Volume 4, Issue 10, October 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Role of Scaling in Data Classification Using SVM

[1]Minaxi Arora, [2]Lekha Bhambhu

[1]M.Tech Scholar

[2]Associate Prof, Department of CSE/IT, JCDCOE, Sirsa, Haryana, India

Abstract: Classification is the most important task which is used in various types of applications today. In machine learning, classification belongs to the act of identifying to which set of categories a given observation belongs. In practice, we can acquire finite samples of data and can perform classification. In this paper, a classifier called the library for support vector machine (LIBSVM) is applied on different scaled values. LIBSVM was developed by Vapnik and due to its excellent features and efficient performance it is being widely used in the field of machine learning. In this paper, we have used heart scale data. For classification, training samples are obtained on which testing is performed. We first scaled data at different scaling values and then comparative results are shown on these different scaling values.

Keywords- SVM, classification, scaling, cross validation.

I. INTRODUCTION

The Support Vector Machine (SVM) was developed by Vapnik and since then it is being widely used in the field of machine learning. Recent studies have shown that SVM (support vector machine) is capable of giving high performance in classification accuracy as compared to other classifiers. It has been used in various real world problems such as handwritten digit recognition, image classification, voice recognition, fingerprint matching, iris recognition, data classification etc. Its performance is very high as compared to other machine learning methods. However, in some cases, performance of SVM varies as we vary the scale. Therefore the user has to perform cross validation in order to properly select the appropriate parameters. This process is called as the model selection process. There are some parameters involved in this process which affect the accuracy of the result obtained.

Importance of SVM is to avoid attributes in greater numeric ranges. Another benefit of applying SVM is to avoid some numerical difficulties during calculations. Before applying SVM, we need to scale data. We need to perform scaling of data before testing it. We have taken a dataset which is divided into two sets- training data and testing data. This data is taken from RSES data set. This paper is organized as follows. In next section, some basic introduction about scaling, SVM, kernel selection and cross validation is given. In next sections results of experiment and conclusion are given.

1.1 SCALING

Before applying data to SVM, it is important to perform scaling of that data. Main purpose of scaling data before processing is to avoid attributes in greater numeric ranges. Other purpose is to avoid some types of numerical difficulties during calculation. Large attribute values might cause numerical problems. Generally, scaling is performed in the range of [-1, +1] or [0, 1]. But before applying scaling on the given data we need to ensure that we apply the same method of scaling on the testing data before testing. For example, we scaled the first attribute of training data from [-10, 10] to [-1, 1] and first attribute of testing data lies in the range [-15, +5] then we must scale testing data to [-1.5, +0.5]. Changing the scale is equivalent o redefining distance. This is the reason of applying the same method to training data and testing data. Scaling is very important in case of variables with large variances. Performance improves with scaling. Code for scaling: svm - scale -l -1 -u 1 -s range tr.txt > tr.txt.scale. Where l and u are the lower and upper limits, .scale file contains the output. Finally, we perform testing of scaled data values. In this paper, we have shown the comparative results on the given data. As the scaling varies, accuracy of the data also varies.

II. SUPPORT VECTOR MACHINE

LIBSVM is a tool for support vector machine (SVM).Purpose of SVM is to let users use SVM easily as a tool [1]. For achieving greater accuracy with SVM, parameter selection is important. Users choose the best parameter for training the training dataset.SVM is basically a supervised machine learning technique. It is a two class classifier. Patterns which are determined by the empty area around the decision boundary which define the distance to the nearest training pattern are called as support vectors. These patterns perform classification. SVM is based on the principle of structured risk minimization to maximize the margin between two classes. Margin is a measurement which defines how well the two classes can be separated. Margin is the distance from the hyperplane to the closest points of two classes. Goal of SVM is to find a hyperplane that maximizes the margin. Format of training and testing data file is:

<label> <index1>:<value1> <index2>:<value2> ...

Each line contains an instance and is ended by a '\n' character. For

classification, <label> is an integer indicating the class label. For regression, <label> is the target value which can be any real number. For one-class SVM, it's not used so it can be any number. The pair <index> :< value> gives a feature (attribute) value: <index> is an integer starting from 1 and <value> is a real number.

Two parallel hyperplanes are constructed on each side of the hyperplane that separates the data. Larger the distance between these hyperplanes better the generalization error will be. Distance between the hyperplanes is 2 / |w|. So to maximize the distance between the hyperplanes we need to minimize w.

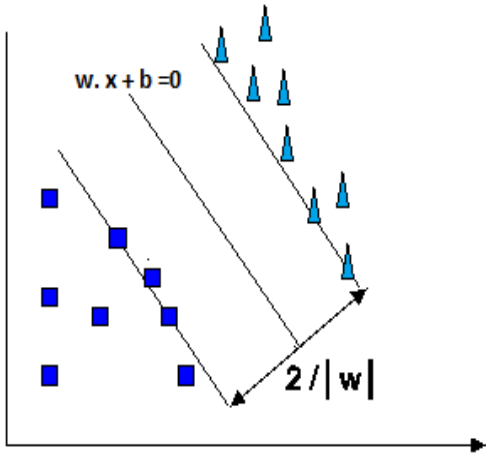

Figure 1 Separating hypeplane and margin

Dividing hyperplane is

w. x + b =0

Where b is scalar and w is k-dimensional vector. Adding a parameter b helps to increase the margin between the hyperplanes.

1.2 KERNEL SELECTION

$K(x_i, x_j) \equiv \Phi(x_i)^T \Phi(x_j)$ is called the kernel function. There are many kernel functions available in SVM. Which kernel function to choose is also an issue. Some of these kernel functions are:-

- Linear kernel: $K(x_i , x_j) = x_i^T x_j$
- Polynomial kernel: $K(x_i , x_j) = (\gamma x_i^T x_j + r)^d , \gamma > 0$
- RBF kernel : $K(x_i , x_j) = \exp(-\gamma \| x_i - x_j \|^2) , \gamma > 0$
- Sigmoid kernel: $K(x_i , x_j) = \tanh(\gamma x_i^T x_j + r)$

Generally RBF is the first choice among all types of kernels.RBF kernel is used in following cases:-

a) When the number of hyperparameters are less. A polynomial kernel uses more number of hyperparameters as compared to RBF kernel.
b) When the relation between class labels and attributes is non-linear. Linear kernels can't handle this case.
c) RBF kernel has less numerical difficulties. In case of polynomial kernels value may go to infinity or zero.

1.3 CROSS VALIDATION

There are two parameters which are considered when we are using RBF kernels. These are c and ϒ. But problem is that we don't know which values are good there we need to find such values which give good results. Main issue in SVM is to find such values of c and ϒ so that classifier can accurately predict testing data. Accuracy determines how well it is classifying the unknown data. Procedure used is cross validation. If we are using x-fold cross validation scheme, we divide the training dataset into x subsets of equal size. We train the classifier onto some of the datasets and after training we perform testing on the remaining datasets. For example, if we have x subsets we train the system on x-1 subsets and test the one subset. Accuracy of this method is calculated by finding the percentage of data which is being correctly classified. Cross validation helps to avoid overfitting problem. And classifiers that don't have overfitting problem gives better accuracy.

III. RESULTS OF EXPERIMENT

Experiments are performed on a heart scale dataset. Data set was taken from RSES data set. This dataset consist of 8 attributes and each attribute have some value. Format of dataset is

<label> <index1> :< value1> <index2> :< value2>...

Where label indicates a class label. Classifier will perform scaling on this data. From this scaled data model file will be produced. We will test this data for accuracy and result of this will be in predict file.

We applied different range of scales on this data set and achieved different accuracy values with different ranges. RBF kernel is used in this method. RBF kernel is the best choice among all the available kernel functions. SVM type used is c-svc and value of gamma is 0.125. Table1 lists the results. Table shows how the data correctly classified and thus accuracy achieved varies with the scale values.

Table 1 comparison of accuracy achieved

Range	Data correctly classified	Total data	Accuracy achieved
[-1,1]	148	200	74%
[0,1]	146	200	73%
[-5,5]	150	200	75%
[-10,10]	143	200	71.5%
[-20,20]	135	200	67.5%

Figure 1 shows the results achieved as the scaling varies accuracy achieved vary.

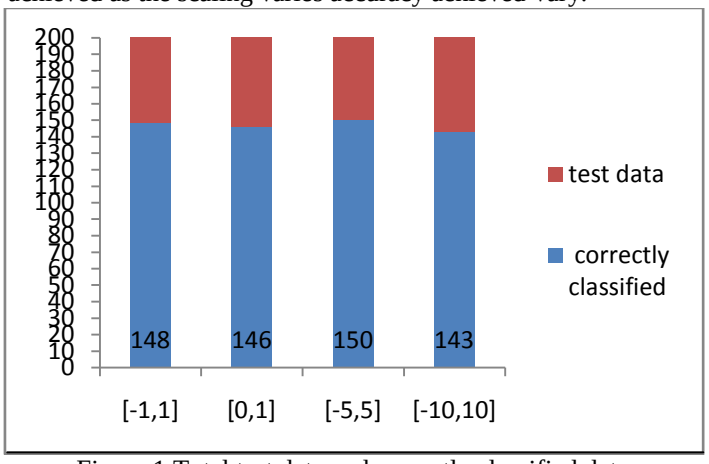

Figure 1 Total test data and correctly classified data

IV. CONCLUSION

In this paper, we have shown the comparative results of accuracy achieved with different ranges. Table1 shows the results achieved. Results are encouraging. Performance of SVM depends on the best choice of parameters. We have used RBF kernel in our experiment and it is the best choice when number of hyperparameters are less and the relationship is non-linear. Results show that as we increase the scale accuracy decreases.

V. FUTURE SCOPE

In future, this technique can be applied on more no. of training datasets. As we increase the no of training data, accuracy also increases. So, in future this technique can be used to enhance performance.

REFERENCES

[1] Chang, C.-C. and C. J. Lin (2001). LIBSVM: a library for support vector machines. http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[2] Junhua Zhang and Yuanyuan Wang: "A rough margin based support vector machine"

[3] Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin. "A Practical Guide to Support Vector Classification". Deptt of Computer Sci. National Taiwan Uni, Taipei, 106, Taiwan http://www.csie.ntu.edu.tw/~cjlin 2007

[4] Silvia Rissino and Germano Lambert-Torres"Rough Set Theory Fundamental Concepts, Principals, Data Extraction, and Applications "Federal University of Rondonia, Itajuba, Federal University, Brazil

[5] Durgesh K.Srivastava, Lekha Bhambhu,"Data Classification using support vector machine", Journal of Theoretical and Applied Information Technology,2009.

[6] Durgesh Srivastava, Shweta Bhalothia: "Efficient Rule Set Generation Using K-Map & Rough Set Theory (RST)"

[7] Daijin Kim:"Data classification based on tolerant rough set". 2001 Pattern Recognition Society. Published by Elsevier Science Ltd

[8] Zdzisław Pawlak:" Rough set theory and its applications". Journal of Telecommunications and Information Technology 3/2002.

[9] Praveer Mansukhani, Sergey Tulyakov, Venu Govindaraju "Using Support Vector Machines to Eliminate False Minutiae Matches during Fingerprint Verification" Center for Unified Biometrics and Sensors (CUBS)

[10] Mahesh Jangid, Renu Dhir, Rajneesh Rani, Kartar Singh "SVM Classifier for Recognition of Handwritten Devanagari Numeral" Department of Computer Science and Engineering Dr. B R Ambedkar National Institute of Technology Jalandhar, India

[11] Eva Kovacs, Losif Ignat, "Reduct Equivalent Rule Induction Based On Rough Set Theory", Technical University of Cluj-Napoca.

[12] Z. Pawlak, Rough set, Int. J. Inform. Computer Sci. 11(1982) 341-56.

[13] V. Vapnik, The Nature of Statistical Learning, Springer-Verlag, New York, 1995.

[14] Ming-Hsuan Yang "Gentle Guide To Support Vector Machines"

[15] Sayan Mukherjee"classifying microarray data using support vector machines"

