



# Support vector frontiers: A new approach for estimating production functions through support vector machines<sup>☆</sup>

Daniel Valero-Carreras, Juan Aparicio\*, Nadia M. Guerrero

Center of Operations Research (CIO), Miguel Hernandez University of Elche (UMH), 03202 Elche (Alicante), Spain.

## ARTICLE INFO

### Article history:

Received 8 September 2020

Accepted 6 May 2021

Available online 19 May 2021

Edited by: Sungmook Lim

### Keywords:

Technical efficiency

Support Vector Regression

Free disposal hull

Data envelopment analysis

## ABSTRACT

In microeconomics, a topic of interest is the estimation of production functions. By definition, a production function is a non-decreasing function that envelops all the observations (firms) from above in the input-output space, capturing the extreme behavior of the data. These characteristics are far from the usual ones assumed by machine learning techniques like Support Vector Regression (SVR) in Support Vector Machines, where the function to be estimated relates the response variable to the covariables in terms of the mean instead of the extremes and, additionally, they try to fit the data as much as possible, determining a function that increases and decreases following a data-driven process. In this paper, we introduce an adaptation of SVR, denominated Support Vector Frontiers (SVF), with the objective of estimating production functions. To do so and seeking meeting points between SVR and the standard non-parametric techniques for estimating production functions, mainly Free Disposal Hull (FDH) and Data Envelopment Analysis (DEA), an estimator is defined in this paper through a specific input transformation function. However, and in contrast to FDH and DEA, SVF overcomes the overfitting problems from using these techniques. Additionally, we show in this paper that standard FDH and DEA could be reinterpreted, in some sense, as Support Vector Regression techniques. Moreover, a new robust notion of efficiency is introduced, called  $\varepsilon$ -insensitive technical efficiency, directly inherited from Support Vector Machines. Finally, the performance of SVF is measured through several experiments using synthetic data, showing that the new approach considerably reduces the bias and mean squared error associated with the estimation of the true production function in comparison with standard FDH and DEA, although at the expense of a more computational burden.

© 2021 The Author(s). Published by Elsevier Ltd.

This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

## 1. Introduction

Support Vector Machines (SVM) is a well-known machine learning technique based on advances in Statistical Learning. It is mainly rooted in the principle of structural risk minimization. In particular, the technique aims to minimize the bound on the generalization error (i.e., the error made by the machine learning on data outside the training set) rather than directly minimizing the empirical error, such as the traditional mean square error over the observed data set. The introduction of SVM in Vapnik [55,56] has led to a torrent of applications and theoretical analysis, which has now established SVM as one of the standard tools for machine learning.

In the last few years, there have been significant developments in the theoretical understanding of SVM as well as the application of the approach to many different empirical contexts. The technique has reached the point at which it is clearly viewed as one of the most fruitful research subareas within machine learning, a subarea that has reached a high degree of maturity in both theoretical insights and practical usefulness. However, there is currently no adaptation of SVM in the literature for approximating production functions in microeconomics, a field that is related to measuring the technical efficiency of firms. This is a special framework that requires specific needs regarding the surface to be estimated, as we will go on to explain.

Technical efficiency assessment is concerned with measuring the performance of firms, which convert inputs into outputs. Efficiency evaluation in production has been and is a relevant topic for managers and policy makers, as well as an area that merits attention from a practical and methodological point of view in

<sup>☆</sup> Area: Data-Driven Decision Making and Analytics; Sustainable Operations. This manuscript was processed by Associate Editor Sungmook Lim.

\* Corresponding author.

E-mail addresses: [dvalero@umh.es](mailto:dvalero@umh.es) (D. Valero-Carreras), [japaricio@umh.es](mailto:japaricio@umh.es) (J. Aparicio), [nguerrero@umh.es](mailto:nguerrero@umh.es) (N.M. Guerrero).

both engineering and economics (see, for example, [3,5,45]). From a methodological perspective, the main aim of such assessment is to analyze the technical efficiency of a set of entities, generally known as Decision Making Units (DMUs), by comparing their performance with respect to the so-called production possibility set or technology, which is unknown and must be estimated from a data sample (a learning sample in the terminology of machine learning). In this context, the cornerstone for the efficiency analysis of DMUs is the notion of production function. A production function represents the maximum product obtainable from the input combination at the existing state of technical knowledge. Its estimation allows calculating the corresponding technical inefficiency value as the deviation of each DMU to the boundary of the technology, characterized by the production function. In fact, given a level of input consumption, the most usual and natural measure of technical efficiency for firms is defined as the ratio of the actual produced output and the maximal producible output, determined by the production function.

As for the estimation of production functions in practice, before Farrell's [28] seminal contribution, economists used to specify the corresponding production functions parametrically, e.g., a Cobb-Douglas function, relying on Ordinary Least Squares (OLS) regression analysis to estimate an 'average' production function, and assuming that disturbance terms had zero mean. This was a clearly unsatisfactory estimation, as it did not follow the accepted definition of production functions in microeconomics as the 'maximal' feasible output for each resource combination considered. In this sense, Farrell [28] was the first in showing, for a single output and multiple inputs, how to estimate a surface enveloping all the observations. Farrell's contribution was based on the construction of a technology that satisfied two usual axioms in production theory: convexity and monotonicity (free disposability). Convexity establishes that if two input-output bundles are feasible (producible), then any convex combination of them is also feasible; whereas monotonicity states that if inputs increase, then the producible output must, at least, not decrease. However, many estimators meet both properties and, consequently, additional requirements are needed. In particular, the most conservative estimation of the production function would be that associated with a surface enveloping the data and, at the same time, as close as possible to them. This is the principle of conservation, known also as 'minimal extrapolation', which in the case of Farrell's estimator leads to a piece-wise linear surface.

Farrell's approach can be categorized in the current area of non-parametric techniques for estimating production functions since it is not necessary to identify, a priori, the specific mathematical formulation of the production function to be estimated. This line of research, initiated by Farrell, was later taken up by Charnes et al. [16] and Banker et al. [9], resulting in the development of the Data Envelopment Analysis (DEA) approach, in which the determination of the frontier is only restricted via its axiomatic foundation (mainly convexity, free disposability and minimal extrapolation). In this case, the axiom of convexity is translated into an additional requirement of the production function: concavity. Another paper working in this same line, is that by Afriat [1], showing how to determine a production function with the property P (e.g., non-decreasing concavity) that represents the set of observations to be as nearly efficient as possible. A more natural sequel than the DEA literature of the previous work done by econometricians, even before Farrell's contribution, would be Aigner and Chu [2], who showed how to estimate a 'parametric' Cobb-Douglas production function enveloping the data cloud from above. Furthermore, in a parallel way, Deprins et al. [22] introduced the alternative technique known as Free Disposal Hull (FDH), which relies exclusively on monotonicity and minimal extrapolation in contrast to DEA, which additionally assumes convexity.

Nowadays, two famous non-parametric approaches for estimating production functions are DEA and FDH [21,45]. In the case of DEA, the estimator is a piece-wise linear function, while in the case of FDH, the estimator is a step function. It is worth mentioning that FDH may be considered the 'skeleton' of DEA since the convex hull of the frontier estimated by FDH coincides with the DEA frontier [20]. As for when to use FDH or DEA, Simar and Wilson [52], p. 56] state that, if the technology is nonconvex, then there is no choice and one must use FDH, since DEA is inconsistent. If the technology is convex, then DEA is preferred over FDH since it offers a faster rate of convergence (but FDH still remains consistent). Moreover, the Data Generating Process (DGP) assumed in both techniques can be summarized as follows. If we assume that  $m$  inputs are involved in the production process, an unknown non-decreasing function  $f(\mathbf{x}) : \mathbb{R}_+^m \rightarrow \mathbb{R}_+$  indicates the maximal output that is producible from an input bundle  $\mathbf{x}$ . However, in practice, technical inefficiency can occur. Following Farrell [28], technical inefficiency reflects the inability of a DMU to obtain maximal output from a given set of inputs, due, for example, to a mismanagement of resources. Therefore, the actual observed output for a DMU coincides with  $y = f(\mathbf{x}) - u \geq 0$ , where  $u$  is conceptualized as a non-negative random variable linked to technical inefficiency. Note that then  $y \leq f(\mathbf{x})$ , which is associated with the enveloping nature of the production function described above in the text. In this way, in practice, for each DMU we observe  $y$  instead of  $f(\mathbf{x})$ , but we need to determine the ratio  $y/f(\mathbf{x})$ , which measures the level of technical inefficiency of the considered unit. This is the methodological problem that is faced by researchers in the field of efficiency measurement. Additionally, other recent alternative non-parametric techniques for estimating production frontiers are those that apply Kernel-based approaches and local regression techniques. See, for example, Du et al. [23], where the authors propose a kernel smoothing method that can handle multiple shape constraints (e.g., monotonicity) for multivariate functions, generalizing Hall and Huang [29]. Another interesting contribution is Parmeter et al. [47], who showed how constraint weighted bootstrapping may be applied to impose smoothness conditions on linear estimates. In particular, these authors estimated an input distance function both parametrically and non-parametrically, resorting in the latter to local linear generalized kernel regression. See also Henderson and Parmeter [31].

Analyzed as a machine learning technique, by construction, the FDH estimator  $\hat{f}_{FDH}$  is not endowed with generalization (out-of-sample) capability. In particular, the principle of minimal extrapolation focuses all its attention on minimizing the empirical error over the observed data set (i.e., minimizing the quantities  $\hat{f}_{FDH}(\mathbf{x}) - y$ ), provided that  $\hat{f}_{FDH}(\mathbf{x}) : \mathbb{R}_+^m \rightarrow \mathbb{R}_+$  is a non-decreasing function enveloping the data sample from above. In this sense, FDH suffers from the usual overfitting problem. FDH underestimates the real technical inefficiency of the observations, as this technique yields estimators always located below the (underlying) theoretical frontier  $f$ . Nevertheless, Free Disposal Hull is able to correctly 'describe' the situation of a particular set of observations from a 'relative' to the sample or sample-specific efficiency evaluation point of view, standing out as a descriptive statistic tool with little inferential capability for smaller samples (see its asymptotic properties and rates of convergence in [53]). It is worth mentioning that something similar happens with the DEA estimator  $\hat{f}_{DEA}$ . Consequently, despite the recognized data-driven nature of FDH and DEA, a conceptual gap remains in the literature between these common techniques in the field of efficiency evaluation and the world of machine learning. Anyway, there are certain key previous papers in the literature that have dealt with FDH and DEA techniques in order to transform them into inferential tools. Chronologically speaking, Banker and Maindiratta [10] and Banker [11] showed that DEA can be interpreted as a Maximum

Likelihood estimator. Later, Simar and Wilson [50] and Simar and Wilson [51,52] introduced how to determine confidence intervals for the efficiency score of each DMU, the ratio  $y/f(\mathbf{x})$  in our context, through adapting the bootstrapping methodology by Efron [24] to the context of Free Disposal Hull and Data Envelopment Analysis. More recently, Kuosmanen and Johnson [36] and Kuosmanen and Johnson [37] have shown that DEA may be interpreted as non-parametric least-squares regression subject to shape constraints on the production frontier and sign constraints on residuals. Additionally, these authors introduced the Corrected Concave Non-parametric Least Squares and showed that, if the data-generating process is deterministic and the inefficiency terms are identically and independently distributed, their estimator is consistent and asymptotically unbiased. However, none of them addresses the problem through machine learning techniques, despite the data-driven nature of DEA and FDH. One recent exception is Esteve et al. [25], where Classification and Regression Trees (CART) are adapted for estimating production functions through step functions, so competing against the standard Free Disposal Hull technique. See also Hong et al. [32], Pendharkar & Troutt [48], Misiunas et al. [41], Badiezadeh et al. [7], Zhu et al. [60], Zhu [59], Arreola et al. [6], Charles et al. [15], Lee and Cai [38], Mirmozaffari et al. [40] and Li et al. [39] to read about the recent interest of the DEA community in bridging the gap between Data Envelopment Analysis and data science, machine learning and big data.

In this paper, for the first time, the SVM technique is adapted for estimating production functions, satisfying the usual axioms established in microeconomics textbooks. The new approach will be named Support Vector Frontiers (SVF). SVF will allow estimating production functions by applying the structural risk minimization principle. In particular, the technique will also consider the generalization<sup>1</sup> and empirical errors, and not just the minimization of the empirical error as happens with FDH and DEA. In this way, this paper contributes to bridging the conceptual gap, pointed out above, by demonstrating that FDH and DEA can fit well within a more complex machine learning technique. More specifically, we prove that the FDH and DEA estimators are always feasible production functions generated by the optimization model linked to SVF, although they are not necessarily the optimal ones, except under restrictive conditions. To prove that, we introduce a specific input transformation function that transforms the original regressor hyperplane in order to provide a suitable step function as an estimation surface. In this way, the SVF method offers an FDH-type estimator. At the same time, the convex hull of our step function yields a convex estimation of the production possibility set, therefore, generating a DEA-type estimator. The notion of  $\varepsilon$ -insensitive technical efficiency will be also introduced in this paper. This concept is directly related to the estimated (SVM) margin through cross-validation and endows the traditional notion of efficiency with more robustness than usual. Additionally, the performance of SVF is checked through several experiments using synthetic data, showing that the new approach significantly reduces the mean squared error and bias associated with the estimation of

<sup>1</sup> Standard SVM tries to make inference but not in the traditional way, in the sense that SVM aims to identify the actual function that is behind the Data Generating Process. SVM is based on Statistical Learning (Vapnik, 2000). If the right balance is struck between the accuracy attained on some particular data (the observations), and the ability of the approach to learn any dataset without error, then a suitable estimation of the underlying function to be estimated will be achieved. This ability to learn any possible dataset is linked to the notion of the generalization error (also called prediction error, out-of-sample error or test error in the literature). The theoretical generalization error of a model cannot be calculated but it may be approximated by resorting to test samples or cross-validation and the traditional concept of prediction. Our approach inherits the feature of the standard SVM technique for trying to infer the underlying function behind the Data Generating Process that generated the observations.

the true frontier in comparison with standard FDH and DEA techniques. Nevertheless, several drawbacks in this new approach must be highlighted. For example, from a computational perspective, the new technique is more intensive than previous non-parametric approaches, such as FDH or DEA. Additionally, several user-defined parameters must be tuned through cross-validation, to name some of the limitations.

As far as we are aware, this paper represents the first contribution that adapts the SVM technique, from a methodological point of view, for estimating production functions following the usual axioms in production theory. The previous contributions that considered these two types of worlds, SVM and production frontiers, did not redefine SVM but combined it with the calculation of the standard DEA model at some stage. This means that the final efficiency estimations generated by these approaches inherit the same overfitting problems of traditional DEA. Examples of that are Song and Zhang [54], where, in a first stage, DEA is used to evaluate the efficiency scores of a set of oil refining enterprises (DMUs), and, in a second stage, traditional SVM for regression is applied on a database consisting of all inputs and outputs as predictors and the DEA efficiency score as response variable; Yeh et al. [58], where the objective is predicting business failure (the response variable) using the DEA efficiency score and other information as predictive variables; Poitier and Cho [49] use standard SVM for regression to predict the DEA score of best and worst DMUs in the data sample and aggregate all the information through an average function; Kao et al. [33] use a similar methodology to that introduced by Song and Zhang [54] in a multiclass classification problem, where each class is associated with a certain level of technical efficiency (estimated again by applying standard DEA); another similar paper is Farahmand et al. [26], which applies an analogous methodology to that of Song and Zhang [54]; or, finally, Chen et al. [17], where in a first stage, within the ambit of the satisficing Data Envelopment Analysis model, the probabilities of achieving a minimal performance threshold are computed, and in a second subsequent stage, SVM regression is applied to discriminate between high/low efficiency groups within each performance threshold.

The paper is organized as follows. Section 2 is devoted to briefly introducing the backgrounds. In Section 3, we extend Support Vector Machines to the context of estimating production functions, developing a new technique called Support Vector Frontiers (SVF). Performance of SVF is investigated through several experiments using synthetic data in Section 4. Finally, Section 5 concludes.

## 2. Background: Support Vector Regression (SVR)

The construction of machines capable of learning from experience has long attracted the attention of researchers. In this line, machine learning (ML) is the study of computer algorithms that improve automatically through experience. ML algorithms build a model based on sample data (training data) for making predictions or decisions without being explicitly programmed to do so. There are different types of learning within ML. On the one hand, under supervised learning, the training data are input/output pairs, with the objective of determining the functional relationship that links the output with the inputs. A learning problem with a binary output is referred to as a classification problem, while for a real-valued output, the problem becomes known as a regression problem. On the other hand, under unsupervised learning, there are no output values and the learning task is to gain some understanding of the process that generated the data (density estimation, clustering, and so on). In this paper, we focus on supervised learning and the regression problem due to the nature of our response variable (the output produced by a firm) (see [19,42]).

Support Vector Machines [55,56] is one of the recognized machine learning techniques. From a theoretical viewpoint, SVM is

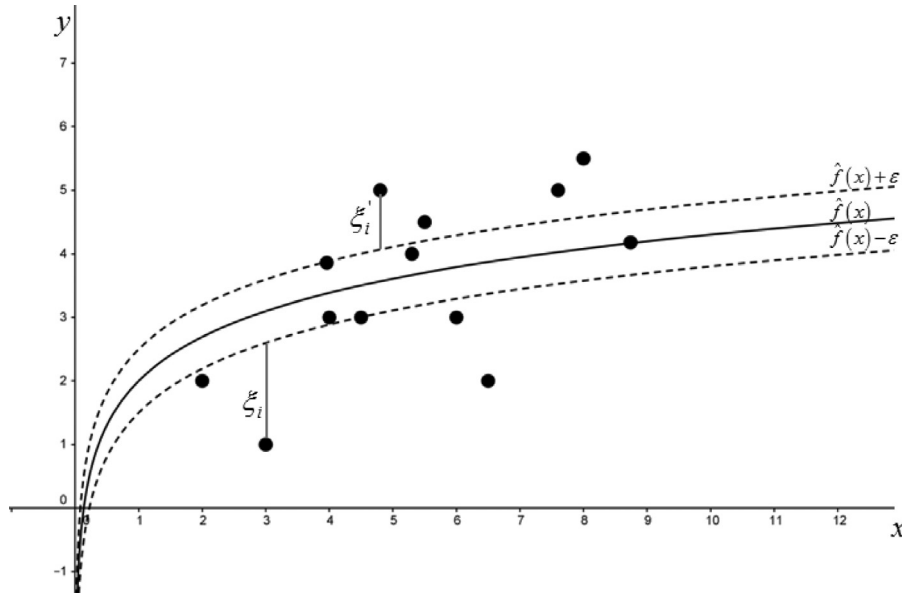


Fig. 1. Example of the standard SVR.

a constructive learning procedure grounded on statistical learning theory and the principle of structural risk minimization. It aims to minimize the bound on the generalization error instead of exclusively minimizing the empirical error such as the mean square error over the data set [55,56]. This results in good generalization capability: SVM tends to perform well when applied to data outside the training set. Because SVM is suitable for dealing with a limited number of samples, regardless of the dimension of the examples (the number of feature variables), it has been widely used in machine learning, data mining, pattern recognition, function approximation, regression, etc.

Support Vector Regression (SVR) is a particular model in the family of Support Vector Machines. As the rest of the regression procedures, SVR tries to construct a function that predicts the behavior of the response variable that is involved in the study, receiving, in this case, the main benefits of machine learning. Standard Support Vector Regression is constructed through a set of techniques whose aim is to predict the value of a response variable  $y \in \mathbb{R}$  given a vector of covariables  $\mathbf{x} \in \mathbb{R}_+^m$ . In this way, SVR sets a function  $\hat{f} : \mathbb{R}_+^m \rightarrow \mathbb{R}$  such that  $\hat{f}(\mathbf{x}) = \hat{y}$ , where  $\hat{y}$  constitutes the prediction of the response variable. Under SVR, the predictor  $\hat{f}$  is defined as  $\hat{f}(\mathbf{x}) = \mathbf{w}^* \phi(\mathbf{x}) + b^*$ , where  $\mathbf{w}^* \in \mathbb{R}^q$  and  $b^* \in \mathbb{R}$  are optimal solutions of model (1),  $\phi(\cdot)$  is a transformation function of the covariable space and the values of  $C \in \mathbb{R}_+$  and  $\varepsilon \in \mathbb{R}_+$  are obtained by a cross-validation process.

$$\begin{aligned} \text{Min}_{\mathbf{w}, b, \xi'_i, \xi_i} \quad & \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n (\xi'_i + \xi_i) \\ y_i - (\mathbf{w}\phi(\mathbf{x}_i) + b) & \leq \varepsilon + \xi'_i, \quad i = 1, \dots, n \quad (1.1) \quad (1) \\ (\mathbf{w}\phi(\mathbf{x}_i) + b) - y_i & \leq \varepsilon + \xi_i, \quad i = 1, \dots, n \quad (1.2) \\ \xi'_i, \xi_i & \geq 0, \quad i = 1, \dots, n \quad (1.3) \end{aligned}$$

Note that  $\hat{f}(\mathbf{x}) = \mathbf{w}^* \phi(\mathbf{x}) + b^*$  has the structure of a hyperplane in the transformed space  $(\phi(\mathbf{x}), y)$ . The SVR generates an estimator  $\hat{f}(\mathbf{x})$  of the response variable for vector  $\mathbf{x}$  as well as lower and upper 'correcting' surfaces, defined as  $\hat{f}(\mathbf{x}) - \varepsilon$  and  $\hat{f}(\mathbf{x}) + \varepsilon$ , where  $\varepsilon$  is a certain margin that endows the estimator linked to SVR with robustness (see Fig. 1). Additionally, observations below the surface  $\hat{f}(\mathbf{x}) - \varepsilon$  have an associated (empirical) error of  $\xi'_i > 0$  (with  $\xi'_i = 0$ ), while observations above the surface  $\hat{f}(\mathbf{x}) + \varepsilon$  present an (empirical) error of  $\xi_i > 0$  (with  $\xi_i = 0$ ). Observations between the surfaces  $\hat{f}(\mathbf{x}) - \varepsilon$  and  $\hat{f}(\mathbf{x}) + \varepsilon$  have an error of zero (with  $\xi_i = \xi'_i =$

0). Regarding the objective function in (1), it represents the combination of regression and regularization that SVR involves, mixing the empirical error term  $\sum_{i=1}^n (\xi'_i + \xi_i)$  and the regularization term

$\|\mathbf{w}\|$  through a weight  $C$ , which balances the two components [57]. Moreover, although hyperplanes have linear shapes, SVR is able to yield estimation functions that are not necessarily linear in the original  $(\mathbf{x}, y)$  space (see Fig. 1). It is due to the function  $\phi$ , which is a transformation of the covariable space,  $\phi : \mathbb{R}_+^m \rightarrow Z$ . It is important to remark that  $Z = \phi(\mathbb{R}_+^m)$ , known as feature space, can have different dimensions, including infinite. This transformation is the cause of the different nonlinear approaches in  $\mathbb{R}_+^m$ , while the estimation function generates a linear hyperplane in  $Z$ .

Sometimes, in the literature, it is easier or advisable to work with the dual model of program (1) and kernel functions, defined as  $K(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})\phi(\mathbf{x}')$ . However, in this paper, the primal model and the transformation function  $\phi(\cdot)$  will be enough for the development of the new approach.

Both the transformation function and the kernel function depend on several hyperparameters, say  $\tau$ , which must be determined together with  $C$  and  $\varepsilon$  by cross-validation. Cross-validation is a standard technique in machine learning and statistics for adjusting hyperparameters of models. In  $V$ -fold cross-validation, the learning sample  $\Omega$  is randomly divided into  $\Omega_1, \dots, \Omega_V$  disjoint subsamples with the same sample size or as close as possible. Typical values for  $V$  are 5 or 10. Let the  $\nu$ -th learning subsample be  $\Omega^{(\nu)} = \Omega - \Omega_\nu$  and let  $\Omega_i = \Omega - \Omega_{\nu(i)}$ , where  $\Omega_{\nu(i)}$  is the subsample such that  $i \in \Omega_{\nu(i)}$ . For each  $\nu = 1, \dots, V$ , an individual model is built by applying the algorithm to the training data  $\Omega^{(\nu)}$ . This model is then evaluated by means of a cost function using the test data in  $\Omega_\nu$ . In particular, given a set of hyperparameters  $(C, \varepsilon, \tau)$ , the generalization error is determined as  $\frac{1}{n} \sum_{i=1}^n (y_i - \hat{f}_{\Omega_i}(\mathbf{x}_i; C, \varepsilon, \tau))^2$ , where  $\hat{f}_{\Omega_i}(\mathbf{x}_i; C, \varepsilon, \tau)$  is the SVR estimator evaluated at  $\mathbf{x}_i$  when the learning data  $\Omega_i$  and the hyperparameters  $(C, \varepsilon, \tau)$  have been considered. To choose the best combination of hyperparameters using cross-validation, we compute the generalization error for different values of hyperparameters  $(C, \varepsilon, \tau)$ . Finally, one selects the combination  $(C^*, \varepsilon^*, \tau^*)$  with the lowest generalization error and uses it for training a SVR model on the complete data set  $\Omega$ .

### 3. Support Vector Regression for estimating production functions

Let us consider the observation of  $n$  Decision Making Units (DMUs).  $DMU_i$  consumes  $\mathbf{x}_i = (x_i^{(1)}, \dots, x_i^{(m)}) \in \mathbb{R}_+^m$  amounts of inputs for the production of  $\mathbf{y}_i = (y_i^{(1)}, \dots, y_i^{(s)}) \in \mathbb{R}_+^s$  amounts of outputs<sup>2</sup>. The relative efficiency of each DMU in the sample is assessed with reference to the so-called production possibility set or technology, which is the set of technically feasible combinations of  $(\mathbf{x}, \mathbf{y})$ . It is defined in general terms as:

$$T = \{(\mathbf{x}, \mathbf{y}) \in \mathbb{R}_+^{m+s} : \mathbf{x} \text{ can produce } \mathbf{y}\} \tag{2}$$

Certain assumptions are made or maintained on this set, such as free disposability of inputs and outputs, meaning that if  $(\mathbf{x}, \mathbf{y}) \in T$ , then  $(\mathbf{x}', \mathbf{y}') \in T$ , as long as  $\mathbf{x}' \geq \mathbf{x}$  and  $\mathbf{y}' \leq \mathbf{y}$ . Often convexity of  $T$  is also assumed (see, e.g., [27])<sup>3</sup>.

When  $s = 1$ , this context is restricted to the key notion of production function. Accordingly,  $m$  input variables are used to produce a univariate output. In this way, the technology is defined as

$$T = \{(\mathbf{x}, y) \in \mathbb{R}_+^{m+1} : y \leq f(\mathbf{x})\} \tag{3}$$

In this context, the property of monotonicity is translated so that the production function  $f$  is supposed to be monotone non-decreasing: if  $\mathbf{x} \leq \mathbf{x}'$ , then  $f(\mathbf{x}) \leq f(\mathbf{x}')$ . Hereinafter, we will focus our attention on the estimation of technical efficiency in the contexts of production functions.

As far as the measurement of technical efficiency is the concern, a certain part of the boundary of  $T$  is of interest. Specifically, we are referring to the efficient frontier of  $T$ , defined as  $\partial(T) := \{(\mathbf{x}, y) \in T : y = f(\mathbf{x})\}$ . In this way, technical inefficiency is defined as the distance from an interior point to this boundary. If  $(\mathbf{x}', y') \in T$ , then its technical efficiency score is determined by the ratio  $y'/f(\mathbf{x}')$ .

Nowadays, there are two main non-parametric methods based on envelopment techniques for estimating the efficient frontier of  $T$ : FDH and DEA. The FDH estimator was introduced by Deprins et al. [22] and relies only on the free disposability (monotonicity) assumption and the minimal extrapolation principle. In contrast, the DEA estimator requires stronger assumptions, such as convexity of the set  $T$ . The convexity assumption is widely used in economics, but it is not always valid [34]. The production possibility set might admit increasing returns to scale (i.e. output increases faster than the inputs, which graphically cannot be modelled by convexity), or there might be lumpy goods (i.e. fractional values of inputs or outputs do not exist). Hence, the FDH can yield a more general and flexible estimator than DEA (see [4]).

In this section, we introduce a new technique based on a methodological adaptation of the standard SVR for the estimation of production functions, which we call Support Vector Frontiers (SVF). The new approach will allow the determination of production functions, satisfying the usual axioms of microeconomics, through a data-driven approach that does not assume any particular random distribution on the data and generates, at a first stage, a step function as an estimator. At a second stage, the convexification of this step function will allow us to provide a piece-wise linear estimator of the production function. These shapes are shared with the estimators derived from FDH and DEA techniques. However, while the latter suffer the problem of overfitting, the new method will try to overcome this drawback through minimizing the structural risk as SVR does. To start with, we will show how the standard SVR model must be modified in order to estimate enveloping surfaces, satisfying certain classical axioms in production theory (see, for example, [27]):

- (A1) if  $\mathbf{x} = \mathbf{0}_m$ , then  $f(\mathbf{x}) = 0$ ;
- (A2) if  $\mathbf{x} \leq \mathbf{x}'$ , then  $f(\mathbf{x}) \leq f(\mathbf{x}')$ ;
- (A3)  $f$  is a concave function.

Axiom A1 means that if a firm does not consume any input, it cannot produce anything. Axiom A2 means that if a firm consumes more resources, then, it is always possible to produce more output; while A3 is associated with the convexity of the production possibility set  $T$ .

In order to be clear in our exposition, we will adapt the standard SVR model step by step in this section. It means that we will incorporate the satisfaction of each production axiom A1-A3 gradually throughout the text. Additionally, we will show the existing relationship between FDH and DEA and the new machine learning framework. In fact, it is one of our objectives to show under which hypothesis these two standard techniques in efficiency measurement may be reinterpreted as SVM models. Finally, we will introduce a new robust definition of technical efficiency, called  $\varepsilon$ -insensitive technical efficiency, as a natural application of the notion of margin in SVM to the efficiency measurement field.

#### 3.1. The estimation of enveloping surfaces

In this subsection, we will show how to adapt the standard SVR model to estimate enveloping structures of the data. Additionally, we will adjust trivially the model to meet A1.

Through the Data Generating Process mentioned in the Introduction, we know that our target function to be estimated,  $f$ , and the response variable (the output in our production context) have the following relationship with respect to each learning example  $i$ ,  $i = 1, \dots, n$ :  $y_i \leq f(\mathbf{x}_i)$ . Therefore, it seems natural to force the estimator to meet the same condition and add it to the constraints of the SVR model (1):  $y_i \leq \hat{f}(\mathbf{x}_i) = \mathbf{w}\phi(\mathbf{x}_i) + b$ ,  $i = 1, \dots, n$ . Nevertheless, this new type of restriction directly implies the satisfaction of the constraint (1.1),  $y_i - (\mathbf{w}\phi(\mathbf{x}_i) + b) \leq \varepsilon + \xi_i'$ , since  $y_i \leq \mathbf{w}\phi(\mathbf{x}_i) + b \Leftrightarrow y_i - (\mathbf{w}\phi(\mathbf{x}_i) + b) \leq 0$  and, additionally, we have that  $\varepsilon + \xi_i' \geq 0$ , for all  $i = 1, \dots, n$ . Consequently, constraint type (1.1) can be removed from the new optimization model if the conditions  $y_i \leq \mathbf{w}\phi(\mathbf{x}_i) + b$ ,  $i = 1, \dots, n$ , are incorporated. And, for the same reason, the decision variables  $\xi_i'$ ,  $i = 1, \dots, n$ , can also be deleted<sup>4</sup>. Moreover, a simple manner of satisfying

<sup>2</sup> We use bold for denoting vectors, and non-bold for scalars. The different components of a vector are denoted by a superscript and parenthesis.

<sup>3</sup> Let  $\mathbf{z} = (z^{(1)}, \dots, z^{(m)})$  and  $\mathbf{q} = (q^{(1)}, \dots, q^{(m)})$ . Hereinafter,  $\mathbf{z} \leq \mathbf{q}$  means  $z^{(j)} \leq q^{(j)}$  for all  $j = 1, \dots, m$ .

<sup>4</sup> In the new approach, as happens with the standard SVR, the upper and lower bounds work together to minimize errors related to the production function to be estimated, which lies somewhere between these upper and lower limits. In this framework, the region delimited by the upper and lower bounds could be related, in some way, to an order- $m$  or order- $\alpha$  frontier or to a quantile regression (see, for example, [4]). In the latter cases, because of the nature of the approaches, the frontier estimators do not envelop all the data points, and so they may be seen as more robust to extreme values than the standard FDH and DEA estimators.

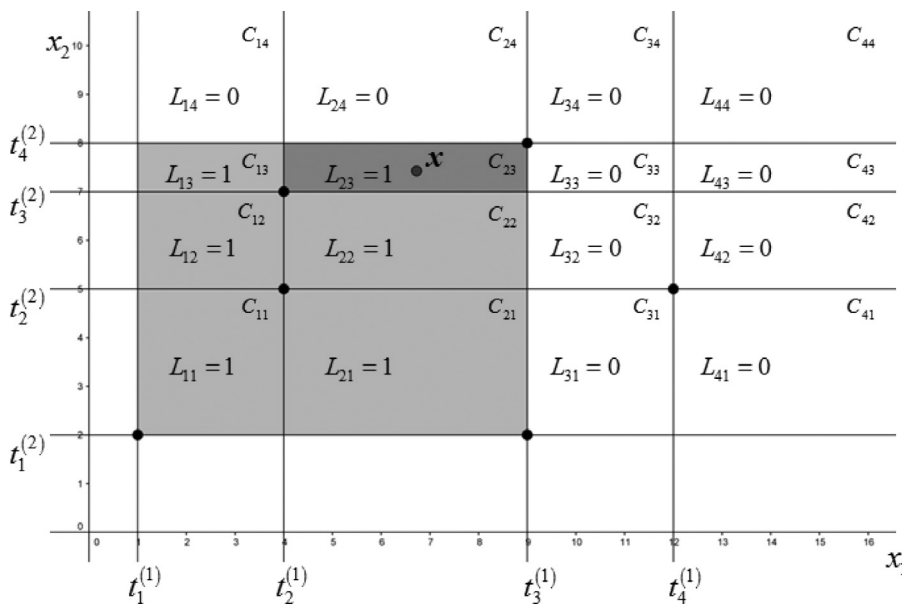


Fig. 2. Example of a grid.

axiom A1 consists of forcing the term  $b$  to be directly zero as long as  $\phi(\mathbf{0}_m) = \mathbf{0}_q$ . It guarantees that  $\hat{f}(\mathbf{0}_m) = \mathbf{w}\phi(\mathbf{0}_m) + b = 0$ . In this way, at this point, the adapted SVR model would be as follows:

$$\begin{aligned}
 \text{Min}_{\mathbf{w}, \xi_i} \quad & \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n \xi_i \\
 & y_i - \mathbf{w}\phi(\mathbf{x}_i) \leq 0, \quad i = 1, \dots, n \quad (4.1) \\
 & \mathbf{w}\phi(\mathbf{x}_i) - y_i \leq \varepsilon + \xi_i, \quad i = 1, \dots, n \quad (4.2) \\
 & \xi_i \geq 0, \quad i = 1, \dots, n \quad (4.3)
 \end{aligned}
 \tag{4}$$

Models (1) and (4) are not mathematically equivalent. Model (4) is the adaptation of the standard SVR model (1) for enveloping the observations from above, thanks to constraint (4.1).

Regarding the shape of the estimator, it is worth mentioning that standard efficiency measurement approaches like FDH and DEA resort to certain particular types of surfaces. FDH uses step functions and DEA resorts to the convexification of the FDH step function to get a concave estimator of the production function. Although, in this respect, standard SVM is flexible, allowing to use several complex transformation families of functions (polynomial, radial, Laplacian kernels, among others), we will focus on a step function as the transformation function  $\phi(\cdot)$ , seeking to establish a bridge between the standard approaches for measuring technical efficiency and the SVR method. In this way, and by analogy with FDH and DEA, we suggest in this paper to apply a transformation function in such a way that our estimator is a step function, which could be subsequently transformed into a concave function by convexification. To do that, we will base our approach on the spline approximation of the estimator constructed from a finite number of knots by Vapnik [56], p. 464].

Next, we introduce the necessary definitions and notation for establishing the transformation function  $\phi_{SVF}(\cdot)$  that will be used in this paper. For each input dimension  $j$ ,  $j = 1, \dots, m$ , the set of knots is defined as  $T_j = \{t_l^{(j)} : l_j = 1, \dots, k_j\}$  satisfying  $0 < t_1^{(j)} < t_2^{(j)} < \dots < t_{k_j}^{(j)}$  with  $t_l^{(j)} \in \mathbb{R}$ ,  $l_j = 1, \dots, k_j$ . The disjoint subsets  $C_{1\dots 1}, \dots, C_{k_1\dots k_m}$  such that  $C_{l_1\dots l_m} = \{\mathbf{x} \in \mathbb{R}_+^m : t_{l_j}^{(j)} \leq x^{(j)} < t_{l_{j+1}}^{(j)}, j = 1, \dots, m\}$ ,  $l_1 \in \{1, \dots, k_1\}, \dots, l_m \in \{1, \dots, k_m\}$ , with  $t_{k_j+1}^{(j)} := \infty, \forall j = 1, \dots, m$ , define a grid  $G$  on  $\mathbb{R}_+^m$ . Each subset  $C_{l_1\dots l_m}$  is called a cell of the grid  $G$ . See Fig. 2, where an example of grid and its cells are shown. For each cell  $C_{l_1\dots l_m}$  of a grid  $G$ ,  $\mathbf{a}_{l_1\dots l_m} = (t_{l_1}^{(1)}, \dots, t_{l_m}^{(m)})$  and  $\mathbf{b}_{l_1\dots l_m} = (t_{l_1+1}^{(1)}, \dots, t_{l_m+1}^{(m)})$  are the lower extreme knot-point and the upper extreme knot-point, respectively. Hence, each cell can be equivalently rewritten as  $C_{l_1\dots l_m} = \{\mathbf{x} \in \mathbb{R}_+^m : a_{l_1\dots l_m}^{(j)} \leq x^{(j)} < b_{l_1\dots l_m}^{(j)}, j = 1, \dots, m\}$ . Now, given an input vector  $\mathbf{x} \in \mathbb{R}_+^m$ , we need to identify the cell where it is located. The following binary ‘activation’ function defined for each cell  $C_{l_1\dots l_m}$  will indicate with a value of one if  $\mathbf{x} \in C_{l_1\dots l_m}$ :  $\mathbf{x} \rightarrow L_{l_1\dots l_m}(\mathbf{x}) = \prod_{j=1}^m B(\mathbf{x}^{(j)} - t_{l_j}^{(j)})$ , where

$$B(\mathbf{x}^{(j)} - t_{l_j}^{(j)}) = \begin{cases} 1, & \text{if } \mathbf{x}^{(j)} - t_{l_j}^{(j)} \geq 0 \\ 0, & \text{if } \mathbf{x}^{(j)} - t_{l_j}^{(j)} < 0 \end{cases}, \forall l_j = 1, \dots, k_j, \forall j = 1, \dots, m.
 \tag{5}$$

Note also that when a cell  $C_{l_1\dots l_m}$  is activated, then all the cells  $C_{s_1\dots s_m}$  with  $s_j \leq l_j$ ,  $j = 1, \dots, m$ , are also activated, i.e., if  $L_{l_1\dots l_m}(\mathbf{x}) = 1$ , then  $L_{s_1\dots s_m}(\mathbf{x}) = 1, \forall s_j \leq l_j, j = 1, \dots, m$ . We will say that all these cells are cells dominated by  $C_{l_1\dots l_m}$ . Fig. 2 shows how the activation of cells in a grid of  $\mathbb{R}_+^m$  for two input dimensions works.

In this context, given a grid  $G$ , we define the transformation function  $\phi_{SVF}^G$ , which will be used in our approach, as the mapping of the space  $\mathbb{R}_+^m$  given by

$$\mathbf{x} \rightarrow \phi_{SVF}^G(\mathbf{x}) = (L_{11\dots 11}(\mathbf{x}), L_{11\dots 12}(\mathbf{x}), \dots, L_{11\dots 1k_m}(\mathbf{x}), L_{11\dots 21}(\mathbf{x}), L_{11\dots 22}(\mathbf{x}), \dots, L_{11\dots 2k_m}(\mathbf{x}), \dots, L_{k_1 k_2 \dots k_m}(\mathbf{x}))
 \tag{6}$$

For each input vector  $\mathbf{x} \in \mathbb{R}_m^+$ ,  $\phi_{SVF}^G(\mathbf{x})$  corresponds to a vector of ones and zeros generated from the  $k_1 \cdot k_2 \cdot \dots \cdot k_m$  activation functions on grid G. Following our previous discussion, the components of the vector  $\phi_{SVF}^G(\mathbf{x})$  value one for the cell where  $\mathbf{x}$  is located and all its corresponding dominated cells. In this way, at this point, the adapted SVR model for estimating production functions would be as follows:

$$\text{Min}_{\mathbf{w}, \xi_i} \quad \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n \xi_i \quad (7.0)$$

$$y_i - \mathbf{w} \phi_{SVF}^G(\mathbf{x}_i) \leq 0, \quad i = 1, \dots, n \quad (7.1)$$

$$\mathbf{w} \phi_{SVF}^G(\mathbf{x}_i) - y_i \leq \varepsilon + \xi_i, \quad i = 1, \dots, n \quad (7.2)$$

$$\xi_i \geq 0, \quad i = 1, \dots, n \quad (7.3)$$

The estimator linked to (7) would be  $\hat{f}(\mathbf{x}) = \mathbf{w} \phi_{SVF}^G(\mathbf{x})$ . Note that  $\hat{f}(\mathbf{0}_m) = 0$  since  $\phi_{SVF}^G(\mathbf{0}_m) = \mathbf{0}_q$ , with  $q = k_1 \cdot k_2 \cdot \dots \cdot k_m$ . It guarantees the satisfaction of axiom A1. Moreover, by (6), if  $\mathbf{x} \in C_{l_1 \dots l_m}$ , then we have that  $\hat{f}(\mathbf{x}) = \mathbf{w} \phi_{SVF}^G(\mathbf{x}) = \sum_{s_1=1, \dots, l_1} w_{s_1 \dots s_m} \cdot$

Additionally, it is worth noting that the number of components of the vector of parameters  $\mathbf{w}$  coincides with the total number of cells of the corresponding grid G. Explicitly, the parametric vector  $\mathbf{w}$  can be written as follows.

$$\mathbf{w} = (w_{11 \dots 11}, w_{11 \dots 12}, \dots, w_{11 \dots 1k_m}, w_{11 \dots 21}, w_{11 \dots 22}, \dots, w_{11 \dots 2k_m}, \dots, w_{k_1 k_2 \dots k_m}) \quad (8)$$

Each component of  $\mathbf{w}$  is linked to a cell on the grid and the order of the components is the same as that followed by the vector  $\phi_{SVF}^G(\mathbf{x})$  in (6). This relationship allows introducing the notion of a parametrized grid: A grid G in which is assigned a parameter  $w_{l_1 \dots l_m} \in \mathbb{R}$  for each cell  $C_{l_1 \dots l_m}$ ,  $l_j = 1, \dots, k_j$ ,  $j = 1, \dots, m$ , is a parameterized grid with parameters  $\mathbf{w}$ .

Next, we are going to show that the estimator derived from model (7) is a step function and, additionally, any step function  $f(\mathbf{x})$  defined on a grid G may be equivalently rewritten in the way  $f(\mathbf{x}) = \mathbf{w} \phi_{SVF}^G(\mathbf{x})$  for a certain parametric vector  $\mathbf{w}$ . Nevertheless, we first need to introduce the definition of a step function on a grid and the notion of a recoverable function through a parametrized grid: A function  $f: \mathbb{R}_+^m \rightarrow \mathbb{R}$  is a step function on a grid G if  $f(\mathbf{x})$  is constant  $\forall \mathbf{x} \in C_{l_1 \dots l_m}$ ,  $\forall l_1 = 1, \dots, k_1, \dots, \forall l_m = 1, \dots, k_m$ . The value of the function  $f$  for any  $\mathbf{x} \in C_{l_1 \dots l_m}$  will be denoted as  $f(C_{l_1 \dots l_m})$ . Additionally, a function  $f: \mathbb{R}_+^m \rightarrow \mathbb{R}$  is a recoverable function through a parameterized grid G with parameters  $\mathbf{w}$  if and only if  $\forall \mathbf{x} \in \mathbb{R}_+^m$   $f(\mathbf{x}) = \mathbf{w} \phi_{SVF}^G(\mathbf{x}) = \sum_{s_1=1, \dots, l_1} w_{s_1 \dots s_m}$ , with  $\mathbf{x} \in C_{l_1 \dots l_m}$ .

Given a grid G, it is clear that  $\hat{f}(\mathbf{x}) = \mathbf{w} \phi_{SVF}^G(\mathbf{x}) = \sum_{s_1=1, \dots, l_1} w_{s_1 \dots s_m}$  is constant for each  $\mathbf{x} \in C_{l_1 \dots l_m}$ . Consequently, the estimator derived

from (7) is a step function defined on G. Additionally, it can be proved that, given a step function  $f(\mathbf{x})$  defined on a grid G, there will always be parameters  $\mathbf{w}$  such that  $f(\mathbf{x})$  may be equivalently expressed as  $f(\mathbf{x}) = \mathbf{w} \phi_{SVF}^G(\mathbf{x}) = \sum_{s_1=1, \dots, l_1} w_{s_1 \dots s_m}$ .

**Proposition 1.** Any step function on a grid G is a recoverable function through the parameterized grid G for a certain unique parametric vector  $\mathbf{w}$ .

**Proof.** See Appendix. ■

This result shows the flexibility of the transformation suggested in this paper, at least, with respect to any step function that can be defined on a grid in the input space. Later in the text, this proposition will be the key for showing that the step function associated with the FDH technique is always a feasible step function linked to the new approach, as long as we resort to a specific ‘empirical’ grid.

At this point, the estimator derived from model (7) is able to generate step functions enveloping the data cloud from above and satisfying axiom A1. However, the model can yield non-monotonic estimates, which are not valid when the objective is to determine a (production) function meeting axiom A2. Additionally, concavity of the estimator is not assured, which contradicts axiom A3. We will try to overcome these drawbacks of the technique in the next section.

### 3.2. The axioms of monotonicity and concavity of the production function

Monotonicity and concavity are two basic axioms of production functions in microeconomics. Consequently, in this section, we try to endow the estimator derived from the new approach based on SVR with the satisfaction of these two properties. Accordingly, the first part of this subsection is devoted to defining the necessary conditions for assuring a monotonic non-decreasing step function as an estimator from the SVF approach. In the final part of this subsection, we will show how to trivially determine concave functions as estimators of the production functions by directly applying convexification of the previously defined step function.

We start by establishing the conditions that assure that the final estimator derived from SVF meets monotonicity for any recoverable function defined on a parametric grid G. To do that, we first need a technical lemma.

**Lemma 1.** Let G a grid and let  $\mathbf{x} \in C_{h_1 \dots h_m}$  and  $\mathbf{z} \in C_{l_1 \dots l_m}$  with  $\mathbf{x} \leq \mathbf{z}$ . Then,  $h_j \leq l_j, \forall j = 1, \dots, m$ .

**Proof.** See Appendix. ■

Before showing the result on monotonicity, let us introduce the following notation:  $W_{l_1 \dots l_m} := \sum_{s_1=1, \dots, l_1} \dots \sum_{s_m=1, \dots, l_m} W_{s_1 \dots s_m}$ .

**Theorem 1.** Let  $f : \mathbb{R}_+^m \rightarrow \mathbb{R}_+$  a recoverable function through the parameterized grid  $G$ . Then,  $f$  is a monotonic non-decreasing function if and only if  $W_{l_1 \dots l_m} \geq W_{h_1 \dots h_m}, \forall h_j \leq l_j, \forall j = 1, \dots, m$ , with  $h_j, l_j = 1, \dots, k_j$ .

**Proof.** See Appendix. ■

It is possible to state a similar result but using a smaller number of constraints than in Theorem 1. This statement is formally established in Proposition 2.

**Proposition 2.** The system of inequalities  $W_{h_1 \dots h_m} \leq W_{l_1 \dots l_m}, \forall h_j \leq l_j, \forall j = 1, \dots, m$ , with  $h_j, l_j = 1, \dots, k_j$  is equivalent to the system of inequalities  $W_{l_1 l_2 \dots s_j \dots l_m} \leq W_{l_1 l_2 \dots l_m}, \forall s_j = l_j - 1, \forall l_j = 1, \dots, k_j, \forall j = 1, \dots, m$ .

**Proof.** See Appendix. ■

Once we have identified the conditions that characterize monotonicity for step functions defined on a parametrized grid, we are able to introduce the final optimization model corresponding to the new technique for estimating production functions, the so-called Support Vector Frontiers (SVF):

$$\text{Min}_{\mathbf{w}, \xi_i} \quad \|\mathbf{w}\|_1 + C \sum_{i=1}^n \xi_i \tag{9.0}$$

$$\text{s.t.} \quad y_i - \mathbf{w} \phi_{SVF}^G(\mathbf{x}_i) \leq 0, \quad i = 1, \dots, n \tag{9.1}$$

$$\mathbf{w} \phi_{SVF}^G(\mathbf{x}_i) - y_i \leq \varepsilon + \xi_i, \quad i = 1, \dots, n \tag{9.2}$$

$$W_{l_1 l_2 \dots s_j \dots l_m} \leq W_{l_1 l_2 \dots l_m}, \quad \forall l_1, \dots, l_m, \forall s_j = l_j - 1, \forall j = 1, \dots, m \tag{9.3}$$

$$\xi_i \geq 0, \quad i = 1, \dots, n \tag{9.4}$$

The constraints (9.1)-(9.4) correspond to the restrictions of model (7) with the new set of constraints (9.3) for ensuring monotonicity. Regarding the objective function, (9.0) coincides with (7.0) but, in the final model (9), the L1 norm is chosen. Although the Euclidean norm is the most usual metric when standard SVM is utilized, other possibilities exist in the literature (see, for example, [12]). In particular, in this paper, we consider the norm L1 because one of our objectives is comparing the new approach with standard FDH and DEA, which are based on linear programming. Note that, by assuming the norm L1, and given that  $\mathbf{w} \phi_{SVF}^G(\mathbf{x}_i) = \sum_{s_1=1, \dots, l_1} \dots \sum_{s_m=1, \dots, l_m} w_{s_1 \dots s_m}$  with  $\mathbf{x}_i \in C_{l_1 \dots l_m}$ ,

model (9) is a linear program. Given an optimal solution of (9),  $(\mathbf{w}^*, \xi_1^*, \dots, \xi_n^*)$ , the estimator of the production frontier linked to SVF is defined as  $f_{SVF}(\mathbf{x}) = \mathbf{w}^* \phi_{SVF}^G(\mathbf{x})$ , which is equal to  $\sum_{s_1=1, \dots, l_1} \dots \sum_{s_m=1, \dots, l_m} w_{s_1 \dots s_m}^*$  if  $\mathbf{x} \in C_{l_1 \dots l_m}$ . By the previous discussion,  $f_{SVF}(\mathbf{x})$  is a step monotonic

non-decreasing function that envelops the learning data. In order to additionally achieve a concave function, which in production theory is translated to convex technologies, it is enough to convexificate the estimator  $f_{SVF}(\mathbf{x})$ . A similar strategy is followed to get a convex production possibility set in DEA from a step production frontier estimated by FDH. To do that, in the case of SVF, we must identify a certain set of extreme points. In particular, each cell  $C_{l_1 \dots l_m}$  in the grid generates one of these points: one with input vector  $\mathbf{a}_{l_1 \dots l_m}$ , the lower extreme knot-point of cell  $C_{l_1 \dots l_m}$ , and output equals the estimation of the response variable for  $\mathbf{a}_{l_1 \dots l_m}$ , i.e.,  $f_{SVF}(\mathbf{a}_{l_1 \dots l_m})$ . Given the set of 'virtual' input-output points  $\{(\mathbf{a}_{l_1 \dots l_m}, f_{SVF}(\mathbf{a}_{l_1 \dots l_m}))\}_{l_1=1, \dots, k_1, \dots, l_m=1, \dots, k_m}$ , a convex production possibility set can be determined by applying a

DEA-type estimation:

$$\hat{T}_{CSVF} = \left\{ (\mathbf{x}, y) \in \mathbb{R}_+^{m+1} : y \leq \sum_{l_1=1, \dots, k_1} \lambda_{l_1 \dots l_m} f_{SVF}(\mathbf{a}_{l_1 \dots l_m}), \mathbf{x}^{(j)} \geq \sum_{l_1=1, \dots, k_1} \lambda_{l_1 \dots l_m} a_{l_1 \dots l_m}^{(j)}, \forall j, \sum_{l_1=1, \dots, k_1} \lambda_{l_1 \dots l_m} = 1, \lambda_{l_1 \dots l_m} \geq 0, \forall l_1, \dots, \forall l_m \right\}, \tag{10}$$

where the acronym CSVF denotes 'Convexificated Support Vector Frontiers'.

Fig. 3a shows a graphical example of an estimation of a production function from a learning sample with an input and an output. The solid line shows the step function linked to the estimator directly derived from the solution of model (9). It is a monotonic non-decreasing function that envelops the data. In contrast to FDH, SVF does not consider the principle of minimal extrapolation and, consequently, the SVF estimator can be located strictly above the FDH estimator with certain contact points. On the one hand, the hyperparameter C in the objective function of model (9) - which will be fitted by a cross-validation process as in the standard SVM - is the value that helps to locate the frontier more or less above the data. Greater values for C give more importance to the observations and are associated with an estimated frontier located closer to the data. On the other hand, the hyperparameter  $\varepsilon$ , which will be also fitted by cross-validation,

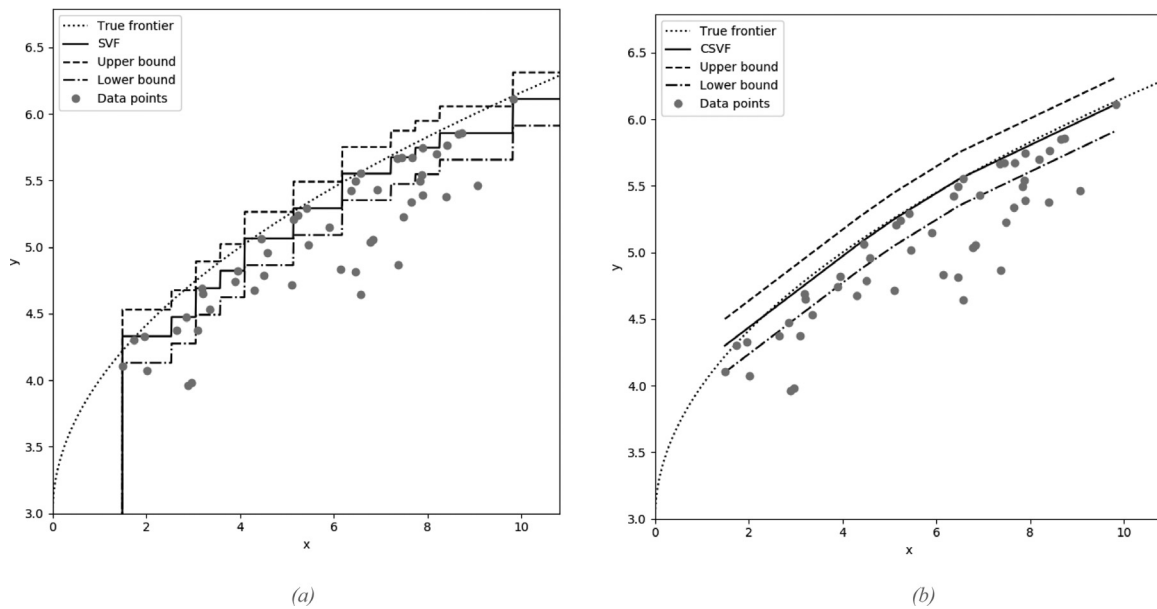


Fig. 3. Example of an estimation of SVF (a) and CSVF (b).

allows a certain degree of robustness to be considered in the analysis. The upper and lower ‘correcting’ surfaces, defined as  $f_{SVF}(\mathbf{x}) + \varepsilon$  and  $f_{SVF}(\mathbf{x}) - \varepsilon$ , respectively, are also illustrated in the figure by dashed lines. Observations between the upper and lower correcting surfaces have an empirical error of zero. Additionally, in Fig. 3b, we show the convexification of the SVF estimator. Fig. 3b shows a concave estimated production function, which implies that the area below this frontier defines a convex set. Furthermore, Fig. 3b also shows the upper and lower correcting surfaces for the convex setting, which is determined in a natural way by convexification from  $f_{SVF}(\mathbf{x}) + \varepsilon$  and  $f_{SVF}(\mathbf{x}) - \varepsilon$  in Fig. 3a, respectively.

Fig. 3 and the notion of margin  $\varepsilon$  in standard SVM inspire a new and more robust notion of technical efficiency in production theory. We are referring to Definition 1, where, in particular, the concept of  $\varepsilon$ -insensitive technical efficiency is introduced.

**Definition 1.** The DMU  $(\mathbf{x}_i, y_i)$  is  $\varepsilon$ -insensitive technically efficient if and only if  $f_{SVF}(\mathbf{x}_i) - \varepsilon \leq y_i \leq f_{SVF}(\mathbf{x}_i) + \varepsilon$ .

Note that, of course, the set of  $\varepsilon$ -insensitive technically efficient units include the traditional technically efficient DMUs, i.e., those located onto the (estimated) production frontier  $f_{SVF}$ . In rough terms, the model does not discriminate, regarding the degree of technical efficiency, among the units located between the estimator and the upper and lower correcting surfaces. For the SVF approach, all are equally technically efficient.<sup>5</sup> Additionally, for any DMU  $(\mathbf{x}_i, y_i)$  that belongs to the learning sample, we have that it is  $\varepsilon$ -insensitive technically efficient if and only if  $f_{SVF}(\mathbf{x}_i) - \varepsilon \leq y_i \leq f_{SVF}(\mathbf{x}_i)$ , since  $y_i \leq f_{SVF}(\mathbf{x}_i)$ . This is not the case for units that do not belong to the learning sample. For instance, in the case of intertemporal analysis, technological changes over time can lead to units belonging to a certain period that may be located outside the technology corresponding to a previous or subsequent period of time.

Finally, to finish this subsection, it is worth mentioning that model (9) needs the prior determination of certain parameters. Both  $C$  and  $\varepsilon$  were previously mentioned in the text. However, these are not the only parameters to be determined. As in the standard SVM, the transformation function (or, instead, the associated kernel) depends on another set of parameters. Within our context, the SVF transformation function  $\phi_{SVF}^C$  is linked to the determination of a set of knots for each input dimension considered in the problem. In this paper, we apply a simplified search method to determine these points since their identification could turn out to be a computationally hard task if their number and position are left free. Instead, the key parameter to be determined by cross-validation will be  $d$ , the number of cells to be defined for each input dimension. In an empirical context, we observe a maximum and minimum value in each input variable, making it possible to calculate the range and split it into  $d$  cells with the same width. This process will be illustrated in Section 4.

### 3.3. Free disposal hull and data envelopment analysis as support vector frontiers

Free Disposal Hull (FDH) and Data Envelopment Analysis (DEA) are well-known non-parametric approaches to technical efficiency analysis. In this subsection, we show that FDH and DEA may be alternatively interpreted as Support Vector Regression subject to shape constraints (monotonicity and concavity) that generates enveloping surfaces of the learning sample. This reinterpretation reveals the nature of FDH and DEA as part of machine learning techniques. In particular, we prove that these two standard techniques are always feasible solutions of the Support Vector Frontiers optimization model, being optimal in the simplest case of working with only one input and one output.

The non-parametric model as FDH is particularly appealing since it does not rely on restrictive hypothesis on the Data Generating Process, a feature shared with usual machine learning techniques, which are clearly data-driven approaches. In particular, Deprins et al. [22] proposed the Free Disposal Hull of the set of observations (DMUs) to estimate  $T$ , defined as follows:  $\hat{T}_{FDH} =$

<sup>5</sup> The  $\varepsilon$ -insensitive technical efficiency is somehow related to the notion of efficiency of some DEA models that deal with the concepts of imprecision and uncertainty, such as fuzzy DEA [30] or Chance Constrained DEA [46], to name just a few.

$\{(\mathbf{x}, \mathbf{y}) \in \mathbb{R}_+^{m+s} : \mathbf{y} \leq \mathbf{y}_i, \mathbf{x} \geq \mathbf{x}_i, i = 1, \dots, n\}$ . In the univariate output case, the production function would be estimated by  $f_{FDH}(\mathbf{x}) = \max_{i: \mathbf{x} \geq \mathbf{x}_i} \{y_i\}$ , which is a step function. The FDH technique is engaging because it relies on few assumptions, but, by construction, it suffers overfitting due to the minimal extrapolation principle. This principle forces the non-decreasing step function to be as close as possible to the data cloud, whereas it envelops all the observations from above.

In contrast to Free Disposal Hull, Data Envelopment Analysis additionally assumes convexity. This means that if  $(\mathbf{x}, y)$  and  $(\mathbf{x}', y')$  belong to  $T$ , then  $\lambda(\mathbf{x}, y) + (1 - \lambda)(\mathbf{x}', y') \in T$ , for all  $\lambda \in [0, 1]$ . Banker et al. [9] proposed the DEA estimator of the production possibility set  $T$  as follows:  $\hat{T}_{DEA} = \{(\mathbf{x}, \mathbf{y}) \in \mathbb{R}_+^{m+s} : \mathbf{y}^{(r)} \leq \sum_{i=1}^n \lambda_i \mathbf{y}_i^{(r)}, \forall r, \mathbf{x}^{(j)} \geq \sum_{i=1}^n \lambda_i \mathbf{x}_i^{(j)}, \forall j, \sum_{i=1}^n \lambda_i = 1, \lambda_i \geq 0, \forall i\}$ . In a single output context, the DEA estimator of a production function is a piece-wise linear function enveloping the data from above. Convexity of the production possibility set implies concavity of the production function. Also, the convexification of the FDH estimator yields the DEA estimator [20]. DEA suffers the same problem of overfitting as FDH. Again, the minimal extrapolation principle compels the determination of a ‘perfect’ description of the observed sample at a frontier level. It is done by calculating the smallest convex set that contains the data cloud and satisfies free disposability (monotonicity).

Throughout this section, we will focus our attention on FDH, achieving similar results with respect to DEA by direct convexification. We start by showing that the step function yielded by applying the FDH technique is always feasible in the SVF model (9), when a specific grid based on the observations is considered. We are referring to the ‘empirical’ grid, which uses the observations as knots for each input dimension and will be formally introduced in this subsection. Before, let us introduce new notation. Let  $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$  be a learning sample with  $\mathbf{x}_i \in \mathbb{R}_+^m$  and  $y_i \in \mathbb{R}_+$ ,  $\forall i = 1, \dots, n$ . Then, the set of elements  $\{\tilde{x}_{l_j}^{(j)}\}_{l_j=1}^{n^{(j)}}$ , for each  $j = 1, \dots, m$ , is the set of ordered values of the observations once repetitions have been removed in each dimension. Consequently,  $n^{(j)} \leq n$  and  $\tilde{x}_1^{(j)} < \dots < \tilde{x}_{n^{(j)}}^{(j)}$ . In this way, the empirical grid  $G^E$  is the grid defined from the following sets of knots  $T_j^E = \{\tilde{x}_{l_j}^{(j)}, l_j = 1, \dots, n^{(j)} + 1\}$ ,  $j = 1, \dots, m$ , with  $\tilde{x}_{n^{(j)+1}}^{(j)} := \infty$ ,  $\forall j = 1, \dots, m$ . The cells in the empirical grid will be denoted by  $C_{l_1 \dots l_m}^E$  and the extreme knot-points for each one as  $a_{l_1 \dots l_m}^E$  and  $b_{l_1 \dots l_m}^E$ . Another necessary definition is one that introduces the notion of set of dominated observations. Additionally, given an input vector  $\mathbf{x} \in \mathbb{R}_+^m$ , the set of dominated observations is defined as  $X = \{i \in \{1, \dots, n\} : \mathbf{x}_i \leq \mathbf{x}\}$ .

Next, we are going to prove that the estimator generated by FDH is a step function defined on the empirical grid  $G^E$ . This result is key for proving that the FDH estimator is always feasible of the SVF model. In order to prove that, we first need to state some previous technical results.

**Lemma 2.** Let  $i \in \{1, \dots, n\}$  and  $\mathbf{x}_i \leq a_{l_1 \dots l_m}^E$ . Then,  $\mathbf{x}_i \leq \mathbf{x}, \forall \mathbf{x} \in C_{l_1 \dots l_m}^E$ .

**Proof.** See Appendix. ■

**Corollary 1.** Let  $\mathbf{x} \in C_{l_1 \dots l_m}^E$ . Then  $A_{l_1 \dots l_m}^E := \{i \in \{1, \dots, n\} : \mathbf{x}_i \leq a_{l_1 \dots l_m}^E\} \subset X$ .

**Proof.** The result is a direct consequence of Lemma 2. ■

**Lemma 3.** Let  $i \in \{1, \dots, n\}$  and  $\mathbf{x} \in C_{l_1 \dots l_m}^E$ . If  $\mathbf{x}_i \leq \mathbf{x}$ , then  $\mathbf{x}_i \leq a_{l_1 \dots l_m}^E$ .

**Proof.** See Appendix. ■

**Corollary 2.** Let  $\mathbf{x} \in C_{l_1 \dots l_m}^E$ . Then  $X \subset A_{l_1 \dots l_m}^E$ .

**Proof.** The result is a direct consequence of Lemma 3. ■

**Proposition 3.** Let  $\mathbf{x} \in C_{l_1 \dots l_m}^E$ . Then  $X = A_{l_1 \dots l_m}^E$ .

**Proof.** This result is a consequence of Corollaries 1 and 2. ■

The next theorem proves that the estimator associated with the FDH technique is a step function defined on the empirical grid. Note that it is well-known in the literature that the FDH methodology always generates step functions. However, we also need to prove that it is a step function defined on a grid. In this case, we identify such grid with the empirical one.

**Theorem 2.** The estimator yielded from the FDH technique is a step function defined on the grid  $G^E$ . In particular,  $f_{FDH}(\mathbf{x}) = f_{FDH}(a_{l_1 \dots l_m}^E)$ ,  $\forall \mathbf{x} \in C_{l_1 \dots l_m}^E$ .

**Proof.** See Appendix. ■

By Theorem 2 and Proposition 1, we know that there is a vector  $\mathbf{w}^{FDH}$  such that the estimator associated with the FDH technique can be equivalently expressed as  $f_{FDH}(\mathbf{x}) = \mathbf{w}^{FDH} \phi_{SVF}^E(\mathbf{x})$ . Moreover,  $f_{FDH}$  is a recoverable function through the parameterized grid  $G^E$  with parameters  $\mathbf{w}^{FDH}$ . Additionally, if we define the empirical errors as

$$\xi_i^{FDH} := \begin{cases} (f_{FDH}(\mathbf{x}) - \varepsilon) - y_i, & \text{if } (f_{FDH}(\mathbf{x}) - \varepsilon) - y_i \geq 0 \\ 0, & \text{if } (f_{FDH}(\mathbf{x}) - \varepsilon) - y_i < 0 \end{cases}, \forall i = 1, \dots, n, \tag{11}$$

then we can state that FDH always generates a feasible solution of model (9). Expression (11) may be equivalently written in a compact way as  $\xi_i^{FDH} = \max\{(f_{FDH}(\mathbf{x}) - \varepsilon) - y_i, 0\}$ .

**Theorem 3.**  $(\mathbf{w}^{FDH}, \xi_1^{FDH}, \dots, \xi_n^{FDH})$  is a feasible solution of model (9).

**Proof.** See Appendix. ■

**Table 1**  
Description of the three scenarios.

Scenario	Inputs	Functional form
(I)	$x_1$	$y = 3 + x_1^{0.5} - u$
(II)	$x_1, x_2$	$y = 3 + x_1^{0.2} + x_2^{0.3} - u$
(III)	$x_1, x_2, x_3$	$y = 3 + x_1^{0.05} + x_2^{0.15} + x_3^{0.3} - u$

Accordingly, the step function generated by the standard FDH technique can be always recovered from the new approach, the so-called Support Vector Frontiers technique. The classical FDH estimated production function is always considered by model (9), given any parameters  $C$  and  $\varepsilon$ , as a possible SVF estimator. Another thing is that the FDH step function is optimal. This is not always true. Nevertheless, we are able to prove that, for any parameters  $C$  and  $\varepsilon$ , in the simplest framework, i.e. when the number of inputs equals one, the FDH estimator is always optimal of model (9) when the empirical grid is considered. It will be [Theorem 4](#). Before proving it, we will establish a useful lemma.

**Lemma 4.** Let  $m = 1$  and let  $(w', \xi'_1, \dots, \xi'_n)$  be a feasible solution of model (9) when the empirical grid is considered. Then,  $w' \phi_{SVF}^C(\max_{1 \leq i \leq n} \{x_i\}) \geq \max\{y_1, \dots, y_n\}$ .

**Proof.** See Appendix. ■

**Theorem 4.** Let  $m = 1$ . Then,  $(w^{FDH}, \xi_1^{FDH}, \dots, \xi_n^{FDH}) \geq 0$  is an optimal solution of model (9).

**Proof.** See Appendix. ■

However, [Theorem 4](#) cannot be extended to the general framework, i.e., when the number of inputs is arbitrary. Considering the empirical grid, a possible counterexample would be related to the following learning sample:  $(x_1^A, x_2^A, y^A) = (1, 4, 2)$ ,  $(x_1^B, x_2^B, y^B) = (2, 2, 1)$  and  $(x_1^C, x_2^C, y^C) = (3, 1, 3)$ .

By convexification, and taking into account that the standard DEA can be derived from the FDH step function by applying this same technique, all the previous results may be trivially extended to the estimation of convex production technologies. In this way, the traditional DEA estimation of the production function is always one of the possible solutions associated with the Convexified Support Vector Frontiers technique. Additionally, in the context of one input and one output, the DEA estimation is obtained from the optimal solution of the Support Vector Frontiers after applying the convexification of the SVF step function, regardless of the values of the hyperparameters  $C$  and  $\varepsilon$ .

#### 4. Experiments using synthetic data

This section describes simulation results that serve for the comparison of methods: FDH vs SVF and DEA vs CSVF. To do that, we present a comparison of all these methods in three alternative simulated environments. Their descriptions appear in [Table 1](#).

These three scenarios exemplify several Cobb-Douglas functions, widely known in the economic literature. Scenario I represents a single-input case and scenarios II and III represent multi-input cases. For all of them, we tested with data set sizes of 20, 30, 40, 50, 60, 70, 80, 90 and 100. The input data were randomly sampled from  $Uni[1,10]$ , independently for each input and observation. Then, the efficient output level was calculated and a random inefficiency term  $u \sim |N(0, 0.4)|$  was subtracted to obtain the data used for the analysis. We ran 50 trials ( $t = 1, \dots, 50$ ) for each combination of scenario and data set size to investigate the relative performance of the methods. Additionally, and due to the nature of SVF, as happens with SVM and SVR, the best model must be selected through the determination of the best combination of hyperparameters. This was carried out through a cross-validation process based on five folds. In our framework, the hyperparameters are  $C$ ,  $\varepsilon$  and  $d$ , the number of cells with the same width to be defined for each input dimension. In practice, it is necessary to state a finite number of possible combinations of these hyperparameters, since the computational cost of testing the infinite combinations of hyperparameters would be unmanageable. Arbitrarily, we fixed the following values for each hyperparameter:  $C \in \{0.1, 0.5, 1, 2, 5\}$ ,  $\varepsilon \in \{0, 0.001, 0.01, 0.1, 0.2\}$  and  $d = 0.1 \cdot t \cdot n$  (rounded to the nearest integer), with  $t = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10$ . These values generate a total quantity of 250 different combinations of hyperparameters for the SVF problem. Additionally, performance of each method is evaluated by two standard criteria: the mean squared error (MSE) and the bias. The MSE statistic is defined as  $\sum_{t=1}^{50} \sum_{i=1}^n (f(x_i^t) - \hat{f}(x_i^t))^2 / 50n$ , whereas

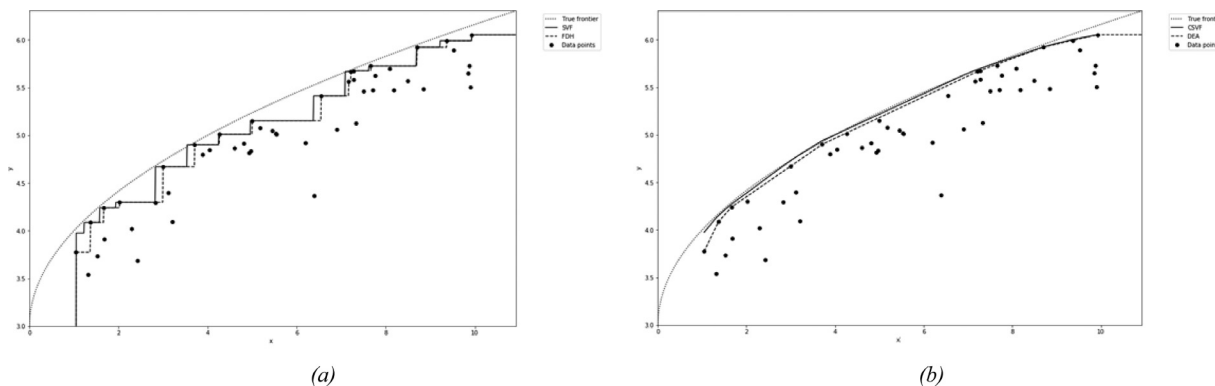
the bias is calculated as  $\sum_{t=1}^{50} \sum_{i=1}^n |f(x_i^t) - \hat{f}(x_i^t)| / 50n$ .

[Table 2](#) reports the mean and standard deviation (in brackets) of the best hyperparameters in our simulations. All of them were estimated by cross-validation except for scenario III, where, due to computational issues, we fixed the hyperparameter in  $d = 20$ . Regarding the results, the hyperparameter  $C$  was indifferent for the scenario with an input (scenario I), because the cross-validated generalization error was always the same. However, the value of the hyperparameter  $C$  was relevant for scenarios with more inputs, as II and III. Something similar happened with the value of the hyperparameter  $\varepsilon$  in scenario I with small sample sizes (20 and 30 observations). Regarding the hyperparameter  $d$ , it determines the size of the grid  $G$  (the number of cells), and, consequently, the number of decision variables and constraints in model (9). The observed trend shows that the greater the number of observations, the larger the value of the hyperparameter  $d$ .

[Table 3](#) describes the performance of the different methods studied: FDH, SVF, DEA and CSVF based on MSE. The first two columns indicate the scenario and the sample size. The next four columns show the mean and the standard deviation (in brackets) of the methods considered. The following two columns report the fraction of trials in which the SVF improves or equals the MSE of the FDH and the percentage of improvement of this method with respect to the other. The last two columns are like the previous ones but compare CSVF versus DEA. Regarding the results, all the methods were affected by the increase in dimensionality, from one to several inputs, since the

**Table 2**  
The best SVF hyperparameters.

Scenario	Number of obs.	Hyperparameter						
		C	$\varepsilon$			d		
		MEAN (STD)	MEAN (STD)	MIN	MAX	MEAN (STD)	MIN	MAX
I	20	indifferent	indifferent	0.001	0.001	14.2(4.005)	6	20
	30		0.001	0.001	23.16(5.751)	9	30	
	40		0.037(0.077)	0	0.2	31.68(6.944)	12	40
	50		0.079(0.095)	0	0.2	41.8(8.003)	25	50
	60		0.088(0.089)	0	0.2	49.08(9.207)	24	60
	70		0.094(0.121)	0	0.3	60.9(9.083)	35	70
	80		0.061(0.087)	0	0.2	72.32(8.698)	48	80
	90		0.033(0.068)	0	0.2	76.86(12.226)	45	90
	100		0.016(0.047)	0	0.2	89.8(12.204)	60	100
	II		20	1.804(1.924)	0.047(0.073)	0	0.2	11.52(5.339)
30		3.064(1.941)	0.025(0.045)	0	0.2	16.92(7.756)	6	30
40		3.332(1.811)	0.009(0.024)	0	0.1	25.28(9.23)	8	40
50		3.66(1.695)	0.022(0.049)	0	0.2	24.6(12.529)	5	50
60		3.15(1.785)	0.012(0.033)	0	0.2	38.04(16.603)	12	60
70		2.98(1.635)	0.006(0.02)	0	0.1	48.72(17.26)	14	70
80		3.96(1.551)	0.002(0.004)	0	0.01	49.76(23.214)	16	80
90		3.38(1.589)	0.011(0.027)	0	0.1	62.82(22.082)	18	90
100		3.46(1.717)	0.011(0.034)	0	0.2	63.2(25.67)	10	100
III		20	1.9(1.897)	0.054(0.08)	0	0.2		
	30	1.964(1.755)	0.019(0.041)	0	0.2			
	40	1.978(1.799)	0.023(0.044)	0	0.2			
	50	2.364(1.841)	0.028(0.058)	0	0.2			
	60	2.216(1.836)	0.011(0.027)	0	0.1	20(0)		
	70	1.996(1.697)	0.014(0.032)	0	0.1			
	80	2.104(1.664)	0.012(0.027)	0	0.1			
	90	2.572(1.773)	0.023(0.053)	0	0.2			
	100	3.25(1.882)	0.025(0.045)	0	0.2			



**Fig. 4.** Graphical illustrations of SVF vs FDH and CSVF vs DEA in a simulated dataset.

MSE increases as the number of inputs increases. Likewise, the improvement of the SVF in relation to the FDH was quite substantial, giving better results in all the simulations. The improvements ranged from 14.3% to 34.4% on average. As for the comparison between CSVF and DEA, the percentages of enhancement were higher than in the previous comparison between SVF and FDH, with improvements ranging from 38.7% to 78.4% on average. One significant result is that the superiority of SVF and CSVF compared to traditional approaches is high even under contexts with a small number of observations.

Table 4 shows the results based on the bias instead of the MSE. The structure of Table 4 is like that of Table 3. Regarding the bias, SVF outperforms FDH for all the computational experiences carried out, with a reduction ranging from 8% to 19.9%. As for the convex technologies, CSVF works better than the traditional DEA, with improvements from 27.3% to 56.1%. For CSVF, the percentage of improvement increases as the number of inputs augments. As in the case of the MSE, the dominance of the new approaches compared to traditional ones is clear, even under frameworks based on few observations.

Finally, Fig. 4a and 4b show a graphical example of the result of one of our simulations.

The new approach is based on Support Vector Regression (SVR), which combines the minimization of the empirical error with the minimization of the structural error. The minimization of the structural error comes from a complex theory, where the focus is the determination of a bound of the so-called PAC (probably approximately correct) learning (see, for example, [19]). The goal is that the estimated function will most likely have a low generalization error (i.e., a measure of how accurately an approach is able to estimate outcome values for previously unseen data). Consequently, Support Vector Frontiers (SVF), based upon SVR, learn what the hidden Data Generating Process is like by avoiding making a simple ‘description’ of the data sample at a frontier level, something that happens with FDH and DEA, which are exclusively based on the minimization of the empirical error due to the minimal extrapolation principle. Additionally, the experiments using synthetic data showed how the produc-

**Table 3**  
Relative performance of estimation methods based on the MSE performance criteria.

Scenario	Number of obs.	Mean squared error				Fraction of trials	Improvement (%)	Fraction of trials	Improvement (%)
		FDH	SVF	DEA	CSVF				
I	20	0.06(0.026)	0.046(0.024)	0.021(0.017)	0.012(0.011)	1	23.126(18.476)	0.96	39.28(29.892)
	30	0.048(0.016)	0.038(0.014)	0.017(0.011)	0.01(0.008)	1	20.47(13.607)	0.98	39.604(27.292)
	40	0.036(0.01)	0.029(0.009)	0.01(0.007)	0.006(0.006)	1	18(9.163)	0.98	41.54(22.823)
	50	0.033(0.011)	0.026(0.01)	0.008(0.006)	0.004(0.005)	1	19.686(13.358)	1	49.347(24.152)
	60	0.027(0.008)	0.022(0.008)	0.007(0.006)	0.004(0.004)	1	18.776(10.464)	1	47.294(20.497)
	70	0.026(0.008)	0.021(0.007)	0.007(0.005)	0.003(0.004)	1	20.539(10.891)	1	52.575(21.751)
	80	0.022(0.007)	0.018(0.006)	0.005(0.004)	0.003(0.002)	1	17.649(10.122)	1	48.986(23.684)
	90	0.02(0.005)	0.017(0.005)	0.004(0.003)	0.002(0.002)	1	14.712(8.33)	1	40.99(17.318)
	100	0.018(0.005)	0.015(0.004)	0.004(0.003)	0.002(0.002)	1	15.18(9.632)	1	38.709(22.785)
	II	20	0.093(0.031)	0.062(0.029)	0.048(0.023)	0.019(0.012)	1	32.906(21.515)	0.96
30		0.088(0.03)	0.065(0.027)	0.039(0.02)	0.014(0.01)	1	25.792(15.893)	0.98	62.702(21.254)
40		0.074(0.021)	0.058(0.021)	0.031(0.013)	0.011(0.008)	1	21.86(16.536)	1	64.682(19.001)
50		0.067(0.015)	0.052(0.016)	0.022(0.009)	0.009(0.009)	1	23.709(13.996)	0.96	55.665(45.693)
60		0.064(0.015)	0.05(0.015)	0.021(0.01)	0.008(0.007)	1	21.534(13.903)	0.98	61.858(24.149)
70		0.06(0.012)	0.05(0.013)	0.02(0.007)	0.008(0.007)	1	17.301(13.545)	0.98	58.134(21.477)
80		0.055(0.01)	0.046(0.01)	0.017(0.007)	0.006(0.003)	1	15.941(11.554)	0.98	63.6(16.983)
90		0.051(0.011)	0.044(0.011)	0.016(0.007)	0.007(0.005)	1	14.314(11.453)	0.98	60.297(19.467)
100		0.047(0.008)	0.04(0.01)	0.014(0.005)	0.005(0.003)	1	15.497(12.1)	0.98	62.138(18.285)
III		20	0.125(0.046)	0.085(0.034)	0.078(0.036)	0.029(0.019)	1	30.401(20.041)	1
	30	0.11(0.03)	0.075(0.028)	0.062(0.03)	0.019(0.013)	1	31.182(19.658)	0.98	66.747(18.605)
	40	0.106(0.032)	0.073(0.03)	0.054(0.02)	0.015(0.01)	1	31.681(18.429)	0.98	70.078(20.382)
	50	0.093(0.021)	0.063(0.018)	0.042(0.015)	0.011(0.006)	1	31.281(17.332)	1	73.877(15.085)
	60	0.094(0.018)	0.063(0.017)	0.042(0.014)	0.011(0.006)	1	32.396(14.453)	1	73.107(12.174)
	70	0.092(0.019)	0.061(0.018)	0.04(0.015)	0.01(0.007)	1	33.688(15.497)	1	74.851(14.008)
	80	0.082(0.017)	0.054(0.018)	0.034(0.011)	0.007(0.003)	1	34.423(15.374)	1	78.375(12.496)
	90	0.08(0.015)	0.053(0.015)	0.03(0.009)	0.009(0.013)	1	33.81(15.881)	0.96	66.782(53.812)
	100	0.079(0.013)	0.056(0.013)	0.03(0.009)	0.007(0.007)	1	29.867(11.365)	0.96	74.618(22.103)

**Table 4**  
Relative performance of estimation methods based on the bias performance criteria.

Scenario	Number of obs.	Bias							
		FDH_ABS	SVF_ABS	DEA_ABS	Fraction of trials CSVF_ABS	Improvement (%) SVF_ABS<=FDH_ABS	Fraction of trials SVF_ABS vs FDH_ABS	Improvement (%) CSVF_ABS<=DEA_ABS	CSVF_ABS vs DEA_ABS
I	20	0.198(0.05)	0.17(0.05)	0.104(0.042)	0.074(0.034)	1	14.211(10.371)	0.9	27.252(24.578)
	30	0.18(0.034)	0.156(0.034)	0.089(0.029)	0.06(0.023)	1	13.383(7.095)	0.96	30.993(18.712)
	40	0.155(0.024)	0.137(0.026)	0.066(0.022)	0.045(0.021)	1	12.288(6.549)	1	33.668(15.439)
	50	0.15(0.023)	0.132(0.024)	0.06(0.016)	0.04(0.016)	1	11.767(7.046)	0.98	35.347(16.288)
	60	0.134(0.018)	0.119(0.02)	0.053(0.016)	0.035(0.014)	1	11.152(5.373)	1	36.312(11.539)
	70	0.131(0.019)	0.116(0.02)	0.049(0.015)	0.032(0.015)	1	11.157(4.782)	1	37.062(13.023)
	80	0.123(0.02)	0.111(0.019)	0.046(0.015)	0.031(0.013)	1	9.347(3.65)	1	33.281(11.379)
	90	0.116(0.013)	0.106(0.014)	0.039(0.01)	0.027(0.009)	1	9.009(3.961)	1	31.853(10.586)
	100	0.11(0.015)	0.101(0.015)	0.038(0.011)	0.027(0.01)	1	8.722(3.565)	0.98	30.588(11.75)
	20	0.249(0.045)	0.2(0.05)	0.17(0.042)	0.101(0.032)	1	19.867(13.551)	0.96	39.658(18.271)
II	30	0.246(0.044)	0.212(0.046)	0.151(0.038)	0.082(0.031)	1	14.097(9.055)	0.98	44.811(18.545)
	40	0.223(0.035)	0.197(0.04)	0.132(0.029)	0.07(0.026)	1	12.283(8.81)	1	46.997(15.129)
	50	0.218(0.026)	0.189(0.032)	0.114(0.021)	0.063(0.027)	1	13.594(8.496)	0.96	43.884(22.736)
	60	0.207(0.025)	0.183(0.028)	0.104(0.02)	0.057(0.022)	1	11.829(7.956)	0.98	45.423(17.326)
	70	0.205(0.022)	0.187(0.023)	0.1(0.013)	0.061(0.021)	1	8.593(6.036)	0.98	39.713(18.1)
	80	0.193(0.017)	0.177(0.018)	0.091(0.013)	0.05(0.014)	1	8.19(5.649)	0.96	44.273(14.527)
	90	0.187(0.02)	0.173(0.022)	0.087(0.017)	0.05(0.017)	1	7.24(5.748)	0.98	42.471(15.682)
	100	0.182(0.017)	0.168(0.021)	0.081(0.013)	0.046(0.013)	1	8.035(6.362)	0.96	43.309(16.463)
	20	0.283(0.049)	0.234(0.048)	0.214(0.045)	0.117(0.036)	1	16.94(12.771)	0.98	44.533(15.679)
	30	0.271(0.038)	0.224(0.045)	0.19(0.042)	0.098(0.032)	1	17.024(12.169)	0.98	47.769(15.344)
III	40	0.264(0.039)	0.22(0.046)	0.173(0.034)	0.085(0.026)	1	17.113(10.631)	0.96	50.106(16.306)
	50	0.25(0.03)	0.206(0.03)	0.153(0.027)	0.072(0.023)	1	17.063(9.746)	1	52.36(15.557)
	60	0.25(0.027)	0.207(0.031)	0.15(0.024)	0.07(0.02)	1	17.391(8.809)	1	52.876(11.499)
	70	0.248(0.023)	0.204(0.031)	0.144(0.023)	0.069(0.022)	1	17.825(9.214)	0.96	52.017(16.303)
	80	0.235(0.026)	0.192(0.032)	0.134(0.021)	0.058(0.015)	1	18.561(8.696)	1	56.066(12.685)
	90	0.231(0.021)	0.189(0.027)	0.124(0.017)	0.064(0.035)	1	18.151(9.2)	0.94	47.215(31.566)
	100	0.232(0.019)	0.195(0.022)	0.124(0.016)	0.058(0.024)	1	15.712(6.472)	0.96	53.283(18.828)

tion frontier estimated by SVF is closer to the theoretical frontier than FDH and DEA, which somehow represent overfitted models. The estimated SVF frontier deviates from FDH and DEA production frontiers, thereby avoiding in some respects the overfitting problem.

Finally, we want to point out that our simulation study presents some limitations. We focused our attention exclusively on the quality of the frontier estimators with the objective of correctly identifying the production frontier, using the traditional bias and MSE measures. However, many interesting questions remain open. Most studies that apply FDH or DEA estimate pointwise efficiency through many different technical efficiency measures (radial, directional distance function, weighted additive measures, ...). This is clearly an extension of SVF to be carried out in the future. Additionally, studying robustness of the efficiency scores, preservation of the ranking of units, stability and other issues could also be of interest from theoretical and practical viewpoints.

## 5. Conclusions and future work

In this paper, we built a new bridge between non-parametric frontier analysis and Support Vector Machines (SVM). So far, these two fields have been growing in parallel with few significant contact points. However, they present certain points in common and the new tendency in economics, engineering and operations research on big data and machine learning strongly encourages efficiency analysis researchers to join the data analytics field (see, for example, the recent papers by [35] and [59]). In our case, this has meant introducing a new approach to estimate production frontiers by adapting Support Vector Regression, generating the so-called Support Vector Frontiers (SVF). Specifically, standard Support Vector Regression is subjected to shape constraints and adapted in such a way that the estimator surface envelops all the observations from above. Monotonicity is one of the key shape constraints. It is related to the classical axiom of free disposability, which any production function must meet in microeconomics. In a first stage and motivated by the Free Disposal Hull (FDH) technique, a specific transformation function of the input space was introduced that allowed determining monotonic non-decreasing step functions as estimator of the production functions. In a second stage, by convexification, we were able to yield concave predictors, which are directly linked to convex production possibility sets and Data Envelopment Analysis (DEA). This resulted in the introduction of the so-called Convexified Support Vector Frontiers (CSVF). In contrast to standard SVM, SVF and CSVF are able to estimate production functions since they have been defined to capture maximum trends in the data instead of mean trends and to guarantee the satisfaction of monotonicity and concavity.

Both FDH and DEA are well-known non-parametric approaches in the literature for measuring technical efficiency. In this paper, we also showed that these two standard methods could be interpreted as feasible estimators of the introduced adaptation of the Support Vector Regression technique. Under the simplest context of producing one output by consuming only one input, FDH and DEA are also optimal solutions of Support Vector Frontiers. This reinterpretation reveals the nature of FDH and DEA as part of existing machine learning techniques.

Despite the similarities between the traditional non-parametric approaches for determining technical efficiency and Support Vector Frontiers, FDH and DEA suffer from a problem of overfitting, a problem inherited from the axiom of minimal extrapolation assumed in the classical literature [1,9,28]. In contrast, the new technique does not assume this axiom, which endows SVF with more flexibility thus allowing to locate the estimated production frontier above the FDH and DEA estimators following a cross-validation process. Overall, the introduction of SVF in the literature opens up new ways for adapting and integrating machine learning techniques to the efficiency analysis world. Furthermore, performance of the new approach was investigated in this paper through several experiments using synthetic data. Our results indicated that SVF and CSVF outperform FDH and DEA, respectively, with respect to several traditional error measures like the mean squared error (MSE) and the bias. Regarding the MSE, we observed that the determined improvements ranged from 14.3% to 78.4% in our simulations. As for the bias, the enhancement ranged from 8% to 56.1%.

The new technique presents both extra advantages and drawbacks in comparison with FDH and DEA. As for the extra returns, the adaptation of the standard SVR allowed translating certain notions of this machine learning technique to the efficiency measurement world. In particular, we introduced the robust concept of  $\varepsilon$ -insensitive technical efficiency, which incorporates the notion of margin to the definition of performance efficiency. Regarding the drawbacks, the new technique takes up more computing time in comparison with FDH and DEA. Although the optimization program linked to SVF that must be determined is a Linear Program; the high number of constraints associated with the satisfaction of monotonicity and, additionally, the cross-validation process, need more computational time than the traditional FDH and DEA techniques. Other drawbacks include the fact that there are several user-defined parameters to be tuned; that there is a combinatorial nature to the size of the model; and that the determination of the "knots" and dimension of the cells may be problematic, to name just a few.

Regarding the implementation in practice of the methods mentioned in this paper, it is worth stating that the new approach is clearly more intensive with respect to the computational burden than traditional FDH and DEA, based on simple Linear Programming. Support Vector Frontiers depend on several parameters to be tuned through cross-validation and a lot of computing and storage capacity. Consequently, if the objective of the researcher is exclusively the description of the extreme behavior of a particular set of units, determining a 'relative' to the sample efficiency score, then both FDH and DEA are the most suitable techniques. However, sometimes the evaluation of the particular set of units in the sample is not the main objective for the researchers. For example, it is usual to resort to random samples of schools from international tests such as PISA (Programme for International Student Assessment) for evaluating the performance of educational systems. In this case, the interest is not the determination of the level of technical efficiency of each anonymous school, but estimating the underlying educational production function associated with each country in the OECD (The Organisation for Economic Co-operation and Development). This is when it makes sense to apply machine learning techniques, such as Support Vector Frontiers, which implements the notion of out-of-sample estimation, focusing on the actual Data Generating Process. Even in the case of being interested in the evaluation of a particular sample of units, the researchers might show interest in finding out, for example, whether the units identified as technically efficient at a relative to the sample level are also out-of-sample efficient or not. Again, in this situation, techniques as the new approach introduced in this paper might be of some help. Additionally, the use of SVF should always be accompanied by a preliminary descriptive analysis of the data at a frontier level. FDH and DEA would be, in this case, suitable tools to be applied.

We finish this section by mentioning several lines that state interesting avenues for further research with Support Vector Frontiers. The first one is the possibility of extending the new technique to the context of producing multiple outputs. In this respect, Vazquez and Walter [57] extended SVR by considering the so-called Cokriging method, which is a multi-output version of Kriging that exploits the

correlations due to the proximity in the space of factors and outputs (see also the survey by [13]). Moreover, in the same line, the new methodology could also be applied for estimating the cost frontier in the general multiple output context. While Briec et al. [14] have developed the theoretical case for distinguishing between DEA- and FDH-based cost frontiers, Balaguer-Coll et al. [8] and Narbón-Perpiñá et al. [43,44] have convincingly shown that these distinctions definitely make a difference from an empirical point of view. Additionally, we resorted in this paper to the L1-norm to define a linear optimization program, in the line of FDH and DEA. However, it is possible to use other norms to improve the results associated with the accuracy of the SVM estimator (see, for example, [12]). Nowadays, large sample properties of FDH and DEA estimators are well-known in the literature (see, for example, [53]). Support Vector Machines are known to be consistent and robust for classification and regression under certain hypotheses defined on the used loss function and kernel (see, for example, [18]). Consequently, in the new framework introduced by the SVF model, where some new constraints are incorporated to the standard SVR model, it is necessary to gain an in-depth understanding on whether the same results become true or not. Another possibility is to extend the new approach for dealing with Constant Returns to Scale and other types of returns to scale. Finally, an evident research line to be followed is the application of the new approach to real databases in different empirical contexts, thus checking the validity of the technique in practice.

**Declaration of Competing Interest**

No.

**CRediT authorship contribution statement**

**Daniel Valero-Carreras:** Methodology, Software, Data curtion, Visualization, Investigation, Validation, Writing – original draft, Writing – review & editing. **Juan Aparicio:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing, Funding acquisition. **Nadia M. Guerrero:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing.

**Acknowledgments**

We thank four anonymous reviewers for providing constructive comments and help in improving the contents and presentation of this paper. Additionally, the authors are grateful for the financial support from the Spanish Ministry for Economy and Competitiveness, the State Research Agency and the European Regional Development Fund under grant PID2019-105952GB-I00. This work was also supported by the *Generalitat Valenciana* under Grant ACIF/2020/155.

**Appendix**

**Proof of Proposition 1.** Let  $f : \mathbb{R}_+^m \rightarrow \mathbb{R}$  a step function on a grid  $G$ . Then, we have that  $f(x) = f(C_{l_1 \dots l_m}) \forall x \in C_{l_1 \dots l_m}, \forall l_1 = 1, \dots, k_1, \dots, \forall l_m = 1, \dots, k_m$ . Now, we need to find out whether the following system has a solution or not.

$$\sum_{s_1=1, \dots, l_1} w_{s_1 \dots s_m} = f(C_{l_1 \dots l_m}), l_1 = 1, \dots, k_1, \dots, l_m = 1, \dots, k_m \tag{12}$$

$$\vdots$$

$$s_m=1, \dots, l_m$$

The expression in (12) represents  $k_1 \cdot \dots \cdot k_m$  equations. The equations corresponding to  $l_1 = l_2 = \dots = l_{m-1} = 1$  and  $l_m = 1, \dots, k_m$  are, in matrix format, as follows.

$$\underbrace{\begin{pmatrix} 1 & 0 & & 0 \\ 1 & 1 & & 0 \\ \vdots & \vdots & \ddots & \\ 1 & 1 & \dots & 1 \end{pmatrix}}_{A_{1 \dots 1 l_m}} \underbrace{\begin{pmatrix} w_{1 \dots 11} \\ w_{1 \dots 12} \\ \vdots \\ w_{1 \dots 1 k_m} \end{pmatrix}}_{\hat{W}_{1 \dots 1 l_m}} = \underbrace{\begin{pmatrix} f(C_{1 \dots 11}) \\ f(C_{1 \dots 12}) \\ \vdots \\ f(C_{1 \dots 1 k_m}) \end{pmatrix}}_{F(C_{1 \dots 1 l_m})}, \tag{13}$$

with  $|A_{1 \dots 1 l_m}| = 1$  because  $A_{1 \dots 1 l_m}$  is a lower triangular matrix (the determinant of a triangular matrix is the product of the elements of its diagonal). Regarding the system of equations for  $l_1 = l_2 = \dots = l_{m-2} = 1, l_{m-1} = 1, \dots, k_{m-1}$  and  $l_m = 1, \dots, k_m$ , we have the following



there exist  $\bar{x}_{1_j'}^{(j')}$  and  $\bar{x}_{1_{j'+1}}^{(j')}$ , successive elements in the ordered set of values  $\{\bar{x}_{1_j'}^{(j')}\}_{j'=1}^n$  such that  $\bar{x}_{1_j'}^{(j')} < x_i^{(j')} < \bar{x}_{1_{j'+1}}^{(j')}$ , with  $\bar{x}_{1_j'}^{(j')} = a_{1 \dots l_m}^{E(j')}$  and  $\bar{x}_{1_{j'+1}}^{(j')} = b_{1 \dots l_m}^{E(j')}$ . However, this is a contradiction with the fact that  $x_i^{(j')} \in \{\bar{x}_{1_j'}^{(j')}\}_{j'=1}^n$ . Consequently,  $x_i \leq a_{1 \dots l_m}^E$ . ■

**Proof of Theorem 2.** If  $x \in C_{1 \dots l_m}^E$ , then  $f_{FDH}(x) = \max_{i: x \geq x_i} \{y_i\} = \max_{i \in X} \{y_i\} = \max_{i \in A_{1 \dots l_m}^E} \{y_i\} = \max_{i: x_i \leq a_{1 \dots l_m}^E} \{y_i\} = f_{FDH}(a_{1 \dots l_m}^E)$ , where the third equality is true by Proposition 3, the last equality by the monotonicity of the FDH function and the other equalities by definition. ■

**Proof of Theorem 3.** Conditions (9.1) are trivially satisfied because  $w^{FDH} \phi_{SVF}^{GE}(x) = f_{FDH}(x)$  is an enveloping function of the data. Constraints (9.2) and (9.4) hold thanks to the definition of the empirical errors  $\xi_i^{FDH}, \forall i = 1, \dots, n$ . Additionally,  $f_{FDH}$  is a recoverable function through the parameterized grid  $G^E$  with parameters  $w^{FDH}$ . Then, given that it is well-known that  $f_{FDH}$  is a monotonic non-decreasing function, invoking Theorem 1 and Proposition 2, conditions (9.3) are satisfied by the vector of parameters  $w^{FDH}$ . ■

**Proof of Lemma 4.** Let  $i' = 1, \dots, n$  be such that  $y_{i'} = \max\{y_1, \dots, y_n\}$ . Then, by (9.1),  $w' \phi_{SVF}^{GE}(x_{i'}) \geq y_{i'}$ . By (9.3),  $h(x) := w' \phi_{SVF}^{GE}(x)$  is a monotonic non-decreasing function. Therefore,  $\max_{1 \leq i \leq n} \{x_i\} \geq x_{i'}$  implies  $w' \phi_{SVF}^{GE}(\max\{x_i\}) \geq w' \phi_{SVF}^{GE}(x_{i'}) \geq y_{i'} = \max\{y_1, \dots, y_n\}$ . ■

**Proof of Theorem 4.** Let  $(w', \xi'_1, \dots, \xi'_n)$  be any feasible solution of model (9). By (9.3),  $w'_1 \leq w'_1 + w'_2, w'_1 + w'_2 \leq w'_1 + w'_2 + w'_3, \dots, w'_1 + w'_2 + w'_3 + \dots + w'_{n(1)-1} \leq w'_1 + w'_2 + w'_3 + \dots + w'_{n(1)}$ . Therefore,  $w'_1 \geq 0, i = 2, \dots, n^{(1)}$ . Let  $i', i'' = 1, \dots, n$  be such that  $x_{i'} = \min_{1 \leq i \leq n} \{x_i\}$  and  $x_{i''} = \max\{x_i\}$ . By (9.1),  $w' \phi_{SVF}^{GE}(x_{i'}) \geq y_{i'}$ , which is equivalent to  $w'_1 \geq y_{i'} \geq 0$ . Regarding the objective function in (9),  $\|w'\|_1 + C \sum_{i=1}^n \xi'_i = w'_1 + \dots + w'_{n(1)} + C \sum_{i=1}^n \xi'_i = w' \phi_{SVF}^{GE}(x_{i''}) + C \sum_{i=1}^n \xi'_i$ . Additionally, it is well-known that  $f_{FDH}(x_{i''}) = \max\{y_1, \dots, y_n\}$ . Then,  $w^{FDH} \phi_{SVF}^{GE}(x_{i''}) = f_{FDH}(x_{i''}) = \max\{y_1, \dots, y_n\}$ , which is, by Lemma 4, the minimum value that  $w' \phi_{SVF}^{GE}(x_{i''})$  can take for any feasible solution  $(w', \xi'_1, \dots, \xi'_n)$  of (9). Moreover, given any feasible solution  $(w', \xi'_1, \dots, \xi'_n)$ ,  $\xi'_i = \max\{w' \phi_{SVF}^{GE}(x_i) - \varepsilon - y_i, 0\}$  because we are minimizing (9.0), which includes the term  $C \sum_{i=1}^n \xi'_i$ , and also thanks to constraints (9.2). Finally, note that, by the principle of minimal extrapolation,  $f_{FDH}(x_i) = w^{FDH} \phi_{SVF}^{GE}(x_i) \leq w' \phi_{SVF}^{GE}(x_i), \forall i = 1, \dots, n$ , which is equivalent to  $(w^{FDH} \phi_{SVF}^{GE}(x_i) - \varepsilon) - y_i \leq (w' \phi_{SVF}^{GE}(x_i) - \varepsilon) - y_i, \forall i = 1, \dots, n$ . These last inequalities imply that  $\xi_i^{FDH} \leq \xi'_i, i = 1, \dots, n$ , for any feasible solution  $(w', \xi'_1, \dots, \xi'_n)$ . Consequently,  $w^{FDH} \phi_{SVF}^{GE}(x_{i''}) + C \sum_{i=1}^n \xi_i^{FDH} \leq w' \phi_{SVF}^{GE}(x_{i''}) + C \sum_{i=1}^n \xi'_i$  for any feasible solution  $(w', \xi'_1, \dots, \xi'_n)$ , which means that  $(w^{FDH}, \xi_1^{FDH}, \dots, \xi_n^{FDH})$  is an optimal solution of model (9). ■

**References**

- [1] Afriat SN. Efficiency estimation of production functions. *Int Econ Rev* 1972;568–98.
- [2] Aigner DJ, Chu SF. On estimating the industry production function. *Am Econ Rev* 1968;58(4):826–39.
- [3] Aparicio J, Pastor JT, Vidal F, Zofio JL. Evaluating productive performance: a new approach based on the product-mix problem consistent with Data Envelopment Analysis. *Omega* 2017;67:134–44.
- [4] Aragon Y, Daouia A, Thomas-Agnan C. Nonparametric frontier estimation: a conditional quantile-based approach. *Econom Theory* 2005;358–89.
- [5] Arnaboldi M, Azzone G, Giorgino M. Performance measurement and management for engineers. Academic Press; 2014.
- [6] Arreola JLP, Yagi D, Johnson AL. Insights from machine learning for evaluating production function estimators on manufacturing survey data. *J Prod Anal* 2019;1–45.
- [7] Badieezadeh T, Saen RF, Samavati T. Assessing sustainability of supply chains by double frontier network DEA: a big data approach. *Comput Oper Res* 2018;98:284–90.
- [8] Balaguer-Coll MT, Prior D, Tortosa-Ausina E. On the determinants of local government performance: a two-stage nonparametric approach. *Eur Econ Rev* 2007;51(2):425–51.
- [9] Banker RD, Charnes A, Cooper WW. Some models for estimating technical and scale inefficiencies in data envelopment analysis. *Manag Sci* 1984;30(9):1078–92.
- [10] Banker RD, Maindiratta A. Maximum likelihood estimation of monotone and concave production frontiers. *J Prod Anal* 1992;3(4):401–15.
- [11] Banker RD. Maximum likelihood, consistency and data envelopment analysis: a statistical foundation. *Manag Sci* 1993;39(10):1265–73.
- [12] Blanco V, Puerto J, Rodríguez-Chía AM. On lp-support vector machines and multidimensional kernels. *J Mach Learn Res* 2020;21(14):1–29.
- [13] Borchani H, Varando G, Bielza C, Larrañaga P. A survey on multi-output regression. *Wiley Interdisc Rev* 2015;5(5):216–33.
- [14] Briec W, Kerstens K, Eeckaut PV. Non-convex technologies and cost functions: definitions, duality and nonparametric tests of convexity. *J Econ* 2004;81(2):155–92.
- [15] Charles V, Aparicio J, Zhu J. Data science and productivity analytics. Springer; 2020.
- [16] Charnes A, Cooper WW, Rhodes E. Measuring the efficiency of decision making units. *Eur J Oper Res* 1978;2(6):429–44.
- [17] Chen Z, Matousek R, Wanke P. Chinese bank efficiency during the global financial crisis: a combined approach using satisficing DEA and Support Vector Machines\*. *North Am J Econ Finance* 2018;43:71–86.
- [18] Christmann A, Steinwart I, van Messem A. On consistency and robustness properties of support vector machines for heavy-tailed distributions. *Stat Interface* 2009;2(3):311–27.
- [19] Cristianini N, Shawe-Taylor J. An introduction to support vector machines and other Kernel-based learning methods. Cambridge: Cambridge University Press; 2000.
- [20] Daraio C, Simar L. Introducing environmental variables in nonparametric frontier models: a probabilistic approach. *J Prod Anal* 2005;24(1):93–121.
- [21] Daraio C, Simar L. Advanced robust and nonparametric methods in efficiency analysis: methodology and applications. Springer Science & Business Media; 2007.
- [22] Deprins, D., & Simar, L. (1984). Measuring labor efficiency in post offices, the performance of public enterprises: concepts and measurements, M. Marchand, P. Pestieau and H. Tulkens.
- [23] Du P, Parmeter CF, Racine JS. Nonparametric Kernel regression with multiple predictors and multiple shape constraints. *Stat Sin* 2013;1347–71.
- [24] Efron B. Bootstrap methods: another look at the Jackknife. *Ann Stat* 1979;7(1):1–26.
- [25] Esteve M, Aparicio J, Rabasa A, Rodríguez-Sala JJ. Efficiency analysis trees: a new methodology for estimating production frontiers through decision trees. *Expert Syst Appl* 2020;113783.
- [26] Farahmand M, Desa MI, Nilashi M. A combined data envelopment analysis and support vector regression for efficiency evaluation of large decision making units. *Int J Eng Technol (IJET)* 2014;2310–21.
- [27] Färe R, Primont D. Multi-output production and duality: theory and applications. Kluwer Academic Publishers; 1995.
- [28] Farrell MJ. The measurement of productive efficiency. *J R Stat Soc: Series A (General)* 1957;120(3):253–81.
- [29] Hall P, Huang LS. Nonparametric kernel regression subject to monotonicity constraints. *Ann Stat* 2001;624–47.

- [30] Hatami-Marbini A, Emrouznejad A, Tavana M. A taxonomy and review of the fuzzy data envelopment analysis literature: two decades in the making. *Eur J Oper Res* 2011;214(3):457–72.
- [31] Henderson DJ, Parmeter CF. Imposing economic constraints in nonparametric regression: survey, implementation, and extension. *Adv Econom* 2009;25:433–69.
- [32] Hong HK, Ha SH, Shin CK, Park SC, Kim SH. Evaluating the efficiency of system integration projects using data envelopment analysis (DEA) and machine learning. *Expert Syst Appl* 1999;16(3):283–96.
- [33] Kao HY, Chang TK, Chang YC. Classification of hospital web security efficiency using data envelopment analysis and support vector machine. *Math Probl Eng* 2013;2013.
- [34] Kerstens K, O'donnell C, Van de Woestyne I. Metatechnology frontier and convexity: a restatement. *Eur J Oper Res* 2019;275(2):780–92.
- [35] Khezrimotlagh D, Zhu J, Cook WD, Toloo M. Data envelopment analysis and big data. *Eur J Oper Res* 2019;274(3):1047–54.
- [36] Kuosmanen T, Johnson AL. Data envelopment analysis as nonparametric least-squares regression. *Oper Res* 2010;58(1):149–60.
- [37] Kuosmanen T, Johnson A. Modeling joint production of multiple outputs in StoNED: directional distance function approach. *Eur J Oper Res* 2017;262(2):792–801.
- [38] Lee CY, Cai JY. LASSO variable selection in data envelopment analysis with small datasets. *Omega* 2020;91:102019.
- [39] Li Y, Wang L, Li F. A data-driven prediction approach for sports team performance and its application to National Basketball Association. *Omega* 2021;98:102123.
- [40] Mirmozaffari M, Yazdani M, Boskabadi A, Dolatsara HA, Kabirifar K, Golilarz NA. A novel machine learning approach combined with optimization models for eco-efficiency evaluation. *Appl Sci* 2020;10(15):5210.
- [41] Misiunas N, Oztekin A, Chen Y, Chandra K. DEANN: A healthcare analytic methodology of data envelopment analysis and artificial neural networks for the prediction of organ recipient functional status. *Omega* 2016;58:46–54.
- [42] Mitchell T. *Machine learning*. New York: McGraw Hill; 1997.
- [43] Narbón-Perpiñá I, Balaguer-Coll M, Tortosa-Ausina E. Evaluating local government performance in times of crisis. *Local Gov Stud* 2019;45(1):64–100.
- [44] Narbón-Perpiñá I, Balaguer-Coll MT, Petrović M, Tortosa-Ausina E. Which estimator to measure local governments' cost efficiency?. In: *The case of Spanish municipalities*. *SERIEs*; 2020. p. 1–32.
- [45] O'Donnell CJ. *Productivity and efficiency analysis*. Singapore: Springer; 2018.
- [46] Olesen OB, Petersen NC. Stochastic data envelopment analysis—a review. *Eur J Oper Res* 2016;251(1):2–21.
- [47] Parmeter CF, Sun K, Henderson DJ, Kumbhakar SC. Estimation and inference under economic restrictions. *J Prod Anal* 2014;41(1):111–29.
- [48] Penderhark PC, Troutt MD. Interactive classification using data envelopment analysis. *Ann Oper Res* 2014;214(1):125–41.
- [49] Poitier K, Cho S. Estimation of true efficient frontier of organisational performance using data envelopment analysis and support vector machine learning. *Int J Inf Decis Sci* 2011;3(2):148–72.
- [50] Simar L, Wilson PW. Sensitivity analysis of efficiency scores: how to bootstrap in nonparametric frontier models. *Manag Sci* 1998;44(1):49–61.
- [51] Simar L, Wilson PW. A general methodology for bootstrapping in non-parametric frontier models. *J Appl Statist* 2000;27(6):779–802.
- [52] Simar L, Wilson PW. Statistical inference in nonparametric frontier models: The state of the art. *J Prod Anal* 2000;13(1):49–78.
- [53] Simar, L., & Wilson, P. W. (2008). Statistical inference in nonparametric frontier models: recent developments and perspectives. *The measurement of productive efficiency and productivity growth*, 421–521.
- [54] Song J, Zhang Z. Oil refining enterprise performance evaluation based on DEA and SVM. In: *2009 Second international workshop on knowledge discovery and data mining*. *IEEE*; 2009. p. 401–4.
- [55] Vapnik V. *The nature of statistical learning theory*. New York: Springer; 1995.
- [56] Vapnik V. *Statistical learning theory*. New York: Wiley; 1998.
- [57] Vazquez E, Walter E. Multi-output support vector regression. *IFAC Proceed Vol* 2003;36(16):1783–8.
- [58] Yeh CC, Chi DJ, Hsu MF. A hybrid approach of DEA, rough set and support vector machines for business failure prediction. *Expert Syst Appl* 2010;37(2):1535–41.
- [59] Zhu J. DEA under big data: data enabled analytics and network data envelopment analysis. *Ann Oper Res* 2019:1–23.
- [60] Zhu Q, Wu J, Song M. Efficiency evaluation based on data envelopment analysis in the big data context. *Comput Oper Res* 2018;98:291–300.