Classification SVM Algorithms with Interval-Valued Training Data using Triangular and Epanechnikov Kernels

Lev V. Utkin
Saint Petersburg State Forest Technical University, Russia
lev.utkin@gmail.com

Anatoly I. Chekh
Saint Petersburg State Electrotechnical University, Russia
anatoly.chekh@gmail.com

Yulia A. Zhuk
Saint Petersburg State Forest Technical University, Russia
zhuk_yua@mail.ru

Abstract

Classification algorithms based on different forms of support vector machines (SVMs) for dealing with interval-valued training data are proposed in the paper. L_2-norm and L_∞-norm SVMs are used for constructing the algorithms. The main idea allowing us to represent the complex optimization problems as a set of simple linear or quadratic programming problems is to approximate the Gaussian kernel by the well-known triangular and Epanechnikov kernels. The minimax strategy is used to choose an optimal probability distribution from the set and to construct optimal separating functions.

Keywords. Classification, support vector machine, kernel, interval-valued data, minimax strategy, linear programming, quadratic programming, extreme points.

1 Introduction

The binary classification problem can be formally written as follows. Given n training data (examples, patterns) $S = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), ..., (\mathbf{x}_n, y_n)\}$, in which $\mathbf{x}_i \in \mathbb{R}^m$ represents a feature vector involving m features and $y_i \in \{-1, 1\}$ indices the class of the associated examples, the task of classification is to construct an accurate classifier $c : \mathbb{R}^m \to \{-1, 1\}$ that maximizes the probability that $c(\mathbf{x}) = y_i$ for $i = 1, ..., n$. Generally $\mathbf{x}_i$ may belong to an arbitrary set $\mathcal{X}$, but we consider the special case $\mathcal{X} = \mathbb{R}^m$ for simplicity. One of the ways for classification is to find a real valued separating function $f(\mathbf{x}, \mathbf{w}, b)$ having parameters $\mathbf{w}$ and b such that $\mathbf{w} = (w_1, ..., w_m) \in \mathbb{R}^m$ and $b \in \mathbb{R}$, for example, $f(\mathbf{x}, \mathbf{w}, b) = \langle \mathbf{w}, \mathbf{x} \rangle + b$. Here $\langle \mathbf{w}, \mathbf{x} \rangle$ denotes the dot product of two vectors $\mathbf{w}$ and $\mathbf{x}$. The sign of the function determines the class label prediction or $c(\mathbf{x})$. We also introduce the notation $x_i^{(k)}$ for the k-th element of the vector $\mathbf{x}_i$.

There are a lot of classification algorithms, but most of them are based on using a training set consisting of precise or point-valued data. However, training examples in many real applications can be obtained only in the interval form. Interval-valued data may result from imperfection of measurement tools or imprecision of expert information, from missing data. It should be noted that the interval-valued data can be regarded as a special case of a more general form of imprecise data. For example, we cannot observe some feature, but we know that the difference between values of the feature for data from different classes is less than some known value. In this case, we have imprecise training data.

Many classification algorithms have been presented for dealing with interval-valued data [11, 14, 17]. Most algorithms use an obvious approach when interval-valued observations are replaced by precise values based on some additional assumptions, for example, by taking middle points of intervals [12]. This approach is rather efficient when intervals are small and do not intersect each other. If intervals in training data are very large, then this approach may lead to incorrect classification.

One of the classification algorithms taking into account all points of intervals has been proposed by Utkin and Coolen [21]. However, this algorithm uses a weak assumption which restricts its usage. According to this assumption, the separating function f is monotone, for example, linear, because its lower and upper bounds in this case are determined only by the bounds of pattern intervals. However, in spite of the restricted application of the algorithm, it looks for "optimal" points to some extent of the expected classification risk, but not for points of intervals of training data. This is an important peculiarity of the algorithm. Similar approaches have been used by Hüllermeier [10], by Antonucci et al. [1] in their interesting classification algorithms under interval and fuzzy training data.

We propose a general approach for constructing robust classification algorithms dealing with imprecise training data which can be represented in the form of closed intervals or some compact convex sets of values

of training data. In contrast to the algorithms where intervals are replaced by points, the proposed algorithm searches for optimal precise points by applying the robust or maximin strategy of decision. In fact, we select a single probability distribution or a point in the interval of expected risk values in accordance with a certain decision strategy instead of points in intervals of training data.

We use the term robust in the sense defined by Xu et al. [26]. The robustness property means here reducing sensitivity of a classifier to incorrect replacement of intervals by point-valued analogues. There are different definitions of robustness. We use robustness stemmed from the robust optimization where a minimax optimization is performed over all possible values of intervals. This definition differs from robustness in statistics which studies how an estimator behaves under a small perturbation of the statistics model.

In order to construct new classification algorithms dealing with interval-valued training data, we propose to use the following three ideas:

1. Interval-valued observations produce a set of probability distributions such that the lower and upper expected classification risk measures can be determined in terms of the belief functions in a simple way.

2. There are many variants of SVMs. It is proposed to choose standard L_2-norm SVM. Moreover, it is proposed to use one of the L_∞-norm SVMs such that constraints in its dual form do not depend on vectors of observations $\mathbf{x}_i$, $i = 1, ..., n$. This allows us to solve the corresponding optimization problem by using extreme points of the polytope produced by the constraints.

3. It is proposed to replace the Gaussian kernel by the well-known triangular kernel and Epanechnikov kernel which can be regarded as two approximations of the Gaussian kernel. This replacement allows us to get a set of linear or quadratic optimization problems with variables $\mathbf{x}_i$ restricted by intervals $\mathbf{A}_i$, $i = 1, ..., n$.

It should be noted that the idea of approximating the Gaussian kernel by the triangular kernel in one-class classification problems has been studied by the authors [22]. This idea and other ones are exploited below for constructing new binary classification algorithms.

2 A Standard L_2-Norm SVM by Precise Data

Suppose we have training data $(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n) \in \mathbb{R}^m \times \{-1, +1\}$. Let ϕ be a feature map $\mathbb{R}^m \to G$ such that the data points are mapped into an alternative higher-dimensional feature space G. In other words, this is a map into an inner product space G such that the inner product in the image of ϕ can be computed by evaluating some simple kernel $K(\mathbf{x}, \mathbf{y}) = (\phi(\mathbf{x}), \phi(\mathbf{y}))$, such as the Gaussian kernel

$$K(\mathbf{x}, \mathbf{y}) = \exp\left(-\left\|\mathbf{x} - \mathbf{y}\right\|^2 / \sigma^2\right).$$

Here σ is the kernel parameter determining the geometrical structure of the mapped samples in the kernel space [24]. It is important to note that Gaussian kernels are very popular because they support many complex models and are rather flexible. Moreover, they show good features and strong learning capability [25].

The SVM minimizes the empirical risk measure

$$R = n^{-1} \sum_{i=1}^{n} l(\mathbf{x}_i),$$

as an approximation of the expected risk, which can be regarded as a bound depending on the so-called VC dimension introduced by Vapnik [23]. Here l is a loss function. The minimization of the above functional is an ill-posed problem because it admits an infinite number of solutions. In order to overcome this difficulty, regularization theory [19] provides a framework for solving the problem by adding appropriate constraints on the solution. This can be done by introducing a smoothness or penalty term $J(f)$ and a tuning "cost" parameter C which balances the tradeoff between the empirical risk measure and the penalty term. As a result, a general class of regularization problems has the form:

$$\min_f \left(C \sum l(\mathbf{x}_i) + J(f)\right).$$

Standard penalty terms are the L_s-norms such that $L_s = \|\mathbf{w}\|_s$, $s > 0$. In particular, the most popular penalty in the SVM classifier is $\|\mathbf{w}\|_2$. Hence, the SVM classifier can be represented in the form of the following convex optimization problem (the quadratic programming problem):

$$\min_{\xi, \mathbf{w}, b} R = \min_{\xi, \mathbf{w}, b} \left(\frac{1}{2} \|\mathbf{w}\|_2 + C \sum_{i=1}^{n} \xi_i\right), \qquad (1)$$

subject to

$$\xi_i \geq 0, \quad y_i \left(\langle \mathbf{w}, \phi(\mathbf{x}_i) \rangle + b\right) \geq 1 - \xi_i, \ i = 1, ..., n. \qquad (2)$$

Here ξ_i, $i = 1, ..., n$, are the slack variables. The quantity $C\xi_i$ is the "penalty" for any data point $\mathbf{x}_i$ that either lies within the margin on the correct side of the hyperplane ($\xi_i \leq 1$) or on the wrong side of the hyperplane ($\xi_i > 1$). The above optimization problem is obtained under condition that the so-called hinge loss function is used, i.e., $l(\mathbf{x}) = \max(0, 1 - y_i f\left\langle \mathbf{w}, \phi(\mathbf{x}_i) \right\rangle)$.

Instead of minimizing the primary objective function (1), a dual objective function, the so-called Lagrangian, can be formed of which the saddle point is the optimum. The dual programming problem is of the form:

$$\max_{\alpha} \left(\sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \right), \quad (3)$$

subject to

$$\sum_{i=1}^{n} \alpha_i y_i = 0, \ 0 \leq \alpha_i \leq C, \ i = 1, ..., n. \quad (4)$$

After substituting the obtained solution into the expression for the decision function f, we get the "dual" separating function

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b.$$

The above SVM is often called the L_2-norm SVM due to the definition of the regularization term. The parameter b is defined by using support vectors $\mathbf{x}_i$ from the following equation $b = y_j - \sum_{i=1}^{n} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}_j)$.

At the same time, there are other forms of the SVM defined by different L_s-norms of the penalty term. It turns out that the SVM with the L_∞-norm can be very useful when we deal with interval-valued data.

3 Interval-Valued Training Data and Belief Functions

Suppose we have training data $(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)$. We again have two classes, i.e., $y_i \in \{-1, 1\}$. However, in contrast to training data considered in the previous sections, $\mathbf{x}_i$ are interval-valued, i.e., $\mathbf{x}_i \in \mathbf{A}_i$, $i = 1, ..., n$. Here $\mathbf{A}_i = [\underline{a}_i^{(1)}, \overline{a}_i^{(1)}] \times ... \times [\underline{a}_i^{(m)}, \overline{a}_i^{(m)}]$, i.e., $\underline{a}_i^{(k)} \leq x_i^{(k)} \leq \overline{a}_i^{(k)}$, $k = 1, ..., m$; $\underline{a}_i^{(k)}$, $\overline{a}_i^{(k)}$ are bounds for values of the k-th feature in the i-th training example.

There are several ways in which one could deal with interval-valued data. In this paper, we consider the expected risk by interval-valued data in the framework of belief functions or Dempster-Shafer theory. Below, we give some basic definitions in the framework of belief functions.

Let $\mathcal{X}$ be a universal set under interest, usually referred to in evidence theory as the frame of discernment. Suppose n observations were made of an element $u \in \mathcal{X}$, each of which resulted in an imprecise (non-specific) measurement given by a set $\mathbf{A}$ of values. Let c_i denote the number of occurrences of the set $\mathbf{A}_i \subseteq \mathcal{X}$, and $\mathcal{P}o(\mathcal{X})$ the set of all subsets of $\mathcal{X}$ (power set of $\mathcal{X}$). A frequency function m, called basic probability assignment (BPA), can be defined such that [6, 16]:

$$m : \mathcal{P}o(\mathcal{X}) \to [0, 1], \quad m(\emptyset) = 0, \quad \sum_{\mathbf{A} \in \mathcal{P}o(\mathcal{X})} m(\mathbf{A}) = 1.$$

According to [6], this function can be obtained as follows:

$$m(\mathbf{A}_i) = c_i / n.$$

According to [16], the belief $Bel(A)$ and plausibility $Pl(\mathbf{A})$ of an event $\mathbf{A} \subseteq \mathcal{X}$ can be defined as

$$\mathrm{Bel}(\mathbf{A}) = \sum_{\mathbf{A}_i : \mathbf{A}_i \subseteq \mathbf{A}} m(\mathbf{A}_i),$$

$$\mathrm{Pl}(\mathbf{A}) = \sum_{\mathbf{A}_i : \mathbf{A}_i \cap \mathbf{A} \neq \varnothing} m(\mathbf{A}_i).$$

As pointed out in [9], a belief function can formally be defined as a function satisfying axioms which can be viewed as a weakening of the Kolmogorov axioms that characterize probability functions. Therefore, it seems reasonable to understand a belief function as a generalized probability function [6] and the belief $Bel(\mathbf{A})$ and plausibility $Pl(\mathbf{A})$ measures can be regarded as lower and upper bounds for the probability of $\mathbf{A}$, i.e., $\mathrm{Bel}(\mathbf{A}) \leq \Pr(\mathbf{A}) \leq \mathrm{Pl}(\mathbf{A})$. This implies that for a function $l(\mathbf{x})$, we can define the lower expectation $\underline{R}$ and the upper expectation $\overline{R}$ of the function $l(\mathbf{x})$ in the framework of belief functions as follows [13, 18]:

$$\underline{R} = \sum_{i=1}^{n} m(\mathbf{A}_i) \inf_{\mathbf{x}_i \in \mathbf{A}_i} l(\mathbf{x}_i),$$

$$\overline{R} = \sum_{i=1}^{n} m(\mathbf{A}_i) \sup_{\mathbf{x}_i \in \mathbf{A}_i} l(\mathbf{x}_i).$$

The above definition provides a simpler way for determining the bounds for the expected risk. By using the assumption accepted in the empirical expected risk, we can conclude that $m(\mathbf{A}_i) = 1/n$ for all $i = 1, ..., n$. Hence, we get

$$\underline{R} = \frac{1}{n} \sum_{i=1}^{n} \inf_{\mathbf{x}_i \in \mathbf{A}_i} l(\mathbf{x}_i), \quad \overline{R} = \frac{1}{n} \sum_{i=1}^{n} \sup_{\mathbf{x}_i \in \mathbf{A}_i} l(\mathbf{x}_i).$$

It follows from the above that we have the interval $[\underline{R}, \overline{R}]$ of the expected risk measure instead of its precise value. In order to use this interval in solving the classification problem, we have to determine a strategy of decision making which selects one point within this interval for searching optimal classification parameters $\mathbf{w}$, ξ and b in (1)-(2) or $\alpha_1, ..., \alpha_n$ in (3)-(4).

One of the well-known and popular ways for dealing with the interval of the expected risk is to use the minimax (pessimistic or robust) strategy. According to the minimax strategy, we select a probability distribution from the set of distributions such that the expected risk R achieves its maximum for fixed values of parameters. It should be noted that the "optimal" probability distributions may be different for different values of parameters. If to return to the interval $[\underline{R}, \overline{R}]$, then the minimax strategy assumes the largest risk, i.e., the upper bound $\overline{R}$. The minimax strategy can be explained in a simple way. We do not know a precise probability distribution and every distribution from their predefined set can be selected. Therefore, we should take the "worst" distribution providing the largest value of the expected risk. The minimax criterion can be interpreted as an insurance against the worst case because it aims at minimizing the expected loss in the least favorable case [15]. This criterion of decision making can be regarded as the well-known Γ-minimax [4, 7].

Robust algorithms have been exploited in classification problems due to the opportunity to avoid some strong assumptions underlying the standard classification algorithms. As pointed out by Xu et al. [26], the use of robust optimization in classification is not new. One of the popular robust classification algorithms is based on the assumption that inputs are subject to an additive noise, i.e., $\mathbf{x}_i^* = \mathbf{x}_i + \triangle\mathbf{x}_i$, where noise $\triangle\mathbf{x}_i$ is governed by a certain distribution. The simplest way for dealing with noise is to assume that every "true" data point is only known to belong to the interior of an Euclidean ball centered at the "nominal" data point $\mathbf{x}_i$ and each point can move around within the Euclidean ball. This algorithm has a very clear intuitive geometric interpretation [3]. One can see that the algorithm with interval-valued data and the robust algorithms [3, 26] are very close.

Finally, we can write the optimization problem for computing the optimal classification parameters ($\mathbf{w}$, ξ, b or α, b) as follows:

$$\overline{R} = \sup_{\mathbf{x}_i \in \mathbf{A}_i, i=1,...,n} \min_{\xi, \mathbf{w}, b} \sum_{i=1}^{n} l(\mathbf{x}_i),$$

4 L_2-Norm SVM by Interval-Valued Data

4.1 A General Problem and a New Kernel

Let us rewrite the objective function of problem (3)-(4) by taking into account interval-valued elements of the training set

$$\sup_{\mathbf{x}_i \in \mathbf{A}_i, i=1,...,n} \max_{\alpha} \times \left(\sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \right). \quad (5)$$

This is a nonlinear optimization problem whose solution is generally a hard problem. Therefore, we propose a method for its solution which can reduce this problem to a finite set of linear programming problems.

One of the ideas underlying the proposed algorithm is to approximate the Gaussian kernel $K(\mathbf{x}, \mathbf{y})$ by another kernel which could somehow simplify the optimization problem. It is proposed to introduce a new kernel function

$$K_1(\mathbf{x}, \mathbf{y}) = \max\{0, 1 - \|\mathbf{x} - \mathbf{y}\|^1 / \sigma^2\}, \quad (6)$$

This is the well-known triangular kernel. Its main peculiarity is that K_1 is linear. This peculiarity allows us to solve the above optimization problem.

Let us fix the values of α and write the dual optimization problem with the introduced kernel K_1 having optimization variables $\mathbf{x}_i \in \mathbf{A}_i$, $i = 1, ..., n$:

$$\inf_{\mathbf{x}_i, i=1,...,n} \left(\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j G_{ij} - \sum_{i=1}^{n} \alpha_i \right), \quad (7)$$

subject to

$$G_{ij} = \max\left\{0, 1 - \frac{\|\mathbf{x}_i - \mathbf{x}_j\|^1}{\sigma^2}\right\}, \ i, j = 1, ..., n, \quad (8)$$

$$\underline{a}_i^{(k)} \le x_i^{(k)} \le \overline{a}_i^{(k)}, \ k = 1, ..., m, \ i = 1, ..., n. \quad (9)$$

Here G_{ij} is a new variable such that $G_{ij} = K_1(\mathbf{x}_i, \mathbf{x}_j)$.

We do not add constraints (4) to the set of constraints (8)-(9) because the values of α are fixed, i.e., we consider the problem with variables $\mathbf{x}_i$, $i = 1, ..., n$. One can see from (7)-(9) that this problem is linear in case of the triangular kernel. According to some general results from linear programming theory, an optimal solution to the above problem is achieved at extreme

points or vertices of the polytope produced only by constraints (8)-(9). This is the first main feature of the proposed approach and the main reason for introducing the triangular kernels. Moreover, it can be seen from constraints (8)-(9) that they do not depend on variables α. This implies that the extreme points do not depend on α. This is the second feature which is used below. The linearity of the above problem and the independence of vertices of the polytope of variables α allow us to represent the initial optimization problem with objective function (5) as a finite set of standard quadratic programming problems which are formed by substituting extreme points $\mathbf{x}_i^*$ of the polytope produced by (8)-(9) into the kernel function $K_1(\mathbf{x}_i, \mathbf{x}_j)$ instead of $\mathbf{x}_i$.

We do not consider details of the optimization problem representation as a set of quadratic programming problems. However, we discuss about a set of extreme points $\mathbf{x}_i^*$, $i = 1, ..., n$. It is interesting to note that G_{ij} totally depends on $\mathbf{x}_i$, $i = 1, ..., n$. This implies that only constraints for $\mathbf{x}_i$ define the extreme points which are trivial and coincide with the bounds of intervals $\mathbf{A}_i, i = 1, ..., n$. Moreover, we do not need to represent constraints (8) in the form of standard inequalities. By enumerating the extreme points $\mathbf{x}_i^*$, we compute all values G_{ij} and substitute them into objective function (7). Finally, we have one of the standard quadratic programming problems corresponding to one combination of bounds of intervals $\mathbf{A}_i, i = 1, ..., n$, whose solution can be found, for example, by means of the packages "kernlab", "e1071", "wSVM" in the R-project.

The optimal values of α correspond to the *largest* value of objective function (7) over all extreme points $\mathbf{x}_i^*$. After substituting the obtained solution into the expression for the decision function f, we get

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b. \tag{10}$$

If we have $n^* \leq n$ interval-valued observations such that all their features are interval-valued, then we have to solve m^{n^*} quadratic programming problems. Of course, when n^* is rather large or the training examples are characterized by many interval-valued features m, then the obtained algorithm leads to extremely hard computations. Therefore, we propose below another classification algorithm whose complexity does not depend on the number of features m.

5 L_∞-Norm SVM

5.1 The Primal Form

We aim to find such a form of the SVM that would separate classification parameters, for example, $\alpha_1, ..., \alpha_n$, and intervals of $\mathbf{x}_1, ..., \mathbf{x}_n$. The SVM whose dual form satisfies this condition was proposed by Zhou et al. in [27]. It is based on using the L_∞-norm for writing the regularization term $\|\mathbf{w}\|$. The L_∞-norm leads to one of the possible variants of the SVM. Suppose that we have fixed precise values $\mathbf{x}_1, ..., \mathbf{x}_n$ from $\mathbf{A}_1, ..., \mathbf{A}_n$, respectively. According to [27], the optimization problem for computing the separating function parameters is of the form:

$$\min R = \min \left(-r + C \sum_{i=1}^{n} \xi_i \right), \tag{11}$$

subject to

$$y_j \left(\sum_{i=1}^{n} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}_j) + b \right) \geq r - \xi_j, \ j = 1, ..., n, \tag{12}$$

$$-1 \leq \alpha_i \leq 1, \ i = 1, ..., n, \tag{13}$$

$$r \geq 0, \ \xi_j \geq 0, \ j = 1, ..., n. \tag{14}$$

Here α_j, ξ_j, $j = 1, ..., n$, r, b are optimization variables; $C \geq 0$ is a constant. One can see that the separating function f is written in constraints in terms of Lagrange multipliers α_i (see (10)).

The authors of [27] show that the VC dimension in this case is bounded and the separating function f can be approached by minimizing the empirical expected risk measure. It is indicated in [27] that training SVMs is simpler than the L_2-norm SVMs, especially for large-scale problems.

5.2 The Dual Form

It should be noted that the SVM algorithm proposed by Zhou et al. in [27] is an interesting version of the SVM. However, its direct use does not help us in solving the classification problem with interval-valued data, which is viewed as a optimization problem with the objective function

$$R = \max_{\mathbf{x}_i} \min_{r, b, \alpha_j, \xi_j} \left(-r + C \sum_{i=1}^{n} \xi_i \right),$$

and constraints (12)-(14), $\mathbf{x}_i \in \mathbf{A}_i$, $i = 1, ..., n$.

Another advantage of the above SVM is very important for us. This is a special form of the dual problem which allows us to get a simple way for dealing with interval-valued data. Therefore, let us write the dual form for the above problem *by fixed* $\mathbf{x}_i$, $i = 1, ..., n$.

First of all, we replace the variables α_j in (11)-(14) by non-negative variables $a_j \geq 0$ and $c_j \geq 0$ in order to have only non-negative variables, i.e., $\alpha_j = a_j - c_j$. By using the standard method for constructing the

dual form, we get the following linear programming problem:

$$\max \sum_{i=1}^{n} (-g_i - h_i),$$

subject to $g_i, h_i \geq 0$,

$$\sum_{i=1}^{n} z_i \geq 1,\ 0 \leq z_j \leq C,\ j = 1, ..., n,$$

$$\sum_{i=1}^{n} z_i y_i = 0,$$

$$g_j - h_j = y_j \left(\sum_{i=1}^{n} z_i y_i K(\mathbf{x}_j, \mathbf{x}_i) \right),\ j = 1, ..., n.$$

Here $z = (z_1, ..., z_n)$, g_i, h_i, $i = 1, ..., n$, are optimization variables. By substituting the last constraint into the objective function, we get another objective function

$$\max \sum_{i=1}^{n} \left(-2g_i - y_i \left(\sum_{j=1}^{n} z_j y_j K(\mathbf{x}_i, \mathbf{x}_j) \right) \right).$$

Note that the maximum of the objective function is achieved when variable g_i is as small as possible, i.e., $g_i = 0$ for all $i = 1, ..., n$. Hence, we get the following simplified optimization problem

$$\min_{z} \sum_{i=1}^{n} y_i \left(\sum_{j=1}^{n} z_j y_j K(\mathbf{x}_i, \mathbf{x}_j) \right), \tag{15}$$

subject to

$$\sum_{i=1}^{n} z_i \geq 1,\ 0 \leq z_j \leq C,\ j = 1, ..., n, \tag{16}$$

$$\sum_{i=1}^{n} z_i y_i = 0. \tag{17}$$

At first glance, the above dual form of the optimization problem does not differ from the primal form from the viewpoint of its use. However, one can see that constraints of the dual form do not contain terms $K(\mathbf{x}_i, \mathbf{x}_j)$ and do not contain vectors $\mathbf{x}_i$. This is a very important feature of the dual form, which allows us to introduce interval-valued data into the SVM. It should be noted that the same property cannot be obtained by considering the standard SVM based on the L_1-norm or the L_2-norm. Therefore, problem (15)-(17) plays a key role in constructing the algorithm of classification with interval-valued data.

5.3 Extreme Points of the Polytope

If we assume that the values of $K(\mathbf{x}_i, \mathbf{x}_j)$ are precisely known, i.e., the values $\mathbf{x}_i$, $i = 1, ..., n$, are precise or fixed, then one of the ways for solving the linear programming problem (15)-(17) is to find the extreme points or vertices of the polytope produced by constraints (16)-(17) and denoted by $z^{(l)}$, $l = 1, ..., N$. Here N is the total number of extreme points. An optimal solution to the above problem is achieved at one of the extreme points.

Proposition 1 *Let n_- and n_+ be numbers of training examples in classes labelled $y = -1$ and $y = 1$, respectively. All extreme points of the polytope produced by constraints (16)-(17) can be divided into two subsets. The first subset consists of*

$$N_1 = \sum_{t=\lceil 1/2C \rceil}^{\min(n_-, n_+)} \binom{n_-}{t} \binom{n_+}{t}$$

extreme points such that every point contains t elements from every class equal to C and other elements are 0. Here t is an integer determined from the condition

$$\frac{1}{2C} < t \leq \min(n_-, n_+).$$

Let s be an integer determined from the condition

$$\frac{1}{2C} - 1 \leq s < \min\left(\frac{1}{2C}, n_-, n_+ \right).$$

If there exists $s \geq 0$, then the second subset consists of

$$N_2 = (n_- - s)(n_+ - s) \binom{n_-}{s} \binom{n_+}{s}$$

extreme points such that every point contains s (if there exists $s > 0$) elements from every class equal to C, one element from every class is $1/2 - sC$, other elements are 0.

Proposition 1 can be regarded as an extension of Proposition 5 in [20].

5.4 L_∞-Norm SVM by Interval-Valued Data

Let us rewrite the objective function of problem (15)-(17) by taking into account the interval-valued elements of the training set

$$\min_{l=1,...,N} \min_{\mathbf{x}_i \in \mathbf{A}_i, i=1,...,n} \sum_{i=1}^{n} \sum_{j=1}^{n} z_j^{(l)} y_i y_j K(\mathbf{x}_i, \mathbf{x}_j). \tag{18}$$

By having extreme points, we can replace the optimization problem (15)-(17) by a set of $N = N_1 + N_2$

(see Proposition 1) objective functions provided above. However, we cannot solve the obtained set of optimization problems with variables $\mathbf{x}_i \in \mathbf{A}_i$, $i = 1, ..., n$, in a simple way because the function $K(\mathbf{x}_i, \mathbf{x}_j)$ is nonlinear. Therefore, we again apply the idea of replacement the Gaussian kernel by its approximations. According to this idea, the Gaussian kernel can be approximated by another kernel which could somehow simplify the optimization problem. It is proposed to introduce two kernel functions

$$K_1(\mathbf{x}, \mathbf{y}) = \max\{0, 1 - \|\mathbf{x} - \mathbf{y}\|^1 / \sigma^2\},$$
$$K_2(\mathbf{x}, \mathbf{y}) = \max\{0, 1 - \|\mathbf{x} - \mathbf{y}\|^2 / \sigma^2\}.$$

Both the kernels can be regarded as approximations of the Gaussian kernel. The first one is the triangular kernel considered in the previous sections. The second kernel is known as the Epanechnikov kernel.

Let us fix the values of $z^{(l)} = (z_1^{(l)}, ..., z_n^{(l)})$ and write the dual optimization problem with the introduced kernels K_r, $r = 1, 2$, for the l-th extreme point $z^{(l)}$ of (16)-(17) as follows:

$$\min_{\mathbf{x}_i, i=1,...,n} \sum_{i=1}^{n} \sum_{j=1}^{n} z_j^{(l)} y_i y_j G_{ij}, \tag{19}$$

subject to

$$G_{ij} = \max\left\{0, 1 - \|\mathbf{x}_i - \mathbf{x}_j\|^r / \sigma^2\right\}, \ i, j = 1, ..., n, \tag{20}$$
$$\underline{a}_i^{(k)} \le x_i^{(k)} \le \overline{a}_i^{(k)}, \ k = 1, ..., m, \ i = 1, ..., n. \tag{21}$$

Here $x_i^{(j)}$ is the value of the j-th feature of the i-th example; G_{ij} is a new variable such that $G_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$; r is 1 or 2 if we use the triangular or Epanechnikov kernel, respectively.

Finally, we get the set of N linear programming problems in case of using the triangular kernel. In case of the Epanechnikov kernel, we have the same number of quadratically constrained linear programs (QCLPs). It can be numerically solved by means of several methods, for example, by using the sequential quadratic programming [5] which efficiently implemented by means of SNOP [8]. The optimal values of $\mathbf{x}_i$ correspond to the *smallest* value of objective function (19) over all extreme points $\mathbf{x}_i^*$.

It is interesting to note that the number N of optimization problems does not depend on the number of features m. This is an important peculiarity of the proposed algorithm, which allows us to apply the algorithm to application problems with many features.

The function $f(\mathbf{x})$ can be rewritten in terms of Lagrange multipliers as

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i^* y_i K(\mathbf{x}_i^*, \mathbf{x}) + b.$$

However, we do not know the optimal values of α_i because we used the dual optimization problem. Here we have two ways for computing the separating function. The first way is based on the fact that, by knowing the optimal solution z^* of the dual problem, the optimal solution α^* of the primal problem can be found by well-known algorithms. In particular, if the algorithm is implemented by using R-project, then the function "solveLP" in the package "linprog" has the output variable "con$dual" which provides the dual solution.

The second way is simpler. If we know precise optimal values $\mathbf{x}_i^*$ of intervals $\mathbf{A}_i$, $i = 1, ..., n$, then we can return to the initial problem (11)-(14) or to its dual form (15)-(17) and solve them by given fixed $\mathbf{x}_i^*$.

5.5 Comments about Constraints with the Triangular and Epanechnikov Kernels

It should be noted that constraints (20) are written in the short form. In order to solve the corresponding optimization problems, they have to be represented by the standard linear or quadratic inequalities. We do not consider in detail the representation of (20) because it is trivial due to the following two tricks.

First, the "standard" representation of (20) depends on the sign of the product $y_i y_j$. If $y_i y_j \ge 0$, then we get two constraints of the form:

$$G_{ij} \ge 1 - \|\mathbf{x}_i - \mathbf{x}_j\|^r / \sigma^2, \ G_{ij} \ge 0.$$

If $y_i y_j < 0$, then we use the well-known equation $\max(0, w) = w/2 + |w/2|$.

Second, in order to represent the absolute values, we use interesting results proposed by Beaumont [2]. According to [2], if we know some interval of values $[\underline{w}, \overline{w}] \subset \mathbb{R}$ of a variable w, then we can write $\forall w \in [\underline{w}, \overline{w}]$, $|w| \le uw + v$, where

$$u = \frac{|\overline{w}| - |\underline{w}|}{\overline{w} - \underline{w}}, \quad v = \frac{\overline{w}|\underline{w}| - \underline{w}|\overline{w}|}{\overline{w} - \underline{w}}.$$

6 Conclusion Remarks

New classification algorithms dealing with interval-valued training data have been proposed in the paper. A part of proposed algorithms using the triangular kernel instead of the Gaussian kernel comes to a finite set of simple linear programming problems whose solution does not meet difficulties. Another part using the triangular kernel comes to a finite set of quadratic programming problems whose solution are implemented by many standard procedures. The third part of algorithms is based on quadratically constrained linear programs which can be solved by using

the package "cplexAPI" available in several programming languages, for instance, in R-project.

It is important to note that the proposed algorithms indirectly find "optimal" points of intervals corresponding to the robust or maximin decision strategy. However, they fundamentally differ from the algorithm using some point-valued counterpart of intervals. The obtained "optimal" points of intervals are optimal in the sense that they maximize the expected classification error or risk if we apply the robust or maximin strategy. These "optimal" points compose a single probability distribution among a set of distributions produced by intervals in the framework of Dempster-Shafer theory.

Of course, all algorithms have a bottle neck which is their complexity. However, the proposed algorithms should not be used when a training set is large and intervals are rather small. Moreover, the algorithms based on the L_2-norm SVM should be used when the number of features is small. At the same time, the algorithms based on the L_∞-norm SVM do not depend on the number of features. It does not mean that the value m does not impact on the complexity of these algorithms. One can see from constraints (21) that the number of constraints strongly depends on m.

Finally, we have to stress on the main idea allowing us to construct the above algorithms. This is the replacement of the Gaussian kernel by the triangular and Epanechnikov kernels. This idea can be also used for constructing the support vector regression algorithms when dependent as well as independent variables are interval-valued.

Acknowledgement

The reported study was partially supported by RFBR, research project No. 15-01-01414-a.

References

[1] A. Antonucci, R. de Rosa, A. Giusti, and F. Cuzzolin. Temporal data classification by imprecise dynamical models. In *Proc. of the 8th International Symposium on Imprecise Probability: Theories and Applications*, pages 13–22, Compiegne, France, 2013. SIPTA.

[2] O. Beaumont. Solving interval linear systems with linear programming techniques. *Linear Algebra and Its Applications*, 281:293–309, 1998.

[3] A. Ben-Tal, L.E. Ghaoui, and A. Nemirovski. *Robust optimization*. Princeton University Press, Princeton, New Jersey, 2009.

[4] J.O. Berger. *Statistical Decision Theory and Bayesian Analysis*. Springer-Verlag, New York, 1985.

[5] P.T. Boggs and J.W. Tolle. Sequential quadratic programming. *Acta numerica*, 4:1–51, 1995.

[6] A.P. Dempster. Upper and lower probabilities induced by a multi-valued mapping. *Annales of Mathematical Statistics*, 38(2):325–339, 1967.

[7] I. Gilboa and D. Schmeidler. Maxmin expected utility with non-unique prior. *Journal of Mathematical Economics*, 18(2):141–153, 1989.

[8] P.E. Gill, W. Murray, and M.A. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM Journal on Optimization*, 12(4):979–1006, 2002.

[9] J.Y. Halpern and R. Fagin. Two views of belief: Belief as generalized probability and belief as evidence. *Artificial Intelligence*, 54(3):275–317, 1992.

[10] E. Hullermeier. Learning from imprecise and fuzzy observations: Data disambiguation through generalized loss minimization. *International Journal of Approximate Reasoning*, 55(7):1519–1534, 2014.

[11] H. Ishibuchi, H. Tanaka, and N. Fukuoka. Discriminant analysis of multi-dimensional interval data and its application to chemical sensing. *International Journal of General Systems*, 16(4):311–329, 1990.

[12] E.A. Lima Neto and F.A.T. de Carvalho. Centre and range method to fitting a linear regression model on symbolic interval data. *Computational Statistics and Data Analysis*, 52:1500–1515, 2008.

[13] H.T. Nguyen and E.A. Walker. On decision making using belief functions. In R.Y. Yager, M. Fedrizzi, and J. Kacprzyk, editors, *Advances in the Dempster-Shafer theory of evidence*, pages 311–330. Wiley, New York, 1994.

[14] P. Nivlet, F. Fournier, and J.-J. Royer. Interval discriminant analysis: An efficient method to integrate errors in supervised pattern recognition. In *Second International Symposium on Imprecise Probabilities and Their Applications*, pages 284–292, Ithaca, NY, USA, 2001.

[15] C.P. Robert. *The Bayesian Choice*. Springer, New York, 1994.

[16] G. Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, 1976.

[17] A. Silva and P. Brito. Linear discriminant analysis for interval data. *Computational Statistics*, 21:289–308, 2006.

[18] T.M. Strat. Decision analysis using belief functions. *International Journal of Approximate Reasoning*, 4(5):391–418, 1990.

[19] A.N. Tikhonov and V.Y. Arsenin. *Solution of Ill-Posed Problems*. W.H. Winston, Washington DC, 1977.

[20] L.V. Utkin. A framework for imprecise robust one-class classification models. *International Journal of Machine Learning and Cybernetics*, 5(3): 379-393, 2014.

[21] L.V. Utkin and F.P.A. Coolen. Interval-valued regression and classification models in the framework of machine learning. In F. Coolen, G. de Cooman, Th. Fetz, and M. Oberguggenberger, editors, *Proc. of the Seventh Int. Symposium on Imprecise Probabilities: Theories and Applications, ISIPTA'11*, pages 371–380, Innsbruck, Austria, 2011. SIPTA.

[22] L.V. Utkin, Y.A. Zhuk, and A.I. Chekh. A robust one-class classification model with interval-valued data based on belief functions and minimax strategy. In Petra Perner, editor, *Machine Learning and Data Mining in Pattern Recognition*, volume 8556 of *Lecture Notes in Computer Science*, pages 107–118. Springer, 2014.

[23] V. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.

[24] J. Wang, H. Lu, K.N. Plataniotis, and J. Lu. Gaussian kernel optimization for pattern classification. *Pattern Recognition*, 42(7):1237 – 1247, 2009.

[25] W. Wang, Z. Xu, W. Lu, and X. Zhang. Determination of the spread parameter in the Gaussian kernel for classification and regression. *Neurocomputing*, 55(3):643–663, 2003.

[26] H. Xu, C. Caramanis, and S. Mannor. Robustness and regularization of support vector machines. *The Journal of Machine Learning Research*, 10(7):1485–1510, 2009.

[27] Weida Zhou, Li Zhang, and Licheng Jiao. Linear programming support vector machines. *Pattern Recognition*, 35(12):2927–2936, 2002.