DB-DRIFT

Concept drift aware density-based anomaly detection for maritime trajectories

Amelia Henriksen, Sandia National Laboratories

Sensor Signal Processing for Defence, 2023
September 13, 2023
MARITIME ANOMALY DETECTION
BAD THINGS HAPPEN TO PEOPLE ON THE OCEAN

- **Cargo loss**
  - Photo by Petty Officer 3rd Class Matthew West (10.28.2018), DVIDS

- **Sinking**
  - Photo by Petty Officer 3rd Class Matthew West (10.28.2018), DVIDS

- **Grounding**
  - Photo by U.S. Coast Guard District 8, (12.21.2002), DVIDS

- **Medical Emergencies**
  - Photo by U.S. Coast Guard District 7 PADET Tampa Bay (08.26.2012), DVIDS

- **Losing power/propulsion**
  - James Brickwood/SMH 07.05.2022, 9News

- **Fire**
PEOPLE DO BAD THINGS ON THE OCEAN

- Trade sanction dodging
- Illegal fishing
- Vessel type spoofing
- Smuggling
- False route reporting (High collision risk)
- Terrorism
HOW DO WE FIND ANOMALIES?

• Many data sources, both public and proprietary, for monitoring vessel tracks.

• Example: All large ships are required by international law to be equipped with **automatic identification system data (AIS)**

HOW DO WE FIND ANOMALIES?

The Problem:
Maritime vessel data is
• Unlabelled
• Large
• Noisy
• Prone to changes in the underlying distribution
HOW DO WE FIND ANOMALIES?

The Common Problem:
Maritime vessel data is
• **Unlabelled**
• Large
• Noisy
• Prone to changes in the underlying distribution

Anomaly detection algorithms for vessel tracks are largely **unsupervised** (UAD)
**HOW DO WE FIND ANOMALIES?**

**The Common Problem:**
Maritime vessel data is
- Unlabelled
- **Large**
- Noisy
- Prone to changes in the underlying distribution

Expensive (and sometimes impossible) for experts* to assess all tracks.

* Domain experts OR expensive expert algorithms
The Problem: Maritime vessel data is
- Unlabelled
- Large
- Noisy
- Prone to changes in the underlying distribution

Basic UAD: Assume the majority of samples are normal and identify outliers.
**The Problem:**
Maritime vessel data is
- Unlabelled
- Large
- Noisy
- Prone to changes in the underlying distribution

**Basic UAD:**
Assume the majority of samples are **normal** and identify **outliers**.

If the norm changes, our model needs to change with it.
The Problem:
Maritime vessel data is
• Unlabelled
• Large
• Noisy
• Prone to changes in the underlying distribution

This is concept drift.

Today we focus on two kinds of drift:
1. Gradual drift
2. Seasonal drift
GRADUAL CONCEPT DRIFT

- One of the most well understood forms of concept drift
- Describes the slow, consistent evolution of data over time.
SEASONAL CONCEPT DRIFT

- Describes patterns that appear repeatedly in the data in a periodic way

- Vessel movements are affected by the earth’s literal meteorological seasons.
ACCOUNTING FOR MULTIPLE TYPES OF CONCEPT DRIFT

• Almost all modern vessel track UAD pipelines don’t account for any concept drift.

• In the few cases where it is incorporated, only gradual drift is addressed.

• How do we solve this problem?
**DBSCAN**: Density-based spatial clustering of applications with noise [1]

- Can automatically identify outliers
- Has few hyperparameters
- Does not need a pre-set number of clusters (unlike k-means).

---


https://github.com/NSHipster/DBSCAN
DBSCAN AND MARITIME ANOMALY DETECTION

**DBSCAN**: Density-based spatial clustering of applications with noise [1]

- Can automatically identify outliers
- Has few hyperparameters
- Does not need a pre-set number of clusters (unlike k-means).

**Examples in Maritime Anomaly Detection:**


**DBSCAN**: Density-based spatial clustering of applications with noise [1]

- Can automatically identify outliers
- Has few hyperparameters
- Does not need a pre-set number of clusters (unlike k-means).

1. DBSCAN is still fundamentally a static method.

2. We’d like to incorporate multiple forms of drift

---

THE DB-DRIFT ALGORITHM

Step 1.
Automatic Identification System (AIS) Data

Step 2.
Trajectory Processing

Step 3.
Gradual Drift Model

Step 4.
Seasonal Drift Model

Combined Score
Threshold

Raw Data Stream

Trajectory n-features

Outlier ID/Score

Outlier ID/Score

Outlier ID/Score

Combined Score
Threshold
THE DB-DRIFT ALGORITHM

Step 1.
Automatic Identification System (AIS) Data → Raw Data Stream

Step 2.
Trajectory Processing

This setup starts most UAD pipelines on trajectories.

The algorithm must process an incoming stream of trajectories (or trajectory segments.)
THE DB-DRIFT ALGORITHM

Step 1.

Automatic Identification System (AIS) Data → Raw Data Stream

Step 2.

Trajectory Processing

This setup starts most UAD pipelines on trajectories.

The algorithm must process an incoming stream of trajectories (or trajectory segments.)

This setup starts most UAD pipelines on trajectories.

The algorithm must process an incoming stream of trajectories (or trajectory segments.)

Sandia National Labs’ Tracktable is our BEST FRIEND for this stage in the pipeline.

https://tracktable.sandia.gov/
The **CONTRIBUTION**

**THE DB-DRIFT ALGORITHM**

**Step 3.**
- Gradual Drift Model
- Seasonal Drift Model

**Step 4.**
- Combined Score
- Threshold

**Outlier ID/Score**
GRADUAL DRIFT

We want:
• Model that works quickly on a stream with low overheads
• Model that emphasizes more recent “normal behavior” over past behavior.

Naïve approach: Simple sliding window model
GRADUAL DRIFT

• Better approach: Damped window model.
  • Fades (re-weights) old samples as new samples arrive
  • Controlling the fade factor lets you control the rate at which the model evolves.

\[ w_t(x) = 2^{-\lambda(t - T_0(x))} \]

- $w_t(x)$: Weight at time $t$ for sample $x$
- $\lambda$: Fade factor
- $T_0(x)$: Arrival time for sample $x$
We choose DenStream [1] as our core model:

1. Uses the damped window model
2. It can easily identify outliers in real time
3. It has low memory requirements
4. It requires very short burn in period (typically only a few days worth of data)

---

Algorithm 1 Denstream for outlier detection at time $t$

Parameters: Max radius $\epsilon$, minimum p-microcluster weight $\mu/\beta$, pruning stepsize $T_p$

\[
\lambda \leftarrow 1/T_p \ast \log_2(\beta \mu/(\beta \mu - 1))
\]

\begin{algorithmic}
\ForEach {sample $x$ s.t. $T(x) = t$} \Comment{Merge step}
    \State Find the nearest p-microcluster $p^*_t \in P_t$.
    \If{radius of $\{p^*_t, x\} \leq \epsilon$}
        \State Add $x$ to $p^*_t$.
    \Else
        \State \text{Report the outlier score as } \min_{p^*_t \in P_t} \|x - \epsilon(p^*_t)\|
        \State Find the nearest o-microcluster $o^*_t \in O_t$
        \If{radius of $\{o^*_t, x\} \leq \epsilon$}
            \State Add $x$ to $o^*_t$
            \If{weight $w(o^*_t, t) > \mu/\beta$}
                \State Move $o^*_t$ from $O_t$ to $P_t$
            \Else
                \State Add $\{x\}$ to $O_t$ as a new o-microcluster.
        \EndIf
    \EndIf
\EndFor
\If{$t \% T_p = 0$} \Comment{Pruning step}
    \For{$p \in P_t$}
        \If{$w(p,t) \leq \mu/\beta$}
            \State Remove $p$ from $P_t$
        \EndIf
    \EndFor
    \For{$o \in O_t$}
        \If{$w(o,t) \leq \xi(t, T_p, o) = \frac{2^{-(\lambda(t-T_0(o) + T_p)} - 1}{2^{-\lambda T_p} - 1}$}
            \State Remove $o$ from $O_t$
        \EndIf
    \EndFor
\EndIf
\end{algorithmic}

---

HOW DENSTREAM WORKS

p-microclusters (aka “normal” points)
HOW DENSTREAM WORKS

o-microclusters (aka “outlier” points)
If enough points are added to an o-microcluster, its weight passes a threshold.

The o-microcluster then becomes a p-microcluster.
If new points aren’t added to a p-microcluster, the weight decreases.

If the weight decreases enough, the p-microcluster is pruned.
Similarly, if no points have been added to an o-microcluster, the weight goes below a threshold $\xi$ and it is pruned.
If an incoming point is an outlier, output the distance to the nearest p-microcluster as the outlier score.
THE DB-DRIFT ALGORITHM

Step 3.

Gradual Drift Model

Seasonal Drift Model

Step 4.

Combined Score

Threshold

Outlier ID/Score

Outlier ID/Score
Handling seasonal drift for density-based clustering is an open field of research—nearly no prior work.
SEASONAL MODEL

• General idea: assign a separate model for each season [1, 2].

SEASONAL MODEL

- General idea: assign a separate model for each season [1, 2].
- For known periodic seasons, there can be seasonal anomaly detectors at multiple scales (months, quarters, weeks, etc).
SEASONAL MODEL

• General idea: assign a separate model for each season [1, 2].
• For known periodic seasons, there can be seasonal anomaly detectors at multiple scales (months, quarters, weeks, etc).
For our preliminary experiments, we have focused on the **monthly model**.
IMPORTANT NOTE:
Seasonal drift is a subset of recurrent drift.

Expanding this algorithm to find seasons (in addition to defining known seasons) is a very challenging ongoing effort.
Step 4.

How do we define the combined outlier score?

How do we choose (and update) the appropriate threshold for a point to be considered an outlier?
THE OUTLIER CONDITION

For sample $x$ at time $t = T(x)$:

$$\hat{O}(x) = w_g \min_{g_{t,i} \in G_t} \|x - c(g_{t,i})\| + w_s \min_{s_{t,i} \in S_t} \|x - c(s_{t,i})\| \geq \theta_{i,j}$$

$G_t$: The set of p-micro-clusters $g_{t,i}$ for the gradual model at time $t$

$S_t$: The set of p-micro-clusters $s_{t,i}$ for the seasonal model at time $t$

$c(\cdot)$: The center of a given microcluster (the fade-weighted sum of the points)

$w_g, w_s$: The outlier-score weights for the gradual and seasonal models.

We set $w_s = \frac{2}{3}, w_g = \frac{1}{3}$ to emphasize the importance of seasonal anomalies.

$\theta_{i,j}$: For time $T(x)$ after some sample time period $[t_i, t_j]$, the minimum threshold for a sample to be considered an outlier based on the desired anomalous subset size.
THE OUTLIER CONDITION

For sample $x$ at time $t = T(x)$:

$$\hat{O}(x) = w_g \min_{g_{t,i} \in G_t} \|x - c(g_{t,i})\| + w_s \min_{s_{t,i} \in S_t} \|x - c(s_{t,i})\| \geq \theta_{i,j}$$

$G_t$: The set of p-micro-clusters $g_{t,i}$ for the gradual model at time $t$

$S_t$: The set of p-micro-clusters $s_{t,i}$ for the seasonal model at time $t$

$c(\cdot)$: The center of a given microcluster (the fade-weighted sum of the points)

$w_g, w_s$: The outlier-score weights for the gradual and seasonal models.

We set $w_s = \frac{2}{3}, \ w_g = \frac{1}{3}$ to emphasize the importance of seasonal anomalies.

$\theta_{i,j}$: For time $T(x)$ after some sample time period $[t_i, t_j]$, the minimum threshold for a sample to be considered an outlier based on the desired anomalous subset size.
The whole point of anomaly detection on maritime surveillance to:

1. Process data too big for experts to process
2. Find potential anomalies outside expert detection.
What is the point of UAD at sea?

We want to output a tractable subset of points that contain the trajectories of interest—not a final anomaly decision.

Could be a human expert, or an expensive expert algorithm.
THE OUTLIER CONDITION

For sample \( x \) at time \( t = T(x) \):

\[
\hat{\Theta}(x) = w_g \min_{g_{t,i} \in G_t} \| x - c(g_{t,i}) \| + w_s \min_{s_{t,i} \in S_t} \| x - c(s_{t,i}) \| \geq \theta_{i,j}
\]

\( G_t \): The set of \( p \)-micro-clusters \( g_{t,i} \) for the gradual model at time \( t \)

\( S_t \): The set of \( p \)-micro-clusters \( s_{t,i} \) for the seasonal model at time \( t \)

\( c(\cdot) \): The center of a given microcluster (the fade-weighted sum of the points)

\( w_g, w_s \): The outlier-score weights for the gradual and seasonal models.

We set \( w_s = \frac{2}{3}, w_g = \frac{1}{3} \) to emphasize the importance of seasonal anomalies.

\( \theta_{i,j} \): For time \( T(x) \) after some sample time period \( [t_i, t_j] \), the minimum threshold for a sample to be considered an outlier based on the desired anomalous subset size.
THE OUTLIER CONDITION

\[ \theta_{i,j} = \begin{cases} 
Q \left( \{ \hat{\theta} > 0 \}_{T(x) \in [t_i, t_j]}, 1 - q_{i,j} \right) & q_{i,j} < 1 \\
0 & \text{otherwise}
\end{cases} \]

\[ q_{i,j} = \frac{n_{t_i,t_j}^r}{\hat{n}_{t_i,t_j}} \]

- \( r \): The desired percentage of the dataset to return as an anomalous subset.
- \([t_i, t_j]\): A sample time period used to determine \( \theta \) for incoming points.
- \( Q(X, q) \): The \( q \)'th sample quantile for a set of scalar values \( X \).
- \( n_{t_i,t_j} \): The number of samples that arrived during period \([t_i, t_j]\).
- \( \hat{n}_{t_i,t_j} \): The number of samples \( x \) with outlier score \( \hat{\theta}(x) > 0 \).
EXAMPLE: THE HAWAIIICOAST_GT DATASET
THE HAWAIICOAST_GT DATASET

• Fully open and FAIR dataset available at https://zenodo.org/record/8253611 [1].

• Curated AIS data from MarineCadastre.gov [2]-[5].

• Includes 208 labelled tracks corresponding to 154 real-world incidents.

PRELIMINARY RESULTS

Our goal was to **improve performance** for UAD pipelines with DBSCAN.

We compare it to **sliding window DBSCAN** over a range of window sizes (reporting best results over a range of hyperparameters).

<table>
<thead>
<tr>
<th>Method</th>
<th>Real incidents captured</th>
<th>Intersection with DB-Drift</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB-Drift</td>
<td>22</td>
<td></td>
</tr>
<tr>
<td>DBSCAN 2w</td>
<td>14</td>
<td>13</td>
</tr>
<tr>
<td>DBSCAN 3w</td>
<td>17</td>
<td>15</td>
</tr>
<tr>
<td>DBSCAN 4w</td>
<td>17</td>
<td>14</td>
</tr>
<tr>
<td>DBSCAN 8w</td>
<td>18</td>
<td>14</td>
</tr>
</tbody>
</table>

Test: anomalous fishing vessel trajectories from HawaiiCoast_GT. Total real world incidents: 74 (varied anomaly types)
PRELIMINARY RESULTS

DB-Drift captures more real incidents than DBSCAN

DBSCAN vs DB-Drift Detection

- Total real incidents captured
- Intersection with DB-Drift

DB-Drift captures more real incidents than DBSCAN
PRELIMINARY RESULTS

DB-Drift captures most of the incidents captured by DBSCAN
More advantages:

- DB-Drift requires a burn in of only a few days, DBSCAN requires at least 1 window period.

- Significantly lower memory requirements
NEXT STEPS:

**Experiments:**
- Trajectory feature optimization to improve overall performance.
- Additional tests for each vessel class and specific anomaly types.
- Curating further datasets using our ground truth technique for additional benchmarking.

**Algorithm:**
- Season discovery vs known seasons.
- Adding abrupt drift detection to reweight historical information.
BIG THANKS

Feedback and Mentorship

Ben Newton

Andy Wilson

David Stracuzzi

Making so much data publicly available