On the Support Vector Machine with the kernel of the q-normal distribution

Hirofumi JOGUCHI and Masaru TANAKA
Department of Information & Computer Sciences,
Saitama University
255 Shimo-okubo, Saitama-shi, Saitama, 338-8570 Japan
Tel. +81-48-858-3957, Fax.: +81-48-858-3716
E-mail: {joguchi,mtanaka}@mi.ics.saitama-u.ac.jp

Abstract: Support Vector Machine (SVM) is one of the methods of pattern recognition that separate input data using hyperplane. This method has high capability of pattern recognition by using the technique, which says kernel trick, and the Radial basis function (RBF) kernel is usually used as a kernel function in kernel trick. In this paper we propose using the q-normal distribution to the kernel function, instead of conventional RBF, and compare two types of the kernel function.

1. Introduction

Usually, it is easy to recognize what it is, when we see the certain pattern. Although we have the exceptional pattern discernment and recognition capability, it is very difficult to make the machine, which has such capability. Various methods have been proposed, Support Vector Machine (SVM) is known as one of the methods of the most excellent pattern recognition. Besides this method is known as having the high capability of nonlinear discernment by combining Kernel Trick. Kernel trick is to separate the data in the feature space that mapped the data to the high-dimensional space by hyperplane using kernel function. Radial Basis Function (RBF) is known as major kernel function. In this paper we propose using the q-normal distribution to the kernel function, instead of RBF kernel. Moreover since both kernel functions are parametric model, we investigate the influence of parameters.

2. SVM

2.1 The linear separable case

The linear separable case, it is very difficult to choose the optimal hyperplane that classify the l training data $\mathbf{x}_i$, $(\mathbf{x}_i \in R^n, i = 1, \cdots, l)$ which has linear separable class label $\mathbf{y}_i$, $(y_i \in \{\pm 1\})$ into two classes, because there are countless hyperplanes in input space.

SVM defines hyperplane with maximal distance (margin) from hyperplane to the closest data points as optimal hyperplane. The optimal hyperplane like follow

$$\mathbf{w} \cdot \mathbf{x} + b = 0, \tag{1}$$

where $\mathbf{w}$ is the normal vector to the hyperplane, and $|b|/||\mathbf{w}||$ is the perpendicular distance from the hyperplane to the origin. In linear separable case, all the data points can satisfy following inequality constraints

$$y_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1 > 0. \tag{2}$$

This constraints represent the optimal hyperplane is between the two parallel hyperplanes $H_1 : \mathbf{x}_i \cdot \mathbf{w} + b = 1$, $H_2 : \mathbf{x}_i \cdot \mathbf{w} + b = -1$ and that there are no data points between them. The closest data points called support vector, lie on the hyperplane $H_{1,2}$. Only the support vector has determined the optimal hyperplane, other data points do not contribute to the composition of the optimal hyperplane. The perpendicular distance from the hyperplane $H_{1,2}$to the origin are $H_1 : |1 - b|/||\mathbf{w}||$ and $H_2 : |-1 - b|/||\mathbf{w}||$. Hence margin is defined as $2/||\mathbf{w}||$. Therefore we can get the hyperplane with maximum margin by minimizing $||\mathbf{w}||^2$, under these constraints. Thus, we introduce Lagrange multipliers $\alpha \geq 0$ for the inequality constraints, the Lagrangean is

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{l} \alpha_i y_i (\mathbf{x}_i \mathbf{w} + b) + \sum_{i=1}^{l} \alpha_i. \tag{3}$$

Requiring $\nabla_{\mathbf{w}} L(\mathbf{w}, b, \alpha) = 0, \nabla_b L(\mathbf{w}, b, \alpha) = 0$, and substitute them into Eq.(3), it is represented as follows maximize:

$$L_d(\alpha) = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \mathbf{x}_j, \tag{4}$$

Subject to:

$$0 \leq \alpha_i, \ \sum_i \alpha_i y_i = 0. \tag{5}$$

To solve this dual problem we get the optimal hyperplane.

2.2 The linear non-separable case

In linear non-separable case, it uses combining "soft margin" and "kernel trick". Soft margin is to add the penalty to the error C to the inequality constraint (5). It is the parameter to be chosen by the user to allow some errors. Hence maximize Eq.(4) is calculated under new inequality constraint $\sum_i \alpha_i y_i = 0, 0 \leq \alpha \leq C$.
Kernel trick is the method that avoid computing higher dimensional map $\Phi(\mathbf{x})$, and compute the dot product in the higher dimensional space with few computational task. It shows as following.

$$K(\mathbf{x}_1, \mathbf{x}_2) = \Phi(\mathbf{x}_1) \cdot \Phi(\mathbf{x}_2). \tag{6}$$

Typical kernel function is RBF. It is the following

$$K(\mathbf{x}, \mathbf{y}) = \exp\left\{\frac{-||\mathbf{x} - \mathbf{y}||^2}{2\sigma^2}\right\}. \tag{7}$$

Therefore, SVM in linear non-separable case as follows,
maximize:

$$L_d(\alpha) = \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i \mathbf{x}_j), \quad (8)$$

Subject to:

$$0 \leq \alpha_i \leq C, \ \sum_i \alpha_i y_i = 0. \quad (9)$$

3. The q-normal distribution

The q-normal distribution is the set of probability density functions rely on parameter q. The case of $q = 1$ represents normal distribution, the case of $q = 2$ represents Cauchy distribution, the case of $q = 1 + \frac{2}{n+1}$ represents t-distribution. Like this it represents some familiar probability density functions on the value of q. Moreover it has the feature of smooth about q, it links several probability density functions smooth. It is the following,

$$P_q(x, y) = \left\{ 1 - \frac{1-q}{3-q} \frac{||\mathbf{x} - \mathbf{y}||^2}{\sigma^2} \right\}^{\frac{1}{1-q}}. \quad (10)$$

where $q \leq 1 + \frac{2}{D}$ (D: The dimension of input data). Note that Eq. (10) needs to be real, if $\{\cdots\} < 0$, then $P_q(x, y) = 0$. In addition, the normalization constant is omitted.

4. Experiment method

In this section we conducted two experiments, in order to investigate the performance of the q-normal distribution. First we investigate about clustering, next investigate predict. In addition, the penalty to the error C is fixed to 100.

4.1 Support Vector Clustering (SVC)

We investigate the performance of the q-normal distribution for clustering. We use the spiral training data like Fig.1 (number of data is 34) and the rectangle training data like Fig.5 (number of data is 87), to the SVM, and compare the result of two types of kernel functions. The kernel functions are RBF kernel and the q-normal distribution kernel. The variances of each kernel function are $0.01, 0.2, 0.3$ and 0.7. Furthermore examine the result of SVM when changed the parameter of q in the q-normal distribution. The upper bound of q is set to 2.0, because the dimension of training data is two.

4.2 Predict iris data

We use the iris data set [2]. It can be obtained from the UCI repository [3]. This data set contains 150 instances, divided into every 50 three classes. Each instance contains four measurements of an iris flower. We select training data from each class, carry out SVM learning using these training data. After learning, we predict the remaining data. The number of training data is 10 and 15. This experiment is conducted on RBF kernel and the q-normal distribution kernel, compare each result. The variances of each kernel function are 0.25, 0.2 and 0.15. The upper bound of q is set to 1.5, because the dimension of training data is four.

5. The result

5.1 SVC

We show the results of RBF, the q-normal distribution ($q = 1.5$) and, the q-normal distribution ($q = 1.5$) in Fig.2 $\sim$ Fig8, the number of support vector in Table 1 and Table 2. Support vector is indicated as white circle.

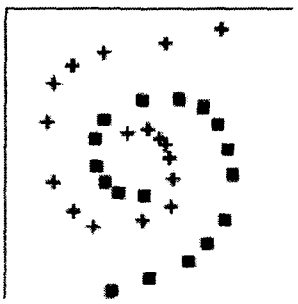

Figure 1. Input data 1

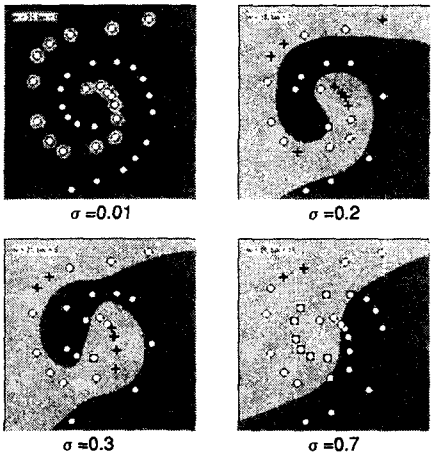

Figure 2. The result of clustering by RBF kernel about input data 1

Table 1. The number of Support Vector about each variances for input data 1

	σ =0.01	0.2	0.3	0.7
RBF	34	18	21	29
The q-normal distribution (q = 1.5)	34	29	21	29
The q-normal distribution (q = 2.0)	34	24	22	24

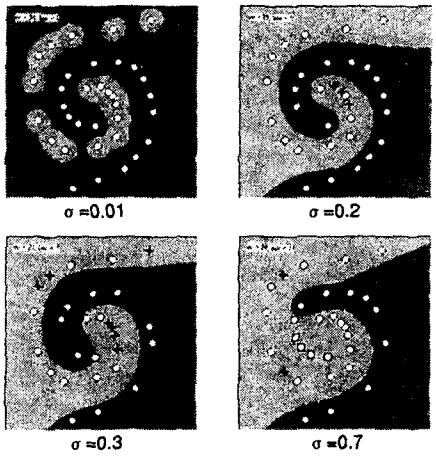

Figure 3. The result of clustering by the q-normal distribution (q = 1.5) about input data 1

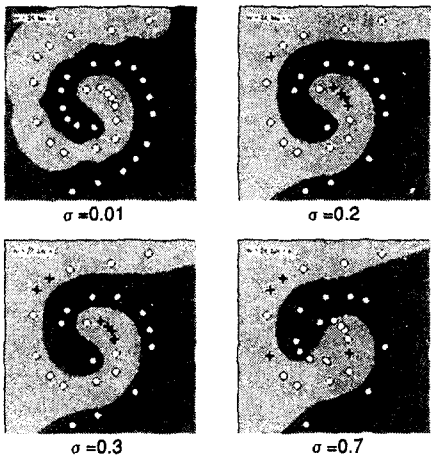

Figure 4. The result of clustering by the q-normal distribution (q = 2.0) about input data 1

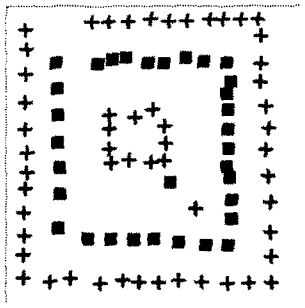

Figure 5. Input data2

Table 2. The number of Support Vector about each variances for input data 1

	σ =0.01	0.2	0.3	0.7
RBF	87	32	26	64
The q-normal distribution (q = 1.5)	87	35	31	64
The q-normal distribution (q = 2.0)	87	47	33	41

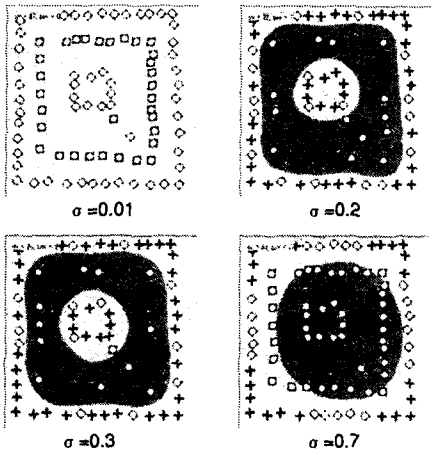

Figure 6. The result of clustering by RBF kernel about input data 2

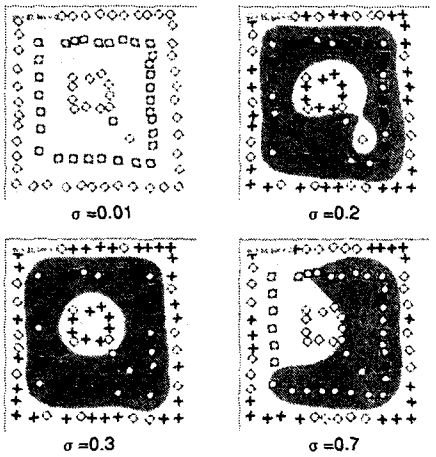

Figure 7. The result of clustering by the q-normal distribution (q = 1.5) about input data 2

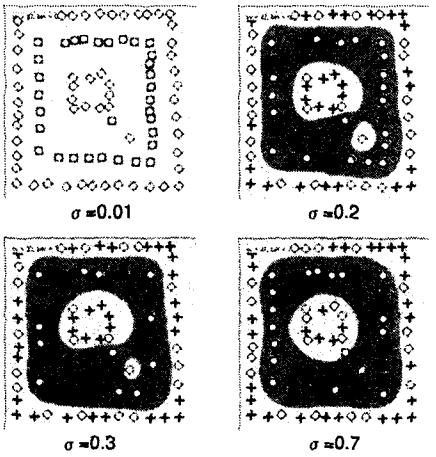

Figure 8. The result of clustering by the q-normal distribution (q = 2.0) about input data 2

First we consider about the σ to investigate the influence of the parameter common to both of kernels. All results show good clustering power that a σ becomes small, but the generalization power is down. See the table 1, the number of support vector is almost the same as the number of input data for the both kernel functions except for $\sigma = 0.2, 0.3$. If many support vectors are used, generalization power has down and calculating takes long time, accordingly we consider $\sigma = 0.3$ is best for the q-normal distribution kernel, $\sigma = 0.2$ is best for the RBF kernel. Next we compare the two type of kernels in the case of best σ, the q-normal distribution kernel clustering better than RBF kernel. As the parameter of q is larger, the result is clustering more clearly.

5.2 Iris data

The result of Accuracy of predict shows in Table 3, Table 4. Best accuracy shows in the bold letter.

Table 3. Accuracy about each variances of 10 training data

	$\sigma = 0.3$	$\sigma = 0.25$	$\sigma = 0.2$	$\sigma = 0.15$
RBF	95.833%	92.833%	**98.333%**	96.667%
q=1.5	95.833%	96.667%	**98.333%**	**98.333%**
q=1.2	96.667%	95.833%	**98.333%**	97.5%
q=0.8	95.833%	96.667%	**98.333%**	96.667%

Table 4. Accuracy about each variances of 15 training data

	$\sigma = 0.3$	$\sigma = 0.25$	$\sigma = 0.2$	$\sigma = 0.15$
RBF	**96.191%**	**96.191%**	**96.191%**	95.238%
q=1.5	**97.143%**	**97.143%**	**97.143%**	96.191%
q=1.2	**96.191%**	**96.191%**	**96.191%**	95.238%
q=0.8	**96.191%**	**96.191%**	**96.191%**	95.238%

See Table 3, the q-normal distribution kernel discriminates and RBF kernel shows same best accuracy in $\sigma = 0.2$. But the q-normal distribution kernel ($q = 1.5$) shows best accuracy in $\sigma = 0.15$ too, it is the area of high accuracy wider than RBF kernel. That is, the q-normal distribution kernel is considered to be flexible about variances. See Table 4, both kernel shows the same accuracy in $q = 0.8, 1, 2$, in $q = 1.5$ show the result better than the RBF kernel. Both results shows that as the parameter of q is larger, the area of best accuracy wider.

6. Conclusion

In this paper, we propose SVM with the kernel of the q-normal distribution, and we experiment about clustering and predict. The case of clustering, the q-normal distribution kernel showed the good result rather than the conventional RBF kernel. The case of prediction, although it was the same accuracy, shown that flexibility is high. Moreover, about Parameter q, larger one shows the result better. This suggests that the q-normal distribution kernel is good for SVM. Note that the q-normal distribution can be considered as one of possible extension of non-homogeneous polynomial kernels. In future, we'll compare another kernel function and more study about the behavior of changing the parameter of q.

References

[1] C.J.C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," Data Mining and Knowledge Discovery, 1998.

[2] A. Ben-Hur, D. Horn, H.T. Siegelmann, and V. Vapnik, "A Support Vector Method for Clustring",NIPS 2000, pp.367-373, 2001

[3] C.L. Blake and C.J. Merz, UCI repository of machine learning databases, 1998.

[4] Masaru Tanaka, "A Consideration on A Family of q-normal distribution", the paper of IEICE D-II, 2002

[5] Chih-Chung Chang and Chih-Jen Lin, LIBSVM: a library for support vector machines, 2001. Software available at
http://www.csie.ntu.edu.tw/ cjlin/libsvm