mathematics 2022;10(6):909. https://doi.org/10.3390/MATH10060909.
[17] Guillen MD, Aparicio J, Esteve M. Gradient tree boosting and the estimation of production frontiers. Expert Syst Appl 2023;214:119134. https://doi.org/10.1016/J.ESWA.2022.119134.
[18] Guillen MD, Aparicio J, Esteve M. Performance evaluation of decision-making units through boosting methods in the context of free disposal hull: some exact and heuristic algorithms. Int J Inf Technol Decis Mak 2023. https://doi.org/10.1142/S0219622023500050. published online.
[19] Hailu A, Veeman TS. Environmentally sensitive productivity analysis of the Canadian pulp and paper industry, 1959-1994: an input distance function approach. J Environ Econ Manage 2000;40(3):251–74. https://doi.org/10.1006/JEEM.2000.1124.
[20] Kneip A, Park BU, Simar L. A note on the convergence of nonparametric DEA estimators for production efficiency scores. Econ Theory 1998;14(6):783–93. https://doi.org/10.1017/S0266466698146042.
[21] Kuosmanen T, Johnson AL. Data envelopment analysis as nonparametric least-squares regression. Oper Res 2010;58(1):149–60. https://doi.org/10.1287/opre.1090.0722.
[22] Li Q. Nonparametric testing of closeness between two unknown distribution functions. Econom Rev 1996;15(3):261–74. https://doi.org/10.1080/07474939608800355.
[23] Lovell CAK, Pastor JT. Units invariant and translation invariant DEA models. Operations Research Letters 1995;18(3):147–51.
[24] Madden P. Concavity and optimization in microeconomics. B. Blackwell; 1986.
[25] Mahlberg B, Sahoo BK. Radial and non-radial decompositions of Luenberger productivity indicator with an illustrative application. Int J Prod Econ 2011;131(2):721–6. https://doi.org/10.1016/J.IJPE.2011.02.021.
[26] Meeusen W, van Den Broeck J. Efficiency estimation from Cobb-Douglas production functions with composed error. Int Econ Rev (Philadelphia) 1977;18(2):435. https://doi.org/10.2307/2525757.
[27] Moragues R, Aparicio J, Esteve M. An unsupervised learning-based generalization of data envelopment analysis. Oper Res Perspect 2023;11:100284. https://doi.org/10.1016/j.orp.2023.100284.
[28] Moragues R, Aparicio J, Esteve M. Measuring technical efficiency for multi-input multi-output production processes through OneClass support vector machines: a finite-sample study. Oper Res 2023;23(3):47. https://doi.org/10.1007/s12351-023-00788-4.
[29] Olesen OB, Ruggiero J. The hinging hyperplanes: an alternative nonparametric representation of a production function. Eur J Oper Res 2022;296(1):254–66. https://doi.org/10.1016/j.ejor.2021.03.054.
[30] Panagopoulos OP, Xanthopoulos P, Razzaghi T, Şeref O. Relaxed support vector regression. Ann Oper Res 2019;276(1–2):191–210. https://doi.org/10.1007/S10479-018-2847-6/TABLES/6.
[31] Parmeter, C.F., & Racine, J.S. (2013). Smooth constrained frontier analysis. *Recent Advances and Future Directions in Causality, Prediction, and Specification Analysis: Essays in Honor of Halbert L. White Jr*, 463–88. 10.1007/978-1-4614-1653-1_18/FIGURES/6.
[32] Pedroso JP, Murata N. Support vector machines with different norms: motivation, formulations and results. Pattern Recognit Lett 2001;22(12):1263–72. https://doi.org/10.1016/S0167-8655(01)00071-X.
[33] Simar L, Zelenyuk V. On testing equality of distributions of technical efficiency scores. Econom Rev 2006;25(4):497–522. https://doi.org/10.1080/07474930600972582.
[34] Tone K. A slacks-based measure of efficiency in data envelopment analysis. Eur J Oper Res 2001;130(3):498–509. https://doi.org/10.1016/S0377-2217(99)00407-5.
[35] Tsionas M. Efficiency estimation using probabilistic regression trees with an application to Chilean manufacturing industries. Int J Prod Econ 2022;249:108492. https://doi.org/10.1016/J.IJPE.2022.108492.
[36] Valero-Carreras D, Aparicio J, Guerrero NM. Support vector frontiers: a new approach for estimating production functions through support vector machines. Omega (Westport) 2021;104:102490. https://doi.org/10.1016/j.omega.2021.102490.
[37] Valero-Carreras D, Aparicio J, Guerrero NM. Multi-output support vector frontiers. Comput Oper Res 2022;143:105765. https://doi.org/10.1016/J.COR.2022.105765.
[38] Vapnik V. Principles of risk minimization for learning theory. Adv Neural Inf Process Syst 1991;4:831–8.
[39] Vapnik V. Statistical learning theory. Wiley; 1998.
[40] Vazquez E, Walter E. Multi-output support vector regression. IFAC Proc Volumes 2003;36(16):1783–8. https://doi.org/10.1016/S1474-6670(17)35018-8.
[41] Zhu J, Rosset S, Tibshirani R, Hastie T. 1-norm support vector machines. Adv Neural Inf Process Syst 2003;16:1–8.