#83 Machine Learning to Recognize Phenomena in Large Scale Simulations

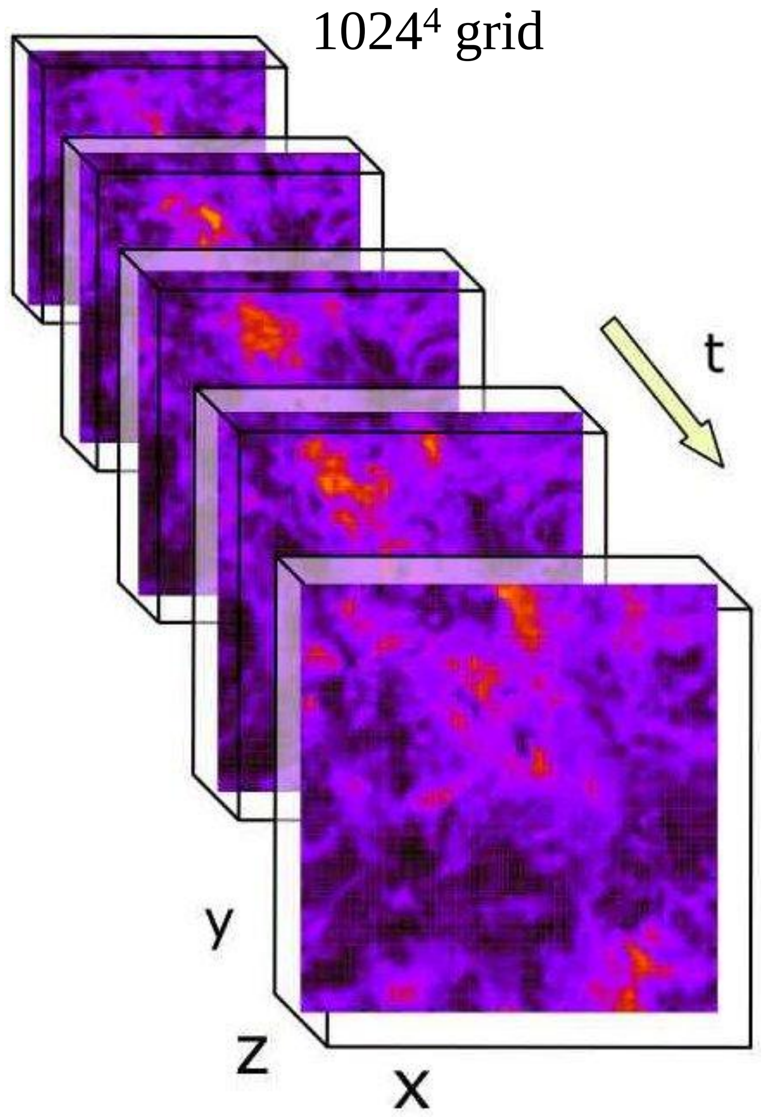
Carnegie Mellon

Auton Lab

Jeff Schneider, Barnabas Poczos, Liang Xiong, Dougal Sutherland

JHU Turbulence Dataset [3]

1024^4 grid



PROBLEM

Given a petabyte of simulation data
- what interesting happened?
- how often did it happen?
- when? where?

CONCEPT

- A "phenomenon" is a spatial group of data points from a grid or particle-based simulation
- Need machine learning algorithms (anomaly detectors, classifiers) to operate on groups of data points
- Treat a group as a set of i.i.d. samples from an underlying feature distribution for the groups
- Develop algorithms to do machine learning on distributions using only samples from them

METHOD

- Propose Support Distribution Machine (SDM)
- Develop non-parametric methods to estimate inner products between distributions using i.i.d. samples from them
- Use estimates to create Gram matrix
- Project estimated Gram matrix to SPD cone
- Use 2-class/1-class SVM for classification/anomaly detection

Benchmark 8-class scene classification problem [5]

- Use SIFT features from black&white images
- Reduce dimensionality to 19 with PCA
- Approximately 350 images in each of 8 categories
- 10-fold cross validation

• **SDM Accuracy:**	**89.95%**
• Best published with similar feature set:	85.60%
• Best published using spatial dependence, color:	91.57%

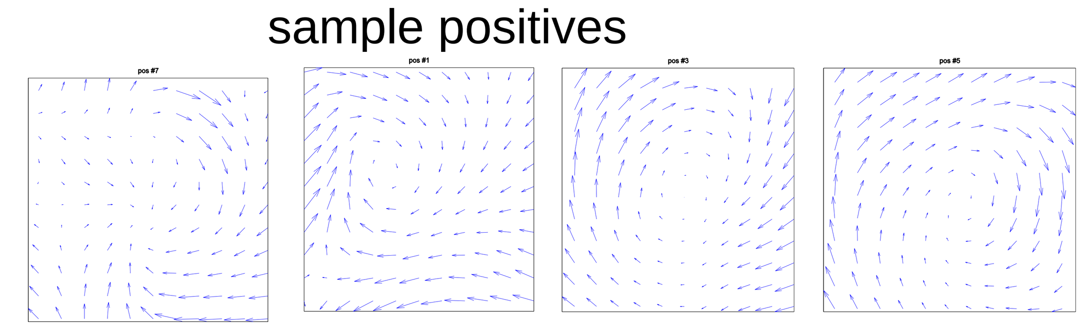
SAMPLE IMAGES

Finding Vortices in JHU Turbulence Simulation [5]

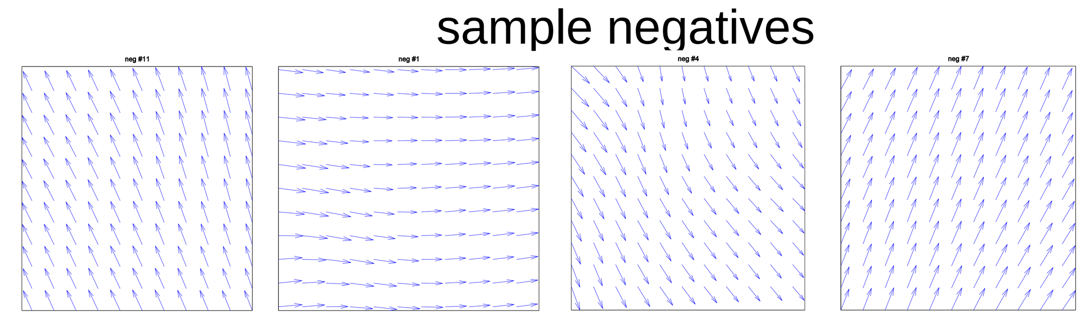
- Features are velocity field, pressure, distance from center
- Anomaly detection: train 1-class SDM on 100 randomly chosen regions
- Classification: train SDM with 20 hand-labeled negatives and 11 positives

- **SDM 2-class leave-one-out accuracy: 97%**

sample positives



sample negatives



Constructing Kernels

- must estimate K(p,q) for distributions p and q
- many useful kernels have a form that can be derived from:

$$D_{\alpha,\beta}(p\,\|\,q)=\int p^{\alpha}(x)q^{\beta}(x)p(x)dx$$

- linear: $K(p,q)=\int pq$
- polynomial: $K(p,q)=(\int pq+c)^s$
- Gaussian: $K(p,q)=\exp(-\frac{\mu^2(p,q)}{2\sigma^2})$

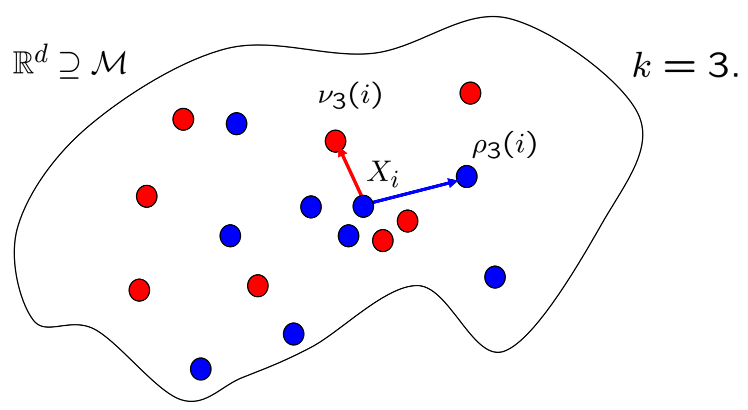
where μ =

Euclidean: $\mu(p,q)=\int (p-q)^2$

Hellinger: $\mu(p,q)=1-\int p^{1/2}q^{1/2}$

Renyi: $\mu(p,q)=\frac{1}{\alpha-1}\log\int p^{\alpha}q^{1-\alpha}$

Non-parametric Estimation



$k \geq 1$, fixed.

$\rho_k(i)$: the distance of the k-th nearest neighbor of X_i in $X_{1:n}$

$\nu_k(i)$: the distance of the k-th nearest neighbor of X_i in $Y_{1:m}$

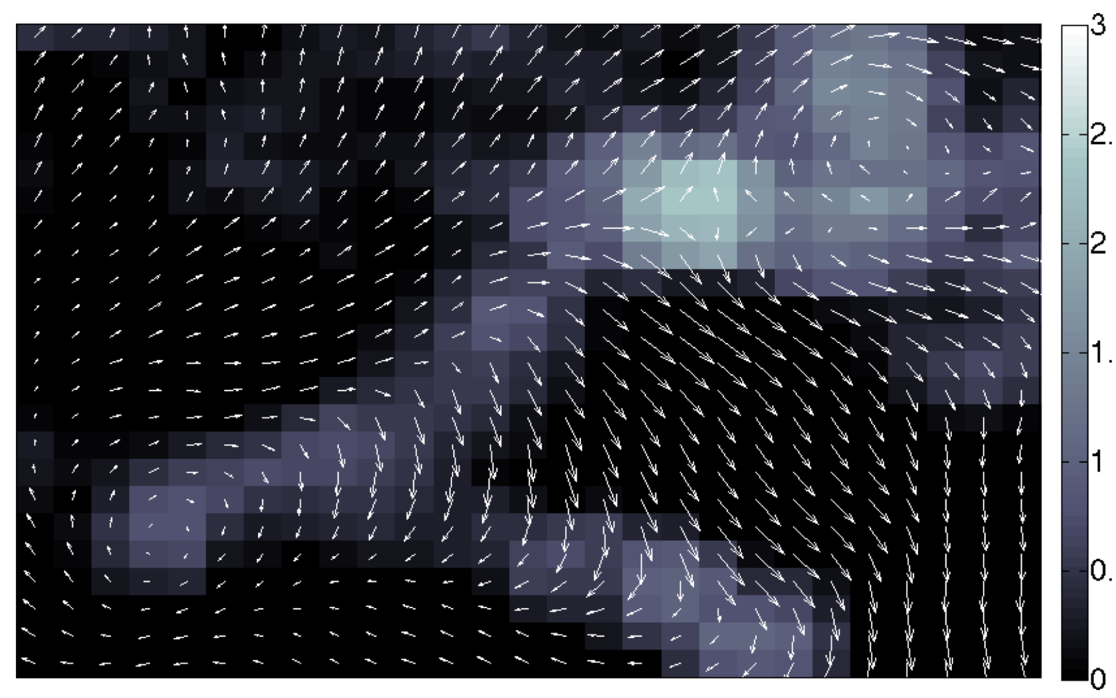
$$\hat{D}_{\alpha,\beta}(X_{1:n}\,\|\,Y_{1:m})=\frac{1}{n}\sum_{i=1}^{n}(n-1)^{-\alpha}m^{-\beta}\rho_k^{-d\alpha}(i)\nu_k^{-d\beta}(i)B_{k,\alpha,\beta}$$

Estimator is provably consistent with minor assumptions [1]

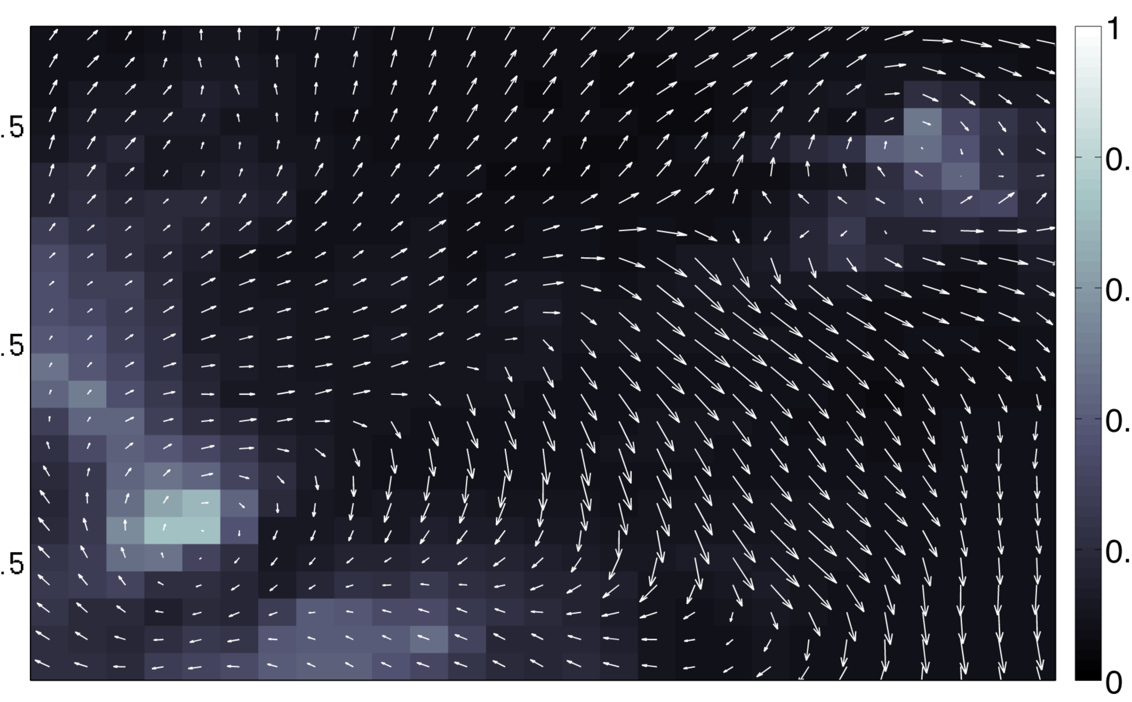
problem: resulting Gram matrix may not be positive semidefinite

solution: project back to the cone of positive semidefinite matrices using the alternating projections method of [2]

Anomaly Score



Vortex Class Probability



Support Distribution Machine (SDM)

- Use estimated Gram matrix
- Plug into standard dual form of the "soft SVM" [4]
- Result is a quadratic program that may be solved with many different tools

References

[1] B. Poczos, J. Schneider, "On the Estimation of alpha-Divergences", AI and Statistics, 2011.

[2] N. Higham, "Computing the Nearest Correlation Matrix: A Problem from Finance", IMA Journal of Numerical Analysis, 2002.

[3] JHU Turbulence Database Cluster, http://turbulence.pha.jhu.edu/

[4] B. Scholkopf, A. Smola, "Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond, MIT Press, 2002.

[5] A. Oliva, A. Torralba, "Modeling the shape of the scene: a holistic representation of the spatial envelope". *International Journal of Computer Vision,* 2001.