Case Study - AutoML for Robust Anomaly Detection

Simon Kluettermann

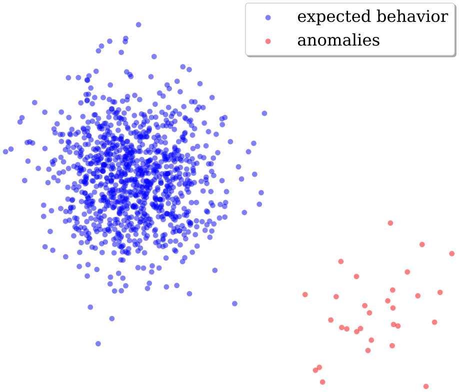
ls9 tu Dortmund

14. September 2022

Simon Kluettermann

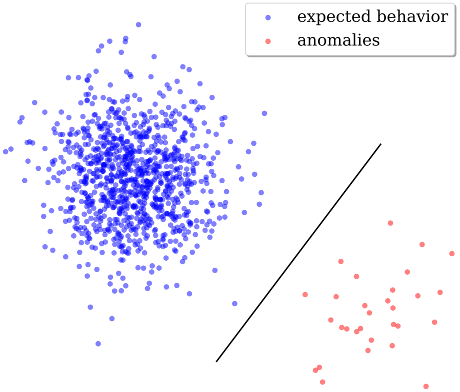
Anomaly Detection

- Two distributions
  - One known (=normal)
  - One unknown (=anomalies)
- Seperate them



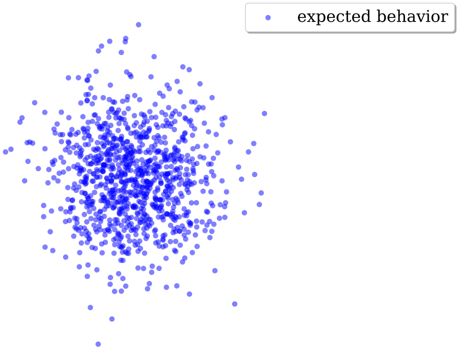
Anomaly Detection

- Two distributions
  - One known (=normal)
  - One unknown (=anomalies)
- Seperate them
- Problem: few anomalies



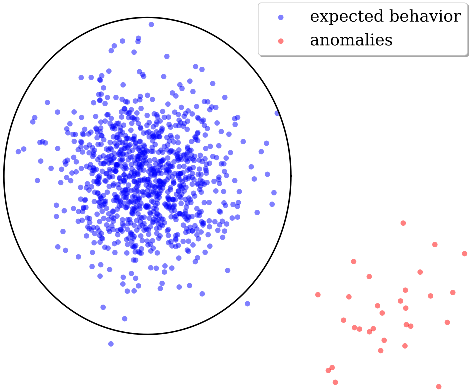
Anomaly Detection

- Anomalies are rare, so often only a few datapoints known (e.g. Machine Failure in an Aircraft)
- In practice, anomalies might appear that are not known during testing
- $\Rightarrow$So train the model only on normal samples
- Unsupervised Machine Learning
  - What can we say without knowing anomalies?
  - "Understand you dataset"



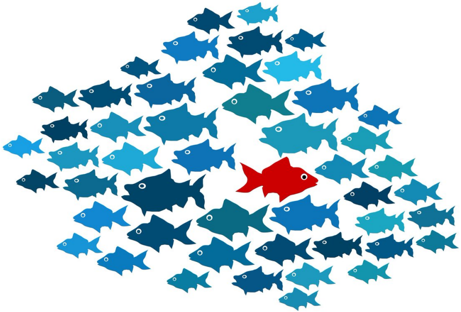
Anomaly Detection

- Anomalies are rare, so often only a few datapoints known (e.g. Machine Failure in an Aircraft)
- In practice, anomalies might appear that are not known during testing
- ⇒So train the model only on normal samples
- Unsupervised Machine Learning
  - What can we say without knowing anomalies?
  - "Understand you dataset"



Anomaly Detection

- Seems easy? Now do this
  - in thousands of dimensions
  - with complicated distributions
  - and overlap between anomalies and normal points



AutoML

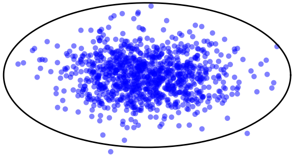
- Most machine learning requires Hyperparameter Optimisation
- (Find model parameters that result in the best results)
- ⇒AutoML: Do this automatically as fast as possible

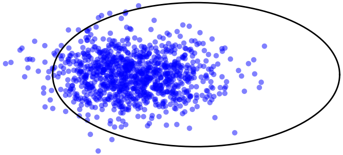
AutoAD

- So lets combine both (Auto Anomaly Detection)
- $\Rightarrow$Problem
  - AutoMl requires Evaluation (loss, accuracy, AUC) to optimize
  - AD can only be evaluated with regards to the anomalies
  - $\Rightarrow$no longer unsupervised
- So most Anomaly Detection is "unoptimized"

Solution 1 Metrics

- So how to solve this?
- One option: Think of some function to evaluate only the normal points
- $\Rightarrow$A bit hard to do in a case study



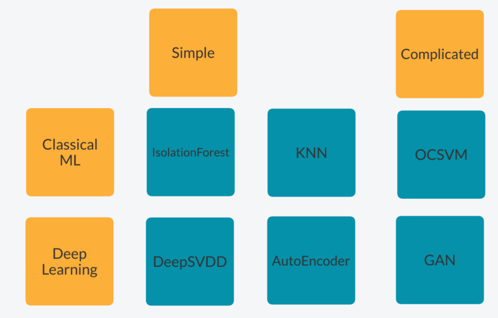


Solution 2 OneShot Learning

- So how to solve this?
- One option: "Just find the best solution directly"
- $\Rightarrow$Zero Shot AutoML
- Find best practices for hyperparameters
- Requires optimisation for each model seperately $\Rightarrow$ matches the case study structure quite well!

Course

- Basics of Scientific Computing
- Basics of AD
- Basics of AutoML
- Build groups for each algorithm
  - Choose a set of Hyperparameters
  - Find "best practice's" for them
  - Maybe consider more complicated Transformations (Preprocessing, Ensemble)
- Compare between groups (best algorithm for current situation)
- Evaluate on new datasets
- Write a report/Present your work



Questions

- Requirements:
  - MD Req 1$\Rightarrow$MD Req 8
  - Basic Python/Math Knowledge
  - Motivation to learn something new;)