Network Data Analysis Laboratory

Proposed projects

Francesco Musumeci

Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB)

Politecnico di Milano, Milano, Italy

2022-2023

General requirements (1/2) – Projects tasks

- All projects **must** include the following main steps:
  1. Raw data visualization/analysis
     - Does data suggest anything (ML algorithm, proper features)?
  2. Data preprocessing (e.g., generate features from raw data)
  3. ML models optimization and training
     - Hyperparameters tuning with cross-validation
  4. Testing the performance and evaluating different scenarios (specified in the assignment):
     - What is the impact of including/excluding different features?
     - What is the impact of feature normalization?
     - What is the impact of training set size on algorithms' performance?
     - What happens if some features are missing for some data points?

General requirements (2/2) – Methodology

- All projects **must** include (for 13 points)
  - 2/3 ML algorithms:
    - the one recommended in the assignment: Neural Network or a tree-based model (e.g., Random Forest, XGBoost)
    - a "simpler" one: linear/logistic regression, K-nearest-neighbors
  - Performance metrics: MSE, MAE, Accuracy, Precision, Recall, F-score, training duration, …

- Additionally, projects **may** include "advanced" tasks (we will discuss in another lab in detail) (for 2 points)
  - Transfer learning: train on one domain, test on another domain (e.g., different datasets, different tasks, etc.)
  - Federated learning: compare "global" models (all data available at one location) vs. "local" models that share knowledge through Fed. Learning
  - Explainability: apply XAI (eXplainable Artificial Intelligence) frameworks to interpret/explain model reasoning and validate the model

Project #4 – Anomaly detection
Background

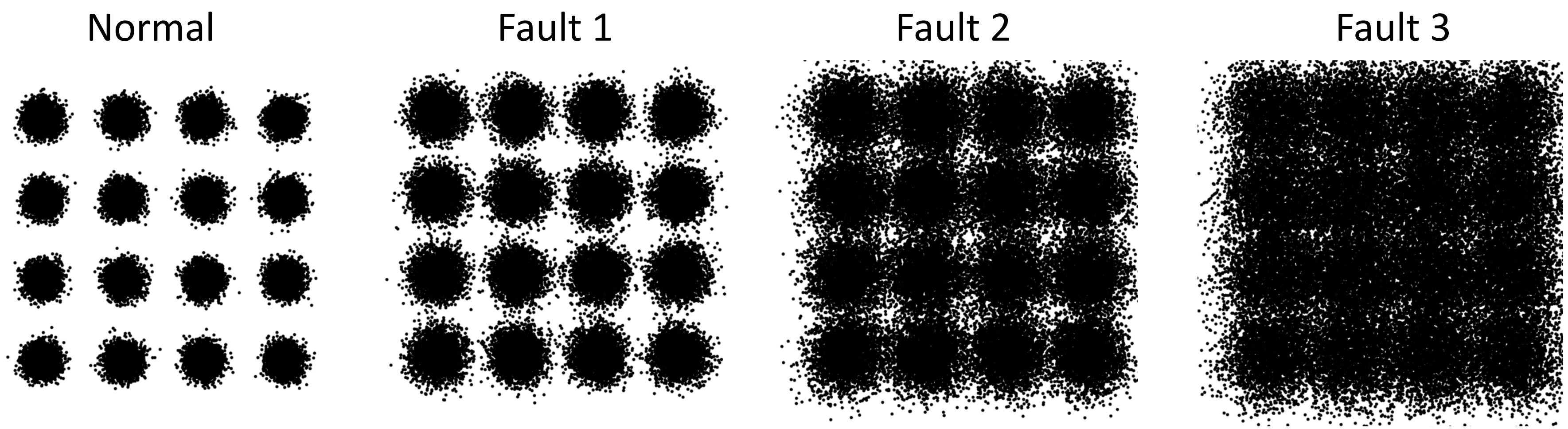
- Lasers in optical transponders can develop faults or be misconfigured
  - Signal can have higher power than expected
  - Channel central frequency can drift
- Signals transmitted by faulty transponders will create extra interference for the neighboring channels

- We may not have any examples of signals experiencing unusual interference from an operational network
- However, we have lots of examples of signals operating under normal conditions

- Can we use ML to automatically detect faults without any data for faulty scenarios during training?

Project #4 – Anomaly detection
Dataset

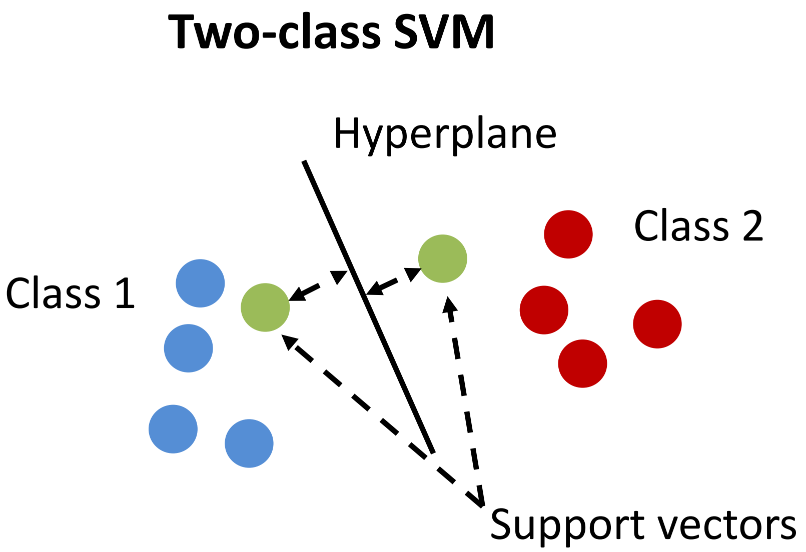
- Images of the 16QAM constellation at the receiver of Channel Under Test (CUT) [1]
- 5 classes with 100 image each
  - 2 normal classes: no close interferes or interferer 50GHz away and with the same power as CUT
  - 3 faulty classes (faulty transceiver lasers):
    - Fault 1: Interferer 50GHz away but with 6 dB higher power than CUT
    - Fault 2: Interferer 43.75 GHz away with 3 dB higher power than CUT
    - Fault 3: Interferer 37.5 GHz away with the same power as CUT

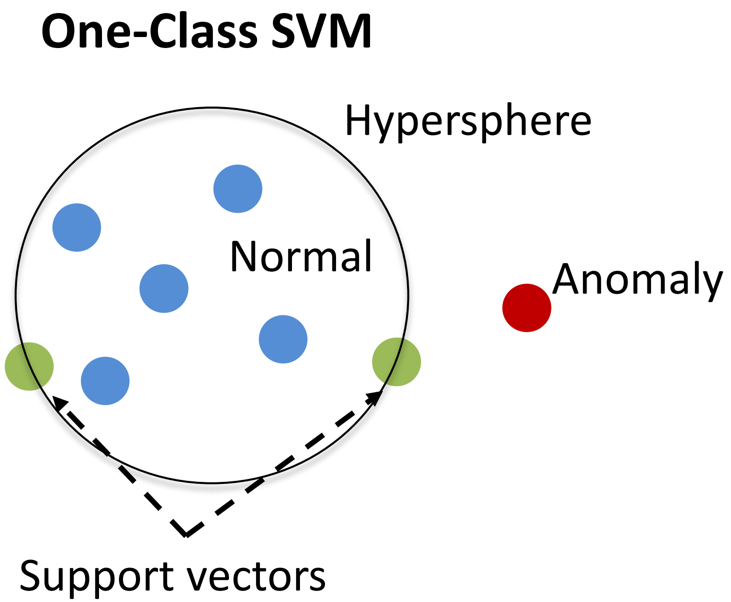


[1] https://ieee-dataport.org/open-access/constellation-diagrams-spectrum-anomaly-detection-optical-networks

Project #4 – Anomaly detection
One-Class SVM

- Two-class SVM finds a hyperplane to separate two classes, maximizing the distance from the instances closest to the hyperplane (support vectors)
- **One-Class SVM** finds the hypersphere to enclose all instances of the normal class, minimizing its radius. Instances on the surface of the hypersphere are support vectors





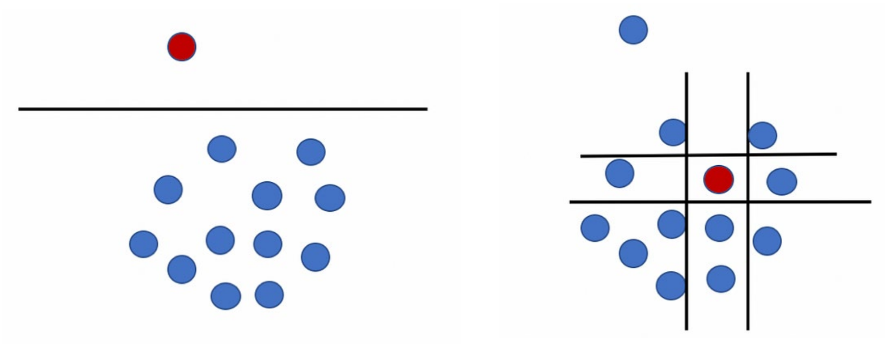
[1] https://scikit-learn.org/stable/modules/generated/sklearn.svm.OneClassSVM.html

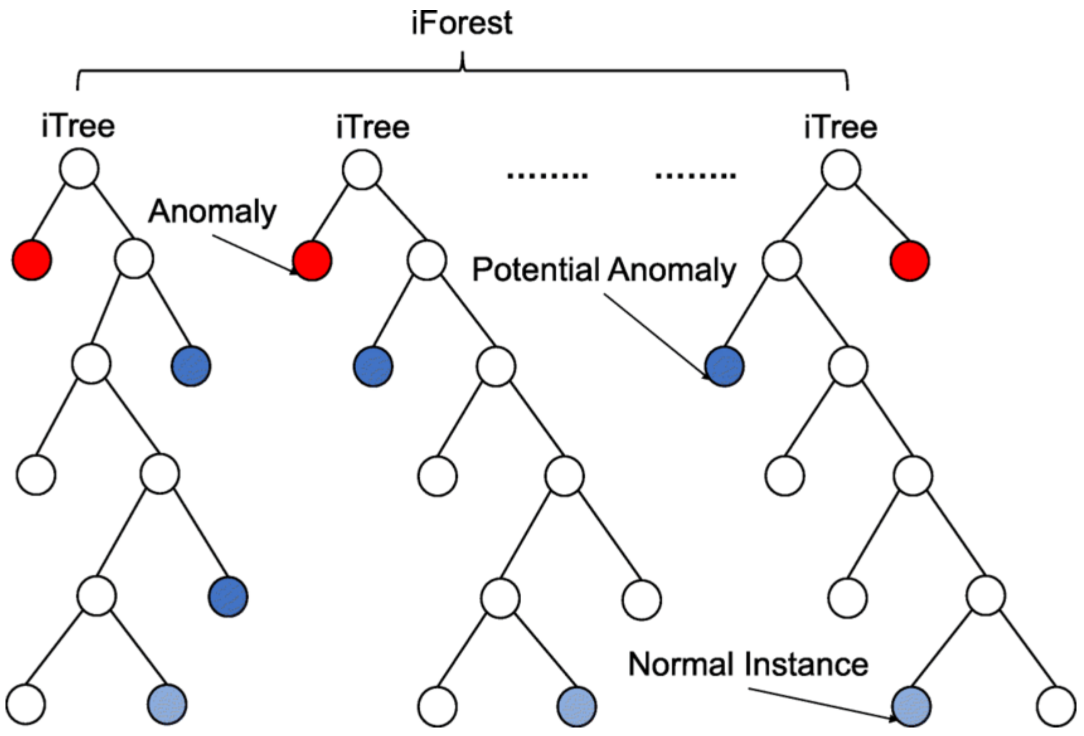
Project #4 – Anomaly detection
Isolation Forest

- **Isolation Forest** creates decision trees by randomly selecting a feature and a split value between the maximum and minimum values of the selected feature
- Random partitioning produces noticeably shorter paths for anomalies

Easy to isolate an anomalous point

Hard to isolate a normal point





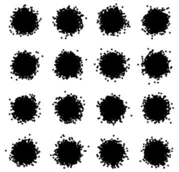
[1] https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.IsolationForest.html

Project #4 – Anomaly detection
Assignment

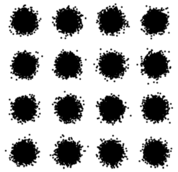
- **Brief summary**:
  - Given constellation image as input, decide if corresponding transponder is faulty → anomaly detection
  - Use one-class SVM and Isolation forest

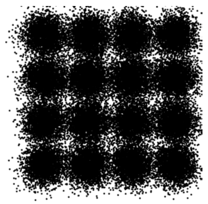
- Compare detection accuracy of one-class SVM and Isolation forest
- Do we need the whole constellation or a single symbol can be enough to detect the anomaly?
  - Compare accuracy and model complexity with two approaches

Train on normal



Test on normal and faulty

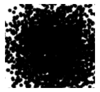




vs.

Train on normal

Test on normal and faulty



- Hint: stack 2D images into 1D vectors