International Journal of Soft Computing and Engineering (IJSCE)
ISSN: 2231-2307, Volume-4 Issue-5, November 2014

Reducing the Size of Very Large Training Set for Support Vector Machine Classification

Mahmoudreza Ahmadi, Hamidreza Ghaffari

Abstract— ***Normal support vector machine (SVM) algorithms are not suitable for classification of large data sets because of high training complexity. In this paper, we introduce a method based on edge recognition technique to find low-value data, where to keep input data distribution, we use clustering algorithm like k-means to compute clusters centers. Data is selected through edge recognition algorithm and cluster centers, are used to build a training data set. Reconstructed data set with small size, increase the speed of training process procedure without decreasing classification precision. But, as we used k-means algorithm, it is required to initially specify the number of classes. We try to get a proper procedure by improving edge recognition algorithm to reduce data, also using hierarchical clustering algorithm and similarity percent to compute number of clusters instead of using k-means algorithm, and compare results of these two algorithms.***

Index Terms— Support vector machine, k-means, optimization, edge recognition, cluster, hierarchical, similarity percent.

I. INTRODUCTION

Svm is one of the most salient programs in the case of utilizing kernel techniques. Most kernel methods are formulated based on Quadprog programming. If we assume training samples, N then the temporal complexity is O(n3) and spatial complexity is O(n2). When confronting the problem of multi-class classifier using kernel method, the problem raised from these temporal complexities, goes doubled. Grammer and Singer [1] algorithm is one these methods that is not applicable for large data sets. Therefore, the main problem is that how we can decrease temporal and spatial complexities for large training data sets. In order to reduce spatial and temporal complexity, many improvement methods are proposed. Among those is getting low-rank approximation over kernel matrix by one of greedy approximation [2], sampling and or decomposition matrix. Although the results of kernel matrix is good but is not used in act. Another modifying method is chunking or advanced decomposition methods [3]. Though chunking method needs full modification of non-zero Lagrange multiplying system, but still setting kernel matrix up is exceeded than present memory. The third type of methods, to avoid QP problem, is CVM [1] algorithm [4], scale methods [5] and Lagrange SVM(LSVM). However for nonlinear kernels already needs large matrices. Another type of these methods is that scale down the training data before implementing SVM, and here we concentrate on this type of method.

Manuscript Received on November 2014.
Mahmoudreza Ahmadi, MA. Student, Department of Computer Engineering, Islamic Azad University of Ferdows Ferdows, Khorasan, Iran.
Dr. Hamidreza Ghaffari, Department of Computer Engineering, Islamic Azad University of Ferdows Ferdows, Khorasan, Iran.

Povlow[6] and Collobert[7] composed small Svms using neural networks based on Gater. The problem of using these methods is that how we can recognize irrelevant samples in training data sets. Most of above methods can decrease size of training data sets but yet many of irrelevant training data are used as training data. Therefore, we need a effective manner to keep relevant data. In this paper we use an edge recognition technique [8] that decrease the size of training data set and benefited from clustering to keep input data feature distribution. In above mentioned method, k-means algorithm is used. In k-means algorithm, at first we must specify number of clusters. In image processing, edge recognition method is a technique to reduce information and filtering unhelpful data. Whereas keeping important image properties, we can use this method to reduce training data set. In this paper we tried to initially solve problem of determining cluster numbers using hierarchical algorithm and characterizing similarity percent in each step of clustering. Also, through changing edge recognition algorithm ,we try to improve its performance. Finally, new training data sets include data recognized by improved edge recognition algorithm, and cluster centers that gained from hierarchical clustering algorithm. Through exerted changes, the number of two parameters for precision adjustment of edge recognition algorithm and cluster numbers are changed to one parameter i.e. precision adjustment of edge recognition algorithm parameter. Also we improved he performance of edge recognition algorithm. At last, we apply this method to improve training time of Singer and Grammer algorithm in order to classifying three-classes data, synchronously. The organization of this paper is as follow. In next section, we proceed to sample reduction method via edge recognition algorithm and way of improving it. Finally we present the results of experiments, hypothesis test and conclusion.

II. EDGE RECOGNITION ALGORITHM

Because Svm is a QP problem, spatial and temporal complexity are O(n2) and O(n3) respectively where n is number of training data. Thus, decreasing size of training data sets prior to selecting Support vectors can be valuable from speed and amount of required space and memory aspects. Meanwhile, reducing training data should not effect of classifier results. Separator line in classifier Svm builds based on support vectors. In order to keep same accuracy, maintaining samples that could be support vectors, is so critical [9]. Assume that we can state total data set as a picture and each class has individual color. We can assume decision bound as a edge on picture. According to Svm theory, data close to separator line have more possibility for support vectors. Therefore, we can suppose data close to decision bound as data that are close to edge. Edge identification is a

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

term in image processing and machine vision. Recognizing bounds edge is a critical problem in image processing.

Similarly, are important in problem of classifying data close to decision edge, and these samples have more possibility for support vectors. Thus, it is required to keep these data in training data reduction process. In image processing, the aim of edge recognition technique is to identifying some point in digital images where image brightness changed sharply or generally, have discontinuity. Edges are regions including intense contrast [10]. Edge recognition can significantly reduce data rate and keep important properties and features. In classification problems, our aim of identifying intense changes between different classes is gaining important data and samples around boundary. Edge identification in classification problems is simpler than image processing. Discontinuity of edge recognition in image brightness is according to: discontinuity along level direction, changing basic features and changes in Scene illumination although in classification problems, we only consider changes in classes' label. A typical edge may contain, for example, boundary between red data block and a blue data block. Similarly, a typical classification problem may be binary. In image processing, we need to look for neighbor pixels to find a pixel with changes in light intensity and color. In edge recognition model, this rule has changed, where we should find M neighbors. As we can see in Fig. 1, we assumed M=5 for a given training sample named P. if some of its neighbors have the label of other class, then P select and keep as edge data.

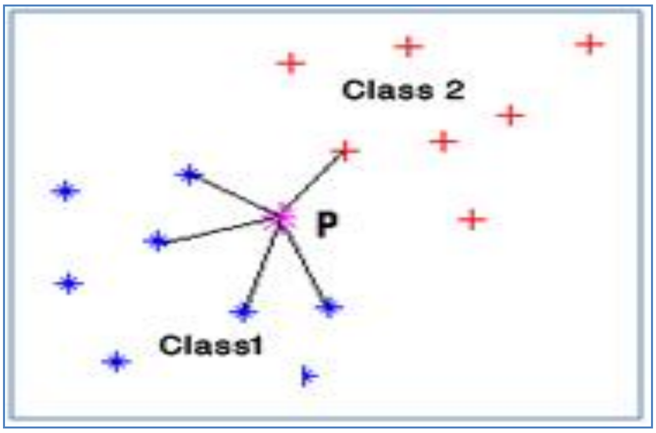


Fig. 1 Assumed M=5 for a Given Training Data Named P

In this case, due to using edge detector on training data set, a data set is obtained that is around the separator boundary of both classes. However, implementing edge recognition method on training data set can decrease being processed data and may filter low-value information but critical data near separator line, are kept. Because support vectors are data that placed around boundary, edge recognition method can maintain most support vectors and put any improper effect on classification outcome. Sometimes using output data from edge recognition method to train Svm algorithm can face classifier to over-fitting. In other words, classifier may be improper for solving all data thus we often need to add some data to input data set in order to distribute structural features. Therefore, we use clustering method for each class and keep cluster centers as critical data. K-means clustering is an algorithm for grouping according to data attributes into k group where k is and integer and positive number. Grouping is done through minimizing sum of data-distance squares to center of each cluster. Selected data via edge recognition algorithm and cluster centers via k-means algorithm are used for training data. This process has four steps:

Step1: Start with decision about m variable (the number of a data neighbors).

Step2: Using edge recognition algorithm to selecting data that are around edge. For this right, finding m nearest neighbor for each data in training set and checked the label of neighbors. If one of them have different label from test data label, test data kept and otherwise would be deleted.

Step3: Making new training data set with data selected by edge recognition algorithm and resulted clusters center via k-means method. We can see the procedure of this algorithm in Fig. 2. It is inevitable that this method can decrease the size of training data sets and speedup training process of Svm algorithm.

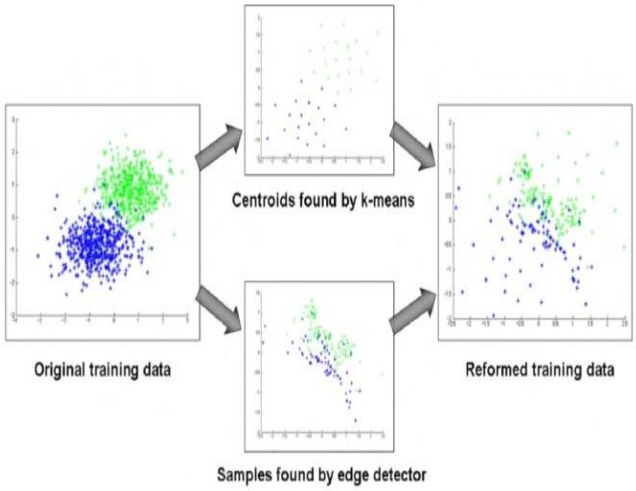


Fig. 2 Select Data with Edge Recognition Algorithm

A. Disadvantageous of Edge Recognition Algorithm

Considering above algorithm in section II, while algorithm efficiency, we must consider two cases.

- ***Determining cluster numbers***

In this algorithm due to using K-means algorithm to cluster data, we must initially specify some of them which can possible for low number of data or data that their distribution process is known. But this is impossible for large and unknown data sets and if this guess is true, yet algorithm may run in non-optimum way, also due to variable response of K-means algorithm, in each run, selected representatives are different.

- ***Data deletion***

As we describe edge recognition algorithm, in step1 a number of neighbors are investigated based on basic data to distinguish class labels. Because of data deletion, the order of this job is very important. Assume that selected data have 5 neighbors with similar class labels and neighbor 6th has different label. According to algorithm, these data must be deleted because has 5 neighbors with same type of itself. But if one of the neighbors removes early than our selected data, then neighbor 6th of that data selects as neighbor 5th and as has different label, our selected data will be deleted. In Fig. 3, if initially we investigate sample 1, then according algorithm this data should be removed but if initially investigate sample 2 then sample 1, because sample 2 removed according to algorithm so the color of 5th neighbor of sample 1 would be red, therefore

Published By:
Blue Eyes Intelligence Engineering & Sciences Publication Pvt. Ltd.

don not remove as recognized edge. Considering this example we can see that the order of removed data is important.

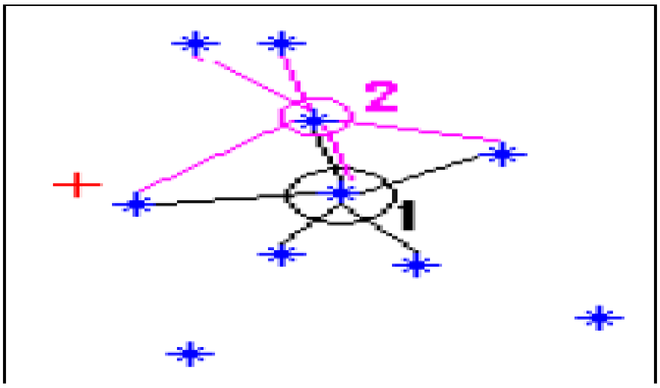


Fig. 3 Process of Data Deletion

III. RECOMMENDED ALIGORITHM

A. Computing Clusters Number

According to given explanations in section 2.1.1, we use hierarchical algorithm instead of K-means algorithm, to clustering data. We use similarity percent index to find cluster number according to input data. In this way, in addition to find algorithmically clusters number and prevent any guess, we can obtain clusters centers that are independent of program running times (K-means algorithm have different response depend on program running times). Therefore, there is no need to initialize parameters that specify number of clusters. To obtain clusters number through this method we can analysis dendrogram plot on different levels to get some data clusters. But what is important for us in this paper is that in what level the plot would be broken and how many clusters could be produced. Each level in this plot or in other words, each of these data that want to build a cluster and merge together, mixed under a index called similarity percent where the bigger index implies to more similarity between two clusters and must merge together. This index is computed by following formula:

$$S_{ij} = 100 * \left(1 - \frac{d(i,j)}{M}\right) \qquad (1)$$

Where d(i,j) is distance between two clusters ith and jth and M is maximum distance or similarity matrix in step of merging two clusters. S is similarity percent of two clusters i and j where 75-80% range is a proper justification for similarity of two clusters. Therefore we can continue algorithm until S is higher than similarity number, and after stopping algorithm, obtain clusters number and data of each class to choose representative.

B. Modifying Edge Recognition Algorithm

According the problem of edge recognition algorithm in section II, ever change must imposed by second labeling and data deletion must done together or in other words, step2 must done on primary data set.

Thus, algorithm in section 2 changes to:

Step 1: begin with decision about variable m (numbers of a data neighbors)

Step2: using edge recognition algorithm to select data that are around edge

- Finding m nearest neighbor for each data in training set
- Checked the label of neighbors. If one of them has different class label from test data label, then test data takes K label, otherwise takes D label.

Step3: removing data including label D

Step4: running hierarchical clustering algorithm and similarities percent for achieve clusters, to compute their centers for all data.

Step5: making new training data sets via data that are selected by edge recognition algorithm and clusters centers that are obtained using similarity percent.

IV. EFFECT RATE OF PARAMETER M

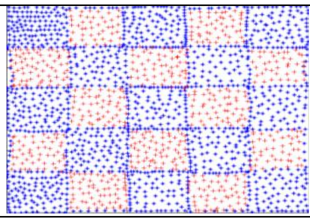
This algorithm only needs to parameter m and must be defined in the beginning of implementation. If m is large, all present data in classes boundary would be selected and much larger is this value, more data around this boundary would be selected, otherwise, if m is small, more boundary data are removed, then some important data like support vectors may be removed. Therefore, a proper m can play an important role in optimum program implementation. Of course, if this parameter is unreliable, large m's relative to small m's, have less negative effect in response accuracy.

V. RESULTS

It must be noted that all experiments are implemented on computer systems with following specifications and Matlab 2011 software: *P4: Intel Core 2 Duo E7500 2.93GHZ, 4 G Ram DDR3 1033*. In first step, we experiment on raster data. All data are created randomly. We implemented edge recognition algorithm by first modified method. As Table1 shows, number of removed data in modification algorithm is higher where there is no initialization for cluster n umber and according similarity percent, this is done by algorithm. Fig. 4 shoes a profile of primary data and reminded data by both algorithms.

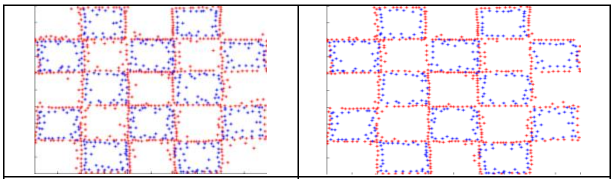
Table 1. Experiment on Raster Data

	The basic algorithm	Improved Algorithm
Total	1923	1923
The number of deleted	957	1062
The number of remaining	966	861
Run Time (Second)	110	100



Initial data(a)

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.



Removing data via primary algorithm(b)	Removing data via modification algorithm(c)

Fig. 4 Implementing Proposed Algorithm on Raster Data

In step2, experiments are done on random data that are compared by both algorithms together using classifier of two-classes Svm and each time, error rate classification is done by proper *Crossvalind* software, 10 times computation and finally their means is showed in Table2. Error rate measure function is as follows:

Crossvalind('LeaveMOut',TArray_Lable,10);

Table 2. Comparing Integrity Rate and Support Vector Numbers among 2 Algorithms

	Svm Classifier		
	The original data set(No reduction)	Reduce with The basic algorithm	Reduce with The Improved Algorithm
Total	425	425	425
The number of deleted	0	215	246
The number of remaining	425	210	179
Number of support vectors	76	76	76
Accuracy rate	91%	81%	84%

As you can see in Table2, support vector numbers is fixed in all 2 algorithms and integrity rate of modified algorithm is higher than initial algorithm. Data classification procedure has shown in Fig.5.

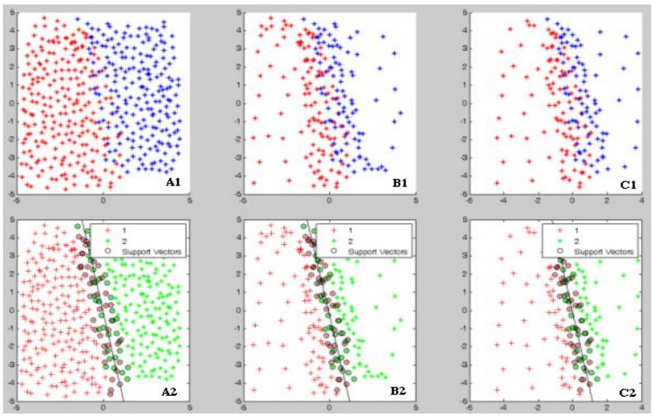


Fig. 5 The Data Classification

In Fig.5 (A), consider basic data. In part (B) and (C) you can see data that are selected by initial algorithm and modified algorithm, respectively. In second row, Svm classifier for all three cases has shown. In step3, we present experiments for seeds multi-classes data sets by UCI Machine Learning Repository that includes three classes and eight features. In this experiment we use features number 5, 6 and investigate both algorithms, then present results in Table 3.

Table 3. Implementing Algorithm on Reduced Data Sets

Data set Seeds	Grammer Algorithm		
	The original data set(No reduction)	Reduce with The basic algorithm	Reduce with The Improved Algorithm
Total	210	210	210
The number of deleted	0	46	68
The number of remaining	210	164	142
Accuracy rate	82%	83%	83%

How data segmentation is showed in Fig6. Other comments about order of images are same as before.

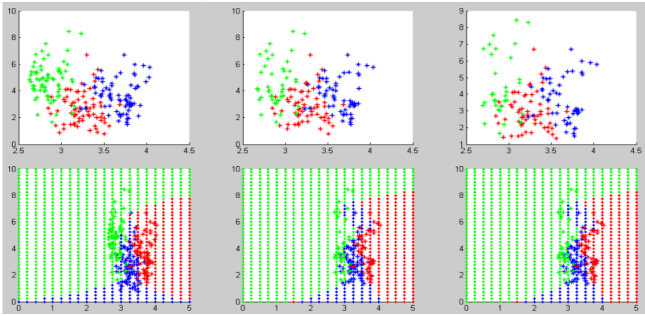


Fig. 6 Grammer Algorithm Classification Procedure on Modified Data Set

Finally, proposed algorithm is compared with some algorithms in the case of data reduction on 20 data sets. Results have shown in Tables 4, 5. Proposed algorithm is compared with edge detection algorithm, modified NN[13], NCN[14] and Crisp[9]. All resulted data sets by Svm classifier are tuned and their RBF kernel and integrity rate via k-fold are computed. (tuned Svm is usual Svm algorithm except can find best C and gamma values by several iterations on a data set and accordingly, run classification). As Tables 4, 5 have shown, however precision is kept, proposed algorithm can identify and remove more low-value data. Also in diagrams 1, 2 rate of precision and data reduction of above algorithms are showed. In diagram1, results of integrity rate related to running tuned Svm algorithm in basic data set and modified data set by above algorithm have shown. In diagram2, data reduction mean rate of algorithms are showed. Among those, Edge reduction algorithm has the highest percent among these methods.

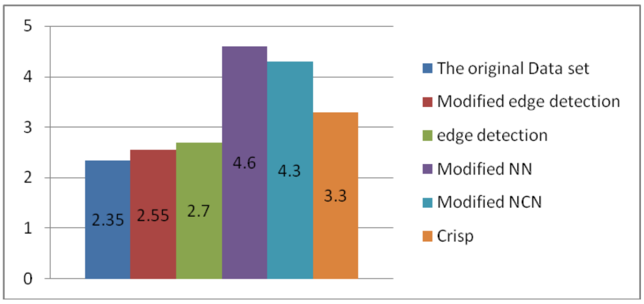


Diagram 1: Average Rank of Algorithms in Accuracy Rate

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

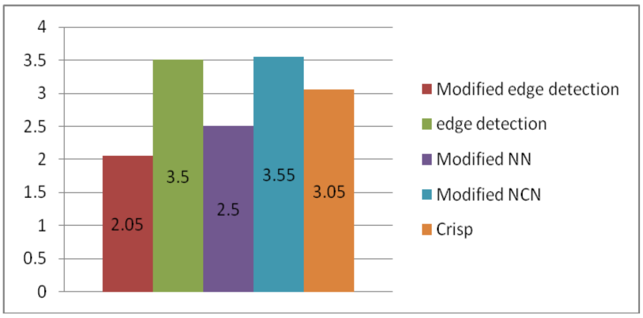


Diagram 2: Average Rank of Algorithms in Reduce Rate

According diagram1, and comparing integrity rate of algorithms in above data sets, Crisp and edge reduction methods are in the same level and Modified NN algorithm has the least level. But as implementing Crisp algorithm needs to manual adjusting of 4 parameters to reach optimum result and propose algorithm needs to initializing one algorithm, so proposed algorithm is preferred. Also, if we concentrate on diagram2 we can see that proposed algorithm has the highest reduction rate. As we said, decreasing data does not have any effect on integrity and validity of algorithm classification and it shows that deleted data don't have any effect on classifier.

Table 4. Integrity Rate of Algorithms on Reduced Data Sets by Above Algorithms

	Data set	Classification accuracy rate by Teun Svm											
		The original Data set		Reduced data set									
				Modified edge detection		edge detection		Modified NN		Modified NCN		Crisp	
		Accuracy Rate	Rank	Accuracy Rate	Rank	Accuracy Rate	Rank	Accuracy Rate	Rank	Accuracy Rate	Rank	Accuracy Rate	Rank
1	Cloud	0.91	3	0.97	1	0.93	2	0.86	4	0.91	3	0.93	2
2	Ecoli	0.93	2	0.75	4	0.74	5	0.60	6	0.85	3	0.94	1
3	Glass	0.76	2	0.71	3	0.71	3	0.65	4	0.62	5	0.80	1
4	Ionosphere	0.92	4	0.99	1	0.96	2	0.85	5	0.82	6	0.93	3
5	IRIS	0.92	4	0.93	3	0.92	4	0.85	5	0.94	1	0.94	2
6	Magic	0.90	1	0.85	3	0.86	2	0.85	3	0.82	4	0.85	3
7	Mammographic	0.69	1	0.68	3	0.70	2	0.53	5	0.50	6	0.61	4
8	sonar	0.66	3	0.94	1	0.91	2	0.65	4	0.50	6	0.60	5
9	segmentation	0.80	2	.093	1	0.93	1	0.71	5	0.76	3	0.75	4
10	Wine	0.88	4	0.99	1	0.97	2	.084	5	0.82	6	0.89	3
11	indians-diabetes	0.58	3	0.66	2	0.68	1	0.50	5	0.45	6	0.52	4
12	Wall_Following	0.96	1	0.79	4	0.85	3	0.65	5	0.58	6	0.87	2
13	appendicitis	0.88	1	0.79	3	0.71	5	0.68	6	0.78	4	0.82	2
14	banana	0.62	3	0.69	2	0.73	1	0.59	6	0.60	5	0.61	4
15	bupa	0.84	1	0.69	5	0.64	6	0.80	4	0.84	2	0.81	3
16	haberman	0.79	1	0.56	6	0.57	5	0.70	2	0.68	3	0.59	4
17	phoneme	0.64	3	0.67	2	0.72	1	0.57	5	0.54	6	0.59	4
18	pima	0.83	1	0.64	3	0.63	4	0.64	3	0.65	2	0.54	5

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

19	wdbc	0.88	4	0.97	1	0.94	2	0.82	6	0.90	3	0.88	5
20	heart	0.73	3	0.75	2	0.76	1	0.62	4	0.59	6	0.60	5
Mean of rank		2.35		2.55		2.7		4.6		4.3		3.3	

Table 5. Size of Data Sets after Running Sample Reduction Algorithms

Row	Data set	The original Data set size	Reduce rate									
			The size of the Reduced data set									
			Modified edge detection		edge detection		Modified NN		Modified NCN		Crisp	
			Reduce Rate	Rank	Reduce Rate	Rank	Reduce Rate	Rank	Reduce Rate	Rank	Reduce Rate	Rank
1	Cloud	1024	0.99	1	0.98	2	0.66	3	0.22	5	0.27	4
2	Ecoli	336	0.34	2	0.26	3	0.56	1	0.08	5	0.21	4
3	Glass	214	0.30	1	0.20	3	0.29	2	0.14	4	0.14	4
4	Ionosphere	351	0.12	3	0.04	4	0.68	1	0.57	2	0.02	5
5	IRIS	150	0.79	1	0.67	2	0.18	3	0.12	5	0.16	4
6	Magic	2500	0.32	3	0.25	5	0.41	1	0.27	4	0.36	2
7	Mammographic	961	0.47	4	0.38	5	0.61	1	0.53	3	0.57	2
8	sonar	208	0.33	1	0.25	2	0.05	3	0	5	0.02	4
9	segmentation	210	0.33	1	0.18	3	0.03	4	0	5	0.23	2
10	Wine	178	0.33	1	0.29	4	0.06	5	0.31	2	0.30	3
11	indians-diabetes	768	0.30	4	0.20	5	0.51	3	0.66	1	0.65	2
12	Wall_Followi	5456	0.55	1	0.42	4	0.47	3	0.48	2	0.48	2
13	appendicitis	106	0.68	1	0.62	2	0.58	3	0	5	0.41	4
14	banana	5300	0.67	2	0.62	3	0.79	1	0.59	4	0.49	5
15	bupa	345	0.14	4	0.09	5	0.45	1	0.29	3	0.43	2
16	haberman	306	0.33	2	0.26	5	0.30	3	0.29	4	0.45	1
17	phoneme	5404	0.60	3	0.52	4	0.63	1	0.62	2	0.62	2
18	pima	652	0.27	4	0.21	5	0.52	3	0.64	2	0.65	1
19	wdbc	569	0.90	1	0.86	2	0.58	3	0.21	4	0.10	5
20	heart	270	0.43	1	0.33	2	0.07	5	0.18	4	0.19	3
Mean of rank			**2.05**		**3.5**		**2.5**		**3.55**		**3.05**	

VI. STATISTICAL TEST

When some data (a small sub-set from society)form society are available and we want to decide about some characteristic of society include mean, variance, mean difference ratio, … statistical tests are proper approaches for this case. In order using parametric tests, a stochastic sample must have following characteristics:

- sample is selected stochastically
- minimum 20 samples

According to results in section 8 and in order to running statistical test to achieve total index to compare above methods for cited data sets, also have a total view over society data, we use classic hypothesis test type II.

R=(t: t>tα , n-1)

Number of data sets and procedure of selecting data; provide the condition for applying this test. We define two variables as follow:

X_{ij} : Integrity rate of ith algorithm for jth set

D_{1i}^{j} : Difference between integrity rate of first algorithm (proposed algorithm) and ith algorithm for data set jth:

$$D_{1i}^{j} = X_{1j} - X_{ij}$$

Where j =1... 20 and i=2… 6. (3)

In this test, determining critical section is based on following test statistical:

$$T_i = \frac{D_{1i}}{(s_{1i} / \sqrt{n})} \tag{4}$$

In this formula, Ti of test statistic for first algorithm (proposed algorithm)is with ith algorithm. Dij is the mean of difference between integrity rate of ith and jth algorithm :

$$D_{1i} = \frac{1}{n-1}\sum_{j=1}^{n} D_{1i}^{j} \qquad n = 20 \tag{5}$$

Also, S_{1i}^{2} is the variance of difference between integrity rate:

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

$$S_{1i}^2 = \frac{1}{n-1}\sum_{j=1}^{n}(D_{1i}^{j} - D_{1i})^2 \qquad n = 20 \qquad (6)$$

Where in all formulas, n represents the number of tested data sets. Regarding to statements, all conditions are provided for running this test. Total procedure is that we compute test statistic (Ti) of proposed algorithm for all algorithms and data sets in both precision rate and reduction rate regarding to test hypothesizes:

H_0: integrity rate difference mean is 'zero'; $\mu_{D1i}=0$

H_1: integrity rate difference mean for proposed algorithm is higher than integrity rate difference mean of ith algorithm; $\mu_{D1i}>0$

Test hypothesis for computing statistic in order to rate reduction is as above but instead of computing integrity arte difference, we compute data reduction rate. Considering above definitions, Ti is obtained for all algorithms and data sets of later section and results have shown in Table6.

Table 6. Results of Test Statistic

classification SVM					
Test statistics	Main data set	edge detection	Modified NN	Modified NCN	Crisp
The rate of correctly	0.47	0.58	4.45	2.96	1.75
The rate of decrease	8.91	10.97	0.62	1.97	1.78

Considering accepts and rejects area of test with α=0.05, that by using Matlab software command or reference to above Table, accept/reject boundary is 1.7291. Therefore, conditions of test hypothesis are provided in Table7.

Tinv(0,95،19)=1,7291 α=0,05

Table7. Results of Test Conditions

Test statistics	Original data set		edge detection		Modified NN		Modified NCN		Crisp	
	H0	H1	H0	H1	H0	H1	H0	H1	H0	H1
The rate of correctly	**pass**	**Fail**	**pass**	**Fail**	**Fail**	**pass**	**Fail**	**pass**	**Fail**	**pass**
The rate of decrease	**Fail**	**pass**	**Fail**	**pass**	**pass**	**Fail**	**Fail**	**pass**	**Fail**	**pass**

As results of Table7 show, proposed algorithm regardless to basic data set size has the same precision as running Svm algorithm on primary data set. It shows that in deletion process, data that are removed from set, have not effect on training procedure and known as low-value or even non-value data. Generally, considering comparison in Table7, we can say that precision of proposed algorithm remained fixed and its performance is more better than other algorithms, also, by looking at results of second row of Table7, we can see that proposed algorithm in addition to proper performance in removing low-value data, can remove much data whereas keep precision compare to other algorithms. Finally, if we put both Table 6,7 together, it will apparent that proposed algorithm, in addition to removing more data, has same/higher precision rate as/than other algorithms.

VII. CONCLUSIONS

In this paper we improved a classifier method to solve problems with big data set. In Svm, support vectors are samples near decision boundary. Here we used edge recognition method that adopted from image processing. Samples that are detected by edge recognition algorithm are similar to support vectors and their performance is increased by improving algorithm. Also, to keep properties and data distribution instead of using clustering algorithm k-means to find cluster centers, we used hierarchical algorithm and similarity percent in order to free from initializing k parameter (cluster number). In 'implementing' section, we compared proposed method with normal Svm and multi-classes algorithm and results are provided in relevant Tables. Also, by running hypothesis test, we proved integrity of proposed algorithm in compare with 4 other algorithms and finally we present results. In future works, we can study on more precise estimation of more efficient edges.

REFERENCES

[1] Chih-Wei Hsu, and Chih-Jen Lin,"A Comparison of methods for multiclass support vector machines", IEEE Transactions on neural netwrks, 2002, Vol. 13, No. 2.
[2] A. Smola, and B. Scholkopf, "Sparse greedy matrix approximation for machine learning", In Proceedings of the Seventeenth International Conference onMachine Learning, 2000, Stanford, USA, 911-918.
[3] S. Vishwanathan, and M. N. Murty, "SSVM: a simple SVM algorithm". Proceedings of the 2002 International Joint Conference, 2002, Vol.3, 2393-2398.
[4] I. W. Tsang, and J. T. Kwok, and P-M. Cheung, "Core Vector Machines: Fast SVM Training on Very Large Data Sets". In Journal of Machine Learning Research, 2005, 6:363-392.
[5] C. Yang, and R. Duraiswami, and L. Davis, "Efficient kernel machines using the improved fast Gauss transform". In Advances in Neural Information Processing Systems, 2005, Vol.17.
[6] D. Pavlov, and 1. Mao, and B. Dom, "Scaling-up support vector machines using boosting algorithm". In Proceedings of the International Conference on Pattern Recognition, 2000, vol.2, 2219-2222.
[7] R. Collobert, and S. Bengio, and Y. Bengio, "A parallel mixture of SVMs for very large scale problems", In Neural Computation, 2002.
[8] B. LI, and Q.WANG, and J. HU,"A Fast SVM Training Method for Very Large Datasets", Proceedings of International Joint Conference on Neural Networks, 2009, Atlanta, Georgia, USA.
[9] R. Koggalage, and S. Halgamuge, "Reducing the Number of Training Samples for Fast Support Vector Machine Classification", In Neural Information Processing, 2004, Vol.2, No.3, 57-65.
[10] D. Ziou, and S. Tabbone, "Edge detection techniques - an overview". In International Journal of Pattern Recognition and Image Analysis, 1997.
[11] S. Wei, "Building Boundary Extraction Based On LIDAR Point Clouds Data", Ocean College of Shanghai Fisheries University, Shanghai2009.
[12] E. Reddy, and J. Bellary, "Multi-Class Support Vector Machines – A Comparative Approach" , International Journal of Applied Physics and Mathematics, Vol. 2,No.4, 2012.
[13] G. Chen, J. Xu, X. Xiang," Neighborhood Preprocessing SVM for Large-scale Data Sets Classification", Fifth International Conference on Fuzzy Systems and Knowledge Discovery.
[14] M. Lozano, J. S´anchez, F. Pla, "Reducing Training Sets by NCN-based Exploratory Procedures",Dept. Lenguajes y Sistemas Inform´aticos, Universitat Jaume I Campus Riu Sec, 12071 Castell´on, Spain.
[15] J. GOU, L. DU, T. XIONG, "Weighted K-nearest Centroid Neighbor Classi_cation", Journal of Computational Information Systems 8: 2 (2012) 851–860.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.