Rule-Based Support Vector Machine Classifiers Applied to Tornado Prediction

Theodore B. Trafalis[1], Budi Santosa[1], and Michael B. Richman[2]

[1] School of Industrial Engineering, The University of Oklahoma
202 W. Boyd, CEC 124, Norman, OK 73019
{ttrafalis,bsant}@ou.edu
[2] School of Meteorology, The University of Oklahoma
100 E. Boyd, SEC 1310, Norman, OK 73019
mrichman@ou.edu

Abstract. A rule-based Support Vector Machine (SVM) classifier is applied to tornado prediction. Twenty rules based on the National Severe Storms Laboratory's mesoscale detection algorithm are used along with SVM to develop a hybrid forecast system for the discrimination of tornadic from non-tornadic events. The use of the Weather Surveillance Radar 1998 Doppler data, with continuous data streaming in every six minutes, presents a source for a dynamic data driven application system. Scientific inquiries based on these data are useful for dynamic data driven application systems (DDDAS). Sensitivity analysis is performed by changing the threshold values of the rules. Numerical results show that the optimal hybrid model outperforms the direct application of SVM by 12.7 percent.

1 Introduction

Rule-based classification methods have shown promise in physical systems applications [1]. One builds a rule-based model by incorporating prior information. In the case of Support Vector Machines (SVMs), prior knowledge is incorporated into the model as additional constraints in the form of polyhedral rule sets in the input space of the given data. These rule sets are supposed to belong to one of two categories into which all the data are divided [2, 3].

Tornado forecasting is an active area of research in the meteorological community [4, 5]. State-of-the-science weather radar scans volumes of the atmosphere, producing a large amount of information that is updated every 5 to 6 minutes. Scientific inquiries based on these data are useful for Dynamic Data Driven Applications Systems (DDDAS). Once the data are collected, they are quickly processed by algorithms that look for signatures of tornadoes in near-real time, since an extra minute of lead-time can save lives. The dynamic nature of DDDAS problems requires us to address the time dependency or real time nature of the applications. Certain applications (e.g., tornado formation) require real time response to observations from data. Typically, in the prediction of severe weather potential, data from observations taken hours previous to the formation are used and these are not updated with real data as they become available. Incorporating new dynamically injected data is a fundamental

M. Bubak et al. (Eds.): ICCS 2004, LNCS 3038, pp. 678–684, 2004.

change in the design. The use of the Weather Surveillance Radar 1998 Doppler (WSR-88D) data, with continuous data streaming in every six minutes, presents a source for data driven simulations. One of the severe weather detection algorithms, created by the National Severe Storms Laboratory (NSSL) and in use at the WSR-88D, is the Mesocyclone Detection Algorithm (MDA) [4]. This dynamic algorithm uses the data stream outputs of the WSR-88D and is designed to detect storm–circulations associated with regions of rotation in thunderstorms. The MDA is used by meteorologists as one input in their decision to issue tornado warnings. Recent work by Trafalis et al. [4, 5] has shown that SVMs applied to the MDA offer a promising role in improved tornado classification. We present a novel approach by incorporating rules into SVMs of the MDA attributes as they stream in just prior to tornado formation. These rule based sets classify the data into one of three categories, tornado, non-tornado and unclassified. Thus, the rules partition the input space into regions for which we know, with a high degree of certainty, the label of points located in those regions. Our approach is different from [3] in the sense that the rules are combined with SVM in a sequential approach. This paper is organized as follows. In section 2, the data description is given. In section 3, we provide a description of rule-based SVM classifiers. Section 4 describes the experimentation procedure. Section 5 provides computational results and, in section 6, analysis and conclusions are provided.

2 Data

The MDA data set used for this research is based on the outputs from the WSR-88D radar that is collected just prior to the formation of a pre-tornadic circulation. Any circulation detected on a particular volume scan of the radar can be associated with a report of a tornado. In the severe weather database, supplied by NSSL, there is a label for tornado ground truth that is based on temporal and spatial proximity. If there is a tornado reported between the beginning and ending of the volume scan, and the report is within a reasonable distance of a circulation detection, then the ground truth value is flagged. If a circulation detection falls within the prediction "time window" of -20 to +6 minutes of the ground truth report duration, then the ground truth value is also flagged. The key idea behind these timings is to determine whether a circulation will produce a tornado within the next 20 minutes, a suitable lead time for advanced severe weather warnings by the National Weather Service. Owing to the autocorrelation in the MDA attributes, a sampling strategy is used to minimize serial correlation. These sampled data are divided into independent training and testing sets, with 749 and 618 observations, respectively.

3 Rule-Based Support Vector Machine Classifiers

3.1 Rule Generation

In this work, we consider a rule-based approach of a decision tree type as shown in Fig. 1. Nodes in the decision tree involve testing a particular attribute. The test at a

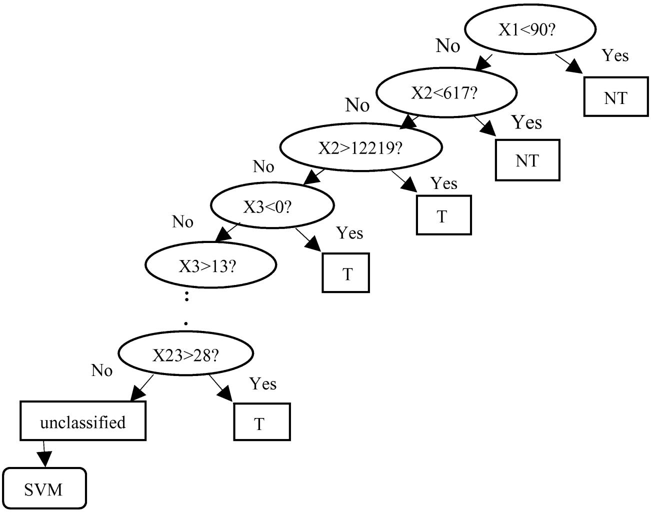

Fig. 1. Tree diagram of rule generation and decisions. Ovals represent nodes, squares represents leaves

node compares an attribute value with a constant threshold. Leaf nodes give a classification that applies to all instances that reach the leaf. When a leaf is reached, the instance is classified according to the class assigned to the leaf. Note that the output of the last node referring to the unclassified category becomes an input to the SVM that provides the final label to the unclassified cases.

There were 23 MDA attributes available for discriminating tornadoes from non-tornadoes [4]. For each attribute, we considered the corresponding probability distribution function for tornado and non-tornado cases arising from the training data. The selection of the threshold for each rule was based on eliminating misclassification by investigating if the minimum for a non-tornado case had a value less than the minimum for a tornado case for a specific attribute. If such a condition holds, then a region unique to non-tornadoes is found.

Similarly, if the maximum for a non-tornado case had a value less than the maximum for a tornado case, for a specific attribute, a region unique to tornado cases is found. Of the 23 attributes, only 20 were found to be useful for rule generation. The thresholds used for tornado and non-tornado discrimination are shown in Table 1.

3.2 Support Vector Machines (SVMs)

Given a set of data points $\{(x_i, y_i),\ i = 1,\dots,\ell\}$ with $x_i \in \Re^n$ and $y_i = \pm 1$, the SVM finds a classifier that separates the two classes of points with maximum margin separation (Fig. 2). The SVM formulation can be written as follows [6],

$$\min_{w,b,\eta} C\sum_{i=1}^{\ell} \eta_i + \tfrac{1}{2}\| w \|^2 \tag{1}$$

$$st \qquad y_i(wx+b)+\eta_i \geq 1 \qquad\qquad \eta_i \geq 0 \quad i = 1,\dots \ell$$

Table 1. Threshold values for each MDA attribute. See [4] for description of attributes

Non-tornado thresholds	Tornado thresholds
if x1 < 90, then non-tornado	
if x2 < 617, then non-tornado	if x2 > 12219, then tornado
if x3 < 0, then non-tornado	if x3 > 13, then tornado
if x4 < 813, then non-tornado	
if x5 < 1091, then non-tornado	
if x6 < 124, then non-tornado	
if x7 < 6, then non-tornado	
if x8 < 10, then non-tornado	
if x9 < 122, then non-tornado	
if x10 <2, then non-tornado	if x10 > 77, then tornado
	if x11 > 83, then tornado
if x12 < 106, the non-tornado	
if x13 < 3, then non-tornado	
if x14 < 11, then non-tornado	
if x15 < 122, then non-tornado	
if x16 < 106, then non-tornado	
if x17 < 617, then non-tornado	
	if x18 > 113, then tornado
	if x22 > 26, then tornado
	if x23 > 28, then tornado

where C is a parameter to be chosen by the user that controls misclassifications, w is referring to the vector perpendicular to the separating hyperplane, η_i refers to the misclassification error variables and b is the bias of the separating hyperplane. A larger C corresponds to assigning a larger penalty to errors. Introducing positive Lagrange multipliers α_i, to the inequality constraints in model (1) we obtain the following dual formulation:

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^{\ell} \sum_{j=1}^{\ell} y_i y_j \alpha_i \alpha_j x_i x_j - \sum_{i=1}^{\ell} \alpha_i$$

$$st \quad \sum_{i=1}^{\ell} y_i \alpha_i = 0 \, , \tag{2}$$

$$0 \leq \alpha_i \leq C \qquad i = 1, \ldots \ell$$

The solution of the primal problem is then given by $w = \Sigma_i \alpha_i y_i x_i$ where w is the vector that is perpendicular to the separating hyperplane. The free coefficient b can be found from the relation $\alpha_i (y_i (w\, x_i + b) - 1) = 0$, for any i such that α_i is not zero. The use of a kernel function allows the SVM to operate efficiently in nonlinear high-dimensional feature space [7].

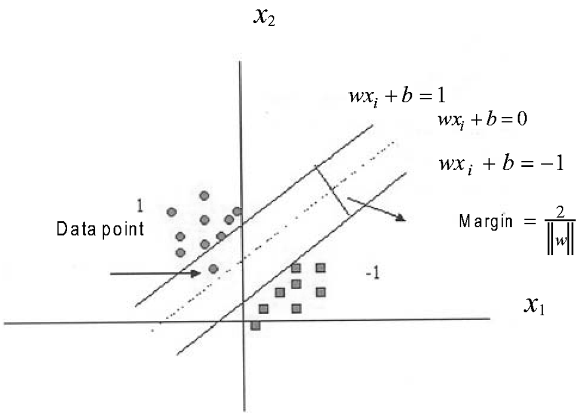

Fig. 2. The geometric illustration of SVM

4 Experiments

In our experimentation, the data are split into training and testing sets. The testing set is sampled independently five times. The first set of experiments is performed by using SVM only on the five testing samples. The total misclassification error is computed as the average of the misclassification error of each sample. The second set of experiments is performed by extracting the rules from the training data and applying those rules in the testing phase. Based on the rules, each testing sample is divided into three different sets: non-tornado, unclassified, and tornado. In the testing phase, those observations not classified by the rules are used as inputs to SVM. The SVM is trained on the training set then tested on five different unclassified samples. For each testing set, the misclassification error for the non-tornado rules set, SVM and tornado rules set are computed. The OSU SVM Classifier Matlab Toolbox [8] was used to run experiments of SVM.

5 Computational Results

The results of the experiments are presented in Fig. 3 and Table 2. The values in the table are misclassification error rates for non-tornado and tornado and SVM components of the total hybrid system. After initial experimentation, it was noted that the rules components of the system had a lower error rate than the SVM component of the system. Accordingly, altering the rules to admit additional cases was considered by creating a multiplier for the threshold values in Table 1. This multiplier controls the level of threshold values (e.g., in Table 1, for attribute 1, the original threshold, 90, corresponds to multiplier 1 and 1.05 times 90 equals 94.5 and this value admits additional observations into the non-tornado category). Table 2 and Fig. 3 illustrate the sensitivity of misclassification error with respect to the threshold. Table 3 shows the misclassification error for SVM for each testing sample and the average of the five samples.

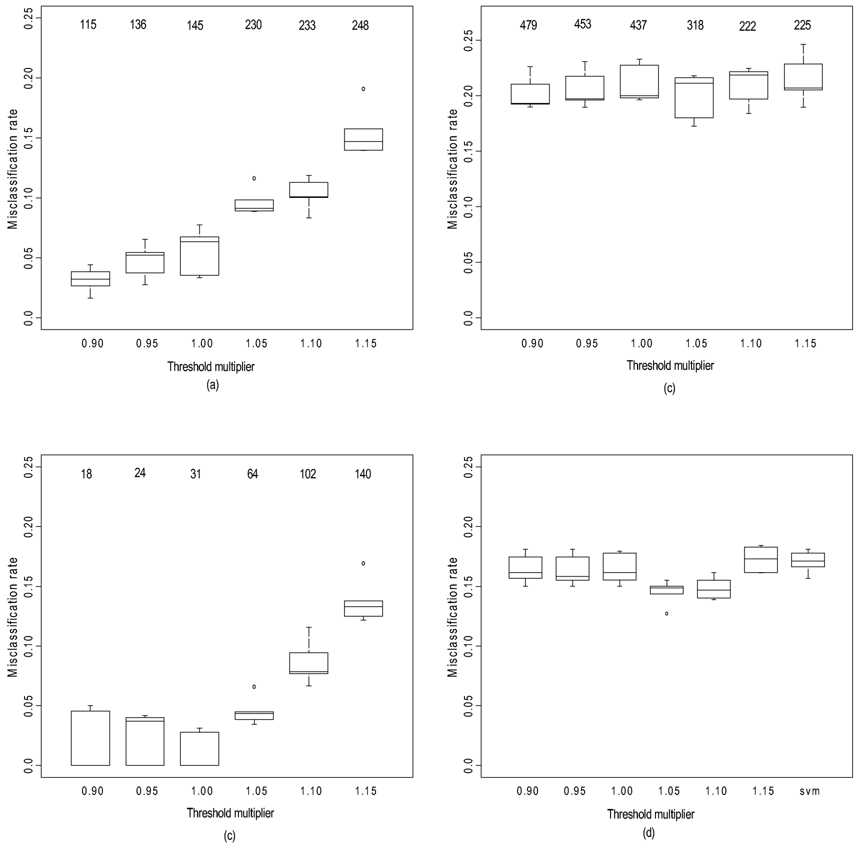

Fig. 3. Boxplots of misclassification error due to (a) non-tornado rules set, (b) SVM, (c) tornado rules set and (d) total hybrid system. Threshold multipliers are shown on X-axis and the numbers of cases classified are shown above the boxplots in (a), (b) and (c)

Table 2. Misclassification error of the hybrid system components and total system

Multiplier	0.90	0.95	1.00	1.05	1.10	1.15
Non-tornado rules	0.0191	0.0237	0.0174	0.0454	0.086	0.1373
Tornado rules	0.0316	0.0474	0.0550	0.0968	0.1032	0.1550
SVM	0.2024	0.2063	0.2110	0.1997	0.2093	0.2154
Total system	0.1648	0.1638	0.1648	0.1449	0.1485	0.1725

Table 3. Misclassification error for SVM

	Sample 1	Sample 2	Sample 3	Sample 4	Sample 5	Average
SVM	0.1664	0.1778	0.1713	0.1811	0.1566	0.1706

6 Analysis and Conclusions

Tables 2 and 3, show that the best misclassification error for the hybrid model (0.1449) is 12.7% lower than the one for the model based solely on SVM (0.1706). The reason for the total system improvement can be seen in Figure 3a,c and Table 2, where the non-tornado rules, based on the threshold given in Table 1, have a mean error rate of 0.0174. Similarly, the tornado rules have a mean error rate of 0.055 at the same multiplier. In contrast, the SVM component has an error rate of 0.211. The behavior of the rules, as seen in Fig. 3 a, c is interesting as the misclassification rate is remarkably low (approximately 5 percent) for threshold multipliers of 0.90 to 1.00. The trade-off is that fewer observations are classified as tornadoes or non-tornadoes. As the threshold multipliers increase to 1.05 and beyond, the misclassification error increases considerably to approximately 15 percent indicating a poorer discrimination between tornadoes, and non-tornadoes. In contrast, the SVM, based on unclassified data (Fig. 3c), is insensitive to the threshold multiplier.

Therefore, given the lower rates in the non-tornado and tornado rules, it is logical to create a hybrid system to capitalize on the disparity in error rate. By admitting additional observations into the leaves of the decision tree prior to sending the remaining observations to the SVM, the optimal system is found. This occurs at a multiplier of 1.05 times the threshold values in Table 1. Experiments are planned for injection of information from successive volume scans to assess additional predictive capability in a constantly updating form of DDDAS.

Acknowledgements. This work has been supported by NSF grant EIA-0205628.

References

1. Mitchell, T.M., Machine Learning, McGraw-Hill, New York, 1997
2. Fung, G.M., Mangasarian, O.L., Shavlik, J.W.: Knowledge-based Support Vector Machines Classifiers, Data Mining Institute. Technical Report 01-09. Computer Sciences Department, University of Wisconsin (2001)
3. Fung, G.M., Mangasarian, O.L., Shavlik, J.W.: Knowledge-based Nonlinear Kernel Classifiers. Data Mining Institute Technical Report 03-02. Computer Sciences Department, University of Wisconsin (2003)
4. Trafalis, T.B., Santosa B., Richman, M.B.: Tornado Detection with Kernel-based Methods. In: Dagli, C.H., Buczak, A.L., Ghosh, J., Embrechts, M., Ersoy, O. (eds.): Intelligent Engineering Systems Through Artificial Neural Networks. ASME Press, Vol. 13 (2003) 677-682
5. Trafalis, T.B., Santosa B., Richman, M.B.: Tornado Detection with Kernel-Based Classifiers From WSR-D88 Radar. Submitted to: Darema, F. (ed.) Dynamic Data Driven Application Systems, Kluwer (2004)
6. Haykin, S.: Neural Networks: A Comprehensive foundation, 2nd edition, Prentice-Hall, Upper Saddle River New Jersey (1999)
7. Schölkopf, B., Smola, A.: Learning with Kernels. MIT Press, Cambridge Massachusetts (2002)
8. Junshui, M., Zhao, Y. Ahalt, S.: OSU SVM Classifier Matlab Toolbox. Available at `http://eewww.eng.ohio-state.edu/~maj/osu_SVM/`