Overview of EPF activities

James Catmore

Machine learning in experimental particle physics

- Under the name “multi-variate analyses” we have been using ML in high energy physics for decades, e.g. Boosted Decision Trees and shallow neural networks

- Opportunities to make use of the more sophisticated tools/techniques developed elsewhere

- Applications, current and future

  - Trigger / data acquisition

  - Distributed computing

  - Reconstruction (potentially using image-recognition for whole-event reconstruction)

  - Monitoring and anomaly detection

  - Physics analysis

Activities in Oslo EPF

- Distributed computing
  - Optimising/automating computing operations (currently labour-intensive), using the analytics platform as a source of training data, and techniques such as supervised and unsupervised anomaly detection
- Physics analysis
  - Applying new techniques and software to the task of signal/background separation
  - Model independent searches
  - One-class (background-only) training
- Data quality monitoring
  - Automating the comparison between two datasets which are supposed to be the same at a statistical level (reference and test): “anomaly detection”
- Two examples today from Eirik and myself
  - Me: one-class classification/anomaly detection
  - Eirik: studying the importance of different variables in a neural network classifier

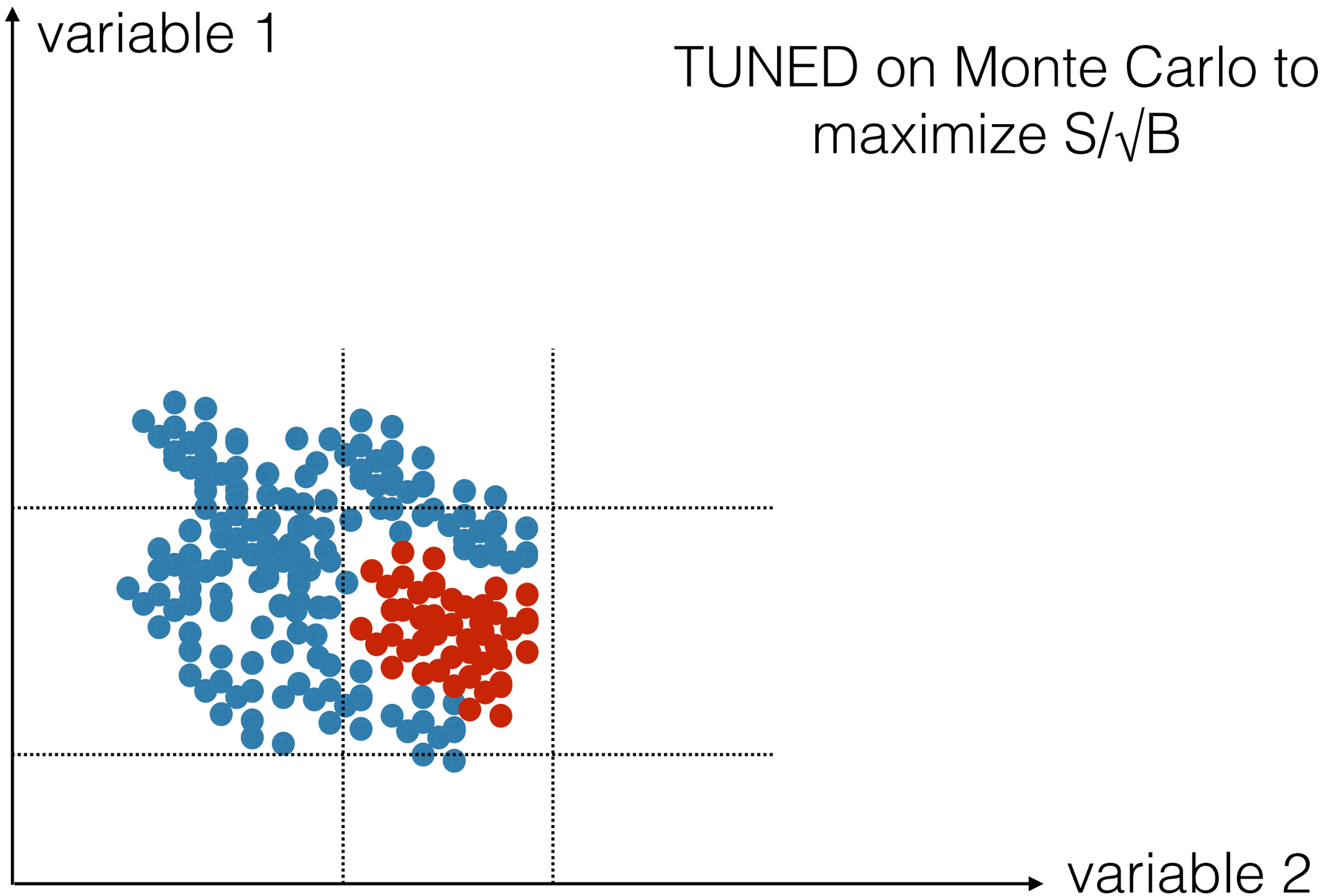
An very simple example

- In general we have a signal that we want to separate from the background
  - It might be something that we know about, like the Z or the Higgs
  - Or it might be something like Z' that we suspect might be there, so we want to suppress the background
- The key question is: how to we minimise the background whilst keeping as much of the signal as possible?

The traditional way: rectangular cuts

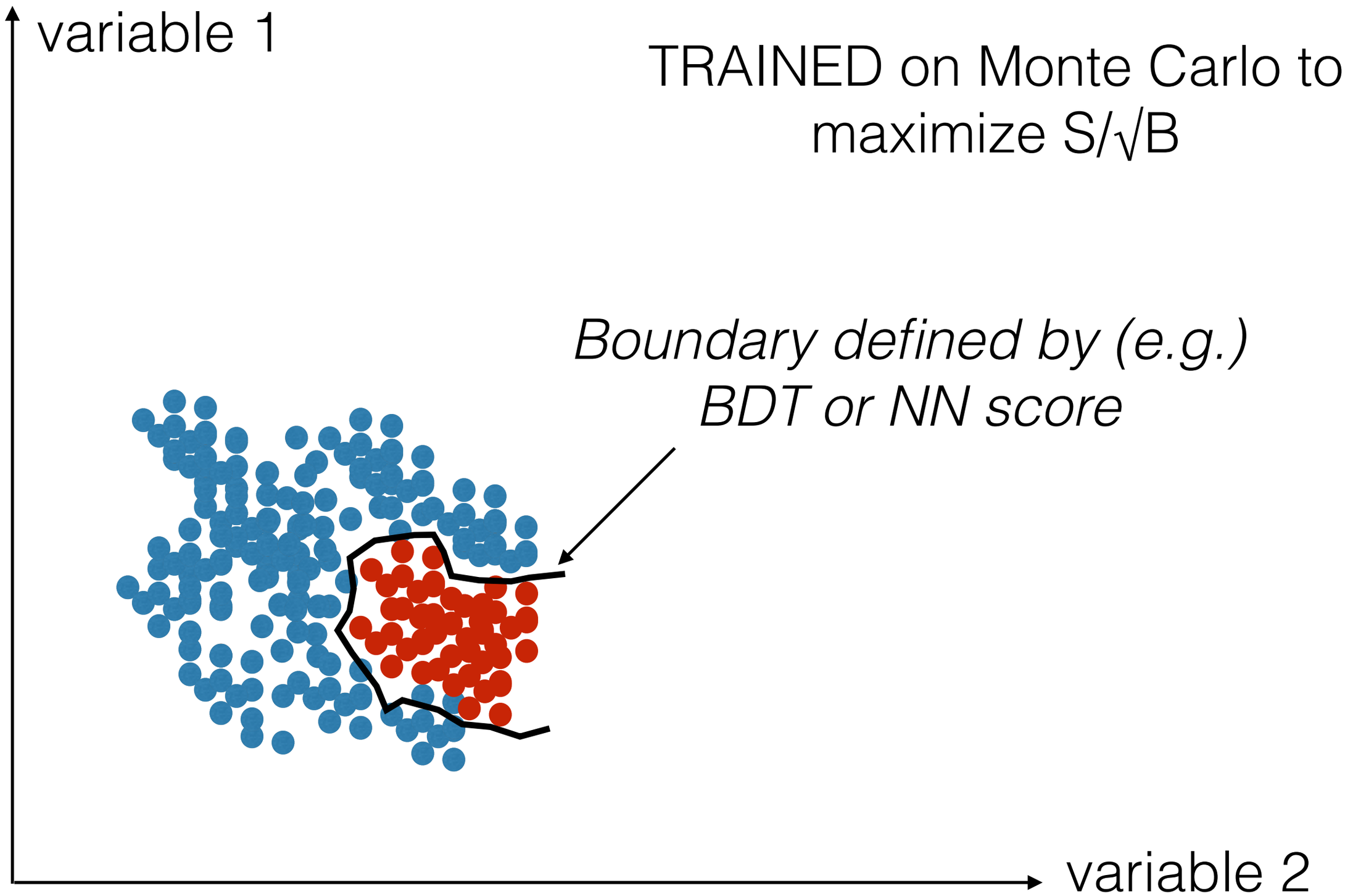
		ATLAS	CMS
Z’	ee	2e E_T > 30 GeV Primary vertex Isolated No opposite charge requirement	2e E_T > 35 GeV $\|\eta_{det}\|$ < 1.4442 (barrel) 1.566 < $\|\eta_{det}\|$ < 2.5 (endcap) At least one in the barrel Isolated No opposite charge requirement
	μμ	2μ p_T > 30 GeV Primary vertex Isolated Opposite charge requirement	2μ > 53 GeV, $\|\eta\|$ < 2.4 Isolated Common vertex fit χ^2/dof < 20 Opposite charge requirement

The traditional way: rectangular cuts



TUNED on Monte Carlo to maximize $S/\sqrt{B}$

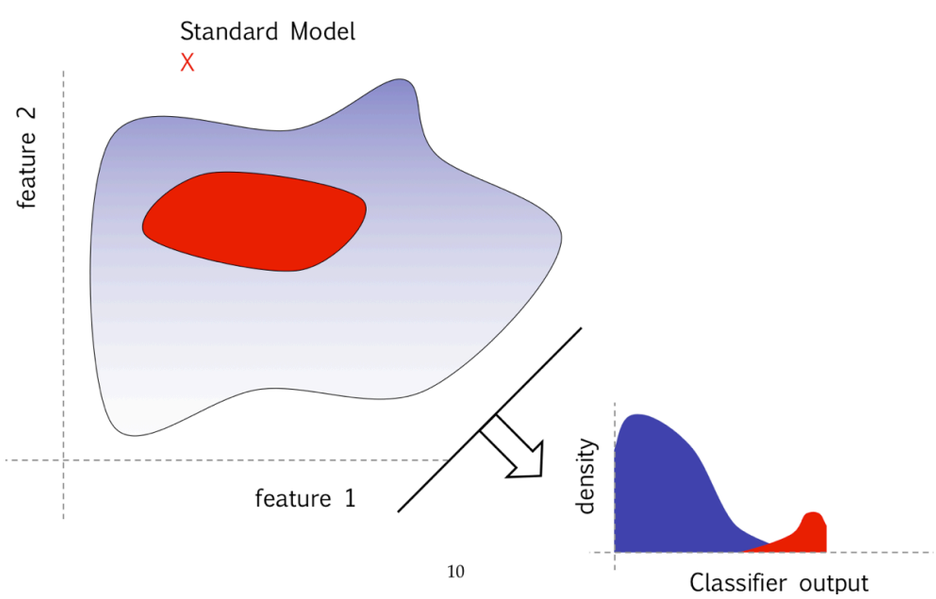
Multi-variate analysis



Function

Find a function

$$f(\mathcal{X}^{N} \rightarrow \mathcal{R}^{1})$$



Neural networks can learn these shapes in high-dim and summarize in a 1D output

$$\frac{L_{SM+X}}{L_{SM}} = \frac{P(\text{data} \mid SM+X)}{P(\text{data} \mid SM)}$$

And require a histogram in only one dimension

Measuring the performance

With a binary classifier there are four possible outcomes:

True Positive (TP)

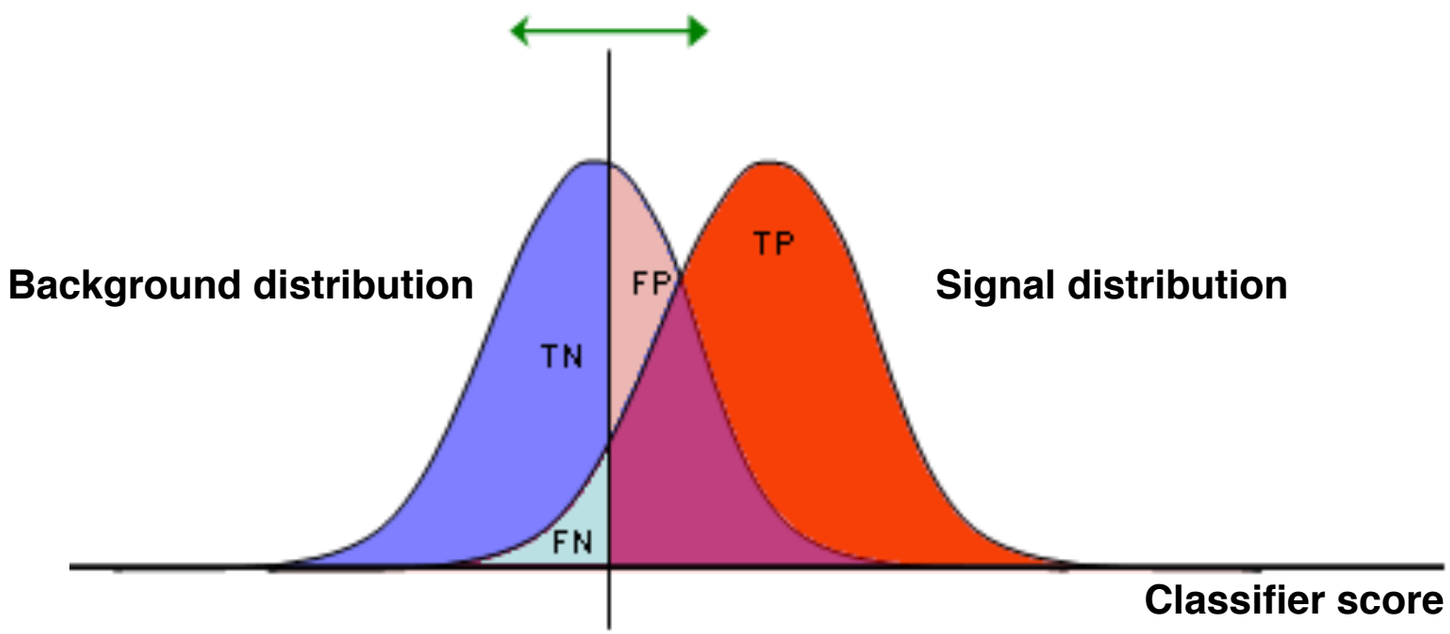
False Negative (FN)

Signal identified as signal	Signal identified as background
Background identified as signal	Background identified as background

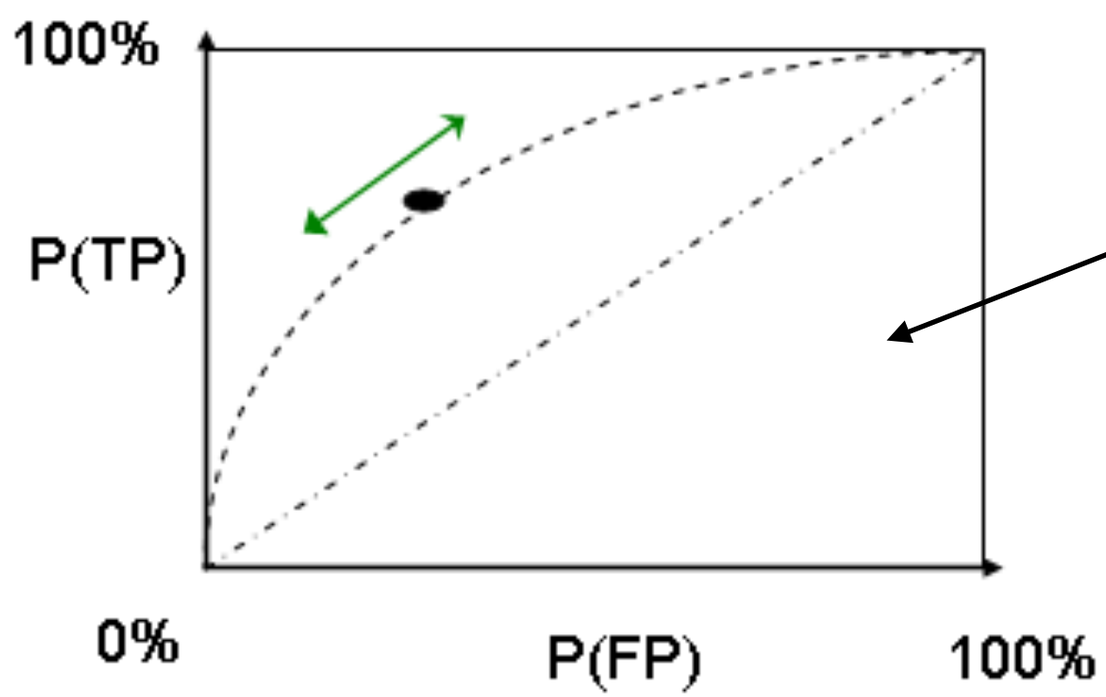
False Positive (FP)

True Negative (TN)

Measuring the performance - receiver operating characteristic (ROC)



TP	FP
FN	TN
1	1



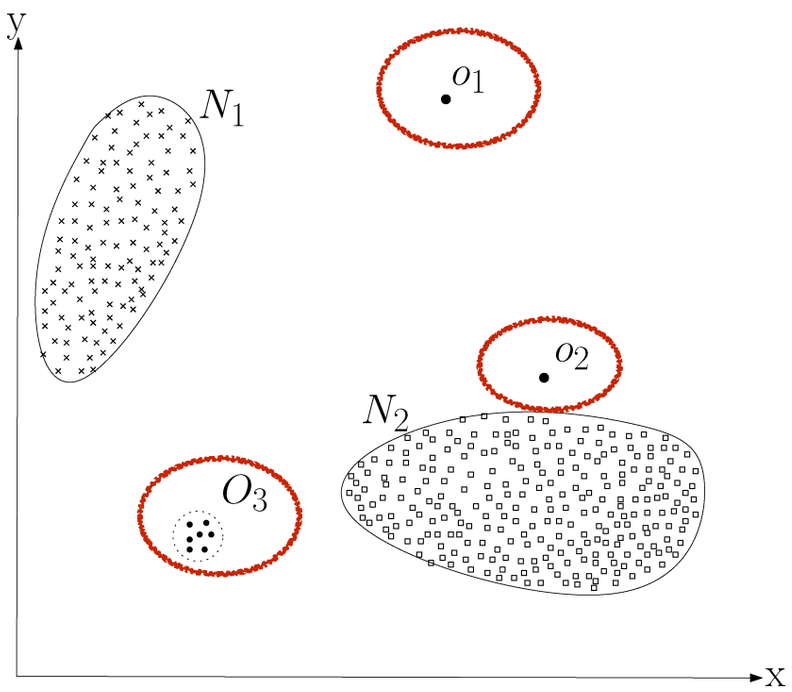
Area Under Curve
(AUC) gives overall performance measure
0.5 = no better than a random guess
1.0 = perfect classifier

Anomaly detection / one class classification

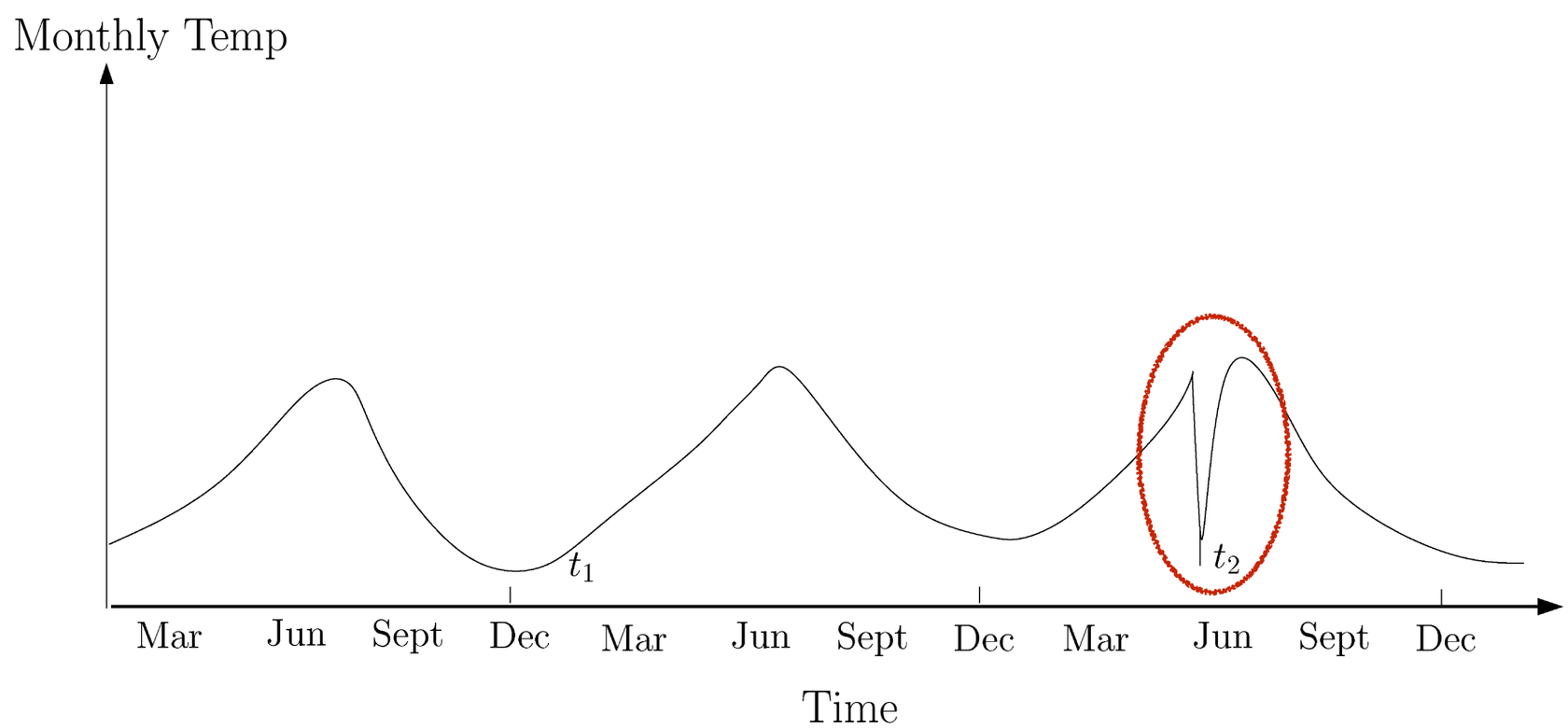
- Automatic identification of data instances (events) that are in some way different from the bulk of the data and which need detailed scrutiny by experts
  - Usually implied: fewer anomalies than normal instances
- Can be
  - supervised: train to recognise specific anomalous cases
  - semi-supervised: train only on the bulk of the data without anomalies
  - unsupervised: algorithm automatically identifies the bulk by some means and thence the anomalies
- Difficult problem because in general we don't know what the anomalies look like, and there may be very few of them
  - Testing is particularly challenging: how do we evaluate the performance of an algorithm on a type of event that we have never seen before?

Anomaly detection

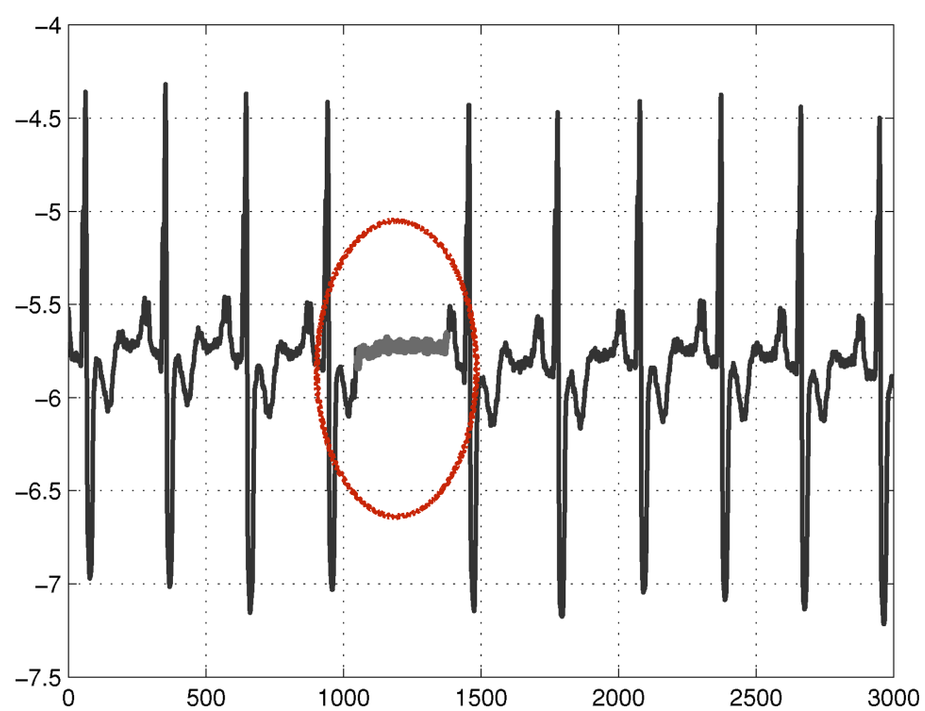
POINT



CONTEXTUAL



COLLECTIVE

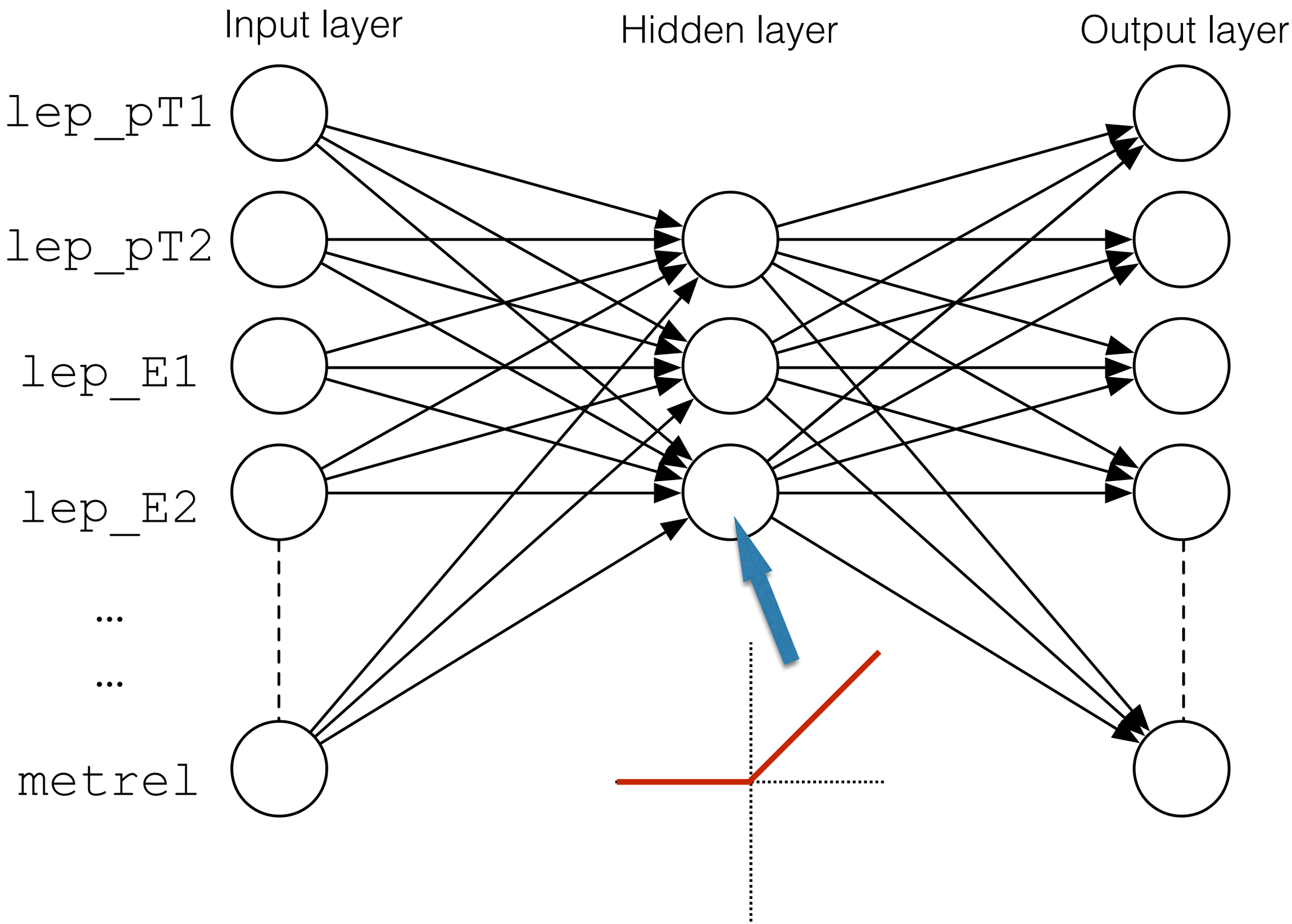


Images:
ACM Computing Surveys,
Vol 41, No. 3, Article 15

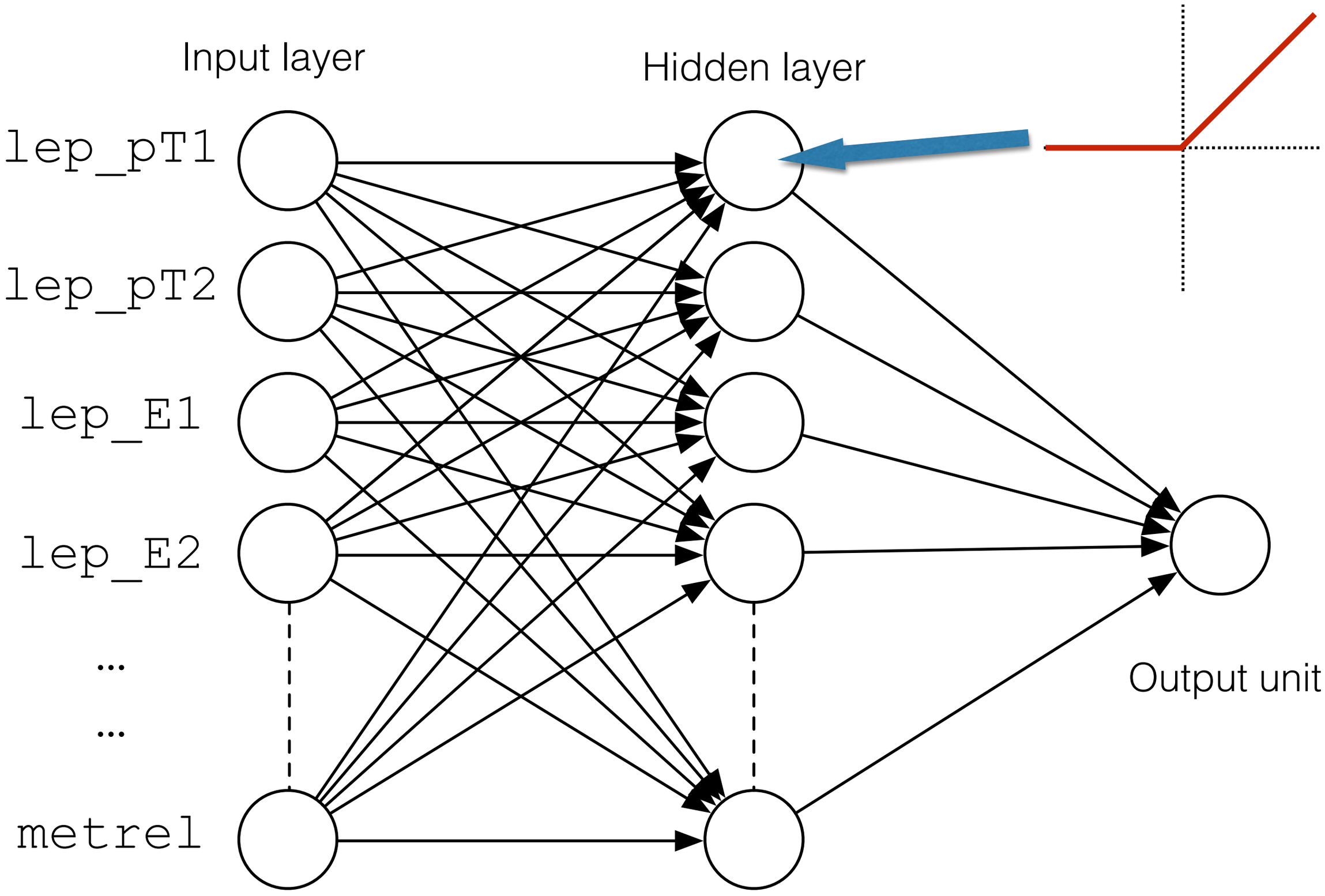
Techniques

- Density methods
  - Data points in areas of high density are likely to be “normal”, those in low density “anomalous”
    - Examples: kNN, minimum spanning trees, Parzen windows
- Boundary methods
  - Algorithm defines a boundary around the “normal” data and anything lying outside the boundary is “anomalous”
    - Example: one-class support vector machines
- Reconstruction methods
  - Algorithms trained to rebuild “normal” data will do badly on “anomalous” events, leading to a high reconstruction error
    - Example: auto-encoder

Example: auto-encoder



Counter-example: classifier



Reconstruction error definition (auto-encoder only)

Reconstruction error per event =

$$\sum_{i=1}^{N} (x_i^{in} - x_i^{out})^2$$

Sum is over the variables, e.g. N=36 in this case

Software overview

- https://github.com/jcatmore/susy-neuralnetworks