The Research of Algorithm for Data Mining Based on Fuzzy Theory

Wang Aimin[1], Li Jie[1, 2]
School of Computer and Information Engineering
Anyang Normal University
Anyang, China
[2]Shenyang Institute of Computing Technology
Chinese Academy of Sciences
Shenyang, China
wam508@126.com, ljjacky@vip.sina.com

Journal of Digital
Information Management

ABSTRACT: *Data Mining is a new filed in data processing research. Support Vector Machine (SVM) is one of the new methods using in data mining, which has gained great applicable success. However, there are still plenty of limitations in SVM. For example, SVM won't work if its training set contains uncertain information. In order to solve the problem presented above, this paper discusses the constraining programming of fuzzy chance and the characteristic of fuzzy classification as well as its expression methods. The algorithm for classifying Support Vector Machine is also included in this paper.*

Categories and Subject Descriptors:
H.2.8 [Database Applications]: Data Mining; **I.2.3 [Deduction and Theorem Proving]:** Fuzzy

General Terms:
Data Mining, Fuzzy Theory, Mining Algorithm

Keywords: Data Mining, Fuzzy Programming, Fuzzy Classification, Fuzzy Linear Separable, Algorithm

Received: 18 April 2013, Revised 6 June 2013, Accepted 12 June 2013

1. Introduction

With the advent of the information age, humankind is faced with more and more information processing problems such as data storage, organization and searches. These problems gradually increase in complexity in the hierarchy. Their scales of space activities are growing. It is faster in time scale and more extensive and far-reaching in consequences and implications. Data mining is a new field in data processing. It is a process that digs out various models, summary and export values from the known data sets. Data mining is an interdisciplinary that involves database technology, machine learning, statistics, neural networks, knowledge engineering, and high-performance calculation and so on. It has been widely applied in industry, agriculture, business, economics, health and many other industries.

Support Vector Machine (SVM) is one of the new methods using in data mining. It was proposed by Vapnik et al [1] [2] [3]. Using SVM, the problem of classification and regression can be deal with better. SVM has been a research focus in machine learning and was applied in many fields successfully. But there are many limitations in SVM. For example, when the training sets of SVM contain uncertain information, SVM will be incapable. In 2002, Chunfu Lin and Shengde Wang, the professors of Taiwan University put forward Fuzzy Support Vector Machine (FSVM) method with Shigeo Abe and Takuya Inoue, the professors of Kobe University. They made some improvements to SVM, but did not build FSVM on algorithm level. Their work lacks of the research on SVM that contains uncertain information.

To solve the problem of SVM containing uncertain information (fuzzy parameters) in common conditions, we discuss the constraining programming of fuzzy chance and the characteristic of fuzzy classification as well as its expression methods. The algorithm for classifying Support Vector Machine is also included in this paper.

2. Preliminary Knowledge

The research of this paper is carried on in the given possible space which is mentioned in [6].

2. 1 Fuzzy Chance-constrained Programming

Definition 2.1 Suppose $\tilde{A}$ is a fuzzy subset in universe of discourse U. If $U = R$ (set of real number) and $\tilde{A}$ is a regular closed convex fuzzy set, $\tilde{A}$ is called fuzzy number, written $\tilde{a}$.

Definition 2.2 Suppose $\tilde{a}$ is a fuzzy number. If $\tilde{a}$'s membership function:

$$\mu_{\tilde{a}}(x) = \begin{cases} \frac{x - r_1}{r_2 - r_1}, r_1 \leq x \leq r_2 \\ 1, x = r_2 \\ \frac{x - r_3}{r_2 - r_3}, r_2 \leq x \leq r_3 \end{cases}$$

$r_1 \leq r_2 \leq r_3, r_j \in R\ (j = 1, 2, 3),$

$\tilde{a}$ is called triangular fuzzy number, written $\tilde{a} = (r_1, r_2, r_3)$. Real number r_2 and r_1, r_3 are called triangular fuzzy number $\tilde{a}$'s center, left and right endpoints. Center is the main location of triangular fuzzy number. Real number a can be expressed as a special triangular fuzzy number $a = (a, a, a)$.

Definition 2.3 Suppose $f : R \times R \rightarrow R$ is a binary function in real number field. $\tilde{a}$, $\tilde{b}$ are fuzzy numbers. The membership function of fuzzy number $\tilde{c} = f(\tilde{a}, \tilde{b})$ can be defined as:

$$\mu_{\tilde{c}}(z) = \sup\{ \min(\mu_{\tilde{a}}(x), \mu_{\tilde{b}}(x)) \mid x, y \in R, z = f(x, y)\}, z \in R$$

Theorem 2.1 $\tilde{a} = (r_1, r_2, r_3)$ and $\tilde{b} = (t_1, t_2, t_3)$ are triangular fuzzy numbers. ρ is a real number. Then:

$$\tilde{a} + \tilde{b} = (r_1 + t_1, r_2 + t_2, r_3 + t_3); \qquad (1)$$

$$\rho\tilde{a} = \begin{cases} (\rho r_1, \rho r_2, \rho r_3), \rho \geq 0 \\ (\rho r_3, \rho r_2, \rho r_1), \rho < 0 \end{cases}. \qquad (2)$$

Proof: The conclusion can be directly deduced from Definition 2.3.

Theorem 2.2 Suppose $\tilde{a} = (r_1, r_2, r_3)$ is triangular fuzzy numbers. For any given confidence level $\lambda\,(0 < \lambda \leq 1)$, $Pos\,\{\tilde{a} \leq 0\} \geq \lambda \Leftrightarrow \}(1 - \lambda)\, r_1 + \lambda r_2$.

Proof: If $Pos\,\{\tilde{a} \leq 0\} \geq \lambda$, there must be an established between $r_2 \leq 0$ and $\frac{r_1}{r_1 - r_2} \geq \lambda$. If $r_2 \leq 0$, then $r_1 < r_2 \leq 0$, so $(1-\lambda)\, r_1 + \lambda r_2 \leq 0$. If, $\frac{r_1}{r_1 - r_2} \geq \lambda$, $r_1 \leq \lambda\,(r_1 - r_2)$ due to $r_1 < r_2$, i.e $(1-\lambda)\, r_1 + \lambda r_2 \leq 0$.

On the contrary, there are two cases when $(1-\lambda)\, r_1 + \lambda r_2 \leq 0$. When $r_2 \leq 0$, $Pos\,\{\tilde{a} \leq 0\} = 1$ i.e. $Pos\,\{\tilde{a} \leq 0\} \geq \lambda$; when $r_2 > 0$, $r_1 - r_2 < 0$, $\frac{r_1}{r_1 - r_2} \geq \lambda$ i.e $Pos\,\{\tilde{a} \leq 0\} \geq \lambda$ due to $(1-\lambda)\, r_1 + \lambda r_2 \leq 0$.

Definition 2.4 Programming

$$\begin{cases} \min \bar{f} \\ s.t.\ Pos\{ f(x, \xi) \leq \bar{f}\} \geq \beta \\ Pos\{ g_j(x, \xi') \leq 0, j = 1, 2, ..., p\} \geq \alpha \end{cases}$$

is called fuzzy chance-constrained programming [8]. In the programming, x is decision variable. ξ, ξ' are fuzzy parameter vectors. $f(x, \xi)$ is objective function. $g_j(x, \xi') \leq 0$, $j = 1, 2, ..., p$ is constraint conditions. $\alpha, \beta\,(0 < \alpha, \beta \leq 1)$ are the given confidence levels of constraint condition and objective function. $Pos\,\{.\}$ is the possibility measure of fuzzy event $\{.\}$.

Common fuzzy chance-constrained programming is

$$\begin{cases} \min f(x) \\ s.t.\ Pos\{ h(x)\,\xi) + c \leq 0\}\, f\} \geq \lambda \end{cases} \qquad (2-1)$$

$f(x)$ is the function of decision variable x. It is an objective function that does not contain fuzzy parameters. $h(x)$ is the function of decision variable x. ξ is fuzzy number. c is real number. $h(x)\ \xi + c \leq 0$ is the constraint condition. $(0 < \lambda \leq 1)$ is the given confidence level.

In [8], a solution of fuzzy chance-constrained programming under normal circumstances was put forward. It converts fuzzy chance-constrained programming to clearly equivalent programming. But it does not available to the fuzzy chance-constrained programming described as (2-2). In this paper, we provide a new solution of fuzzy chance-constrained programming described as (2-2).

Theorem 2.3 If ξ in fuzzy chance-constrained programming equation (2-1) is triangular fuzzy number i.e. $\xi = (r_1, r_2, r_3)$, clearly equivalent programming of equation (2-1) is:

$$\begin{cases} \min f(x) \\ s.t. (1 - \lambda)\,[r_1 h^+(x) - r_3 h^-(x)] + \lambda r_2 h(x) + c \leq 0 \end{cases} \qquad (2-2)$$

$$h^+(x) = \begin{cases} h(x), h(x) \geq 0 \\ 0, h(x) < 0 \end{cases}, \quad h^-(x) = \begin{cases} 0, h(x) \geq 0 \\ -h(x), h(x) < 0 \end{cases}.$$

It is an ordinary programming that equals to fuzzy chance-constrained programming described as (2-1).

Proof: First, we prove the clear equivalent class of $Pos\{h(x)\,\xi + c \leq 0\} \geq \lambda$ is:

$$(1 - \lambda)\,[r_1 h^+(x) - r_3 h^-(x)] + \lambda r_2 h(x) + c \leq 0$$

From the calculation of triangular fuzzy number, $h(x)\xi + c$ is still a triangular fuzzy number. Set

$$h^+(x) = \begin{cases} h(x), h(x) \geq 0 \\ 0, h(x) < 0 \end{cases},$$

$$h^-(x) = \begin{cases} 0, h(x) \geq 0 \\ -h(x), h(x) < 0 \end{cases}.$$

Then $h^+(x)$, $h^-(x)$ are nonnegative and $h(x) = h^+(x) - h^-(x)$.

So

$$h(x)\xi + c = (h^+(x) - h^-(x))\xi + c = h^+(x)\xi - h^-(x)\xi + c$$
$$= (h^+(x)r_1, h^+(x)r_2, h^+(x)r_3) + (h^-(x)r_1, h^-(x)r_2, h^-(x)r_3) + c$$
$$= (h^+(x)r_1 - h^-(x)r_3 + c, h^+(x)r_2 - h^-(x)r_2 + c, h^+(x)r_3 - h^-(x)r_1 + c$$
$$= (h^+(x)r_1 - h^-(x)r_3 + c, (h^+(x)r_2 - h^-(x))r_2 + c, h^+(x)r_3 - h^-(x)r_1 + c$$
$$= (h^+(x)r_1 - h^-(x)r_3 + c, h(x)r_2 + c, h^+(x)r_3 - h^-(x)r_1 + c$$

From Theorem 2.2, the clear equivalent class of $Pos\{h(x)\xi + c\} \geq \lambda\,(0 < \lambda \leq 1)$ is

$$(1-\lambda)\,[r_1 h^+(x) - r_3 h^-(x)] + \lambda r_2 h(x) + c \leq 0$$

Because fuzzy chance-constrained programming equation (2-1) and ordinary programming equation (2-2) have the same objective function, and their constraint conditions are equal, fuzzy chance-constrained programming equation (2-1) is equal to ordinary programming equation (2-2).

2.2 Characteristics of Fuzzy Classification

First, we give the definition of classification.

Definition 2.5 Find a rule for training set $S = \{(x_1, \tilde{y}_1),...,(x_l, \tilde{y}_l)\}$ ($x_j \in R^n$, $\tilde{y}_j$ is fuzzy number that means its fuzzy categories $j = 1,..., l$) of fuzzy number according to the output of given training points to deduce the corresponding fuzzy number $\tilde{y}$ (means fuzzy categories of x) of any mode x. We call this problem fuzzy classification.

The integral fuzzy information studied in this paper has three fuzzy characteristics: fuzzy positive class (sample points' positive degree of membership are more than negative one), fuzzy negative class (sample points' negative degree of membership are more than positive one) and center class (sample points' positive degree of membership and negative one are equal).

Suppose sample points' positive or negative degrees of membership are δ^+ or δ^-, $\delta^+, \delta^- \in [0.5, 1]$. For convenience, we use $\delta \in [-1, -0.5] \cup [0.5, 1]$, make $\delta^+ = \delta$, $\delta^- = -\delta$. So, three fuzzy characteristics can be described by a special triangular fuzzy number as follow:

$$\tilde{y} = (r_1, r_2, r_3) =$$
$$\begin{cases} \left(\dfrac{2\delta^2 + \delta - 2}{2}, 2\delta - 1, \dfrac{2\delta^2 - 3\delta + 2}{2}\right), 0.5 \leq \delta \leq 1 \\ \left(\dfrac{2\delta^2 - 3\delta + 2}{2}, 2\delta + 1, \dfrac{2\delta^2 - \delta - 2}{2}\right), -1 \leq \delta \leq -0.5 \end{cases} \quad (2\text{-}3)$$

(1) **Example of fuzzy positive class:** sample points' positive degree of membership is 0.9, and its negative degree of membership is 0.1. This fuzzy characteristic can be described by triangular fuzzy number $\tilde{a} = (0.58, 0.8, 1.02)$. In this equation, 0.8 is the center of $\tilde{a}$.

(2) **Example of fuzzy negative class:** sample points' negative degree of membership is 0.8, and its positive degree of membership is 0.2. This fuzzy characteristic can be described by triangular fuzzy number $\tilde{b} = (-1.1, -0.6, -0.1)$. In this equation, 0.6 is the center of $\tilde{b}$.

(3) **Example of center class:** sample points' positive degree of membership is 0.5, and its negative degree of membership is 0.5. This fuzzy characteristic can be described by triangular fuzzy number $\tilde{c} = (-2, -0, 2)$. In this equation, 0 is the center of $\tilde{c}$.

3. FSVM

Suppose the training set is $S = \{(x_1, \tilde{y}_1),...,(x_l, \tilde{y}_l)\}$ (3-1). $x_j \in R^n$, $\tilde{y}_j$ is a triangular fuzzy number as equation (2-3), $j = 1,..., l$. $(x_j, \tilde{y}_j)$ $(j = 1,..., l)$ is fuzzy training points. S is a fuzzy training set.

Definition 3.1 In equation (2-3), if $\delta \in (0.5, 1]$, the corresponding fuzzy training points are fuzzy positive points. If $\delta \in [-1, -0.5)$, the corresponding fuzzy training points are fuzzy negative points.

For simplicity, we do not consider the situation of $\delta = 0.5$ or $\delta = -0.5$. Because the triangular fuzzy number $\tilde{y} = (-2, 0, 2)$ does not provide positive or negative class information in this situation.

To facilitate our research, we reorder the fuzzy training points in the set. Put the fuzzy positive class points in front, and negative ones on back. Thus, we get the fuzzy training set:

$$S = \{(x_1, \tilde{y}_1),..., (x_p, \tilde{y}_p), (x_{p+1}, \tilde{y}_{p+1}),..., (x_l, \tilde{y}_l)\} \quad (3\text{-}2)$$

In equation (3-2), $(x_t, \tilde{y}_t)$ are fuzzy positive class points $t = 1,..., p$. $(x_i, \tilde{y}_i)$ are fuzzy negative class points $i = p + 1,..., l$.

Definition 3.2 Suppose the fuzzy training set is described as equation (3-2), for the given confidence level $\lambda\,(0 < \lambda \leq 1)$, if it has $w \in R^n$, $b \in R$ to make:

$$Pos\{\tilde{y}_j((w \cdot (x_j) + b) \geq 1\} \geq \lambda, \; j = 1,..., l, \quad (3\text{-}3)$$

We call training set (3-2) fuzzy linear separable in the confidence level λ. Now, we also call the question of fuzzy classification fuzzy linear separable under confidence level.

As we all know, the basic condition of fuzzy linear separable is the inputs of fuzzy positive class points and negative ones are separated by possibility $\lambda\,(0 < \lambda \leq 1)$. If the outputs of fuzzy training points are all 1 or –1, the fuzzy training set will degenerate to ordinary training set. Now, the linear separable of fuzzy training set becomes the linear separable of ordinary one.

The fuzzy linear separable of fuzzy training set is the extension of ordinary sets' linear separable. In fuzzy training set, if only we choose an appropriate confidence level, the fuzzy training set is still linear separable, although some training points with smaller membership are misclassified in the set.

For example, this is shown as Figure 1:

Suppose there are three fuzzy training points. Every point's input is one-dimensional. The outputs of fuzzy training points (x_2, y_2) and (x_3, y_3) are determinate $y_2 = 1$ and $y_3 = -1$. The membership degree of the first fuzzy training point that belongs to negative class is δ_1^-. Suppose $\delta_1^- = 0.51$, $0.6, 0.7$.

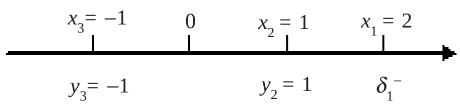

Figure 1. The shown of fuzzy linear separable

(i) When $\delta_1^- = 0.51$, according to equation (2-3), the corresponding triangular fuzzy number $\tilde{y}_1 = (-1.94, -0.02, 1.9)$. So, the fuzzy training set is:

$$S = \{(x_1, \tilde{y}_1), (x_2, y_2), (x_3, y_3)\}$$

Choose $\lambda = 0.72$. If use separating hyper-plane $x = 0$, $(w.\, x_1) + b = 2$. So

$$Pos\{(\tilde{y}_1((w.\, x_1) + b) \geq 1\} = 0.722 > 0.7$$

And

$$Pos\{(\tilde{y}_3((w.\, x_3) + b) \geq 1\} = 1 > 0.7$$

$$Pos\{(\tilde{y}_3((w.\, x_3) + b) \geq 1\} = 1 > 0.7$$

So, fuzzy training set S is linear separable under confidence level $\lambda = 0.72$.

(ii) When $\delta_1^- = 0.6$, the corresponding triangular fuzzy number $\tilde{y}_1 = (-1.53, -0.2, 1.13)$.

Choose $\lambda = 0.47$, similarly, fuzzy training set is linear separable under confidence level $\lambda = 0.47$.

(iii) When $\delta_1^- = 0.7$, the corresponding triangular fuzzy number $\tilde{y}_1 = (-1.26, -0.4, 0.46)$.

Choose $\lambda = 0.08$, similarly, fuzzy training set S is linear separable under confidence level $\lambda = 0.08$.

The problem of fuzzy classification has three types: fuzzy linear separable (all the fuzzy training points meet the condition of definition 3.2); approximate fuzzy linear separable (most of the fuzzy training points meet the condition of definition 3.2); fuzzy nonlinear (most of the fuzzy training points do not meet the condition of definition 3.2).

Theorem 3.1 Under confidence level $\lambda (0 < \lambda \leq 1)$, the clear equivalence class of equation (3-3) is:

$$\begin{cases} ((1-\lambda)\, r_{t3} + \lambda r_{t2})\, ((w.\, x_t) + b) \geq 1,\ t = 1,..., p \\ ((1-\lambda)\, r_{i1} + \lambda r_{i2})\, ((w.\, x_i) + b) \geq 1,\ i = p+1,..., l \end{cases} \quad (3\text{-}4)$$

Proof: for any $j = 1,..., l$, set $h_j(w, b) = -((w.\, x_j) + b)$,

$$h_j^+(w, b) = \begin{cases} 0, (w.\, x_j) + b > 0 \\ -((w.\, x_j) + b), (w.\, x_j) + b \leq 0 \end{cases}$$

$$h_j^-(w, b) = \begin{cases} (w.\, x_j) + b, (w.\, x_j) + b > 0 \\ 0, (w.\, x_j) + b \leq 0 \end{cases}$$

According to theorem 2.3, get the clear equivalence class of $Pos\{(\tilde{y}_j((w.\, x_3) + b) \geq 1, j = 1,..., l\} = Pos\{(1 - \tilde{y}_j((w.\, x_j) + b) \leq 0\} = 1, j = 1,..., l\} \geq \lambda$ is:

$$(1-\lambda)\,[\, r_{j1} h_j^+(w, b) - r_{j3} h_j^-(w, b)] + \lambda r_{j2} h_j(w, b) + 1 \leq 0,\ j = 1) \quad (3\text{-}5)$$

Under confidence level λ $(0 < \lambda \leq 1)$, equation (3-5) can be written:

$$\begin{cases} ((1-\lambda)\, r_{t3} + \lambda r_{t2})\, ((w.\, x_t) + b) \geq 1,\ t = 1,..., p \\ ((1-\lambda)\, r_{i1} + \lambda r_{i2})\, ((w.\, x_i) + b) \geq 1,\ i = p+1,..., l \end{cases}$$

That is to say, under confidence level $\lambda (0 < \lambda \leq 1)$, the clear equivalence class of equation (3-3) is equation (3-4).

In equation (3-4), set

$$k_t = \frac{1}{(1-\lambda)\, r_{t3} + \lambda r_{t2}},\ t = 1,..., p$$
$$l_i = \frac{1}{(1-\lambda)\, r_{i1} + \lambda r_{i2}},\ i = p+1,..., l \quad (3\text{-}6)$$

Then (3-5) can be expressed as

$$\begin{cases} (w.\, x_t) + b \geq k_t,\ t = 1,..., p \\ (w.\, x_i) + b \leq l_i,\ i = p+1,..., l \end{cases}$$

Definition 3.3 Consider the fuzzy linear separable problem given by fuzzy training set equation (3-3). We call two parallel hyper-planes $(w.\, x) + b = k^+$ and $(w.\, x) + b = l^-$ supporting hyper-planes about fuzzy training set (3-3). If:

$$\begin{cases} (w.\, x_t) + b \geq k_t,\ t = 1,..., p \\ \min\limits_{t = 1,...p}\{((w.\, x_t) + b\} = k^+ \\ (w.\, x_i) + b \leq l_i,\ i = p+1,..., l \\ \max\limits_{i = p+1,...\, l}\{((w.\, x_t) + b\} = l^- \end{cases} \quad (3\text{-}7)$$

$k_t\,(t = 1,..., p)$, $l_i\,(i = 1,..., p)$ is shown as equation (3-7). $k^+ = \min\limits_{t = 1,...p}\{k_t\}$, $l^- = \max\limits_{i = p+1,...\, l}\{l_i\}$.

Obviously, the pair of supporting hyper-planes about fuzzy training set (3-3) is unique.

Distance between the pair of supporting hyper-planes $(w.\, x) + b = k^+$ and $(w.\, x) + b = l^-$ are $\frac{|\, k^+ - l^-|}{\| w \|}$. We call it interval ($k^+ > 0$ and $l^- < 0$ are constants).

4. Fuzzy Linear Separable FSVMS

Now we consider the linear separable problem of fuzzy training set described as equation (3-2). Under confidence level λ ($0 < \lambda \le 1$), the problem of fuzzy classification transform into fuzzy chance-constraint programming question with decision variable $(w, b)^T$:

$$\begin{cases} \min_{w,b} \frac{1}{2} \| w \|^2 \\ s.t\, Pos\, \{\tilde{y}_j ((w. x_j) + b) \ge 1\} \ge \lambda\,, j = 1,..., l \end{cases} \quad (4\text{-}1)$$

In equation (4-1), $\tilde{y}_j$ ($j = 1,..., l$) is the triangular fuzzy number of training set described as equation (3-2). $Pos\{.\}$is possibility measure of fuzzy event$\{.\}$.

Theorem 4.1 Under confidence level $\lambda (0 < \lambda \le 1)$, the clear equivalent programming of uncertain chance-constraint programming described as equation (4-1) is the following quadratic programming:

$$\begin{cases} \min_{w,b} \frac{1}{2} \| w \|^2 \\ s.t ((1-\lambda)\, r_{t3} + \lambda r_{t2})\, ((w. x_t) + b) \ge 1, t = 1,..., p \\ ((1-\lambda)\, r_{i1} + \lambda r_{i2})\, ((w. x_i) + b) \ge 1, i = p+1,..., l \end{cases} \quad (4\text{-}2)$$

Proof: From the conclusion of Theorem 3.1, we can draw this conclusion.

In [1], we know the optimal solution of programming (4-2) exists. Now, we will solve the dual programming of quadratic programming described as equation (4-2).

Theorem 4.2 the dual programming of quadratic programming described as equation (4-2) is:

$$\begin{cases} \min_{w,b} \frac{1}{2} (A + 2B + C) - (\sum_{t=1}^{p} \beta_t + \sum_{i=p+1}^{l} \alpha_i) \\ s.t. \sum_{t=1}^{p} \beta_t ((1-\lambda)\, r_{t3} + \lambda r_{t2}) + \sum_{i=p+1}^{l} \alpha_i ((1-\lambda)\, r_{i1} + \lambda r_{i2}) = 0 \\ \beta_t \ge 0, t = 1,..., p \\ \alpha_i \ge 0, i = p+1,..., l \end{cases} \quad (4\text{-}3)$$

In equation (4-3):

$$A = \sum_{t=1}^{p} \sum_{s=1}^{p} \beta_t \beta_s ((1-\lambda)\, r_{t3} + \lambda r_{t2})\, ((1-\lambda)\, r_{s3} + \lambda r_{s2})\, (x_t \cdot x_s)$$

$$B = \sum_{t=1}^{p} \sum_{i=p+1}^{l} \beta_t \alpha_i ((1-\lambda)\, r_{t3} + \lambda r_{t2})\, ((1-\lambda)\, r_{i1} + \lambda r_{i2})\, (x_t \cdot x_i)$$

$$C = \sum_{i=p+1}^{l} \sum_{q=p+1}^{l} \alpha_i \alpha_q ((1-\lambda)\, r_{i1} + \lambda r_{i2})\, ((1-\lambda)\, r_{q1} + \lambda r_{i2})\, (x_i \cdot x_q)$$

(t, s is the subscript of fuzzy positive class points, i, q is the subscript of fuzzy negative class points), $\beta = (\beta_1,..., \beta_p)^T$, $\alpha = (\alpha_{p+1},..., \alpha_l)^T$, $(\beta, \alpha)^T$ are decision variables.

Proof: First, we induct Lagrange function

$$L(w, b, \beta, \alpha) = \frac{1}{2} \| w \|^2 - \sum_{t=1}^{p} \beta_t (((1-\lambda)\, r_{t3} + \lambda r_{t2})\, ((w. x_t) + b) - \sum_{i=p+1}^{l} \alpha_i ((1-\lambda)\, r_{i1} + \lambda r_{i2})\, ((w. x_i) + b) - 1$$

In equation (4-4), $\beta = (\beta_1,..., \beta_p)^T \in R_+^p$, $\alpha = (\alpha_{p+1},..., \alpha_l)^T \in R_+^{l-p}$, β_t, α_i are Lagrange multipliers.

According to the definition of Wolfe antithesis, we solve the minimum of Lagrange function about w, b. From the condition of extreme value:

$$\nabla_b L(w, b, \beta, \alpha) = 0, \nabla_w L(w, b, \beta, \alpha) = 0$$

Get

$$\sum_{t=1}^{p} \beta_t ((1-\lambda)\, r_{t3} + \lambda r_{t2}) + \sum_{i=p+1}^{l} \alpha_i ((1-\lambda)\, r_{i1} + \lambda r_{i2}) = 0 \quad (4\text{-}5)$$

$$w = \sum_{t=1}^{p} \beta_t ((1-\lambda)\, r_{t3} + \lambda r_{t2})\, x_t + \sum_{i=p+1}^{l} \alpha_i ((1-\lambda)\, r_{i1} + \lambda r_{i2})\, x_i \quad (4\text{-}6)$$

Substitute equation (4-6) into Lagrange function described as equation (4-4), and use equation (4-5), we can get the dual programming of quadratic programming described as equation (4-2):

$$\begin{cases} \max_{\beta, \alpha} (\sum_{t=1}^{l} \beta_t + \sum_{i=p+1} \alpha_i) - \frac{1}{2} (A + 2B + C) \\ s.t. \sum_{t=1}^{p} \beta_t ((1-\lambda)\, r_{t3} + \lambda r_{t2}) + \sum_{i=p+1}^{l} \alpha_i ((1-\lambda)\, r_{i1} + \lambda r_{i2}) = 0 \\ \beta_t \ge 0, t = 1,..., p \\ \alpha_i \ge i = p+1,..., l \end{cases} \quad (4\text{-}7)$$

In equation (4-7):

$$A = \sum_{t=1}^{p} \sum_{s=1}^{p} \beta_t \beta_s ((1-\lambda)\, r_{t3} + \lambda r_{t2})\, ((1-\lambda)\, r_{s3} + \lambda r_{s2})\, (x_t \cdot x_s)$$

$$B = \sum_{t=1}^{p} \sum_{i=p+1}^{l} \beta_t \alpha_i ((1-\lambda)\, r_{t3} + \lambda r_{t2})\, ((1-\lambda)\, r_{i1} + \lambda r_{i2})\, (x_t \cdot x_i)$$

$$C = \sum_{i=p+1}^{l} \sum_{q=p+1}^{l} \alpha_i \alpha_q ((1-\lambda)\, r_{i1} + \lambda r_{i2})\, ((1-\lambda)\, r_{q1} + \lambda r_{i2})\, (x_i \cdot x_q)$$

$\beta = (\beta_1,...\beta_p)^T$, $\alpha = (\alpha_{p+1},...\alpha_l)^T$, $(\beta, \alpha)^T$ are decision variable

Convert the objective function of quadratic programming described as equation (4-7) to minimum; we can get equation (4-3).

The programming described as equation (4-3) is a convex quadratic programming. We solve its optimal solution $(\beta^*, \alpha^*)^T = (\beta_1^*,...\beta_p^*, \alpha_{p+1}^*,...\alpha_l^*)^T$. Set:

$$w^* = \sum_{t=1}^{p} \beta_t^* ((1-\lambda)\, r_{t3} + \lambda r_{t2})\, x_t + \sum_{i=p+1}^{l} \alpha_i^* ((1-\lambda)\, r_{i1} + \lambda r_{i2})\, x_i$$

$$b^* = ((1-\lambda)\, r_{s3} + \lambda r_{s2}) - (\sum_{t=1}^{p} \beta_t^* ((1-\lambda)\, r_{t3} + \lambda r_{t2})\, (x_t \cdot x_s) + \sum_{i=p+1}^{l} \alpha_i^* ((1-\lambda)\, r_{i1} + \lambda r_{i2})\, (x_i \cdot x_s))$$

$s \in \{s \mid \beta_s^* > 0\}$. or

$$b^* = ((1-\lambda)\, r_{qi} + \lambda r_{q2}) - (\sum_{t=1}^{p} \beta_t^* ((1-\lambda)\, r_{t3} + \lambda r_{t2})\, (x_t \cdot x_q) + \sum_{i=p+1}^{l} \alpha_i^* ((1-\lambda)\, r_{i1} + \lambda r_{i2})\, (x_i \cdot x_q))$$

$q \in \{q \mid \alpha_q^* > 0\}$. or

And set $g(x) = (w^* \cdot x) + b^*$. We can prove the optimal classification function (the method mentioned in [1]) is:

$$f(x) = sgn\,(g(x)) = sgn\,((w^* \cdot x) + b^*),\ x \in R^n \qquad (4-8)$$

The optimal separating hyper-plane is $(w^* \cdot x) + b^* = 0$

The membership function of optimal classification function is

$$\mu(u) = \begin{cases} \varphi_+(u),\ 0 < u \le \varphi_+^{-1}(1) \\ \varphi\_(u),\ \varphi_-^{-1}(1) \le u < 0 \\ 1,\ u > \varphi_+^{-1}(1) \text{ 或 } u < \varphi_-^{-1}(1) \end{cases} \qquad (4-9)$$

In equation (4-9), $\varphi_+(u)$ is the regression function (monotone increasing function about u) getting from $\varepsilon-$ support vector regression. The construction method of $\varepsilon-$support vector regression is:

(a) construct the training set of regression

$$\{(0, 0.5), (g(x_1), \delta_1^+), ..., (g(x_p), \delta_p^+)\} \qquad (4-10)$$

(b) use the training set as equation (4-10), choose appropriate $\varepsilon > 0$, penalty parameter $C > 0$, choose linear kernel as kernel function, construct ε – support vector regression.

Similarly, $\varphi\_(u)$ is the regression function (monotone decreasing function about u) getting from $\varepsilon-$support vector regression. The construction method of $\varepsilon-$support vector regression is:

(b) construct the training set of regression

$$\{(0, 0.5), (g(x_{p+1}), \delta_{p+1}^-), ..., (g(x_l), \delta_l^-)\} \qquad (4-11)$$

(c) use the training set as equation (4-11), choose the same ε and C with above, choose linear kernel as kernel function, construct $\varepsilon-$support vector regression.

$\varphi_+^{-1}(1)$ is the value of function $\varphi_+(u)$'s inverse function at 1. $\varphi_-^{-1}(1)$ is the value of function $\varphi\_(u)$'s inverse function at 1.

Given an input $\bar{x}$ of test point, substitute it into equation (4-8) and (4-9), we can get $f(\bar{x}) = 1$ (or -1) and its degree of membership $\mu(g(\bar{x}))$. Convert them to triangular fuzzy number $\tilde{y}$. It is the output of test points that can reflect the fuzzy classification of test points $(\bar{x}, \tilde{y})$. This classification shows the test point's membership of positive and negative class.

Through above discussion, we can get the algorithm for classifying Support Vector Machines. This algorithm is fuzzy linear separable.

(1) Given the training set of fuzzy linear separable question as equation (3-2), choose an appropriate confidence level $\lambda\,(0 < \lambda \le 1)$ to construct quadratic programming as equation (4-3).

(2) Solve quadratic programming as equation (4-3), get the optimal solution $(\beta^*, \alpha^*)^T = (\beta_1^*,...\beta_p^*, \alpha_{p+1}^*,...\alpha_l^*)^T$

(3) Computing

$$w^* = \sum_{t=1}^{p} \beta_t^*((1-\lambda)\, r_{t3} + \lambda r_{t2})\, x_t + \sum_{i=p+1}^{l} \alpha_i^*((1-\lambda)\, r_{i1} + \lambda r_{i2})\, x_i\,;$$

choose β^*, s positive component β_s^* or α^*'s positive component α_q^*. Accordingly, compute

$$b^* = ((1-\lambda)\, r_{s3} + \lambda r_{s2}) - (\sum_{t=1}^{p} \beta_t^*((1-\lambda)\, r_{t3} + \lambda r_{t2})\,(x_t \cdot x_s) + \sum_{i=p+1}^{l} \alpha_i^*((1-\lambda)\, r_{i1} + \lambda r_{i2})\,(x_i \cdot x_s))$$

or

$$b^* = ((1-\lambda)\, r_{qi} + \lambda r_{q2}) - (\sum_{t=1}^{p} \beta_t^*((1-\lambda)\, r_{t3} + \lambda r_{t2})\,(x_t \cdot x_q) + \sum_{i=p+1}^{l} \alpha_i^*((1-\lambda)\, r_{i1} + \lambda r_{i2})\,(x_i \cdot x_q))$$

(4) Construct optimal separating hyper-plane $(w^* .\ x) + b^* = 0$. Accordingly, get the optimal classification function as equation (4-8).

(5) Use the training set as equation (4-10) and (4-11) separately, construct $\varepsilon-$support vector regression (choose appropriate ε, penalty parameter C, choose linear kernel as kernel function). Get regression function $\varphi^+(u)$ and $\varphi^-(u)$. Accordingly, construct the membership function of optimal classification function as equation (4-9).

Normally, there is only a little β_t^* and α_i^* are not zero in the optimal solution $(\beta_1^*,...\beta_p^*, \alpha_{p+1}^*,...\alpha_l^*)^T$ of quadratic programming as equation (4-3). The corresponding fuzzy training points' input x_t and x_i are called uncertain support vectors. So, the optimal classification function can be expressed as:

$$f(x) = \text{sgn}\,\{(\sum_{(FSV)^+} \beta_t^*((1-\lambda)\, r_{t3} + \lambda r_{t2})\,(x_t \cdot x) + \sum_{(FSV)^-} \alpha_i^*((1-\lambda)\, r_{i1} + \lambda r_{i2})\,(x_i \cdot x)) + b^*\}$$

In this equation, $(FSV)^+$ is the set composed by all the uncertain support vectors in fuzzy positive class points. $(FSV)^-$ is the set composed by all the uncertain support vectors in fuzzy negative class points.

If the outputs of fuzzy training points are all 1 or –1, the fuzzy training set will degenerate to ordinary training set. Now, the linear separable of fuzzy training set becomes the linear separable of ordinary one.

5. Numerical Experimentation

The uncertain Support Vector Classification Machine is built on the basis of classic SVM and fuzzy mathematics.

To explain the rationality of our algorithm, we give specific data to do numerical experimentation.

Suppose the input of 6 training points and their membership degrees of positive (or negative) class are:

$x_1 = (2, 2)^T$, its membership degree of positive class is 1 (its membership degree of negative class is 0);

$x_2 = (1.7, 2)^T$, its membership degree of positive class is 0.95 (its membership degree of negative class is 0.05);

$x_3 = (1.5, 1)^T$, its membership degree of positive class is 0.8 (its membership degree of negative class is 0.2);

$x_4 = (0, 0)^T$, its membership degree of negative class is 1 (its membership degree of positive class is 0);

$x_5 = (0.8, 0.5)^T$, its membership degree of negative class is 0.85 (its membership degree of positive class is 0.15);

$x_6 = (1, 0.5)^T$, its membership degree of negative class is 0.8 (its membership degree of positive class is 0.2).

According to the fuzzy characteristics and their expressions in fuzzy classification mentioned in 2.2, convert the membership degrees to triangular fuzzy numbers; get the following fuzzy training set:

$$S = \{(x_1, \tilde{y}_1), (x_2, \tilde{y}_2), (x_3, \tilde{y}_3), (x_4, \tilde{y}_4), (x_5, \tilde{y}_5), (x_6, \tilde{y}_6)\}$$

The outputs of training points are triangular fuzzy numbers.

$$\tilde{y}_1 = 1 = (1, 1, 1), \tilde{y}_2 = (0.755, 0.9, 1),$$

$$\tilde{y}_3 = (0.1, 0.6, 1.1), \tilde{y}_4 = -1 = (-1, -1, -1)$$

$$\tilde{y}_5 = (-1.05, -0.7, -0.34),$$

$$\tilde{y}_6 = (-1.1, -0.6, -0.1),$$

Choose confidence level $\lambda = 0.8$, we can verify the fuzzy linear separable of training set S from Definition 3.14. So:

The optimal separating hyper-plane is $2\,[x]_1 + [2x]_2 - 4 = 0$
Shown as Figure 2:

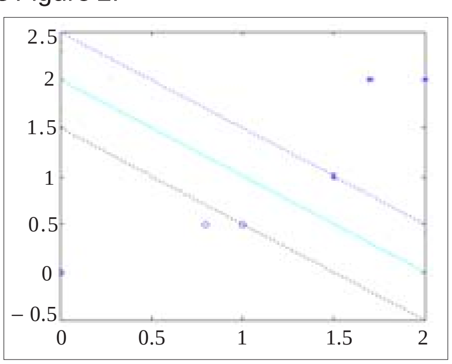

Figure 3. Numerical experimental (1)

The optimal classification function is:

$$f(x) = sgn\,(g(x)) = sgn\,(2\,[x]_1 + [2x]_2 - 4)$$

We construct the membership function of optimal classification function:

Use $S_1 = \{(0, 0.5), (4, 1), (3.4, 0.95), (1, 0.8)\}$ as fuzzy training set, choose $\varepsilon = 0.1$, $C = 10$ and the linear kernel, construct support vector regression, we can get regression function $\varphi\_(u) = 0.11u + 0.6$.

The membership function of optimal classification function is

$$\mu(g(x)) = \begin{cases} 0.10\,g(x) + 0.6,\ 0 < g(x) \le 4 \\ -0.11\,g(x) + 0.6,\ -3.64 \le g(x) \le 0 \\ 1,\ g(x) > 4 \text{ 或 } g(x) < -3.64 \end{cases}$$

Suppose there are test points input $x_7 = (1, 2)^T$, $x_8 = (1, 0)^T$ Put them into $f(x)$ and $\mu\,(g\,(x))$, we get $f(x_7) = 1$ (fuzzy positive class), its membership degree of positive class is 0.8 and negative is 0.2, so $\tilde{y}_7 = (0.1, 0.6, 1.1)$ (triangular fuzzy number); $f(x_8) = -1$ (fuzzy negative class), its membership degree of negative class is 0.82 and positive is 0.18, so $\tilde{y}_8 = (-1.08, -0.64, -0.2)$ (triangular fuzzy number).

Compare Fuzzy Support Vector Machine with classic support vector machine. First, suppose fuzzy training set is:

$S_1 = \{(0, 0), -1)\,((0.8, 0.5), -1), ((1, 0.5), -1), ((2, 2), 1), ((1.7, 2), 1), ((1.5, 1), 1)\}$

Then the output of training points $((1.5, 1), 1)$ will change: $1 = (1, 1, 1) \rightarrow (0.1, 0.6, 1.1) \rightarrow (-1.1, -0.6, -0.1) \rightarrow (-1, -1, -1) = -1$

So the other fuzzy training sets are:

$S_2 = \{((0, 0), -1)\,((0.8, 0.5), -1), ((1, 0.5), -1), ((2, 2), 1), ((1.7, 2), 1), ((1.5, 1), 1), (-1.4, 0.6, 2.6))\}$

$S_3 = \{((0, 0), -1)\,((0.8, 0.5), -1), ((1, 0.5), -1), ((2, 2), 1), ((1.7, 2), 1), ((1.5, 1), 1), (-1.4, 0.6, 2.6))\}$

$S_4 = \{((0, 0), -1)\,((0.8, 0.5), -1), ((1, 0.5), -1), ((2, 2), 1), ((1.7, 2), 1), ((1.5, 1), -1)\}$

Choose $\lambda = 0.8$, we can verify that fuzzy training sets S_1, S_2, S_3, S_4 are fuzzy linear separable according Definition 3.14. So, on the basis of the algorithm we propose in this paper, we arrive at the optimal separating hyper-plane are 4 straight lines:

$$l_1: [x_1] + [x_2] = 2;\ l_2: [x_1] + [x_2] = 2.4;$$

$$l_3: 0.385\,[x_1] + 1.923\,[x_2] = 1.4;$$

$$l_4: 0.385\,[x_1] + 1.923\,[x_2] = 1.76;$$

Shown as Figure 3, the output of fuzzy training points change:

$(1, 1, 1) \rightarrow (0.1, 0.6, 1.1) \rightarrow (-1.1, -0.6, -0.1) \rightarrow (-1, -1, -1)$ and the straight line move as: $l_1 \rightarrow l_2 \rightarrow l_3 \rightarrow l_4$. This change shows that the fuzzy training points' membership degrees

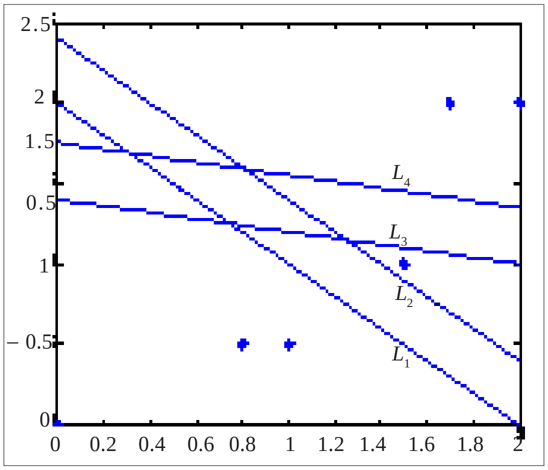

Figure 2. Numerical experimental (2)

of negative class are increased and positive one decreased. From this, it can be seen that the result coincides with intuitive judgments.

6. Acknowledgment

The authors wish to thank Computer and Information Engineering School of Anyang Normal University.

References

[1] Vapnik, V. N. (1998). *Statistical Learning Theory,* New York: Wiley.

[2] Vapnik, V. N. (1995) *The Nature of Statistical Learning Theory,* New York: Springer-Verlag.

[3] Nello, C. et al. (2000). *Introduction to Support Vector Machines*. Cambridge University Press.

[4] Sakawa, M., Nishizaki, I., Uemura, Y. (2000). Interactive fuzzy programming for multi-level linear programming problems with fuzzy parameters, *Fuzzy Sets and Systems,* 109, p. 3–19.

[5] Liu, B. D., Zhao, R. Q. (1998). *Stochastic Programming and Fuzzy Programming.* Tsinghua University Press.

[6] Deng, N. Y.,Tian, Y. J. (2004). *New Method in Data Mining: Support Vector Machines.* Science Press.

[7] Mitra, S., Pal, S. K., Mitra, P. (2002). Data mining in soft computing Communicating framework survey. *IEEE Trans. Neural Networks,* 13 (1) 3-14.

[8] Kou, G., Xiantao, L., Peng, Y. (2003). Multiple criteria linear programming approach J. C. to data mining. *Models, algorithm designs and software development, Optimization Methods and Software* 18 (4) 453-473.

Author Biography

Wang Aimin, born in 1957. Professor and master supervisor of computer science at Anyang Normal University, China. His major research interests include intelligent computing and fuzzy control.

Li Jie, born in 1978. Lecture of computer science at Anyang Normal University, China. His major research interests include real-time system and intelligent control.