

# TORNADO DETECTION WITH SUPPORT VECTOR MACHINES

*Theodore B. Trafalis\**, *Huseyin Ince\**, and  
*Michael B. Richman\*\**

*School of Industrial Engineering\**, *School of Meteorology\*\**  
*University of Oklahoma*  
[www.lois.ou.edu](http://www.lois.ou.edu)

*ICCS, June 2-4, Melbourne, Victoria, Australia*



# Outline

---

- Introduction
- Mesocyclone Detection Algorithm (MDA)
- Support Vector Machines (SVM)
- Experiments
- Conclusion



# Introduction

---

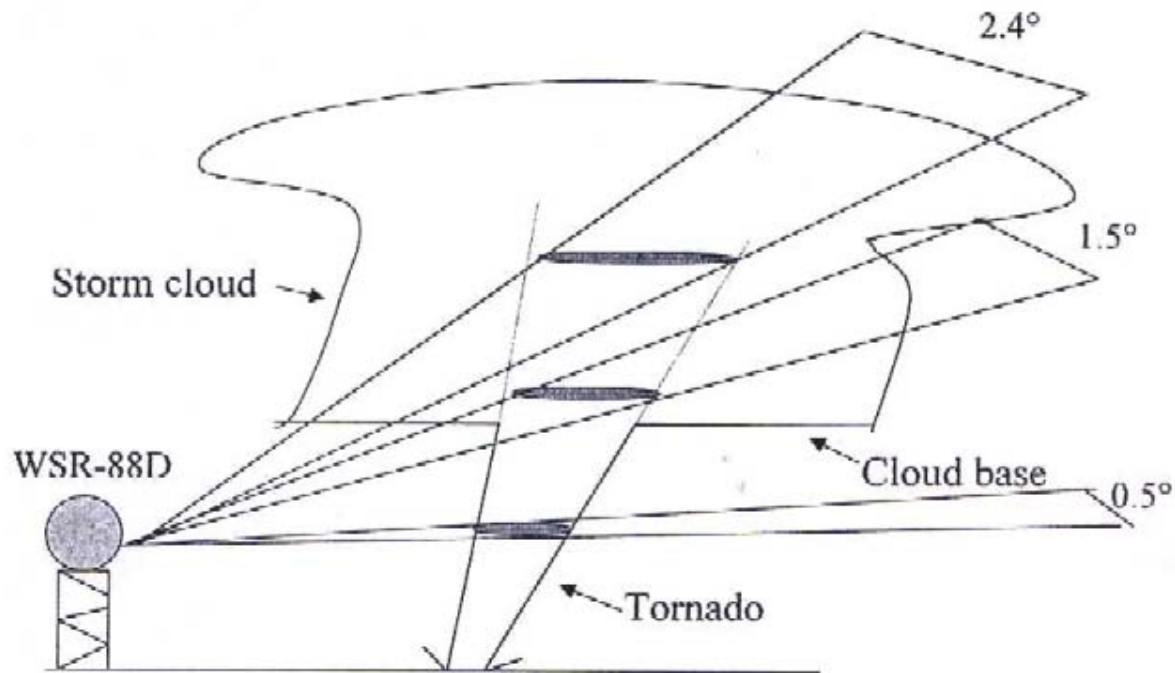
- National Weather Service (NWS) Severe Weather Detection Algorithms
  - Empirical rule based algorithms
    - MDA algorithm
  - Data mining techniques applied to MDA algorithms
    - Support vector machines
    - Neural networks

# How do we begin?

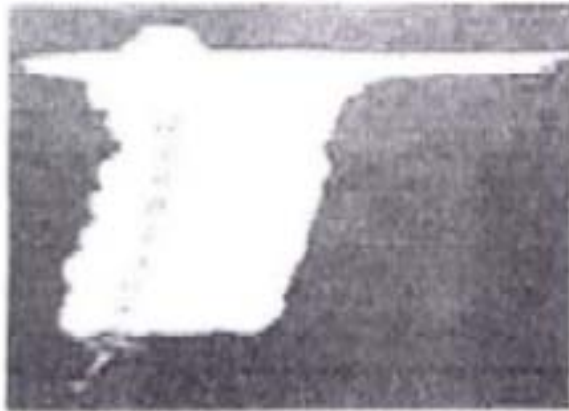
- Data!
- Radar data provide the best observational tool for gathering weather information
- Weather Surveillance Radar – 1988 Doppler (WSR-88D)



# *Radar scans the atmosphere*



# Formation of a tornado







# MDA Algorithm

---

- The current Mesocyclone Detection Algorithm (MDA) was created at the National Severe Storms Laboratory (NSSL) to work with variables derived from the WSR-88D.
- In order to detect circulations associated with vortices that eventually spin up into tornadoes, the radar velocity data are exploited.
- The data are measured for circulation depth, height above the ground and strength of the circulation.
- By relaxing previous threshold values, the MDA is capable of detecting weaker circulations that may eventually spin up into tornadoes.





# MDA continued

---

- The downside is that a number of these weak circulations do not progress into rotating thunderstorms that yielded tornadoes.
- Information we get from radar data
  - Reflectivity (Z): ratio of the radiant energy reflected by a given surface to the total incident energy
  - Velocity (V): velocity of target
  - Spectrum Width (W): measure of dispersion of velocities within the radar sample



# MDA Attributes

---

- 1. base (m) [0-12000]
- 2. depth (m) [0-13000]
- 3. strength rank [0-25]
- 4. low-level diameter (m) [0-15000]
- 5. maximum diameter (m) [0-15000]
- 6. height of maximum diameter (m) [0-12000]
- 7. low-level rotational velocity (m/s) [0-65]
- 8. maximum rotational velocity (m/s) [0-65]
- 9. height of maximum rotational velocity (m) [0-12000]
- 10. low-level shear (m/s/km) [0-175]
- 11. maximum shear (m/s/km) [0-175]
- 12. height of maximum shear (m) [0-12000]
- 13. low-level gate-to-gate velocity difference (m/s) [0-130]
- 14. maximum gate-to-gate velocity difference (m/s) [0-130]
- 15. height of maximum gate-to-gate velocity difference (m) [0-12000]
- 16. core base (m) [0-12000]
- 17. core depth (m) [0-9000]
- 18. age (min) [0-200]
- 19. strength index (MSI) wghtd by avg density of integrated layer [0-13000]
- 20. strength index (MSIr) "rank" [0-25]
- 21. relative depth (%) [0-100]
- 22. low-level convergence (m/s) [0-70]
- 23. mid-level convergence (m/s) [0-70]



# Support Vector Machines(SVM)

---

- SVM is based on statistical learning theory
- Find an optimal hyperplane that separates two classes (classification problem)
- This can be done with empirical risk minimization or structural risk minimization
  - Neural networks are based on empirical risk minimization
  - SVM are based on structural risk minimization (Vapnik 1995,98)

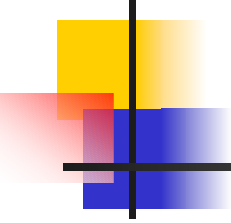
# SVM Formulation



---

Given  $\{x_i, y_i\}$  with  $y_i \in \{-1, 1\}$ , find a hyperplane that separates the two classes with a maximum margin.

- Margin is the distance between the two classes shown in the following figure.
- Distance can be computed by using different norms. Each norm provides a different optimization problem.
- The SVM learning problem is equivalent to the following optimization problem



---

$$\begin{aligned} \min \quad & \phi(w, \xi) = \|w\| + C \sum_{i=1}^l \xi_i \\ \text{Subject to} \quad & y_i (wx_i + b) \geq 1 - \xi_i \\ & \xi \geq 0 \end{aligned} \tag{1}$$

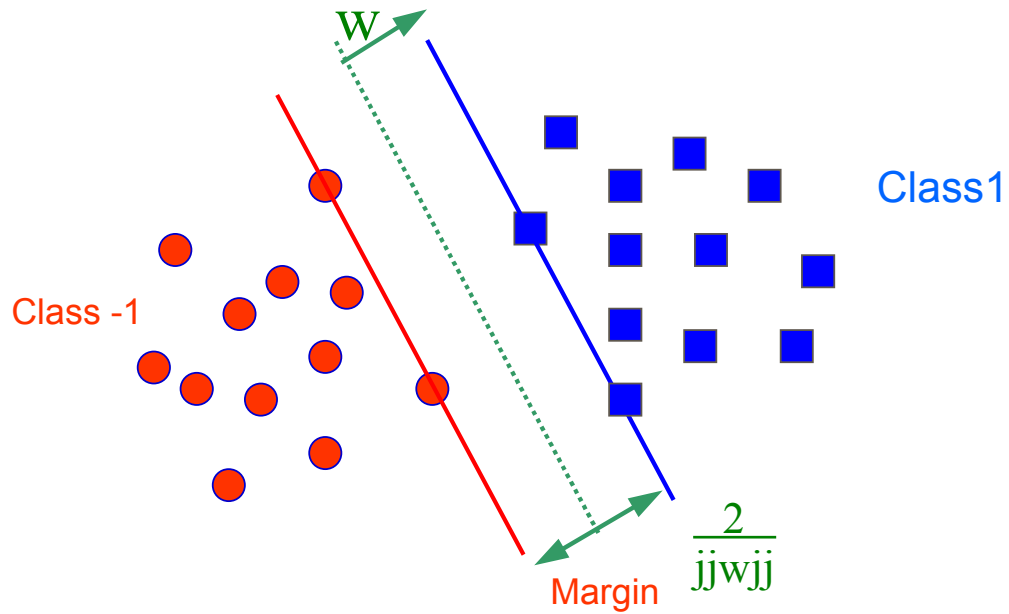


Figure 1: Maximum Margin



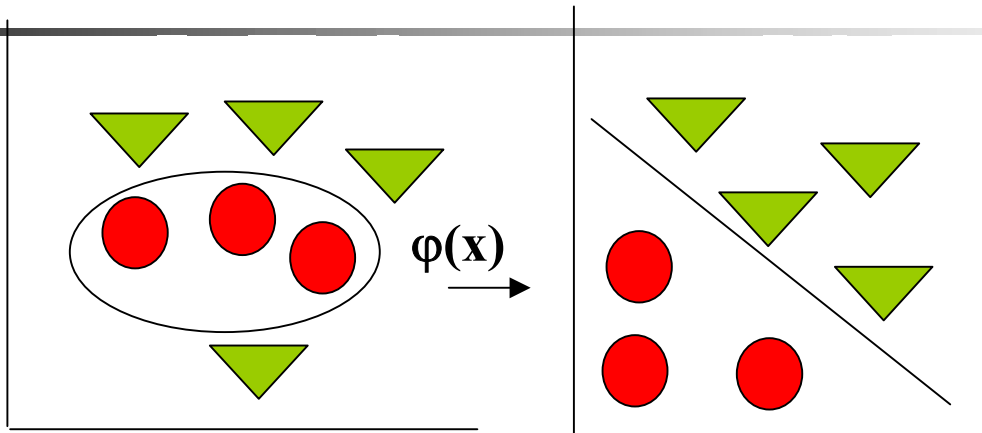
# Optimal point

---

$$w^* = \sum_{i=1}^l \lambda_i^* y_i x_i$$

- Support vector: a training vector for which

$$\lambda_i^* > 0.$$



**Non-Linear Separation**

**Linear Separation**

**Figure 2. The Kernel Method for Classification**

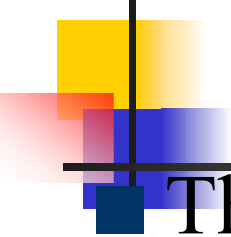


# Kernelization



---

- To handle non-linear discriminant functions .
- Linear classification methods can be applied in feature space to solve problems that are non-linear in the input space.
- This can be done by making the necessary transformation  $\varphi$  from the input space to a higher dimensional space known as the feature space.



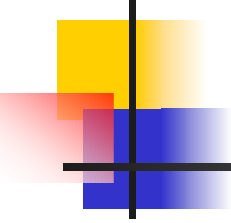
■ This mapping creates the corresponding classification problem in feature space, allowing linear separation of the problem data in feature space. Dot product in feature space is defined as

$$K(x_1, x_2) = \varphi(x_1) \cdot \varphi(x_2)$$

Most commonly used kernel functions are:

Polynomials:  $K(x_1, x_2) = ((x_1 \cdot x_2) + 1)^d$ , where  $d$  is the degree of the polynomial.

Radial Basis:  $K(x_1, x_2) = \exp(-\|x_1 - x_2\|^2 / 2\sigma^2)$ .



---

$$\max \quad F(\Lambda) = \sum_{i=1}^l \lambda_i - \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l \lambda_i \lambda_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$$

*Subject to*

$$\sum_{i=1}^l \lambda_i y_i = 0$$

$$\lambda_i \leq C \quad i = 1, 2, \dots, l$$

$$\lambda_i \geq 0 \quad i = 1, 2, \dots, l$$

$$f(x) = \text{sign} \left( \sum_{i=1}^l \lambda_i^* y_i K(x, x_i) + b^* \right)$$



# Experiments

---

- Datasets are taken from WSR-88D MDA output
- Input variables to SVM/ANN are the outputs of the MDA algorithm
- The number of input variables (attributes) is 23.
- Definition of the variables can be found in Marzban and Stumpf (1996) and the Table (just presented).
- The number of observations is more than 60,000.
- 10 different datasets are used with 2,500 observations which are chosen randomly from the whole dataset.

# Confusion matrix and forecast indices

- Confusion matrix

		<b>Observed</b>		
		Yes	No	Total
<b>Predicted</b>	Yes	Hits (a)	False alarm (b)	Forecast Yes
	No	Misses (c)	Correct negative (d)	Forecast No
Total		Observed yes	Observed No	



# (Continued)

---

- The performance of the techniques used is assessed by using the following forecast evaluation indices

- Probability of Detection (POD) =  $\frac{a}{a + c}$

- False Alarm Rate (FAR) =  $\frac{b}{a + b}$

- Critical Success Index (CSI) =  $\frac{a}{a + b + c}$

- Bias =  $\frac{a + b}{a + c}$

- Probability of false Detection (POFD) =  $\frac{b}{b + d}$

- True skill statistic (H-K Skill) =  $\frac{a}{a + c} - \frac{b}{b + d}$

- Odds Ratio (OR) =  $\frac{(a)(d)}{(b)(c)}$



# continued

---

- POD measures the fraction of observed “yes” events that were correctly forecast. Its range is 0 to 1 and a perfect score is 1 (or 100%). Note that POD is sensitive to hits, therefore, good for rare events.
- FAR measures the fraction of “yes” forecasts in which the event did not occur. Its range is 0 to 1, and 0 is a perfect rate. FAR is sensitive to false alarms and it ignores misses.
- CSI measures the fraction of observed and/or forecast events that were correctly forecast. Its range is 0 to 1, with a perfect score being 1. CSI is sensitive to hits, penalizing both misses and false alarms.



# continued

---

- Accuracy measures the fraction of all forecasts that were correct, which makes it a seemingly intuitive measure. The range is 0 to 1, with 1 being best.
- Bias measures the ratio of the frequency of forecast events to the frequency of observed events. The range is from 0 to infinity. A perfect score is 1. Bias indicates whether the forecast system has a tendency to underforecast ( $\text{bias} < 1$ ) or overforecast ( $\text{bias} > 1$ ) events.
- POFD measures the ratio of false alarms to the total number of no observations. The probability of false detection is a measure of inaccuracy with respect to the observations and provides a measure of the extent to which the forecasts provide a false warning for the occurrence of an event. POFD varies from 0 to 1. A perfect score is zero.





# continued

---

- The concept of skill is one where a forecast is superior to some known reference forecast (e.g., random chance). Skill ranges from  $-1$  (anti-skill) to  $0$  (no skill over the reference) to  $+1$  (perfect skill). Heidke's skill formulation is commonly used in meteorology since it uses all elements in the confusion matrix and works well for rare event forecasting (e.g. tornadoes) (Doswell et al., 1990).



# SVM specific parameters(Continued)

---

- For the SVM algorithm, the radial basis kernel function is used with different sigma values.
- First, we compared the SVM with different sigma values.
- Second, we compared SVM with Neural Networks.
- The following two tables show the results of the comparison.



Table 1. The Accuracy, CSI, POD, Bias, ODDS, FAR, POFD and H-K Skill for different kernel parameters.

$\sigma$	ACCURACY	CSI	POD	Bias	ODDS	FAR	POFD	H-K Skill
	(%)	(%)	(%)	(%)	(Ratio)	(%)	(%)	(%)
1.25	83.6	44.65	60.13	94.76	14.08	36.43	9.77	50.36
2.5	85.54	48.7	62.38	90.48	19.54	30.99	7.93	54.45
7.25	88.27	53.46	61.24	75.8	37.44	19.19	4.11	57.13
15	87.98	51.75	58.71	72.15	36.53	18.61	3.79	54.92

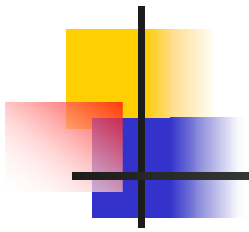


Table 2. The validation CSI for SVM, neural networks, MDA and discriminant analysis for ten different training and validation sets. The values for MDA, discriminant analysis and neural networks are taken from [5].

Seed	CSI <sub>MDA</sub>	CSI <sub>DA</sub>	CSI <sub>NN</sub>	CSI <sub>SVM</sub>
1	26.9	31	36.9	47.78
2	24	29.2	35.7	47.49
3	24.7	28.1	38.3	50.2
4	28.7	27.7	33.6	49.38
5	27.4	30	34.2	47.5
6	28	28.8	32.5	44.93
7	29.9	26.1	33.1	51.62
8	21.3	28.7	29.1	48.19
9	27.7	30.6	37.8	50.5
10	21.5	26.5	31.7	49.41
Average	26.01	28.67	34.29	48.7



# Remarks

---

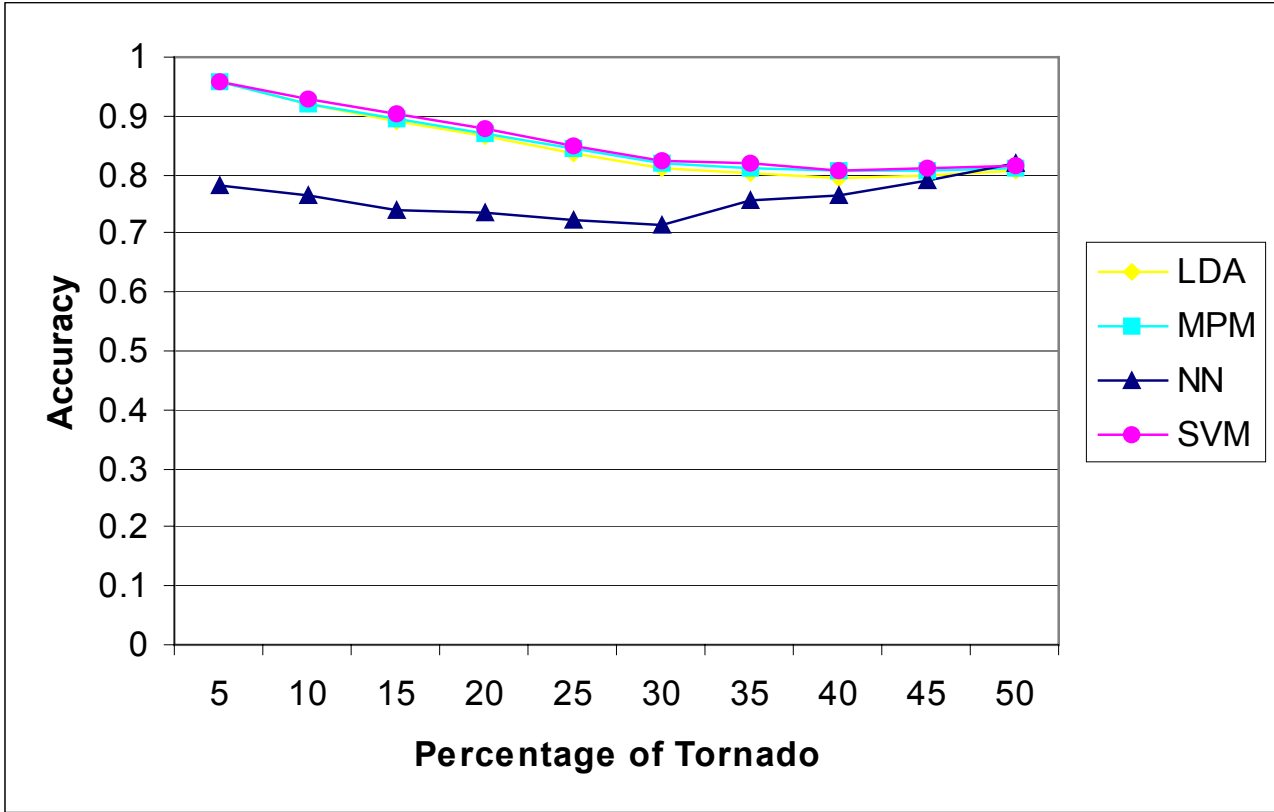
- Selection of the kernel related parameters is important and has to be chosen carefully
- Comparison of SVM with other techniques such as neural networks, discriminant analysis, mesocyclone detection algorithm shows that SVM is superior than other methods

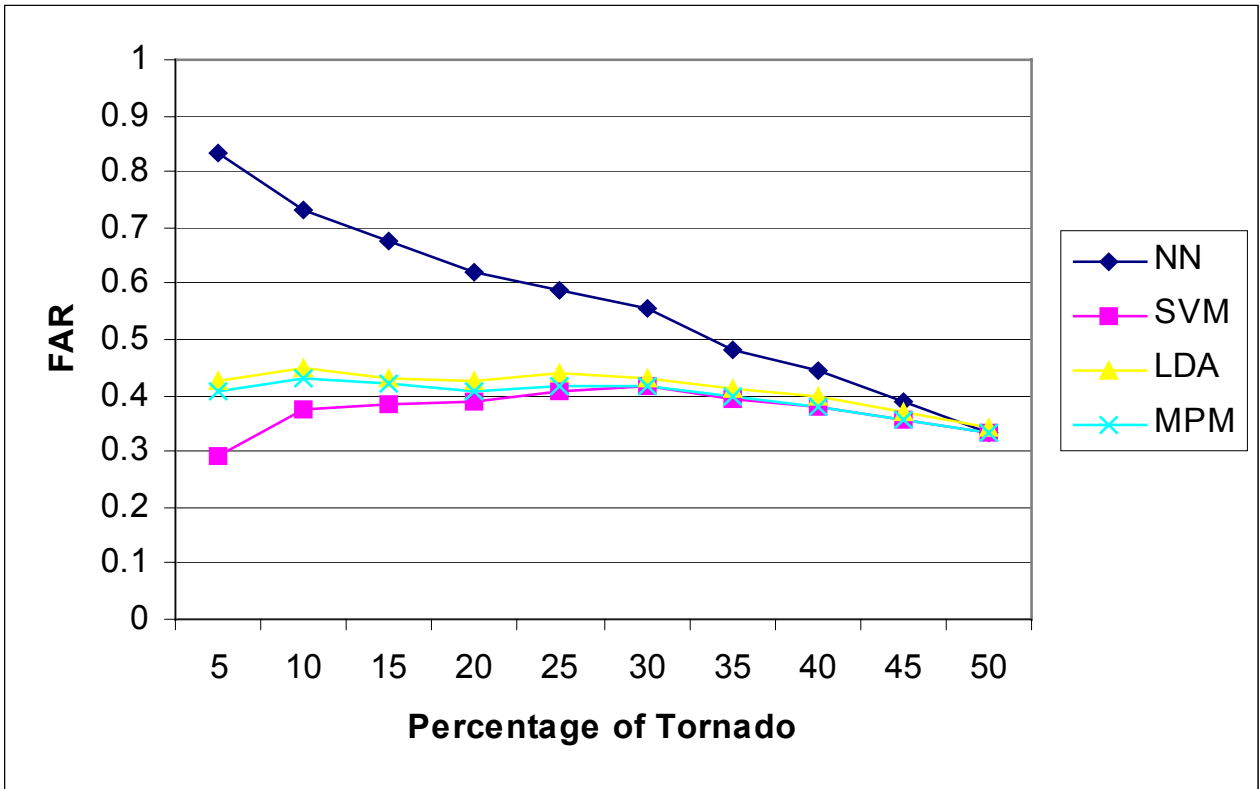


# Results based on new sampling strategies

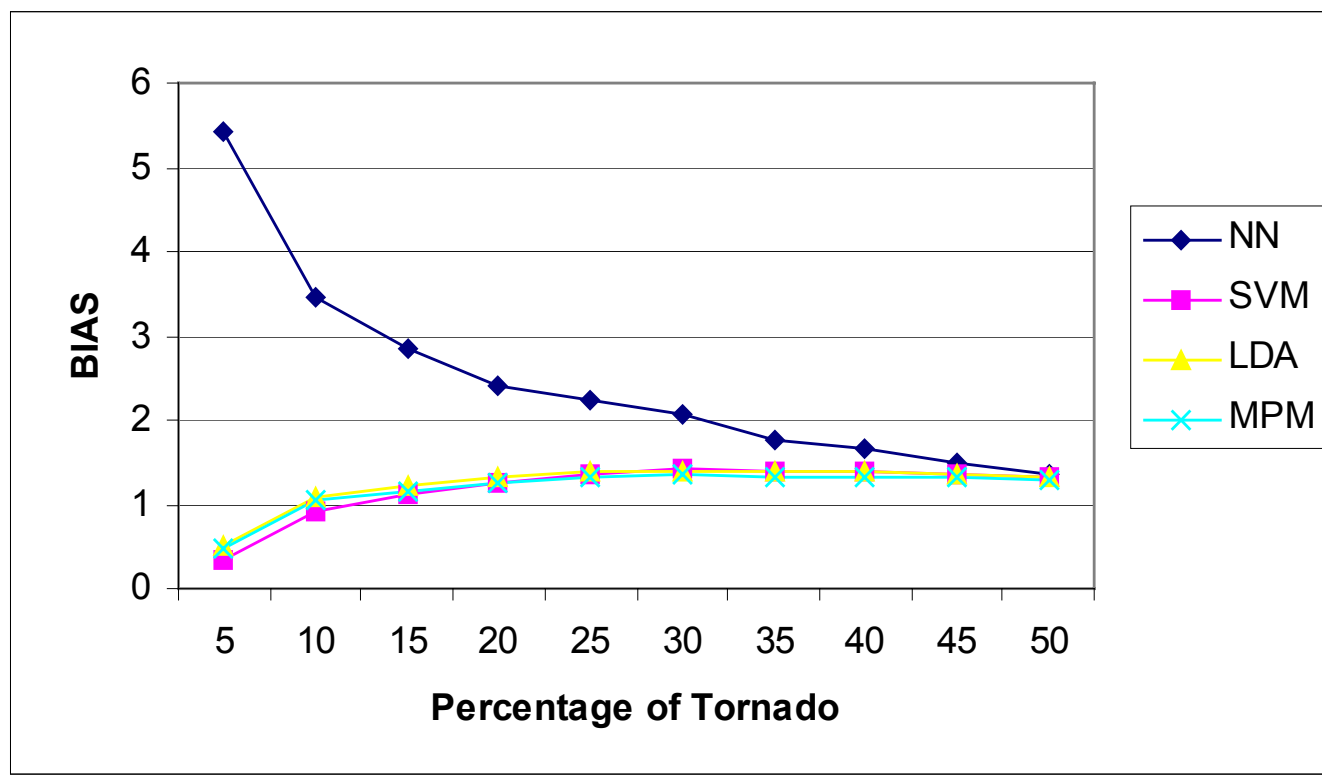
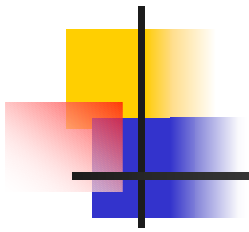
---

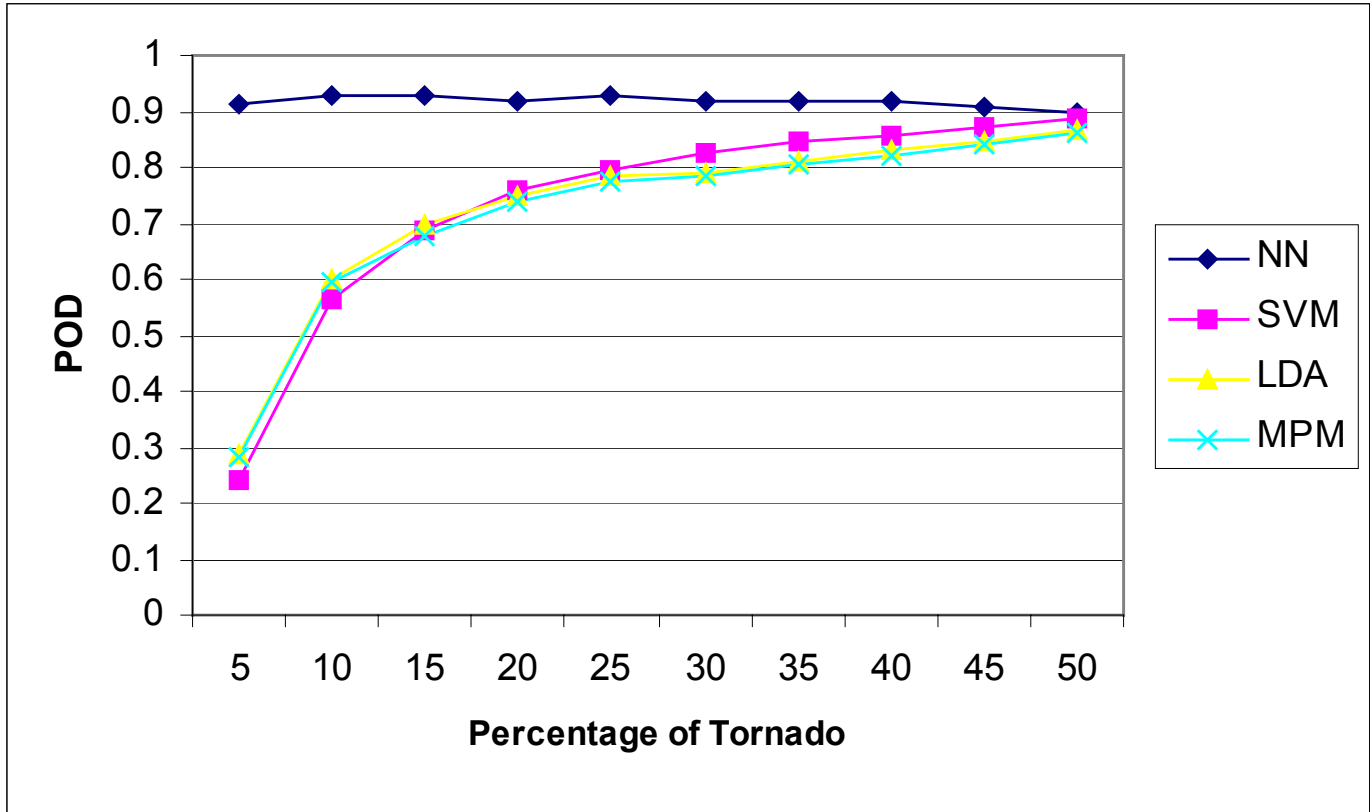
- Randomized sampling has been altered (tornado and nontornado observations selected from different dates)
- Added a new kernel based learning machine (MPM)
- Vary the ratio of tornado to nontornado observations

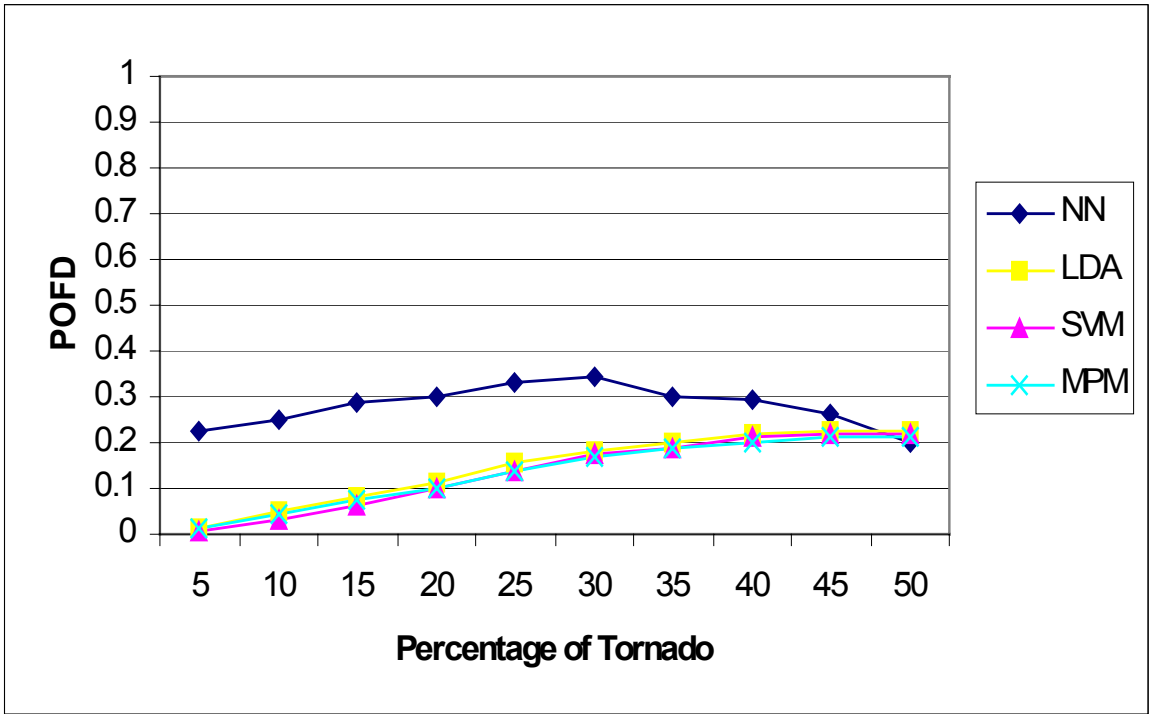


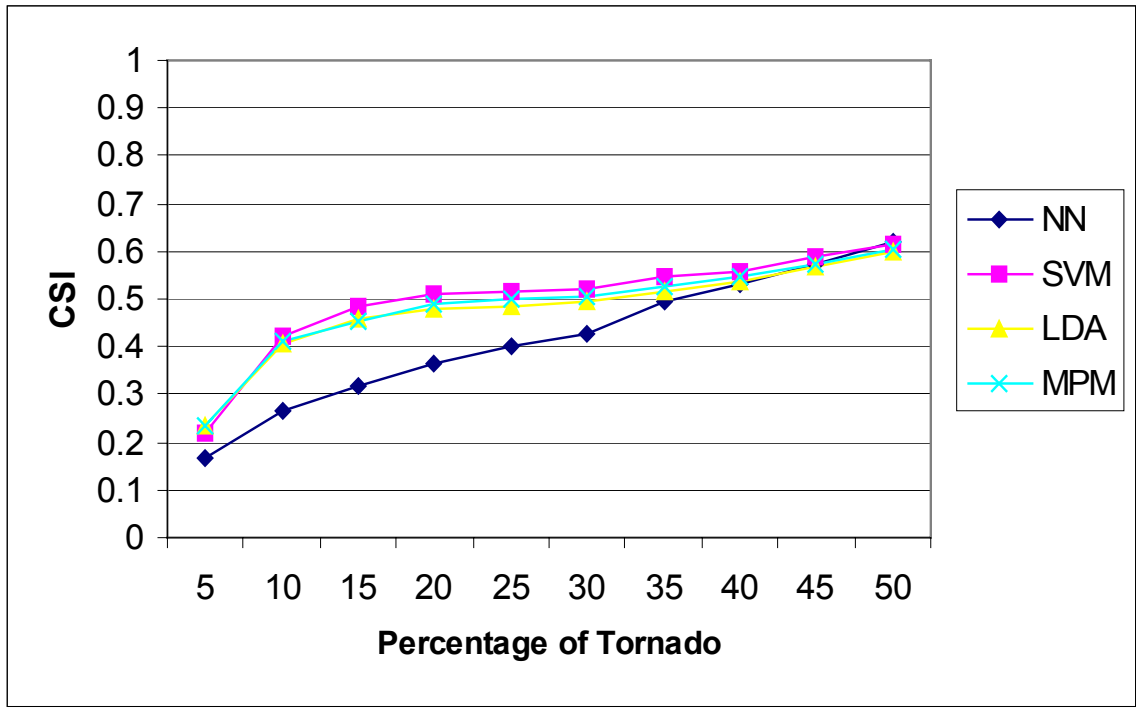
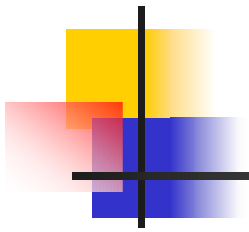


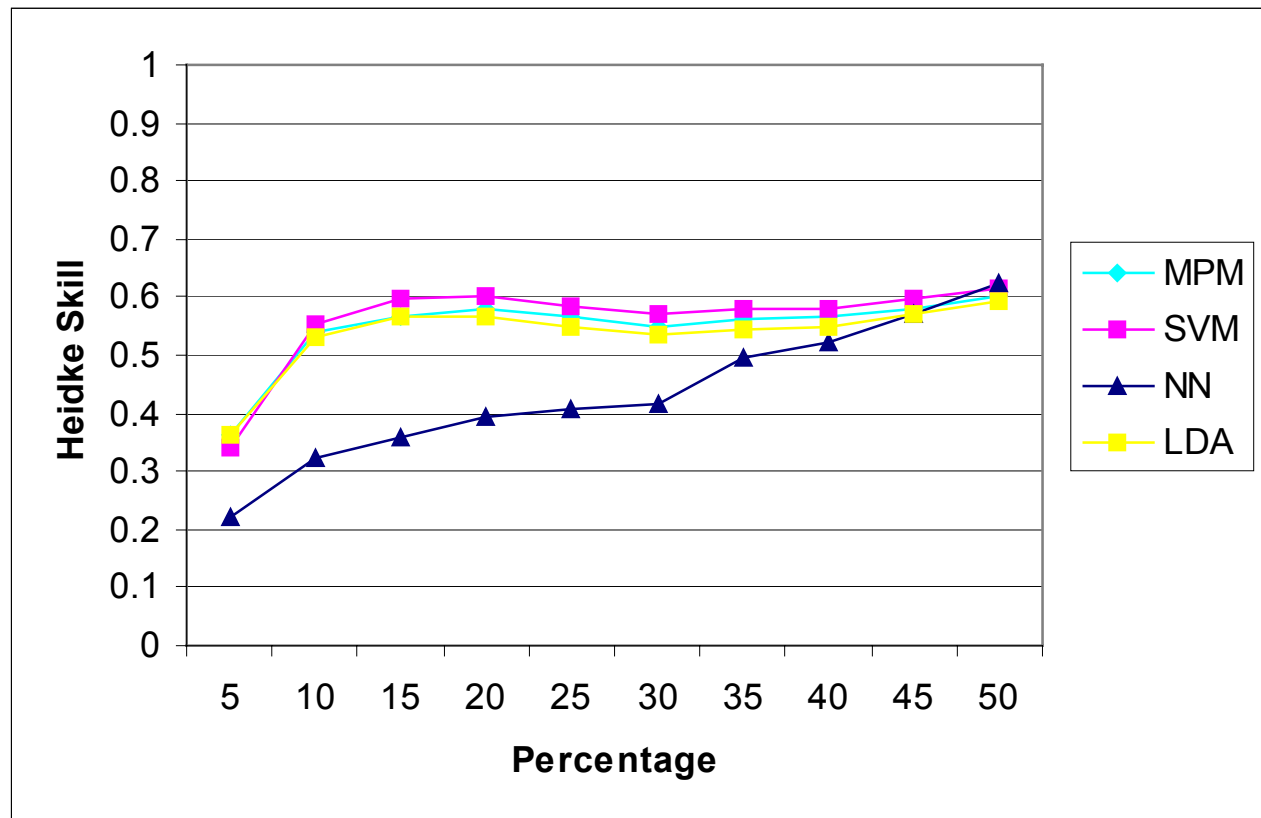
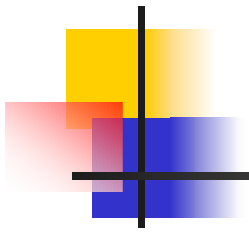














# Conclusion

---

- A new method has been used for forecasting tornadoes
- Accuracy of the SVM method depends on the free parameters of the kernel. SVM is compared with other data-driven techniques
- Several forecasting indices are used for comparison
- Next step is to incorporate the above algorithms in the tornado detection algorithms of NSSL

Funded through NSF EIA-0205628