Distributionally robust joint chance-constrained support vector machines

Article *in* Optimization Letters · March 2022

DOI: 10.1007/s11590-022-01873-x

4 authors:

Rashed Khanjani Shiraz
University of Tabriz

Ali Babapour-Azar

Zohreh Hosseini Nodeh
University of Tabriz

Panos Pardalos
University of Florida

Optimization Letters
https://doi.org/10.1007/s11590-022-01873-x

ORIGINAL PAPER

Distributionally robust joint chance-constrained support vector machines

Rashed Khanjani-Shiraz[1] · Ali Babapour-Azar[1] · Zohreh Hosseini-Nodeh[1] · Panos M. Pardalos[2]

Received: 12 September 2020 / Accepted: 4 March 2022

Abstract
In this paper, we investigate the chance-constrained support vector machine (SVM) problem in which the data points are virtually uncertain although some properties of distributions are available. Thus the robust joint chance-constrained SVM is applied to consider the probability of any existing misclassification in the uncertain data. We transform the chance-constrained SVM into a semidefinite programming problem and a deterministic problem of second-order cone programming.We present new techniques for handling these problems. By assuming the fact that the rows related to the separation constraint matrix of the chance-constrained SVM model are dependent, it is possible to present a new approach for connecting copulas to a stochastic separation constrained support vector machine. Hence, we can use a marginal distribution of the archimedean copula functions, instead of the distribution functions. The experimental results indicate that the stochastic separation constrained support vector machine with copula theory is able to achieve an efficient performance.

Keywords Support vector machines · Copula approach · Joint chance-constrained · Second-order moment information · Robust set

✉ Rashed Khanjani-Shiraz
rashed.shiraz@gmail.com ; r.khanjani@tabrizu.ac.ir

Ali Babapour-Azar
babapoor@tabrizu.ac.ir

Zohreh Hosseini-Nodeh
z.hoseininoudeh@tabrizu.ac.ir

Panos M. Pardalos
pardalos@ufl.edu

1 School of Mathematics, University of Tabriz, Tabriz, Iran

2 Center for Applied Optimization, Industrial and Systems Engineering, University of Florida, Gainesville, FL, USA

Published online: 31 March 2022

1 Introduction

In recent years, SVM type algorithms (as one of the supervised learning algorithms) have been extensively studied. Additionally, an introduced method uses a set of data to produce input or output mapping functions in which every classification function such as regression function is accessible. In 1998, SVM was proposed as a maximum margin classifier and then tutorials on SVM were found [4, 12, 43]. Then, researchers made use of SVM issues in various problems such as solving the large quadratic programming problem of SVM design in data mining [30, 40, 46]. Moreover, certain well-known methods are found in [2], where authors analyze methods based on standard benchmarks in order to evaluate their effectiveness. Kernels adaptation through magnifying the riemannian metric in the boundary vicinity is an example in which geometric methods are being used to improve SVM efficiency. SVMs process is able to approximate each multivariate function to the desired degree like the neural networks.

In the recent decade, a survey on SVM with uncertainties is presented. Initially, the basic SVM models were used for cases in which exact values of points were available, and then different models were offered for the support vector machine with uncertainties. One of the models which were presented against the uncertainty was the Bi and Zhang model [8]. They considered the data points are subject to an additive noise that is bounded by the norm. Therefore, a robust model is presented in order to ensure the most optimal yield in the worst-case scenario constraints in case of existence. Trafalis et al. used the norm to bound the disturbance of uncertainty in the data [41, 47]. Other presented models were expressed when the uncertainty was at intervals [20, 44]. Khanjani et al. discussed a distributionally robust joint chance constrained optimization model and applied it for the shortest path problem under resource uncertainty [25]. Another method that has been considered is robust optimization, where there is a chance-constraint with uncertain data in the problem to ensure a low probability of misclassifications. Lanckriet et al. controlled misclassification probabilities in a worst-case setting: that is, under all possible choices of class-conditional densities with given mean and covariance matrix, they minimize the worst-case (maximum) probability of misclassification of future data points. Also, they interpreted the problem by minimizing the maximum Mahalanobis distances to the two classes [27]. They propose a convex optimization based strategy to deal with uncertainty in the observations of a classification problem. Cao et al. extended the model of fuzzy chance constrained support vector machine for uncertain classification, and following that, Han and Cao concentrate on least squares twin support vector machine classification when data distributions are uncertain statistically [11, 21].

Stochastic and non-stochastic processes are among the factors that create uncertain data in real life. Therefore, creating machine learning techniques and developing them to deal with data uncertainty and decision-making in the face of data turbulence is of great importance. Recent developments in robust optimization have been able to have a great impact on models with data uncertainty. Bhattacharyya et al. Studied the existence of uncertainty in observations and proposed a strategy based on convex optimization to deal with uncertainty in observations of a classification problem [7].

The chance-constrained SVM model is reformulated as a Second-Order Cone Programming (SOCP) and Semi-definite Programming (SDP) when the information of the second-order moment is available and you are able to solve them by the conic linear programming [32, 38, 39, 44]. For this, a direct way is used to reformulate the robust chance-constrained SVM model into SDP model [5, 6, 27, 35]. Besides, the stochastic subgradient descent method is investigated for robust chance-constrained SVM [9, 45, 48]. However, no fully efficient for numerical problem-solving techniques have been proposed yet.

The relationship between the individual probability distributions of random variables and their joint probability distribution is called the copula approach which is offered by Sklar [36]. Also, modeling of multivariate joint distribution and anomalous multivariate discussions are studied using the copula approach [33]. In this way, some studies have been presented due to the fact that the constraint's random coefficient vectors are considered dependent, and the dependence of the random vectors is handled through copulas [26]. The copula approach is used to describe non-parametric measures of dependency for every pair of random variables, plus where we manage to show that a multivariate joint distribution is completely characterized by its respective marginal distributions. Consequently, this approach might be helpful to consider dependency and marginal distribution as two separate but at the same time related subjects. Sklar's Theorem which is probably the most fundamental connection between copulas and statistics helps us replace the joint distribution functions with copulas. The reason that the marginal distribution functions of random variables are estimated based on the historical data is that a joint distribution function using a copula is selected to convert the joint probabilistic constraints into individual probabilistic constraints. The copula functions collect the dependent structure of a set including variables. This paper proposes a technique based on copula for solving robust joint chance-constrained SVM on large-scale data sets that is dependent on the different designs of the reformulations for SVM.

To summarize, in this paper, first of all, a robust model was considered, then a robust set was used. Therefore, we were able to provide a robust SVM model, and we examined the problem in the case where the constraint was the form of joint chance-constrained. We study reformulation of the robust chance-constrained SVM into equivalent SOCP and SDP model as well as the model with second-order moment information of the uncertain data provided. For this purpose, Chebyshev inequalities are used in which the problem can be posed as a SOCP [31]. In the paper, a small example is provided to compare the problem at different values of the confidence level for the separation of the robust chance-constrained with the case given in [44]. Some properties of the distribution such as the first and second moments are available but the exact probability distribution of the random variables is unknown. The separation constraint variables of the SVM are assumed to be dependent, normally or not normally distributed random variables, whereas in [44] the constrained SVM problem is developed and is not considered dependent. The method used to convert joint chance-constrained problems into deterministic problems of second-order cone programming is the marginal distribution of the archimedean copula functions. The uniqueness of this article is to show the issue jointly and assign it to copula. Moreover, we implement numerical experiments to display the efficiency of the proposed approaches.

The organization of this paper is as follows:

Section 2 gives an introduction to the basic SVM models and chance-constrained support vector machines and presents semidefinite programming and second order cone programming for SVM. In this section, the SVM with uncertainties, states both of the robust SVM with bounded uncertainty and chance-constrained SVM through robust optimization. Section 3 discusses the SVM model when we use marginal distribution instead of a joint distribution function. Section 4 shows the numerical results on the equivalence and on the estimation and performance issues, which include two separate sections for random data and real data. Section 5 concludes this paper.

2 Preliminaries and general definitions

Support vector machine is one of the modern tools in classification techniques. It is noteworthy that in the beginning, the support vector theory was not acknowledged and the first articles written by Chervonenkis et al. and others, were ignored till 1992, and were taken seriously only when excellent results on practical learning benchmarks were achieved in digit recognition, computer vision and text categorization [17, 28]. One of the issues to consider for the SVMs is that, some nonlinear and unknown dependency exists in the input vector (e.g., x) which is generated using available moment information (e.g., $x_1 \sim (\mu_1, (\sigma_1)^2)$) and scalar output (e.g., λ). In this case, we only have some information on the pairs of the training data set $D = \{(x_i, \lambda_i) \in X \times \Lambda, \quad i = 1, \ldots k\}$ without having any access to underlying joint probability functions where k is the size of the training data set. SVMs are often called non-parametric, however, it does not mean that no parameters exist. Namely, the parameters are data-driven and their number depends on the training data where you used in the model.

Below we want to investigate the linear maximal margin classifier for linearly separable data. Consider the problem of binary classification which training data are as follows

$$(x_1, \lambda_1), (x_2, \lambda_2), \ldots, (x_k, \lambda_k), \tag{1}$$

where $x \in \mathbb{R}^n$ and $\lambda \in \{+1, -1\}$ and data are linearly separable. Besides, there are some hyperplane that can separate data (Fig. 1) and point separation is done by plane $v_1 x_1 + v_2 x_2 + b = 0$. So, the important thing is to find the best plane where we don't know the underlying probability distribution.

We now need to define an optimal canonical hyperplane and maximal margin from limited training data. Consider the margin M as follows [24]

$$M = \frac{2}{\|v\|}. \tag{2}$$

Equation (2) shows that maximizing of margin M is equivalent to minimization of $\|v\|$. So, learning problem is as follows

$$\min \ \frac{1}{2} v^T v, \tag{3}$$

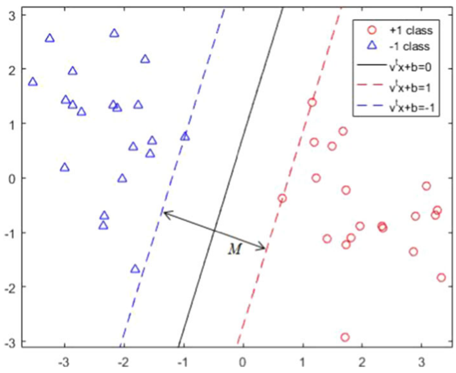


Fig. 1 Illustration of SVM optimization of the margin in the feature space where by minimizing $v^T v$, the margin between both classes is maximized

Table 1 Illustration of SVM optimization for Fig. 1

Optimal value	v^T	b	CPU time
0.562961	(1.021346, -0.287705)	0.219679	0.2423

and constraint are hyperplanes in proper form of avaliable training data as

$$\lambda_i[v^T x_i + b] \geq 1, \quad i = 1, \ldots, K. \tag{4}$$

Table 1 is related to the numerical results of Fig. 1 where obtained from the implementation of Model (3) with constraints (4). In this example, 20 points are considered for each class in which obtain the maximum margin between both classes as a optimal value. It is noted that all the points of the two sets are separable.

In data classification, we sometimes have to ignore some points (see Fig. 2), and these points remain unclassified. The model can now be presented below to evaluate incorrect classification

$$\begin{aligned} \min \quad & \frac{1}{2} v^T v + C \sum_{i=1}^{K} \beta_i, \\ s.t: \quad & \lambda_i[(v^T x_i + b)] \geq 1 - \beta_i, \quad i = 1, \ldots, K \\ & \beta_i \geq 0, \quad i = 1, \ldots, K, \end{aligned} \tag{5}$$

where $\sum_{i=1}^{K} \beta_i$ shows the sum of distances of the wrong side points and $0 \leq C \leq \dfrac{1}{2nk}$ is a penalty parameter, trading off the margin size for the number of misclassified data points and k is a small parameter. Model (5) is named as a soft margin SVM model.

Figure 2 and Table 2 are related to classifiers with two inseparable classes and 50 points for each class. As seen, it is not possible to gain the maximum margin from the linear model due to the interference of the points. Therefore, we have obscured a number of points in order to obtain a reasonable and acceptable margin. We have generated class points randomly by normal distribution with mean's $[-1, 2]^T$ and

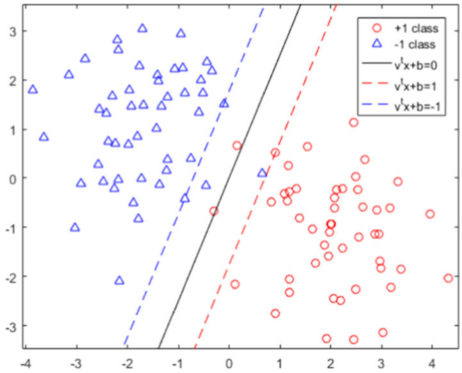


Fig. 2 SVM optimization of the inseparable points

Table 2 Illustration of SVM optimization for Fig. 2

Optimal value	v^T	b	CPU time
452.651	(1.415252, −0.565995)	0.00309	0.4164

$[-2, 1]^T$ for +1 class and −1 class, respectively and the covariance is the identity matrix.

2.1 Chance-constrained support vector machines

The robust optimization has been proposed to guaranty optimal performance for ensuring a small probability of misclassification in uncertain data [3, 6, 41]. Robust classifiers were used in the last years but, the case study of using this method was that these classifiers use only limited partial information. As a result, the researchers tended to be more conservative so that they were to increase the classification margin as well as enhancing the ability to generalize the classifier. Since the goal is to maximize traceable margins by applying support and second-order moment information from uncertain data to make a decision boundary, then the proposed classifiers of the partial information yield a better generalization and they use the boundary knowledge instead of exact moments which are sometimes uncertain and also robust to instantaneous estimation errors. Accordingly, one idea is to accomplish a model with the goal of maximizing margins with chance-constraints on uncertain training datapoints which guarantees classification with high probability. The chance-constrained SVM formulation is defined as follows [35]

$$\begin{aligned} \min\ & \frac{1}{2}v^T v + C\sum_{i=1}^{K}\beta_i, \\ s.t:\ & prob[\lambda_i(v^T x_i + b) \ge 1 - \beta_i] \ge \alpha, \quad i = 1, \ldots, K, \\ & \beta_i \ge 0, \quad i = 1, \ldots, K. \end{aligned} \tag{6}$$

Table 3 Robust SVM classfier for data with different uncertainty sets

α	Optimal value	v^T	b	CPU time
0.95	0.460005	(0.58410, 0.76082)	0.30434	0.33811
0.9	0.357613	(0.53767, 0.65279)	0.25411	0.31990
0.8	0.303155	(0.51058, 0.58789)	0.22372	0.33591
Original data	0.226818	(0.46838, 0.484)	0.17479	0.19722

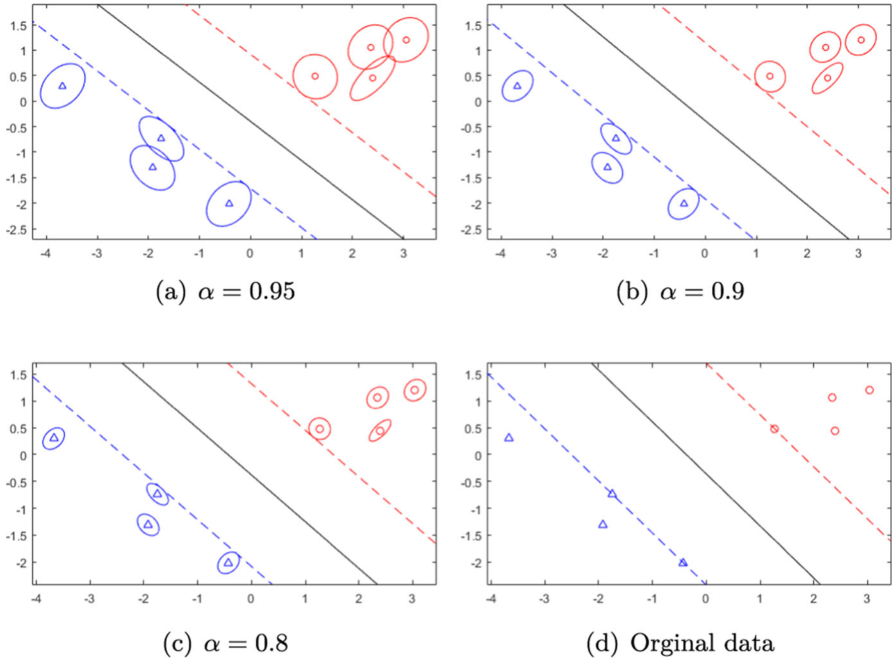


Fig. 3 Robust chance-constrained SVM classifier for data with different confidence level

Table 3 shows the numerical results for a robust chance-constrained example with two inseparable classes and 50 points for each class, which data-points belong to their corresponding ellipsoid uncertainty sets and the mean is in [1,2] for +1 class and $[-2, -1]$ for -1 class and covariance is identity matrix. The optimal value of the last row is based on original data and the rest of the Table rows are obtained for different confidence levels by considering the same ellipsoid uncertainty sets. Since we consider the worst case in the robust chance-constrained, the optimal value is higher than the case in which the data is original, and it results that the margin has decreased as confidence levels have increased. The lower the confidence level, the better the margin obtained.

Figure 3 shows the results of Table 3 intuitively. As you see, the higher the confidence level, leads to a larger uncertainty set. Figure (3) demonstrates the robust chance-constrained SVM with a different confidence level. The black line is the separating line $v^T x + b = 0$ and the red and blue dashed line are the lines determining the margin, i.e., the lines $v^T x + b = \pm 1$. Figure 3d is obtained based on original data with the method given in [44]. In Fig. 3a–c, we set the confidence level $\alpha = 0.95$, $\alpha = 0.9$, and $\alpha = 0.8$ respectively and the data are into ellipsoid uncertainty sets. Then, instead of the dashes which are separating all the points, we make a selection of ellipsoid uncertainty sets so that the dashes do not cross the ellipsoids and the dashed lines are forced to be tangent to the ellipsoids or having a distance from the inner side but still obtaining the maximum margin classifier. As shown in all plots of Fig. 3, the optimal value between these two dash lines are maximized. Besides, the size of the uncertainty set would affect the classifier.

Now, consider that the datapoints x_i are uncertain. The joint chance-constrained program (JCCP) SVM formulation is

$$\begin{aligned} \min \quad & \frac{1}{2} v^T v + C \sum_{i=1}^{K} \beta_i, \\ s.t: \quad & prob[\lambda_i (v^T x_i + b) \geq 1 - \beta_i, \;\; i = 1, \ldots, K] \geq \alpha, \\ & \beta_i \geq 0, \;\; i = 1, \ldots, K. \end{aligned} \tag{7}$$

Assume $x_i \sim (\mu_i, \Sigma_i)$, so we have $v^T x_i \sim (v^T \mu_i, v^T \Sigma_i v)$. Notice, in the assumtions of the robust problem, no kind of distribution is considered and only the distribution is assumed to be elliptical. Robust JCCP with independent raw case and normal distribution is as follows

$$\min \quad \frac{1}{2} v^T v + C \sum_{i=1}^{K} \beta_i, \tag{8}$$

$$\begin{aligned} s.t: \quad & \inf_{x_i \sim (\mu_i, \Sigma_i)} prob[\lambda_i (v^T x_i + b) \geq 1 - \beta_i, \;\; i = 1, \ldots, K] \geq \alpha, \\ & \beta_i \geq 0, \;\; i = 1, \ldots, K. \end{aligned} \tag{9}$$

Then model (8) can be reduce to

$$\begin{aligned} \min \quad & \frac{1}{2} v^T v + C \sum_{i=1}^{K} \beta_i, \\ s.t: \quad & \prod_{i=1}^{K} \inf_{x_i \sim (\mu_i, \Sigma_i)} prob[\lambda_i (v^T x_i + b) \geq 1 - \beta_i] \geq \alpha, \\ & \beta_i \geq 0, \;\; i = 1, \ldots, K, \end{aligned} \tag{10}$$

where for the robust JCCP of SVM we can write $\alpha^{\sum_{i=1}^{K} z_i} = \alpha$ s.t. $\sum_{i=1}^{K} z_i = 1$ [22]. On the other hand the joint chance-constrained of model can be written as K individual chance-constrained

$$\prod_{i=1}^{K} \inf_{x_i \sim (\mu_i, \Sigma_i)} prob\{\lambda_i(v^T x_i + b) \geq 1 - \beta_i\} \geq \prod_{i=1}^{K} \alpha^{z_i}. \tag{11}$$

Thus, by some logarithmic manipulations, (11) reduces to

$$\begin{aligned} &\inf_{x_i \sim (\mu_i, \Sigma_i)} prob\{\lambda_i(v^T x_i + b) \geq 1 - \beta_i\} \geq \alpha^{z_i}, \quad i = 1, \ldots, K, \\ &\sum_{i=1}^{K} z_i = 1, \\ &z_i \geq 0, \quad i = 1, \ldots, K, \end{aligned} \tag{12}$$

constraint (12) can be obtained as follows

$$\inf_{x_i \sim (\mu_i, \Sigma_i)} \{prob[\lambda_i(v^T x_i + b) \geq 1 - \beta_i]\} \geq \alpha^{z_i},$$

or equivalently

$$\sup_{x_i \sim (\mu_i, \Sigma_i)} \{prob[\lambda_i(v^T x_i + b) < 1 - \beta_i]\} \leq 1 - \alpha^{z_i}.$$

Now we can use the multivariate Chebyshev inequality (see, e.g., [27])

$$\sup_{x_i \sim (\mu_i, \Sigma_i)} \{prob[\lambda_i(v^T x_i + b) < 1 - \beta_i]\} = \frac{1}{1 + t^2}, \tag{13}$$

where $t^2 = \inf_{\lambda_i(v^T x_i + b) < 1 - \beta_i} \| (x_i - \mu_i)^T \Sigma^{\frac{-1}{2}} \|^2$. Now assuming that if $\lambda_i(v^T x_i + b) \geq 1 - \beta_i$, so t^2 can be obtained as

$$t^2 = \frac{(\lambda_i(v^T \mu_i + b) - 1 + \beta_i)^2}{v^T \Sigma_i v}, \tag{14}$$

furthermore, since (13) is established, it can be concluded that

$$\frac{1}{1 + t^2} \leq 1 - \alpha^{z_i} \Rightarrow t^2 \geq \frac{1}{1 - \alpha^{z_i}} - 1. \tag{15}$$

Put (14) in (15)

$$\frac{(\lambda_i(v^T \mu_i + b) - 1 + \beta_i)^2}{v^T \Sigma_i v} \geq \frac{1}{1 - \alpha^{z_i}} - 1. \tag{16}$$

Finally, we come to the following inequality as the constraint of Model (10)

$$\sqrt{\frac{\alpha^{z_i}}{1-\alpha^{z_i}}}\sqrt{v^T\Sigma_i v} \le \lambda_i(v^T\mu_i + b) - 1 + \beta_i, \tag{17}$$

thus, model (10) reduces to

$$\begin{aligned}
\min \quad & \frac{1}{2}v^T v + C\sum_{i=1}^{K}\beta_i,\\
s.t: \quad & -\lambda_i(v^T\mu_i + b) + \sqrt{\frac{\alpha^{z_i}}{1-\alpha^{z_i}}}\sqrt{v^T\Sigma_i v} \le \beta_i - 1, \quad i = 1,\ldots,K,\\
& \sum_{i=1}^{K} z_i = 1,\\
& \beta_i \ge 0, \quad i = 1,\ldots,K,\\
& z_i \ge 0, \quad i = 1,\ldots,K.
\end{aligned} \tag{18}$$

The next theorem shows the convexity of $\sqrt{\frac{\alpha^z}{1-\alpha^z}}$. So, by approximating the function $\sqrt{\frac{\alpha^z}{1-\alpha^z}}$ and applying it to the problem, the convex approximation is obtained from the nonlinear constraint.

Theorem 2.1 *For $z \in (0, 1]$, $\sqrt{\frac{\alpha^z}{1-\alpha^z}}$ is convex and decreasing when $\alpha \in (0, 1)$. So, by given any set of values $z_s \in (0, 1]$, $s = 1, \ldots, S$, we can have*

$$f(z) \ge \max_{s=1,\ldots,S} \{a_s z + b_s\}, \tag{19}$$

where

$$a_s = \left.\frac{\partial(f(z))}{\partial(z)}\right|_{z=z_s} \quad \text{and} \quad b_s = \sqrt{\frac{\alpha^{z_s}}{1-\alpha^{z_s}}} - a_s z_s.$$

Proof [14]. □

2.2 Approximation of upper and lower bounds for robust joint chance-constrained SVM

In this section, we discuss the upper and lower bounds for chance constraints. In General, considering the two cases below, it is required to use the approximation and in truth, in this case, the linear approximation. First of all, reliable and relevant risk measurement is required to be processed for the chance constraint. The second case is bounding below the true optimal value. In the following case, we can process a robust best approximation from an ellipsoidal set involving interpolation constraints and uncertain inequality constraints in a measurable space that is immunized against the data uncertainty using a nonsmooth piecewise method. We assumed the input data of the inequality constraints are not exactly known. In this part, we illustrate that finding the best robust approximation is equivalent to solving a complementarity second-order cone problem. That is, following Cheng and Lisser's [14, 22] idea, we find the upper and lower bounds of the objective function of Model (18) by replacing (17) in it through the piecewise linear approximation procedure and the piecewise tangent approximation procedure, respectively.

Now, by Theorem (2.1) we approximate the convex function $\sqrt{\frac{\alpha^z}{1-\alpha^z}}$ with the piecewise linear approximation function. Hence, the final model reduces to

$$\begin{aligned}
\min \quad & \frac{1}{2}v^T v + C\sum_{i=1}^{K}\beta_i,\\
s.t: \quad & -\lambda_i(v^T\mu_i + b) + \max_{s=1,\dots,S}\{a_s z_i + b_s\}\sqrt{v^T\Sigma_i v} \le \beta_i - 1, \quad i = 1,\dots,K,\\
& \sum_{i=1}^{K} z_i = 1,\\
& \beta_i \ge 0, \quad i = 1,\dots,K,\\
& z_i \ge 0, \quad i = 1,\dots,K.
\end{aligned} \tag{20}$$

We can write (20) as

$$\begin{aligned}
\min \quad & \frac{1}{2}v^T v + C\sum_{i=1}^{K}\beta_i,\\
s.t: \quad & -\lambda_i(v^T\mu_i + b) + \sqrt{h_i^2 v^T\Sigma_i v} \le \beta_i - 1, \quad i = 1,\dots,K,\\
& h_i \ge a_s z_i + b_s, \quad i = 1,\dots,K, \ s = 1,\dots,S,\\
& \sum_{i=1}^{K} z_i = 1,\\
& \beta_i \ge 0, \quad i = 1,\dots,K,\\
& z_i \ge 0, \quad i = 1,\dots,K.
\end{aligned} \tag{21}$$

Since v is free, then we can write it as the difference of two non-negative variables v_p and v_n. Finally upper bound model is as follows

$$
\begin{aligned}
\min \quad & \frac{1}{2} \parallel v_p - v_n \parallel_2^2 + C \sum_{i=1}^{K} \beta_i, \\
s.t: \quad & -\lambda_i \Big((v_p - v_n)^T \mu_i + b \Big) + \parallel \Sigma_i^{\frac{1}{2}} (\tau_p^i - \tau_n^i) \parallel_2 \leq \beta_i - 1, \;\; i = 1, \ldots, K, \\
& \tau_p^i \geq a_s \eta_p^i + b_s v_p, \;\; i = 1, \ldots, K, \; s = 1, \ldots, S, \\
& \tau_n^i \geq a_s \eta_n^i + b_s v_n, \;\; i = 1, \ldots, K, \; s = 1, \ldots, S, \\
& \sum_{i=1}^{K} \eta_p^i = v_p, \\
& \sum_{i=1}^{K} \eta_n^i = v_n, \\
& \beta, \eta_p, \eta_n, \tau_p, \tau_n, v_p, v_n \geq 0,
\end{aligned} \tag{22}
$$

where $\eta_p^i = z_i v_p$ and $\tau_p^i = h_i v_p$. Since $\sum_{i=1}^{K} z_i = 1$, so equation $\sum_{i=1}^{K} \eta_p^i = v_p$ is established. Then model (21) will be calculated with the new values a_s and b_s as

$$
a_s = \frac{\sqrt{\frac{\alpha^{z_s}}{1-\alpha^{z_s}}} - \sqrt{\frac{\alpha^{z_{s-1}}}{1-\alpha^{z_{s-1}}}}}{z_s - z_{s-1}},
$$

and

$$
b_s = \frac{z_{s-1}\Big(\sqrt{\frac{\alpha^{z_s}}{1-\alpha^{z_s}}}\Big) - z_s\Big(\sqrt{\frac{\alpha^{z_{s-1}}}{1-\alpha^{z_{s-1}}}}\Big)}{z_{s-1} - z_s}.
$$

2.3 Semidefinite programming and second order cone programming for SVM

It has been shown in some references that some of the distributionally robust optimization models might be solved in polynomial times, while the distribution's mean and covariance matrix are constrained along with being modeled as a SDP reformulation [18]. This process is applied to cases where the chance constraint is present in the model and the distribution is known and the chance constraint is used as an impervious structure. In addition, researches have been conducted on cases where the chance constraints were joint, so that, the uncertainty set is able to be tested with a SDP [10, 13, 49]. SDP model is used to obtain an approximation model where the polynomial convergence rate is known. This application is based on the relaxation scheme for binary constraints. Thus the chance-constrained problem where is solved in the polynomial time becomes SOCP [1].

Now, by [31, 44] we present a formulation we use to solve the above discussed classification problem. Thus SDP and SOCP reformulation of SVM program are as follows respectively

$$\begin{aligned}
\min \quad & \frac{1}{2}v^T v + C\sum_{i=1}^{K}\beta_i, \\
s.t: \quad & \delta_i - \frac{1}{1-\alpha}\, tr(\Omega_i \mathcal{M}_i) \geq 0, \quad i = 1, \ldots, K, \\
& \mathcal{M}_i + \begin{bmatrix} 0 & \frac{1}{2}\lambda_i\, v \\ \frac{1}{2}\lambda_i\, v^T & \lambda_i b + \beta_i - 1 - \delta_i \end{bmatrix} \succeq 0, \quad i = 1, \ldots, K, \\
& \mathcal{M}_i \succeq 0, \quad i = 1, \ldots, K, \\
& \beta_i \geq 0, \quad i = 1, \ldots, K,
\end{aligned} \tag{23}$$

where $\Omega_i = \begin{bmatrix} \Sigma_i + \mu_i\mu_i^T & \mu_i \\ \mu_i^T & 1 \end{bmatrix}$, $i = 1, \ldots, K$ and SOCP model is as follows

$$\begin{aligned}
\min \quad & \frac{1}{2}v^T v + C\sum_{i=1}^{K}\beta_i, \\
s.t: \quad & -\lambda_i(v^T\mu_i + b) \leq \beta_i - 1 - \sqrt{\frac{\alpha}{1-\alpha}}\|\Sigma_i^{\frac{1}{2}} v\|_2, \quad i = 1, \ldots, K, \\
& \beta_i \geq 0, \quad i = 1, \ldots, K.
\end{aligned} \tag{24}$$

3 SVM model based on dependent structure

One of the important problems in mathematical analysis is the relationship between the individual probability distributions of random variables and their joint probability distribution. In the context of non-normal multivariate discussions, the copula method is suggested to model multivariate joint distributions. This approach can be used to illustrate that a multivariate joint distribution is entirely defined by its respective marginal distributions and the linking function of copula, along with the individual of the set of individual marginal distributions. The copula approach is often used to describe non-parametric dependency measures for each pair of random variables and given the marginal distributions, is a helpful tool for deducting joint distributions. It is very helpful to model joint distributions using copulas as marginal distributions and dependence can be considered as two distinct but related topics.In fact, based on historical evidence, the marginal distribution functions of random variables are approximate, so they have appointed a common distribution function using a copula, and then the joint use it to transform the joint probabilistic constraints into individual probabilistic constraints. The dependant configuration of a set of variables derived from the copula functions. Therefore, when researchers are faced with the cumulative distribution function, copulas are a typical tool in designing the dependency among time series. An important characteristic of the copula function is that the multivariate structure and multivariate structure of joint distribution function are alike. In conclusion, the researchers job is becomes easier by using copula models to combine marginal distribution for modeling the dependence structure [34]. Consequently, now copula is briefly used in models with joint chance-constrained random data in order to determine the dependence between rows of constraints. In this section we mention only some basic facts about copulas needed for our following investigation.

Definition 3.1 [29] The copula is the distribution function $C : [0, 1]^k \rightarrow [0, 1]$ of some $k-$ dimensional random vector where its marginal is uniformly distributed on $[0, 1]$.

Definition 3.2 [15, 23] A copula C is called archimedean if there exists a continuous strictly decreasing function $\Psi : [0, 1] \rightarrow [0, +\infty]$, called generator of C, such that $\Psi(1) = 0$ and

$$C(u) = \Psi^{-1}\left(\sum_{i=1}^{n} \Psi(u_i)\right).$$

If $\lim_{u\rightarrow 0} \Psi(u_i) = +\infty$ then C is called a strict archimedean copula and Ψ is called a strict generator.

In general, the distribution information of random variables can be derived from historical data. The process of obtaining a joint probability distribution from historical data is very complex, especially when the random variables follow different marginal probability distributions.

Theorem 3.3 (Sklar's theorem [29]) *For any $k-$dimensional distribution function F : $\mathbb{R}^k \rightarrow [0, 1]$ with marginal $F_1 \times \ldots \times F_k$ there exists a copula C such that*

$$\forall z \in \mathbb{R}^k \quad F(z) = C(F_1(z_1), \ldots, F_k(z_k)).$$

If, moreover, F_k are continuous, then C is uniquely given by

$$C(u) = F(F_1^{(-1)}(u_1), \ldots, F_k^{(-1)}(u_k)),$$

otherwise, C is uniquely determined on $rang F_1 \times \ldots \times rang F_k$.

Also another generator of archimedean copula explained as follows [42]
Joe's copula: $C_\theta(u) = -\ln(1 - (1 - u)^\theta), \quad \theta \geq 1$,
Frank's copula: $C_\theta(u) = -\ln(\frac{e^{-\theta u} - 1}{e^{-\theta} - 1}), \quad \theta \in \mathbb{R}\backslash\{0\}$,
Ali-Mikhail-Haqas copula: $C_\theta(u) = -\ln(\frac{1-\theta}{u} + \theta), \quad \theta \in [-1, 1)$,
Gumbel copula: $C_\theta(u_1, u_2) := \exp(-((-\ln u_1)^\theta + (-\ln u_2)^\theta)^{\frac{1}{\theta}})$,
Clayton copula: $C_\theta(u) = \frac{1}{\theta}(u^{-\theta} - 1), \quad \theta \in [-1, \infty)\backslash\{0\}$.

Sklar's theorem provides a direct relationship between a copula and the joint distribution function of a random vector. According to Sklar's theorem, the copula represents a practical tool for describing the dependency structure of the considered random vector. In addition, Sklar's theorem guarantees the existence and uniqueness of a copula for any distribution function and all its marginals.

Theorem 3.4 [16, 29] *For the matrix $\tilde{a}$ in feasibility set $\{x \in \mathbb{R}^n | prob\{\tilde{a}^T x \leq d\} \geq p\}$, we assume that $\tilde{a}_i^T \sim \Xi(\mu_i, \Sigma_i, \Phi_i)$ (Ξ is multivariate elliptically symmetric distribution [1]) with $\Sigma_i \succ 0$[2], and the "row dependence[3]" is depicted by an Archimedean copula which is independent of x. Note that this assumption holds if the rows of the matrix are normally distributed and independent. Thus the feasible set of the problem (7) can be equivalently recorded as*

$$X(p) = \{x | \exists\, y_i \geq 0 : \sum y_i = 1,\ \mu_i^T x + \Phi^{-1}(\Psi^{-1}(y_i \Psi(p)))\sqrt{x^T \Sigma x} \leq d_i,\ \ i = 1, \ldots, k\},$$

where Φ_i is the distribution function of the i-th constraints' random parameters ($\tilde{a}_i^T$), and Ψ is the generator of an archimedean copula describing the dependence properties of the rows of the matrix $\tilde{a}$.

We extend this idea to the multi-row case by introducing archimedean copula to describe the dependency structure of the optimization problem. According to the linear model of chance-constrained, for nonlinear chance-constrained problems the following model is established as

$$prob\{x \in X \,| h_i(x) \leq \kappa_i,\ i = 1, \ldots, K\} \geq \alpha, \tag{25}$$

where h_i is deterministic continuous function and $\kappa = (\kappa_1, \ldots, \kappa_K)$ is a random variable. If C is archimedean copula, then, we can have this chance-constrained equivalently as [22]

$$\mathbb{M}_{\Phi_i} = \left\{ x \in X | \exists \tau_i \geq 0 :\ \Psi[\Phi_i(h_i(x))] \geq \Psi(\alpha)\tau_i,\ \forall i = 1, \ldots, K, \sum_{i=1}^{K} \tau_i = 1 \right\}. \tag{26}$$

Given what has been said in the introduction to copula, we intend to re-model the chance constraint where the rows of the problem are dependent normally distributed random variables. For this let

$$C[\lambda_i(v^T x_i + b) \geq 1 - \beta_i,\ \ i = 1, \ldots, K] \geq \alpha, \tag{27}$$

and let

$$U = [\lambda\, v^T \quad \lambda\, b - 1 - \beta]^T \qquad and \qquad \hat{x} = [x^T \quad 1]^T.$$

[1] The second-order distributions with probability densities whose contours of equal height are ellipses.

[2] A positive definite matrix is a symmetric matrix where every eigenvalue is positive.

[3] Given matrix A, determine whether the row vectors are linearly dependent.

Assume $\tau_i = \lambda_i(v^T x_i + b) - 1 + \beta_i \geq 0$. Thus, by implementing the mean and the variance of the normal distribution,

$$\frac{\tau_i - \mathbb{E}(x_i)}{\sigma_{\tau_i}} \geq \frac{-\mathbb{E}(x_i)}{\sigma_{\tau_i}}, \tag{28}$$

and let $\kappa_i = \dfrac{\tau_i - \mathbb{E}(x_i)}{\sigma_{\tau_i}}$ and $h_i(x) = \dfrac{-\mathbb{E}(x_i)}{\sigma_{\tau_i}}$. The coeficients corresponding to the deterministic reformulation is obtained by adding the values of κ_i and h_i in the relation (27), as follows

$$C(\kappa_i \geq h_i(x),\ i = 1, \ldots, K),$$

Therefore, from (25) and (26), we get the equivalent description of distribution functions

$$\begin{aligned} &\Psi(\Phi_i(h_i(x))) \geq \tau_i \Psi(\alpha) \\ &\Phi_i(h_i(x)) \geq \Psi^{-1}(\tau_i \Psi(\alpha)) \Rightarrow\ h_i(x) \geq \Phi_i^{-1}(\Psi^{-1}(\tau_i \Psi(\alpha))). \end{aligned} \tag{29}$$

By substituting (29) in (30), we have

$$\mathbb{E}(x_i) + \sigma_{\tau_i} \Phi_i^{-1}(\Psi^{-1}(\tau_i \Psi(\alpha))) \leq 0, \quad i = 1, \ldots, K, \quad \sum_{i=1}^{K} \tau_i = 1. \tag{30}$$

Now, we can model the SVM model by chance constraints, taking into account the dependence between random variables and copula's approach as follows

$$\begin{aligned} \min\ \ & \frac{1}{2} v^T v + C \sum_{i=1}^{K} \beta_i, \\ s.t:\ \ & \lambda_i(v^T x_i + b) + \lambda_i \Phi_i^{-1}(\Psi^{-1}(\Psi(\alpha)\tau_i))\sqrt{x^T \Sigma_i x} \leq 1 - \beta_i, \quad i = 1, \ldots, K \\ & \sum_{i=1}^{K} \tau_i = 1 \\ & \beta_i \geq 0, \quad i = 1, \ldots, K, \\ & \tau_i \geq 0, \quad i = 1, \ldots, K. \end{aligned} \tag{31}$$

By using Theorem (3.4) and Gamble Capula, the constraint of model (31) is written as follows:

$$\lambda_i(v^T x_i + b) + \lambda_i \Phi_i^{-1}(\alpha^{(\tau_i)^{\frac{1}{\theta}}})\sqrt{x^T \Sigma_i x} \leq 1 - \beta_i, \quad i = 1, \ldots, K,$$

where approximations upper and lower bounds are obtained for $\Phi_i^{-1}(\alpha^{(\tau_i)^{\frac{1}{\theta}}})$ as well [22].

3.1 Approximation upper and lower bounds for copula approach

Piecewise linear approximation is a useful tool allowing researchers to tackle optimization problems from another perspective. Piecewise linear approximation use an infinite number of linear parts until it matches the function's curve. The goal is to find the most accurate approximation with the least linear parts. So in this section, for finding the best approximate let

$$g(\tau_i) = \Phi_i^{-1}(\Psi^{-1}(\Psi(\alpha)\tau_i)),$$

assume $s = 1, \dots, S$ be approximated points from (0, 1], thus

$$\begin{aligned}\hat{g}(\tau_i) &= g(\tau_s) + \frac{\tau_i - \tau_s}{\tau_{s+1} - \tau_s}(g(\tau_{s+1}) - g(\tau_s)) \\ &= \frac{(g(\tau_{s+1}) - g(\tau_s))}{\tau_{s+1} - \tau_s}\tau_i + \frac{\tau_{s+1}(g(\tau_{s+1})) - \tau_s(g(\tau_s))}{\tau_{s+1} - \tau_s},\end{aligned}$$

now by replacing $g(\tau_i) = \Phi_i^{-1}(\Psi^{-1}(\Psi(\alpha)\tau_i))$ we have

$$\begin{aligned}\hat{g}(\tau_i) &= \frac{\Phi_i^{-1}(\Psi^{-1}(\Psi(\alpha)\tau_{s+1})) - \Phi_i^{-1}(\Psi^{-1}(\Psi(\alpha)\tau_s))}{\tau_{s+1} - \tau_s}\tau_i \\ &\quad + \frac{\tau_{s+1}\Phi_i^{-1}(\Psi^{-1}(\Psi(\alpha)\tau_s)) - \tau_s\Phi_i^{-1}(\Psi^{-1}(\Psi(\alpha)\tau_{s+1}))}{\tau_{s+1} - \tau_s} \\ &= a_s\tau_i + b_s,\end{aligned} \tag{32}$$

thus in SVM model with copula approach we should use a_s and b_s as picewise linear approximate where

$$\begin{aligned}a_s &= \frac{\tau_{s+1}\Phi_s^{-1}(\Psi^{-1}(\Psi(\alpha)\tau_s)) - \tau_s\Phi_{s+1}^{-1}(\Psi^{-1}(\Psi(\alpha)\tau_{s+1}))}{\tau_{s+1} - \tau_s} \\ b_s &= \frac{\Phi_{s+1}^{-1}(\Psi^{-1}(\Psi(\alpha)\tau_{s+1})) - \Phi_s^{-1}(\Psi^{-1}(\Psi(\alpha)\tau))}{\tau_{s+1} - \tau_s}.\end{aligned}$$

Now by replacing $\Phi_i^{-1}(\Psi^{-1}(\Psi(\alpha)\tau_i))$ by $\max\{a_s\tau_i + b_s\}$ we can achive the model same as model (22) but with different a_s and b_s. By using first-order Taylor series expansion around $\tau = \tau_s, \quad s = 1, \dots, S$ we earn the tangent approximation of $\Phi_i^{-1}(\Psi^{-1}(\Psi(\alpha)\tau_i))$

$$\hat{g}(\tau_i) = \Phi_i^{-1}(\Psi^{-1}(\Psi(\alpha)\tau_i)) + (\tau - \tau_s)(\Phi_i^{-1})'(\Psi^{-1}(\Psi(\alpha)\tau_i)).$$

So a_s and b_s are as follows

$$\begin{aligned}a_s &= \tau(\Phi_i^{-1})'(\Psi^{-1}(\Psi(\alpha)\tau_i)) \\ b_s &= \Phi_i^{-1}(\Psi^{-1}(\Psi(\alpha)\tau_i)) - \tau_s(\Phi_i^{-1})'(\Psi^{-1}(\Psi(\alpha)\tau_i)).\end{aligned}$$

4 Numerical examples

4.1 Stochastic data

In this section, we present experiments performed on datasets. All the experiments were carried out in MATLAB using CVX by SeDuMi solver and a PC with windows system and Intel (R) Core (TM) i7-7700K CPU@ 4.20 GH_z and 8 GB of RAM.

Many real-world problems took account of the factor of uncertainty or perturbation in the input data and were solved at these conditions and as the level of uncertainty decreases we get closer to the solution of the precise data. Furthermore, there are other cases where, the distribution properties are often so unknown that need to be estimated from data points. In this part, our experiments were performed on data sets which produced randomly, which contain high- and low-noise problems. We used a random partition to separate points with the same property, and finally, we tried to put the state of this partition in the best condition. In Sect. 4.1.1, we compare the two cases of robust joint chance-constrained SVM and robust dis-joint chance-constrained SVM problem. Note that, we selected the same factors, and also the confidence level is the same for both models so that we can have a fair comparison. The first test is implemented to certify the better performance of the robust joint chance-constrained SVM than the robust chance-constrained SVM. Section 4.1.2 contains the robust joint chance-constrained SVM results for the larger data set. It is thus important to note that our implementation is based on the second-order cone programming model (22). In Sect. 4.1.3, we examine the joint chance-constrained SVM with the copula approach. The Tables and Figures presented provide significant results.

During this part, assume that the first and second-order moment information of the random variable x_i are known. Let mean and covariance matrix of a random variable as follows:

$$\mu_i = \mathbb{E}[x_i] \quad and \quad \Sigma_i = \mathbb{E}[(x_i - \mu_i)(x_i - \mu_i)^T], \quad i = 1, \dots, K,$$

where $x_i = [x_{i1}, \dots, x_{ik}]^T$.

4.1.1 Robust chance-constrained support vector machine

In this part, we present an example with 12 inseparable points in which data points belong to their corresponding ellipsoid uncertainty sets. The example is presented in such a way that it compares the two modes of robust joint chance-constrained SVM and robust disjoint chance-constrained SVM with the same information and conditions. There are four different confidence levels for this example. The information obtained by testing the uniformly random generated data is as follows:

Similar to the experiments in data, the penalty parameters $C = 10$, The mean $\mu_i,\ i = 1, \dots, K$ related to robust joint chance-constrained SVM is in the interval [0,5] for class +1 and in the interval $[-5, 0]$ for class -1 where obtained by normal distribution. Moreover, interval $[-1, 4]$ for class +1 and the interval $[-3, 2]$ for class -1 are related to robust chance-constrained SVM, and variance-covariance matrix is

Table 4 Computational results for robust joint chance-constrained SVM

α		Calculated values	v^T	b	CPU time	Gap
0.95	UB	0.5965212825	(0.857986346, 0.675945251)	−0.114675591	0.7073148	$1.8189E-06$
	LB	0.59652127165	(0.857986342, 0.675945248)	−0.114675589	0.7681027	
0.9	UB	0.59652127448	(0.857986344, 0.675945249)	−0.11467559	0.712278	$6.0015E-07$
	LB	0.59652127090	(0.857986342, 0.675945248)	−0.114675589	0.8088554	
0.85	UB	0.59652127483	(0.857986346, 0.675945249)	−0.114675591	0.6938122	$1.2741E-07$
	LB	0.59652127407	(0.857986345, 0.675945249)	−0.11467559	0.7398756	
0.8	UB	0.59652127234	(0.857986342,0.675945250)	−0.11467559	0.7050473	$6.5379E-08$
	LB	0.59652127195	(0.857986343, 0.675945249)	−0.114675589	0.752736	

Table 5 Computational results for robust chance-constrained SVM

α	Optimal values	v	b	CPU time
$\alpha = 0.95$	2.23393	(1.801277, 1.106013)	0.028357	0.362455
$\alpha = 0.9$	1.33209	(1.362273, 0.899107)	−0.03413	0.369685
$\alpha = 0.85$	1.09459	(1.221661, 0.834703)	−0.0553	0.366938
$\alpha = 0.8$	0.97934	(1.147453, 0.801265)	−0.06681	0.369083

identity matrix. For this example, confidence levels $\alpha = 0.95, 0.9, 0.85,$ and 0.8 are selected.

In this section, we present the experimental results which extensively compare the proposed (see Table 4) and existing methodologies (see Table 5) for classifying uncertain data. Table 4 presents the results for robust joint chance-constrained SVM model (22). The first column of the Table is different values of the confidence level (α). The second column contains the upper bound (UB) and the lower bound (LB) obtained. Columns 3 and 4 show the vectors obtained for v and the value b, respectively. Column 5 shows the run time of the program for each boundary, and finally, column 6 shows the gap obtained as

$$Gap = \frac{upper\ bound - lower\ bound}{upper\ bound} \times 100.$$

Table 5 presents the results for robust chance-constrained SVM. The first column of the Table is different values of the confidence level. The second column contains the optimal values. Columns 3 and 4 show the vectors obtained for v and the value b, respectively. Column 5 shows the run time.
Note the results obtained in Tables 4 and 5, if we want to compare the two models, we will clearly see the improvement of the optimal value in model provided it is a joint. For example, if $\alpha = 0.9$, both the upper and lower bounds offer better values than the dis-joint model with the optimal value of 1.33209. As the confidence level decreases, we see better results in Table 4, and the upper and lower boundaries are closer together and near to 0. In addition, look at the ratio of changes in each Table. Optimal value

changes in Table 5 from $\alpha = 0.95$ to $\alpha = 0.9$ are almost one unit, while the changes in Table 4 are imperceptibly seen. The changes in the values of b are also clearly shown in Table 5 ($0.028357 \rightarrow -0.03413$).

Figure 4 illustrates the results of Table 4 and Table 5. In other words, this Figure shows a comparison between robust joint chance-constrained SVM and robust chance-constrained SVM with different confidence levels and 7 interpolation points. The first column of the Figure, including $\alpha = 0.95, 0.9$, and 0.85, shows the results in the case where the problem is of the joint type and the second column is related to dis joint one. Although Tables 4 and 5 show the efficiency of the joint method compared to dis-joint, Fig. 4 provides other information. In the points that are closest to the dashed line, the uncertainty set is almost the point itself, and the uncertainty extends to farther points, while the uncertainty in the dis-joint model is uniformly distributed to the points. In other words, the farther we go from the margin, the set of uncertainties gets bigger. In addition, as the confidence level decreases, we see better results and the set of uncertainties gets smaller.

4.1.2 Robust joint chance-constrained support vector machine

This example supports for the stability of the proposed models. The following information were used in the experiments.

The penalty parameters $C = 1{,}000{,}000$, the mean μ_i, $i = 1, \ldots, K$ related to robust joint chance-constrained SVM is in the interval [2,1] for class +1 and in the interval $[-2, -1]$ for class -1 where obtained by normal distribution and variance-covariance matrix is identity matrix. For this example, confidence levels $\alpha = 0.95, 0.9, 0.85$, and 0.8 are selected.

Table 6 shows the results of a random example with 100 points for Model (22). In this model, we have considered four different confidence levels as well as three different interpolation points. The structure of this Table is as follows

The confidence level and the number of interpolation points (IP) are in columns one and two, respectively. Columns 3–6 are the values obtained for the lower bound and columns 7–10 are for the upper bound. Finally, the last column shows the amount of gap between the lower and upper bounds. We tested the model to evaluate the efficiency for a larger number of points. Note that the gap of classified testing decreases as the confidence level decreases in all data. Moreover, there was no significant change in the classified testing points since the changes in optimal values were very close to each other. Consider confidence level 0.8 with the number of interpolation points 3, 4, and 7. We see that as the number of interpolation points increases, the gap created has decreased significantly ($0.0123 \rightarrow 8.9883E-04 \rightarrow 6.5374E-04$). The results show that although we examine the worst case by considering robust, the results are acceptable and close to the answers of the deterministic model. Besides, Fig. 5 shows the accuracy of the results based on the sample size in the graph form.

In the following, we want to examine the superiority of the joint over the dis-joint in terms of the loss of some points during the calculations. If we consider Table 6, the values obtained for the beta at the upper and lower bounds indicate that we do not lose any points and the points are completely separable. The margins themselves are separated by points, but the set of uncertainties is not disturbed by the margins,

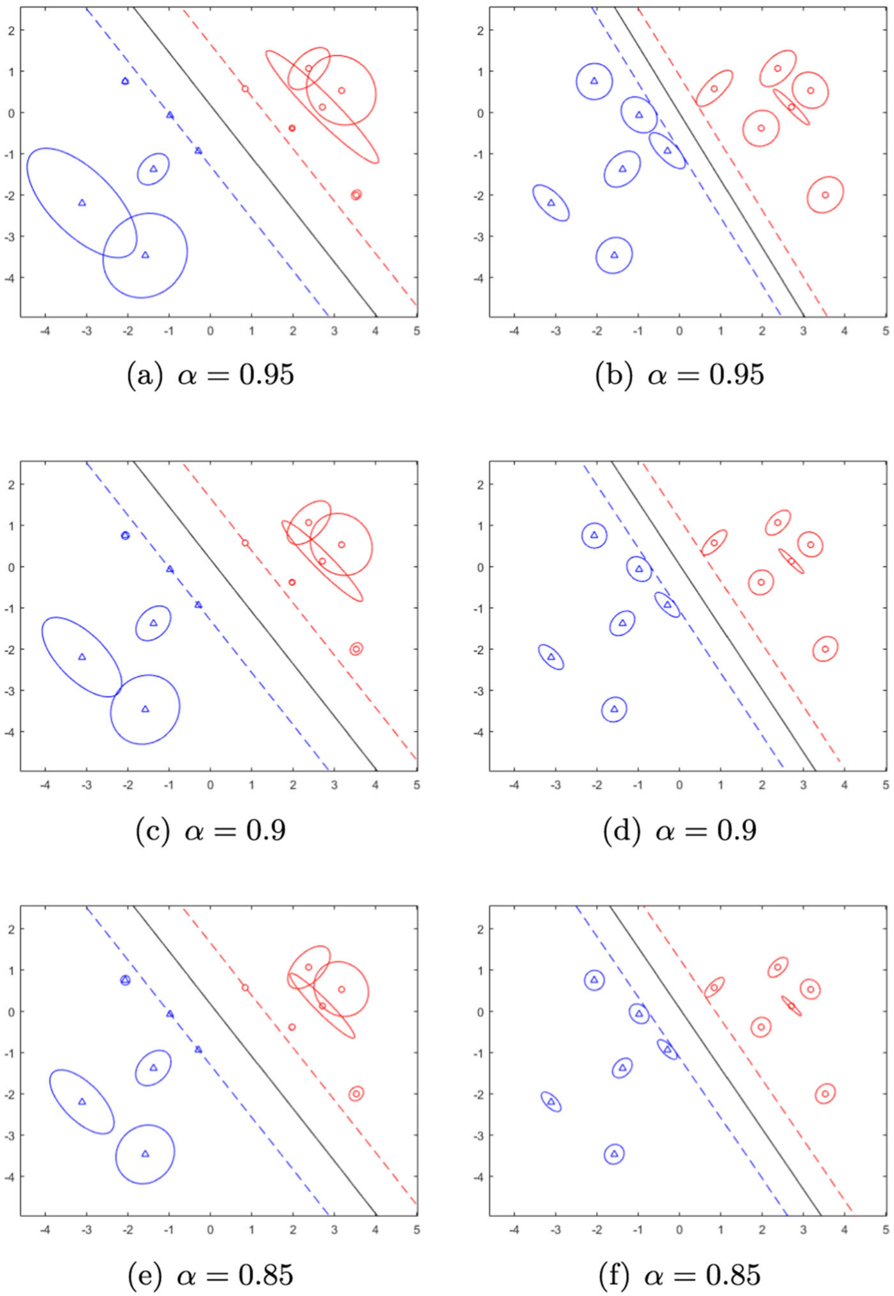
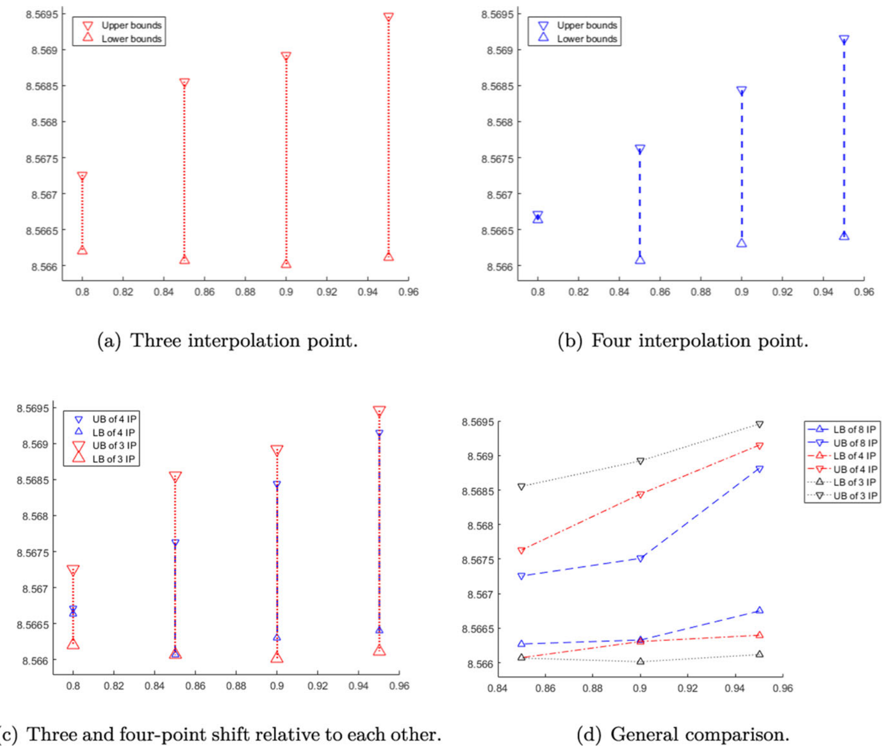


Fig. 4 Comparison between robust joint chance-constrained SVM and robust chance-constrained SVM

Table 6 Computational results for robust joint chance-constrained Support vector machine

α	IP	LB	v^T	b	CPU(s)	$\sum_{i=1}^{K} \beta_i$	UB	v^T	b	CPU(s)	$\sum_{i=1}^{K} \beta_i$	Gap %
0.95		8.566221	(3.2454, 2.5689)	0.08422	5.1433	$9.03E-10$	8.568549	(3.2442, 2.5713)	0.08417	4.6462	$3.70E-09$	0.0272
0.9	7	8.566330	(3.2453, 2.5691)	0.08422	5.0608	$8.16E-10$	8.567511	(3.2448, 2.5701)	0.08420	4.5572	$7.080E-09$	0.0138
0.85		8.566272	(3.2453, 2.5691)	0.08422	4.9617	$9.04E-10$	8.567257	(3.2450, 2.5698)	0.08421	4.5548	$2.97E-09$	0.0115
0.8		8.566241	(3.2454, 2.5689)	0.08423	5.0453	$5.92E-10$	8.566498	(3.2455, 2.5689)	0.08424	4.6232	$3.33E-09$	0.0030
0.95		8.566399	(3.2454, 2.5689)	0.08423	3.3249	$1.40E-09$	8.569151	(3.2440, 2.5718)	0.08418	2.8887	$3.07E-09$	0.0321
0.9	4	8.566308	(3.2454, 2.5690)	0.08422	3.3634	$9.32E-10$	8.568442	(3.2460, 2.5691)	0.08432	2.9211	$2.12E-09$	0.0249
0.85		8.566074	(3.2451, 2.5692)	0.08421	3.3183	$1.61E-10$	8.567633	(3.2458, 2.5689)	0.08429	2.9075	$3.37E-09$	0.0182
0.8		8.566635	(3.2456, 2.5688)	0.08425	3.3417	$1.11E-09$	8.566712	(3.2454, 2.5691)	0.08423	2.9382	$1.54E-09$	$8.9883E-04$
0.95		8.566119	(3.2453, 2.5689)	0.08422	2.9992	$6.25E-09$	8.569456	(3.2470, 2.5681)	0.08436	2.5456	$5.19E-09$	0.0389
0.9		8.566015	(3.2453, 2.5689)	0.08422	2.9253	$3.00E-10$	8.568922	(3.2460, 2.5692)	0.08434	2.5229	$3.45E-09$	0.0339
0.85	3	8.566068	(3.2452, 2.5690)	0.08422	2.9507	$1.30E-09$	8.568554	(3.2443, 2.5712)	0.08419	2.4824	$2.75E-09$	0.0290
0.8		8.566208	(3.2453, 2.5690)	0.08422	2.8851	$3.60E-10$	8.567262	(3.2460, 2.5686)	0.08430	2.4814	$1.69E-09$	0.0123
0.75		8.566091	(3.2453, 2.5690)	0.08422	2.9482	$2.61E-10$	8.566692	(3.2448, 2.5698)	0.08420	2.5450	$8.43E-10$	0.0070
0.7		8.565994	(3.2453, 2.5690)	0.08422	2.9246	$8.75E-11$	8.566312	(3.2454, 2.5690)	0.08423	2.5351	$1.08E-09$	0.0037



(a) Three interpolation point.

(b) Four interpolation point.

(c) Three and four-point shift relative to each other.

(d) General comparison.

Fig. 5 Changes related to robust joint chance-constrained support vector machine with different confidence level and interpolation points

and we have been able to find a margin that gives the best result considering the set of uncertainties. However, in the case of dis-joint, we have to ignore some points where the point or set of uncertainty itself penetrates into the margin. In Table 7 for confidence level greater than 0.8, $\sum_{i=1}^{K} \beta_i$ values indicate that a number of points are missing and are located inside the dashed lines. However, the points have become inseparable for us, but lowering the confidence level reduces the risk of losing points, so that at confidence level 0.8 no point is lost, but it offers a worse optimal value than the corresponding joint state. In addition, Fig. 6 clearly shows all the results. In Fig. 6, we used the zoom of the shapes for greater clarity so that the missing points could be seen. Note, the first column of the Figure, including = 0.95,0.9, and 0.8, shows the results in the case where the problem is of the joint type and the second column is related to dis-joint one.

4.1.3 Chance-constrained support vector machine based on dependency

In this part, we use copulas as a useful tool to design the dependency of chance-constrained random variables. As said, an important characteristic of the copula function is that the multivariate structure of the joint distribution function is alike. Here, we use copula models to combine marginal distribution for modeling the depen-

Table 7 Computational results for robust chance-constrained SVM with 100 points

α	Optimal values	v	b	CPU time	$\sum_{i=1}^{K} \beta_i$
$\alpha = 0.95$	$6.52E + 06$	(2.5488, 0.4628)	0.0390	1.4803	6.516748
$\alpha = 0.9$	$4.37E + 06$	(2.7772, 1.4507)	0.3748	1.4632	4.369038
$\alpha = 0.85$	$2.13E + 06$	(9.3089, 7.3867)	1.1749	1.4667	2.130911
$\alpha = 0.8$	$4.33E + 02$	(23.0508, 18.2839)	0.30352	1.5154	$1.40E - 08$
$\alpha = 0.75$	$1.31E + 02$	(12.6779, 10.0535)	0.1982	1.4600	$7.00E - 09$
$\alpha = 0.7$	72.5220	(9.4370, 7.4819)	0.15904	1.4579	$2.83E - 09$

dence structure. For this purpose, Table 8 based on the copula approach is presented for the example which we tested in the previous section.

In this part, to make the results easier to interpret, firstly, synthetic data is generated using a 2-dimensional normal distribution with the +1 class and -1 class generated by a normal distribution with the same information of the previous example in which each class has 50 points. The structure of Table 8 is the same as in Table 4, but the results have been significantly improved. We use Gumbel copula with parameter $\theta \geq 1$ to handle the dependency of random variables. In this part, our goal is to find the maximum margin by randomly considering some parameters. As the number of interpolation points and the amount of θ increases, the Gap decreases in the problem. Note that, the problem is a cone model and exist of complexity in the solving process, thus by considering the complexity of the problem it seems cpu time is acceptable. The results show that by increasing the value of θ and α, we need fewer interpolation points to achieve convergence. An increase in the θ is fully visible in the process of improving the Table. In addition, the number of interpolation points and the confidence level have had a significant impact on the improvement of this process. The gaps from the upper and lower bounds are better at each step and close to 0.

To compare how the copula approach can be controller the optimal value and would affect the performance of the models, we presented Table 9. Table 9 presents joint chance-constrained SVM results for example with 100 data points which we use $\Phi^{-1}(\alpha^{z_i})$ instead of $\Phi_i^{-1}(\Psi^{-1}(\Psi(\alpha)z_i))$. In the calculations of Table 9, we used normal distribution and 6-point interpolation for all confidence levels. The numerical results show that the optimal value obtained for copula is closer to the actual distribution, and we can use copula to obtain an acceptable approximation in cases where another distribution is used instead of the normal distribution. For example, if $\alpha = 0.9$ upper bound of joint chance-constrained SVM is 8.566132, this is while, for both $\theta = 2$, and 4 the optimal value with copula is better than joint chance-constrained SVM (8.566099 for $\theta = 2$, and 8.565992 for $\theta = 4$). Thus, we can rely on the copula functions to obtain results close to the actual distribution.

4.2 Real data analysis

In this section, besides the synthetic random data, we also tested on real data, the Breast Cancer Wisconsin (Original) data from UCI dataset [44]. There are 699 samples

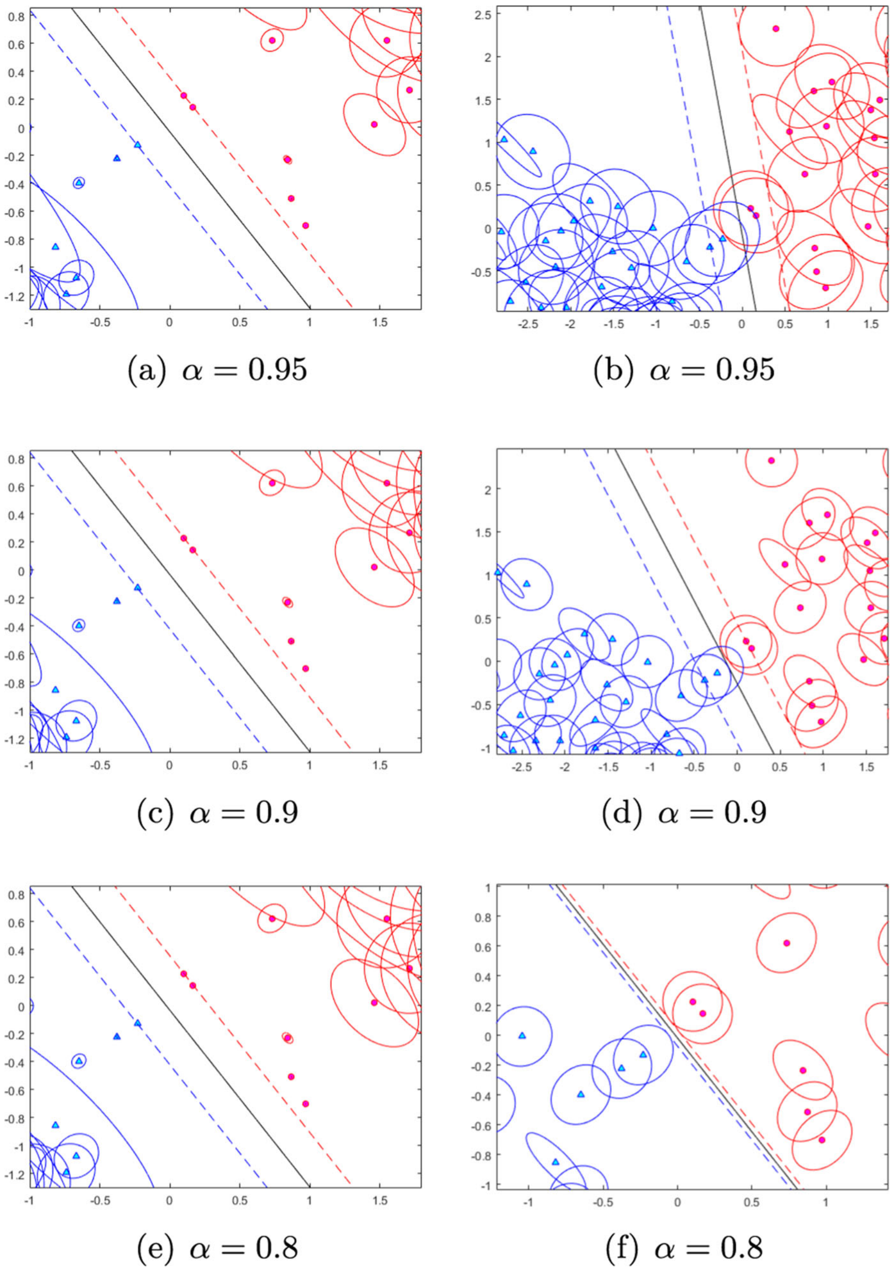


Fig. 6 Compare of robust joint chance-constrained SVM and robust chance-constrained SVM based on losing points

Table 8 Computational results based on dependency and copula approach

θ	IP	α		Calculated values	v^T	b	CPU time	Gap
2		0.95	UB	8.566197	(3.245172, 2.569291)	0.084218	4.135265	0.0021
			LB	8.566019	(3.245370, 2.568971)	0.084223	4.599681	
	7	0.9	UB	8.566099	(3.245395, 2.568970)	0.084228	4.080396	0.0019
			LB	8.565939	(3.245348, 2.568968)	0.084221	4.555879	
		0.85	UB	8.566039	(3.245395, 2.568947)	0.084225	4.172046	7.0044e-04
			LB	8.565979	(3.245356, 2.568973)	0.084222	4.519899	
	3	0.95	UB	8.566551	(3.245470, 2.569051)	0.084242	2.426281	0.0064
			LB	8.565999	(3.245364, 2.56897)	0.084223	2.881934	
4	7	0.95	UB	8.566080	(3.245449, 2.568895)	0.084229	4.079845	5.0198e-04
			LB	8.566037	(3.245378, 2.568968)	0.084226	4.537607	
		0.9	UB	8.565992	(3.245309, 2.569038)	0.084220	4.109181	1.0507e-04
			LB	8.565983	(3.245362, 2.568967)	0.084222	4.548901	
	3	0.95	UB	8.566027	(3.245421, 2.568956)	0.084228	2.404224	5.6035e-04
			LB	8.565979	(3.245370, 2.568956)	0.084223	2.845477	
5	3	0.95	UB	8.565979	(3.245366, 2.568961)	0.084222	2.429382	1.1674e-05
			LB	8.565978	(3.245368, 2.568957)	0.084223	2.856382	

Table 9 Comparing robust joint chance-constrained SVM with copula approach and joint chance-constraint SVM

α		Calculated values	v^T	b	CPU time	Gap
0.95	UB	8.566229	(3.245417, 2.568993)	0.084227	3.2488	0.0030
	LB	8.565973	(3.245371, 2.568952)	0.084223	3.6288	
0.9	UB	8.566169	(3.245436, 2.568947)	0.084228	3.2708	0.00210
	LB	8.565989	(3.245355, 2.568978)	0.084222	3.6084	
0.85	UB	8.566172	(3.245441, 2.568942)	0.084228	3.2305	0.00208
	LB	8.565993	(3.245354, 2.568981)	0.084222	3.6280	
0.8	UB	8.566190	(3.245450, 2.568937)	0.084228	3.2455	0.0014
	LB	8.566070	(3.245447, 2.568894)	0.084230	3.6265	

in this info, while 16 samples have missing values so that we do not use them in the calculation process, resulting in 683 samples. 444 samples are benign and 239 are malignant, and we record them as class +1 and −1, respectively. Each of these samples has 10 attributes, but the first one is Id number so we do not use it into the experiments. Therefore, $d = 9$ dimensions are considered. Besides, Ionosphere data with 351 samples is also used in the experiments and we record them in two classes of 225 and 126 as good class (+1) and bad class (−1), respectively. Also, for Ionosphere data d is equals 34. These samples are deterministic real and there is no probability in them, but our model is chance-constrained so we need random samples. Therefore, we generate random data from real samples. For this purpose, we consider real samples as means and the covariance matrix is $0.01 I_n$ when I_n is the $n \times n$ identity matrix.

Table 10 shows the results obtained for the Wisconsin breast cancer and Ionosphere data from UCI dataset. We tested the model to evaluate the efficiency for a larger number of points. This table contains two sections for Breast Cancer Wisconsin and Ionosphere data at the $\alpha = 0.95$ and 0.9 confidence levels. For both data types, numerical results are reported for the SVM-SOCP model [44], Robust Joint chance-constrained ($R - JCC$), and Capula models. Because the samples are not linearly separable, some of the samples are lost and are not classified. Also, Table 10 shows that k=40 samples are lost in $R - JCC$ and 48 samples are lost in SVM-SOCP. This indicates that our model has fewer missing samples. In addition, according to the minimization model, the optimal value of $R - JCC$ had better results than SVM-SOCP. The results show that although we examine the worst case by considering robust, the results are acceptable and close to the answers of the deterministic model. In addition, v and b are reported in "Appendix".

Figure 7 shows the data overlap. In order to be able to provide a two-dimensional diagram of how the data is distributed, we have used PCA to extract the first two principle components. In addition, we used PCA to reduce the Ionosphere data dimension from 34 to 10. Figures 7a, b show the overlap of Wisconsin breast cancer and Ionosphere data, respectively. As you can see, benign data are greater than malignant data, but malignant data are more spread than benign data.

Table 10 Comparing robust joint chance-constrained SVM with copula approach and SVM-SOCP [44]

Real data	Compared methods	α		$\sum_{i=1}^{K} \beta_i$	Optimal value	$\frac{1}{2} v^T v$	CPU time	k
	$R - JCC$	0.95	UB	43.97674925	4397.78086890221	0.105943495	76.6754857	40
			LB	43.97674913	4397.78085595417	0.105943217	90.8685959	40
Wisconsin breast cancer		0.9	UB	43.97674906	4397.78084884344	0.105943193	79.7892989	40
$c = 100$			LB	43.97674902	4397.78084518318	0.105943183	92.6608547	40
	SVM-SOCP	0.95		54.01181722	5401.29	0.112110913	9.9145959	50
		0.9		50.63481078	5063.59	0.110131052	9.7983306	48
	Capula ($\theta = 2$)	0.95	UB	43.97674081	4397.78002436045	0.105943284	68.3071166	40
			LB	43.97674535	4397.78047840811	0.105943201	90.7299878	40
	$R - JCC$	0.95	UB	120.8992237	12091.9198122286	1.997437721	20.5870737	116
			LB	120.8992177	12091.9192054231	1.997437597	24.3662693	116
Ionosphere		0.9	UB	120.8992142	12091.9188534116	1.99743737	25.0663682	116
$c = 100$			LB	120.8992143	12091.9188636893	1.997437367	28.4166857	116
	SVM-SOCP	0.95		240.9757369	24098.2	0.618958865	4.9685689	237
		0.9		203.669797	20367.9	0.917239064	5.0091544	200
	Capula ($\theta = 2$)	0.95	UB	120.8992141	12091.9188431952	1.997437512	20.9970876	116
			LB	120.8992139	12091.9188241003	1.997437361	22.9978902	116

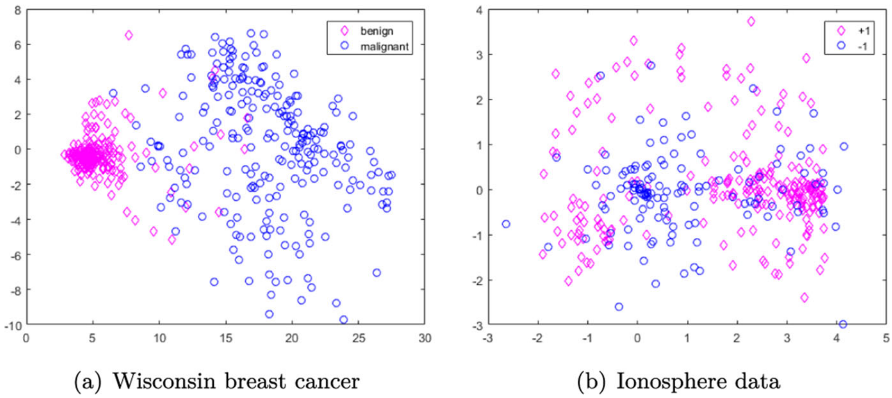


Fig. 7 First two principal components of Wisconsin breast cancer and Ionosphere data

Table 11 Accuracy, sensitivity, and specificity for Wisconsin breast cancer data (α=0.95)

Used data (%)		t_p	t_n	f_p	f_n	Accuracy	Sensitivity	Specificity	Correctly classified data(%)
25	Joint	426	197	18	43	0.91081	0.90831	0.91627	91.21
	Disjoint	406	195	38	45	0.87865	0.90022	0.83690	87.99
75	Joint	416	215	28	25	0.92251	0.94331	0.88477	92.38
	Disjoint	414	204	30	36	0.90350	0.92000	0.87179	90.48
100	Joint	424	219	20	20	0.94143	0.95495	0.91631	94.14
	Disjoint	418	216	26	24	0.92690	0.94570	0.89256	92.82

4.3 Numerical results in terms of training point and accuracy

In the previous section we compared our approaches with respect to the optimal value and CPU time. Nevertheless, in order to compare methods, we want to apply a measure that shows how well the method classifies some test data. One measure to compare machine learning methods is the accuracy. So, we now validate the effectiveness of robust joint chance-constrained SVM on Wisconsin breast cancer data. In our experiments, we conduct empirical comparisons on the classification accuracy between robust joint chance-constrained SVM and disjoint chance-constrained SVM considering that robust joint chance-constrained SVM is more justifiable than disjoint chance-constrained SVM.

The subset of data used to guide the selection of hyperparameters is called the validation set. To compare how the chance constraint probability controller α would affect the performance of the models, we change the training and test point. Thus, the classification accuracy on the test set and results in terms of training point are reported in Table 11.

Table 11 ensures that for robust joint chance-constrained SVM and disjoint one the proposed method has acceptable accuracy given the percentage of data used. Also, the table shows that with this level of confidence, the data is placed at the top and bottom of the separator hyperplane. The first column show the percentage of data used. A value of 100 indicates that all data was used in the implementation of the method, and a value of 25 means that only 25% of the data was used in the conclusion where this percentage of data is randomly selected using function 'randi()' . The third to sixth columns show t_p, t_n, f_p, and f_n values in reference [37], where t_p are true positive, t_n are true negative, f_p are false positive, and f_n are false negative counts (see [37] section 2 relationships 1 and 2, respectively). The last column is the percentage of data that is correctly classified. Note that for joint chance constraint case the upper bound values are reported for the minimization problem with five interpolation points.

5 Conclusion

In this paper, a novel methodology for constructing robust classifiers by employing partial information on the support and moments of the uncertain training datapoints is presented. The main idea was to investigate the chance-constrained SVM problem. Thus the robust joint chance-constrained SVM is applied to consider the probability of any existing misclassification in the uncertain data. We transformed the joint chance-constrained SVM model into a deterministic problem of second-order cone programming to handle these types of problems. During this paper, the rows related to the separation constraint matrix of chance-constrained the SVM model were considered to be dependent. The type of problem allowed us to present a new approach for connecting copulas to a stochastic separation constrained support vector machine. The numerical results showed improvement of the optimal value in the model provided it is a joint chance-constrained problem. This is a key feature of the proposed model that provides acceptable results and higher margins without knowing the probability distribution of the random variables. In addition, in the last section, the copula approach was able to achieve an efficient performance. To emphasize the veracity of our claims, the robust joint chance-constrained SVM problem examined in terms of accuracy.

Acknowledgements Rashed Khanjani Shiraz, Ali Babapour Azar, Zohreh Hosseini Nodeh have been supported by University of Tabriz, International and Academic Cooperation Directorate, in the framework of TabrizU-300 program.
P.M. Pardalos was supported by a Humboldt Research Award (Germany) and the Paul and Heidi Brown preeminant professorship at ISE, University of Florida (USA).

Appendix

See Table 12.

Table 12 The obtained value of v and b for Table 10

		$\alpha = 0.95$	UB	v	(−0.235362844, 0.022801313, −0.17137641, −0.112116219, −0.094577742, −0.177681498, −0.180143072, −0.091044312, −0.181039338)
				b	4.274537494
	Robust JCC		LB	v	(−0.235362872, 0.022800748, −0.171376282, −0.112116016, −0.094577825, −0.17768161, −0.180142703, −0.091044452, −0.181038233)
				b	4.27453696
		$\alpha = 0.9$	UB	v	(−0.235362872, 0.022800705, −0.171376266, −0.112116002, −0.094577834, −0.177681616, −0.180142677, −0.091044463, −0.18103814)
				b	4.274536899
Wisconsin breast cancer			LB	v	(−0.235362873, 0.022800685, −0.171376262, −0.112115993, −0.094577836, −0.17768162, −0.180142664, −0.091044468, −0.181038101)
				b	4.274536879
		$\alpha = 0.95$		v	(−0.220785247, −0.220756862, −0.012588911, −0.186458442, −0.101185752, −0.123642355, −0.201996403, −0.174171153, −0.096797203)
				b	4.521004846
	SVM-SOCP	$\alpha = 0.9$		v	(−0.231822043, −0.000542362, −0.187001797, −0.095347773, −0.117516723, −0.192705348, −0.178054704, −0.102281334, −0.171315497)
				b	4.481584871
	Capula	$\alpha = 0.95$	UB	v	(−0.235362873, 0.022800816, −0.171376356, −0.112116102, −0.094577827, −0.177681628, −0.180142703, −0.091044445, −0.181038455)
				b	4.274537477
			LB	v	(−0.235362875, 0.022800685, −0.171376304, −0.112116025, −0.094577836, −0.177681639, −0.180142634, −0.091044475, −0.181038145)
				b	4.274537106
		$\alpha = 0.95$	UB	v	(0.205237201, 0.06119248, 1.528798593, −0.937569319, −0.466806521, −0.06322833,−0.073598848, 0.461902482, −0.390071378, −0.374051769)
				b	−1.649419085
	Robust JCC		LB	v	(0.205237224, 0.061192469, 1.528798611, −0.937569372, −0.466806424, −0.063228327, −0.073598797, 0.46190233, −0.390071111, −0.374051816)

Table 12 continued

				b	−1.649419044
		$\alpha = 0.9$	UB	v	(0.205237237, 0.061192486, 1.528798631, −0.937569333, −0.466806366, −0.063228276, −0.073598735, 0.461902183, −0.390070844, −0.37405177)
				b	−1.649418981
Ionosphere			LB	v	(0.205237238, 0.061192486, 1.52879863, −0.93756933, −0.466806365, −0.063228276, −0.073598734, 0.461902185, −0.390070844, −0.374051768)
				b	−1.64941898
		$\alpha = 0.95$		v	(0.304724017, 0.155488872, 0.829666019, −0.556670097, −0.100647094, 0.041531576, 0.052366296, 0.133387781, −0.223606803, −0.200668067)
				b	−0.695841754
	SVM-SOCP	$\alpha = 0.9$		v	(0.331996334, 0.15009671, 1.0137561, −0.633468565, −0.111136641, −0.036354249, 0.048598023, 0.256174208, −0.312355045, −0.305807184)
				b	−1.049222485
	Capula	$\alpha = 0.95$	UB	v	(0.205237237, 0.06119247, 1.528798618, −0.93756937, −0.466806384, −0.063228313, −0.073598781, 0.461902265, −0.390070992, −0.374051818)
				b	−1.649419022
			LB	v	(0.205237238, 0.061192486, 1.528798631, −0.937569332, −0.466806363, −0.063228275, −0.073598733, 0.461902178, −0.390070834, −0.374051769)
				b	−1.649418979

References

1. Alizadeh, F., Goldfarb, D.: Second-order cone programming. Math. Program. **95**(1), 3–51 (2003)
2. Anguita, D., Boni, A., Ridella, S., Rivieccio, F., Sterpi, D.: Theoretical and practical model selection methods for support vector classifiers. In: Support Vector Machines: Theory and Applications. Springer, Berlin, Heidelberg, pp. 159–179 (2005)
3. Ben-Hur, A., Weston, J.: A users guide to support vector machines. In: Data Mining Techniques for the Life Sciences. Berlin: Springer, pp. 223–239 (2010)
4. Ben-Tal, A., Bhadra, S., Bhattacharyya, C., Nath, J.S.: Chance-constrained uncertain classification via robust optimization. Math. Program. **127**(1), 145–173 (2011)
5. Bertsimas, D., Popescu, I.: Optimal inequalities in probability theory: a convex optimization approach. SIAM J. Optim. **15**(3), 780–804 (2005)
6. Bhattacharyya, C., Pannagadatta, K.S., Smola, A.J., El Ghaoui, L., Mian, I.S.: Robust sparse hyperplane classifiers: application to uncertain molecular profiling data. J. Computat. Biol. **11**(6), 1073–1089 (2004)
7. Bhattacharyya, C., Grate, L.R., Jordan, M.I., El Ghaoui, L., Mian, I.S.: A second order cone programming formulation for classifying missing data. In: Neural Information Processing Systems (NIPS), pp. 153–160 (2005)
8. Bi, J., Zhang, T.: Support vector classification with input data uncertainty. Adv. Neural Inf. Process. Syst. 161–168 (2005)
9. Bottou, L.: Large-scale machine learning with stochastic gradient descent. In: Proceedings of COMPSTAT. Springer, pp. 177–186 (2010)
10. Calafiore, G. C., El Ghaoui, L.: On distributionally robust chance-constrained linear programs. J. Optim. Theory Appl. 130(1): 1–22 (2006)
11. Cao, Q., Lu, Y., Dong, D., Tang, Z., Li, Y.: The roles of bridging and bonding in social media communities. J. Am. Soc. Inf. Sci. Technol. 64(8):1671–1681 (2013)
12. Chang, C.C., Lin, C.J.: A library for support vector machines. ACM Trans. Intell. Syst. Technol. **2**(3), 1–27 (2011)
13. Chen, W., Sim, J., Sun, J., Teo, C.P.: From CVaR to uncertainty set: implications in joint chance-constrained optimization. Oper. Res. **58**(2), 470–485 (2010)
14. Cheng, J., Delage, E., Lisser, A.: Distributionally robust stochastic knapsack problem. SIAM J. Optim **24**(3), 1485–1506 (2014)
15. Cheng, J., Houda, M., Lisser, A.: Chance-constrained 0–1 quadratic programs using copulas. Optim. Lett. **9**(7), 1283–1295 (2015)
16. Cheng, J., Houda, M., Lisser, A.: Second-order cone programming approach for elliptically distributed joint probabilistic constraints with dependent rows. Technical Report, Optimization (2014)
17. Chu, F., Wang, L.: Gene expression data analysis using support vector machines. In: IEEE International Joint Conference on Neural Networks, pp. 2268-2271 (2003)
18. Delage, E., Ye, Y.: Distributionally robust optimization under moment uncertainty with application to data-driven problems. Oper. Res. **58**(3), 595–612 (2010)
19. Fan, N., Sadeghi, E., Pardalos, P.M.: Robust support vector machines with polyhedral uncertainty of the input data. In: International Conference on Learning and Intelligent Optimization, pp. 291-305 (2014)
20. Ghaoui, L.E., Lanckriet, G.R., Natsoulis, G.: Robust classification with interval data. Technical Report UCB/CSD-03-1279, Computer Science Division, University of California, Berkeley (2003)
21. Han, R., Cao, Q.: Fuzzy chance constrained least squares twin support vector machine for uncertain classification. J. Intell. Fuzzy Syst. **33**(5), 3041–3049 (2017)
22. Hosseini Nodeh, Z., Babapour, A., Khanjani, S.R., Khodayifar, S., Pardalos, P.M.: Joint chance constrained shortest path problem with copula theory. J. Comb. Optim. **13**, 1–31 (2020)
23. Houda, M., Lisser, A.: On the use of copulas in joint chance-constrained programming. In: Proceedings of the 3rd International Conference on Operations Research and Enterprise Systems. SCITEPRESS-Science and Technology Publications, Lda, pp. 72–79 (2014)
24. Kecman, V.: Support vector machines—an introduction in support vector machines: theory and applications. Springer, Berlin, pp. 1–47 (2005)
25. Khanjani-Shiraz, R., Babapour-Azar, A., Hosseini-Noudeh, Z., Pardalos, P.M.: Distributionally robust maximum probability shortest path problem. J. Comb. Optim. **1**, 1–28 (2021)

26. Khanjani-Shiraz, R., Khodayifar, S., Pardalos, P.M.: Copula theory approach to stochastic geometric Programming. J. Glob. Optim. **1**, 1 (2021). https://doi.org/10.1007/s10898-021-01062-7
27. Lanckriet, G.R., Ghaoui, L.E., Bhattacharyya, C., Jordan, M.I.: A robust minimax approach to classification. J. Mach. Learn. Res. **3**, 555–582 (2002)
28. Meyer, D., Leisch, F., Hornik, K.: The support vector machine under test. Neurocomputing **55**(1–2), 169–186 (2003)
29. Michal, H.: A note on the use of copulas in chance-constrained programming. In: Proceedings of 32nd International Conference on Mathematical Methods in Economics MME, pp. 327-332 (2014)
30. Mitra, P., Murthy, C.A., Pal, S.K.: Active support vector learning with statistical queries. In: Support Vector Machines: Theory and Applications. Springer, Berlin, pp. 99–111 (2005)
31. Nath, J.S., Bhattacharyya, C.: Maximum margin classifiers with specified false positive and false negative error rates. In: Proceedings of the 2007 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics, pp. 35–46 (2007)
32. Nesterov, Y, Nemirovskii, A.: Interior-point polynomial algorithms in convex programming. Society for Industrial and Applied Mathematics (1994)
33. Nelsen, R. B.: An Introduction to Copulas. Springer (2007)
34. Papaefthymiou, G., Kurowicka, D.: Using copulas for modeling stochastic dependence in power system uncertainty analysis. IEEE Trans. Power Syst. **24**(1), 40–49 (2008)
35. Shivaswamy, P.K., Bhattacharyya, C., Smola, A.J.: Second order cone programming approaches for handling missing and uncertain data. J. Mach. Learn. Res. **7**, 1283–1314 (2006)
36. Sklar, M.: Fonctions de répartition à n dimensions et leurs marges., Publications de l′Institut de Statistique de l′Université de Paris **8**, 229–231 (1959)
37. Sokolova, M., Japkowicz, N., Szpakowicz, S.: Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation. In: Australasian Joint Conference on Artificial Intelligence. Springer, Berlin, pp. 1015–1021 (2006)
38. Sturm, J.F.: Implementation of interior point methods for mixed semidefinite and second order cone optimization problems. Optim. Methods Softw. **17**(6), 1105–1154 (2002)
39. Sturm, J.F., Zhang, S.: Symmetric primal-dual path-following algorithms for semidefinite programming. Appl. Numer. Math. **29**(3), 301–315 (1999)
40. Tian, Y., Shi, Y., Liu, X.: Recent advances on support vector machines research. Technol. Econ. Dev. Econ. **18**(1), 5–33 (2012)
41. Trafalis, T.B., Gilbert, R.C.: Robust classification and regression using support vector machines. Eur. J. Oper. Res. **173**(3), 893–909 (2006)
42. Van Ackooij, W., de Oliveira, W.: Convexity and optimization with copula structured probabilistic constraints. Optimization **65**(7), 1349–1376 (2016)
43. Vapnik, V.N.: An overview of statistical learning theory. IEEE Trans. Neural Netw. **10**(5), 988–999 (1999)
44. Wang, X., Fan, N., Pardalos, P.M.: Robust chance-constrained support vector machines with second-order moment information. Ann. Oper. Res. **263**(1–2), 45–68 (2018)
45. Wang, X., Fan, N., Pardalos, P.M.: Stochastic subgradient descent method for large-scale robust chance-constrained support vector machines. Optim. Lett. **11**(5), 1013–1024 (2017)
46. Wang, X., Pardalos, P.M.: A survey of support vector machines with uncertainties. Ann. Data Sci. **1**(3–4), 293–309 (2014)
47. Xanthopoulos, P., Pardalos, P.M., Trafalis, T.B.: Robust Data Mining. Springer (2012)
48. Zhang, T.: Solving large scale linear prediction problems using stochastic gradient descent algorithms. In: Proceedings of the 21st International Conference on Machine Learning. ACM, pp. 116–123 (2004)
49. Zymler, S., Kuhn, D., Rustem, M.: Distributionally robust joint chance constraints with second-order moment information. Math. Program. **137**(1–2), 167–198 (2013)