Handwritten Digits Recognition Based on SVM

Apply PCA to Accelerate the Clustering Process

1st Yusheng Chen
School of Computer Science
Nanjing University of Posts and Telecommunications
Nanjing, China
1023040817@njupt.edu.cn

***Abstract*—Support vector machine (SVM), proposed by Vapnik in 1995 for classification and regression problems, based on the statistical learning theory of VC dimension and structural risk minimization theory, which has come to be a powerful method of overcoming the curse of dimensionality and over learning. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier. Due to its excellent clustering performance, SVM has been applied to many fields, such as face recognition, Web page classification and the like. In this paper, we will introduce its performance in handwritten digit recognition in UCI dataset, MINIST.PKL, which is consisted of 60000 training samples and 10000 testing samples. At the beginning of the paper, the maximum margin algorithm is introduced to simplify the problem. Through the experiments, we know that PCA is optional, which will sharply shorten the clustering time. However, it will reduce the accuracy in some cases, as we all know that.**

***Index Terms*—SVM, handwritten, recognition**

I. Introduction

Handwritten digit recognition is a branch of optical character recognition, mainly study of how to allow the computer to automatically identify the figures on the Arabic numerals. There are some barriers to do this job. Firstly, digits have fewer strokes than characters, bringing more difficulties in recognition. Secondly, It is difficult to make a universally recognizable digital recognition system that can take into account all kinds of wordings in the world, and the recognition rate is very high, so it is very difficult to make a digital signature system. The problem is to solve, and there are many theories can be used to solve the problem, such as KNN, CNN. We apply the SVM to the problem for the reason that the SVM model is much smaller than KNN and ANN, however, it achieves amazing accuracy.

II. Simplify the Problem with Maximum Margin Algorithm [1]

A. Maintaining the Integrity of the Specifications

Support vector machine is a binary classification model. The algorithm means to find a decision function (Hyperplane). Sample point x of dimension n belongs to either of two class A and B. The input to the training algorithm is a set of p examples X_i with labels 1 or -1. From these training examples, the algorithm finds a decision function $D(x)$ during a learning phase. When the training finished, the classification of unknown patterns is predicted according to the following rule:

$$x \in \begin{cases} A, & \text{if } D(x) > 0 \\ B, & \text{if } D(x) \leq 0 \end{cases} \tag{1}$$

The decision functions must be linear in their parameters but are not restricted to linear dependence of. These functions can be expressed either in direct or in dual space. In other words, It is meant to the following optimization problem (2) or its dual problem need to be solved:

$$\begin{aligned} \min_{\omega,b} \quad & \frac{1}{2}\|\omega\|^2 \\ \text{s.t.} \quad & y_i(\omega^T X_i + b) \geq 1, i = 1, 2, \ldots, m \end{aligned} \tag{2}$$

In (2) if the distance of x_i to two different hypersurfaces

$$\gamma = \frac{2}{\|\omega\|^2} \tag{3}$$

x_i will be called support vector(SV). (2) is a CQP(Convex Quadratic programming) problem, with so many methods to solve it, such as SMO(Sequential Minimal Optimization, proposed by Platt,1998), Newton. In the direct space notation is identical to the Perceptron decision function:

$$D(x) = \sum_{i=1}^{n} \omega_i \varphi_i(x) + b \tag{4}$$

In this equation φ_i are predefined functions of and non-linear transformation, transforming x to a high dimensional feature-space. It is noteworthy that φ_i is predefined to obtain better clustering results.

In dual space, kernel function is introduced to deal with the linearly inseparable case. The decision function is of the form:

$$D(x) = \sum_{j=1}^{n} \alpha_j K(X_k, x) + b \tag{5}$$

The coefficients α_j are the parameters to be adjusted and X_k is the training patterns. The function K is a predefined

Identify applicable funding agency here. If none, delete this.

kernel, for example, a potential function or any Radial Basis Function.

$$K(x, x') = \varphi(x)\varphi(x') \tag{6}$$

The proposed training algorithm is based on the 'generalized portrait' method described in [Vap82] that constructs separating hyperplanes with maximum margin. Here this algorithm is extended to train classifiers linear in their parameters, Firstly, the margin between the class boundary and the training patterns is formulated in the direct space. This problem description is then transformed into the dual space by the principle of duality. The resulting problem is that of maximizing a quadratic form with constraints and is amenable to efficient numeric optimization algorithms.

III. Apply SMO to Solve (4) (Processes of Clustering)

In 1998, Platt proposed the Sequential Minimal Optimization algorithm (SMO) based on the decomposition algorithm. During iteration, only the corresponding two sample points are selected to adjust, meaning need to solve an optimization problem with two variables. When it comes to train model, it is necessary to transform (4) into the form as follows:

$$\begin{aligned} \min_{\alpha} \quad & \sum_{i=1}^{l}\sum_{j=1}^{l} y_i y_j \alpha_i \alpha_j K(x_i, x_j) - \sum_{j=1}^{l} \alpha_j \\ \text{s.t.} \quad & \sum_{i=1}^{l} y_i \alpha_i = 0 \\ & 0 \le \alpha_i \le C \quad i = 1, 2, \ldots, l \end{aligned} \tag{7}$$

The Laplace equation of (4) is just (6). C is Penalty factor, while α_i and α_j are Lagrange coefficient. Next, I want to introduce a process of that apply SMO to train the (6). In order to accelerate the convergence of the algorithm, it is necessary to select a good working set at each iteration, i.e., to select two suitable optimization variables or appropriate training points. The first training point will be selected from those points that violate the KKT condition [2]. Specifically, traverse all support vectors. If point (x_j, y_j) violates the condition (7)

$$y_j(\sum_{i=1}^{l} \alpha_i y_i K(x_i, x_j) + b) \begin{cases} \ge 1 & \{x_j | \alpha_j = 0\} \\ = 1 & \{x_j | 0 < \alpha_j < C\} \\ \le 1 & \{x_j | \alpha_j = C\} \end{cases} \tag{8}$$

most seriously, it will be chosen as the first training point. However, if in one iteration, a traversal found that there is no support vector within the corresponding training points need to be adjusted, it traverses the entire training set then the first point that violates the condition (7) will be chosen as first training point. In the algorithm, the above process acts as an inner loop. As for the second training point, it just bases on the first point. We make the following agreement:

$$g(x) = \sum_{i=1}^{l} \alpha_i y_i K(x_i, x) + b \tag{9}$$

$$E_i = g(x_i) - y_i = (\sum_{j=1}^{l} \alpha_j y_j K(x_j, x_i) + b) - y_i, i = 1, 2 \tag{10}$$

If the first point is (x_j, y_j), The second point, supposed as (x_i, y_i), should meet the following condition:

$$\max |E_i - E_j| \tag{11}$$

(x_i, y_i) should also be a support vector and not the same with (x_j, y_j). Experiments show that if we use a temporary array to store E, the practical complexity of the algorithm will be reduced. If it fails to find the second points in the above manners, traverse the entire training set will be necessary.

When two training points have been chosen, the usual practice is that supposing their Lagrange multipliers are α_i and α_j. Then (6) will be simplified as follows:

$$\begin{aligned} \min \quad & W(\alpha_1, \alpha_2) = \frac{1}{2} K(x_1, x_2)\alpha_2^2 + y_1 y_2 K(x_1, x_2)\alpha_1 \alpha_2 \\ & - (\alpha_1 + \alpha_2) + y_1 \alpha_1 \sum_{i=3}^{l} y_i \alpha_i K(x_i, x_1) \\ & + y_2 \alpha_2 \sum_{i=3}^{l} y_i \alpha_i K(x_i, x_2) \\ \text{s.t.} \quad & \alpha_1 y_1 + \alpha_2 y_2 = \sum_{i=3}^{l} y_i \alpha_i = C \end{aligned} \tag{12}$$

Each time the minimum optimization is done (11), E of each training point must be updated so that the next training can use it to select the second training point.

SMO is summarized as follows:

IV. Processes of Handwritten Digits Recognition

A. Processes of Digits Recognition

Handwritten digits recognition integrates the knowledge of image processing, pattern recognition, data analysis and machine learning. It is an interdisciplinary problem. The recognition system is usually composed of image preprocessing feature extraction and classification recognition, as Fig. 1 [3]:

B. Image Preprocess

In order to improve the accuracy of handwritten digits recognition, according to the quality of database samples, it is suitable to take appropriate preprocessing operations. Preprocessing is an important part of handwritten numeral recognition. However, because the focus of this paper is not on preprocessing, and a lot of researchers have done a lot

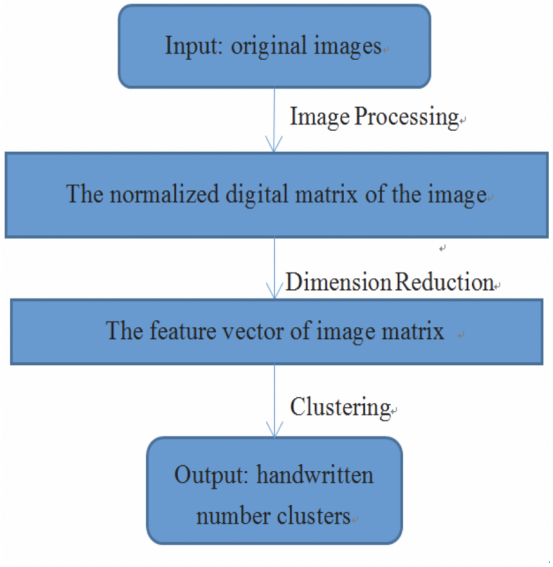

Fig. 1. Processes of Digits Recognition

of research in this field, this paper simply introduces the pre-processing image enhancement, refinement, and normalization. Then the normalized digital matrix, the values of its elements between one and zero, of the image will be obtained. Next, Specific practices can be got references.

C. Dimension Reduction

PCA (Principal Component Analysis), as one of the most commonly used data dimensionality reduction algorithms, can also be regarded as a kind of multivariate statistical analysis method, which is the most commonly used feature extraction method, and has been the focus of attention and research. [4] It makes the difficulty and complexity greatly of the problem simplified, can improve the data signal to noise ratio, improving the original data anti-interference ability.

1) Principle Component Contribution Rate: The greater the ratio of the eigenvalues of the ith principal component to the sum of all the eigenvalues of the covariance matrix, the stronger the ability of the first principal component to synthesize the original indicator information. The characteristic value corresponding to the ith principal component is calculated as:

$$\alpha_i = \frac{\lambda_j}{\sum_{i=1}^{p} \lambda_i} \tag{13}$$

2) Cumulative Contribution Rate: The higher the ratio of the sum of the eigenvalues of the first K principal components to the sum of all eigenvalues, the more the former K principal components can represent the information of the original data. The formula is:

$$M_k = \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{p} \lambda_i} \tag{14}$$

In order to solve the practical problem, we usually select the former principal component to make the cumulative variance contribution rate meet certain requirements (usually more than 80%), and use the first k principal components instead of the original variables to analyze. The purpose of dimension reduction can also be regarded as a feature selection.

3) Steps for Principal Component Analysis Algorithm:

1) Calculate the mean vector μ of the samples in the sample data set X, that is:
$$\mu = \frac{1}{n} \sum_{i=1}^{n} x_i \tag{15}$$
2) Centering the sample data is done by subtracting the sample mean for each sample, as following:
$$\tilde{X} = X - \mu \tag{16}$$
3) Construct the covariance matrix V of the data matrix,
$$V = \frac{1}{n} \tilde{X} \tilde{X}^T \tag{17}$$
4) The eigenvalues and the corresponding eigenvectors are obtained by decomposing the matrix V, and the eigenvalues are descending;
5) According to the size of the contribution rate, the former d eigenvalues will be selected:
$$F = W_d^T \tilde{X} \tag{18}$$
Here, F is the principal component function;
6) Reconstruct the original data with the selected principal components:
$$X = WF + \mu \tag{19}$$

The application of PCA is mainly to reduce the model and the time to cluster.

D. Clustering

Basic support vector machine is mainly applied to two solve the two-classification problem, and handwritten digits recognition belongs multiclass (10 categories) classification problem. So it is necessary to transform SVM to a complex model. In this paper, determine the multi-class objective function algorithm will be introduced.

We can change the original support vector classifier in the previously described problem so that it can simultaneously calculate the multi-class classification decision function, as follows.

$$\begin{aligned} \min_{\omega_r \in H, b_r \in R, \xi_r \in R^l} \quad & \frac{1}{2} \sum_{r=1}^{M} \|\omega_r\|^2 + \frac{C}{m} \sum_{i=1}^{m} \sum_{r \neq y_1} \xi_i^T \\ \text{s.t.} \quad & (\omega_{y_i} \cdot x_l) + b_{y_i} \geq (\omega_r \cdot x_l) + b_r + 2 - \xi_l^T \\ & \xi_l^T \geq 0 \end{aligned} \tag{20}$$

$$m \in \{1, 2, \ldots, M\} \backslash \{y_i\}, y \in \{1, 2, \ldots, M\} \tag{21}$$

Then SMO is also a good choice to solve this problem. The advantage of the algorithm is that the accuracy of the result

can be compared with the widely used one class method, and the method is more suitable for the category of very many problems. But the calculation of it is very large, that is to say, it possesses a bad time complexity.

V. Experiment and Results Analysis

A. Introduction to Dataset

The MNIST database (Mixed National Institute of Standards and Technology database) is a large database of handwritten digits that is commonly used for training various image processing systems. [5] The database is also widely used for training and testing in the field of machine learning. It was created by "remixing" the samples from NIST's original datasets. The creators felt that since NIST's training dataset was taken from American Census Bureau employees, while the testing dataset was taken from American high school students, it was not well-suited for machine learning experiments. Furthermore, the black and white images from NIST were normalized to fit into a 20x20 pixel bounding box and anti-aliased, which introduced grayscale levels. The MNIST database contains 60,000 training images and 10,000 testing images. Half of the training set and half of the test set were taken from NIST's training dataset, while the other half of the training set and the other half of the test set were taken from NIST's testing dataset. There have been a number of scientific papers on attempts to achieve the lowest error rate; one paper, using a hierarchical system of convolutional neural networks, manages to get an error rate on the MNIST database of 0.23 percent. The original creators of the database keep a list of some of the methods tested on it. In their original paper, they use a support vector machine to get an error rate of 0.8 percent.

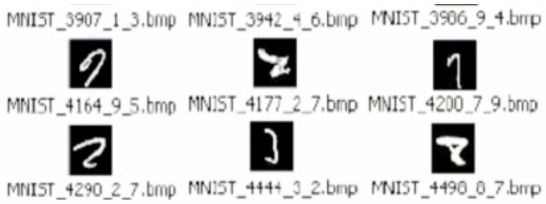

Fig. 2. the Images of Minist.pkl

B. Introduction to the Experimental Environment

We conducted this experiment on our own computer, the configuration of which is introduced as the following table.

TABLE I
Environment of Experiment

Hardware Environment		**Software Environment**	
CPU	**Main Memory**	**OS**	**Programming Language**
I7-4700	12GB	Windows 10	Python

C. Results of Experiment

In the experiment, SVC (Support Vector Cluster) function of klearn package was called, with $C = 1$(penalty factor) and $tol = 0.0001$. Some kinds of kernel functions are applied to verify the performance of SVM, and the results will be shown in TABLE II.

TABLE II
Performance of Different Kernel Functions

Kernel function	linear	rbf	sigmoid
Correct rate	0.9989	0.9332	0.9183
Run time	68s	158s	187s

TABLE III
Performance of Different Kernel Functions (Applied PCA ,50 Features are Reserved)

Kernel function	linear	rbf	sigmoid
Correct rate	0.9609	0.9884	0.8876
Run time	5s	8s	8s

Different scholars at home and abroad based on this database (including the NIST database and MNIST database) research results as the following table;

TABLE IV
Performance of Other Algorithm

Algorithm	**Error rate (%)**
linear classifier(1-layer NN)	12.00
K-nearest-neighbors, Euclidean	5.00
3-layer NN	2.50
k-NN, shape context matching	0.67

Compared with the above methods, the 0.11% error rate obtained by SVM is so amazing that we can conclude that SVM is an effective method to be applied in handwritten digits recognition. From TABLE II, it could be speculated that selections of kernel functions will affect the performance of the algorithm SVM, to a certain extent. Kernel function selection is a critical step to apply SVC, with kernel function rbf not performing as well as expected. PCA will sharply shorten the time of clustering, with little accuracy loss. Note that the number of the feature to be reserved needs great experimentation to obtain. In other words, theoretical guidance is necessary here, being the target of our efforts.

VI. Conclusion

Support Vector Machine (SVM) is a statistical learning theory proposed by Vapnik for classification and regression problems. It is a learning system that uses linear function hypothesis space in high-dimensional feature space. It is composed of a learning algorithm from optimization theory Training, the algorithm implements a learning bias derived from the statistical learning theory and is a standard and powerful method. Over the years since its inception, SVM has outperformed most other learning systems in a wide range of

applications. Because the SVM method has many advantages and promising experimental performance, it has been paid more and more attention. It has become a hotspot in the field of machine learning research and has achieved very good results, such as face recognition, handwritten digits recognition and page classification. But, nowadays, selection of kernel function is mainly based on experience of scholars. And the PCA algorithm is introduced in this paper, mainly aiming to accelerate the process of clustering, however, it will reduce the accuracy of SVM. So there are extra experiments to be needed to balance accuracy and efficiency.

REFERENCES

[1] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the fifth annual workshop on Computational learning theory*. ACM, 1992, pp. 144–152.

[2] A. Ben-Hur, D. Horn, H. T. Siegelmann, and V. Vapnik, "Support vector clustering," *Journal of machine learning research*, vol. 2, no. Dec, pp. 125–137, 2001.

[3] Q. Li, L. Chen, and W. Wang, "Research on handwritten digit rapid identification method based on svm," *Compute technology and development*, vol. 24, no. 2, pp. 205–208.

[4] I. Jolliffe, "Principal component analysis," in *International encyclopedia of statistical science*. Springer, 2011, pp. 1094–1096.

[5] Y. LeCun, C. Cortes, and C. Burges, "Mnist handwritten digit database," *AT&T Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, vol. 2, 2010.