Anomaly detection in CERN LHC injection magnets

Thiebout Dewitte[2], Armin Halilovic[2], Niels Wéry[2], Elia Van Wolputte[1], Pieter Van Trappen[3], Wannes Meert[1], Hendrik Blockeel[1]
[1]{firstname.lastname}@cs.kuleuven.be, [2]{firstname.lastname}@student.kuleuven.be, [3]pieter.van.trappen@cern.ch
[1,2]KU Leuven, Belgium - [3]CERN

Context

Injection Kicker Magnets (MKI) inject proton beams into the LHC.

PROBLEM:

Manual discovery and analysis of anomalies in MKI data is a tedious process that does not scale.

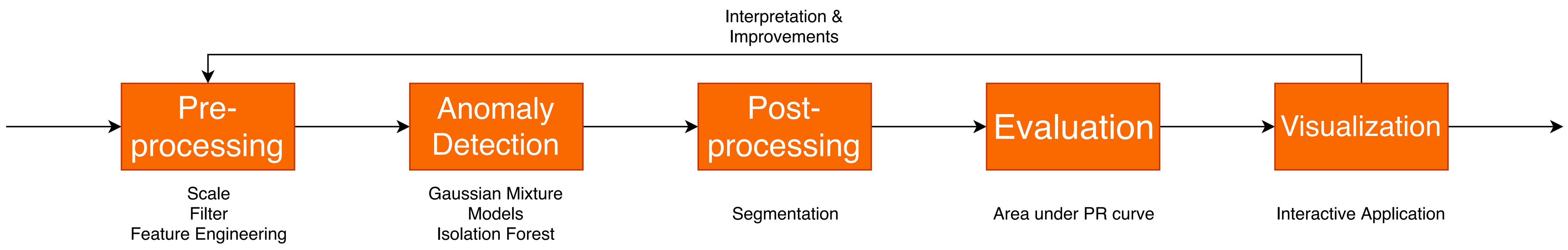
GOAL:

Develop an anomaly detection application that can automatically detect anomalous behavior, based on historical data.

Main Challenges

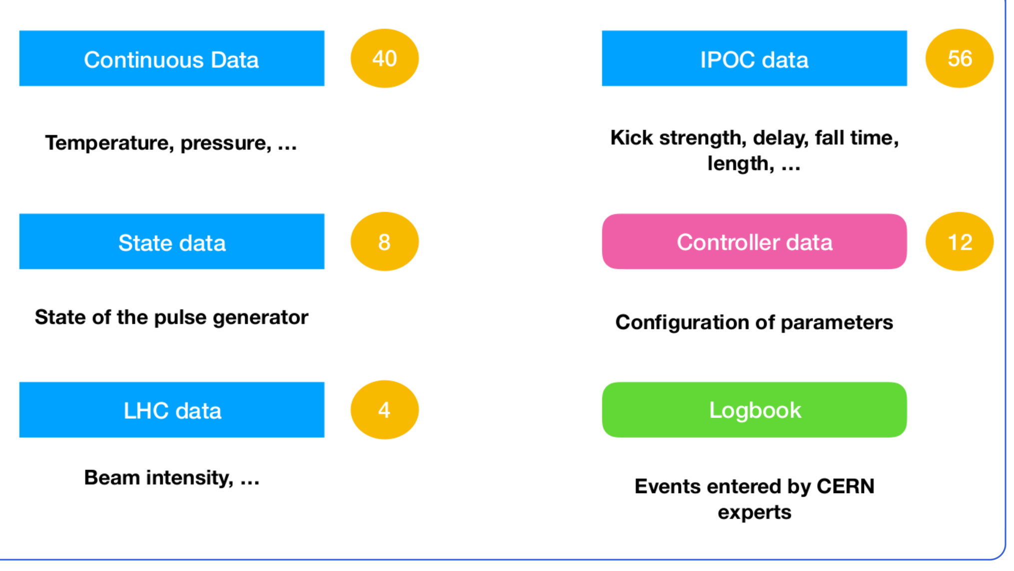
1. How to deal with the **high-variety** and **high-volume** of **data**?
2. How to deal with **non-consistent "*normal*" behaviour**?
3. How to **incorporate manual logbook entries** that label the historic anomalies?
4. How to **evaluate** our algorithm?
5. How to **interpret and act upon** the outputs of our algorithms?

Pipeline



1. Preprocessing

- Design **filters** for measurement errors
- **Merge data** from different sources, to build a sensible **feature vector**.
- Use sliding windows to compute ***"temporal"* features**.

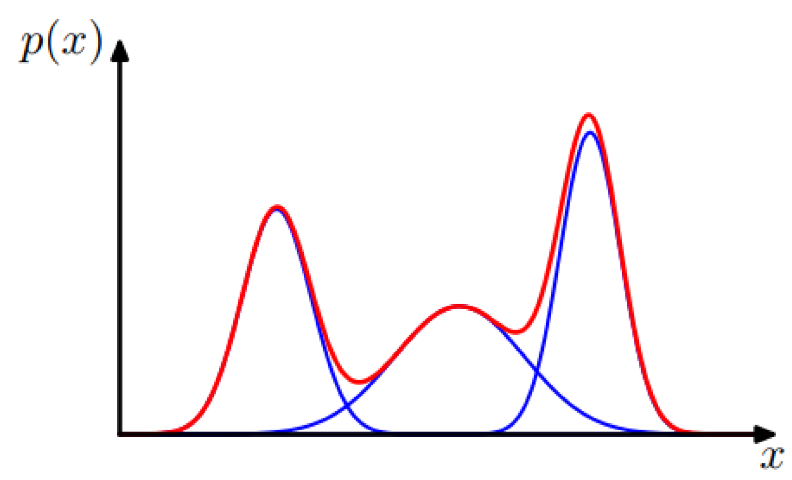


2. Anomaly Detector

Gaussian Mixture Models

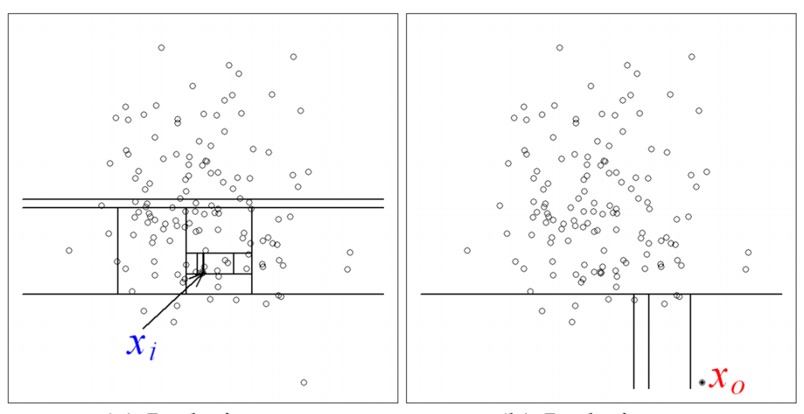
What? Fit all the data to a *mixture* of a finite number of *Gaussian distributions* with unknown parameters.
Why? The mixture model represents the underlying data-generation procedure. If a datapoint has a very low probability, this indicates it was anomalous.
Upsides? Scales OK, low number of components is typically alright.
Downsides? Interpretability is limited. Correct number of components is hard to determine. Can have difficulty in high-dimensional problems.

$$p(\vec{x}) = \sum_{k=1}^{K} \pi_k \cdot \mathcal{N}(\vec{x}|\vec{\mu}_k, \Sigma_k)$$



Isolation Forests

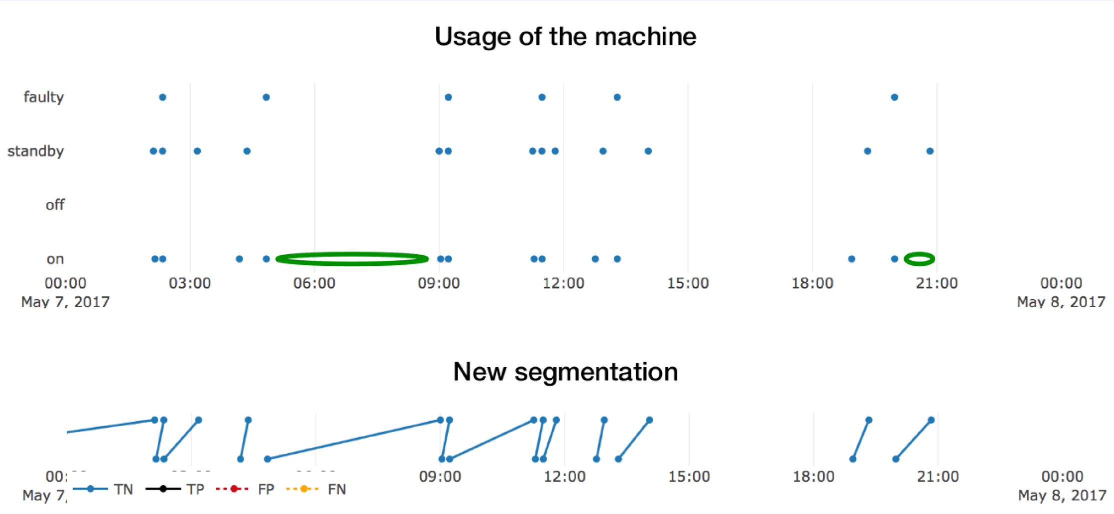
What? Learn an ensemble of *isolation trees*. This is a random tree structure which aims to isolate individual points
Why? The assumption is that anomalies are easier to isolate. Anomalous points will be found in leaf nodes with a shorter average path length to the root node.
Upsides? No need for data normalization. Performs in high-dimensional problems (looks in sub-spaces anyway.) Some interpretability since trees can explicitly be inspected.
Downsides? Heavier computation, more algorithmic, less principled.



(a) Isolating x_i (b) Isolating x_o

3. Segmentation

- Our algorithms yields **anomaly scores** $\in [0,1]$ **for each datapoint**.
- This has little meaning, we are interested **if a certain period of time was anomalous**.
- Based on controller data, we build **meaningful segments**, based on usage intervals.



4. Evaluation

- Use *top* $-$ *k* scores of individual points as a **score for the entire segment**.
- Ground truth comes from **manual logbook entries**. A detection within a **12-hour timeframe** preceding such a logbook entry is considered a successful detection. Our evaluation metric is adjusted accordingly.
- We use **precision and recall** as metrics. For hyperparameter tuning, we consider the area under the PR-curve.

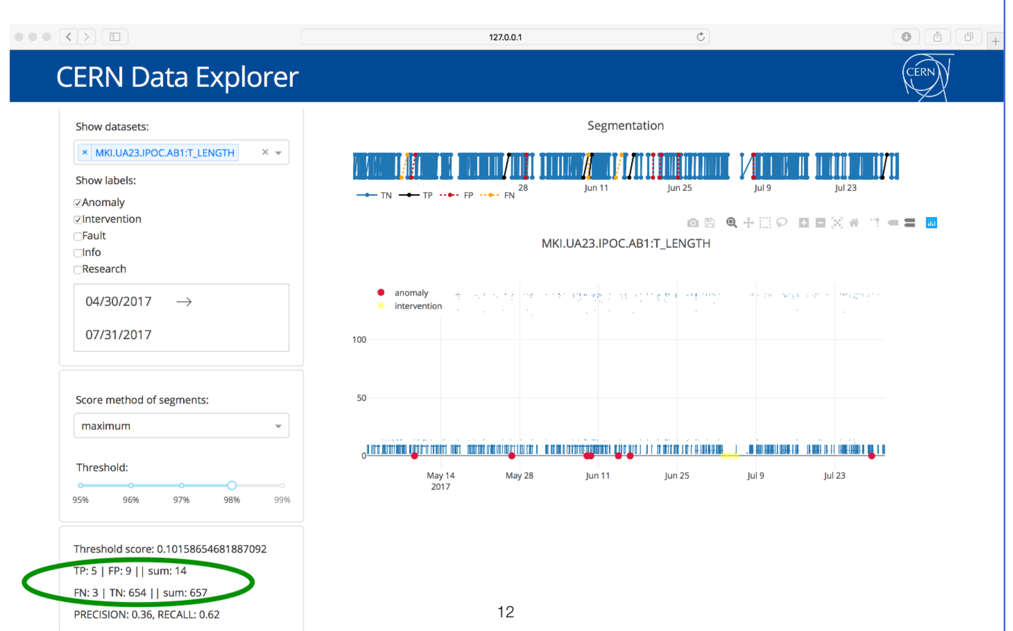
Results for a 3 month period. The detector succeeds reasonably well in separating normal from anomalous behaviour. If the detector flags something as anomalous, it is worthwhile to investigate this in detail. These false positives are possibly real anomalies that went undetected. The false negatives are more of a concern, since some kinds of anomalies still go undetected.

	Anomaly	Normal
Detected	$TP = 7$	$FP = 5$
Undetected	$FN = 3$	$TN = 585$

$$Precision = 0.58$$
$$Recall = 0.70$$

5. Visualization

- Did the **algorithm** make a **mistake**?
- Did **we overlook** an anomaly?
- Why did the algorithm indicate an anomaly?
- What **caused** the anomaly?
- Step one: **Interactive visualization** of results.



6. Future Work

Interpretability

How? Using **multi-directional models (e.g. MERCS)**, we hope to build our own, interpretable anomaly detector on top of an ensemble of predictive functions.
Why? Now we have to guess why certain segments were flagged by the detector, slowing down the analysis. An interpretable anomaly detector allows us learn more, faster.

Supervision

How? Semi-supervised clustering (e.g. COBRAS) listens to users subjective preferences and incorporates them into its clustering. Ideally, this happens in an interactive manner.
Why? The only way we currently interact with the algorithm is by adjusting hyperparameters after the fact. If we learn something new, we would like to directly incorporate this knowledge into the algorithm.

CERN erc fwo DTAI KU LEUVEN

LaTeX TikZposter