European Journal of Operational Research xxx (xxxx) xxx

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Analytics, computational intelligence and information management

A novel robust optimization model for nonlinear Support Vector Machine

Francesca Maggioni *, Andrea Spinelli

Department of Management, Information and Production Engineering, University of Bergamo, Viale G. Marconi 5, Dalmine 24044, Italy

ARTICLE INFO

Keywords:
Machine learning
Nonlinear Support Vector Machine
Robust optimization

ABSTRACT

In this paper, we present new optimization models for Support Vector Machine (SVM), with the aim of separating data points in two or more classes. The classification task is handled by means of nonlinear classifiers induced by kernel functions and consists in two consecutive phases: first, a classical SVM model is solved, followed by a linear search procedure, aimed at minimizing the total number of misclassified data points. To address the problem of data perturbations and protect the model against uncertainty, we construct bounded-by-norm uncertainty sets around each training data and apply robust optimization techniques. We rigorously derive the robust counterpart extension of the deterministic SVM approach, providing computationally tractable reformulations. Closed-form expressions for the bounds of the uncertainty sets in the feature space have been formulated for typically used kernel functions. Finally, extensive numerical results on real-world datasets show the benefits of the proposed robust approach in comparison with various SVM alternatives in the machine learning literature.

1. Introduction

Support Vector Machine (SVM) is one of the main supervised *Machine Learning* (ML) techniques commonly deployed for classification and regression purposes. Within the *Operational Research* (OR) domain, supervised ML methods are designed to support better decision-making and solve hard optimization problems (Gambella, Ghaddar, & Naoum-Sawaya, 2021). To this end, a plethora of methodologies have been devised and applied to various OR fields (De Bock et al., 2024). In particular, combinatorial optimization (Bengio, Lodi, & Prouvost, 2021; Wei, Hao, Ren, & Glover, 2023), customer churn prediction (Benítez-Peña, Blanquero, Carrizosa, & Ramírez-Cobo, 2024; Chen, Fan, & Sun, 2012; Maldonado, López, & Vairetti, 2020; Szelag & Słowiński, 2024), banking (Doumpos, Zopounidis, Gounopoulos, Platanakis, & Zhang, 2023; Katsafados, Leledakis, Pyrgiotakis, Androutsopoulos, & Fergadiotis, 2024; Yao, Crook, & Andreeva, 2017) and maritime industry (Mi et al., 2019; Raeesi, Sahebjamnia, & Mansouri, 2023).

Currently, deep learning algorithms are adopted whenever classical ML methods fail to capture complex relationships between input data both for classification and regression tasks (Gambella et al., 2021). Nevertheless, the advantage of mathematical programming approaches to model deep neural networks has been explored only for small-sized datasets, and without a guarantee on the effectiveness of the performance (Gunnarsson, vanden Broucke, Baesens, Óskarsdóttir, & Lemahieu, 2021). For this reason, the investigation of novel ML techniques is a relevant ongoing research issue (Maldonado, López, & Carrasco, 2022).

Introduced in Vapnik and Chervonenkis (1974), SVM has outperformed most other ML systems, due to its simplicity and better performance. Therefore, it has been applied in many practical research fields, such as finance (Luo, Yan, & Tian, 2020; Tay & Cao, 2001), chemistry (Li, Liang, & Xu, 2009; Marcelli & De Leone, 2020), medicine (Maggioni, Faccini, Gheza, Manelli, & Bonetti, 2023; Wang, Zheng, Yoon and Ko, 2018), and vehicles smog rating classification (De Leone, Maggioni, & Spinelli, 2024; Maggioni & Spinelli, 2024), to name a few.

Hard Margin-SVM (HM-SVM) is the original approach formulated in Vapnik and Chervonenkis (1974), consisting in finding a hyperplane classifying observations into two classes, such that the margin, i.e. the ℓ_2-distance from the hyperplane to the nearest point of each class, is maximized. The underlying hypothesis of the HM-SVM is that training data can always be linearly separated, such that no observation is misclassified. To overcome the assumption of linear separability, in Cortes and Vapnik (1995) the *Soft Margin*-SVM (SM-SVM) is proposed. In this case, the optimal hyperplane seeks a trade-off between the maximization of the margin and the minimization of the training error of misclassification.

In order to improve the accuracy of the method, several SVM variants have been devised in the literature. Specifically, in this paper we focus our attention on the one presented in Liu and Potra (2009). The advantages of this technique over other SVM approaches are mainly due to a two-step procedure. Indeed, rather than considering a

* Corresponding author.
E-mail address: francesca.maggioni@unibg.it (F. Maggioni).

https://doi.org/10.1016/j.ejor.2024.12.014
Received 17 August 2023; Accepted 9 December 2024
Available online 20 December 2024

Please cite this article as: Francesca Maggioni, Andrea Spinelli, *European Journal of Operational Research*, https://doi.org/10.1016/j.ejor.2024.12.014

single hyperplane, training data are firstly separated by means of two parallel hyperplanes as solutions of a SM-SVM model. The final optimal hyperplane is then searched in the strip between them, such that the total number of misclassified points is minimized. Compared to classical SM-SVM, numerical experiments show that this formulation achieves higher levels of computational accuracy.

Nevertheless, training observations may not be always separable by means of hyperplanes and, even with *ad hoc* variants of linear SVM, the misclassification error may be significant. In Boser, Guyon, and Vapnik (1992), the extension of the linear HM-SVM model is introduced, by considering nonlinear transformation of the data. According to this technique, kernel functions are used to embed data points onto a higher-dimensional space (the so-called *feature space*), without increasing the computational complexity of the problem. Several variants of this methodology have been proposed in the ML literature (see for example Bennett & Mangasarian, 1992; Blanco, Puerto, & Rodríguez-Chía, 2020; Cervantes, Garcia-Lamont, Rodríguez-Mazahua, & Lopez, 2020; Ding & Hua, 2014; Ding, Zhao, Zhang, Zhang, & Xue, 2019; Du et al., 2021; Gao, Fang, Luo, & Medhin, 2021; Hao, 2010; Jayadeva, Khemchandani, & Chandra, 2007; Jiménez-Cordero, Morales, & Pineda, 2021; Mangasarian, 1998; Peng, 2011; Schölkopf, Smola, Williamson, & Bartlett, 2000; Yajima, 2005).

For the methods mentioned above, all data points are implicitly assumed to be known exactly. However, in real-world observations this condition may not be always true. Indeed, measurement errors during data collection, random perturbations, presence of noise and other forms of uncertainty may corrupt the quality of input values, resulting in worsening performance of the classification process. In recent years, different techniques have been investigated with the aim of facing uncertainty in ML methods. Among them, *Robust Optimization* (RO) is recognized as one of the main paradigms to protect optimization models against uncertainty (see for example Ben-Tal, El Ghaoui, & Nemirovski, 2009; Bertsimas, Brown, & Caramanis, 2011; Xu, Caramanis, & Mannor, 2009). RO assumes that all possible realizations of the uncertain parameter belong to a prescribed uncertainty set. The corresponding robust model is then derived by optimizing against the worst-case realization of the parameter across the entire uncertainty set (Bertsimas, Dunn, Pawlowski, & Zhuo, 2019). The application of RO strategies typically results in higher predictiveness (Faccini, Maggioni, & Potra, 2022; Maldonado et al., 2020). For this reason, it is worth designing novel RO models with the aim of improving the accuracy of the classification process.

In this paper, we present novel SVM models aiming at separating classes of data points. The formulation extends the approach of Liu and Potra (2009) to the context of multiclass and nonlinear classification. In order to protect the model against perturbations, we introduce bounded-by-norm uncertainty sets around each training observation and rigorously derive the robust counterpart of the deterministic approach, providing computationally tractable reformulations. In addition, our proposal represents a valid contribution to the state of the art on SVM thanks to the computation of the uncertainty set bounds in the feature space as function of the bounds in the input space. This is a novel development in the ML domain.

The main contributions of the paper are four-fold and can be summarized as follows:

- To extend the binary linear SVM approach of Liu and Potra (2009) to the cases of multiclass and nonlinear classification;
- To formulate the robust extension of the SVM model with nonlinear classifiers using bounded-by-ℓ_p-norm uncertainty sets and provide computationally tractable reformulations;
- To rigorously derive bounds on the radii of the uncertainty sets in the feature space for some of the most used kernel functions in the ML literature;
- To provide extensive numerical experiments based on real-world datasets with the aim of evaluating the performance of the proposed models and comparing the results with extant SVM methods in the literature.

The remainder of the paper is organized as follows. Section 2 reviews the existing literature on the problem. In Section 3, the notation is introduced, along with a brief discussion on related SVM-type problems. In Section 4, the novel deterministic model with nonlinear classifier is introduced for both binary and multiclass classification. Section 5 considers the robust extension together with the construction of the uncertainty sets. In Section 6, the computational results are shown. Finally, Section 7 concludes the paper and discusses future works.

2. Literature review

The nonlinear SVM approach presented in Boser et al. (1992) has been explored in several works, leading to alternative formulations. In Lee, Mangasarian, and Wolberg (2000) and Mangasarian (1998) a kernel-induced decision boundary is derived by considering quadratic and piecewise-linear objective function, resulting in a convex model. In Schölkopf et al. (2000) the formulation of *ν-Support Vector Classification* (ν-SVC) is proposed for both linear and nonlinear classifiers. This algorithm differs from the classical SVM paradigm of Vapnik (1995) since it involves a new parameter ν in the objective function, controlling the number of support vectors. In Jayadeva et al. (2007) the *TWin Support Vector Machine* (TWSVM) is designed. Contrary to standard SVM, TWSVM determines a pair of nonparallel hyperplanes by solving two small-sized SVM-type problems. TWSVM is combined in Peng (2011) with a flexible parametric margin model (Hao, 2010), deriving the *Twin Parametric Margin Support Vector Machine* (TPMSVM). Recently, in Blanco et al. (2020) the classical ℓ_2-norm problem has been extended to the general case of ℓ_p-norm with $p > 1$, resulting in a *Second-Order Cone Programming* formulation (SOCP, Maggioni, Potra, Bertocchi, & Allevi, 2009). Within the field of *Double Well Potential* functions (DWP), a kernel-free DWP model for SVM is derived in Gao et al. (2021) for classifying nonlinearly separable data. The problem of feature selection in nonlinear SVM is explored in Jiménez-Cordero et al. (2021), where a method based on a min–max optimization model is proposed. With respect to the extant literature on nonlinear SVM, the first contribution of this work is the extension of the linear SVM variant developed in Liu and Potra (2009) to the case of nonlinear classifiers. The model benefits from such extension since it handles cases of nonlinearly separable data with a low misclassification error.

In order to prevent low accuracies in the classification process when training data are plagued by uncertainty, RO techniques are applied in the SVM context (Wang & Pardalos, 2014). In Bhattacharyya (2004) hyperellipsoids around data points are considered, and the robust model results in a SOCP problem. A tractable robust counterpart of the linear SM-SVM approach is derived in Bertsimas et al. (2019). The authors robustify the model by considering additive and bounded-by-norm perturbations in the training data. In El Ghaoui, Lanckriet, Natsoulis, et al. (2003) the binary classification problem under feature uncertainty is formulated with uncertainty sets in the form of hyper-rectangles and hyperellipsoids around input data. The same choices of uncertainty sets is made in Faccini et al. (2022), where the RO extension of the linear SVM variant presented in Liu and Potra (2009) is proposed. In this work, we further extend such approach by formulating a robust SVM model tailored for a general class of bounded-by-ℓ_p-norm uncertainty sets. This improves the generalization capability of the model as the choice of the ℓ_p-norm can be made according to the information available on the training dataset and the desired degree of conservatism.

As far as it concerns RO techniques applied to nonlinear SVM, various approaches exist in the literature. In Ben-Tal, Bhadra, Bhattacharyya, and Nemirovski (2012) and Bhadra, Bhattacharya, Bhattacharyya, and Ben-Tal (2010) the kernel matrix is assumed to be affected by uncertainty, due to feature perturbations in the input data. Such matrix is decomposed as a linear combination of positive

semidefinite matrices with bounded-by-ℓ_p-norm coefficients. The main limitation of this approach is that the functional form of the matrices in the combination is typically unknown. Thus, it is not obvious how to characterize the elements in the uncertainty set, unless by using a sampling procedure. In Bi and Zhang (2005) and Trafalis and Gilbert (2006) training data points are subject to uncertain but bounded-by-ℓ_p-norm perturbations. Robustified models are derived for both linear and nonlinear classifiers. A related work on bounded-by-norm uncertainty sets is Xu et al. (2009), where a link between regularization and robustness is provided. In Trafalis and Alwazzi (2010) the stability of SVM models with bounded perturbations is investigated by using discriminant functions. Polyhedral uncertainty sets are considered in Fan, Sadeghi, and Pardalos (2014), Fung, Mangasarian, and Shavlik (2002) and Ju and jie Tian (2012), based on the nonlinear classifier proposed in Mangasarian (1998). In all these works on robust SVM with nonlinear classifiers, only the case with Gaussian kernel has been investigated. Indeed, for such kernel there exists a closed-form expression for the radius of the uncertainty set in the feature space based on the corresponding one in the input space (see Xu et al., 2009). In this paper, we prove further theoretical results valid for other classes of kernels, i.e. homogeneous and inhomogeneous polynomial kernels. These findings are beneficial for all robust SVM models with bounded-by-ℓ_p-norm uncertainty sets and kernel-induced classifiers.

RO techniques are also applied to variants of the classical SVM model. In Peng and Xu (2013) a robust TWSVM classifier is proposed, by including uncertainty in the variance matrices of the two classes. In Qi, Tian, and Shi (2013) the robust extension of TWSVM is derived. For the nonlinear case, only Gaussian kernel and ellipsoidal uncertainty sets are considered, resulting in a SOCP formulation. In De Leone, Maggioni, and Spinelli (2023) the robust and multiclass extension of the TPMSVM is provided. A complete survey on recent developments on TWSVM models can be found in Tanveer, Rajani, Rastogi, and Shao (2022).

When partial or complete information on the probability distribution of the training data are available, other solution techniques dealing with uncertainties such as *Chance-Constrained Programming* (CCP) and *Distributionally Robust Optimization* (DRO) have been considered in the SVM literature (Jiang & Peng, 2024; Ketkov, 2024). The *Minimax Probability Machine* (MPM) is the first distributionally robust SVM approach that minimizes the worst-case probability of misclassification (Lanckriet, Ghaoui, Bhattacharyya, & Jordan, 2002). In Maldonado et al. (2020) the MPMs are extended and applied to the robust profit-driven churn prediction. Within the MPM framework, the use of Cobb–Douglas function for maximizing the expected class accuracies under a worst-case distribution setting is proposed in Maldonado et al. (2022). As far as it concerns DRO methods applied to SVM, we mention the recent work of Faccini et al. (2022) where a moment-based distributionally robust extension of the Liu and Potra (2009) formulation is designed. The problem of robust feature selection with CCP is explored in López, Maldonado, and Carrasco (2018) by using difference of convex functions. Within the multiclass context, in López, Maldonado, and Carrasco (2017) a robust CCP formulation for multiclass classification via TWSVM is proposed. Finally, a combination of CCP and DRO techniques applied to linear and nonlinear SVM models with uncertain data is explored in Khanjani-Shiraz, Babapour-Azar, Hosseini-Nodeh, and Pardalos (2023), Wang, Fan and Pardalos (2018) and in Lin, Fang, Fang and Gao (2024a), Lin, Fang, Fang, Gao and Luo (2024b), respectively.

All the approaches discussed so far are listed in Table 1. For a comprehensive review of RO techniques applied to SVM models the reader is referred to Singla, Ghosh, and Shukla (2020).

In summary, the contributions of this paper differ from the literature described above in several aspects. First of all, we present a novel optimization model with nonlinear classifiers, extending the approach of Liu and Potra (2009), both in the case of binary and multiclass classification. Secondly, we consider general bounded-by-ℓ_p-norm uncertainty sets around training observations. This increases the flexibility of the model, adapting the formulation to more complex perturbations in input data. In addition, it results in a generalization of the robust approach of Faccini et al. (2022) where only box and ellipsoidal uncertainty sets have been considered. Thirdly, we derive closed-form expressions of the bounds in the feature space for some of typically used kernel functions in ML literature. Finally, we deduce the robust counterpart of the deterministic formulations, protecting the models against data uncertainty.

3. Background and notation

In this section, we report the notation (Section 3.1) and briefly recall the methods that are relevant for our proposal (Section 3.2).

3.1. Notation

In the following, the set of nonnegative real numbers will be denoted by $\mathbb{R}^+$, whereas if zero is excluded we write $\mathbb{R}_0^+$. Hereinafter, all vectors will be column vectors, unless transposition by the superscript "$\top$". If a is a vector in $\mathbb{R}^n$, then its ith component will be denoted by a_i. The scalar product in a inner product space $\mathcal{H}$ will be denoted by $\langle\cdot,\cdot\rangle$. If $\mathcal{H}=\mathbb{R}^n$ and $a,b\in\mathbb{R}^n$, the dot product will be indifferently denoted as $a^\top b$ or $\langle a,b\rangle$. For $p\in[1,\infty]$, $\|a\|_p$ is the ℓ_p-norm of a. Finally, if $c\in\mathbb{R}$, the indicator function $\mathbb{1}(c)$ has value 1 if c is positive and 0 otherwise. By convention, we assume that $\frac{1}{\infty}:=0$ and $\|a\|_\infty^\infty:=\|a\|_\infty$.

3.2. A selected review of SVM models

Let $\{(x^{(i)},y^{(i)})\}_{i=1}^m$ be the set of training data points, where $x^{(i)}\in\mathbb{R}^n$ is the vector of features, and $y^{(i)}\in\{-1,+1\}$ is the label representing the class to which the ith data point belongs. In particular, we denote by $\mathcal{A}$ and $\mathcal{B}$ the *positive* (label "+1") and *negative* (label "−1") classes, respectively.

The *Soft Margin*-SVM approach (SM-SVM, Cortes & Vapnik, 1995) finds the best separating hyperplane $H:=(w,\gamma)$ defined by the equation $w^\top x=\gamma$, where $w\in\mathbb{R}^n$ and $\gamma\in\mathbb{R}$, as solution of the following ℓ_q-model, $q\in[1,\infty]$:

$$\begin{aligned}\min_{w,\gamma,\xi}\quad & \|w\|_q^q+\nu\sum_{i=1}^m\xi_i\\ \text{s.t.}\quad & y^{(i)}(w^\top x^{(i)}-\gamma)\geqslant 1-\xi_i \qquad i=1,\dots,m\\ & \xi_i\geqslant 0 \qquad\qquad\qquad\quad i=1,\dots,m.\end{aligned}\tag{1}$$

The vector $\xi\in\mathbb{R}^m$ is the soft margin error vector and $\nu\geqslant 0$ is a regularization parameter. Data point $x^{(i)}$ is correctly classified by the separating hyperplane H if $0\leqslant\xi_i\leqslant 1$, otherwise is misclassified.

Whenever a new observation $x\in\mathbb{R}^n$ occurs, it is classified as *positive* or *negative* depending on the decision function $\mathbb{1}(w^\top x-\gamma)$.

Instead of a single hyperplane, in Liu and Potra (2009) a pair of parallel hyperplanes $H_\mathcal{A}$ and $H_\mathcal{B}$ is constructed, satisfying the following properties:

- **(P1)** all points of class $\mathcal{A}$ lie on one halfspace of $H_\mathcal{A}$;
- **(P2)** all points of class $\mathcal{B}$ lie on the opposite halfspace of $H_\mathcal{B}$;
- **(P3)** the intersection of the convex hulls of $\mathcal{A}$ and $\mathcal{B}$ is contained in the region between $H_\mathcal{A}$ and $H_\mathcal{B}$.

The starting point of the formulation consists in solving the SM-SVM model (1) with $q=1$, determining an initial separating hyperplane $H_0:=(w,\gamma)$ and the soft margin vector ξ. Then, H_0 is shifted in order to identify $H_\mathcal{A}:=(w,\gamma-1+\omega_A)$ and $H_\mathcal{B}:=(w,\gamma+1-\omega_B)$, where:

$$\omega_\mathcal{A}:=\max_{i:x^{(i)}\in\mathcal{A}}\{\xi_i\},\quad \omega_\mathcal{B}:=\max_{i:x^{(i)}\in\mathcal{B}}\{\xi_i\}.\tag{2}$$

The choice of $\omega_\mathcal{A}$ and $\omega_\mathcal{B}$ according to condition (2) guarantees that $H_\mathcal{A}$ and $H_\mathcal{B}$ satisfy properties **(P1)–(P3)**.

Table 1
A selected SVM literature review. In the first row of the table the methodological contributions are listed in chronological order. Second and third rows specify the type of SVM classifier (linear or nonlinear). Finally, the optimization under uncertainty methodologies employed in the articles are explored in rows four to ten.

		Vapnik and Chervonenkis (1974)	Boser et al. (1992)	Vapnik (1995)	Mangasarian (1998)	Lee et al. (2000)	Schölkopf et al. (2000)	Fung et al. (2002)	Lanckriet et al. (2002)	El Ghaoui et al. (2003)	Bhattacharyya (2004)	Bi and Zhang (2005)	Trafalis and Gilbert (2006)	Jayadeva et al. (2007)	Liu and Potra (2009)	Xu et al. (2009)	Bhadra et al. (2010)	Trafalis and Alwazzi (2010)	Peng (2011)	Ben-Tal et al. (2012)	Ju and jie Tian (2012)	Peng and Xu (2013)	Qi et al. (2013)	Fan et al. (2014)	López et al. (2017)	López et al. (2018)	Wang, Fan et al. (2018)	Bertsimas et al. (2019)	Blanco et al. (2020)	Maldonado et al. (2020)	Gao et al. (2021)	Jiménez-Cordero et al. (2021)	Faccini et al. (2022)	Maldonado et al. (2022)	De Leone et al. (2023)	Khanjani-Shiraz et al. (2023)	Lin et al. (2024a)	Lin et al. (2024b)
SVM	Linear classifier	✓		✓			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			✓	✓	✓	✓		
	Nonlinear classifier		✓	✓	✓	✓	✓	✓	✓			✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			✓	✓	✓	✓		✓	✓		✓	✓
Type of robust methodology	Box RO									✓																							✓					
	Ellipsoidal RO								✓	✓	✓													✓									✓					
	Polyhedral RO							✓													✓			✓													
	Bounded-by-norm RO											✓	✓			✓		✓		✓								✓							✓			
	Matrix RO																✓			✓		✓																
	Chance-Constrained								✓								✓								✓	✓	✓			✓			✓			✓	✓	✓
	Distributionally RO								✓																		✓				✓				✓	✓	✓

Finally, the optimal separating hyperplane $H := (w, b)$ is such that is parallel to $H_{\mathcal{A}}$ and $H_{\mathcal{B}}$, lies in their strip, and the number of misclassified points is minimized. These conditions are satisfied finding the optimal parameter b as solution of the following problem:

$$\begin{aligned} \min_{b} \quad & \sum_{i:x^{(i)}\in\mathcal{A}} \mathbb{1}\left(w^\top x^{(i)} - b\right) + \sum_{i:x^{(i)}\in\mathcal{B}} \mathbb{1}\left(b - w^\top x^{(i)}\right) \\ \text{s.t.} \quad & \gamma + 1 - \omega_{\mathcal{B}} \leqslant b \leqslant \gamma - 1 + \omega_{\mathcal{A}}. \end{aligned} \tag{3}$$

From a computational perspective, model (3) is solved through a linear search procedure. Specifically, the interval $[\gamma+1-\omega_{\mathcal{B}}, \gamma-1+\omega_{\mathcal{A}}]$ is divided into $N_{\max}$ sub-intervals of equal length and the problem is solved on each of them. The optimal solution b is the one providing the overall minimum value of the objective function.

Similarly to SM-SVM, a new data point $x \in \mathbb{R}^n$ is classified in class $\mathcal{A}$ or $\mathcal{B}$ depending on the decision rule $\mathbb{1}\left(w^\top x - b\right)$.

Whenever training observations are not linearly separable, the so-called *kernel trick* can be applied (Cortes & Vapnik, 1995). The key idea is to introduce a function $\phi(\cdot)$, usually referred to as *feature map*, to translate data from the *input space* $\mathbb{R}^n$ to a higher-dimensional space $\mathcal{H}$, equipped with the dot product $\langle\cdot,\cdot\rangle$. In $\mathcal{H}$, the transformed data $\{\phi(x^{(i)})\}_{i=1}^m$ are assumed to be linearly separable. Thus, model (1) can be written in the feature space $\mathcal{H}$ as:

$$\begin{aligned} \min_{\overline{w},\gamma,\xi} \quad & \|\overline{w}\|_{\mathcal{H}} + \nu \sum_{i=1}^m \xi_i \\ \text{s.t.} \quad & y^{(i)}(\langle \overline{w}, \phi(x^{(i)})\rangle - \gamma) \geqslant 1 - \xi_i \qquad i = 1, \dots, m \\ & \xi_i \geqslant 0 \qquad\qquad\qquad\qquad i = 1, \dots, m. \end{aligned} \tag{4}$$

Vector $\overline{w} \in \mathcal{H}$ defines the linear classifier in the feature space and the norm $\|\cdot\|_{\mathcal{H}}$ is induced by the inner product $\langle\cdot,\cdot\rangle$.

Unfortunately, the expression of the mapping $\phi(\cdot)$ is usually unknown and, consequently, model (4) cannot be solved in practice. To overcome this limitation, a symmetric and positive semidefinite kernel $k : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ is introduced. Examples of kernel functions typically used in ML literature are reported in Table 2. For a comprehensive overview, the reader is referred to Schölkopf and Smola (2001).

As in Cortes and Vapnik (1995), $\overline{w}$ can be decomposed into a finite linear combination of $\{\phi(x^{(j)})\}_{j=1}^m$, as $\overline{w} = \sum_{j=1}^m y^{(j)} u_j \phi(x^{(j)})$, for some coefficients $u_j \in \mathbb{R}$. Consequently, for all $i = 1, \dots, m$ the dot product $\langle \overline{w}, \phi(x^{(i)})\rangle$ in the first set of constraints of model (4) can be formulated as $\langle \overline{w}, \phi(x^{(i)})\rangle = \sum_{j=1}^m K_{ij} y^{(j)} u_j$, where $K_{ij} := k(x^{(i)}, x^{(j)}) = \langle \phi(x^{(i)}), \phi(x^{(j)})\rangle$. The properties of the kernel function imply that the Gram matrix $K = [K_{ij}]$ is a real, symmetric and positive semidefinite $m \times m$ matrix (Piccialli & Sciandrone, 2018).

As in Lee et al. (2000) and Mangasarian (1998), in the objective function of model (4) the $\mathcal{H}$-norm $\|\overline{w}\|_{\mathcal{H}}$ is replaced by $\|u\|_q^q$, where $u := [u_1, \dots, u_m]^\top$. This choice guarantees the convexity of the optimization problem. Therefore, model (4) can be rewritten as:

$$\begin{aligned} \min_{u,\gamma,\xi} \quad & \|u\|_q^q + \nu \sum_{i=1}^m \xi_i \\ \text{s.t.} \quad & y^{(i)}\left(\sum_{j=1}^m K_{ij} y^{(j)} u_j - \gamma\right) \geqslant 1 - \xi_i \qquad i = 1, \dots, m \\ & \xi_i \geqslant 0 \qquad\qquad\qquad\qquad i = 1, \dots, m. \end{aligned} \tag{5}$$

Within this context, the separating hyperplane in the feature space translates into a nonlinear decision boundary $S := (u, \gamma)$ in the input space, defined by the following equation:

$$\sum_{i=1}^m k(x, x^{(i)}) y^{(i)} u_i = \gamma. \tag{6}$$

Finally, each new observation $x \in \mathbb{R}^n$ is classified either in class $\mathcal{A}$ or $\mathcal{B}$ according to the decision function $\mathbb{1}\left(\sum_{i=1}^m k(x, x^{(i)}) y^{(i)} u_i - \gamma\right)$.

4. A novel approach for deterministic nonlinear SVM

In this section, we propose an extension of the SVM approach presented in Liu and Potra (2009) to the nonlinear case. Specifically, we classify input observations by means of kernel-induced decision boundaries, such that the corresponding hyperplanes in the feature space satisfy properties **(P1)–(P3)**.

In Section 4.1 we tackle binary classification tasks, whereas in Section 4.2 we extend the approach to the case of multiclass classification.

4.1. Binary classification

According to our proposal, binary classification problems are handled as follows. First of all, we start solving model (5), finding an initial decision boundary $S_0 := (u, \gamma)$. In the input space, hypersurface S_0 induces an initial nonlinear separation of training data points. Accordingly, in the feature space the corresponding hyperplane H_0 performs a linear classification of transformed observations.

Then, for each of the two classes, we compute the greatest misclassification error through the following extended version of formulas (2):

$$\omega_{\mathcal{A}} := \max_{i=1,\dots,m} (D\xi)_i \qquad \omega_{\mathcal{B}} := \max_{i=1,\dots,m} (-D\xi)_i, \tag{7}$$

Table 2
Examples of kernel functions. The first column reports the name of the functions. The second column provides their mathematical expressions. Finally, the third column contains the related relevant parameters.

Kernel function	$k(x, x')$	Parameter
Homogeneous polynomial	$k(x,x') = \langle x, x'\rangle^d$	$d \in \mathbb{N}$
Inhomogeneous polynomial	$k(x,x') = (c + \langle x, x'\rangle)^d$	$c \in \mathbb{R}^+,\ d \in \mathbb{N}$
Gaussian Radial Basis Function (RBF)	$k(x,x') = \exp\left(-\frac{\|x-x'\|_2^2}{2\alpha^2}\right)$	$\alpha \in \mathbb{R}_0^+$

where D is a diagonal matrix with entries $D_{ii} := y^{(i)}$, for all $i = 1, \dots, m$.

Due to the structure of problem (5), the modulus of $-1 + \omega_{\mathcal{A}}$ represents the deviation of the farthest misclassified point of class $\mathcal{A}$ from H_0 and similarly for $1 - \omega_{\mathcal{B}}$. Nevertheless, it may happen that H_0 already correctly classifies all the data points of one or both classes. In that case, the moduli are just the deviations of the closest data points from hyperplane H_0. According to the classic literature of SVM (see Cortes & Vapnik, 1995), we call *support vectors* of class $\mathcal{A}$ and $\mathcal{B}$ the transformed points that deviate $|-1 + \omega_{\mathcal{A}}|$ and $|1 - \omega_{\mathcal{B}}|$ from H_0, respectively.

At this stage, similarly to Liu and Potra (2009), we shift hyperplane H_0 by $-1+\omega_{\mathcal{A}}$ and $1-\omega_{\mathcal{B}}$, obtaining $H_{\mathcal{A}}$ and $H_{\mathcal{B}}$, respectively. Such a pair of parallel hyperplanes passes through the support vectors of the corresponding class and satisfies properties **(P1)–(P3)** in the feature space. According to Eq. (6), the corresponding hypersurfaces $S_{\mathcal{A}} := (u, \gamma - 1 + \omega_{\mathcal{A}})$ and $S_{\mathcal{B}} := (u, \gamma + 1 - \omega_{\mathcal{B}})$ are then derived in the input space.

Finally, the optimal kernel-induced decision boundary $S := (u, b)$ is deduced, where b is the solution of the following nonlinear version of model (3):

$$\begin{aligned} \min_{b} \quad & \sum_{i=1}^{m} \mathbb{1}\left(y^{(i)} b - y^{(i)} \sum_{j=1}^{m} K_{ij} y^{(j)} u_j\right) \\ \text{s.t.} \quad & \gamma + 1 - \omega_{\mathcal{B}} \leqslant b \leqslant \gamma - 1 + \omega_{\mathcal{A}}. \end{aligned} \tag{8}$$

We observe that hypersurface S in the input space is induced by hyperplane H in the feature space, which is parallel to $H_{\mathcal{A}}$ and $H_{\mathcal{B}}$ and lies in the region between them. Therefore, a new observation $x \in \mathbb{R}^n$ is classified according to the decision function $\mathbb{1}\left(\sum_{i=1}^{m} k(x, x^{(i)}) y^{(i)} u_i - b\right)$.

For the sake of clarity, all the steps of the approach discussed so far are schematically reported in Pseudocode 1.

Pseudocode 1 A novel approach for nonlinear SVM.

Input: $\{(x^{(i)}, y^{(i)})\}_{i=1}^m$, $q \in [1, \infty]$, $\nu \geqslant 0$, $k(\cdot,\cdot) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$.

1: Calculate matrix K as $K_{ij} = k\left(x^{(i)}, x^{(j)}\right), i, j = 1, \dots, m$.
2: Solve model (5).
3: Find the initial separating hypersurface $S_0 = (u, \gamma)$, defined by equation Eq. (6).
4: Construct diagonal matrix D as $D_{ii} = y^{(i)}, i = 1, \dots, m$, and compute $\omega_{\mathcal{A}}$ and $\omega_{\mathcal{B}}$ according to formulas (7).
5: Shift S_0 to get the separating hypersurface for each class, $S_{\mathcal{A}} = (u, \gamma - 1 + \omega_{\mathcal{A}})$ and $S_{\mathcal{B}} = (u, \gamma + 1 - \omega_{\mathcal{B}})$, defined by Eq. (6).
6: Solve model (8), obtaining parameter b.

Output: The optimal decision boundary $S = (u, b)$, defined by Eq. (6).

The computational complexity of nonlinear SVM models is between $O(m^2)$ and $O(m^3)$ (Peng, 2011). Since model (8) requires at most $N_{\max}$ iterations to be solved through a linear search procedure, the computational complexity of our approach is between $O(\max\{m^2, N_{\max}\})$ and $O(\max\{m^3, N_{\max}\})$.

By way of illustration, in Fig. 1 we depict the separating surfaces obtained by applying the proposed SVM methodology to a bidimensional toy example. In model (5) we set $q = 1$, $\nu = 1$ and consider linear and Gaussian RBF kernels. The graphical interpretation of the novel approach is illustrated in Fig. 2.

4.2. Multiclass classification

In this section, we derive the multiclass extension of the approach presented so far. We focus our attention on one of the most commonly used multiclass SVM framework, the *one-versus-all* (Vapnik, 1995; Weston & Watkins, 1998). According to this methodology, L binary classifiers are constructed, where L is the number of classifying categories, such that each class is independently separated by all the others grouped together. Formally, let $\{(x^{(i)}, y^{(i)})\}_{i=1}^m$ be the set of training observations, with $x^{(i)} \in \mathbb{R}^n$ and $y^{(i)} \in \{1, \dots, L\}$. For each class $l = 1, \dots, L$, we find an initial separating hypersurface $S_{l,0} := (u_l, \gamma_l)$, where $u_l \in \mathbb{R}^n$ and $\gamma_l \in \mathbb{R}$ are the solutions of the following multiclass version of model (5):

$$\begin{aligned} \min_{u_l, \gamma_l, \xi_l} \quad & \|u_l\|_q^q + \nu \sum_{i=1}^{m} \xi_{l,i} \\ \text{s.t.} \quad & \hat{y}_l^{(i)} \left(\sum_{j=1}^{m} K_{ij} \hat{y}_l^{(j)} u_{l,j} - \gamma_l\right) \geqslant 1 - \xi_{l,i} \qquad i = 1, \dots, m \\ & \xi_{l,i} \geqslant 0 \qquad\qquad\qquad\qquad\quad i = 1, \dots, m, \end{aligned} \tag{9}$$

with $\hat{y}_l^{(i)} = 1$ if $y^{(i)} = l$, and $\hat{y}_l^{(i)} = -1$ otherwise. Then, we construct the diagonal matrix $\hat{D}_l$, with $\hat{D}_{l,ii} := \hat{y}_l^{(i)}$, $i = 1, \dots, m$, and compute:

$$\omega_l := \max_{i=1,\dots,m} (\hat{D}_l \xi_l)_i \qquad \omega_{-l} := \max_{i=1,\dots,m} (-\hat{D}_l \xi_l)_i.$$

Hypersurface $S_{l,0}$ is then shifted to get $S_l := (u_l, \gamma_l - 1 + \omega_l)$ and $S_{-l} := (u_l, \gamma_l + 1 - \omega_{-l})$ in the input space. The corresponding hyperplanes in the feature space satisfy properties **(P1)–(P3)**. Finally, the optimal decision boundary for class l versus all the others is $S_{l,-l} := (u_l, b_l)$, with b_l solution of the following model:

$$\begin{aligned} \min_{b_l} \quad & \sum_{i=1}^{m} \mathbb{1}\left(\hat{y}_l^{(i)} b_l - \hat{y}_l^{(i)} \sum_{j=1}^{m} K_{ij} \hat{y}_l^{(j)} u_{l,j}\right) \\ \text{s.t.} \quad & \gamma_l + 1 - \omega_{-l} \leqslant b_l \leqslant \gamma_l - 1 + \omega_l. \end{aligned} \tag{10}$$

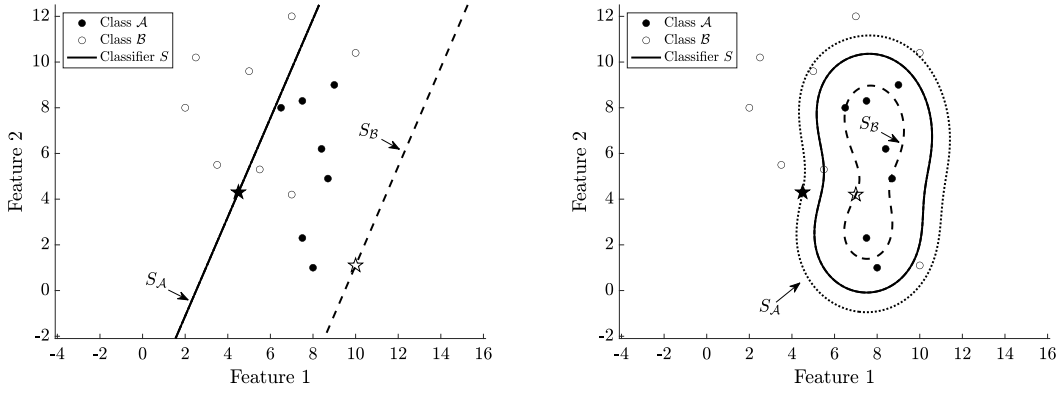
The decision function of the lth class is given by $f_l(x) := \sum_{i=1}^m k(x, x^{(i)}) \hat{y}_l^{(i)} u_{l,i} - b_l$, and each new observation $x \in \mathbb{R}^n$ is assigned to the class $l^* := \arg\max_{l=1,\dots,L} f_l(x)$ (López et al., 2017).

Since the *one-versus-all* strategy generates L binary classifiers, one for each class, the computational complexity of our multiclass approach is between $O(L \cdot \max\{m^2, N_{\max}\})$ and $O(L \cdot \max\{m^3, N_{\max}\})$.

We represent in Fig. 3 the results of the proposed methodology in the case of a multiclass classification task. The parameters q and ν are the same as in Fig. 1. Similarly to the binary case (see Fig. 1(a)), it may happen that either S_l or S_{-l} coincides with $S_{l,-l}$. This is due to the fact that in model (10) the optimal parameter b_l may be equal to $\gamma_l - 1 + \omega_l$ or $\gamma_l + 1 - \omega_{-l}$, respectively.

5. A robust model for nonlinear SVM

In this section, we derive the robust counterpart of the deterministic approach discussed so far, when input data are plagued by uncertainties. According to the RO framework, we construct an uncertainty set around each observation and optimize against the worst-case realization across the entire uncertainty set (Bertsimas et al., 2019).



(a) Linear kernel (b) Gaussian RBF kernel ($\alpha = 1.9$)

Fig. 1. Separating surfaces obtained with linear and Gaussian RBF kernel functions. Support vectors are depicted as stars.

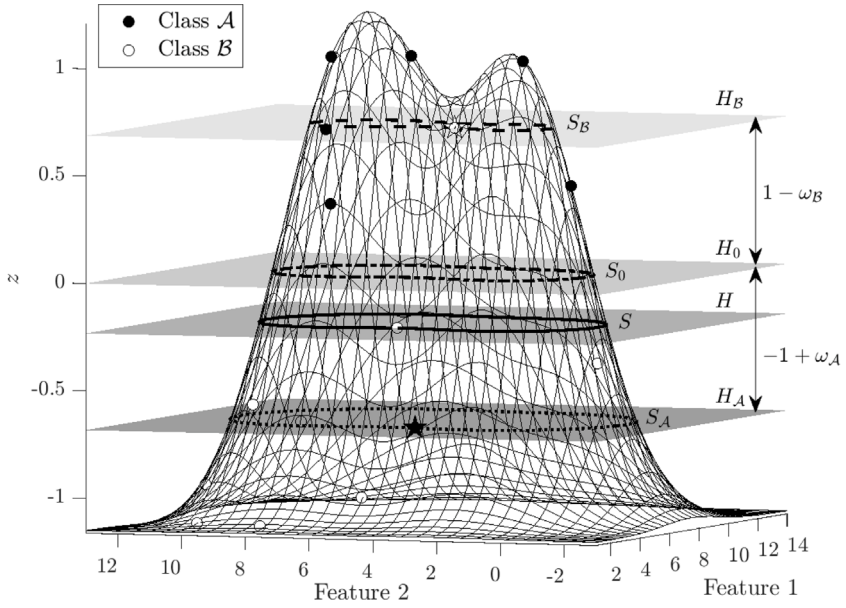
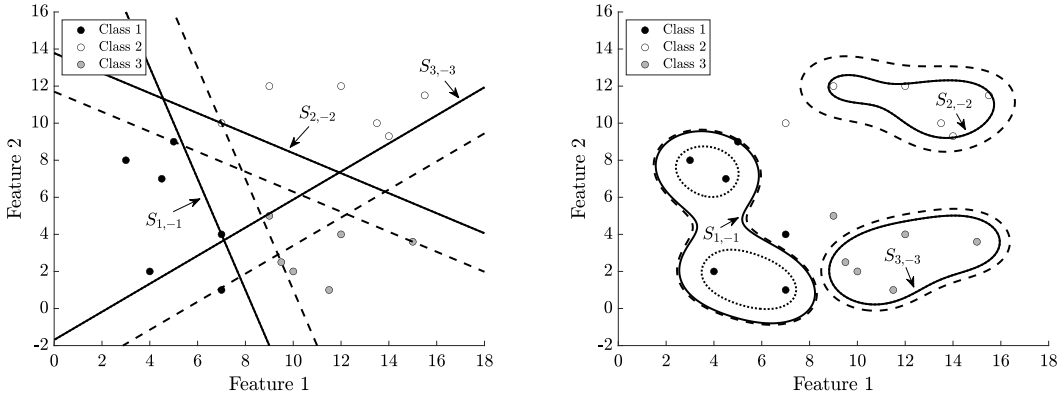


Fig. 2. Graphical representation of the implicit function defined by Eq. (6), in the case of Gaussian RBF kernel ($\alpha = 1.9$), along with the separating hyperplanes and decision boundaries. Support vectors are drawn as stars.



(a) Linear kernel (b) Gaussian RBF kernel ($\alpha = 1.9$)

Fig. 3. Separating surfaces obtained with linear and Gaussian RBF kernel functions in the case of a three-classes classification task. For each class $l = 1, 2, 3$, the dotted line and the dashed line represent respectively S_l and S_{-l}.

Contrariwise to RO models dealing with linear classification (see, for instance, Faccini et al., 2022), in the nonlinear context data points $x^{(i)}$ are mapped into the feature space $\mathcal{H}$ via $\phi(\cdot)$ and uncertainty sets $\mathcal{U}_{\mathcal{H}}\left(\phi(x^{(i)})\right)$ have to be constructed. Unfortunately, a closed-form expression of $\phi(\cdot)$ is rarely available and an *a priori* control about $\mathcal{U}_{\mathcal{H}}\left(\phi(x^{(i)})\right)$ is not possible. Therefore, further assumptions on the uncertainty set $\mathcal{U}_{\mathcal{H}}\left(\phi(x^{(i)})\right)$ in the feature space are necessary.

The remainder of the section is organized as follows. In Section 5.1 bounded-by-ℓ_p-norm uncertainty sets $\mathcal{U}_p(x^{(i)})$ are constructed, together with the corresponding ones $\mathcal{U}_{\mathcal{H}}(\phi(x^{(i)}))$ in the feature space. Bounds on the radii of $\mathcal{U}_{\mathcal{H}}(\phi(x^{(i)}))$ are derived in Section 5.2. Finally, in Section 5.3 the robust counterpart of models (5) and (9) is rigorously deduced, together with computationally tractable reformulations.

5.1. The construction of the uncertainty sets

We assume that each observation $x^{(i)}$ in the input space is subject to an additive and unknown perturbation vector $\sigma^{(i)}$, whose ℓ_p-norm, with $p \in [1, \infty]$, is bounded by a nonnegative constant $\eta^{(i)}$. Consequently, the uncertainty set around $x^{(i)}$ has the following expression:

$$\mathcal{U}_p(x^{(i)}) := \left\{ x \in \mathbb{R}^n : x = x^{(i)} + \sigma^{(i)}, \|\sigma^{(i)}\|_p \leqslant \eta^{(i)} \right\}. \tag{11}$$

Parameter $\eta^{(i)}$ calibrates the degree of conservatism: if $\eta^{(i)} = 0$, then $\sigma^{(i)}$ is the zero vector of $\mathbb{R}^n$ and $\mathcal{U}_p(x^{(i)})$ coincides with $x^{(i)}$. Popular choices for the ℓ_p-norm in the RO literature are $p = 1, 2, \infty$, leading to polyhedral, spherical and box uncertainty sets, respectively.

In order to consider the extension towards the feature space, we now assume that, if x belongs to $\mathcal{U}_p(x^{(i)})$, then:

$$\phi(x) = \phi(x^{(i)} + \sigma^{(i)}) = \phi(x^{(i)}) + \zeta^{(i)},$$

where the perturbation $\zeta^{(i)}$ belongs to the feature space $\mathcal{H}$ and its $\mathcal{H}$-norm is bounded a nonnegative constant $\delta^{(i)}$. The latter may be unknown but it depends on the known bound $\eta^{(i)}$ in the input space, i.e. $\delta^{(i)} = \delta^{(i)}(\eta^{(i)})$. If no uncertainty occurs in the input space, no uncertainty will occur in the feature space too: $\eta^{(i)} = 0$ implies $\delta^{(i)} = 0$. Hence, the uncertainty set around $\phi(x^{(i)})$ in the feature space is modeled as:

$$\mathcal{U}_{\mathcal{H}}(\phi(x^{(i)})) := \left\{ z \in \mathcal{H} : z = \phi(x^{(i)}) + \zeta^{(i)}, \|\zeta^{(i)}\|_{\mathcal{H}} \leqslant \delta^{(i)} \right\}. \tag{12}$$

5.2. Bounds on the uncertainty sets in the feature space

Let $k(\cdot,\cdot)$ be a symmetric and positive semidefinite kernel, with corresponding feature map $\phi(\cdot)$. In the following, we derive closed-form expressions for the radius $\delta^{(i)}$ in the feature space given the bound $\eta^{(i)}$ in the input space, when $k(\cdot,\cdot)$ is the polynomial kernel or the Gaussian RBF kernel. Below, we provide the results and relegate the proofs to Appendix A.

Proposition 1 (*Polynomial Kernel*). *Let $\mathcal{U}_p(x^{(i)})$ and $\mathcal{U}_{\mathcal{H}}(\phi(x^{(i)}))$ be the uncertainty sets in the input and in the feature space as in (11) and (12), respectively, with $p \in [1, \infty]$. Consider the inhomogeneous polynomial kernel of degree $d \in \mathbb{N}$ and additive constant $c \geqslant 0$, with radius $\delta^{(i)} \equiv \delta^{(i)}_{d,c}$, and:*

$$C = C(n,p) = \begin{cases} 1, & 1 \leqslant p \leqslant 2 \\ n^{\frac{p-2}{2p}}, & p > 2. \end{cases}$$

(i) If $d = 1$, then the radius of $\mathcal{U}_{\mathcal{H}}(\phi(x^{(i)}))$ is:

$$\delta^{(i)}_{1,c} = C\eta^{(i)}. \tag{13}$$

(ii) If $d > 1$, then:

$$\delta^{(i)}_{d,c} = \sqrt{\left(\delta^{(i)}_{d,0}\right)^2 + \sum_{k=1}^{d-1} \binom{d}{k} c^k \left[\sum_{j=1}^{d-k} \binom{d-k}{j} \|x^{(i)}\|_2^{d-k-j} \left(C\eta^{(i)}\right)^j\right]^2}, \tag{14}$$

where $\delta^{(i)}_{d,0}$ is the bound for the corresponding homogeneous polynomial kernel:

$$\delta^{(i)}_{d,0} = \sum_{k=1}^{d} \binom{d}{k} \left\|x^{(i)}\right\|_2^{d-k} \left(C\eta^{(i)}\right)^k. \tag{15}$$

Notice that when $c = 0$, Eq. (14) reduces to Eq. (15).

Proposition 2 (*Gaussian RBF Kernel*). *Let $\mathcal{U}_p(x^{(i)})$ and $\mathcal{U}_{\mathcal{H}}(\phi(x^{(i)}))$ be the uncertainty sets in the input and in the feature space as in (11) and (12), respectively, with $p \in [1, \infty]$. Consider the Gaussian RBF kernel with parameter $\alpha > 0$ and radius $\delta^{(i)} \equiv \delta^{(i)}_{\alpha}$. If:*

$$C = C(n,p) = \begin{cases} 1, & 1 \leqslant p \leqslant 2 \\ n^{\frac{p-2}{2p}}, & p > 2, \end{cases}$$

then:

$$\delta^{(i)}_{\alpha} = \sqrt{2 - 2\exp\left(-\frac{\left(C\eta^{(i)}\right)^2}{2\alpha^2}\right)}. \tag{16}$$

We observe that Propositions 1–2 are consistent with Lemma 7 presented in Xu et al. (2009). However, in this paper we specify the bounds for particular choices of the kernel functions. In addition, we extend the result for a bounded-by-ℓ_p-norm uncertainty set for a generic $p \in [1, \infty]$.

5.3. The robust model

Robustifying model (5) against the uncertainty set $\mathcal{U}_p(x^{(i)})$ yields the following optimization program:

$$\begin{aligned} \min_{u,\gamma,\xi} \quad & \|u\|_q^q + \nu \sum_{i=1}^{m} \xi_i \\ \text{s.t.} \quad & y^{(i)} \sum_{j=1}^{m} k(x, x^{(j)}) y^{(j)} u_j \geqslant 1 - \xi_i + y^{(i)}\gamma \quad \forall x \in \mathcal{U}_p(x^{(i)}),\ i = 1, \dots, m \\ & \xi_i \geqslant 0 \qquad\qquad i = 1, \dots, m. \end{aligned} \tag{17}$$

Model (17) cannot be solved in practice due to the infinite possibilities for choosing x in $\mathcal{U}_p(x^{(i)})$. Nevertheless, it can be reformulated in a tractable form, as stated in the following theorem.

Theorem 1. *Let $\mathcal{U}_p(x^{(i)})$ and $\mathcal{U}_{\mathcal{H}}(\phi(x^{(i)}))$ be the uncertainty sets in the input and in the feature space as in (11) and (12), respectively, with $p \in [1, \infty]$. Model (17) is equivalent to:*

$$\begin{aligned} \min_{u,\gamma,\xi} \quad & \|u\|_q^q + \nu \sum_{i=1}^{m} \xi_i \\ \text{s.t.} \quad & y^{(i)} \sum_{j=1}^{m} K_{ij} y^{(j)} u_j - \delta^{(i)} \sum_{j=1}^{m} \sqrt{K_{jj}} \left|u_j\right| \geqslant 1 - \xi_i + y^{(i)}\gamma \qquad i = 1, \dots, m \\ & \xi_i \geqslant 0 \qquad\qquad i = 1, \dots, m. \end{aligned} \tag{18}$$

Proof. The first set of constraints of model (17) is equivalent to:

$$\min_{x \in \mathcal{U}_p(x^{(i)})} \quad y^{(i)} \sum_{j=1}^{m} k(x, x^{(j)}) y^{(j)} u_j \geqslant 1 - \xi_i + y^{(i)}\gamma \qquad i = 1, \dots, m. \tag{19}$$

Due to the definition of $\mathcal{U}_p(x^{(i)})$, for all $i = 1, \dots, m$ the left-hand side of (19) can be re-stated as follows:

$$\begin{aligned} \min_{\sigma^{(i)}} \quad & y^{(i)} \sum_{j=1}^{m} k(x^{(i)} + \sigma^{(i)}, x^{(j)}) y^{(j)} u_j \\ \text{s.t.} \quad & \|\sigma^{(i)}\|_p \leqslant \eta^{(i)}. \end{aligned} \tag{20}$$

According to the definition of the kernel function and the assumption on $\mathcal{U}_{\mathcal{H}}(\phi(x^{(i)}))$, we have that:

$$k(x^{(i)} + \sigma^{(i)}, x^{(j)}) = \langle \phi(x^{(i)} + \sigma^{(i)}), \phi(x^{(j)}) \rangle = \langle \phi(x^{(i)}) + \zeta^{(i)}, \phi(x^{(j)}) \rangle.$$

Moreover, the linearity of the dot product in the feature space $\mathcal{H}$ implies that model (20) can be written as follows:

$$\begin{aligned} \min_{\zeta^{(i)}} \quad & y^{(i)} \sum_{j=1}^{m} \langle \zeta^{(i)}, \phi(x^{(j)}) \rangle \; y^{(j)} u_j \\ \text{s.t.} \quad & \|\zeta^{(i)}\|_{\mathcal{H}} \leqslant \delta^{(i)}, \end{aligned} \tag{21}$$

where the term $y^{(i)} \sum_{j=1}^{m} \langle \phi(x^{(i)}), \phi(x^{(j)}) \rangle \; y^{(j)} u_j$ is equivalent to $y^{(i)} \sum_{j=1}^{m} K_{ij} y^{(j)} u_j$. Being independent of $\zeta^{(i)}$, it is moved to the right-hand side of (19).

Then, the modulus of the objective function of model (21) can be bounded by $\sum_{j=1}^{m} \left| \langle \zeta^{(i)}, \phi(x^{(j)}) \rangle \right| \cdot \left| u_j \right|$. By applying the Cauchy–Schwarz inequality in $\mathcal{H}$ and the boundedness condition on $\left\| \zeta^{(i)} \right\|_{\mathcal{H}}$, we get:

$$\left| \langle \zeta^{(i)}, \phi(x^{(j)}) \rangle \right| \leqslant \left\| \zeta^{(i)} \right\|_{\mathcal{H}} \cdot \left\| \phi(x^{(j)}) \right\|_{\mathcal{H}} \leqslant \delta^{(i)} \cdot \sqrt{\langle \phi(x^{(j)}), \phi(x^{(j)}) \rangle} = \delta^{(i)} \cdot \sqrt{K_{jj}}.$$

The value K_{jj} is nonnegative, due to the positive semidefiniteness of the Gram matrix K. Therefore, we obtain:

$$\left| y^{(i)} \sum_{j=1}^{m} \langle \zeta^{(i)}, \phi(x^{(j)}) \rangle \; y^{(j)} u_j \right| \leqslant \delta^{(i)} \sum_{j=1}^{m} \sqrt{K_{jj}} \left| u_j \right|. \tag{22}$$

Thus, the optimal value of model (21) is $-\delta^{(i)} \sum_{j=1}^{m} \sqrt{K_{jj}} \left| u_j \right|$. By replacing the minimization term with this optimal value in the first set of constraints of (19), the thesis follows. □

When no uncertainty occurs in the data, $\delta^{(i)} = 0$ for all $i = 1, \dots, m$ and the robust model (18) reduces to the deterministic formulation (5).

Model (18) is a convex nonlinear optimization model due to the presence of the ℓ_q-norm of u. Nevertheless, it can be reformulated as a *Linear Programming* (LP) problem when $q = 1$ or $q = \infty$ and as a SOCP problem when $q = 2$, as stated in the following result. The proof is provided in Appendix A.

Corollary 1. *Model (18) can be expressed as a LP problem or as a SOCP problem in the following cases:*

(a) Case $q = 1$: LP problem

$$\begin{aligned} \min_{u,\gamma,\xi,s} \quad & \sum_{i=1}^{m} s_i + \nu \sum_{i=1}^{m} \xi_i \\ \text{s.t.} \quad & y^{(i)} \sum_{j=1}^{m} K_{ij} y^{(j)} u_j - \delta^{(i)} \sum_{j=1}^{m} \sqrt{K_{jj}} s_j \geqslant 1 - \xi_i + y^{(i)} \gamma && i = 1, \dots, m \\ & s_i \geqslant -u_i && i = 1, \dots, m \\ & s_i \geqslant u_i && i = 1, \dots, m \\ & s_i, \xi_i \geqslant 0 && i = 1, \dots, m. \end{aligned} \tag{23}$$

(b) Case $q = 2$: SOCP problem

$$\begin{aligned} \min_{u,\gamma,\xi,s,r,t,v} \quad & r - v + \nu \sum_{i=1}^{m} \xi_i \\ \text{s.t.} \quad & y^{(i)} \sum_{j=1}^{m} K_{ij} y^{(j)} u_j - \delta^{(i)} \sum_{j=1}^{m} \sqrt{K_{jj}} s_j \geqslant 1 - \xi_i + y^{(i)} \gamma && i = 1, \dots, m \\ & t \geqslant \|u\|_2 \\ & r + v = 1 \\ & r \geqslant \sqrt{t^2 + v^2} \\ & s_i \geqslant -u_i && i = 1, \dots, m \\ & s_i \geqslant u_i && i = 1, \dots, m \\ & s_i, \xi_i \geqslant 0 && i = 1, \dots, m. \end{aligned} \tag{24}$$

(c) Case $q = \infty$: LP problem

$$\begin{aligned} \min_{u,\gamma,\xi,s,s_\infty} \quad & s_\infty + \nu \sum_{i=1}^{m} \xi_i \\ \text{s.t.} \quad & y^{(i)} \sum_{j=1}^{m} K_{ij} y^{(j)} u_j - \delta^{(i)} \sum_{j=1}^{m} \sqrt{K_{jj}} s_j \geqslant 1 - \xi_i + y^{(i)} \gamma && i = 1, \dots, m \\ & s_\infty \geqslant -u_i && i = 1, \dots, m \\ & s_\infty \geqslant u_i && i = 1, \dots, m \\ & s_i \geqslant -u_i && i = 1, \dots, m \\ & s_i \geqslant u_i && i = 1, \dots, m \\ & s_\infty \geqslant 0 \\ & s_i, \xi_i \geqslant 0 && i = 1, \dots, m. \end{aligned} \tag{25}$$

As in the deterministic setting, once u, γ and ξ are obtained as solutions of model (18), then $\omega_{\mathcal{A}}$ and $\omega_{\mathcal{B}}$ are computed according to formulas (7). Finally, the optimal separating hypersurface $S = (u, b)$ is derived, where b is the optimal solution of the following robust counterpart of problem (8):

$$\begin{aligned} \min_{b} \quad & \sum_{i=1}^{m} \mathbb{1} \left[\left(y^{(i)} b - y^{(i)} \sum_{j=1}^{m} K_{ij} y^{(j)} u_j + \delta^{(i)} \sum_{j=1}^{m} \sqrt{K_{jj}} \left| u_j \right| \right)_i \right] \\ \text{s.t.} \quad & \gamma + 1 - \omega_{\mathcal{B}} \leqslant b \leqslant \gamma - 1 + \omega_{\mathcal{A}}. \end{aligned} \tag{26}$$

When dealing with a multiclass classification task, the robust extension of model (9) for the lth class is given by:

$$\begin{aligned} \min_{u_l, \gamma_l, \xi_l} \quad & \|u_l\|_q^q + \nu \sum_{i=1}^{m} \xi_{l,i} \\ \text{s.t.} \quad & \hat{y}_l^{(i)} \sum_{j=1}^{m} K_{ij} \hat{y}_l^{(j)} u_{l,j} - \delta^{(i)} \sum_{j=1}^{m} \sqrt{K_{jj}} \left| u_{l,j} \right| \geqslant 1 - \xi_{l,i} + \hat{y}_l^{(i)} \gamma_l && i = 1, \dots, m \\ & \xi_{l,i} \geqslant 0 && i = 1, \dots, m. \end{aligned} \tag{27}$$

The optimal parameter b_l is the solution of:

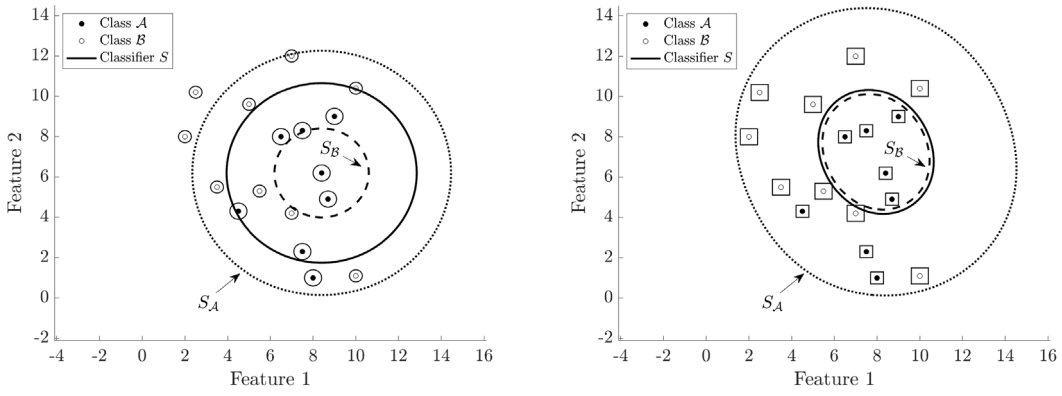
$$\begin{aligned} \min_{b_l} \quad & \sum_{i=1}^{m} \mathbb{1} \left[\left(\hat{y}_l^{(i)} b_l - \hat{y}_l^{(i)} \sum_{j=1}^{m} K_{ij} \hat{y}_l^{(j)} u_{l,j} + \delta^{(i)} \sum_{j=1}^{m} \sqrt{K_{jj}} \left| u_{l,j} \right| \right)_i \right] \\ \text{s.t.} \quad & \gamma_l + 1 - \omega_{-l} \leqslant b_l \leqslant \gamma_l - 1 + \omega_l. \end{aligned} \tag{28}$$

Since the structural form of the robust models (18) and (27) is the same as their deterministic equivalent, the time complexity analysis provides analogous results.

For the sake of illustration, we depict in Fig. 4 the kernel-induced decision boundaries of the robust model (23), considering the same dataset of Fig. 1. The model is trained for both spherical (see Fig. 4(a)) and box (see Fig. 4(b)) uncertainty sets.

6. Computational results

In this section, we evaluate the performance of the deterministic models presented in Section 4 and their robust counterparts of Section 5 on a selection of 12 benchmark datasets taken from the UCI Machine Learning Repository (Kelly, Longjohn, & Nottingham, 2023). The models were implemented in MATLAB (v. 2021b) and solved using CVX (v. 2.2, see Grant & Boyd, 2008, 2014) and MOSEK solver (v. 9.1.9, see MOSEK ApS, 2019). All computational experiments were run on a MacBookPro17.1 with a chip Apple M1 of 8 cores and 16 GB of RAM memory. The MATLAB codes developed for the current proposal are made publicly available on GitHub (https://github.com/aspinellibg/NonlinearSVM).



(a) Gaussian RBF kernel ($\alpha = 1.9$), $p = 2$

(b) Gaussian RBF kernel ($\alpha = 1.9$), $p = \infty$

Fig. 4. Separating surfaces obtained with Gaussian RBF kernel function from the robust model (23). The ℓ_p-norms defining the uncertainty sets are $p = 2$ (on the left) and $p = \infty$ (on the right).

The benchmark datasets are listed in the first column of Table 3, along with the corresponding number of observations m and of features n. In this study we examine 10 binary classification problems and 2 multiclass classification problems.

The experimental setting is as follows. Each dataset was split into *training set*, composed by the $\beta\%$ of the observations, and *testing set*, composed by the remaining $(100-\beta)\%$. We accounted for three different values of β, leading to the holdouts 75%–25%, 50%–50%, and 25%–75%. The partition was performed inline with the *proportional random sampling* strategy (Chen, Tse, & Yu, 2001), meaning that the original class balance in the entire dataset was maintained in both training and testing set. Once the partition was complete, a kernel function $k(\cdot,\cdot)$ was chosen and the training set used to train the deterministic classifier for different values of input parameter ν. Specifically, the deterministic formulation was solved on the training dataset through a grid-search strategy with five logarithmically spaced values of ν between 10^{-3} and 10^{0}, and setting $N_{\max} = 10^4$ as number of sub-intervals in the linear search procedure (see Faccini et al., 2022). The optimal classifier was chosen among the five candidates as the one minimizing the misclassification error on the training set. Finally, the out-of-sample error on the testing set was computed, as the ratio between the total number of misclassified points in the testing set and its cardinality. In order to get stable results, the partition in training and testing set was performed 96 times in a *repeated holdout* fashion (Kim, 2009). The choice of this number is motivated by the use of the Parallel Computing Toolbox in MATLAB, since the code was parallelized on the 8 cores of the working machine. The final results were then averaged.

As in the original work of Liu and Potra (2009) and in the robust linear extension presented in Faccini et al. (2022), we considered $q = 1$ in the objective function of the models. This choice provides a good compromise between structural risk minimization, related to the misclassification error, and parsimony since it automatically performs feature selection (Labbé, Martínez-Merino, & Rodríguez-Chía, 2019; Lee, Yoon, & Won, 2022; Liao, Dai, & Kuosmanen, 2024; López, Maldonado, & Carrasco, 2019).

As far as it concerns the kernel function $k(\cdot,\cdot)$, we tested seven different alternatives: homogeneous linear ($d = 1$, $c = 0$), homogeneous quadratic ($d = 2$, $c = 0$), homogeneous cubic ($d = 3$, $c = 0$); inhomogeneous linear, inhomogeneous quadratic, inhomogeneous cubic; Gaussian RBF. For simplicity, parameter α in the Gaussian RBF kernel was set as the maximum value of the standard deviation across features for the dataset under consideration. Similarly for parameter c in the inhomogeneous polynomial kernels.

Since models (5), (9) and their robust extensions (18), (27) are distance-based, imbalances in the order of magnitude of the features may result in distorted weights when classifying. For this reason, we considered *min–max normalization* and *standardization* as pre-processing techniques of data transformation (Han, Kamber, & Pei, 2011). On one hand, in the min–max normalization each dataset was linearly scaled feature-wise into the n-dimensional hypercube $[0,1]^n$. On the other hand, in the standardization the values of a specific feature j, with $j = 1,\dots,n$, were normalized based on its mean μ_j and standard deviation std_j.

Among all the optimal deterministic classifiers found for each pair *data transformation-kernel function*, the best configuration was chosen as the one minimizing the overall out-of-sample testing error. Within this choice of *data transformation-kernel function*, the robust model was solved. The bounds $\eta^{(i)}$ on the perturbation vectors defining the uncertainty sets $\mathcal{U}_p(x^{(i)})$ were adjusted as:

$$\eta^{(i)} = \eta_{\mathcal{A}} := \rho_{\mathcal{A}} \max_{j=1,\dots,n} std_{j,\mathcal{A}} \qquad \forall i : x^{(i)} \in \mathcal{A}$$

$$\eta^{(i)} = \eta_{\mathcal{B}} := \rho_{\mathcal{B}} \max_{j=1,\dots,n} std_{j,\mathcal{B}} \qquad \forall i : x^{(i)} \in \mathcal{B},$$

where $\rho_{\mathcal{A}}$ is a nonnegative parameter allowing the user to tailor the degree of conservatism and $\max_{j=1,\dots,n} std_{j,\mathcal{A}}$ is the maximum standard deviation feature-wise for training points of class $\mathcal{A}$. Similarly for $\rho_{\mathcal{B}}$ and $\max_{j=1,\dots,n} std_{j,\mathcal{B}}$. Once $\eta^{(i)}$ had been determined, the computation of the bound $\delta^{(i)}$ in the feature space was performed according to Propositions 1–2. For simplicity, we set $\rho_{\mathcal{A}} = \rho_{\mathcal{B}} = \rho$, and considered 7 logarithmically spaced values for ρ between 10^{-7} and 10^{-1}. When the number of classes is greater than two, an analogous approach was applied class-wise. As in the deterministic setting, we averaged the out-of-sample testing errors for 96 random partitions of the dataset.

For each dataset, we report in Table 3 the best configuration *data transformation-kernel function*, along with the average out-of-sample testing errors and standard deviations for the deterministic and robust models (holdout 75% training set–25% testing set). We considered three types of uncertainty set, defined respectively by ℓ_1-, ℓ_2- and ℓ_∞-norm. Detailed results are reported in Tables B.7–B.21 in Appendix B.

We notice that all the considered robust formulations outperform the corresponding deterministic models. In 6 out of 12 datasets the best results are achieved by the box robust formulation ($p = \infty$). Since box uncertainty sets are the widest around data points, this implies that the proposed formulations benefit from a more conservative approach when treating uncertainties. The last column of Table 3 shows the robust *Improvement Ratio* (IR) over the deterministic counterpart. The IR was computed as in Faccini et al. (2022) and according to the following formula:

$$\text{IR} := \frac{\tau^{\text{det}} - \tau^{\text{rob}^*}}{\tau^{\text{det}}},$$

Table 3
Average out-of-sample testing errors and standard deviations over 96 runs for the deterministic and robust models. Best results are highlighted. The last column displays the robust improvement ratios over the deterministic counterparts. Holdout: 75% training set-25% testing set.

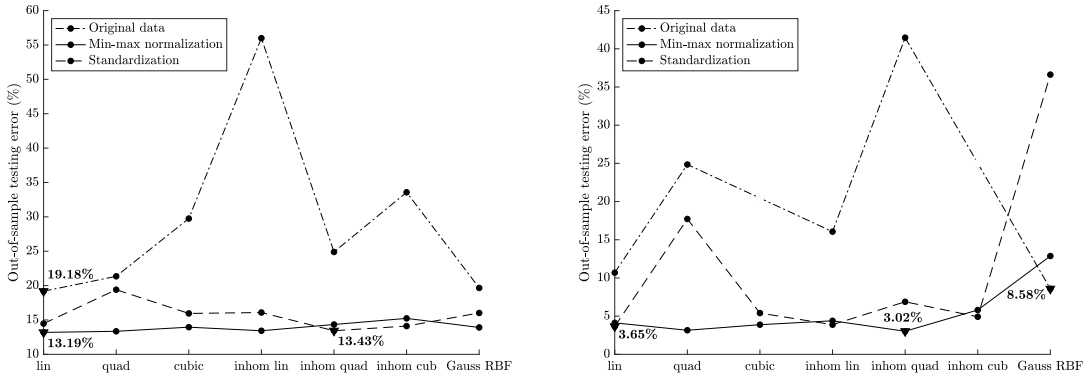
Dataset	Data transformation	Kernel	Deterministic	Robust			Robust
$m \times n$				$p = 1$	$p = 2$	$p = \infty$	improvement ratio
Arrhythmia 68 × 279	–	Gaussian RBF	20.47% ± 0.07	**19.12% ± 0.08**	19.30% ± 0.07	19.61% ± 0.07	6.60%
CPU time (s)			0.289	0.290	0.288	0.295	
Parkinson 195 × 22	Min–max normalization	Hom. linear	13.19% ± 0.03	12.98% ± 0.03	**12.37% ± 0.03**	12.61% ± 0.04	6.22%
CPU time (s)			3.626	3.421	3.454	3.418	
Heart Disease 297 × 13	Standardization	Inhom. linear	17.48% ± 0.04	16.84% ± 0.04	17.53% ± 0.03	**16.36% ± 0.04**	6.41%
CPU time (s)			12.253	11.602	11.477	11.417	
Dermatology 358 × 34	–	Inhom. quadratic	1.64% ± 0.02	1.65% ± 0.01	1.57% ± 0.01	**0.55% ± 0.01**	66.46%
CPU time (s)			20.173	20.055	20.420	20.147	
Climate Model Crashes 540 × 18	–	Hom. linear	5.01% ± 0.02	4.47% ± 0.02	4.50% ± 0.01	**4.34% ± 0.01**	13.37%
CPU time (s)			68.069	66.762	67.169	67.381	
Breast Cancer Diagnostic 569 × 30	Min–max normalization	Inhom. quadratic	3.02% ± 0.02	2.63% ± 0.01	2.65% ± 0.01	**2.39% ± 0.01**	20.86%
CPU time (s)			77.786	77.968	78.267	77.543	
Breast Cancer 683 × 9	Standardization	Hom. linear	3.17% ± 0.01	**2.97% ± 0.01**	3.07% ± 0.01	3.06% ± 0.01	6.31%
CPU time (s)			135.765	135.651	137.039	136.286	
Blood Transfusion 748 × 4	Standardization	Inhom. cubic	20.72% ± 0.02	20.60% ± 0.02	**20.55% ± 0.02**	20.64% ± 0.02	0.82%
CPU time (s)			178.136	178.751	179.682	180.083	
Mammographic Mass 830 × 5	Standardization	Inhom. quadratic	15.71% ± 0.02	15.49% ± 0.02	**15.42% ± 0.02**	15.54% ± 0.02	1.85%
CPU time (s)			241.205	241.810	242.614	241.929	
Qsar Biodegradation 1055 × 41	Min–max normalization	Gaussian RBF	12.88% ± 0.02	**11.78% ± 0.01**	12.72% ± 0.02	12.86% ± 0.02	8.54%
CPU time (s)			484.908	498.235	495.073	491.748	
Iris 150 × 4 (3 classes)	–	Gaussian RBF	3.10% ± 0.03	3.07% ± 0.03	3.21% ± 0.03	**2.87% ± 0.03**	7.42%
CPU time (s)			5.391	5.684	5.627	5.604	
Wine 178 × 13 (3 classes)	Standardization	Inhom. linear	2.77% ± 0.02	2.63% ± 0.02	2.63% ± 0.02	**2.51% ± 0.02**	9.39%
CPU time (s)			7.916	8.361	8.352	8.605	

where τ^{det} and $\tau^{\text{rob}*}$ are the average out-of-sample testing errors of the deterministic and the best robust performing model, respectively. The results on the IR further confirm that robust methods provide superior accuracy when the uncertainty is handled in the classification process. Extensive results on the improvement ratio are reported in Table B.22 in Appendix B.

For the sake of completeness, we explore in details the performance of the proposed models when classifying datasets "Parkinson" and "Breast Cancer Diagnostic". First of all, we discuss the results of the deterministic approach, with respect to both data transformation and kernel function. The out-of-sample testing errors for the holdout 75%–25% are depicted in Fig. 5, while detailed results are reported in Table B.7 in Appendix B. We note that the worst performance occurs when no data transformations are applied. Conversely, min–max normalization and standardization provide good and comparable results. Similar conclusions can be drawn for holdouts 50%–50% and 25%–75% (see Tables B.8–B.9 in Appendix B).

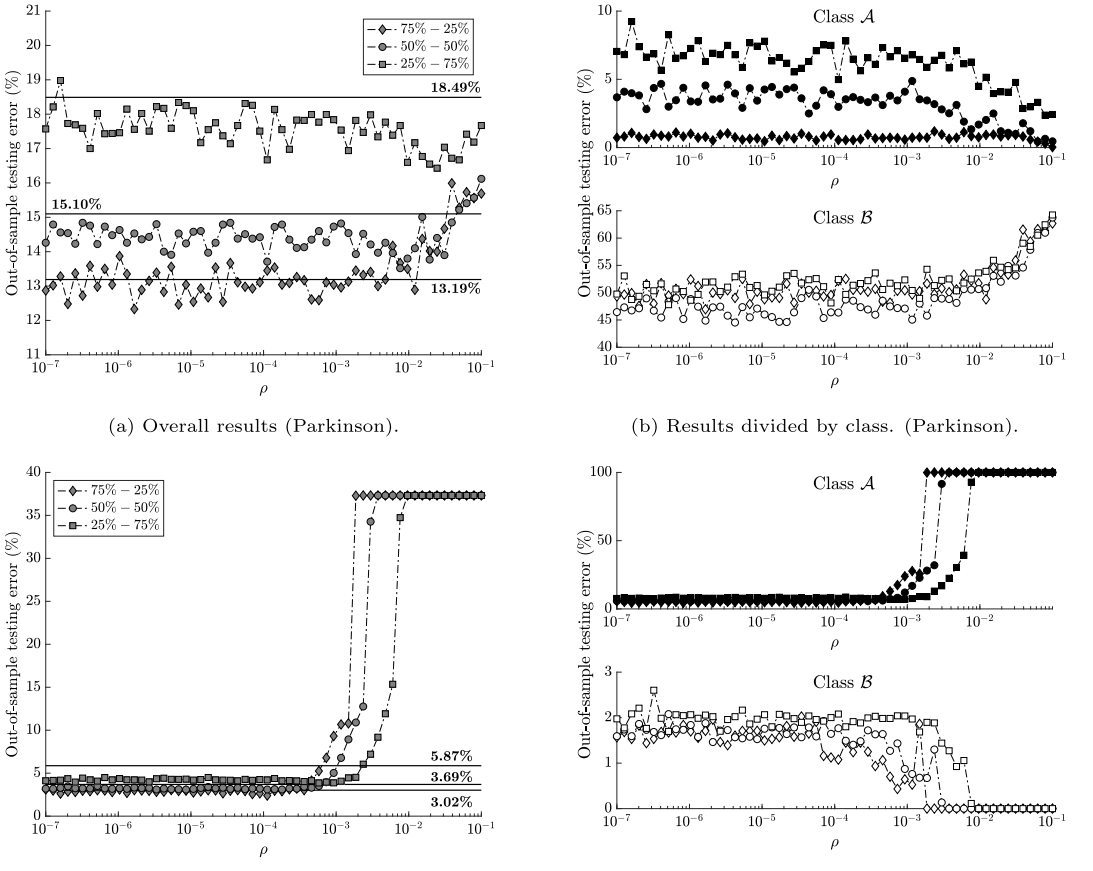
In order to evaluate the performance of the robust model, we consider 60 logarithmically spaced values of ρ between 10^{-7} and 10^{-1}. The results are depicted in Fig. 6. We notice that increasing the value of β leads to better performance in terms of the overall out-of-sample testing error (see Figs. 6(a), 6(c)), since more data points in the training set are available as input of the optimization model. In addition, when perturbations are included in the model, the performance improves with respect to the deterministic case. Indeed, the great majority of the points lies below the corresponding horizontal line, representing the out-of-sample testing error of the deterministic classifier. Interestingly, the increase of the uncertainty impacts differently on the two classes (see Figs. 6(b), 6(d)). For instance, the "Breast Cancer Diagnostic" dataset is not able to bear high levels of uncertainty ($\rho > 10^{-3}$) since all data points of class $\mathcal{A}$, representing patients with a malignant tumor, are misclassified. On the other hand, all observations in class $\mathcal{B}$ (patients with a benign tumor) are assigned to the correct category. From a practical perspective, given that classifying people with a malignant tumor as people with a benign tumor is worse than the opposite, robust models with low degree of perturbation should be considered in this case.

In Table 4 we report a comparison between the best results of Table 3 and the out-of-sample testing errors provided by the SVM classifier of *scikit-learn*, a popular ML library implemented in Python (Pedregosa et al., 2011). We tested the seven different kernels and reported in column 5 the best choice in terms of the lowest out-of-sample testing error. From column 6, it can be noted that in 8 out of 10 datasets the formulation proposed in this study outperforms the one implemented in the scikit-learn library for SVM.



(a) Parkinson. (b) Breast Cancer Diagnostic.

Fig. 5. Out-of-sample testing error of the deterministic formulation applied to the datasets "Parkinson" and "Breast Cancer Diagnostic". Each triangle represents the lowest error for the corresponding data transformation technique. Holdout: 75% training set–25% testing set.



(a) Overall results (Parkinson). (b) Results divided by class. (Parkinson).

(c) Overall results. (Breast Cancer Diagnostic). (d) Results divided by class. (Breast Cancer Diagnostic).

Fig. 6. Out-of-sample testing error of the robust formulation applied to the datasets "Parkinson" and "Breast Cancer Diagnostic". Overall results are on the left, with the performance of the deterministic classifier depicted as horizontal line for each holdout. Results divided by class are on the right. The values of ρ are in logarithmic scale.

In addition, we compare the performance of our proposal with results from other SVM formulations present in the ML literature (see Table 5). Specifically, as deterministic models we consider the linear classifiers proposed in Bertsimas et al. (2019), Jayadeva et al. (2007), and Liu and Potra (2009), as well as the kernelized TWSVM classifier from Jayadeva et al. (2007). For all of these models, we tuned the hyperparameter in the objective function using the same grid-search strategy employed in this paper. Following Peng (2011), to prevent issues related to ill-conditioning, we included a regularization term in the objective function of the kernelized TWSVM approach (see Suman, 2018 for further details on the MATLAB implementation). Finally, our robust formulation was compared with the robust classifiers from Bertsimas et al. (2019) and Faccini et al. (2022). As shown in Table 5a, in 5 out of 10 datasets the results of our deterministic classifiers outperform the other methods. Consequently, the linear approaches benefit from a generalization towards nonlinear classifier. Table 5b further shows that our robust formulation achieves even better accuracy in most of the cases.

To assess the good performance of the proposed approach over the other methods, we applied the Friedman test and the Holm test

Table 4
Out-of-sample testing error comparison among best results of Table 3 and simulations from the scikit-learn SVM library (Pedregosa et al., 2011). The lowest out-of-sample testing error within a dataset is highlighted.

Dataset	Data transformation	Table 3		Scikit-learn SVM library	
		Best kernel	Result	Best kernel	Result
Arrhythmia	–	Gaussian RBF	**19.12% ± 0.08**	Gaussian RBF	19.48% ± 0.07
Parkinson	Min–max normalization	Hom. linear	12.37% ± 0.03	Inhom. cubic	**9.41% ± 0.04**
Heart Disease	Standardization	Inhom. linear	**16.36% ± 0.04**	Inhom. linear	16.63% ± 0.04
Dermatology	–	Inhom. quadratic	0.55% ± 0.01	Inhom. linear	**0.11% ± 0.01**
Climate Model Crashes	–	Hom. linear	**4.34% ± 0.01**	Inhom. linear	4.78% ± 0.01
Breast Cancer Diagnostic	Min–max normalization	Inhom. quadratic	**2.39% ± 0.01**	Hom. cubic	2.78% ± 0.01
Breast Cancer	Standardization	Hom. linear	**2.97% ± 0.01**	Gaussian RBF	3.04% ± 0.01
Blood Transfusion	Standardization	Inhom. cubic	**20.55% ± 0.02**	Inhom. cubic	21.65% ± 0.02
Mammographic Mass	Standardization	Inhom. quadratic	**15.42% ± 0.02**	Inhom. quadratic	16.05% ± 0.02
Qsar Biodegradation	Min–max normalization	Gaussian RBF	**11.78% ± 0.01**	Inhom. quadratic	12.57% ± 0.02

Table 5
Out-of-sample testing error comparison among deterministic and robust results obtained from SVM formulations in the literature. For each approach and dataset, the best result is underlined.

(a) Deterministic formulations.

Dataset	SVM classifier				
	This paper	Linear Liu and Potra (2009)	Linear Bertsimas et al. (2019)	Linear TWSVM Jayadeva et al. (2007)	Kernelized TWSVM Jayadeva et al. (2007)
Arrhythmia	20.47%	25.65%	43.08%	20.34%	24.33%
Parkinson	13.19%	14.13%	14.36%	16.10%	15.71%
Heart Disease	17.48%	16.68%	15.93%	16.96%	16.31%
Dermatology	1.64%	0.56%	3.38%	1.12%	0.18%
Climate Model Crashes	5.01%	4.99%	5.00%	13.67%	5.92%
Breast Cancer Diagnostic	3.02%	4.89%	6.49%	3.62%	4.50%
Breast Cancer	3.17%	3.49%	5.00%	4.08%	4.00%
Blood Transfusion	20.72%	23.49%	23.62%	37.12%	23.22%
Mammographic Mass	15.71%	–	18.07%	17.32%	17.92%
Qsar Biodegradation	12.88%	–	12.51%	14.69%	13.24%

(b) Robust formulations.

Dataset	SVM classifier		
	This paper	Robust linear Faccini et al. (2022)	Robust linear Bertsimas et al. (2019)
Arrhythmia	19.12%	23.00%	29.23%
Parkinson	12.37%	13.00%	16.41%
Heart Disease	16.36%	16.20%	16.61%
Dermatology	0.55%	0.13%	1.13%
Climate Model Crashes	4.34%	4.34%	4.07%
Breast Cancer Diagnostic	2.39%	3.89%	4.04%
Breast Cancer	2.97%	3.12%	4.26%
Blood Transfusion	20.55%	22.55%	23.62%
Mammographic Mass	15.42%	–	19.28%
Qsar Biodegradation	11.78%	–	12.42%

(Demšar, 2006). First of all, we computed the average rank R_j for each of the methods on the basis of the out-of-sample testing error (see columns 2 and 4 in Table 6). Then, the Friedman test with Iman–Davenport correction is applied to verify whether such ranks are statistically similar (null hypothesis). The statistic F_F associated with the test is given by:

$$F_F = \frac{(N_d - 1)\chi_F^2}{N_d(N_m - 1) - \chi_F^2},$$

with

$$\chi_F^2 = \frac{12N_d}{N_m(N_m+1)}\left[\sum_{j=1}^{N_m} R_j^2 - \frac{N_m(N_m+1)^2}{4}\right],$$

where $N_d = 8$ is the number of datasets (we excluded "Mammographic Mass" and "Qsar Biodegradation" since they were not considered in Faccini et al., 2022) and N_m is the number of methodologies (5 for the deterministic and 3 for the robust). Under the null hypothesis, F_F is distributed according to the F-distribution with $N_m - 1$ and $(N_m - 1)(N_d - 1)$ degrees of freedom. In our case, the p-values associated with the Friedman test are 0.243 and 0.014 for the deterministic and robust approach, respectively. This implies that for the robust classifiers the null hypothesis of equal ranks is rejected with a significance level lower than $\alpha_R = 5\%$. Since such hypothesis does not hold, we performed pairwise comparisons between the robust classifier with the highest rank R^* and those remaining. To this extent, we considered the Holm test (Demšar, 2006) whose statistic z_j for comparing the best classifier with the jth one is computed as follows:

$$z_j = (R^* - R_j)\sqrt{\frac{6N_d}{N_m(N_m+1)}}.$$

Under the null hypothesis of outperformance of the best method over the others, the test statistic is distributed as a standard normal distribution. The results of the Holm test are presented in Table 6 (see columns 3–7). The null hypothesis is rejected when the p-value of the test is below the significance thresholds of column 6. It can be seen that the proposed model achieves the highest rank in both the deterministic and robust formulation, outperforming the robust linear SVM approach presented in Bertsimas et al. (2019). On the other hand, there are no statistically significant differences between our proposal and the robust method devised in Faccini et al. (2022), even if in most cases the results confirm the good performance of the proposed methodology (see Table 5b).

Table 6
Mean ranks of the deterministic formulations (columns 1–2). Holm test for pairwise comparison of robust formulations, with $\alpha_R = 0.05$ and $j = 2, 3$ (columns 3–7).

Deterministic formulation		Robust formulation				
SVM classifier	Mean rank	SVM classifier	Mean rank	p-value	$\alpha_R/(j-1)$	Action
This paper	2.250	This paper	1.438	–	–	–
Faccini et al. (2022)	2.625	Faccini et al. (2022)	1.813	0.453	0.050	Not reject
Kernelized TWSVM (Jayadeva et al., 2007)	2.750	Bertsimas et al. (2019)	2.750	0.009	0.025	Reject
Linear TWSVM (Jayadeva et al., 2007)	3.625					
Bertsimas et al. (2019)	3.750					

From Table 3 it can be noticed that the choice of the best data transformation method strongly depends on the dataset. In order to guide the final user among the three possible techniques, we report in Table B.23 in Appendix B summary statistics on the 10 datasets deployed for binary classification task. Specifically, for each feature we compute the mean and the corresponding coefficient of variation, defined as the ratio between the standard deviation and the mean. In Table B.23 we list the minimum and the maximum values of the two considered indices for each dataset, along with the corresponding best data transformation. We argue that, whenever the values of the observations are close, the best approach is to classify the original data without any transformation (see datasets "Arrhythmia", "Dermatology" and "Climate Model Crashes"). In the extreme case of constant features, pre-processing techniques of data transformation cannot be applied (see dataset "Arrhythmia"). On the other hand, the min–max normalization is a suitable choice when the order of magnitude across the features varies a lot. For instance, in datasets "Parkinson" and "Breast Cancer Diagnostic" there are 7 and 5 orders of magnitude of difference between the minimum and the maximum value of the mean of the features, respectively. Finally, standardization is an appropriate technique in all other cases, where no significant differences occur among the orders of magnitude of the features (see datasets "Heart Disease", "Breast Cancer", "Blood Transfusion" and "Mammographic Mass").

Finally, numerical results show that the computational time is significantly high for datasets with a large number of observations, especially when considering 75% of the instances as training set (see Table B.7 in Appendix B). The performing speed benefits from a reduction of β, even if at the cost of worsening the accuracy. Nevertheless, when datasets are equally split in training and testing set, the out-of-sample testing error does not increase significantly if compared to the holdout 75%–25% (see Table B.8). A similar conclusion can be drawn for the robust model (see Tables B.13–B.16). Conversely, from the time complexity analysis, it should be noticed that the number $N_{\max}$ of sub-intervals chosen to solve problem (8) and its variants impacts on the overall computational time especially when the number of observations is not significantly high. Therefore, the final user should properly choose the values of β and $N_{\max}$ to guarantee high accuracy in a reasonable time.

7. Conclusions

In this paper, we have proposed novel optimization models for solving binary and multiclass classification tasks through a Support Vector Machine (SVM) approach. From a methodological perspective, we have extended the techniques presented in Faccini et al. (2022) and Liu and Potra (2009) to the nonlinear context through the introduction of kernel functions. Data are mapped from the input space to the feature space where a first classification via kernelized SVM is performed. The optimal classifier is then constructed as the solution of a linear search procedure aiming to minimize the overall misclassification error.

Motivated by the uncertain nature of real-world data, we have adopted a Robust Optimization (RO) approach by constructing around each training data a bounded-by-ℓ_p-norm uncertainty set, with $p \in [1, \infty]$. Perturbation propagates from the input space to the feature space through the feature map associated with the kernel function. To face this problem, we have rigorously derived closed-form expressions for the uncertainty set bounds in the feature space, extending the results present in the literature. Thanks to this, we have formulated the robust counterpart of the deterministic models in the case of nonlinear classifiers. To enhance generalization, in all the proposed formulations we have considered a ℓ_q-norm with $q \in [1, \infty]$ as measure of the SVM-margin. Since the resulting robust problem turns to be convex but nonlinear, we have proved that in specific cases it can be reformulated as a LP or a SOCP problem, with clear advantages in terms of computational efficiency.

The proposed models have been tested on real-world datasets, considering different combinations of data transformations and kernel functions. The results show that our robust formulation outperforms other linear and kernelized SVM approaches in most cases. This has been confirmed by classical statistical tests deployed to compare the performance of machine learning techniques. Overall, the models benefit from including uncertainty in the training process. The accuracy is clearly affected by the choice of the kernel function and of the data transformation before training. Therefore, we have provided insights to guide the final user in choosing the best configuration.

Regarding future advancements, various streams of research can originate from this work. First of all, extend the approach to handle uncertainties in the labels of training data. This could increase the generalization capability of the models. Additionally, in this work we have followed the classical RO approach of including uncertainty during the training phase (see, for instance, Bertsimas et al., 2019). It could be noteworthy to consider perturbations both in the training and in the testing sets. However, this choice increases the complexity of the models and novel measures to quantify the accuracy have to be devised, since it is not obvious how to classify an entire uncertainty set in one class or another as opposed to the case of single data point. The main limitation of the current proposal is the complexity of the two-step procedure, leading to a time-consuming process. Further techniques could be employed to speed up the approach and to optimize the tuning phase of the model's parameters (see, for example, the Bayesian optimization in Snoek, Larochelle, & Adams, 2012). Finally, different methodologies could be applied to further robustify the models. For instance, Chance-Constrained Programming and Distributionally Robust Optimization with ambiguity sets defined by moments, phi-divergences or Wasserstein distance merit further research too.

CRediT authorship contribution statement

Francesca Maggioni: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. **Andrea Spinelli:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation.

Acknowledgments

This work has been supported by "ULTRA OPTYMAL - Urban Logistics and sustainable TRAnsportation: OPtimization under uncertainTY and MAchine Learning", a PRIN2020 project funded by the Italian University and Research Ministry (grant number 20207C8T9M).

This study was also carried out within the MOST - Sustainable Mobility National Research Center and received funding from the European Union Next-GenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR) – MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.4 – D.D. 1033 17/06/2022, CN00000023), Spoke 5 "Light Vehicle and Active Mobility". This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

The authors finally acknowledge the support received from Gruppo Nazionale per il Calcolo Scientifico (GNCS-INdAM).

Appendix A. Supplementary proofs

We first recall a lemma that will be useful to prove Propositions 1–2.

Lemma 1 (*Inequalities in ℓ_p-norm*). *Let x be a vector in $\mathbb{R}^n$. If $1 \leqslant p \leqslant q \leqslant \infty$, then:*

$$\|x\|_q \leqslant \|x\|_p \leqslant n^{\frac{1}{p}-\frac{1}{q}} \|x\|_q . \tag{A.1}$$

Proof. We consider the two inequalities separately, starting from $\|x\|_q \leqslant \|x\|_p$. First of all, if $x = 0$, then the inequality is obviously true. Otherwise, let $y \in \mathbb{R}^n$ such that $y_i := |x_i| / \|x\|_q$ for $i = 1, \dots, n$. Therefore, $0 \leqslant y_i \leqslant 1$. Indeed:

$$\|x\|_q^q = \sum_{i=1}^n |x_i|^q \geqslant |x_i|^q ,$$

for all $i = 1, \dots, n$ and thus $|x_i| / \|x\|_q \leqslant 1$. The hypothesis $p \leqslant q$ and the decreasing property of the exponential function with basis lower than one imply that:

$$y_i^p \geqslant y_i^q, \qquad i = 1, \dots, n.$$

By summing we have:

$$\|y\|_p \geqslant \|y\|_q .$$

Finally, by definition of y we derive that:

$$\frac{\|x\|_p}{\|x\|_q} \geqslant \frac{\|x\|_q}{\|x\|_q} = 1,$$

from which the thesis follows.

On the other hand, to prove the second inequality we recall the Hölder inequality (see, for instance, Rudin, 1987). Let a and b be in $\mathbb{R}^n$. If r and r' are conjugate exponents, i.e. $\frac{1}{r} + \frac{1}{r'} = 1$, with $1 \leqslant r, r' \leqslant \infty$, then:

$$\|ab\|_1 \leqslant \|a\|_r \cdot \|b\|_{r'} ,$$

or, equivalently:

$$\sum_{i=1}^n |a_i|\,|b_i| \leqslant \left(\sum_{i=1}^n |a_i|^r\right)^{\frac{1}{r}} \cdot \left(\sum_{i=1}^n |b_i|^{r'}\right)^{\frac{1}{r'}} . \tag{A.2}$$

First of all, we rewrite the ℓ_p-norm of x as:

$$\|x\|_p^p = \sum_{i=1}^n |x_i|^p = \sum_{i=1}^n |x_i|^p \cdot 1.$$

In the Hölder inequality (A.2), let $a = x$ and $b = e$, i.e. the vector of ones in $\mathbb{R}^n$, and consider as conjugate exponents $r = \frac{q}{p}$ and $r' = \frac{q}{q-p}$. Both r and r' are greater than or equal to 1 because, by hypothesis, $p \leqslant q$. Consequently, we can bound the ℓ_p-norm of x by:

$$\begin{aligned}\|x\|_p^p &\leqslant \left(\sum_{i=1}^n \left(|x_i|^p\right)^{\frac{q}{p}}\right)^{\frac{p}{q}} \cdot \left(\sum_{i=1}^n 1^{\frac{q}{q-p}}\right)^{1-\frac{p}{q}} \\ &= \left(\sum_{i=1}^n |x_i|^q\right)^{\frac{p}{q}} n^{1-\frac{p}{q}} = \|x\|_q^p \, n^{1-\frac{p}{q}} .\end{aligned}$$

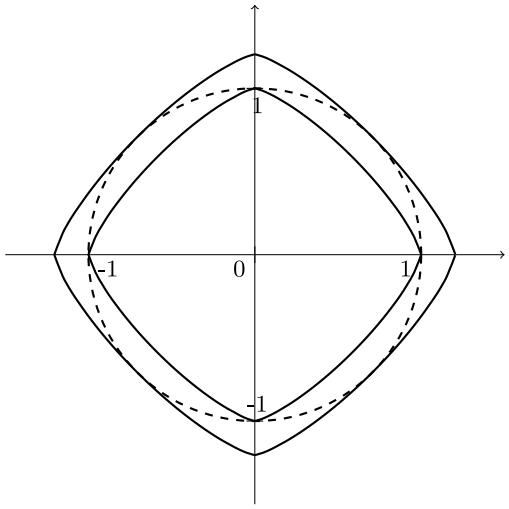


Fig. A.7. Graphical representation of Lemma 1 in the case of $p = 1.3$, $q = 2$, $n = 2$. The dashed ℓ_2 unit ball lies between the $\ell_{1.3}$ unit ball and the $\ell_{1.3}$ ball with radius $2^{\frac{1}{1.3}-\frac{1}{2}} \approx 1.205$.

Finally, the thesis follows by taking the pth root of both sides of the inequality. □

A graphical representation of inequality (A.1) is depicted in Fig. A.7.

As special cases, Lemma 1 implies that, whenever $1 \leqslant p \leqslant 2$, then:

$$\|x\|_2 \leqslant \|x\|_p . \tag{A.3}$$

Conversely, if $p > 2$, then:

$$\|x\|_2 \leqslant n^{\frac{p-2}{2p}} \|x\|_p . \tag{A.4}$$

Thus, combining these results, we can write:

$$\|x\|_2 \leqslant C \, \|x\|_p ,$$

with:

$$C = C(n,p) = \begin{cases} 1, & 1 \leqslant p \leqslant 2 \\ n^{\frac{p-2}{2p}}, & p > 2. \end{cases} \tag{A.5}$$

Proof of Proposition 1. The $\mathcal{H}$-norm of the vector of perturbation $\zeta^{(i)}$ in the feature space can be expanded as:

$$\begin{aligned}\left\|\zeta^{(i)}\right\|_{\mathcal{H}}^2 &= \left\|\phi(x) - \phi(x^{(i)})\right\|_{\mathcal{H}}^2 \\ &= \left\|\phi(x^{(i)} + \sigma^{(i)}) - \phi(x^{(i)})\right\|_{\mathcal{H}}^2 \\ &= \langle \phi(x^{(i)} + \sigma^{(i)}) - \phi(x^{(i)}), \phi(x^{(i)} + \sigma^{(i)}) - \phi(x^{(i)}) \rangle \\ &= \langle \phi(x^{(i)} + \sigma^{(i)}), \phi(x^{(i)} + \sigma^{(i)}) \rangle - 2\langle \phi(x^{(i)} + \sigma^{(i)}), \phi(x^{(i)}) \rangle \\ &\quad + \langle \phi(x^{(i)}), \phi(x^{(i)}) \rangle \\ &= k(x^{(i)} + \sigma^{(i)}, x^{(i)} + \sigma^{(i)}) - 2k(x^{(i)} + \sigma^{(i)}, x^{(i)}) + k(x^{(i)}, x^{(i)}).\end{aligned} \tag{A.6}$$

By definition of the inhomogeneous polynomial kernel of degree d, the last right-hand side of (A.6) becomes:

$$\begin{aligned}\left\|\zeta^{(i)}\right\|_{\mathcal{H}}^2 &= \left(\left\|x^{(i)} + \sigma^{(i)}\right\|_2^2 + c\right)^d - 2\left(\langle x^{(i)} + \sigma^{(i)}, x^{(i)} \rangle + c\right)^d \\ &\quad + \left(\left\|x^{(i)}\right\|_2^2 + c\right)^d \\ &= \left(\left\|x^{(i)}\right\|_2^2 + \left\|\sigma^{(i)}\right\|_2^2 + 2\,\langle \sigma^{(i)}, x^{(i)} \rangle + c\right)^d \\ &\quad - 2\left(\left\|x^{(i)}\right\|_2^2 + \langle \sigma^{(i)}, x^{(i)} \rangle + c\right)^d + \left(\left\|x^{(i)}\right\|_2^2 + c\right)^d .\end{aligned}$$

By applying the Cauchy–Schwarz inequality in $\mathbb{R}^n$ to the terms containing the dot product, the previous expression simplifies further, leading to:

$$\begin{aligned}\left\|\zeta^{(i)}\right\|_{\mathcal{H}}^2 \leqslant & \left(\left\|x^{(i)}\right\|_2^2+\left\|\sigma^{(i)}\right\|_2^2+2\left\|\sigma^{(i)}\right\|_2\left\|x^{(i)}\right\|_2+c\right)^d \\ & -2\left(\left\|x^{(i)}\right\|_2^2+\left\|\sigma^{(i)}\right\|_2\left\|x^{(i)}\right\|_2+c\right)^d+\left(\left\|x^{(i)}\right\|_2^2+c\right)^d \\ = & \left[\left(\left\|x^{(i)}\right\|_2+\left\|\sigma^{(i)}\right\|_2\right)^2+c\right]^d \\ & -2\left[\left\|x^{(i)}\right\|_2\left(\left\|x^{(i)}\right\|_2+\left\|\sigma^{(i)}\right\|_2\right)+c\right]^d+\left(\left\|x^{(i)}\right\|_2^2+c\right)^d.\end{aligned}$$

Applying the binomial expansion to the three dth powers implies that:

$$\begin{aligned}\left\|\zeta^{(i)}\right\|_{\mathcal{H}}^2 \leqslant & \sum_{k=0}^{d}\binom{d}{k}c^k\left(\left\|x^{(i)}\right\|_2+\left\|\sigma^{(i)}\right\|_2\right)^{2(d-k)} \\ & -2\sum_{k=0}^{d}\binom{d}{k}c^k\left\|x^{(i)}\right\|_2^{d-k}\left(\left\|x^{(i)}\right\|_2+\left\|\sigma^{(i)}\right\|_2\right)^{d-k} \\ & +\sum_{k=0}^{d}\binom{d}{k}c^k\left\|x^{(i)}\right\|_2^{2(d-k)}.\end{aligned}$$

We now split all the three sums by considering separately the cases when $k=0$, $k=d$ and, then, all the intermediate cases. Firstly, let us call a_0 the addendum of the sum corresponding to $k=0$. Therefore:

$$\begin{aligned}a_0 &= \left(\left\|x^{(i)}\right\|_2+\left\|\sigma^{(i)}\right\|_2\right)^{2d}-2\left\|x^{(i)}\right\|_2^d\left(\left\|x^{(i)}\right\|_2+\left\|\sigma^{(i)}\right\|_2\right)^d+\left\|x^{(i)}\right\|_2^{2d} \\ &= \left[\left(\left\|x^{(i)}\right\|_2+\left\|\sigma^{(i)}\right\|_2\right)^d-\left\|x^{(i)}\right\|_2^d\right]^2 \\ &= \left[\sum_{k=0}^{d}\binom{d}{k}\left\|x^{(i)}\right\|_2^{d-k}\left\|\sigma^{(i)}\right\|_2^k-\left\|x^{(i)}\right\|_2^d\right]^2 \\ &= \left[\sum_{k=1}^{d}\binom{d}{k}\left\|x^{(i)}\right\|_2^{d-k}\left\|\sigma^{(i)}\right\|_2^k+\left\|x^{(i)}\right\|_2^d-\left\|x^{(i)}\right\|_2^d\right]^2 \\ &= \left[\sum_{k=1}^{d}\binom{d}{k}\left\|x^{(i)}\right\|_2^{d-k}\left\|\sigma^{(i)}\right\|_2^k\right]^2.\end{aligned}$$

We notice that a_0 is the only addendum of the sum that does not contain c. This implies that a_0 is related to the bound $\delta_{d,0}^{(i)}$ for the homogeneous polynomial kernel.

Secondly, if $k=d$, we have no contribution because $c^d-2c^d+c^d=0$. Before considering the cases $k=1,\dots,d-1$, we now investigate what happens when the degree d is equal to 1. Here, the index k of the sums goes from 0 to 1, and therefore, as seen before:

$$\left\|\zeta^{(i)}\right\|_{\mathcal{H}}^2 \leqslant \left(\delta_{\text{hom}}^{(i)}\right)^2=\left(C\eta^{(i)}\right)^2.$$

Hence, when $d=1$, then $\delta_{1,c}^{(i)}=C\eta^{(i)}$. Conversely, when $d>1$, we have all the addenda between $k=1$ and $k=d-1$. Thus, by combining all the three sums together we have:

$$\begin{aligned}\left\|\zeta^{(i)}\right\|_{\mathcal{H}}^2 \leqslant & a_0+\sum_{k=1}^{d-1}\binom{d}{k}c^k\left[\left(\left\|x^{(i)}\right\|_2+\left\|\sigma^{(i)}\right\|_2\right)^{2(d-k)}\right. \\ & \left.-2\left\|x^{(i)}\right\|_2^{d-k}\left(\left\|x^{(i)}\right\|_2+\left\|\sigma^{(i)}\right\|_2\right)^{d-k}+\left\|x^{(i)}\right\|_2^{2(d-k)}\right] \\ = & a_0+\sum_{k=1}^{d-1}\binom{d}{k}c^k\left[\left(\left\|x^{(i)}\right\|_2+\left\|\sigma^{(i)}\right\|_2\right)^{d-k}-\left\|x^{(i)}\right\|_2^{d-k}\right]^2.\end{aligned}$$

Again, by applying the binomial expansion to the $(d-k)$-th power of $\left(\left\|x^{(i)}\right\|_2+\left\|\sigma^{(i)}\right\|_2\right)$ and by splitting the sum, we are able to simplify the last term. Hence:

$$\begin{aligned}\left\|\zeta^{(i)}\right\|_{\mathcal{H}}^2 &\leqslant a_0+\sum_{k=1}^{d-1}\binom{d}{k}c^k\left[\sum_{j=0}^{d-k}\binom{d-k}{j}\left\|x^{(i)}\right\|_2^{d-k-j}\left\|\sigma^{(i)}\right\|_2^j-\left\|x^{(i)}\right\|_2^{d-k}\right]^2 \\ &= a_0+\sum_{k=1}^{d-1}\binom{d}{k}c^k\left[\sum_{j=1}^{d-k}\binom{d-k}{j}\left\|x^{(i)}\right\|_2^{d-k-j}\left\|\sigma^{(i)}\right\|_2^j\right]^2.\end{aligned}$$

Therefore, by taking the square root:

$$\left\|\zeta^{(i)}\right\|_{\mathcal{H}} \leqslant \sqrt{a_0+\sum_{k=1}^{d-1}\binom{d}{k}c^k\left[\sum_{j=1}^{d-k}\binom{d-k}{j}\left\|x^{(i)}\right\|_2^{d-k-j}\left\|\sigma^{(i)}\right\|_2^j\right]^2}.$$

According to inequalities (A.3)–(A.4) and to hypothesis $\left\|\sigma^{(i)}\right\|_p \leqslant \eta^{(i)}$, we obtain that:

$$\left\|\sigma^{(i)}\right\|_2 \leqslant \begin{cases}\left\|\sigma^{(i)}\right\|_p \leqslant \eta^{(i)}, & 1\leqslant p\leqslant 2 \\ n^{\frac{p-2}{2p}}\left\|\sigma^{(i)}\right\|_p \leqslant n^{\frac{p-2}{2p}}\eta^{(i)}, & p>2.\end{cases}$$

Finally, whenever $1\leqslant p\leqslant 2$, we have that:

$$\begin{aligned}a_0 &\leqslant \left[\sum_{k=1}^{d}\binom{d}{k}\left\|x^{(i)}\right\|_2^{d-k}\left\|\sigma^{(i)}\right\|_p^k\right]^2 \leqslant \left[\sum_{k=1}^{d}\binom{d}{k}\left\|x^{(i)}\right\|_2^{d-k}\left(\eta^{(i)}\right)^k\right]^2 \\ &= \left(\delta_{d,0}^{(i)}\right)^2,\end{aligned}$$

and the second addendum in the square root can be bounded by:

$$\sum_{k=1}^{d-1}\binom{d}{k}c^k\left[\sum_{j=1}^{d-k}\binom{d-k}{j}\left\|x^{(i)}\right\|_2^{d-k-j}\left(\eta^{(i)}\right)^j\right]^2.$$

On the other hand, if $p>2$, then:

$$\begin{aligned}a_0 &\leqslant \left[\sum_{k=1}^{d}\binom{d}{k}\left\|x^{(i)}\right\|_2^{d-k}n^{\frac{k(p-2)}{2p}}\left\|\sigma^{(i)}\right\|_p^k\right]^2 \\ &\leqslant \left[\sum_{k=1}^{d}\binom{d}{k}\left\|x^{(i)}\right\|_2^{d-k}\left(n^{\frac{p-2}{2p}}\eta^{(i)}\right)^k\right]^2=\left(\delta_{d,0}^{(i)}\right)^2,\end{aligned}$$

and similarly the second addendum in the square root is always less than or equal to:

$$\sum_{k=1}^{d-1}\binom{d}{k}c^k\left[\sum_{j=1}^{d-k}\binom{d-k}{j}\left\|x^{(i)}\right\|_2^{d-k-j}\left(n^{\frac{p-2}{2p}}\eta^{(i)}\right)^j\right]^2. \quad \square$$

Proof of Proposition 2. For all x in $\mathbb{R}^n$, we have that $k(x,x)=1$ and, thus, Eq. (A.6) reduces to:

$$\left\|\zeta^{(i)}\right\|_{\mathcal{H}}^2=1-2\exp\left(-\frac{\left\|x^{(i)}+\sigma^{(i)}-x^{(i)}\right\|_2^2}{2\alpha^2}\right)+1=2-2\exp\left(-\frac{\left\|\sigma^{(i)}\right\|_2^2}{2\alpha^2}\right).$$

Therefore:

$$\left\|\zeta^{(i)}\right\|_{\mathcal{H}}=\sqrt{2-2\exp\left(-\frac{\left\|\sigma^{(i)}\right\|_2^2}{2\alpha^2}\right)}.$$

The thesis follows by applying inequalities (A.3)–(A.4) and by considering the monotonicity of function $g(x)=-\exp(-x^2)$ when $x>0$. $\square$

Proof of Corollary 1.

(a) If $q=1$, model (18) can be rewritten as model (23) by introducing an auxiliary vector $s\in\mathbb{R}^m$ such that each component s_i is equal to $|u_i|$ and adding the constraints $s_i\geqslant 0$, $s_i\geqslant -u_i$ and $s_i\geqslant u_i$ for all $i=1,\dots,m$.

(b) If $q = 2$, the quadratic term $\|u\|_2^2$ can be transformed from the objective function to the set of constraints by introducing auxiliary variables $r, t, v \in \mathbb{R}$ such that $t \geqslant \|u\|_2$, $r + v = 1$ and $r \geqslant \sqrt{t^2 + v^2}$ (Qi et al., 2013). With the same reasoning at point (a), model (18) reduces to model (24).

(c) If $q = \infty$, by introducing an auxiliary variable $s_\infty \geqslant 0$ equal to $\|u\|_\infty$, and adding the constraints $s_\infty \geqslant -u_i$ and $s_\infty \geqslant u_i$ for all $i = 1, \ldots, m$, model (18) is equivalent to model (25) with the same reasoning at point (a). □

Appendix B. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.ejor.2024.12.014.

References

Ben-Tal, A., Bhadra, S., Bhattacharyya, C., & Nemirovski, A. (2012). Efficient methods for robust classification under uncertainty in kernel matrices. *Journal of Machine Learning Research*, *13*, 2923–2954.

Ben-Tal, A., El Ghaoui, L., & Nemirovski, A. (2009). *Robust optimization*. Princeton University Press.

Bengio, Y., Lodi, A., & Prouvost, A. (2021). Machine learning for combinatorial optimization: A methodological tour d'horizon. *European Journal of Operational Research*, *290*(2), 405–421.

Benítez-Peña, S., Blanquero, R., Carrizosa, E., & Ramírez-Cobo, P. (2024). Cost-sensitive probabilistic predictions for support vector machines. *European Journal of Operational Research*, *314*(1), 268–279.

Bennett, K. P., & Mangasarian, O. L. (1992). Robust linear programming discrimination of two linearly inseparable sets. *Optimization Methods & Software*, *1*, 23–34.

Bertsimas, D., Brown, D. B., & Caramanis, C. (2011). Theory and applications of robust optimization. *SIAM Review*, *53*, 464–501.

Bertsimas, D., Dunn, J., Pawlowski, C., & Zhuo, Y. D. (2019). Robust classification. *INFORMS Journal of Optimization*, *1*, 2–34.

Bhadra, S., Bhattacharya, S., Bhattacharyya, C., & Ben-Tal, A. (2010). Robust formulations for handling uncertainty in kernel matrices. In *Proceedings for the 27th international conference on machine learning* (pp. 71–78).

Bhattacharyya, C. (2004). Robust classification of noisy data using second order cone programming approach. In *International conference on intelligent sensing and information processing, 2004* (pp. 433–438).

Bi, J., & Zhang, T. (2005). Support vector classification with input data uncertainty. In *Advances in neural information processing systems* (pp. 161–168).

Blanco, V., Puerto, J., & Rodríguez-Chía, A. M. (2020). On lp-support vector machines and multidimensional kernels. *Journal of Machine Learning Research*, *21*, 1–29.

Boser, B. E., Guyon, I., & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. *vol. 5*, In *Proceedings of the fifth annual workshop of computational learning theory* (pp. 144–152).

Cervantes, J., Garcia-Lamont, F., Rodríguez-Mazahua, L., & Lopez, A. (2020). A comprehensive survey on support vector machine classification: Applications, challenges and trends. *Neurocomputing*, *408*, 189–215.

Chen, Z.-Y., Fan, Z.-P., & Sun, M. (2012). A hierarchical multiple kernel support vector machine for customer churn prediction using longitudinal behavioral data. *European Journal of Operational Research*, *223*(2), 461–472.

Chen, T. Y., Tse, T. H., & Yu, Y.-T. (2001). Proportional sampling strategy: a compendium and some insights. *Journal of Systems and Software*, *58*(1), 65–81.

Cortes, C., & Vapnik, V. N. (1995). Support-vector networks. *Machine Learning*, *20*, 273–297.

De Bock, K. W., Coussement, K., Caigny, A. D., Słowiński, R., Baesens, B., Boute, R. N., et al. (2024). Explainable AI for operational research: A defining framework, methods, applications, and a research agenda. *European Journal of Operational Research*, *317*(2), 249–272.

De Leone, R., Maggioni, F., & Spinelli, A. (2023). A robust twin parametric margin support vector machine for multiclass classification. URL: https://arxiv.org/abs/2306.06213.

De Leone, R., Maggioni, F., & Spinelli, A. (2024). A multiclass robust twin parametric margin support vector machine with an application to vehicles emissions. In G. Nicosia, V. Ojha, E. La Malfa, G. La Malfa, P. M. Pardalos, & R. Umeton (Eds.), *Lecture notes in computer science*: *14506, Machine learning, optimization, and data science* (pp. 299–310). Cham: Springer Nature Switzerland, http://dx.doi.org/10.1007/978-3-031-53966-4_22.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, *7*, 1–30.

Ding, S., & Hua, X. (2014). Recursive least squares projection twin support vector machines for nonlinear classification. *Neurocomputing*, *130*, 3–9, Track on Intelligent Computing and Applications Complex Learning in Connectionist Networks.

Ding, S., Zhao, X., Zhang, J., Zhang, X., & Xue, Y. (2019). A review on multi-class TWSVM. *Artificial Intelligence Review*, *52*, 775–801.

Doumpos, M., Zopounidis, C., Gounopoulos, D., Platanakis, E., & Zhang, W. (2023). Operational research and artificial intelligence methods in banking. *European Journal of Operational Research*, *306*(1), 1–16.

Du, S.-W., Zhang, M.-C., Chen, P., Sun, H.-F., Chen, W.-J., & Shao, Y.-H. (2021). A multiclass nonparallel parametric-margin support vector machine. *Information*, *12*(12), 515–533.

El Ghaoui, L., Lanckriet, G. R. G., Natsoulis, G., et al. (2003). *Robust classification with interval data*. Computer Science Division, University of California Berkeley.

Faccini, D., Maggioni, F., & Potra, F. A. (2022). Robust and distributionally robust optimization models for linear support vector machine. *Computers & Operations Research*, *147*, Article 105930.

Fan, N., Sadeghi, E., & Pardalos, P. M. (2014). Robust support vector machines with polyhedral uncertainty of the input data. In *Learning and intelligent optimization. international conference on learning and intelligent optimization* (pp. 291–305). Springer-Verlag.

Fung, G., Mangasarian, O. L., & Shavlik, J. W. (2002). Knowledge-based support vector machine classifiers. In *NIPS* (pp. 521–528).

Gambella, C., Ghaddar, B., & Naoum-Sawaya, J. (2021). Optimization problems for machine learning: A survey. *European Journal of Operational Research*, *290*(3), 807–828.

Gao, Z., Fang, S.-C., Luo, J., & Medhin, N. (2021). A kernel-free double well potential support vector machine with applications. *European Journal of Operational Research*, *290*(1), 248–262.

Grant, M., & Boyd, S. (2008). Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, & H. Kimura (Eds.), *Lecture notes in control and information sciences*, *Recent advances in learning and control* (pp. 95–110). Springer-Verlag Limited, http://stanford.edu/~boyd/graph_dcp.html.

Grant, M., & Boyd, S. (2014). CVX: Matlab software for disciplined convex programming, version 2.1. http://cvxr.com/cvx.

Gunnarsson, B. R., vanden Broucke, S., Baesens, B., Óskarsdóttir, M., & Lemahieu, W. (2021). Deep learning for credit scoring: Do or don't? *European Journal of Operational Research*, *295*(1), 292–305.

Han, J., Kamber, M., & Pei, J. (2011). *Data mining: concepts and techniques - 3rd edition*. Morgan Kaufmann.

Hao, P.-Y. (2010). New support vector algorithms with parametric insensitive/margin model. *Neural Networks : the Official Journal of the International Neural Network Society*, *23*(1), 60–73.

Jayadeva, Khemchandani, R., & Chandra, S. (2007). Twin support vector machines for pattern classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *29*(5), 905–910.

Jiang, J., & Peng, S. (2024). Mathematical programs with distributionally robust chance constraints: Statistical robustness, discretization and reformulation. *European Journal of Operational Research*, *313*(2), 616–627.

Jiménez-Cordero, A., Morales, J. M., & Pineda, S. (2021). A novel embedded min-max approach for feature selection in nonlinear support vector machine classification. *European Journal of Operational Research*, *293*, 24–35.

Ju, X., & jie Tian, Y. (2012). Knowledge-based support vector machine classifiers via nearest points. *Procedia Computer Science*, *9*, 1240–1248.

Katsafados, A. G., Leledakis, G. N., Pyrgiotakis, E. G., Androutsopoulos, I., & Fergadiotis, M. (2024). Machine learning in bank merger prediction: A text-based approach. *European Journal of Operational Research*, *312*(2), 783–797.

Kelly, M., Longjohn, R., & Nottingham, K. (2023). UCI machine learning repository. URL: http://archive.ics.uci.edu/ml.

Ketkov, S. S. (2024). A study of distributionally robust mixed-integer programming with wasserstein metric: on the value of incomplete data. *European Journal of Operational Research*, *313*(2), 602–615.

Khanjani-Shiraz, R., Babapour-Azar, A., Hosseini-Nodeh, Z., & Pardalos, P. M. (2023). Distributionally robust joint chance-constrained support vector machines. *Optimization Letters*, *17*, 299–332.

Kim, J.-H. (2009). Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap. *Computational Statistics & Data Analysis*, *53*, 3735–3745.

Labbé, M., Martínez-Merino, L. I., & Rodríguez-Chía, A. M. (2019). Mixed integer linear programming for feature selection in support vector machine. *Discrete Applied Mathematics*, *261*, 276–304.

Lanckriet, G. R. G., Ghaoui, L. E., Bhattacharyya, C., & Jordan, M. I. (2002). A robust minimax approach to classification. *Journal of Machine Learning Research*, *3*, 555–582.

Lee, Y.-J., Mangasarian, O. L., & Wolberg, W. H. (2000). Breast cancer survival and chemotherapy: a support vector machine analysis. *Discrete Mathematical Problems with Medical Applications*, *55*, 1–10.

Lee, I. G., Yoon, S. W., & Won, D. (2022). A mixed integer linear programming support vector machine for cost-effective group feature selection: Branch-cut-and-price approach. *European Journal of Operational Research*, *299*(3), 1055–1068.

Li, H., Liang, Y., & Xu, Q. (2009). Support vector machines and its applications in chemistry. *Chemometrics and Intelligent Laboratory Systems*, *95*(2), 188–198.

Liao, Z., Dai, S., & Kuosmanen, T. (2024). Convex support vector regression. *European Journal of Operational Research*, *313*(3), 858–870.

Lin, F., Fang, S.-C., Fang, X., & Gao, Z. (2024a). Distributionally robust chance-constrained kernel-based support vector machine. *Computers & Operations Research*, *170*, Article 106755.

Lin, F., Fang, S.-C., Fang, X., Gao, Z., & Luo, J. (2024b). A distributionally robust chance-constrained kernel-free quadratic surface support vector machine. *European Journal of Operational Research*, *316*(1), 46–60.

Liu, X., & Potra, F. A. (2009). Pattern separation and prediction via linear and semidefinite programming. *Studies in Informatics and Control*, *18*(1), 71–82.

López, J., Maldonado, S., & Carrasco, M. (2017). A robust formulation for twin multiclass support vector machine. *Applied Intelligence: The International Journal of Artificial Intelligence, Neural Networks, and Complex Problem-Solving Technologies*, *47*, 1031–1043.

López, J., Maldonado, S., & Carrasco, M. (2018). Double regularization methods for robust feature selection and SVM classification via DC programming. *Information Sciences*, *429*, 377–389.

López, J., Maldonado, S., & Carrasco, M. (2019). Robust nonparallel support vector machines via second-order cone programming. *Neurocomputing*, *364*, 227–238.

Luo, J., Yan, X., & Tian, Y. (2020). Unsupervised quadratic surface support vector machine with application to credit risk assessment. *European Journal of Operational Research*, *280*(3), 1008–1017.

Maggioni, F., Faccini, D., Gheza, F., Manelli, F., & Bonetti, G. (2023). Machine learning based classification models for COVID-19 patients. In R. Aringhieri, F. Maggioni, E. Lanzarone, M. Reuter-Oppermann, G. Righini, & M. T. Vespucci (Eds.), *Operations research for health care in red zone* (pp. 35–46). Cham: Springer International Publishing.

Maggioni, F., Potra, F. A., Bertocchi, M., & Allevi, E. (2009). Stochastic second-order cone programming in mobile ad hoc networks. *Journal of Optimization Theory and Applications*, *143*, 309–328.

Maggioni, F., & Spinelli, A. (2024). A robust nonlinear support vector machine approach for vehicles smog rating classification. In M. Bruglieri, P. Festa, G. Macrina, & O. Pisacane (Eds.), *AIRO springer series, Optimization in green sustainability and ecological transition*. Springer Cham, http://dx.doi.org/10.1007/978-3-031-47686-0_19.

Maldonado, S., López, J., & Carrasco, M. (2022). The Cobb–Douglas learning machine. *Pattern Recognition*, *128*, Article 108701.

Maldonado, S., López, J., & Vairetti, C. (2020). Profit-based churn prediction based on minimax probability machines. *European Journal of Operational Research*, *284*(1), 273–284.

Mangasarian, O. L. (1998). Generalized support vector machines. In *Advances in large margin classifiers* (pp. 135–146). MIT Press.

Marcelli, E., & De Leone, R. (2020). Multi-kernel covariance terms in multi-output support vector machines. In G. Nicosia, V. Ojha, E. La Malfa, G. Jansen, V. Sciacca, P. Pardalos, G. Giuffrida, & R. Umeton (Eds.), *Machine learning, optimization, and data science* (pp. 1–11). Cham: Springer International Publishing.

Mi, C., Wang, J., Mi, W., Huang, Y., Zhang, Z., Yang, Y., et al. (2019). Research on regional clustering and two-stage SVM method for container truck recognition. *Discrete and Continuous Dynamical Systems - S*, *12*(4–5), 1117–1133.

MOSEK ApS (2019). The MOSEK optimization toolbox for MATLAB manual. Version 9.1. URL: http://docs.mosek.com/9.1/toolbox/index.html.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Peng, X. (2011). TPMSVM: A novel twin parametric-margin support vector machine for pattern recognition. *Pattern Recognition*, *44*(10), 2678–2692.

Peng, X., & Xu, D. (2013). Robust minimum class variance twin support vector machine classifier. *Neural Computing and Applications*, *22*, 999–1011.

Piccialli, V., & Sciandrone, M. (2018). Nonlinear optimization and support vector machines. *4OR - A Quarterly Journal of Operations Research*, *16*, 111–149.

Qi, Z., Tian, Y., & Shi, Y. (2013). Robust twin support vector machine for pattern classification. *Pattern Recognition*, *46*(1), 305–316.

Raeesi, R., Sahebjamnia, N., & Mansouri, S. A. (2023). The synergistic effect of operational research and big data analytics in greening container terminal operations: A review and future directions. *European Journal of Operational Research*, *310*(3), 943–973.

Rudin, W. (1987). *Real and complex analysis*. McGraw-Hill.

Schölkopf, B., & Smola, A. J. (2001). *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. MIT Press.

Schölkopf, B., Smola, A., Williamson, R. C., & Bartlett, P. L. (2000). New support vector algorithms. *Neural Computation*, *12*(5), 1207–1245.

Singla, M., Ghosh, D., & Shukla, K. K. (2020). A survey of robust optimization based machine learning with special reference to support vector machines. *International Journal of Machine Learning and Cybernetics*, *11*, 1359–1385.

Snoek, J., Larochelle, H., & Adams, R. P. (2012). Practical Bayesian optimization of machine learning algorithms. In F. Pereira, C. Burges, L. Bottou, & K. Weinberger (Eds.), *vol. 25, Advances in neural information processing systems*. Curran Associates, Inc..

Suman, S. (2018). TwinSVM. URL: https://github.com/sumitsomans/TwinSVM.

Szelag, M., & Słowiński, R. (2024). Explaining and predicting customer churn by monotonic rules induced from ordinal data. *European Journal of Operational Research*, *317*(2), 414–424.

Tanveer, M., Rajani, T., Rastogi, R., & Shao, Y. (2022). Comprehensive review on twin support vector machines. *Annals of Operations Research*, 1–46.

Tay, F. E., & Cao, L. (2001). Application of support vector machines in financial time series forecasting. *Omega*, *29*(4), 309–317.

Trafalis, T. B., & Alwazzi, S. A. (2010). Support vector machine classification with noisy data: a second order cone programming approach. *International Journal of General Systems*, *39*, 757–781.

Trafalis, T. B., & Gilbert, R. C. (2006). Robust classification and regression using support vector machines. *European Journal of Operational Research*, *173*, 893–909.

Vapnik, V. N. (1995). *The nature of statistical learning theory*. Springer-Verlag.

Vapnik, V. N., & Chervonenkis, A. Y. (1974). *Theory of pattern recognition*. Moscow: Nauka.

Wang, X., Fan, N., & Pardalos, P. M. (2018). Robust chance-constrained support vector machines with second-order moment information. *Annals of Operations Research*, *263*, 45–68.

Wang, X., & Pardalos, P. M. (2014). A survey of support vector machines with uncertainties. *Annals of Data Science*, *1*, 293–309.

Wang, H., Zheng, B., Yoon, S. W., & Ko, H. S. (2018). A support vector machine-based ensemble algorithm for breast cancer diagnosis. *European Journal of Operational Research*, *267*(2), 687–699.

Wei, Z., Hao, J.-K., Ren, J., & Glover, F. (2023). Responsive strategic oscillation for solving the disjunctively constrained knapsack problem. *European Journal of Operational Research*, *309*(3), 993–1009.

Weston, J., & Watkins, C. (1998). *Multi-class support vector machines*: *Technical report*, Royal Holloway, University of London.

Xu, H., Caramanis, C., & Mannor, S. (2009). Robustness and regularization of support vector machines. *Journal of Machine Learning Research*, *10*, 1485–1510.

Yajima, Y. (2005). Linear programming approaches for multicategory support vector machines. *European Journal of Operational Research*, *162*(2), 514–531.

Yao, X., Crook, J., & Andreeva, G. (2017). Enhancing two-stage modelling methodology for loss given default with support vector machines. *European Journal of Operational Research*, *263*(2), 679–689.