Decision Analytics Journal 16 (2025) 100595

Contents lists available at ScienceDirect

Decision Analytics Journal

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

A robust support vector machine approach for Raman data classification

Marco Piazza [a], Andrea Spinelli [b], Francesca Maggioni [b], Marzia Bedoni [c], Enza Messina [a,*]

[a] Department of Informatics, Systems and Communication, University of Milano-Bicocca, Viale Sarca 336, Milan 20126, Italy
[b] Department of Management, Information and Production Engineering, University of Bergamo, Viale G. Marconi 5, Dalmine 24044, Italy
[c] IRCCS Fondazione Don Carlo Gnocchi ONLUS, Via Capecelatro 66, Milan 20148, Italy

ARTICLE INFO

Dataset link: https://github.com/piazzam/Robust-SVM-Raman

Keywords:
Machine learning
Raman spectroscopy
Support vector machine
Robust optimization
Bayesian optimization
COVID-19

ABSTRACT

Recent advances in healthcare technologies have led to the availability of large amounts of biological samples across several techniques and applications. In particular, in the last few years, *Raman spectroscopy* analysis of biological samples has been successfully applied for early-stage diagnosis. However, spectra's inherent complexity and variability make the manual analysis challenging, even for domain experts. For the same reason, the use of traditional *Statistical Learning* and *Machine Learning* techniques could not guarantee for accurate and reliable results. Machine learning models, combined with robust optimization techniques, offer the possibility to improve the classification accuracy and enhance the resilience of predictive models under data uncertainty. In this paper, we investigate the performance of a novel robust formulation for *Support Vector Machine* (SVM) in classifying COVID-19 samples obtained from Raman spectroscopy. Given the noisy and perturbed nature of biological samples, we protect the classification process against uncertainty through the application of robust optimization techniques. Specifically, we consider the robust counterparts of deterministic SVM formulations using bounded-by-norm uncertainty sets. We explore the cases of both linear and kernel-induced classifiers, addressing binary and multiclass classification tasks. The effectiveness of our approach is evaluated on real-world COVID-19 Raman saliva samples provided by Italian hospitals. We assess the performance of the proposed method by comparing the results of our numerical experiments with those of a state-of-the-art classifier, showing the potential of robust classifiers in handling uncertain Raman data.

1. Introduction

Raman Spectroscopy (RS) is a technique based on the inelastic scattering of monochromatic light to observe low-frequency modes in a molecular system (see [1]). The resulting scattering pattern serves as a "fingerprint", revealing information about the sample's chemical composition, including the presence, concentrations, and interactions of its molecules. In the healthcare field, the spectral information acquired from biological samples can be exploited to diagnose and monitor the emergence of pathologies by detecting certain biomarkers associated with the suspected condition. These applications typically involve a variety of target samples, such as blood-based fluids (serum, plasma) or human tissues (see [2–4]). In recent research (see, for instance, [5,6]), the analysis of saliva samples has demonstrated potential for identifying the presence of relevant biomarkers and their concentration, making saliva one of the most promising targets for analysis, especially considering its straightforward accessibility.

Labelled spectra coming from Raman spectroscopy analysis have been successfully used to train *Machine Learning* (ML) and *Deep Learning* (DL) models for classification purposes. Despite their potential, Deep Learning methods require vast amounts of data for an effective training. This represents a significant challenge in the field of spectroscopy, where the acquisition of data samples may require considerable financial, human, and time resources, potentially compromising the applicability of these models. On the other hand, a plethora of machine learning algorithms have been designed to handle classification problems: *Support Vector Machines* (SVMs, see [7]), *Linear Discriminant Analysis* (LDA, see [8]), *k-Nearest Neighbours* (see [9]) and *Decision Trees* (see [10]), to name a few. Among these, SVMs have received strong attention in the ML literature thanks to their simplicity and strong predictive performance (see [11]). Such methodology has been extensively employed to address classification tasks in medicine and healthcare applications (see [12–14]). In particular, in the context of Raman Spectroscopy, SVMs have recently been explored as fast and efficient tools for the early diagnosis of various diseases (see [15,16]).

Originally introduced in [7], classical SVM aims to find the best separating hyperplane that maximizes the margin between two classes of data. To improve the classification accuracy, many SVM-based models have been proposed in the literature (see, for instance, [17–20]).

* Corresponding author.
E-mail address: enza.messina@unimib.it (E. Messina).

https://doi.org/10.1016/j.dajour.2025.100595
Received 13 November 2024; Received in revised form 24 May 2025; Accepted 13 June 2025
Available online 18 June 2025

In this paper, we focus on the variant introduced in [21] and further extended in [22]. The strength of this approach over other SVMs lies in its two-step procedure. Indeed, rather than constructing a single hyperplane, the method first separates the training data using two parallel hyperplanes derived as solutions of a SVM model. The optimal final hyperplane is then searched within the region between these two, minimizing the total number of misclassified data points.

The interpretation of spectra obtained from RS, particularly salivary samples, can be challenging due to the complex combination of several basic molecules, resulting in a high sensitivity to noise and a possible low signal-to-noise ratio (see [23]). A number of preprocessing steps have been proposed in the literature to address this challenge (see [23]). Although certain noise-related issues, such as outliers and spikes, have been effectively addressed, the resulting data may still be affected by uncertainties, hindering the performance of data-driven methods. For this reason, it is crucial to employ ML models able to protect the classification process against such perturbations. In the mathematical programming literature, various techniques have been developed to address the problem of uncertainty affecting ML methods. Among these, *Robust Optimization* (RO) is widely recognized as one of the main paradigms (see [24,25]). RO assumes that all potential realizations of the uncertain parameters fall within a predefined uncertainty set. The corresponding robust model is then derived by optimizing against the worst-case realizations of the parameters across the entire uncertainty set (see [26]). The application of robust optimization techniques generally leads to improved predictive performance of the ML methods (see [27,28]).

To this aim, in this work, we investigate the performance of a novel robust SVM formulation, introduced in [22], in the context of Raman spectra classification. Specifically, to account for the perturbed nature of Raman data, we start by constructing uncertainty sets around each training observation. These uncertainty sets capture potential measurement errors and variability arising during the spectra acquisition process. Then, we solve the corresponding robust SVM model to ensure that the resulting classifier remains effective under all possible realizations of the data within the uncertainty sets. In terms of numerical experiments, we conduct a comparative analysis between the robust approach and the standard SVM formulation. We address both binary and multiclass classification tasks aimed at diagnosing COVID-19 from real-world Raman saliva samples. Our computational study shows that the robust SVM model exhibits superior performance compared to the standard SVM approach in the majority of investigated conditions, making it a suitable candidate for Raman spectroscopy classification under uncertainty.

The remainder of the paper is organized as follows. Section 2 reviews the existing literature on the problem. In Section 3, the mathematical models and their robust counterpart are presented. Section 4 describes data collection and reports the experimental study. Finally, Section 5 concludes the paper and discusses future works.

2. Related works

Currently, the automatic classification of spectral data is predominantly performed using ML models, with a considerable proportion of these being linear models. Considering the high dimensionality of spectral data, these methods are often combined with feature reduction strategies, such as *Principal Component Analysis* (PCA, see [29]). In recent years, some works also explored the possibility of classifying Raman spectra with DL and neural network algorithms (see [30]). However, the collection of spectral datasets is a time-consuming and costly process. Given the considerable data requirements for the effective training of a Deep Neural Network, the use of ML models remains the predominant approach.

According to the recent literature (see [30]), three main fields of application are identified as the most common in combining machine learning and spectroscopy analysis. The first is the food industry, with ML methods used to detect fraud and identify product alterations (see [31]). The second is forensic science, in which ML techniques are employed to identify illicit drugs (see [32]) or analyse criminal scenes (see [33]). The third is medicine and healthcare, where ML algorithms are applied to recognize bacteria and viruses, or to support automatic diagnosis. Given the focus of this work, the remainder of this section will concentrate on the healthcare domain, with particular emphasis on the development of automated techniques for the diseases' diagnosis.

In the field of healthcare, linear models remain the predominant approach to tackle supervised learning tasks particularly classification problems through LDA (see [34,35]) and SVMs (see [36–38]). In contrast, relatively few studies have employed unsupervised algorithms, such as clustering (see [39]) or k-Nearest Neighbours (see [40]), and only recently have neural networks begun to be explored (see [6,41]). Machine learning methods are often combined with feature reduction techniques to manage the highly dimensionality and noise that characterize spectral data. In this context, PCA is the most commonly adopted method. Among the many applications of Raman spectroscopy combined with machine learning, cancer detection is one of the most promising. In recent years, the use of Raman data combined with multivariate analysis has been investigated for the diagnosis of various types of cancers, including liver (see [42]) and thyroid cancer (see [36]). Other examples include the identification of bladder and breast cancer using SVMs (see [3,43]), LDA (see [44]) and deep learning models (see [45,46]). The detection of neurodegenerative diseases, such as Alzheimer's, Parkinson's, and amyotrophic lateral sclerosis has been explored too by means of SVMs and tree-based ensemble methods (see [5,6]). Following the global diffusion of the novel coronavirus, numerous studies have investigated the potential for automated identification of infected individuals through the integration of Raman spectra and machine learning algorithms. These efforts often leverage spectral data extracted from biological samples, including saliva and blood. Two illustrative examples are [2], where a *Light Gradient Boosting Machine* is trained to classify blood serum spectra, and [47] which employs SVM to analyse spectra derived from saliva samples.

All the ML approaches discussed so far implicitly rely on input data being precisely known at the time of classification. However, this assumption is often unrealistic with real-world observations, especially when dealing with data coming from Raman spectroscopy or saliva-based measurements. These are frequently plagued by noise and perturbations, resulting in worsening performances of the classification process. To address the problem of uncertainty in training samples, robust optimization techniques have been developed within the machine learning literature to prevent the worsening of the solution quality (see [48]). Robust formulations of standard classification methods including logistic regression, SVM and decision trees are discussed in [26]. RO techniques have been also applied to other variants of the classical SVM model. The robust counterpart of the linear approach presented in [21] is extended in [28], introducing a novel *Distributionally Robust Optimization* formulation with moment-based ambiguity sets. An application of such robust and distributionally robust SVM methodology for COVID-19 patient classification is presented in [49]. In [22] the robust extension of the approach designed in [21] is developed employing kernel-induced decision boundaries and bounded-by-ℓ_p-norm uncertainty sets. The performance of the approach is tested on a real-world vehicle emissions task (see [50]). In [51] a robust version of the *TWin Support Vector Machine* (TWSVM, see [18]) classifier is proposed, incorporating uncertainty in the variance matrices of the two classes. The robust extension of TWSVM, formulated as a *Second Order Cone Programming* model (SOCP), is presented in [52]. Additionally, the recent work of [53] introduces a robust and multiclass extension of the *Twin Parametric Margin Support Vector Machine* (TPMSVM, see [19]), with an application in the field of sustainability (see [54]). Finally, [55–57] explore the integration of *Chance-Constrained Programming* (CCP) and distributionally robust optimization techniques into linear and nonlinear SVM models, respectively, accounting for uncertain data.

Table 1
Examples of kernel functions typically used to train SVM models.

Kernel function	$k(x, x')$	Parameters
Homogeneous polynomial	$\langle x, x' \rangle^d$	$d \in \mathbb{N}$
Inhomogeneous polynomial	$(\langle x, x' \rangle + c)^d$	$c \in \mathbb{R}^+, d \in \mathbb{N}$
Gaussian	$\exp(-\frac{\|x-x'\|_2^2}{2\alpha^2})$	$\alpha \in \mathbb{R}_0^+$

Previous research has explored the integration of machine learning techniques, particularly classical SVM approaches, with spectral data for diagnostic purposes. However, these studies have not addressed the impact of perturbations and noise that are inherently present in saliva samples. This variability poses significant challenges for traditional ML models in dealing with such type of data. In this work, we evaluate the performance of a robust SVM formulation, originally proposed in [22], for processing saliva-based Raman spectra under uncertainty. The core novelty of this study lies in its ability to explicitly model and manage input perturbations during the training phase, making the classification process more resilient to real-world measurement noise. To the best of our knowledge, this contribution represents the first application of a robust SVM model that explicitly incorporates uncertainty into the learning process of Raman spectroscopy saliva-based data for diagnostic purposes.

3. Mathematical models

In this section, we describe the SVM model proposed in [22] and based on the works in [17,21] for addressing classification problems with nonlinear decision boundaries. We start by examining the deterministic formulations for binary and multiclass classification tasks (see Section 3.1). Next, we consider the robust counterpart extension in the context of bounded-by-ℓ_p-norm uncertainty sets (see Section 3.2).

3.1. Deterministic formulation

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 the case of spectral data, $x^{(i)}$ corresponds to the Raman spectrum of the ith patient, while $y^{(i)}$ represents the diagnostic outcome (e.g., positive or negative).

The aim of the model proposed in [17] is to find the best separating hypersurface as solution of the following ℓ_1-SVM formulation:

$$\begin{aligned} \min_{u,\gamma,\xi} \quad & \|u\|_1 + \nu \sum_{i=1}^m \xi_i \\ \text{s.t.} \quad & y^{(i)} \left(\sum_{j=1}^m K_{ij} y^{(j)} u_j - \gamma \right) \geq 1 - \xi_i \qquad i = 1, \dots, m \\ & \xi_i \geq 0 \qquad\qquad\qquad\qquad i = 1, \dots, m, \end{aligned} \tag{1}$$

where ν is a positive parameter balancing the terms in the objective function, $K_{ij} := k(x^{(i)}, x^{(j)})$ is the Gram matrix defined according to kernel function $k : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ (see Table 1), and $\xi \in \mathbb{R}^m$ is a slack vector. The kernel function $k(\cdot, \cdot)$ is associated with a feature map $\phi : \mathbb{R}^n \to \mathcal{H}$ that projects training data from the *input space* $\mathbb{R}^n$ to a higher-dimensional space $\mathcal{H}$, called *feature space*, and equipped with the norm $\|\cdot\|_{\mathcal{H}}$. For a comprehensive overview on kernel functions applied to ML methods, and specifically to SVM models, the reader is referred to [58,59].

Once u, γ, ξ are obtained as solutions of (1), an initial nonlinear decision boundary $S_0 := (u, \gamma)$ is defined according to the following equation:

$$\sum_{i=1}^m k(x, x^{(i)}) y^{(i)} u_i = \gamma. \tag{2}$$

Similarly to [21], for each class the greatest misclassification error is computed through formulas:

$$\omega_1 := \max_{i=1,\dots,m} (D\xi)_i \qquad \omega_{-1} := \max_{i=1,\dots,m} (-D\xi)_i, \tag{3}$$

where D is a diagonal matrix with entries $D_{ii} := y^{(i)}$, for all $i = 1, \dots, m$. The values ω_1 and ω_{-1} are used to shift the initial hypersurface S_0, leading to $S_1 := (u, \gamma - 1 + \omega_1)$ and $S_{-1} := (u, \gamma + 1 - \omega_{-1})$ defined as follows:

$$\begin{aligned} S_1 &: \sum_{i=1}^m k(x, x^{(i)}) y^{(i)} u_i = \gamma - 1 + \omega_1 \\ S_{-1} &: \sum_{i=1}^m k(x, x^{(i)}) y^{(i)} u_i = \gamma + 1 - \omega_{-1}. \end{aligned} \tag{4}$$

Finally, the optimal kernel-induced decision boundary $S := (u, b)$ lies in the region between S_1 and S_{-1}, being b the solution of the following model:

$$\begin{aligned} \min_b \quad & \sum_{i=1}^m \mathbb{1} \left(y^{(i)} b - y^{(i)} \sum_{j=1}^m K_{ij} y^{(j)} u_j \right) \\ \text{s.t.} \quad & \gamma + 1 - \omega_{-1} \leq b \leq \gamma - 1 + \omega_1, \end{aligned} \tag{5}$$

where $\mathbb{1}(\cdot) : \mathbb{R} \to \{0, 1\}$ is the indicator function. From a computational standpoint, the solution of model (5) is obtained via a linear search procedure. In particular, the interval $[\gamma + 1 - \omega_{-1}, \gamma - 1 + \omega_1]$ is partitioned into $N_{\max}$ equally spaced sub-intervals, and the objective function is evaluated on each of them. The optimal solution b corresponds to the one yielding the minimum value of the objective function across all sub-intervals. Finally, every new observation $x \in \mathbb{R}^n$ is classified according to the decision function $\mathbb{1}\left(\sum_{i=1}^m k(x, x^{(i)}) y^{(i)} u_i - b\right)$.

As an example, in Fig. 1(a), we show the optimal decision boundaries resulting from the application of the considered SVM methodology to a two-dimensional toy problem. In model (1), we set $\nu = 1$, and consider inhomogeneous quadratic kernel, with $d = 2$ and $c = 0.3$ (see Table 1).

In the case of multiclass classification tasks, a *one-versus-all* approach is considered, classifying training data points of each class against all the other classes. Formally, let $y^{(i)} \in \{1, \dots, L\}$ be the label of the ith observations, with L the number of classes. For each class $l = 1, \dots, L$, an initial separating hypersurface $S_{l,0} := (u_l, \gamma_l)$ is constructed as in (2), where $u_l \in \mathbb{R}^m$ and $\gamma_l \in \mathbb{R}$ are the solutions of the following multiclass version of model (1):

$$\begin{aligned} \min_{u_l,\gamma_l,\xi_l} \quad & \|u_l\|_1 + \nu \sum_{i=1}^m \xi_{l,i} \\ \text{s.t.} \quad & \hat{y}_l^{(i)} \left(\sum_{j=1}^m K_{ij} \hat{y}_l^{(j)} u_{l,j} - \gamma_l \right) \geq 1 - \xi_{l,i} \qquad i = 1, \dots, m \\ & \xi_{l,i} \geq 0 \qquad\qquad\qquad\qquad i = 1, \dots, m, \end{aligned} \tag{6}$$

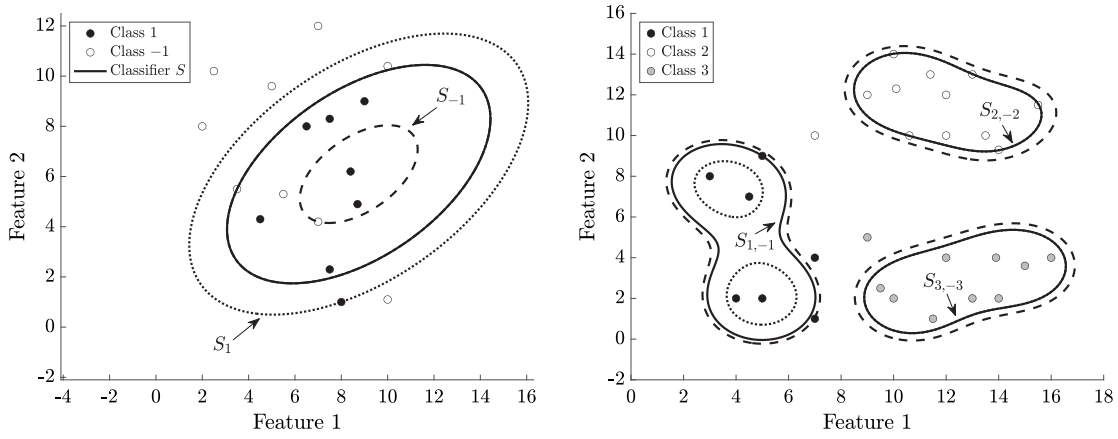
with $\hat{y}_l^{(i)} = 1$ if $y^{(i)} = l$, and $\hat{y}_l^{(i)} = -1$ otherwise. Then, the diagonal matrix $\hat{D}_l$, with $\hat{D}_{l,ii} := \hat{y}_l^{(i)}$, $i = 1, \dots, m$, is constructed and the multiclass equivalent formulas of (3) are computed as follows:

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

Hypersurface $S_{l,0}$ is then shifted to get $S_l := (u_l, \gamma_l - 1 + \omega_l)$ and $S_{-l} := (u_l, \gamma_l + 1 - \omega_{-l})$ (see Eq. (4)). Finally, the optimal decision boundary for class l versus all the others is $S_{l,-l} := (u_l, b_l)$, being b_l the solution of the following model:

$$\begin{aligned} \min_{b_l} \quad & \sum_{i=1}^m \mathbb{1} \left(\hat{y}_l^{(i)} b_l - \hat{y}_l^{(i)} \sum_{j=1}^m K_{ij} \hat{y}_l^{(j)} u_{l,j} \right) \\ \text{s.t.} \quad & \gamma_l + 1 - \omega_{-l} \leq b_l \leq \gamma_l - 1 + \omega_l. \end{aligned} \tag{7}$$

The decision function of the lth class, with $l = 1, \dots, L$, 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 $l^* := \arg\max_{l=1,\dots,L} f_l(x)$ (see [60]).



(a) Binary classification with inhomogeneous quadratic kernel.

(b) Multiclass classification with Gaussian kernel.

Fig. 1. Optimal decision boundaries obtained with inhomogeneous quadratic kernel ($d = 2$, $c = 0.3$) for binary classification (left panel) and Gaussian kernel ($\alpha = 1.9$) for multiclass classification (right panel). For each class $l = 1, 2, 3$, the dotted line and the dashed line represent respectively S_l and S_{-l}.

We represent in Fig. 1(b) the results of the approach for a multiclass classification problem involving three distinct classes. We consider $\nu = 1$ in model (6) and Gaussian kernel with parameter $\alpha = 1.9$ (see Table 1).

3.2. Robust formulation

In this section, we discuss the robust counterpart of the deterministic approaches discussed so far and derived in [22]. According to the robust optimization framework, we assume that input data are plagued by unknown perturbations and construct an uncertainty set around each observation. The best solution is the one optimizing against the worst-case realization across the entire uncertainty set (see [26]).

Formally, let each observation $x^{(i)}$ in the input space $\mathbb{R}^n$ be 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 $\eta^{(i)}$. As a result, the uncertainty set around $x^{(i)}$ can be written as follows:

$$\mathcal{U}_p(x^{(i)}) := \left\{ x \in \mathbb{R}^n : x = x^{(i)} + \sigma^{(i)}, \|\sigma^{(i)}\|_p \le \eta^{(i)} \right\}. \tag{8}$$

The parameter $\eta^{(i)}$ regulates 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 $x^{(i)}$. Common choices for the ℓ_p-norm in the robust optimization literature include $p = 1, 2, \infty$, leading to polyhedral, spherical and box uncertainty sets, respectively.

In the context of Raman spectroscopy, the considered perturbations $\sigma^{(i)}$ are associated with measurement errors and variability introduced during the data acquisition process. These issues arise from instrument limitations, sample inconsistencies, or casual fluctuations, and can significantly affect the quality of the spectra. Such distortions make the input data uncertain, which in turn can compromise the reliability of the predictive SVM model. Accounting for these perturbations is crucial to protect the classification process from adverse variations, especially in sensitive diagnostic applications where small spectral differences are critical.

To incorporate uncertainty within the feature space $\mathcal{H}$, we assume that the uncertainty set around the projected data $\phi(x^{(i)})$ is modelled as:

$$\mathcal{U}_{\mathcal{H}}\left(\phi(x^{(i)})\right) := \left\{ z \in \mathcal{H} : z = \phi(x^{(i)}) + \zeta^{(i)}, \|\zeta^{(i)}\|_{\mathcal{H}} \le \delta^{(i)} \right\}, \tag{9}$$

where the perturbation $\zeta^{(i)}$ belongs to $\mathcal{H}$ 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: if no uncertainty occurs in the input space ($\eta^{(i)} = 0$), no uncertainty will occur in the feature space too ($\delta^{(i)} = 0$). The relation between $\delta^{(i)}$ and $\eta^{(i)}$ has been explored in [22] where a closed-form expression of $\delta^{(i)}$ has been derived as function of $\eta^{(i)}$ for typically used kernel functions.

Once the uncertainty sets (8)–(9) have been constructed, it is possible to derive the robust counterparts of models (1) and (6). Specifically, for binary classification tasks, the robust model is given by:

$$\begin{aligned} \min_{u,\gamma,\xi} \quad & \|u\|_1 + \nu \sum_{i=1}^{m} \xi_i \\ \text{s.t.} \quad & y^{(i)} \sum_{j=1}^{m} K_{ij} y^{(j)} u_j - \delta^{(i)} \sum_{j=1}^{m} \sqrt{K_{jj}} \left|u_j\right| \ge 1 - \xi_i + y^{(i)}\gamma \qquad i = 1, \dots, m \\ & \xi_i \ge 0 \qquad\qquad i = 1, \dots, m. \end{aligned} \tag{10}$$

Similar to the deterministic framework, once u, γ and ξ are determined as solutions of (10), then ω_1 and ω_{-1} are calculated using the expressions in (3). Finally, the optimal separating hypersurface $S = (u, b)$ is obtained, where b is the optimal solution of the following robust version of model (5):

$$\begin{aligned} \min_{b} \quad & \sum_{i=1}^{m} \mathbb{1}\left[\left(y^{(i)} b - y^{(i)} \sum_{j=1}^{m} K_{ij} y^{(j)} u_j + \delta^{(i)} \sum_{j=1}^{m} \sqrt{K_{jj}} \left|u_j\right| \right)_i \right] \\ \text{s.t.} \quad & \gamma + 1 - \omega_{-1} \le b \le \gamma - 1 + \omega_1. \end{aligned} \tag{11}$$

When addressing a multiclass classification problem, the robust extension of model (6) for the lth class, with $l = 1, \dots, L$, is expressed as follows:

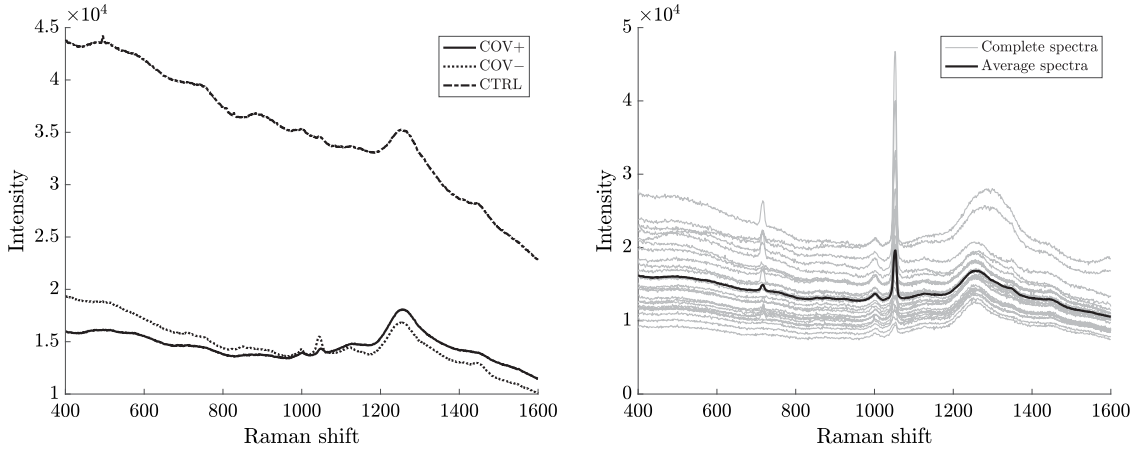
$$\begin{aligned} \min_{u_l,\gamma_l,\xi_l} \quad & \|u_l\|_1 + \nu \sum_{i=1}^{m} \xi_{l,i} \\ \text{s.t.} \quad & \hat{y}_l^{(i)} \sum_{j=1}^{m} K_{ij} \hat{y}_l^{(j)} u_{l,j} - \delta^{(i)} \sum_{j=1}^{m} \sqrt{K_{jj}} \left|u_{l,j}\right| \ge 1 - \xi_{l,i} + \hat{y}_l^{(i)}\gamma_l \qquad i = 1, \dots, m \\ & \xi_{l,i} \ge 0 \qquad\qquad i = 1, \dots, m. \end{aligned} \tag{12}$$

The optimal parameter b_l of the kernel-induced decision boundary $S_{l,-l} := (u_l, b_l)$ is the solution of:

$$\begin{aligned} \min_{b_l} \quad & \sum_{i=1}^{m} \mathbb{1}\left[\left(\hat{y}_l^{(i)} b_l - \hat{y}_l^{(i)} \sum_{j=1}^{m} K_{ij} \hat{y}_l^{(j)} u_{l,j} + \delta^{(i)} \sum_{j=1}^{m} \sqrt{K_{jj}} \left|u_{l,j}\right| \right)_i \right] \\ \text{s.t.} \quad & \gamma_l + 1 - \omega_{-l} \le b_l \le \gamma_l - 1 + \omega_l. \end{aligned} \tag{13}$$

4. Computational experiments

In this section, we discuss the performance of the deterministic models presented in Section 3.1 and their robust counterparts of Section 3.2,



(a) Patient-average spectra of COV+, COV− and CTRL patients.

(b) Single and patient-average spectra set of a COV+ patient.

Fig. 2. Examples of saliva Raman spectra. Left panel: average spectra of three patients (COV+, COV− and CTRL). Right panel: single and average spectra of a COV+ patient.

evaluated on Raman-based COVID-19 datasets. We start by describing the data collection process, outlining the main characteristics of the datasets, and detailing the preprocessing techniques adopted (see Section 4.1). Then, we present and analyse the results of the numerical experiments using classical statistical indicators (see Section 4.2).

All the models were implemented in MATLAB (v. 2021b) and solved using CVX (v. 2.2, see [61,62]) with the MOSEK solver (v. 9.1.9, see [63]). All computational experiments were run on an AMD EPYC 7302 Processor with 16-Core and 512 GB of RAM memory. Unless otherwise specified, a runtime limit of 48 h (172800 s) was imposed. For models (5), (7), (11), and (13), the maximum number of subdivisions $N_{\max}$ was set to 100. In previous works (see [22,28]), larger values of $N_{\max}$ were considered. However, the computational complexity analysis provided in [22] shows that high values of $N_{\max}$ can significantly increase the CPU time, particularly in multiclass settings. In this work, we chose to reduce $N_{\max}$ to balance computational efficiency with solution quality. This choice is especially relevant in diagnostic applications, where achieving reliable results within a relatively short time is essential.

The MATLAB code and data used in this study are made publicly available on GitHub (see https://github.com/piazzam/Robust-SVM-Raman).

4.1. Dataset description

Saliva samples, health records, and clinical data were acquired at *IRCCS Fondazione Don Carlo Gnocchi ONLUS*, and *Santa Maria Nascente Hospital* in Milano (Italy), and at *Centro Spalenza Hospital* in Rovato (Italy), between April 2020 to July 2020. The COVID-19 diagnosis was conducted following the *World Health Organization* guidelines, declaring a positive case after the positive result of sequencing or *Real-Time reverse-transcription Polymerase Chain Reaction* assay of SARS-CoV-2 for nasopharyngeal swabs. The patients were considered COVID-19 negativized after two consecutive tests with negative results.

The total number of subjects involved in the study was 101, composed as follows: 30 patients affected by COVID-19 (COV+), 37 subjects negative to the SARS-CoV-2 test with an ascertained episode of COVID-19 (COV−), and 34 age and sex correlated healthy subjects (CTRL). More information regarding the acquisition protocol and the patients selections are available in [41].

Before applying the machine learning methods to the considered dataset, we carried out the following cleaning and preprocessing steps (see [23,64]):

- *Outlier removal*: a spectrum is considered as an outlier when it encounters issues during the acquisition phase that lead to a low signal-to-noise ratio (see [23]). To ensure the integrity of the dataset, we excluded spectra that contained more than 10% of values equal to zero or sequences of repeated values exceeding a saturation limit, defined as the maximum value that can be found within the single spectrum;
- *Spike removal*: cosmic rays may negatively influence the measuring process, producing anomalous peaks in a spectrum. To remove them we exploited the Whitaker-Hayes algorithm (see [65]), where the series of subsequent differences in a spectrum is taken into account to highlight and remove peak anomalies;
- *Realignment of the Raman shift axis*: the acquisition of Raman spectra may occur at different times and under varying conditions, potentially resulting in slight wavenumber shifts that misalign the Raman peaks. This issue can be addressed by interpolating the spectral data onto a common fixed grid of x-axis points;
- *Removal of the background noise*: the acquisition of Raman spectra is negatively affected by background noise from fluorescence generated by molecules excited by the laser, which compromises the signal-to-noise ratio. Since this effect introduces wavenumber shifts that do not directly relate to the specific compound under investigation, it is usually recommended to remove them. Following recent spectroscopy literature (see [23]) we employed polynomial fitting for this purpose (see [66]);
- *Intensity normalization*: normalization techniques are employed to ensure consistent comparisons between Raman spectra collected under different conditions. Furthermore, since models (1)–(6) and their robust counterparts (10)–(12) are distance-based, imbalances in the magnitude of the features can lead to distorted classifiers. To mitigate these issues, in this study, we implemented the *Standard Normal Variate Normalization* (see [67]), a widely used techniques in the field of spectroscopy;
- *Principal Component Analysis*: for each of the 101 patients, approximately 25 salivary samples were acquired consisting in more than 900 components. Best practices often suggest avoiding the direct computation on high-dimensional raw data in ML applications (see [67]). For this reason, the most informative 15 features were extracted through a PCA method.

After cleaning and preprocessing the original dataset according to the previous steps, we also constructed a reduced dataset consisting of the average spectra for each patient. Fig. 2(a) presents an example of

Table 2
Out-of-sample accuracy and standard deviation for deterministic models based on grid search. Best results are highlighted.

(a) Single-spectra COVID-19 dataset.

Classification task		Kernel						
		Hom. linear	Hom. quadratic	Hom. cubic	Inhom. linear	Inhom. quadratic	Inhom. cubic	Gaussian
COV+ vs COV−	Accuracy (%)	74.27 ± 34.79	**86.69 ± 23.21**	69.93 ± 27.80	74.27 ± 34.79	86.43 ± 22.91	79.04 ± 23.12	81.04 ± 28.25
	CPU time (s)	2887	3190	25 171	3768	3279	7599	3032
COV+ vs CTRL	Accuracy (%)	**80.96 ± 32.40**	67.40 ± 29.85	67.31 ± 30.99	**80.96 ± 32.40**	69.90 ± 34.12	69.50 ± 30.60	78.74 ± 33.96
	CPU time (s)	2792	6887	24 499	3082	3644	14 056	2933
COV− vs CTRL	Accuracy (%)	82.38 ± 29.73	88.25 ± 18.46	86.18 ± 19.41	82.38 ± 29.95	86.89 ± 22.28	86.25 ± 20.08	**89.31 ± 20.93**
	CPU time (s)	12 449	5858	7352	4310	3727	8127	6618
(COV+ ∪ COV−) vs CTRL	Accuracy (%)	**82.32 ± 29.51**	77.12 ± 29.19	74.40 ± 27.69	**82.32 ± 29.51**	80.56 ± 28.59	77.16 ± 27.33	66.34 ± 47.49
	CPU time (s)	2576	3612	11 056	2651	4009	13 771	2410
COV+ vs COV− vs CTRL	Accuracy (%)	69.90 ± 39.11	71.55 ± 31.89	62.30 ± 30.57	69.90 ± 39.11	**75.38 ± 31.98**	68.69 ± 29.22	73.63 ± 33.25
	CPU time (s)	20 189	20 284	183 159	20 365	17 461	271 198	12 835

(b) Patient-average COVID-19 dataset.

Classification task		Kernel						
		Hom. linear	Hom. quadratic	Hom. cubic	Inhom. linear	Inhom. quadratic	Inhom. cubic	Gaussian
COV+ vs COV−	Accuracy (%)	**77.61 ± 42.00**	65.67 ± 47.84	65.67 ± 47.84	**77.61 ± 42.00**	67.16 ± 47.32	70.15 ± 46.11	44.78 ± 50.10
	CPU time (s)	15	15	16	15	15	17	15
COV+ vs CTRL	Accuracy (%)	81.25 ± 39.34	78.12 ± 41.67	76.56 ± 42.70	81.25 ± 39.34	75.00 ± 43.64	**82.81 ± 38.03**	46.88 ± 50.30
	CPU time (s)	15	15	15	15	14	15	15
COV− vs CTRL	Accuracy (%)	**85.92 ± 35.03**	80.28 ± 40.07	83.10 ± 37.74	**85.92 ± 35.03**	83.10 ± 37.74	78.87 ± 41.11	52.11 ± 50.31
	CPU time (s)	16	16	17	16	16	17	16
COV+ vs COV− vs CTRL	Accuracy (%)	**71.29 ± 45.47**	70.30 ± 45.92	65.35 ± 47.82	**71.29 ± 45.47**	68.32 ± 46.76	69.31 ± 46.35	61.39 ± 48.93
	CPU time (s)	68	67	70	66	68	70	65

three average Raman spectra, each corresponding to a specific group of patients (COV+, COV−, CTRL). In Fig. 2(b) both the single and patient-average spectra for a COV+ patient are shown.

Summarizing, in this study we consider the following two datasets:

(a) Single-spectra COVID-19 dataset: $n = 15$ features, $m = 2409$ observations;
(b) Patient-average spectrum COVID-19 dataset: $n = 15$ features, $m = 101$ observations.

4.2. Model validation

The experimental setting is structured as follows. Each dataset was divided into training set and testing set using a *Leave One Patient Out-Cross Validation* (LOPO-CV) strategy. Specifically, in this study, the LOPO-CV corresponds to a 101-fold cross validation procedure, where, in each iteration, all the patients except one were assigned to the training set. Once the classifier was determined on the training set, its performance was assessed on the unique patient in the testing set. This process was then repeated for all patients.

Regarding the kernel function $k(\cdot,\cdot)$, seven different alternatives were explored: 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. Following the approach in [22], the parameters α and c (see Table 1) were set equal to the maximum value of the standard deviation across features for the dataset under consideration.

Since the datasets contain three classes, we performed four binary classification tasks (COV+ vs COV−; COV+ vs CTRL; COV− vs CTRL; (COV+ ∪ COV −) vs CTRL), and a multiclass task (COV+ vs COV− vs CTRL).

To measure the quality of the solution, we considered various statistical indicators, depending on the nature of the classification problem. Specifically, in the case of binary classification, let TP_s, TN_s, FP_s, FN_s be the number of True Positive, True Negative, False Positive and False Negative, respectively, identified by the optimal classifier in fold s. Thus, for each fold $s = 1, \dots, 101$, we computed the following indicators:

$$\text{Accuracy for fold } s := A_s = \frac{TP_s + TN_s}{TP_s + TN_s + FP_s + FN_s}$$

$$\text{Precision for fold } s := P_s = \frac{TP_s}{TP_s + FP_s}$$

$$\text{Sensitivity for fold } s := SN_s = \frac{TP_s}{TP_s + FN_s}$$

$$\text{Specificity for fold } s := SP_s = \frac{TN_s}{TN_s + FP_s}$$

$$\text{Matthews Correlation Coefficient for fold } s := MCC_s = \frac{TP_s \cdot TN_s - FP_s \cdot FN_s}{\sqrt{(TP_s + FP_s)(TP_s + FN_s)(TN_s + FP_s)(TN_s + FN_s)}}.$$

Finally, the results were averaged, leading to:

$$\text{Accuracy} := \frac{1}{101}\sum_{s=1}^{101} A_s \qquad \text{Precision} := \frac{1}{101}\sum_{s=1}^{101} P_s$$

$$\text{Sensitivity} := \frac{1}{101}\sum_{s=1}^{101} SN_s$$

$$\text{Specificity} := \frac{1}{101}\sum_{s=1}^{101} SP_s \qquad \text{MCC} := \frac{1}{101}\sum_{s=1}^{101} MCC_s.$$

Concerning the multiclass classification task, for each fold s let $C_s^{\hat{l},l}$ be the number of observations in class l and classified in class $\hat{l}$, with $l, \hat{l} \in \{1,2,3\} = \{\text{COV+},\text{COV−},\text{CTRL}\}$. If $l = \hat{l}$, then the observations are correctly classified, otherwise they are misclassified. Hence, for each

Table 3
Out-of-sample statistical indicators for deterministic models based on grid search. Best results are highlighted.

(a) Single-spectra COVID-19 dataset.

Classification task		Kernel						
		Hom. linear	Hom. quadratic	Hom. cubic	Inhom. linear	Inhom. quadratic	Inhom. cubic	Gaussian
COV+ vs COV−	Precision (%)	76.13	**87.21**	–	76.13	87.05	82.19	79.52
	Sensitivity (%)	55.67	55.11	53.29	55.67	55.02	53.28	**58.17**
	Specificity (%)	44.33	44.89	46.71	44.33	44.98	**46.72**	41.83
	MCC	0.49	**0.73**	0.40	0.49	**0.73**	0.58	0.62
COV+ vs CTRL	Precision (%)	**80.73**	66.91	66.04	**80.73**	68.00	69.29	80.08
	Sensitivity (%)	52.10	54.09	55.10	52.10	**56.03**	52.56	50.46
	Specificity (%)	47.90	45.91	44.90	47.90	43.97	47.44	**49.54**
	MCC	**0.62**	0.34	0.33	**0.62**	0.39	0.38	0.57
COV− vs CTRL	Precision (%)	78.97	87.23	86.65	78.97	86.95	86.89	**87.47**
	Sensitivity (%)	**49.05**	47.07	45.51	**49.05**	46.15	45.75	47.60
	Specificity (%)	50.95	52.93	**54.49**	50.95	53.85	54.25	52.40
	MCC	0.65	0.76	0.72	0.65	0.74	0.72	**0.78**
(COV+ ∪ COV−) vs CTRL	Precision (%)	82.60	83.87	84.66	82.60	**86.50**	85.84	67.40
	Sensitivity (%)	72.02	71.67	68.39	72.02	70.47	69.16	**100.00**
	Specificity (%)	27.98	28.33	**31.61**	27.98	29.53	30.84	0.00
	MCC	**0.60**	0.49	0.45	**0.60**	0.56	0.5	–
COV+ vs COV− vs CTRL	Sensitivity COV+ (%)	59.84	59.04	47.93	59.84	**64.26**	55.02	64.12
	Sensitivity COV− (%)	71.72	85.97	67.73	71.72	**86.43**	77.65	80.39
	Sensitivity CTRL (%)	**77.58**	67.90	69.30	**77.58**	72.87	71.21	74.52
	Specificity COV+ (%)	83.93	85.93	78.61	83.93	**87.00**	83.97	84.47
	Specificity COV− (%)	84.99	92.62	87.34	84.99	**93.47**	87.73	86.49
	Specificity CTRL (%)	86.15	80.95	78.54	86.15	83.84	82.25	**89.75**

(b) Patient-average COVID-19 dataset.

Classification task		Kernel						
		Hom. linear	Hom. quadratic	Hom. cubic	Inhom. linear	Inhom. quadratic	Inhom. cubic	Gaussian
COV+ vs COV−	Precision (%)	**74.19**	65.22	62.64	**74.19**	66.67	67.86	44.78
	Sensitivity (%)	44.23	34.09	38.64	44.23	35.56	40.43	**100.00**
	Specificity (%)	55.77	**65.91**	61.36	55.77	64.44	59.57	0.00
	MCC	**0.55**	0.30	0.30	**0.55**	0.33	0.39	–
COV+ vs CTRL	Precision (%)	76.47	83.33	77.78	76.47	81.82	**91.30**	46.88
	Sensitivity (%)	50.00	40.00	42.86	50.00	37.50	39.62	**100.00**
	Specificity (%)	50.00	60.00	57.14	50.00	**62.50**	60.38	0.00
	MCC	0.63	0.57	0.53	0.63	0.51	**0.67**	–
COV− vs CTRL	Precision (%)	86.49	89.66	87.88	86.49	**96.30**	82.35	52.11
	Sensitivity (%)	52.46	45.61	49.15	52.46	44.07	50.00	**100.00**
	Specificity (%)	47.54	54.39	50.85	47.54	**55.93**	50.00	0.00
	MCC	**0.72**	0.62	0.67	**0.72**	0.69	0.58	–
COV+ vs COV− vs CTRL	Sensitivity COV+ (%)	63.33	**73.33**	56.67	63.33	53.33	63.33	20.00
	Sensitivity COV− (%)	**75.68**	64.86	67.57	**75.68**	67.57	70.27	**75.68**
	Sensitivity CTRL (%)	73.53	73.53	70.59	73.53	**82.35**	73.53	**82.35**
	Specificity COV+ (%)	88.87	78.85	78.73	88.87	84.36	81.54	**91.56**
	Specificity COV− (%)	84.38	**89.06**	81.25	84.38	84.38	81.25	70.31
	Specificity CTRL (%)	83.61	87.88	88.50	83.61	83.91	**91.14**	79.64

fold $s = 1, \dots, 101$ we computed:

$$\text{Accuracy for fold } s := \text{AC}_s = \frac{\sum_{l=1}^{3} \text{C}_s^{l,l}}{\sum_{\hat{l},l=1}^{3} \text{C}_s^{\hat{l},l}}$$

$$\text{Sensitivity for class } l \text{ and fold } s := \text{SN}_s^l = \frac{\text{C}_s^{l,l}}{\sum_{\hat{l}=1,\hat{l}\neq l}^{3} \text{C}_s^{\hat{l},l}}$$

$$\text{Specificity for class } l \text{ and fold } s := \text{SP}_s^l = \frac{\sum_{\hat{l}=1,\hat{l}\neq l}^{3} \sum_{\tilde{l}=1,\tilde{l}\neq l}^{3} \text{C}_s^{\hat{l},\tilde{l}}}{\sum_{\hat{l}=1}^{3} \sum_{\tilde{l}=1,\tilde{l}\neq l}^{3} \text{C}_s^{\hat{l},\tilde{l}}}.$$

As in the binary case, we averaged the results as follows:

$$\text{Accuracy} := \frac{1}{101} \sum_{s=1}^{101} \text{AC}_s \qquad \text{Sensitivity for class } l := \frac{1}{101} \sum_{s=1}^{101} \text{SN}_s^l$$

$$\text{Specificity for class } l := \frac{1}{101} \sum_{s=1}^{101} \text{SP}_s^l.$$

In the training phase, we explored two different approaches in treating hyperparameter ν in the objective function of models (1), (6), (10) and (12): a grid search procedure and a Bayesian Optimization algorithm (see [68]).

In the first approach, five logarithmically spaced values between 10^{-3} and 10^{0} were considered (see [22,28]), choosing the best one minimizing the training error. The results of the computations in terms of accuracy are reported in Table 2 and specified on each dataset (single-spectra COVID-19 dataset, see Table 2a; patient-average COVID-19 dataset, see Table 2b).

Table 4
Out-of-sample accuracy and precision comparison between the best results from Tables 2–3 and scikit-learn experiments. Best results are highlighted.

(a) Single-spectra COVID-19 dataset.

Classification task	Accuracy (%)		Precision (%)	
	Table 2a	Scikit-learn library	Table 3a	Scikit-learn library
COV+ vs COV−	**86.69**	81.81	**87.21**	81.82
COV+ vs CTRL	**80.96**	80.35	**80.73**	80.35
COV− vs CTRL	**89.31**	89.29	87.47	**89.09**
(COV+ ∪ COV−) vs CTRL	82.32	**84.36**	**86.50**	82.34
COV+ vs COV− vs CTRL	**75.38**	74.39	–	–

(b) Patient-average COVID-19 dataset.

Classification task	Accuracy (%)		Precision (%)	
	Table 2b	Scikit-learn library	Table 3b	Scikit-learn library
COV+ vs COV−	**77.61**	74.63	74.19	**74.45**
COV+ vs CTRL	82.81	**84.12**	**91.30**	84.12
COV− vs CTRL	85.92	**86.13**	**96.30**	86.13
COV+ vs COV− vs CTRL	71.29	**74.04**	–	–

It can be noted that in both cases, the accuracy of the best classifier exceeds 71%, demonstrating that the considered SVM methodology is generally effective at correctly classifying the samples. However, when moving from dataset (a) (single-spectra COVID-19 dataset) to dataset (b) (patient-average COVID-19 dataset), the results worsen overall, except for the task COV+ vs CTRL, where the inhomogeneous cubic kernel achieves the highest accuracy in dataset (b) at 82.81% (compared to 80.96% in dataset (a) with both the homogeneous and inhomogeneous linear kernels). This general decline in accuracy reflects the reduced informative power of the average dataset compared to the original one. Additionally, in dataset (b) the results tend to fluctuate more, showing larger standard deviations and indicating that the reduced data granularity makes it harder to distinguish between classes, especially for more complex tasks. On the other hand, in the reduced dataset CPU times are generally much lower, with most computations taking around 15–17 s for binary classification tasks and 65–70 s for the multiclass tasks. This shows significant computational efficiency compared to the single-spectra dataset. Therefore, we can conclude that there exists a trade-off between accuracy and performing speed. The final user must balance these factors based on their priority: faster results with lower accuracy (average dataset) or more precise classifications with longer computation times (single-spectra dataset).

A similar drop in performance is confirmed by other statistical indicators (see Table 3), confirming that the reduced dataset leads to a decrease in model reliability. As far as it concerns the kernel functions, we notice that the Gaussian kernel tends to maximize sensitivity, reaching 100% in several cases (see Table 3b), but it suffers from extremely poor specificity, making it unsuitable for balanced tasks and severely limiting its usefulness. On the other hand, homogeneous and inhomogeneous polynomial kernels, especially linear and quadratic, generally achieve the best overall performance across multiple indicators, with particularly strong precision and MCC in most tasks, making them effective in both binary and multiclass problems.

To further support the results of our proposal, in Table 4 we conducted a comparison between the best results of Tables 2–3 and the out-of-sample accuracy and precision provided by *scikit-learn*, a popular ML library implemented in Python (see [69]). The comparison illustrates a clear advantage for the proposed approach across several tasks. For instance, in the case of distinguishing COV− from CTRL in the patient-average dataset, our model achieved a precision of 96.30% compared to scikit-learn performance of 86.13% (see Table 4b), indicating a superior ability to minimize false positives in this classification. Similarly, in the task of identifying (COV+ ∪ COV−) vs CTRL in the single-spectra dataset, our method ensured a precision of 86.50%, outperforming the 82.34% of scikit-learn (see Table 4a). In terms of accuracy, our models consistently delivered better results in most tasks when applied to the single-spectra dataset.

As an alternative strategy for identifying the optimal value of the hyperparameter ν, we adopted the Bayesian optimization approach. We utilized the *bayesopt* library in MATLAB, specifically configured to minimize the training error at each iteration. To ensure consistency with the grid search results, we restricted the search for ν to the interval [0,2]. The numerical experiments were conducted using the best-performing kernel functions identified in the grid search procedure (see Table 2). For the robust SVM formulations (see Section 3.2), we employed a box-type uncertainty set ($p = \infty$ in definition (8)), assuming a constant uncertainty radius $\eta = \eta^{(i)}$ across all observations. The value of η was tuned using Bayesian optimization over the interval $[10^{-7}, 10^{-1}]$ to identify the most robust configuration.

The results of the numerical experiments are reported in Table 5. The robust SVM models provide slight improvements in statistical indicators like accuracy and sensitivity compared to the deterministic models across various classification tasks. However, regarding CPU time, Bayesian optimization techniques require nearly double the time compared to the grid search procedure for ν in the deterministic setting (see Table 2). This computational time is further increased in the robust framework due to the simultaneous tuning of both ν and η. In the multiclass classification task using the single-spectra dataset, the Bayesian optimization method fails to provide results within the considered time limit.

Finally, in Table 6 we summarized and compared the best results in terms of accuracy of this study. In the single-spectra dataset (see Table 6a), the robust models consistently outperformed the deterministic models in the COV+ vs COV− and COV+ vs CTRL tasks. In all the other tasks the best results are in favour of deterministic approaches. On the other hand, in the patient-average dataset (see Table 6b), the results exhibit a different trend. In three out of four cases, the scikit-learn library outperforms the other methods. However, as pointed out, the informative power of this dataset is reduced. These results highlight the trade-offs associated with using robust models and Bayesian optimization techniques. While they can improve accuracy in certain classification tasks, there are situations where deterministic approaches perform better.

As a final remark, it is interesting to note that the robust model performs worse on the patient-average dataset. This can be explained by the fact that averaging the data tends to collapse the spectra into similar intensity values, thus reducing the need for and importance of robustness. However, the single-spectra dataset is more relevant from a medical point of view, as it can better capture specific nuances of the biomarkers in saliva.

Table 5
Out-of-sample statistical indicators for deterministic and robust models based on Bayesian optimization. Asterisk indicates time limit reached. Best results are highlighted.

(a) Single-spectra COVID-19 dataset.

Classification task	Kernel		Deterministic	Robust
COV+ vs COV−	Hom. quadratic	Accuracy (%)	86.69 ± 23.21	**87.00 ± 22.08**
		Precision (%)	**87.21**	86.93
		Sensitivity (%)	55.11	**55.52**
		Specificity (%)	**44.89**	44.48
		MCC	0.73	**0.74**
		CPU time (s)	5568	20456
COV+ vs CTRL	Hom. linear	Accuracy (%)	80.96 ± 32.40	**82.52 ± 32.02**
		Precision (%)	80.50	**82.66**
		Sensitivity (%)	**52.10**	51.70
		Specificity (%)	47.90	**48.30**
		MCC	0.62	**0.65**
		CPU time (s)	5351	13485
COV− vs CTRL	Gaussian	Accuracy (%)	**79.63 ± 28.61**	79.54 ± 29.24
		Precision (%)	**92.45**	92.12
		Sensitivity (%)	**37.05**	36.43
		Specificity (%)	62.95	**63.57**
		MCC	**0.61**	**0.61**
		CPU time (s)	13 143	26616
(COV+ ∪ COV−) vs CTRL	Hom. linear	Accuracy (%)	**82.35 ± 29.23**	82.20 ± 29.60
		Precision (%)	**86.34**	86.05
		Sensitivity (%)	71.88	**72.13**
		Specificity (%)	**28.12**	27.87
		MCC	**0.60**	0.59
		CPU time (s)	13 633	87744
COV+ vs COV− vs CTRL	Inhom. quadratic	Accuracy (%)	75.22 ± 31.66	–
		Sensitivity COV+ (%)	62.25	–
		Sensitivity COV− (%)	87.00	–
		Sensitivity CTRL (%)	73.76	–
		Specificity COV+ (%)	87.88	–
		Specificity COV− (%)	92.75	–
		Specificity CTRL (%)	83.45	–
		CPU time (s)	145 923	172800*

(b) Patient-average COVID-19 dataset.

Classification task	Kernel		Deterministic	Robust
COV+ vs COV−	Hom. linear	Accuracy (%)	74.63 ± 43.84	**76.12 ± 42.96**
		Precision (%)	69.70	**71.88**
		Sensitivity (%)	**46.00**	45.10
		Specificity (%)	54.00	**54.90**
		MCC	0.49	**0.52**
		CPU time (s)	710	755
COV+ vs CTRL	Inhom. cubic	Accuracy (%)	**81.25 ± 39.34**	59.38 ± 49.50
		Precision (%)	**87.50**	56.90
		Sensitivity (%)	40.38	**86.84**
		Specificity (%)	**59.62**	13.16
		MCC	**1.03**	0.23
		CPU time (s)	757	702
COV− vs CTRL	Hom. linear	Accuracy (%)	**85.92 ± 35.03**	83.10 ± 37.74
		Precision (%)	86.49	**89.29**
		Sensitivity (%)	**52.46**	42.37
		Specificity (%)	47.54	**57.63**
		MCC	**1.12**	1.07
		CPU time (s)	736	807
COV+ vs COV− vs CTRL	Hom. linear	Accuracy (%)	**71.29 ± 45.47**	68.32 ± 46.76
		Sensitivity COV+ (%)	**63.33**	53.33
		Sensitivity COV− (%)	**72.97**	**72.97**
		Sensitivity CTRL (%)	**76.47**	**76.47**
		Specificity COV+ (%)	85.82	**85.94**
		Specificity COV− (%)	82.21	**85.94**
		Specificity CTRL (%)	**88.19**	81.27
		CPU time (s)	5504	6677

As noted in the final step of the preprocessing techniques described in Section 4.1, PCA was applied to reduce the high dimensionality of the dataset. All results presented so far are based on classification tasks using the 15 most informative principal components selected from over 900 original features. However, the use of artificially derived features can limit the interpretability of the results, as it becomes difficult to relate the reduced principal components to the original spectral features, making it impossible to explain the final diagnosis in terms of actual Raman spectra. To address this limitation, while also managing the computational complexity associated with the original dataset,

Table 6
Out-of-sample accuracy comparison between the best results from Tables 2, 3, 5, and scikit-learn experiments. Best results are highlighted.

(a) Single-spectra COVID-19 dataset.

Classification task	Deterministic model			Robust model
	Scikit-learn library	Grid-search	Bayesian optimization	Bayesian optimization
COV+ vs COV−	81.81	86.69	86.69	**87.00**
COV+ vs CTRL	80.35	80.86	80.96	**82.52**
COV− vs CTRL	89.29	**89.31**	79.63	79.54
(COV+ ∪ COV−) vs CTRL	**84.36**	82.32	82.35	82.20
COV+ vs COV− vs CTRL	74.39	**75.38**	75.22	–

(b) Patient-average COVID-19 dataset.

Classification task	Deterministic model			Robust model
	Scikit-learn library	Grid-search	Bayesian optimization	Bayesian optimization
COV+ vs COV−	74.63	**77.61**	74.63	76.12
COV+ vs CTRL	**84.12**	82.81	81.25	59.38
COV− vs CTRL	**86.13**	85.92	85.92	83.10
COV+ vs COV− vs CTRL	**74.04**	71.29	71.29	68.32

Table 7
Comparison of out-of-sample accuracy between PCA and the ensemble method using three and nine spectral intervals for dimensionality reduction.

Classification task	Deterministic model			Robust model		
	PCA	Three intervals	Ten intervals	PCA	Three intervals	Ten intervals
COV+ vs COV−	74.63	74.63	71.43	**76.12**	60.32	58.21
COV+ vs CTRL	**84.12**	81.25	77.42	59.38	59.38	67.8
COV− vs CTRL	86.13	85.92	**87.32**	83.10	57.75	56.34

we proposed the following approach. We divided the spectral range (see the x-axis in Fig. 2) into non-overlapping, fixed-size intervals. A separate classifier is trained on each interval using only the features within that range, thus preserving the original spectral information while working with smaller subsets of data. The predictions from these classifiers were then combined using a majority voting strategy to assign the final label. The accuracy of the procedure is finally computed by comparing the predicted label with the true label across all patients.

To evaluate the interval-based approach we considered two possible configurations: using three spectral intervals and nine spectral intervals. A summary of the results obtained on the patient-average spectra with binary tasks is presented in Table 7, together with a comparison with the corresponding PCA-based models. As shown, PCA-based classifiers achieve higher out-of-sample accuracy across all binary classification tasks in two over three cases. However, often, the performance gap between the PCA-based and interval-based models is relatively small or even absent. This is especially evident for the *COV− vs CTRL* task when the spectral range is divided into nine intervals, where the deterministic interval-based model exceeds the PCA-based one.

These findings suggest that while models relying on PCA-transformed features achieve high predictive accuracy, the interval-based ensemble approach offers a promising trade-off between performance and interpretability. By working directly with Raman spectral regions, this method allows for a more transparent connection between diagnostic outcomes and specific Raman features. However, it could be more sensitive to noisy data and outliers when applied on single spectra.

5. Conclusions

In this paper, we presented data-driven optimization models to support medical decision-making in diagnosing COVID-19 through a combination of Raman Spectroscopy and Machine Learning methods. Specifically, we applied a novel approach proposed in [22] to handle Support Vector Machines with nonlinear decision boundaries. To account for uncertainties in saliva Raman spectra, we formulated robust optimization classifiers for both binary and multiclass classification tasks. We conducted numerical experiments based on real-world data on COVID-19 diagnosis provided by Italian hospitals. To evaluate the effectiveness of the proposal, we compared the results with a state-of-the-art classifier in machine learning applications. The experiments highlight a trade-off between computational efficiency and classification accuracy. Additionally, we explored two methods for tuning the hyperparameters of the models: a grid search procedure and a Bayesian Optimization algorithm. The combination of Bayesian optimization and robust Support Vector Machine models led to small but consistent improvements in accuracy.

Overall, this work highlights the potential of machine learning techniques, especially robust Support Vector Machines, in improving disease detection through Raman spectroscopy with noisy and limited data. From a methodological perspective, future research could extend this approach to address uncertainties in the labels of spectral data, enhancing the model's generalization capabilities. Additionally, the potential of combining machine learning and Raman Spectroscopy can be explored for designing a rapid, cost-effective, and non-invasive diagnostic tool. Indeed, with recent advancements in spectroscopy and the development of portable Raman Spectroscopy devices, a point-of-care system could be established for use in any setting, without requiring specific human expertise. Finally, given the critical nature of healthcare applications, incorporating explainability in machine learning methods is essential to make the proposed models more transparent. In this context, we proposed an interval-based approach designed to improve model interpretability, representing a promising direction for future research. Preliminary numerical results indicated that, while models based on PCA-transformed features achieve high predictive accuracy, the interval-based method offers a valuable trade-off between performance and interpretability.

Funding

This work has been supported by Fondazione Regionale per la Ricerca Biomedica, project CORSAI ID 383 - JTC 2021 ERA PerMed, GA 779282 - CUP: H45E22000030006.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Enza Messina reports financial support was provided by Regional Foundation for Biomedical Research. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

MP has been partially supported by the MUR under the grant "Dipartimenti di Eccellenza 2023-2027" of the Department of Informatics, Systems and Communication of the University of Milano-Bicocca, Italy

FM and AS have 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).

Data availability

The MATLAB code and data used in this study are made publicly available on GitHub (see https://github.com/piazzam/Robust-SVM-Raman).

References

[1] H.J. Butler, L. Ashton, B. Bird, G. Cinque, K. Curtis, J. Dorney, K. Esmonde-White, N.J. Fullwood, B. Gardner, P.L. Martin-Hirsch, M.J. Walsh, M.R. McAinsh, N. Stone, F.L. Martin, Using Raman spectroscopy to characterize biological materials, Nat. Protoc. 11 (4) (2016) 664–687, http://dx.doi.org/10.1038/nprot.2016.036.

[2] S. Deepaisarn, C. Vong, M. Perera, Exploring machine learning pipelines for Raman spectral classification of covid-19 samples, in: 2022 14th International Conference on Knowledge and Smart Technology, Vol. 9, KST, 2022, pp. 51–56, http://dx.doi.org/10.1109/kst53302.2022.9729081.

[3] S. Chen, S. Zhu, X. Cui, W. Xu, C. Kong, Z. Zhang, W. Qian, Identifying non-muscle-invasive and muscle-invasive bladder cancer based on blood serum surface-enhanced Raman spectroscopy, Biomed. Opt. Express 10 (7) (2019) 3533, http://dx.doi.org/10.1364/boe.10.003533.

[4] H. Zhang, C. Chen, R. Gao, Z. Yan, Z. Zhu, B. Yang, C. Chen, X. Lv, H. Li, Z. Huang, Rapid identification of cervical adenocarcinoma and cervical squamous cell carcinoma tissue based on Raman spectroscopy combined with multiple machine learning algorithms, Photodiagnosis Photodyn. Ther. 33 (2021) 102104, http://dx.doi.org/10.1016/j.pdpdt.2020.102104.

[5] C. Carlomagno, P.I. Banfi, A. Gualerzi, S. Picciolini, E. Volpato, M. Meloni, A. Lax, E. Colombo, N. Ticozzi, F. Verde, V. Silani, M. Bedoni, Human salivary Raman fingerprint as biomarker for the diagnosis of Amyotrophic Lateral Sclerosis, Sci. Rep. 10 (1) (2020) 10175, http://dx.doi.org/10.1038/s41598-020-67138-8.

[6] C. Carlomagno, D. Bertazioli, A. Gualerzi, S. Picciolini, M. Andrico, F. Rodà, M. Meloni, P.I. Banfi, F. Verde, N. Ticozzi, V. Silani, E. Messina, M. Bedoni, Identification of the Raman salivary fingerprint of Parkinson's disease through the spectroscopic– computational combinatory approach, Front. Neurosci. 15 (2021) http://dx.doi.org/10.3389/fnins.2021.704963.

[7] V.N. Vapnik, A.Y. Chervonenkis, Theory of Pattern Recognition, Nauka, Moscow, 1974.

[8] R.A. Fisher, The use of multiple measurements in taxonomic problems, Ann. Eugen. 7 (2) (1936) 179–188, http://dx.doi.org/10.1111/j.1469-1809.1936.tb02137.x.

[9] S.A. Dudani, The distance-weighted k-nearest-neighbor rule, IEEE Trans. Syst. Man Cybern. SMC-6 (4) (1976) 325–327, http://dx.doi.org/10.1109/TSMC.1976.5408784.

[10] S. Safavian, D. Landgrebe, A survey of decision tree classifier methodology, IEEE Trans. Syst. Man Cybern. 21 (3) (1991) 660–674, http://dx.doi.org/10.1109/21.97458.

[11] J.S. Pimentel, R. Ospina, A. Ara, A novel fusion support vector machine integrating weak and sphere models for classification challenges with massive data, Decis. Anal. J. 11 (2024) 100457, http://dx.doi.org/10.1016/j.dajour.2024.100457, URL https://www.sciencedirect.com/science/article/pii/S2772662224000614.

[12] B.A. Akinnuwesi, K.A. Olayanju, B.S. Aribisala, S.G. Fashoto, E. Mbunge, M. Okpeku, P. Owate, Application of support vector machine algorithm for early differential diagnosis of prostate cancer, Data Sci. Manag. 6 (1) (2023) 1–12, http://dx.doi.org/10.1016/j.dsm.2022.10.001, URL https://www.sciencedirect.com/science/article/pii/S2666764922000443.

[13] R. Guido, S. Ferrisi, D. Lofaro, D. Conforti, An overview on the advancements of support vector machine models in healthcare applications: A review, Information 15 (4) (2024) 235, http://dx.doi.org/10.3390/info15040235.

[14] M. T.R., V. Kumar Venkatesan, R. Bhardwaj, S. Bhatia Khan, N.A. Alkhaldi, N. Victor, A. Verma, An artificial intelligence-based decision support system for early and accurate diagnosis of Parkinson's disease, Decis. Anal. J. 10 (2024) 100381, http://dx.doi.org/10.1016/j.dajour.2023.100381, URL https://www.sciencedirect.com/science/article/pii/S2772662223002217.

[15] X. Chen, X. Wu, C. Chen, C. nan Luo, Y. Shi, Z. Li, X. Lv, C. Chen, J. Su, L. Wu, Raman spectroscopy combined with a support vector machine algorithm as a diagnostic technique for primary Sjögren's syndrome, Sci. Rep. 13 (2023) 5137, http://dx.doi.org/10.1038/s41598-023-29943-9.

[16] L.A. Duc, N.T. Tung, Use of Raman spectroscopy to diagnose diabetes with svm, in: C.V. Phan, T.D. Nguyen (Eds.), Nature of Computation and Communication, Springer Nature Switzerland, Cham, 2023, pp. 79–87, http://dx.doi.org/10.1007/978-3-031-28790-9_6.

[17] O.L. Mangasarian, Generalized support vector machines, in: A.J. Smola, P. Barlett, B. Schölkopf, D. Schuurmans (Eds.), Advances in Large Margin Classifiers, in: Neural Information Processing Series, MIT Press, 1998, pp. 135–146, http://dx.doi.org/10.7551/mitpress/1113.003.0012.

[18] Jayadeva, R. Khemchandani, S. Chandra, Twin support vector machines for pattern classification, IEEE Trans. Pattern Anal. Mach. Intell. 29 (5) (2007) 905–910, http://dx.doi.org/10.1109/tpami.2007.1068.

[19] X. Peng, Tpmsvm: A novel twin parametric-margin support vector machine for pattern recognition, Pattern Recognit. 44 (10–11) (2011) 2678–2692, http://dx.doi.org/10.1016/j.patcog.2011.03.031.

[20] A. Jiménez-Cordero, J.M. Morales, S. Pineda, A novel embedded min–max approach for feature selection in nonlinear support vector machine classification, European J. Oper. Res. 293 (1) (2021) 24–35, http://dx.doi.org/10.1016/j.ejor.2020.12.009.

[21] X. Liu, F.A. Potra, Pattern separation and prediction via linear and semidefinite programming, Stud. Inf. Control 18 (1) (2009) 71–82.

[22] F. Maggioni, A. Spinelli, A novel robust optimization model for nonlinear support vector machine, European J. Oper. Res. 322 (1) (2025) 237–253, http://dx.doi.org/10.1016/j.ejor.2024.12.014, URL https://www.sciencedirect.com/science/article/pii/S0377221724009561.

[23] R. Gautam, S. Vanga, F. Ariese, S. Umapathy, Review of multidimensional data processing approaches for Raman and infrared spectroscopy, EPJ Tech. Instrum. 2 (1) (2015) 8, http://dx.doi.org/10.1140/epjti/s40485-015-0018-6.

[24] A. Ben-Tal, L. El Ghaoui, A. Nemirovski, Robust Optimization, Princeton University Press, 2009, http://dx.doi.org/10.1515/9781400831050.

[25] D. Bertsimas, D.B. Brown, C. Caramanis, Theory and applications of robust optimization, SIAM Rev. 53 (3) (2011) 464–501, http://dx.doi.org/10.1137/080734510.

[26] D. Bertsimas, J. Dunn, C. Pawlowski, Y.D. Zhuo, Robust classification, INFORMS J. Optim. 1 (1) (2019) 2–34, http://dx.doi.org/10.1287/ijoo.2018.0001.

[27] S. Maldonado, J. López, C. Vairetti, Profit-based churn prediction based on minimax probability machines, European J. Oper. Res. 284 (1) (2020) 273–284, http://dx.doi.org/10.1016/j.ejor.2019.12.007.

[28] D. Faccini, F. Maggioni, F.A. Potra, Robust and distributionally robust optimization models for linear support vector machine, Comput. Oper. Res. 147 (2022) 105930, http://dx.doi.org/10.1016/j.cor.2022.105930.

[29] H. Hotelling, Analysis of a complex of statistical variables into principal components, J. Educ. Psychol. 24 (6) (1933) 417–441.

[30] F. Lussier, V. Thibault, B. Charron, G.Q. Wallace, J.-F. Masson, Deep learning and artificial intelligence methods for Raman and surface-enhanced Raman scattering, TRAC Trends Anal. Chem. 124 (2020) 115796, http://dx.doi.org/10.1016/j.trac.2019.115796.

[31] H. Mohamadi Monavar, N.K. Afseth, J. Lozano, R. Alimardani, M. Omid, J.P. Wold, Determining quality of caviar from Caspian Sea based on Raman spectroscopy and using artificial neural networks, Talanta 111 (2013) 98–104, http://dx.doi.org/10.1016/j.talanta.2013.02.046, URL http://www.sciencedirect.com/science/article/pii/S0039914013001197.

[32] H. Dies, J. Raveendran, C. Escobedo, A. Docoslis, Rapid identification and quantification of illicit drugs on nanodendritic surface-enhanced Raman scattering substrates, Sensors Actuators B: Chem. 257 (2018) 382–388, http://dx.doi.org/10.1016/j.snb.2017.10.181, URL http://www.sciencedirect.com/science/article/pii/S092540051732097X.

[33] S.R. Khandasammy, M.A. Fikiet, E. Mistek, Y. Ahmed, L. Halámková, J. Bueno, I.K. Lednev, Bloodstains, paintings, and drugs: Raman spectroscopy applications in forensic science, Forensic Chem. 8 (2018) 111–133, http://dx.doi.org/10.1016/j.forc.2018.02.002, URL http://www.sciencedirect.com/science/article/pii/S2468170917301133.

[34] C. Carlomagno, A. Gualerzi, S. Picciolini, F. Rodà, P.I. Banfi, A. Lax, M. Bedoni, Characterization of the copd salivary fingerprint through surface enhanced Raman spectroscopy: A pilot study, Diagnostics 11 (3) (2021) 508, http://dx.doi.org/10.3390/diagnostics11030508.

[35] X. Diao, X. Li, S. Hou, H. Li, G. Qi, Y. Jin, Machine learning-based label-free sers profiling of exosomes for accurate fuzzy diagnosis of cancer and dynamic monitoring of drug therapeutic processes, Anal. Chem. 95 (19) (2023) 7552–7559, http://dx.doi.org/10.1021/acs.analchem.3c00026.

[36] X. Zheng, G. Lv, G. Du, Z. Zhai, J. Mo, X. Lv, Rapid and low-cost detection of thyroid dysfunction using Raman spectroscopy and an improved support vector machine, IEEE Photonics J. 10 (6) (2018) 1–12, http://dx.doi.org/10.1109/jphot.2018.2876686.

[37] A. Rahman, S. Kang, W. Wang, Q. Huang, I. Kim, P.J. Vikesland, Lectin-modified bacterial cellulose nanocrystals decorated with au nanoparticles for selective detection of bacteria using surface-enhanced Raman scattering coupled with machine learning, ACS Appl. Nano Mater. 5 (1) (2022) 259–268, http://dx.doi.org/10.1021/acsanm.1c02760.

[38] C. He, X. Wu, J. Zhou, Y. Chen, J. Ye, Raman optical identification of renal cell carcinoma via machine learning, Spectrochim. Acta Part A: Mol. Biomol. Spectrosc. 252 (2021) 119520, http://dx.doi.org/10.1016/j.saa.2021.119520.

[39] M. Sbroscia, M. Di Gioacchino, P. Ascenzi, P. Crucitti, A. di Masi, I. Giovannoni, F. Longo, D. Mariotti, A.M. Naciu, A. Palermo, C. Taffon, M. Verri, A. Sodo, A. Crescenzi, M.A. Ricci, Thyroid cancer diagnosis by Raman spectroscopy, Sci. Rep. 10 (1) (2020) http://dx.doi.org/10.1038/s41598-020-70165-0.

[40] X. Cui, Z. Zhao, G. Zhang, S. Chen, Y. Zhao, J. Lu, Analysis and classification of kidney stones based on Raman spectroscopy, Biomed. Opt. Express 9 (9) (2018) 4175, http://dx.doi.org/10.1364/boe.9.004175.

[41] C. Carlomagno, D. Bertazioli, A. Gualerzi, S. Picciolini, P.I. Banfi, A. Lax, E. Messina, J. Navarro, L. Bianchi, A. Caronni, F. Marenco, S. Monteleone, C. Arienti, M. Bedoni, Covid-19 salivary Raman fingerprint: innovative approach for the detection of current and past sars-cov-2 infections, Sci. Rep. 11 (4943) (2021) http://dx.doi.org/10.1038/s41598-021-84565-3.

[42] X. Li, T. Yang, S. Li, L. Jin, D. Wang, D. Guan, J. Ding, Noninvasive liver diseases detection based on serum surface enhanced Raman spectroscopy and statistical analysis, Opt. Express 23 (14) (2015) 18361–18372, http://dx.doi.org/10.1364/OE.23.018361.

[43] L. Zhang, C. Li, D. Peng, X. Yi, S. He, F. Liu, X. Zheng, W.E. Huang, L. Zhao, X. Huang, Raman spectroscopy and machine learning for the classification of breast cancers, Spectrochim. Acta Part A: Mol. Biomol. Spectrosc. 264 (2022) 120300, http://dx.doi.org/10.1016/j.saa.2021.120300.

[44] A.C. Talari, S. Rehman, I.U. Rehman, Advancing cancer diagnostics with artificial intelligence and spectroscopy: identifying chemical changes associated with breast cancer, Expert. Rev. Mol. Diagn. 19 (10) (2019) 929–940, http://dx.doi.org/10.1080/14737159.2019.1659727.

[45] M. Kazemzadeh, C.L. Hisey, K. Zargar-Shoshtari, W. Xu, N.G. Broderick, Deep convolutional neural networks as a unified solution for Raman spectroscopy-based classification in biomedical applications, Opt. Commun. 510 (2022) 127977, http://dx.doi.org/10.1016/j.optcom.2022.127977.

[46] R. Kothari, V. Jones, D. Mena, V. Bermúdez Reyes, Y. Shon, J.P. Smith, D. Schmolze, P.D. Cha, L. Lai, Y. Fong, M.C. Storrie-Lombardi, Raman spectroscopy and artificial intelligence to predict the bayesian probability of breast cancer, Sci. Rep. 11 (1) (2021) http://dx.doi.org/10.1038/s41598-021-85758-6.

[47] V. Karunakaran, M.M. Joseph, I. Yadev, H. Sharma, K. Shamna, S. Saurav, R.P. Sreejith, V. Anand, R. Beegum, S. Regi David, T. Iype, K. Sarada Devi, A. Nizarudheen, M. Sharmad, R. Sharma, R. Mukhiya, E. Thouti, K. Yoosaf, J. Joseph, P. Sujatha Devi, S. Savithri, A. Agarwal, S. Singh, K.K. Maiti, A non-invasive ultrasensitive diagnostic approach for covid-19 infection using salivary label-free sers fingerprinting and artificial intelligence, J. Photochem. Photobiol. B: Biol. 234 (2022) 112545, http://dx.doi.org/10.1016/j.jphotobiol.2022.112545.

[48] H. Xu, C. Caramanis, S. Mannor, Robustness and regularization of support vector machines, J. Mach. Learn. Res. 10 (2009) 1485–1510, URL https://www.jmlr.org/papers/volume10/xu09b/xu09b.pdf.

[49] F. Maggioni, D. Faccini, F. Gheza, F. Manelli, G. Bonetti, 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, Springer International Publishing, Cham, 2023, pp. 35–46, http://dx.doi.org/10.1007/978-3-031-38537-7_4.

[50] F. Maggioni, A. Spinelli, A robust nonlinear support vector machine approach for vehicles smog rating classification, in: M. Bruglieri, P. Festa, G. Macrina, O. Pisacane (Eds.), Optimization in Green Sustainability and Ecological Transition, in: AIRO Springer Series, Springer Cham, 2024, http://dx.doi.org/10.1007/978-3-031-47686-0_19.

[51] X. Peng, D. Xu, Robust minimum class variance twin support vector machine classifier, Neural Comput. Appl. 22 (5) (2012) 999–1011, http://dx.doi.org/10.1007/s00521-011-0791-3.

[52] Z. Qi, Y. Tian, Y. Shi, Robust twin support vector machine for pattern classification, Pattern Recognit. 46 (1) (2013) 305–316, http://dx.doi.org/10.1016/j.patcog.2012.06.019.

[53] R. De Leone, F. Maggioni, A. Spinelli, A robust twin parametric margin support vector machine for multiclass classification, 2025, URL https://arxiv.org/abs/2306.06213.

[54] R. De Leone, F. Maggioni, A. Spinelli, 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.), Machine Learning, Optimization, and Data Science, in: Lecture Notes in Computer Science, vol. 14506, Springer Nature Switzerland, Cham, 2024, pp. 299–310, http://dx.doi.org/10.1007/978-3-031-53966-4_22.

[55] R. Khanjani-Shiraz, A. Babapour-Azar, Z. Hosseini-Nodeh, P.M. Pardalos, Distributionally robust joint chance-constrained support vector machines, Optim. Lett. 17 (2) (2022) 299–332, http://dx.doi.org/10.1007/s11590-022-01873-x.

[56] F. Lin, S.-C. Fang, X. Fang, Z. Gao, Distributionally robust chance-constrained kernel-based support vector machine, Comput. Oper. Res. 170 (2024) 106755, http://dx.doi.org/10.1016/j.cor.2024.106755.

[57] F. Lin, S.-C. Fang, X. Fang, Z. Gao, J. Luo, A distributionally robust chance-constrained kernel-free quadratic surface support vector machine, European J. Oper. Res. 316 (1) (2024) 46–60, http://dx.doi.org/10.1016/j.ejor.2024.02.022.

[58] C. Cortes, V.N. Vapnik, Support-vector networks, Mach. Learn. 20 (1995) 273–297.

[59] B. Schölkopf, A.J. Smola, Learning with Kernels: Support Vector Machines, Regularization, Optimization, and beyond, The MIT Press, 2001, http://dx.doi.org/10.7551/mitpress/4175.001.0001.

[60] J. López, S. Maldonado, M. Carrasco, A robust formulation for twin multiclass support vector machine, Appl. Intell. 47 (4) (2017) 1031–1043, http://dx.doi.org/10.1007/s10489-017-0943-y.

[61] M. Grant, S. Boyd, Graph implementations for nonsmooth convex programs, in: V. Blondel, S. Boyd, H. Kimura (Eds.), Recent Advances in Learning and Control, Lecture Notes in Control and Information Sciences, Springer-Verlag Limited, 2008, pp. 95–110, http://stanford.edu/boyd/graph_dcp.html.

[62] M. Grant, S. Boyd, CVX: Matlab software for disciplined convex programming, version 2.1, 2014, http://cvxr.com/cvx.

[63] MOSEK ApS, The MOSEK Optimization Toolbox for MATLAB Manual. Version 9.1, 2019, URL http://docs.mosek.com/9.1/toolbox/index.html.

[64] D. Bertazioli, M. Piazza, C. Carlomagno, A. Gualerzi, M. Bedoni, E. Messina, An integrated computational pipeline for machine learning-driven diagnosis based on Raman spectra of saliva samples, Comput. Biol. Med. 171 (2024) 108028, http://dx.doi.org/10.1016/j.compbiomed.2024.108028.

[65] D.A. Whitaker, K. Hayes, A simple algorithm for despiking Raman spectra, Chemometr. Intell. Lab. Syst. 179 (2018) 82–84, http://dx.doi.org/10.1016/j.chemolab.2018.06.009, URL http://www.sciencedirect.com/science/article/pii/S0169743918301758.

[66] C.A. Lieber, A. Mahadevan-Jansen, Automated method for subtraction of fluorescence from biological Raman spectra, Appl. Spectrosc. 57 (11) (2003) 1363–1367, http://dx.doi.org/10.1366/000370203322554518, pMID: 14658149.

[67] J. Han, M. Kamber, J. Pei, Data Mining: Concepts and Techniques, third ed., Morgan Kaufmann, 2012, http://dx.doi.org/10.1016/c2009-0-61819-5.

[68] M.A.K. Raiaan, S. Sakib, N.M. Fahad, A.A. Mamun, M.A. Rahman, S. Shatabda, M.S.H. Mukta, A systematic review of hyperparameter optimization techniques in convolutional neural networks, Decis. Anal. J. 11 (2024) 100470, http://dx.doi.org/10.1016/j.dajour.2024.100470, URL https://www.sciencedirect.com/science/article/pii/S2772662224000742.

[69] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830, URL https://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf.