

Bilinear Grid Search Strategy Based Support Vector Machines Learning Method

Li Lin, Zhang Xiaolong, Zhang Kai and Liu Jun

School of Computer Science and Technology, Wuhan University of Science and Technology, China

Hubei Province Key Laboratory of Intelligent Information Processing and Real-time Industrial System, China

E-mail: lilin@wust.edu.cn

Keywords: support vector machines, model selection, parameters optimization, protein structure prediction

Received: October 21, 2013

Support Vector Machines (SVM) learning can be used to construct classification models of high accuracy. However, the performance of SVM learning should be improved. This paper proposes a bilinear grid search method to achieve higher computation efficiency in choosing kernel parameters (C , γ) of SVM with RBF kernel. Experiments show that the proposed method retains the advantages of a small number of training SVMs of bilinear search and the high prediction accuracy of grid search. It has been proved that bilinear grid search method (BGSM) is an effective way to train SVM with RBF kernel. With the application of BGSM, the protein secondary structure prediction can obtain a better learning accuracy compared with other related algorithms.

Povzetek: Razvita je nova metoda iskanja parametrov za metodo SVM.

1 Introduction

Support Vector Machines (SVM) is a new machine learning method based on statistical learning theory and structural risk minimization [1-3]. The core function of SVM identifies the maximal margin hyperplane and a set of linearly separable data, classifies data correctly, so as to maximize the minimum distance between data and the hyperplane. A number of recent studies on SVM attempt to explore simple and efficient methods to solve the problem of maximal margin hyperplane [4-6]. Many of these works study the performance of SVM learning [7-9]. Several kernel functions can be used in SVM, such as linear function, polynomial function, RBF function, Gaussian function, MLPs with one hidden layer and spline.

SVM is used to construct accurate classification models and has been widely applied, such as in handwritten character recognition, web page/text automatic classification, gene analysis and so on [10]. However, there is still no widely accepted way of selecting kernel function and its parameters in SVM learning. The selection of parameters for SVM algorithms usually depends on large-scale search.

SVM learning is a kind of quadratic programming (QP) problem. Despite its advantages, there are a number of drawbacks in selecting hyperparameters in the size of matrix involved in the QP problem. Therefore, this paper proposes a bilinear grid search method to compute the penalty parameter and the kernel parameter (C , γ) of SVM using RBF kernel. This method is efficient in reducing the training space in QP. Bilinear grid search algorithm has the advantages of both bilinear search and grid search. The proposed algorithm expands the search range of (C , γ) so that it can perform SVM learning with

a small size of training samples to construct classification models with high accuracy.

The rest of the paper is structured as follows: Section 2 introduces SVM learning and relevant search strategies; Section 3 proposes bilinear grid search method in SVM learning with RBF kernel; In section 4, we conduct experiments to test the efficiency and applicability of the proposed algorithm; Finally, Section 5 is devoted to concluding remarks and future research recommendations.

2 Search strategy for SVM learning

SVM classification can be described as:

Given:

– A training set of instance-label pairs (x_i, y_i) , $i = 1, \dots, l$, where $x_i \in R^n$ and $y_i \in \{1, -1\}^l$.

Find:

– The solution to the minimum value of $\frac{1}{2}w^T w + C \sum_{i=1}^l \xi_i$,

where $y_i(w^T Z_i + b) \geq 1 - \xi_i$, $\xi_i \geq 0$, $i = 1, \dots, l$.

Here, the training vector x_i are mapped into a higher - (probably infinite-) dimensional space using function ϕ as $Z_i = \phi(x_i)$; $C (C > 0)$ is the penalty parameter of the error term.

Usually, the formation (1) can be considered as the following dual problem:

• Minimize $F(\alpha) = \frac{1}{2} \alpha^T Q \alpha - e^T \alpha$, subject to $0 \leq \alpha \leq C$, $i = 1, \dots, l$,

• $y^T \alpha = 0$, where e is the vector of all ones and Q is a 1 by 1 positive semidefinite matrix. Element $(i,j)^{th}$ of Q is given by $Q_{ij} \equiv y_i y_j K(x_i, x_j)$, where $K(x_i, x_j) \equiv \phi(x_i)^T \phi(x_j)$ is the kernel function. Then, the decision function can be given as: $\text{sgn}(w^T \phi(x) + b) = \text{sgn}(\sum_{i=1}^l \alpha_i y_i K(x_i, x) + b)$. The above definition is employed to minimize the predictable error in SVM learning. Several kernel functions can be used in SVM learning, including linear kernel, polynomial kernel, sigmoid kernel, radial basis function (RBF) kernel (also called Gaussian kernel) etc. This paper selects RBF kernel as the SVM kernel function, i.e., RBF kernel $K(x, y) = \exp(-\gamma \|x - y\|^2), \gamma > 0$. The RBF kernel nonlinearly maps the training data into a higher dimensional space, so it can handle non linear relation between the class labels and the attributes. Keerthi and Lin [10] prove that a linear kernel with a penalty parameter \tilde{C} has the same performance as the RBF kernel with (C, γ) (C is the penalty parameter, γ is the kernel parameter). In addition, the application of sigmoid kernel in SVM learning and the similar parameters to RBF kernel are given by [9].

It is known that the number of hyperparameters influences the complexity of model selection. In RBF kernel, $0 < K_{ij} \leq 1$. However, for polynomial kernels, there are two cases: $\gamma x_i^T x_j + \gamma > 1$ means its value is infinite; $0 < \gamma x_i^T x_j + \gamma < 1$ is the opposite. The authors of [12] believe that since there is no inner product of two vectors, the application of the sigmoid kernel has some limitations.

As mentioned above, there are two hyperparameters in model selection for the RBF kernel [11]: the penalty parameter C and the kernel width γ . We can improve SVM learning by optimizing the parameter pair (C, γ) . Several methods can be used to compute these two parameters [12]. (C, γ) can be computed in the same way as $(\log C, \log \gamma)$. When searching for a good set of $\log C$ and $\log \gamma$, it is usual to form a two-dimensional uniform grid $(n \times n)$ in the training space to find a set of $(\log C, \log \gamma)$ which has the smallest generalization error in SVM classification. This method is called *grid search method*. This method searches for n^2 pairs of (C, γ) .

Keerthi and Lin [11] propose a simple and efficient heuristic method for computing (C, γ) . It forms a unit slope which cuts through the middle part of the good region and searches for a good set of (C, γ) within the good region. Suppose line \tilde{C} is the optimal penalty parameter of linear SVM, follow the procedures below (call it *bilinear search method*) to compute \tilde{C} :

• Search for the best C of linear SVM and denote it as \tilde{C} .

• Fix \tilde{C} , and search for the satisfying (C, γ) $\log \gamma = \log C - \log \tilde{C}$ using the RBF kernel.

Keerthi and Lin [11] have difficulty in deciding the range of $\log C$ for the computation of \tilde{C} in the first step. This paper employs an *improved bilinear search method* to solve this problem by searching for $\log \gamma = \log C - \log \tilde{C}$ with $0.5 \tilde{C}$, \tilde{C} and $2 \tilde{C}$ respectively. The best \tilde{C} is computed from the range of $\log C$.

Grid search is time-consuming. Based on the bilinear search method adopted by Keerthi and Lin [11], we propose an improved bilinear search method to decide (C, γ) . First, identify a 'better' region (the range of $\log C$ is larger than that of [11]), and compute a (C_1, γ_1) pair. Then, invoke an *improved grid search* to obtain a better pair (C_2, γ_2) than (C_1, γ_1) for accurate prediction. It is stated that the grid search method can be improved by a improved grid search method, to obtain a better set of (C, γ) and an accurate SVM model.

3 Bilinear grid search algorithm

In SVM learning with RBF kernel, several methods can be applied to compute (C, γ) . As aforementioned, (C_1, γ_1) of the (coarse) grid search can be optimized using the improved grid search, to acquire a more suitable set of (C_2, γ_2) for training accurate SVM models. Bilinear search method is used to search for the best parameter \tilde{C} in linear SVM. These parameters, $0.5 \tilde{C}$, \tilde{C} and $2 \tilde{C}$ are acquired in this paper, and computed with the related $\gamma_{0.5}, \gamma_1, \gamma_2$ respectively. In [13], the advantage of determining (C, γ) with the improved bilinear search method is also presented.

Due to the complexity of search space, grid search method requires n^2 pairs of (C, γ) to be tried, while bilinear search method requires only $2n$. Compared to bilinear search, grid search method usually has a higher accuracy of prediction. The bilinear grid search method proposed in this paper retains the advantages of these two methods: it attempts to search for (C, γ) with less training points while maintaining not the accuracy of SVM models. Details algorithm is presented as follows:

First, compute the best C using bilinear method and denote it as \tilde{C} . Then, compare $0.5 \tilde{C}$, \tilde{C} and $2 \tilde{C}$ to search for the best parameter pair (C_{bt}, γ_{bt}) among (C_j, γ_j) , using the improved bilinear search method. According to (C_{bt}, γ_{bt}) , invoke a finer search using aimproved grid search smaller grid spacing of $2^{0.25}$ in the scope of $[2^{-2}, 2^2]$ around the best (C_{bt}, γ_{bt}) to obtain $(C_{final}, \gamma_{final})$. Denote $(C_{final}, \gamma_{final})$ as the optimized (C, γ) and use it to train a SVM model with RBF kernel and acquire the objective SVM model with the highest accuracy.

Algorithm: Bilinear grid search algorithm

Input: Training examples

Output: Classification model with the best accuracy

Begin

- 1: Map the training data to the SVM space;
- 2: Select a linear kernel SVM;
- 3: Search for the best C of linear SVM and call it \tilde{c} ;
- 4: for $j = 0.5 \tilde{c}, \tilde{c}, 2 \tilde{c}$ do
- 5: Compute the γ according to $\log \gamma = \log C - \log \tilde{c}$ using the RBF kernel;
- 6: Select the best (C_{bt}, γ_{bt}) from the (C_j, γ_j) ;
- 7: For (C_{bt}, γ_{bt}) , invoke improved grid search to do
- 8: For $k = 2^{-2}$ to 2^2 step $2^{0.25}$;
- 9: Compute their (C_k, γ_k) ;
- 10: Select the best $(C_{final}, \gamma_{final})$ among (C_k, γ_k) ;
- 11: Train the SVM with RBF kernel using $(C_{final}, \gamma_{final})$;
- 12: Obtain the classification model with the best accuracy

End.

In the process, evaluate the accuracy of all models with 10-fold cross-validation. For grid search method, we uniformly discretize (C, γ) within a $[-10, 16] \times [-15, 11]$ region i.e., $27^2 = 729$ training points. For bilinear search method, we search for \tilde{c} using the value of uniformly spaced $\log C$ in $[-10, 16]$. Then, discretize $[-15, 11]$ as values of $\log \gamma$ and check all points that matches $\log \gamma = \log C - \log \tilde{c}$ (compared with bilinear search method, the improved bilinear search method takes all three values of $0.5 \tilde{c}, \tilde{c}$, and $2 \tilde{c}$ to satisfy the bilinear equation).

4 Experimental results

The proposed bilinear grid algorithm has been evaluated and compared to existing algorithms. This section presents the experimental results.

Classification accuracies of grid search, bilinear search, improved bilinear search, and bilinear grid

search are compared in this section. The experiments employ 10 sets of data chosen from UCI database [15]. These data are trained on LIBSVM [16] with four methods respectively, namely the grid search method, bilinear search method, improved bilinear search method and bilinear grid search method.

Table 1 presents the basic information of the 10 data sets. For example, the Breast-cancer (BC) data set includes 9 attributes, 683 examples, and 2 classes. Table 2 demonstrates the model errors of these 4 different search algorithms. Figures inside the parentheses indicate set $(C_{final}, \gamma_{final})$, which is computed by our proposed bilinear grid search. It shows that bilinear grid algorithm is very competitive compared with grid search in terms of testing error. Among these 10 data sets, both bilinear grid search and grid search obtain the same accuracy on 6 data sets (i.e., Breast-cancer, Iris, Vowel, Wine, Wpbc, Zoo); Bilinear grid search trains more accurate models than grid search on 2 data sets (Credit-screening, Letter-recognition). On Diabetes and Wdbc, bilinear grid search obtains higher accuracy compared with grid search, even though the latter obtains higher accuracy during training. On all the 10 data sets, bilinear grid search learns more accurate models than bilinear search and improved bilinear search.

Data set	attribute	example	class
Breast-Cancer (BC)	9	683	2
Credit-screening (CS)	15	690	2
Diabetes (DIAB)	8	768	2
Iris(IR)	4	150	3
Letter-recognition (LR)	16	20000	26
Vowel(VO)	10	528	11
Wdbc	10	569	2
Wine	13	768	3
Wpbc	33	194	2
Zoo	16	101	7

Table 1: Training data set.

Table 3 shows the number of training SVMs required by these 4 different algorithms. For all the 10 data sets, grid search needs to run the same

Data	Grid search	Bilinear search	IB search	BG search
BC	0.027(-3,-3)	0.030(-4,-2)	0.030(-4,-2)	0.027(2.8,-3)
CS	0.130(2,-1)	0.139(3,-1)	0.130(2,-1)	0.128(2.5,-1.5)
DIAR	0.225(0,-4)	0.244(-3,0)	0.234(-3,-1)	0.226(-1.8,-2.5)
IR	0.026(2,-3)	0.046(-2,-2)	0.026(0,-1)	0.026(0,-1)
LR	0.020(10,2)	0.019(5,1)	0.019(6,1)	0.019(6,1)
VO	0.003(3,2)	0.003(6,1)	0.003(6,1)	0.003(6,1)
Wdbc	0.019(3,-5)	0.040(-3,-1)	0.031(-2,-1)	0.021(-0.5,-2.3)
Wine	0.005(0,-2)	0.028(-2,0)	0.016(-2,-1)	0.005(0,-2)
Wpbc	0.164(6,-5)	0.201(1,-3)	0.190(2,-3)	0.164(3.8,-3.5)
Zoo	0.039(10,-9)	0.138(-2,-3)	0.049(0,-2)	0.039(1.3,-3)

IB search: Improved Bilinear search. BG search: Bilinear Grid search.

Table 2: Model error comparison of bilinear grid search with other search methods.

training SVMs for 729 times because it trains SVMs with the same grid 27^2 . Both bilinear search and the improved bilinear search require a smaller number of training SVMs. The number of training SVMs of bilinear grid search algorithm is much smaller compare with grid search algorithm.

From Tables 2 and 3, we can see that bilinear grid search algorithm has the best performance in terms of accuracy and the number of training SVMs. For large data sets, bilinear grid search algorithm is preferable over grid search algorithm, since the former checks fewer points on the $(\log C, \log \gamma)$ two-dimension plane, thus saves computing time. The experimental results show that, with the largest training SVMs, grid search method generates higher accuracy of prediction than bilinear search method because the latter searches a smaller number of training SVMs. Bilinear grid search method retains the advantages of both bilinear search and grid search, thus reducing the number of training SVMs (compared with bilinear search and grid search method), while obtaining a competitive accuracy of prediction. Therefore, it is preferable over grid search.

Data set	Grid search	Bilinear search	IB search	BG search
BC	729	47	87	376
CS	729	53	105	394
DIAB	729	46	84	373
IR	729	49	93	382
LR	729	53	105	394
Vowel	729	54	106	395
Wdbc	729	47	87	376
Wine	729	44	83	372
Wpbc	729	53	105	394
Zoo	729	50	96	385

IB search: Improved Bilinear search. BG search: Bilinear Grid search.

Table 3: Comparison of SVM training times.

Set A	256b_A 2aat 8abp 6acn 1acx 8adh 3ait 2ak3_A 2alp 9api_A 9api_B 1azu 1cyo 1bbp_A 1bds 1bmv_1 1bmv_2 3blm 4bp2
Set B	2cab 7cat_A 1cbh 1cc5 2ccy_A 1cdh 1cdt_A 3cla 3cln 4cms 4cpa_I 6cpa 6cpp 4cpv 1cm 1cse_I 6cts 2cyp 5cyt_R
Set C	1eca 6dfr 3ebx 5er2_E 1etu 1fc2_C fdl_H 1dur 1fkf 1fnd 2fxb 1fxi_A 2fox 1g6n_A 2gbp 1a45 1gd1_O 2gls_A 2gn5
Set D	1gp1_A 4gr1 1hip 6hir 3hmg_A 3hmg_B 2hmg_A 5hvp_A 2i1b 3icb 7icd 1i18_A 9ins_B 1158 1lap 5ldh 1gdj 2lhb 1lmb_3
Set E	2ltn_A 2ltn_B 5lyz 1mcp_L 2mev_4 2or1_L 1ovo_A 1paz 9pap 2pcy 4pfk 3pgm 2phh 1pyp 1r09_2 2pab_A 2mhu 1mrt 1ppt
Set F	1rbp 1rhd 4rhv_1 4rhv_3 4rhv_4 3rnt 7rsa 2rsp_A 4rxn 1s01 3sdh_A 4sgb_I 1sh1 2sns 2sod_B 2stv 2tgp_I 1tgs_I 3tim_A
Set G	6tmn_E 2tmv_P 1tnf_A 4ts1_A 1ubq 2utg_A 9wga_A 2wrp_R 1bks_A 1bks_B 4xia_A 2tsc_A 1prc_C 1prc_H 1prc_L 1prc_M

Table 4: 130 Protein sequences name used in experiments

5 BGS on protein secondary structure prediction

Due to potential homology between proteins in the training and testing set, the selection of protein database for secondary structure prediction is complicated. Homologous proteins in the database may generate misleading results. This is because in some cases the learning method memorizes the training set. Therefore protein chains without significant pairwise homology are used for developing our prediction model. To have a fair comparison, we train and test the same 130 protein sequences used by Rost and Sander [17] and Jung-Ying Wang [18]. These proteins, taken from the HSSP (Homology-derived Structures and Sequences alignments of Proteins) database [19], all have less than 25% of the pairwise similarity and more than 80 residues. Meanwhile, we also train and test the same seven-fold cross-validation are used in Rost and Sander [17] and Jung-Ying Wang [18]. Table 4 lists the 130 protein sequences used for seven-fold cross-validation.

The secondary structure assignment was done using the DSSP (Dictionary of Secondary Structures of Proteins) algorithm [20], which distinguishes between the eight secondary structure classes. The eight classes are reclassified into the following three classes: H (α -helix), I (π -helix), and G (310-helix) are classified as helix (α), E (extended strand) as β -strand (β), and all others as coil (c). Table 5 lists the reclassification process. Note that different classification methods influence the prediction accuracy to some extent, as discussed by Cuff and Barton [21]. For an amino acid sequence, the objective of secondary structure prediction is to predict a secondary structure state (α , β , coil) for each residue in the sequence.

Structural character	Structural name
H	α -helix
G	310-helix
I	π -helix
E	Extended strand
B	β -bridge
T	Turn
S	Bend
C	The rest

(a)

Structural character	Structural name
H	helix
E	strand
C	The rest

(b)

Structural character before conversion	Structural character after conversion
H	H
I	
G	
E	E
B	C
T	
S	
C	

(c)

Table 5: (a) Eight types structural character and name (b) Three classes structural character and name (c) Reclassification between eight types and three classes.

For fair comparison, we train and test same 130 protein sequences used by Rost, Sander and Jung-Ying Wang. These proteins are taken from the HSSP database. The secondary structure assignment was done according to the DSSP algorithms, which are distinguished by eight secondary structures classes, and then three classes.

Moving window and multiple alignment methods are used for encoding. We apply the moving window method for the 17 neighbouring residues in our study. Each window has 21 possible values, including 20 amino acids and a null input. Therefore, the number of data points is the same as the number of residues when each data point has $21 \times 17=357$ values. Before testing these proteins, we employ multiple alignment method to acquire more evolutionary information and protein family information. Having replaced single sequence orthogonal coding, input vector is obtained by aligning the similarity between unknown sequences and known sequences. Then, we can obtain evolutionary information by finding out whether these sequences are homologous.

Figure 1 is an example of using evolutionary information for encoding. we align four proteins. In the gray column, the based sequence has residue 'N' while the multiple alignments in this position are 'N', 'A', 'S' and 'E' (indicating point of deletion in this sequence). Finally, we treat frequencies as the values of output coding. Therefore, the coding scheme in this position is as follows: A = 0.2, S = 0.2, E = 0.2, N=0.4.

Prediction is conducted for the central residue in the windows. In order to allow the moving window to overlap the amino- or carboxyl-terminal end of the protein, a null input was added to each residue. Therefore, each data point has $21 \times 17 = 357$ values and

each data can be represented as a vector. Note that data set RS130 consists of 24,387 data points in three classes where 47% are coil, 32% are helix, and 21% are strand.

An important fact about prediction is that training errors are not significant; only test errors (i.e. accuracy for predicting new sequences) count. Therefore, it is important to estimate the overall performance of a learning method. Previous research proposed different methods to evaluate accuracy. The most common method applied in secondary structure prediction is the overall three-state accuracy (Q_3). It is defined as the ratio of correctly predicted residues to the total number of residues in the database under consideration.

Q_3 is calculated by

$$Q_3 = \frac{q_\alpha + q_\beta + q_{coil}}{N} \times 100 \quad ,$$

where N is the total number of residues in the test data sets, and q_s is the number of residues of secondary structure type s that are predicted correctly.

We carry out several experiments to optimize hyperparameters using bilinear grid search method. The ranges of C and γ are both $[2^{-8}, 2^{-7}, \dots, 2^8]$, and cross-validation fold is 7.

Fig.2 and Fig.3 are the running result charts of command-line and contour. In the chart of command-line, <best c=1.0 g=0.03125, rate=70.8123> the best parameter $(C, \gamma) = (1.0, 0.03125)$, and its classification accuracy is 70.8123%.

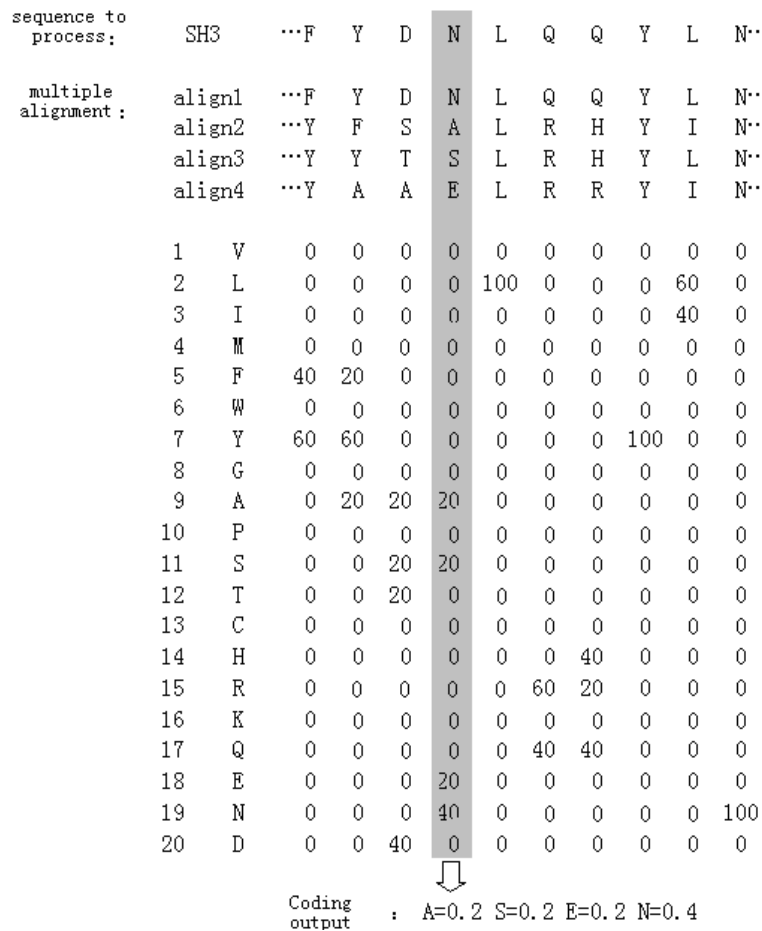


Figure 1: An example of using evolutionary information for coding secondary structure.

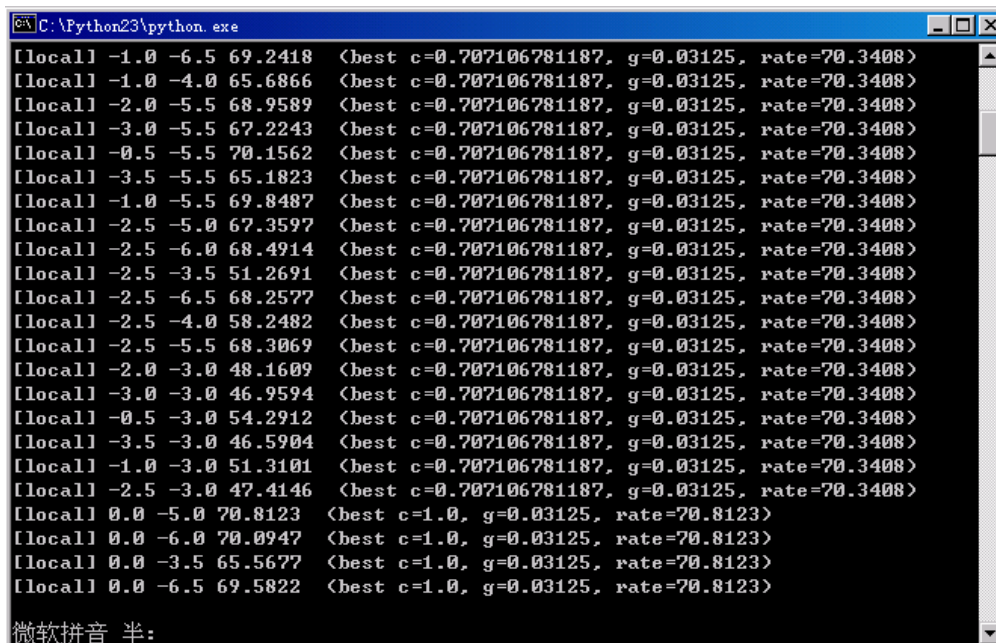


Figure 2: the result chart of command-line.

Table 6 lists the accuracy of different methods on RS130 data set. The average accuracy for bilinear grid search method is 70.8%, which is competitive compared with those methods proposed by Rost, Sander and Jung-Ying Wang. The average accuracy for the method of Rost and Sander [17] is 68.2% , which employs neural

networks for encoding. Other techniques must be incorporated in order to increase accuracy to 70.0%. Jung-Ying Wang utilizes basic SVM to [18] obtain 70.5% of the accuracy .

The experiment used the same data set (including the type of alignment profiles) and secondary structure

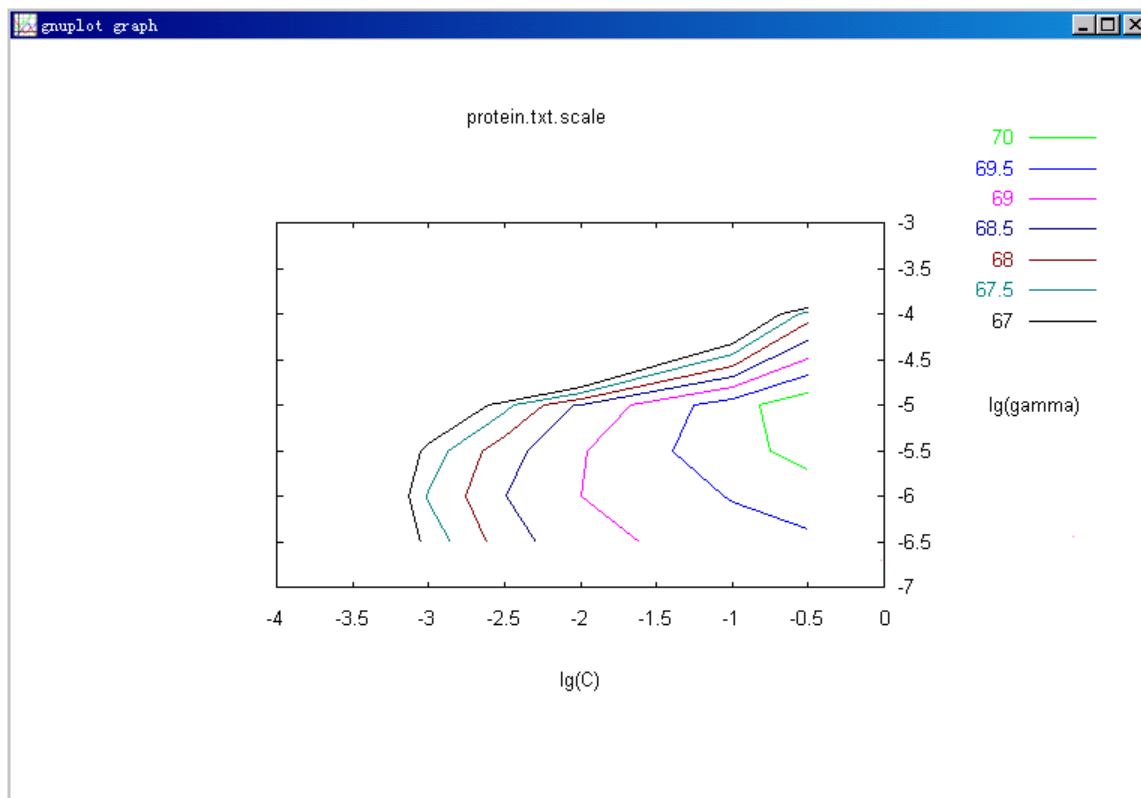


Figure 3: The result chart of contour.

definition (reduction from eight to three secondary structures) as those employed by Rost and Sander [17], and Jung-Ying Wang [18]. The same accuracy assessment of the prior ones is used as well, so as to ensure the fairness of comparison.

Different methods	Secondary structure prediction accuracy %
Neural networks	68.2
Neural networks incorporated With other techniques	70.0
SVM	70.5
SVM with bilinear grid Search method	70.8

Table 6: Comparison of different methods’ accuracy on RS130 data set.

6 Conclusion

In this paper, we demonstrate an approach of optimization in SVM learning. The proposed bilinear grid search method can effectively improve learning performance and enhance the accuracy of prediction. A comparison has been made between grid search method, bilinear search method and bilinear grid search method when selecting optimal parameters for RBF kernel. Experiment results prove that the proposed algorithm retains the advantages of both bilinear search method and grid search method.

With the application of BGSM, the protein secondary structure prediction also obtains better learning accuracy compared with other algorithms.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant No.61273225, No.61100055 and No.31201121, the Natural Science Foundation of Hubei Province under Grant No. 2011CDB233.

Reference

- [1] Vladimir N. Vapnik (1998). Statistical learning theory. J. Wiley & Sons, New York.
- [2] C. Cortes, Vladimir N. Vapnik (1995). Support vector networks. Machine Learning, Vol.20, No.3, pp.273-297.
- [3] Vladimir N. Vapnik (2000). The Nature of Statistical Learning Theory (Second Edition). Springer Press.
- [4] B Schölkopf, AJ Smola (2002). Learning with kernels. MIT Press.
- [5] Kai Zhang, Tsang, I.W. , Kwok, J.T.(2009). Maximum margin clustering made practical. IEEE Transactions on Neural Networks, Vol.20 , No.4, pp. 583 - 596.
- [6] E Blanzieri, F Melgani (2008). Nearest neighbor classification of remote sensing images with the maximal margin principle. IEEE transaction on Geoscience and Remote Sensing, Vol.46, No.6, pp.1804-1811.

- [7] GB Huang, QY Zhu, CK Siew (2006). Extreme learning machine: theory and applications. *Neurocomputing*.
- [8] S. Fine (2001). Efficient SVM training using low-rank kernel representations. *Journal of Machine Learning Research*, Vol.2, pp.243-264.
- [9] K. M. Lin and C. J. Lin(2003), A Study on reduced support machines. *IEEE Trans. on Neural Computation*, Vol.14, No.6, pp. 1449-1559.
- [10] Li. Lin and Zhang Xiaolong (2005). Optimization of SVM with RBF Kernel. *Computer Engineering and Applications*(in Chinese), Vol.29, No. 10, pp.190-193.
- [11] S. S. Keerthi, C. J. Lin(2003). Asymptotic behaviours of support vector machines with Gaussian kernel. *Neural Computation*, No.5:1667–1689.
- [12] O. Chapelle, V. Vapnik et al(2002). Choosing multiple parameters for support vector machines. *Machine Learning*, Vol.46, pp.131–159.
- [13] P. Wang, X. Zhu(2003). Model Selection of SVM with RBF Kernel and its Application. *Computer Engineering and Applications*(in Chinese), Vol.24, pp.72–73.
- [14] H. T. Lin, C. J. Lin(2003), A Study on Sigmoid kernels for SVM and the training of Non-PSD kernels by SMO-type methods. Technical Report, National Taiwan University.
- [15] Blake C., Merz C.(2013), UCI Repository of Machine Learning Databases. <http://www.ics.uci.edu/~mllearn/MLRepository.html>, Dept. of Information and Computer Science, University of California.
- [16] C. C. Chang, C. J. Lin (2013). LIBSVM: A library for support vector machines. Software Available on-line at: <http://www.csie.ntu.edu.tw/~cjlin/libsvm/index.html>.
- [17] B. Rost and C. Sander(1993). Prediction of protein secondary structure at better than 70% accuracy. *Journal of Molecular Biology*, Vol.23, No.2, pp.584–599.
- [18] Jung-Ying Wang(2002). Application of Support Vector Machines in Bioinformatics. Taipei: Department of Computer Science and Information Engineering, National Taiwan University.
- [19] <http://www.cmbi.kun.nl/gv/hssp>.
- [20] W. Kabsch and C. Sander(1983). Dictionary of protein secondary structure: Pat-tern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, Vol.22, No.12, pp.2577–2637.
- [21] J. A. Cuff and G. J. Barton(1999). Evaluation and improvement of multiple sequence methods for protein secondary structure prediction. *Proteins: Struct. Funct. Genet.*, Vol.34, pp.508–519.