Anomaly Detection of Trajectories

Junier B. Oliva

Anomaly Detection

- An anomaly (or outlier) in a dataset is an instance that is abnormal or unlikely based on the rest of the dataset

- If the instances in the dataset are labeled as {normal, abnormal} then standard *supervised* machine learning (ML) techniques may be used to perform anomaly detection

- This thesis focuses on the case where there are no labels, for which one must rely on *unsupervised* ML techniques

Possible Uses for Anomaly Detection in Trajectories

- Security—an agent moving in an abnormal fashion may be up to no good

- Novelty detection—perhaps new pathways may have opened up for agents' travels

- Sensor inspection—faulty odometers, and other localization sensors will likely deliver abnormal trajectories

Notation

Consider dataset $\mathcal{D}$, of trajectories: $\mathcal{D} = \{t^{(1)}, \ldots, t^{(N)}\}$ where each trajectory $t^{(i)}$ is an ordered collection of n_i 2D points that correspond to the location of the agent i at regular intervals; i.e., $t^{(i)} = \langle (x_1^{(i)}, y_1^{(i)}), \ldots, (x_{n_i}^{(i)}, y_{n_i}^{(i)}) \rangle$, and $\forall i, j \; (x_j^{(i)}, y_j^{(i)}) \in \mathcal{S} \subseteq \mathbb{R}^2$.

One-Class SVM

- Find maximum margin hyperplane from origin such that *most* instances are on positive side:

One-Class SVM Optimization

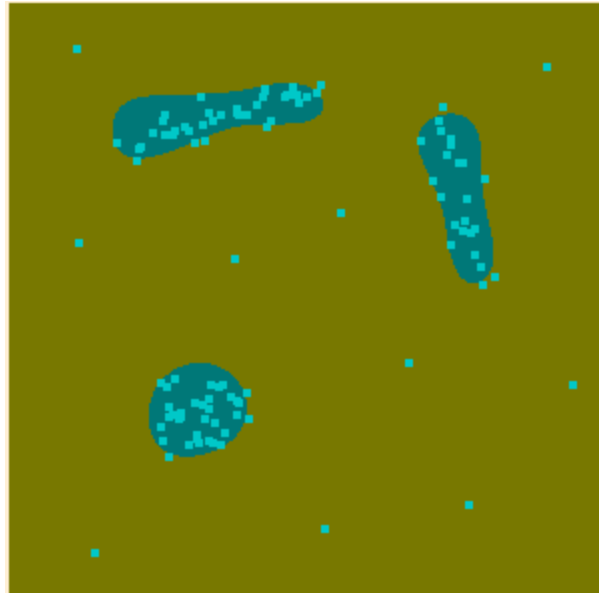- To solve the problem, optimize the following quadratic problem:

$$\min_{w \in F, \boldsymbol{\xi} \in \mathbb{R}^{\ell}, \rho \in \mathbb{R}} \quad \frac{1}{2}\|w\|^2 + \frac{1}{\nu\ell}\sum_i \xi_i - \rho$$

$$\text{subject to } (w \cdot \Phi(\mathbf{x}_i)) \geq \rho - \xi_i, \quad \xi_i \geq 0.$$

Non-linear feature space, via the kernel-trick

# Non-linear Example

- Using Gaussian Kernel: exp(- ɣ * ||x-y||^2)

One-Class Anomaly Detection for Trajectories

- The one-class SVM technique can be applied to trajectories so long as one finds appropriate representations and kernels to use

- It is possible to find kernels that can be used directly with trajectories of different lengths through various different spatial representations

Discrete Spatial Distribution Representation (DSDR)

- A way to build a spatially informative representation of a trajectory is to build a distribution over a quantized space such that the probability of drawing a space is high for spaces near the trajectory.

- One way to build such a distribution:

  1. Turn on indicators on quantized spaces that the trajectory passes through
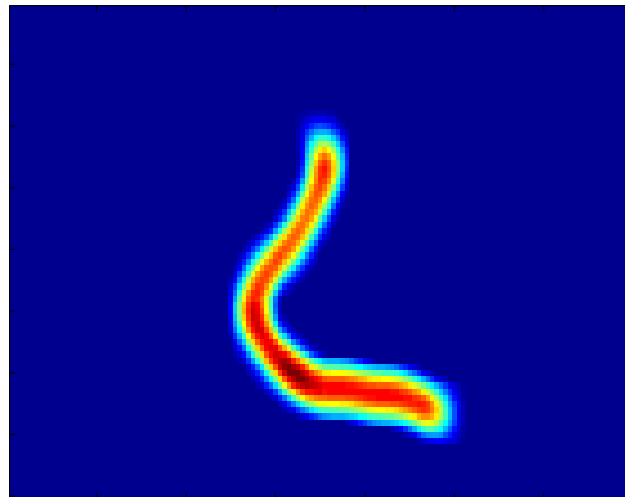 2. Convolve a Gaussian through the map
  3. Normalize

DSDR Example
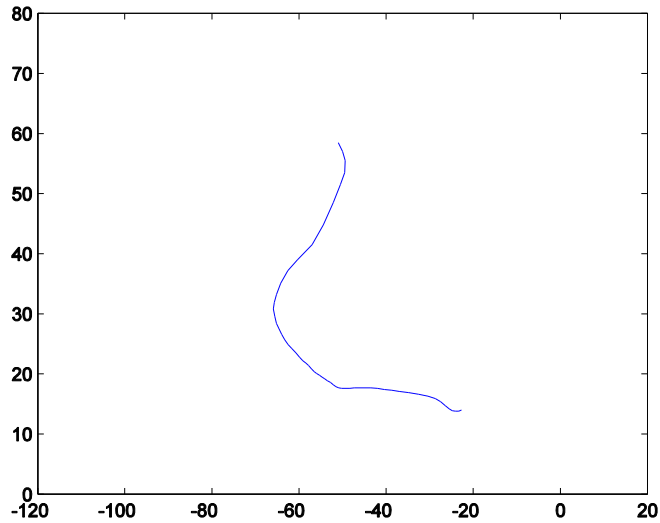
(a) Example Coordinates
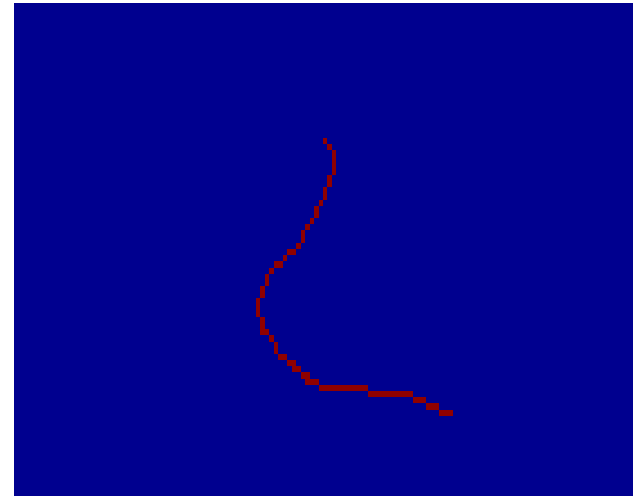
(b) Example Indicators

(c) Example DSDR

Additional Dimensions

- The representations discussed so far have compared trajectories using *first-order* information about the locations traveled. That is, only the snapshot *XY* positions are compared; angular and speed information is not.

- Hence, two trajectories that travel through the same space, but using varying speeds and direction will be indistinguishable. However, this can be easily remedied by extending the kernels to include additional dimensions.

- For example, instead of discretizing the space into a 2D matrix of indicators, one may discretize it into a 3D matrix of indicators where the third dimension is either speed or orientation

Discrete Angular Expectation Representation (DAER)

135º   90º   45º

180º   0º

225º   270º   315º
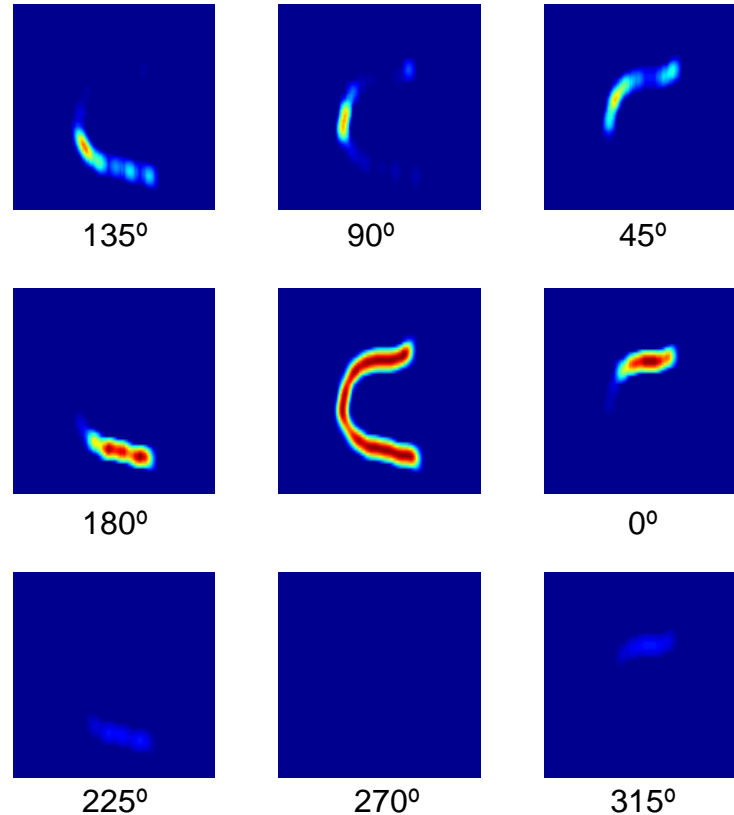
A Gaussian can be convolved on a 3d matrix of indicators where the 3rd dimension is angular position (see above). Again, using a Gaussian Kernel with this representation can help find anomalies.

# Discrete Speed Expectation Representation (DSpER)
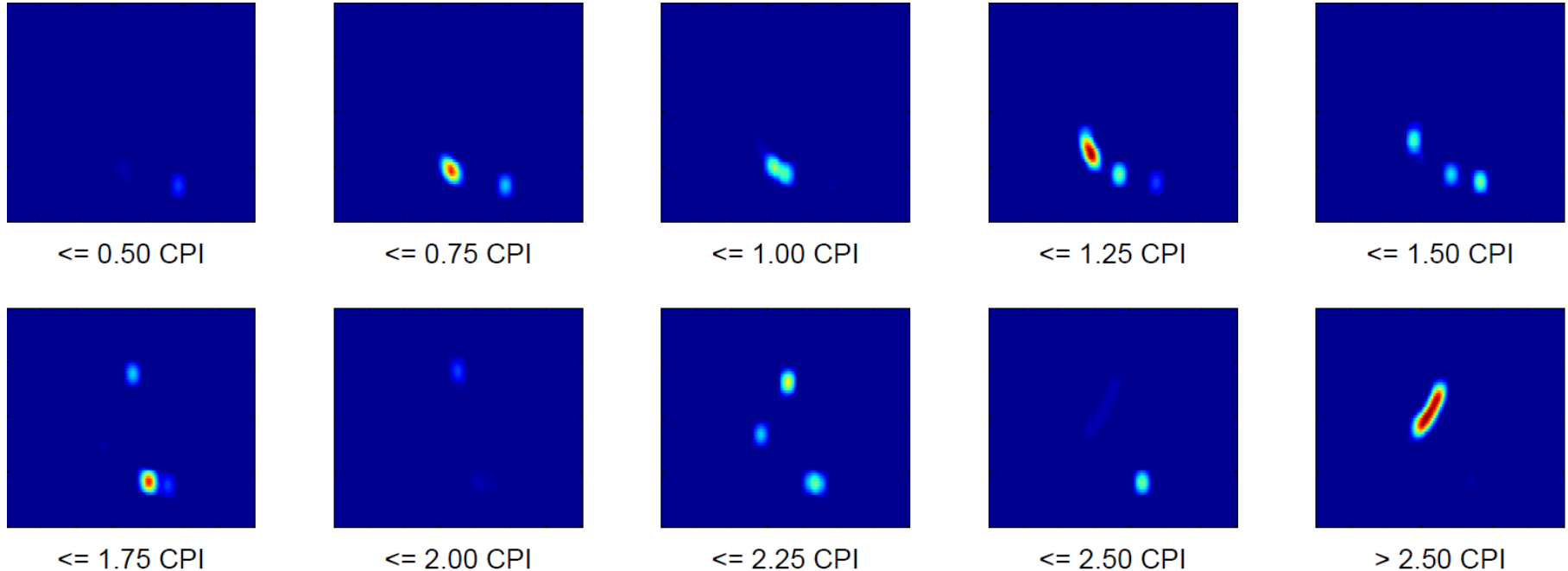
The DSpER for a trajectory. The first two dimensions correspond to spatial location and the third to speed. Speed is measured in terms of coordinates per interval (CPI). In the case of hurricanes the coordinates are Lat/Long and intervals are 6 hours. The third dimension is rolled out in the 10 images shown above, corresponding to the following intervals (top-left to bottom-right): {[0, .5], (.5, .75], …, (2.25, 2.5],(2.5, ∞]}.
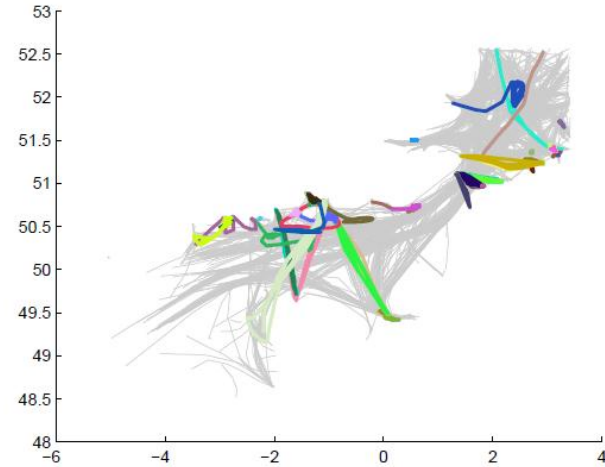
Experiments

- Experiments were ran on both real world data sets using the representations described

- Used LIBSVM implementation
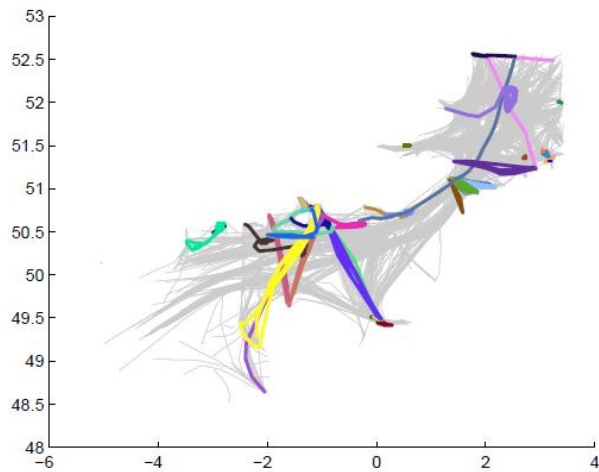 - ν was selected to be .03 leading to roughly 3% of dataset to be labeled as outliers
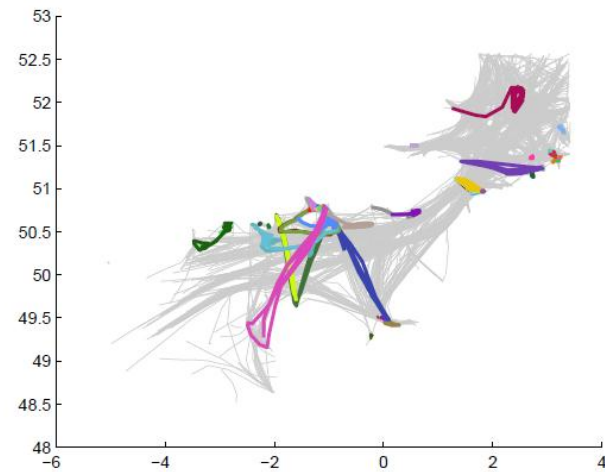
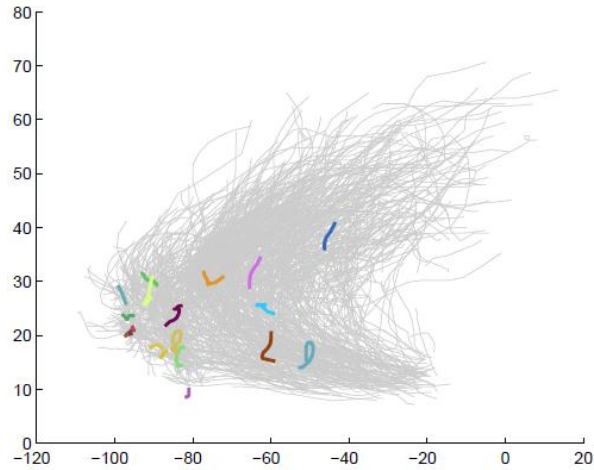# Results: AIS Dataset
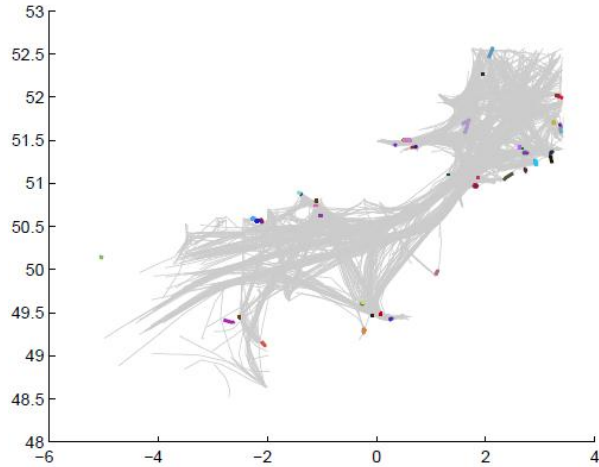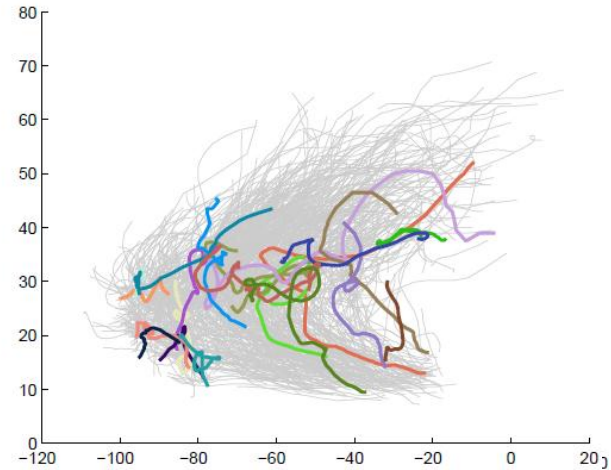

(a) DSDR

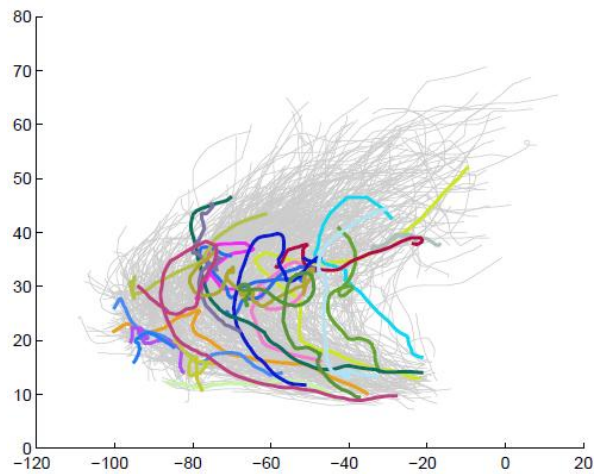(b) DSER

(c) DAER
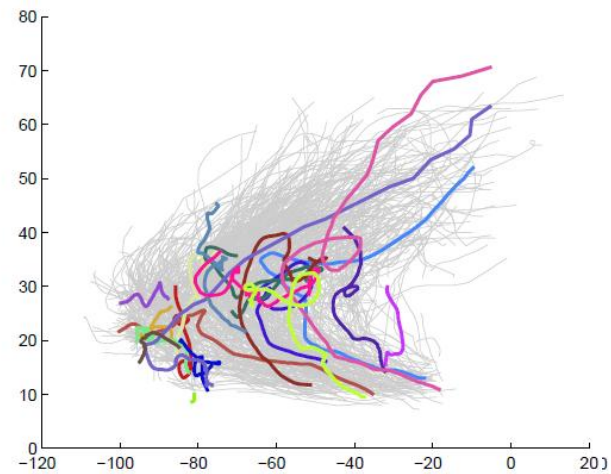
(d) DSpER

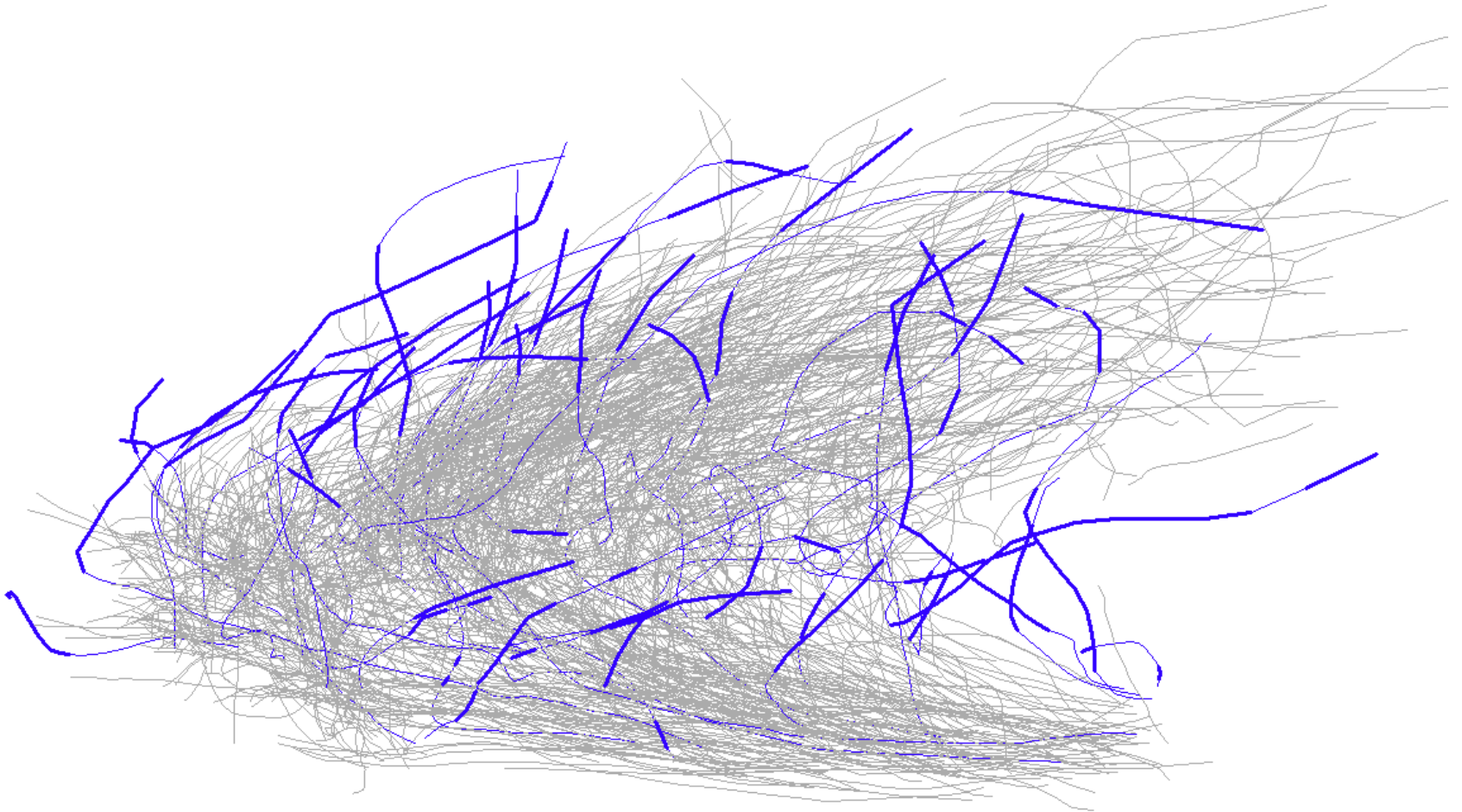# Results: Hurricane Dataset

(a) DSDR

(b) DSER

(c) DAER

(d) DSpER

Comparison

- Lee et. al. 2008

SVM Conclusion:

- Methodology does a good job at capturing what appear to be anomalous trajectories.

  – Trajectories that are going against the grain compared to other nearby trajectories, or ones that are at uncommon locations are found.

- Future work:

  – Parameter selection

  – Assess the quality of outliers returned by unsupervised methods

End

Thank You!