An SVM-cum-Decision Tree Approach to binary classification
(IEE-04: Course Project)

Aashita Kesarwani
kesar01@gmail.com
Roll No: 072070

Kalpna Gupta
kalpana.iitr@gmail.com.
Roll No:070810

Abstract

We have employed a hybrid SVM based decision tree approach to speedup SVMs in its testing phase for binary classification tasks. In this approach, focus is on reducing the number of test data points that are classified with SVMs, hence reducing the total time consumed in testing. The tree first classifies the data points into data pints that are far-off or close to the decision boundary of the SVM. Far-off nodes are classified directly with decision tree only while closer data points need SVM's help for high accuracy. Thus it has both univariate and multivariate (SVM) nodes. The hybrid tree takes SVM's help only in classifying crucial data points lying near decision boundary; remaining less crucial data points are classified by fast univariate nodes without any compromise in classification accuracy.

1. Main Objectives

- Approximate the decision boundary for SVM training data to DT.
- Define a closeness measure and threshold parameter.
- Test the hybrid model and compare the results for different threshold values.

2. Status and other details

- Fully Completed
- Contribution of members:
  - Kalpna – report making (40%).
  - Aashita – coding (60%)
- Total time spent on the project: 2 weeks

3. Major stumbling blocks

- *Construction of Decision Tree*: We initially could not figure out how to build up the optimal decision tree which uses SVMs for the nodes. Later we overcame this problem by using SVM and DT separately for different data points by means of a closeness measure.
- *Deciding upon the Data set*: The data set that we wanted to use should have relevant attributes so that decision tree looks intuitive and interpretable. Also, number of attributes was a major concern due to slow speed of only available resource MATLAB as compared to CPLEX [2]. With the new design [1] it was no more a concern.

4. Introduction

SVMs being computationally powerful tools for supervised learning are widely used in classification and regression problems. The central idea of SVM classifier is to find the optimal separating hyper plane between positive and negative examples. The optimal hyper plane is defined as the one giving maximum margin between training examples that are closest to the hyper plane. SVM classifiers have been successfully applied to a variety of real-world problems like handwritten digits classification, particle identification, face recognition, text categorization and bioinformatics [4]. SVMs enjoy better generalization than many other classification techniques in these applications, but this improved generalization comes with a cost. SVMs are considerably slower in testing phase than other techniques. This is because the computational complexity of SVM's decision function scales with respect to the number of support vectors. Hence if the number of support vectors is very large, SVMs will take more time to classify a new data point. It could be seen that almost all methods aim at reducing the number of

support vectors to decrease testing time. We have implemented the idea proposed by Arun Kumar and Gopal [1] to approximate the decision boundary of SVM using decision tree (DT) to speedup SVM in its testing phase. In this approach (SVMDT) a single binary SVM is trained once and is positioned in some of the (multiple) leafs of the DT. SVMDT approach is different from other methods such as [3] in the sense that, instead of attempting to simplify SVM decisions with DT to an extent or to reduce the number of support vectors, focus is on reducing the number of test data points that require SVM's decision. It uses both SVM and DT to achieve fast classification without any compromise in classification accuracy. Only crucial test data points which lie close to SVM's decision boundary are classified using SVM nodes; remaining less crucial test data points are classified using much faster univariate nodes of Decision Tree. This was a fairly simple and effective approach as compared to other ones we considered for building our Decision Tree [2][3]. In this way, only a very small fraction of data points needs classification by SVM and only these data points take long time while most of the intuitively classifiable data points are classified with DT taking very less time reducing the overall time taken significantly.

The SVMDT approach benefits from both SVM and DT by capturing their advantages of accuracy and speed respectively. SVMs are well known for their high accuracy and good generalization of classification problems. While DTs are good at quickly producing a sufficiently accurate but less generalized classification. We use DT where accuracy is not a concern by data characteristics and SVM for data which inherently needs more accuracy. This reduces the time consumed in testing phase as well as in classifying new data points.

The paper is organized as follows: the idea of hybrid Decision Tree is presented in Section 5. Experimental results are presented and discussed in Section 6. Section 7 and Section 8 give concluding remarks and future scope of SVMDT respectively.

5. SVM based Decision Tree

The time complexity of SVMs can be stated as:

$$\text{Complexity}_{SVM} = \text{d\_num} \times O(nN_{SV})$$

Where,
d_num = total number of data points
n = dimensionality of the input
N_{SV} = number of support vector hyper planes

Most methods aim at reducing number of SVs which can at most be reduced up to a limit that too on the accounts of loosing accuracy. Dimensionality reduction techniques are well formulated statistically. SVMDT focuses on reducing the number of data points being classified with SVM, as said above, and this can achieve a high speed up without loosing the accuracy at all. It exploits the fact that only the data points near to decision boundary need accuracy as high as provided by SVMs, not all of them.

5.1 Motivation

Let's take an example of cancer patients. Cancer has some symptoms analogous to the attributes in the data set, which help in decision making. If we plot these patients as class vs. symptoms similar to plotting the sample data, we can draw a boundary between the points belonging to class of cancer patients and to class of remaining patients. Now the points far away from the decision boundary will have negative results for all the symptoms and will be easily classifiable as negative class. Any sufficiently knowledgeable doctor can draw the conclusion. Similarly for patients heavily showing the symptoms of cancer will be identifiable easily. Complicacy arises in case of points near to decision boundary where slight error may lead to wrong classification. Hence, an expert's advice is required and the patient is referred to specialized doctor, who by properly examining the patient

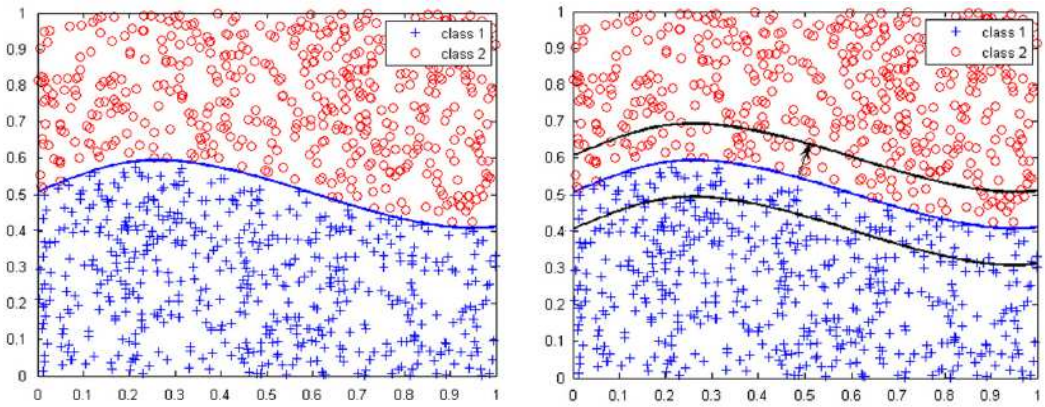

Fig. 1. (a) SVM's decision boundary and (b) δ-region

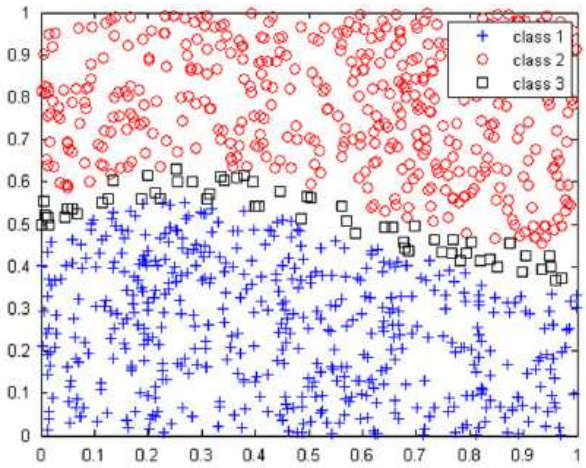

Fig. 2. Three classes resulting from DT

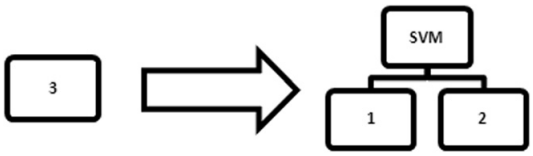

Fig. 3. Class 3 node replaced with SVM. Which in turn returns class 1 and class 2.

announces results. It definitely takes time but the accuracy is high. Overall, we optimized on both time, by taking services of a god doctor and accuracy, by consulting specialist, depending upon the data point characteristics.

Thus the doctor classifies into cancer, no cancer and crucial classes and specialist further classifies crucial into cancer and non cancer classes.

5.2 SVMDT

Let us consider an SVM decision function

$$f(x) = \sum_{i = 1 \text{ to } Nsv} \alpha_i D_i K(x, X_i) + b$$

Where,

$X_i \in R^n$ is a support vector,

D_i is corresponding target value (D_i = 1for class 1 data points and D_i = -1 for class 2 data points),

α_i is Lagrangian multiplier ($0 < \alpha_i < C$),

$b \in R$ is bias,

Nsv represents the number of support vectors,

K represents the kernel function and

$x \in R^n$ is the new data point to be tested

Given SVM's decision function $f(x)$, its decision boundary is defined as $f(x)=0$. This decision boundary dichotomizes the feature space into two mutually exclusive regions, positive region $f(x)>0$ (consisting of all training data points predicted as class 1) and negative region $f(x)<0$ (consisting of all training data points predicted as class 2). Based on this decision boundary, we define a closeness measure S(x) such that, it will give small values for data points closer to $f(x)=0$ and large values for data points away from $f(x)=0$. By selecting a threshold parameter δ, we define a δ-region around the decision boundary, such that this δ-region will include all training data points with closeness measure $S(x) \leq \delta$. this δ-region is shown in Fig. 1. We re-label predictions for all training data points inside δ -region as class 3. Now the feature space will have training data points along with its predictions, representing three different regions (as shown in Fig.2):

(i) Class 1- $S(x) > \delta$ and $f(x) > 0$

(ii) Class 2- $S(x) > \delta$ and $f(x) < 0$

(iii) Class 3- $S(x) \leq \delta$ (δ-region)

Once training data points with its predictions are available to represent these three regions, SVMDT can be obtained in two steps. First we train a DT with this 3-class data set to approximately identify these three regions. After training DT, the second step is to replace each class 3 leaf by a sub tree with binary SVM and two leaves as shown in Fig. 3. This gives final SVMDT where a single binary SVM trained once is positioned in several leaves of the DT. SVMDT consists of both conventional univariate decision nodes and multivariate decision nodes (SVMs). The univariate decision nodes help in arriving at a quick decision for less crucial test data points without consulting the multivariate SVM. On the other hand if test data points are crucial, univariate decision nodes direct them to multivariate SVM. Hence only a small portion of test data points gets classified with SVM nodes and the rest are classified using much faster univariate decision nodes, thereby decreasing the overall testing time.

5.3 Closeness measure and threshold δ

For identifying the δ-region, a closeness measure between training data points and decision boundary $f(x) = 0$ is to be defined. We use the proposed probabilistic output of SVM method [1] as closeness measure. In general, the SVM decision function $f(x)$ outputs uncalibrated values and can be converted to posterior probability estimates by fitting a sigmoid function at its output.

$$P(\text{Class } 1|f(x)) = 1/(1+\exp(-f(x))$$

$$P(\text{Class } 1|f(x)) = 0.5 \text{ when } f(x) = 0$$

The above expression can be modified as

$$\Delta P(x) = P(\text{Class } 1|(f(x)) - 0.5 = 1/(1+\exp(-f(x)) - 0.5$$

where $\Delta P(x)$ indicates the closeness measure S(x) between training data points and decision boundary f(x)=0. It could be noted that, for f(x)=0; S(x)=0, when $f(x) \rightarrow \infty$; $S(x) \rightarrow 0.5$, and $f(x) \rightarrow -\infty$; $S(x) \rightarrow -0.5$. Thus we have a reasonable threshold parameter δ.

5.4 SVMDT algorithm

The algorithm as described by Arun Kumar [1] is stated below. The role of data matrices used is given first as their names suggest:

Train_data: set of training data points
Train_target: corresponding target for Train_data
New_target: targets to be used for DT training Algorithm

SVMDT training:

Given a binary data set, SVMDT training can be performed with the following steps:

(a)Train SVM classifier with Train_data and train target to obtain its decision function f(x).

(b) Classify Train_data with f(x) in to class 1 or class 2.Save these predictions in New_target.

(c) Select a threshold value δ between 0 and 0.5.

(d) Identify data points in Train_data with $S(x) \leq \delta$ and change their corresponding predictions in New_target to class 3.

(e)Train a DT with Train_data and New_target.

(f) Replace all class 3 leaves of DT by a sub tree with SVM and two leaves as shown in Fig. 3.

(g) Save the tree and return.

5.5 Data set

Experimental data was for the detection of Breast Cancer (Wisconsin) [5] with 10 features and 683 samples (283 for training and 400 for testing).

6. Experimental Results

We performed the above steps for more than one data set and achieved faster execution with same accuracy as SVMs.

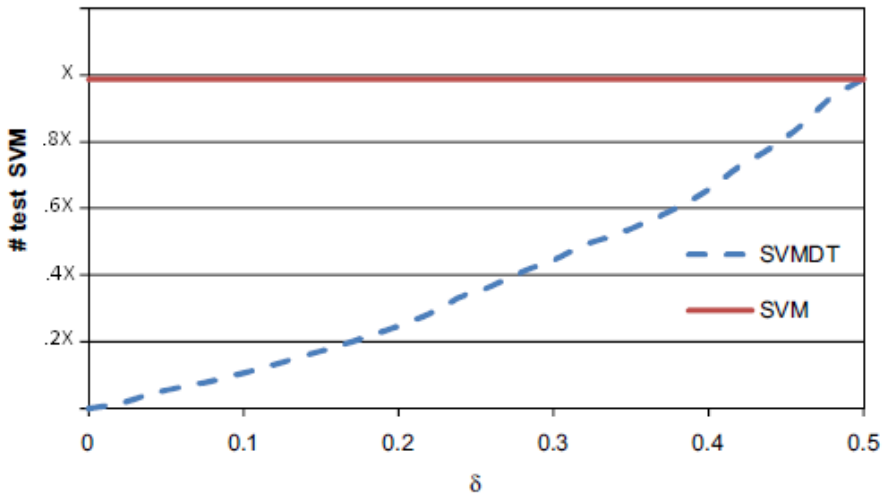

Fig. 4. Ideal results for number of SVM tests vs. total Threshold

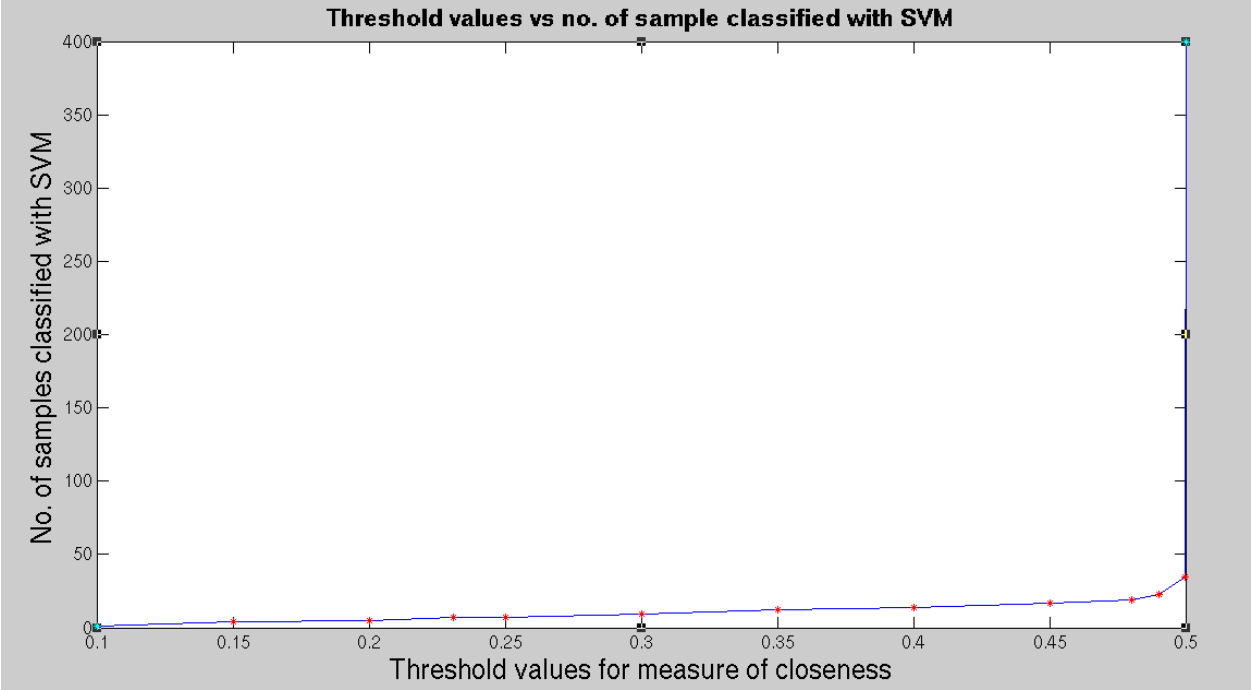

Fig. 4. Experimental results for number of SVM tests vs. total Threshold

Relation of SVM tests required to threshold values is depicted in Table 1.
Plot for the same is depicted in Fig 5. This is very close to the ideal results shown in Fig 4. Number of SVM tests is more or less directly proportional to time taken for testing; therefore the graph for time taken vs. delta values is similar.

Threshold values for measure of closeness δ	**Out of 400, no. of samples classified using SVM**
0.1	1
0.15	4
0.2	5
0.231*	7
0.25	7
0.3	9
0.35	12
0.4	14
0.45	17
0.48	19
0.49	23
0.5	400

Table 1. SVM tests required for δ values

*For support vectors, |f(x)| = 1, and S(x) = 0.231. Hence selecting δ=0.231 will make the δ-region to coincide with the margin area around SVM's decision boundary.

Moreover, we observed that at δ=0.231, the δ-region covers the region covered by hyper planes $f(x)$ = -1 to $f(x)$ = 1 as stated in [1].

7. Conclusions

As shown above, the desired results were achieved fully. With the SVMDT approach, we could achieve the fast classification without any loss of accuracy. Data points far away from the decision boundary were classified with DT and those near the decision boundary were classified with SVM.

The function that we built takes sample and training data sets as arguments; hence it works for all the domains. We performed our experiment with data for Breast Cancer (Wisconsin) [5] with satisfactory results in each case.

In this experiment, the threshold parameter is internal to the SVMDT model and is not given as user input, while classifying new data points. Rather, it is to be tuned at the time of testing of the model. If the user has advanced knowledge of domain then tuning of threshold parameter can be left to him leading to accuracy in different data sets and thus good generalization over range of domains.

Moreover, SVMDT can be seen as a generalized model for both SVM and DT. If the threshold parameter is kept 0, no δ-region is identified and the model is a pure DT. On the other hand, if δ = 0.5, DT classifies all the points into δ-region leading to SVM classification. Thus the model becomes an SVM.

8. Future Scope

The work can be extended to multiclass problems by defining more than one δ-region leading to separate binary SVMs, or one δ-region leading to multiclass SVM. Analytical and comparative study for both these approaches can be done so as to draw a conclusion on their accuracy and time complexity.

References

1. M. Arun Kumar and M. Gopal (2010) 'A hybrid SVM based decision tree', *Pattern Recognition*, vol. 43, no. 12, pp. 3977-3987, September 1992.
2. Bennett, Kristin P. and Auslender, Leonardo (1998) 'On Support Vector Decision Trees for Database Marketing', *R.P.I. Math Report 98-100*.
3. Bennett, K. P. and Blue, J. A. (1998) 'A Support Vector Machine Approach to Decision Trees', *Neural Networks Proceedings, 1998 of IEEE World Congress on Computational Intelligence*, vol. 3, pp. 2396-2401.
4. C.J. Burges (1998) 'A tutorial on support vector machines for pattern recognition', *Data Mining and Knowledge Discovery*, vol. 2, pp. 1–43.
5. http://archive.ics.uci.edu/ml/datasets html?format=&task=cla&att=&area &numAtt=&numIns=&type=&sort= taskUp&view=table
6. http://www.csie.ntu.edu.tw/~cjlin/l ibsvmtools/datasets/