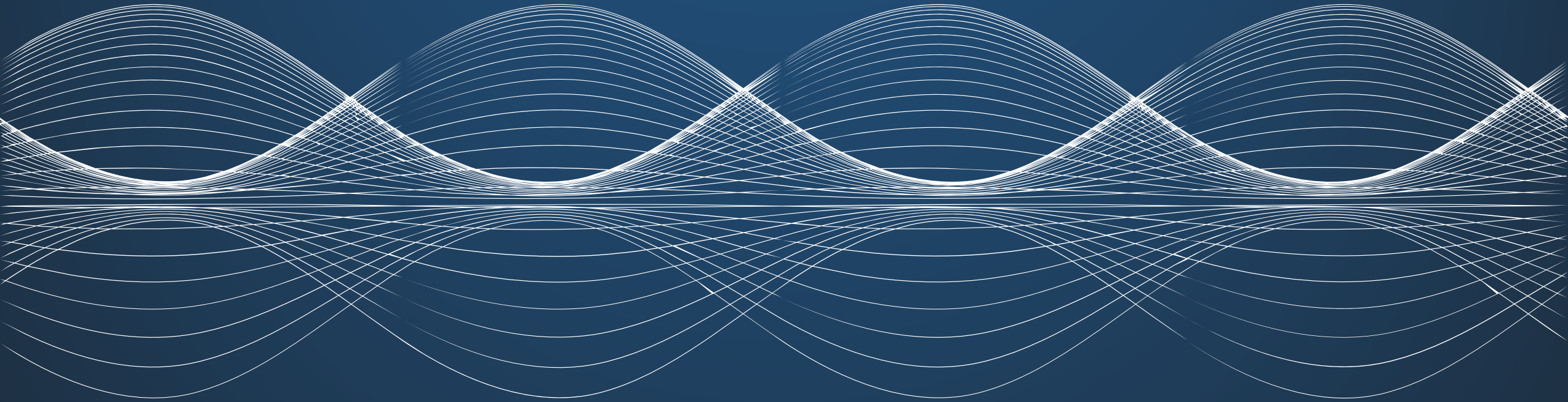


Anomaly Detection Part II

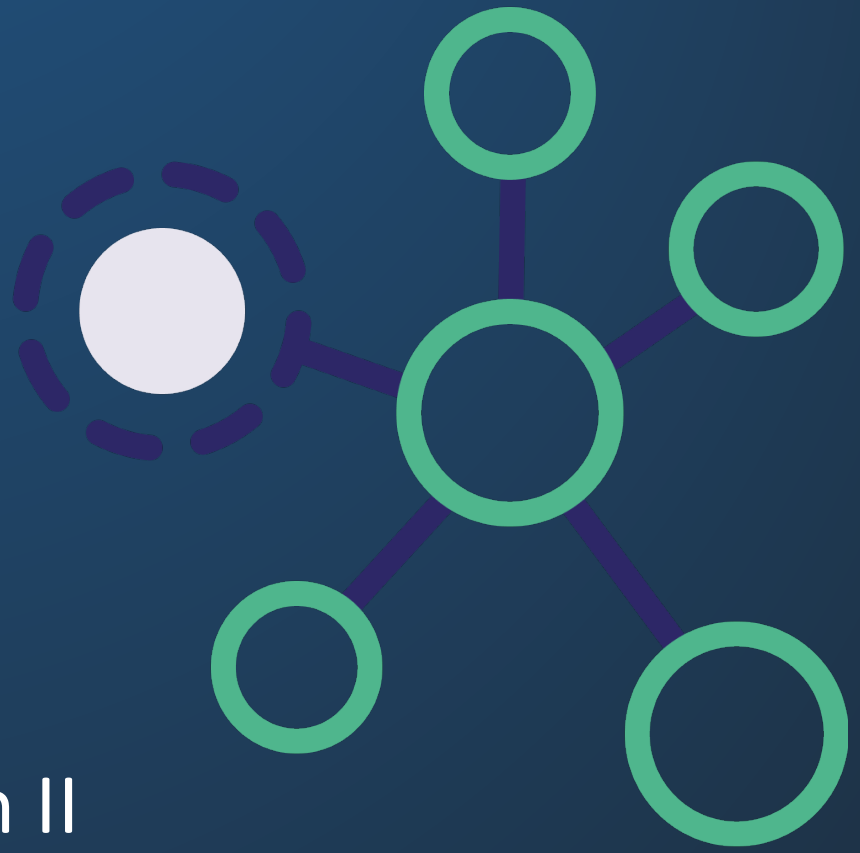
Advanced Research Topics – 7PAM2016

Dr Vanessa Graber (based on slides by Dr William Alston)

Quick recap

Techniques for anomaly detection II

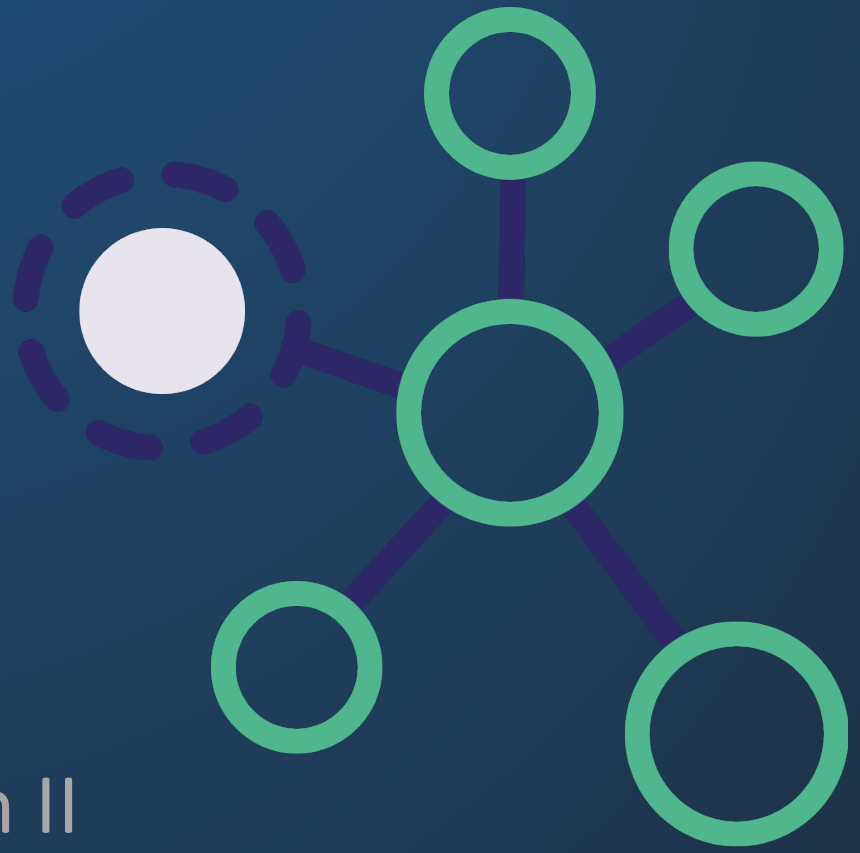
Summary



Quick recap

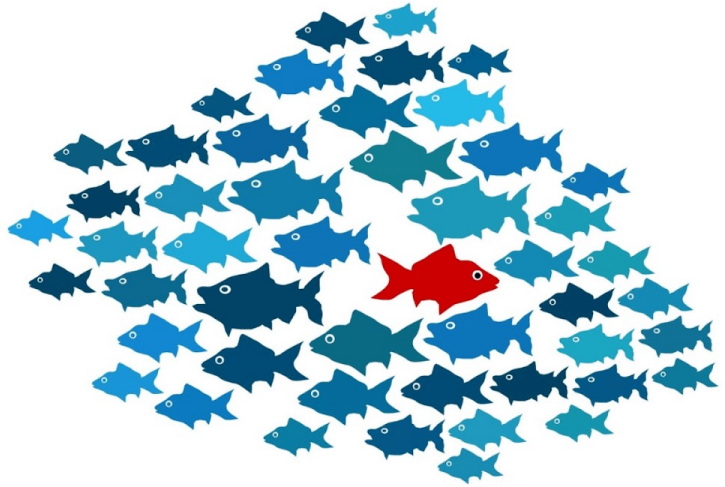
Techniques for anomaly detection II

Summary



Learning outcomes

After the two lectures, you will:



- Understand anomaly detection problems.
- Understand the methods used for anomaly detection.
- Be able to identify, which algorithm to use for a particular problem set.
- Be able to implement this approach in Python.

Reading materials

Two PDFs can be found on Canvas

Anomaly Detection: A Survey

VARUN CHANDOLA, ARINDAM BANERJEE, and VIPIN KUMAR

University of Minnesota

15

Anomaly detection is an important problem that has been researched within diverse research areas and application domains. Many anomaly detection techniques have been specifically developed for certain application domains, while others are more generic. This survey tries to provide a structured and comprehensive overview of the research on anomaly detection. We have grouped existing techniques into different categories based on the underlying approach adopted by each technique. For each category we have identified key assumptions, which are used by the techniques to differentiate between normal and anomalous behavior. When applying a given technique to a particular domain, these assumptions can be used as guidelines to assess the effectiveness of the technique in that domain. For each category, we provide a basic anomaly detection technique, and then show how the different existing techniques in that category are variants of the basic technique. This template provides an easier and more succinct understanding of the techniques belonging to each category. Further, for each category, we identify the advantages and disadvantages of the techniques in that category. We also provide a discussion on the computational complexity of the techniques since it is an important issue in real application domains. We hope that this survey will provide a better understanding of the different directions in which research has been done on this topic, and how techniques developed in one area can be applied in domains for which they were not intended to begin with.

Revisiting Time Series Outlier Detection: Definitions and Benchmarks

Kwei-Herng Lai
Rice University
khlai@rice.edu

Daochen Zha
Rice University
daochen.zha@rice.edu

Junjie Xu
Penn State University
jmx5097@psu.edu

Yue Zhao
Carnegie Mellon University
zhaoy@cmu.edu

Guanchu Wang
Rice University
hegsns@rice.edu

Xia Hu
Rice University
xiahu@rice.edu

Abstract

Time series outlier detection has been extensively studied with many advanced algorithms proposed in the past decade. Despite these efforts, very few studies have investigated how we should benchmark the existing algorithms. In particular, using synthetic datasets for evaluation has become a common practice in the literature, and thus it is crucial to have a general synthetic criterion to benchmark algorithms. This is a non-trivial task because the existing synthetic methods are very different in different applications and the outlier definitions are often ambiguous. To bridge this gap, we propose a behavior-driven taxonomy for time series outliers and categorize outliers into point- and pattern-wise outliers with clear context definitions. Following the new taxonomy, we then present a general synthetic criterion and generate 35 synthetic datasets accordingly. We further identify 4 multivariate real-world datasets from different domains and benchmark 9 algorithms on the synthetic and the real-world datasets. Surprisingly, we observe that some classical algorithms could outperform many recent deep learning approaches. The datasets, preprocessing and synthetic scripts, and the algorithm implementations are made publicly available at https://github.com/datamllab/tods/tree/benchmark.

Anomalies

And what they mean

- “We are drowning in information, while starving for wisdom.” Consi-lience: The Unity of Knowledge (1998), biologist E. O. Wilson
- Anomalous events occur relatively infrequently.
- However, when they do occur, their **consequences** can be **dramatic** and often **negative**.

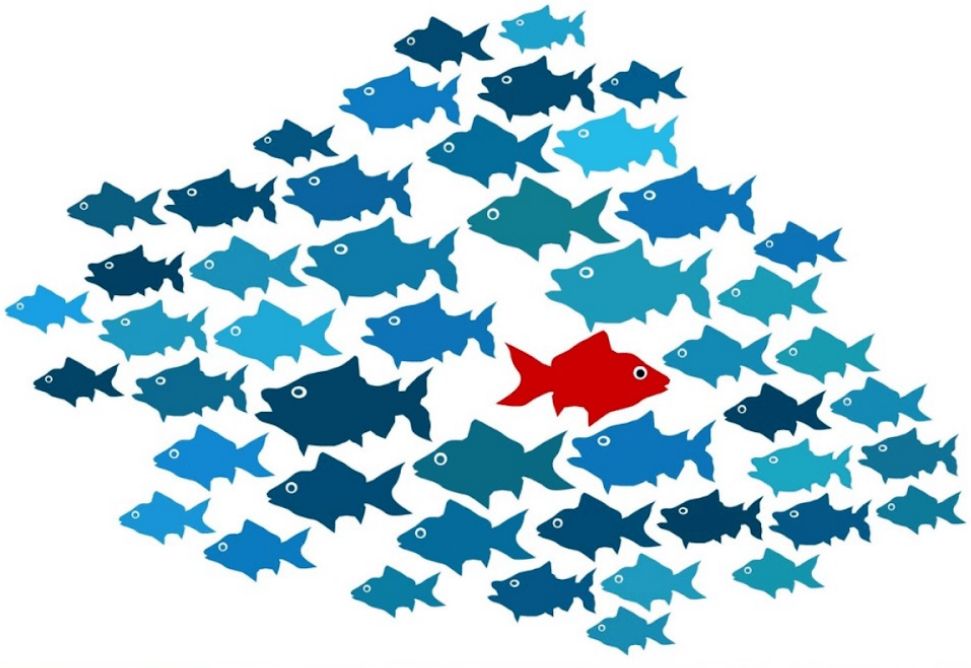
Anomalies can be like needles in a haystack.

Anomaly detection

"The odd one out"

- An anomaly is a pattern in the data that does **not conform with the expected behaviour**. They are also referred to as outliers, exceptions, peculiarities, surprises, novelties, incongruences, etc.

- Historically, the **field of statistics** dealt with anomalies to find and remove outliers to improve analyses. There are now many fields, where **anomalies are the topic** of greatest **interest**.

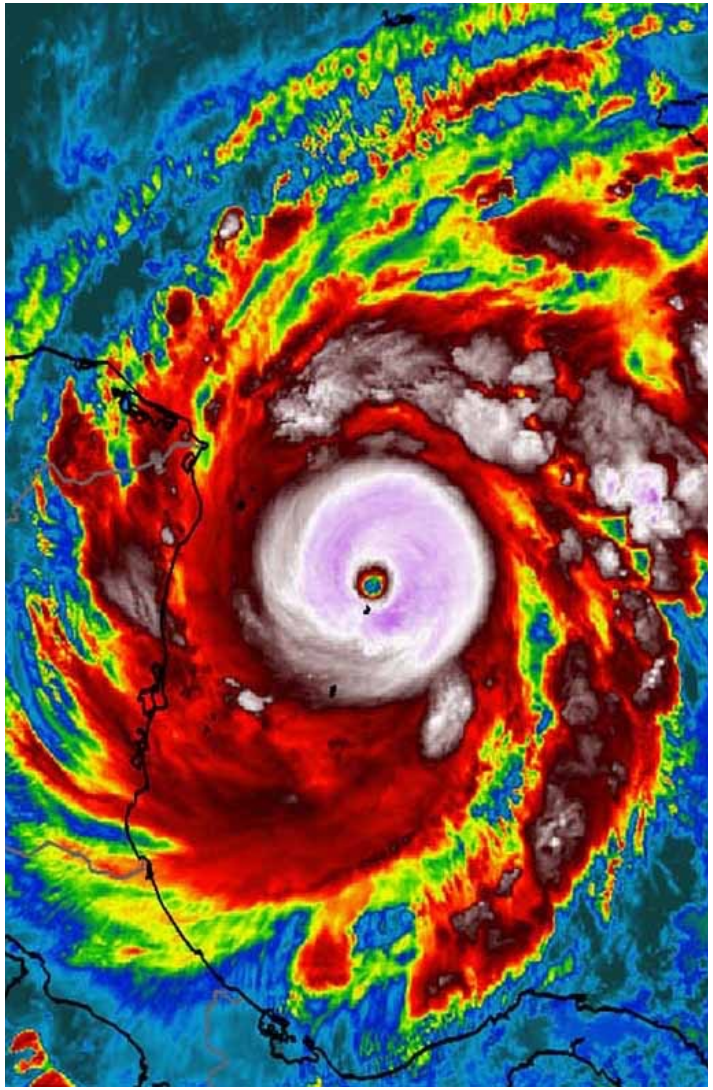


To detect them, we need to identify objects, events, etc. that are different from most other objects, events, etc.

Causes of anomalies

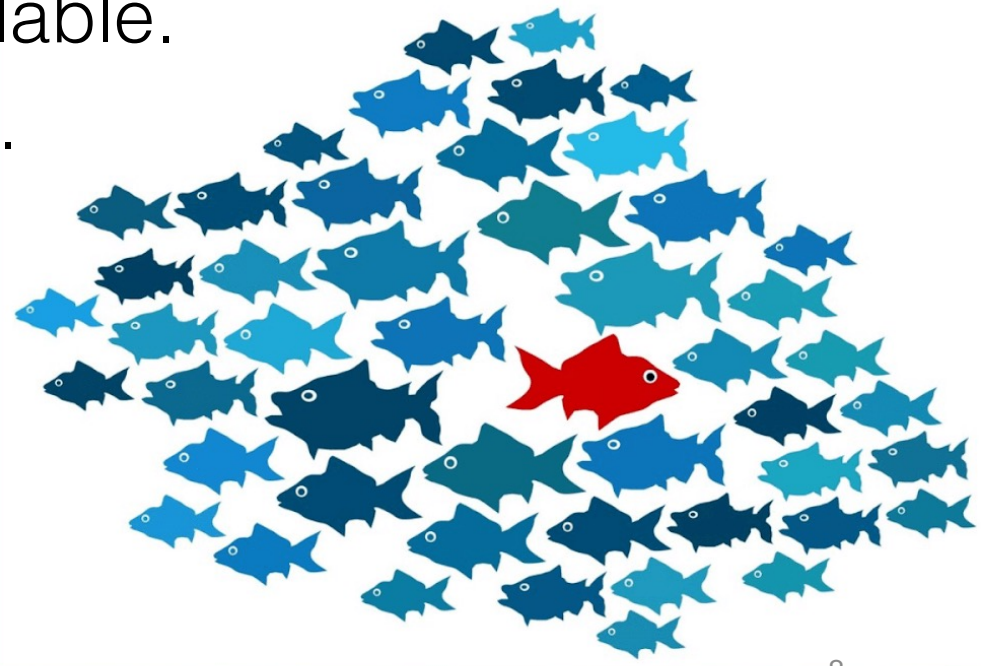
Several primary causes

- Some data elements are part of a different class of objects or produced by a different underlying mechanism (e.g., disease vs. no disease or fraud vs. no fraud).
- Data elements can originate from the tails of an underlying Gaussian distribution.
- The underlying process has a natural variation (e.g., extreme weather events).
- The underlying data was measured, and collection errors were made (e.g., human, equipment).



Key challenges in anomaly detection

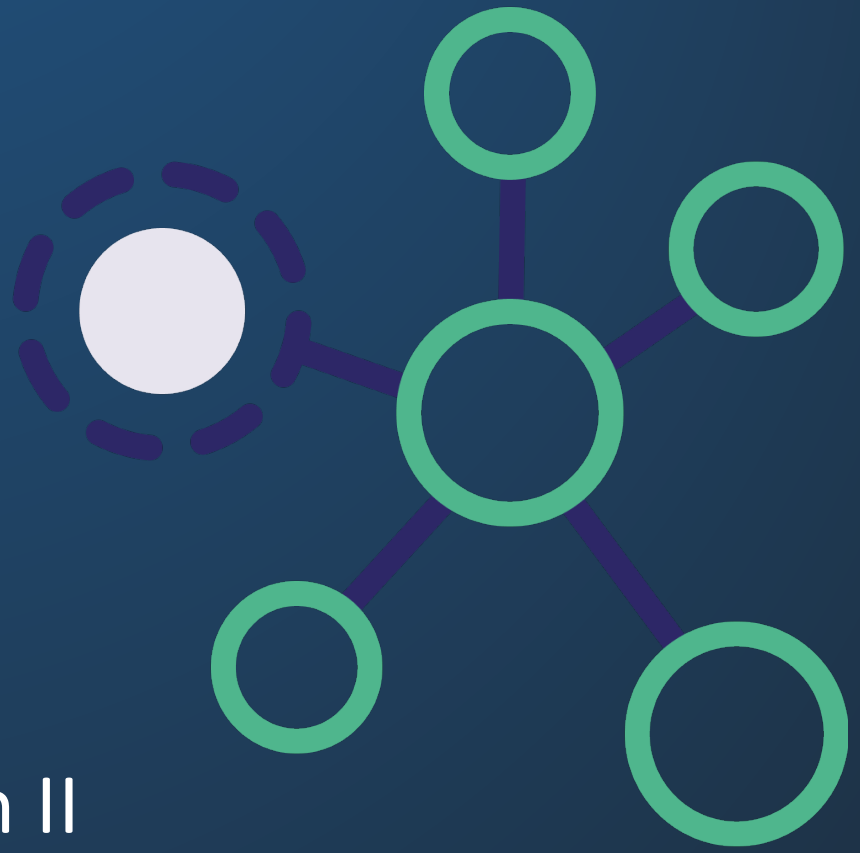
- Defining a representative normal region is challenging.
- Boundary between normal & outlying behaviour is often imprecise.
- Exact notion of an outlier varies between application domains.
- Data labels for training/validation unavailable.
- Malicious adversaries are unpredictable.
- Normal behaviour evolves with time.
- Data might contain noise.
- Selection of relevant features is difficult.



Quick recap

Techniques for anomaly detection II

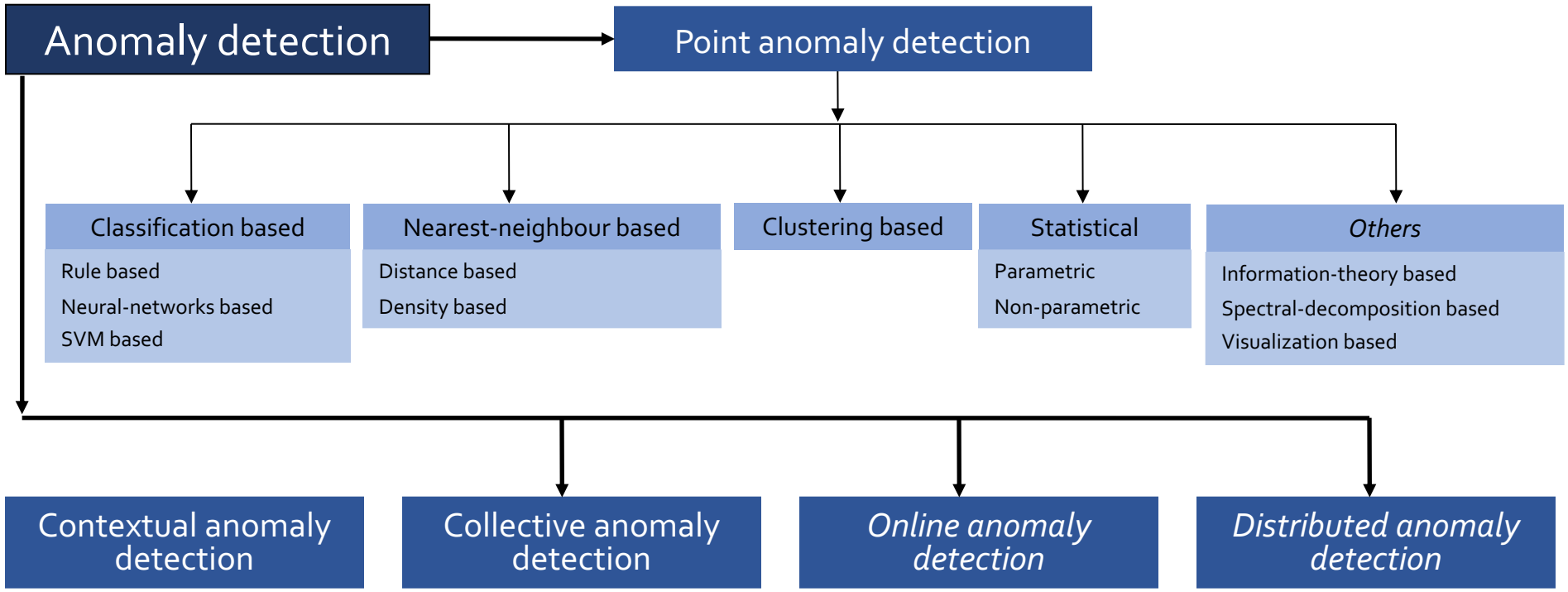
Summary



Taxonomy of techniques

An overview of anomaly detection approaches

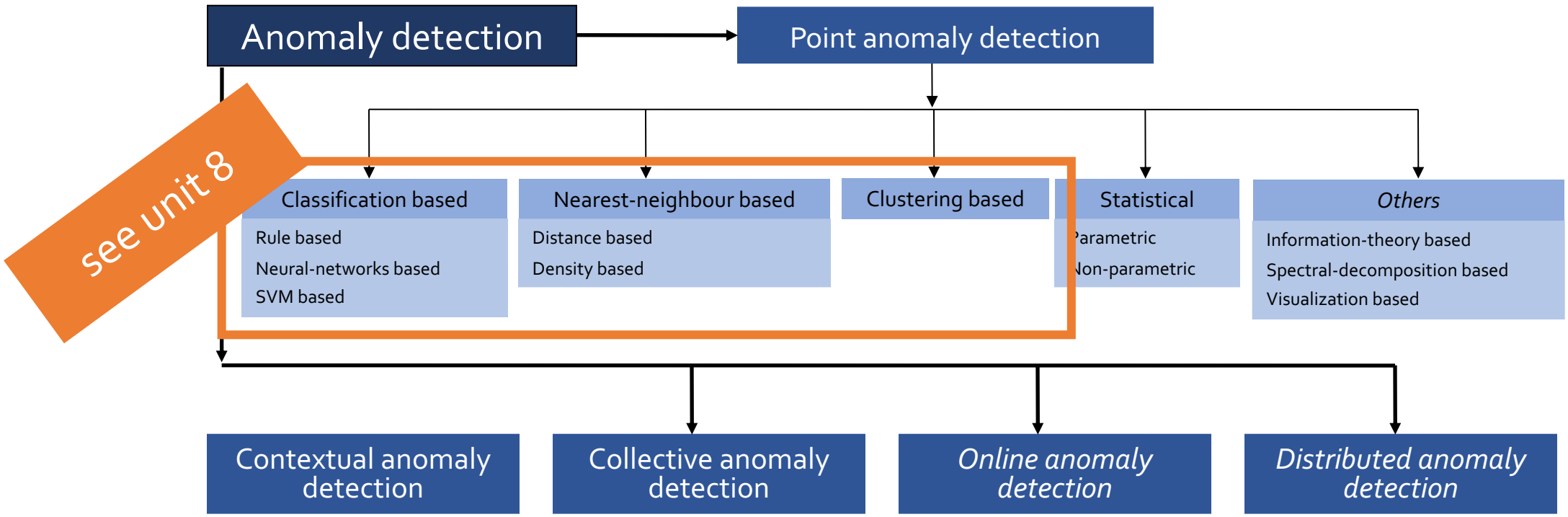
- In the remainder of this class, we will look at several more of the **methods of anomaly detection** (following Chandola et al. (2008) – Anomaly Detection: A Survey; see reading materials):



Taxonomy of techniques

An overview of anomaly detection approaches

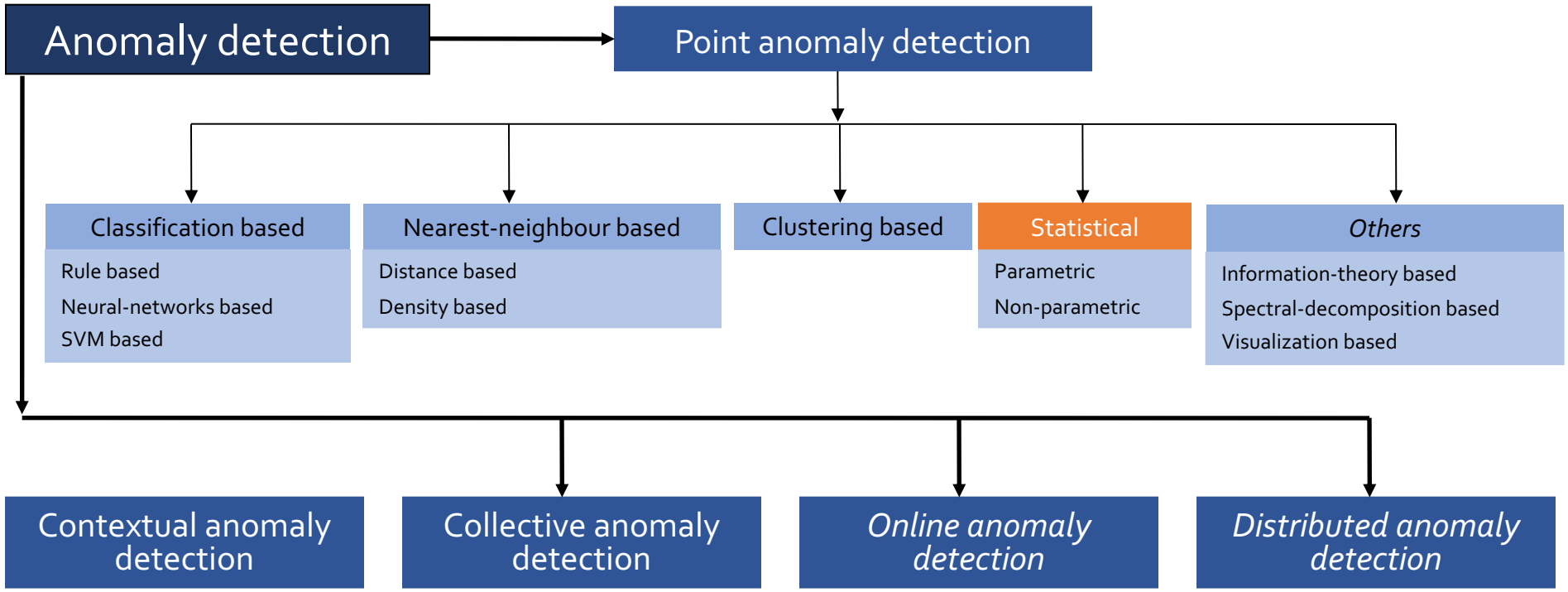
- In the remainder of this class, we will look at several more of the **methods of anomaly detection** (following Chandola et al. (2008) – Anomaly Detection: A Survey; see reading materials):



Taxonomy of techniques

An overview of anomaly detection approaches

- In the remainder of this class, we will look at several more of the **methods of anomaly detection** (following Chandola et al. (2008) – Anomaly Detection: A Survey; see reading materials):



Statistical techniques

An overview

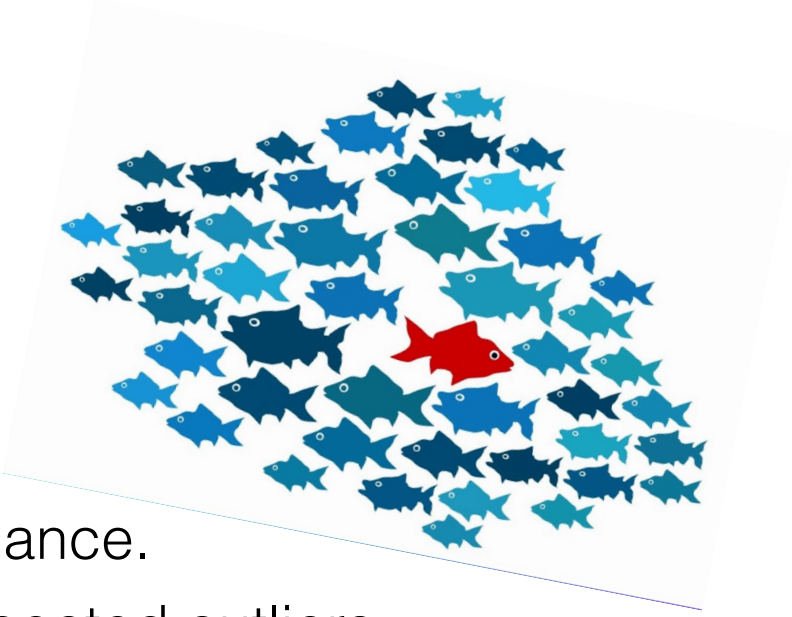
- The key assumption here is that **normal data instances** occur in **regions of high probability** of a statistical distribution, while anomalies occur in the low-probability regions.

- The **general approach** is as follows:
  - Estimate a statistical distribution for a given dataset.
  - Apply a statistical inference test to determine if a test instance belongs to this distribution or not.
  - If an observation is more than 3 standard deviations away from the sample mean, we call it an anomaly.

Anomalies have large values for a given test statistic.

Statistical techniques

An overview



- Applying a statistical test depends on
  - the properties of the test instance.
  - the parameters of the model such as the mean or the variance.
  - the confidence limit, which is related to the number of expected outliers.

- We can distinguish two types of approaches:
  - **Parametric techniques:** We assume that the normal (and possibly anomalous) data are generated from an underlying parametric distribution. We then fit the distribution parameters from the training data.
  - **Non-parametric techniques:** We do not assume any knowledge of underlying distributions and use non-parametric techniques (e.g., histograms) to estimate the density of the distribution and its parameters.

Statistical techniques

Pros and cons

ADVANTAGES

- We can readily apply a range of existing statistical tools.
- Statistical tests are well-understood and well-validated.
- These tools allow a quantitative measure of the degree to which and object is an outlier.
- This provides a statistically justifiable solution to detect anomalies.

DISADVANTAGES

- Datasets might be hard to model parametrically (e.g., if they have multiple modes or varying density).
- The parametric assumptions might not actually hold.
- In high dimensions, we might have insufficient data to estimate the true underlying distribution, which complicates the construction of hypothesis tests.

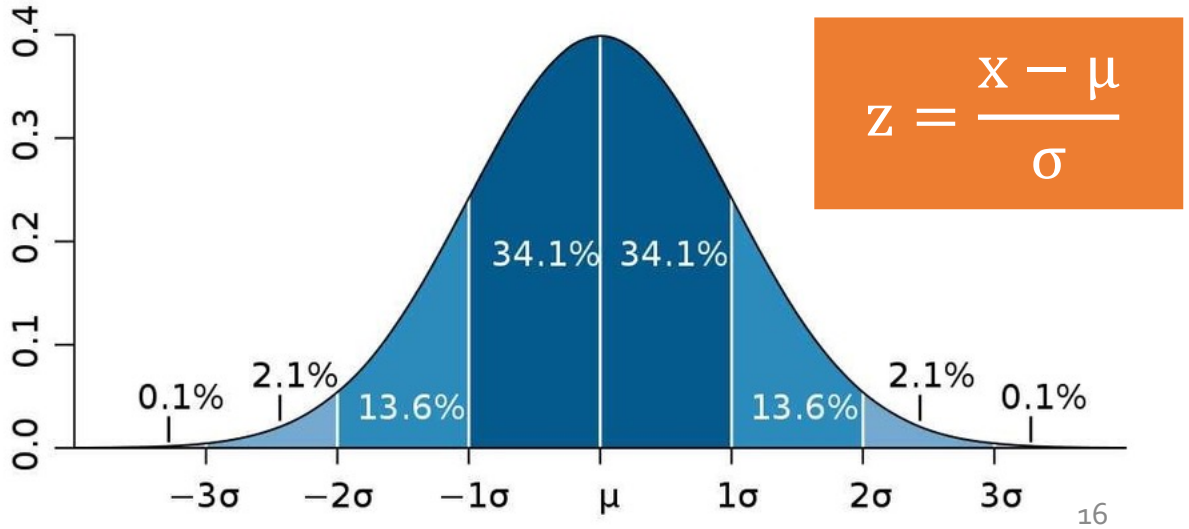
Statistical techniques

Parametric approaches: univariate Gaussians

- A common approach assumes that normal data follow a **Gaussian distribution**. Based on this, we then determine the "shape" of the data and define outliers as those which stand far away from the fit shape.

- For a **univariate Gaussian distribution**, outliers can be obtained by comparing the **z-score** (standard score) to a specific threshold:

z-scores denote the number of standard deviations, σ, with which certain values are above or below the mean, μ.



$$z = \frac{x - \mu}{\sigma}$$

Statistical techniques

Parametric approaches: Grubbs's method for univariate Gaussians

- Another method to detect outliers in univariate data that is assumed to come from a normally distributed population is the **Grubbs test**.

- It detects one outlier at a time. If a point is an outlier, it is removed from the dataset. The test is then **iterated until no outliers remain**. The test is defined for the **hypothesis** (for a dataset of size N):

$$G \equiv \frac{\max_{i=1,\ldots,N} |x_i - \mu|}{\sigma}$$

H_0: There are no outliers in the dataset.
H_1: There is exactly one outlier in the dataset.

Reject H_0 at level α if :

$$G > \frac{N-1}{\sqrt{N}} \sqrt{\frac{t^2_{\alpha/(2N),N-2}}{N-2+t^2_{\alpha/(2N),N-2}}}$$

Statistical techniques

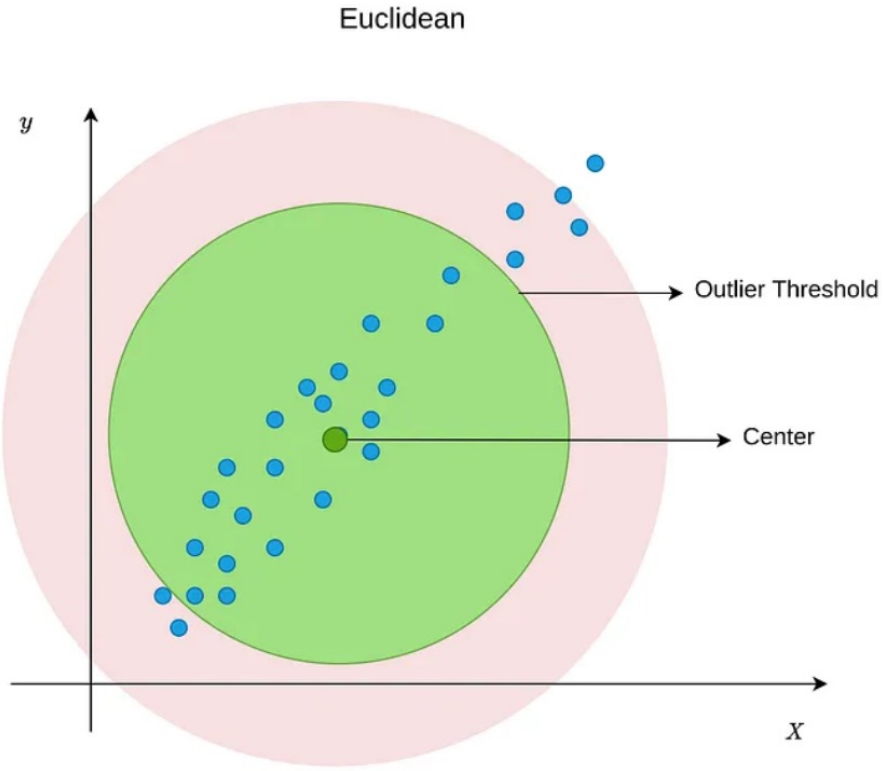
Parametric approaches: multi-variate Gaussians

- For multi-variate **Gaussian distributions**, we need to think carefully about what "far" away from a certain distribution shape means.

- For correlated features, calculating the **standard Euclidean distance** from a data point to the distribution's centre gives misleading results.

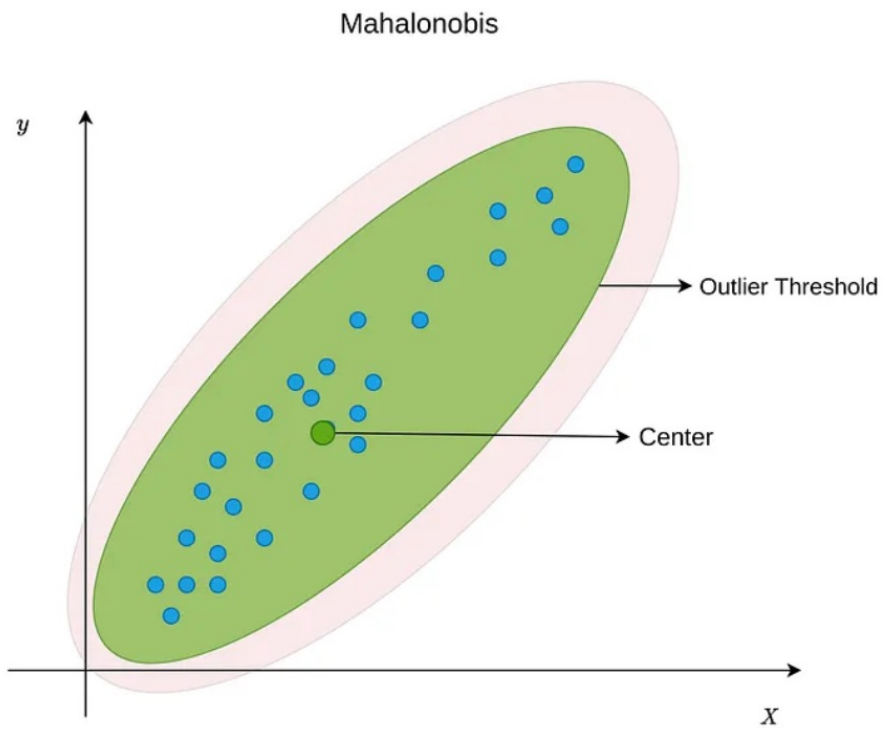
In 2D, we have:

$$d(p_1, p_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$



Statistical techniques

Parametric approaches: multi-variate Gaussians



This is the multi-variate generalisation of the z-score.

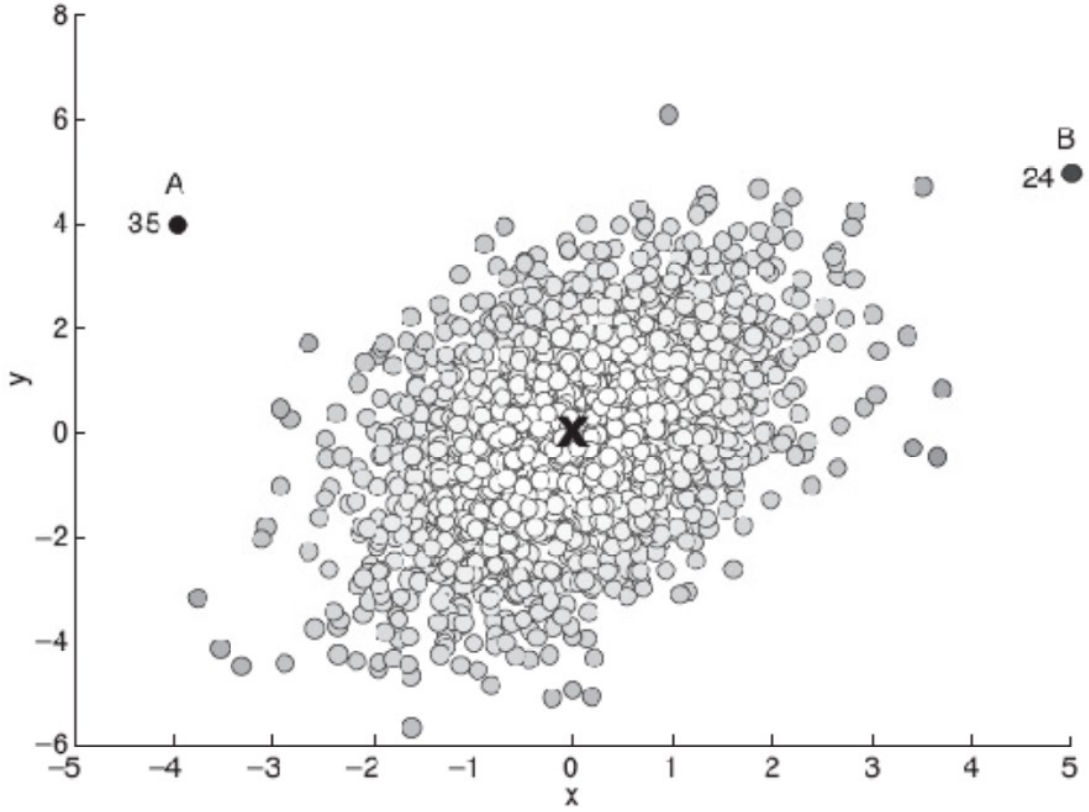
- We require a **“better” distance measure** that includes the **covariances** of our data.
- For a distribution Q, with mean $\vec{\mu}$ and covariance matrix, S, the so-called **Mahalanobis distance** is defined as

$$\mathrm{d}(\vec{x},\vec{y};Q)=\sqrt{(\vec{x}-\vec{y})^{\mathrm{T}}S^{-1}(\vec{x}-\vec{y})}$$

Statistical techniques

Mahalanobis distance

- Let's look at another example:



	Distance	
	Euclidean	Mahalanobis
A	5.7	35
B	7.1	24

As the data are not spherically distributed, the Euclidean distance would identify point B as an outlier. However, point A is less compatible with the data points.

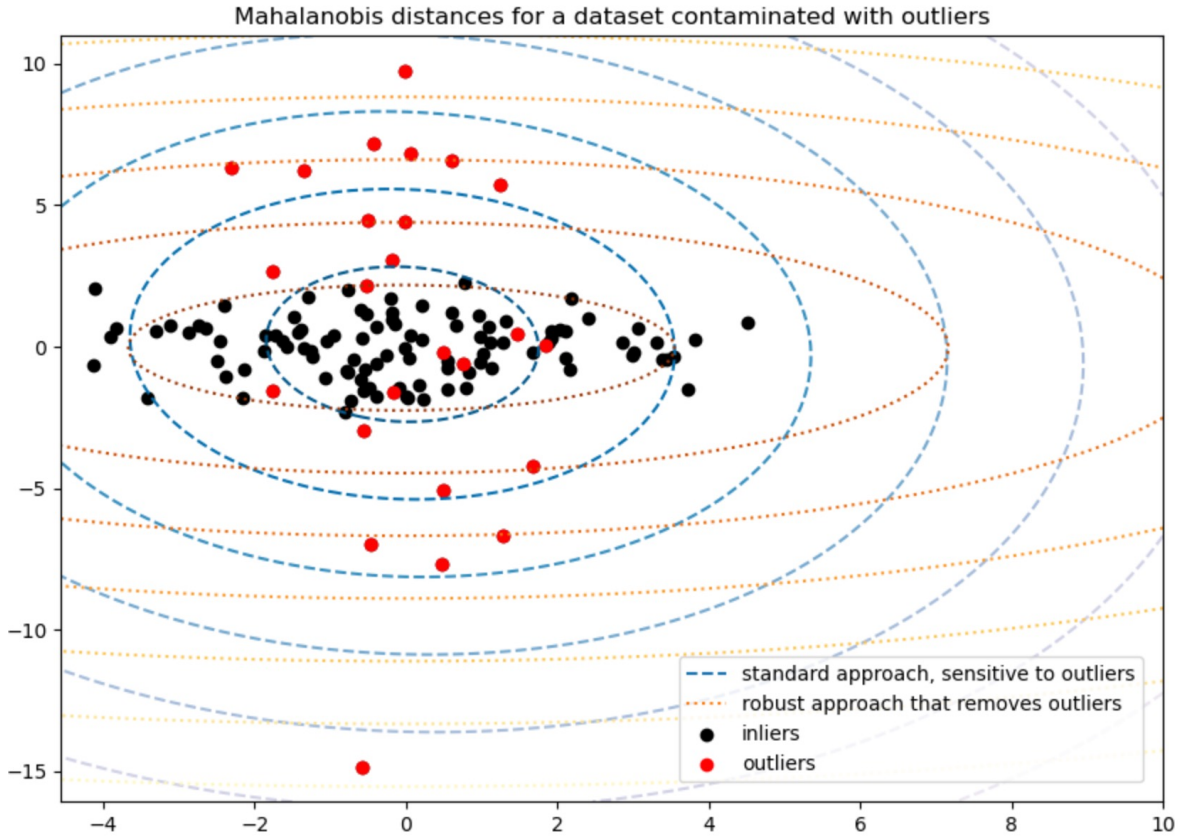
Statistical techniques

Mahalanobis distance: scikit-learn implementation

- Scikit-learn contains a number of covariance estimators. For details see https://scikit-learn.org/stable/modules/covariance.html.

- Several of these are sensitive to the presence of outliers. However, **robust covariance estimation** (using `MinCovDet()`) allows us to down-weigh those contaminations.

The method fits ellipses to the central data points ignoring the points outside the central mode.



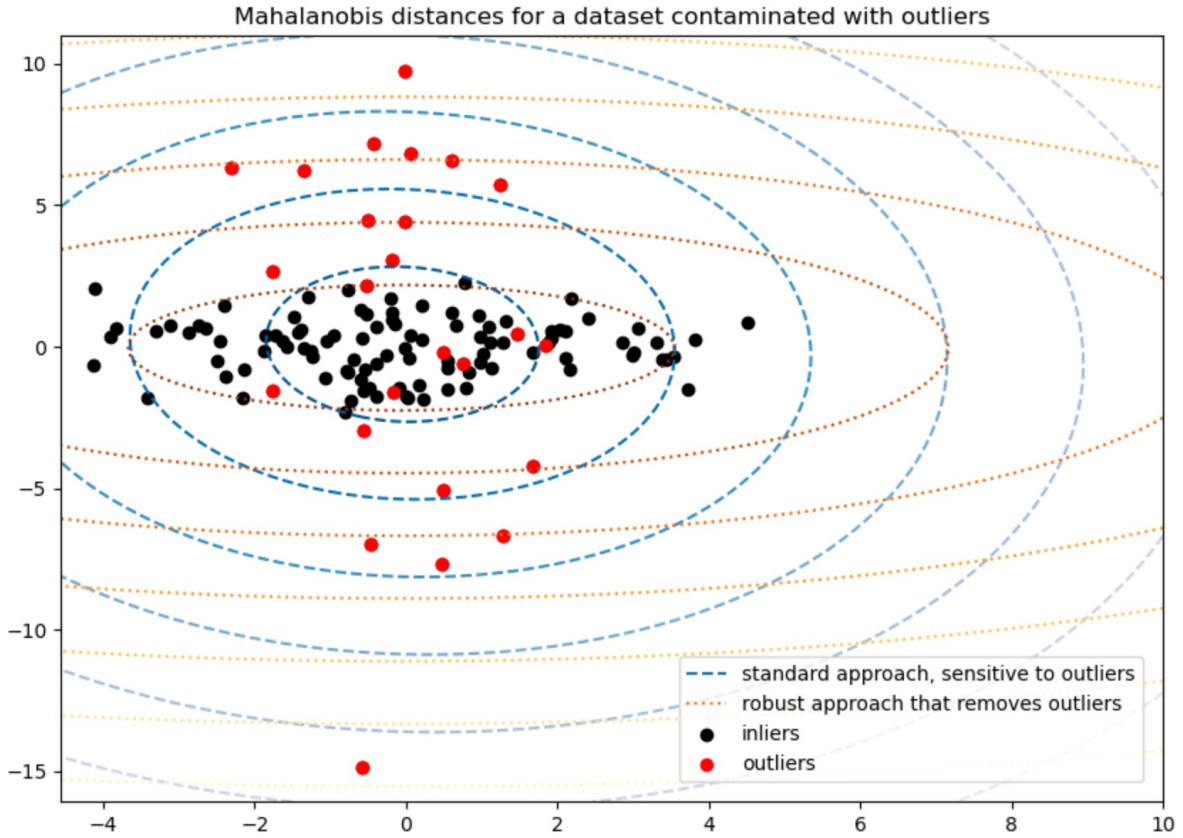
Statistical techniques

Mahalanobis distance: scikit-learn implementation

- Scikit-learn contains a number of covariance estimators. For details see https://scikit-learn.org/stable/modules/covariance.html.

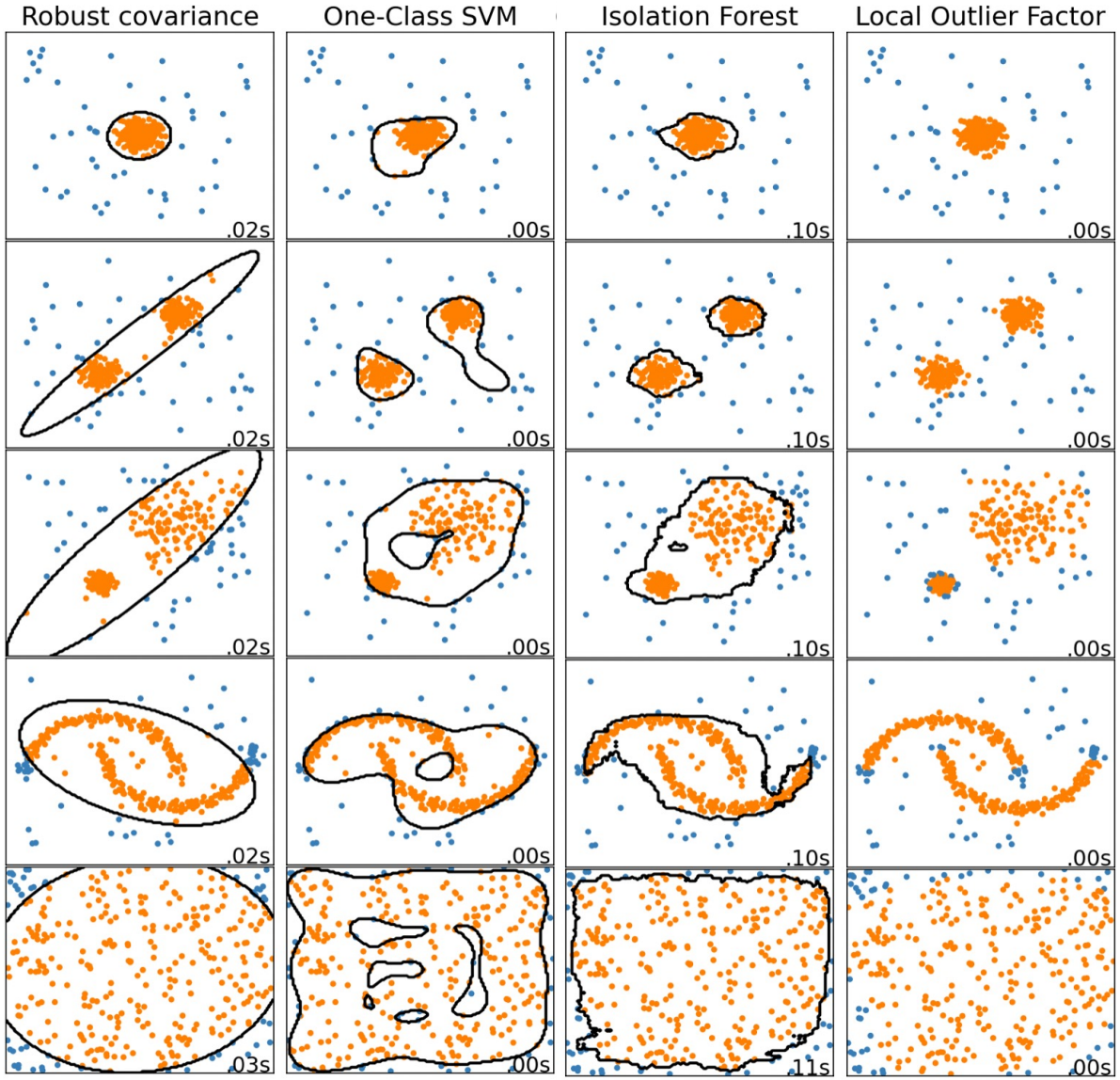
- Several of these are sensitive to the presence of outliers. However, **robust covariance estimation** (using `MinCovDet()`) allows us to down-weigh those contaminations.

The Mahalanobis distance is used as a measure of outlyingness.



Statistical techniques

A comparison with earlier approaches.



Statistical techniques

Pearson's chi-square statistic

- For multi-variate categorical data, we can generally determine whether a point is an outlier by using the **Pearson's chi-square test** for statistical hypothesis testing (provided that the observed dataset is large).

- We can test that a point, **x**, is not an outlier (i.e., it follows the same distribution as the observed data; the **null hypothesis**) by calculating the following test statistic, which follows the **chi-square distribution**:

$$\chi^2 = \sum_{j=1}^{D} \frac{(x_j - \mu_j)^2}{\mu_j}$$

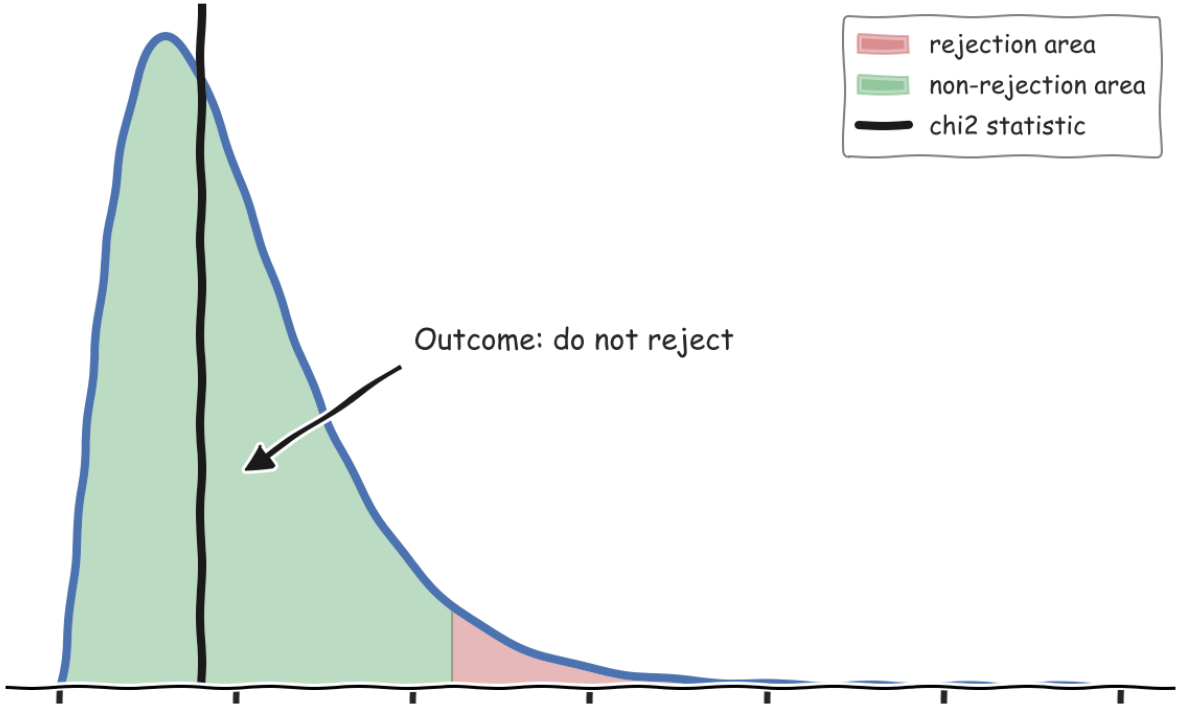
- Here, D is the dimension of the multi-variate data and μ_j the mean along the jth dimension of observed data.

Statistical techniques

Pearson's chi-square statistic

$$\chi^2 = \sum_{j=1}^{D} \frac{(x_j - \mu_j)^2}{\mu_j}$$

- If χ^2 is larger than a given threshold, then we reject the null hypothesis and adopt the alternative hypothesis. Our x is therefore likely an outlier.



Statistical techniques

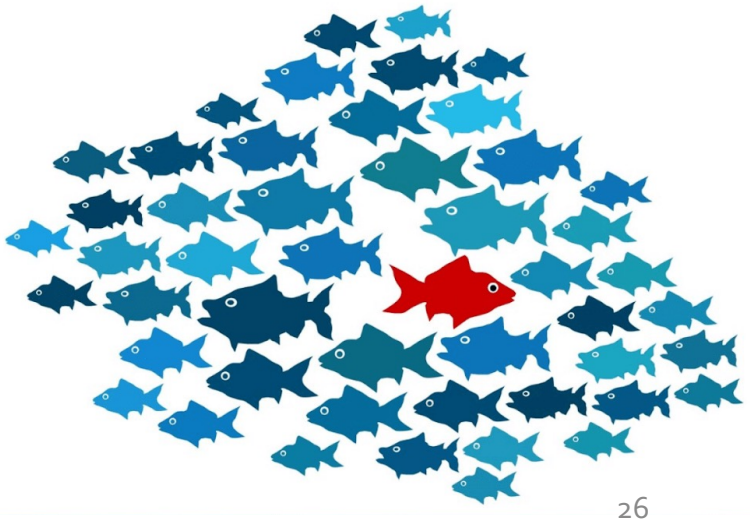
Likelihood-based approaches

- To detect anomalies using these approaches, we assume that a dataset, D, contains **samples from a mixture of two probability distributions**: the majority distribution (M) and the anomalous distribution (A).

- The distribution of our data is then given by

$$D = (1 - \lambda)M + \lambda A$$

- More specifically, M is a probability distribution **estimated from our data** (based on any kind of modelling method, e.g., naïve Bayes, maximum entropy, etc.). In contrast, A is initially assumed to be a **uniform distribution**.



Statistical techniques

Likelihood-based approaches

- We initially assume that all data points belong to M and let $LL_t(D)$ be the **log-likelihood of D** at a given time t. We can calculate the latter as follows:

$$L_t = \prod_{i=1}^{N} P_D(x_i) = \left[(1-\lambda)^{|M_t|} \prod_{x_i \in M_t} P_{M_t}(x_i)\right]\left[\lambda^{|A_t|} \prod_{x_i \in A_t} P_{A_t}(x_i)\right]$$

$$LL_t = |M_t| \log(1-\lambda) + \sum_{x_i \in M_t} \log P_{M_t}(x_i) + |A_t| \log\lambda + \sum_{x_i \in A_t} \log P_{A_t}(x_i)$$

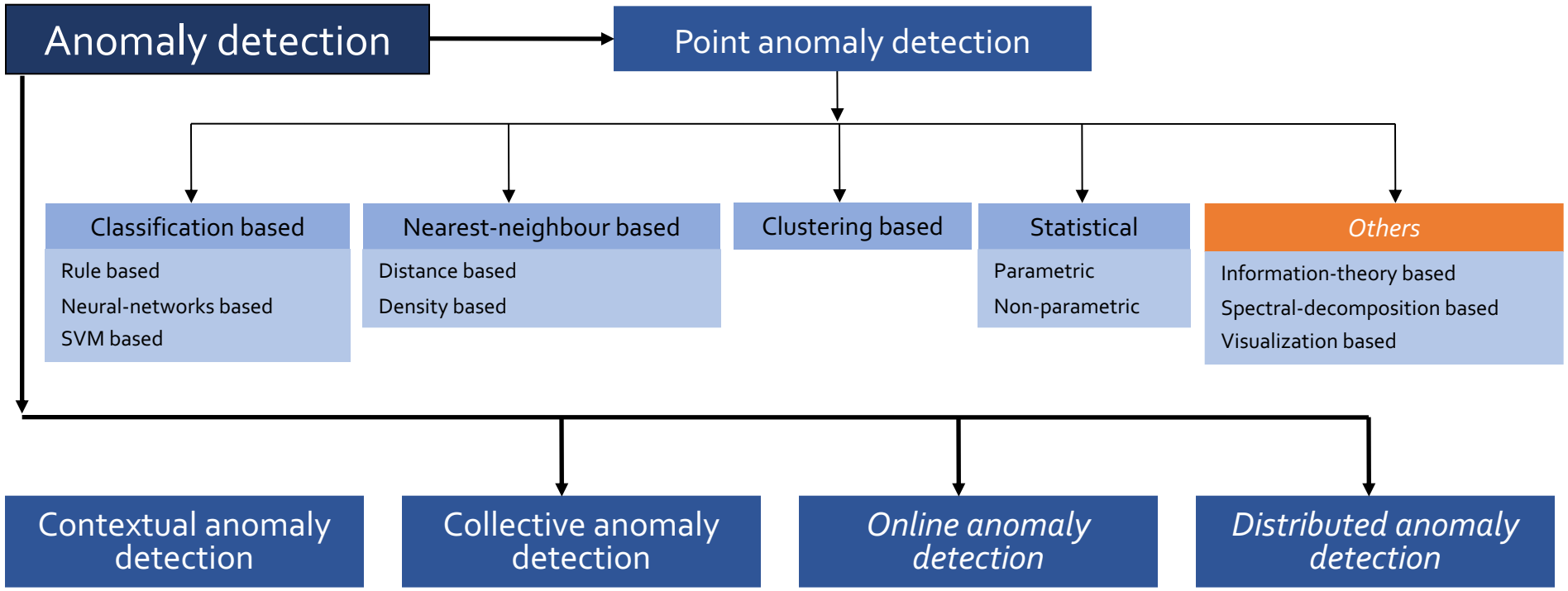
- For each point x_t in M, we move it to A and let $LL_{t+1}(D)$ be the new likelihood. We then **compute the different** $\Delta = LL_{t+1}(D) - LL_t(D)$.

If Δ larger than some threshold, c, we declare x_t an anomaly and move it permanently to A.

Taxonomy of techniques

An overview of anomaly detection approaches

- In the remainder of this class, we will look at several more of the **methods of anomaly detection** (following Chandola et al. (2008) – Anomaly Detection: A Survey; see reading materials):



Other techniques

Visualisation approaches

- An intuitive approach for point anomaly detection uses visualisation tools. These techniques complement other approaches as they (only) provide an alternate view of our data, which is then inspected manually.

ADVANTAGES

- These are quick and suitable for an initial analysis of low-dimensional data.
- The data scientist is kept in the loop during the analysis.

DISADVANTAGES

- Anomalies are detected visually which makes things subjective.
- Unsuitable for outlier detection in aggregation or partial views of high-dimensional data.
- Not suitable for real-time analyses.

Other techniques

Information-theory based approaches

- The key idea here is that outliers significantly alter the **information content in a dataset**. Anomaly detection then requires us to find those data instances that have a strong impact on the data's information content.

ADVANTAGES

- These approaches do not require any labels and can operate in an unsupervised mode.

DISADVANTAGES

- We require an information theoretic measure that is sensitive enough to detect anomalies.

Other techniques

Theoretic measure: entropy

- A range of measures have been used for this purpose. One commonly used example in information theory is the so-called **entropy**.

- For a dataset D, where each data item, x, belongs to a class C_D and P(x) is the probability of x in D, the entropy of D relative to this $|C_D|$-wise classification is defined as:

$$H(D) = \sum_{x \in C_D} P(x) \log \frac{1}{P(x)}$$

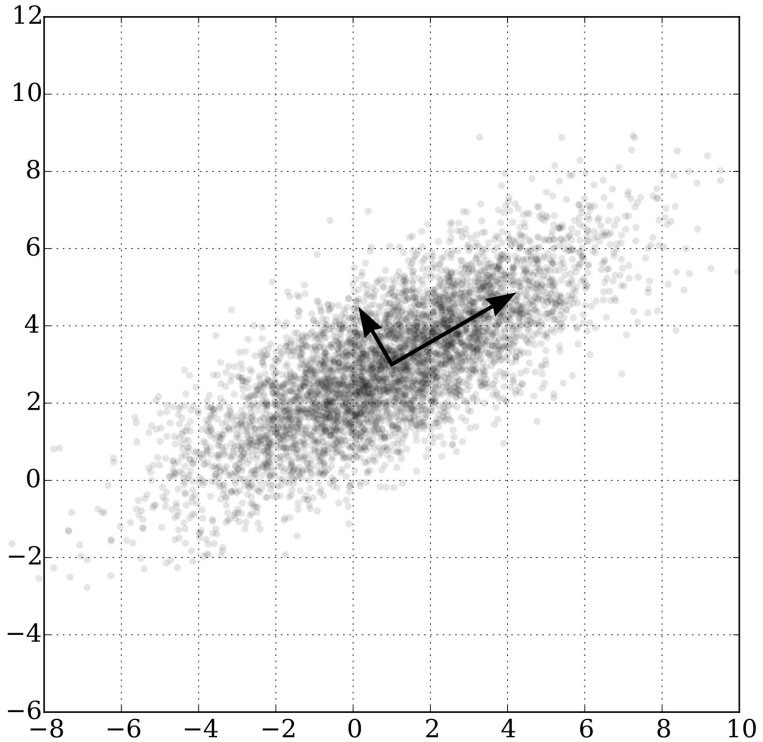
H measures the level of uncertainty / disorder / impurity of a collection of data. Smaller values correspond to ordered / purer datasets.

Other techniques

Principal component analysis (PCA)

- PCA is a dimensionality reduction tool that aims to identify **directions of largest variation** within a dataset. Mathematically, this corresponds to determining eigenvectors, $\vec{\lambda} = (\lambda_1, \ldots, \lambda_p)$, of the covariance matrix.

- Let's consider a test point, $\vec{x}$, and perform PCA to obtain the principal components, $y_1, \ldots, y_p$. Then the **sum of standardised principal component scores** follows a chi-square distribution with $q \leq p$ degrees of freedom.



$$\sum_{i=1}^{q} \frac{y_i^2}{\lambda_i} = \frac{y_1^2}{\lambda_1} + \cdots + \frac{y_q^2}{\lambda_q}$$

Other techniques

Principal component analysis (PCA)

- Given a significance level, α, our point $\vec{x}$ is then an **outlier if**

$$\sum_{i=1}^{q} \frac{y_i^2}{\lambda_i} > \chi_q^2(\alpha)$$

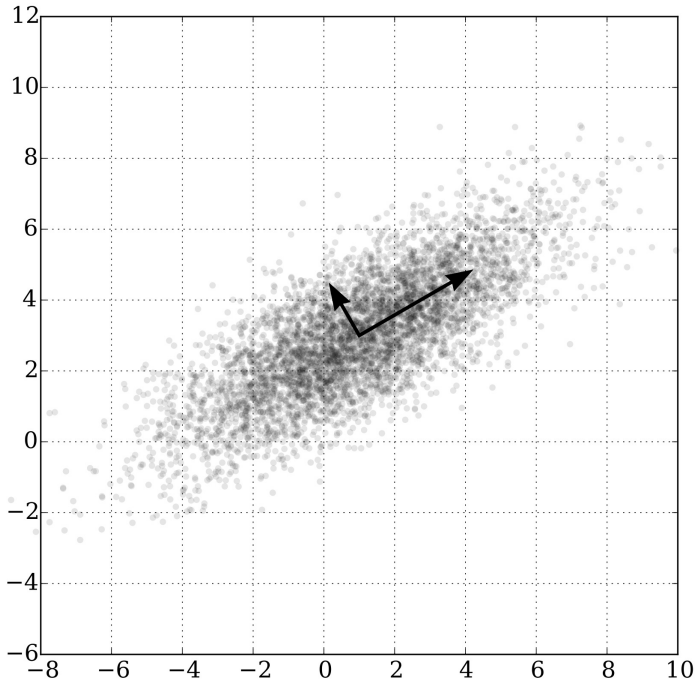
- Alternatively, we could also investigate the **last few components** of the PCA. Anomalies have high values for the following quantity

$$\sum_{i=p-r}^{q} \frac{y_i^2}{\lambda_i}$$

Other techniques

Principal component analysis (PCA)

- In general, we observe that the **first few principal components** capture the variability of the normal data. The higher PCA components are small and remain constant. In contrast, outliers will show **variability in these higher components**.

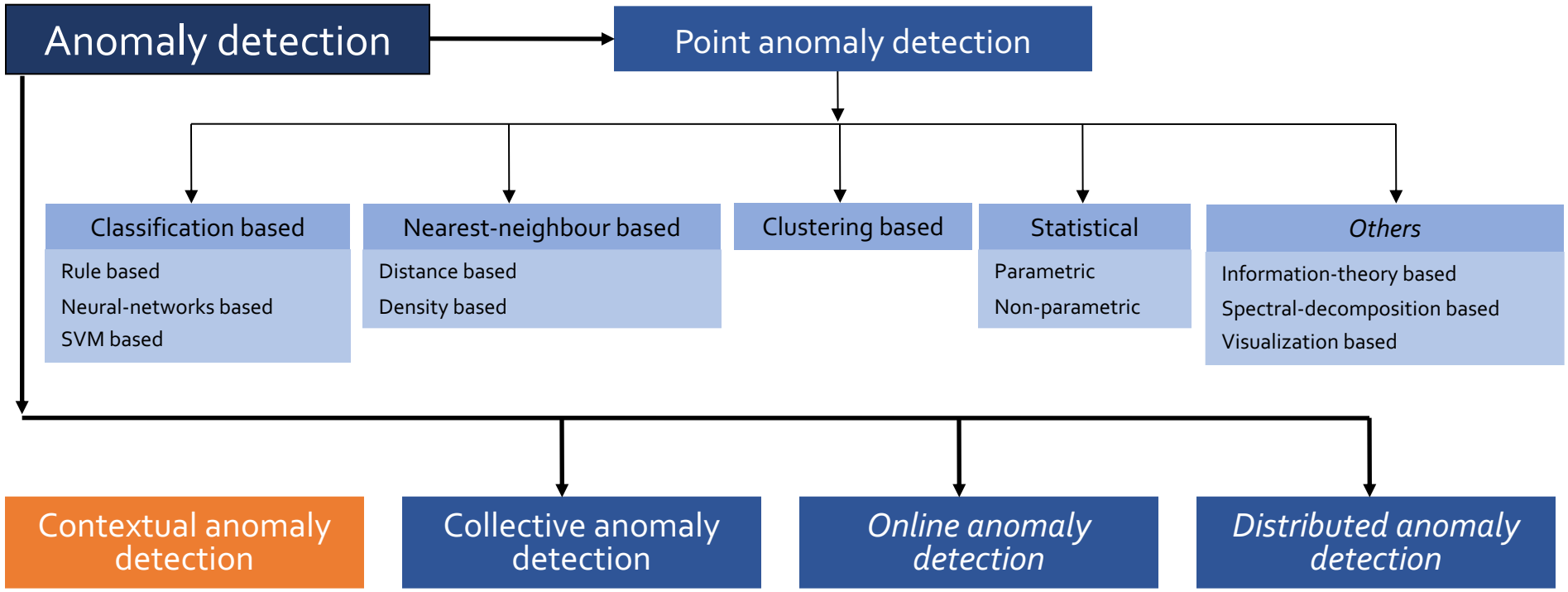


PCA reduces the dimensionality of a dataset and results in a simple classifier that is a function of only a few of the principal components. This makes it a fast method.

Taxonomy of techniques

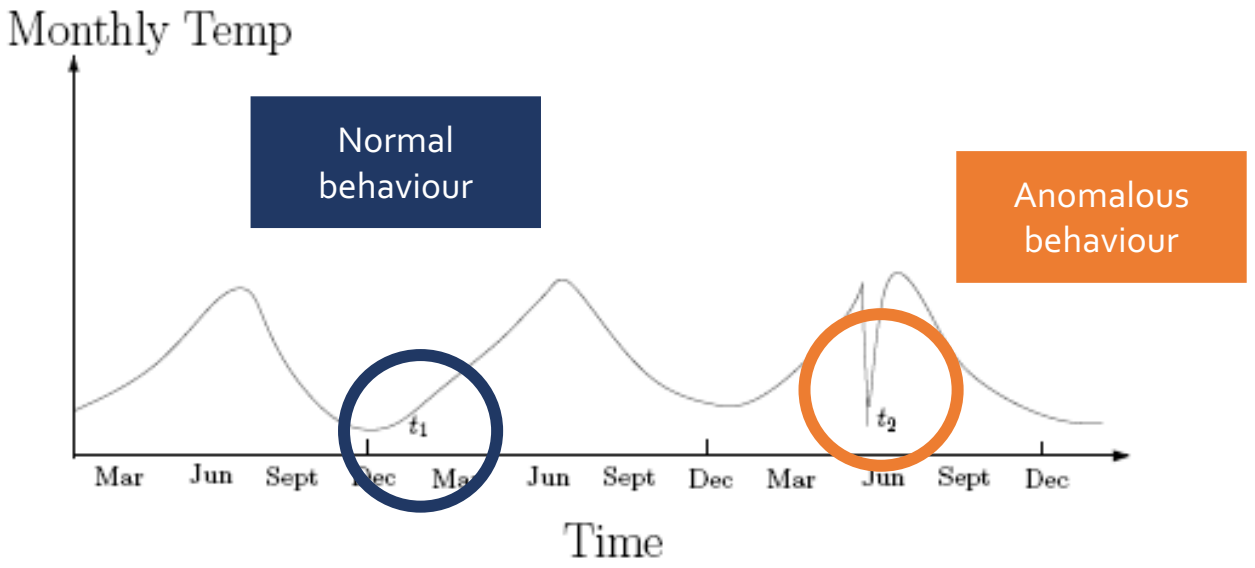
An overview of anomaly detection approaches

- In the remainder of this class, we will look at several more of the **methods of anomaly detection** (following Chandola et al. (2008) – Anomaly Detection: A Survey; see reading materials):



Types of anomalies
Contextual anomalies

- **Contextual anomaly**: An individual data instance is anomalous within a specific context. This requires a **notion of context**.
- Such anomalies are also referred to as **conditional anomalies** (see e.g., Song et al., Conditional Anomaly Detection, IEEE,19, 5, 2006).



Contextual anomalies

An overview

- To detect contextual anomalies, we make the key assumption that all normal instances within a context are similar (in terms of their behavioural attributes), while anomalies will behave differently.

- Anomaly detection then involves the following steps:
  - Identify context around a data instance using a set of **contextual attributes**.
  - Determine if a test data instance is anomalous within this context.

ADVANTAGES

- Detect anomalies hard to identify in a global context.

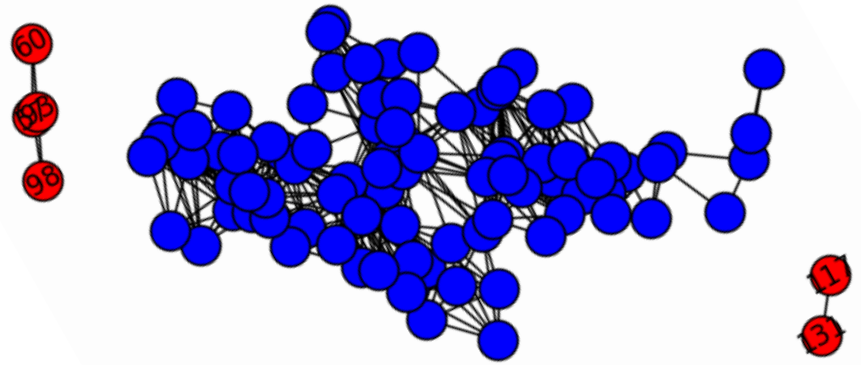
CHALLENGES

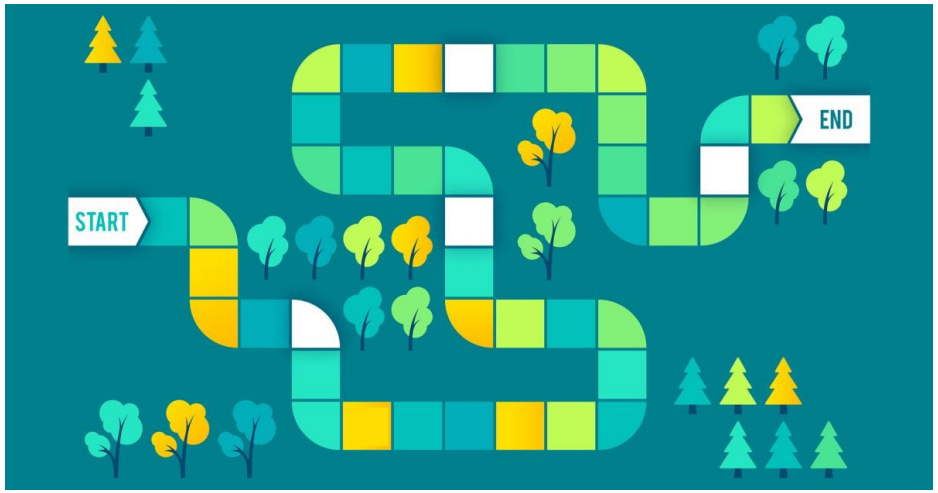
- Identify a suitable set of contextual attributes.
- Determine context using these attributes.

Contextual anomalies

Contextual attributes

- Contextual attributes define a neighbourhood (context) for a given instance. These can, e.g., include
  - Spatial context (e.g., latitude and longitude)
  - Graph context (e.g., edges and weights)
  - Sequential context (e.g., position in a series or time)
  - Profile context (e.g., user demographic)





Contextual anomalies
Techniques

- To detect contextual anomalies, we follow **two main approaches**.
- **Reduction to point anomaly detection**
  - We first segment data using our contextual attributes.
  - Apply a traditional AD algorithm on behavioural attributes in segments.
  - However, contextual attributes often cannot be segmented easily.
- **Utilise structure in data**
  - We can build models from the data using contextual attributes, such as time series models (ARIMA, etc.; see earlier lectures).
  - Models automatically analyse data instances with respect to their context.

Contextual anomalies

An example for conditional anomalies; Song et al. (2006)

- The authors represent data points in a dataset as [x, y], where x denotes contextual attributes and y denotes behavioural attributes. Then, a mixture of N_U Gaussian models, U, is learnt from the contextual data, while a mixture of N_V Gaussian models, V, is learnt from the behavioural data.

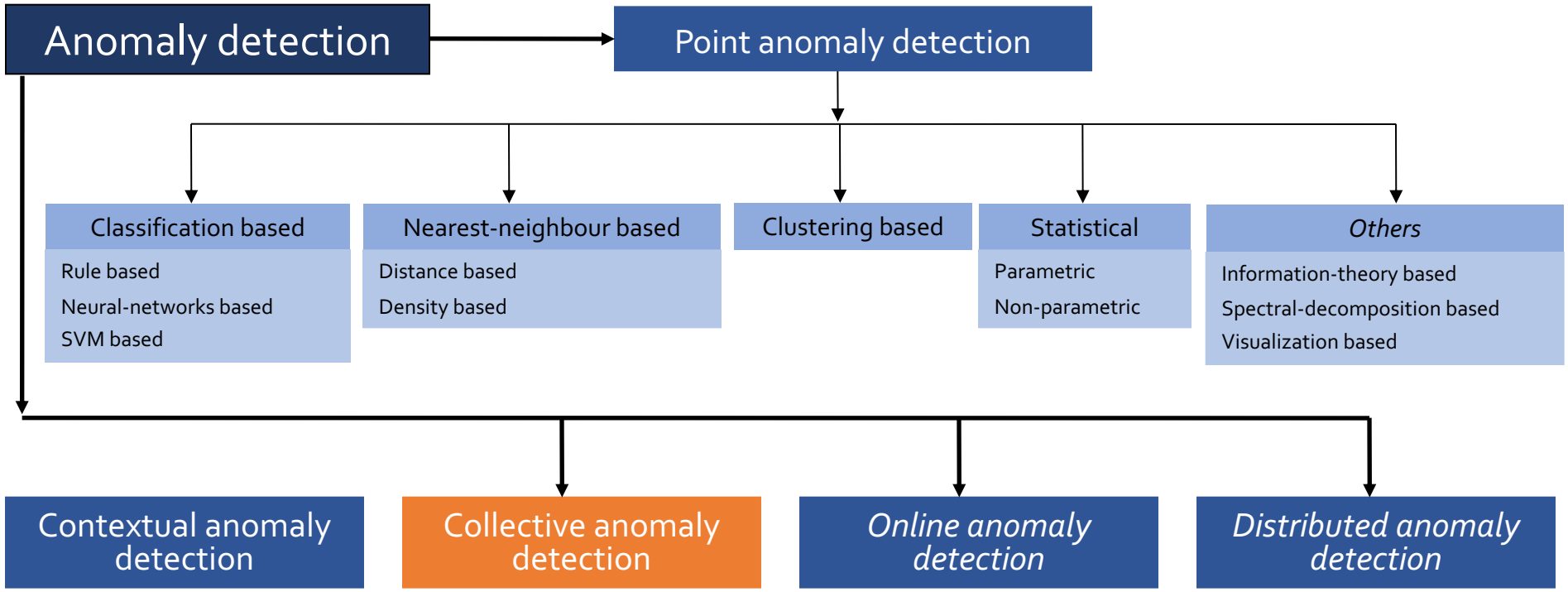
- The authors than determine a mapping $p(V_j|U_i)$ which indicates the probability of the behavioural attributes being generated by component V_j when the contextual aspects are generated by component U_i.

- This approach allows us to answer the following kinds of questions:
  - How likely is the contextual part to be generated by a component U_i of U?
  - What is the probability of a behavioural attribute to be generated by V_j?
  - Given U_i, what is the most likely V_j that will generate the behavioural attributes?

Taxonomy of techniques

An overview of anomaly detection approaches

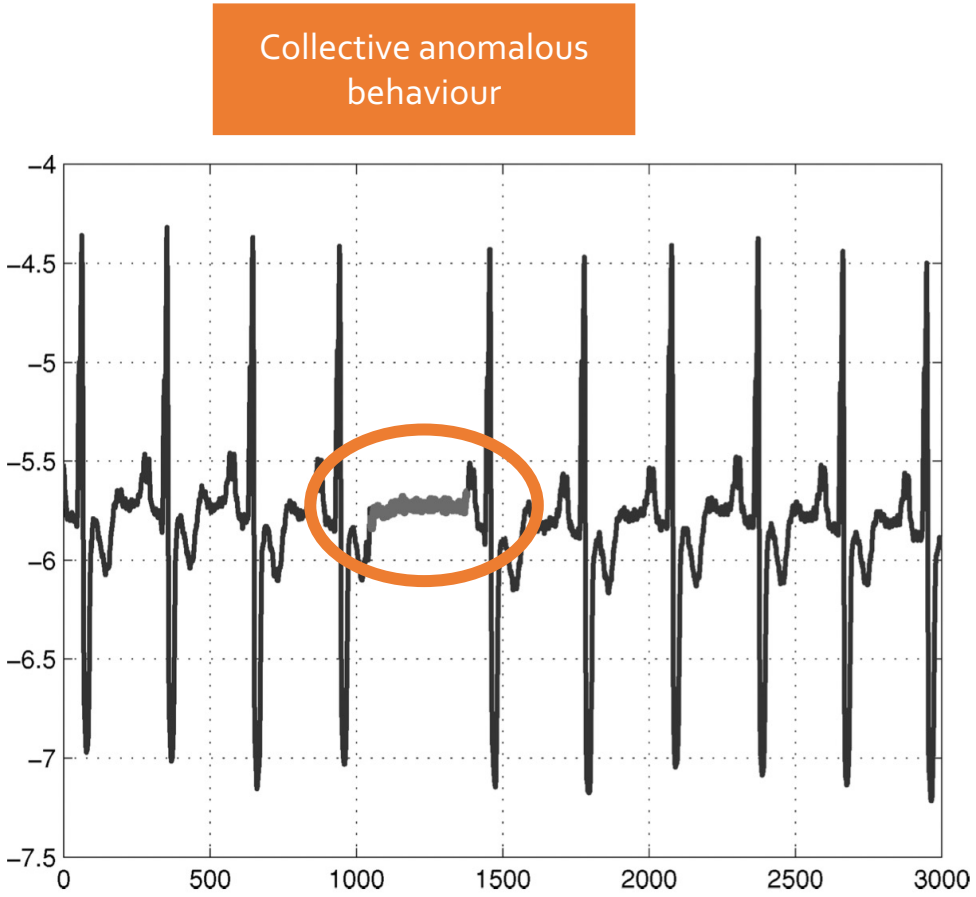
- In the remainder of this class, we will look at several more of the **methods of anomaly detection** (following Chandola et al. (2008) – Anomaly Detection: A Survey; see reading materials):



Types of anomalies
Collective anomalies

- **Collective anomaly**: A collection of related data instances is anomalous.
- This requires a (sequential, spatial, or graph) relationship among data instances.

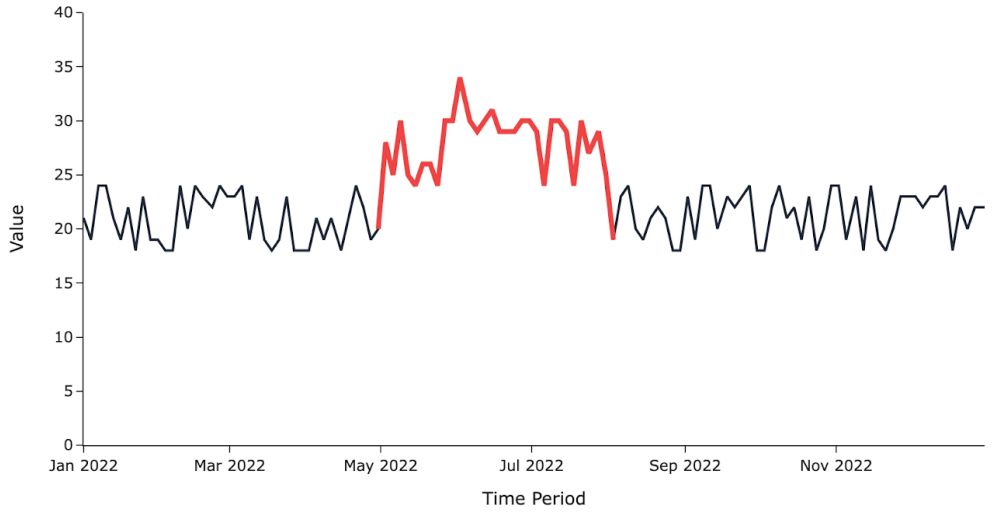
The individual instances within a collective anomaly are not anomalous by themselves.



Collective anomalies

Overview of techniques

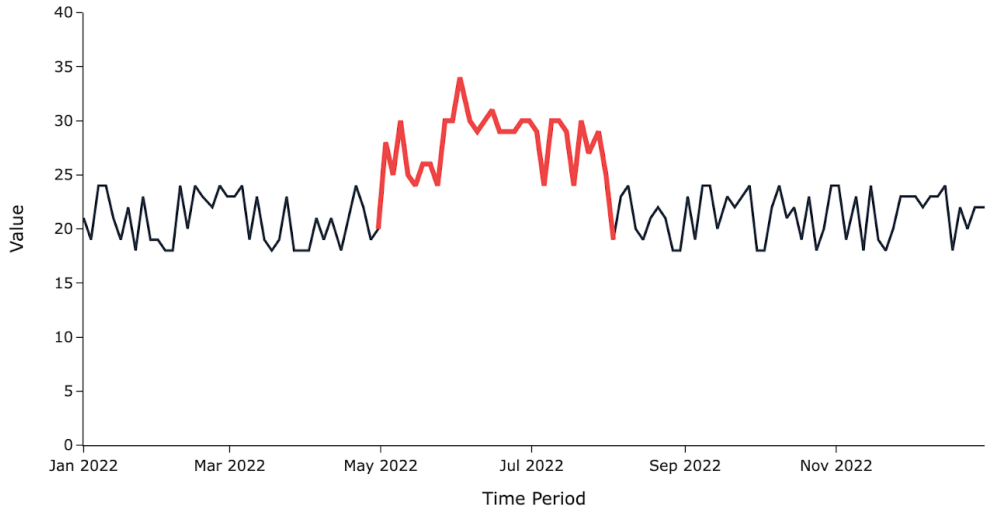
- To detect collective anomalies, we **take advantage of the relation-ships** between data instances in a dataset. We can distinguish:
  - **Sequential anomaly detection** to detect anomalous sequences.
  - **Spatial anomaly detection** to detect anomalous sub-regions in spatial data.
  - **Graph anomaly detection** to detect anomalous sub-graphs in graph data.



Collective anomalies

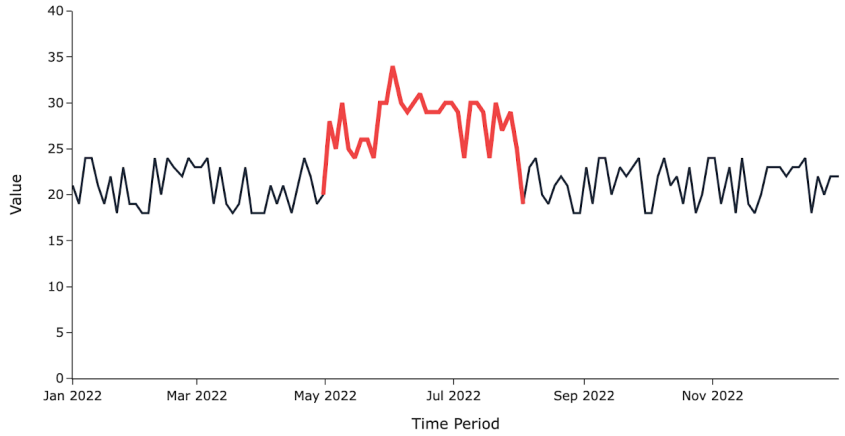
Overview of techniques

- To detect collective anomalies, we **take advantage of the relation-ships** between data instances in a dataset. We can distinguish:
  - **Sequential anomaly detection** to detect anomalous sequences.
  - **Spatial anomaly detection** to detect anomalous sub-regions in spatial data.
  - **Graph anomaly detection** to detect anomalous sub-graphs in graph data.



Collective anomalies

Sequential anomaly detection



- A number of AD techniques have been proposed for symbolic sequences. However,
  - Techniques often apply to a single domain only.
  - Little comparative evaluation across different domains.
  - Evaluation is essential to identify strengths and weaknesses of these techniques.

- To study this problem in **a general context** that is applicable to a range of domains (such as aircraft safety, intrusion detection, etc.) we can **define sequential anomaly detection** as follows

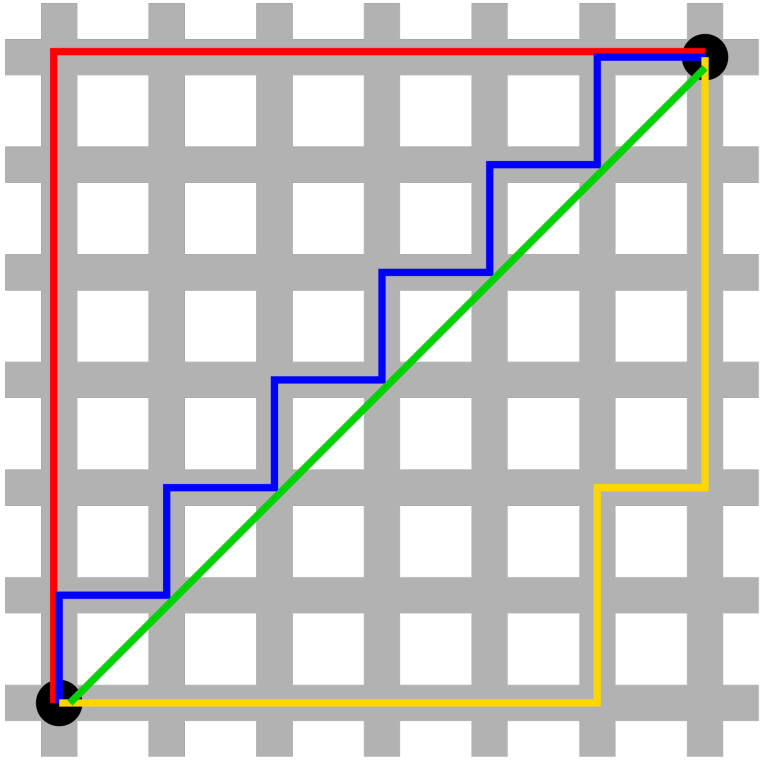
Given a set of N sequences S, and a query sequences S_q, find an anomaly score for S_q with respect to S, where we assume that subsequences in S are (mostly) normal.

Collective anomalies

Sequential AD: kernel-based techniques

- A way to tackle this is via **kernel-based methods,** which define a similarity kernel based on some similarity measure, such as the Manhattan distance or the longest common subsequence (LCS).

- Based on this measure, we can then apply any traditional proximity-based AD technique:
  - **Clustering analysis** to cluster normal sequences into a fixed number of clusters. The anomaly score of a test sequence is then the inverse of its similarity measure with respect to the closest cluster medoid.
  - **Nearest-neighbour analysis**, where the anomaly score of a test sequence is the inverse of its similarity to the k^{th} nearest neighbour in the normal sequence dataset.



S1 = B C D A A C D

S2 = A C D B A C

Longest Common Substring is:

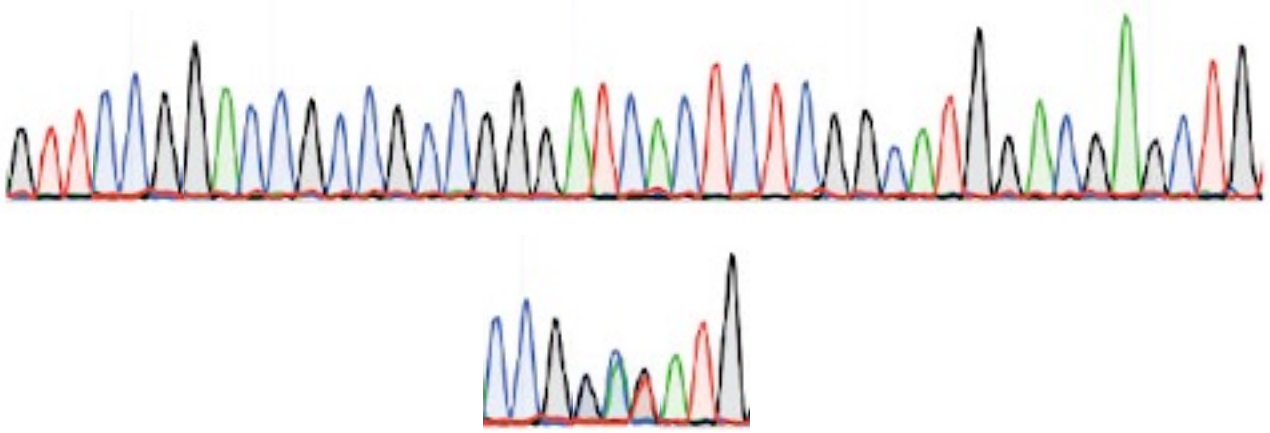
C D A C

Collective anomalies

Sequential AD: window-based techniques

- Another way to identify anomalies is via window-based tools, where we aim to extract **finite-length sliding windows** from a given test sequence.

- For each sliding window, we determine its frequency in the training dataset. The **frequency of occurrence** then acts as an **inverse anomaly score**.



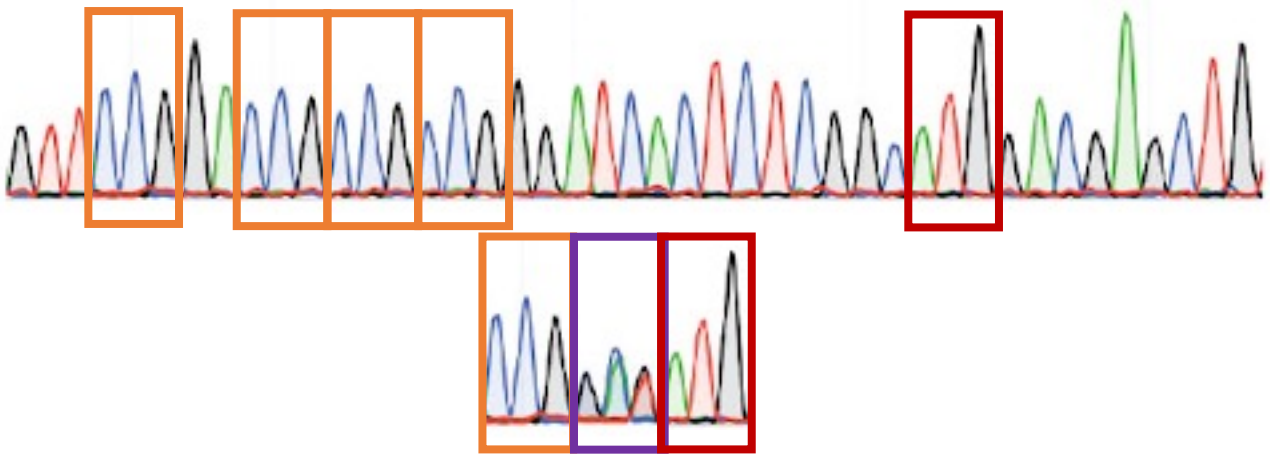
We then average the per-window anomaly score to obtain an overall anomaly score for the entire test sequence.

Collective anomalies

Sequential AD: window-based techniques

- Another way to identify anomalies is via window-based tools, where we aim to extract **finite-length sliding windows** from a given test sequence.

- For each sliding window, we determine its frequency in the training dataset. The **frequency of occurrence** then acts as an **inverse anomaly score**.

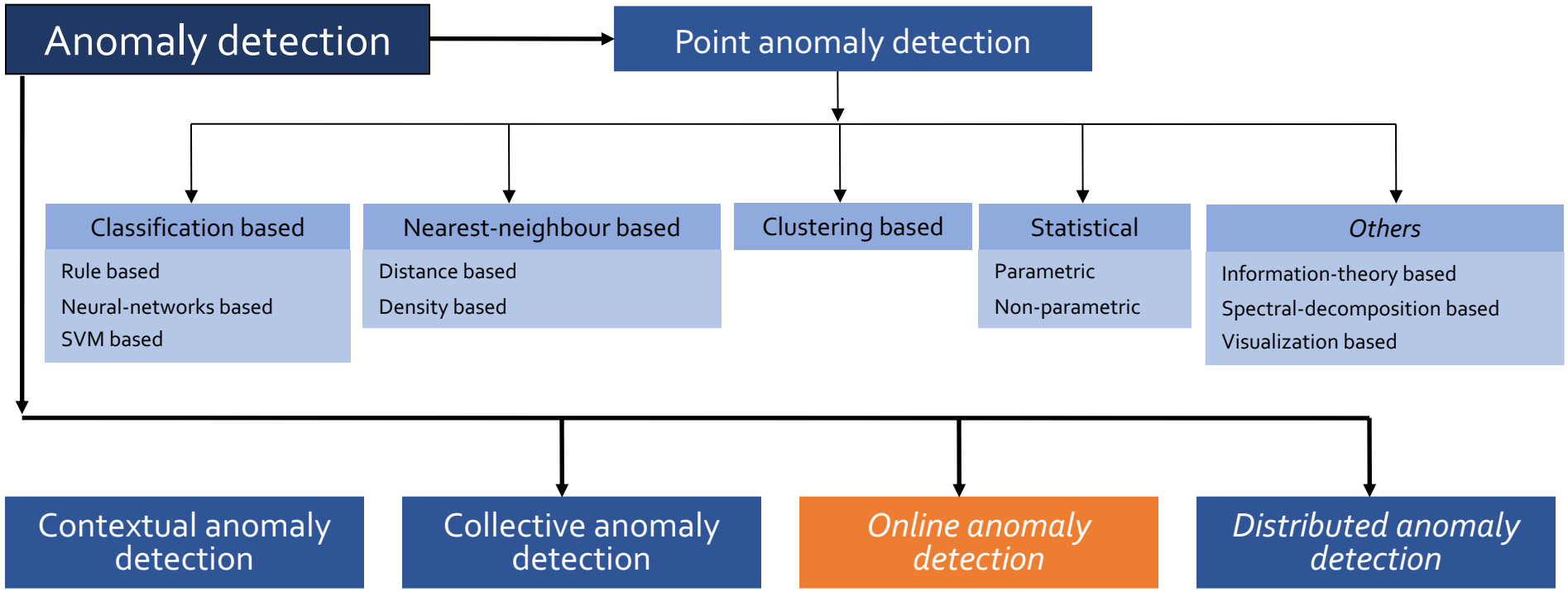


We then average the per-window anomaly score to obtain an overall anomaly score for the entire test sequence.

Taxonomy of techniques

An overview of anomaly detection approaches

- In the remainder of this class, we will look at several more of the **methods of anomaly detection** (following Chandola et al. (2008) – Anomaly Detection: A Survey; see reading materials):



Online anomaly detection

The idea

- When addressing problems like video feeds, network traffic, aircraft safety or credit-card transactions, data often **arrives in streaming mode**. So, anomalies need to be **detected in real time**.

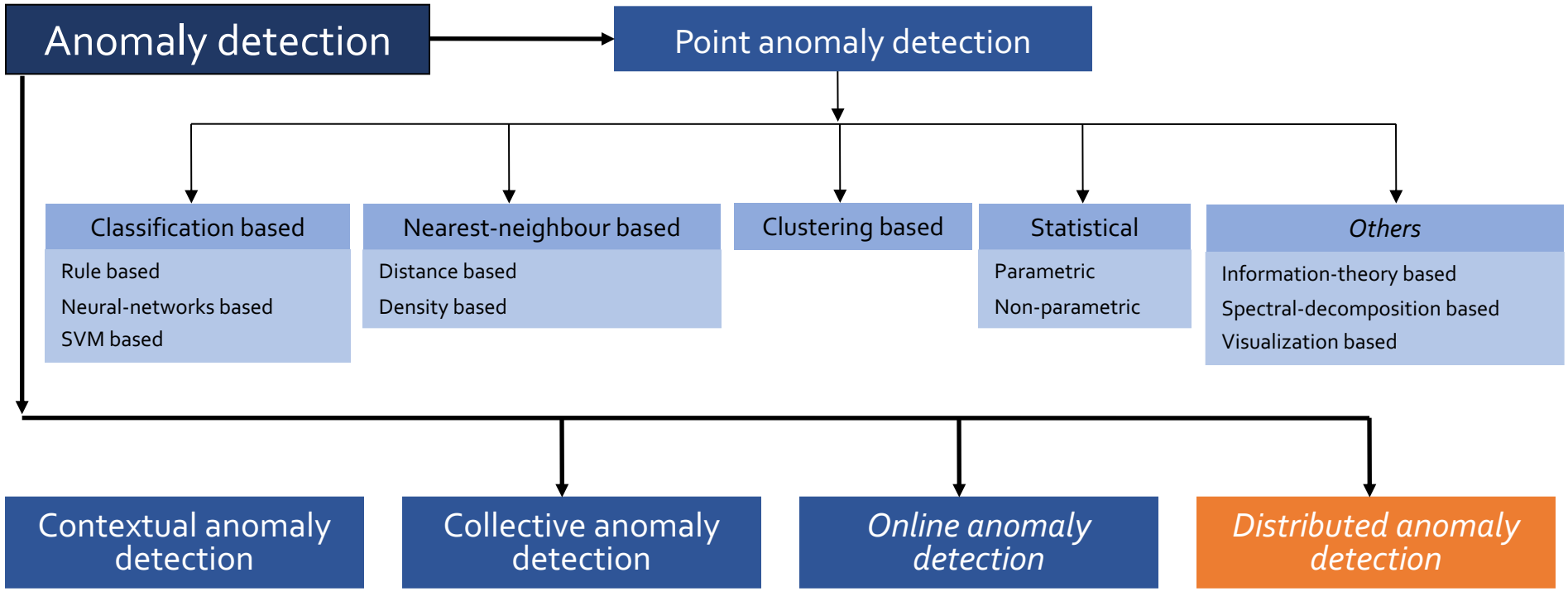
CHALLENGES

- When do we reject an outlier candidate?
- As "normal" behaviour changes, when do we update a model (periodically at fixed times, incremental after every data record, reactively only when needed)?

Taxonomy of techniques

An overview of anomaly detection approaches

- In the remainder of this class, we will look at several more of the **methods of anomaly detection** (following Chandola et al. (2008) – Anomaly Detection: A Survey; see reading materials):

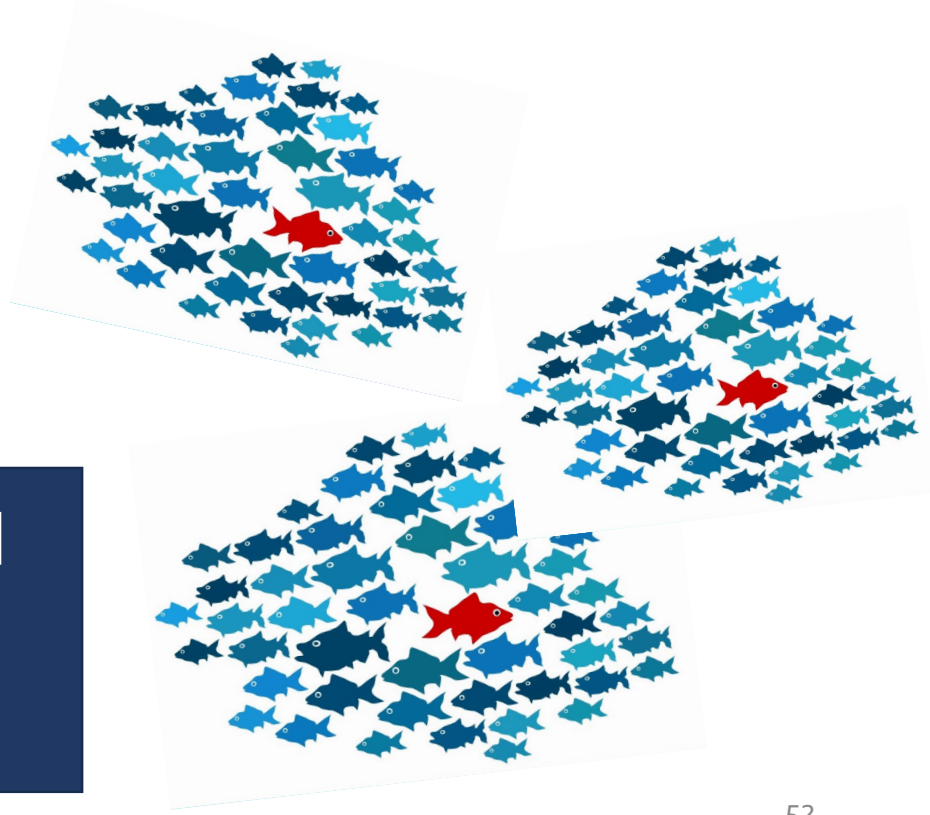


Distributed anomaly detection

The idea

- Data in many anomaly detection applications comes from **different sources** (e.g., users in a network). Outliers that occur across multiple locations simultaneously may be undetected by analysing only data from a single location.

- Detecting anomalies in such complex systems requires the integration of information about detected anomalies from single locations to **detect anomalies at the global level**.



There is a need for high-performance and distributed algorithms for correlation and integration of anomalies.

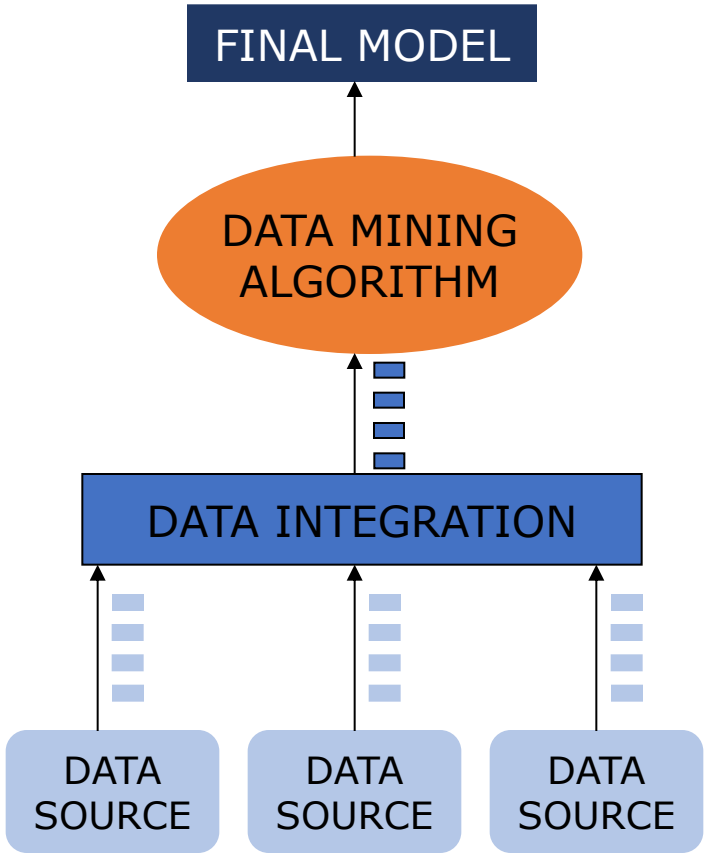
Distributed anomaly detection

Different approaches

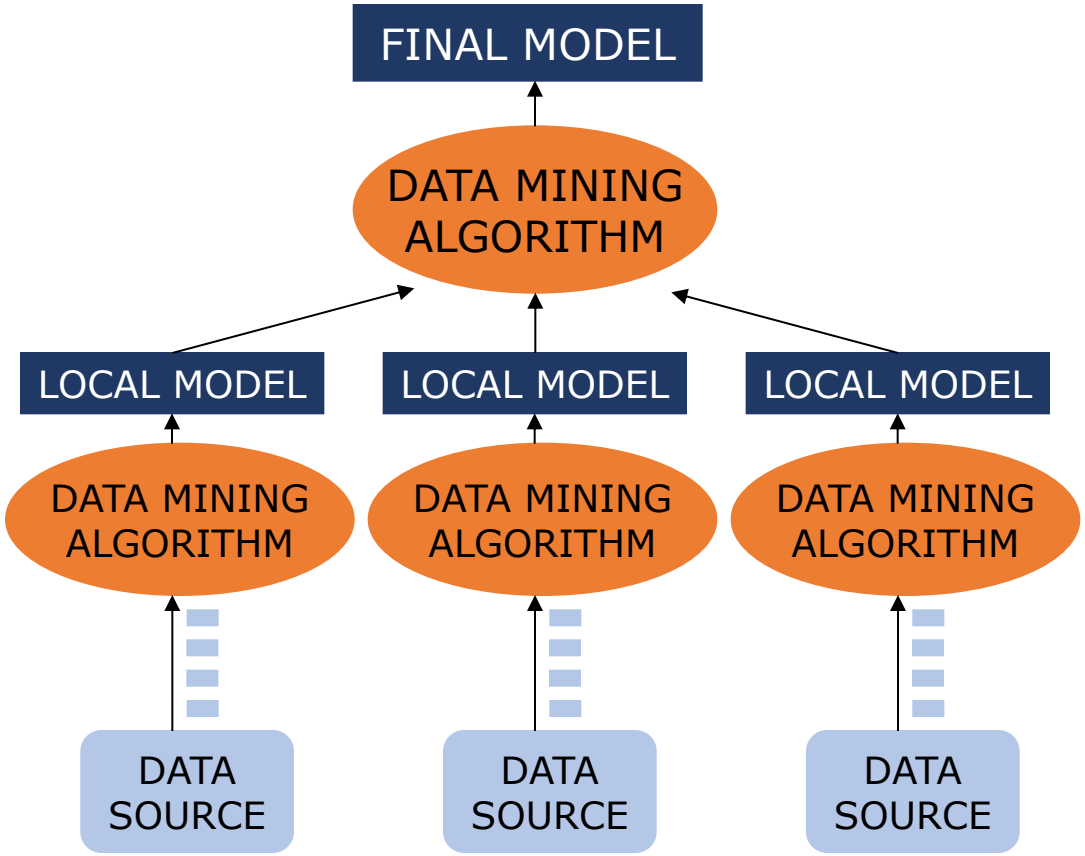
- Simple data exchange approaches
  - Merge data at single locations and then exchange these data between distributed locations.
- Distributed nearest-neighbour approaches
  - Exchange one data record per distance computation to other centres.
  - Computationally inefficient, but e.g., used for privacy-preserving AD algorithms.
- Methods based on exchange of models
  - Develop statistical or data-mining models for normal/anomalous behaviour.
  - Share these models across multiple locations.
  - Combine them at each location to detect global anomalies

Distributed anomaly detection

Centralised vs. distributed architectures



Centralised processing

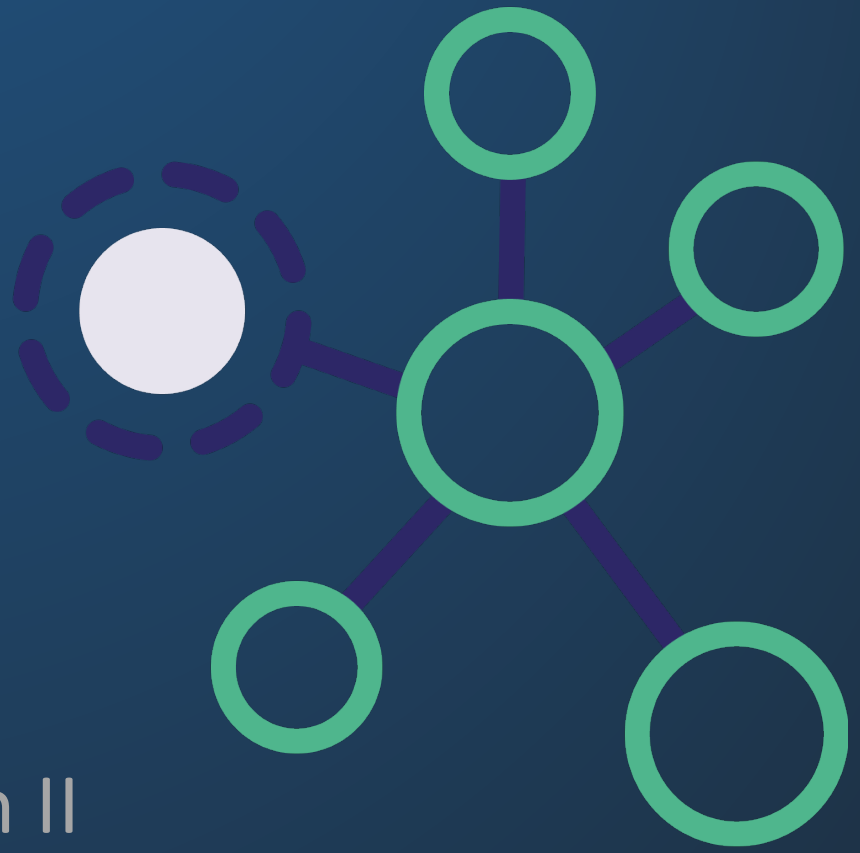


Distributed processing

Quick recap

Techniques for anomaly detection II

Summary



Summary

- Anomaly detection can **detect critical information** in data.
- Anomaly detection **applies to various domains**. Anomalies and outliers are often the piece of information of greatest interest.
- The nature of the anomaly detection problem is dependent on the application domain. We, therefore, need different techniques to solve a particular problem formulation.
- We introduced statistical-based approaches and several others for point anomaly detection. We then focused on contextual and collective anomaly detection and highlighted a few special approaches.