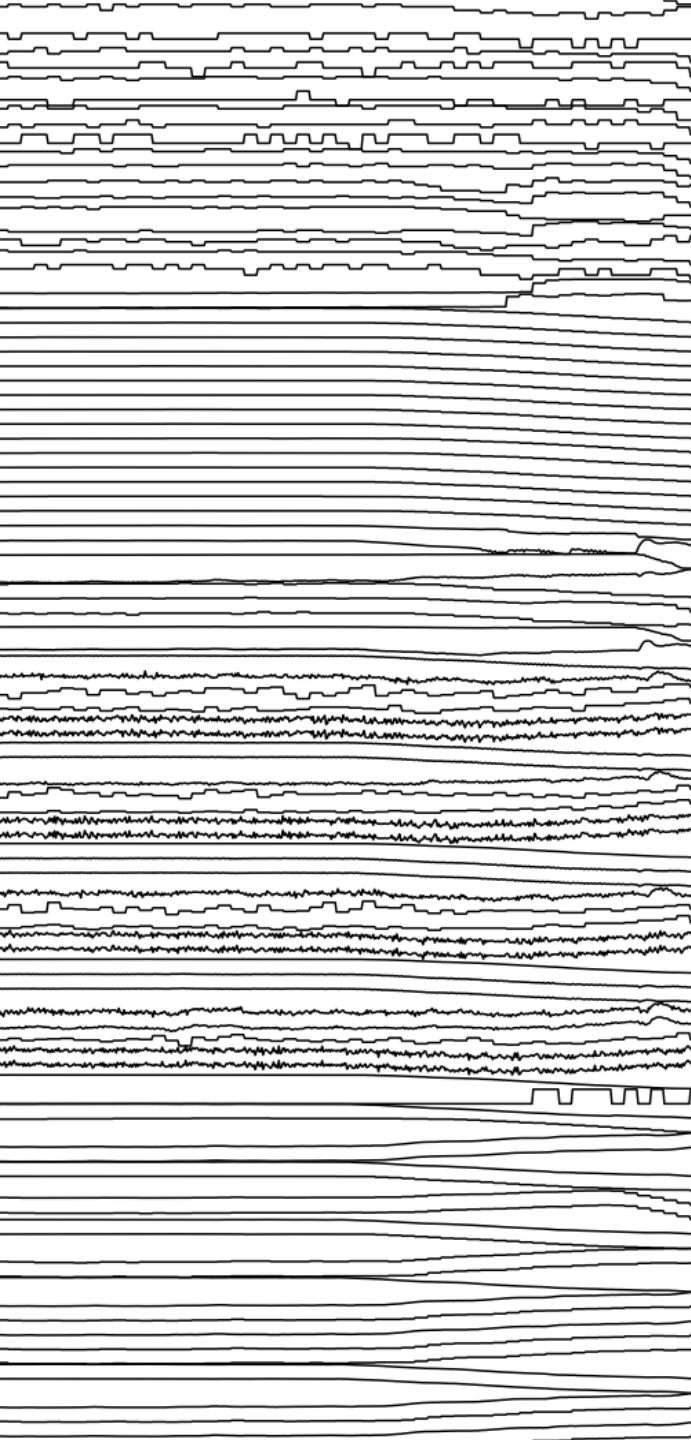




Advances in Time-Series Anomaly Detection:

Algorithms, Benchmarks, and Evaluation Measures

John Paparrizos
The Ohio State University;
Aristotle University
paparrizos.1@osu.edu

Paul Boniol
Inria, ENS, PSL University
Paul.boniol@inria.fr

Qinghua Liu
The Ohio State University
liu.11085@osu.edu

Themis Palpanas
Université Paris Cité; IUF
themis@mi.parisdescartes.fr





Table of Contents

- Part 1: Introduction, Motivation and Foundations
- Part 2: Taxonomy of Anomaly Detection Methods
- Part 3: Evaluation Measures
- Part 4: Anomaly Detection Benchmarks
- Part 5: Automated Solutions for Anomaly Detection
- Part 6: Conclusion and Open Problems

Part 1: Introduction, Motivation and Foundations

Introduction: *Time series are Everywhere*

Energy Production



Edf.fr: tinyurl.com/yc7x5xje

Astrophysics



Virgo: <https://www.virgo-gw.eu/>

Medicine



tinyurl.com/39dx2us4

Volcanology

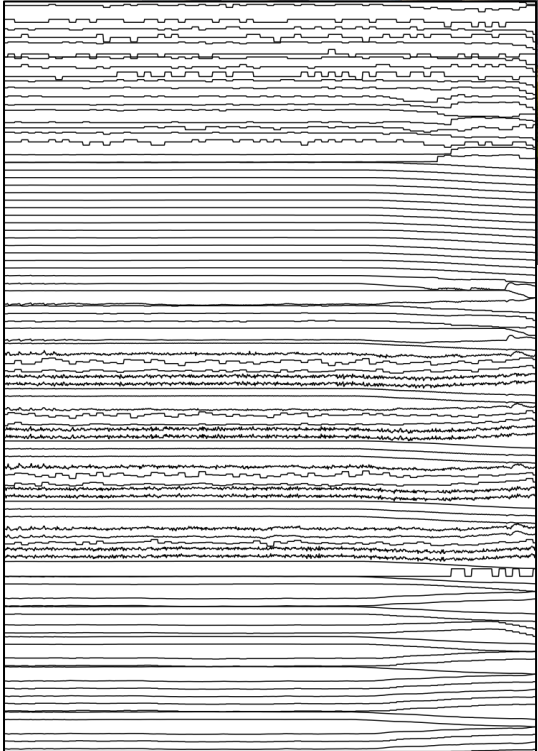


tinyurl.com/ybcttmfz

Introduction: *Time series are Everywhere*

Energy Production

Secondary circuit sensor
measurements



Astrophysics



Virgo: <https://www.virgo-gw.eu/>

Medicine



tinyurl.com/39dx2us4

Volcanology

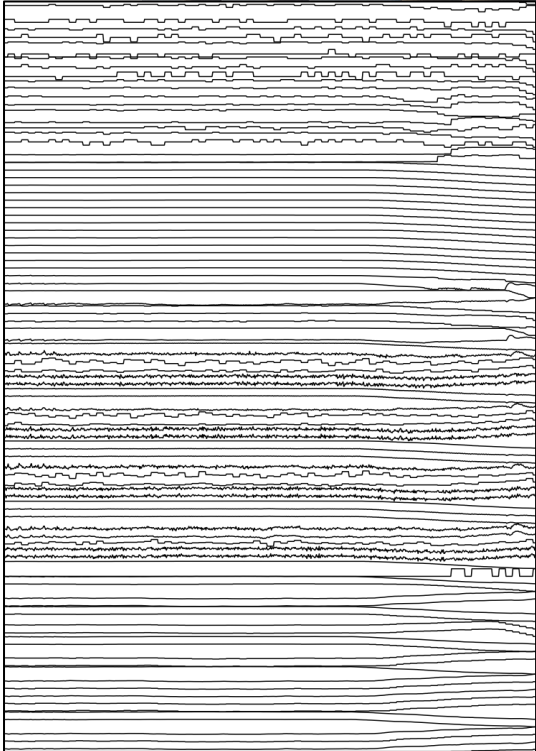


tinyurl.com/ybcttmfz

Introduction: *Time series are Everywhere*

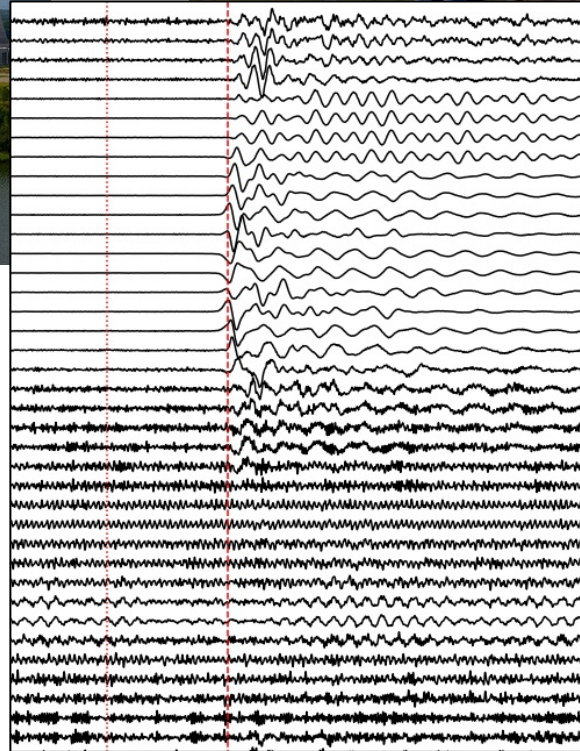
Energy Production

Secondary circuit sensor measurements



Astrophysics

Fiber-acoustic sensors in the VIRGO north building



Medicine



tinyurl.com/39dx2us4

Volcanology

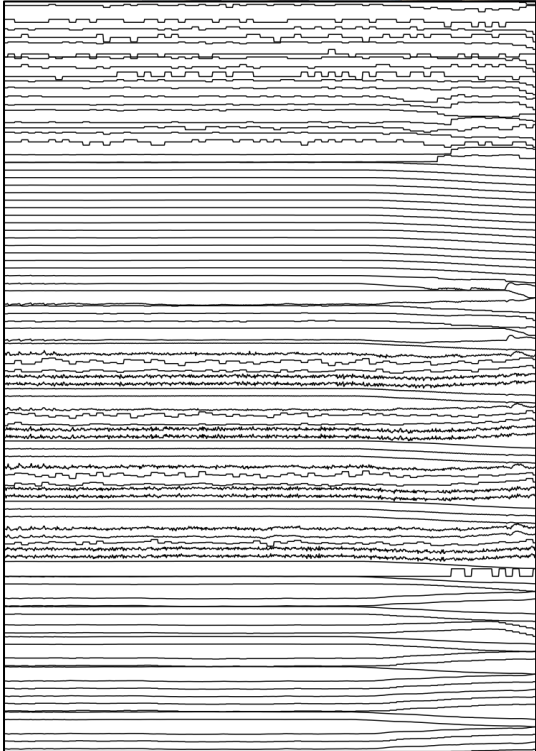


tinyurl.com/ybcttmfz

Introduction: *Time series are Everywhere*

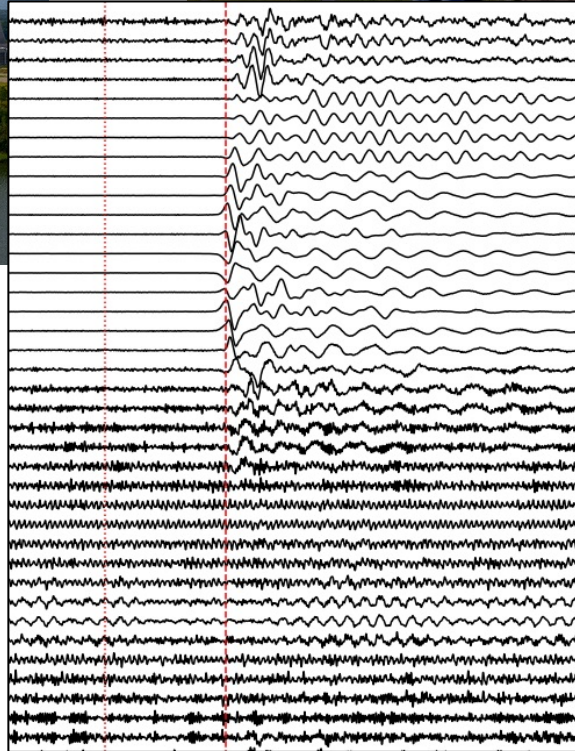
Energy Production

Secondary circuit sensor measurements



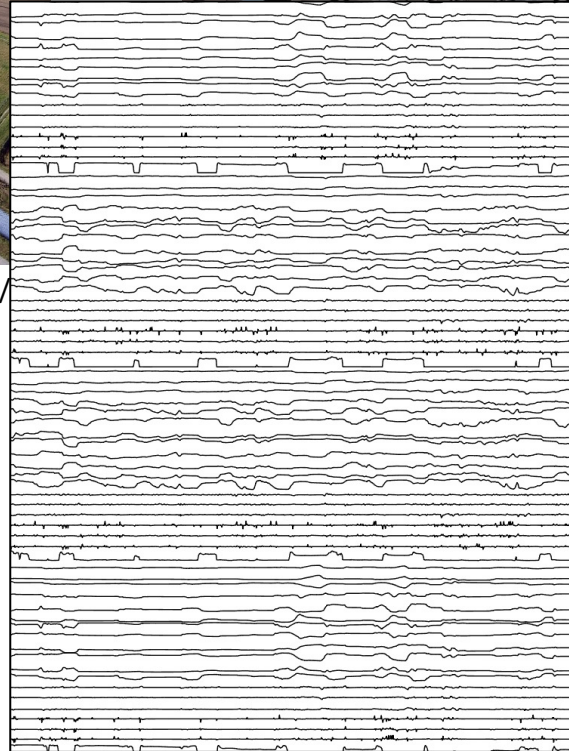
Astrophysics

Fiber-acoustic sensors in the VIRGO north building



Medicine

Sensor measurements of the Da Vinci surgery robot



Volcanology

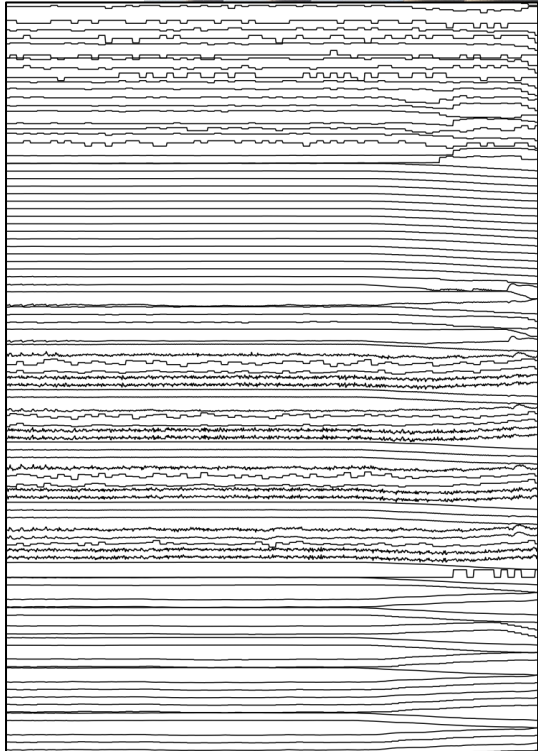


tinyurl.com/ybcttmfz

Introduction: *Time series are Everywhere*

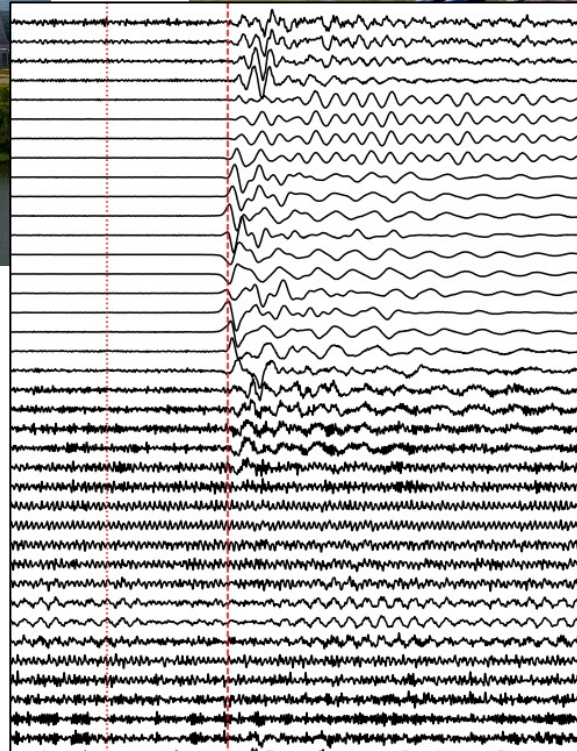
Energy Production

Secondary circuit sensor measurements



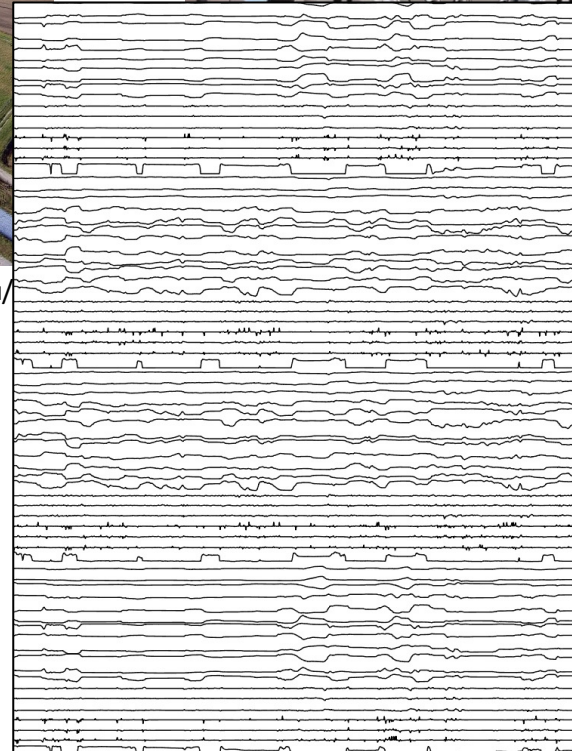
Astrophysics

Fiber-acoustic sensors in the VIRGO north building



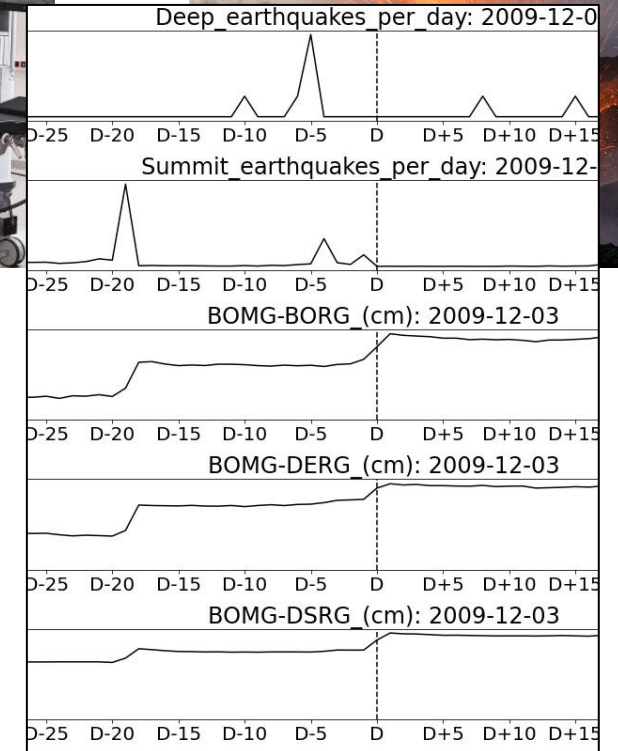
Medicine

Sensor measurements of the Da Vinci surgery robot



Volcanology

Sensor measurements on le Piton de la Fournaise

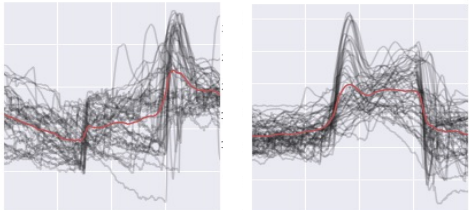


Introduction: *with Important Challenges*

Energy Production

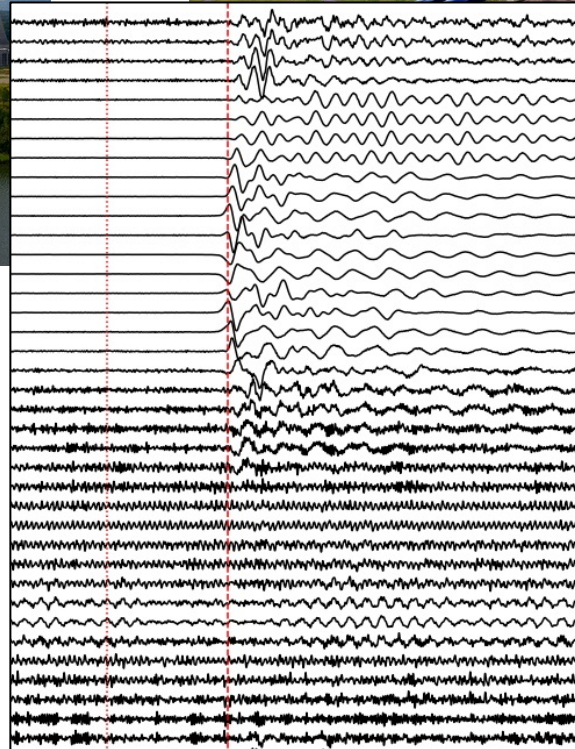
Secondary circuit sensor measurements

Identification of precursors of feed-water pumps vibrations



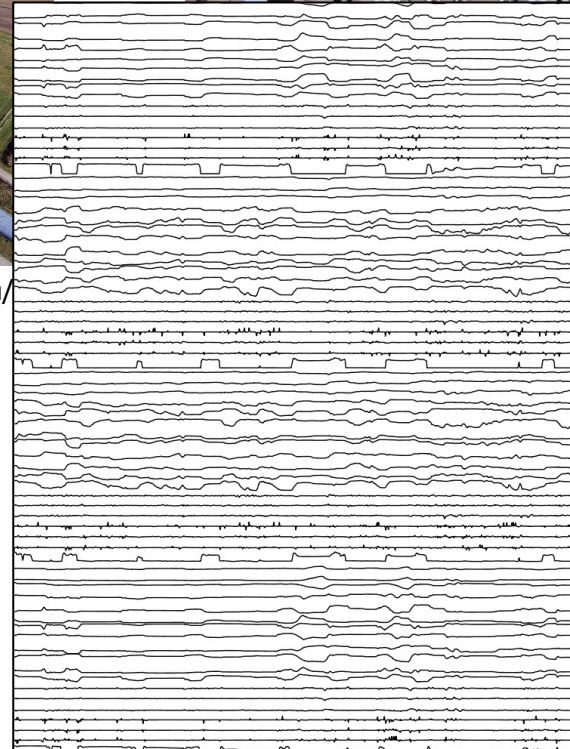
Astrophysics

Fiber-acoustic sensors in the VIRGO north building



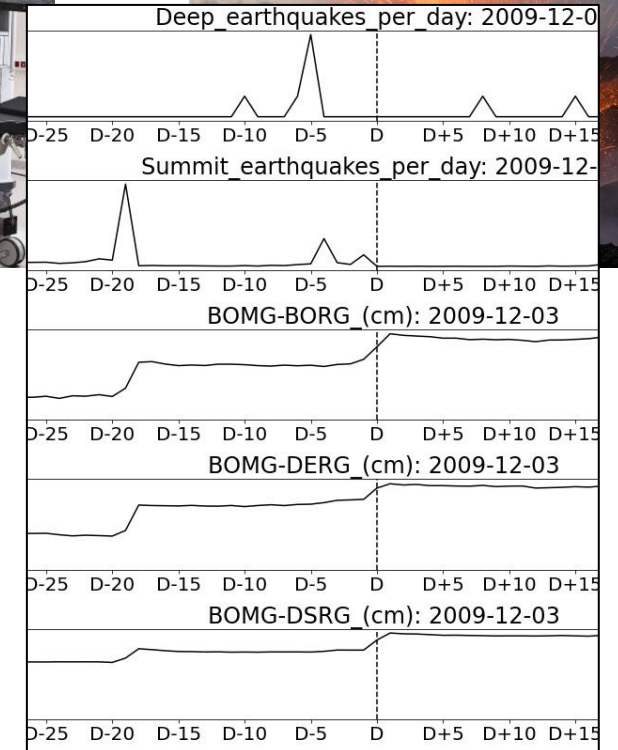
Medicine

Sensor measurements of the Da Vinci surgery robot



Volcanology

Sensor measurements on le Piton de la Fournaise

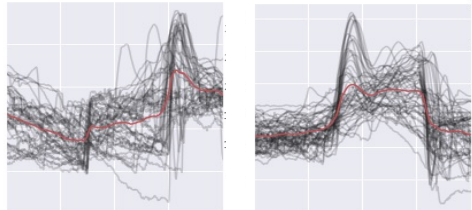


Introduction: *with Important Challenges*

Energy Production

Secondary circuit sensor measurements

Identification of precursors of feed-water pumps vibrations



Astrophysics

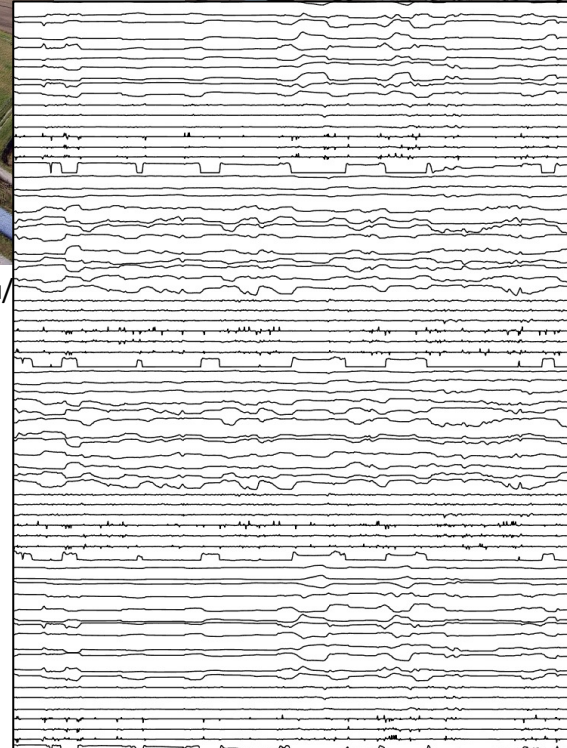
Fiber-acoustic sensors in the VIRGO north building

Noise detection in VIRGO interferometer north building



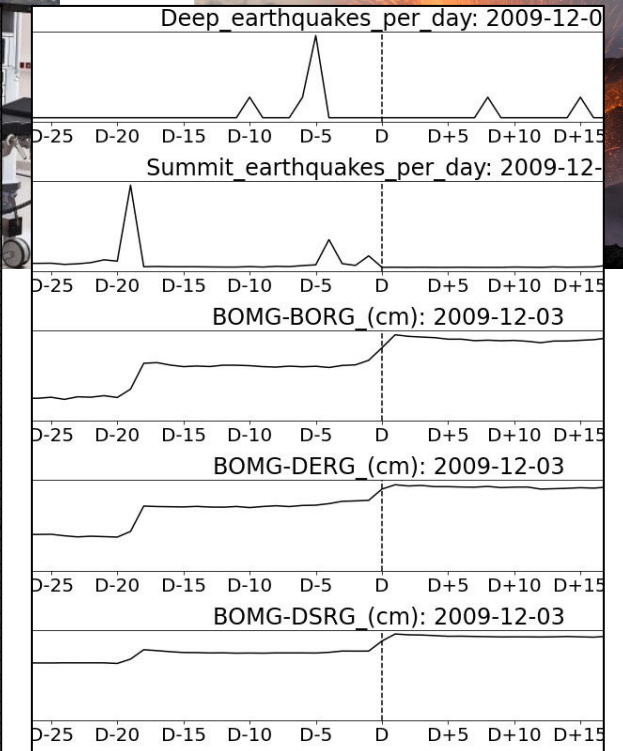
Medicine

Sensor measurements of the Da Vinci surgery robot



Volcanology

Sensor measurements on le Piton de la Fournaise

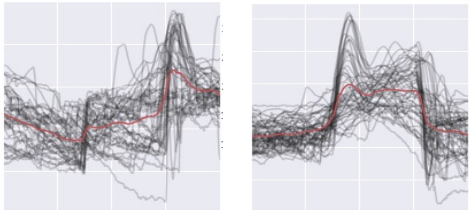


Introduction: *with Important Challenges*

Energy Production

Secondary circuit sensor measurements

Identification of precursors of feed-water pumps vibrations



Astrophysics

Fiber-acoustic sensors in the VIRGO north building

Noise detection in VIRGO interferometer north building



Medicine

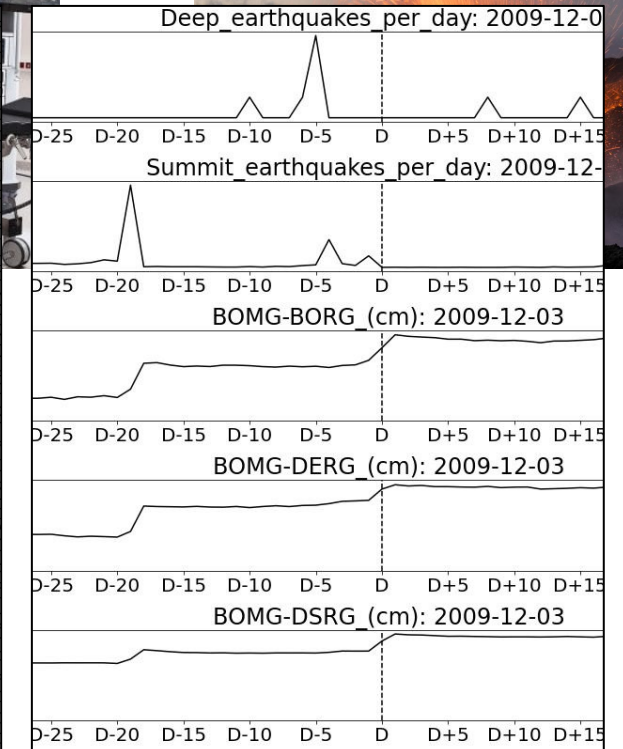
Sensor measurements of the Da-Vinci surgery robot

Unusual surgeons gestures detection



Volcanology

Sensor measurements on le Piton de la Fournaise

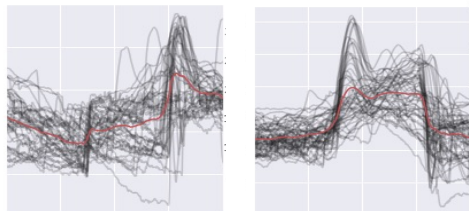


Introduction: *with Important Challenges*

Energy Production

Secondary circuit sensor measurements

Identification of precursors of feed-water pumps vibrations



Astrophysics

Fiber-acoustic sensors in the VIRGO north building

Noise detection in VIRGO interferometer north building



Medicine

Sensor measurements of the Da Vinci surgery robot

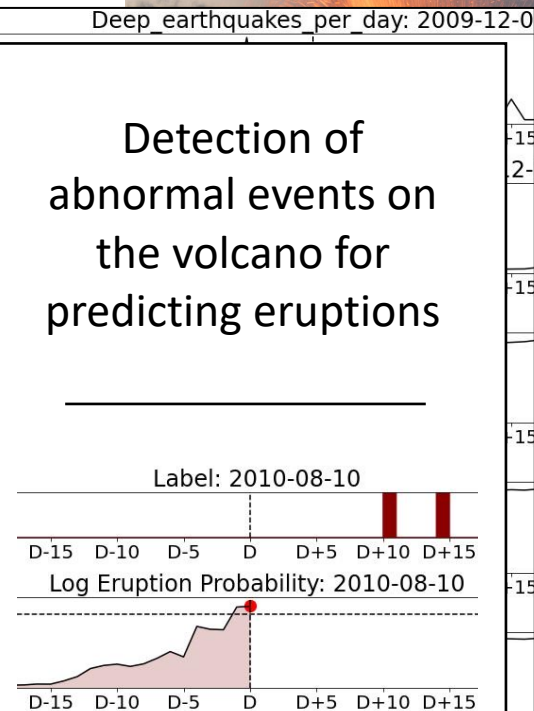
Unusual surgeons gestures detection



Volcanology

Sensor measurements on le Piton de la Fournaise

Detection of abnormal events on the volcano for predicting eruptions



Introduction: *with Important Challenges*

Large-scale time series database

Energy Production

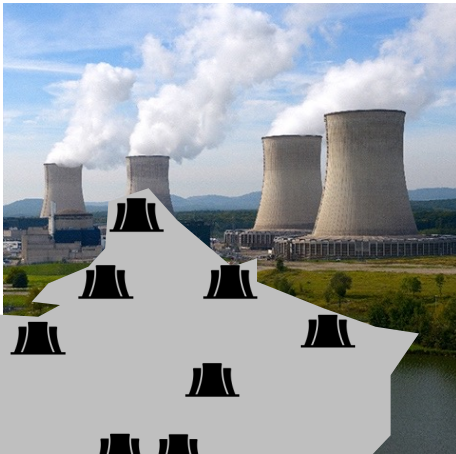


Edf.fr: tinyurl.com/yc7x5xje

Introduction: *with Important Challenges*

Large-scale time series database

Energy Production



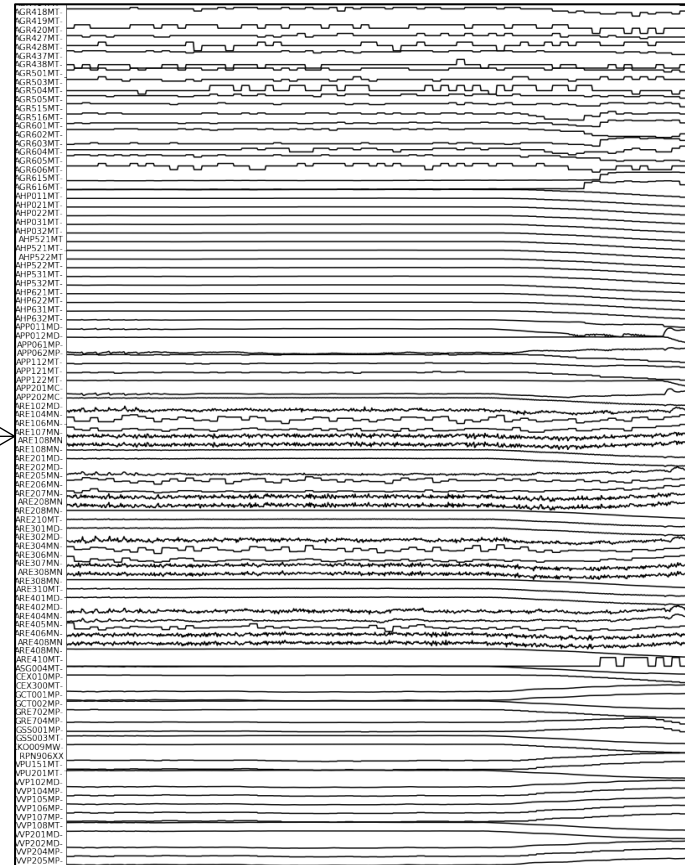
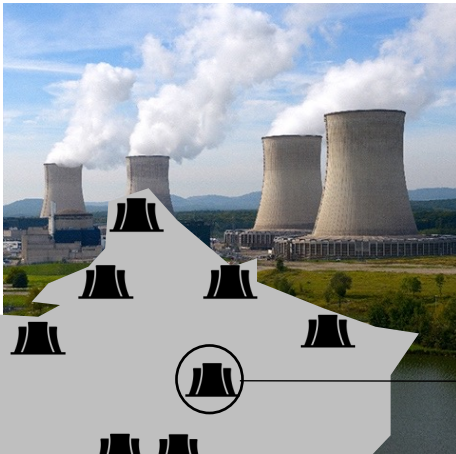
Edf.fr: tinyurl.com/yc7x5xje

Example of Nuclear production

- 58 nuclear power plants across France

Introduction: *with Important Challenges*

Large-scale time series database



Example of Nuclear production

- 58 nuclear power plants across France
- 2000+ sensors per power plant
- 30 years of data collections

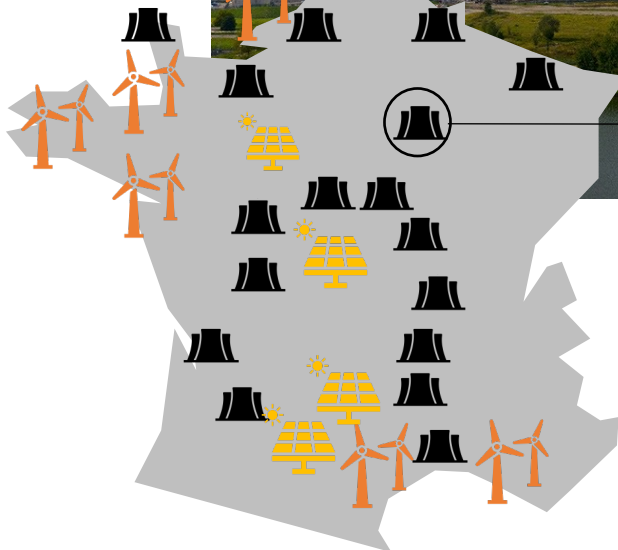
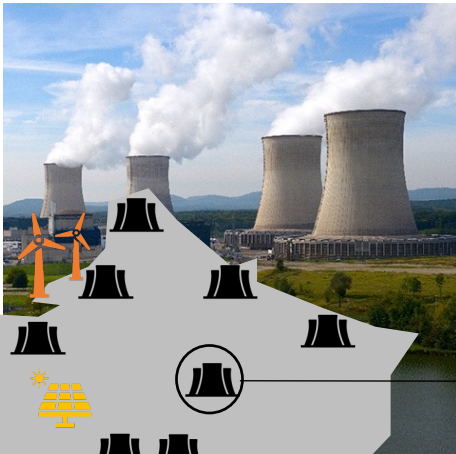
A total of 500 TeraBytes

Edf.fr: tinyurl.com/yc7x5xje

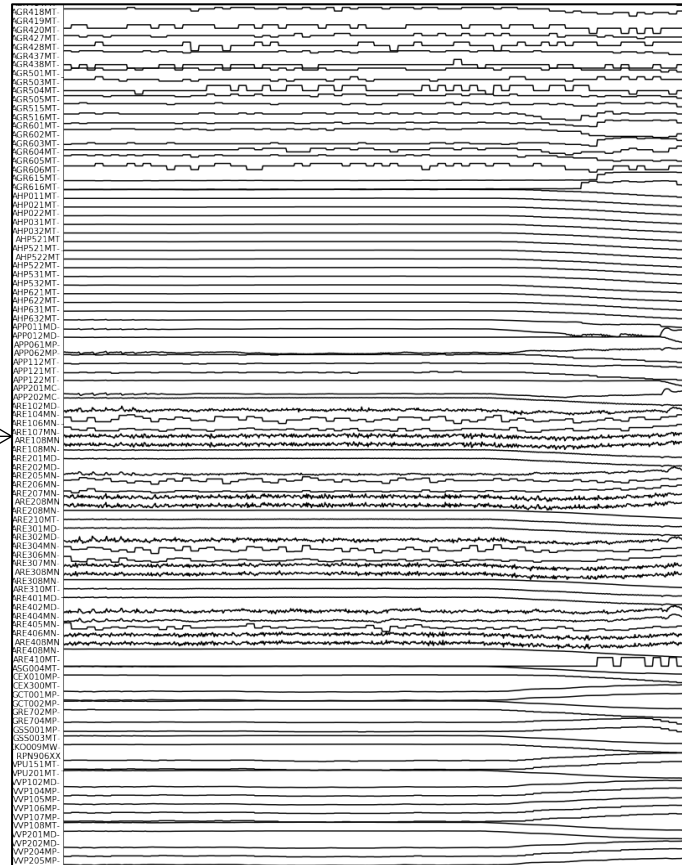
Introduction: *with Important Challenges*

Large-scale time series database

Energy Production



Edf.fr: tinyurl.com/yc7x5xje



Example of Nuclear production

- 58 nuclear power plants across France
- 2000+ sensors per power plant
- 30 years of data collections

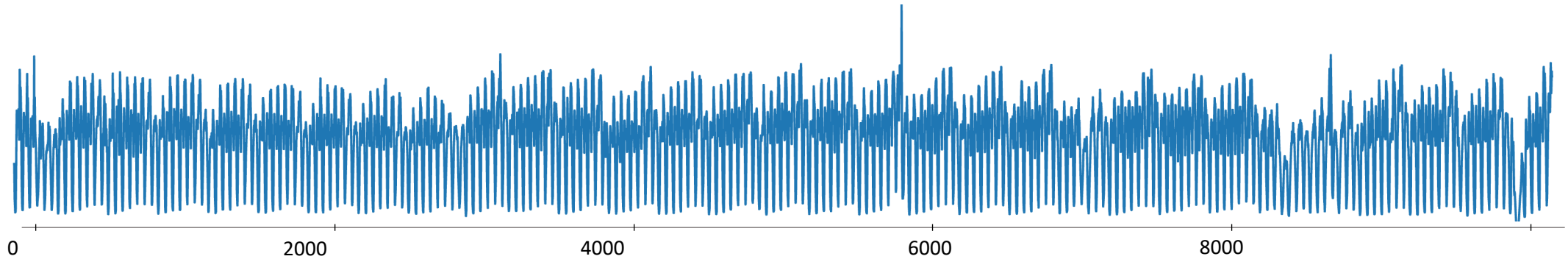
A total of 500 TeraBytes

Other source of production

- New sensors with higher acquisition rate

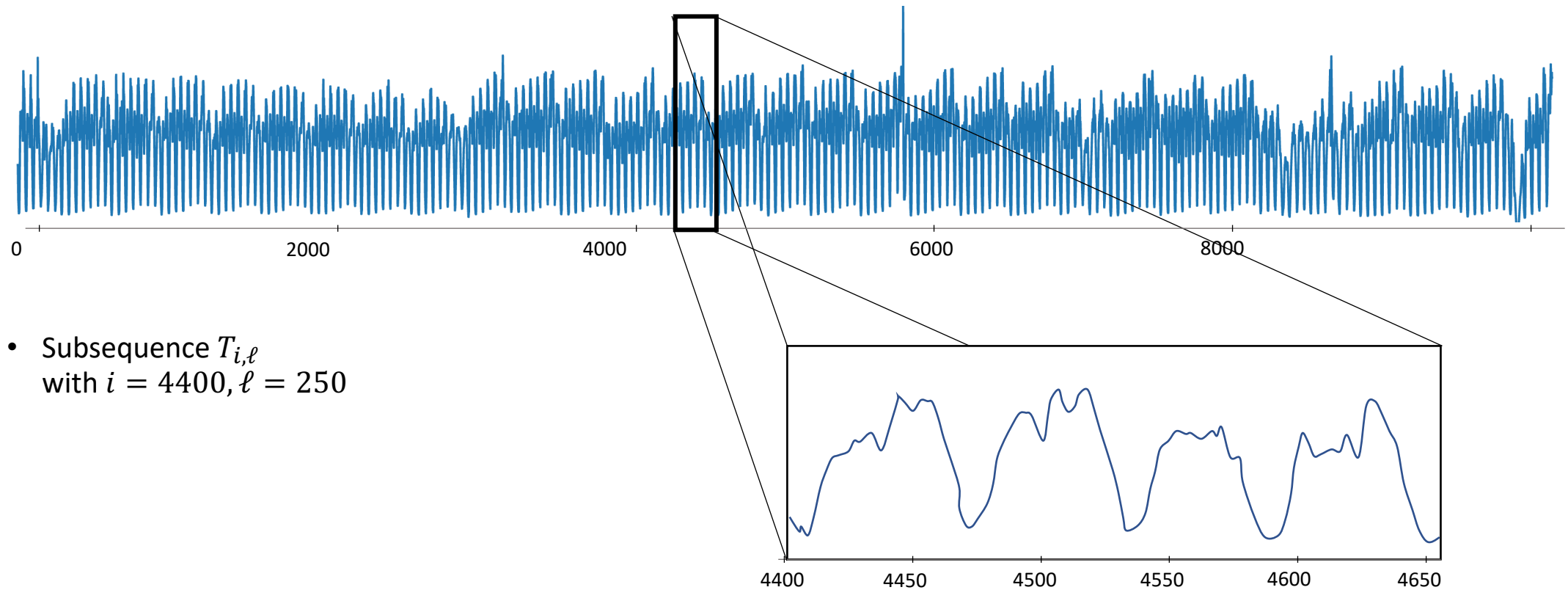
Introduction: *Anomaly Detection in Time Series*

- Time series T (example : number of taxi passengers in New York City)



Introduction: *Anomaly Detection in Time Series*

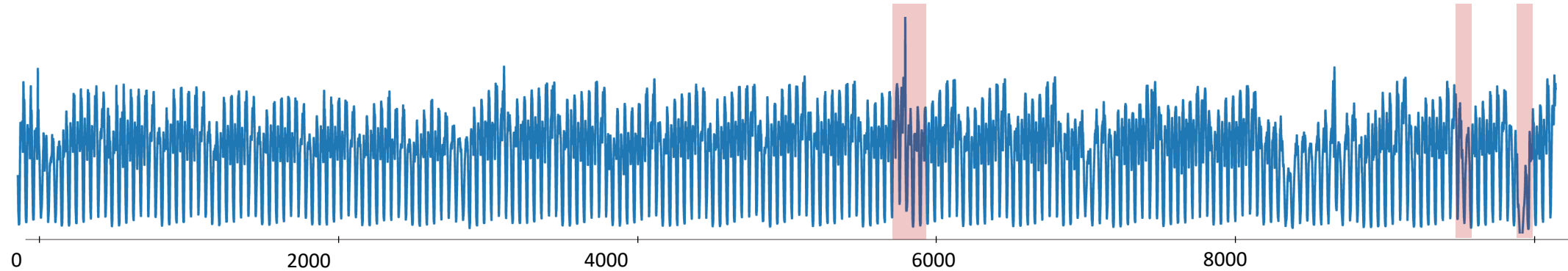
- Time series T (example : number of taxi passengers in New York City)



- Subsequence $T_{i,\ell}$
with $i = 4400, \ell = 250$

Introduction: *Anomaly Detection in Time Series*

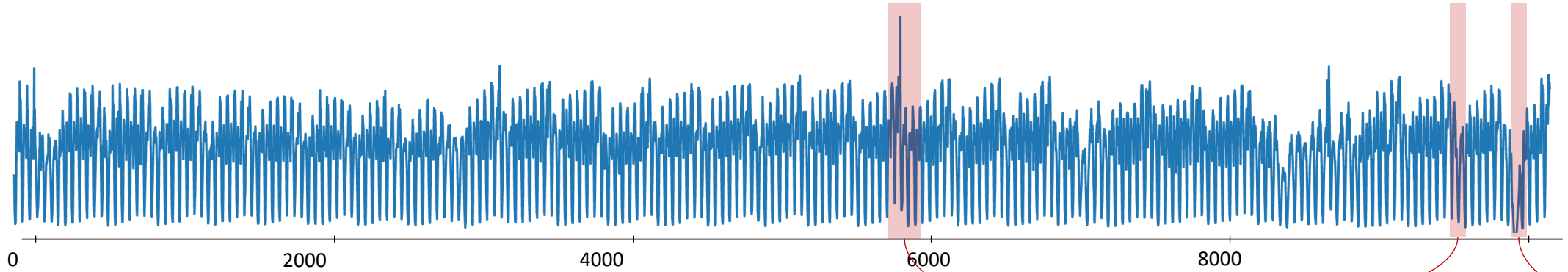
- Time series T (example : number of taxi passengers in New York City)



- Anomaly: *rare* point or sequence (of a given length)
potentially *non-desired*

Introduction: *Anomaly Detection in Time Series*

- Time series T (example : number of taxi passengers in New York City)



- Anomaly: *rare* point or sequence (of a given length) potentially *non-desired*

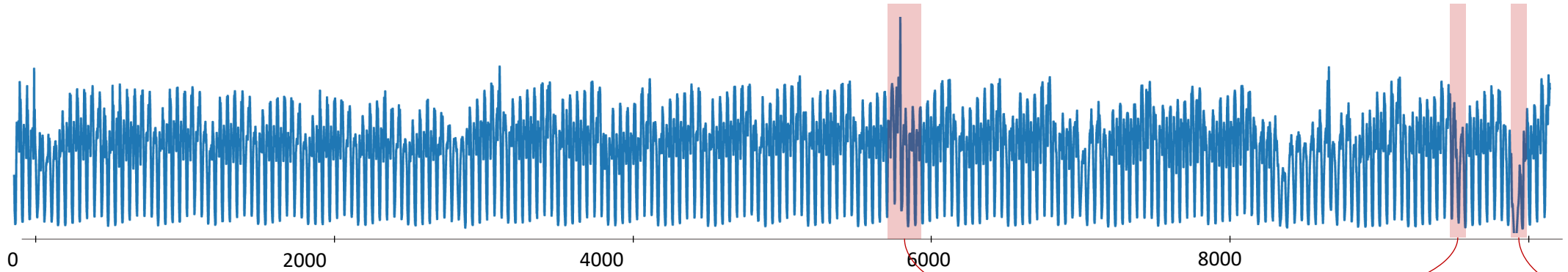
Daylight
Saving Time
(DST)

Flooding

Snowstorm

Introduction: *Anomaly Detection in Time Series*

- Time series T (example : number of taxi passengers in New York City)

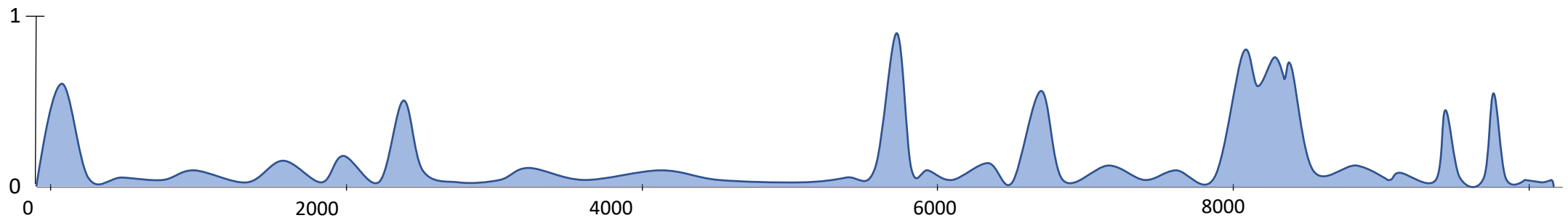


- Anomaly: *rare* point or sequence (of a given length) potentially *non-desired*

Daylight
Saving Time
(DST)

Flooding

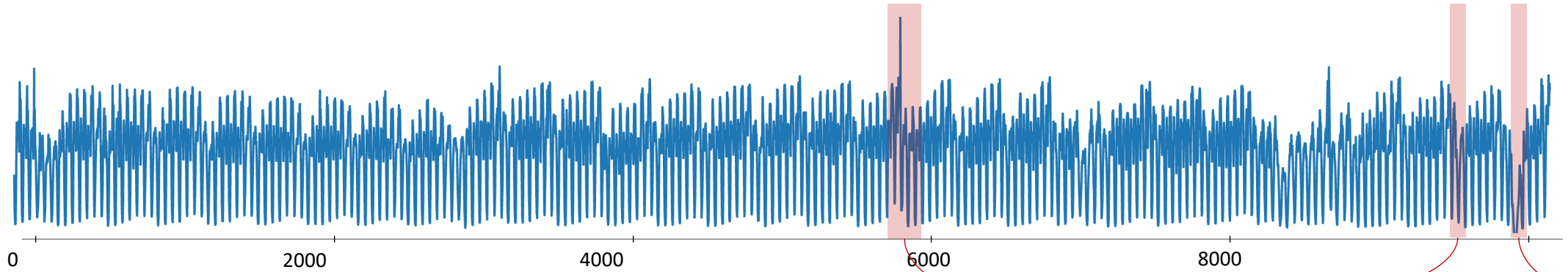
Snowstorm



Anomaly score S_T

Introduction: *Anomaly Detection in Time Series*

- Time series T (example : number of taxi passengers in New York City)

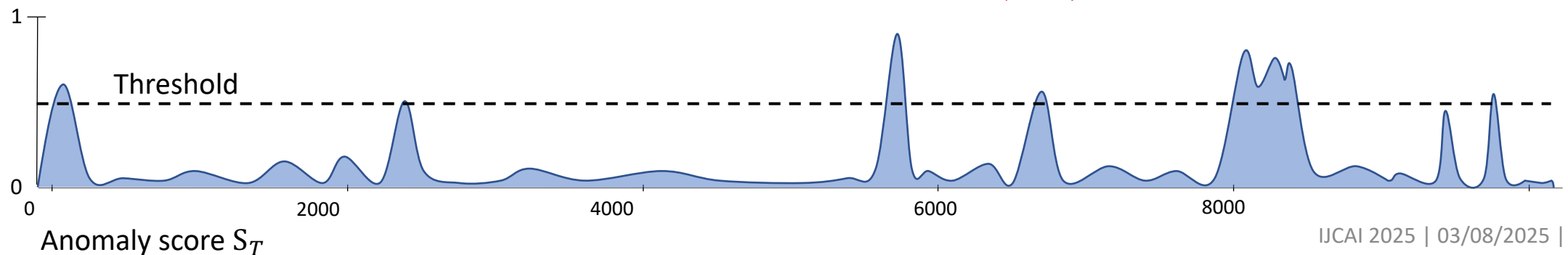


- Anomaly: *rare* point or sequence (of a given length) potentially *non-desired*

Daylight
Saving Time
(DST)

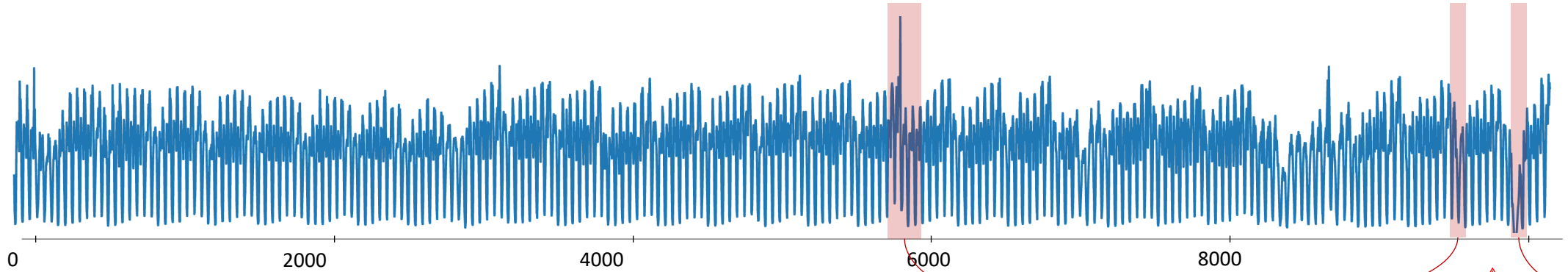
Flooding

Snowstorm

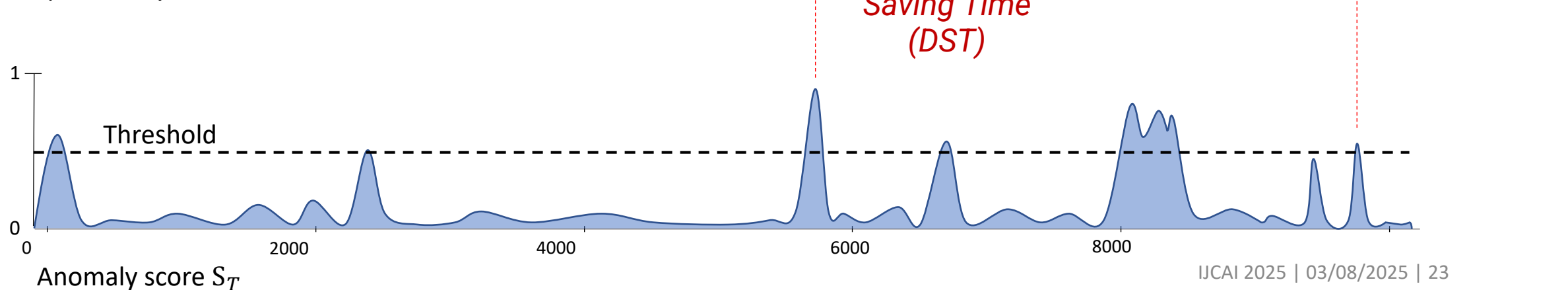


Introduction: *Anomaly Detection in Time Series*

- Time series T (example : number of taxi passengers in New York City)

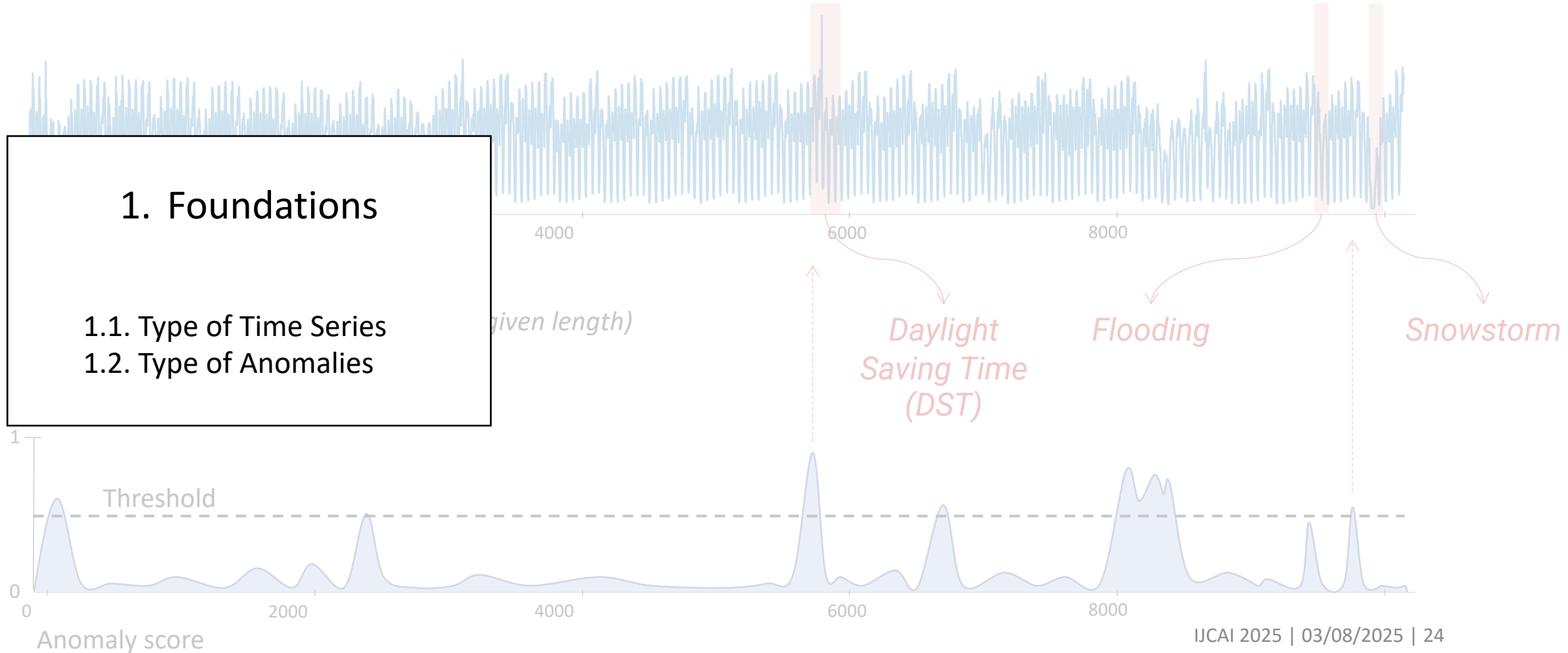


- Anomaly: *rare* point or sequence (of a given length) potentially *non-desired*



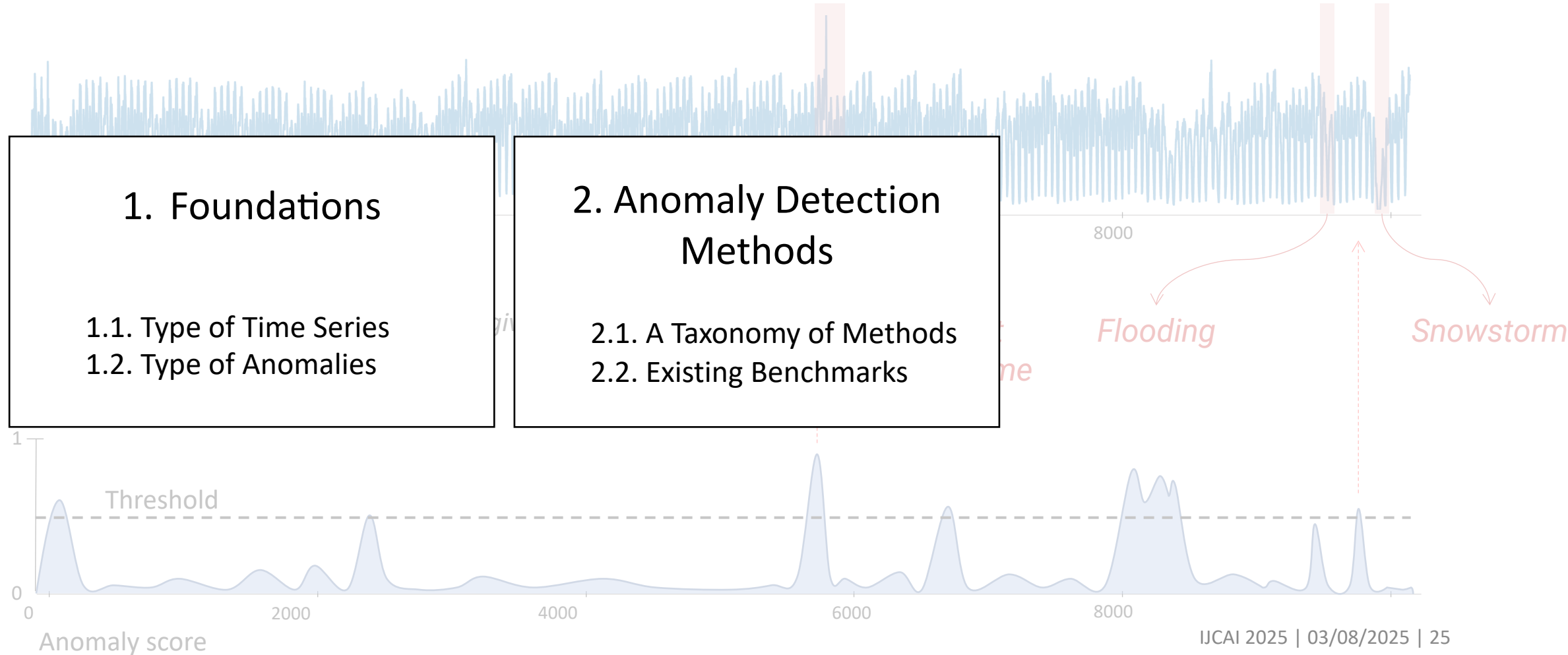
Introduction: *Outline*

- Time series (example : number of taxi passengers in New York City)



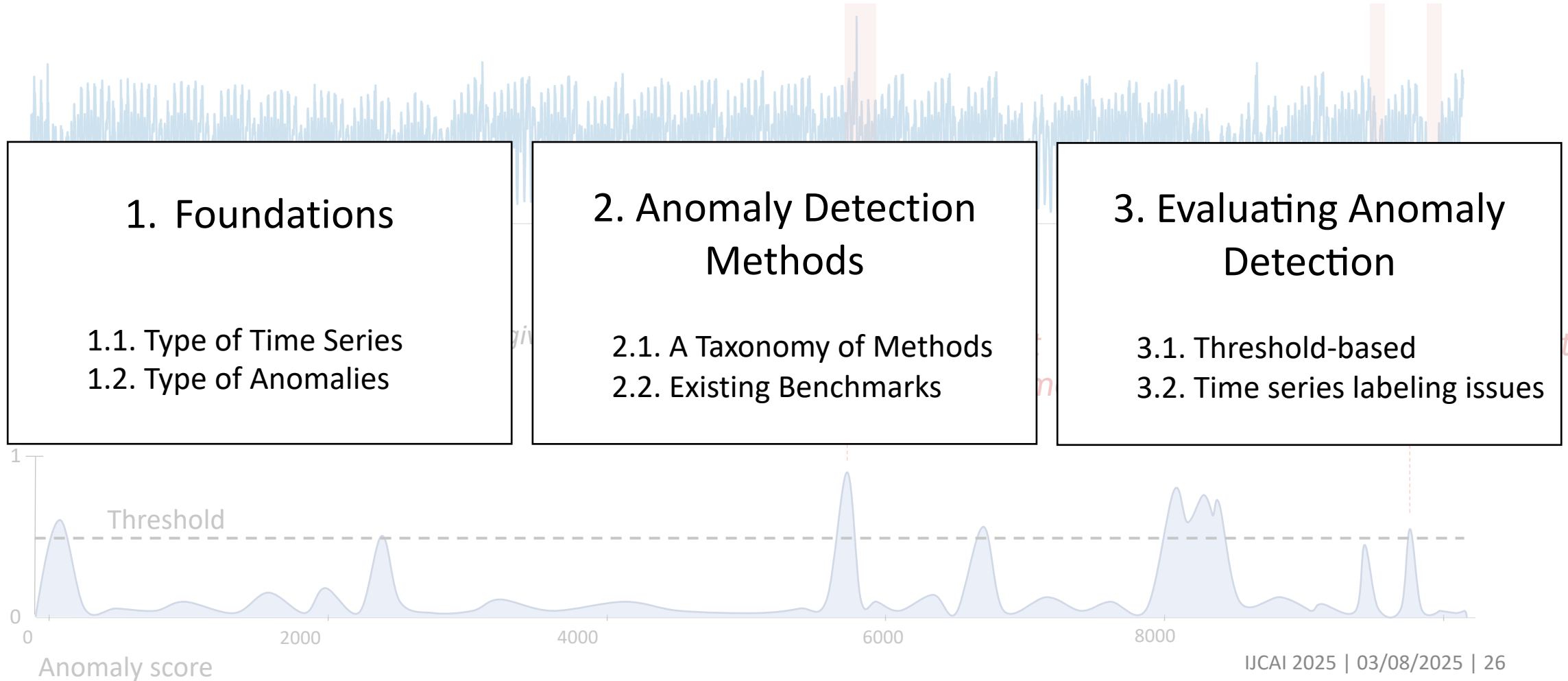
Introduction: *Outline*

- Time series (example : number of taxi passengers in New York City)

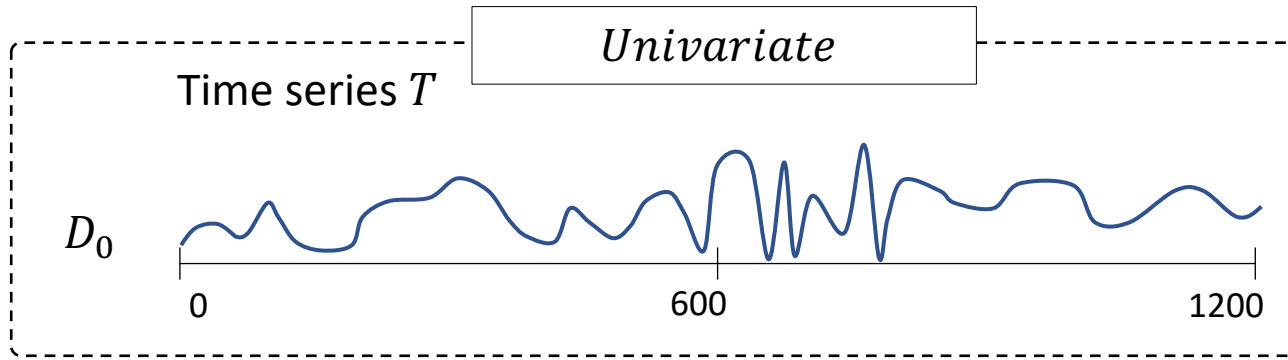


Introduction: *Outline*

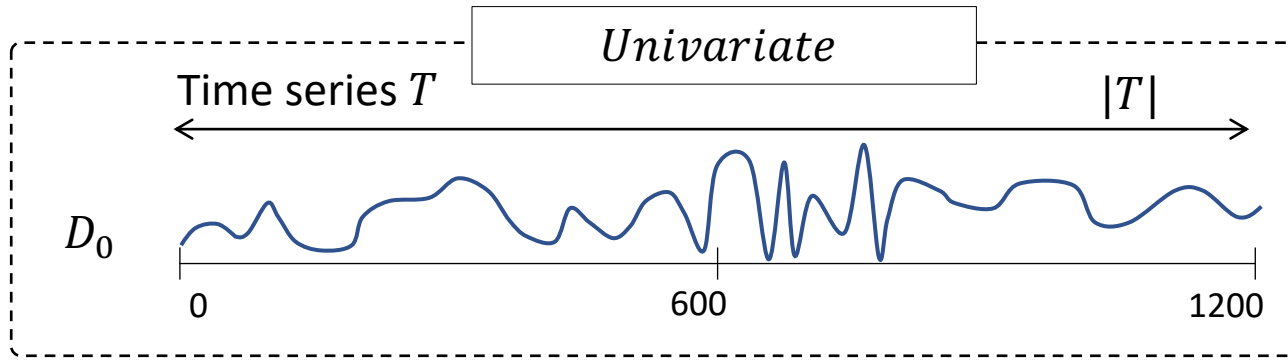
- Time series (example : number of taxi passengers in New York City)



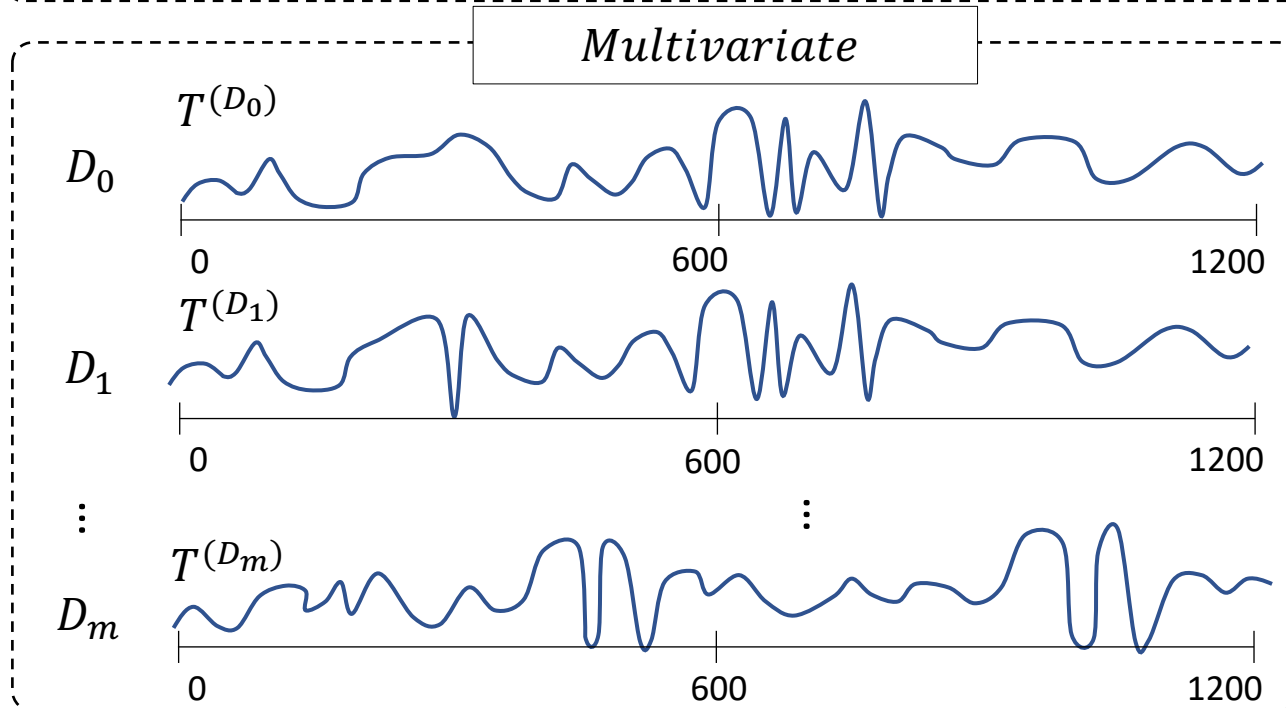
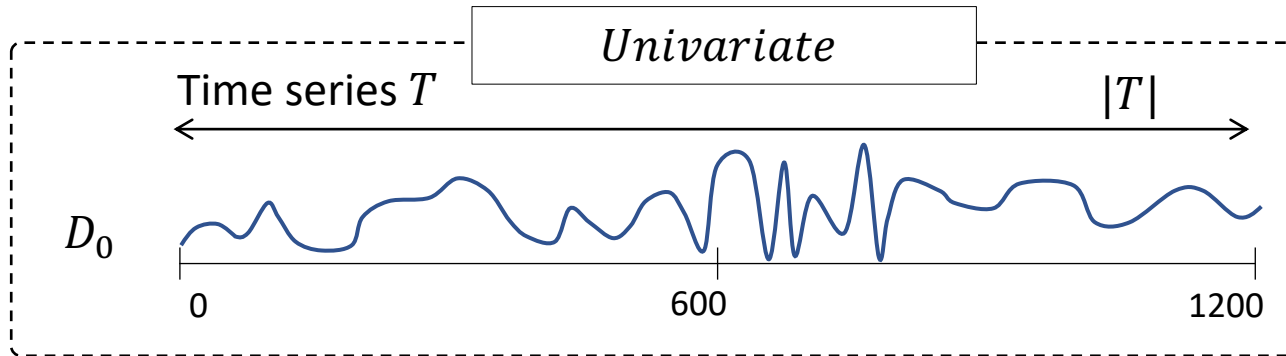
Foundations: *Type of time series*



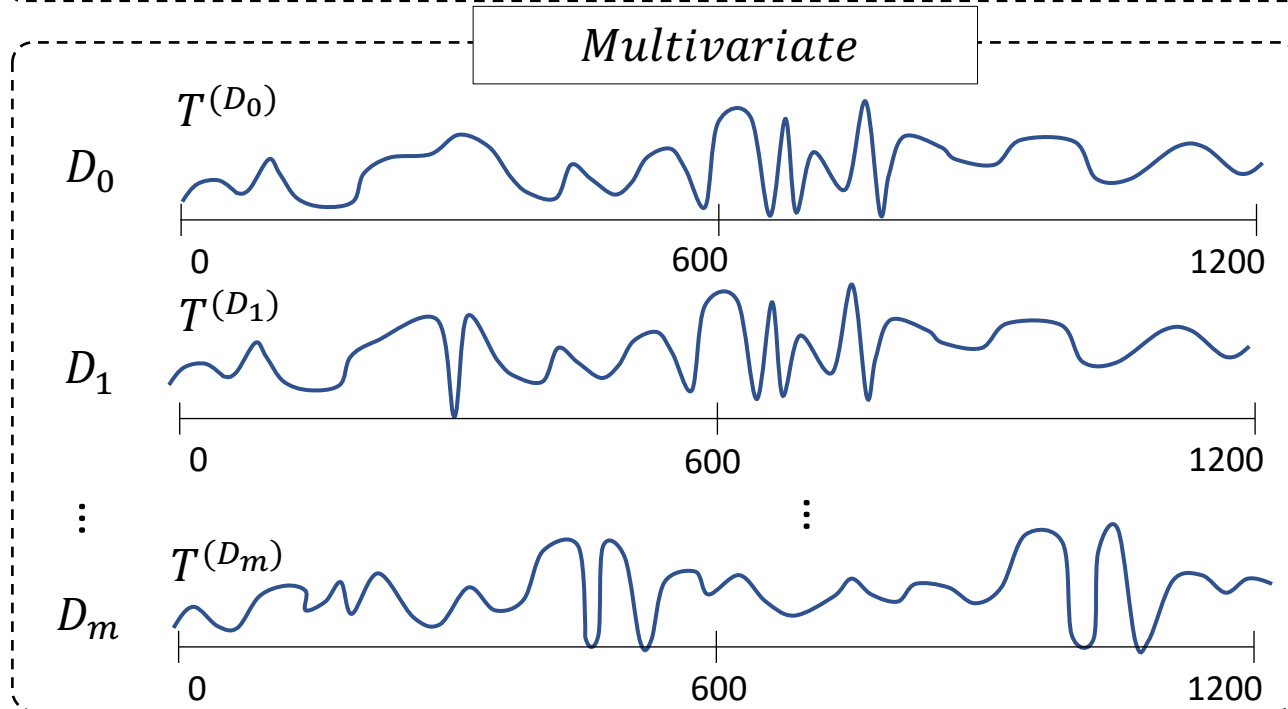
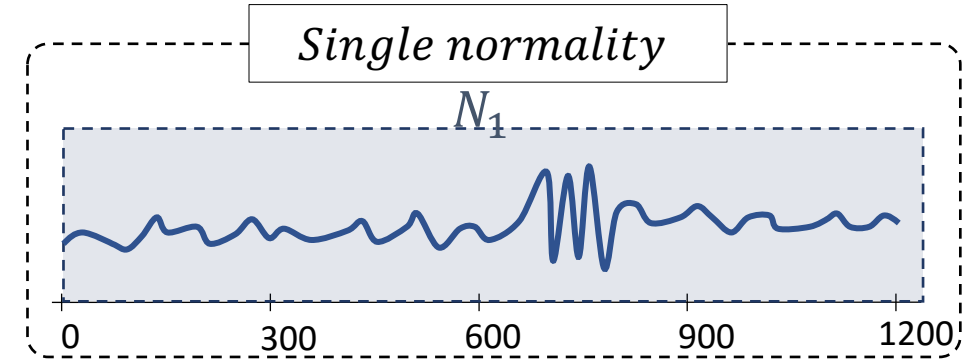
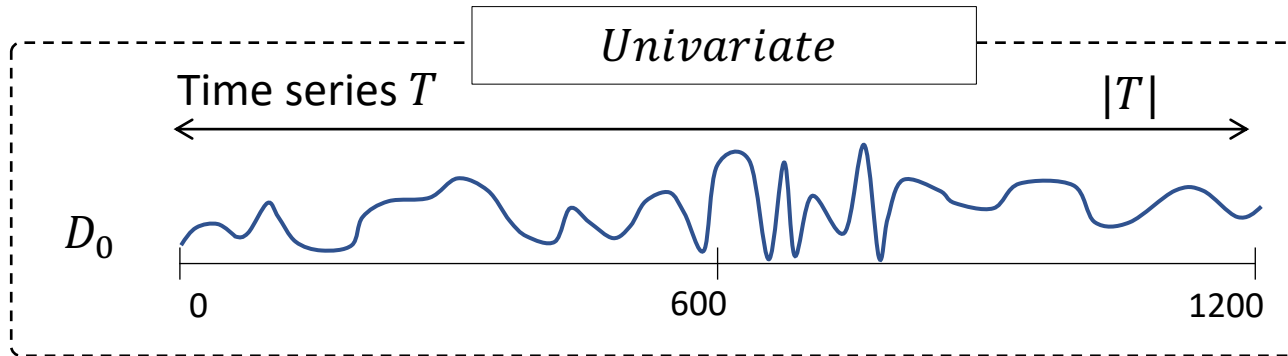
Foundations: *Type of time series*



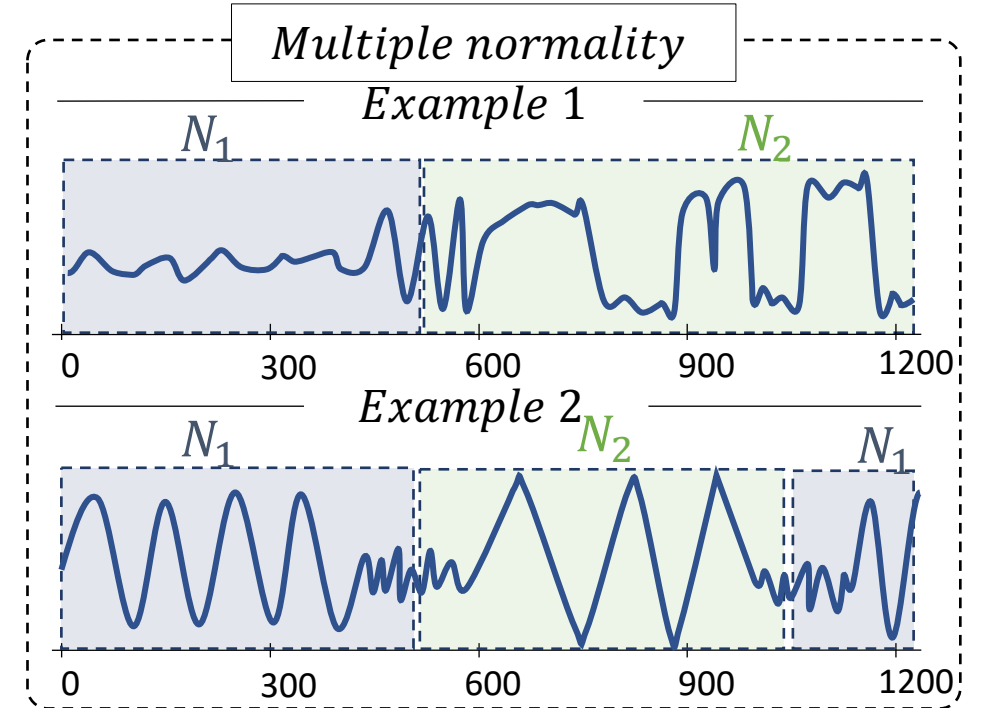
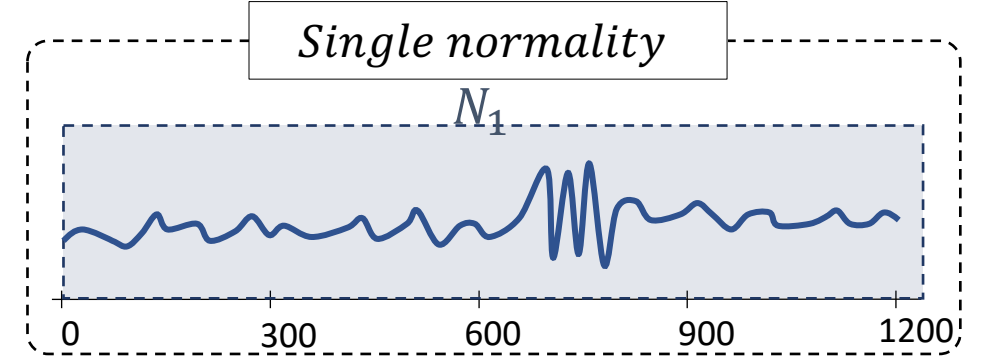
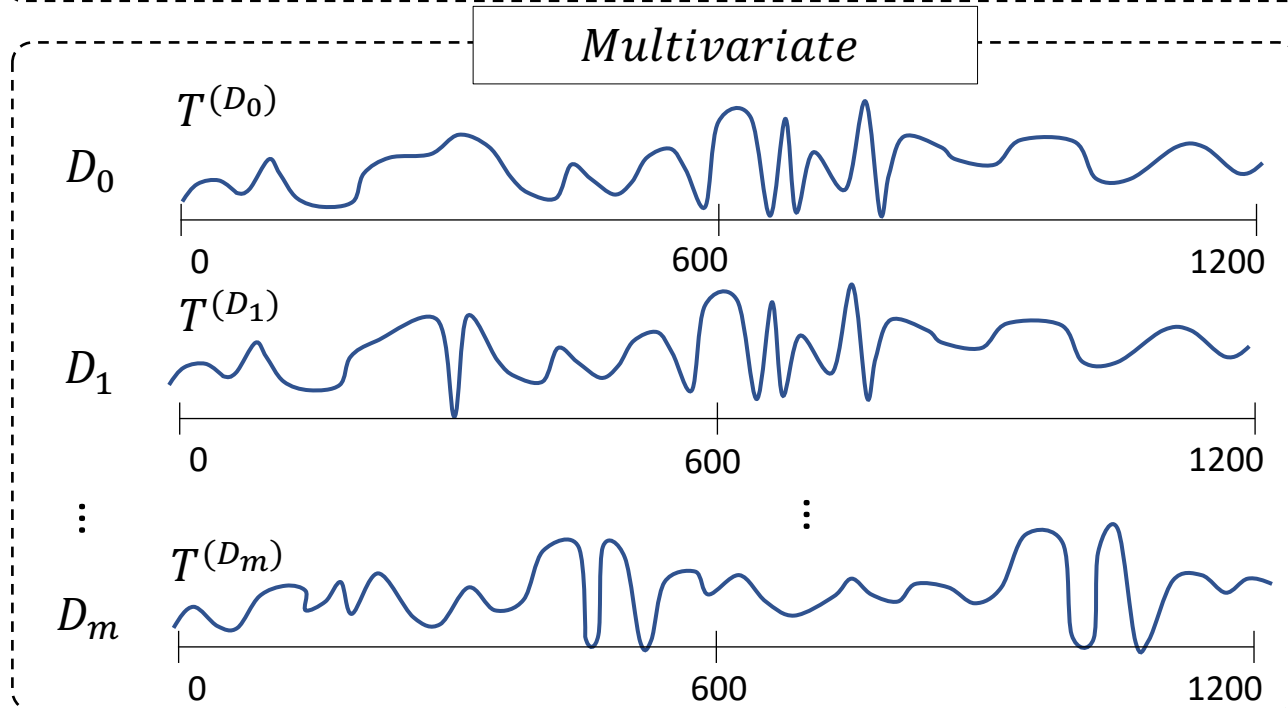
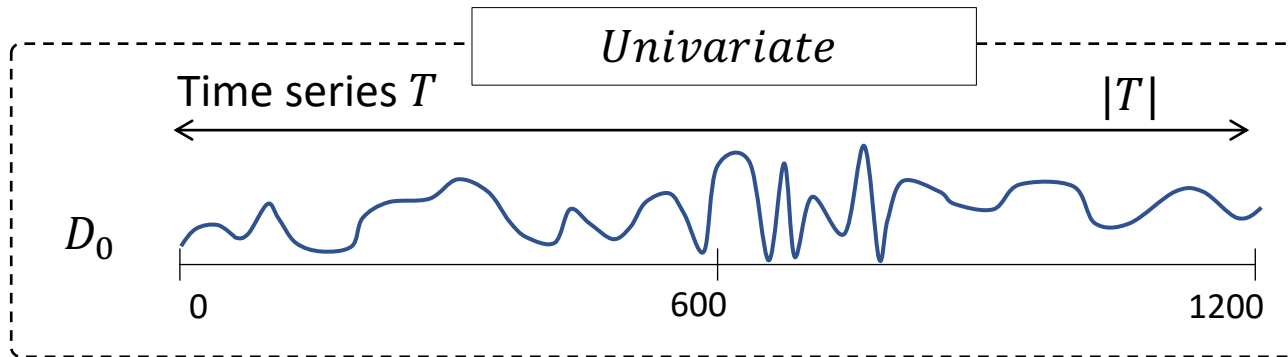
Foundations: *Type of time series*



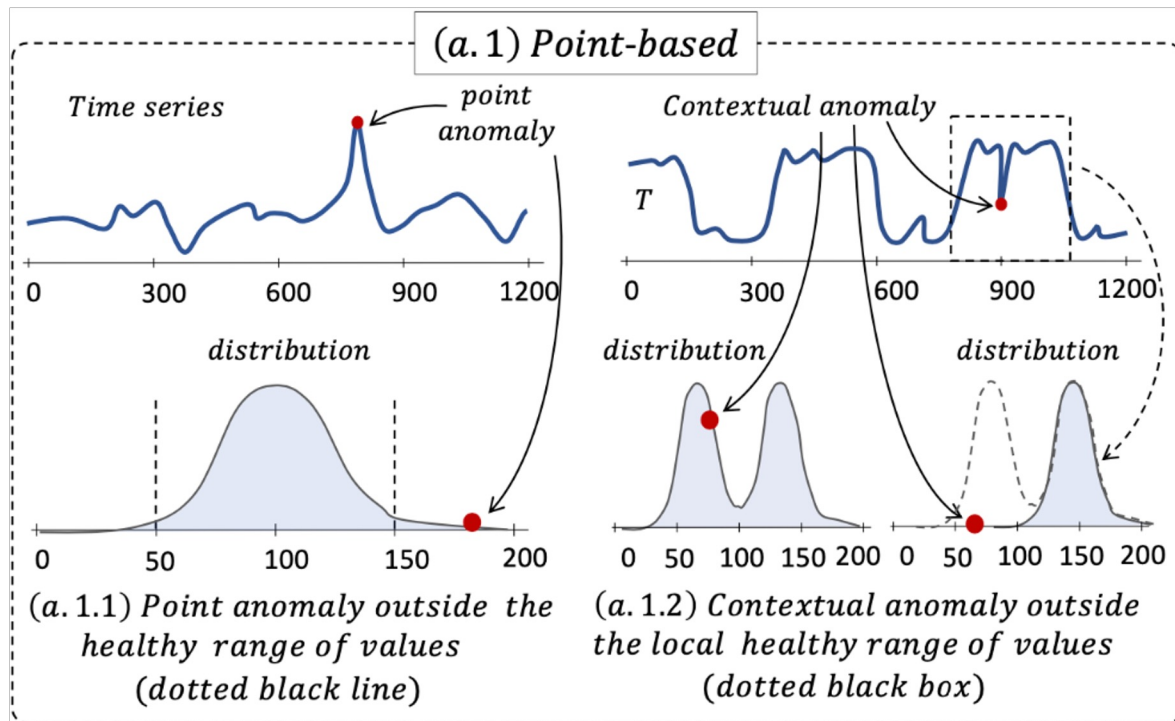
Foundations: *Type of time series*



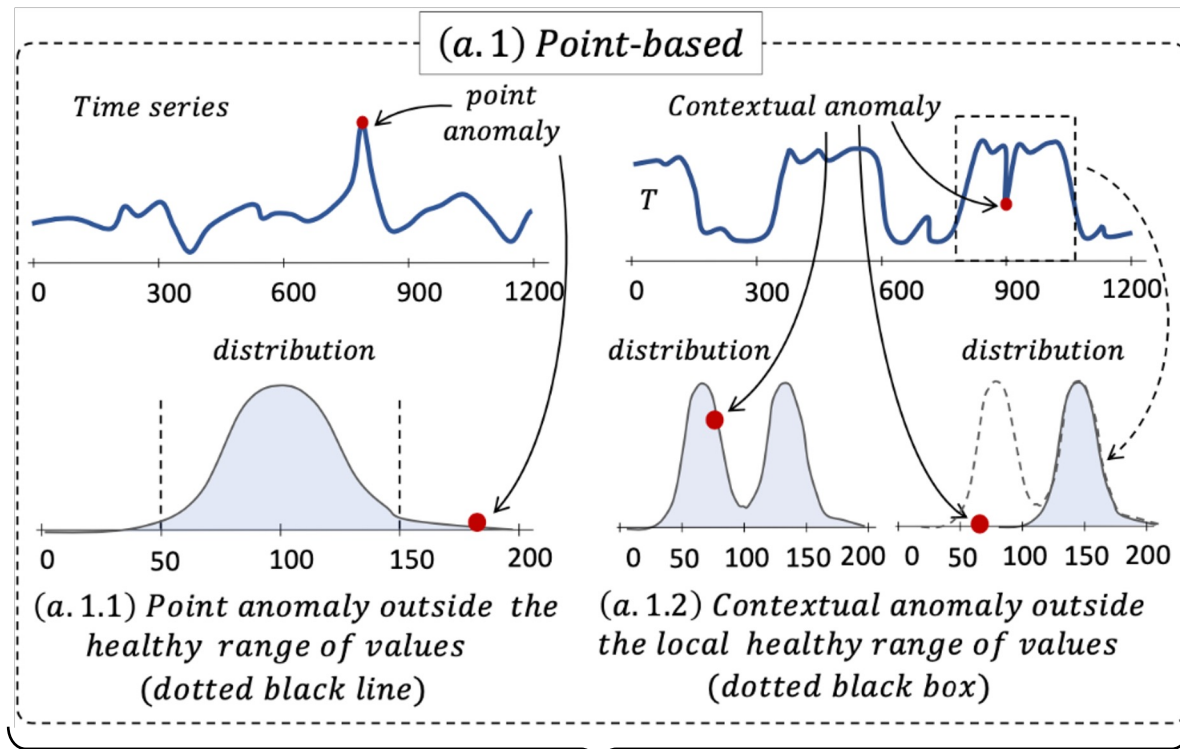
Foundations: *Type of time series*



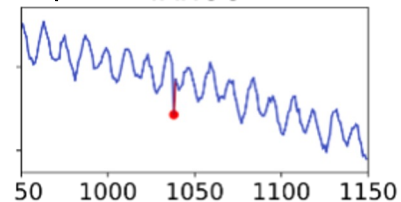
Foundations: *Type of anomalies*



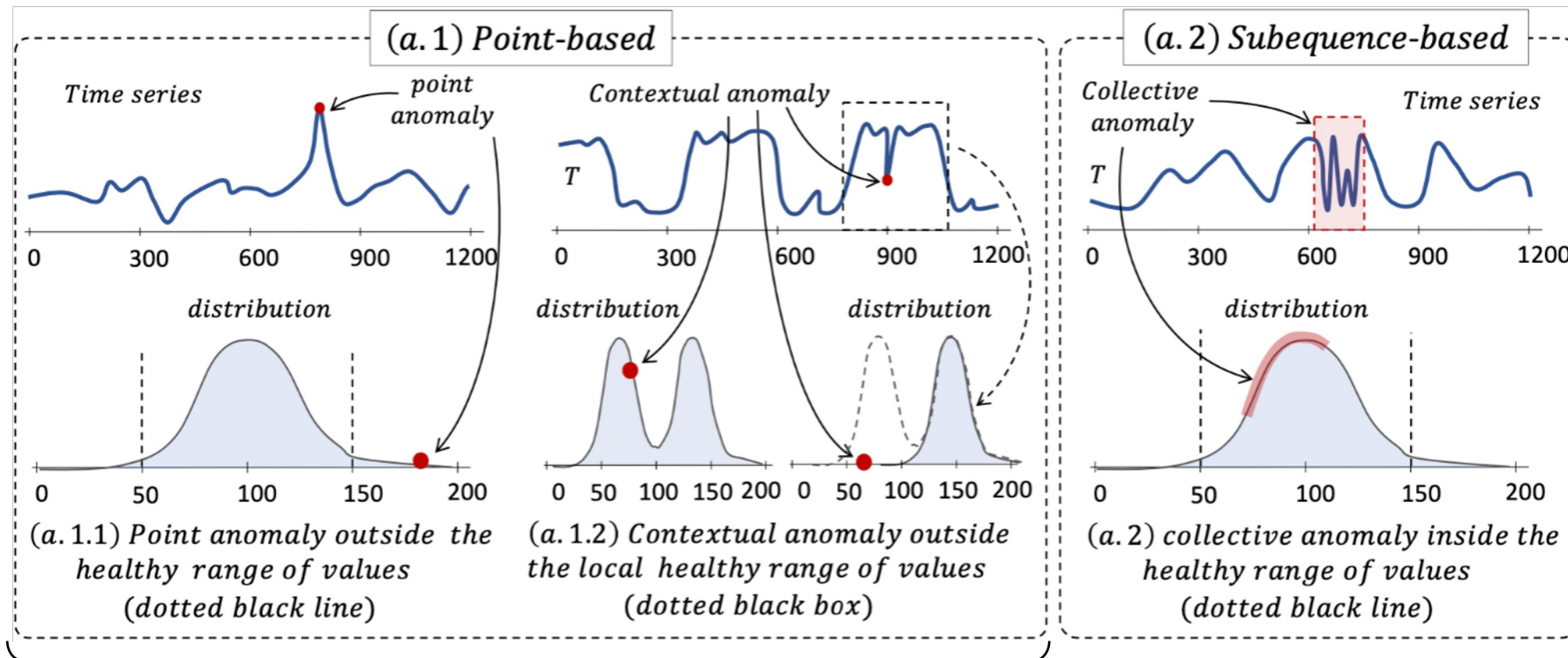
Foundations: *Type of anomalies*



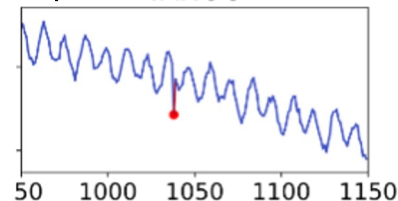
Example of
point-based
anomaly [1]



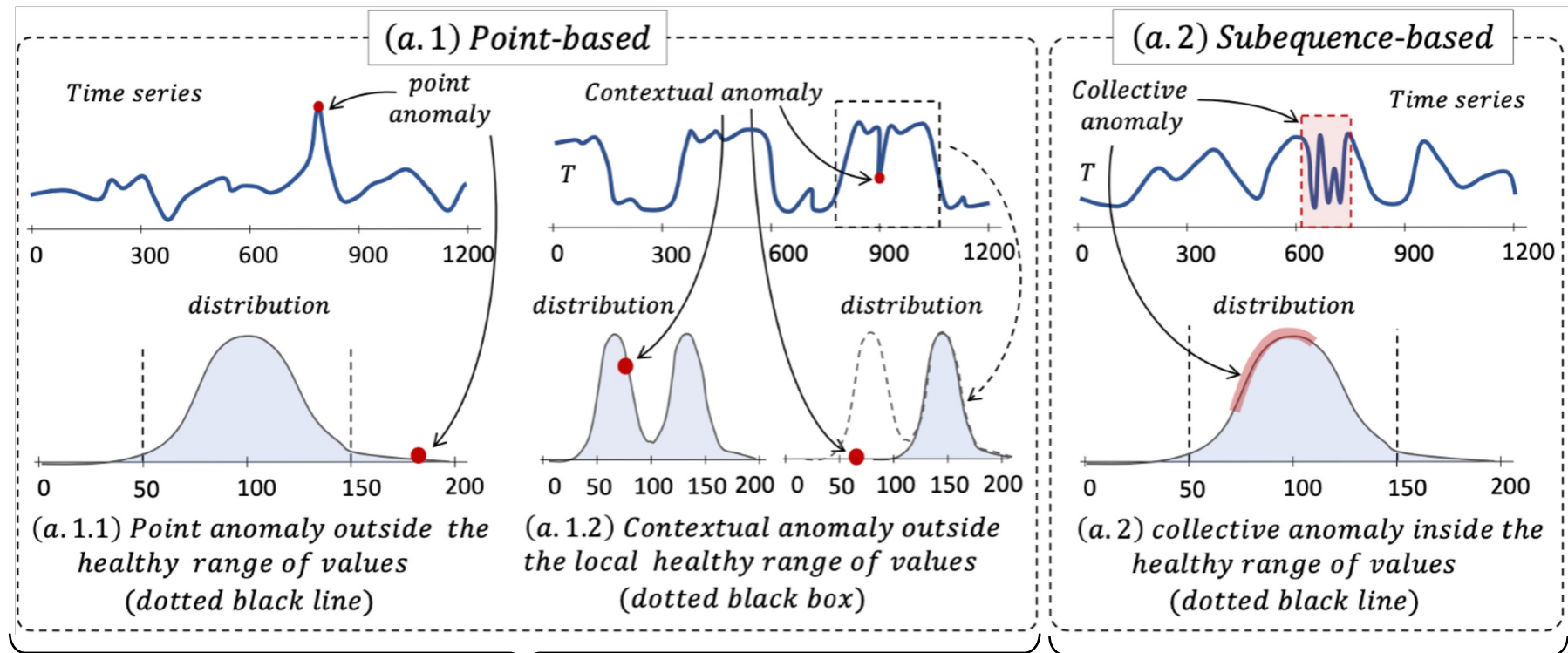
Foundations: *Type of anomalies*



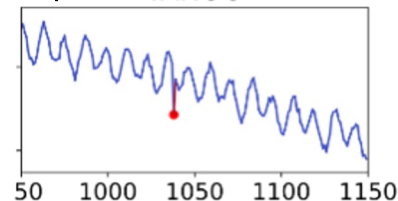
Example of
point-based
anomaly [1]



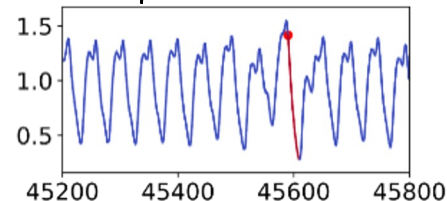
Foundations: *Type of anomalies*



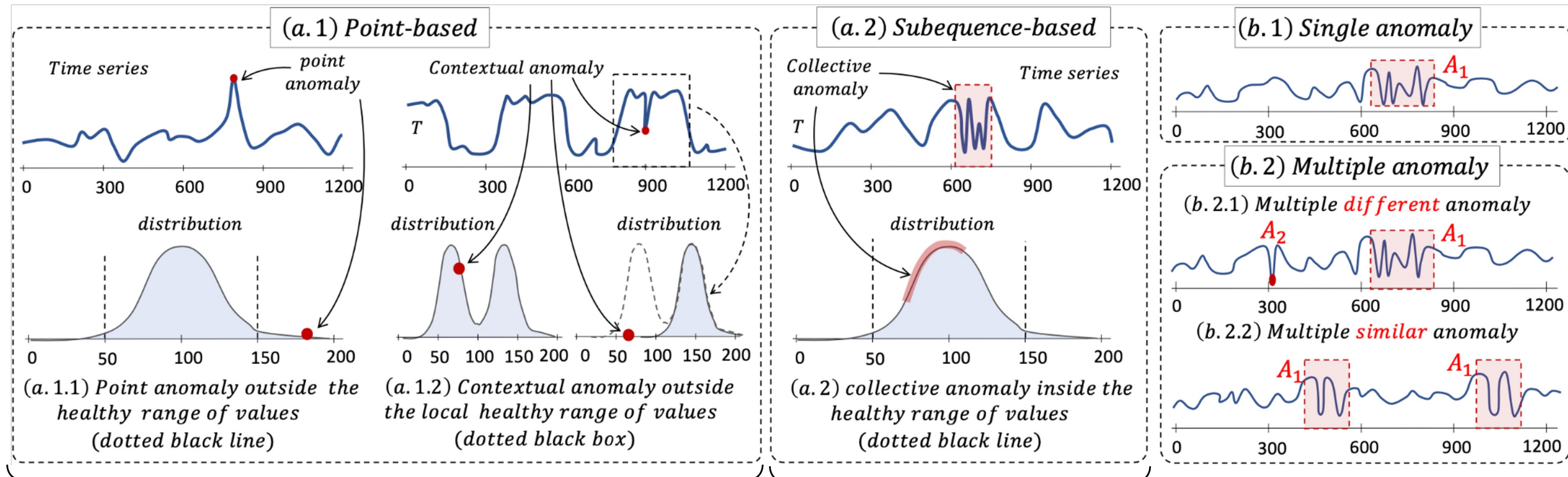
Example of
point-based
anomaly [1]



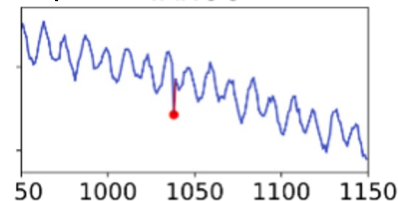
Example of
subsequence-
based anomaly [2]



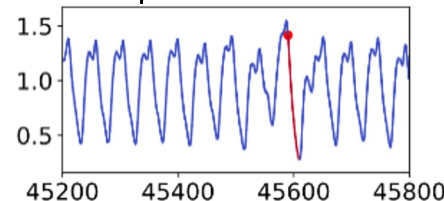
Foundations: *Type of anomalies*



Example of point-based anomaly [1]

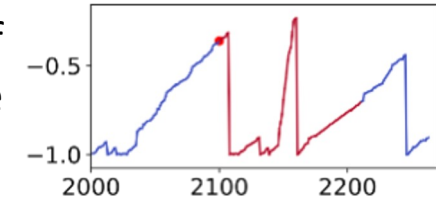


Example of subsequence-based anomaly [2]

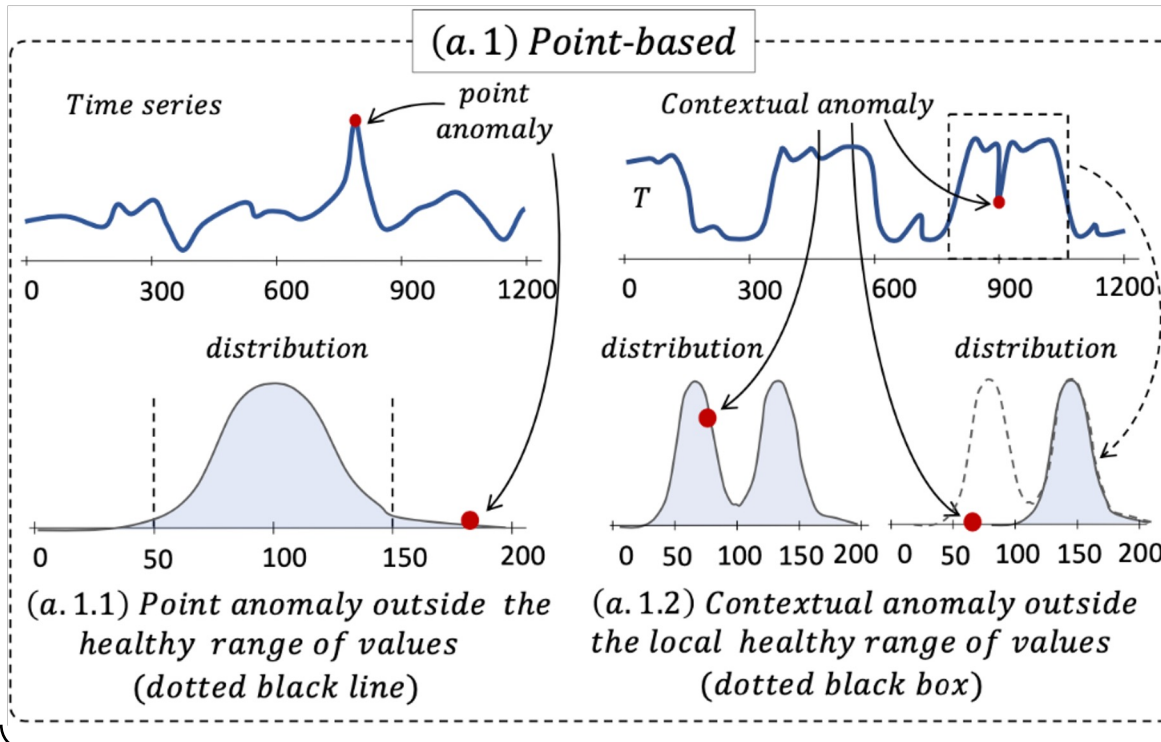


Foundations: *Type of anomalies*

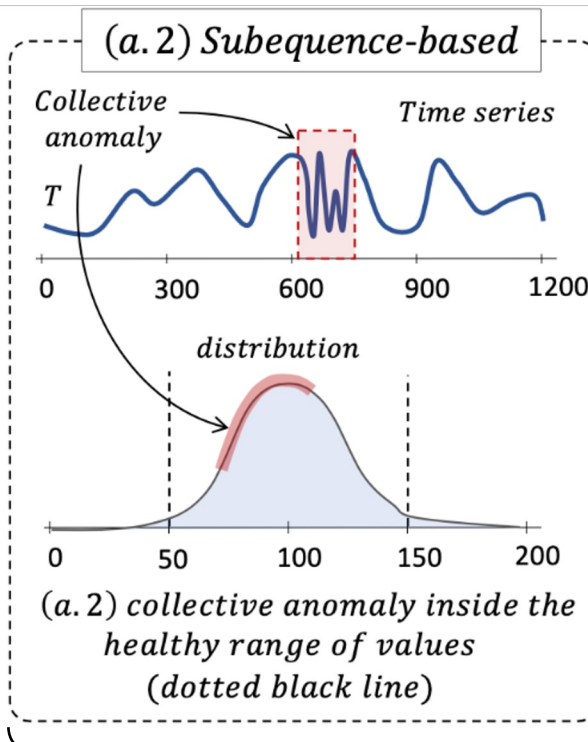
Example of
single
anomaly [3]



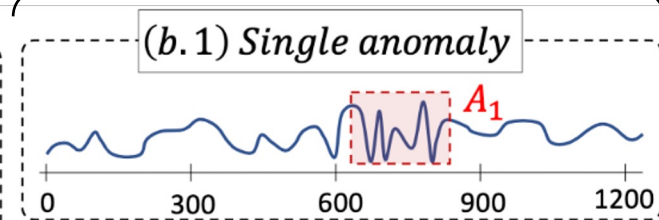
(a.1) *Point-based*



(a.2) *Subsequence-based*

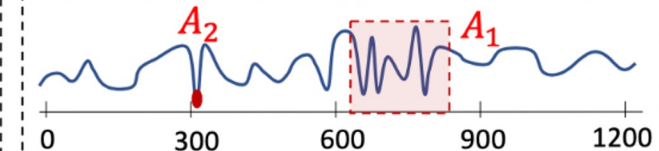


(b.1) *Single anomaly*

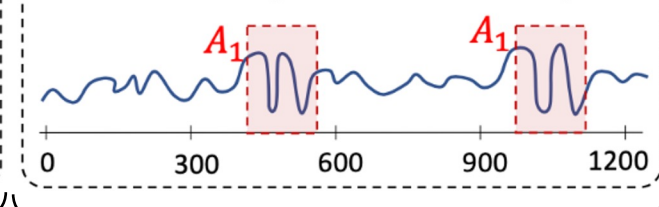


(b.2) *Multiple anomaly*

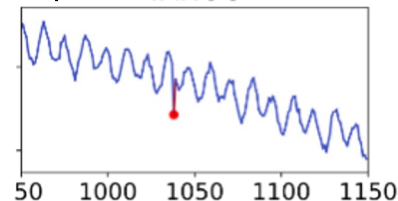
(b.2.1) Multiple *different* anomaly



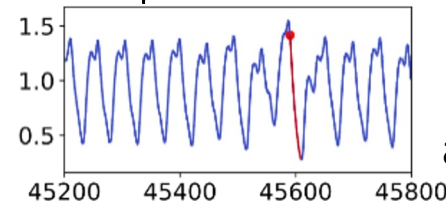
(b.2.2) Multiple *similar* anomaly



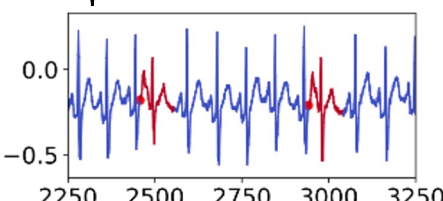
Example of
point-based
anomaly [1]



Example of
subsequence-
based anomaly [2]

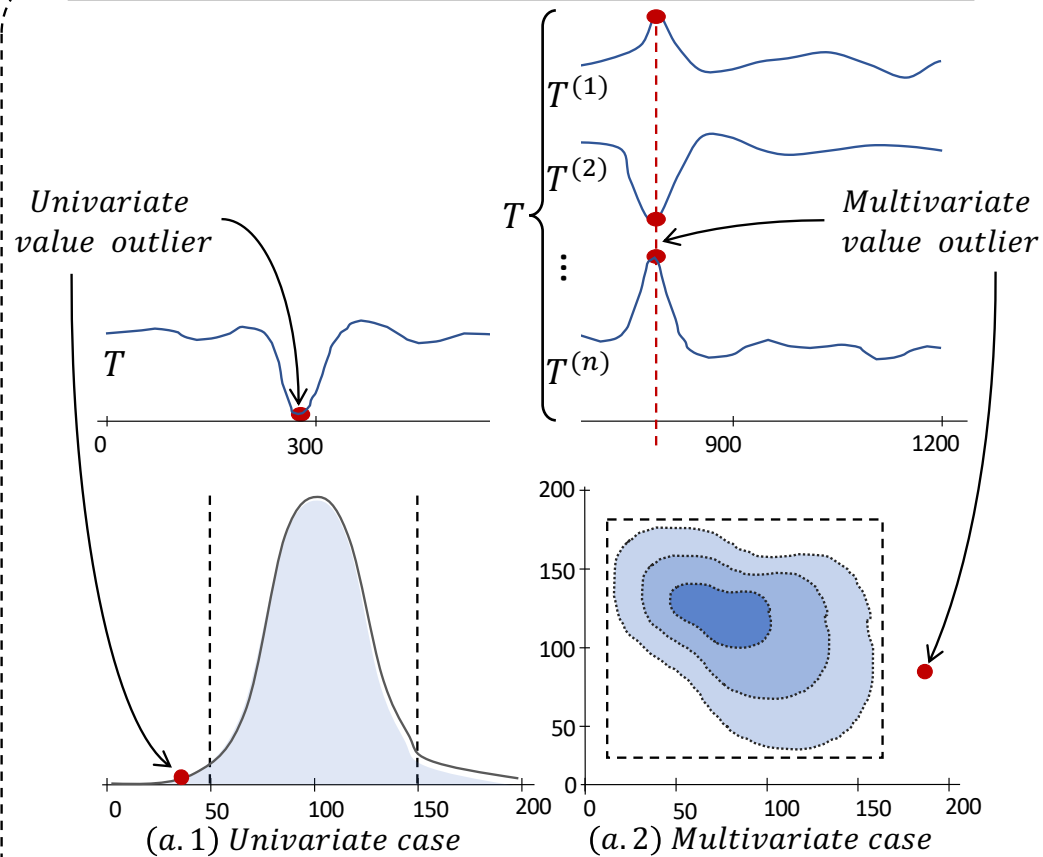


Example of
multiple
anomaly [4]



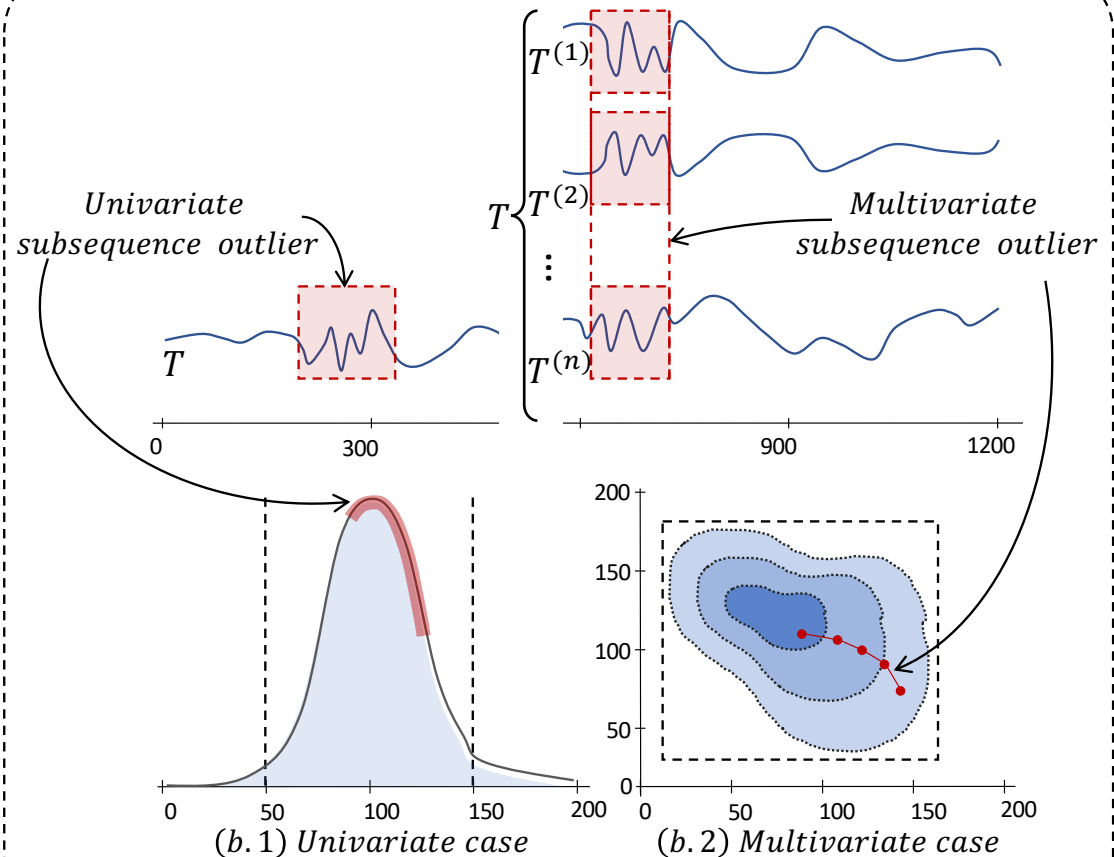
Foundations: *Type of anomalies*

Univariate and Multivariate point anomalies



(a) Point outlier outside the healthy range of values (dotted black line)

Univariate and Multivariate sequence anomalies

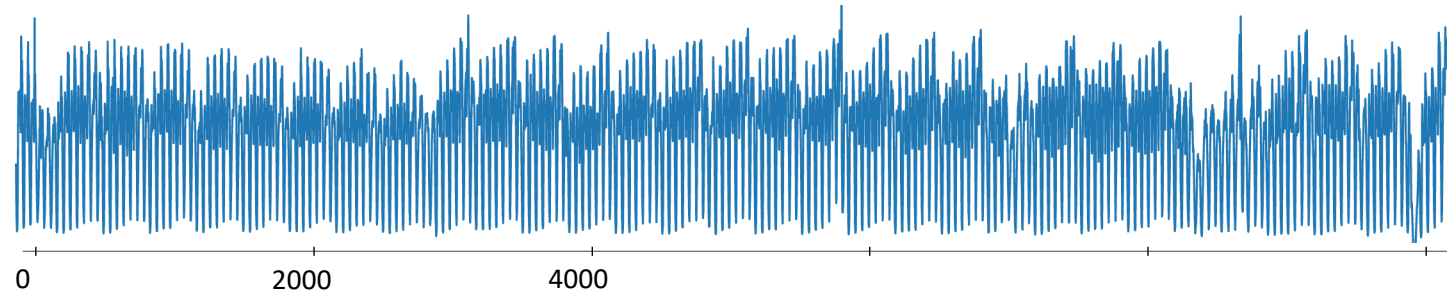


(b) Subsequence outlier inside the healthy range of values (dotted black line)

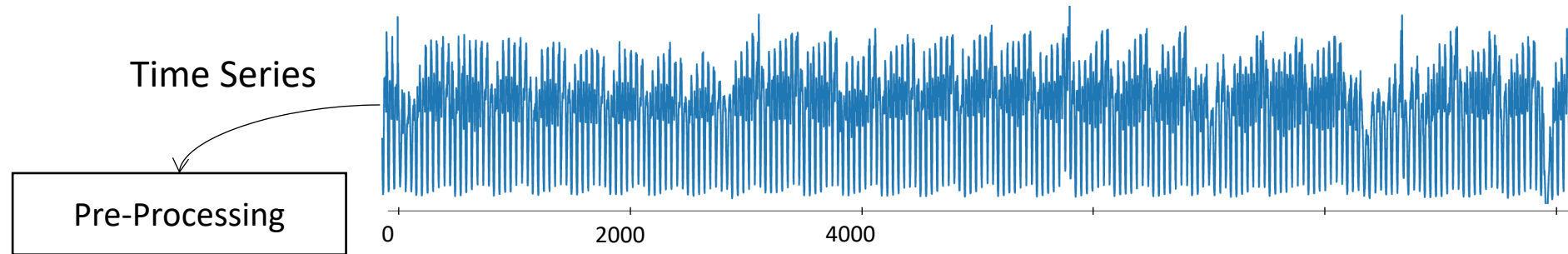
Part 2: Taxonomy of Anomaly Detection Methods

Anomaly Detection methods: *A taxonomy*

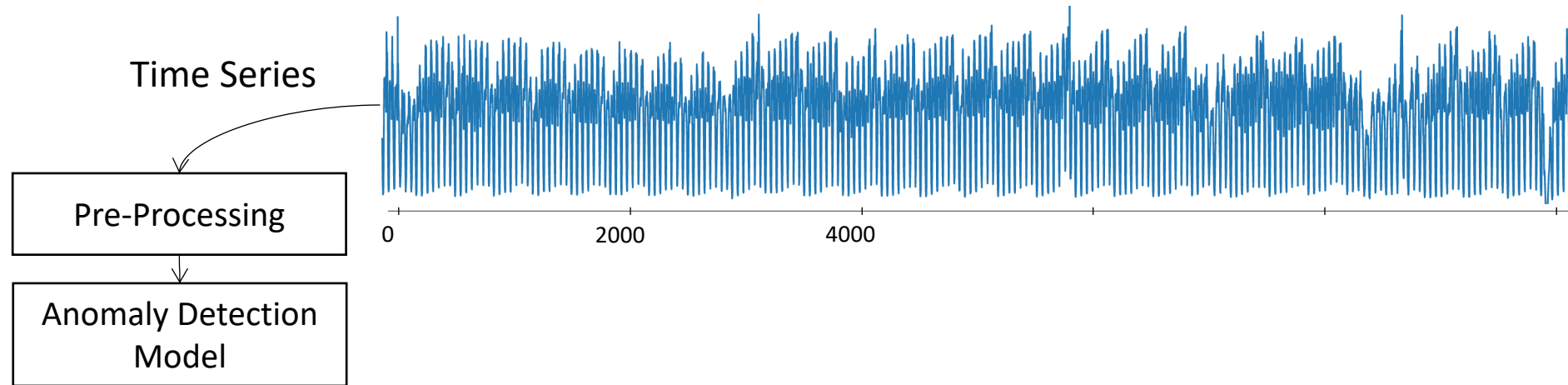
Time Series



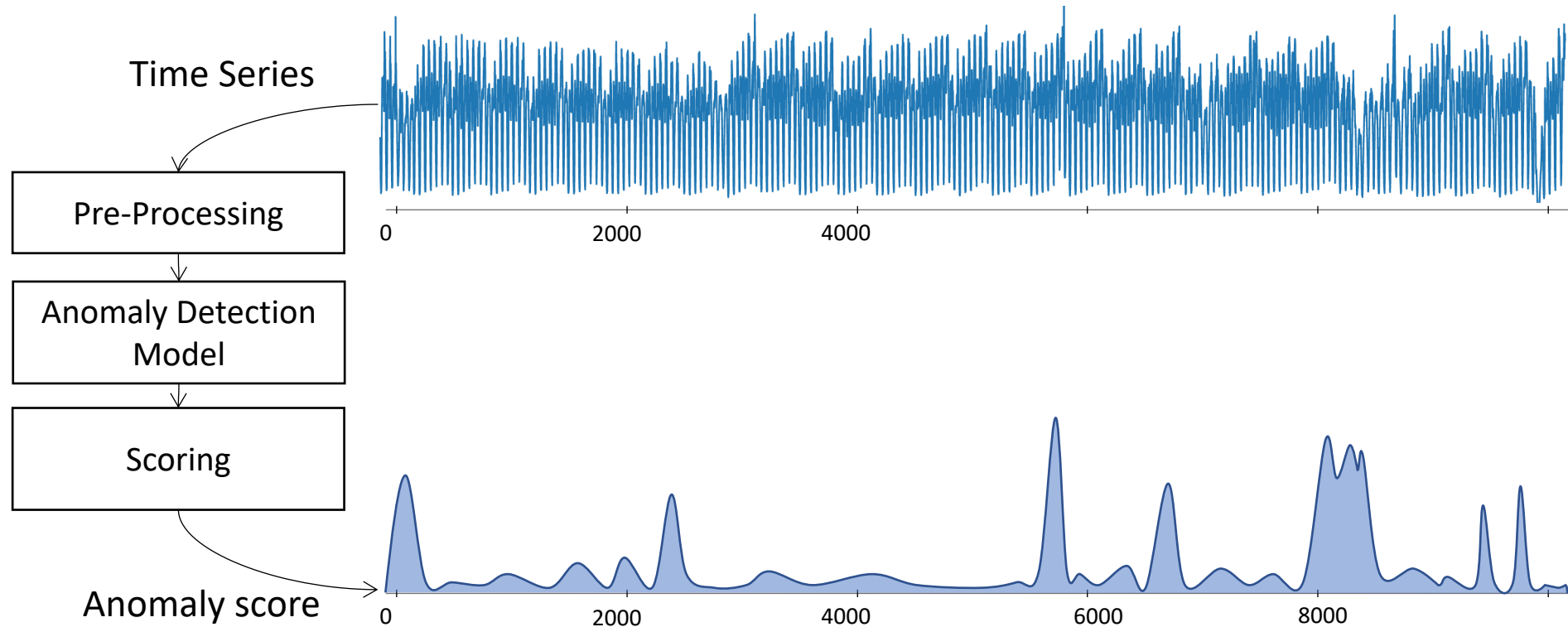
Anomaly Detection methods: *A taxonomy*



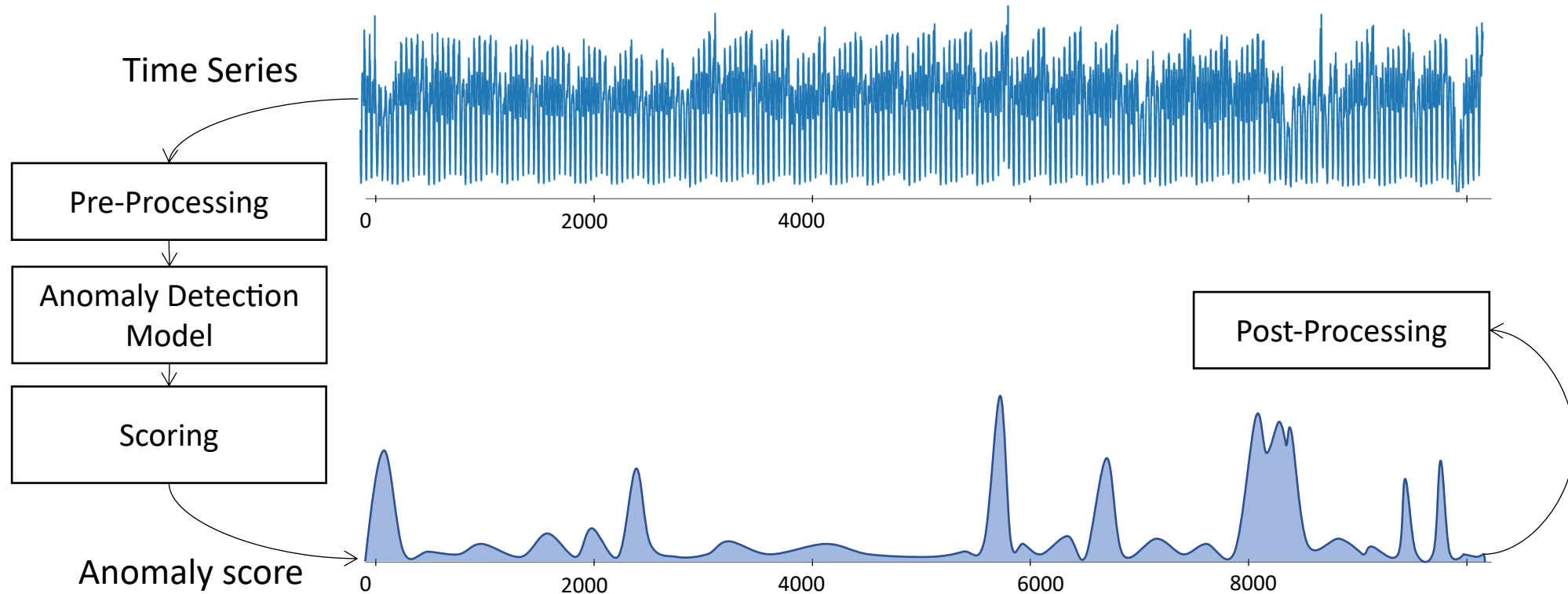
Anomaly Detection methods: *A taxonomy*



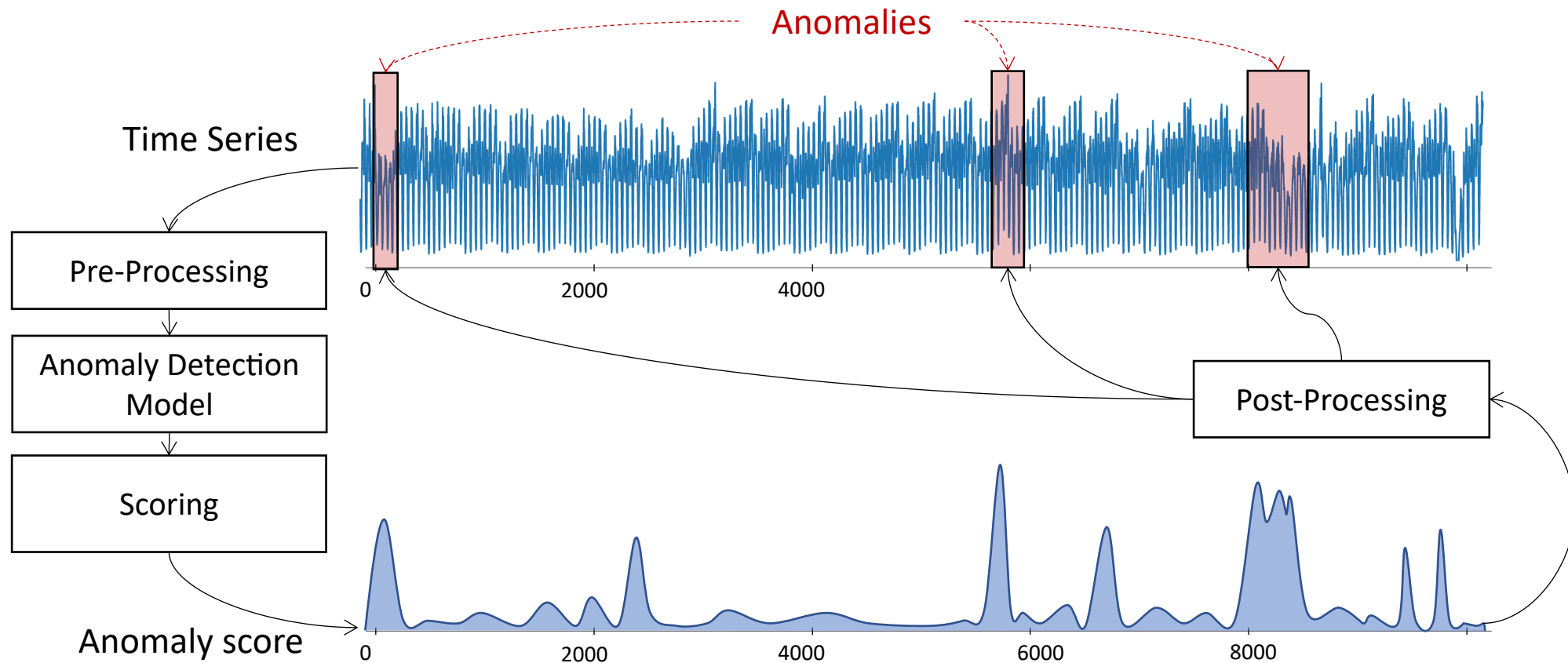
Anomaly Detection methods: *A taxonomy*



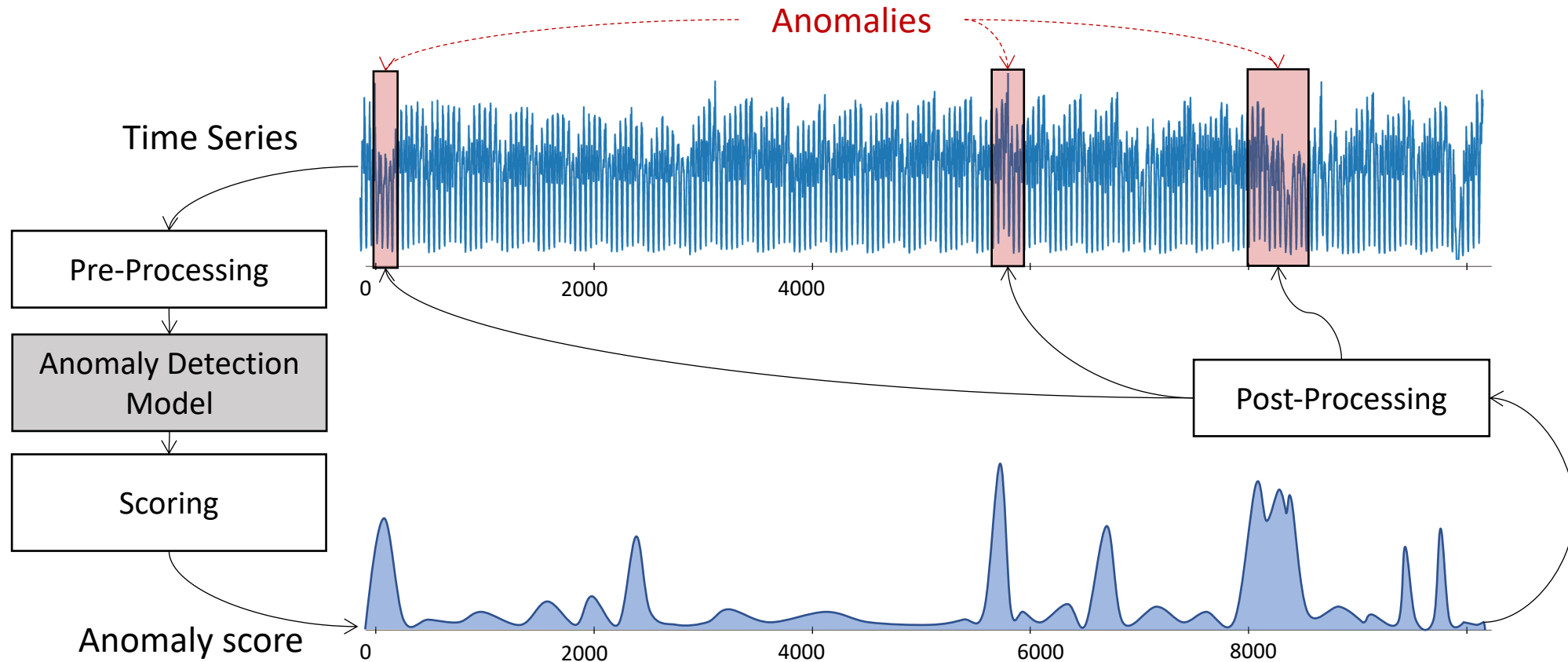
Anomaly Detection methods: *A taxonomy*



Anomaly Detection methods: *A taxonomy*

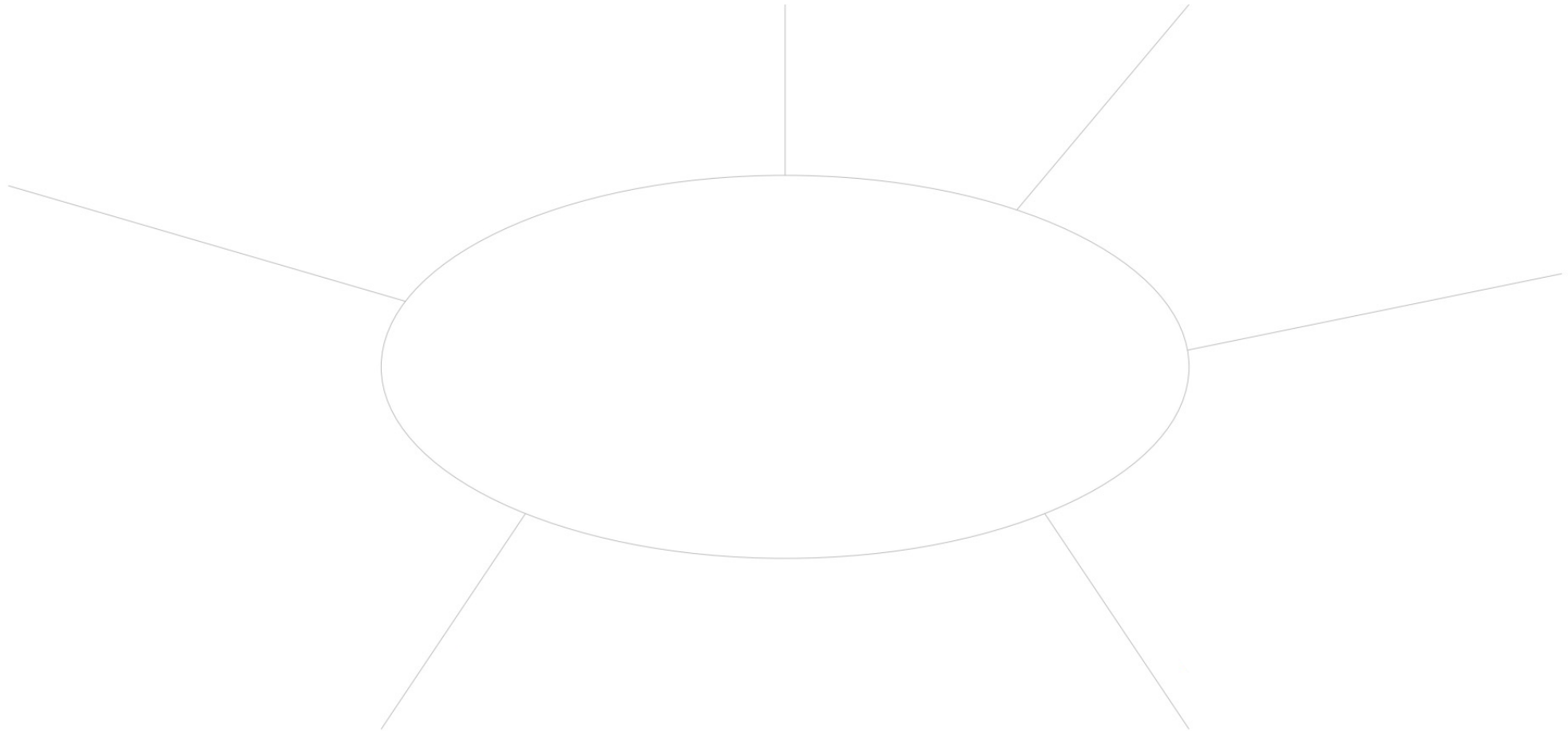


Anomaly Detection methods: *A taxonomy*



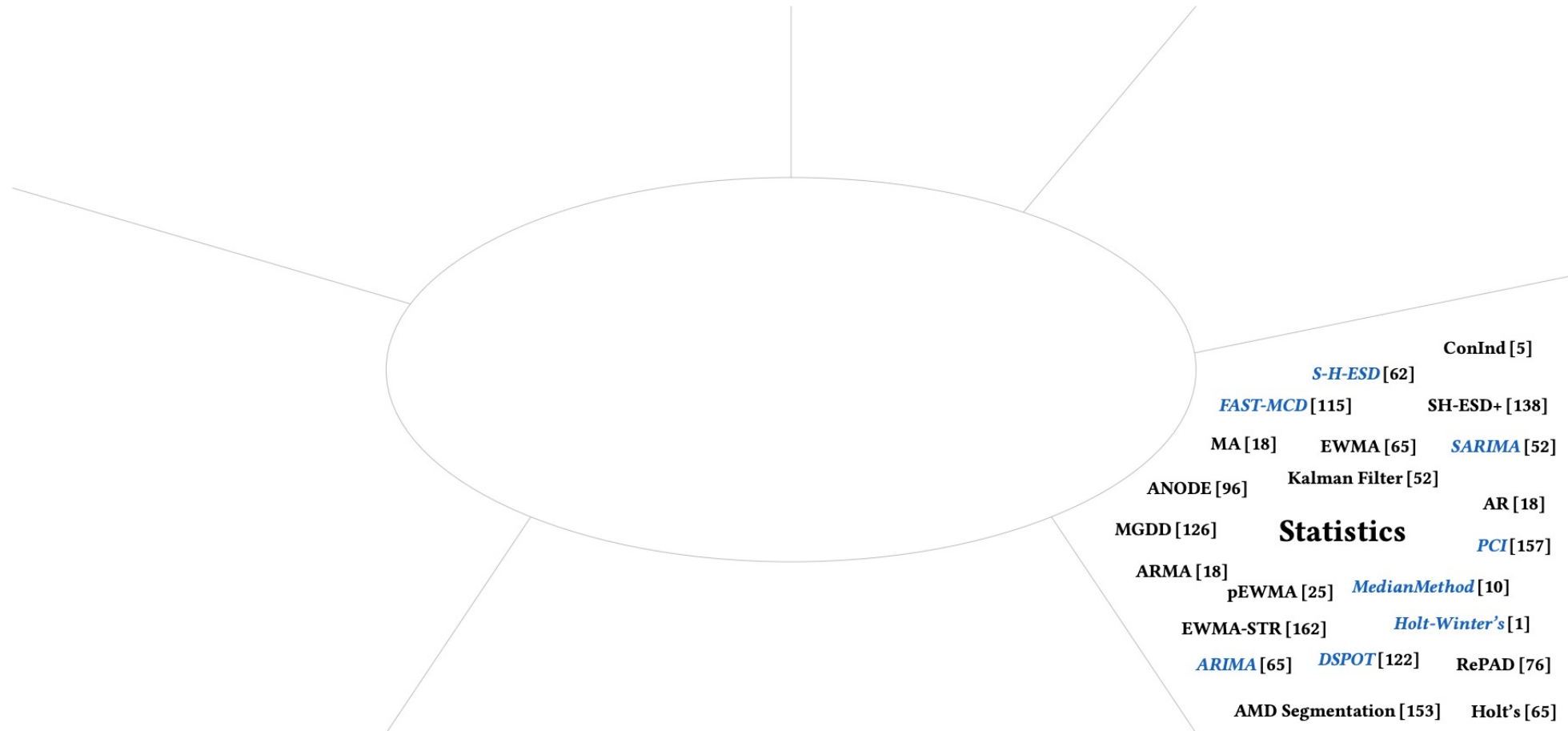
Anomaly Detection methods: *A taxonomy*

By domains [5] ...



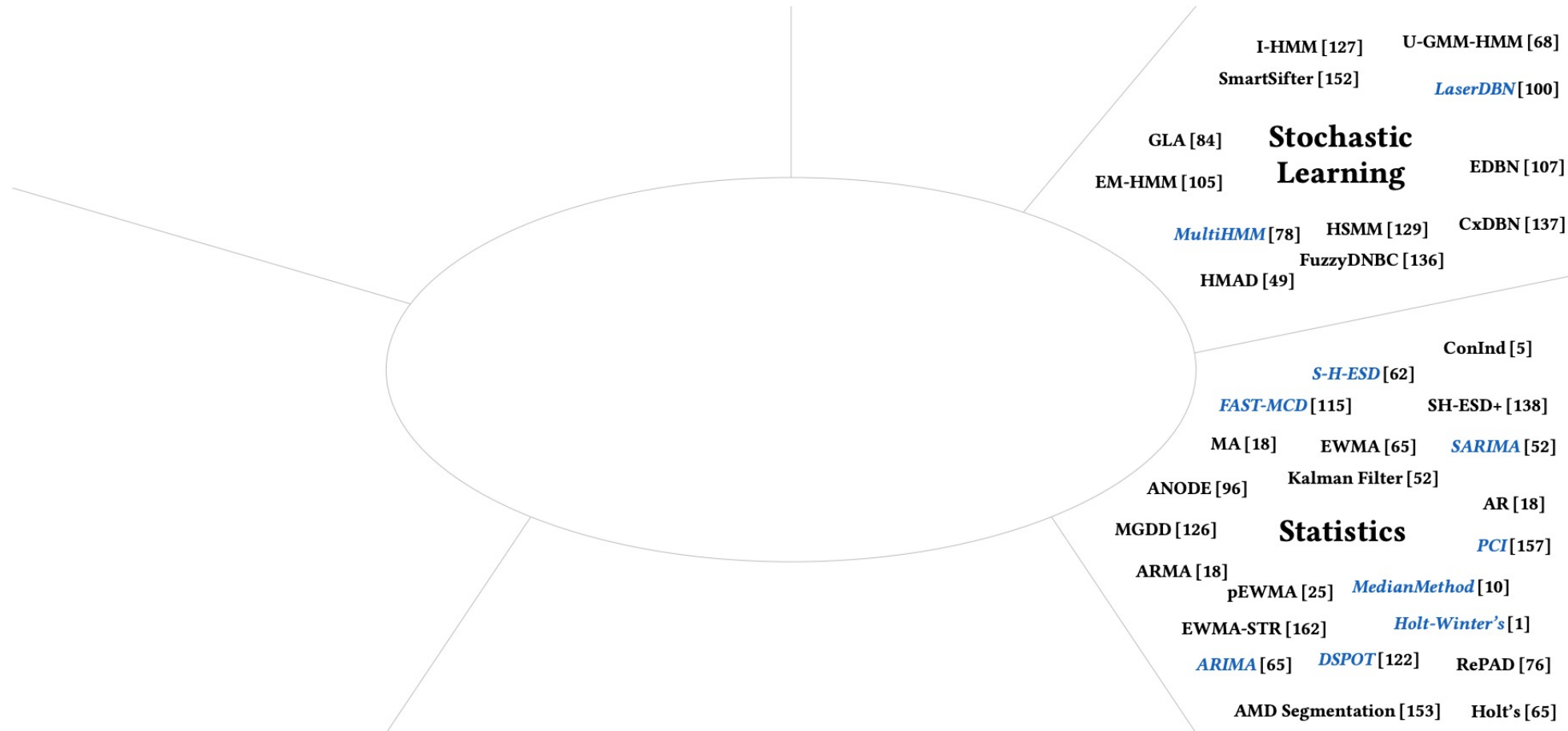
Anomaly Detection methods: *A taxonomy*

By domains [5] ...



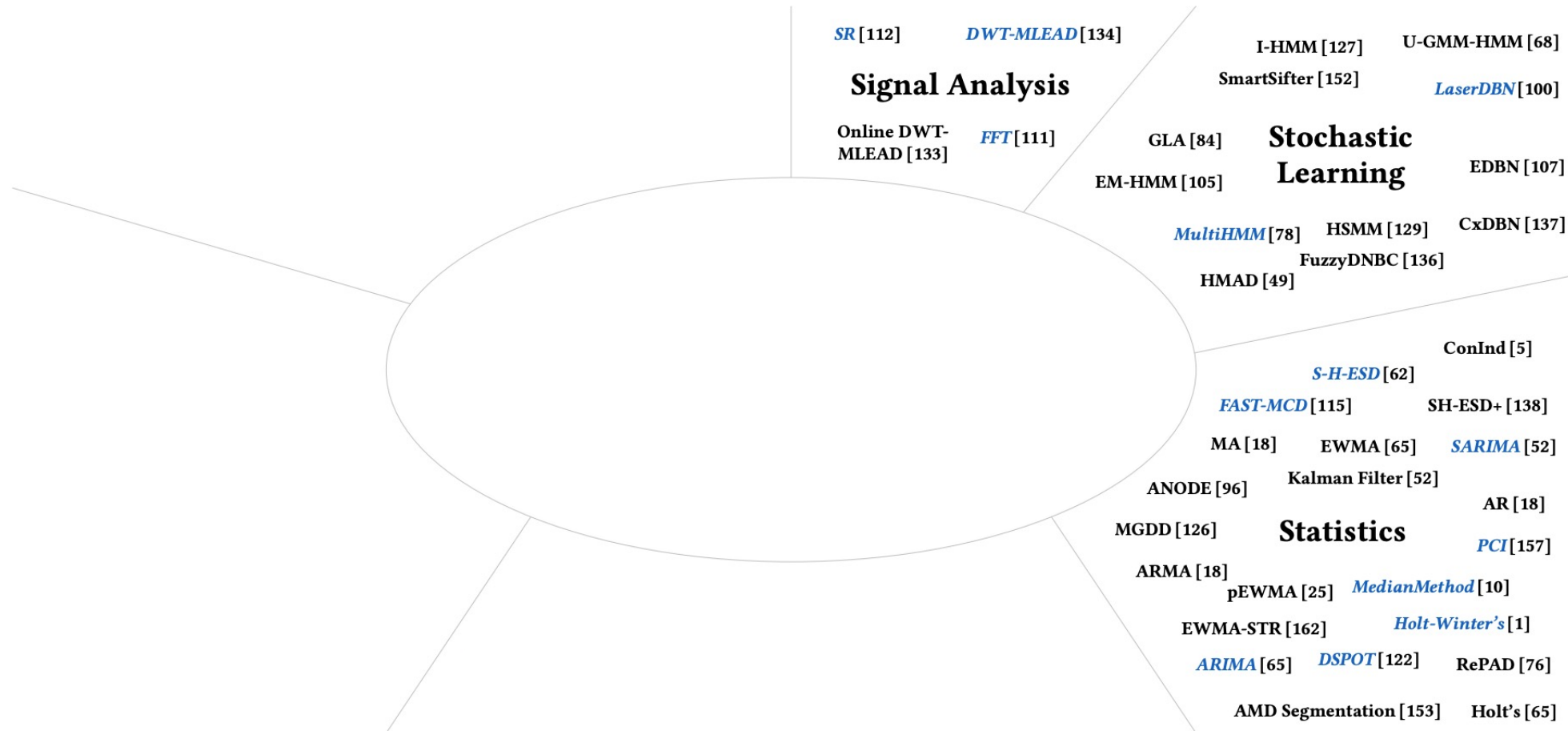
Anomaly Detection methods: *A taxonomy*

By domains [5] ...



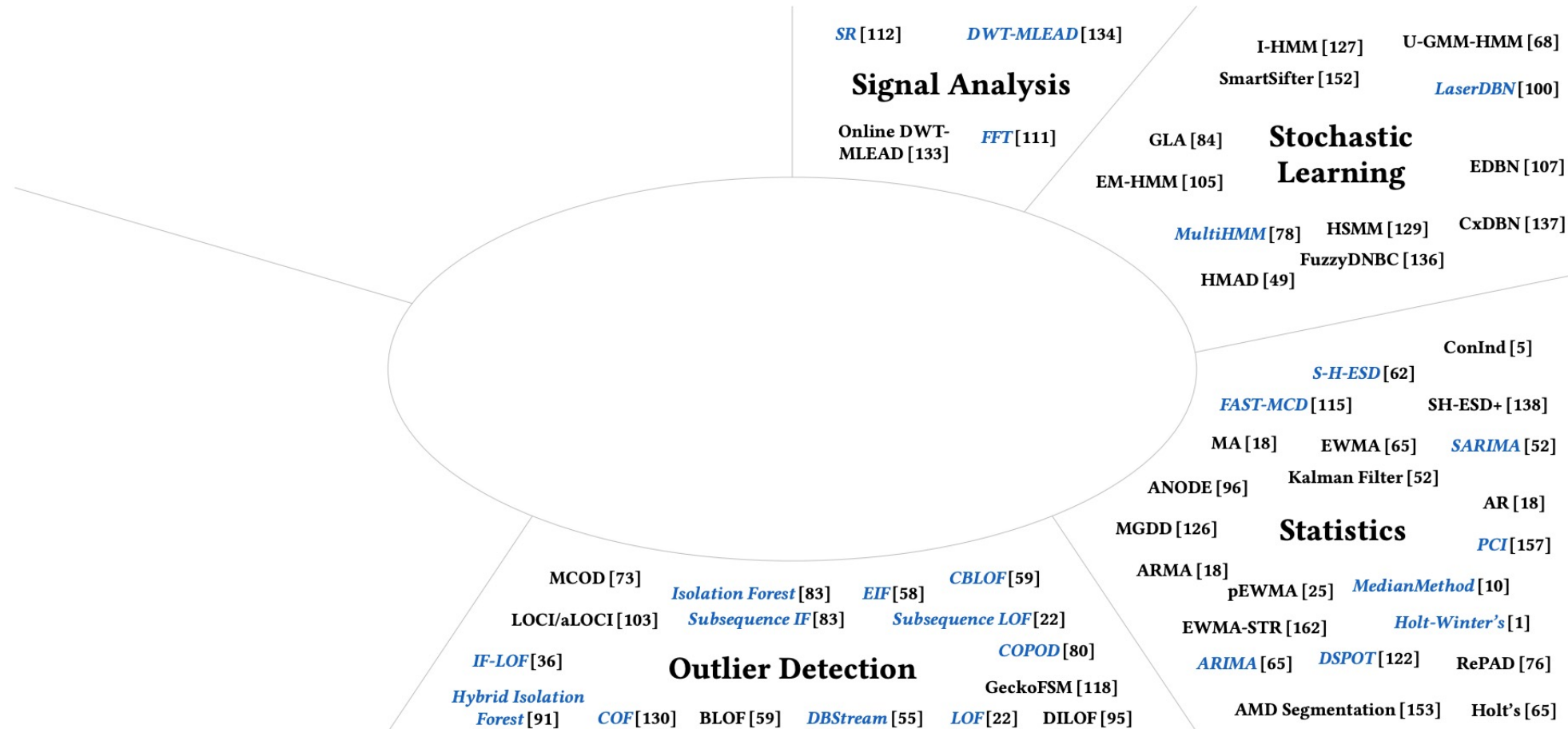
Anomaly Detection methods: *A taxonomy*

By domains [5] ...



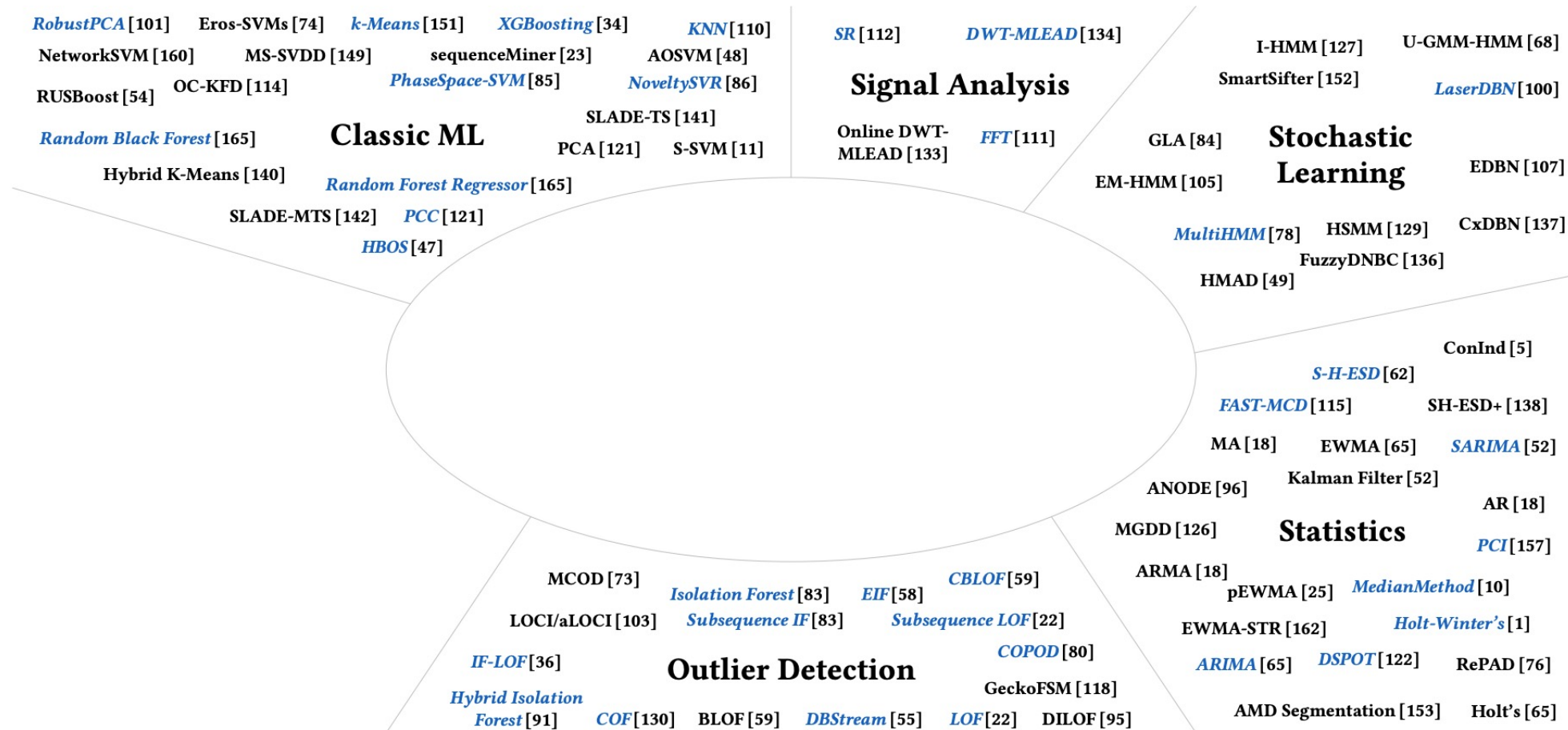
Anomaly Detection methods: *A taxonomy*

By domains [5] ...



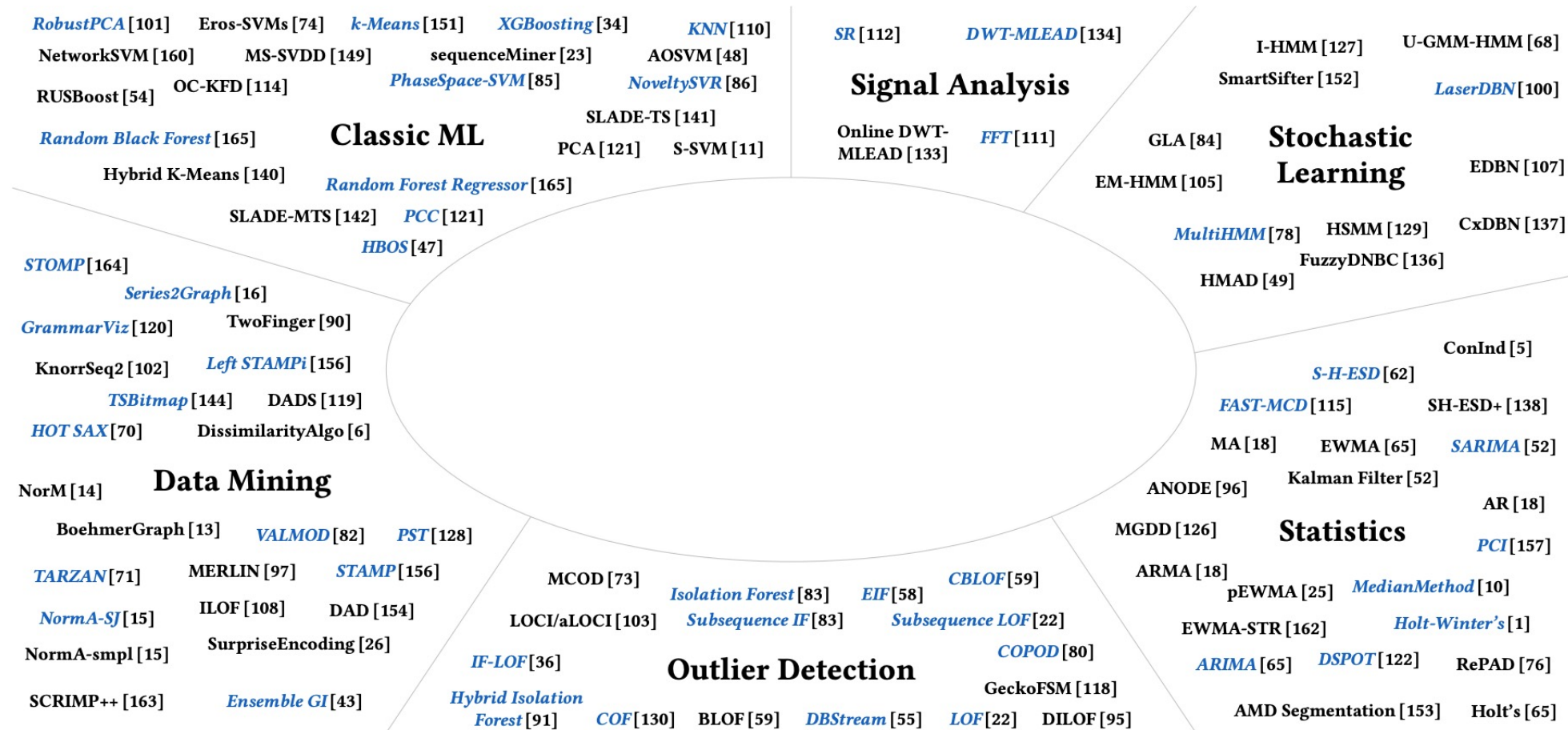
Anomaly Detection methods: *A taxonomy*

By domains [5] ...



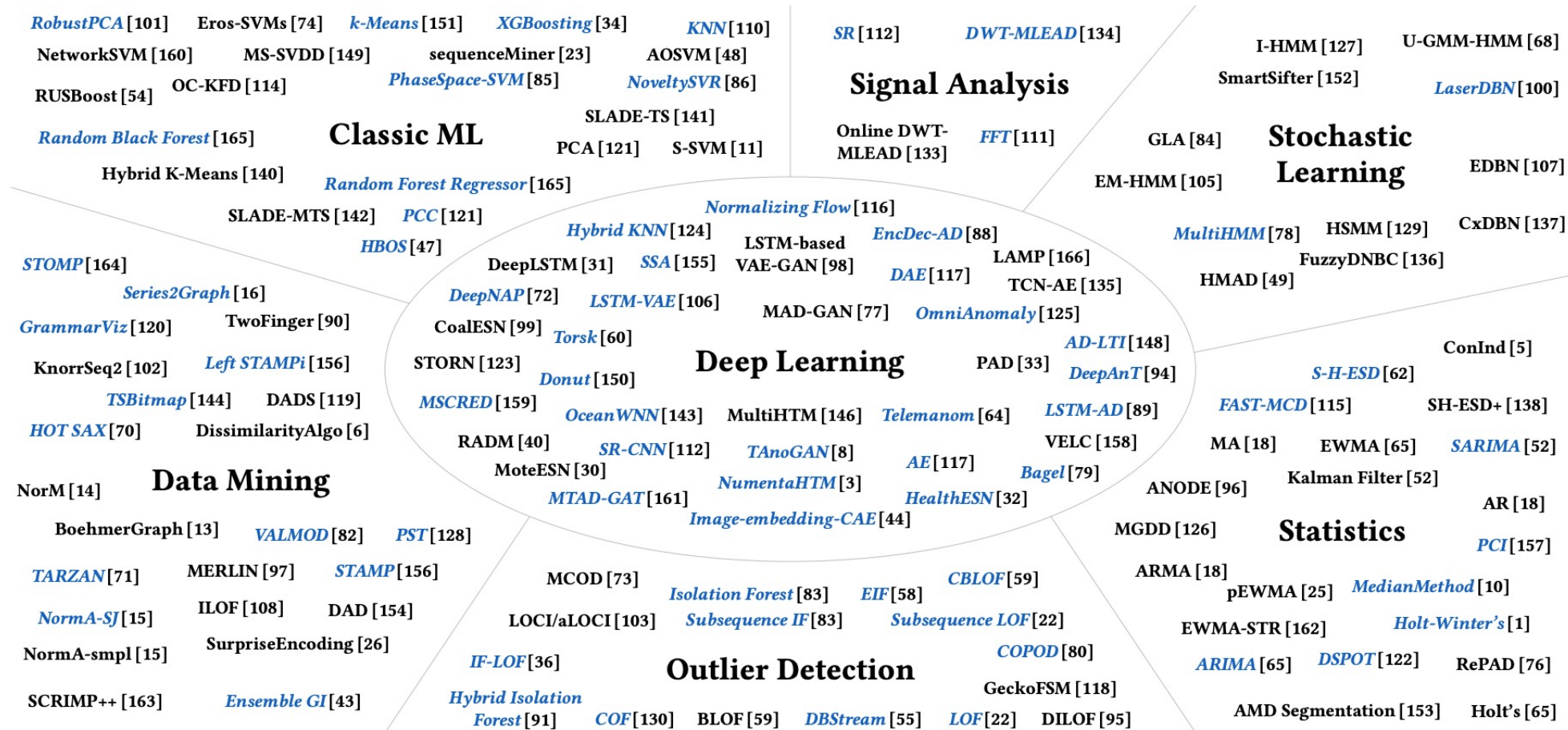
Anomaly Detection methods: *A taxonomy*

By domains [5] ...



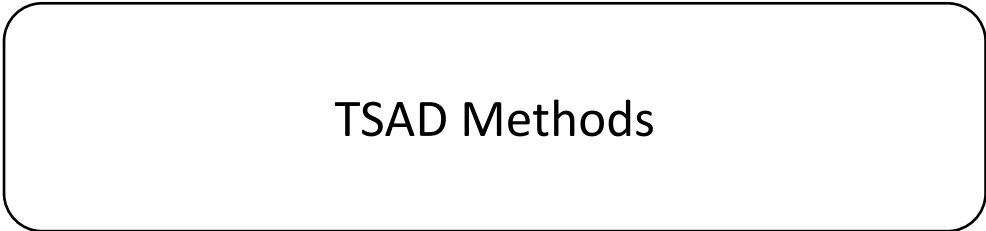
Anomaly Detection methods: *A taxonomy*

By domains [5] ...



Anomaly Detection methods: *A taxonomy*

By inputs...



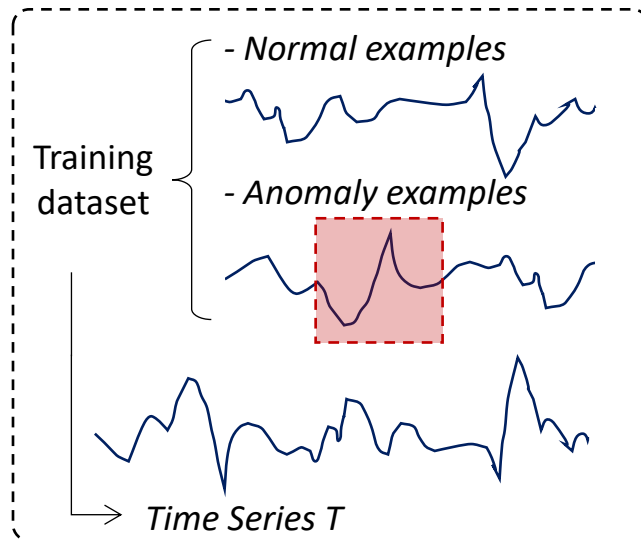
TSAD Methods

Anomaly Detection methods: *A taxonomy*

By inputs...

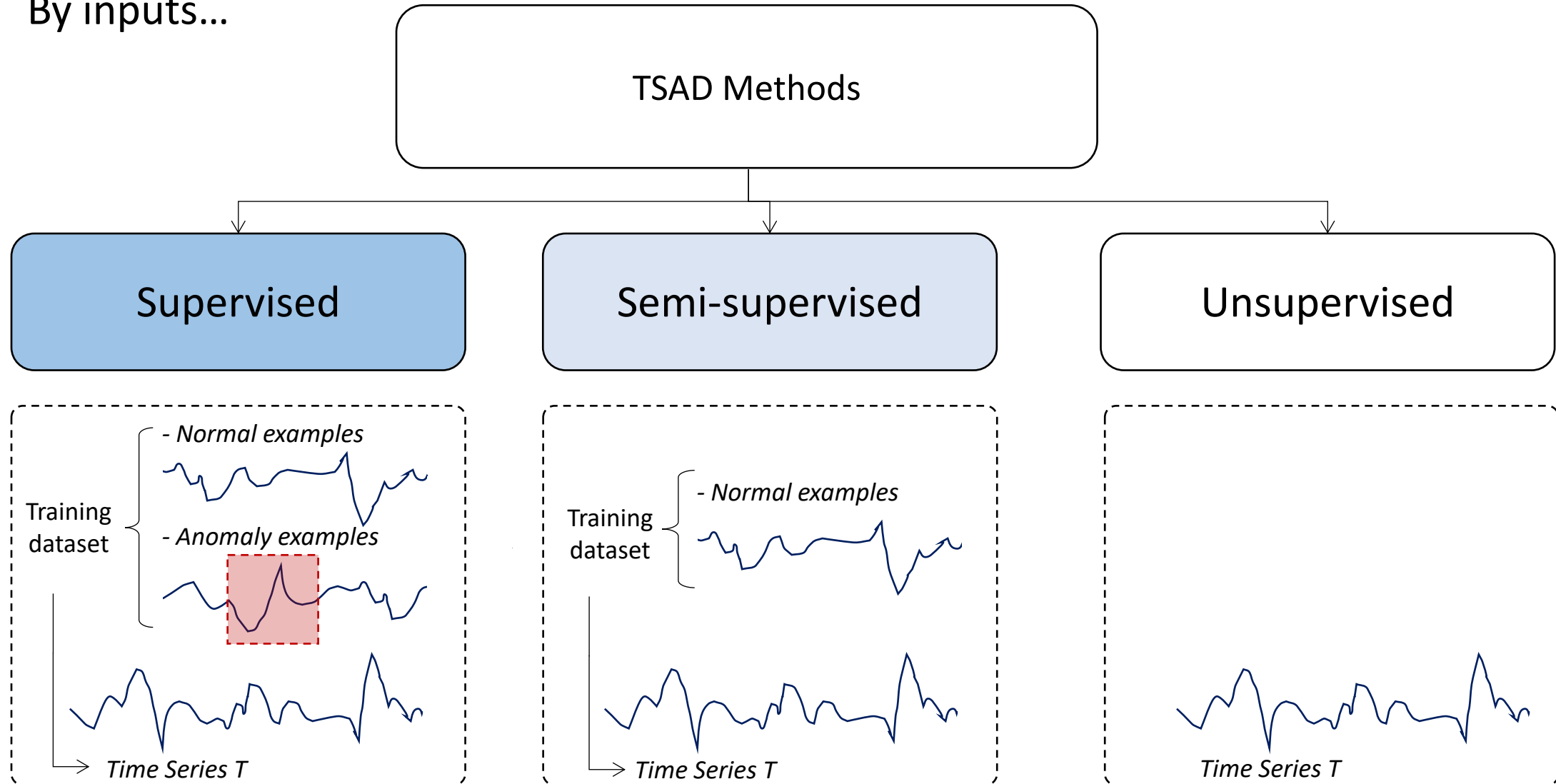
TSAD Methods

Supervised



Anomaly Detection methods: *A taxonomy*

By inputs...



Anomaly Detection

By inputs...

Time

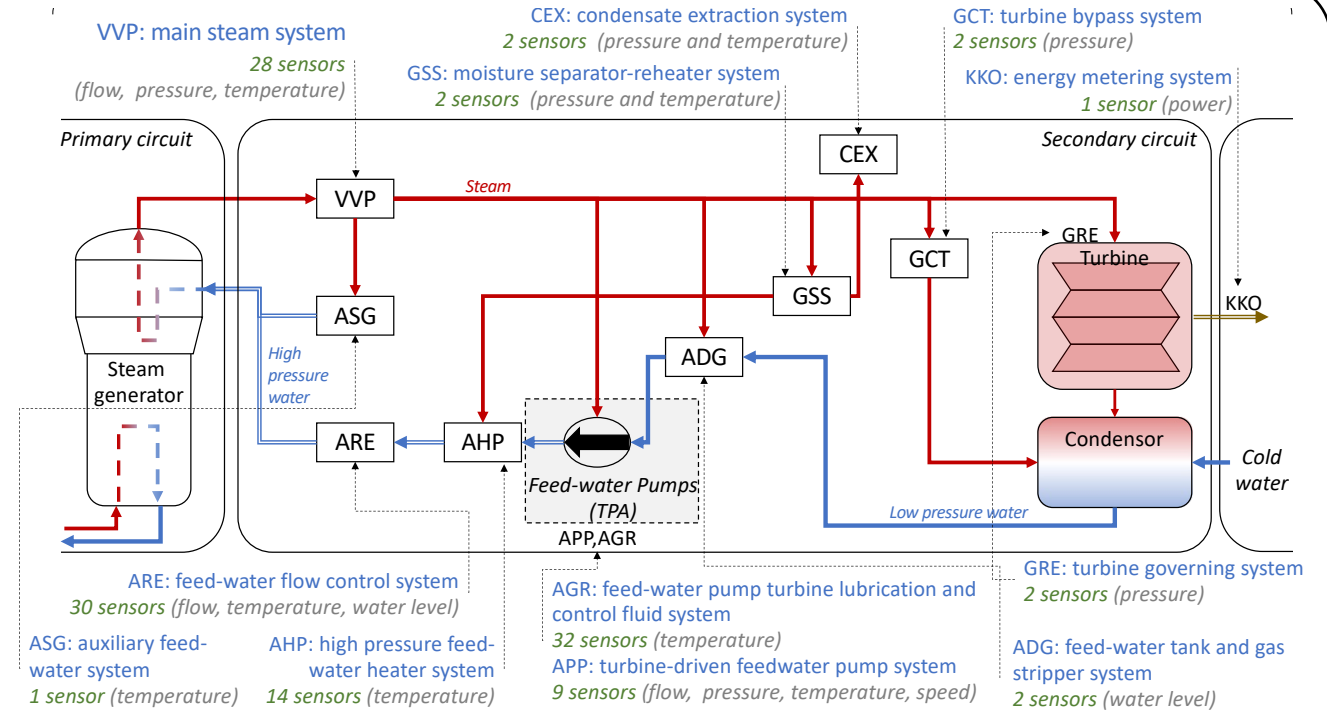
Supervised

Training dataset

- Normal examples

- Anomaly examples

Time Series T



Anomaly Detection

By inputs...

Time

Supervised

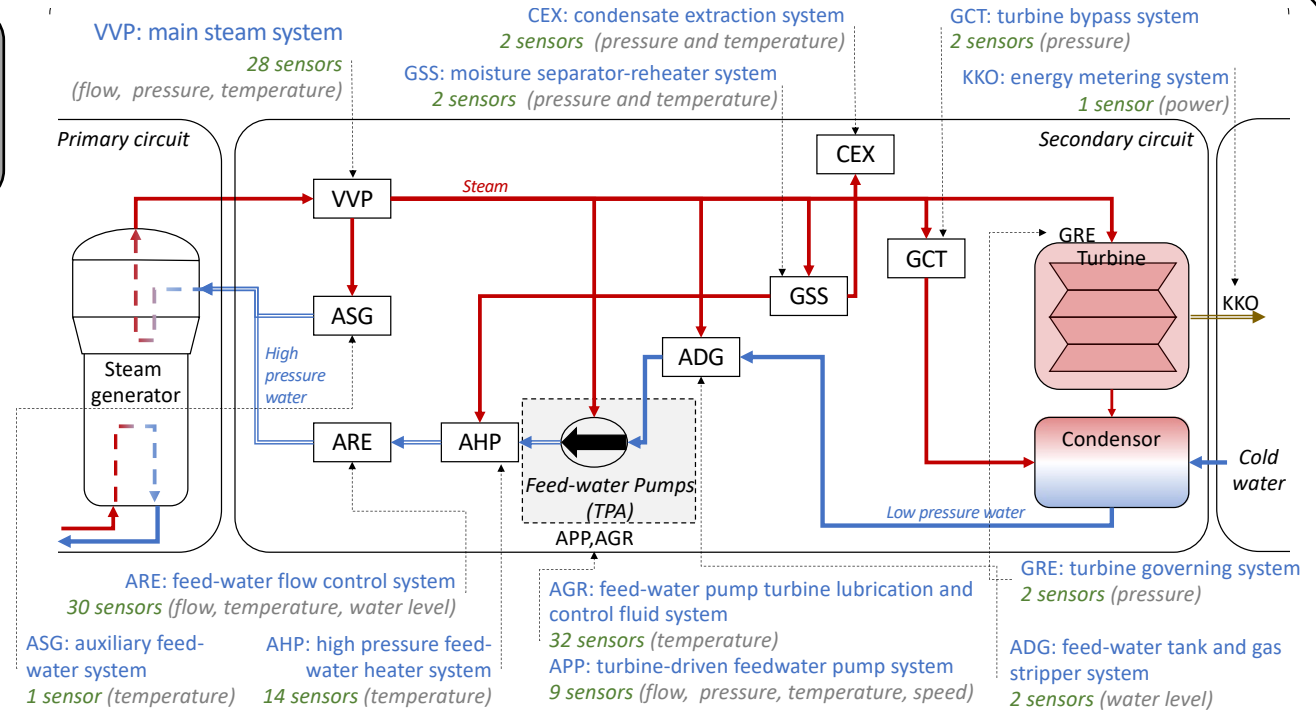
Training dataset

- Normal examples

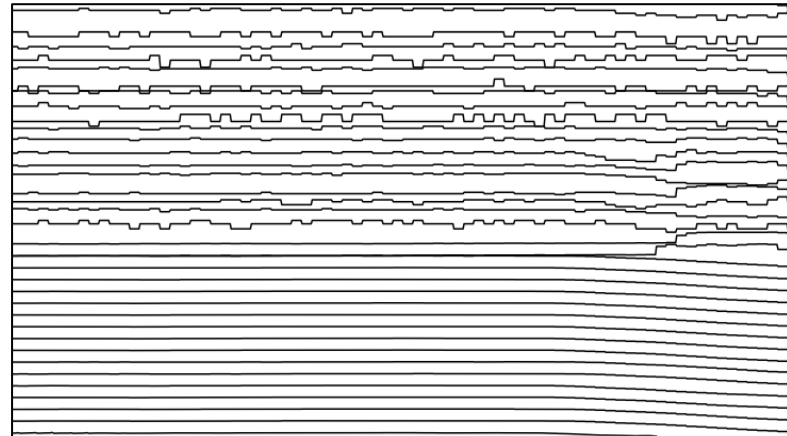
- Anomaly examples

Time Series T

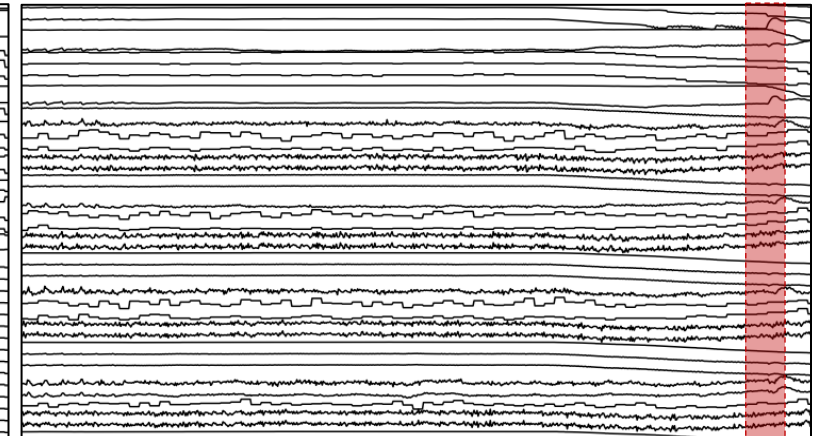
Supervised anomaly detection (e.g., classification)



Class 1: Time series without any vibrations



Class 2: Time series with a vibrations



Vibration

Anomaly Detection

By inputs...

Time

Supervised

Supervised anomaly detection (e.g., classification)

Explanation of the detection

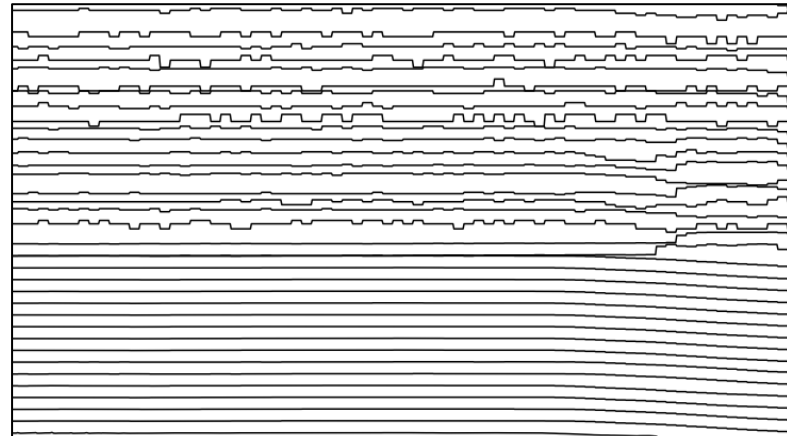
Training dataset

- Normal examples

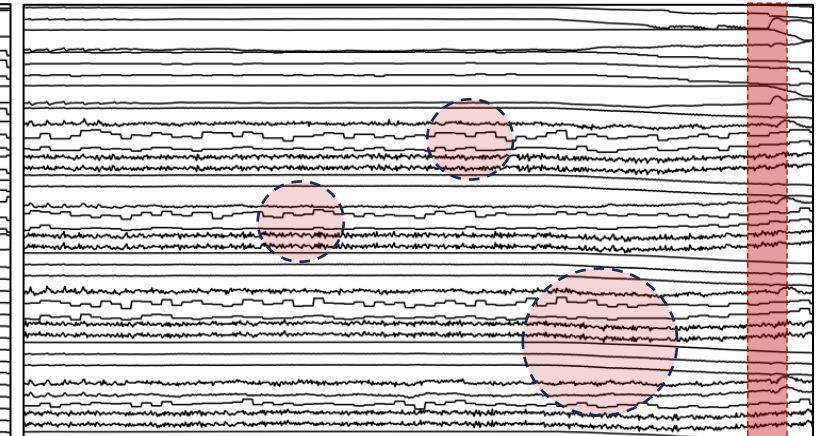
- Anomaly examples

Time Series T

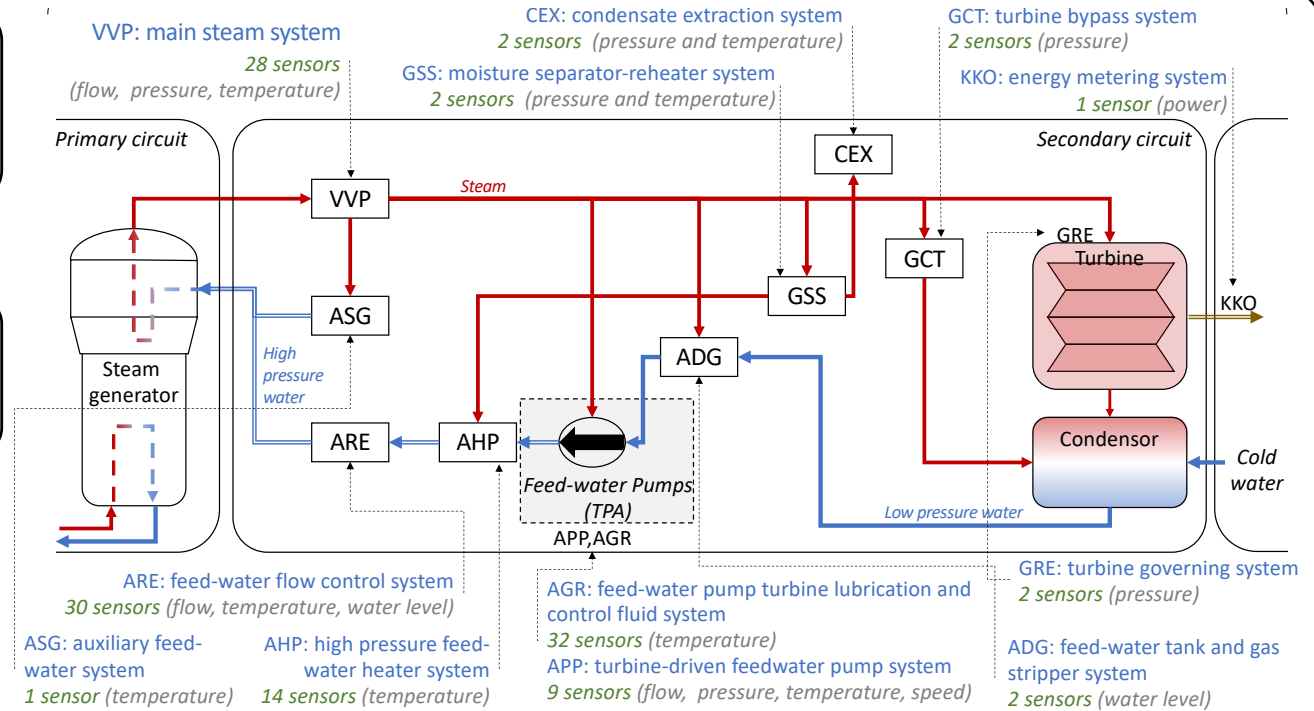
Class 1: Time series without any vibrations



Class 2: Time series with a vibrations



Vibration



Anomaly Detection

By inputs...

Time

Supervised

Training dataset

- Normal examples

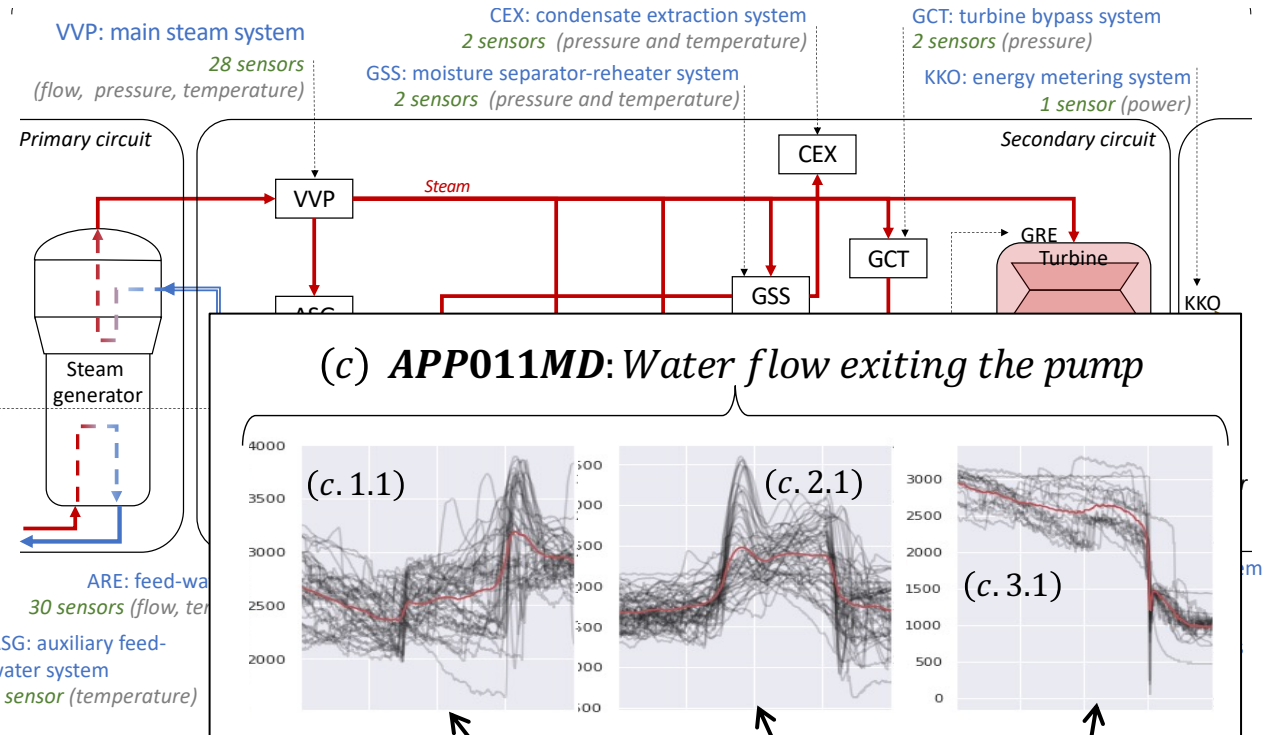
- Anomaly examples

Time Series T

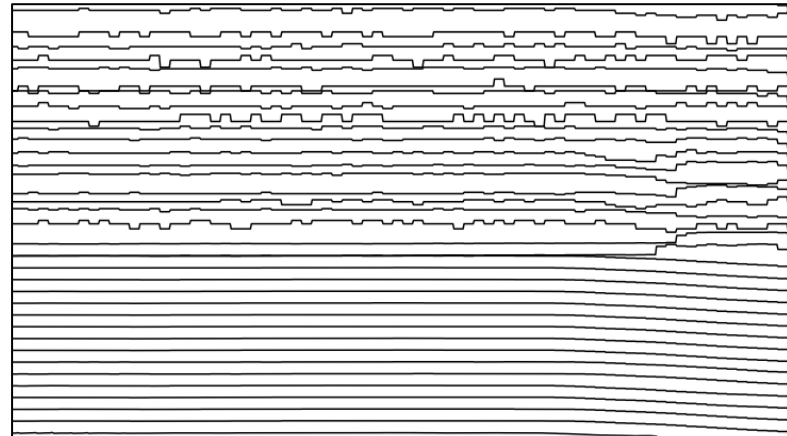
Supervised anomaly detection (e.g., classification)

Explanation of the detection

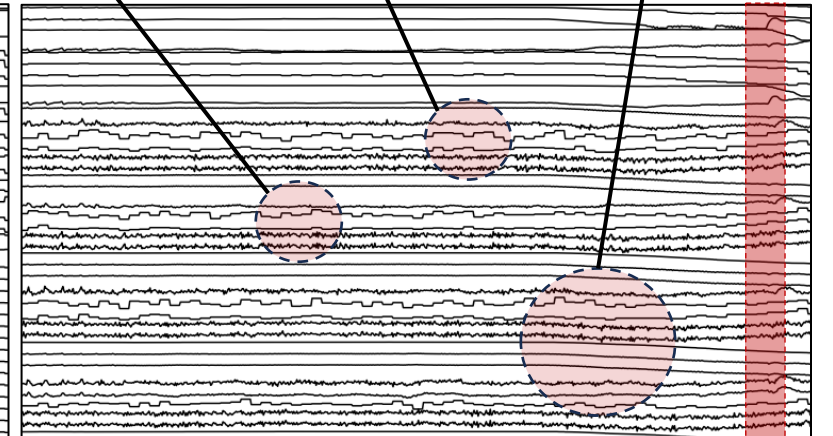
Identification of precursors



Class 1: Time series without any vibrations



Class 2: Time series with a vibrations



Vibration

Anomaly Detection

By inputs...

Time

Supervised

Training dataset

- Normal examples

- Anomaly examples

Time Series T

Supervised anomaly detection (e.g., classification)

Explanation of the detection

Identification of precursors

Class 1: Turbine

More info :

On the use case



DCE journal 2023

On the method



SIGMOD 2022



pump

1)

vibrations

Vibration

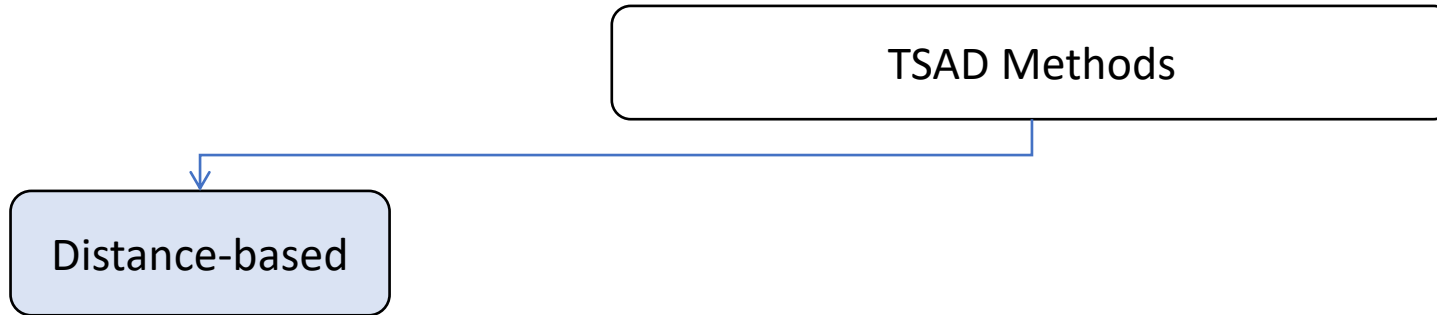
Anomaly Detection methods: *A taxonomy*

By methods...

TSAD Methods

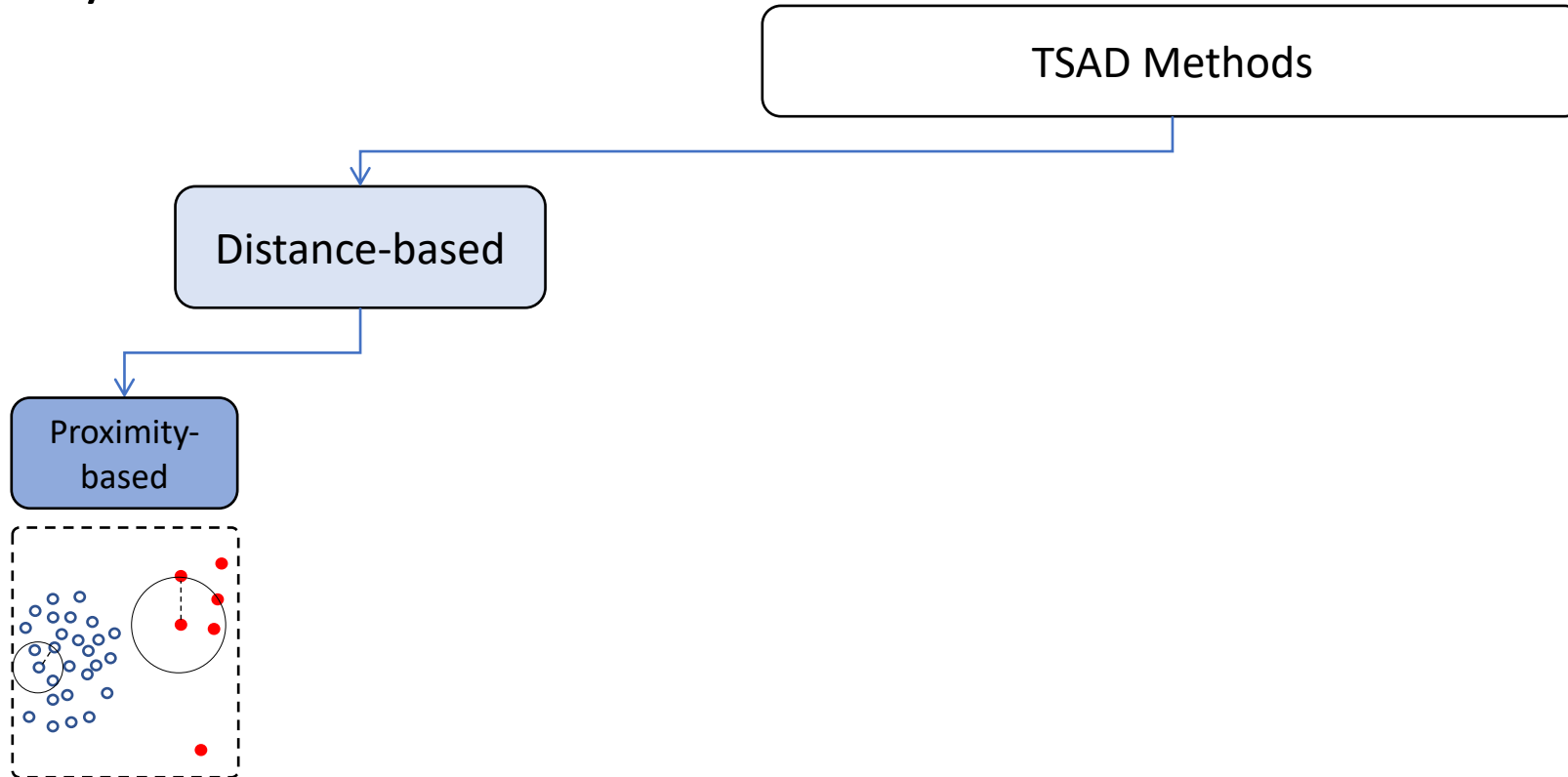
Anomaly Detection methods: *A taxonomy*

By methods...



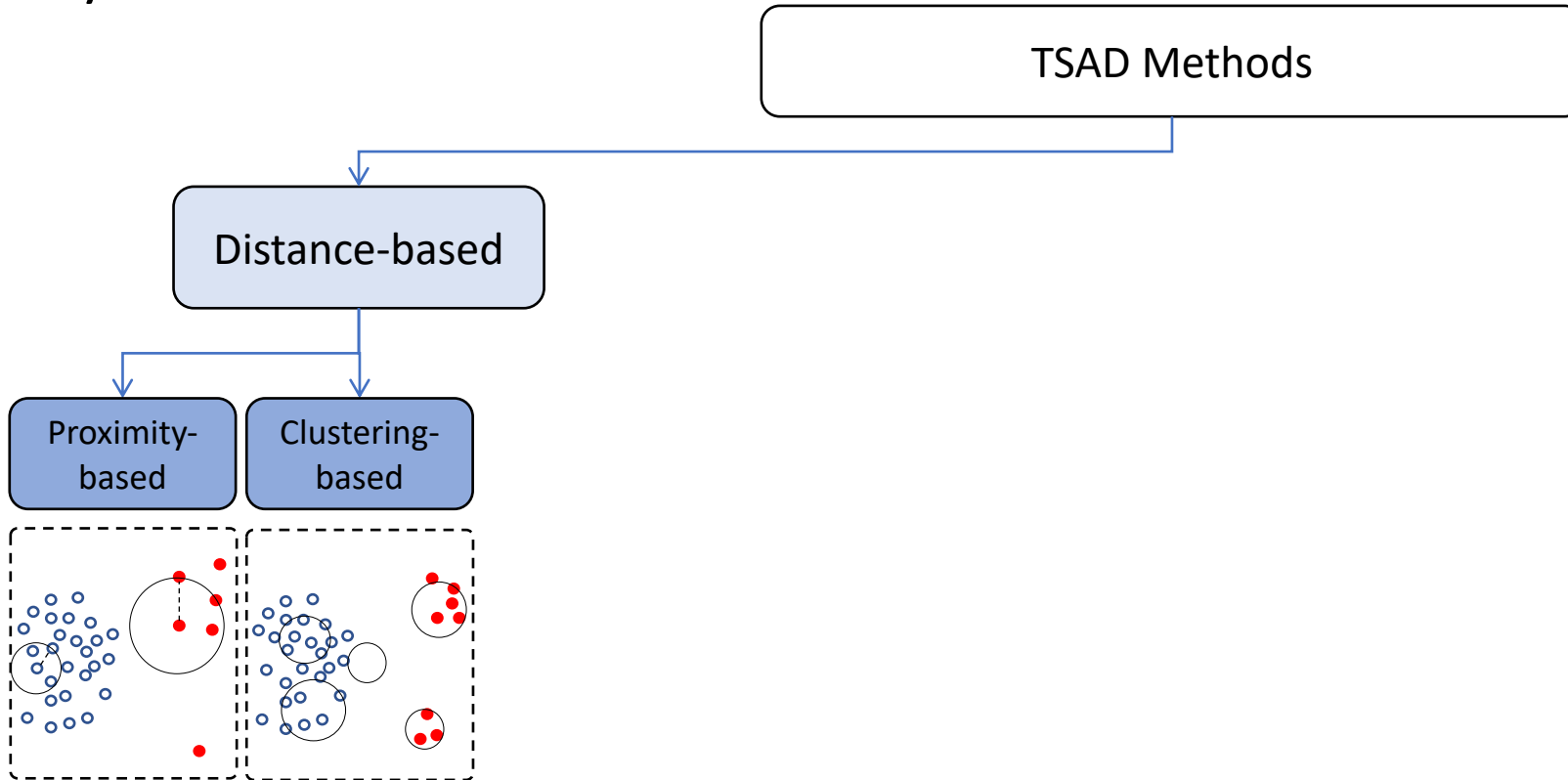
Anomaly Detection methods: *A taxonomy*

By methods...



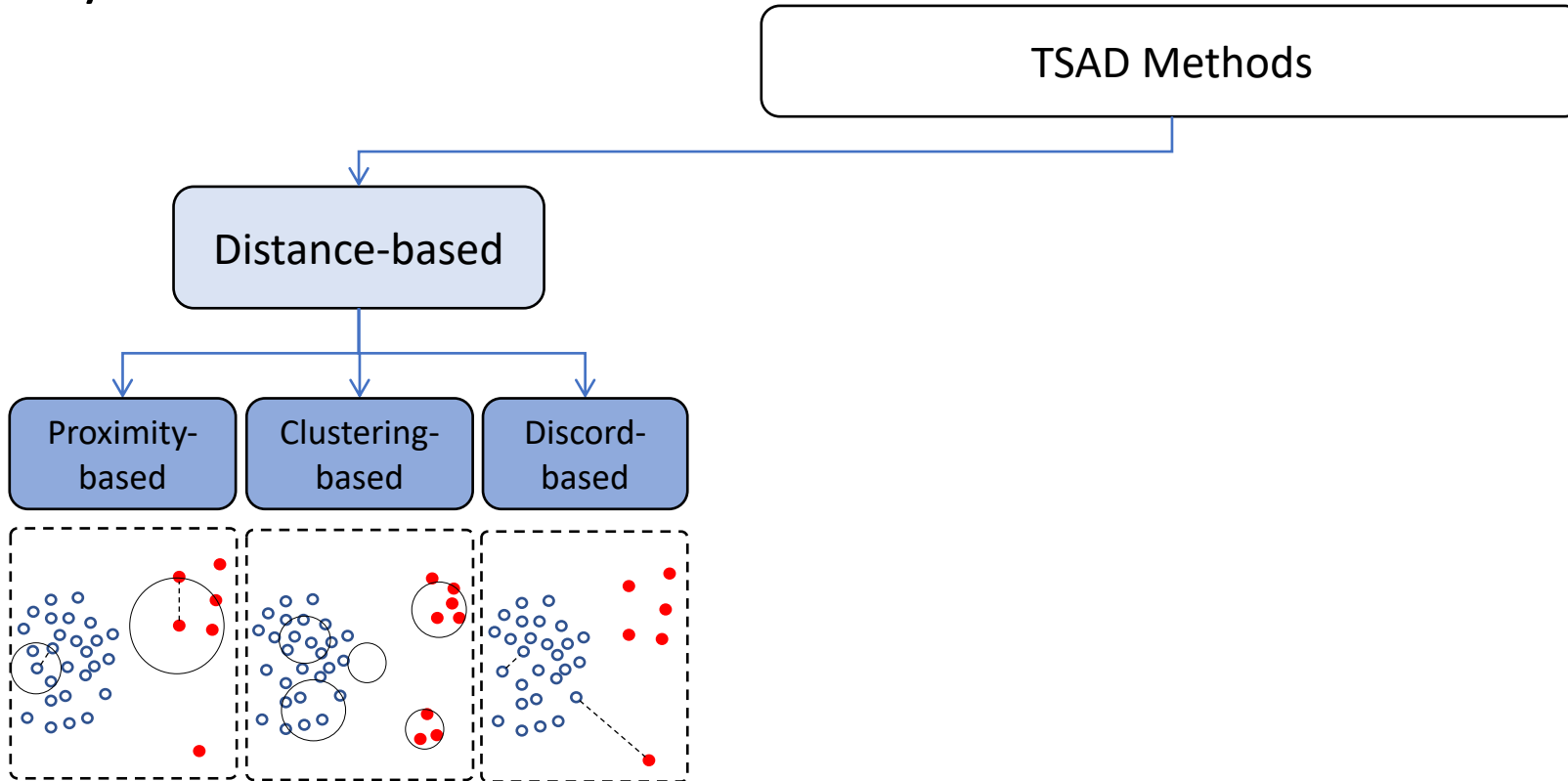
Anomaly Detection methods: *A taxonomy*

By methods...



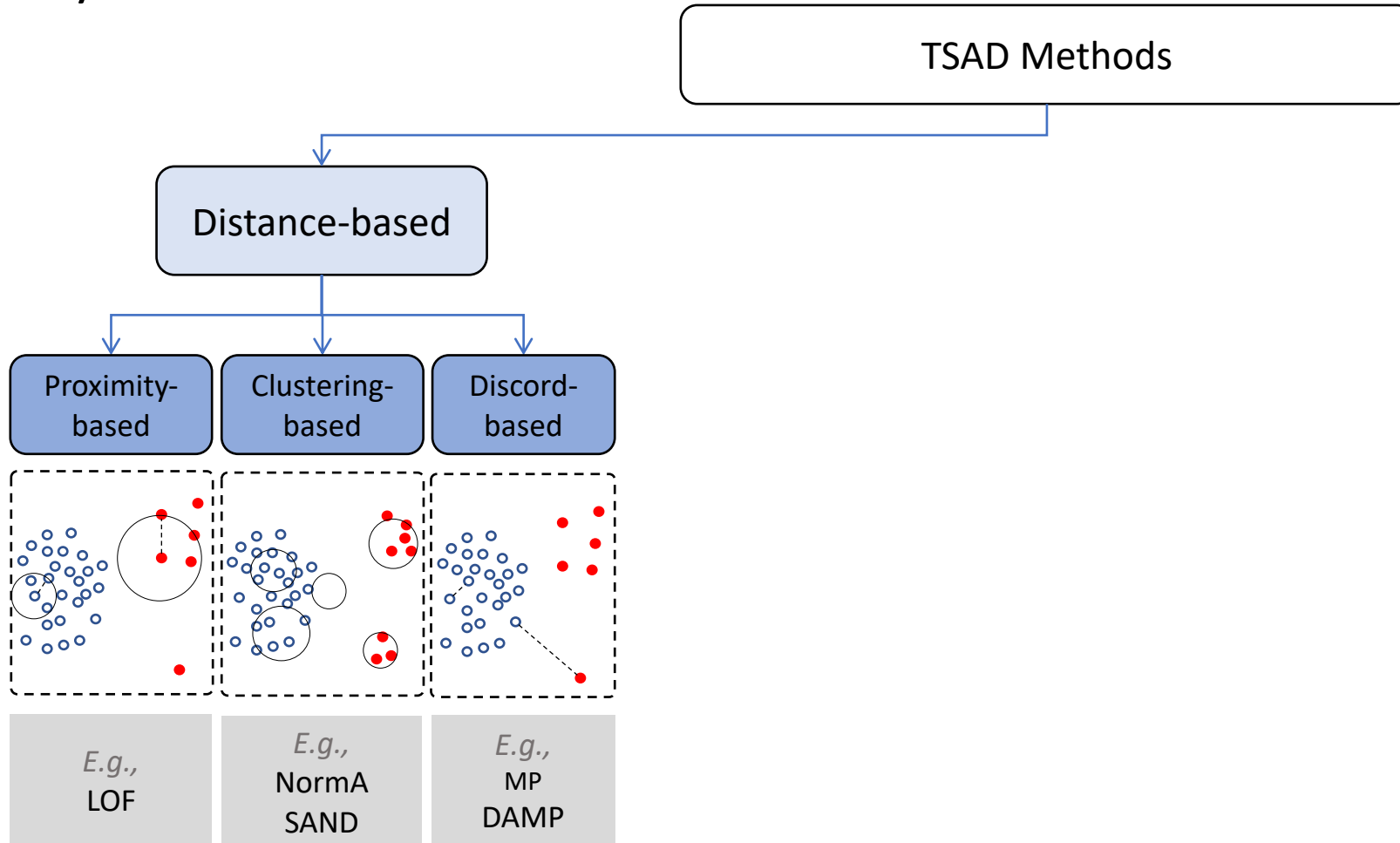
Anomaly Detection methods: *A taxonomy*

By methods...



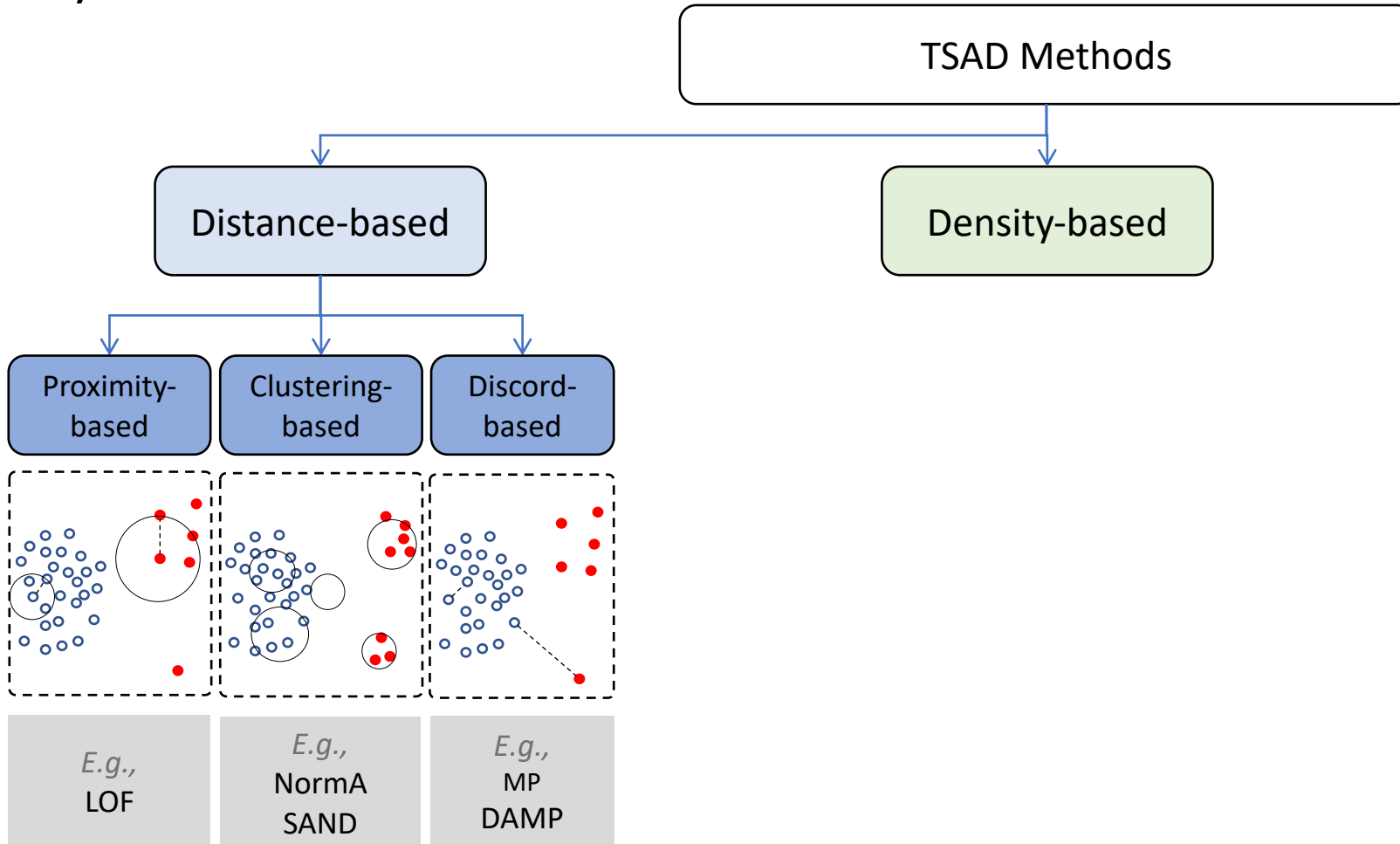
Anomaly Detection methods: *A taxonomy*

By methods...



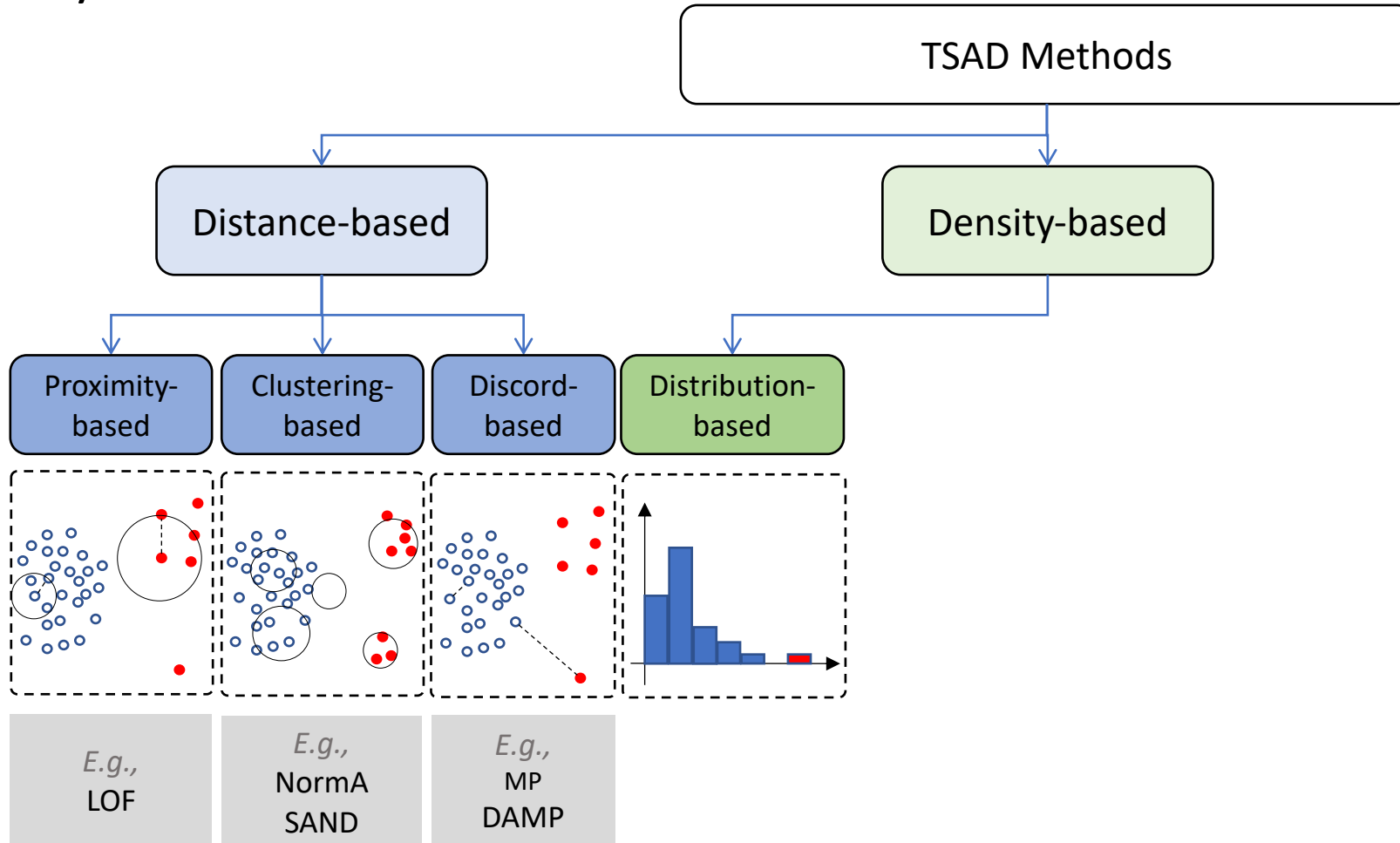
Anomaly Detection methods: *A taxonomy*

By methods...



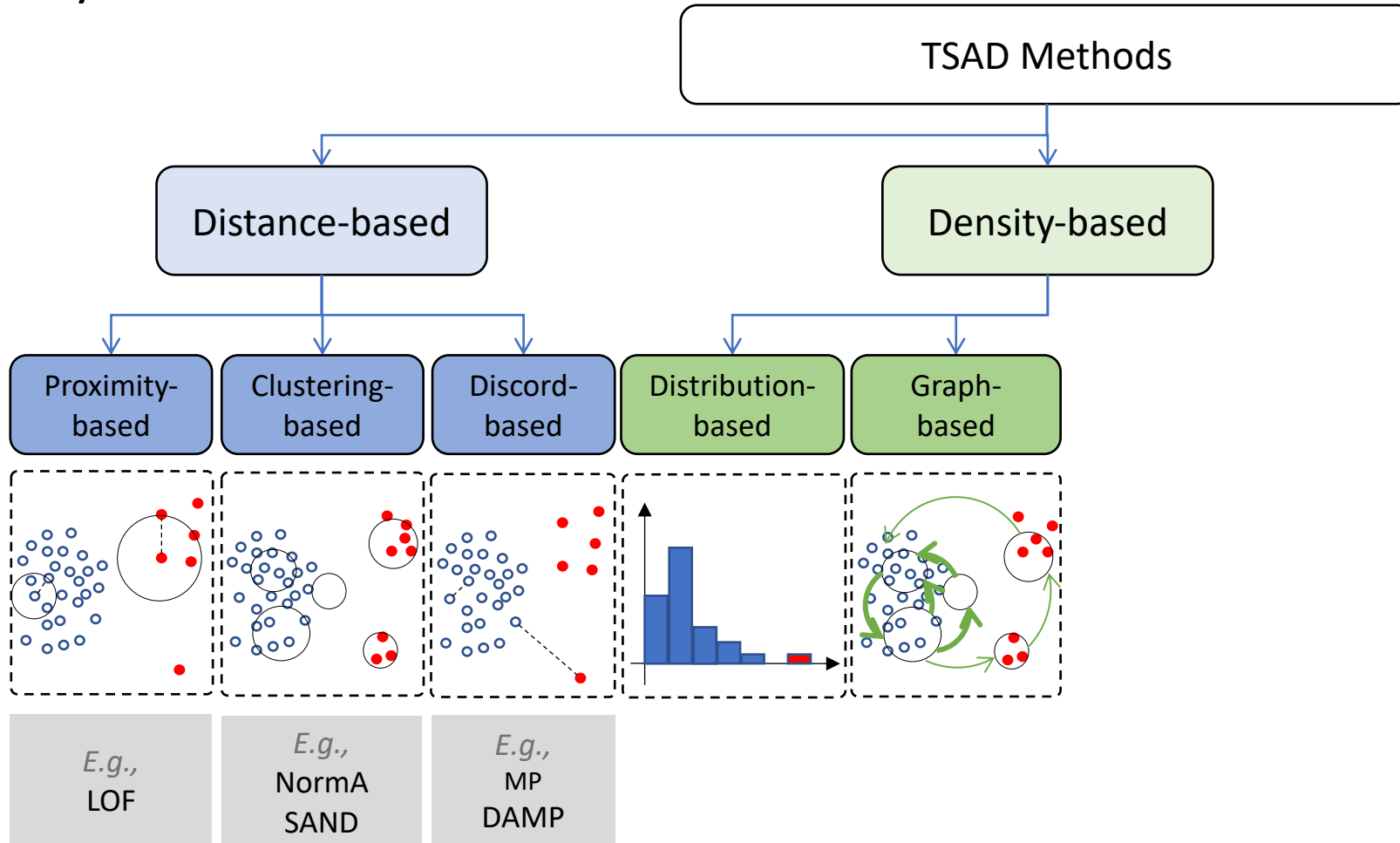
Anomaly Detection methods: *A taxonomy*

By methods...



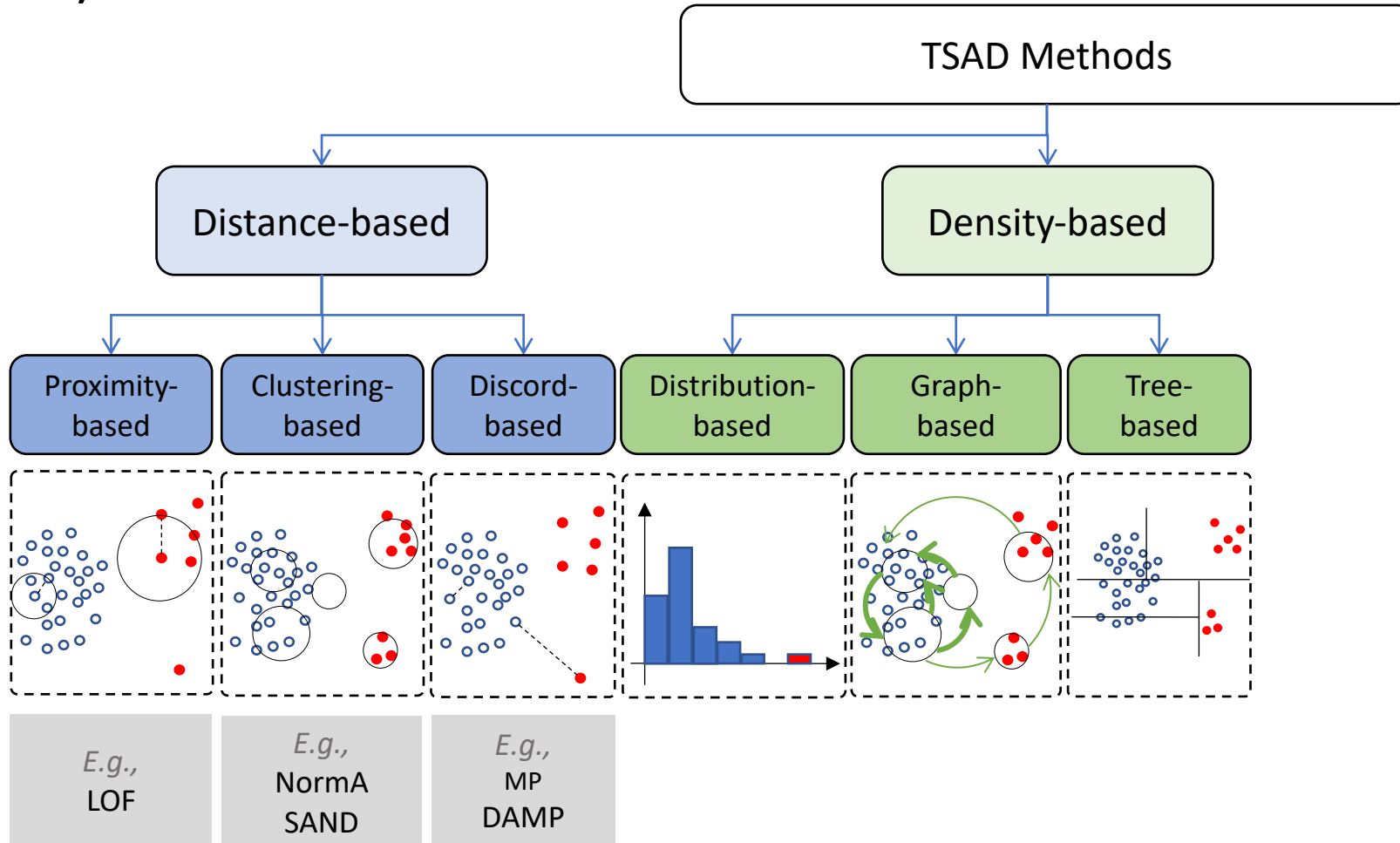
Anomaly Detection methods: *A taxonomy*

By methods...



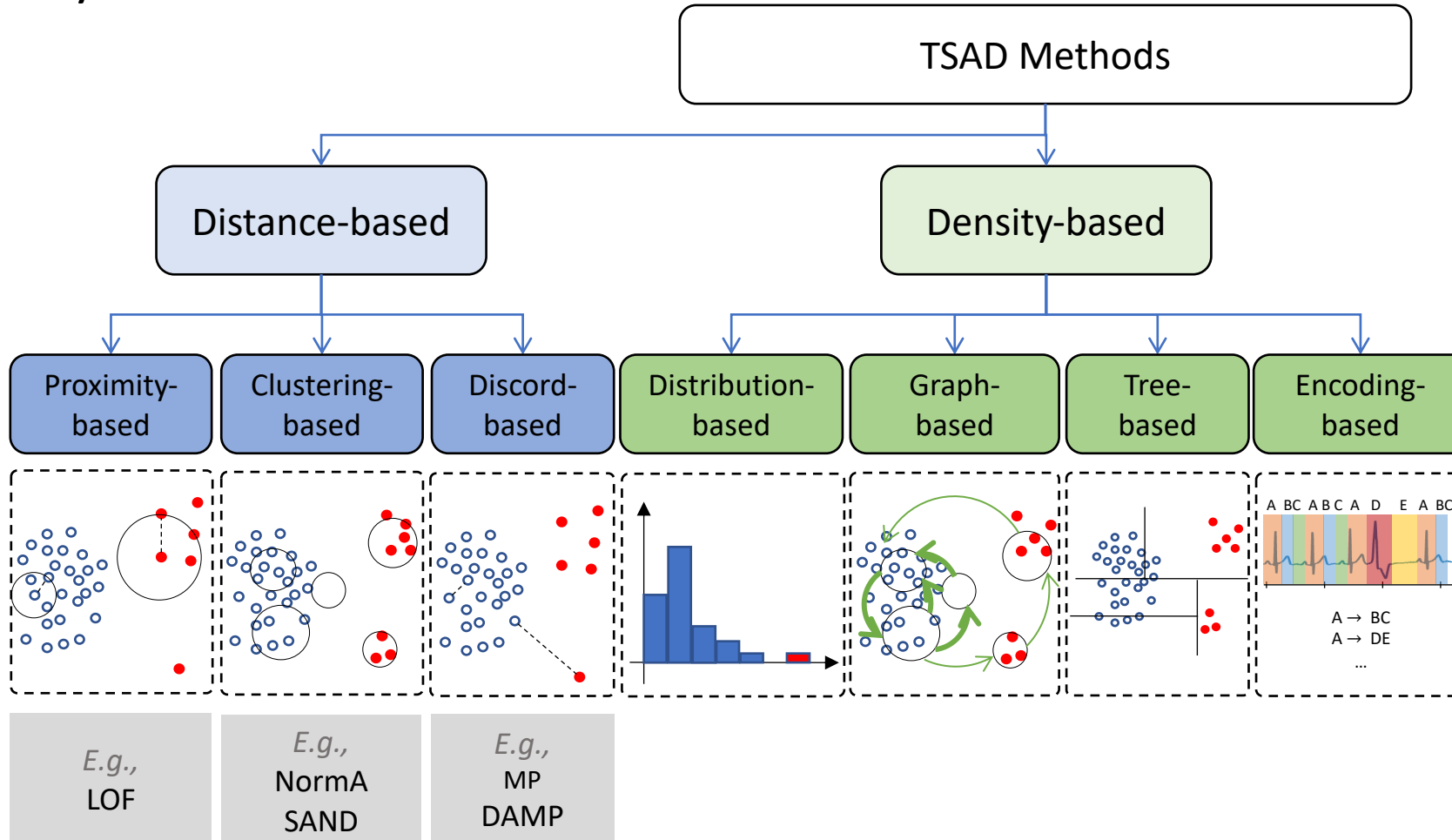
Anomaly Detection methods: *A taxonomy*

By methods...



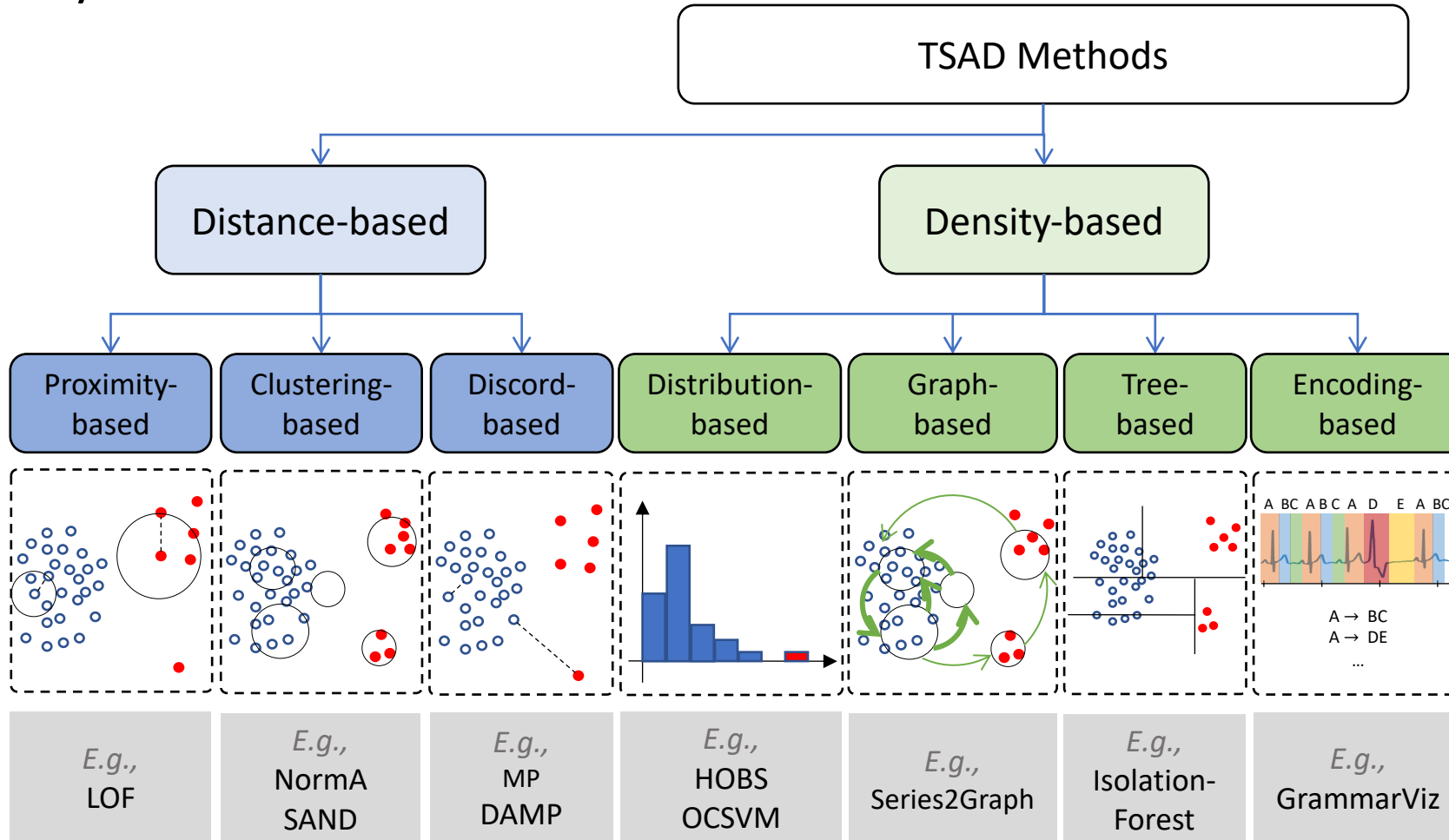
Anomaly Detection methods: *A taxonomy*

By methods...



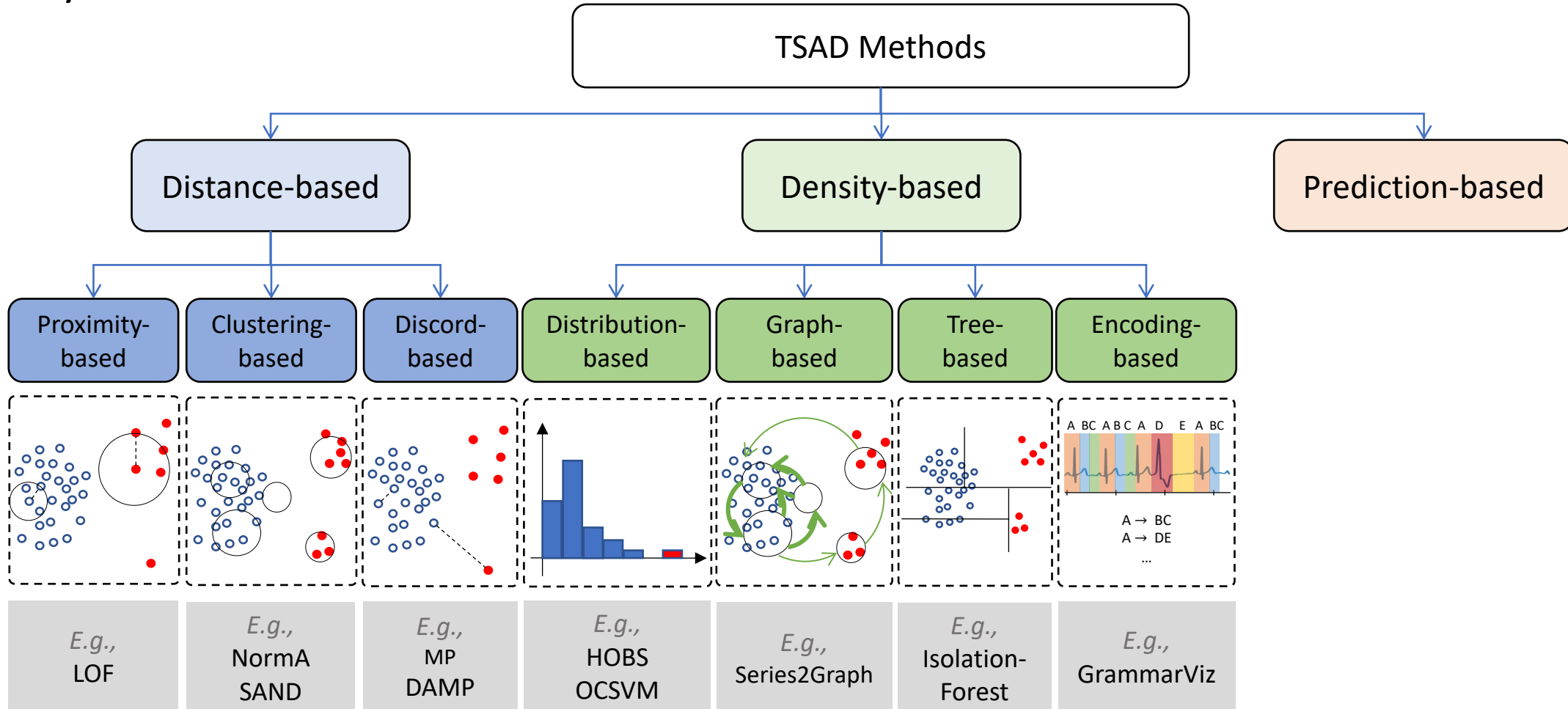
Anomaly Detection methods: *A taxonomy*

By methods...



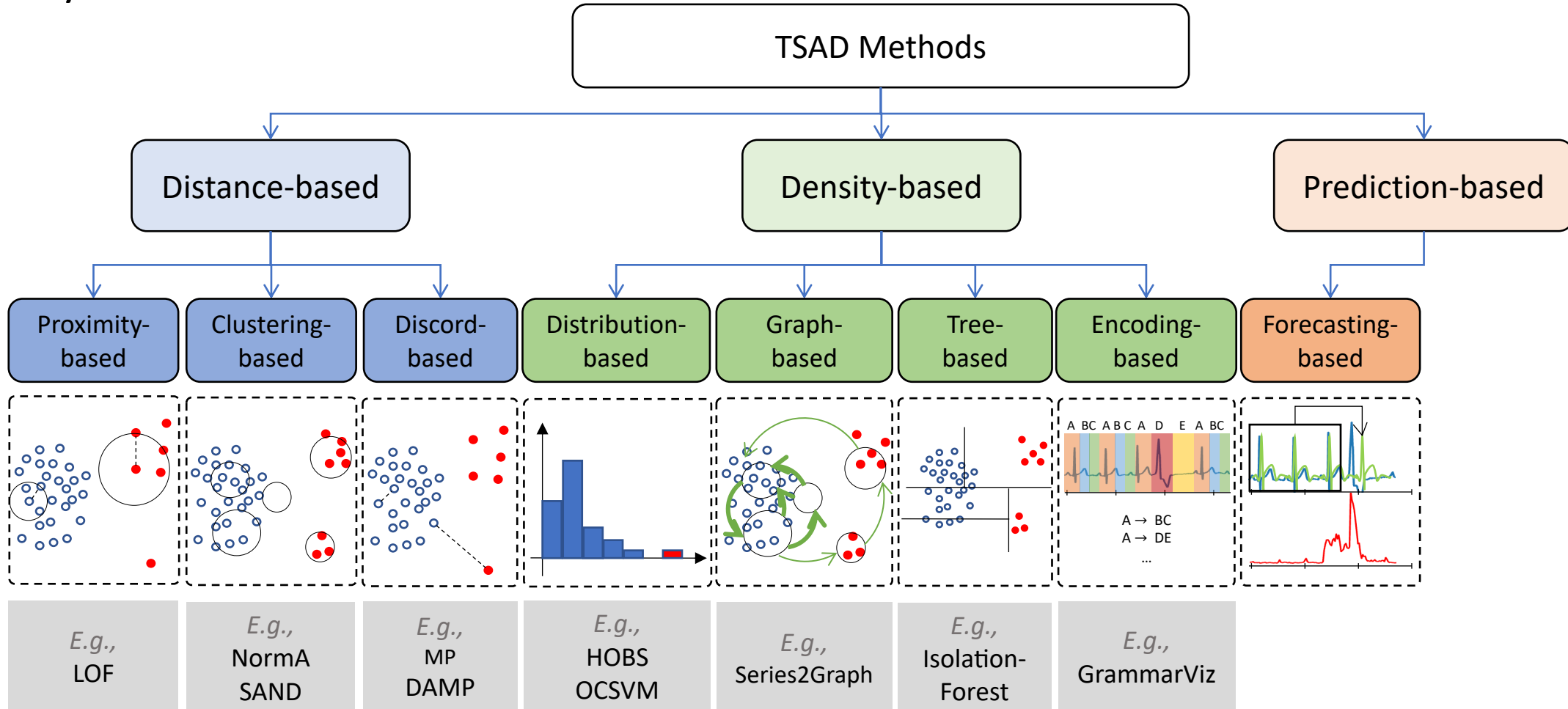
Anomaly Detection methods: *A taxonomy*

By methods...



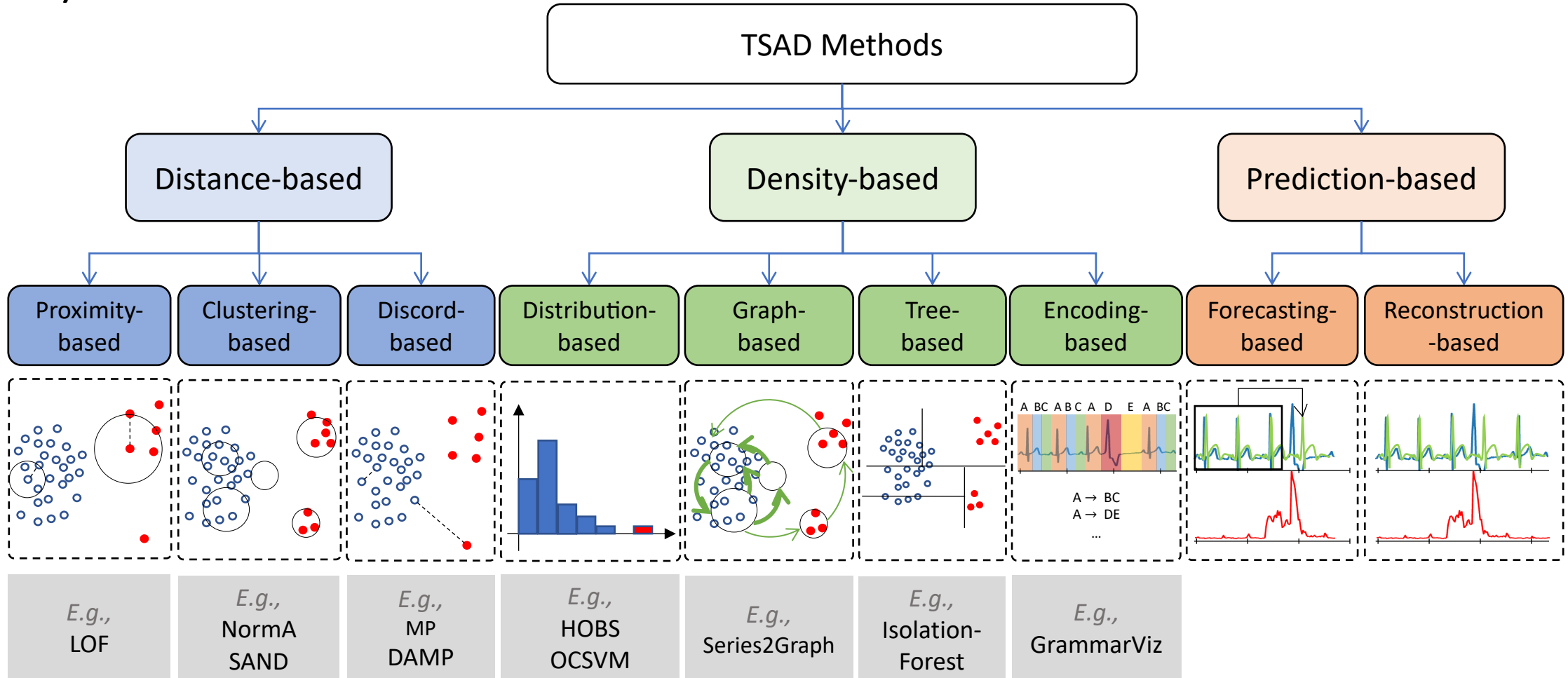
Anomaly Detection methods: *A taxonomy*

By methods...



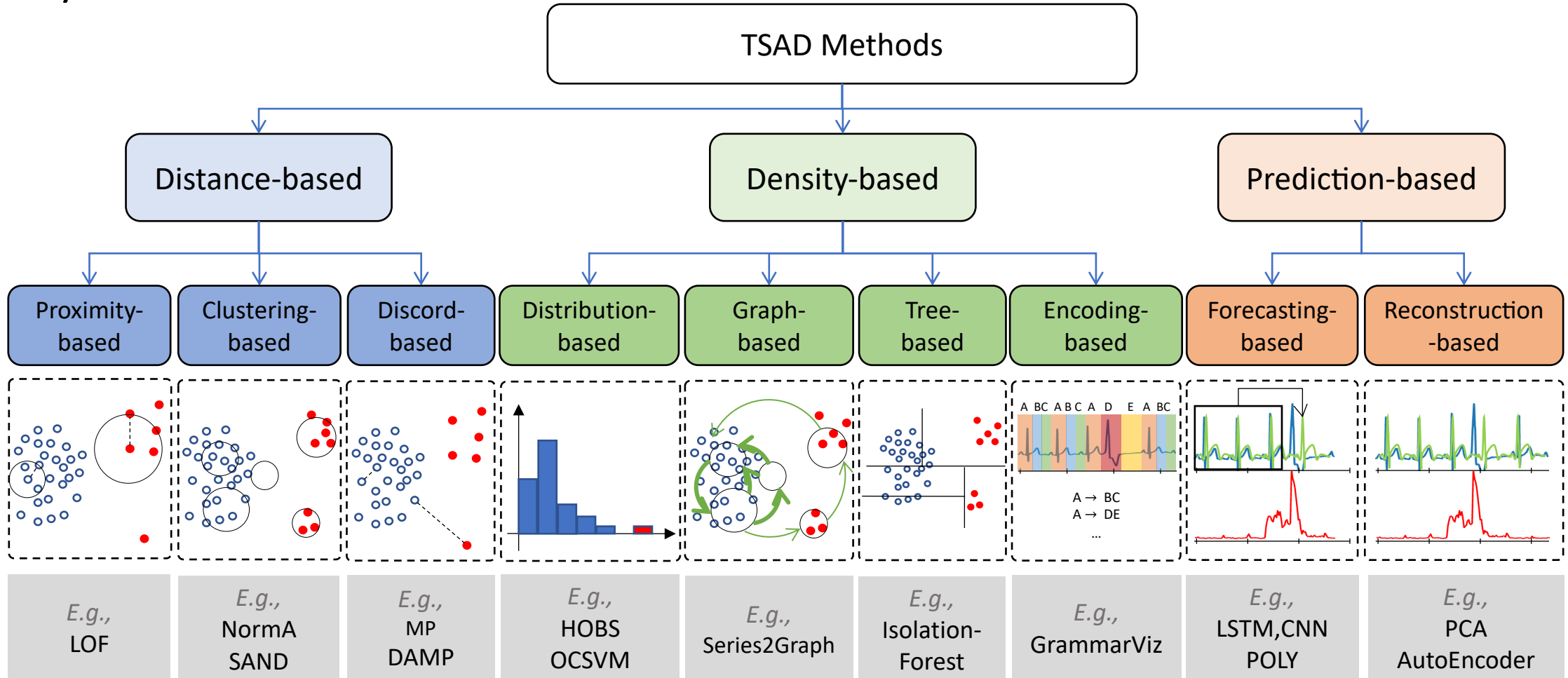
Anomaly Detection methods: *A taxonomy*

By methods...



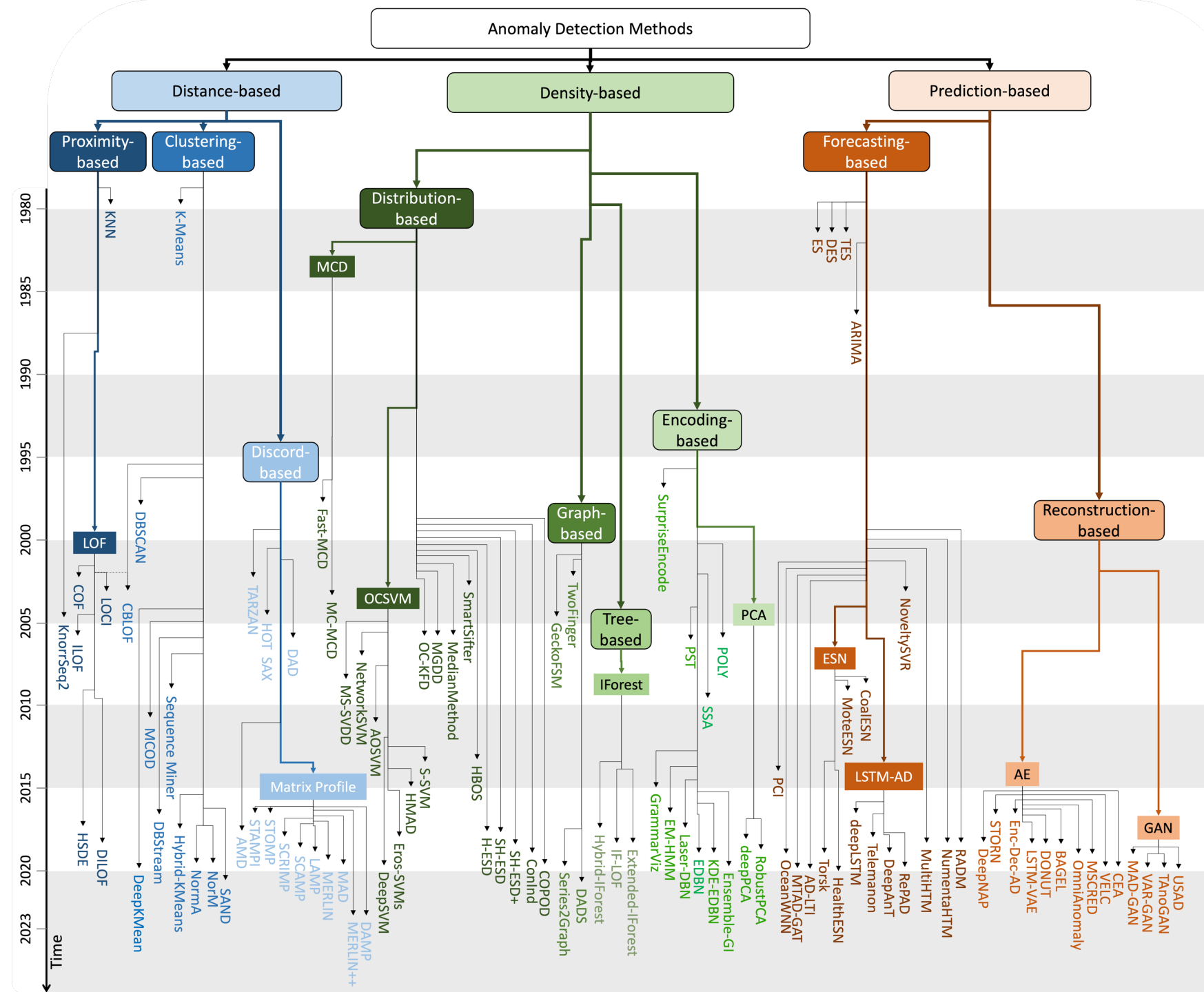
Anomaly Detection methods: *A taxonomy*

By methods...



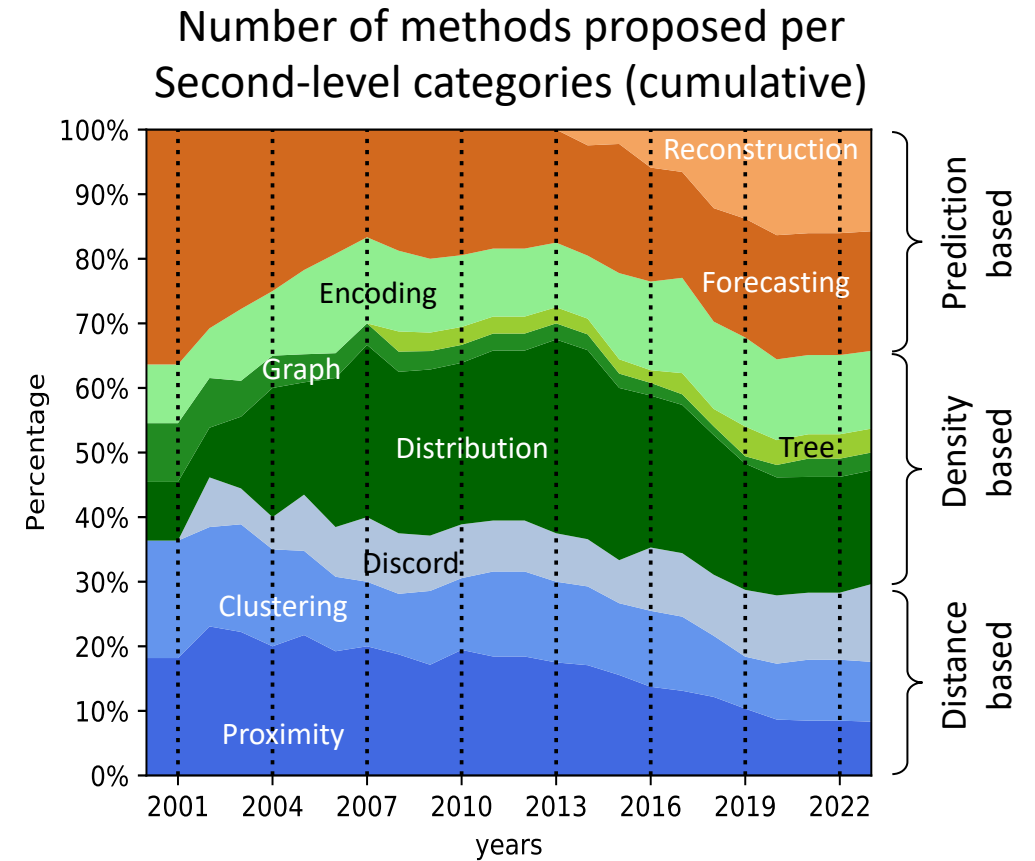
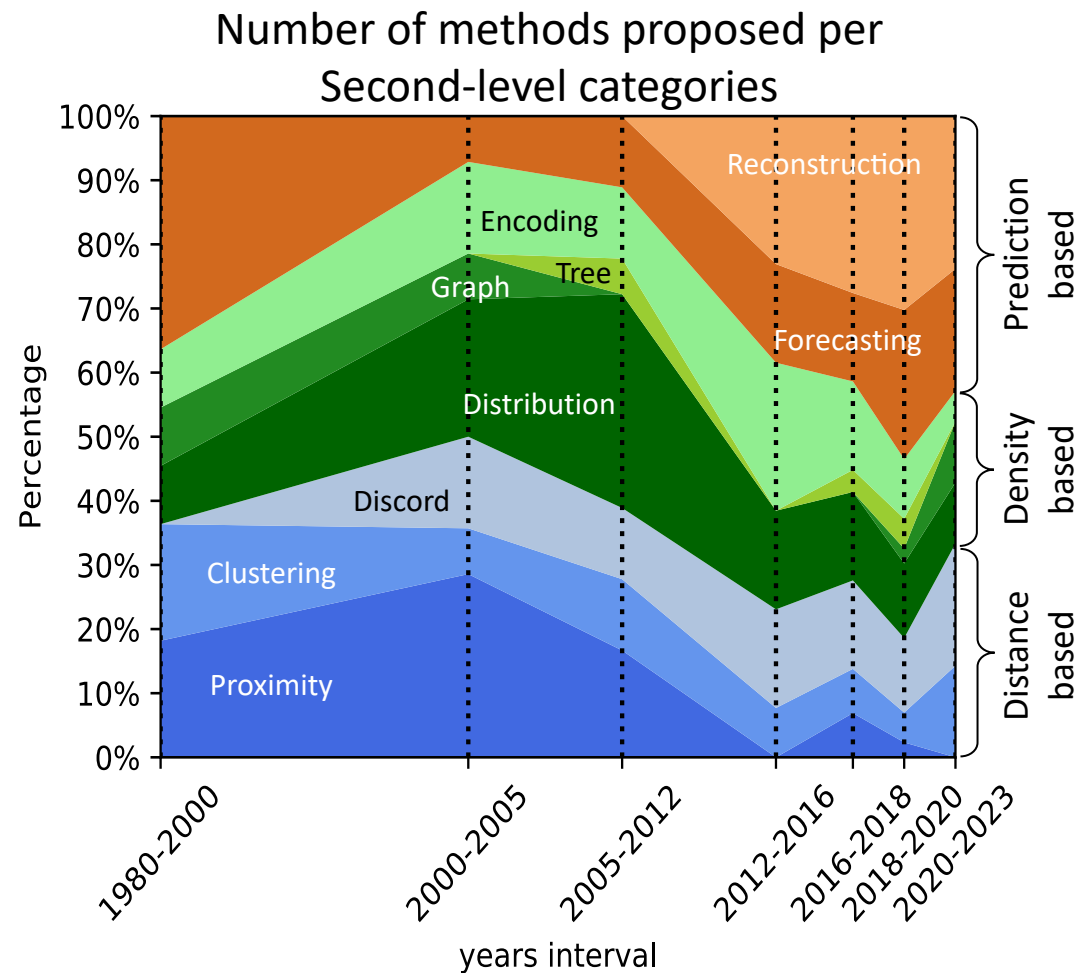
Anomaly Detection methods: *A taxonomy*

By time...



Anomaly Detection methods: *A taxonomy*

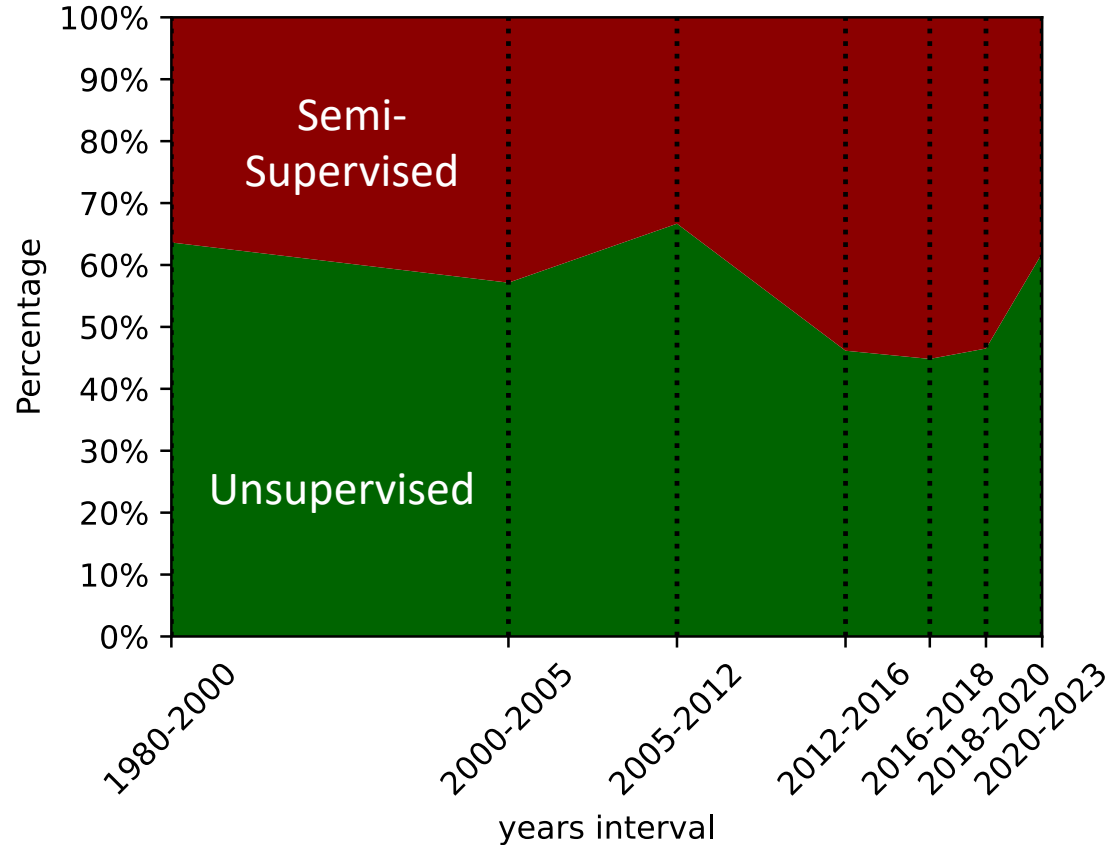
By time...



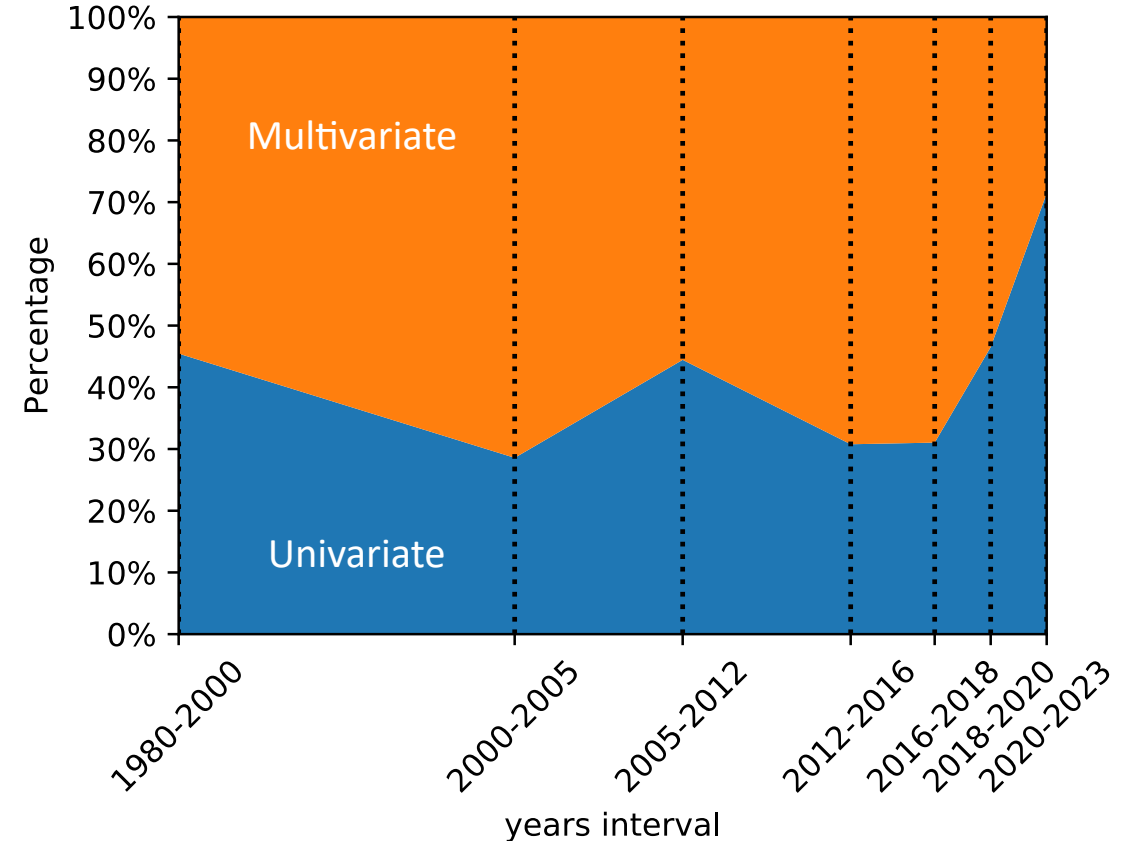
Anomaly Detection methods: *A taxonomy*

By time...

Number of methods proposed that are
Unsupervised or *Semi-Supervised*



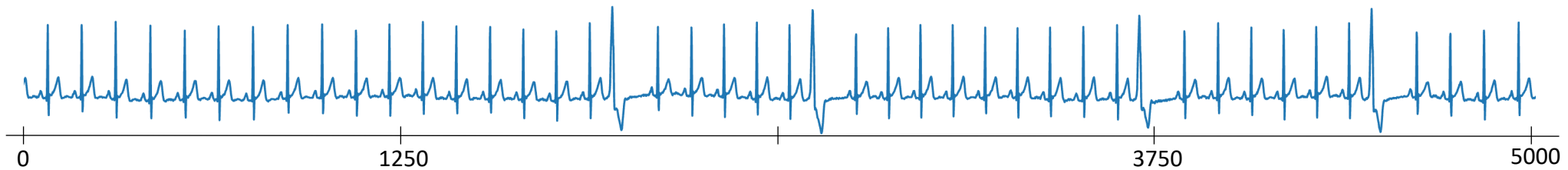
Number of methods proposed that can handle
Univariate or *Multivariate* time series



Anomaly Detection methods: *Distance-based*

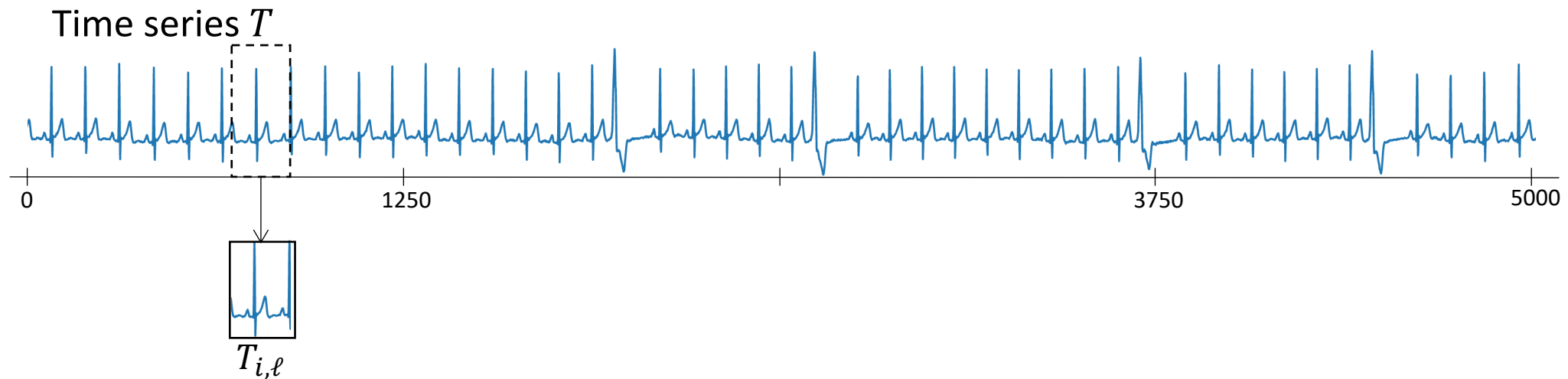
Methods that use **distance computation** between subsequences (or group of subsequences) to detect anomalies.

Time series T



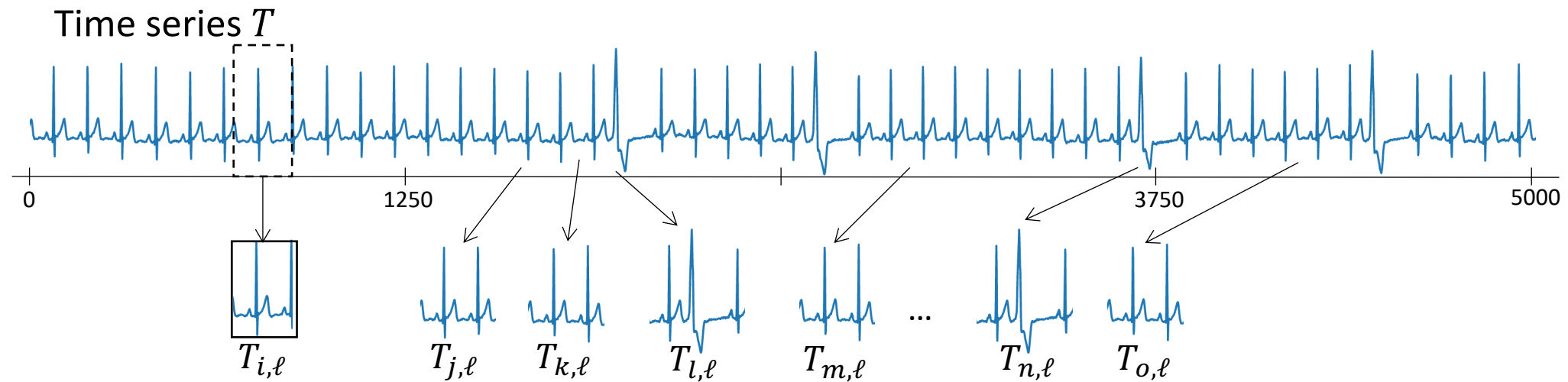
Anomaly Detection methods: *Distance-based*

Methods that use **distance computation** between subsequences (or group of subsequences) to detect anomalies.



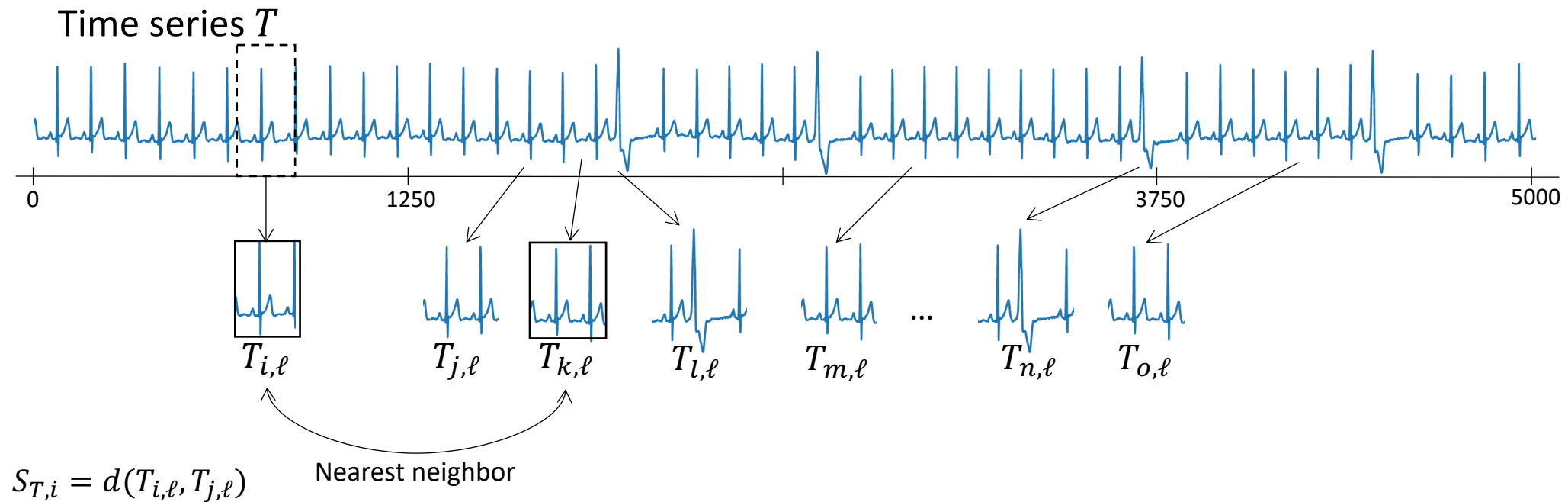
Anomaly Detection methods: *Distance-based*

Methods that use **distance computation** between subsequences (or group of subsequences) to detect anomalies.



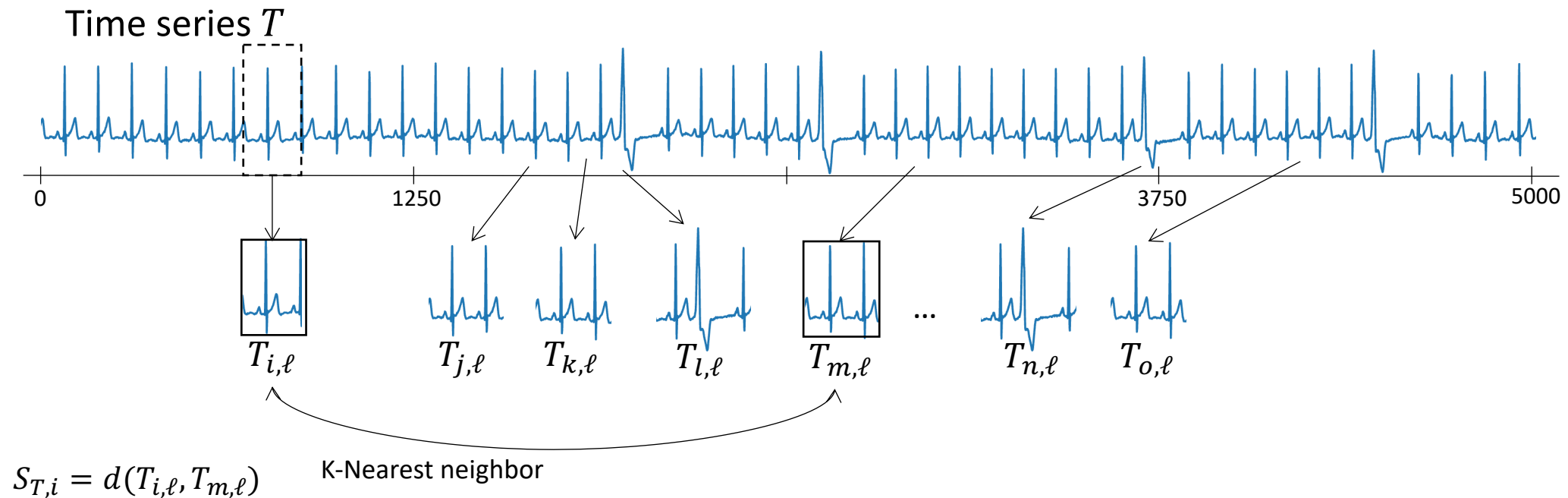
Anomaly Detection methods: *Distance-based*

Methods that use **distance computation** between subsequences (or group of subsequences) to detect anomalies.



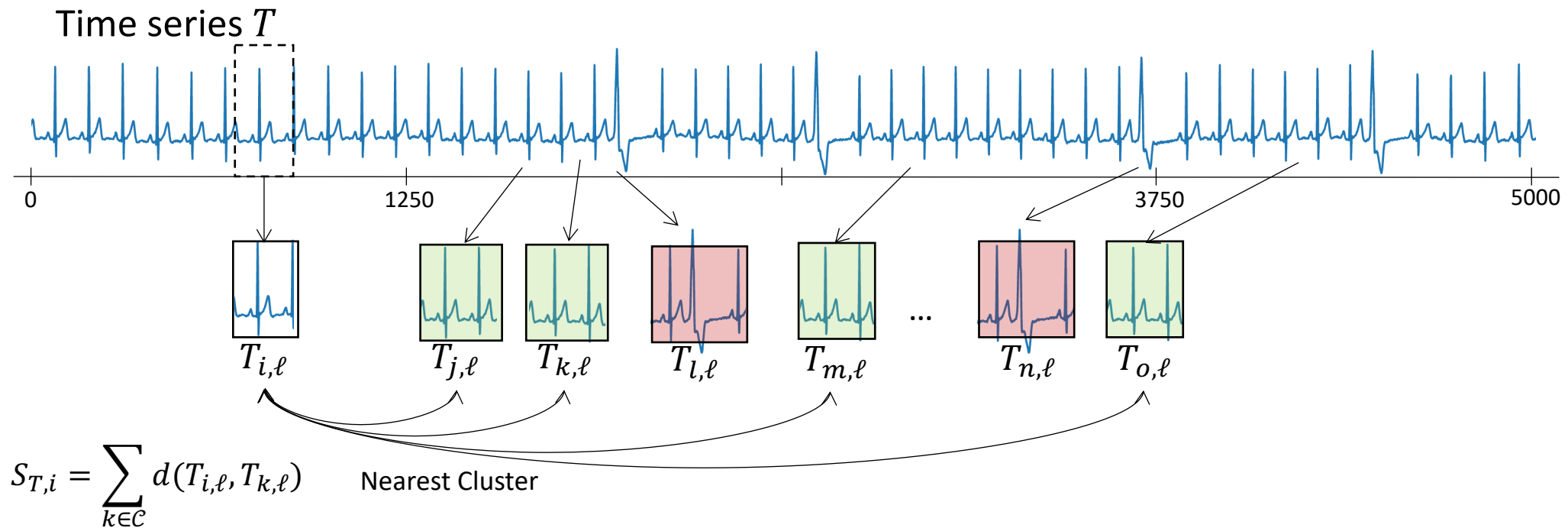
Anomaly Detection methods: *Distance-based*

Methods that use **distance computation** between subsequences (or group of subsequences) to detect anomalies.



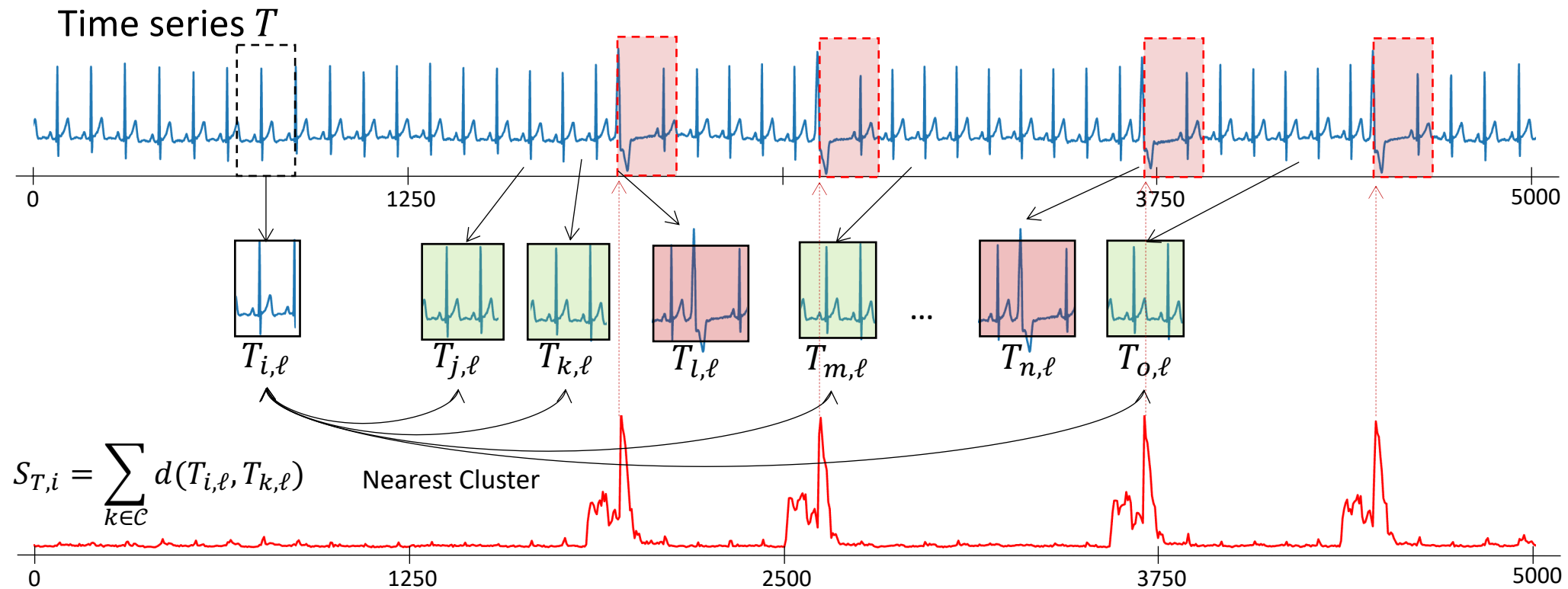
Anomaly Detection methods: *Distance-based*

Methods that use **distance computation** between subsequences (or group of subsequences) to detect anomalies.



Anomaly Detection methods: *Distance-based*

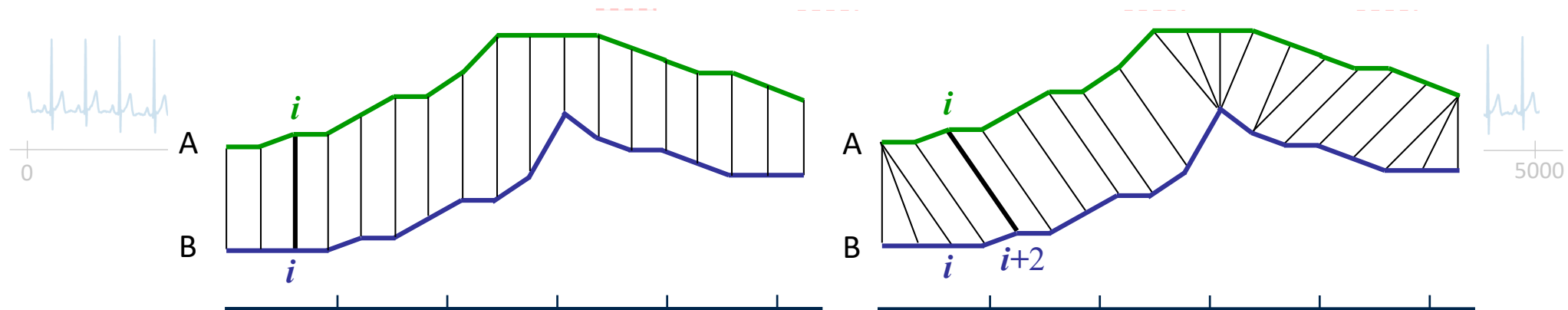
Methods that use **distance computation** between subsequences (or group of subsequences) to detect anomalies.



Anomaly Detection methods: *Distance-based*

Methods that use **distance computation** between subsequences (or group of subsequences) to detect anomalies.

Example of distance computation

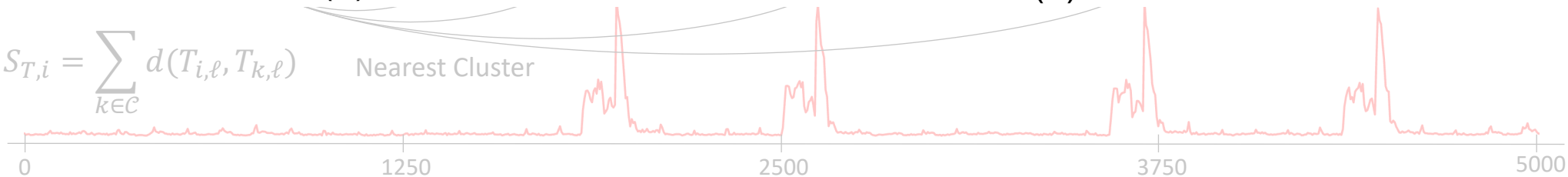


(a) Euclidian Distance

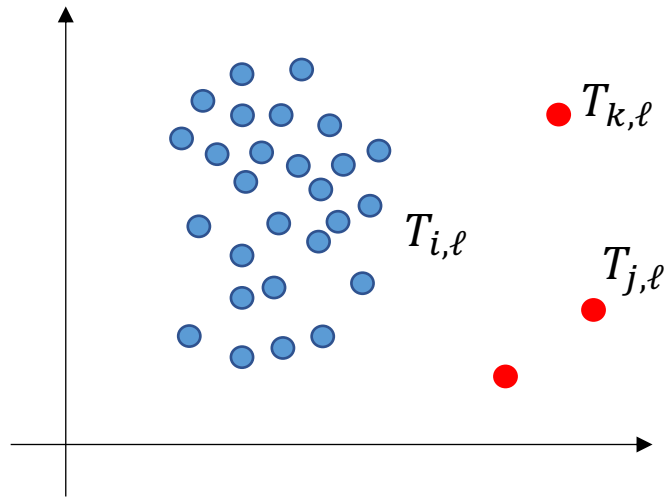
(b) DTW distance

$$S_{T,i} = \sum_{k \in \mathcal{C}} d(T_{i,\ell}, T_{k,\ell})$$

Nearest Cluster



Anomaly Detection methods: *an Example*



Matrix Profile [6] (MP)

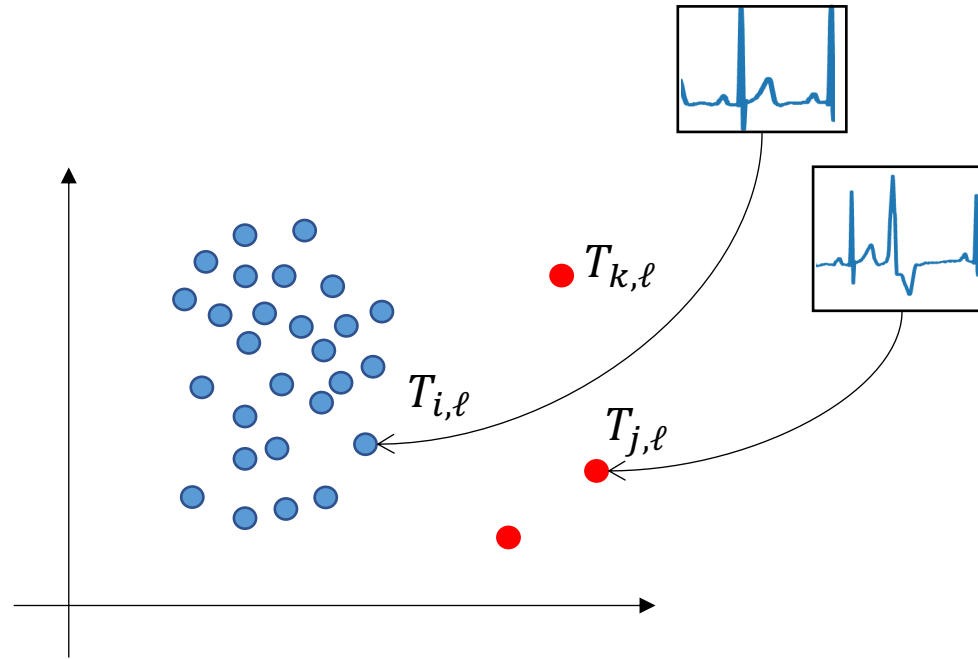
Compute the **distance to the nearest neighbor** (using the MASS algorithm z-norm Euclidean distance computation) and use it as anomaly score

Unsupervised

Univariate

sequence

Anomaly Detection methods: *an Example*



Matrix Profile [6] (MP)

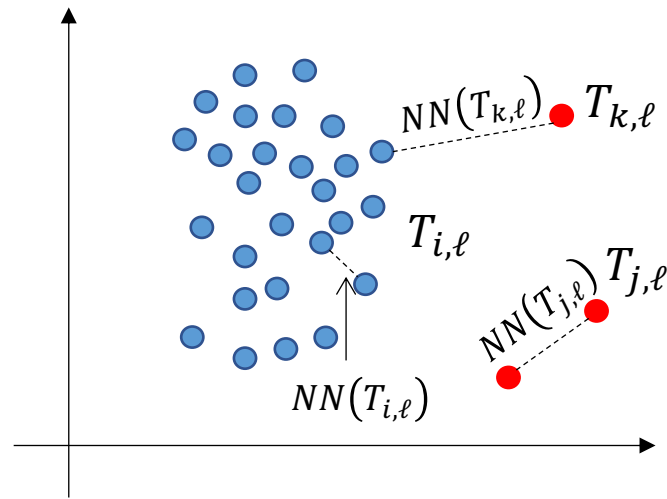
Compute the **distance to the nearest neighbor** (using the MASS algorithm z-norm Euclidean distance computation) and use it as anomaly score

Unsupervised

Univariate

sequence

Anomaly Detection methods: *an Example*



Matrix Profile [6] (MP)

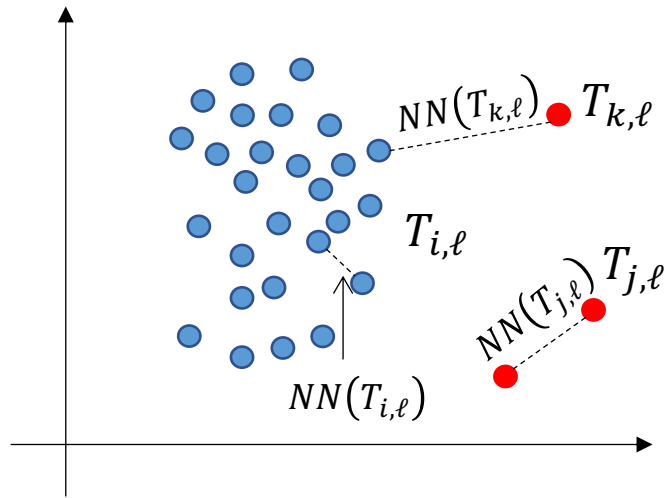
Compute the **distance to the nearest neighbor** (using the MASS algorithm z-norm Euclidean distance computation) and use it as anomaly score

Unsupervised

Univariate

sequence

Anomaly Detection methods: *an Example*



The matrix Profile is computed as follows:

$$S_T = [NN(T_{0,\ell}), NN(T_{1,\ell}), \dots, NN(T_{|T|-\ell,\ell})]$$

Matrix Profile [6] (MP)

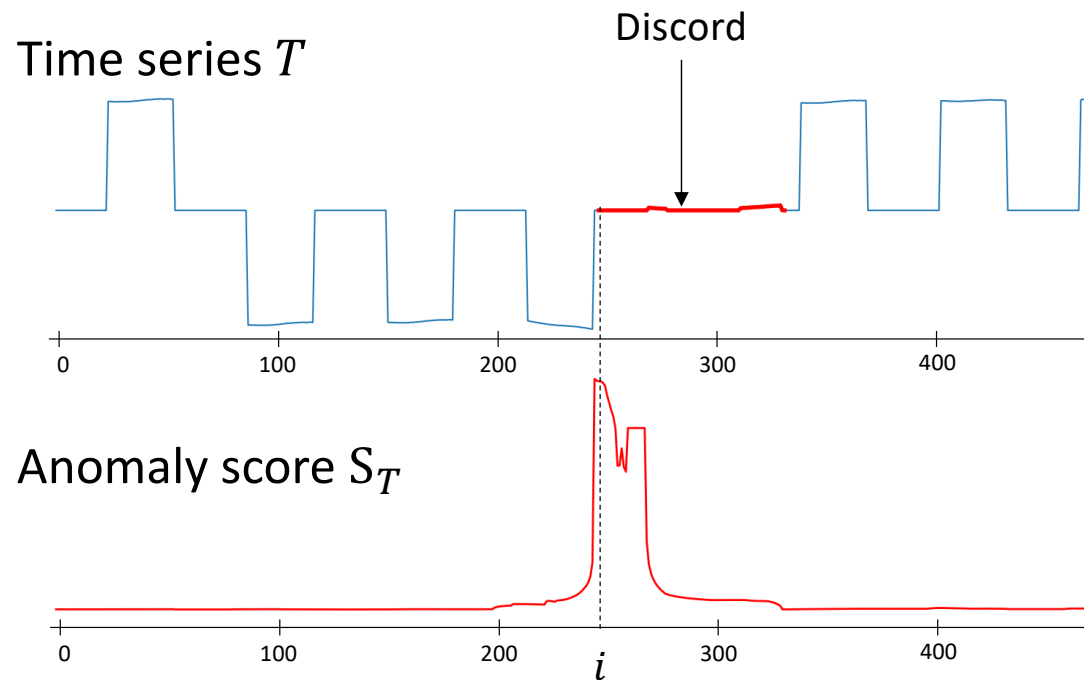
Compute the **distance to the nearest neighbor** (using the MASS algorithm z-norm Euclidean distance computation) and use it as anomaly score

Unsupervised

Univariate

sequence

Anomaly Detection methods: *an Example*



Matrix Profile [6] (MP)

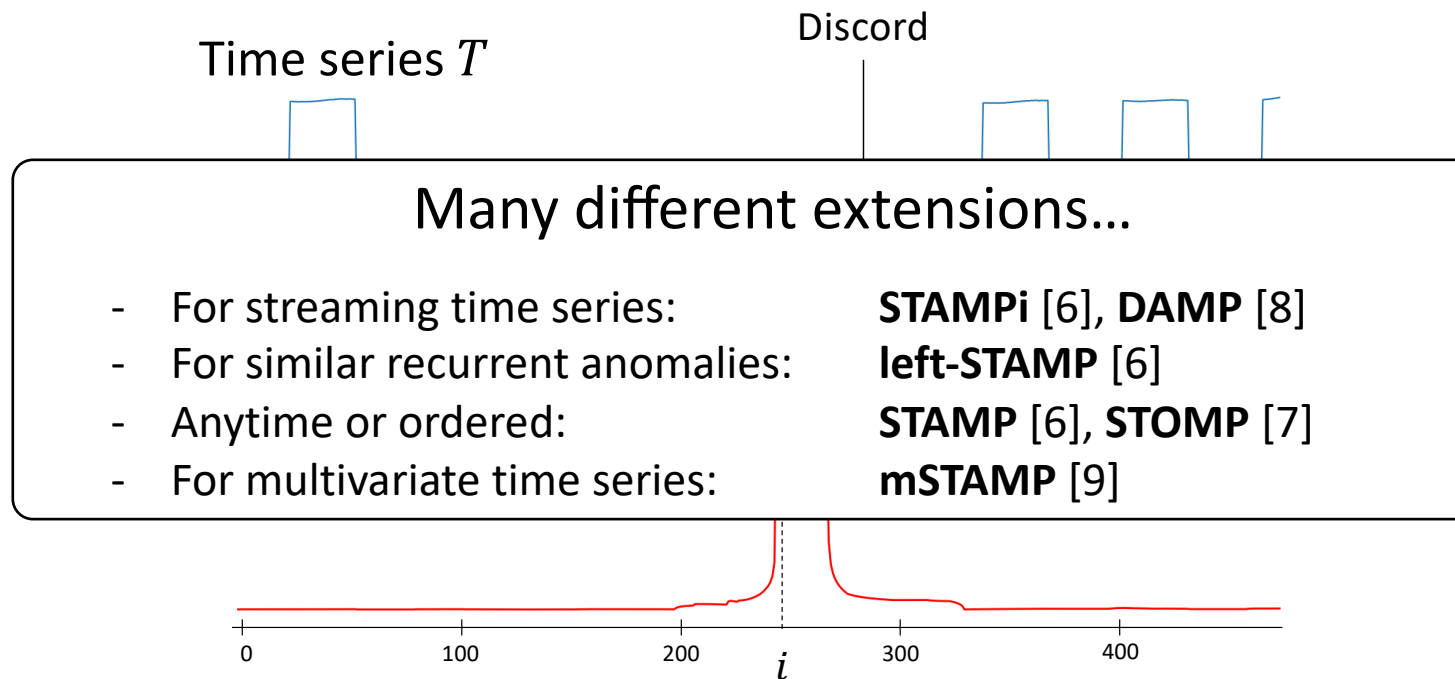
Compute the **distance to the nearest neighbor** (using the MASS algorithm z-norm Euclidean distance computation) and use it as anomaly score

Unsupervised

Univariate

sequence

Anomaly Detection methods: *an Example*



Matrix Profile [6] (MP)

Compute the **distance to the nearest neighbor** (using the MASS algorithm z-norm Euclidean distance computation) and use it as anomaly score

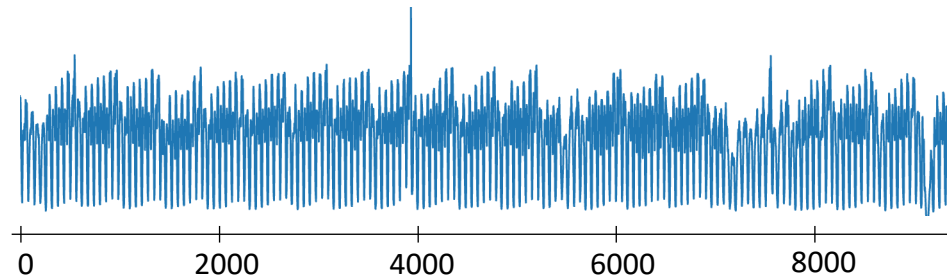
Unsupervised

Univariate

sequence

Anomaly Detection methods: *an Example*

Time series T



NormA [10]

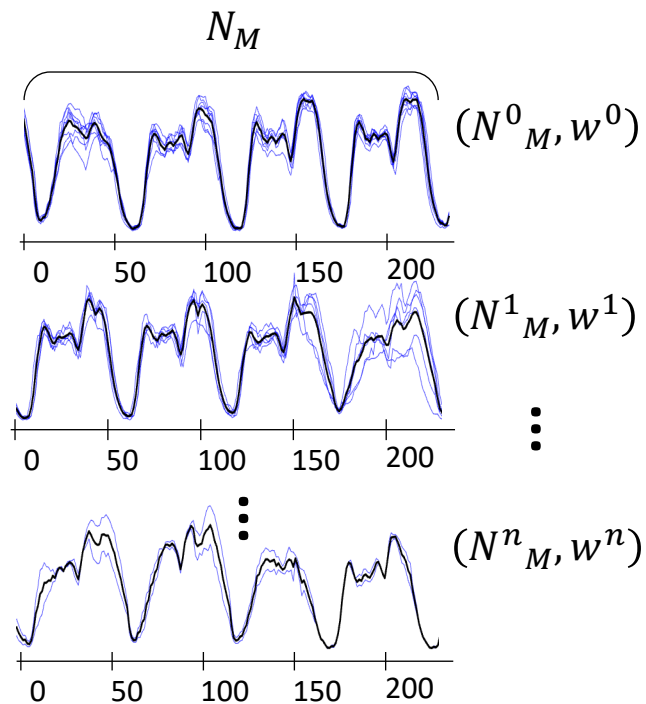
Distance-based approach that **summarize** the time series into **a weighted set of subsequences** and use the distance to them as anomaly score

Unsupervised

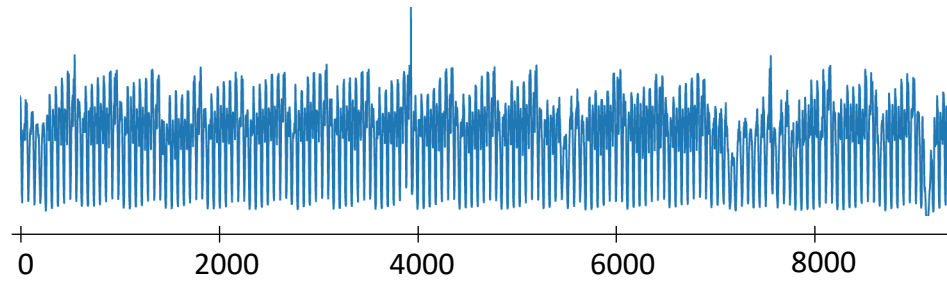
Univariate

sequence

Anomaly Detection methods: *an Example*



Time series T



NormA [10]

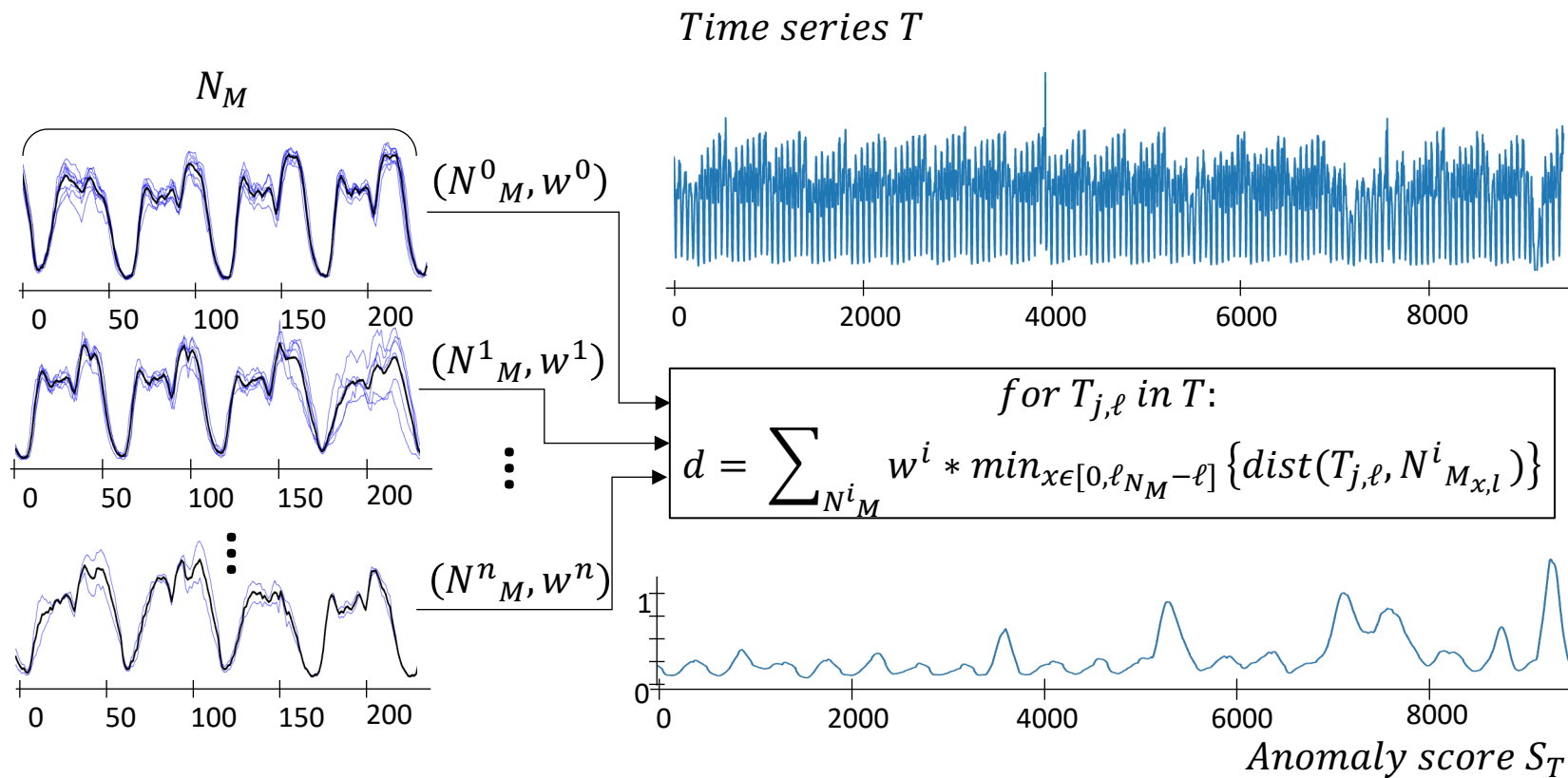
Distance-based approach that **summarize** the time series into **a weighted set of subsequences** and use the distance to them as anomaly score

Unsupervised

Univariate

sequence

Anomaly Detection methods: *an Example*



NormA [10]

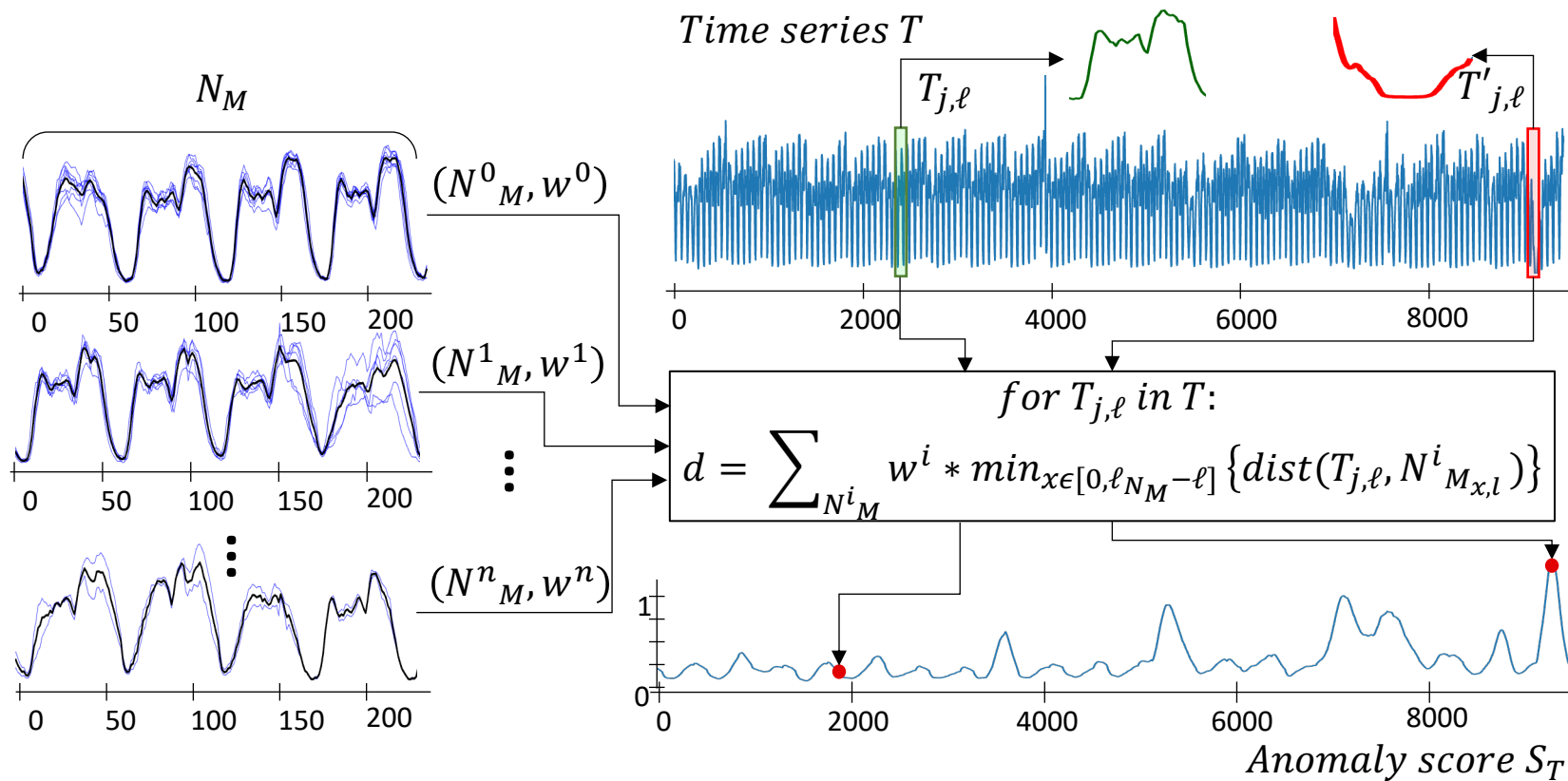
Distance-based approach that **summarize** the time series into **a weighted set of subsequences** and use the distance to them as anomaly score

Unsupervised

Univariate

sequence

Anomaly Detection methods: *an Example*



NormA [10]

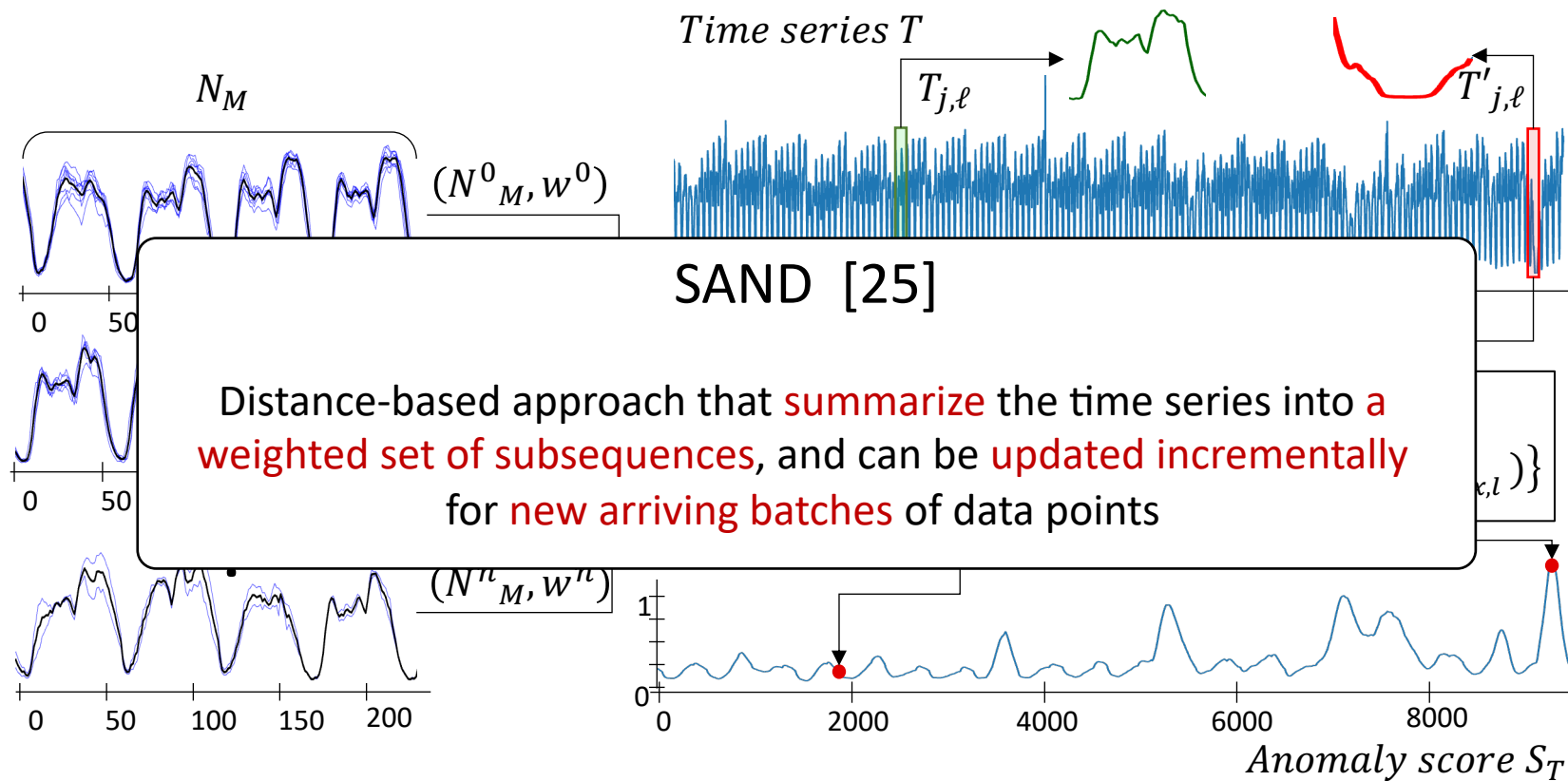
Distance-based approach that **summarize** the time series into **a weighted set of subsequences** and use the distance to them as anomaly score

Unsupervised

Univariate

sequence

Anomaly Detection methods: *an Example*



NormA [10]

Distance-based approach that **summarize** the time series into a **weighted set of subsequences** and use the distance to them as anomaly score

Unsupervised

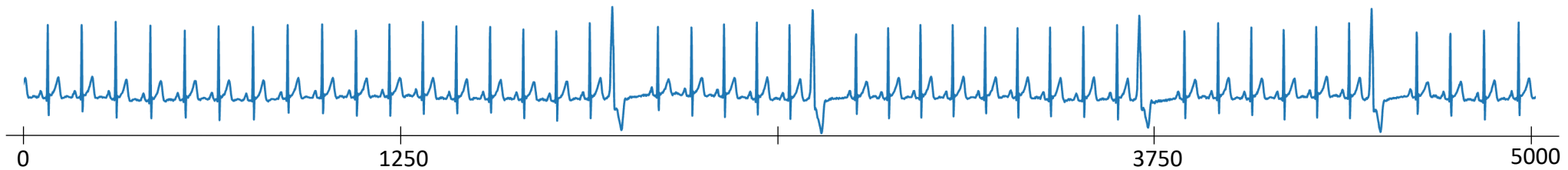
Univariate

sequence

Anomaly Detection methods: *Density-based*

Methods that **estimate the density** of the space (points or subsequences) and identify as anomalies points (or sequences) that are in **low-density subspace**.

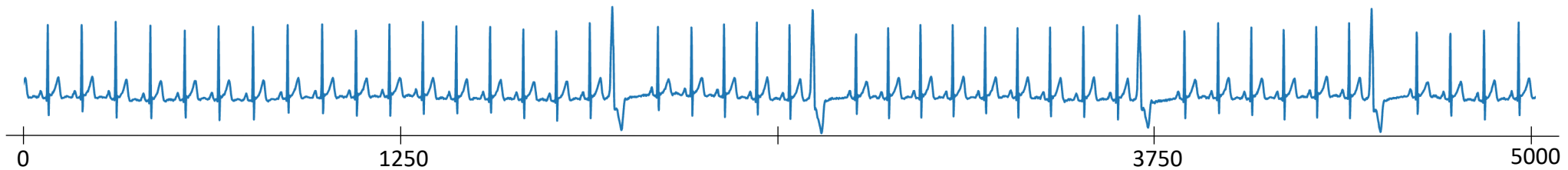
Time series T



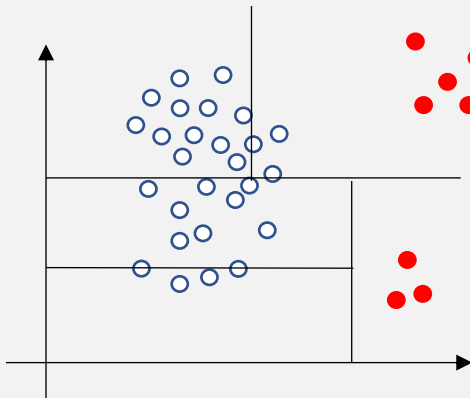
Anomaly Detection methods: *Density-based*

Methods that **estimate the density** of the space (points or subsequences) and identify as anomalies points (or sequences) that are in **low-density subspace**.

Time series T



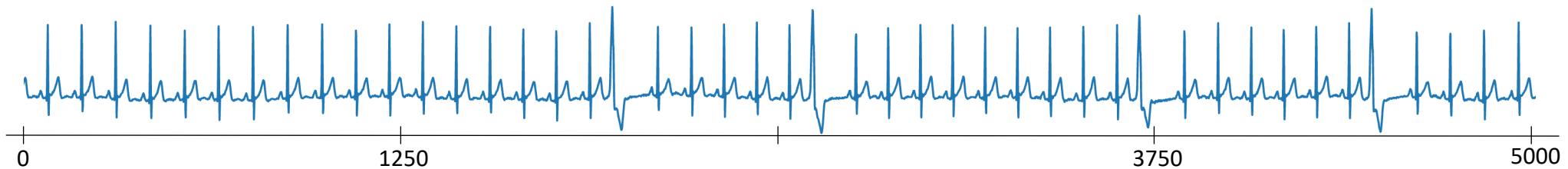
Tree-based approaches [11]



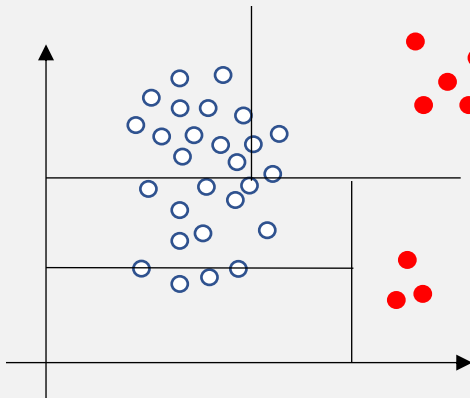
Anomaly Detection methods: *Density-based*

Methods that **estimate the density** of the space (points or subsequences) and identify as anomalies points (or sequences) that are in **low-density subspace**.

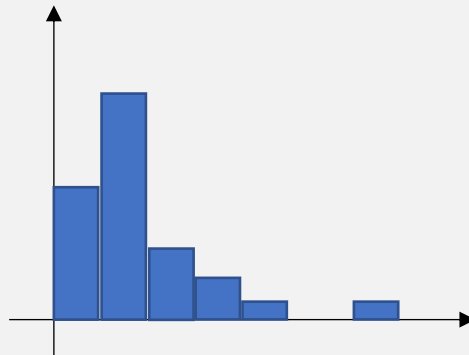
Time series T



Tree-based approaches [11]



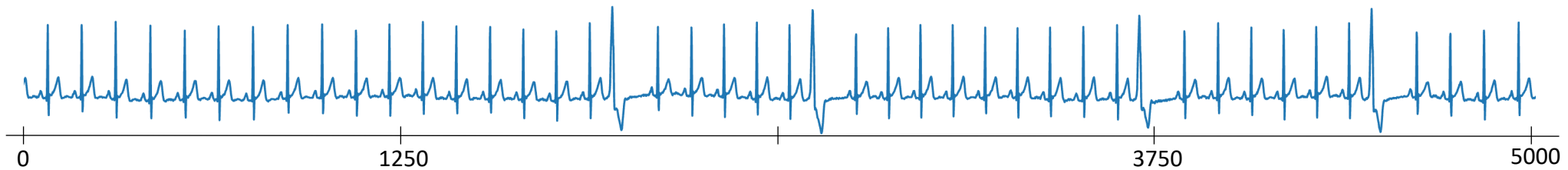
Distribution-based Approaches [12]



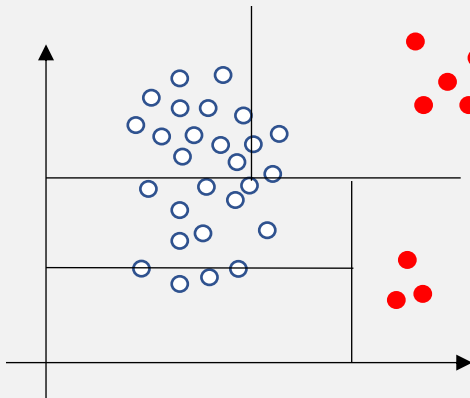
Anomaly Detection methods: *Density-based*

Methods that **estimate the density** of the space (points or subsequences) and identify as anomalies points (or sequences) that are in **low-density subspace**.

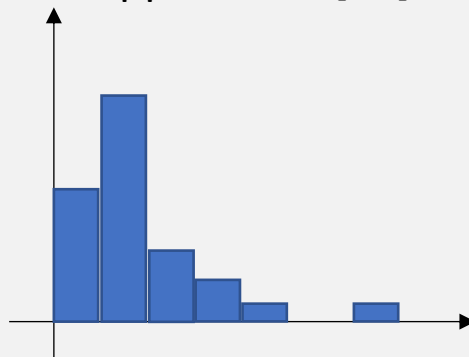
Time series T



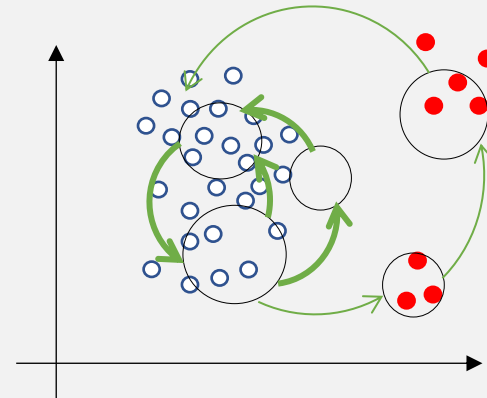
Tree-based approaches [11]



Distribution-based
Approaches [12]

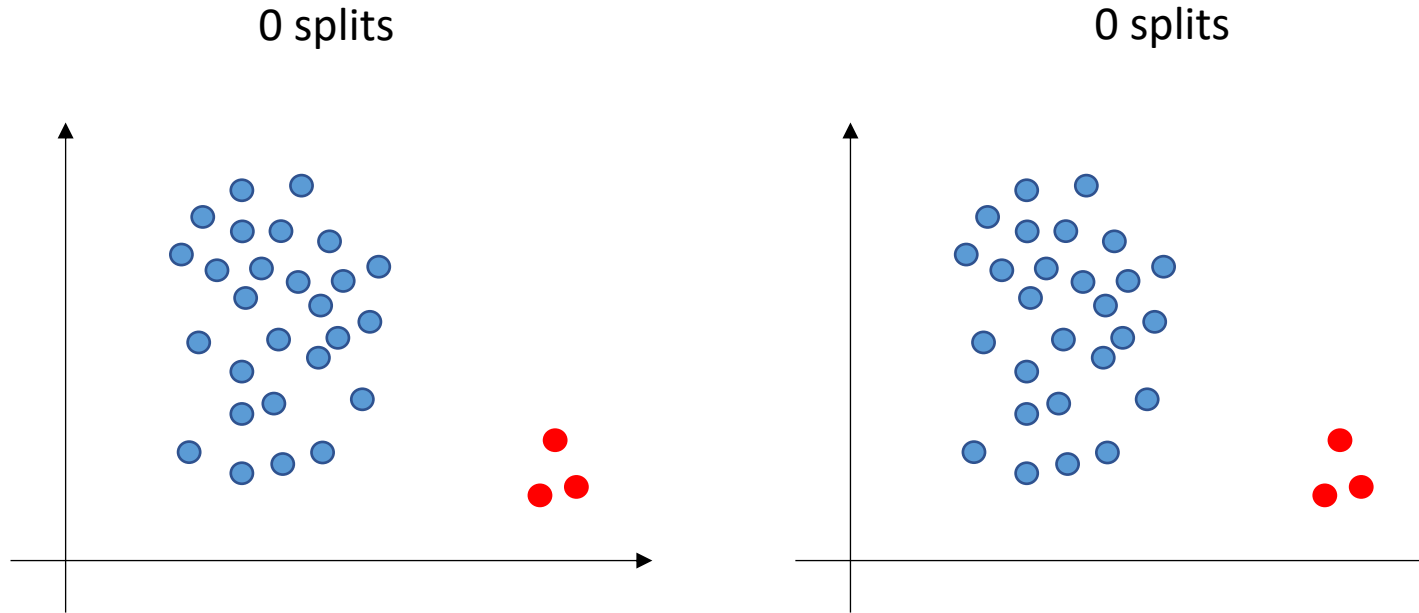


Graph-based approaches [13]



...

Anomaly Detection methods: *an Example*



Isolation Forest [11]

Density-based approach that **split the space randomly** and using the **depth of the trees** to identify anomalies

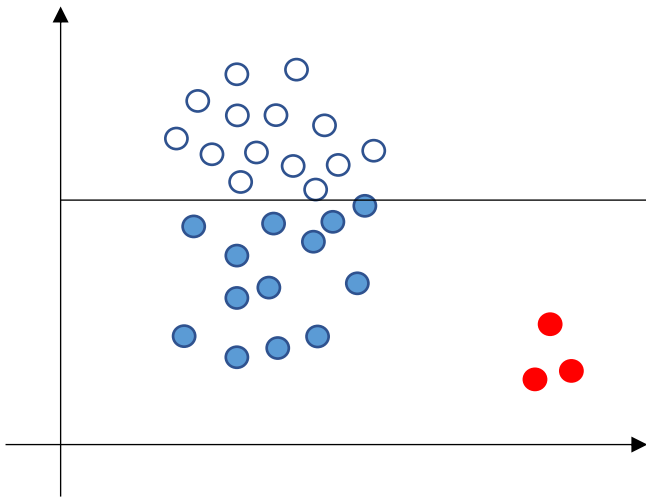
Unsupervised

Univariate/Multivariate

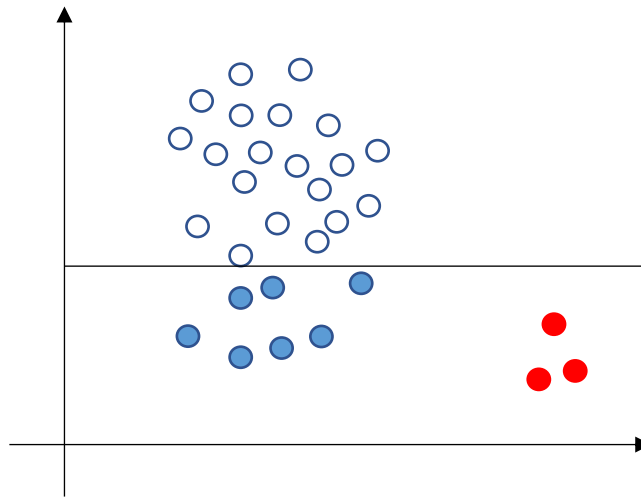
Point/sequence

Anomaly Detection methods: *an Example*

1 splits



1 splits



Isolation Forest [11]

Density-based approach that **split the space randomly** and using the **depth of the trees** to identify anomalies

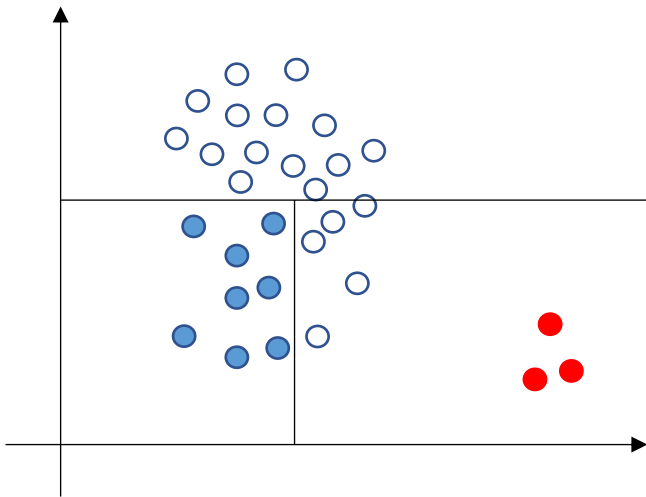
Unsupervised

Univariate/Multivariate

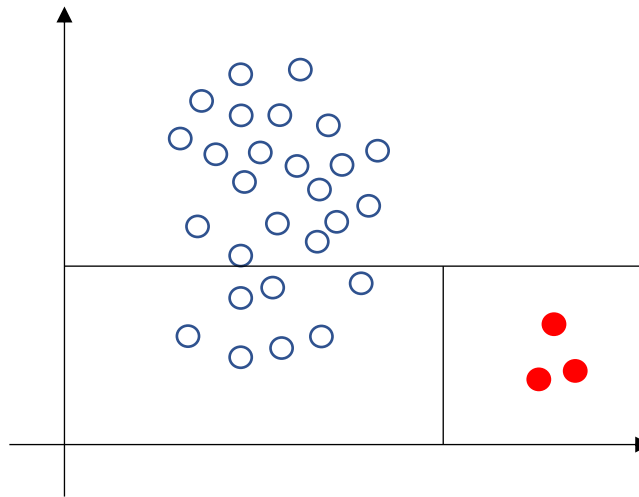
Point/sequence

Anomaly Detection methods: *an Example*

2 splits



2 splits



Isolation Forest [11]

Density-based approach that **split the space randomly** and using the **depth of the trees** to identify anomalies

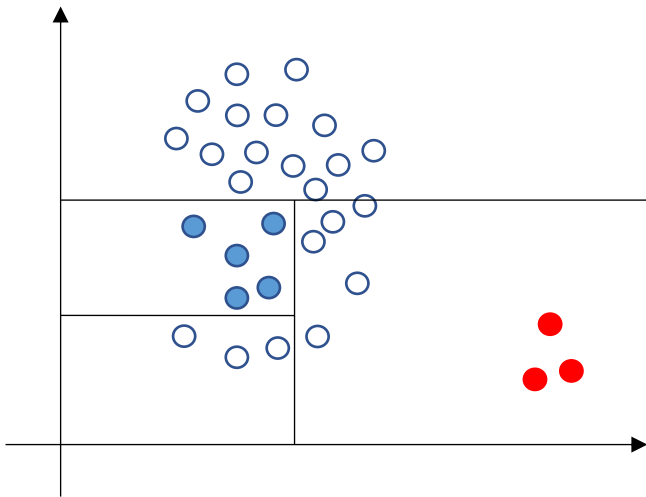
Unsupervised

Univariate/Multivariate

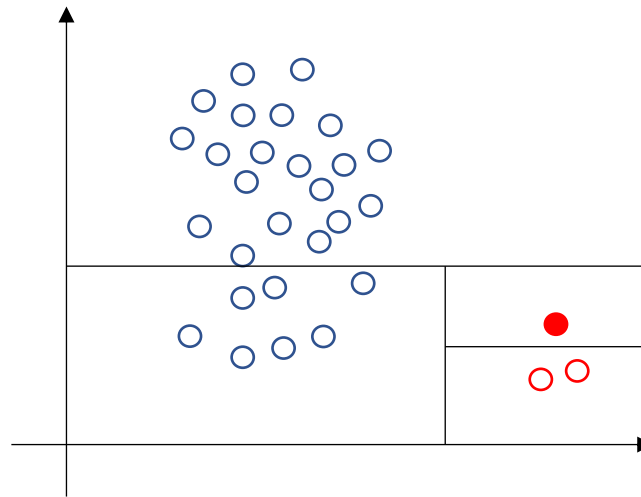
Point/sequence

Anomaly Detection methods: *an Example*

3 splits



3 splits



Isolation Forest [11]

Density-based approach that **split the space randomly** and using the **depth of the trees** to identify anomalies

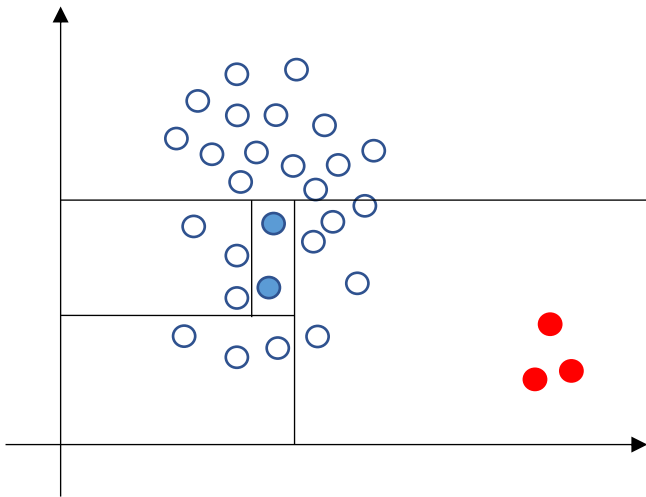
Unsupervised

Univariate/Multivariate

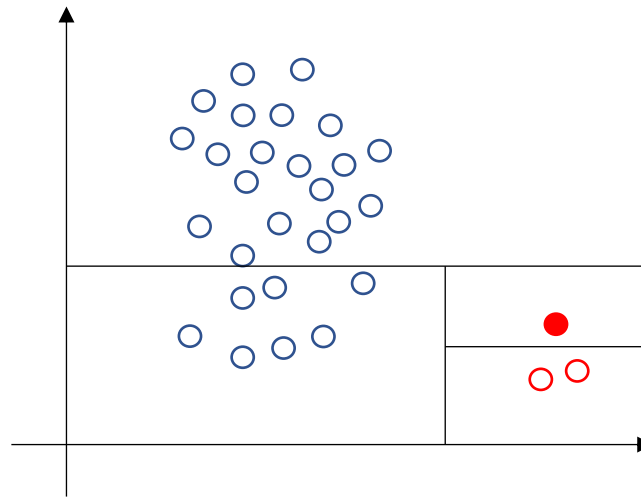
Point/sequence

Anomaly Detection methods: *an Example*

4 splits



3 splits



Isolation Forest [11]

Density-based approach that **split the space randomly** and using the **depth of the trees** to identify anomalies

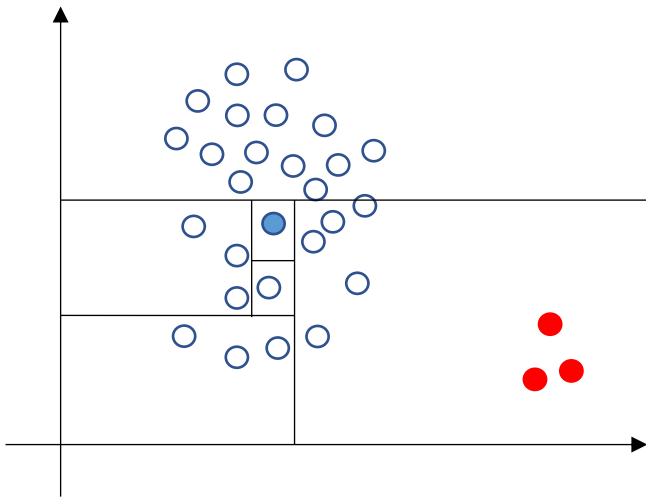
Unsupervised

Univariate/Multivariate

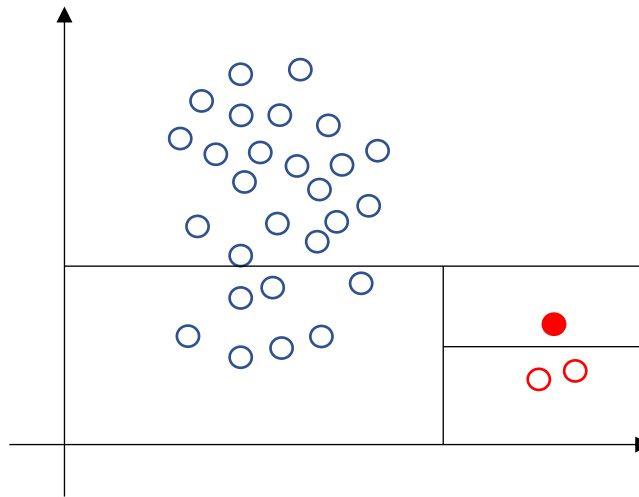
Point/sequence

Anomaly Detection methods: *an Example*

5 splits



3 splits



Isolation Forest [11]

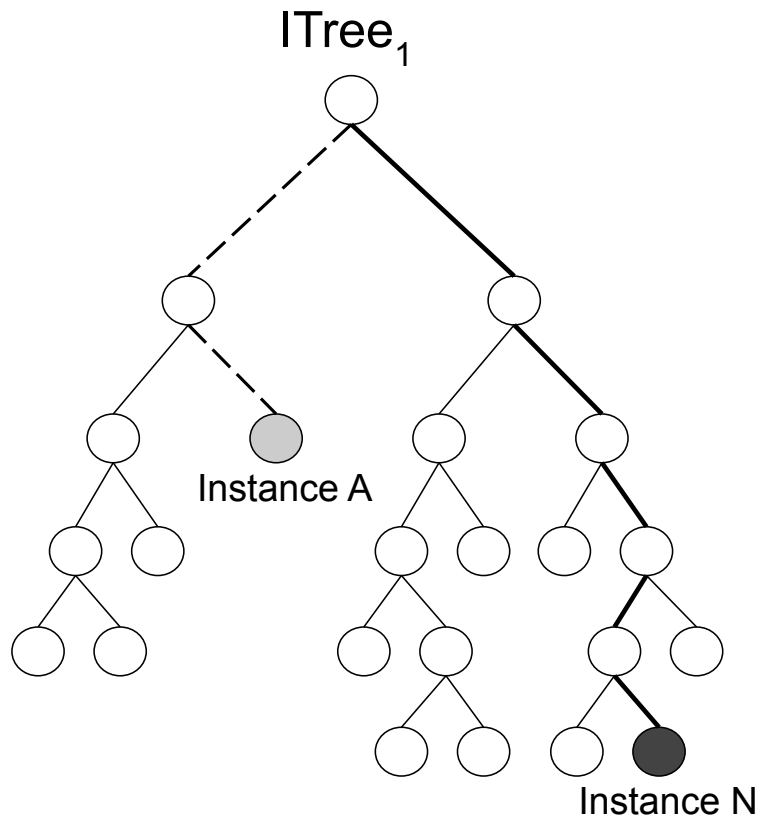
Density-based approach that **split the space randomly** and using the **depth of the trees** to identify anomalies

Unsupervised

Univariate/Multivariate

Point/sequence

Anomaly Detection methods: *an Example*



Isolation Forest [11]

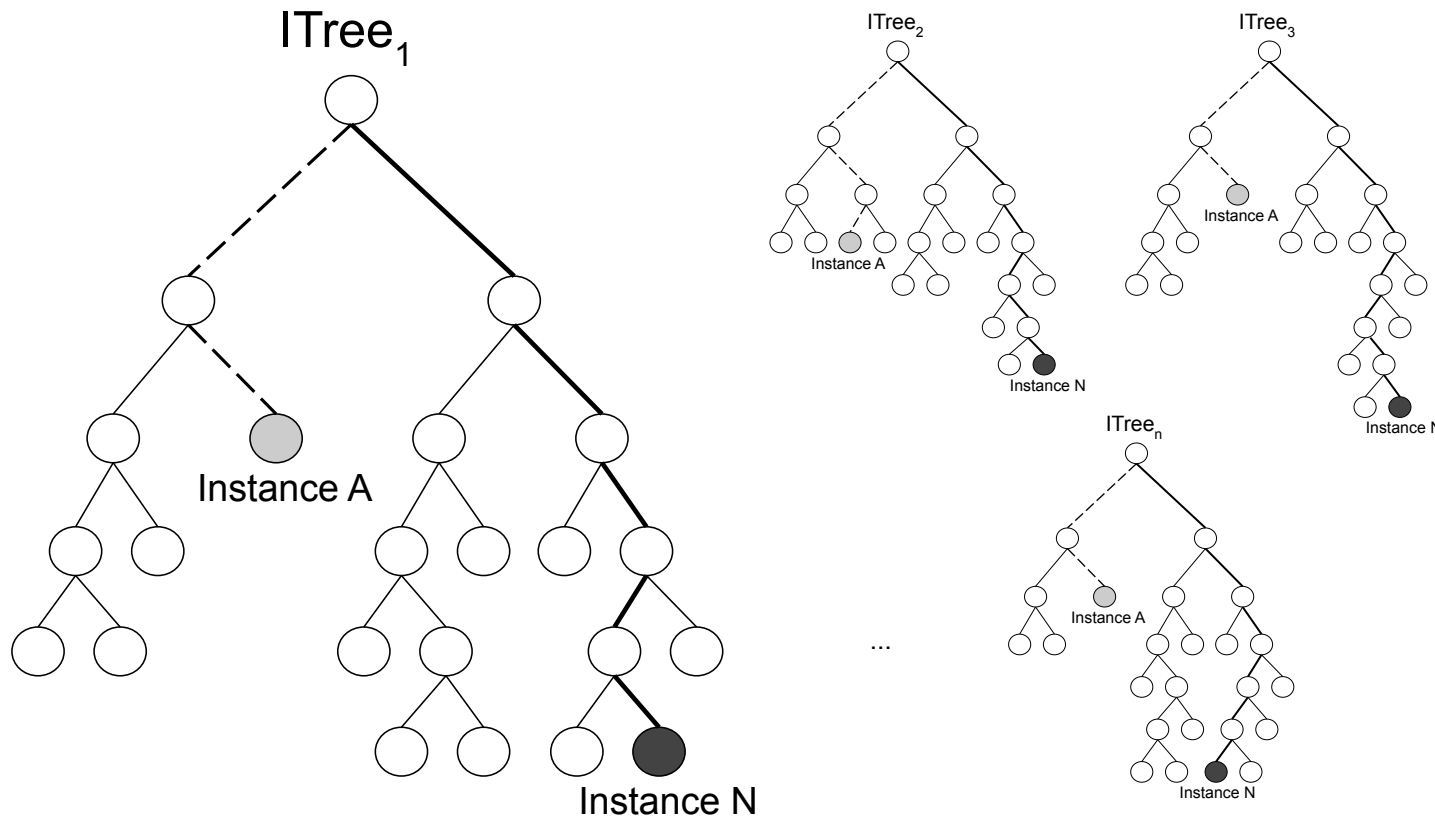
Density-based approach that **split the space randomly** and using the **depth of the trees** to identify anomalies

Unsupervised

Univariate/Multivariate

Point/sequence

Anomaly Detection methods: *an Example*



Isolation Forest [11]

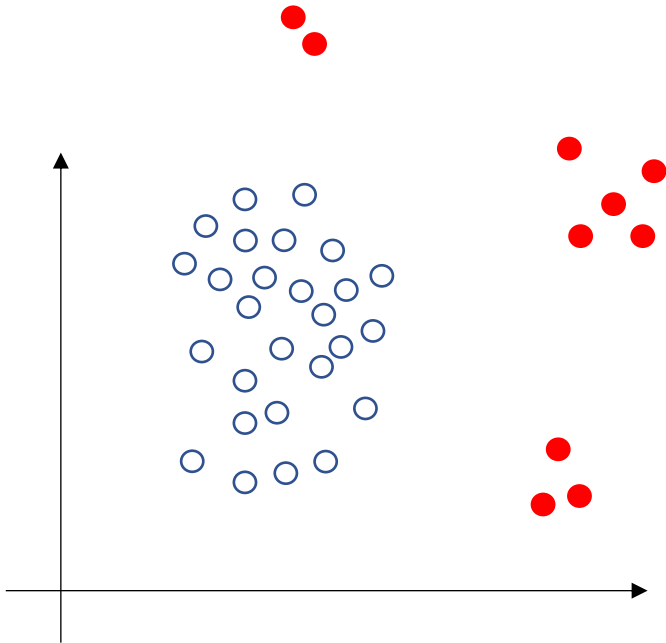
Density-based approach that **split the space randomly** and using the **depth of the trees** to identify anomalies

Unsupervised

Univariate/Multivariate

Point/sequence

Anomaly Detection methods: *an Example*



Series2Graph [13]

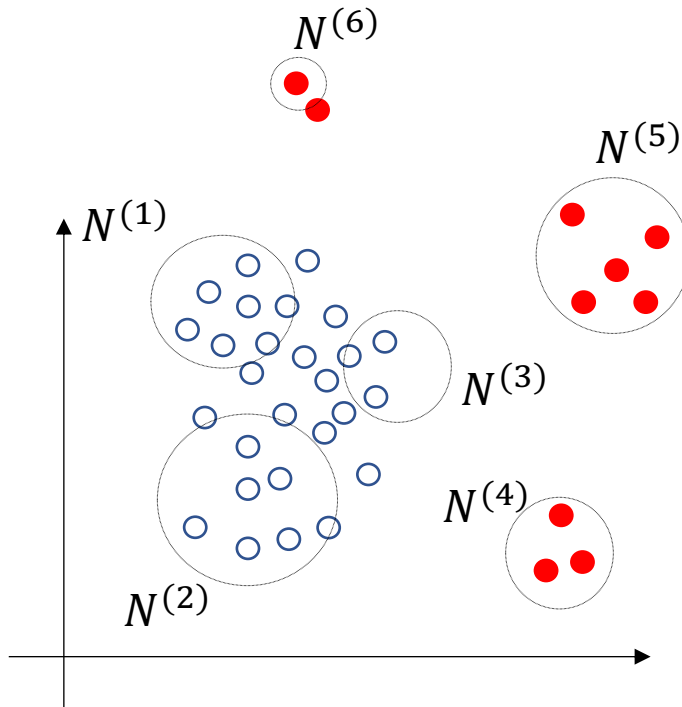
Density-based approach that **convert** the time series into a **graph** and detect **unusual trajectories**

Unsupervised

Univariate

subsequence

Anomaly Detection methods: *an Example*



Each **node** is an ensemble of similar subsequences.

Series2Graph [13]

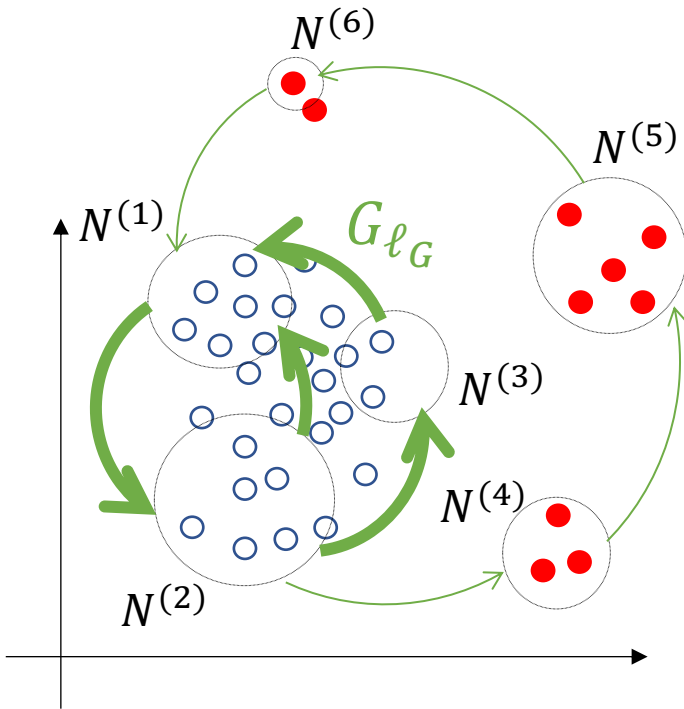
Density-based approach that **convert** the time series into a **graph** and detect **unusual trajectories**

Unsupervised

Univariate

subsequence

Anomaly Detection methods: *an Example*



Each **node** is an ensemble of similar subsequences.

Each **edge** is associated to a weight w that corresponds to the number of times a subsequence move from one node to another.

Series2Graph [13]

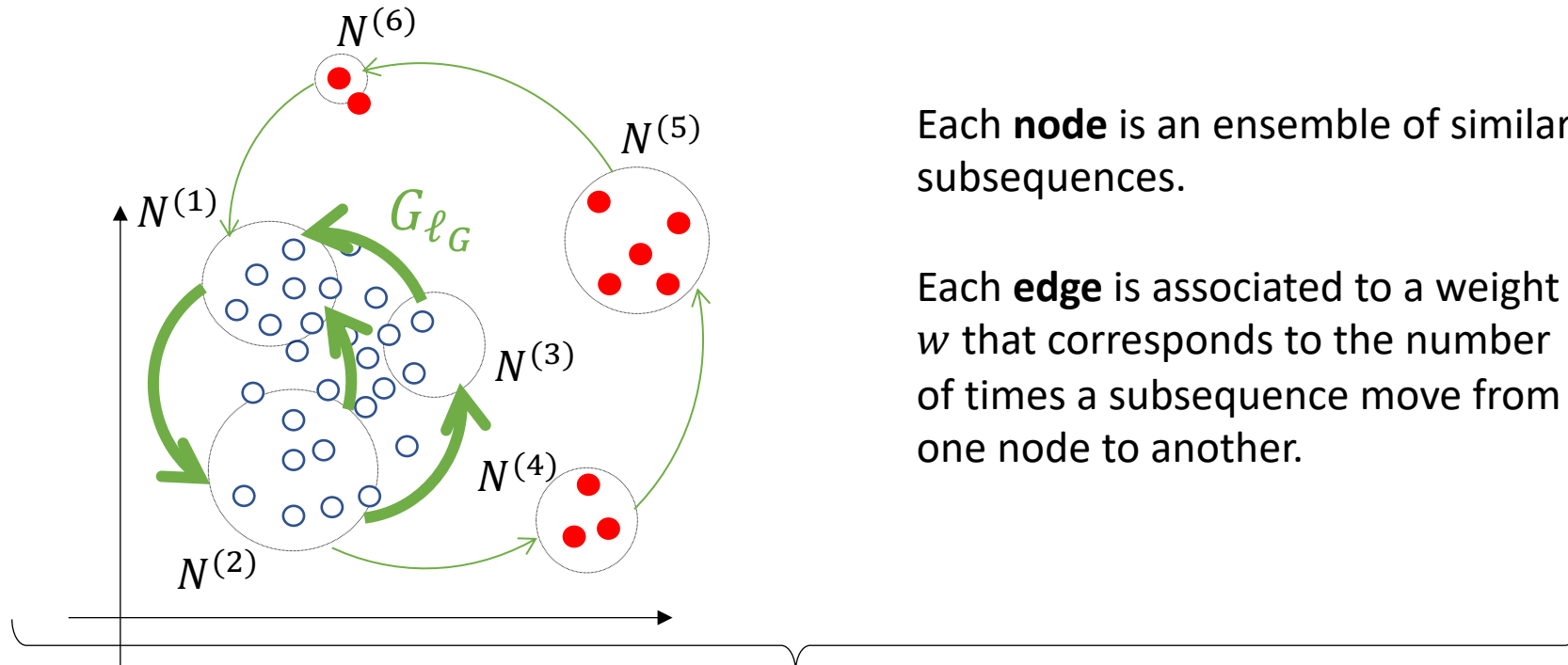
Density-based approach that **convert** the time series into a **graph** and detect **unusual trajectories**

Unsupervised

Univariate

subsequence

Anomaly Detection methods: *an Example*



For a given subsequence $T_{i,\ell}$ and its corresponding path $P_{th} = \langle N^{(i)}, N^{(i+1)}, \dots, N^{(i+\ell)} \rangle$, we define the normality score as follows:

$$Norm(P_{th}) = \sum_{j=i}^{i+\ell-1} \frac{w(N^{(j)}, N^{(j+1)}) \deg(N^{(j)} - 1)}{\ell}$$

Series2Graph [13]

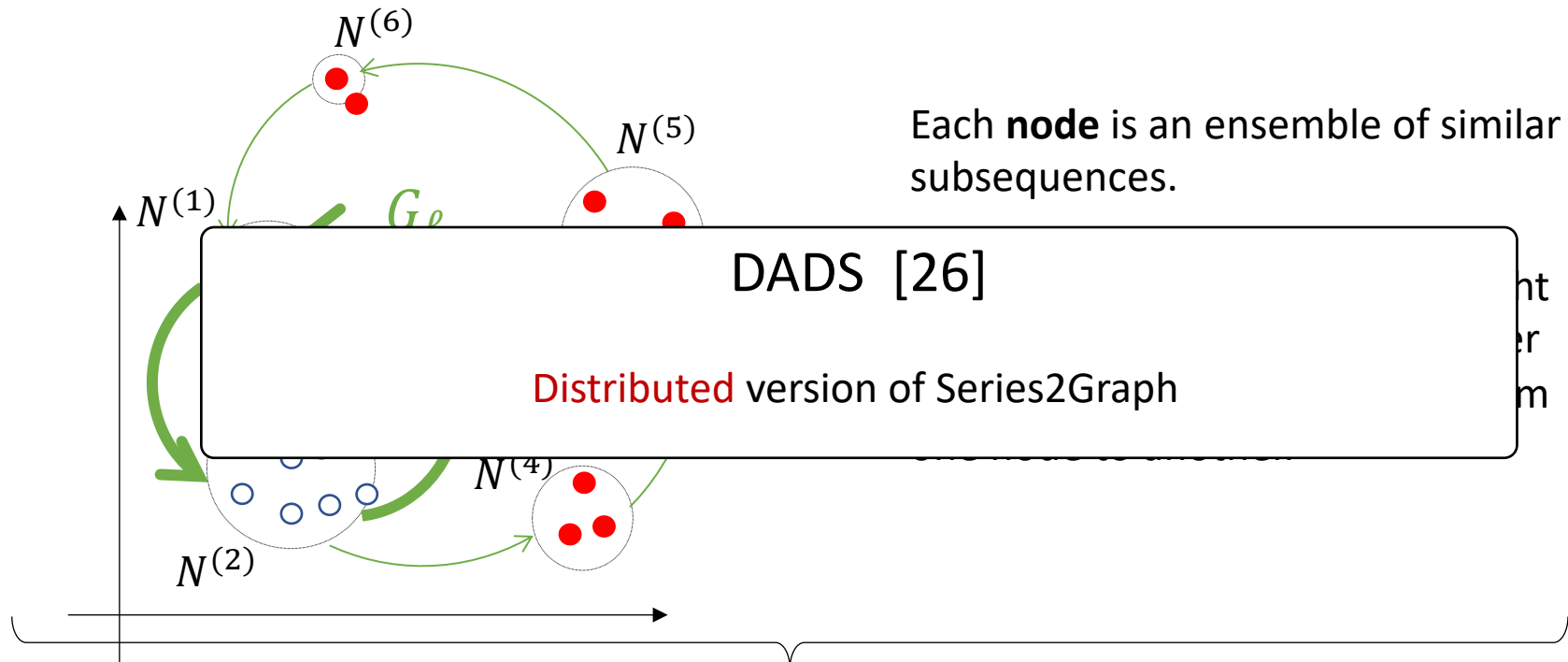
Density-based approach that **convert** the time series into a **graph** and detect **unusual trajectories**

Unsupervised

Univariate

subsequence

Anomaly Detection methods: *an Example*



For a given subsequence $T_{i,\ell}$ and its corresponding path $P_{th} = \langle N^{(i)}, N^{(i+1)}, \dots, N^{(i+\ell)} \rangle$, we define the normality score as follows:

$$Norm(P_{th}) = \sum_{j=i}^{i+\ell-1} \frac{w(N^{(j)}, N^{(j+1)}) \deg(N^{(j)} - 1)}{\ell}$$

Series2Graph [13]

Density-based approach that **convert** the time series into a **graph** and detect **unusual trajectories**

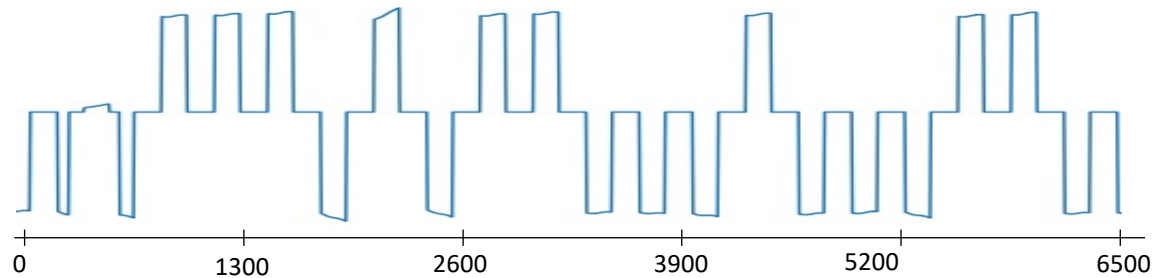
Unsupervised

Univariate

subsequence

Anomaly Detection methods: *an Example*

Snippet of SED time series



Series2Graph [13]

Density-based approach that **convert** the time series into a **graph** and detect **unusual trajectories**

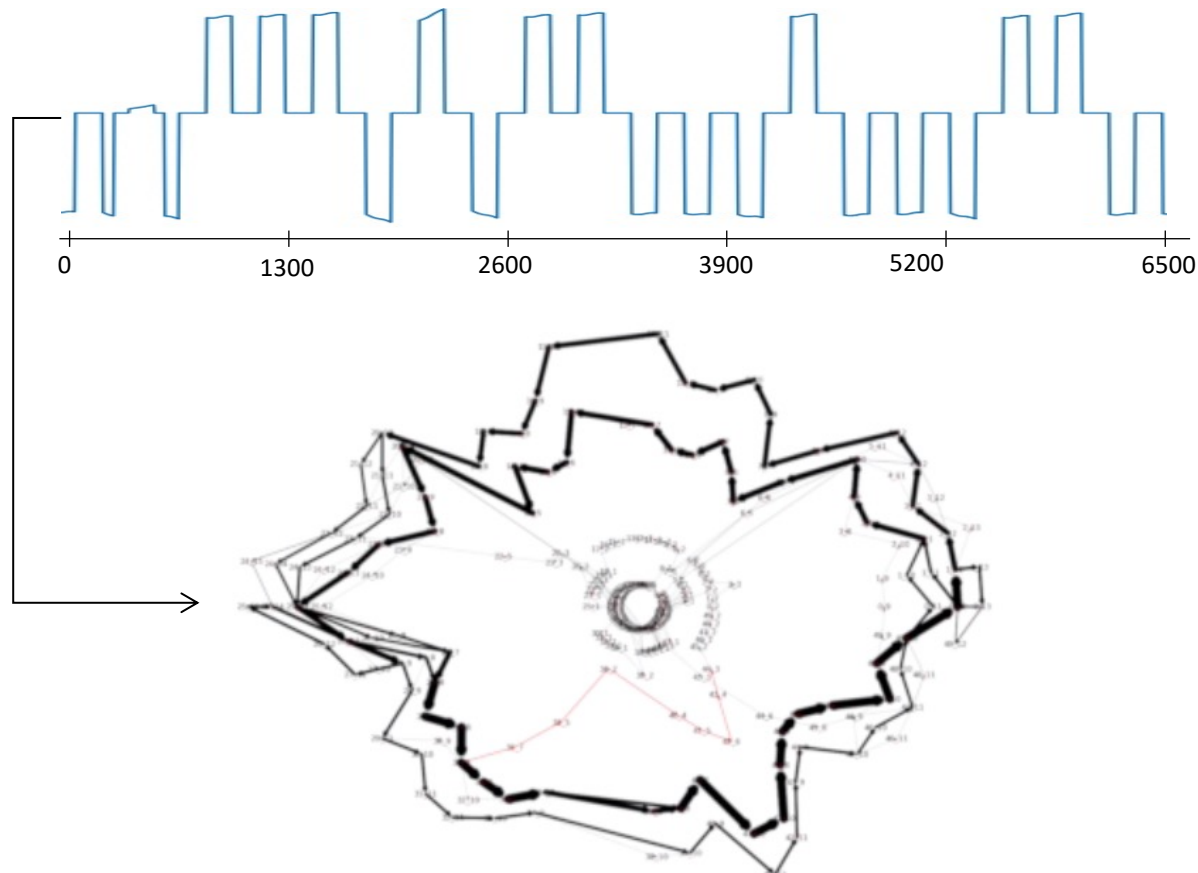
Unsupervised

Univariate

subsequence

Anomaly Detection methods: *an Example*

Snippet of SED time series



Series2Graph [13]

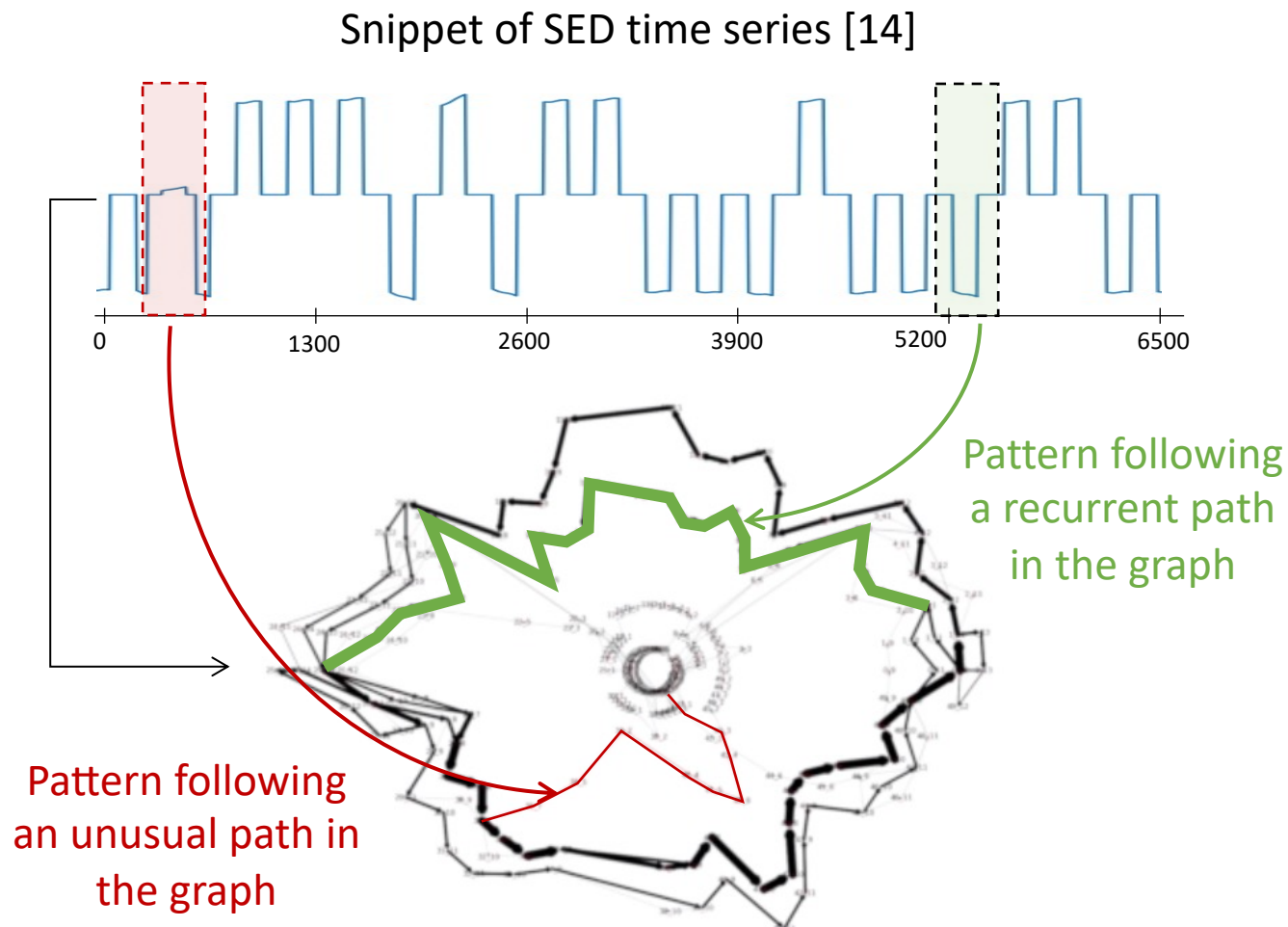
Density-based approach that **convert** the time series into a **graph** and detect **unusual trajectories**

Unsupervised

Univariate

subsequence

Anomaly Detection methods: *an Example*



Series2Graph [13]

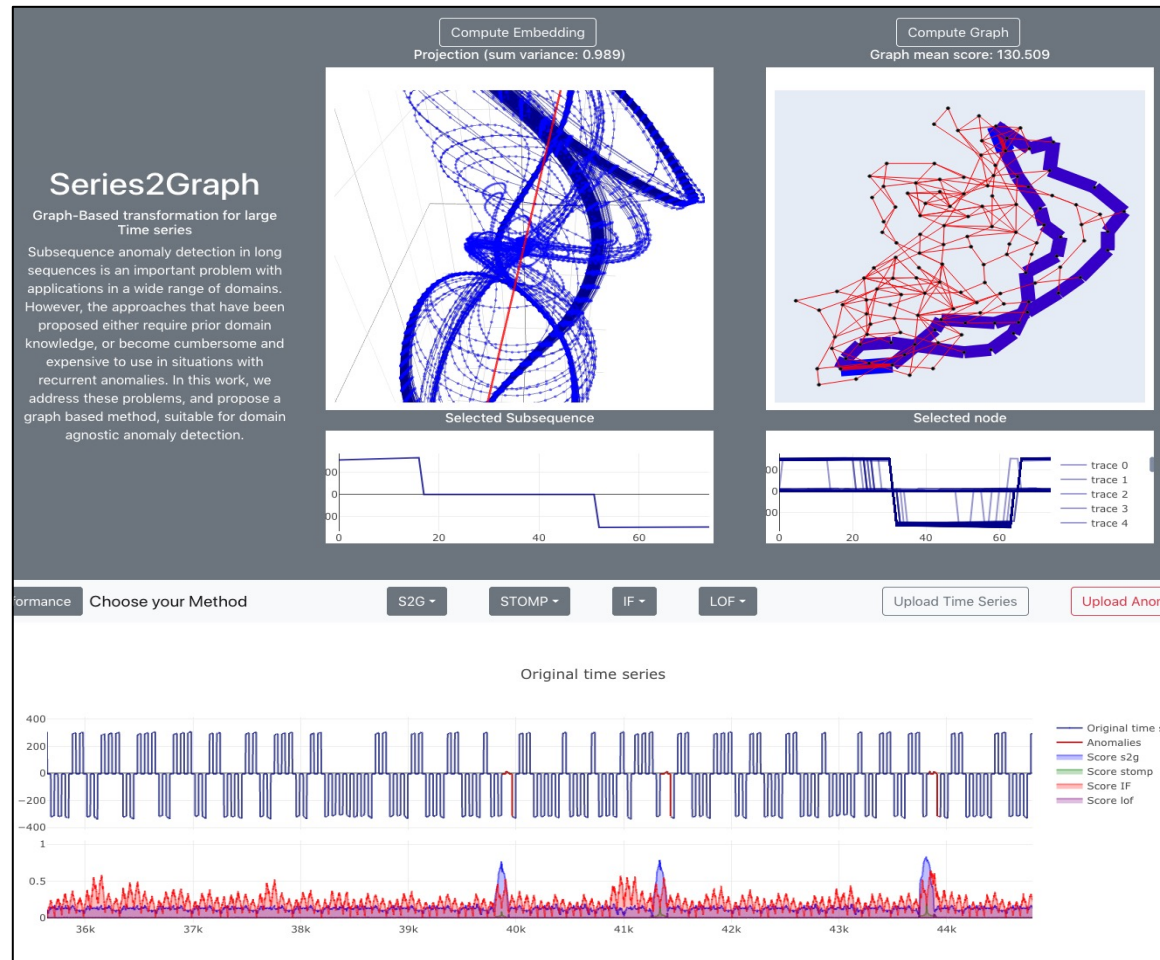
Density-based approach that **convert** the time series into a **graph** and detect **unusual trajectories**

Unsupervised

Univariate

subsequence

Anomaly Detection methods: *an Example*



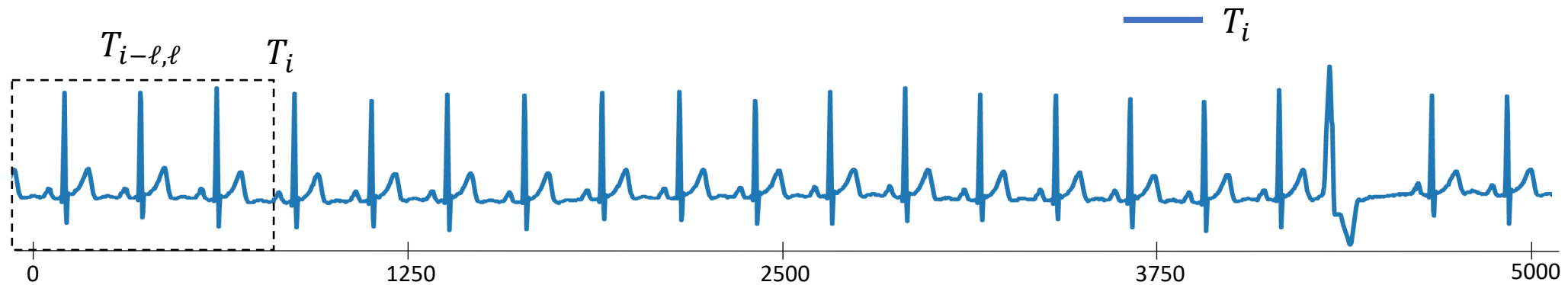
GraphAn [28]

An interactive tool to dive into
the computation steps of
Series2Graph :



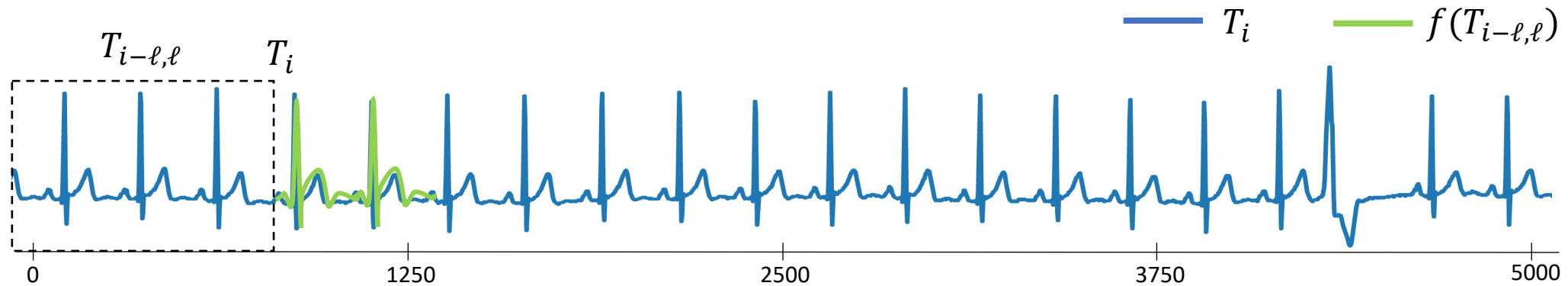
Anomaly Detection methods: *Forecasting-based*

Methods that aims to **predict the next points** based on the previous ones. The **prediction error** is used to detect if there is an anomaly or not.



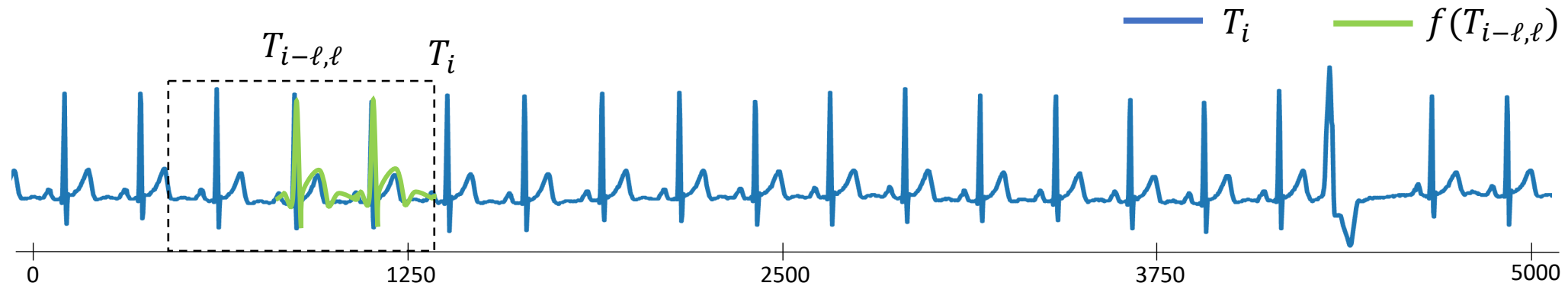
Anomaly Detection methods: *Forecasting-based*

Methods that aims to **predict the next points** based on the previous ones. The **prediction error** is used to detect if there is an anomaly or not.



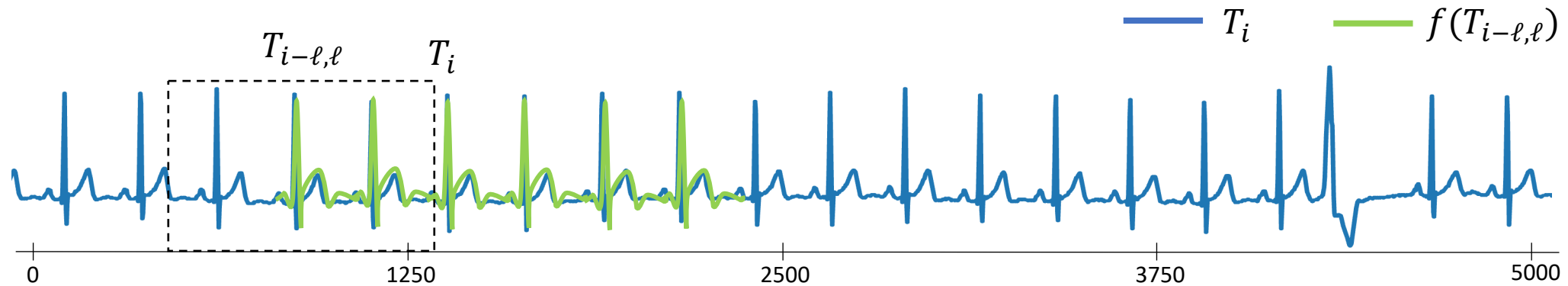
Anomaly Detection methods: *Forecasting-based*

Methods that aims to **predict the next points** based on the previous ones. The **prediction error** is used to detect if there is an anomaly or not.



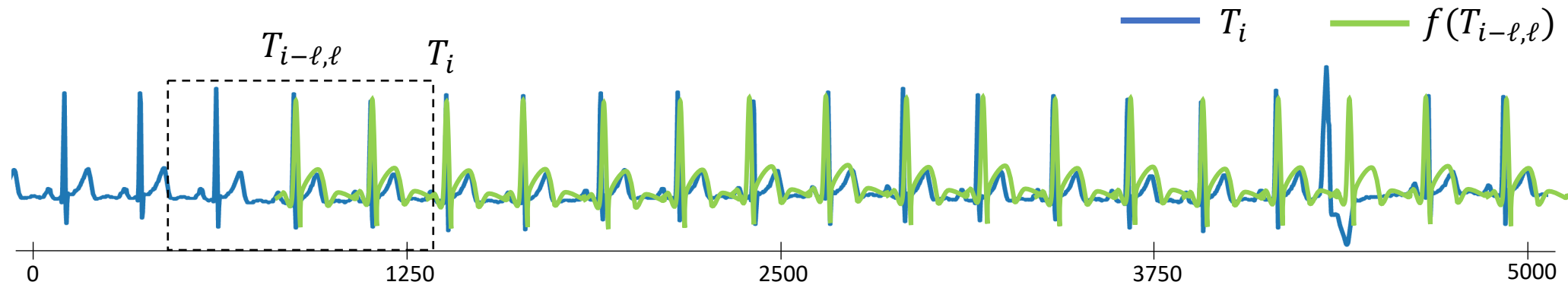
Anomaly Detection methods: *Forecasting-based*

Methods that aims to **predict the next points** based on the previous ones. The **prediction error** is used to detect if there is an anomaly or not.



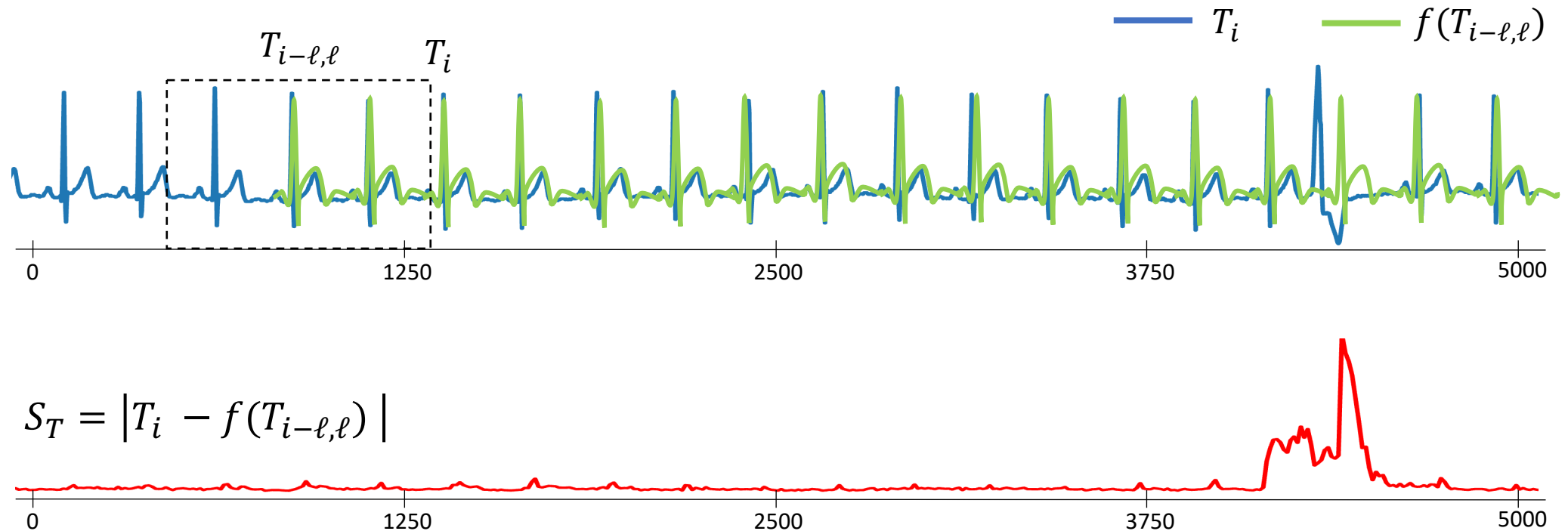
Anomaly Detection methods: *Forecasting-based*

Methods that aims to **predict the next points** based on the previous ones. The **prediction error** is used to detect if there is an anomaly or not.



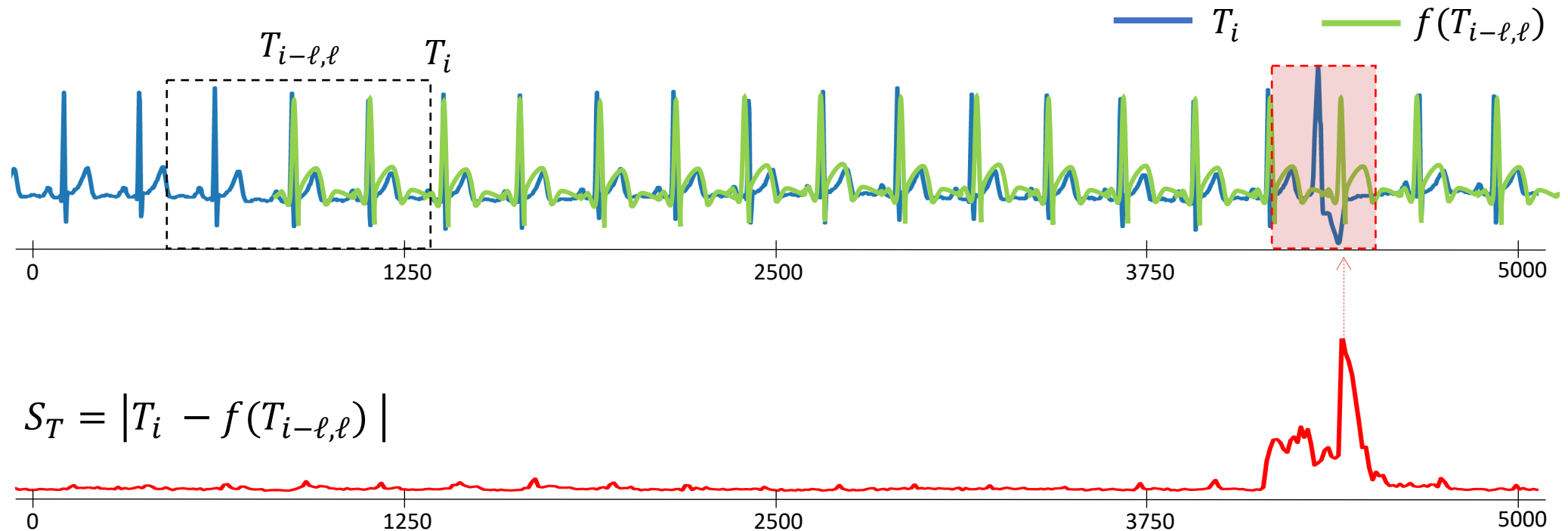
Anomaly Detection methods: *Forecasting-based*

Methods that aims to **predict the next points** based on the previous ones. The **prediction error** is used to detect if there is an anomaly or not.

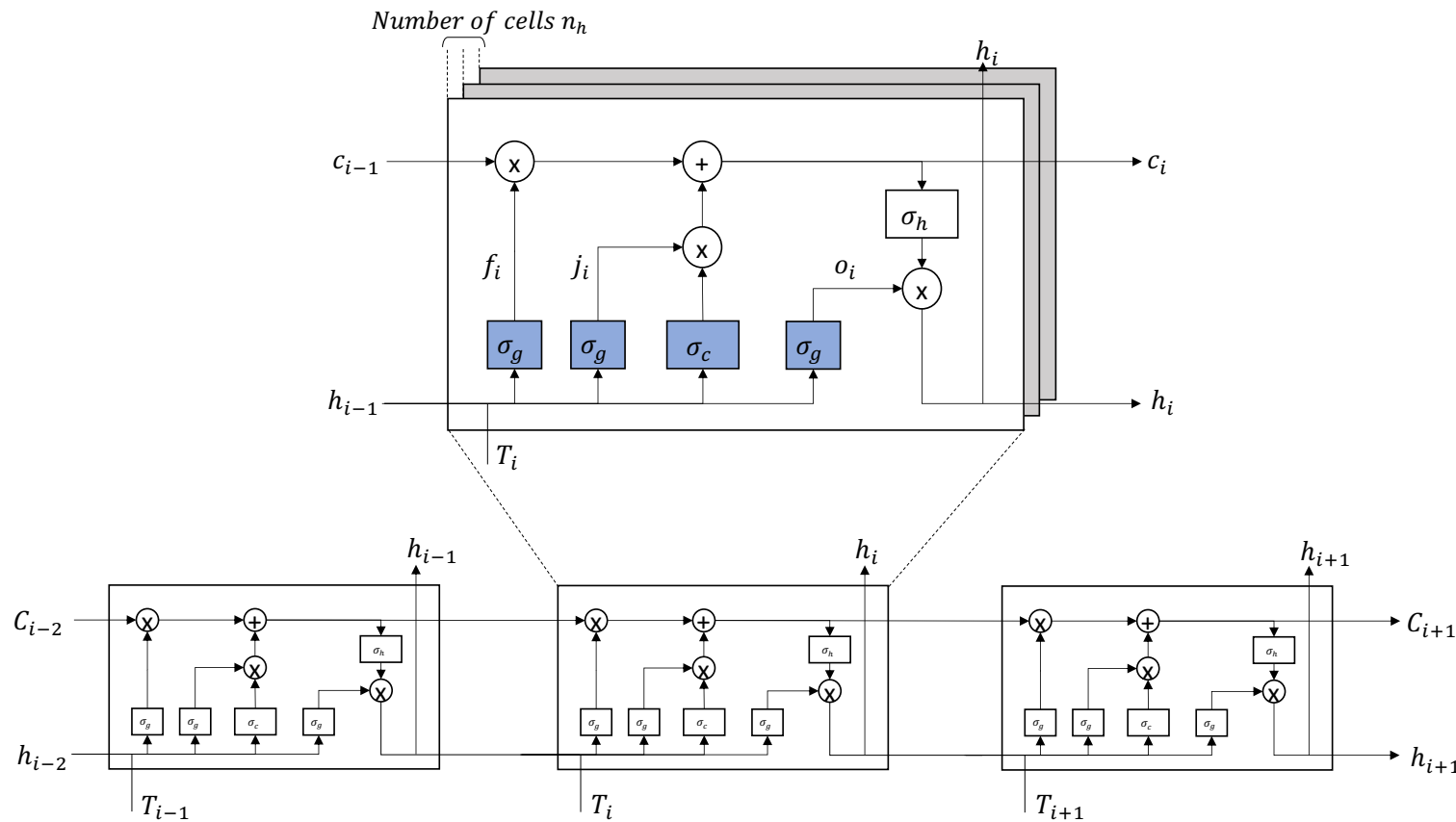


Anomaly Detection methods: *Forecasting-based*

Methods that aims to **predict the next points** based on the previous ones. The **prediction error** is used to detect if there is an anomaly or not.



Anomaly Detection methods: *an Example*



LSTM-AD [15]

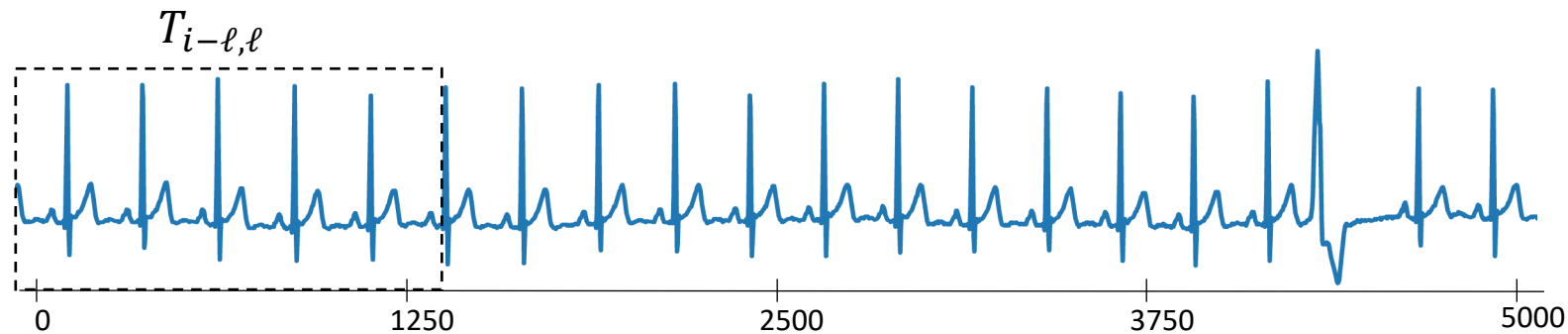
Model that stack multiple LSTM cell and use the output to predict the next value

Semi-supervised

Univariate/Multivariate

Point/sequence

Anomaly Detection methods: *an Example*



DeepAnT [16] (CNN)

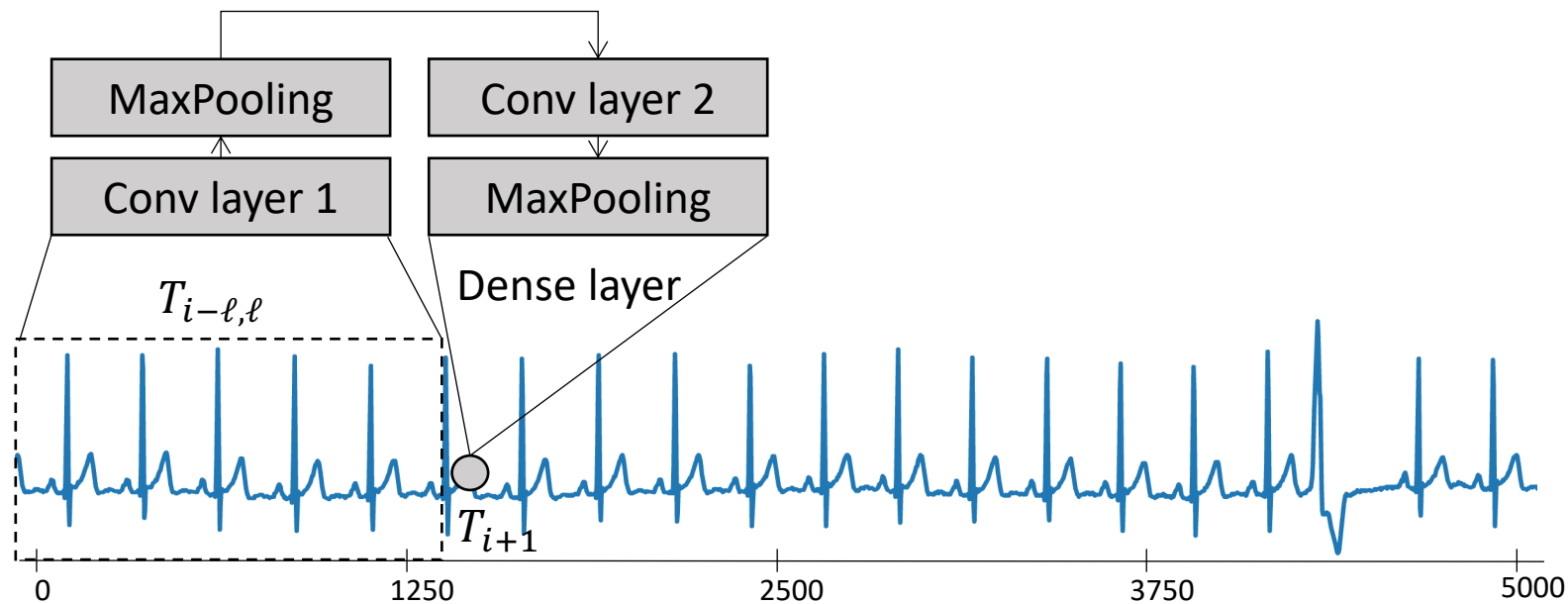
Convolutional-based approach (2 convolutional layers) taking as input a sequence and aims to predict the next value.

Semi-supervised

Univariate/Multivariate

Point/sequence

Anomaly Detection methods: *an Example*



DeepAnT [16] (CNN)

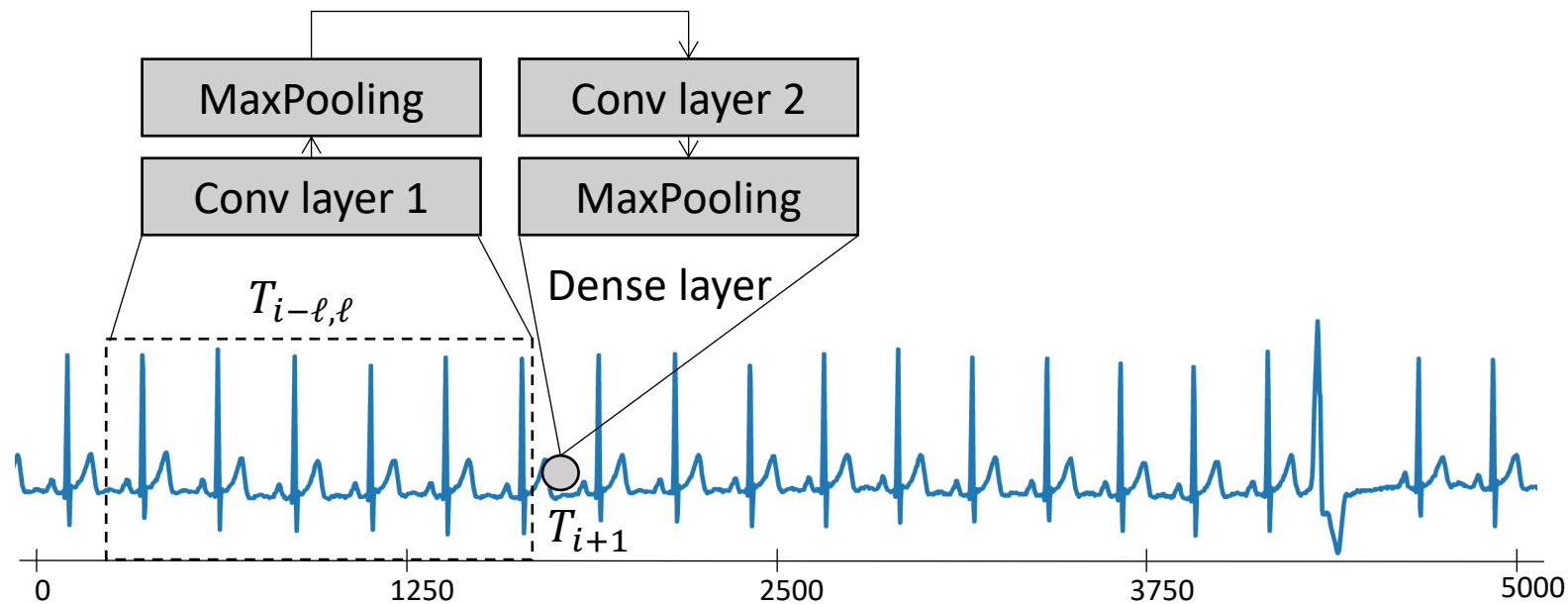
Convolutional-based approach (2 convolutional layers) taking as input a sequence and aims to predict the next value.

Semi-supervised

Univariate/Multivariate

Point/sequence

Anomaly Detection methods: *an Example*



DeepAnT [16] (CNN)

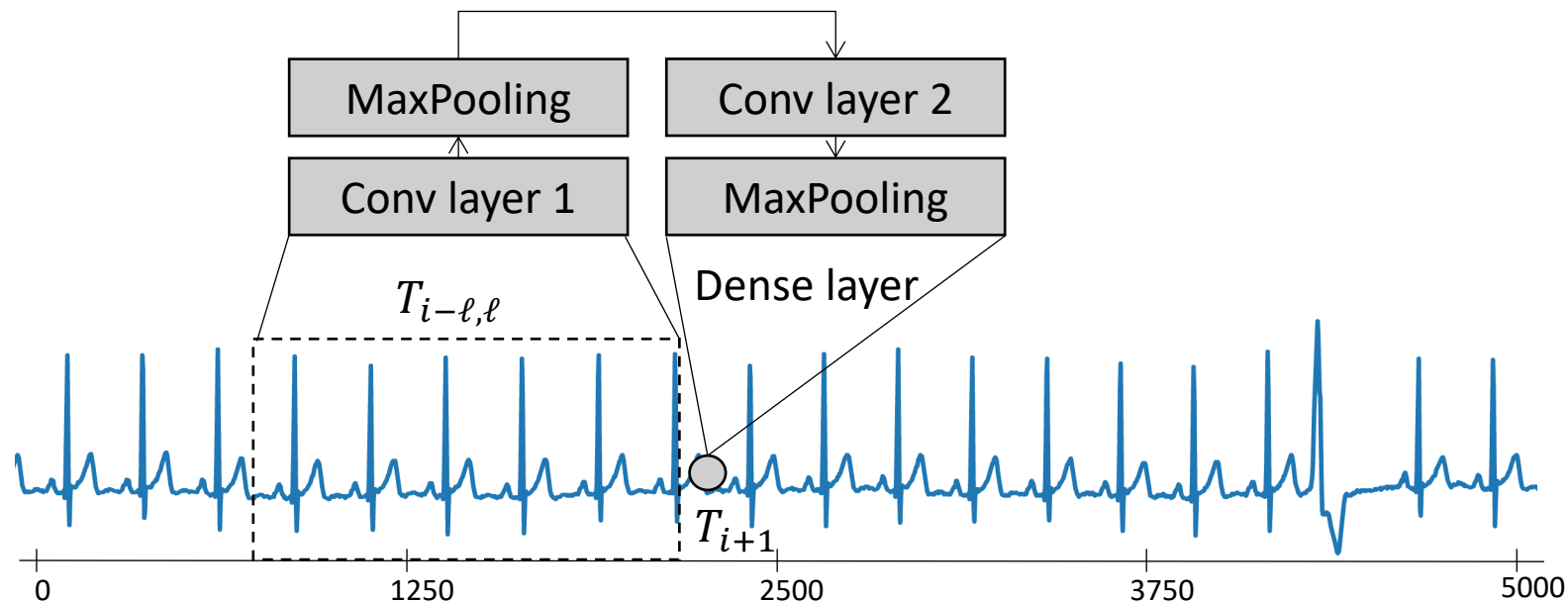
Convolutional-based approach (2 convolutional layers) taking as input a sequence and aims to predict the next value.

Semi-supervised

Univariate/Multivariate

Point/sequence

Anomaly Detection methods: *an Example*



DeepAnT [16] (CNN)

Convolutional-based approach (2 convolutional layers) taking as input a sequence and aims to predict the next value.

Semi-supervised

Univariate/Multivariate

Point/sequence

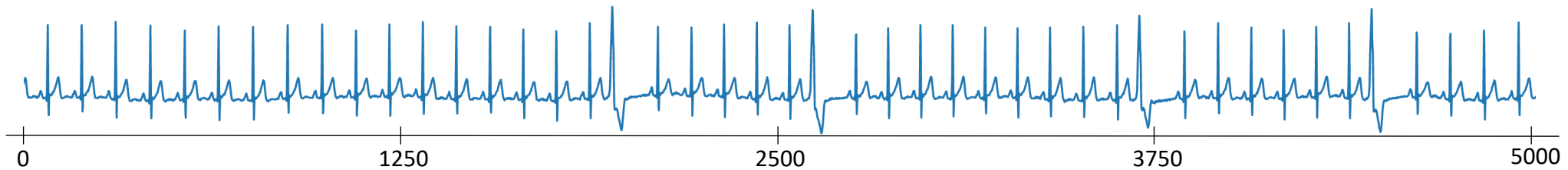
Anomaly Detection methods: *Reconstruction-based*

Methods that aims to **reconstruct** the time series T and use the **reconstruction error** to detect if the time series is an anomaly or not.

Anomaly Detection methods: *Reconstruction-based*

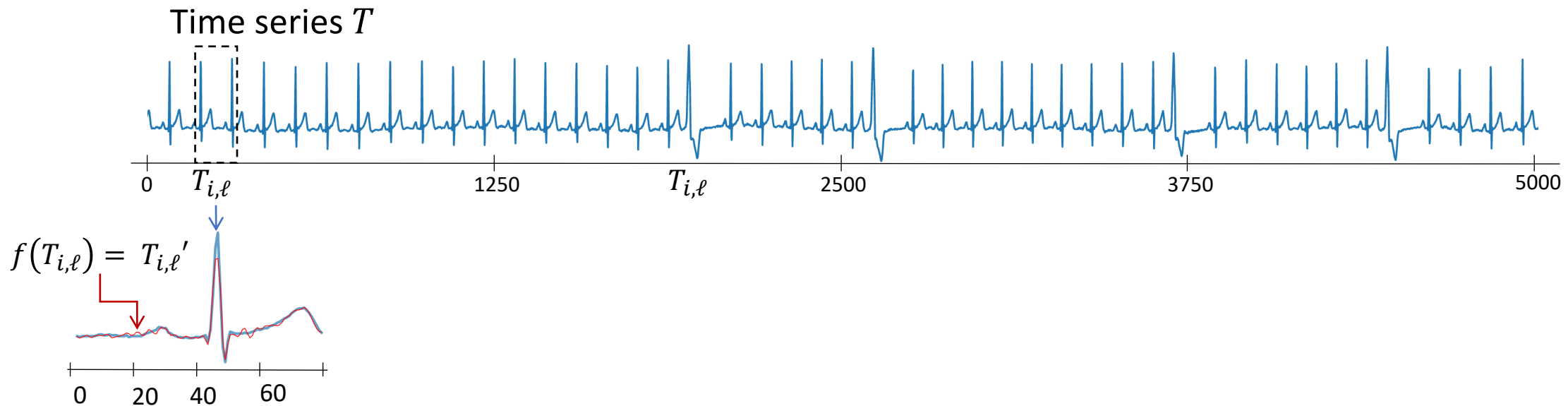
Methods that aims to **reconstruct** the time series T and use the **reconstruction error** to detect if the time series is an anomaly or not.

Time series T



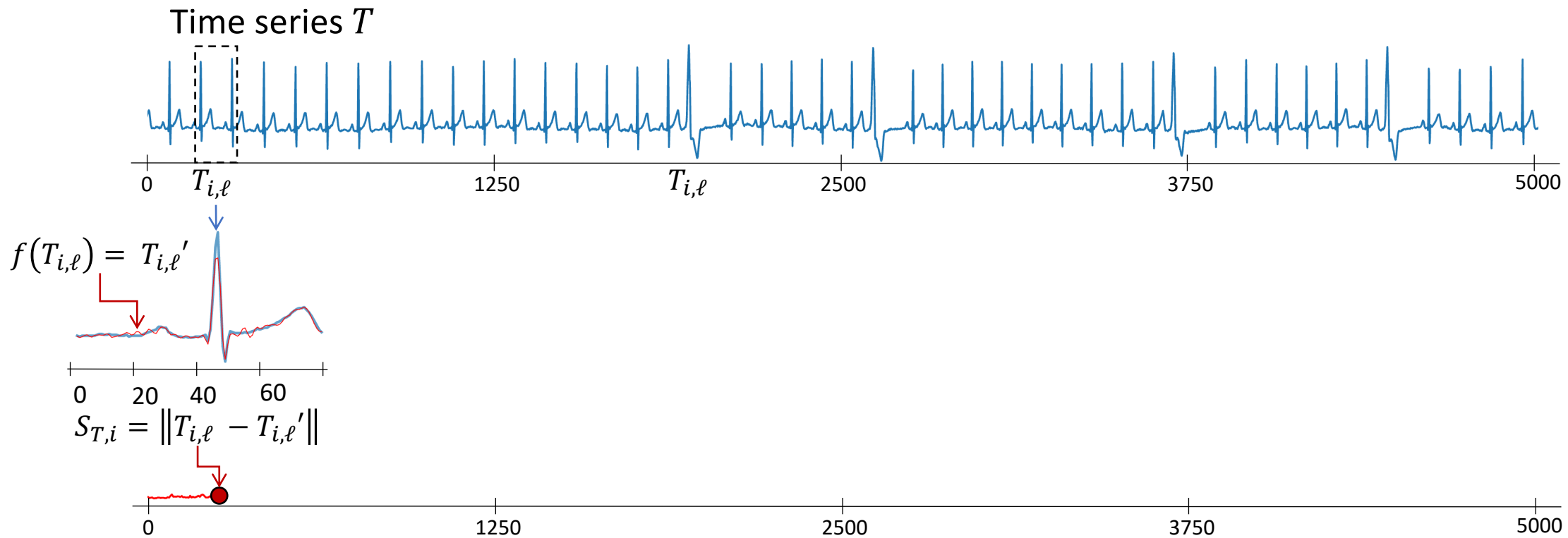
Anomaly Detection methods: *Reconstruction-based*

Methods that aims to **reconstruct** the time series T and use the **reconstruction error** to detect if the time series is an anomaly or not.



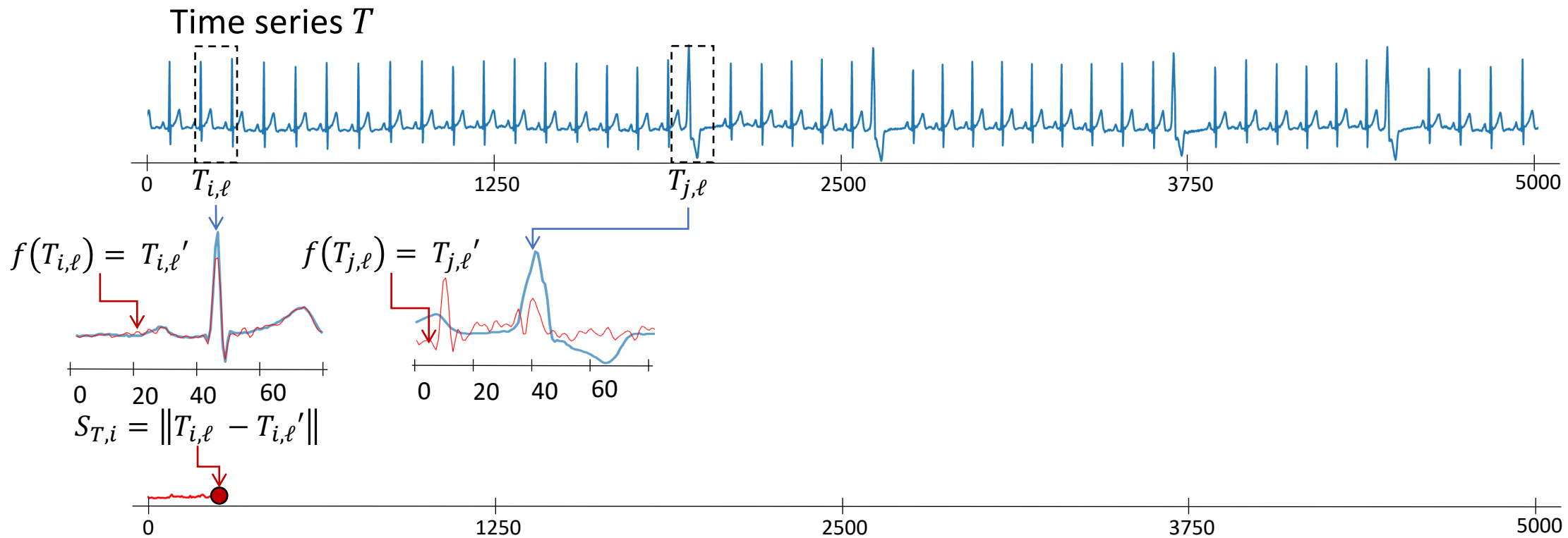
Anomaly Detection methods: *Reconstruction-based*

Methods that aims to **reconstruct** the time series T and use the **reconstruction error** to detect if the time series is an anomaly or not.



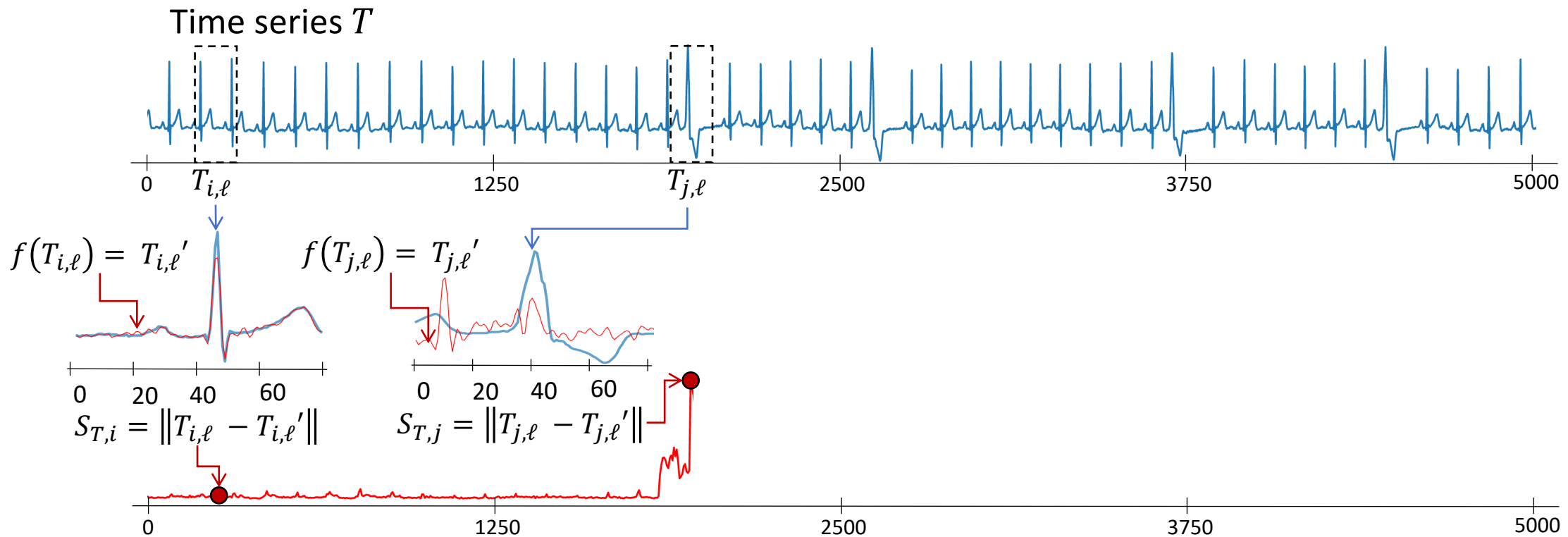
Anomaly Detection methods: *Reconstruction-based*

Methods that aims to **reconstruct** the time series T and use the **reconstruction error** to detect if the time series is an anomaly or not.



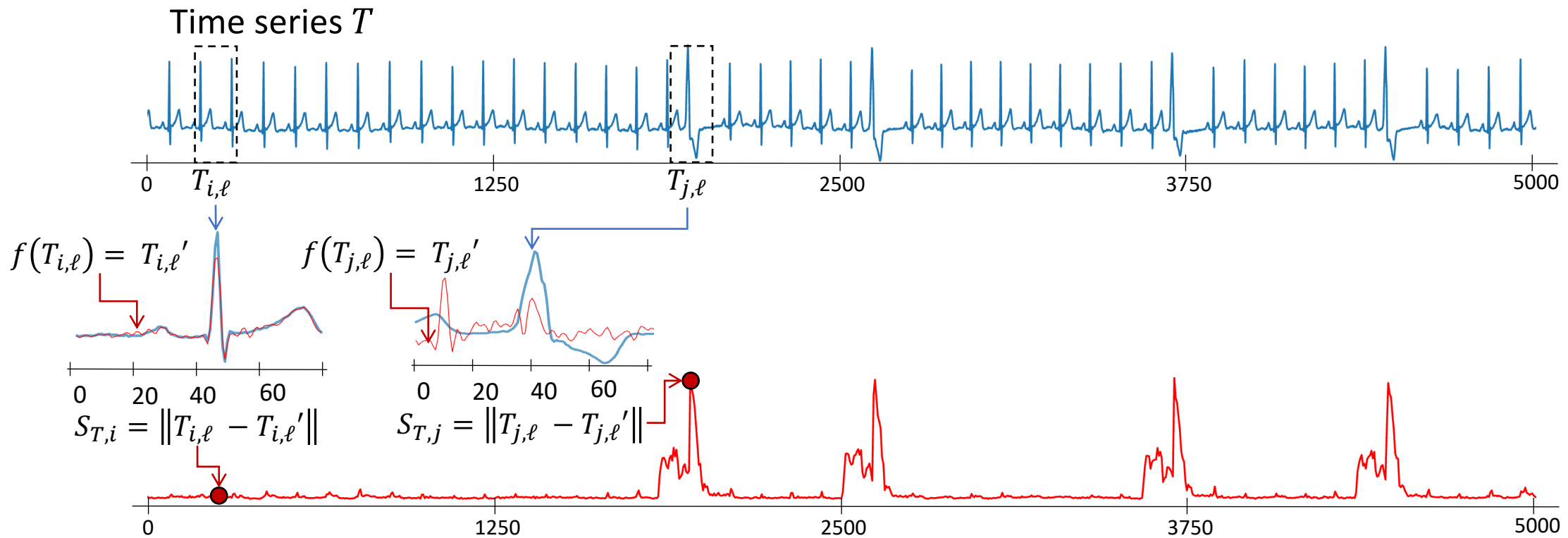
Anomaly Detection methods: *Reconstruction-based*

Methods that aims to **reconstruct** the time series T and use the **reconstruction error** to detect if the time series is an anomaly or not.



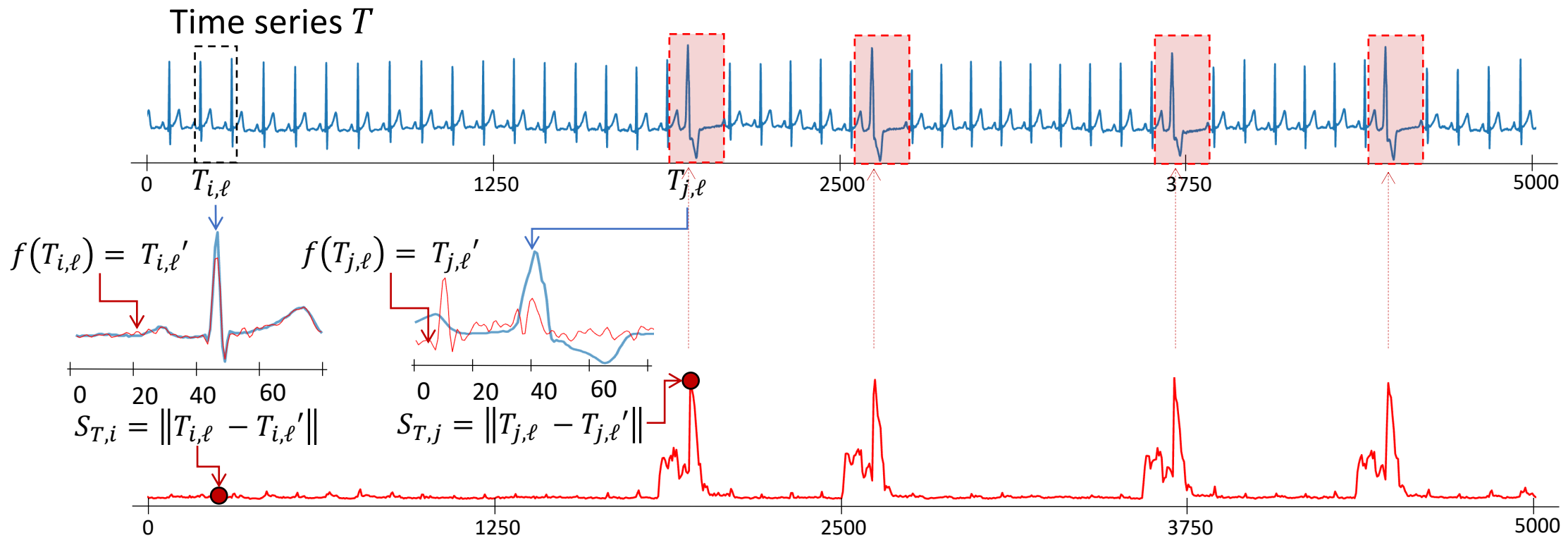
Anomaly Detection methods: *Reconstruction-based*

Methods that aims to **reconstruct** the time series T and use the **reconstruction error** to detect if the time series is an anomaly or not.

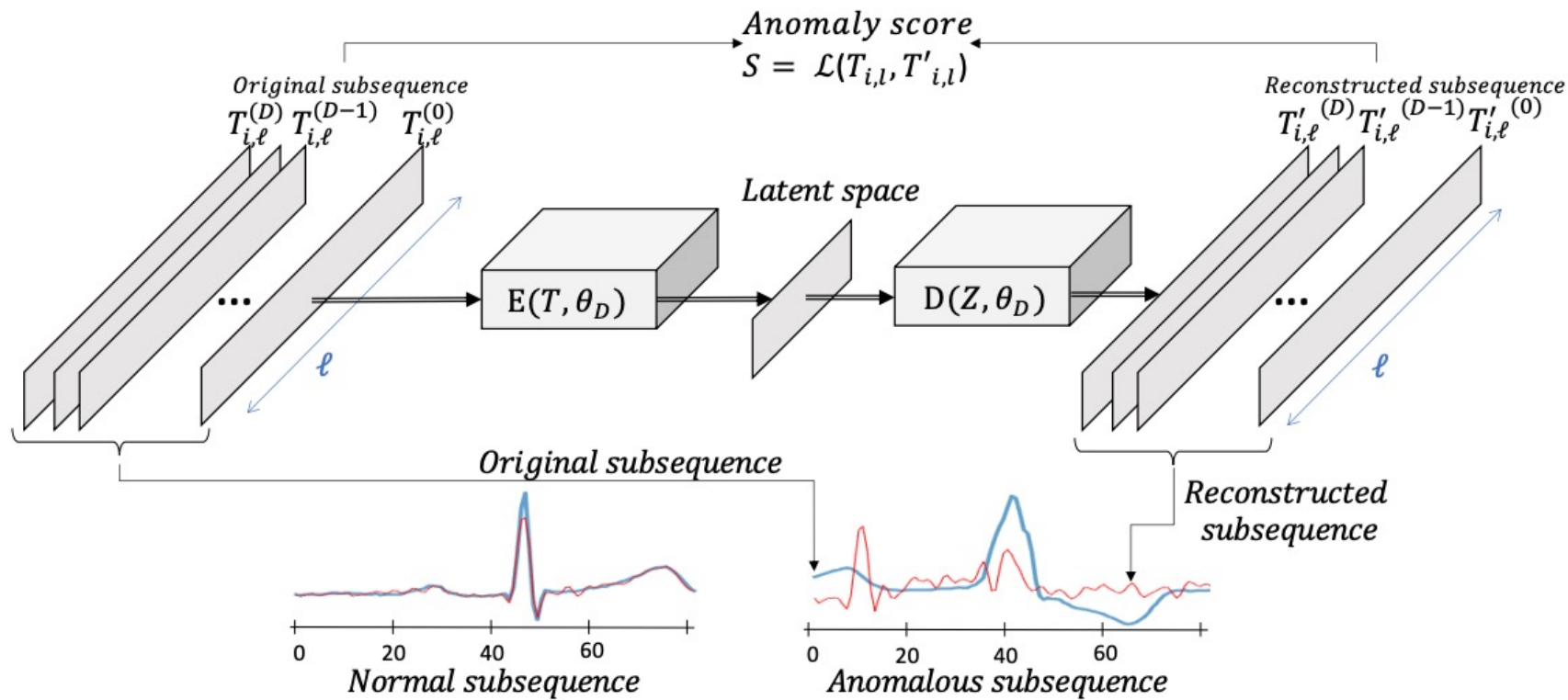


Anomaly Detection methods: *Reconstruction-based*

Methods that aims to **reconstruct** the time series T and use the **reconstruction error** to detect if the time series is an anomaly or not.



Anomaly Detection methods: *an Example*



AutoEncoders [17] (AE)

Neural Network composed of an **encoder** (that reduce the dimensionality) and **decoder** that **reconstruct** the time series. The objective is to **minimize the reconstruction error**.

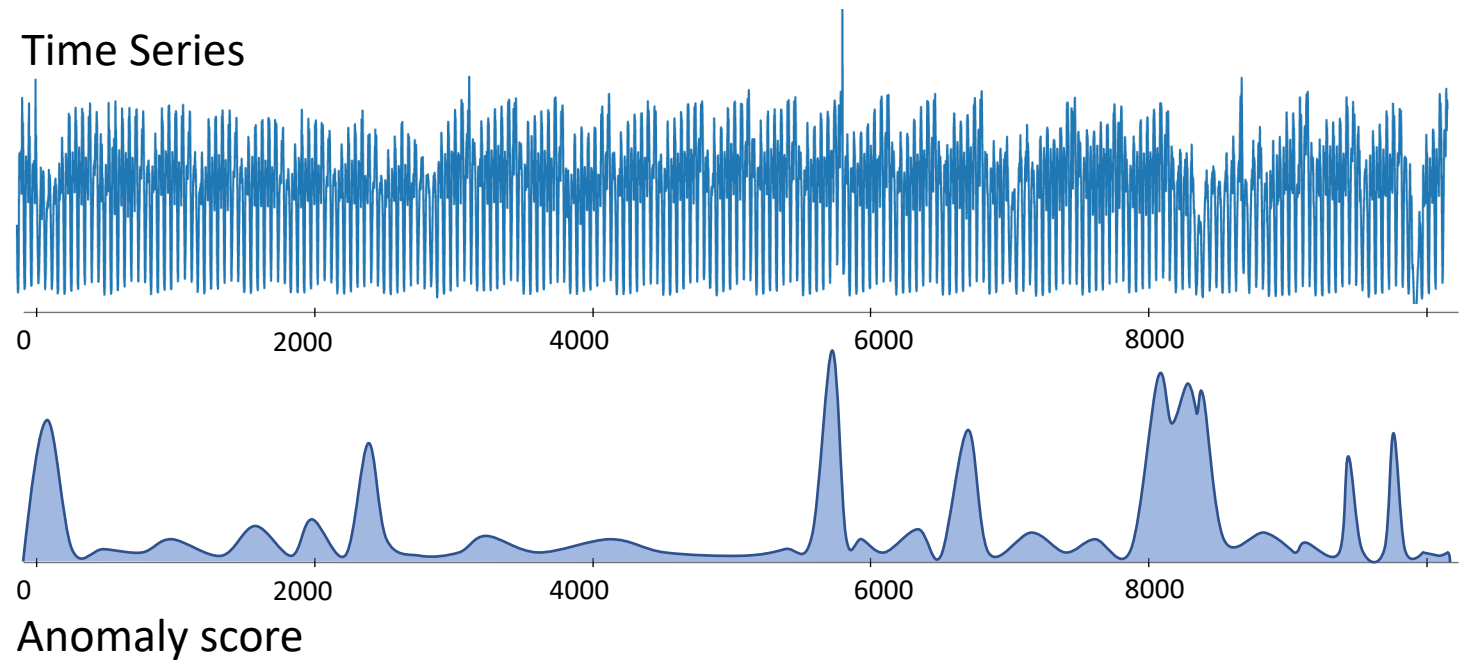
Semi-supervised

Univariate/Multivariate

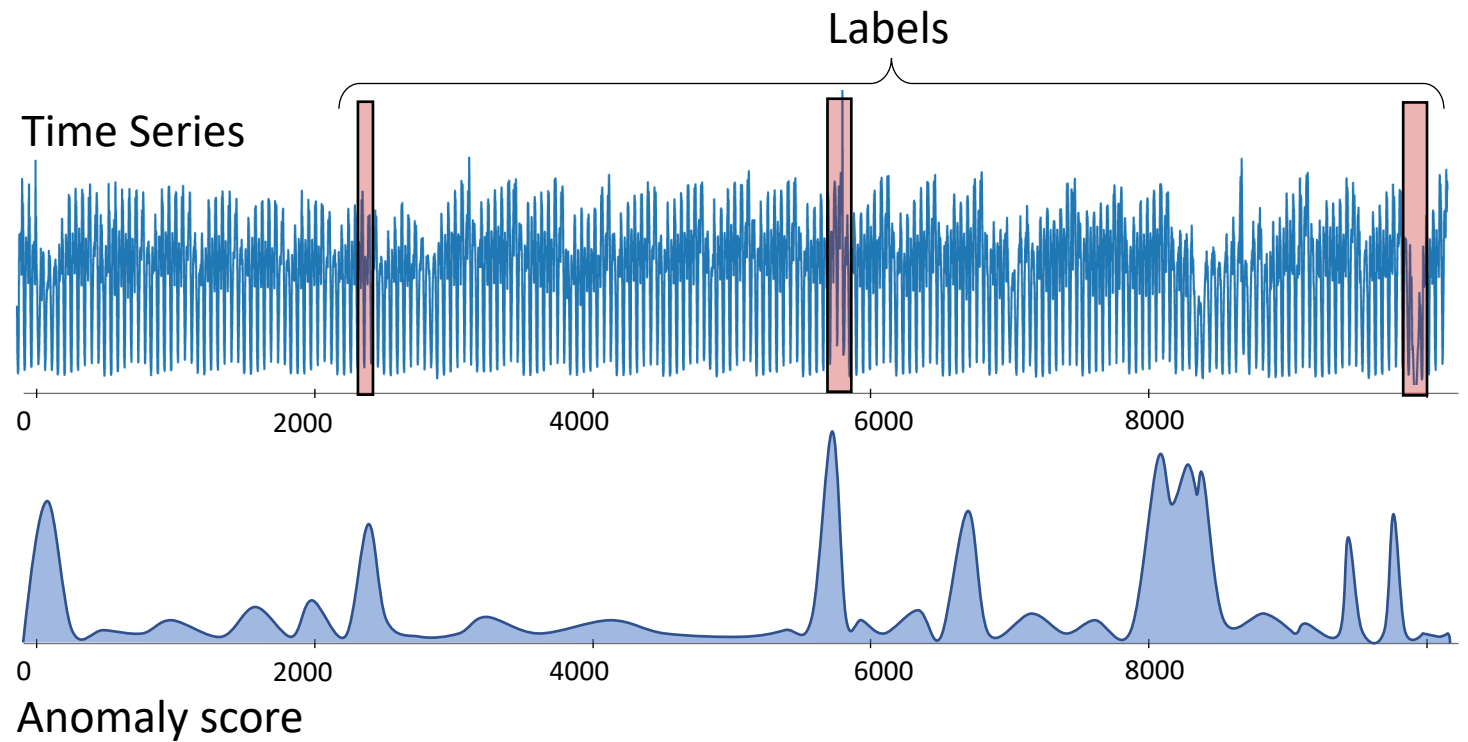
Point/sequence

Part 3: Evaluation Measures

Evaluation measures: *A general overview*

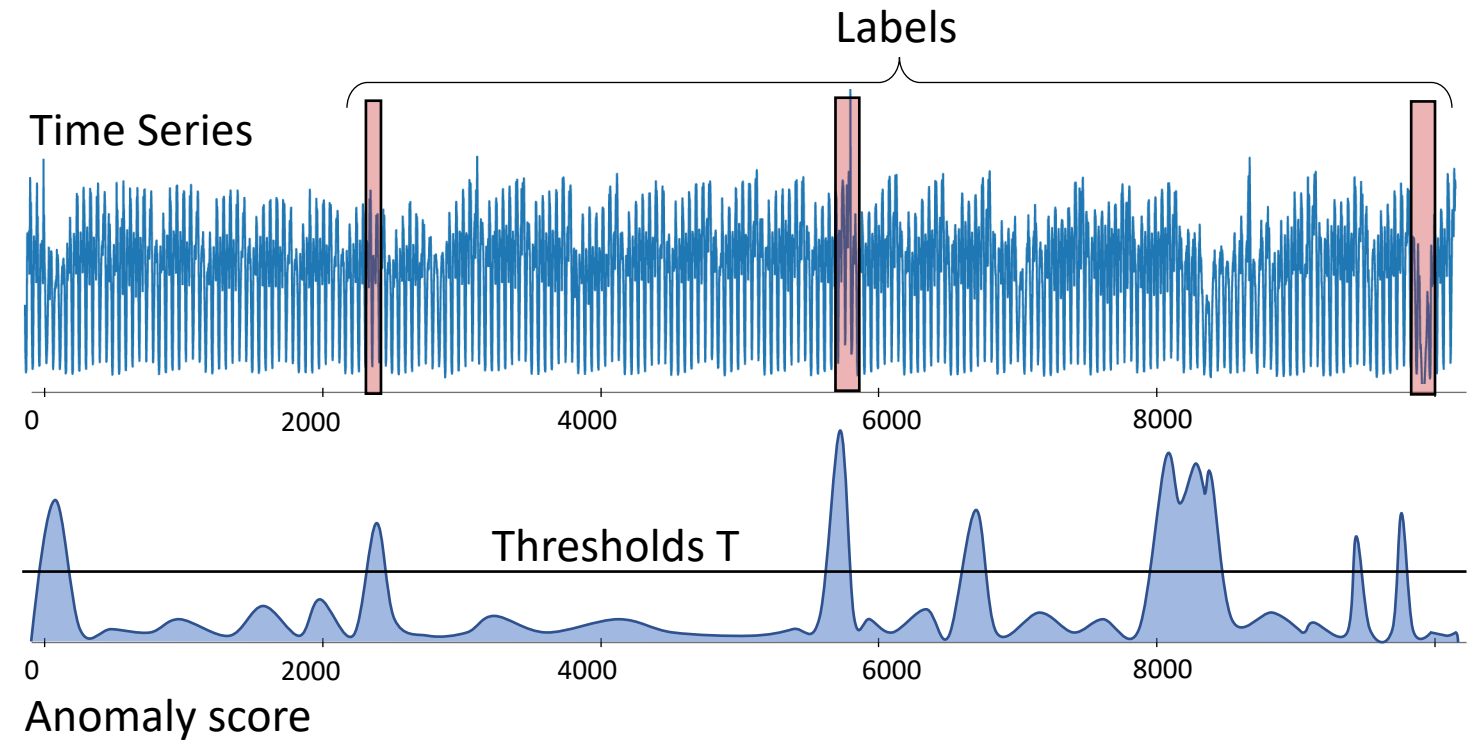


Evaluation measures: *A general overview*



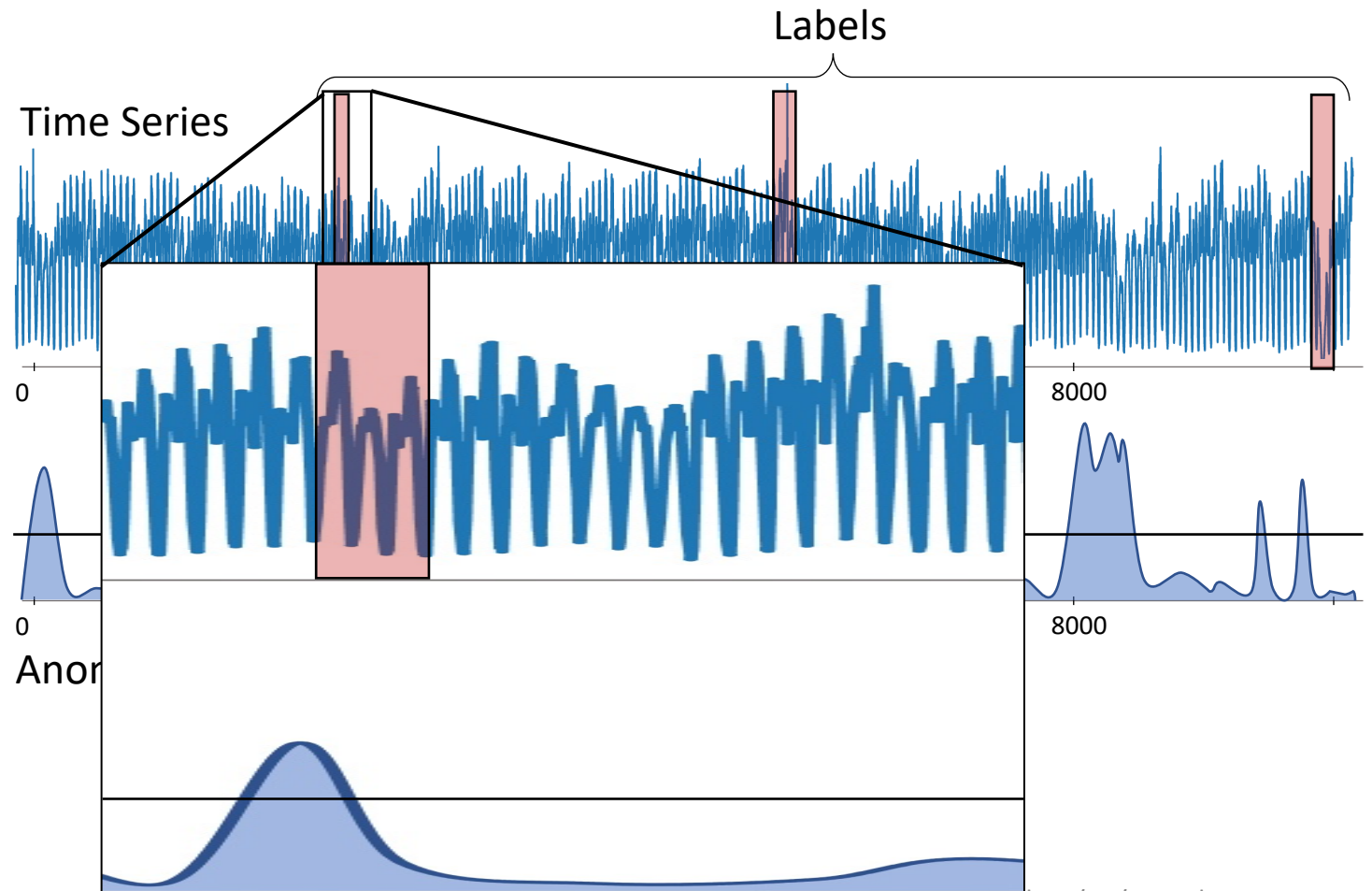
Evaluation measures: *Threshold-based*

Threshold-based Evaluation Measures:



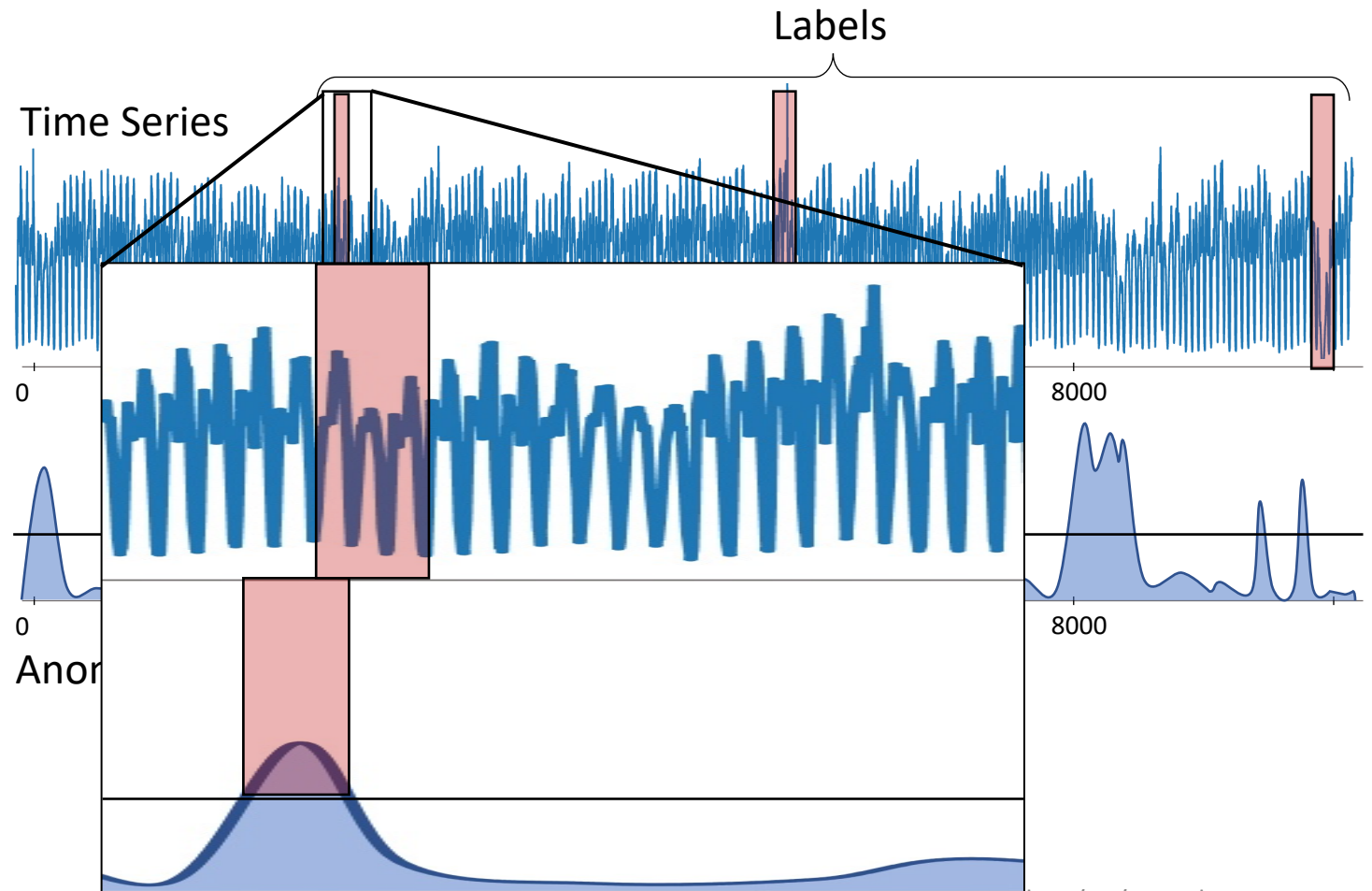
Evaluation measures: *Threshold-based*

Threshold-based Evaluation Measures:



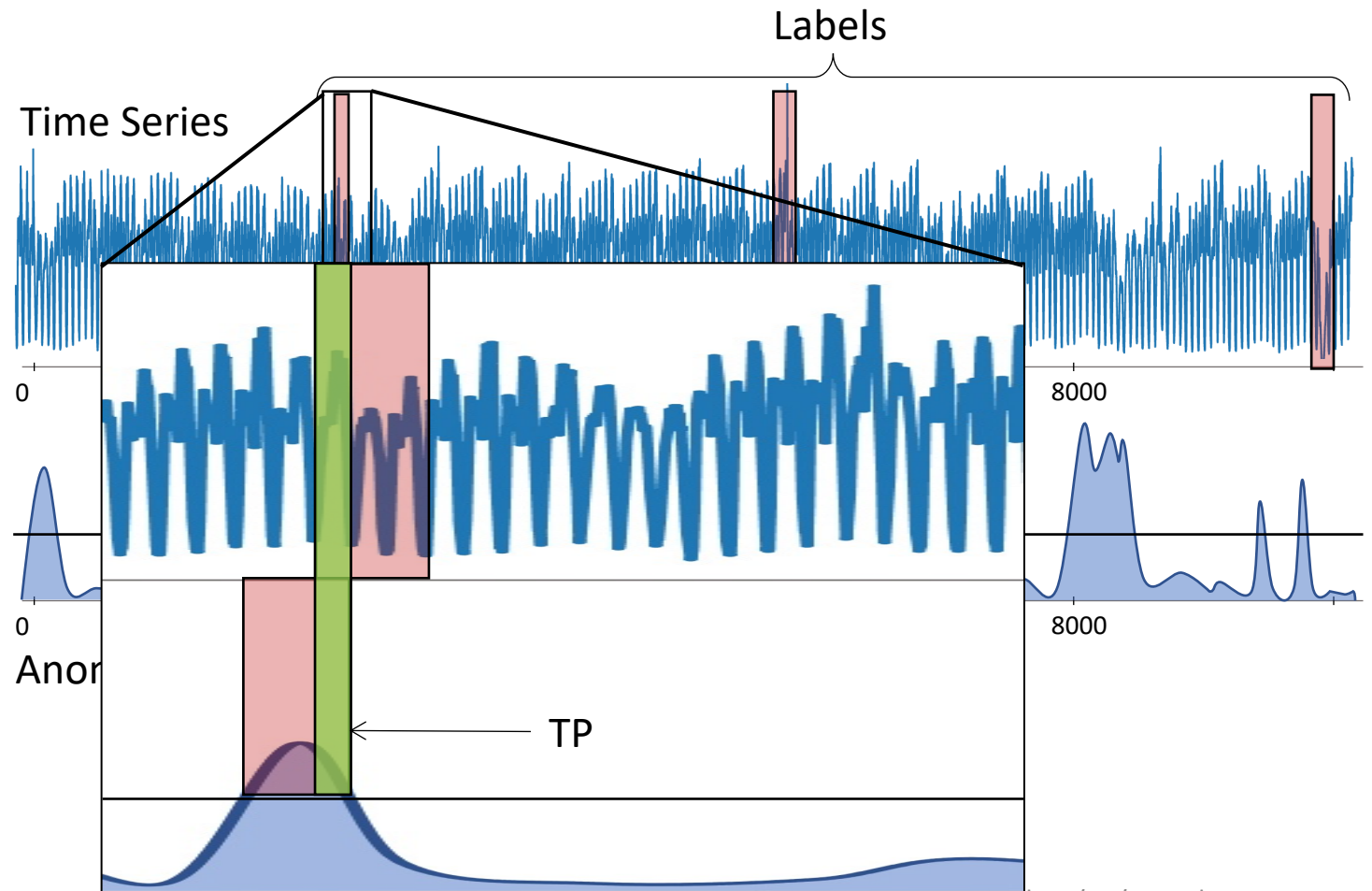
Evaluation measures: *Threshold-based*

Threshold-based Evaluation Measures:



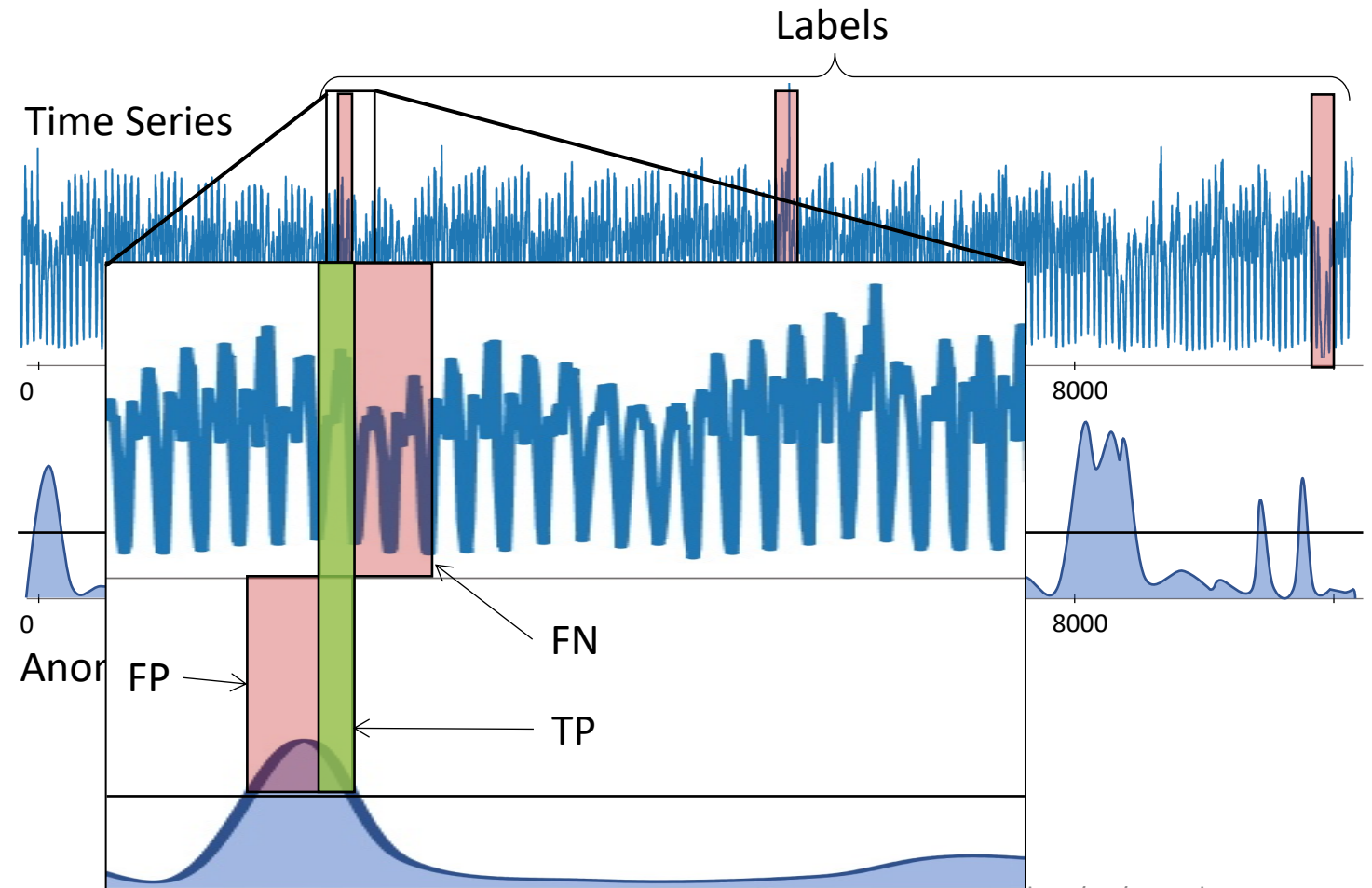
Evaluation measures: *Threshold-based*

Threshold-based Evaluation Measures:



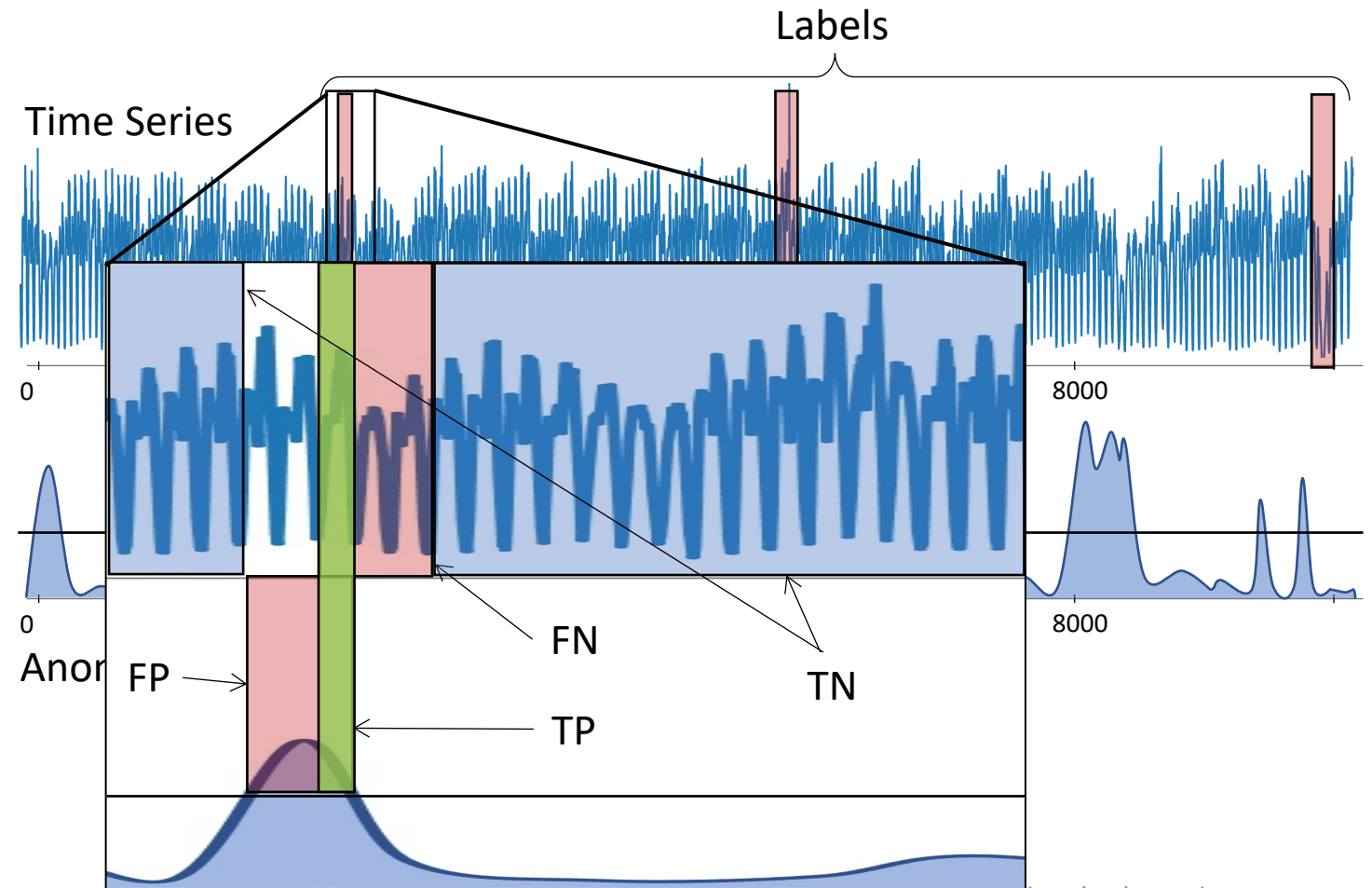
Evaluation measures: *Threshold-based*

Threshold-based Evaluation Measures:



Evaluation measures: *Threshold-based*

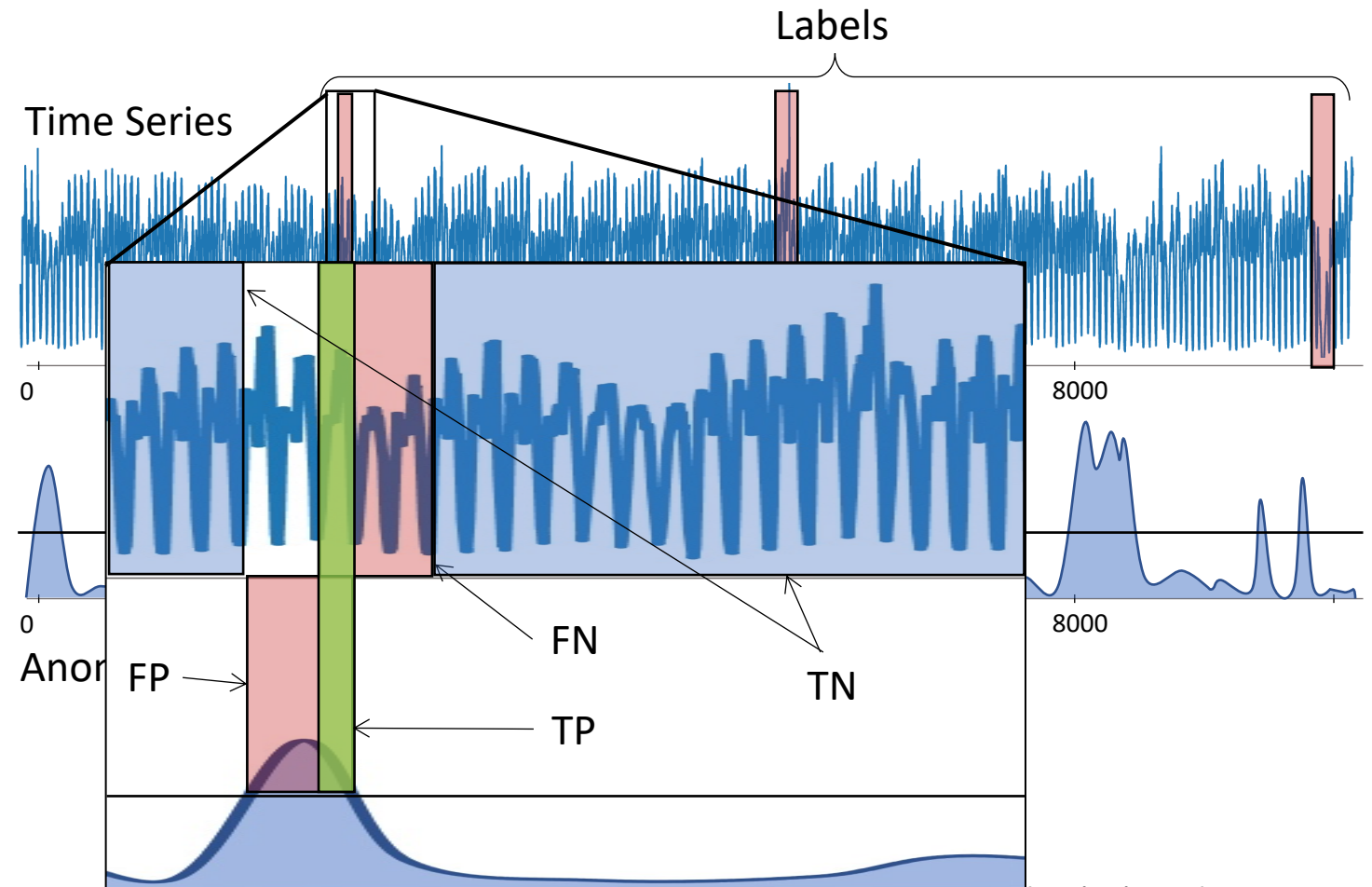
Threshold-based Evaluation Measures:



Evaluation measures: *Threshold-based*

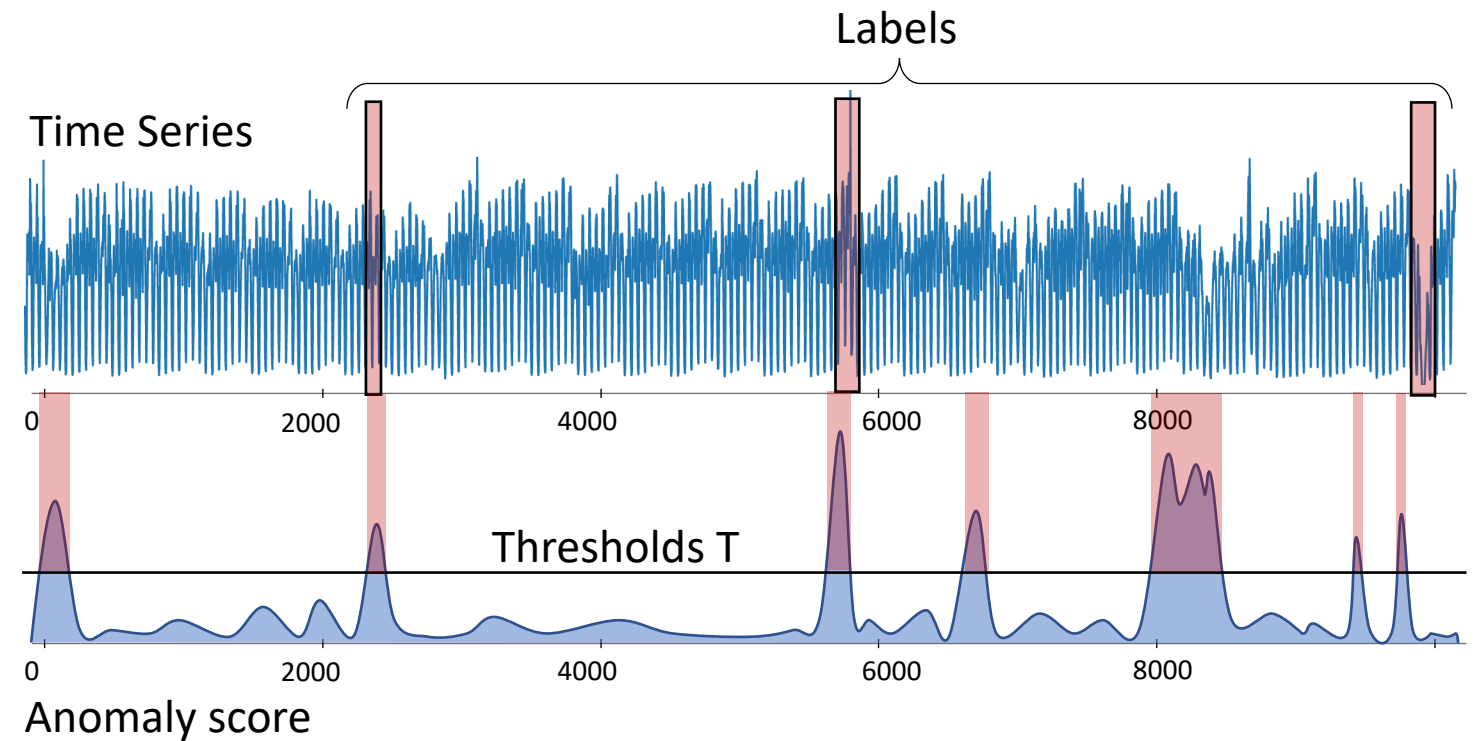
Threshold-based Evaluation Measures:

- Precision: $\frac{TP}{TP+FP}$
- Recall (true positive rate): $\frac{TP}{TP+FN}$
- False positive rate: $\frac{FP}{FP+TN}$
- F-score: $\frac{(1+\beta^2)*Precision}{\beta^2*Precision+Recall}$



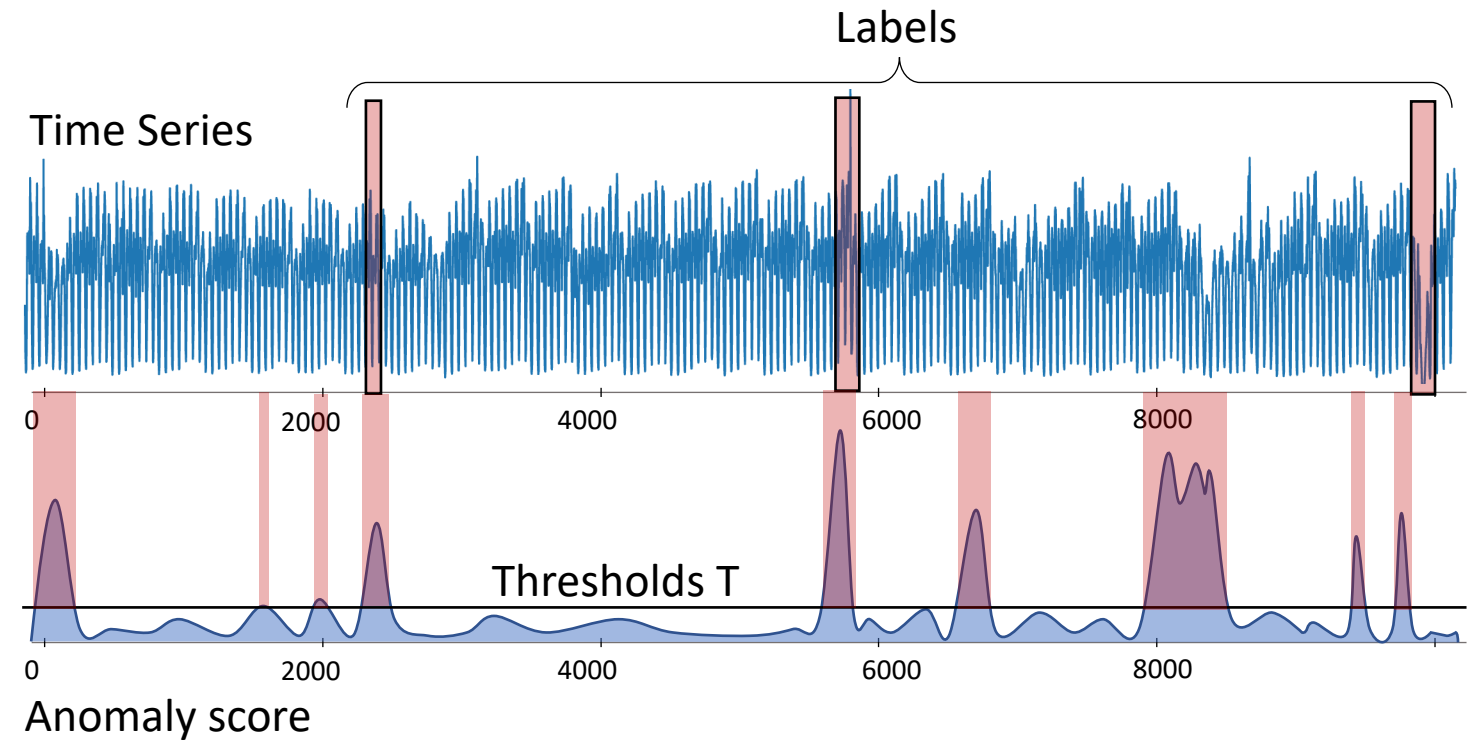
Evaluation measures: *AUC-based*

How do we set the threshold?



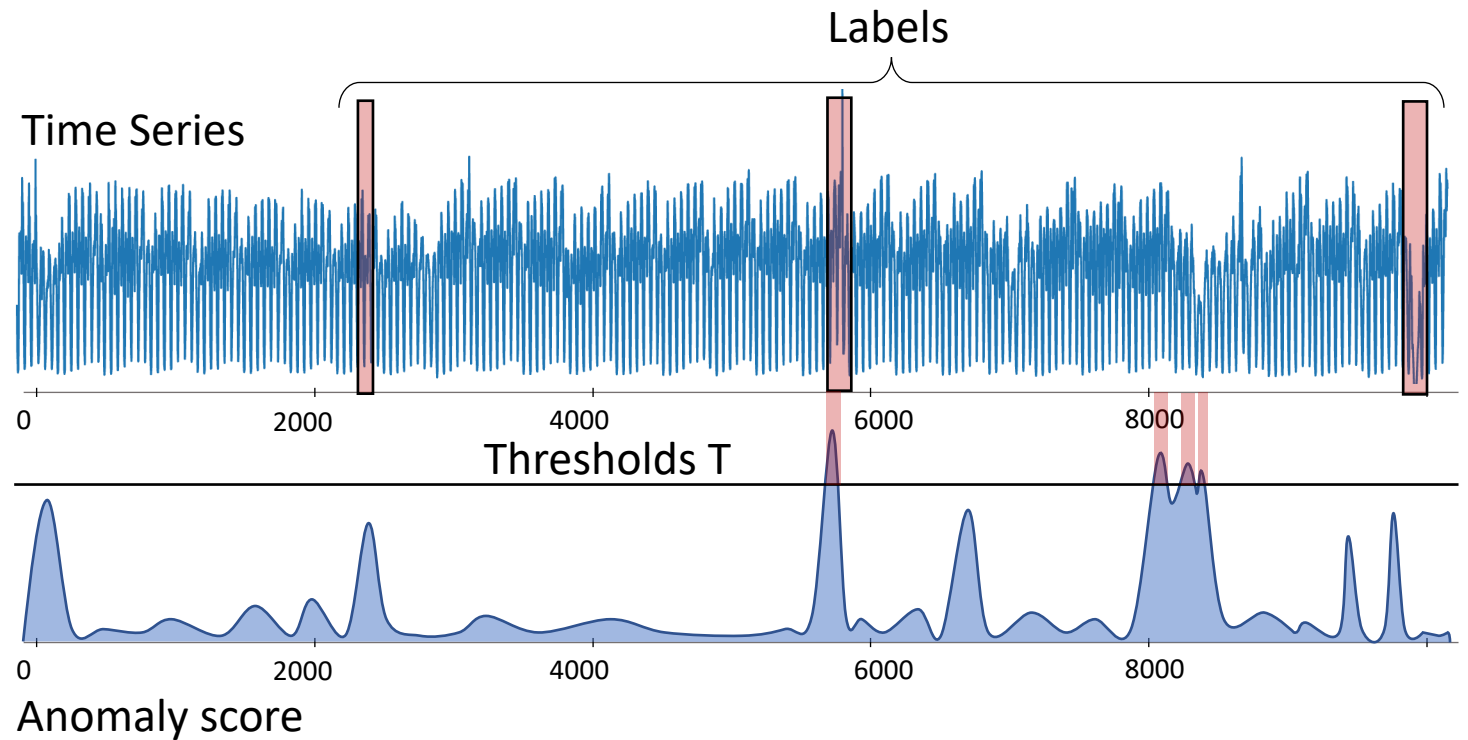
Evaluation measures: *AUC-based*

How do we set the threshold?



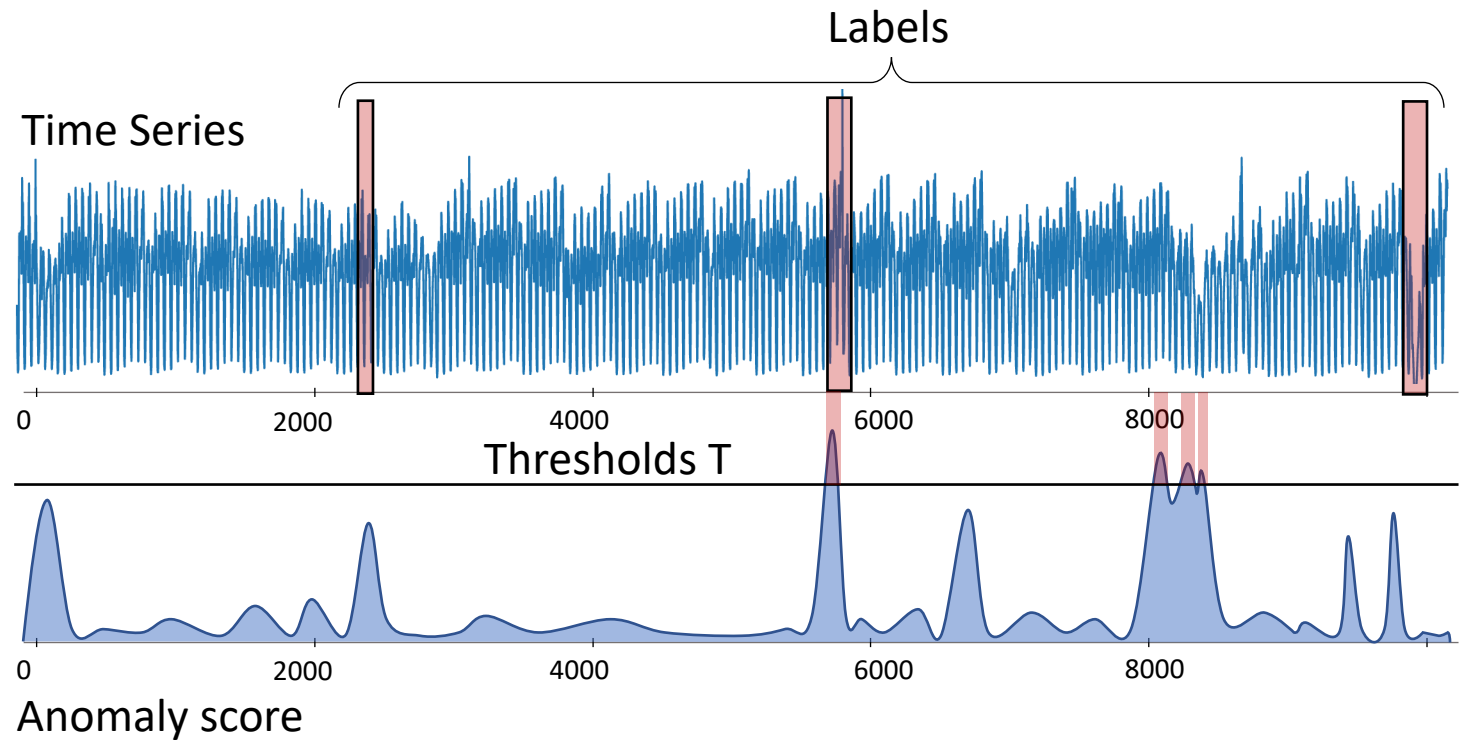
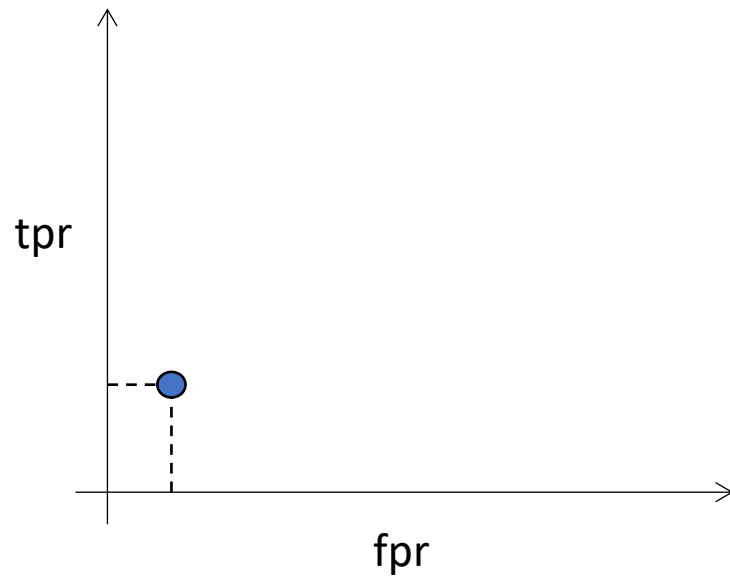
Evaluation measures: *AUC-based*

How do we set the threshold?



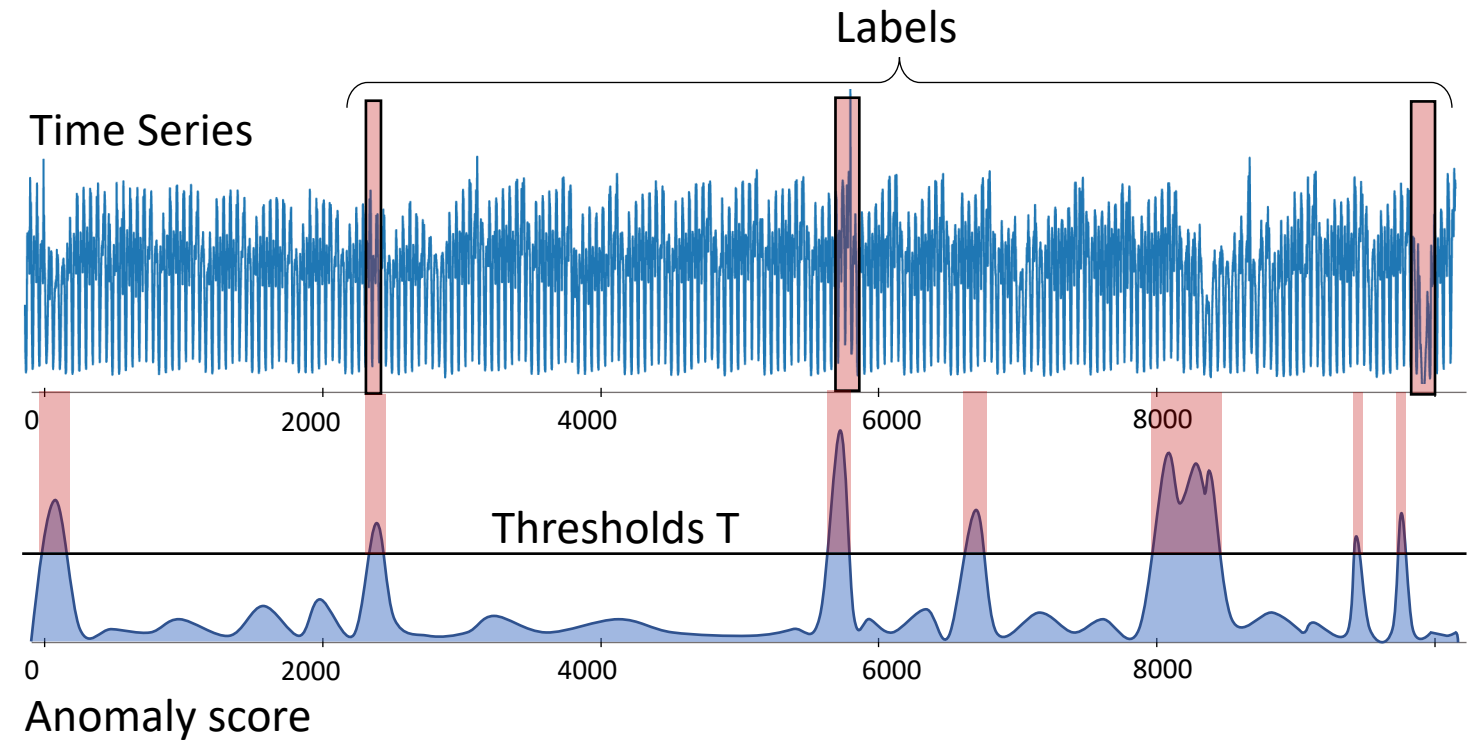
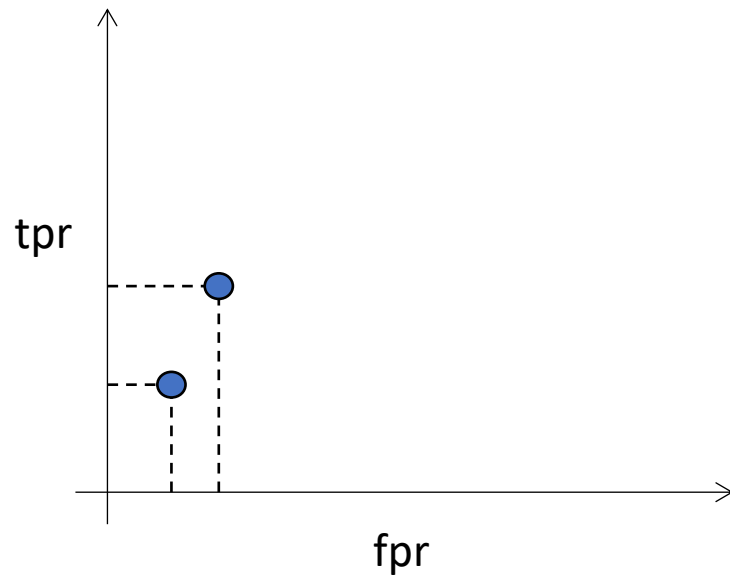
Evaluation measures: *AUC-based*

AUC-based Evaluation Measures:



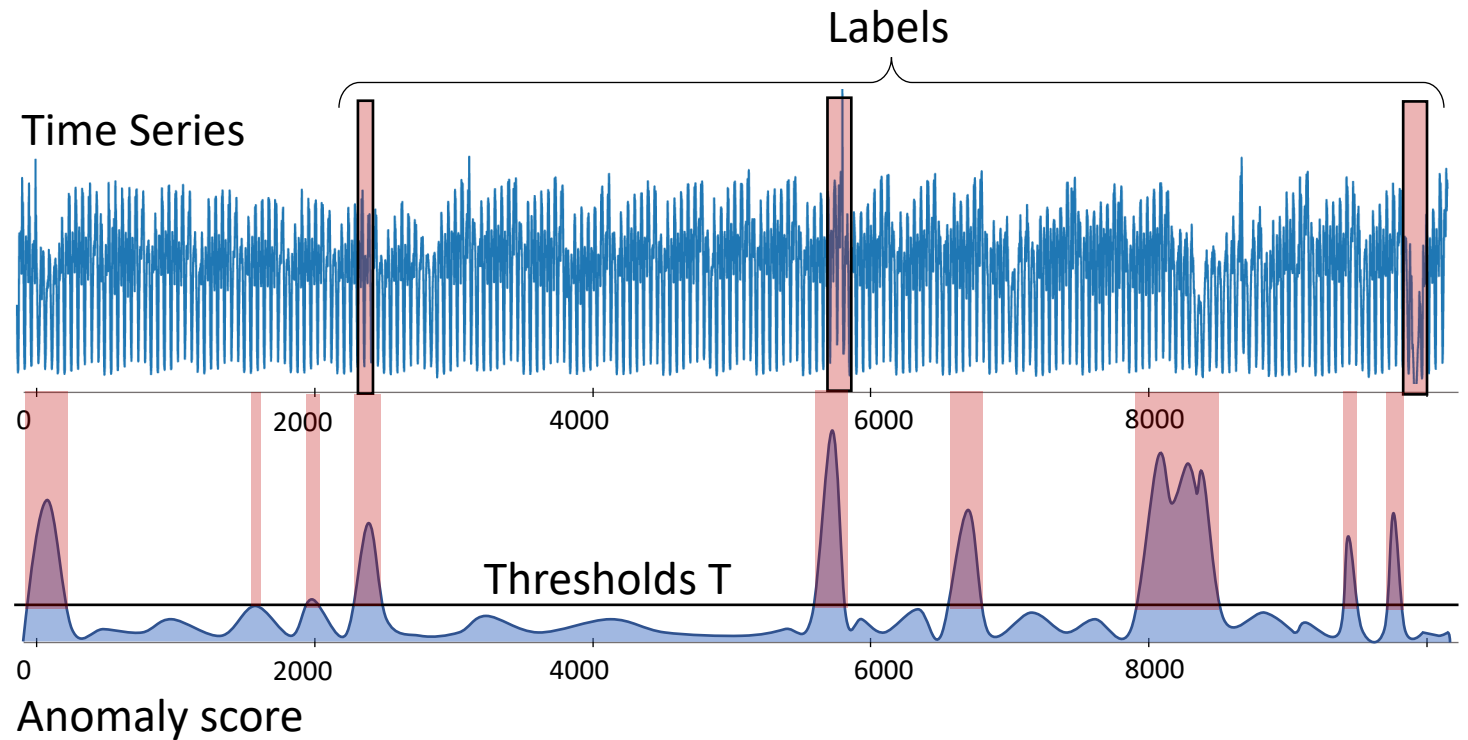
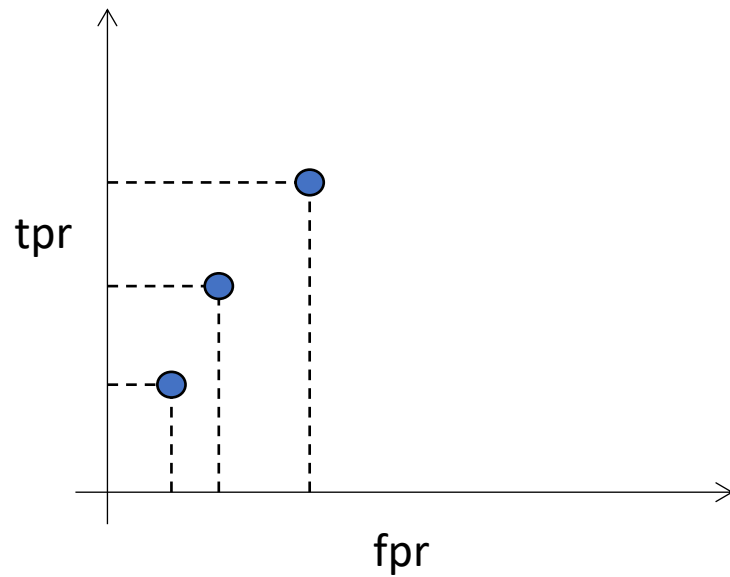
Evaluation measures: *AUC-based*

AUC-based Evaluation Measures:



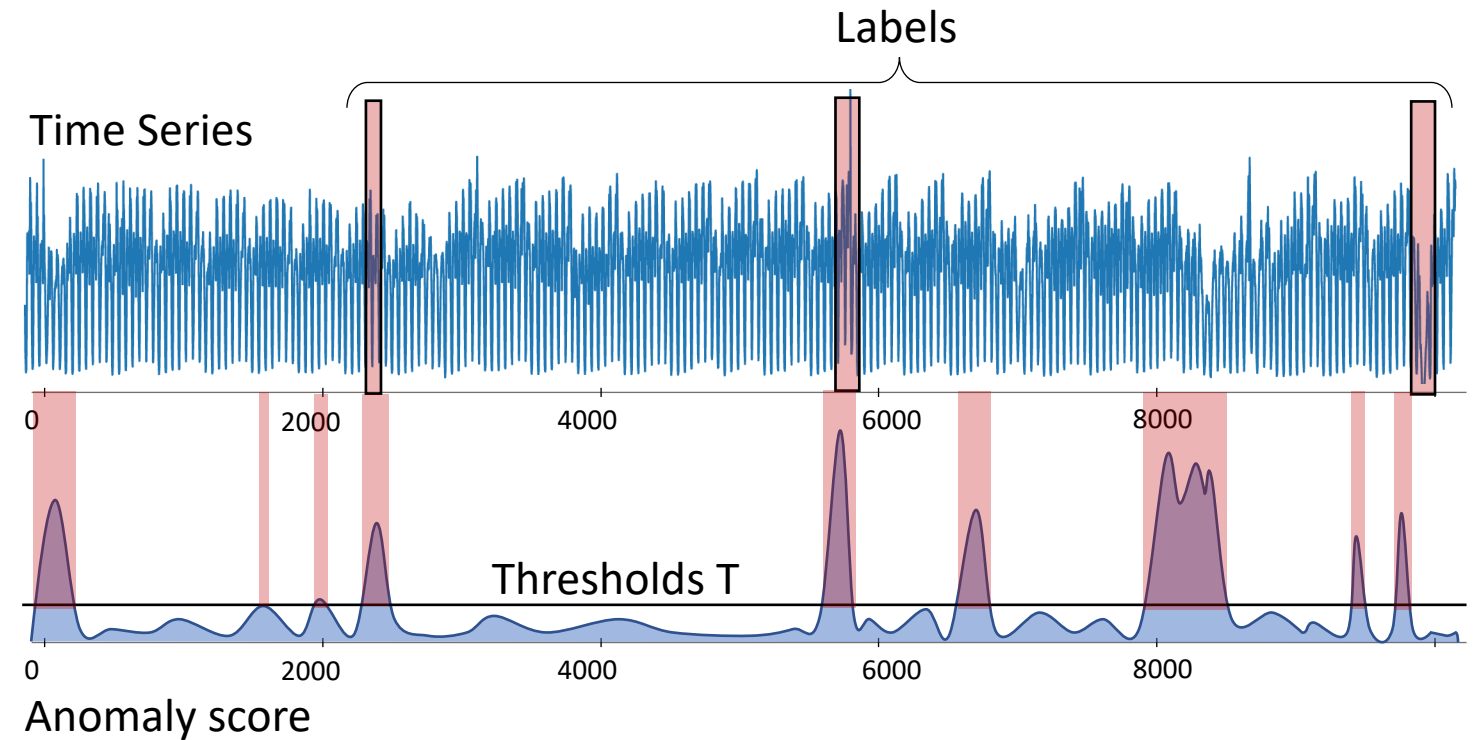
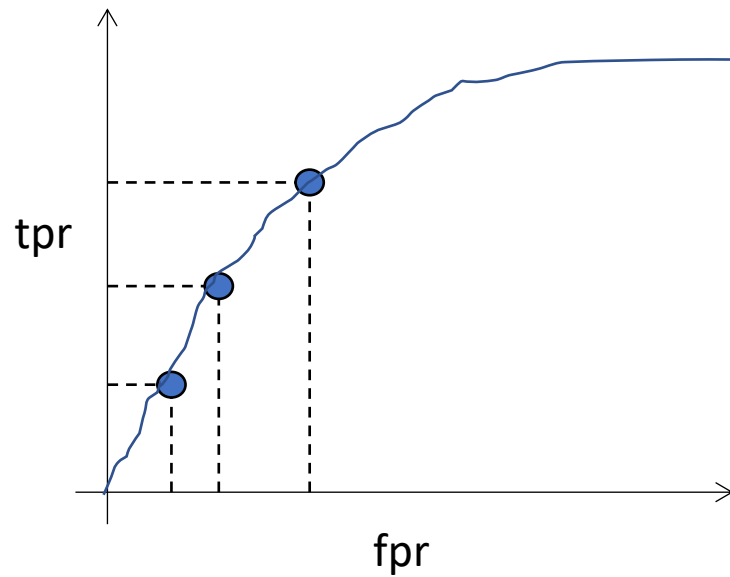
Evaluation measures: *AUC-based*

AUC-based Evaluation Measures:



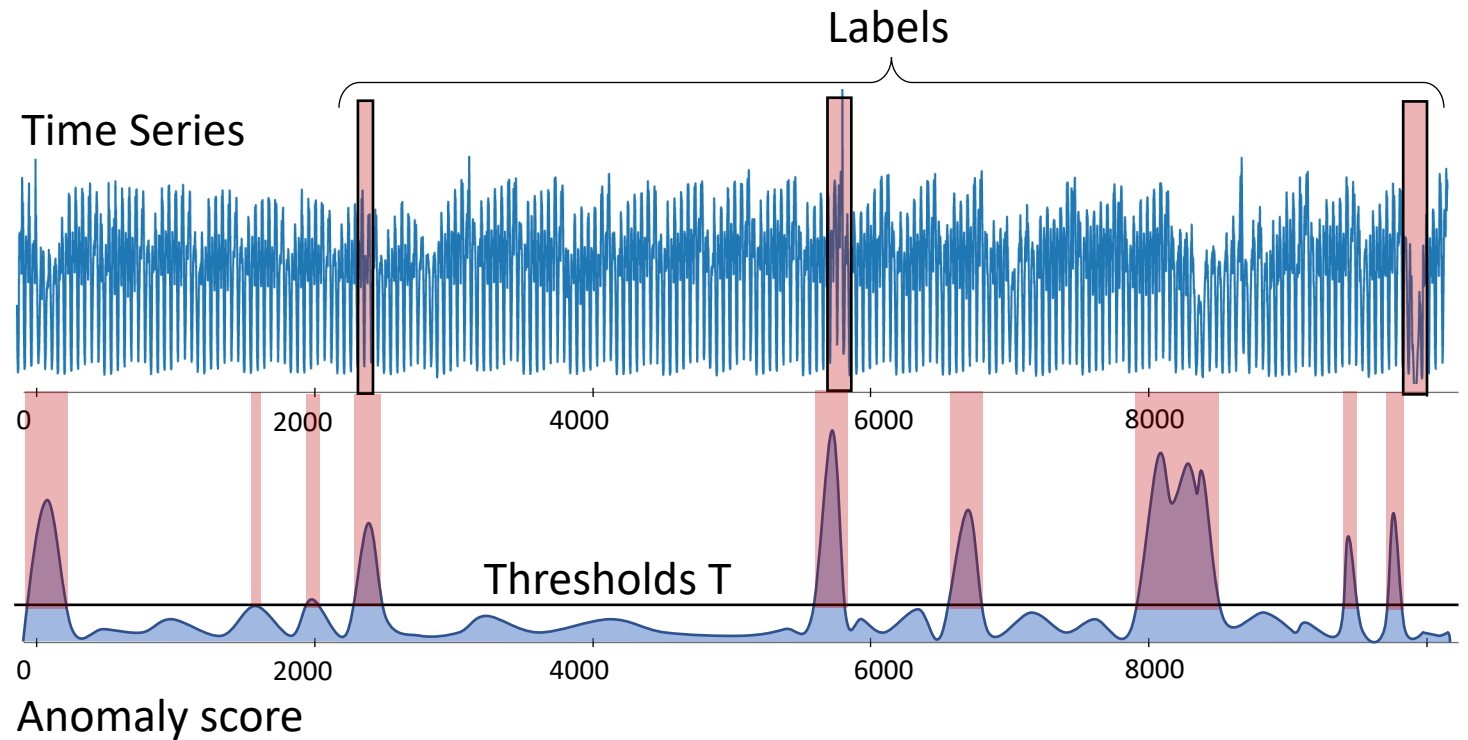
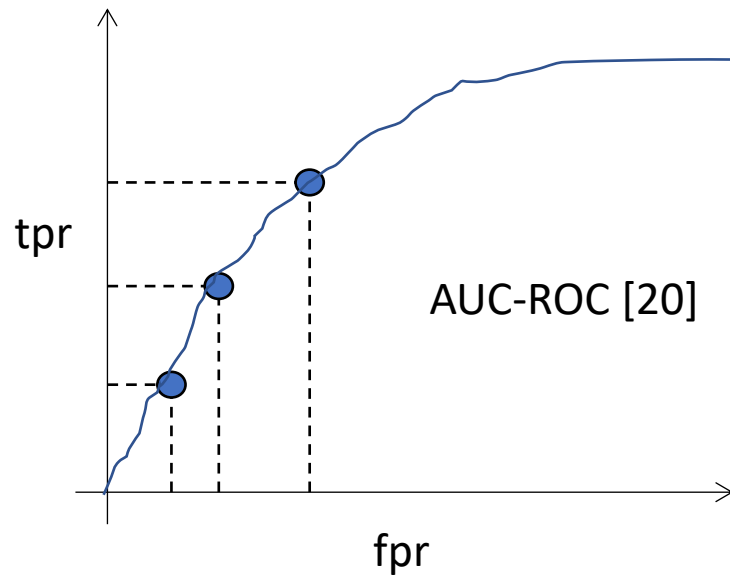
Evaluation measures: *AUC-based*

AUC-based Evaluation Measures:



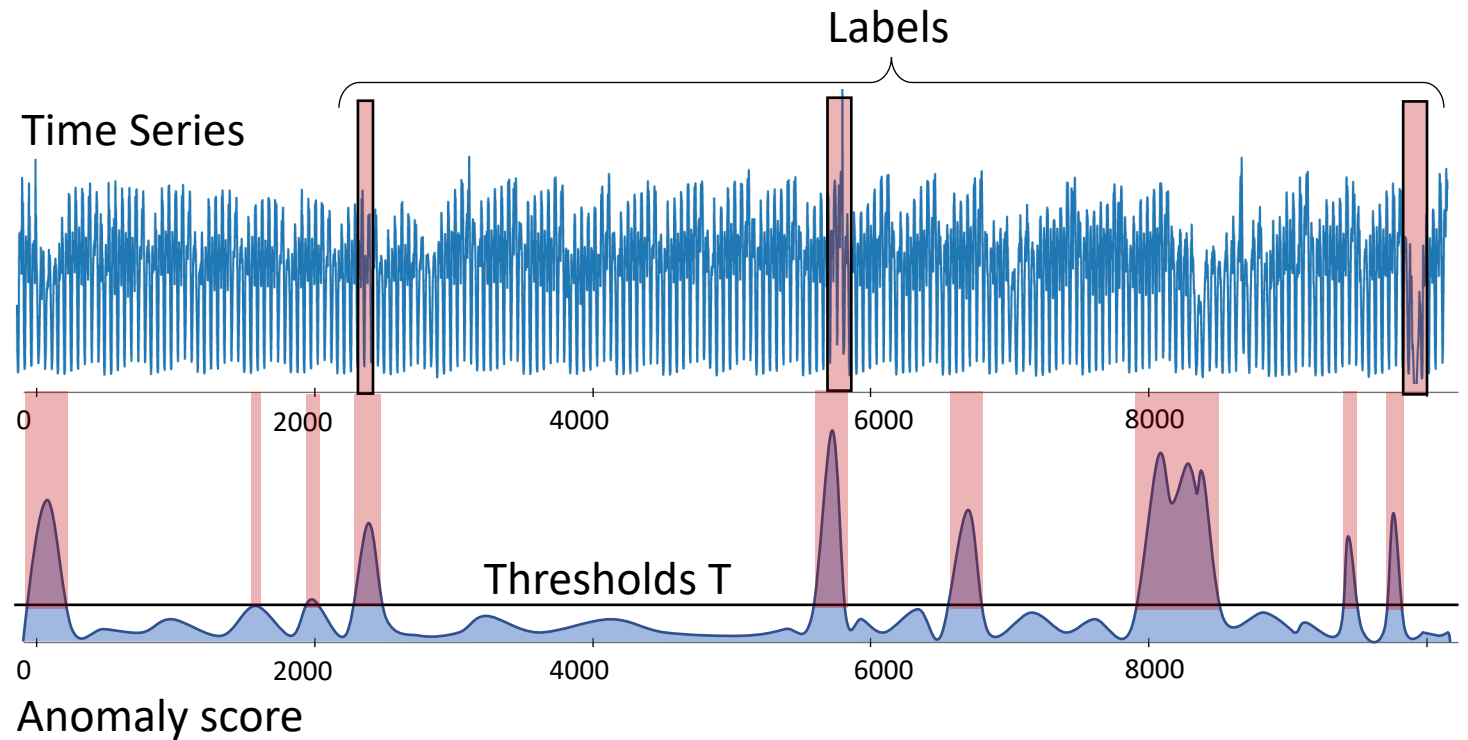
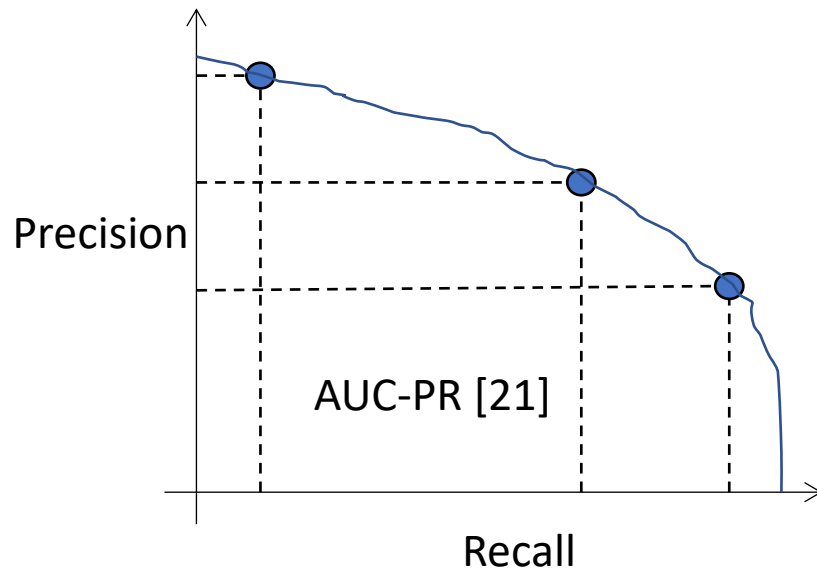
Evaluation measures: *AUC-based*

AUC-based Evaluation Measures:



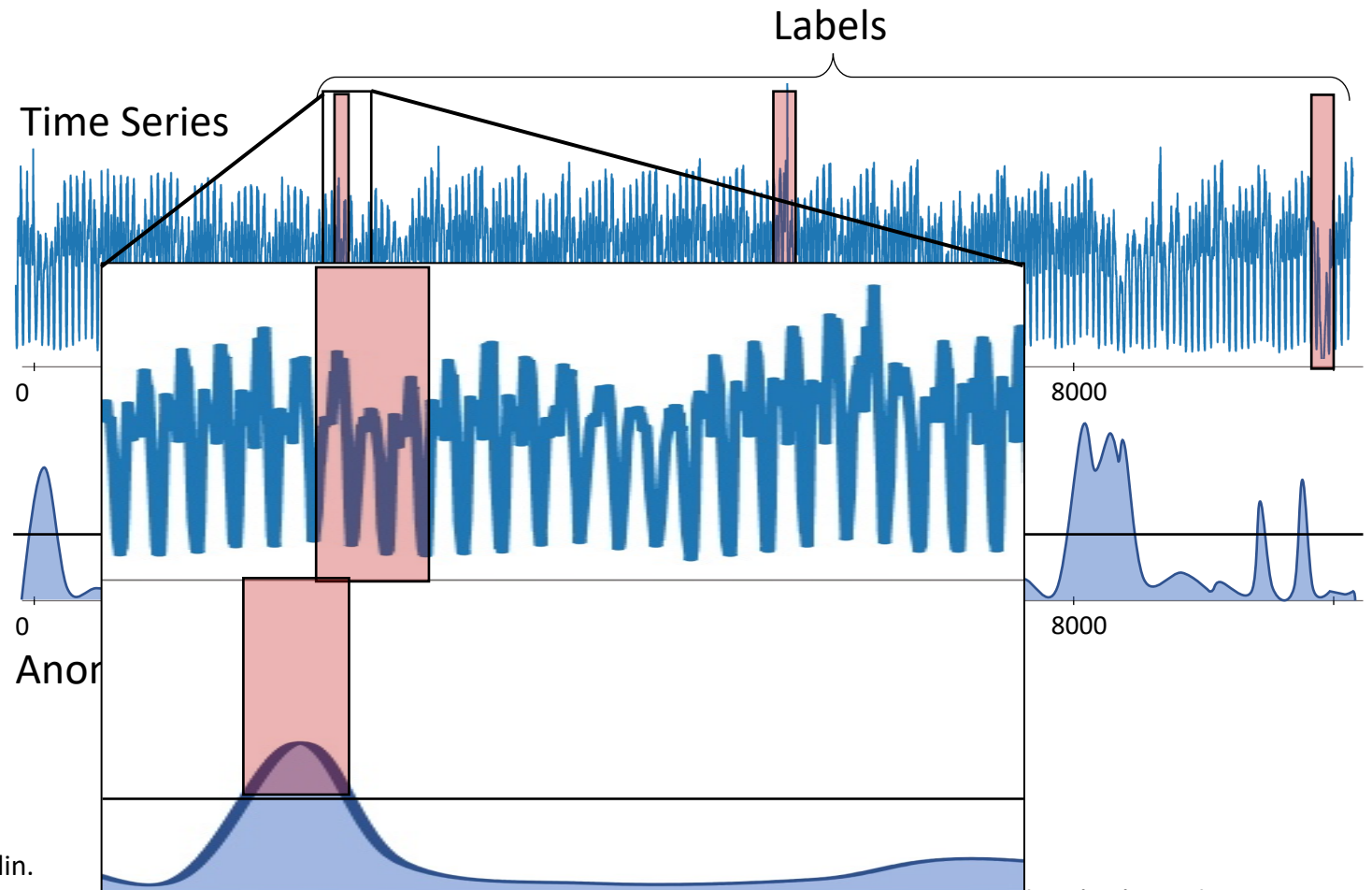
Evaluation measures: *AUC-based*

AUC-based Evaluation Measures:



Evaluation measures: *Labeling issue*

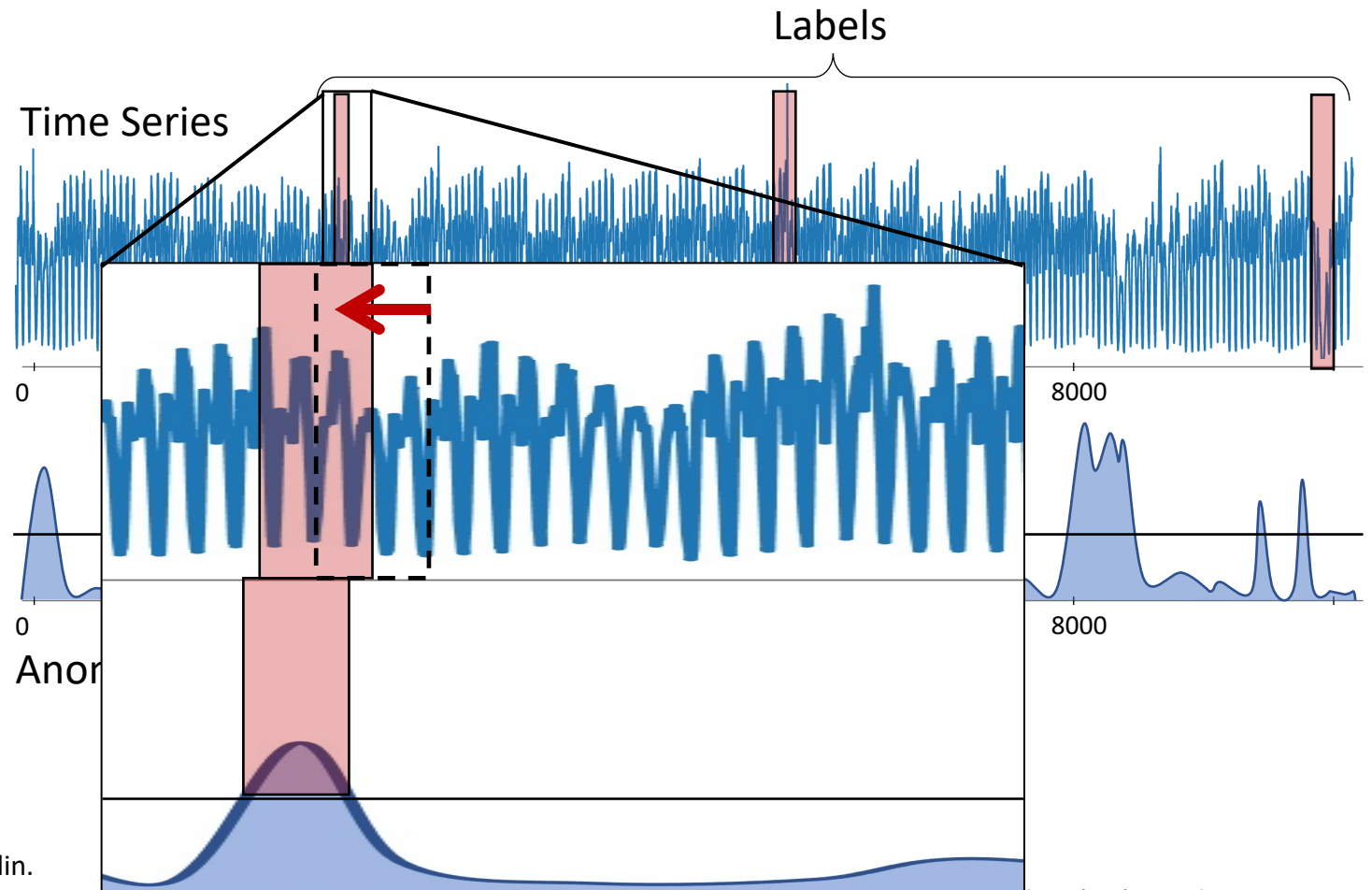
Labeling can be an issue for time series [22]:



[22] J. Paparrizos, P. Boniol, T. Palpanas, R. S. Tsay, A. Elmore, and M. J. Franklin. Volume under the surface: a new accuracy evaluation measure for time-series anomaly detection. *Proc. VLDB Endow.* 15, 11 (2022), 2774–2787.

Evaluation measures: *Labeling issue*

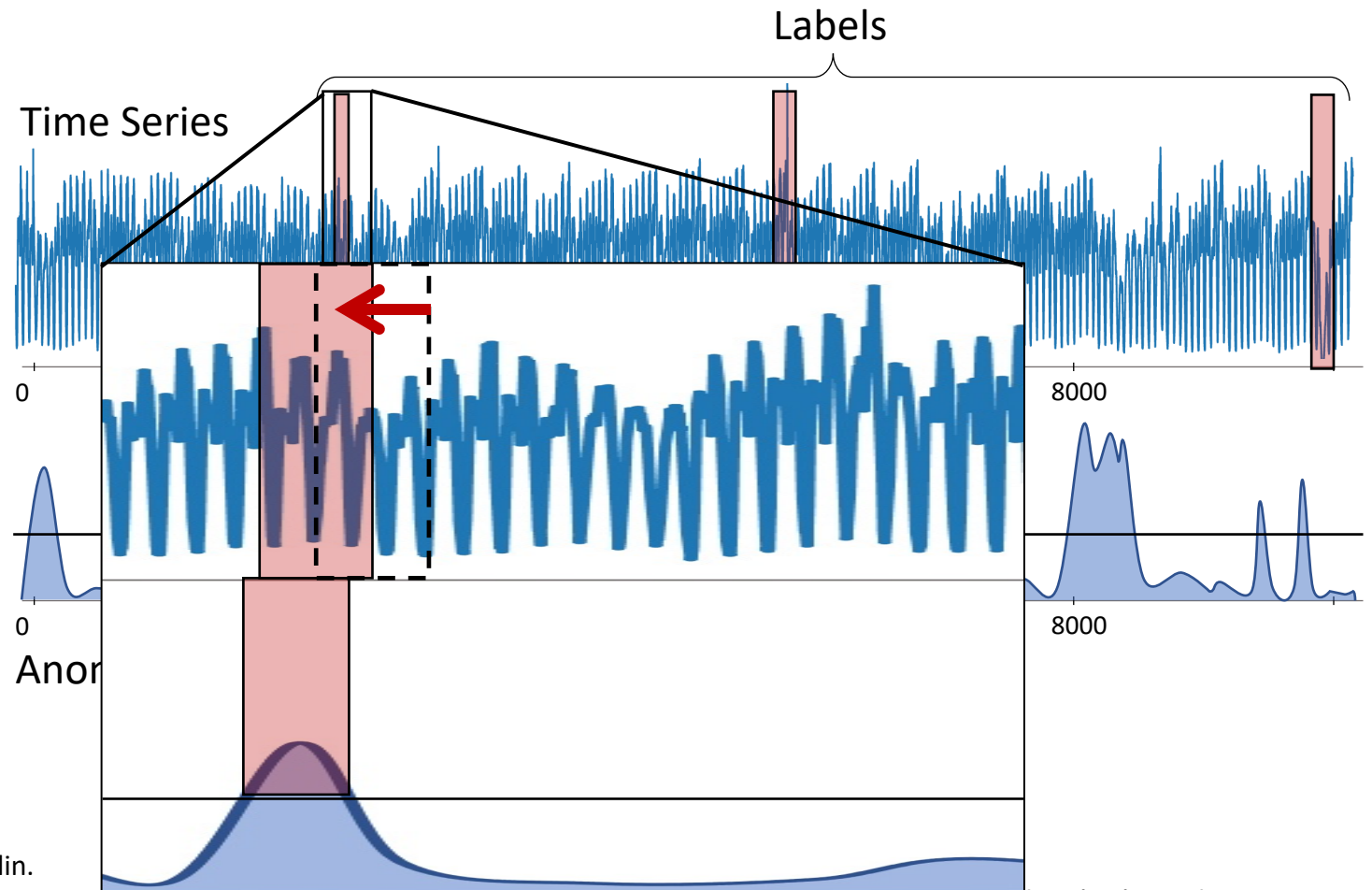
Labeling can be an issue for time series [22]:



Evaluation measures: *Labeling issue*

Labeling can be an issue for time series [22]:

- Misalignment can lead to significant changes of accuracy values.



Evaluation measures: *Labeling issue*

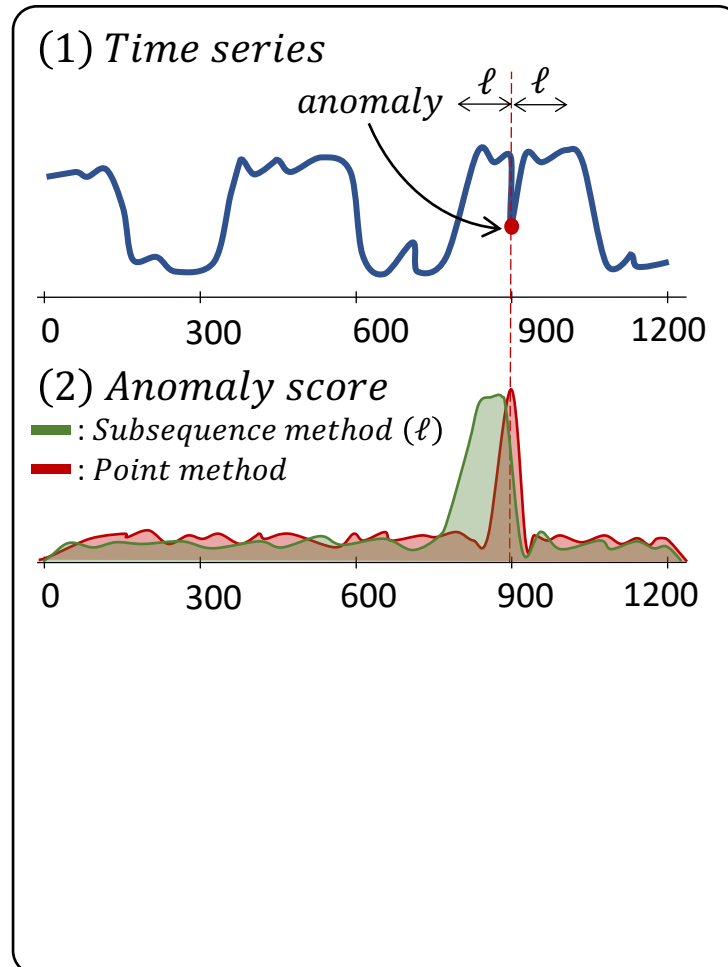
Labeling can be an issue for time series [22]:

- Misalignment can lead to significant changes of accuracy values.

Evaluation measures: *Labeling issue*

Labeling can be an issue for time series [22]:

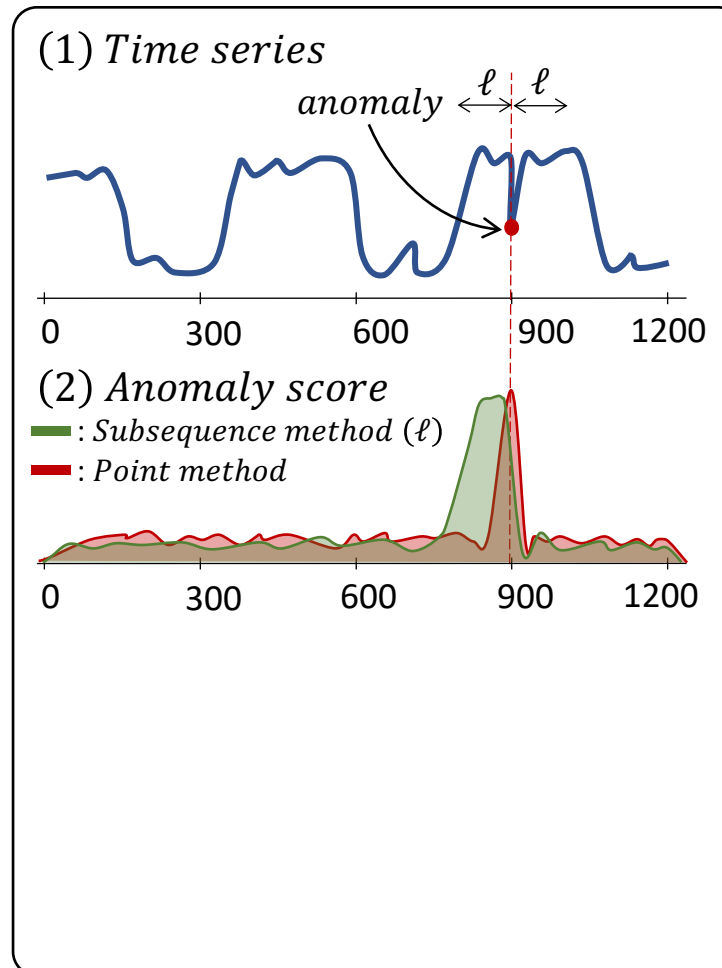
- Misalignment can lead to significant changes of accuracy values.
- This is a real issue because of:
 - Methods that produce **misaligned anomaly scores**.



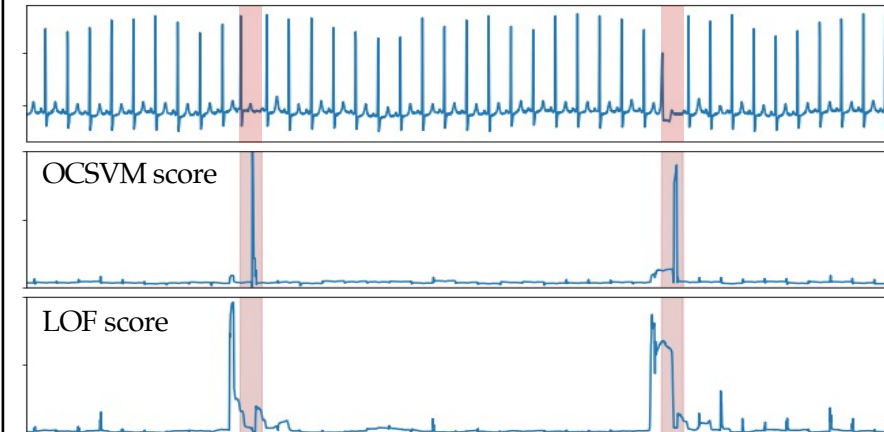
Evaluation measures: *Labeling issue*

Labeling can be an issue for time series [22]:

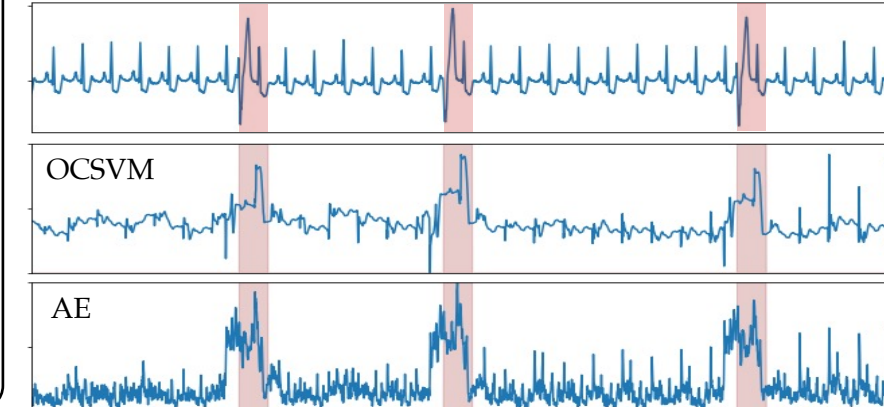
- Misalignment can lead to significant changes of accuracy values.
- This is a real issue because of:
 - Methods that produce **misaligned anomaly scores**.



Example 1



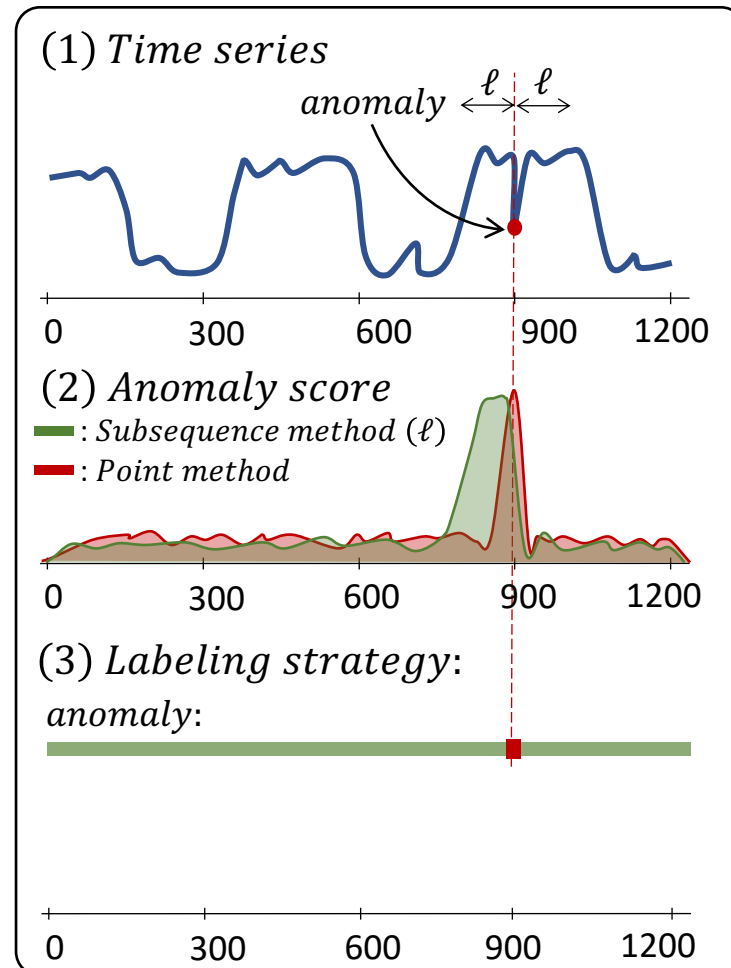
Example 2



Evaluation measures: *Labeling issue*

Labeling can be an issue for time series [22]:

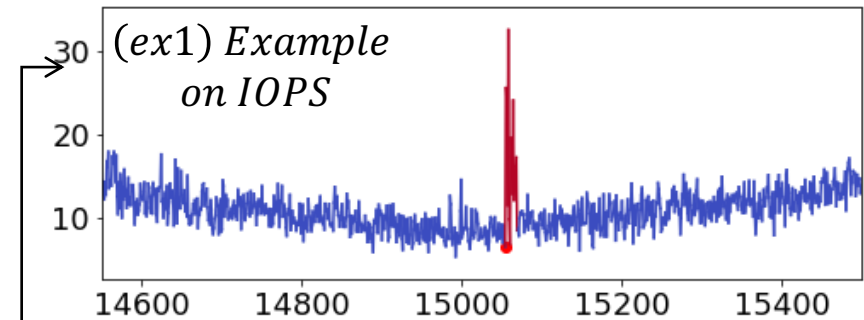
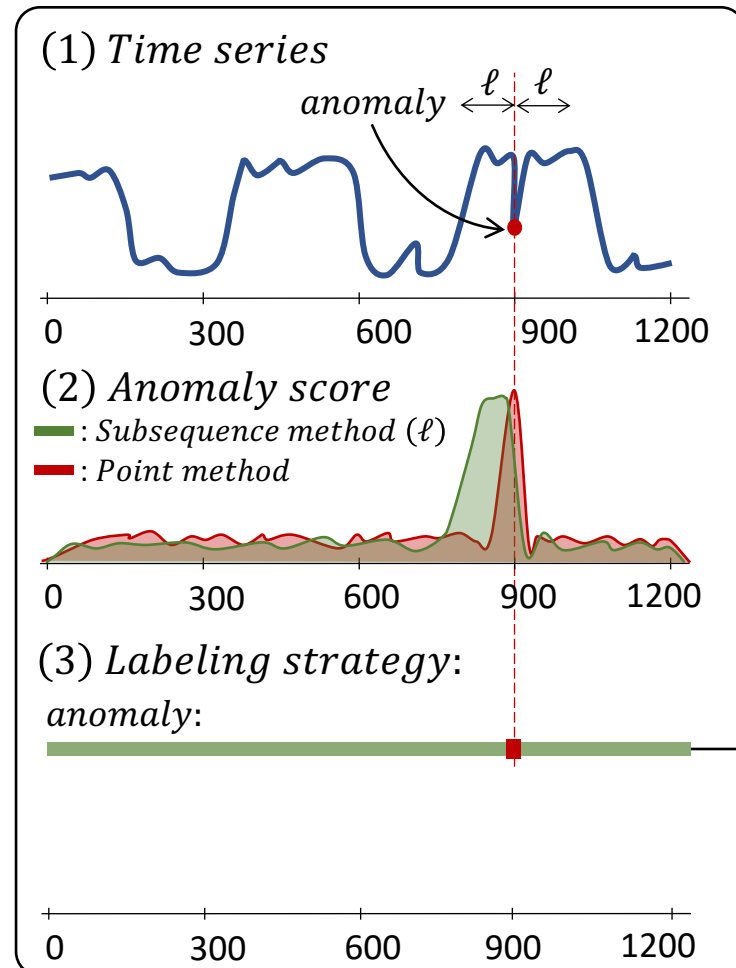
- Misalignment can lead to significant changes of accuracy values.
- This is a real issue because of:
 - Methods that produce **misaligned anomaly scores**.
 - **Different Labeling strategies** between domains and applications



Evaluation measures: *Labeling issue*

Labeling can be an issue for time series [22]:

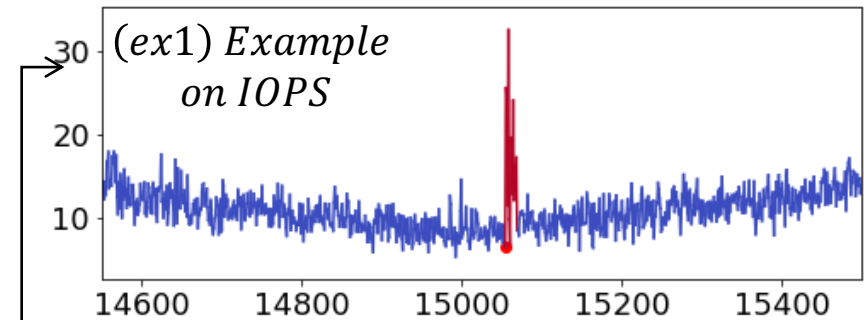
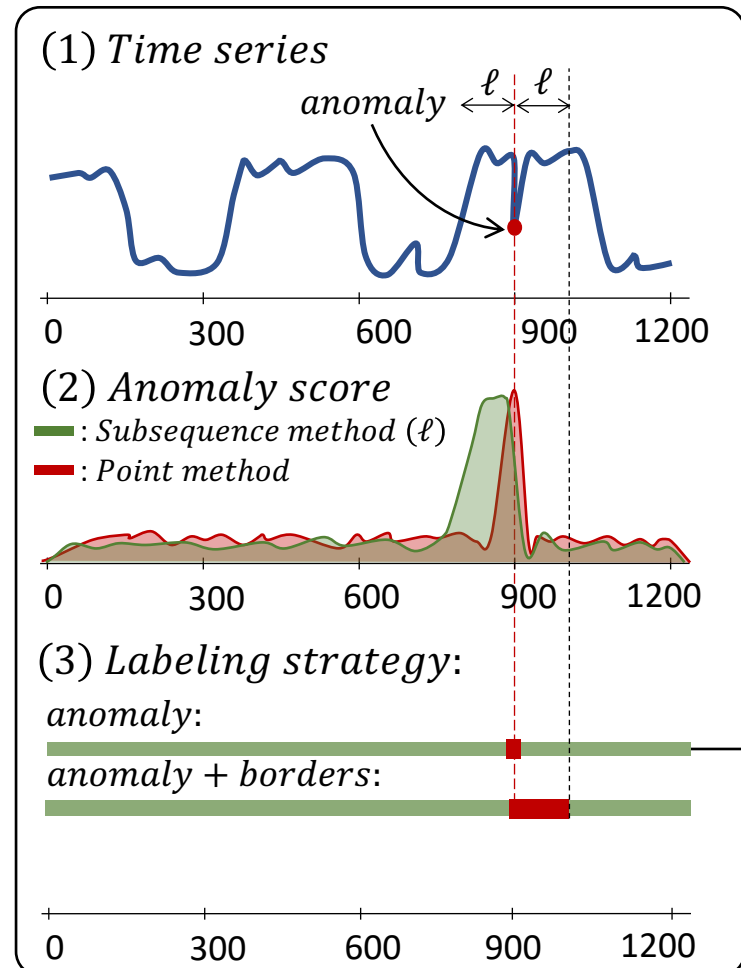
- Misalignment can lead to significant changes of accuracy values.
- This is a real issue because of:
 - Methods that produce **misaligned anomaly scores**.
 - **Different Labeling strategies** between domains and applications



Evaluation measures: *Labeling issue*

Labeling can be an issue for time series [22]:

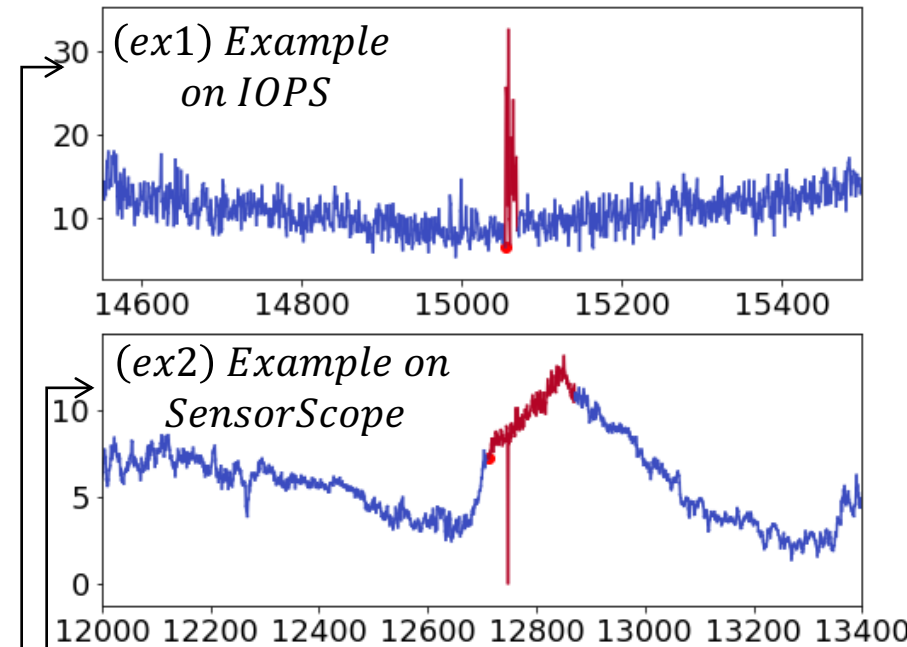
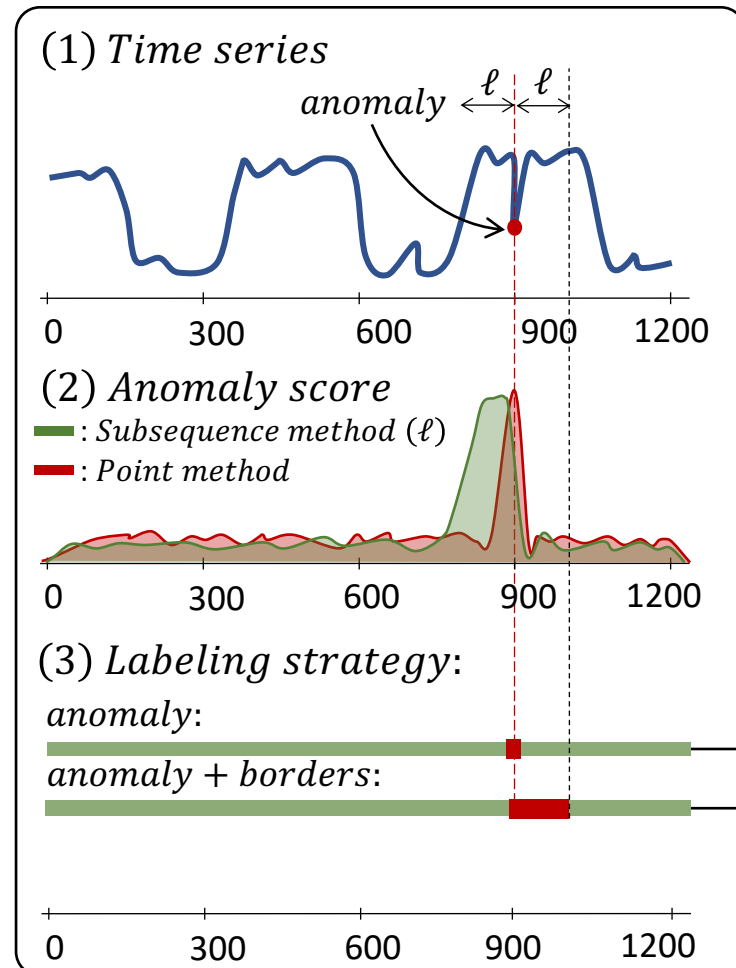
- Misalignment can lead to significant changes of accuracy values.
- This is a real issue because of:
 - Methods that produce **misaligned anomaly scores**.
 - **Different Labeling strategies** between domains and applications



Evaluation measures: *Labeling issue*

Labeling can be an issue for time series [22]:

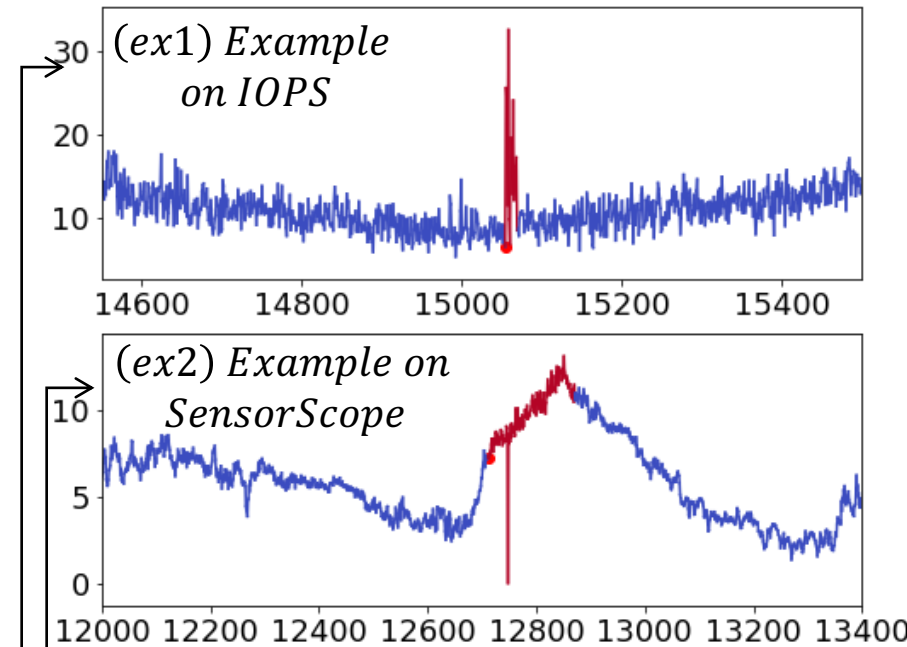
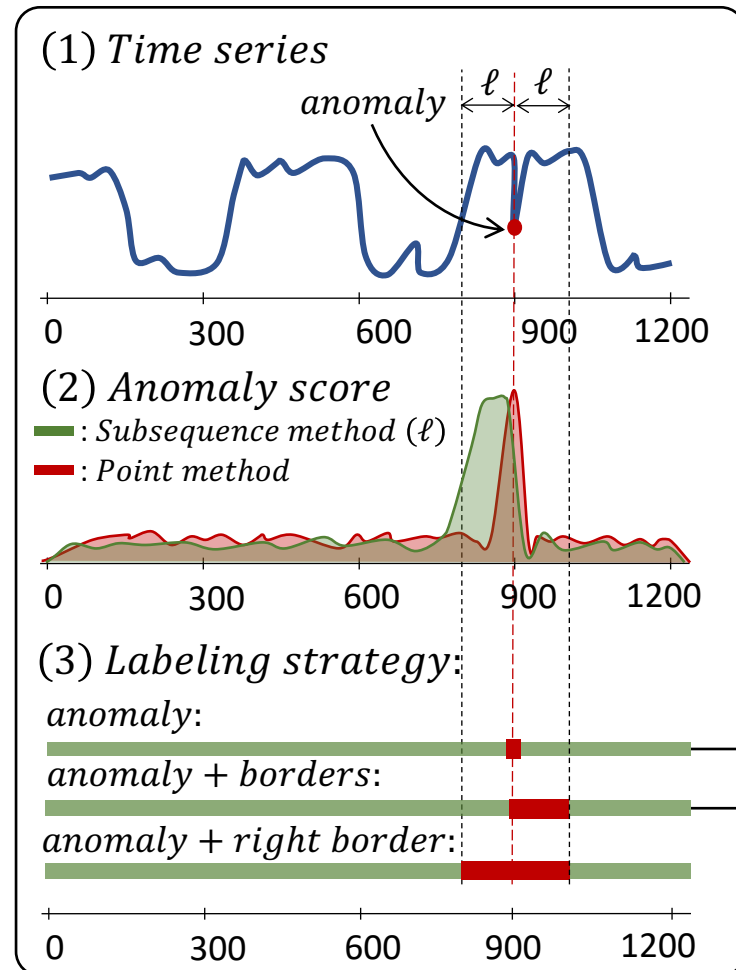
- Misalignment can lead to significant changes of accuracy values.
- This is a real issue because of:
 - Methods that produce **misaligned anomaly scores**.
 - **Different Labeling strategies** between domains and applications



Evaluation measures: *Labeling issue*

Labeling can be an issue for time series [22]:

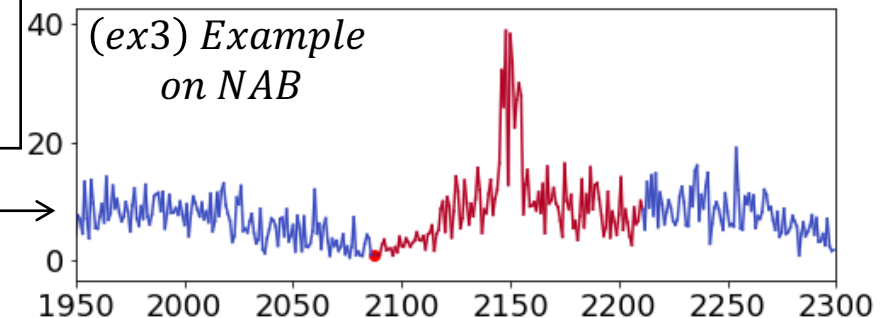
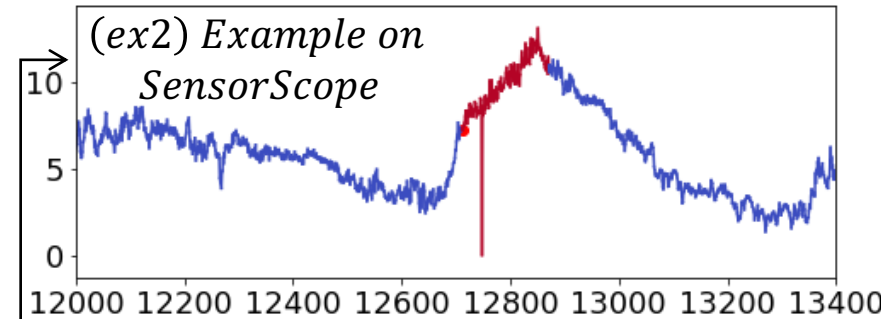
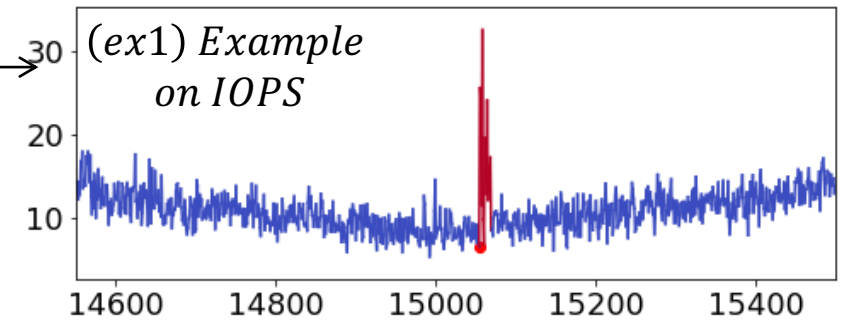
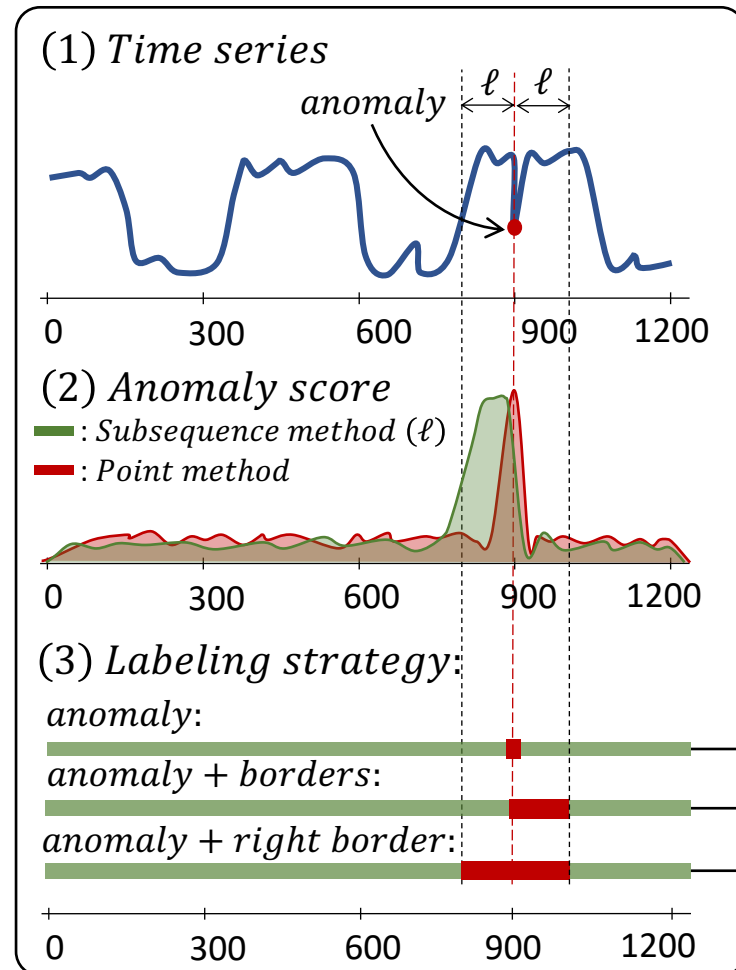
- Misalignment can lead to significant changes of accuracy values.
- This is a real issue because of:
 - Methods that produce **misaligned anomaly scores**.
 - **Different Labeling strategies** between domains and applications



Evaluation measures: *Labeling issue*

Labeling can be an issue for time series [22]:

- Misalignment can lead to significant changes of accuracy values.
- This is a real issue because of:
 - Methods that produce **misaligned anomaly scores**.
 - **Different Labeling strategies** between domains and applications



Evaluation measures: *Labeling issue*

Existing solutions:

- Range Precision and Recall [23]:

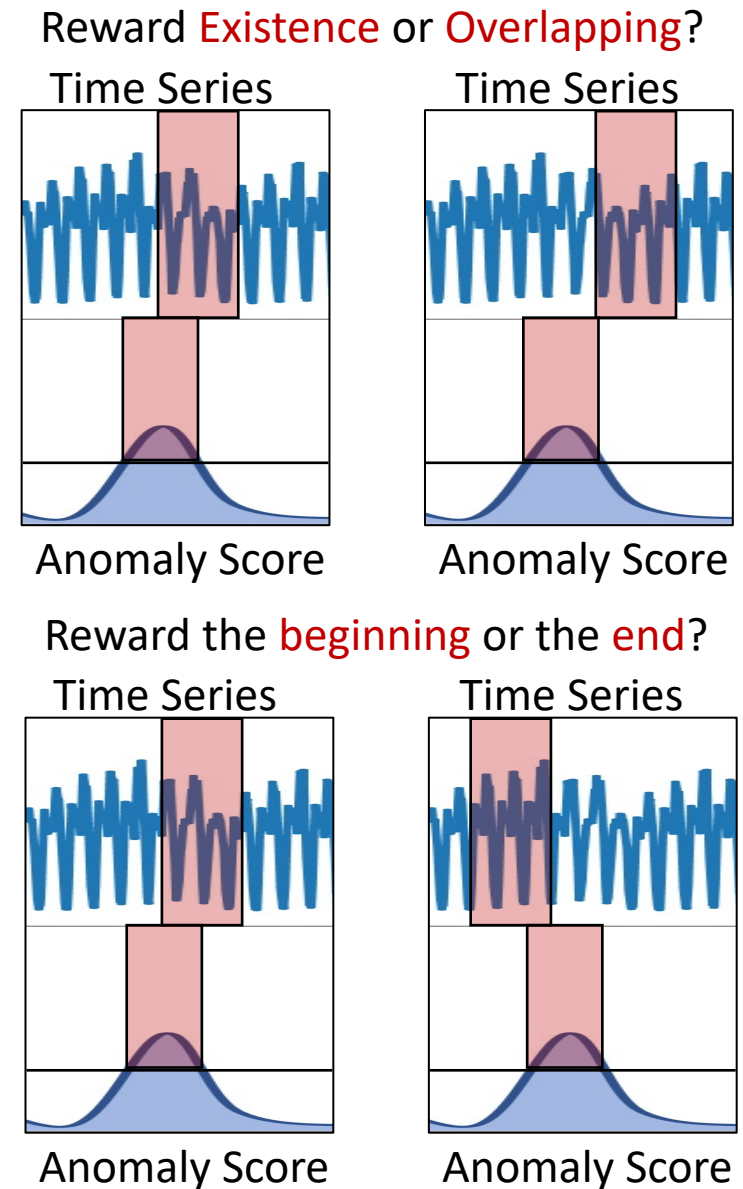
- $Recall_T(R, P) = \frac{\sum_{i=1}^{N_r} Recall_T(R_i, P)}{N_r}$

- $Recall_T(R_i, P) = \alpha * ExistenceR(R_i, P) + (1 - \alpha) * OverlappingR(R_i, P)$

- $Precision_T(R, P) = \frac{\sum_{i=1}^{N_p} Precision_T(R, P_i)}{N_p}$

- $Precision_T(R, P_i) = CardinalityFactor(P_i, R) * \sum_{j=1}^{N_r} w(P_i, P_i \cap R_j, \delta)$

- Functions $w()$, $\delta()$ are tunable functions to represent the overlap size and position respectively.



Evaluation measures: *Labeling issue*

Existing solutions:

- Range Precision and Recall

- $Recall_T(R, P) = \frac{\sum_{i=1}^{N_r} P_i}{N_r}$

- $Recall_T(R_i, P) = \alpha * \frac{P_i}{N_r}$

- $Precision_T(R, P) = \frac{P_i}{N_p}$

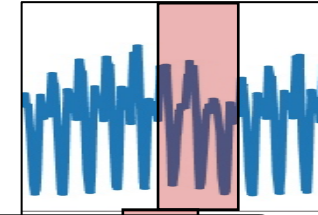
- $Precision_T(R, P_i) = \frac{P_i}{N_p}$

- Functions $w()$, $\delta()$ are tunable functions to represent the overlap size and position respectively.

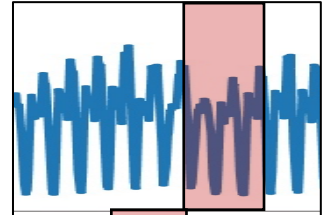
But what to do if the labeling strategy is not the same for all time series?

Reward **Existence** or **Overlapping**?

Time Series



Time Series

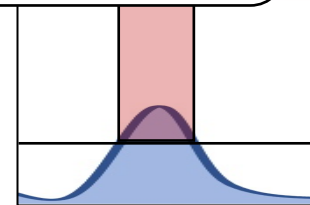
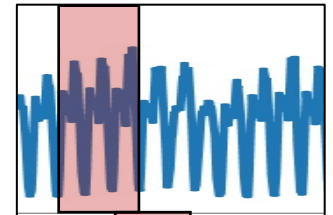


re

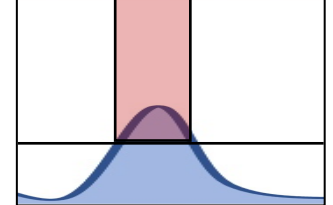
Anomaly Score

beginning or the **end**?

Time Series



Anomaly Score



Anomaly Score

Evaluation measures: *Labeling issue*

Existing solutions:

- Range Precision and Recall

- $Recall_T(R, P) = \frac{\sum_{i=1}^{N_r} R_i}{N_r}$

- $Recall_T(R_i, P) = \alpha * \dots$

- $Precision_T(R, P) = \dots$

- $Precision_T(R, P_i) = \dots$

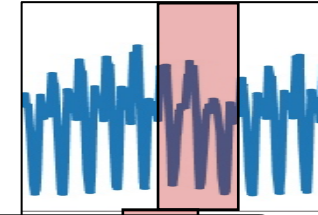
- Functions $w()$, $\delta()$ are tunable functions to represent the overlap size and position respectively.

But what to do if the labeling strategy is not the same for all time series?

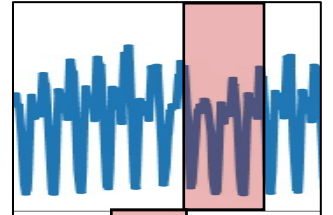
Setting the correct parameters for all time series might not be possible

Reward **Existence** or **Overlapping**?

Time Series



Time Series

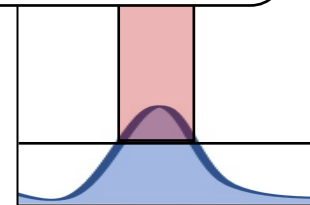
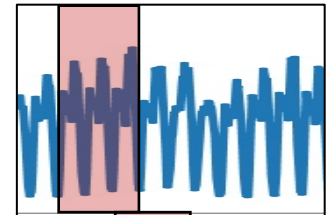


re

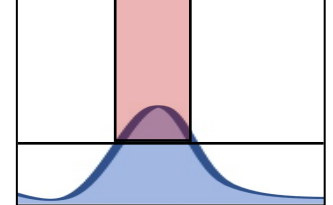
Anomaly Score

beginning or the **end**?

Time Series



Anomaly Score



Anomaly Score

Evaluation measures: *Labeling issue*

Existing solutions:

- Range Precision and Recall

- $Recall_T(R, P) = \frac{\sum_{i=1}^{N_r} R_i}{N_r}$

- $Recall_T(R_i, P) = \alpha * \dots$

- $Precision_T(R, P) = \dots$

- $Precision_T(R, P_i) = \dots$

- Functions $w()$, $\delta()$ are tunable functions to represent the overlap size and position respectively.

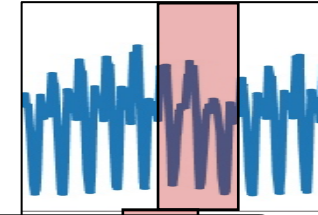
But what to do if the labeling strategy is not the same for all time series?

Setting the correct parameters for all time series might not be possible

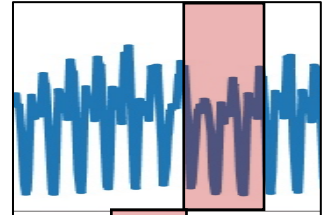
But are these labeling issues a real problem?

Reward **Existence** or **Overlapping**?

Time Series



Time Series

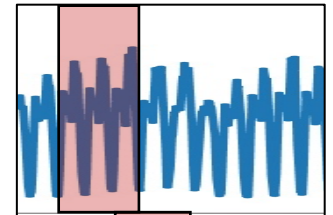


re

Anomaly Score

beginning or the **end**?

Time Series

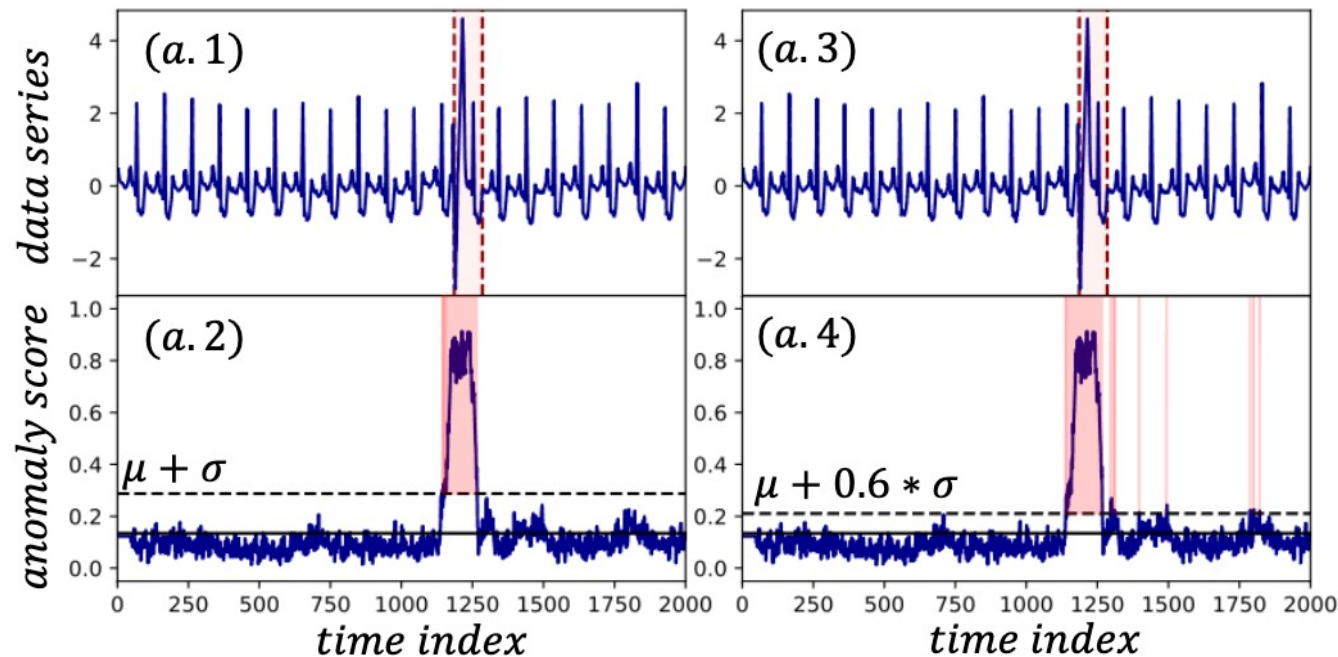


Anomaly Score

Anomaly Score

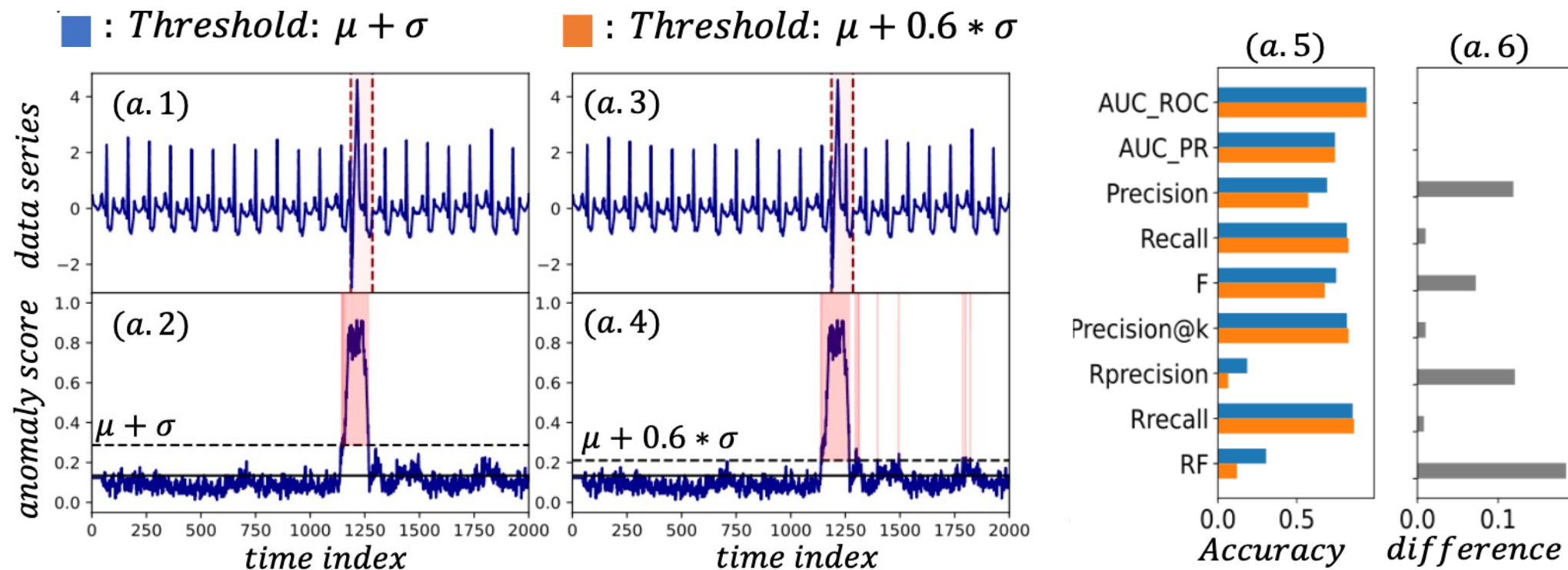
Evaluation measures: *Labeling issue*

(a) *Threshold selection impact on accuracy measures*



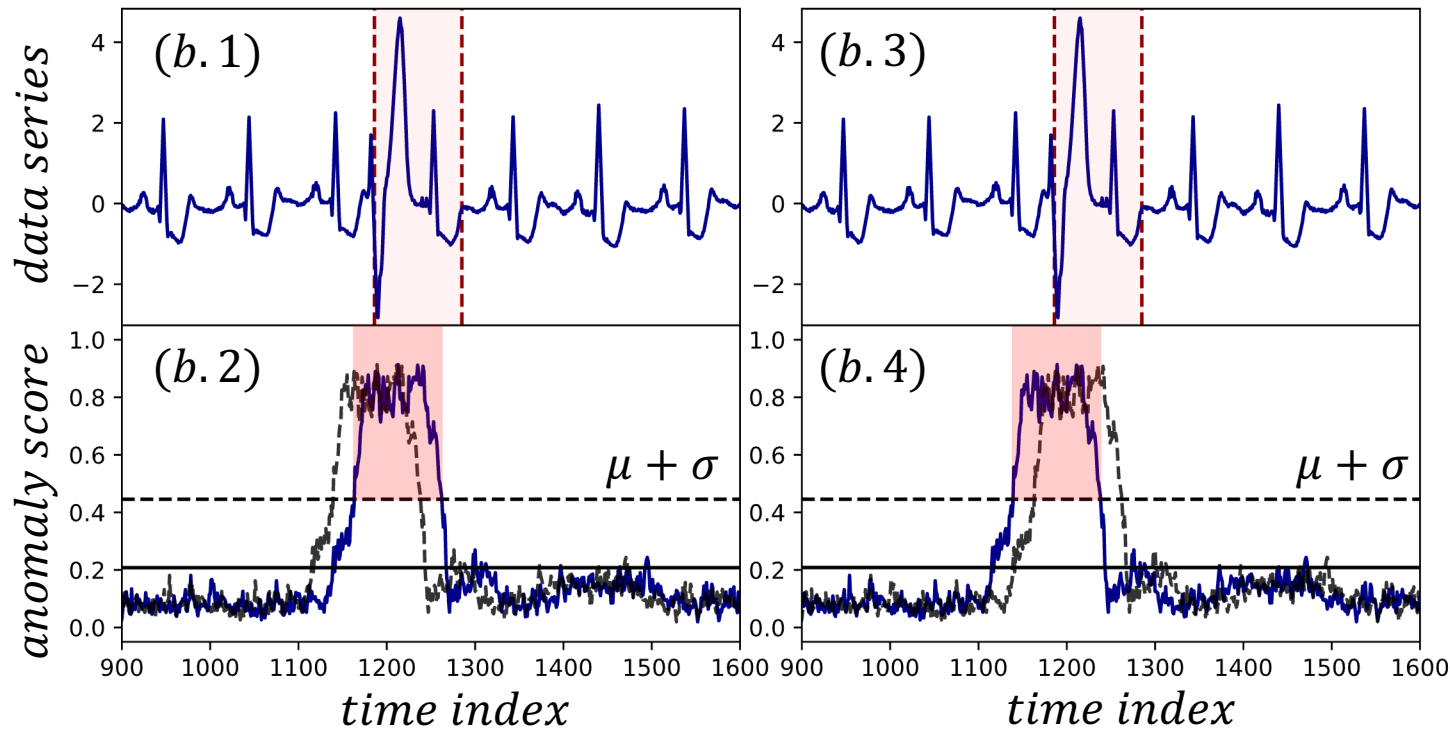
Evaluation measures: *Labeling issue*

(a) *Threshold selection impact on accuracy measures*



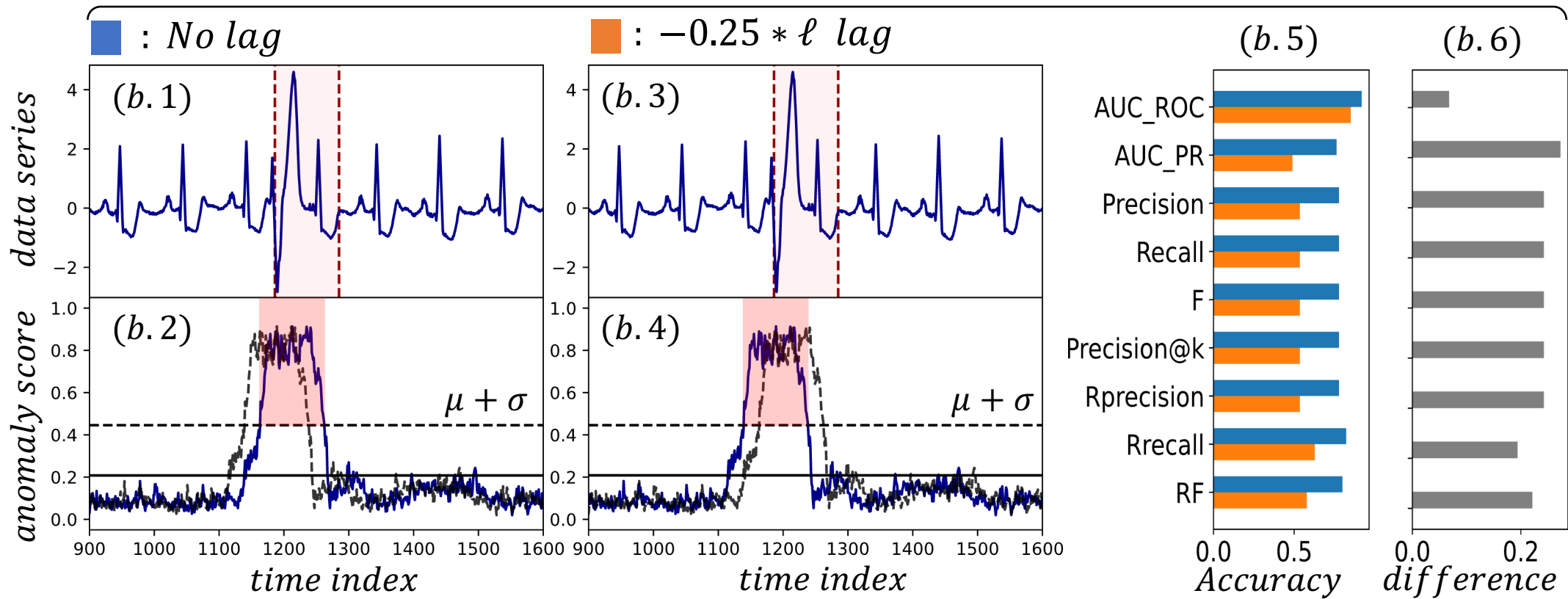
Evaluation measures: *Labeling issue*

(b) *Lag impact on accuracy measures*



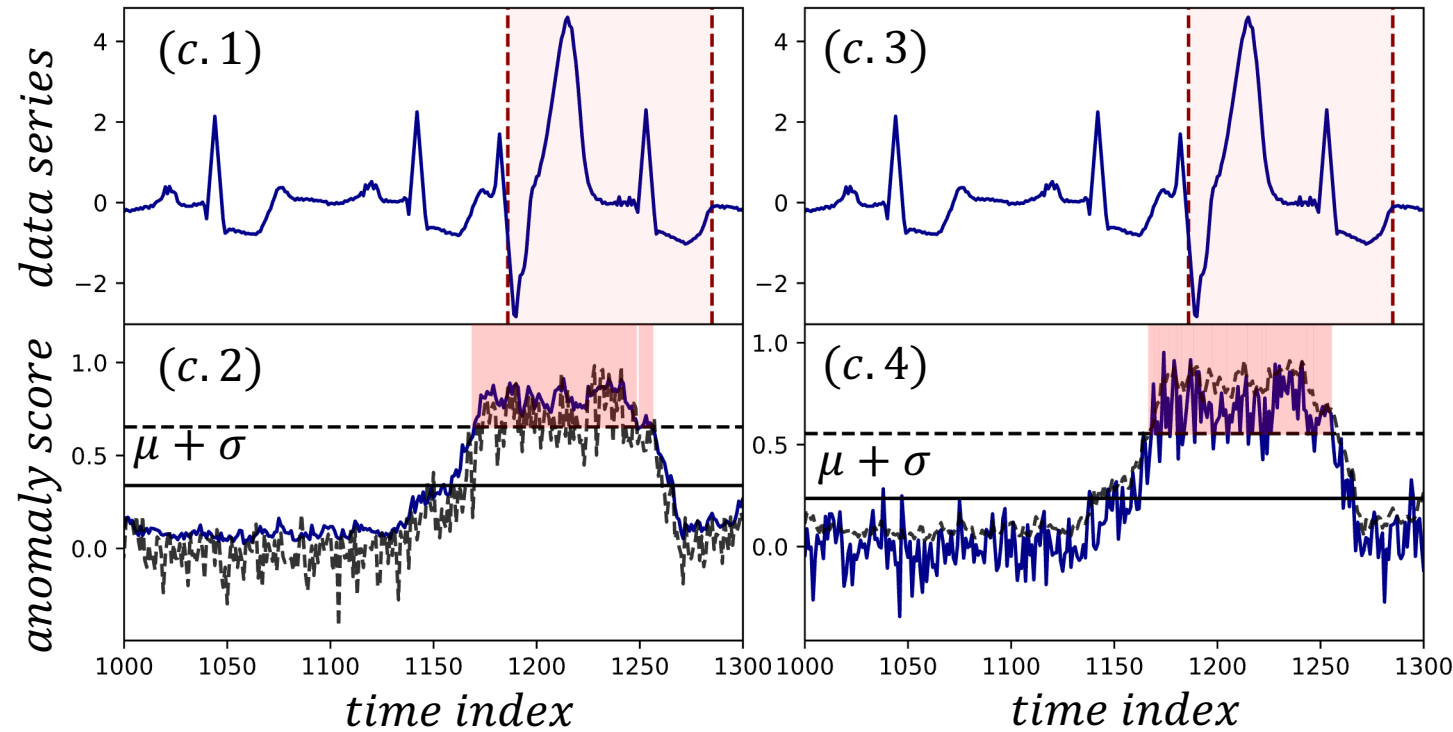
Evaluation measures: *Labeling issue*

(b) *Lag impact on accuracy measures*



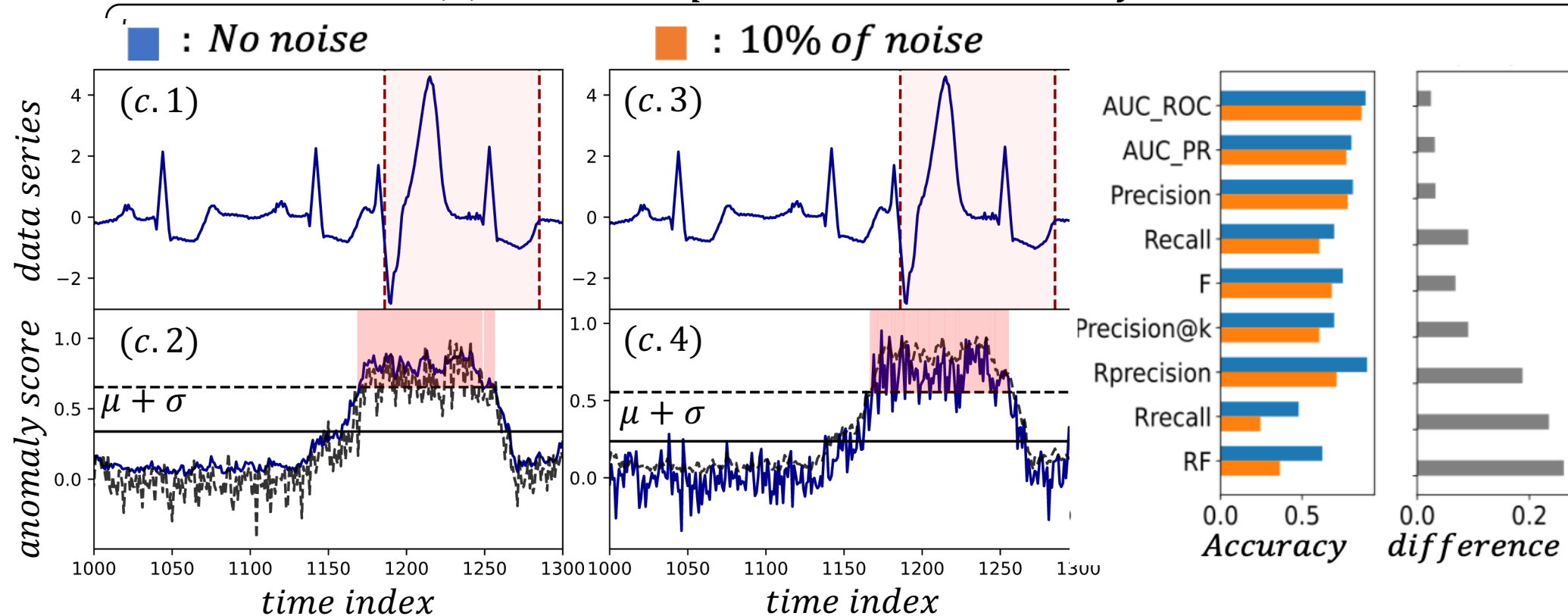
Evaluation measures: *Labeling issue*

(c) *Noise impact on the accuracy measures*



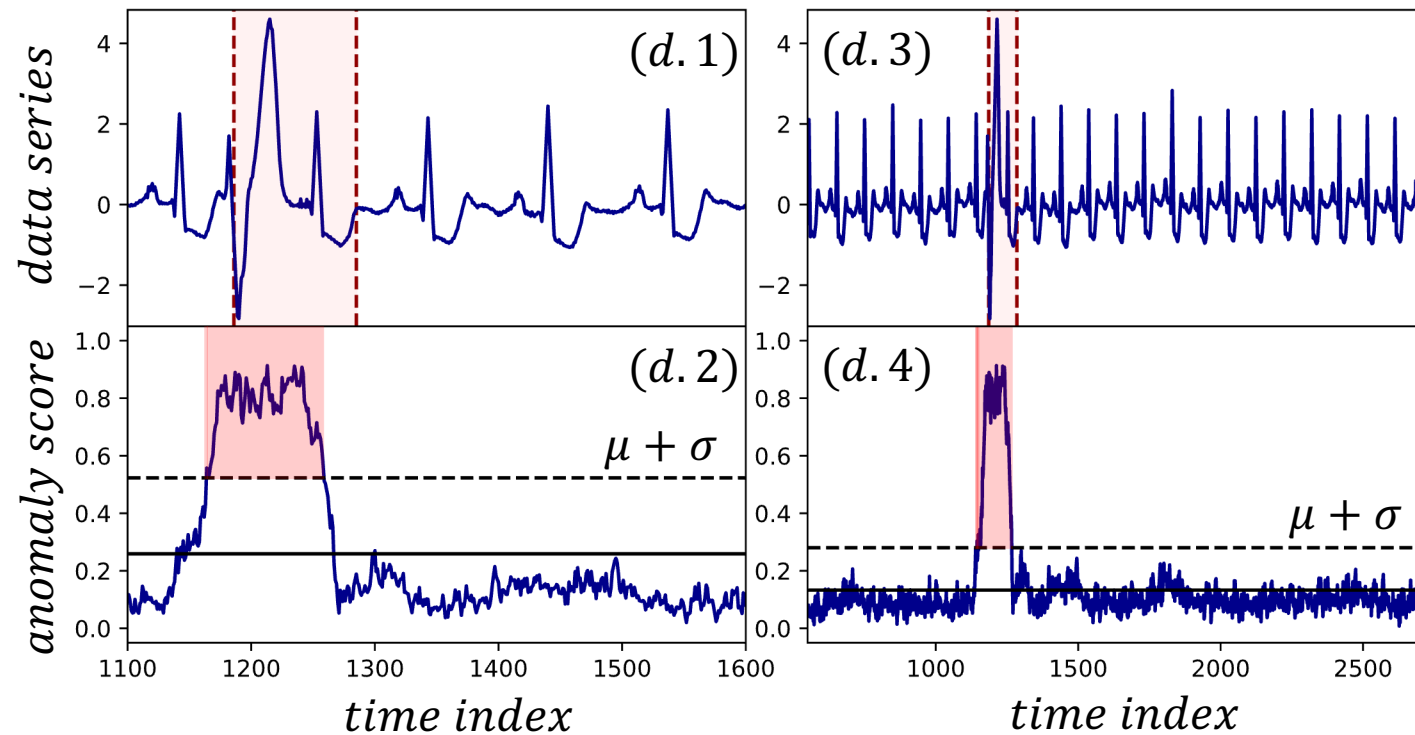
Evaluation measures: *Labeling issue*

(c) Noise impact on the accuracy measures



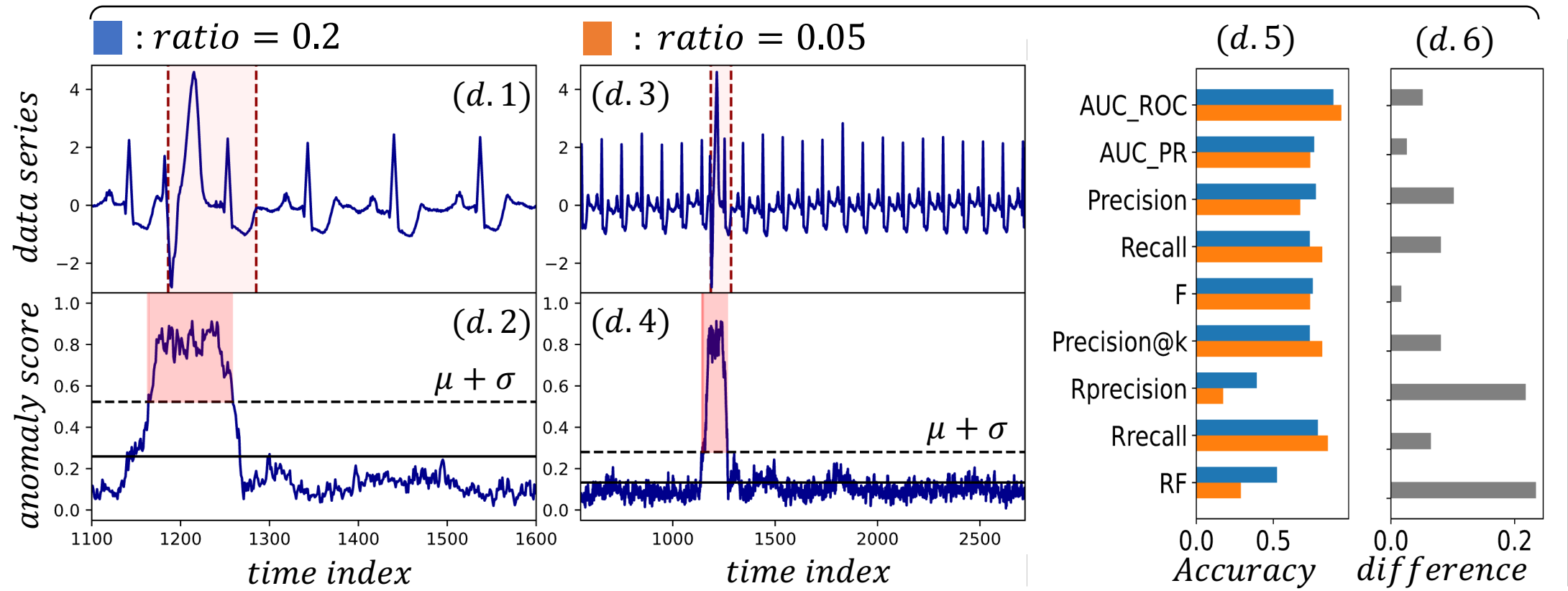
Evaluation measures: *Labeling issue*

(c) *Normal – abnormal ratio impact on accuracy measures*



Evaluation measures: *Labeling issue*

(c) Normal – abnormal ratio impact on accuracy measures



Evaluation measures: *Labeling issue*

(a) Lag impact on accuracy measures

■ : No lag ■ : $-0.25 * \ell$ lag



(b) Noise impact on accuracy measures

■ : No noise ■ : 10% of noise



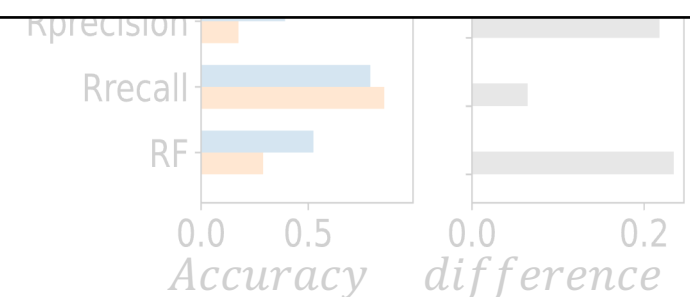
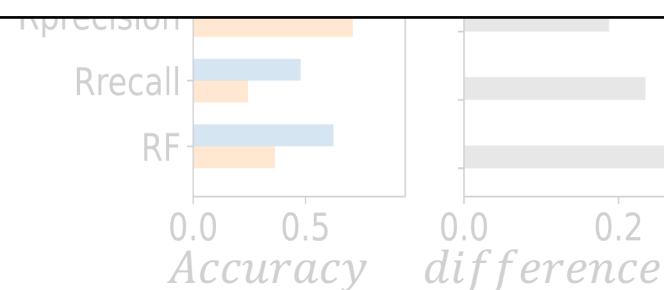
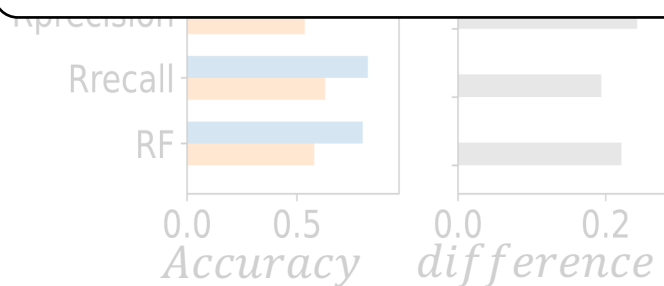
(c) Normal – abnormal ratio impact on accuracy measures

■ : ratio = 0.2 ■ : ratio = 0.05



Some evaluation measures are more robust to Noise and normal/abnormal ratio variations (especially AUC-ROC and AUC-PR).

No measures are robust to **small lags** on anomaly scores or labels.



Evaluation measures: *Labeling issue*

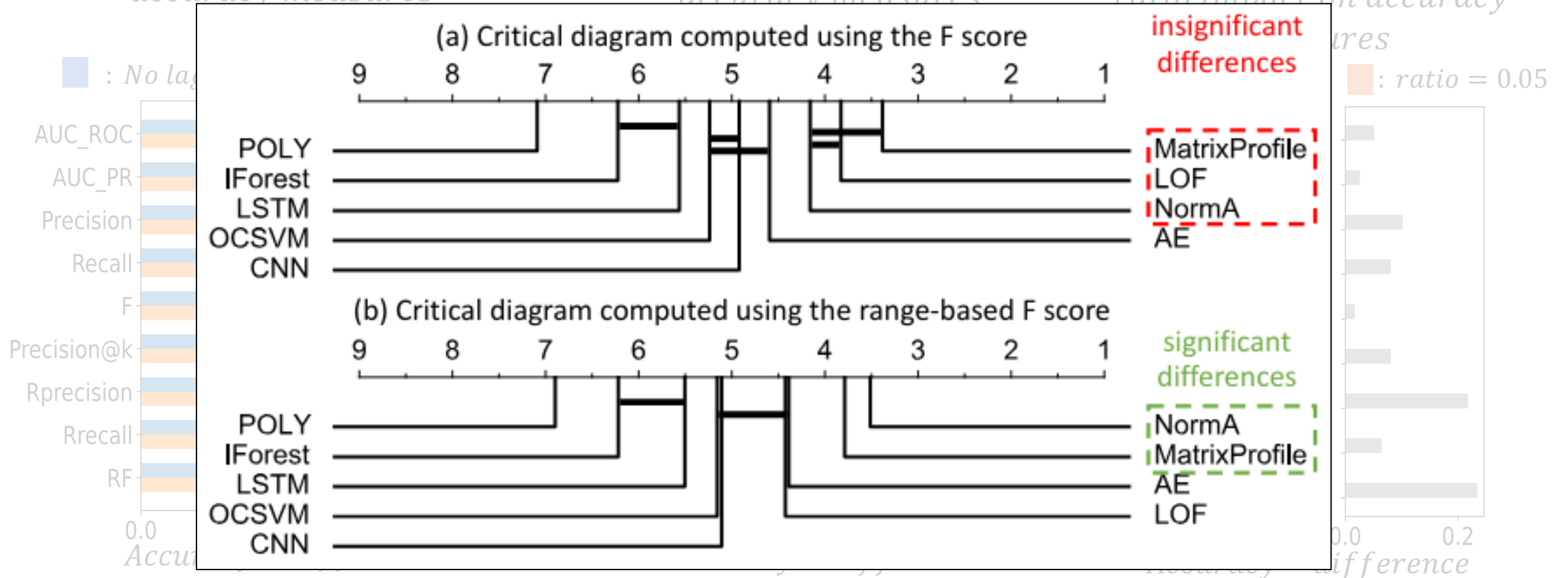
(a) *Lag impact on accuracy measures*

(b) Noise impact on accuracy measures

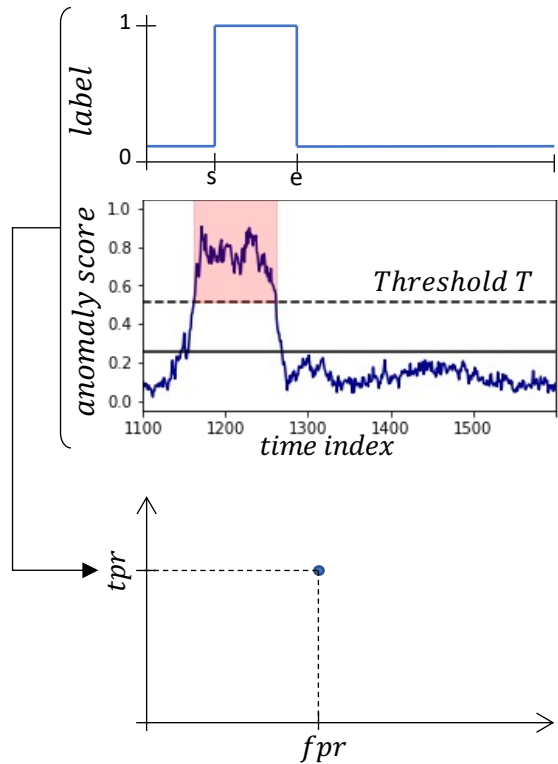
(c) Normal – abnormal ratio impact on accuracy

Ratio	Accuracy
1.0	0.85
1.5	0.85
2.0	0.85
2.5	0.85
3.0	0.85
3.5	0.85
4.0	0.85
4.5	0.85
5.0	0.85
5.5	0.85
6.0	0.85
6.5	0.85
7.0	0.85
7.5	0.85
8.0	0.85
8.5	0.85
9.0	0.85
9.5	0.85
10.0	0.85

insignificant

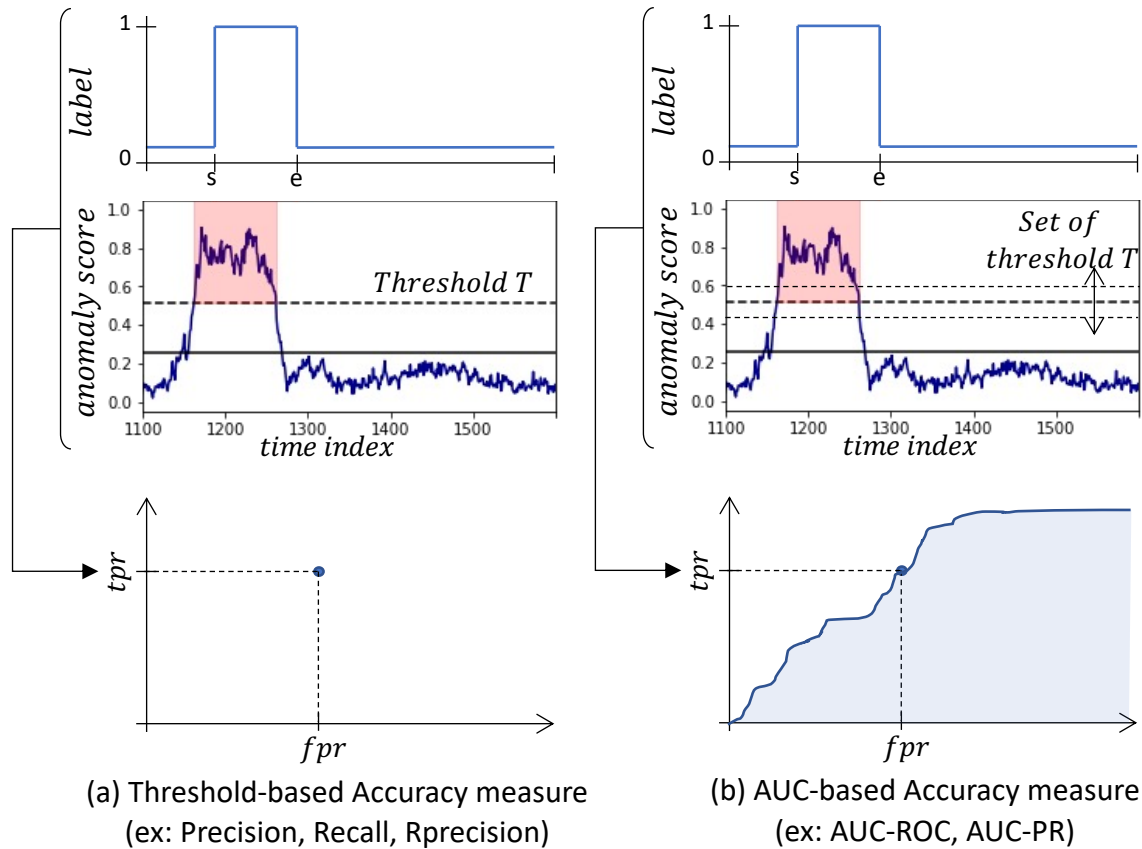


Evaluation measures: *Labeling issue*

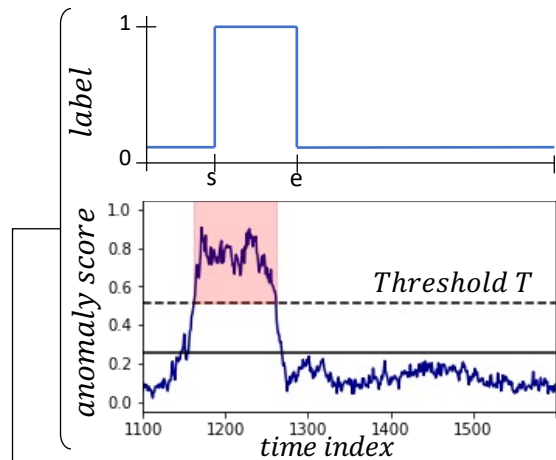


(a) Threshold-based Accuracy measure
(ex: Precision, Recall, Rprecision)

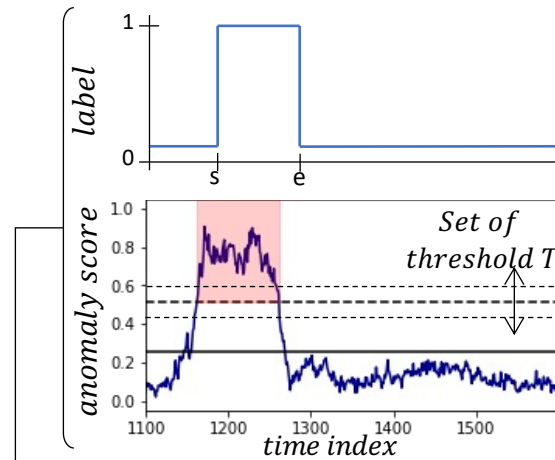
Evaluation measures: *Labeling issue*



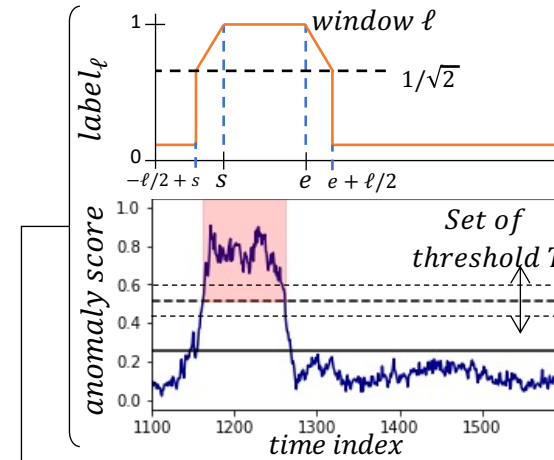
Evaluation measures: *Labeling issue*



(a) Threshold-based Accuracy measure
(ex: Precision, Recall, Rprecision)

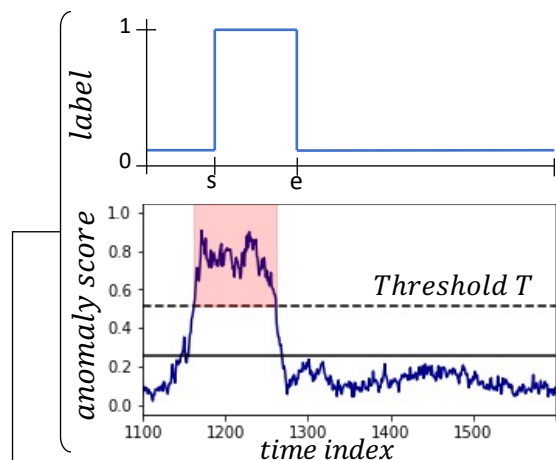


(b) AUC-based Accuracy measure
(ex: AUC-ROC, AUC-PR)

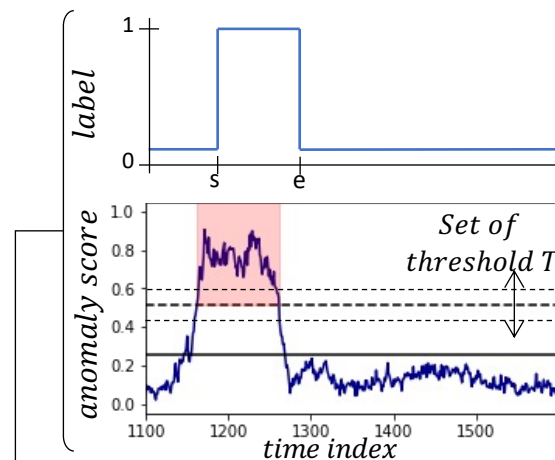


(c) R-AUC-based Accuracy measure
(ex: R-AUC-ROC, R-AUC-PR)

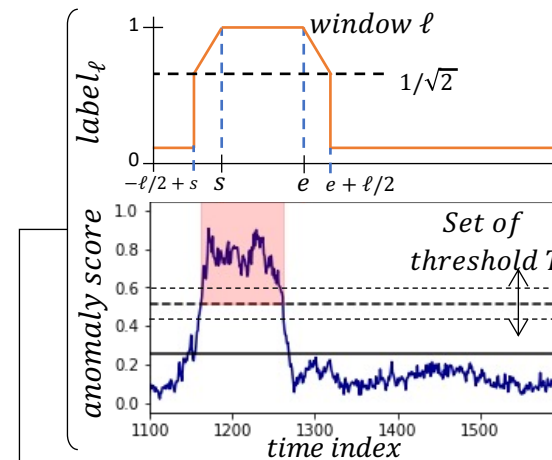
Evaluation measures: *Labeling issue*



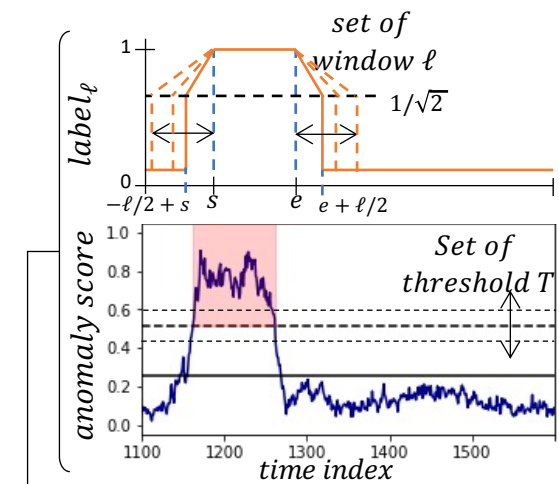
(a) Threshold-based Accuracy measure
(ex: Precision, Recall, Rprecision)



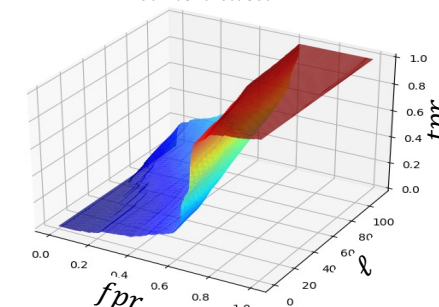
(b) AUC-based Accuracy measure
(ex: AUC-ROC, AUC-PR)



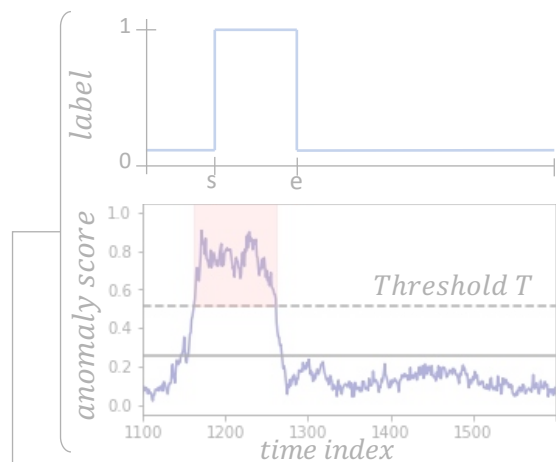
(c) R-AUC-based Accuracy measure
(ex: R-AUC-ROC, R-AUC-PR)



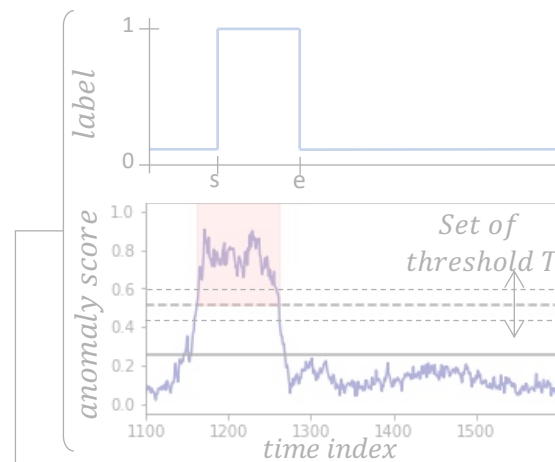
(d) VUS-based Accuracy measure
(ex: VUS-ROC, VUS-PR)



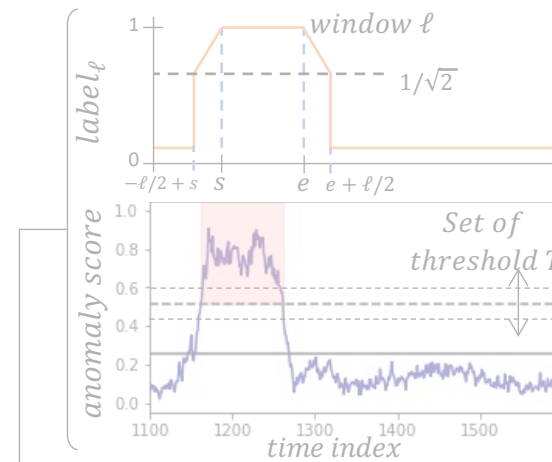
Evaluation measures: *Labeling issue*



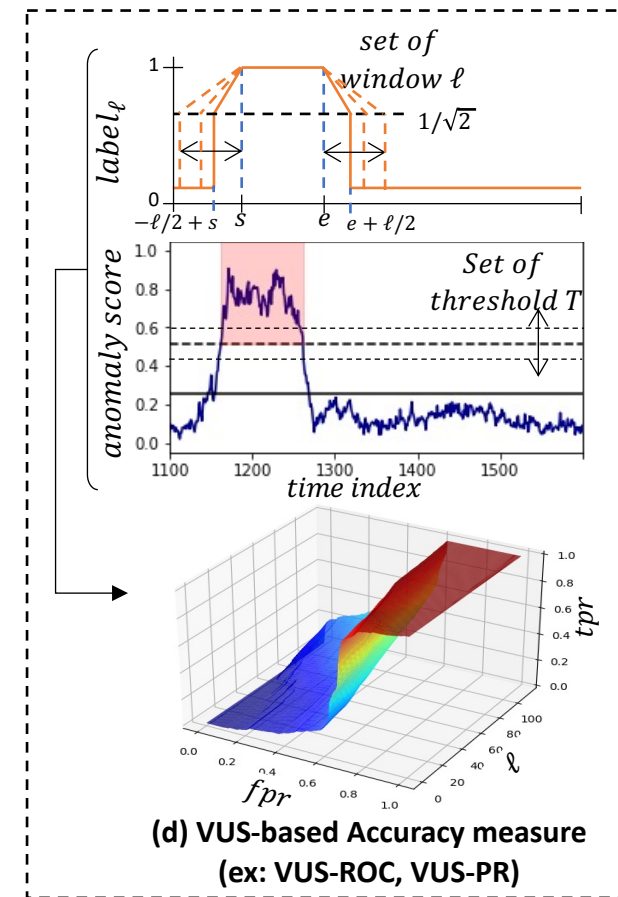
(a) Threshold-based Accuracy measure
(ex: Precision, Recall, Rprecision)



(b) AUC-based Accuracy measure
(ex: AUC-ROC, AUC-PR)

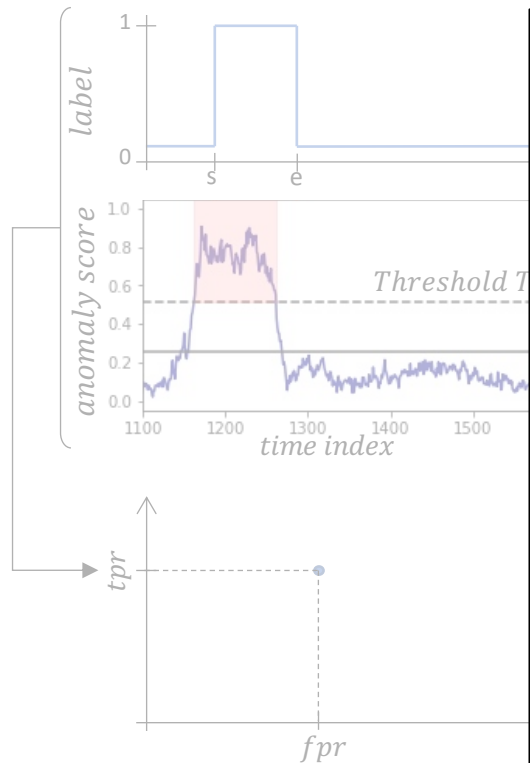


(c) R-AUC-based Accuracy measure
(ex: R-AUC-ROC, R-AUC-PR)



(d) VUS-based Accuracy measure
(ex: VUS-ROC, VUS-PR)

Evaluation measures: *Labeling issue*



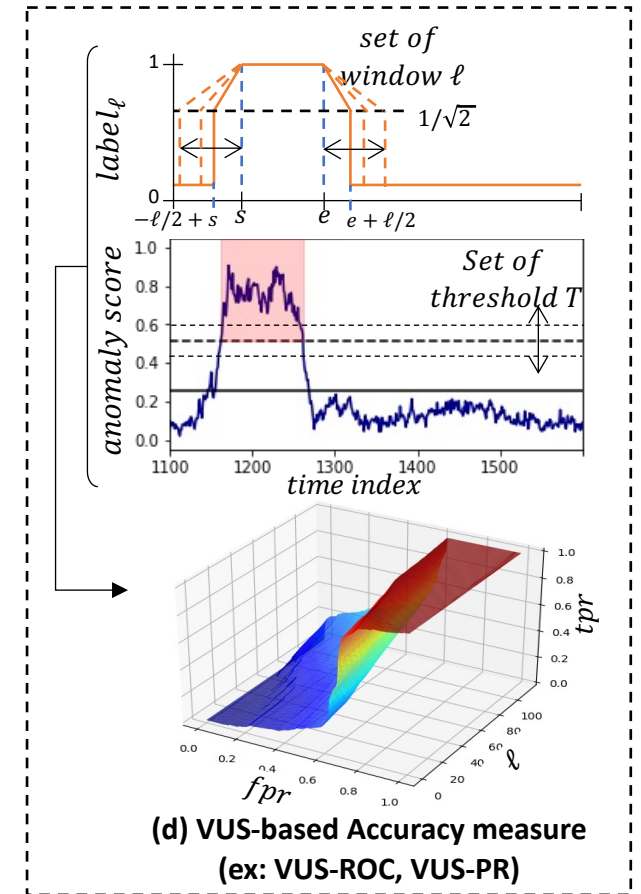
(a) Threshold-based Accuracy measure
(ex: Precision, Recall, Rprecision)

Summary:

- **Volume Under the Surface [22] (VUS):**
- Modify the labels with buffer regions at the beginning and at the end of an anomaly
- We vary the buffer size (as well as the threshold) and we obtain a surface
- We use the volume under the surface (VUS) as accuracy



(b) AUC-based Accuracy measure
(ex: R-AUC-ROC, R-AUC-PR)



(d) VUS-based Accuracy measure
(ex: VUS-ROC, VUS-PR)

Evaluation measures: *VUS*

How is it computed?

$$AUC-ROC = \frac{1}{2} \sum_{k=1}^N \Delta_{TPR}^k * \Delta_{FPR}^k$$

$$\text{with: } \begin{cases} \Delta_{FPR}^k &= FPR(Th_k) - FPR(Th_{k-1}) \\ \Delta_{TPR}^k &= TPR(Th_{k-1}) + TPR(Th_k) \end{cases}$$

$$AUC-PR = \frac{1}{2} \sum_{k=1}^N \Delta_{Precision}^k * \Delta_{Recall}^k$$

$$\text{with: } \begin{cases} \Delta_{Recall}^k &= Recall(Th_k) - Recall(Th_{k-1}) \\ \Delta_{Precision}^k &= Precision(Th_{k-1}) + Precision(Th_k) \end{cases}$$

Evaluation measures: *VUS*

How is it computed?

$$AUC-ROC = \frac{1}{2} \sum_{k=1}^N \Delta_{TPR}^k * \Delta_{FPR}^k$$

$$\text{with: } \begin{cases} \Delta_{FPR}^k &= FPR(Th_k) - FPR(Th_{k-1}) \\ \Delta_{TPR}^k &= TPR(Th_{k-1}) + TPR(Th_k) \end{cases}$$

$$AUC-PR = \frac{1}{2} \sum_{k=1}^N \Delta_{Precision}^k * \Delta_{Recall}^k$$

$$\text{with: } \begin{cases} \Delta_{Recall}^k &= Recall(Th_k) - Recall(Th_{k-1}) \\ \Delta_{Precision}^k &= Precision(Th_{k-1}) + Precision(Th_k) \end{cases}$$

$$VUS-ROC = \frac{1}{4} \sum_{w=1}^L \sum_{k=1}^N \Delta^{(k,w)} * \Delta^w$$

$$\text{with: } \begin{cases} \Delta^{(k,w)} &= \Delta_{TPR_{\ell_w}}^k * \Delta_{FPR_{\ell_w}}^k + \Delta_{TPR_{\ell_{w-1}}}^k * \Delta_{FPR_{\ell_{w-1}}}^k \\ \Delta_{FPR_{\ell_w}}^k &= FPR_{\ell_w}(Th_k) - FPR_{\ell_w}(Th_{k-1}) \\ \Delta_{TPR_{\ell_w}}^k &= TPR_{\ell_w}(Th_{k-1}) + TPR_{\ell_w}(Th_k) \\ \Delta^w &= |\ell_w - \ell_{w-1}| \end{cases}$$

$$VUS-PR = \frac{1}{4} \sum_{w=1}^L \sum_{k=1}^N \Delta^{(k,w)} * \Delta^w$$

$$\text{with: } \begin{cases} \Delta^{(k,w)} &= \Delta_{Pr_{\ell_w}}^k * \Delta_{Re_{\ell_w}}^k + \Delta_{Pr_{\ell_{w-1}}}^k * \Delta_{Re_{\ell_{w-1}}}^k \\ \Delta_{Re_{\ell_w}}^k &= Recall_{\ell_w}(Th_k) - Recall_{\ell_w}(Th_{k-1}) \\ \Delta_{Pr_{\ell_w}}^k &= Precision_{\ell_w}(Th_{k-1}) + Precision_{\ell_w}(Th_k) \\ \Delta^w &= |\ell_w - \ell_{w-1}| \end{cases}$$

Evaluation measures: *VUS*

How is it computed?

$$AUC-ROC = \frac{1}{2} \sum_{k=1}^N \Delta_{TPR}^k * \Delta_{FPR}^k$$

$$(\Delta_{FPR}^k = FPR(Th_k) - FPR(Th_{k-1}))$$

Time Complexity: $O(NT)$

With:

- T : the time series length
- N : the number of thresholds

$$\text{with: } \begin{cases} \Delta_{Recall}^k &= Recall(Th_k) - Recall(Th_{k-1}) \\ \Delta_{Precision}^k &= Precision(Th_{k-1}) + Precision(Th_k) \end{cases}$$

$$VUS-ROC = \frac{1}{4} \sum_{w=1}^L \sum_{k=1}^N \Delta^{(k,w)} * \Delta^w$$

$$\text{with: } \begin{cases} \Delta^{(k,w)} &= \Delta_{TPR_{\ell_w}}^k * \Delta_{FPR_{\ell_w}}^k + \Delta_{TPR_{\ell_{w-1}}}^k * \Delta_{FPR_{\ell_{w-1}}}^k \\ \Delta_{FPR_{\ell_w}}^k &= FPR_{\ell_w}(Th_k) - FPR_{\ell_w}(Th_{k-1}) \\ \Delta_{TPR_{\ell_w}}^k &= TPR_{\ell_w}(Th_{k-1}) + TPR_{\ell_w}(Th_k) \\ \Delta^w &= |\ell_w - \ell_{w-1}| \end{cases}$$

$$VUS-PR = \frac{1}{4} \sum_{w=1}^L \sum_{k=1}^N \Delta^{(k,w)} * \Delta^w$$

$$\text{with: } \begin{cases} \Delta^{(k,w)} &= \Delta_{Pr_{\ell_w}}^k * \Delta_{Re_{\ell_w}}^k + \Delta_{Pr_{\ell_{w-1}}}^k * \Delta_{Re_{\ell_{w-1}}}^k \\ \Delta_{Re_{\ell_w}}^k &= Recall_{\ell_w}(Th_k) - Recall_{\ell_w}(Th_{k-1}) \\ \Delta_{Pr_{\ell_w}}^k &= Precision_{\ell_w}(Th_{k-1}) + Precision_{\ell_w}(Th_k) \\ \Delta^w &= |\ell_w - \ell_{w-1}| \end{cases}$$

Evaluation measures: *VUS*

How is it computed?

Time Complexity: $O(NT)$

With:

- T : the time series length
- N : the number of thresholds

$$\text{with: } \begin{cases} \Delta_{Recall}^k &= Recall(Th_k) - Recall(Th_{k-1}) \\ \Delta_{Precision}^k &= Precision(Th_{k-1}) + Precision(Th_k) \end{cases}$$

$$VUS-ROC = \frac{1}{4} \sum_{w=1}^L \sum_{k=1}^N \Delta^{(k,w)} * \Delta^w$$

$$\left(\Delta^{(k,w)} = \Delta_{TPR_{\ell_w}}^k * \Delta_{FPR_{\ell_w}}^k + \Delta_{TPR_{\ell_{w-1}}}^k * \Delta_{FPR_{\ell_{w-1}}}^k \right)$$

Time Complexity: $O(NLT)$

With:

- T : the time series length
- N : the number of thresholds
- L : the number of buffer lengths

$$\text{with: } \begin{cases} \Delta_{Ret_w}^k &= Recall_{\ell_w}(Th_k) - Recall_{\ell_w}(Th_{k-1}) \\ \Delta_{Pr_{\ell_w}}^k &= Precision_{\ell_w}(Th_{k-1}) + Precision_{\ell_w}(Th_k) \\ \Delta^w &= |\ell_w - \ell_{w-1}| \end{cases}$$

Evaluation measures: *VUS*

How is it computed?

$$AUC-ROC = \frac{1}{2} \sum_{k=1}^N \Delta_{TPR}^k * \Delta_{FPR}^k$$

Time Complexity: $O(NT)$

With:

- T : the time series length
- N : the number of thresholds

with:
$$\begin{cases} \Delta_{Recall}^k = Recall(Th_k) - Recall(Th_{k-1}) \\ \Delta_{Precision}^k = Precision(Th_{k-1}) + Precision(Th_k) \end{cases}$$

$$VUS-ROC = \frac{1}{4} \sum_{w=1}^L \sum_{k=1}^N \Delta^{(k,w)} * \Delta^w$$

Time Complexity: $O(NLT)$

With:

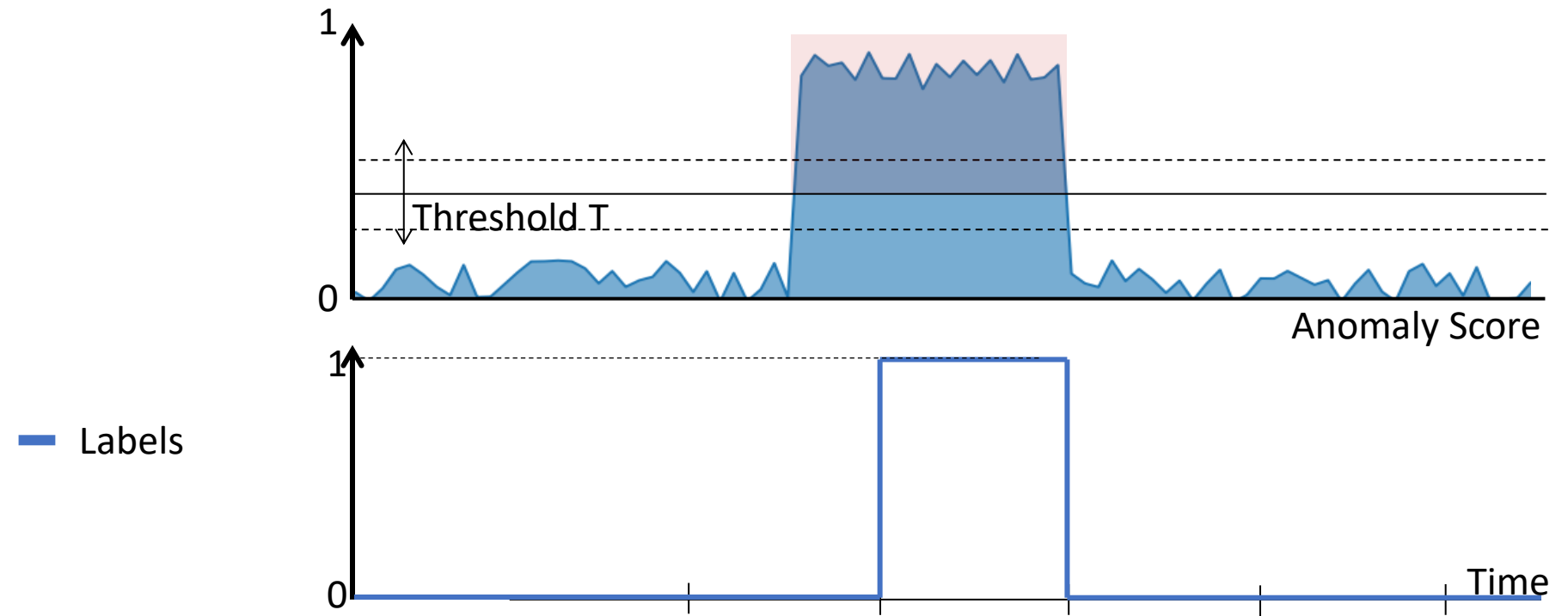
- T : the time series length
- N : the number of thresholds
- L : the number of buffer lengths

with:
$$\begin{cases} \Delta_{Rec_{\ell_w}}^k = Recall_{\ell_w}(Th_k) - Recall_{\ell_w}(Th_{k-1}) \\ \Delta_{Prec_{\ell_w}}^k = Precision_{\ell_w}(Th_{k-1}) + Precision_{\ell_w}(Th_k) \end{cases}$$

VUS is significantly slower to compute, complicating its usage in practice

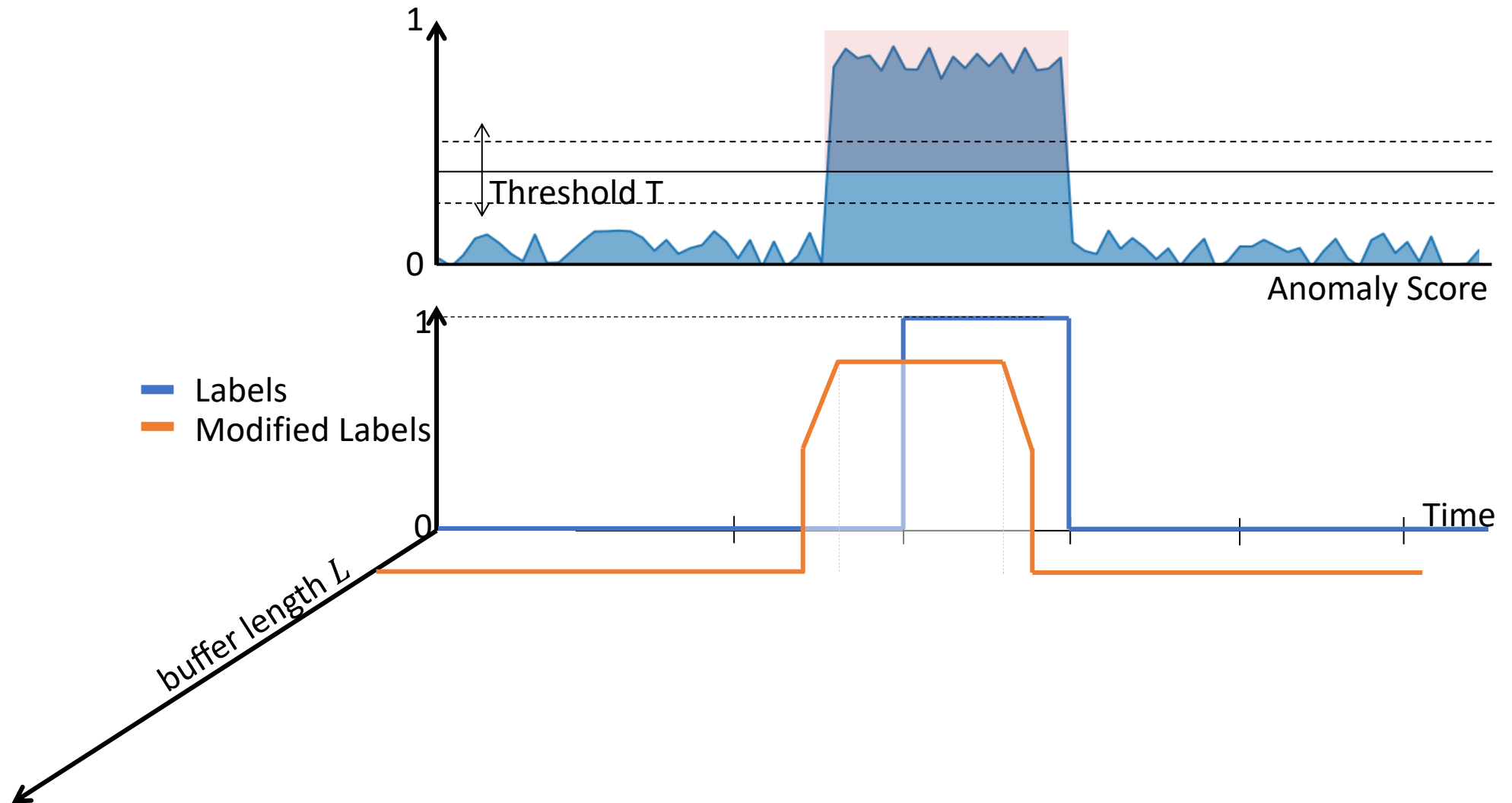
Evaluation measures: *VUS*

A solution?



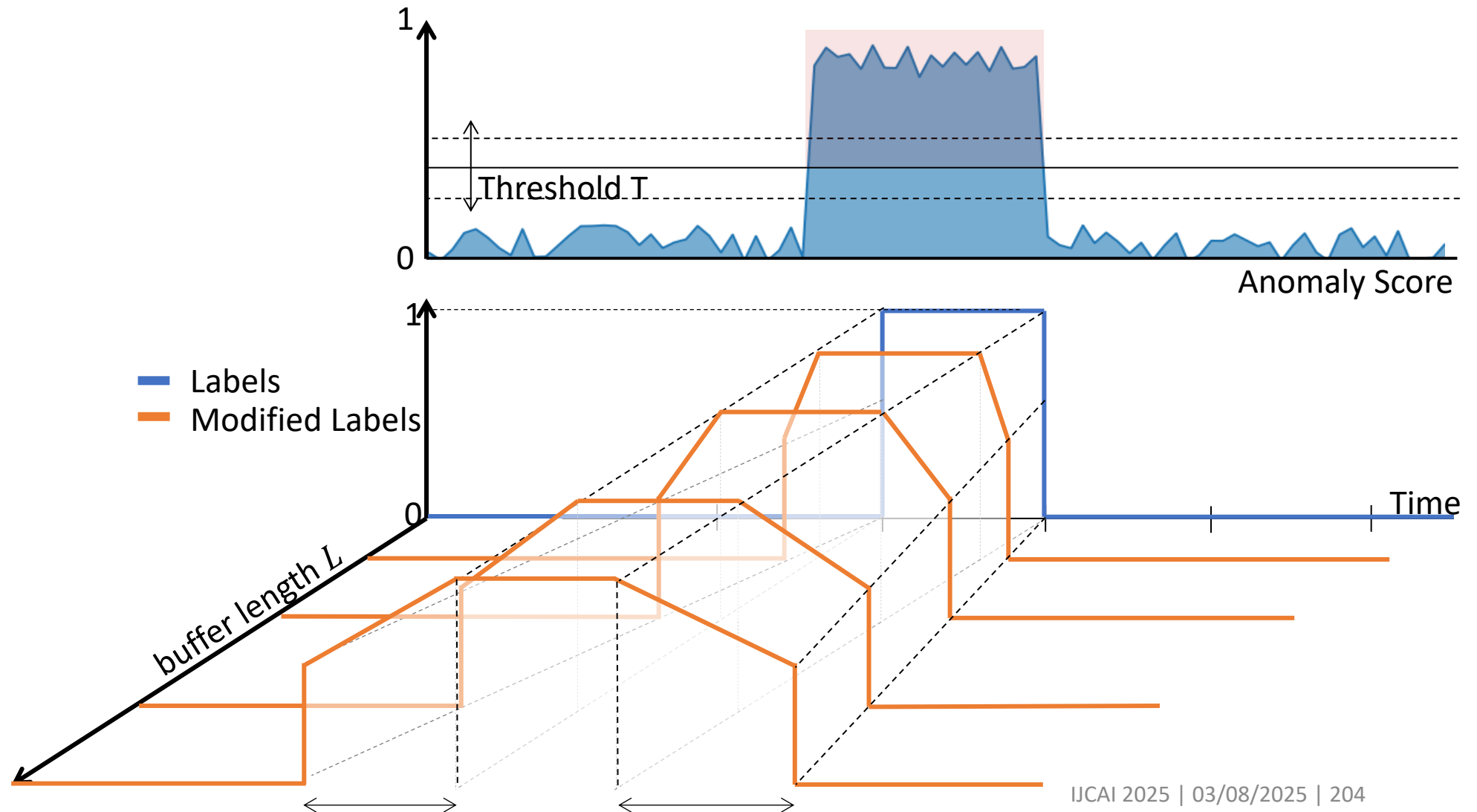
Evaluation measures: *VUS*

A solution?



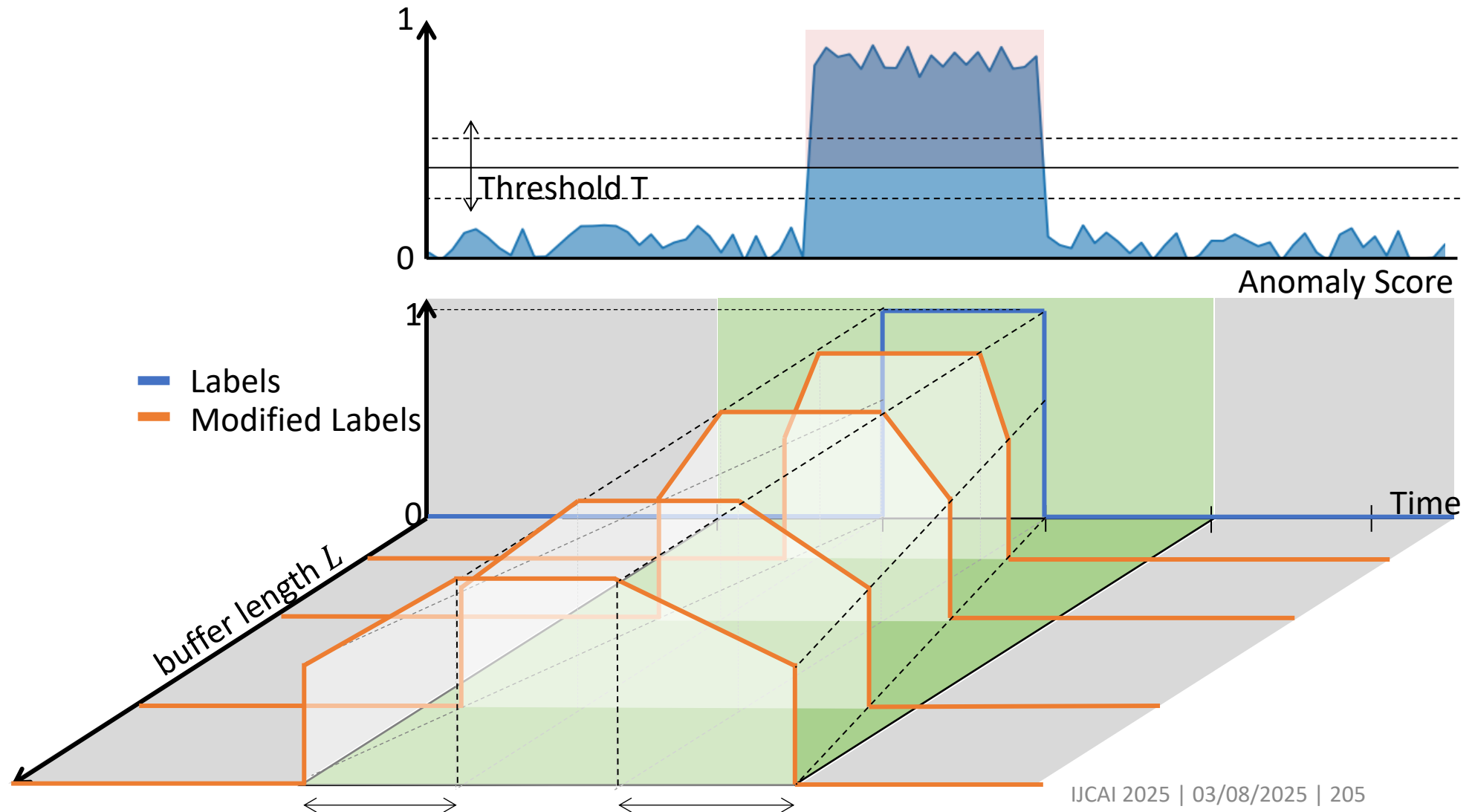
Evaluation measures: *VUS*

A solution?



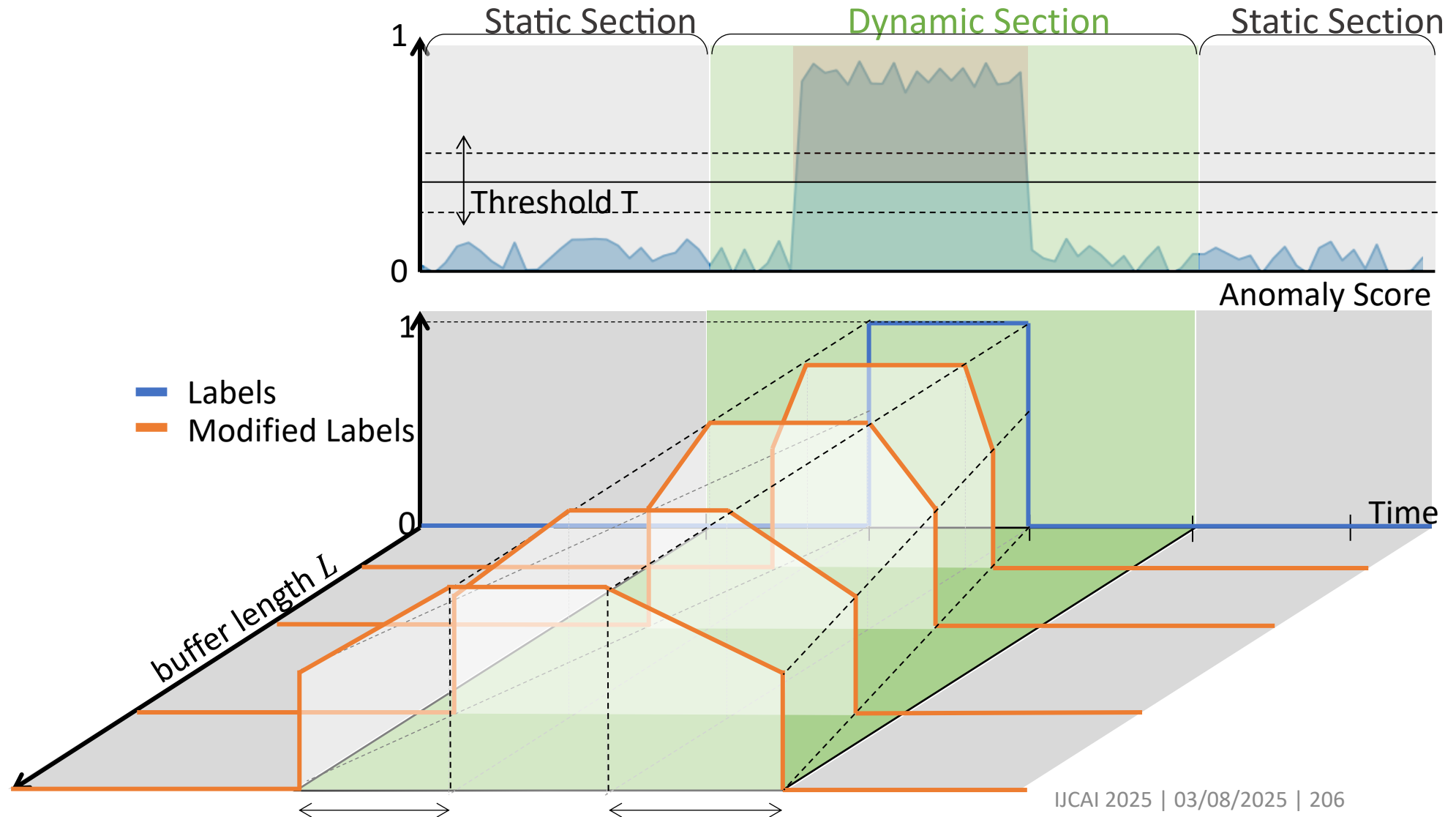
Evaluation measures: *VUS*

A solution?



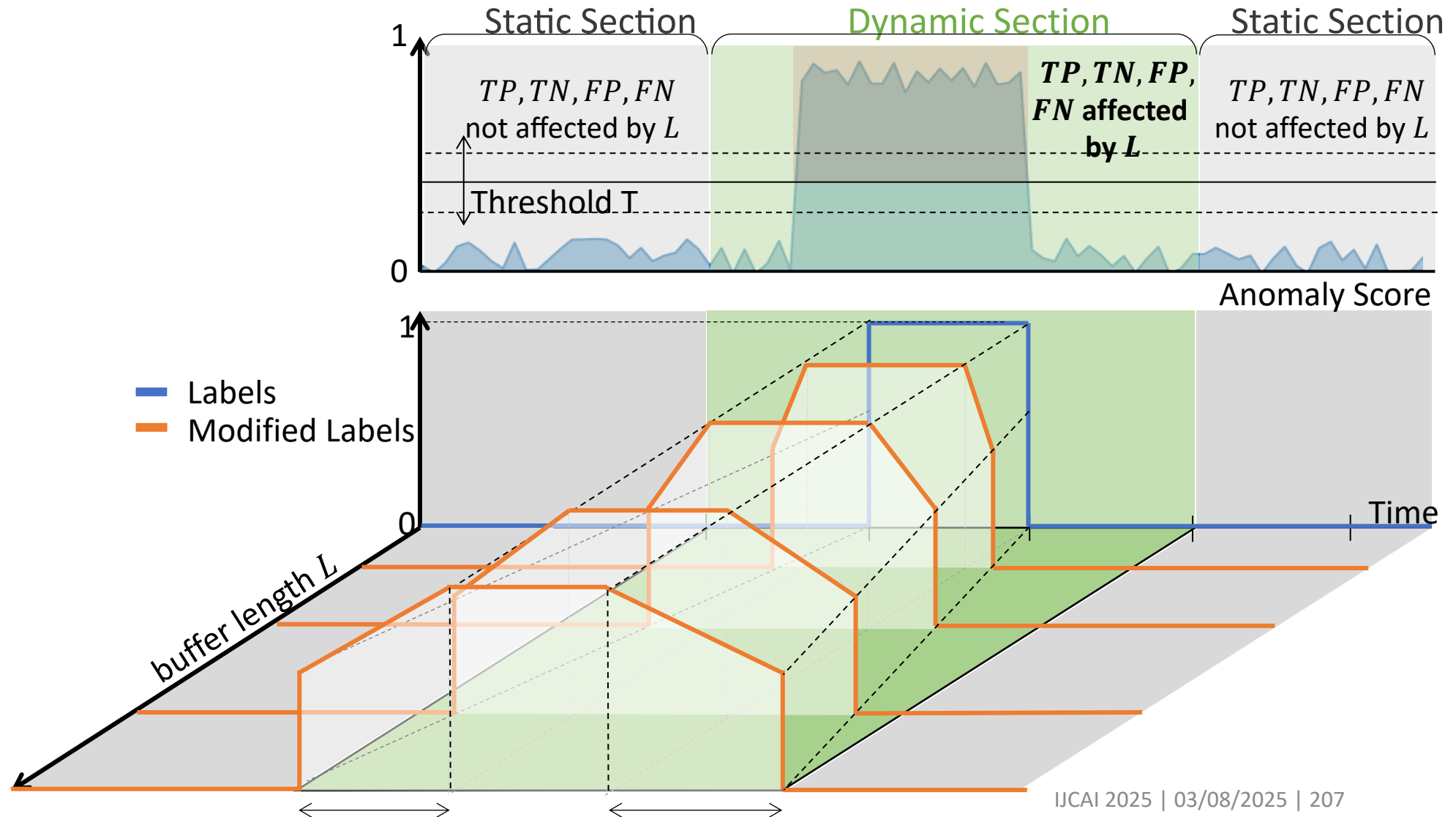
Evaluation measures: *VUS*

A solution?



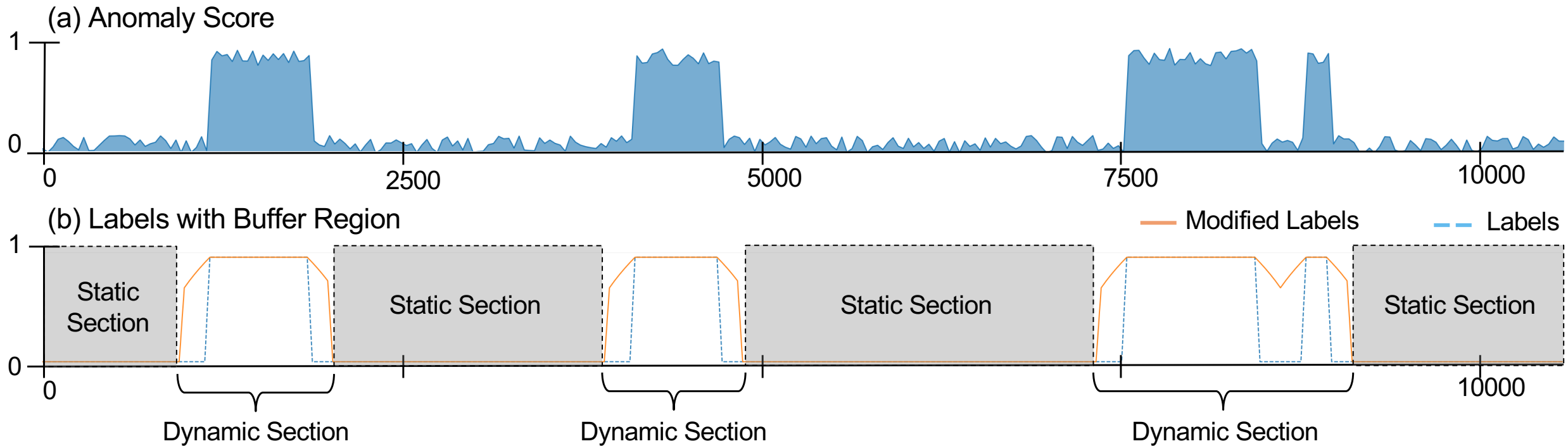
Evaluation measures: *VUS*

A solution?



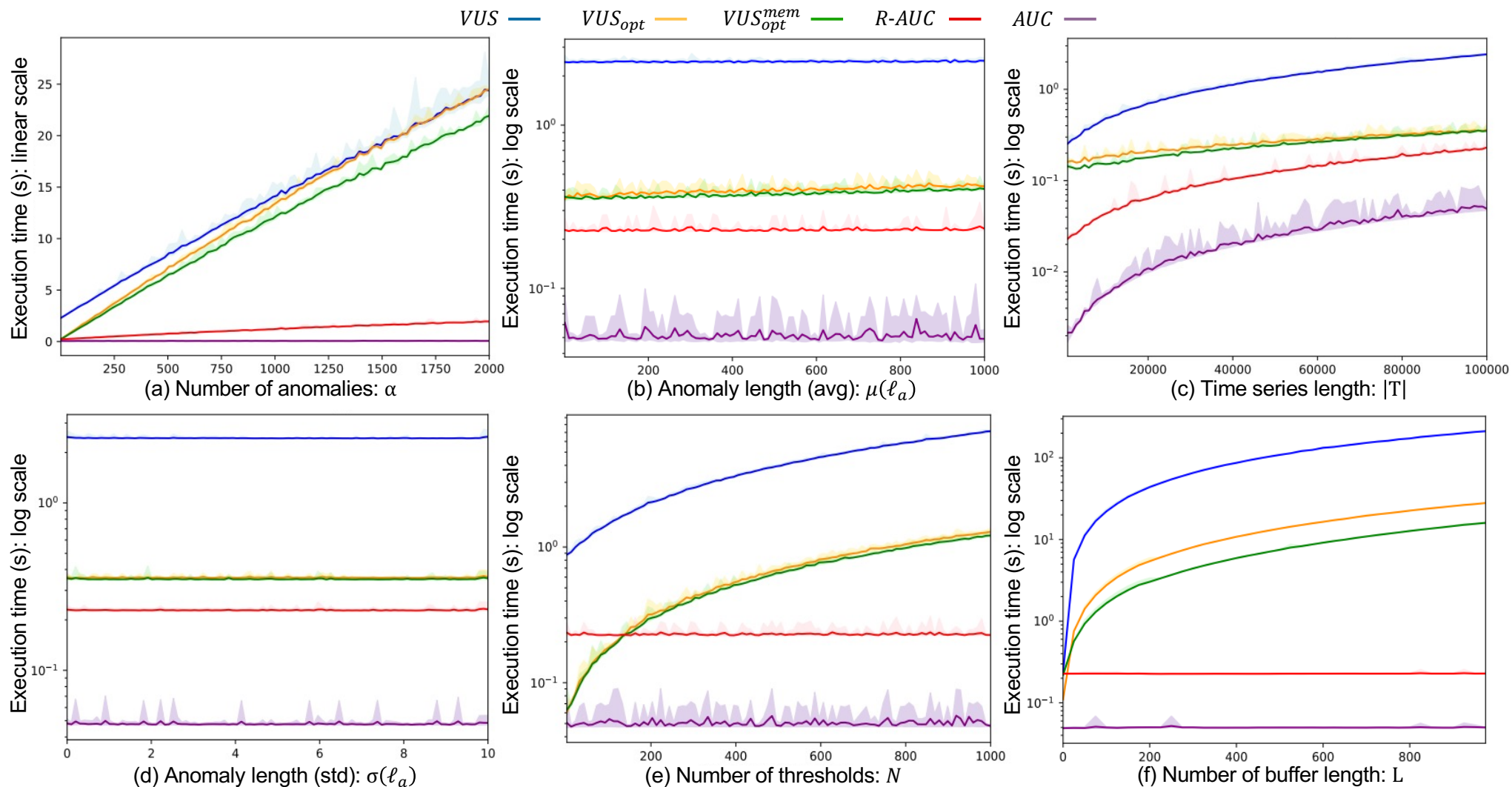
Evaluation measures: *VUS*

A solution?



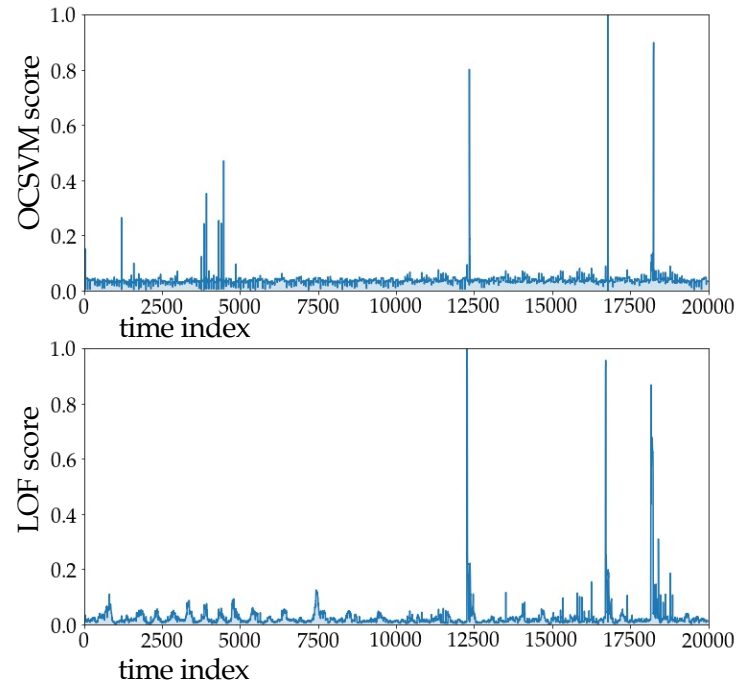
Evaluation measures: VUS

Execution time evaluation on synthetic data to measure the influence of different parameters

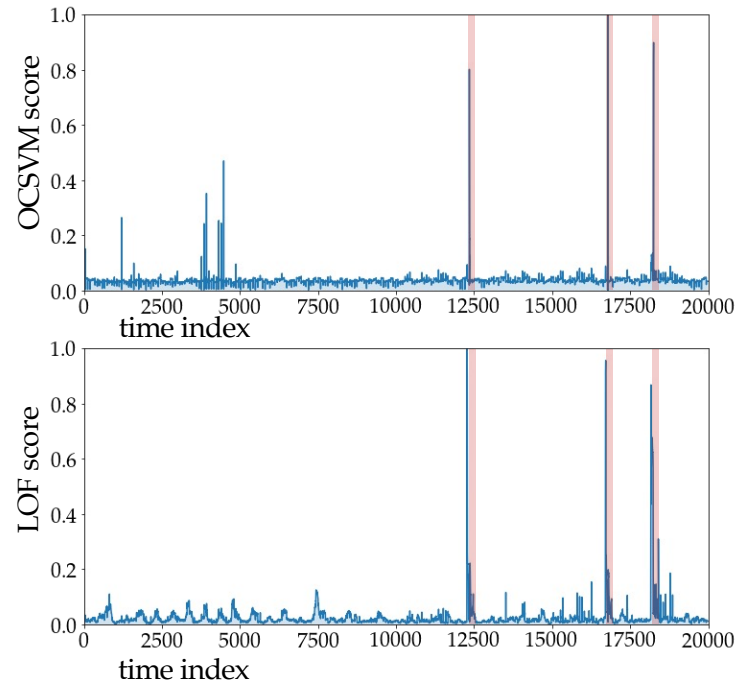


Evaluating measures: A Hard question

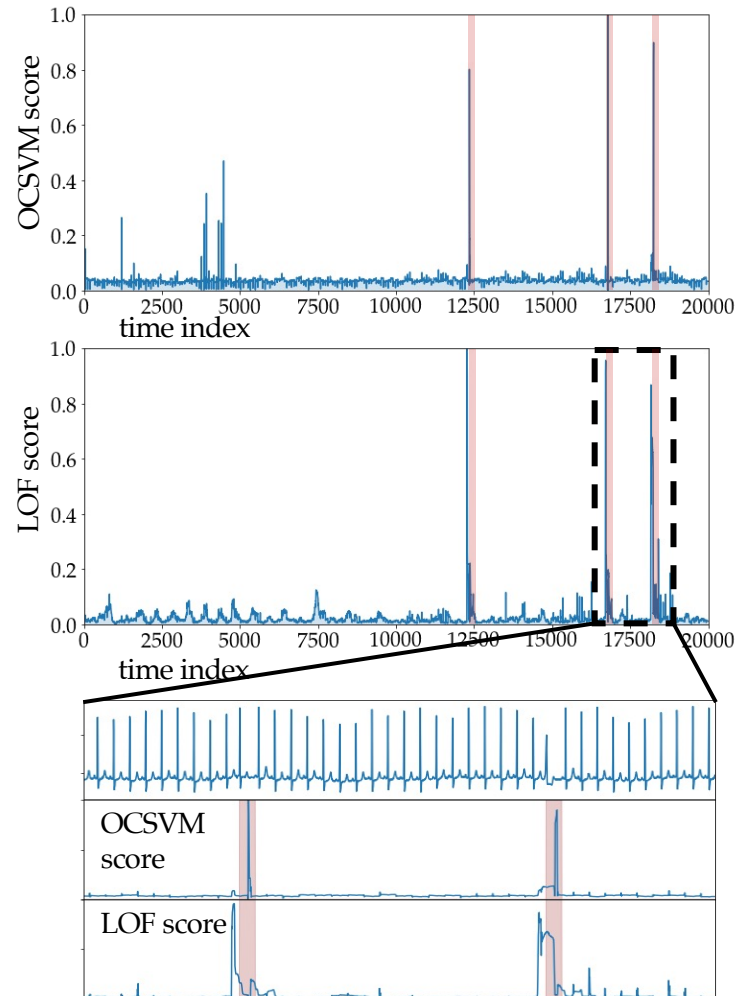
Evaluating measures: A Hard question



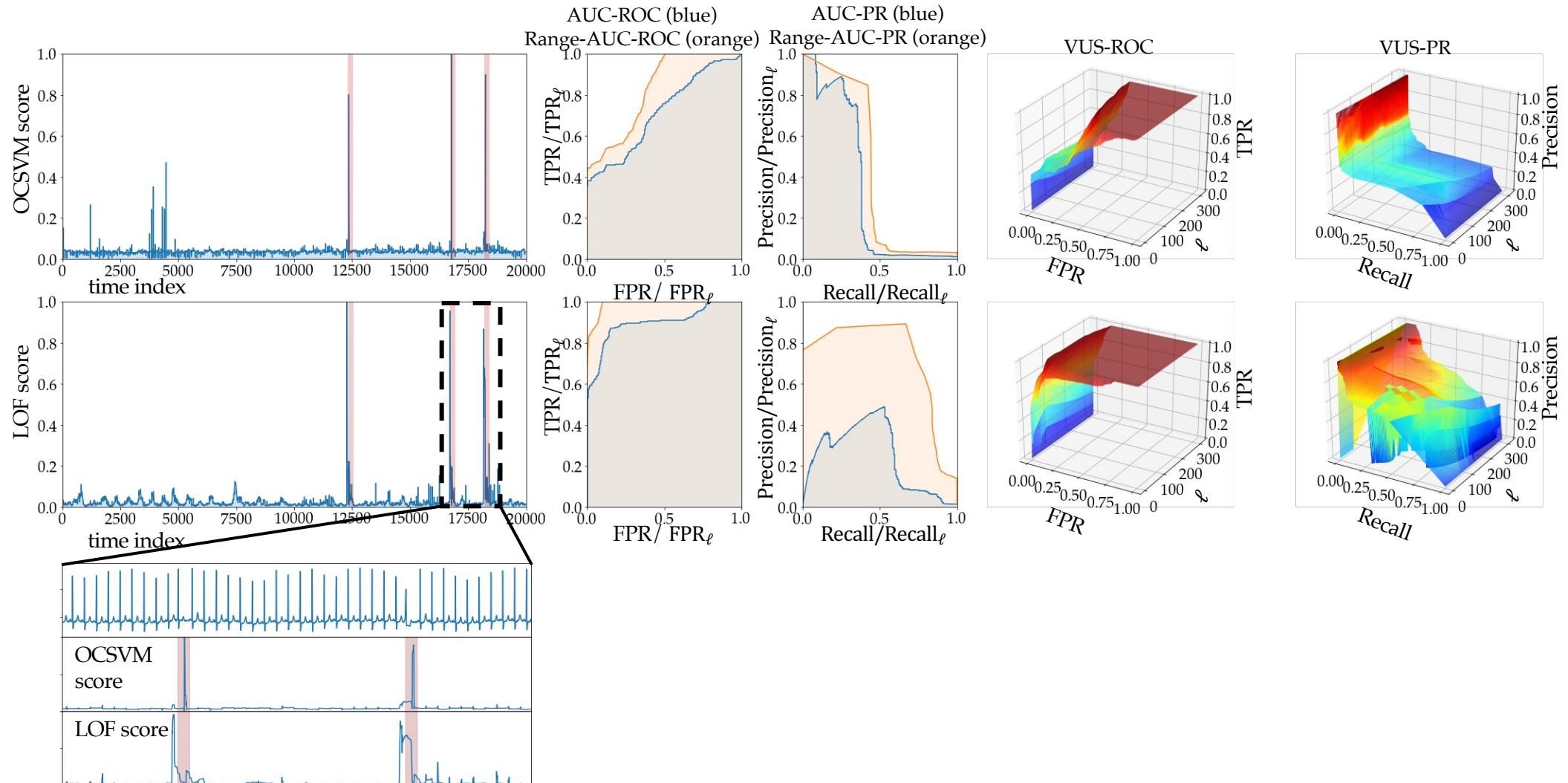
Evaluating measures: A Hard question



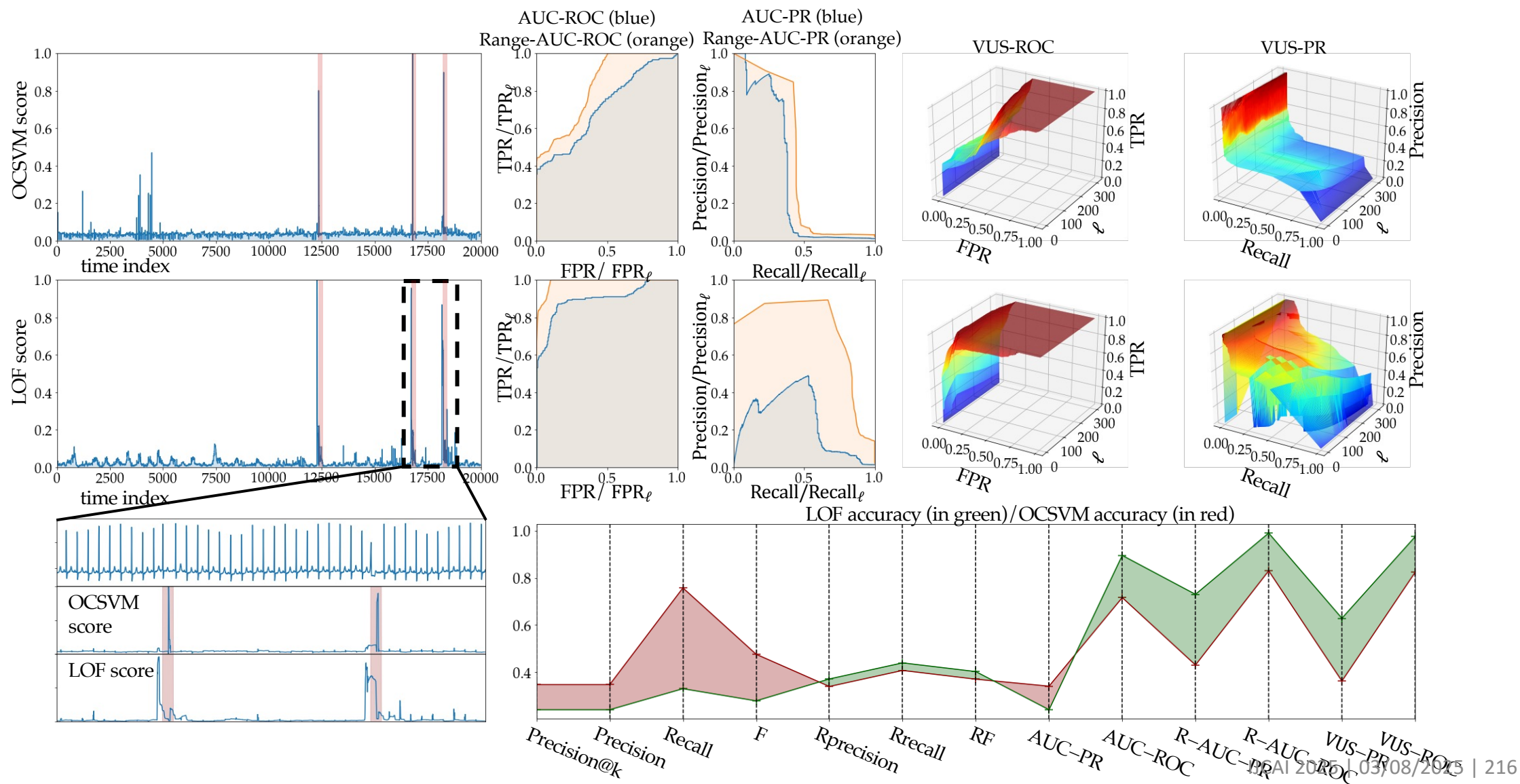
Evaluating measures: A Hard question



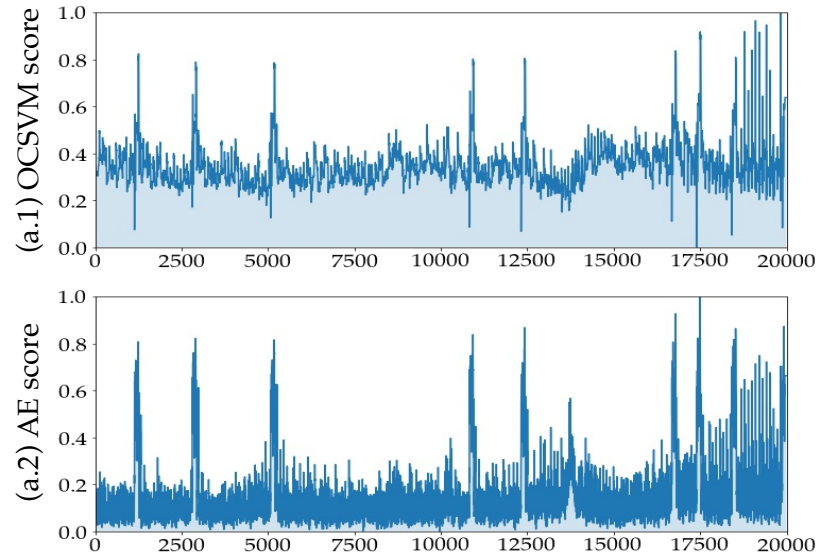
Evaluating measures: A Hard question



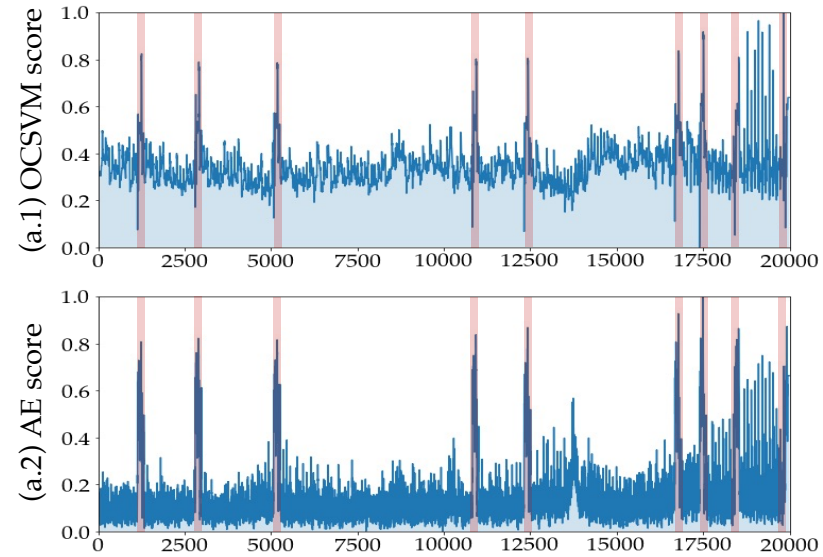
Evaluating measures: A Hard question



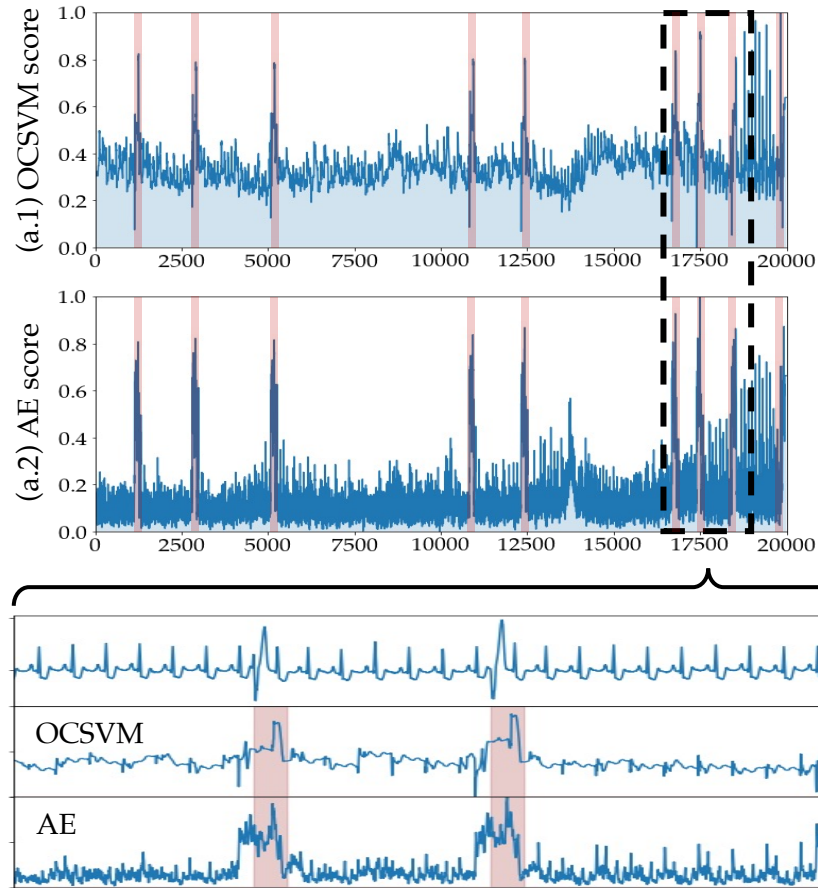
Evaluating measures: A Hard question



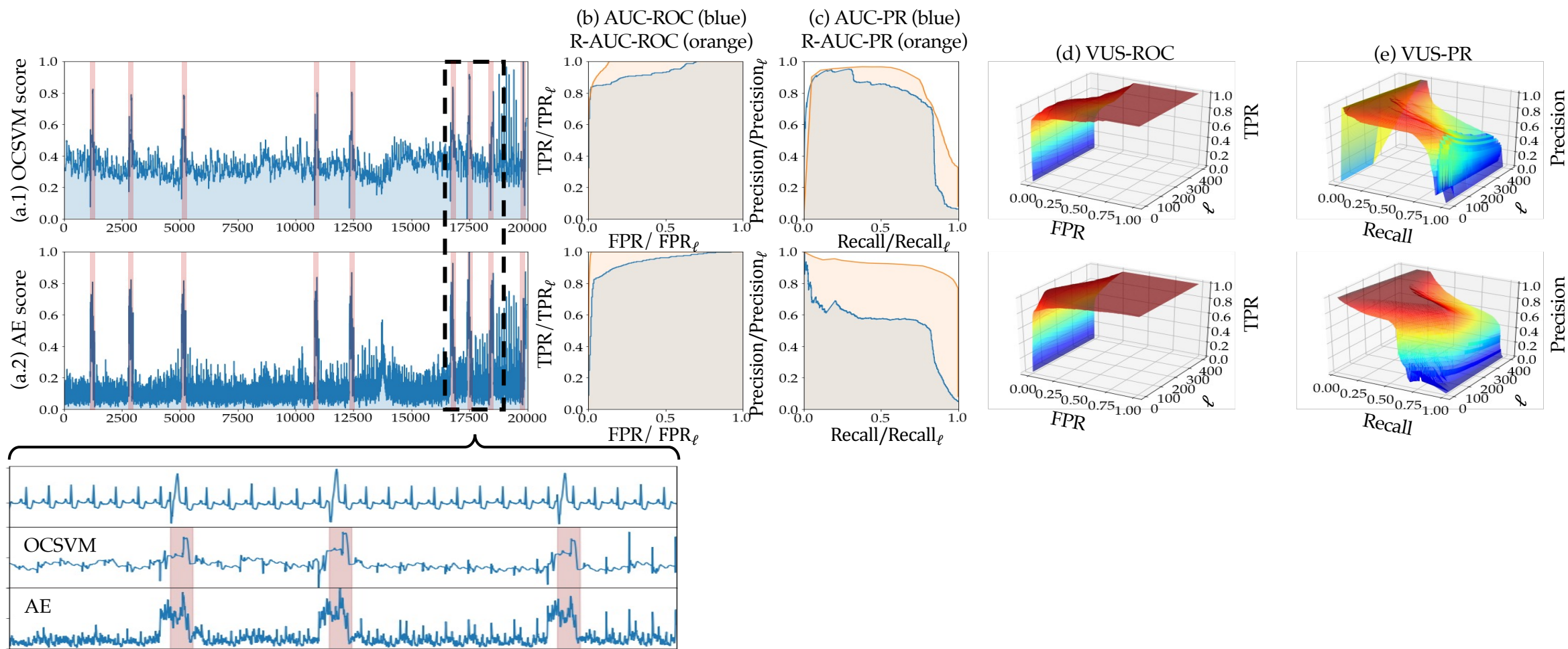
Evaluating measures: A Hard question



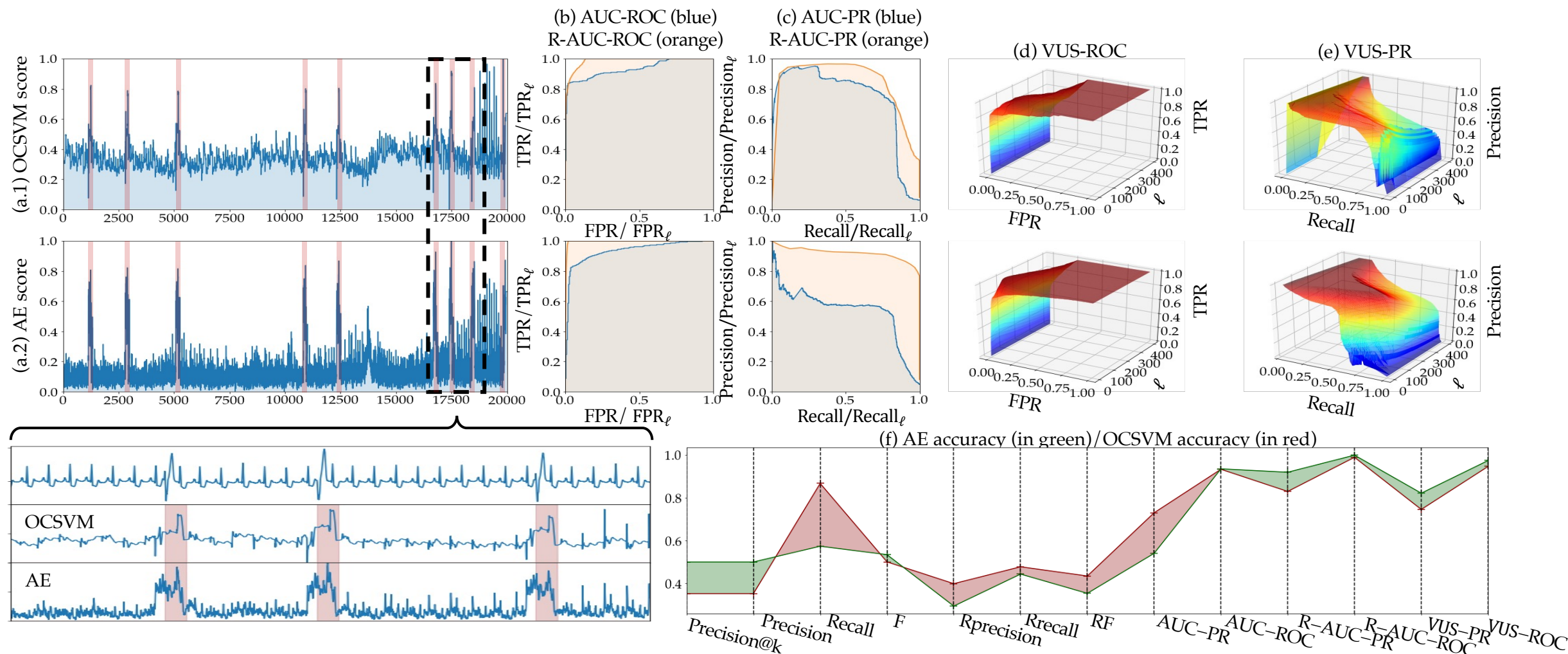
Evaluating measures: A Hard question



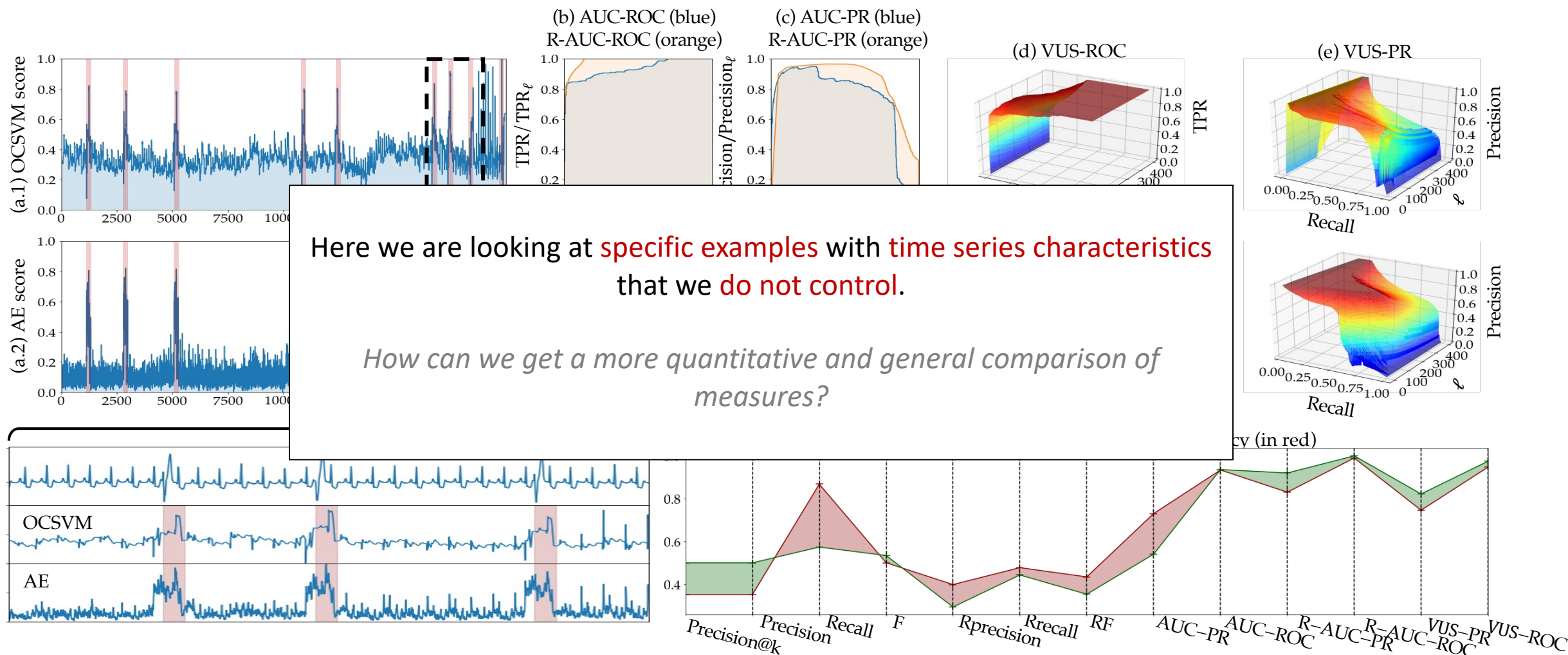
Evaluating measures: A Hard question



Evaluating measures: A Hard question



Evaluating measures: A Hard question



Evaluating measures: A Hard question

Synthetic Evaluation [29]

Evaluating measures: A Hard question

Synthetic Evaluation [29]

	$P@3$	$best_{PW} f_{\beta}$	AUC_{ROC}	AUC_{PR}	VUS^4_{ROC}	VUS^4_{PR}
	0.67	0.75	0.96	0.76	0.96	0.83
	0.67	0.75	0.96	0.76	0.96	0.83
	$P@11$	$best_{PW} f_{\beta}$	AUC_{ROC}	AUC_{PR}	VUS^4_{ROC}	VUS^4_{PR}
	0.41	0.78	0.8	0.78	0.61	0.74
	0.41	0.58	0.6	0.62	0.61	0.69
	$P@7$	$best_{PW} f_{\beta}$	AUC_{ROC}	AUC_{PR}	VUS^4_{ROC}	VUS^4_{PR}
	0.32	0.48	0.61	0.4	0.61	0.49
	0.5	0.63	0.76	0.49	0.73	0.63
	$P@16$	$best_{PW} f_{\beta}$	AUC_{ROC}	AUC_{PR}	VUS^4_{ROC}	VUS^4_{PR}
	0.57	0.73	0.75	0.79	0.65	0.86
	0.57	0.73	0.56	0.62	0.59	0.83
	$P@3$	$best_{PW} f_{\beta}$	AUC_{ROC}	AUC_{PR}	VUS^4_{ROC}	VUS^4_{PR}
	0.18	0.3	0.46	0.18	0.48	0.11
	0.18	0.3	0.46	0.18	0.48	0.11
	0.18	0.3	0.46	0.18	0.5	0.19
	0.18	0.3	0.46	0.18	0.56	0.4
	$P@3$	$best_{PW} f_{\beta}$	AUC_{ROC}	AUC_{PR}	VUS^4_{ROC}	VUS^4_{PR}
	0.0	0.38	0.39	0.17	0.46	0.17
	0.0	0.43	0.54	0.21	0.6	0.22
	0.0	0.5	0.68	0.27	0.75	0.33
	0.33	0.6	0.82	0.39	0.88	0.56

[29] Sørnbø, S., Ruocco, M. Navigating the metric maze: a taxonomy of evaluation metrics for anomaly detection in time series. Data Min Knowl Disc 38, 1027–1068 (2024)

Evaluating measures: A Hard question

Synthetic Evaluation [29]

NAVIGATING THE METRIC MAZE: A TAXONOMY OF EVALUATION METRICS FOR ANOMALY DETECTION IN TIME SERIES

Sondre Sørbo¹
sondre.sorbo@sintef.no

Massimiliano Ruocco^{1,2}
massimiliano.ruocco@sintef.no

¹SINTEF Digital, Trondheim, Norway

²Norwegian University of Science and Technology, Trondheim, Norway

ABSTRACT

The field of time series anomaly detection is constantly advancing, with several methods available, making it a challenge to determine the most appropriate method for a specific domain. The evaluation of these methods is facilitated by the use of metrics, which vary widely in their properties. Despite the existence of new evaluation metrics, there is limited agreement on which metrics are best suited for specific scenarios and domain, and the most commonly used metrics have faced criticism in the literature. This paper provides a comprehensive overview of the metrics used for the evaluation of time series anomaly detection methods, and also defines a taxonomy of these based on how they are calculated. By defining a set of properties for evaluation metrics and a set of specific case studies and experiments, twenty metrics are analyzed and discussed in detail, highlighting the unique suitability of each for specific tasks. Through extensive experimentation and analysis, this paper argues that the choice of evaluation metric must be made with care, taking into account the specific requirements of the task at hand.

Keywords Time series · Anomaly detection · Evaluation · Taxonomy

[29] Sørbo, S., Ruocco, M. Navigating the metric maze: a taxonomy of evaluation metrics for anomaly detection in time series. Data Min Knowl Disc 38, 1027–1068 (2024)

Evaluating measures: A Hard question

Synthetic Evaluation [29]

NAVIGATING THE METRIC MAZE: A TAXONOMY OF EVALUATION METRICS FOR ANOMALY DETECTION IN TIME SERIES

Sondre Sørbo¹
sondre.sorbo@sintef.no

Massimiliano Ruocco^{1,2}
massimiliano.ruocco@sintef.no

¹SINTEF Digital, Trondheim, Norway

²Norwegian University of Science and Technology, Trondheim, Norway

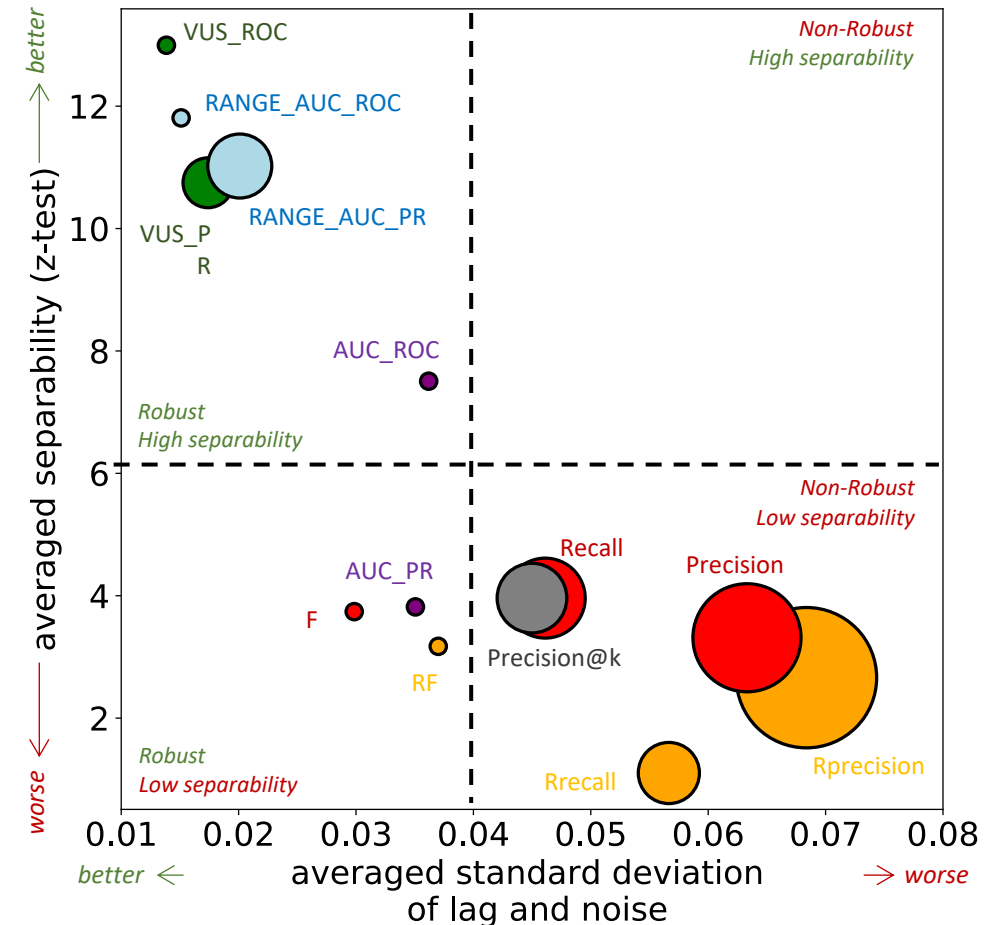
ABSTRACT

The field of time series anomaly detection is constantly advancing, with several methods available, making it a challenge to determine the most appropriate method for a specific domain. The evaluation of these methods is facilitated by the use of metrics, which vary widely in their properties. Despite the existence of new evaluation metrics, there is limited agreement on which metrics are best suited for specific scenarios and domain, and the most commonly used metrics have faced criticism in the literature. This paper provides a comprehensive overview of the metrics used for the evaluation of time series anomaly detection methods, and also defines a taxonomy of these based on how they are calculated. By defining a set of properties for evaluation metrics and a set of specific case studies and experiments, twenty metrics are analyzed and discussed in detail, highlighting the unique suitability of each for specific tasks. Through extensive experimentation and analysis, this paper argues that the choice of evaluation metric must be made with care, taking into account the specific requirements of the task at hand.

Keywords Time series · Anomaly detection · Evaluation · Taxonomy

[29] Sørbo, S., Ruocco, M. Navigating the metric maze: a taxonomy of evaluation metrics for anomaly detection in time series. Data Min Knowl Disc 38, 1027–1068 (2024)

Real data Evaluation [22]



[22] J. Paparrizos, P. Boniol, T. Palpanas, R. S. Tsay, A. Elmore, and M. J. Franklin. Volume under the surface: a new accuracy evaluation measure for time-series anomaly detection. Proc. VLDB Endow. 15, 11 (2022), 2774–2787.

Evaluating measures: A Hard question

Synthetic Evaluation [29]

NAVIGATING THE METRIC MAZE: A TAXONOMY OF EVALUATION METRICS FOR ANOMALY DETECTION IN TIME SERIES

Sondre Sørbo¹
sondre.sorbo@sintef.no

Massimiliano Ruocco^{1,2}
massimiliano.ruocco@sintef.no

¹SINTEF Digital, Trondheim, Norway

²Norwegian University of Science and Technology, Trondheim, Norway

ABSTRACT

The field of time series anomaly detection is constantly advancing, with several methods available, making it a challenge to determine the most appropriate method for a specific domain. The evaluation of these methods is facilitated by the use of metrics, which vary widely in their properties. Despite the existence of new evaluation metrics, there is limited agreement on which metrics are best suited for specific scenarios and domain, and the most commonly used metrics have faced criticism in the literature. This paper provides a comprehensive overview of the metrics used for the evaluation of time series anomaly detection methods, and also defines a taxonomy of these based on how they are calculated. By defining a set of properties for evaluation metrics and a set of specific case studies and experiments, twenty metrics are analyzed and discussed in detail, highlighting the unique suitability of each for specific tasks. Through extensive experimentation and analysis, this paper argues that the choice of evaluation metric must be made with care, taking into account the specific requirements of the task at hand.

Keywords Time series · Anomaly detection · Evaluation · Taxonomy

[29] Sørbo, S., Ruocco, M. Navigating the metric maze: a taxonomy of evaluation metrics for anomaly detection in time series. Data Min Knowl Disc 38, 1027–1068 (2024)

Real data Evaluation [22]

Evaluate measures on three criteria:

- **Robustness:** *The ability of a measure to be invariant to irrelevant changes (i.e., that do not reflect a change of detection quality)*
- **Separability:** *An accuracy value of an accurate method should be easy to differentiate from an accurate value of an inaccurate method.*
- **Consistency:** On similar time series, the ranking of methods provided by an accuracy measure should be similar

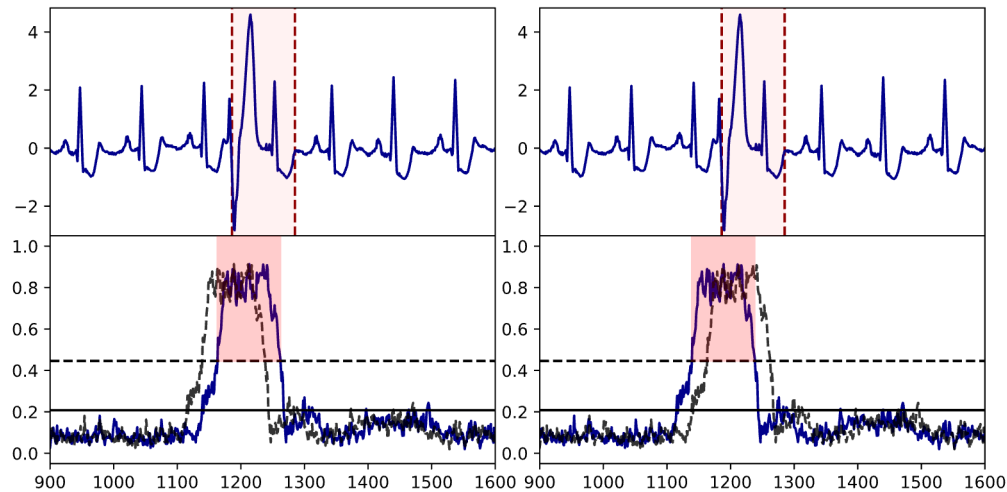
[22] J. Paparrizos, P. Boniol, T. Palpanas, R. S. Tsay, A. Elmore, and M. J. Franklin. Volume under the surface: a new accuracy evaluation measure for time-series anomaly detection. Proc. VLDB Endow. 15, 11 (2022), 2774–2787.

Evaluating measures: About Robustness...

Robustness: *The ability of a measure to be invariant to irrelevant changes (i.e., that do not reflect a change of detection quality)*

Evaluating measures: About Robustness...

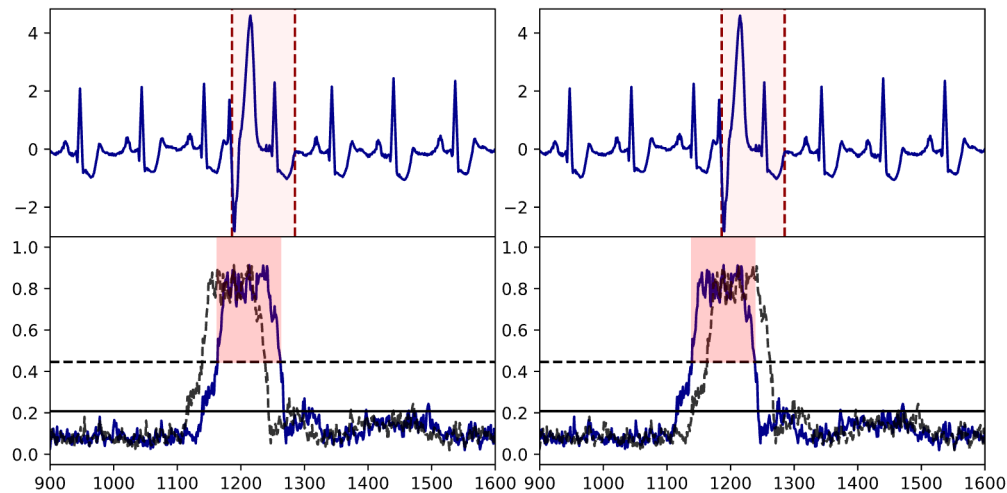
Robustness: *The ability of a measure to be invariant to irrelevant changes (i.e., that do not reflect a change of detection quality)*



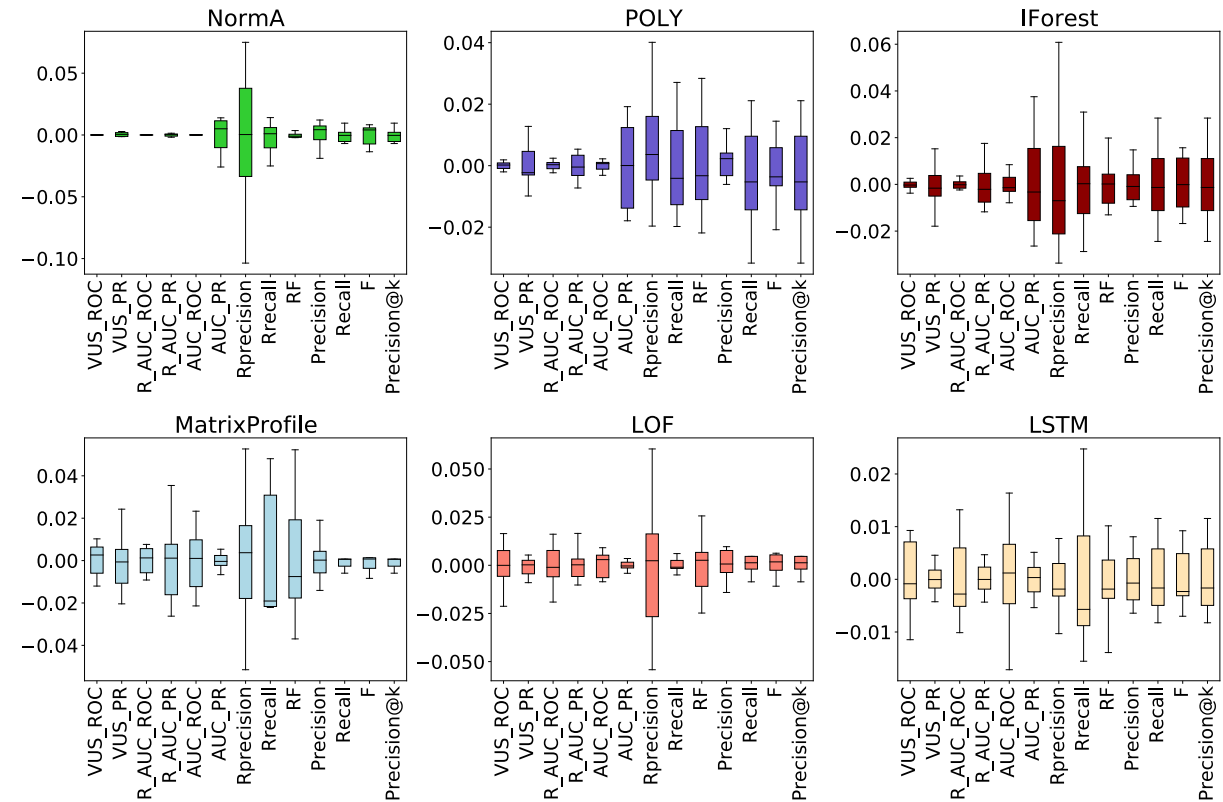
To lag

Evaluating measures: About Robustness...

Robustness: *The ability of a measure to be invariant to irrelevant changes (i.e., that do not reflect a change of detection quality)*



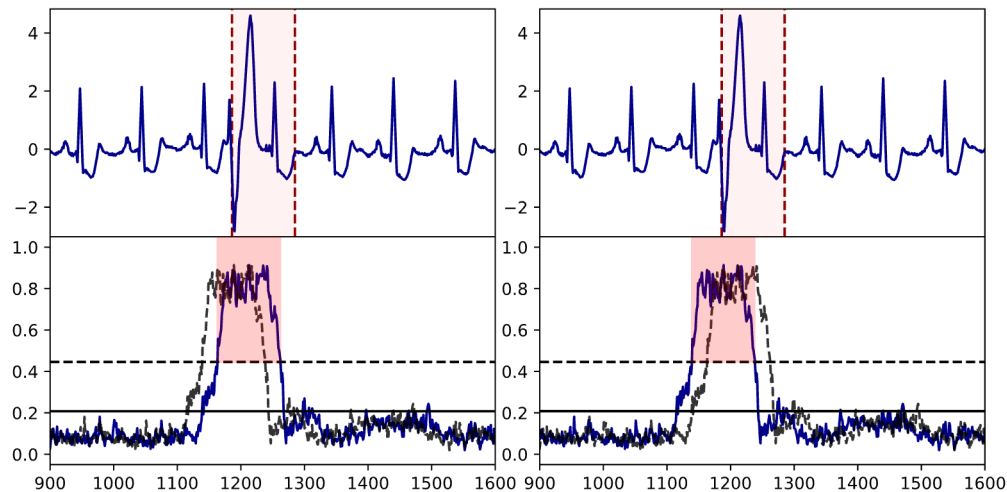
To lag



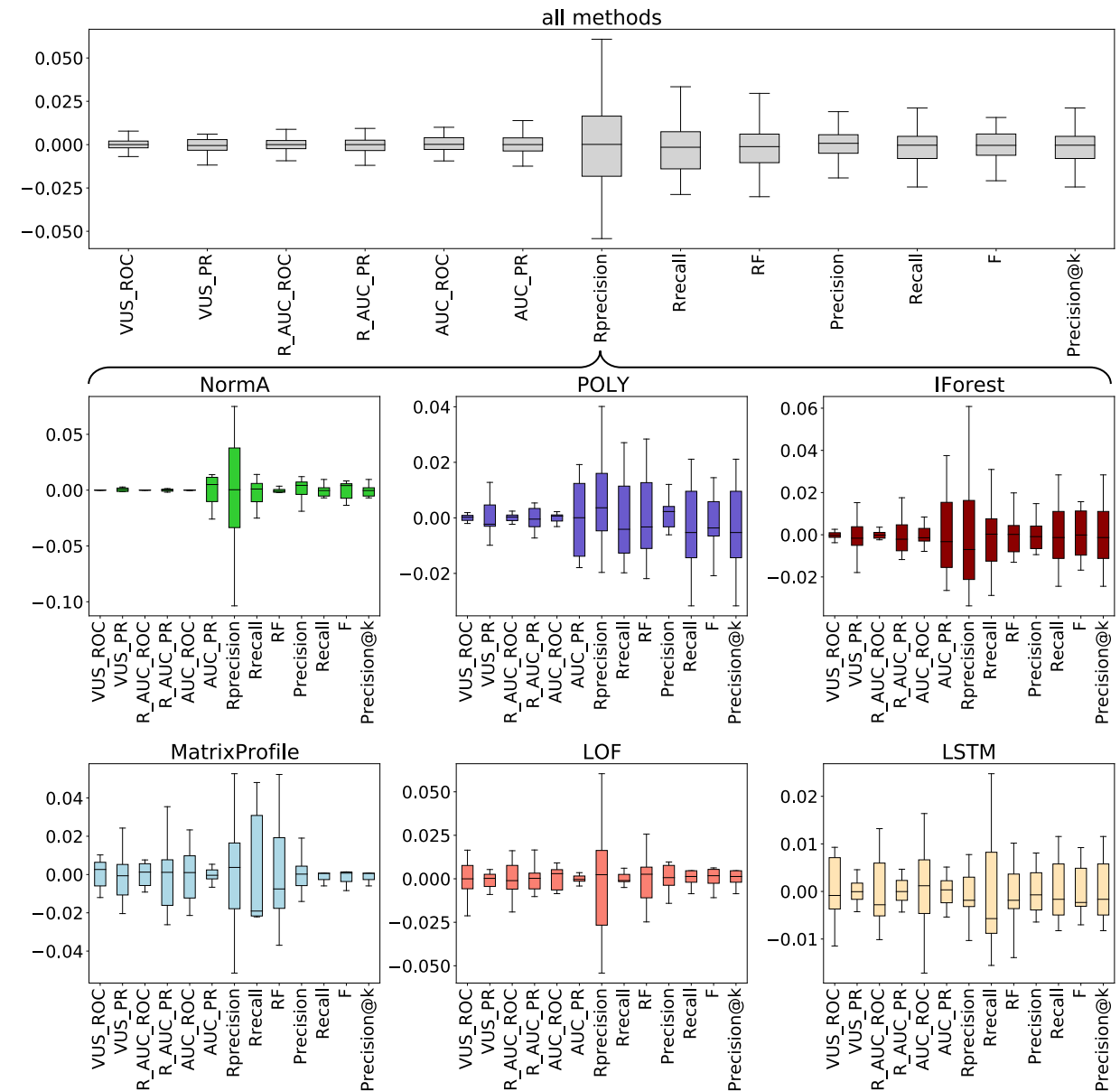
For each methods (and averaged for all methods in (c.1)), we compute 10 times the accuracy measures with random lag lg (with $0.25 * \ell < lg < 0.25 * \ell$) in the anomaly score. We center the accuracy average to 0.

Evaluating measures: About Robustness...

Robustness: *The ability of a measure to be invariant to irrelevant changes (i.e., that do not reflect a change of detection quality)*



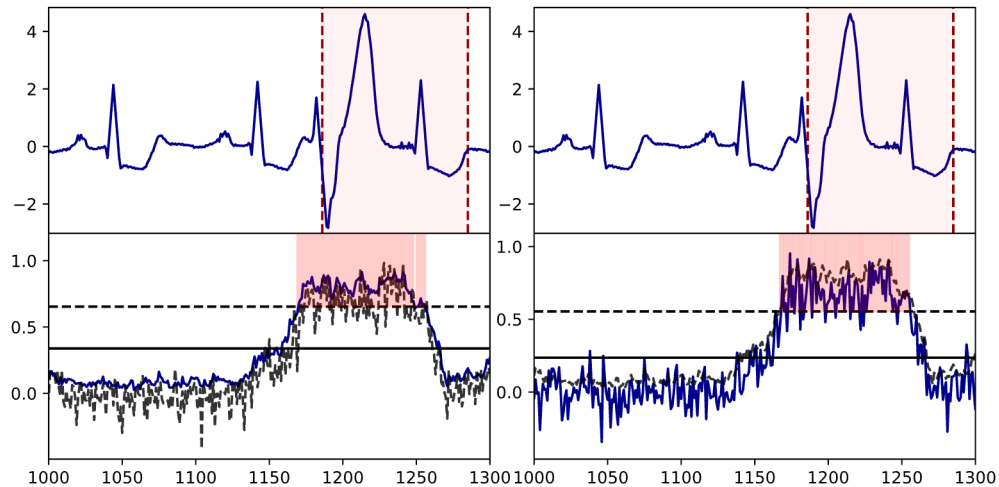
To lag



For each methods (and averaged for all methods in (c.1)), we compute 10 times the accuracy measures with random lag lg (with $0.25 * \ell < lg < 0.25 * \ell$) in the anomaly score. We center the accuracy average to 0.

Evaluating measures: About Robustness...

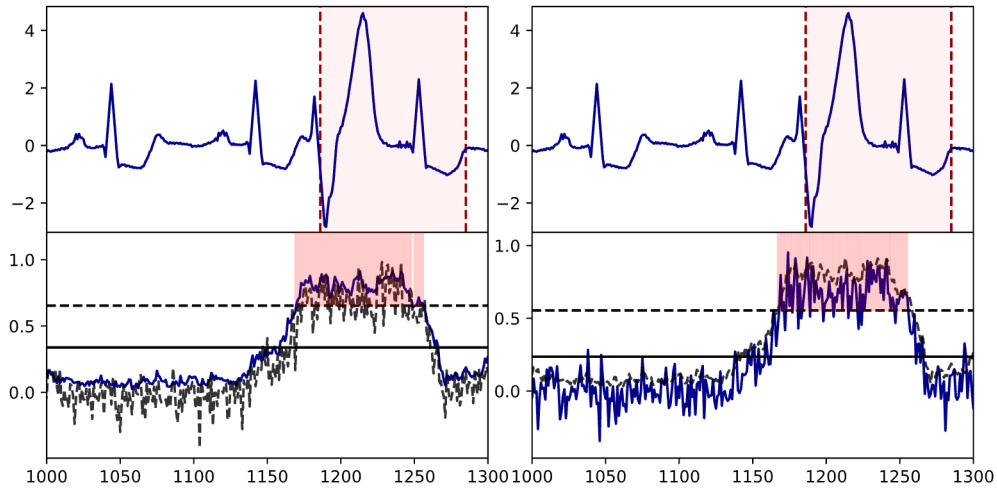
Robustness: *The ability of a measure to be invariant to irrelevant changes (i.e., that do not reflect a change of detection quality)*



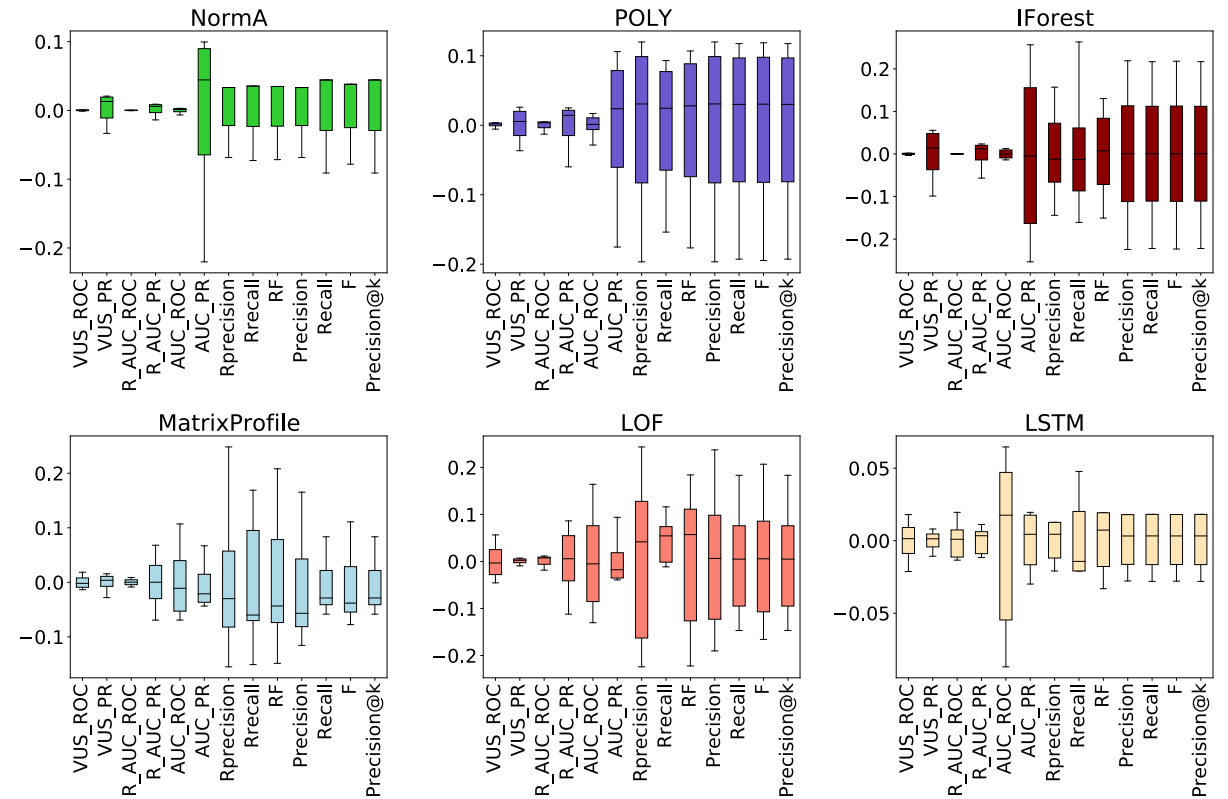
To noise

Evaluating measures: About Robustness...

Robustness: *The ability of a measure to be invariant to irrelevant changes (i.e., that do not reflect a change of detection quality)*



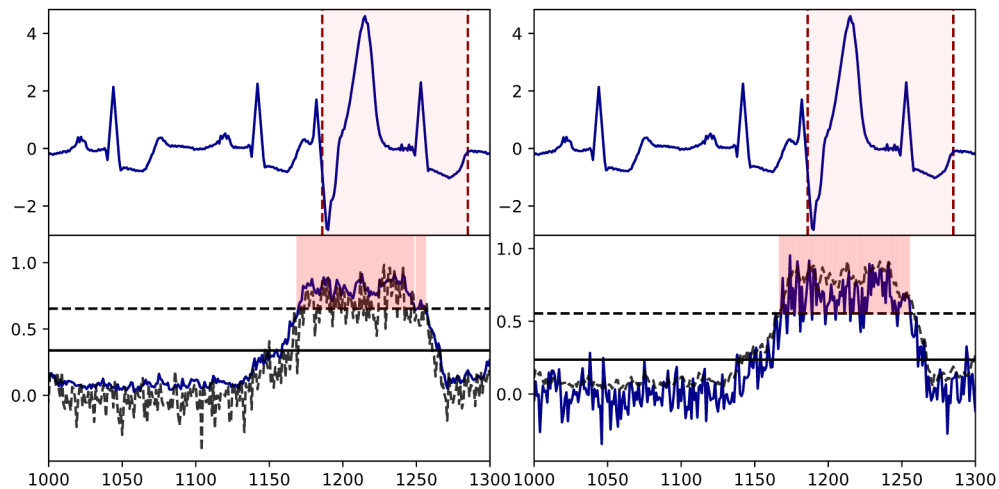
To noise



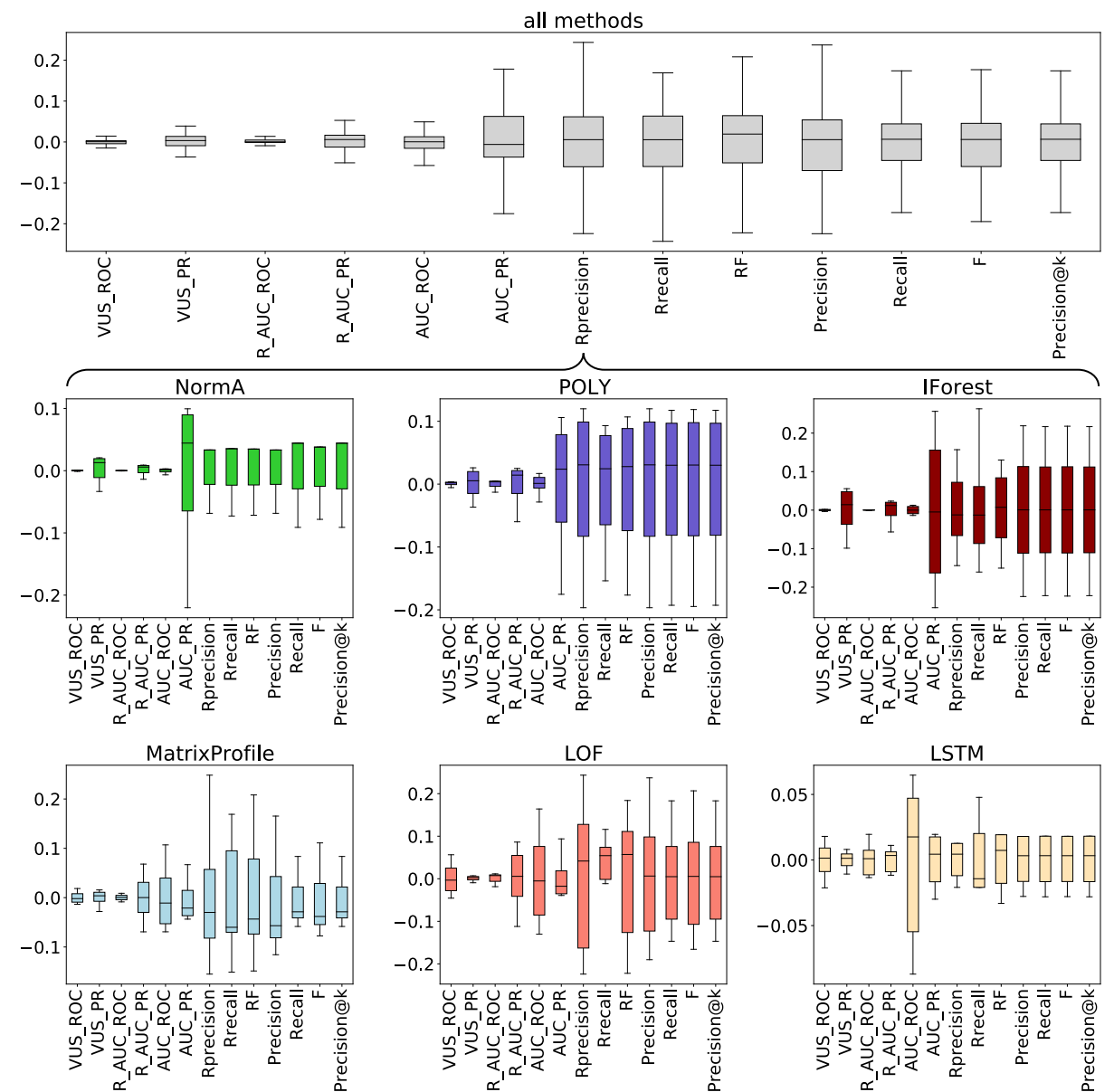
For each methods (and averaged for all methods in (b.1)), we compute 10 times the accuracy measures with random noise (between -0.1 and 0.1) injected in the anomaly score. We center the accuracy average to 0.

Evaluating measures: About Robustness...

Robustness: *The ability of a measure to be invariant to irrelevant changes (i.e., that do not reflect a change of detection quality)*



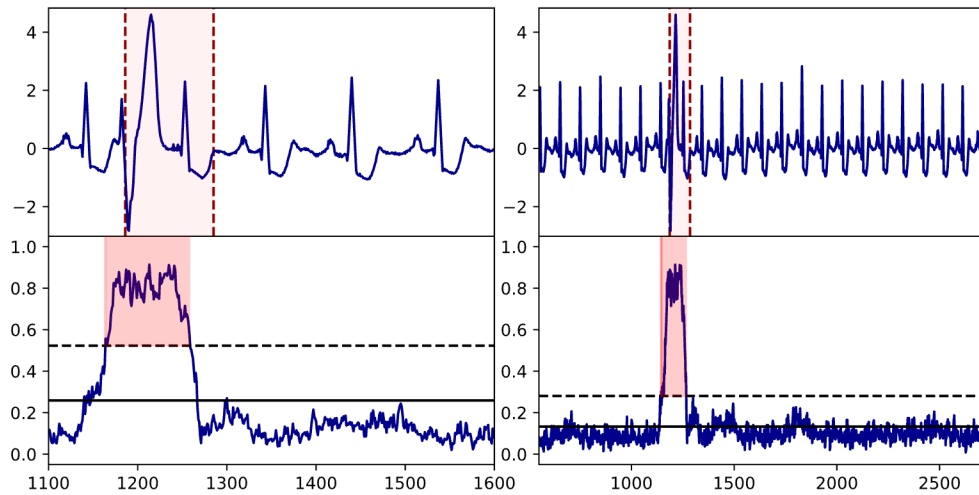
To noise



For each methods (and averaged for all methods in (b.1)), we compute 10 times the accuracy measures with random noise (between -0.1 and 0.1) injected in the anomaly score. We center the accuracy average to 0.

Evaluating measures: About Robustness...

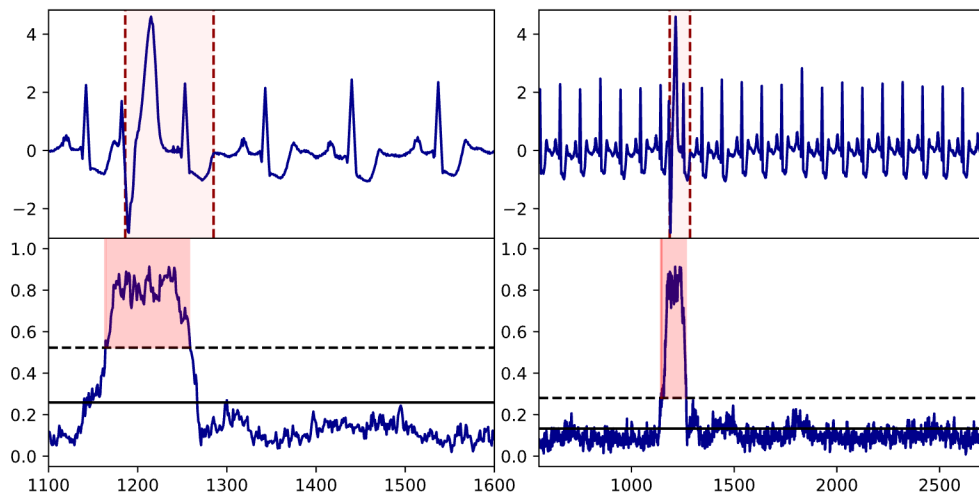
Robustness: *The ability of a measure to be invariant to irrelevant changes (i.e., that do not reflect a change of detection quality)*



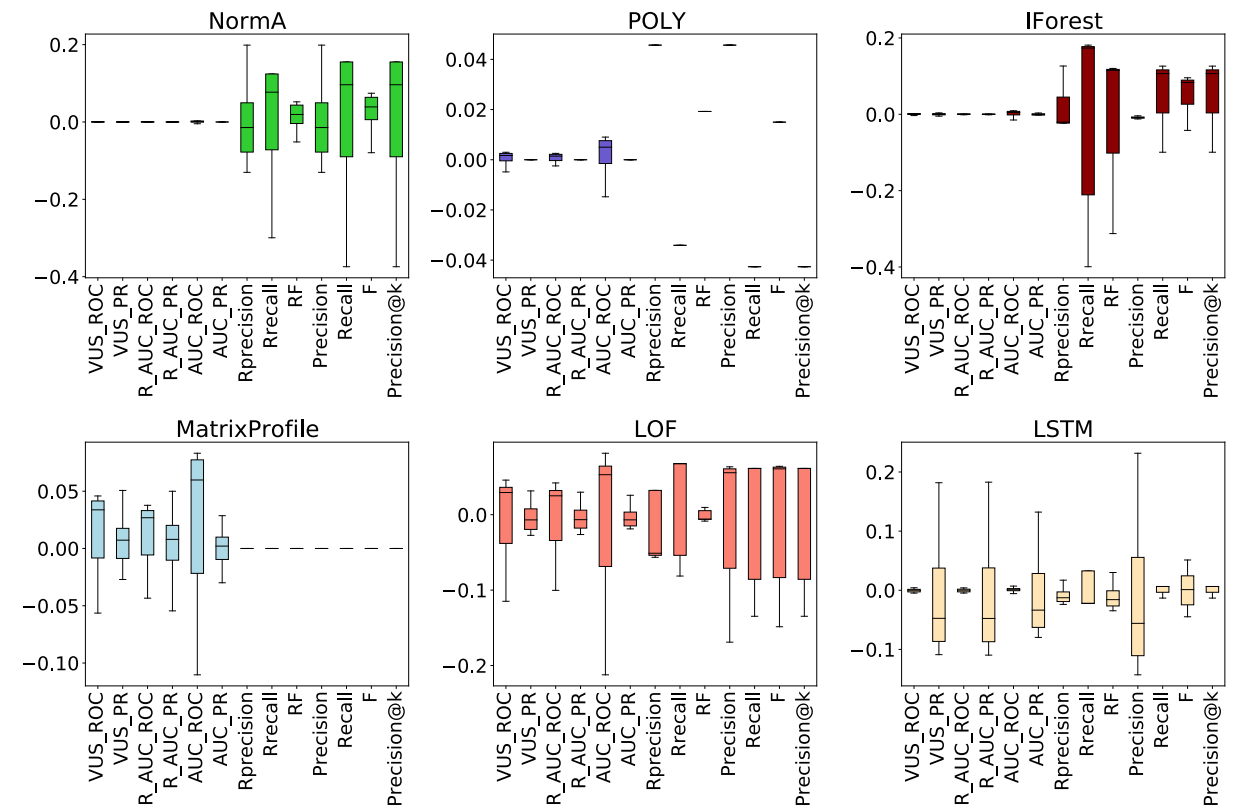
To normal/abnormal ratio

Evaluating measures: About Robustness...

Robustness: *The ability of a measure to be invariant to irrelevant changes (i.e., that do not reflect a change of detection quality)*



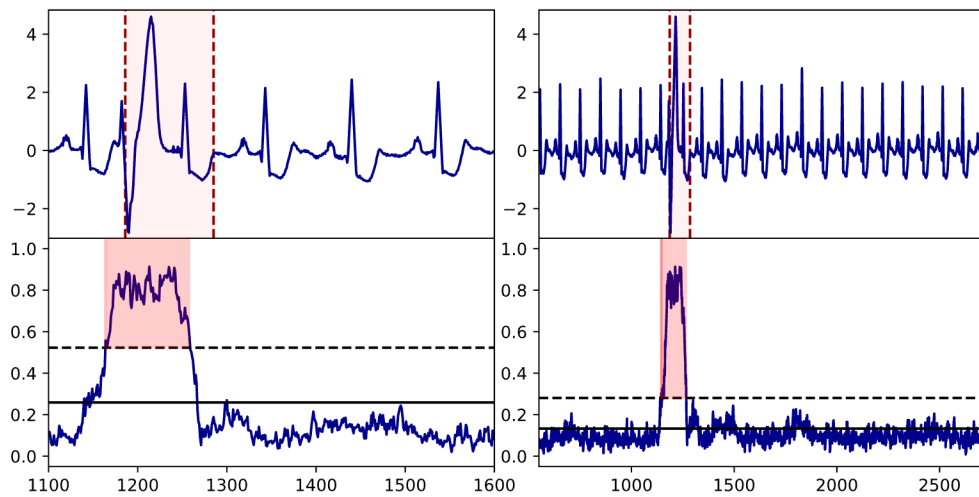
To normal/abnormal ratio



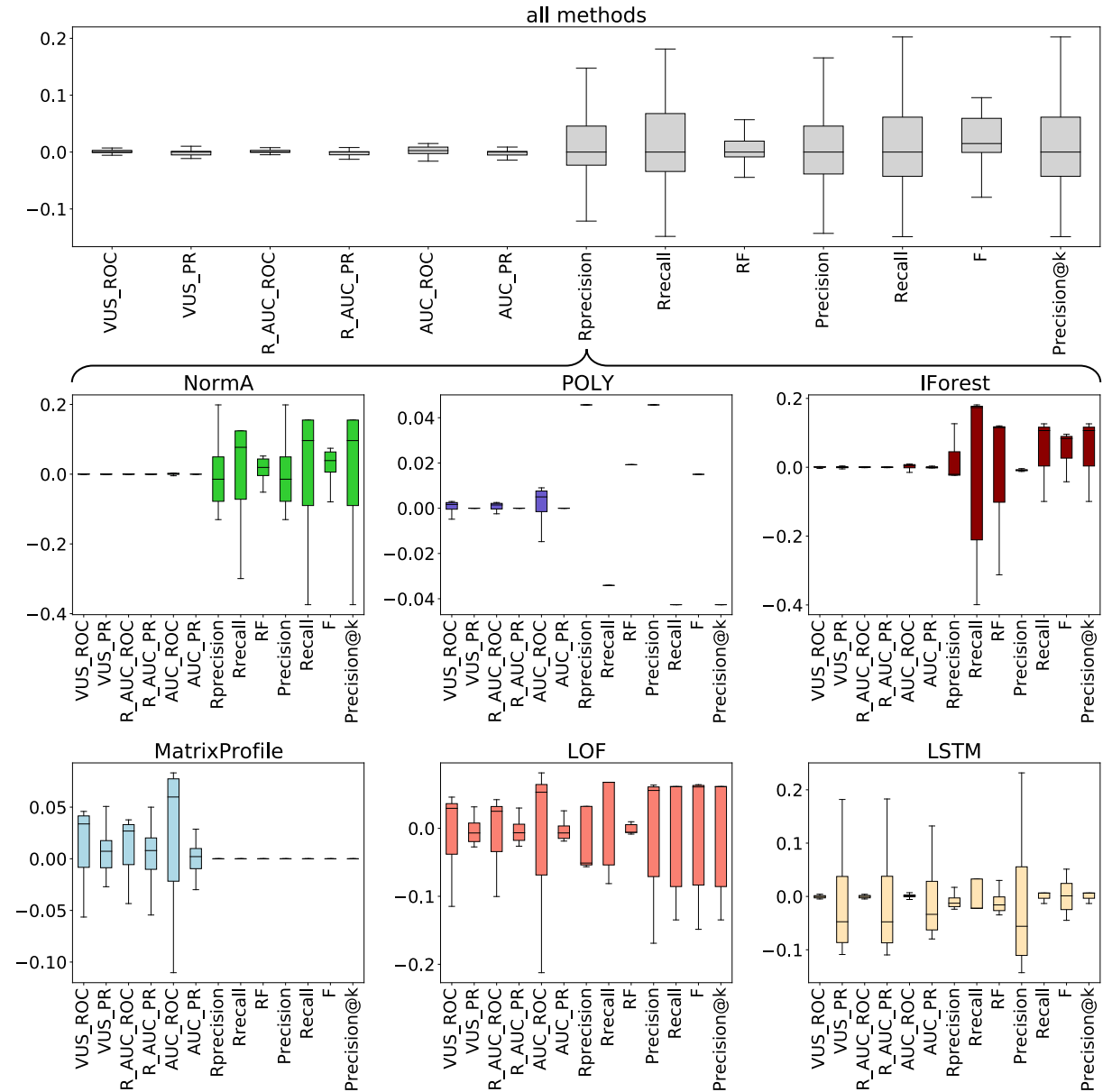
For each methods (and averaged for all methods in (d.1)), we compute 10 times the accuracy measures with random normal/abnormal ratio r (with $0.01 < r < 0.2$) in data series. We center the accuracy average to 0.

Evaluating measures: About Robustness...

Robustness: *The ability of a measure to be invariant to irrelevant changes (i.e., that do not reflect a change of detection quality)*



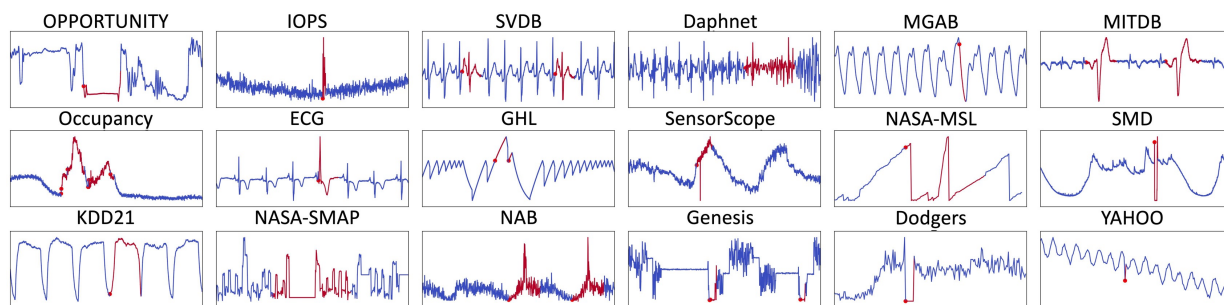
To normal/abnormal ratio



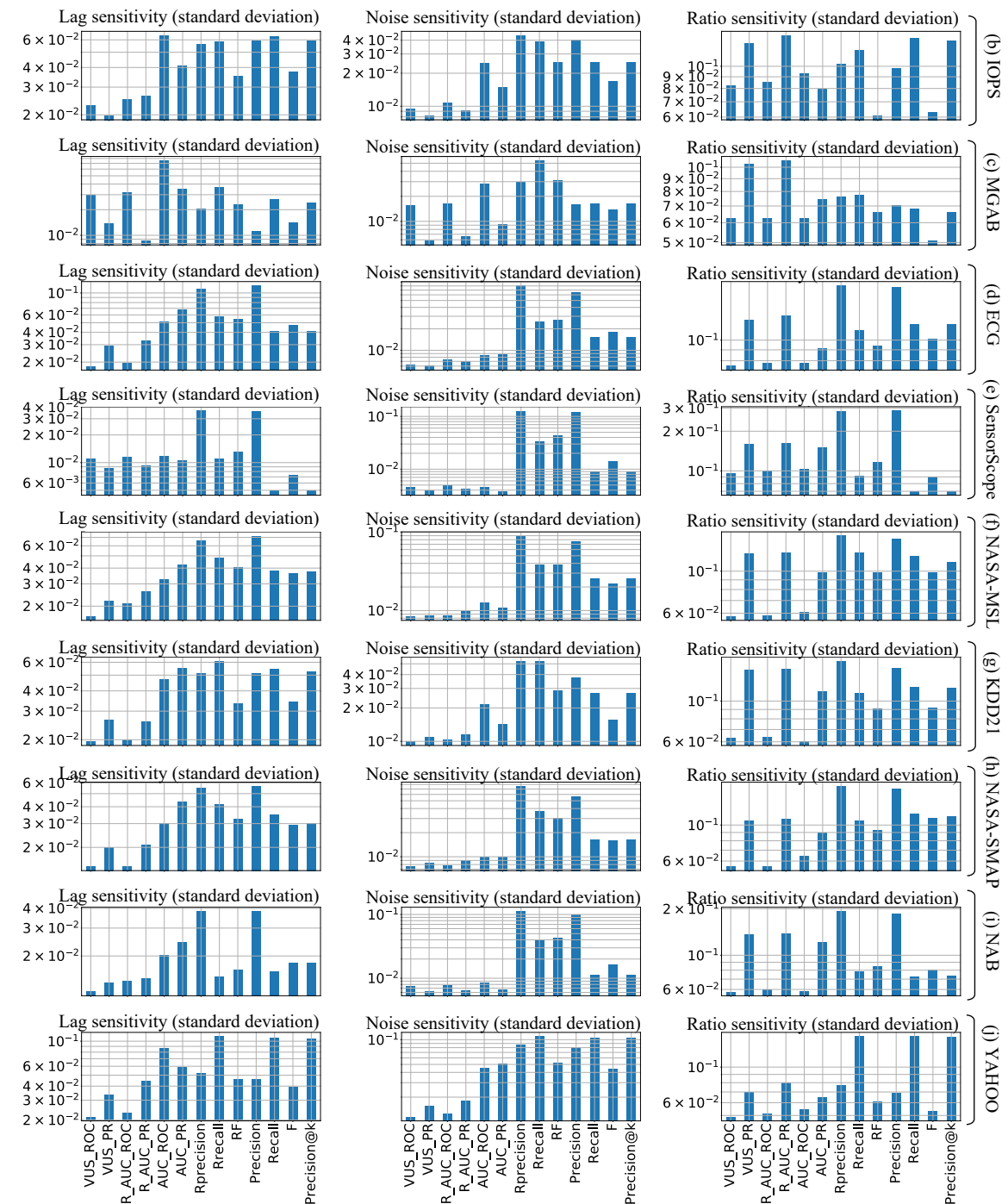
For each methods (and averaged for all methods in (d.1)), we compute 10 times the accuracy measures with random normal/abnormal ratio r (with $0.01 < r < 0.2$) in data series. We center the accuracy average to 0.

Evaluating measures: About Robustness...

Robustness: *The ability of a measure to be invariant to irrelevant changes (i.e., that do not reflect a change of detection quality)*



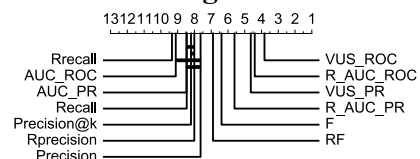
For all TSB-UAD benchmark



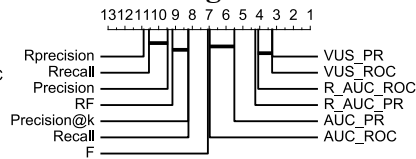
Evaluating measures: About Robustness...

Robustness: *The ability of a measure to be invariant to irrelevant changes (i.e., that do not reflect a change of detection quality)*

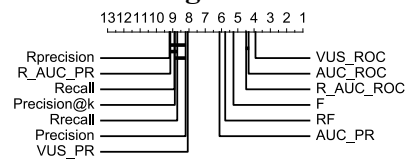
Lag sensitivity critical diagram



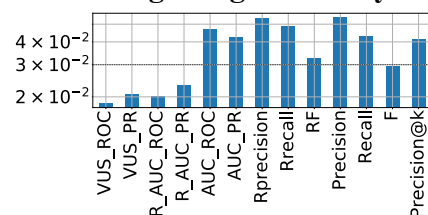
Noise sensitivity critical diagram



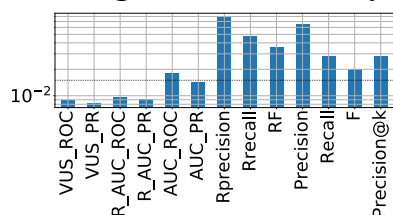
ratio sensitivity critical diagram



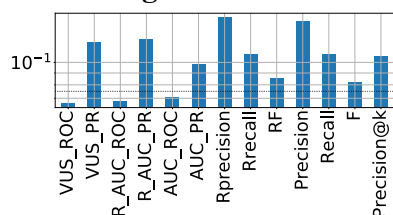
Averaged lag sensitivity



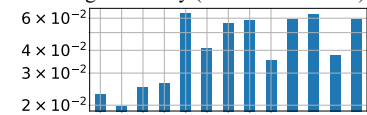
Averaged noise sensitivity



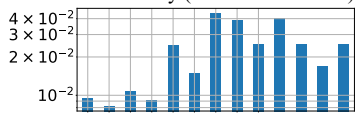
Averaged ratio sensitivity



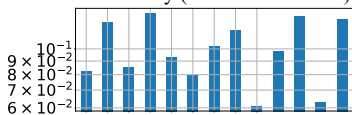
Lag sensitivity (standard deviation)



Noise sensitivity (standard deviation)

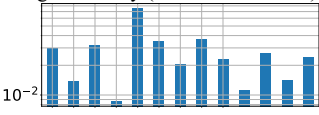


Ratio sensitivity (standard deviation)

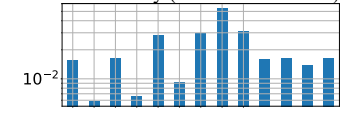


(b) IOPS

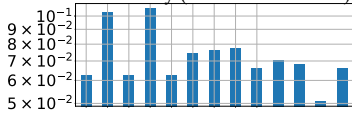
Lag sensitivity (standard deviation)



Noise sensitivity (standard deviation)

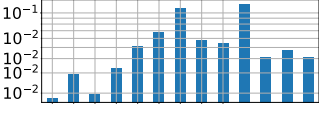


Ratio sensitivity (standard deviation)

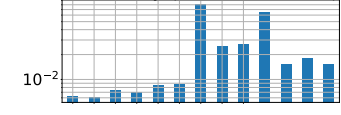


(c) MGAB

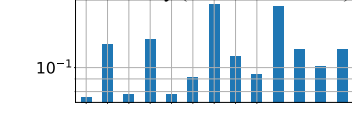
Lag sensitivity (standard deviation)



Noise sensitivity (standard deviation)

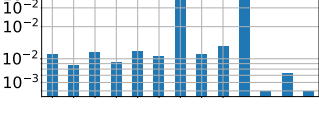


Ratio sensitivity (standard deviation)

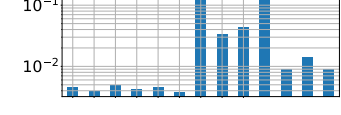


(d) ECG

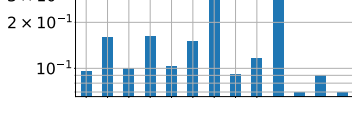
Lag sensitivity (standard deviation)



Noise sensitivity (standard deviation)

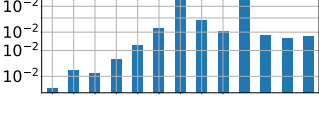


Ratio sensitivity (standard deviation)

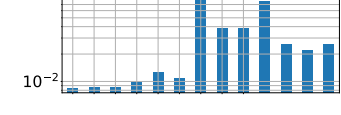


(e) SensorScope

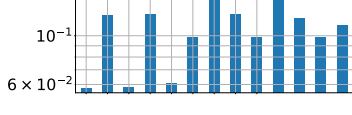
Lag sensitivity (standard deviation)



Noise sensitivity (standard deviation)

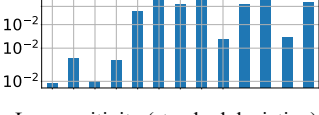


Ratio sensitivity (standard deviation)

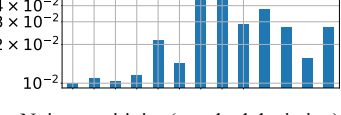


(f) NASA-MSL

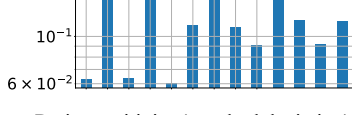
Lag sensitivity (standard deviation)



Noise sensitivity (standard deviation)



Ratio sensitivity (standard deviation)

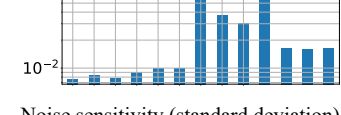


(g) KDD21

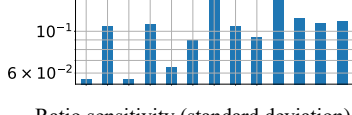
Lag sensitivity (standard deviation)



Noise sensitivity (standard deviation)

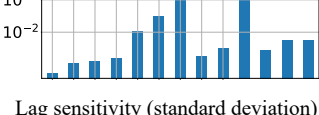


Ratio sensitivity (standard deviation)

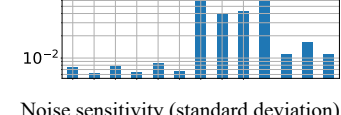


(h) NASA-SMAP

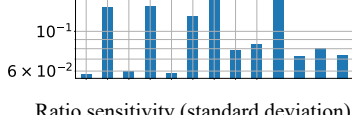
Lag sensitivity (standard deviation)



Noise sensitivity (standard deviation)

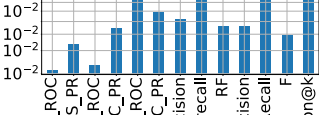


Ratio sensitivity (standard deviation)

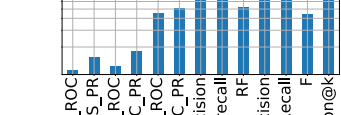


(i) NAB

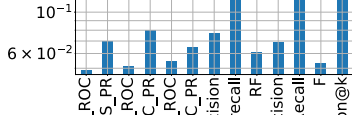
Lag sensitivity (standard deviation)



Noise sensitivity (standard deviation)



Ratio sensitivity (standard deviation)

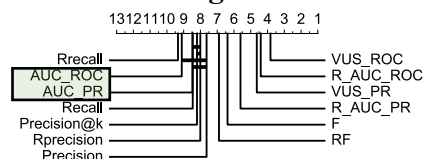


(j) YAHOO

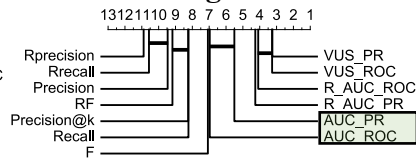
Evaluating measures: About Robustness...

Robustness: *The ability of a measure to be invariant to irrelevant changes (i.e., that do not reflect a change of detection quality)*

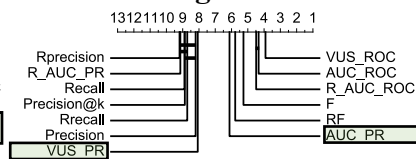
Lag sensitivity critical diagram



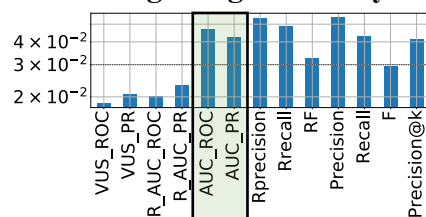
Noise sensitivity critical diagram



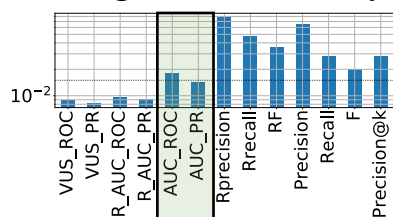
ratio sensitivity critical diagram



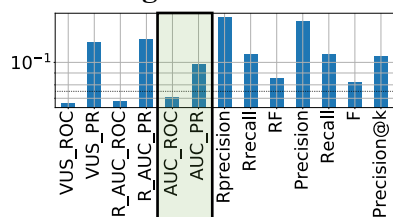
Averaged lag sensitivity



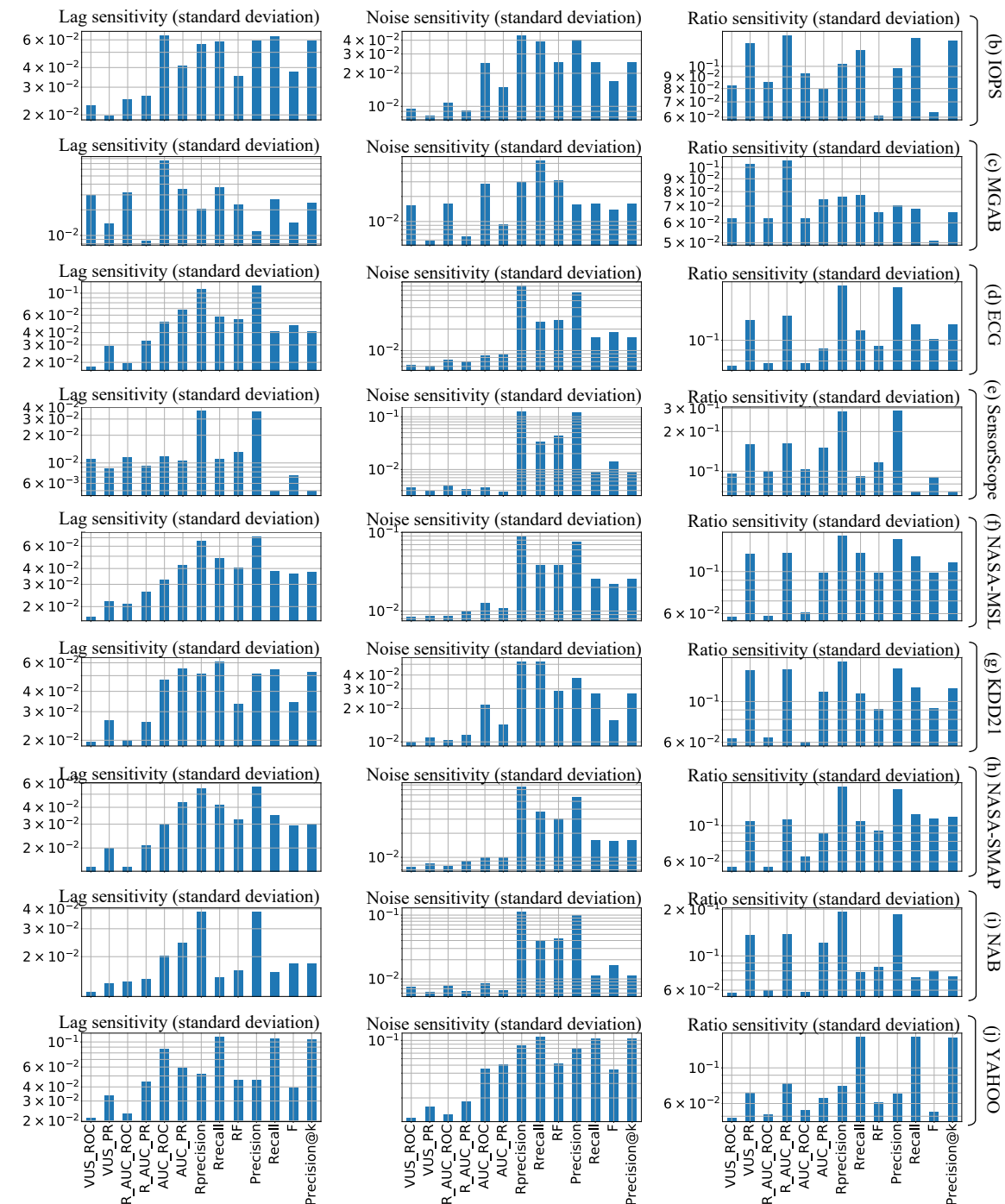
Averaged noise sensitivity



Averaged ratio sensitivity



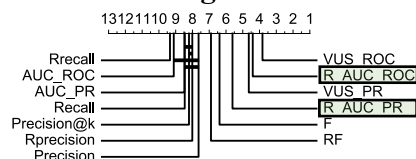
AUC-based



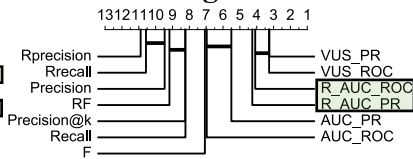
Evaluating measures: About Robustness...

Robustness: *The ability of a measure to be invariant to irrelevant changes (i.e., that do not reflect a change of detection quality)*

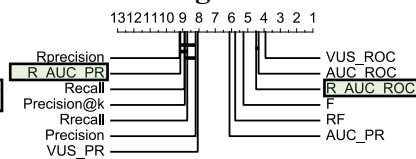
Lag sensitivity critical diagram



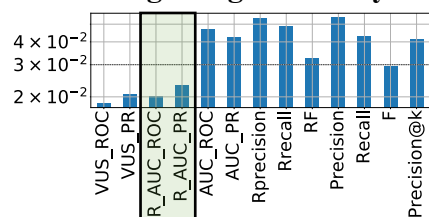
Noise sensitivity critical diagram



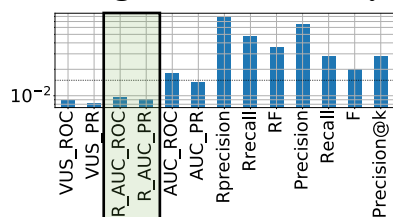
ratio sensitivity critical diagram



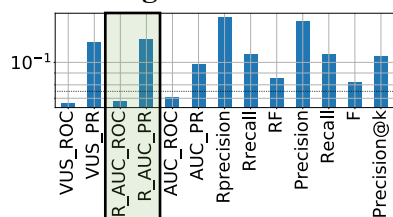
Averaged lag sensitivity



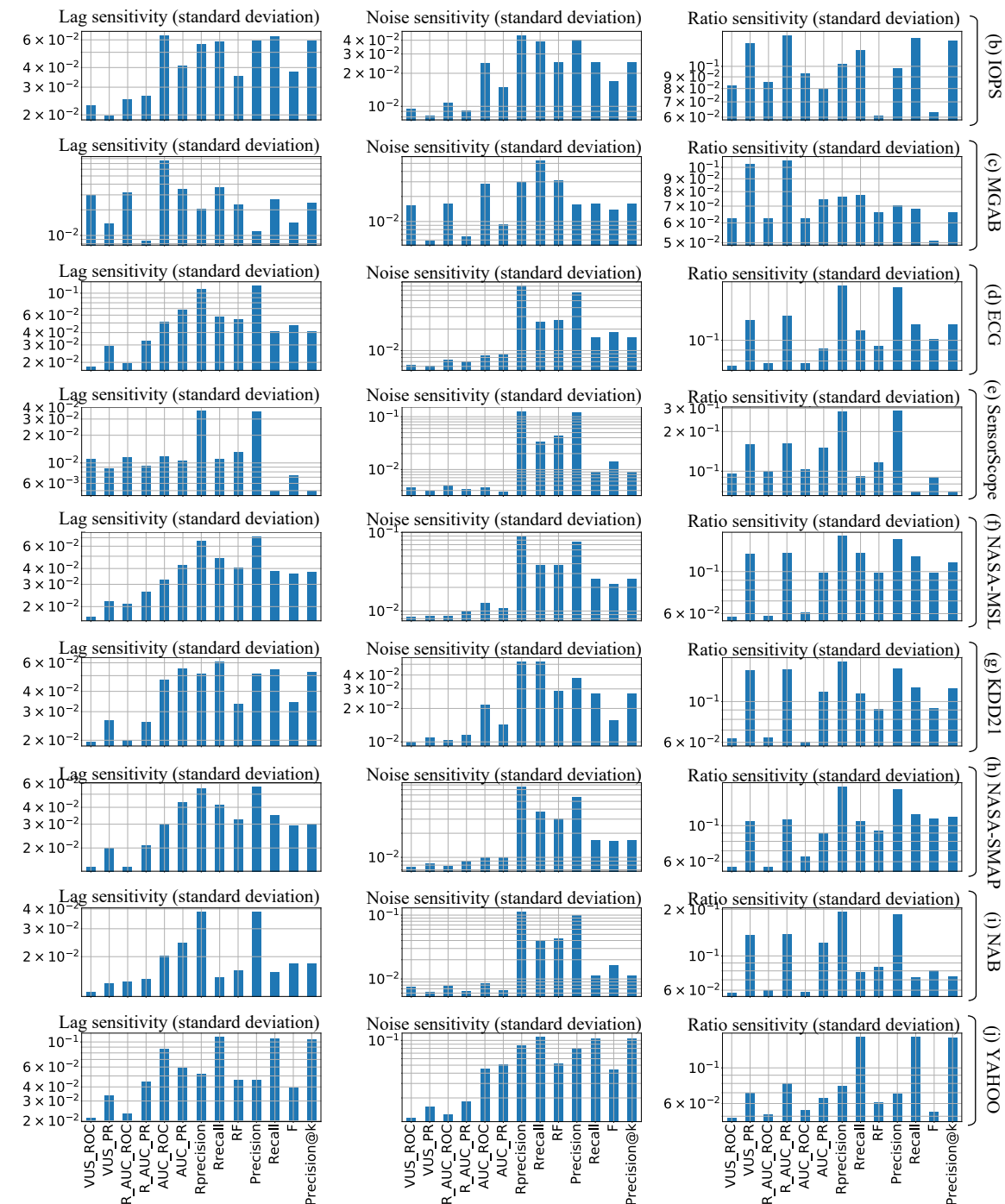
Averaged noise sensitivity



Averaged ratio sensitivity



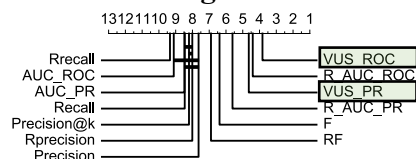
R-AUC-based



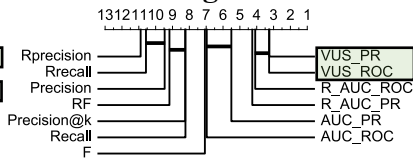
Evaluating measures: About Robustness...

Robustness: *The ability of a measure to be invariant to irrelevant changes (i.e., that do not reflect a change of detection quality)*

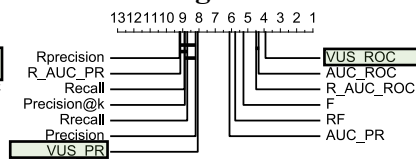
Lag sensitivity critical diagram



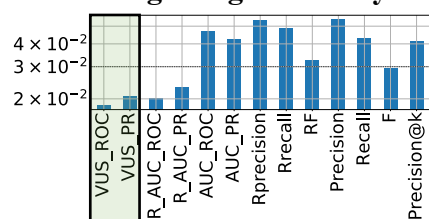
Noise sensitivity critical diagram



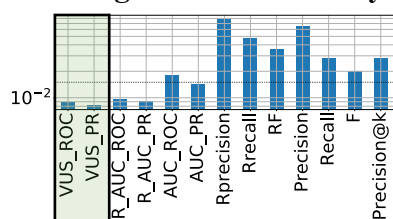
ratio sensitivity critical diagram



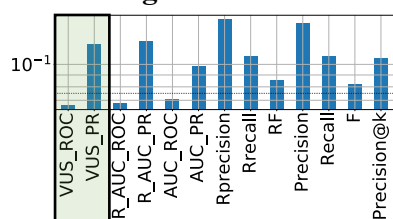
Averaged lag sensitivity



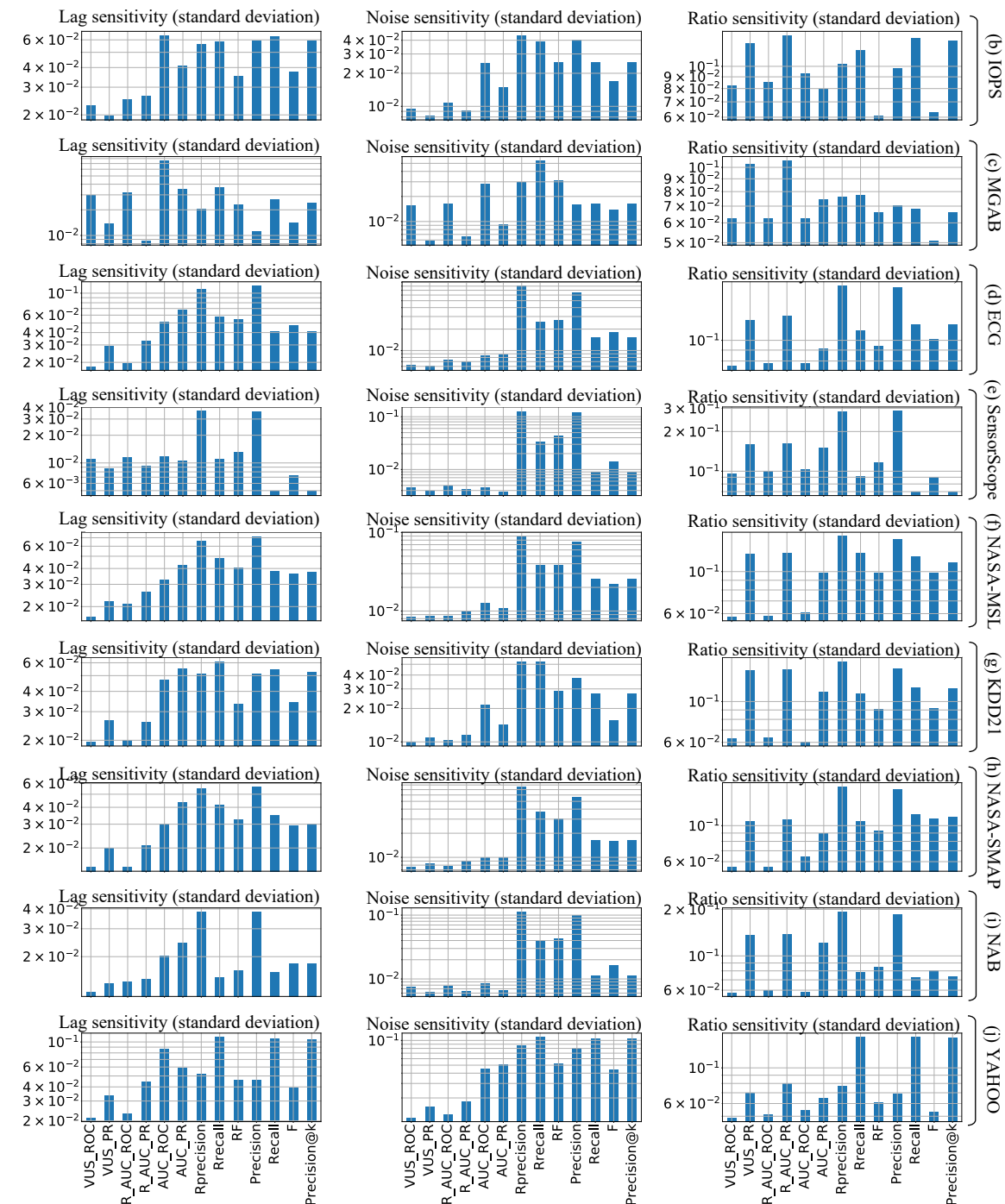
Averaged noise sensitivity



Averaged ratio sensitivity



VUS-based



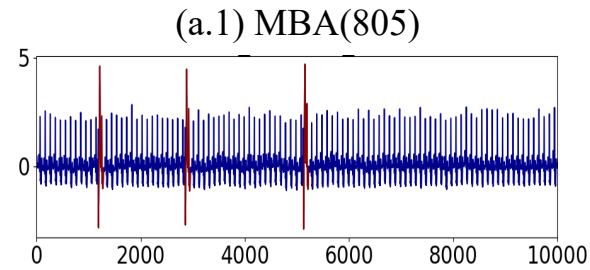
Evaluating measures: About Separability...

Separability: *An accuracy value of an accurate method should be easy to differentiate from an accurate value of an inaccurate method.*

Evaluating measures: About Separability...

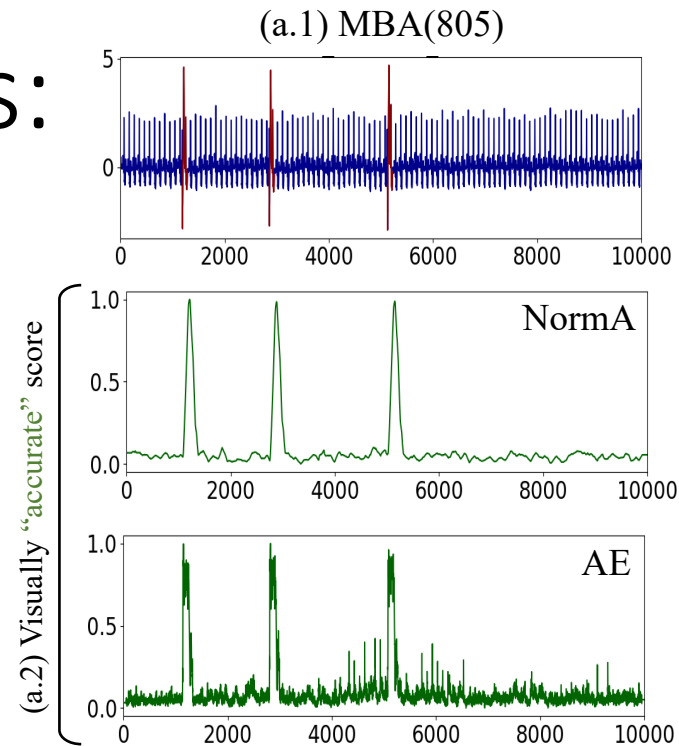
Separability: *An accuracy value of an accurate method should be easy to differentiate from an accurate value of an inaccurate method.*

*But how do we identify accurate methods
without accuracy measures?*



Evaluating measures: About Separability...

Separability: *An accuracy value of an accurate method should be easy to differentiate from an accurate value of an inaccurate method.*

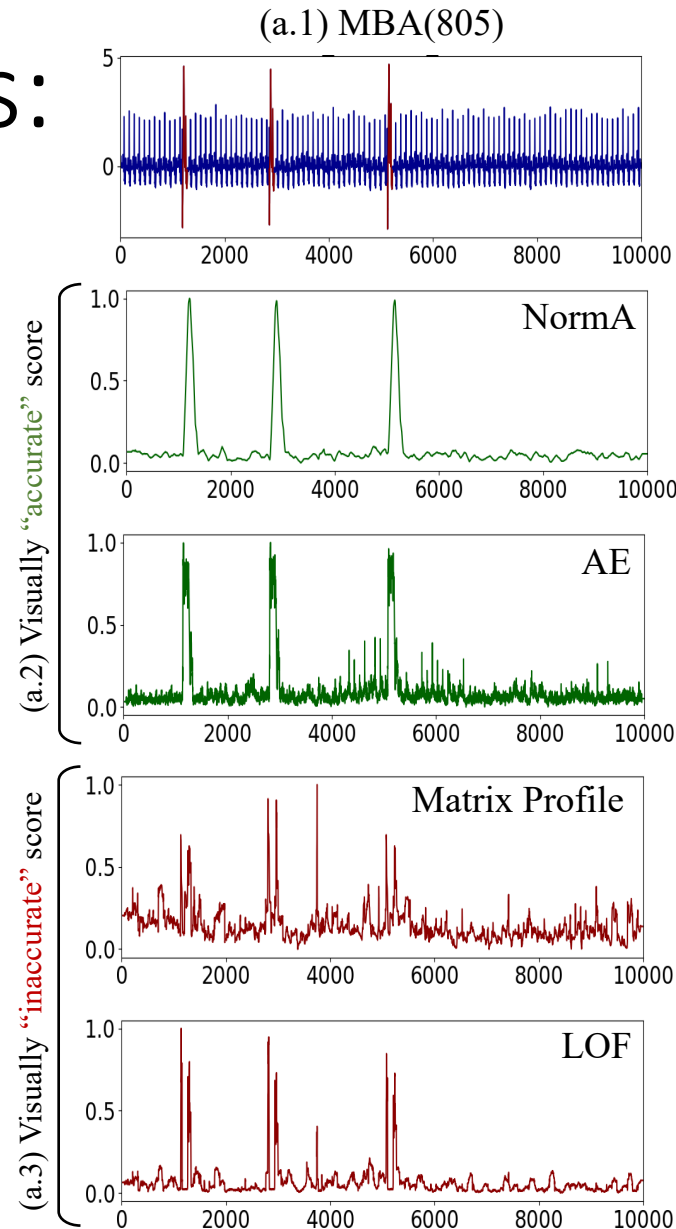


*But how do we identify accurate methods
without accuracy measures?*

Evaluating measures: About Separability...

Separability: *An accuracy value of an accurate method should be easy to differentiate from an accurate value of an inaccurate method.*

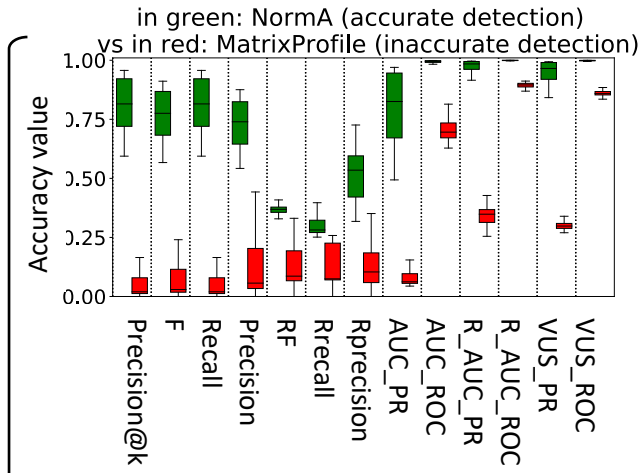
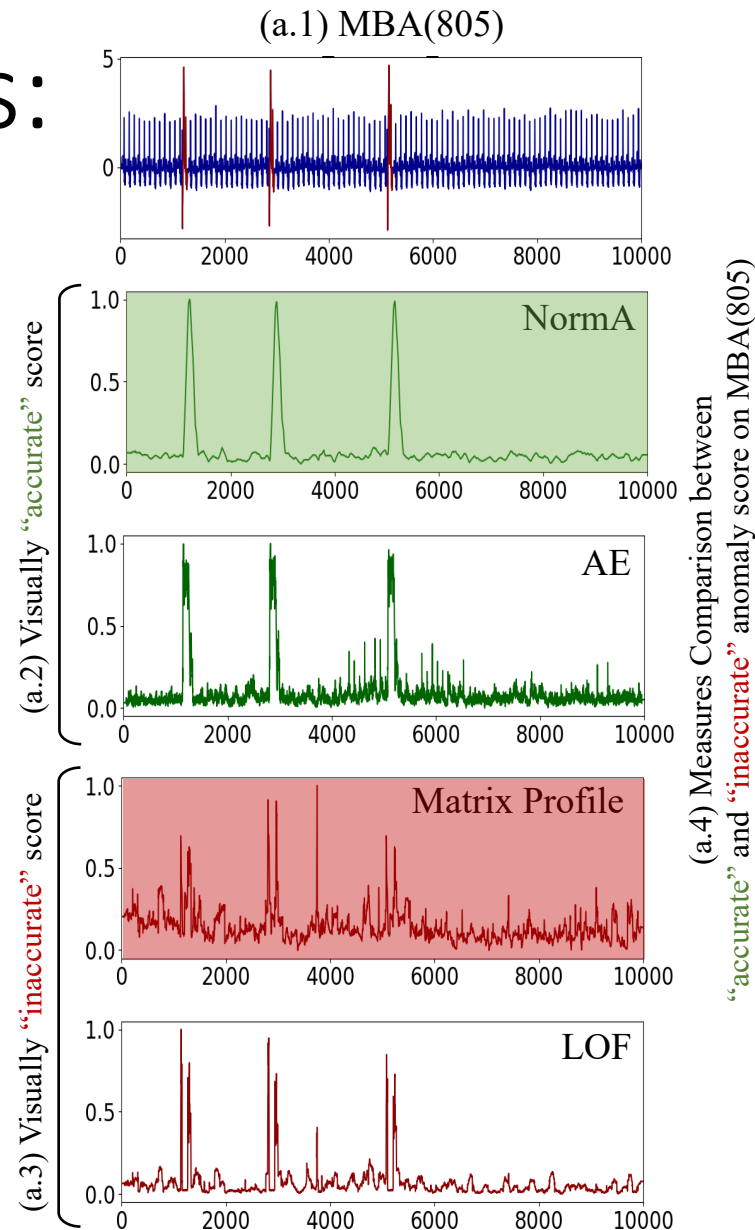
But how do we identify accurate methods without accuracy measures?



Evaluating measures: About Separability...

Separability: *An accuracy value of an accurate method should be easy to differentiate from an accurate value of an inaccurate method.*

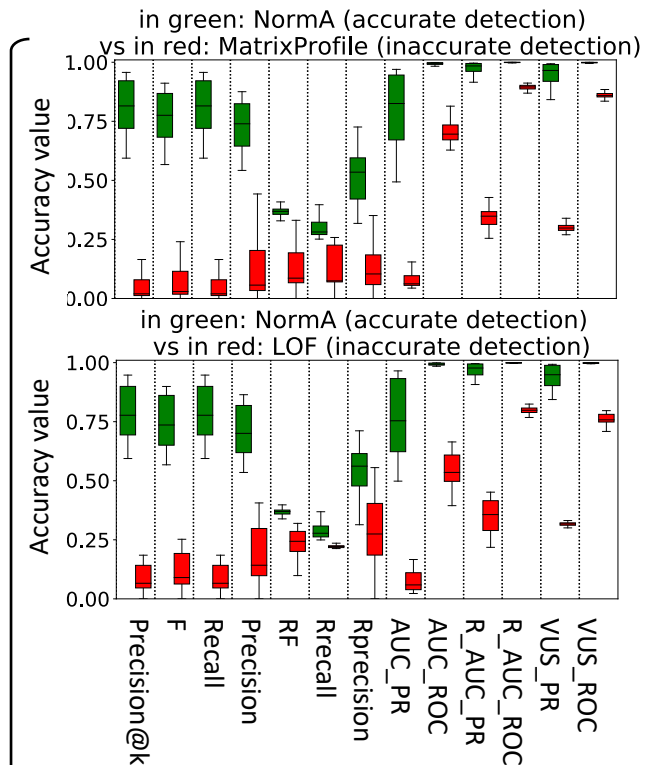
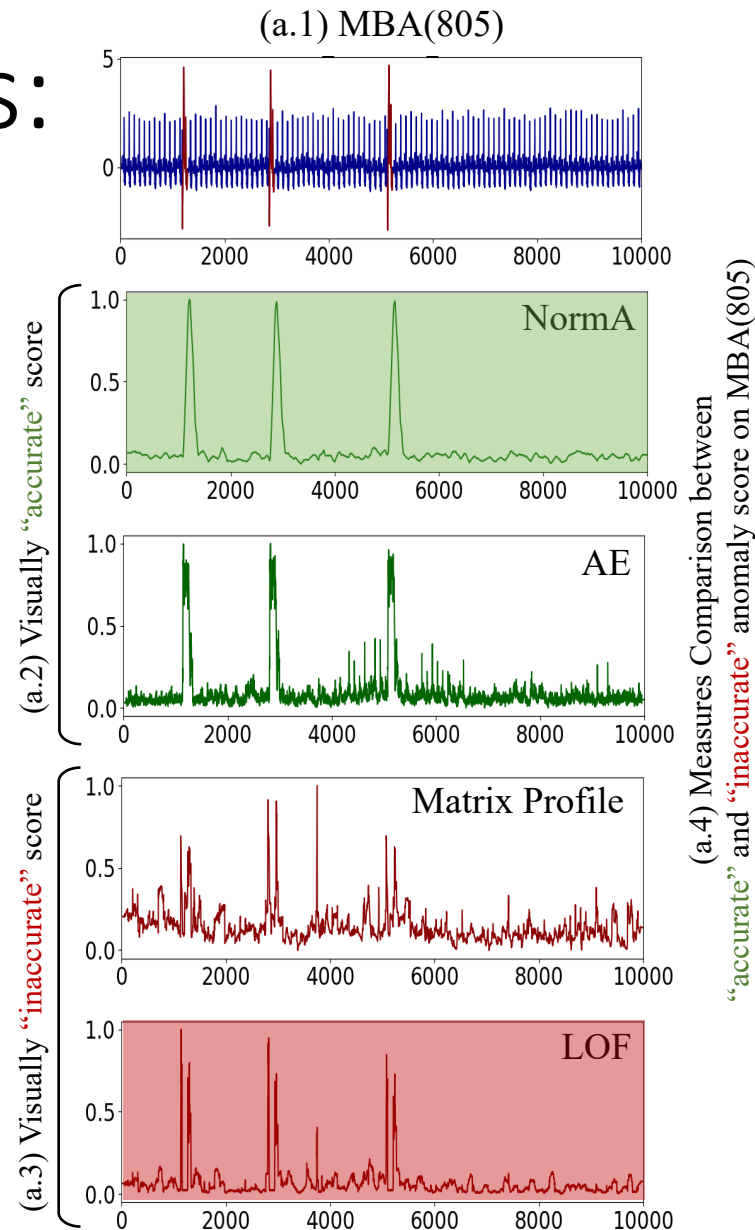
But how do we identify accurate methods without accuracy measures?



Evaluating measures: About Separability...

Separability: *An accuracy value of an accurate method should be easy to differentiate from an accurate value of an inaccurate method.*

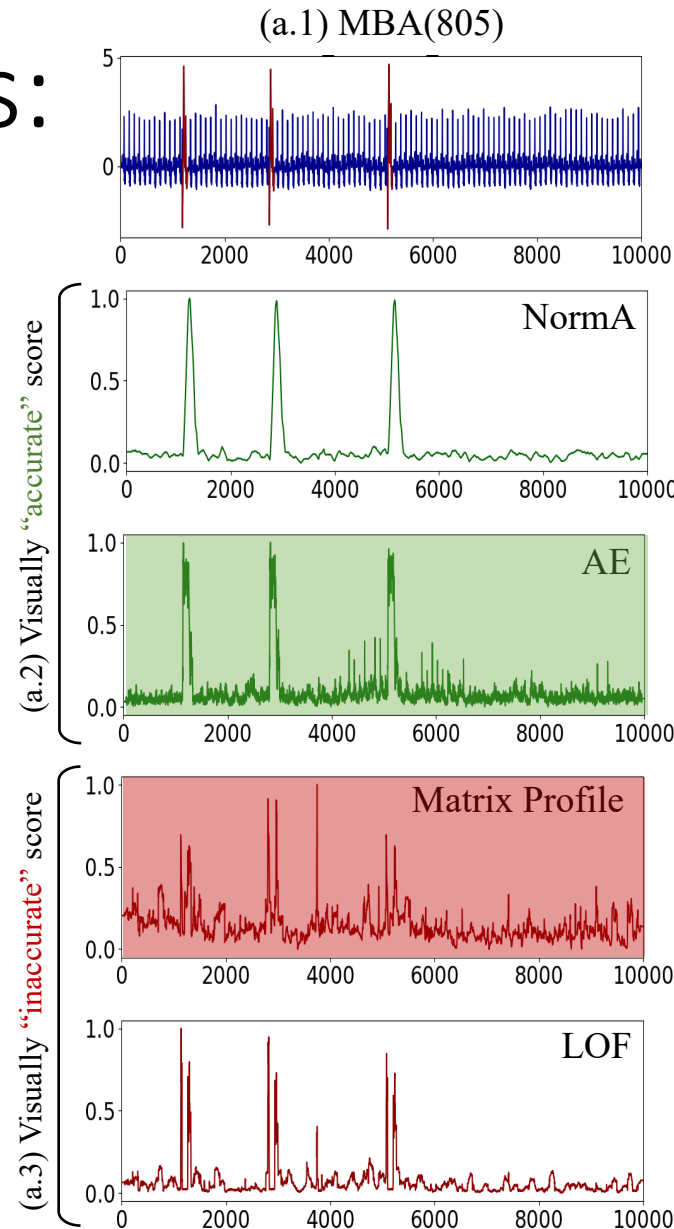
But how do we identify accurate methods without accuracy measures?



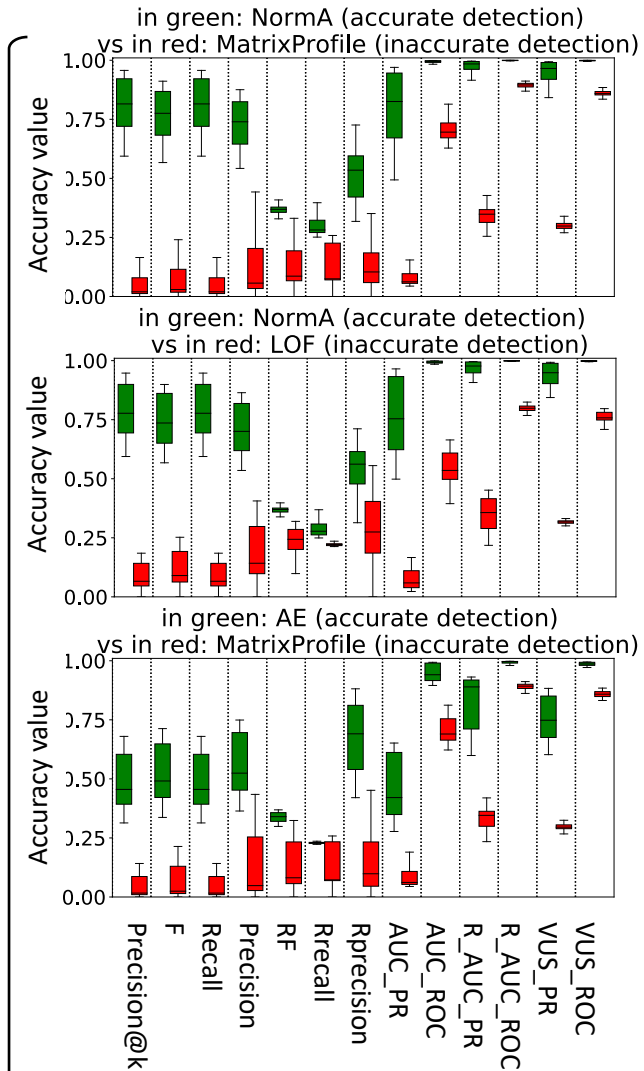
Evaluating measures: About Separability...

Separability: *An accuracy value of an accurate method should be easy to differentiate from an accurate value of an inaccurate method.*

But how do we identify accurate methods without accuracy measures?



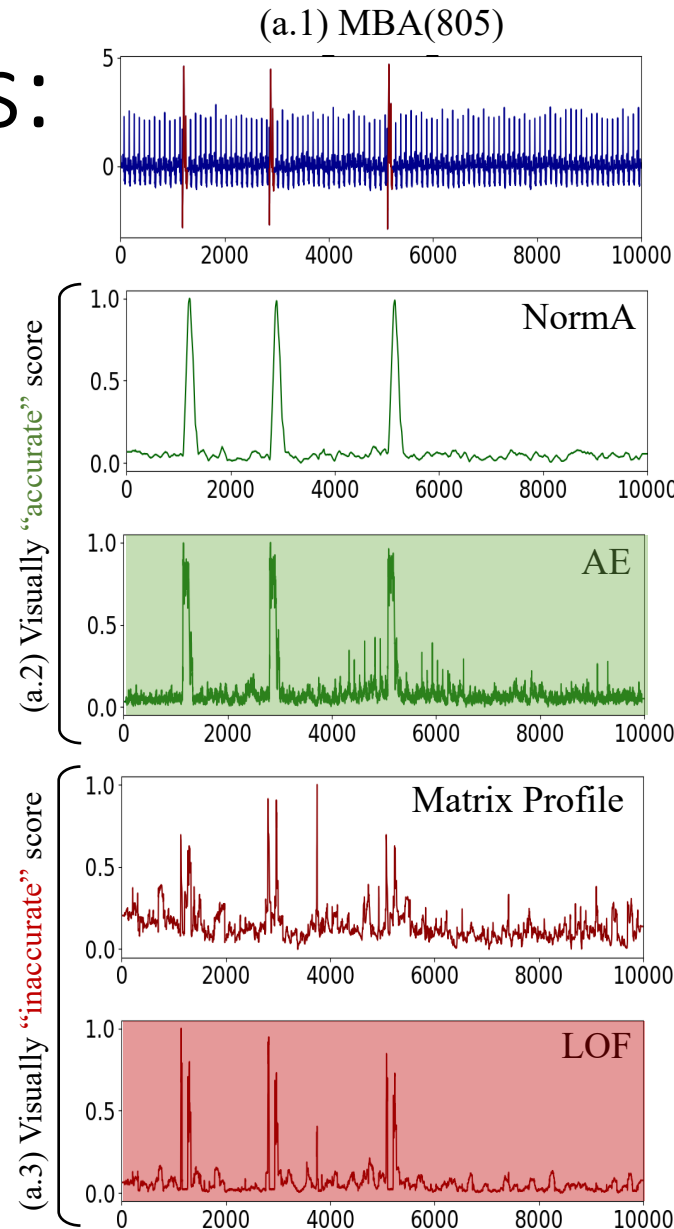
(a.4) Measures Comparison between
“accurate” and “inaccurate” anomaly score on MBA(805)



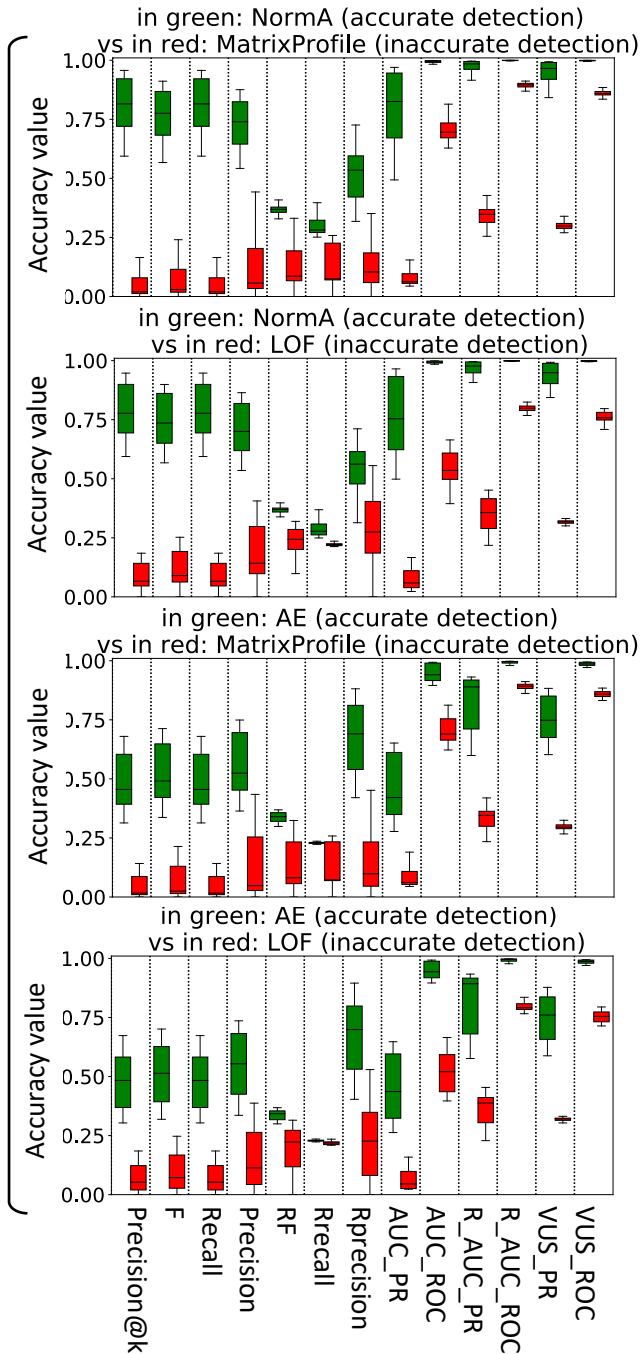
Evaluating measures: About Separability...

Separability: *An accuracy value of an accurate method should be easy to differentiate from an accurate value of an inaccurate method.*

But how do we identify accurate methods without accuracy measures?



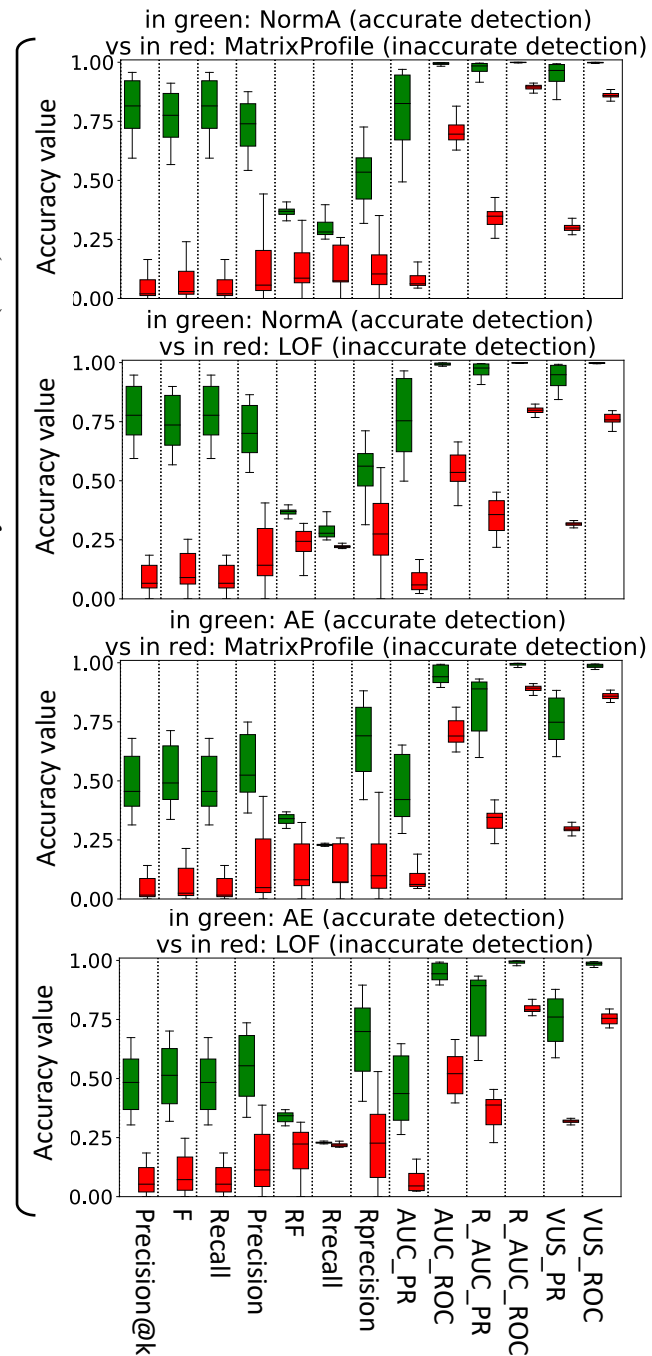
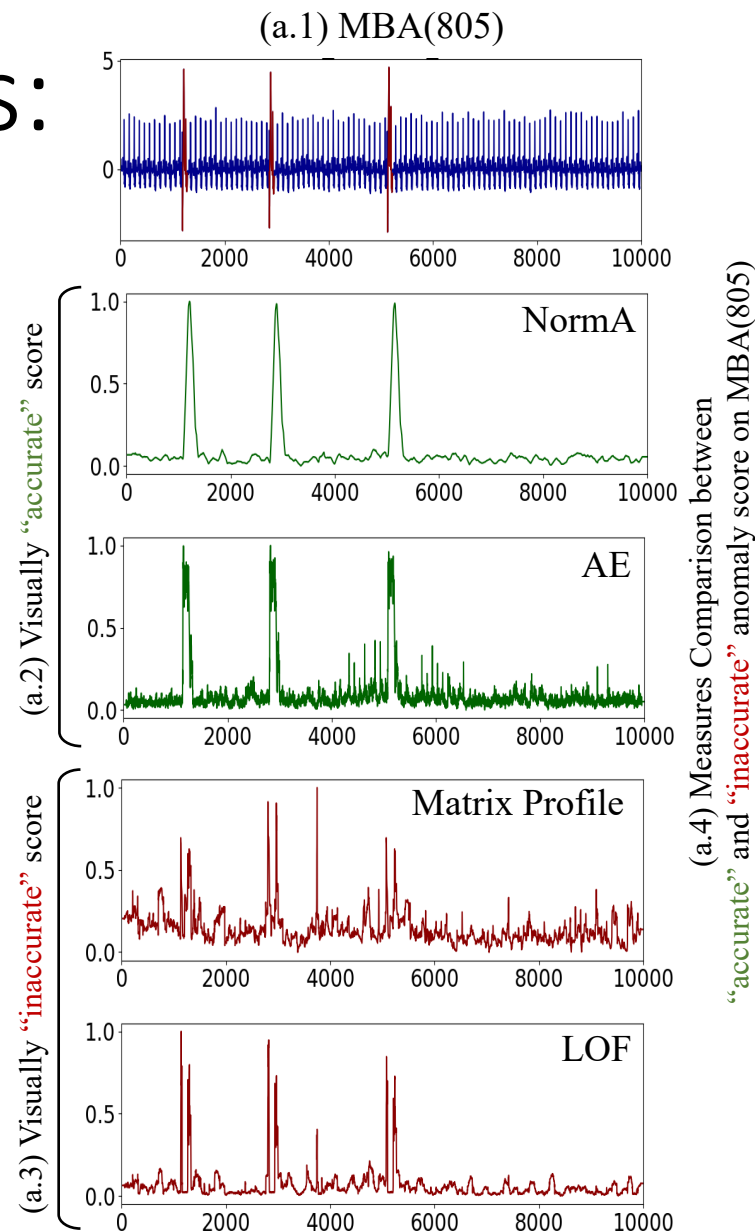
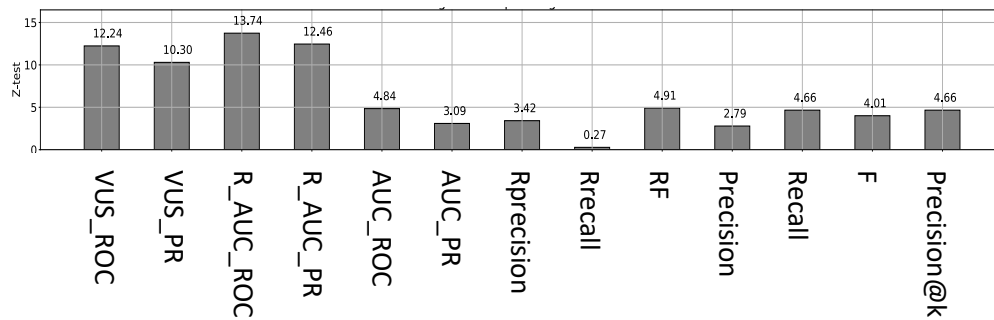
(a.4) Measures Comparison between
"accurate" and "inaccurate" anomaly score on MBA(805)



Evaluating measures: About Separability...

Separability: *An accuracy value of an accurate method should be easy to differentiate from an accurate value of an inaccurate method.*

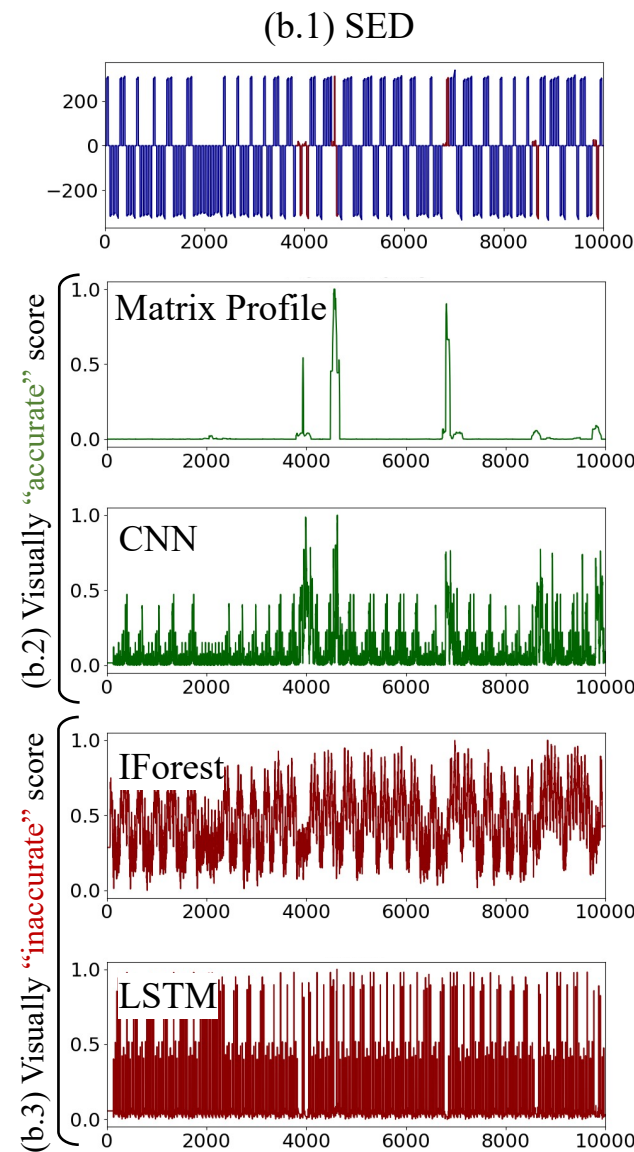
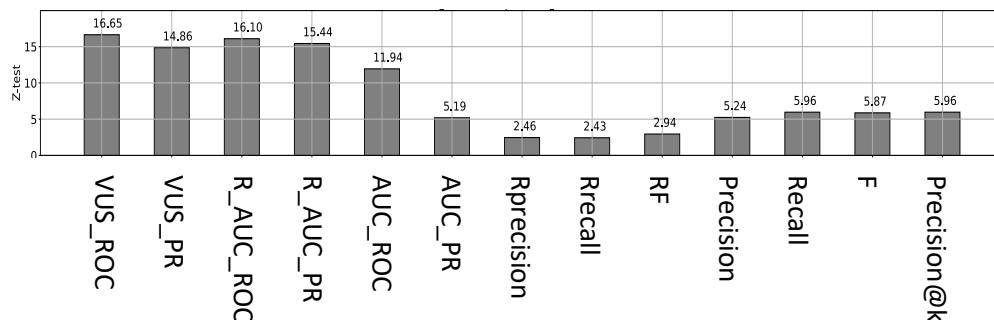
We compute the z-test between the accurate and inaccurate distribution scores.



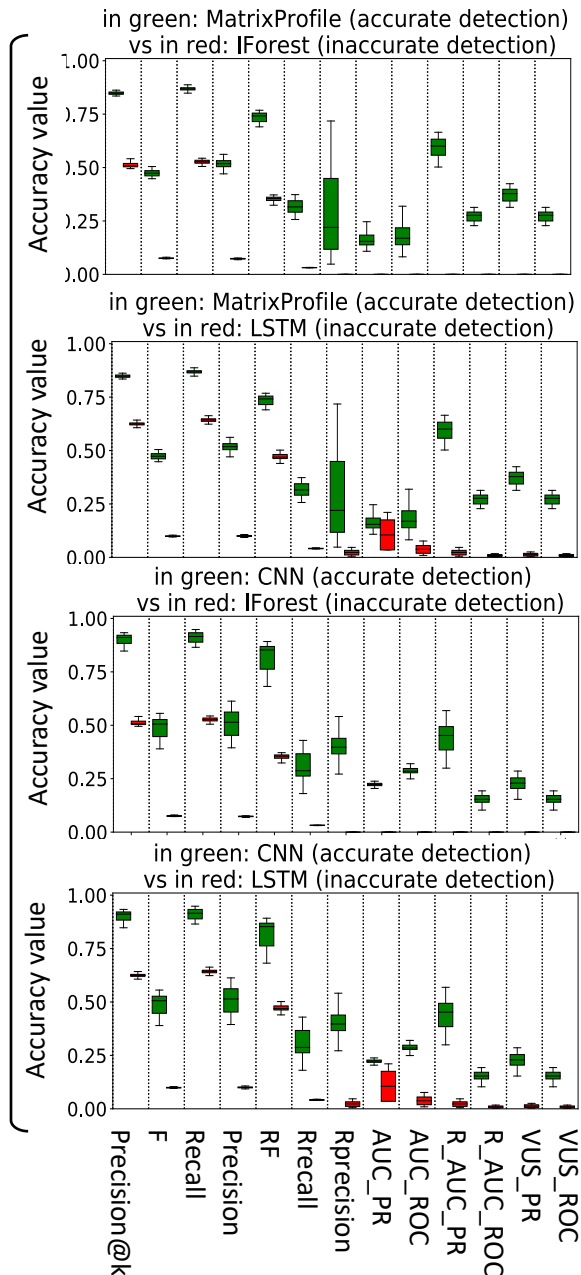
Evaluating measures: About Separability...

Separability: *An accuracy value of an accurate method should be easy to differentiate from an accurate value of an inaccurate method.*

We compute the z-test between the accurate and inaccurate distribution scores.



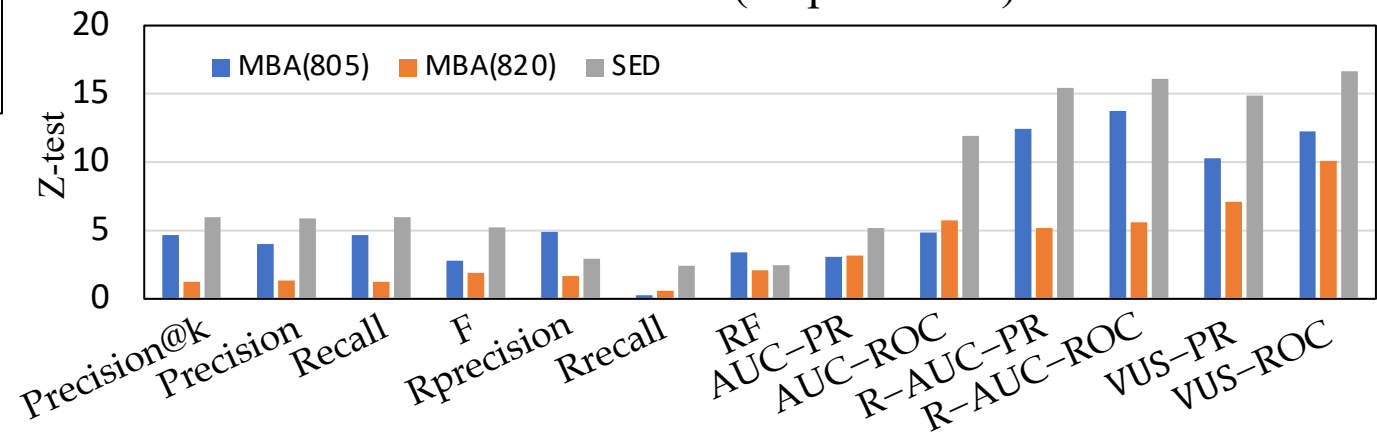
(b.4) Measures Comparison between "accurate" and "inaccurate" anomaly score on SED



Evaluating measures: About Separability...

Separability: *An accuracy value of an accurate method should be easy to differentiate from an accurate value of an inaccurate method.*

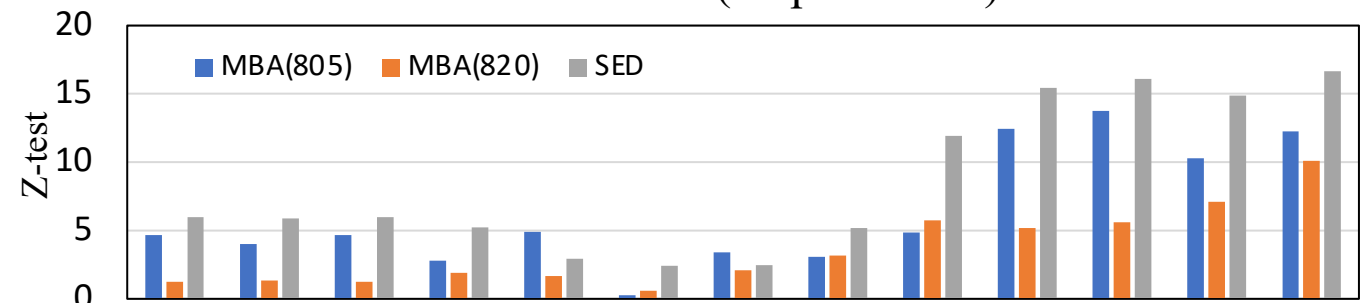
(a) Separability evaluation on MBA(805,820) and SED datasets (12 pairs each)



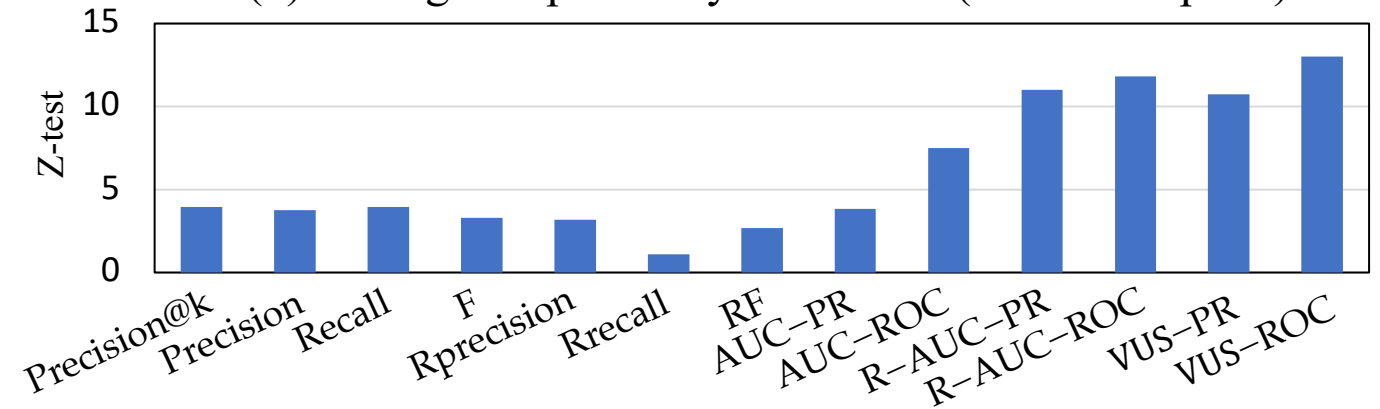
Evaluating measures: About Separability...

Separability: *An accuracy value of an accurate method should be easy to differentiate from an accurate value of an inaccurate method.*

(a) Separability evaluation on MBA(805,820) and SED datasets (12 pairs each)



(b) Averaged separability evaluation (total of 36 pairs)



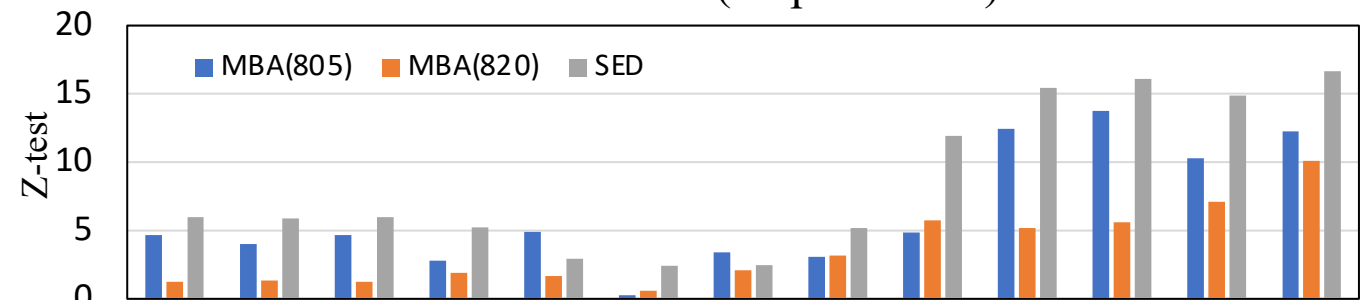
Evaluating measures: About Separability...

Separability: *An accuracy value of an accurate method should be easy to differentiate from an accurate value of an inaccurate method.*

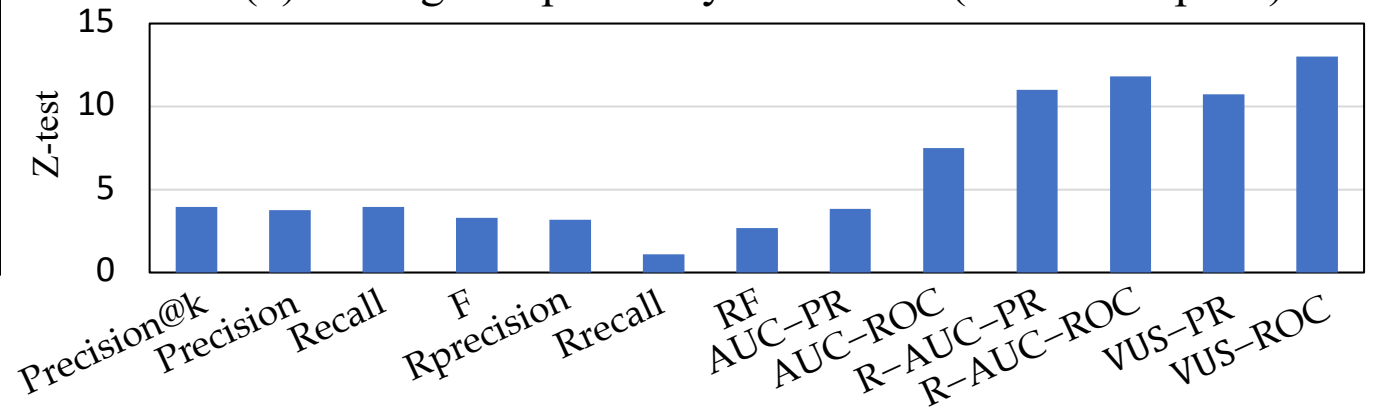
Current Problems:

- *The selection of accurate and inaccurate is subjective*
- *Can only be performed for a few time series*

(a) Separability evaluation on MBA(805,820) and SED datasets (12 pairs each)

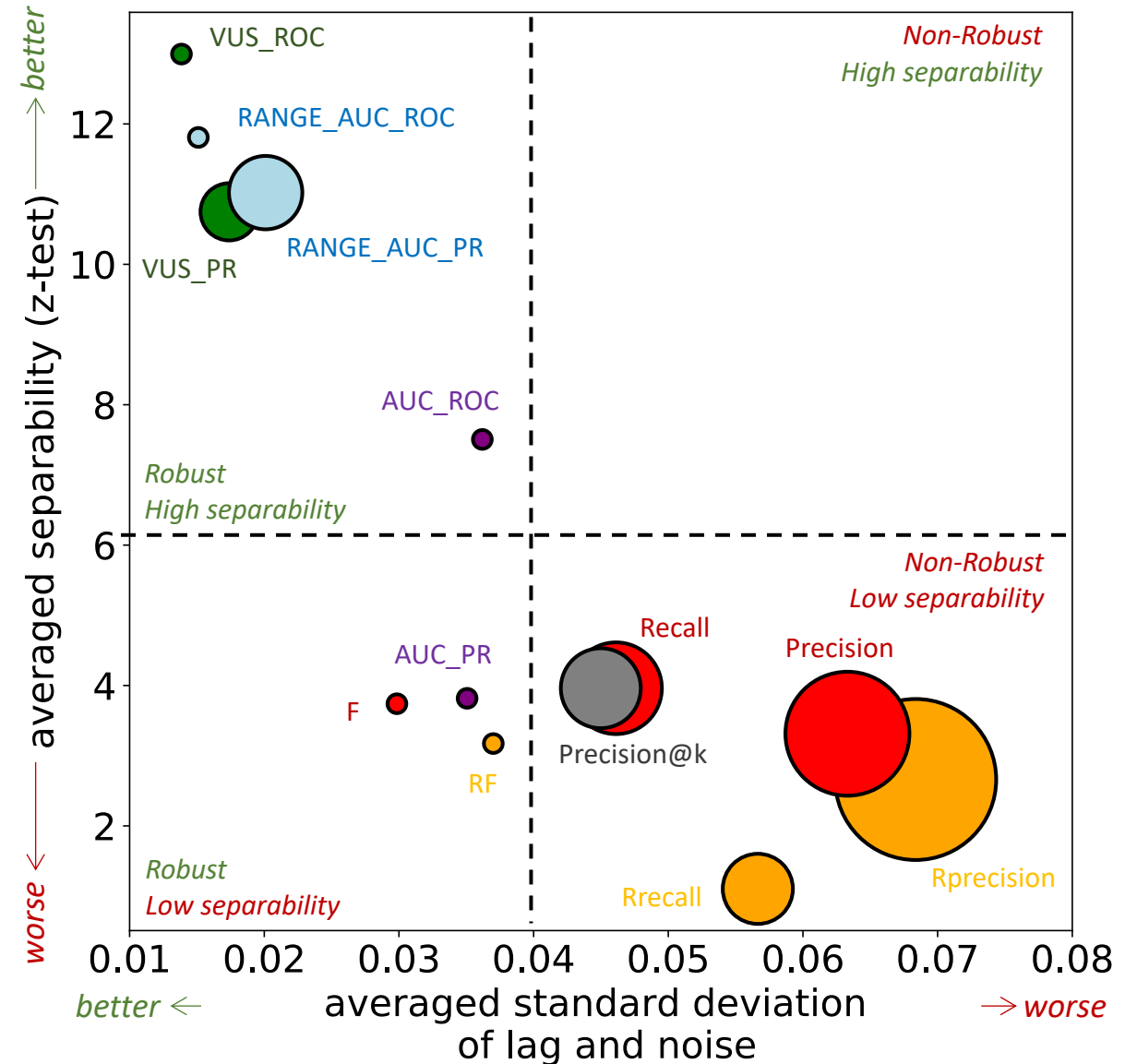


(b) Averaged separability evaluation (total of 36 pairs)



Evaluating measures: Robustness and Separability...

Both robust and separable measures are desirable

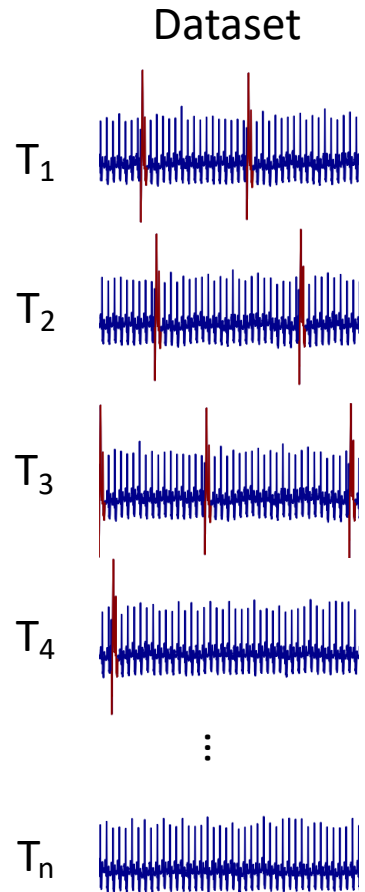


Evaluating measures: About Consistency...

Consistency: On similar time series, the ranking of methods provided by an accuracy measure should be similar

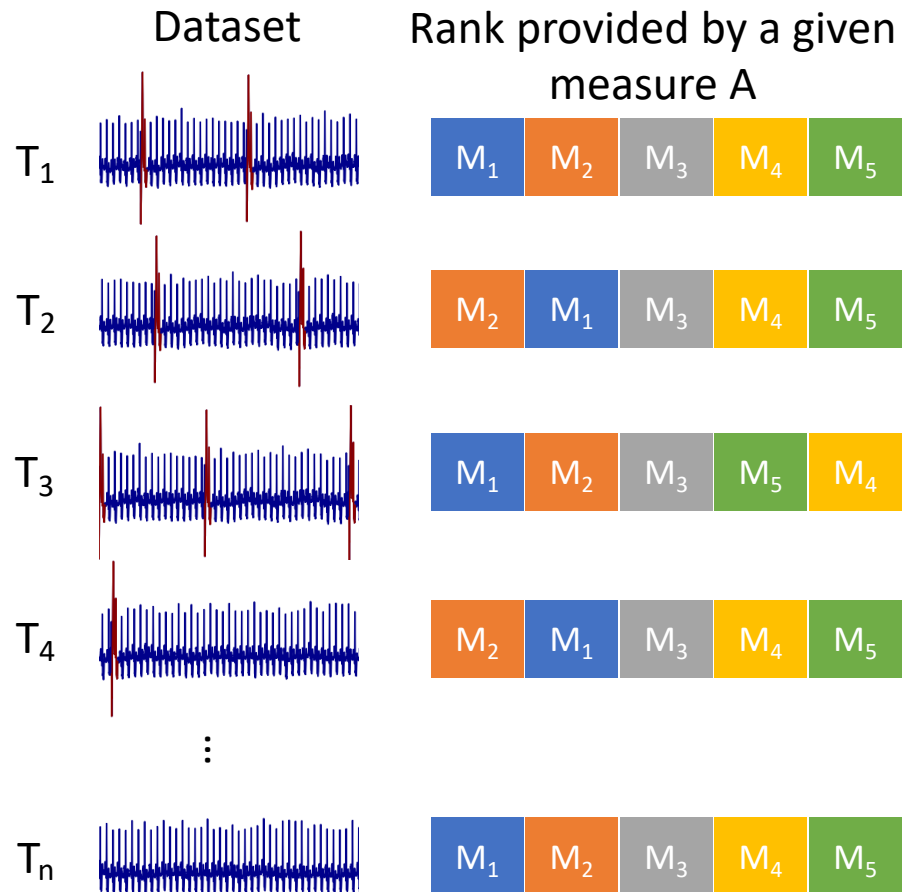
Evaluating measures: About Consistency...

Consistency: On similar time series, the ranking of methods provided by an accuracy measure should be similar



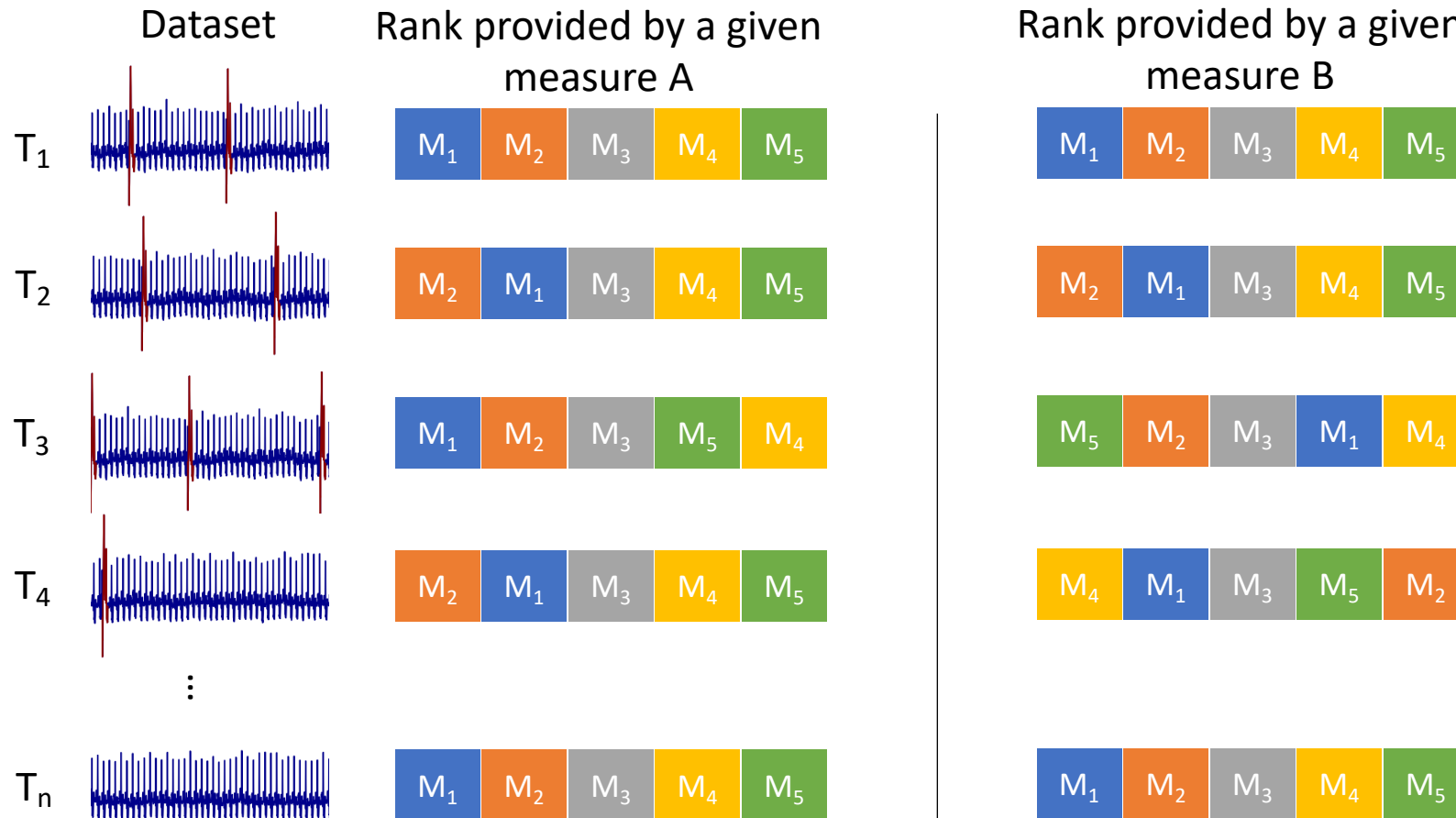
Evaluating measures: About Consistency...

Consistency: On similar time series, the ranking of methods provided by an accuracy measure should be similar



Evaluating measures: About Consistency...

Consistency: On similar time series, the ranking of methods provided by an accuracy measure should be similar



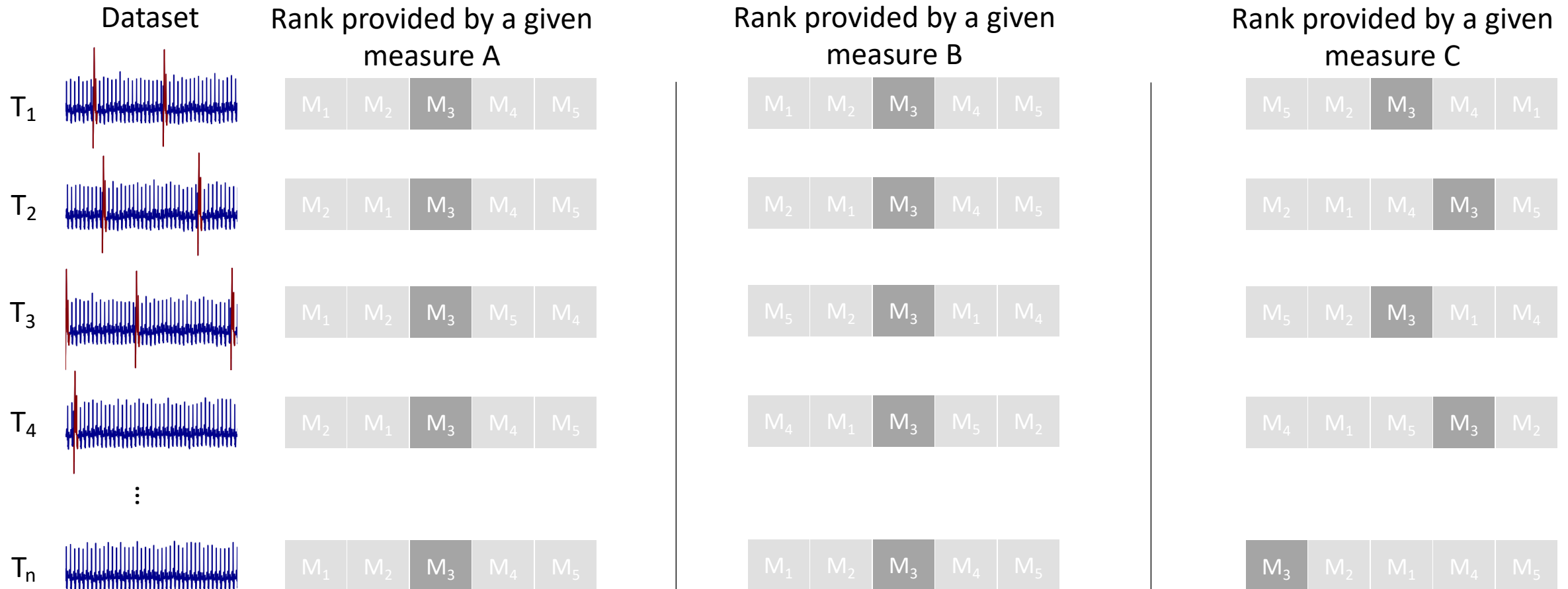
Evaluating measures: About Consistency...

Consistency: On similar time series, the ranking of methods provided by an accuracy measure should be similar



Evaluating measures: About Consistency...

Consistency: On similar time series, the ranking of methods provided by an accuracy measure should be similar



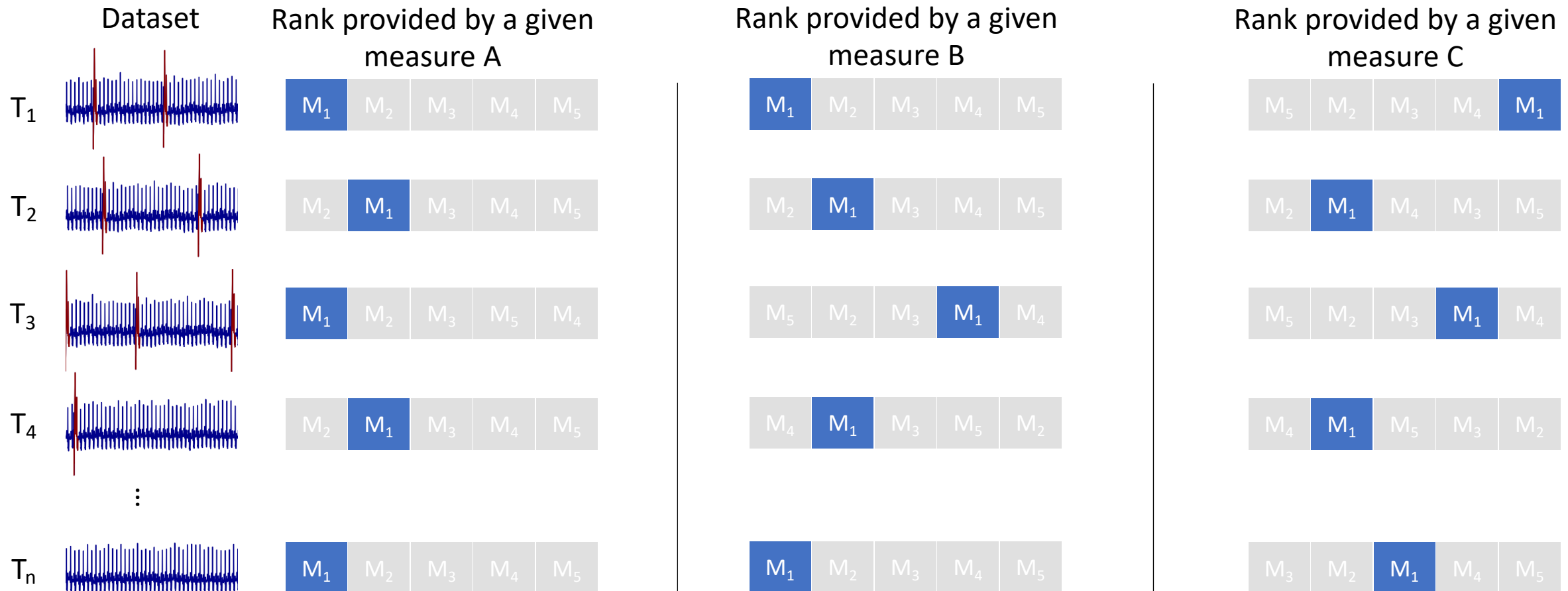
Evaluating measures: About Consistency...

Consistency: On similar time series, the ranking of methods provided by an accuracy measure should be similar



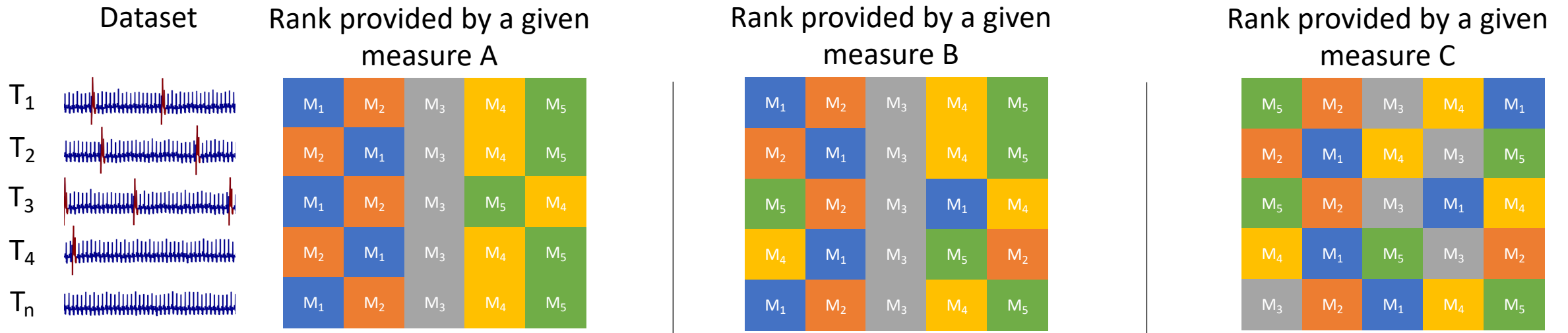
Evaluating measures: About Consistency...

Consistency: On similar time series, the ranking of methods provided by an accuracy measure should be similar



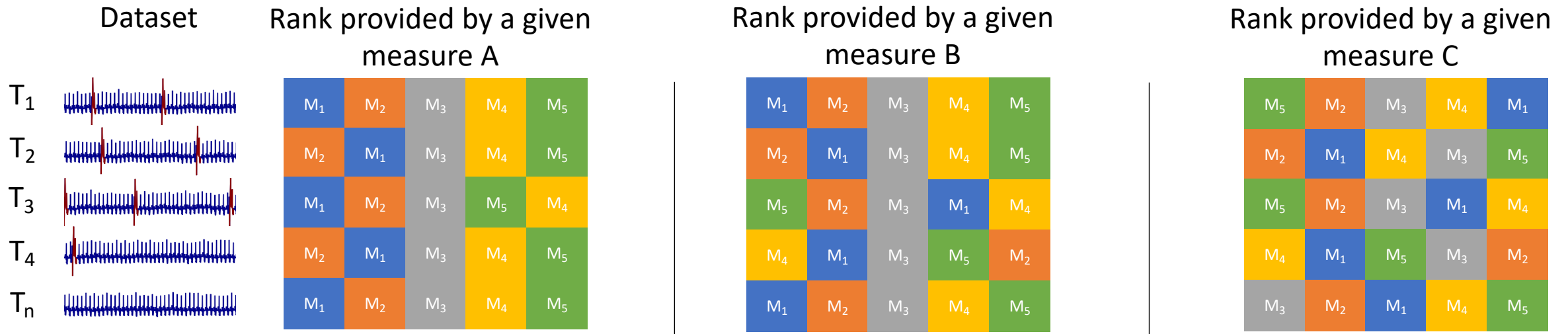
Evaluating measures: About Consistency...

Consistency: On similar time series, the ranking of methods provided by an accuracy measure should be similar



Evaluating measures: About Consistency...

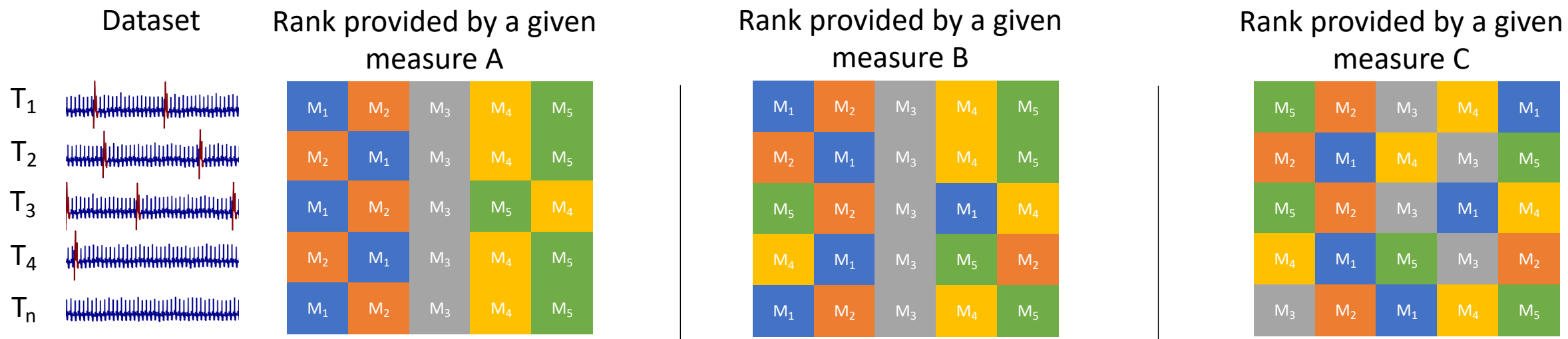
Consistency: On similar time series, the ranking of methods provided by an accuracy measure should be similar



For a dataset (that we assume contains similar time series), we average the Entropy H of the rank per methods

Evaluating measures: About Consistency...

Consistency: On similar time series, the ranking of methods provided by an accuracy measure should be similar

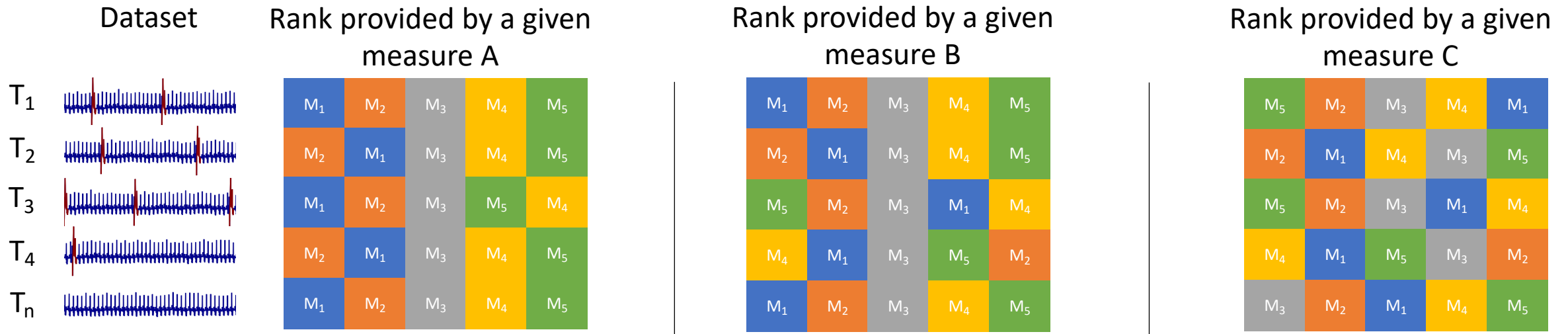


For a dataset (that we assume contains similar time series), we average the Entropy H of the rank per methods

$$\frac{1}{|M|} \sum_i H(\text{Rank}_A(M_i)) < \frac{1}{|M|} \sum_i H(\text{Rank}_B(M_i)) < \frac{1}{|M|} \sum_i H(\text{Rank}_C(M_i))$$

Evaluating measures: About Consistency...

Consistency: On similar time series, the ranking of methods provided by an accuracy measure should be similar



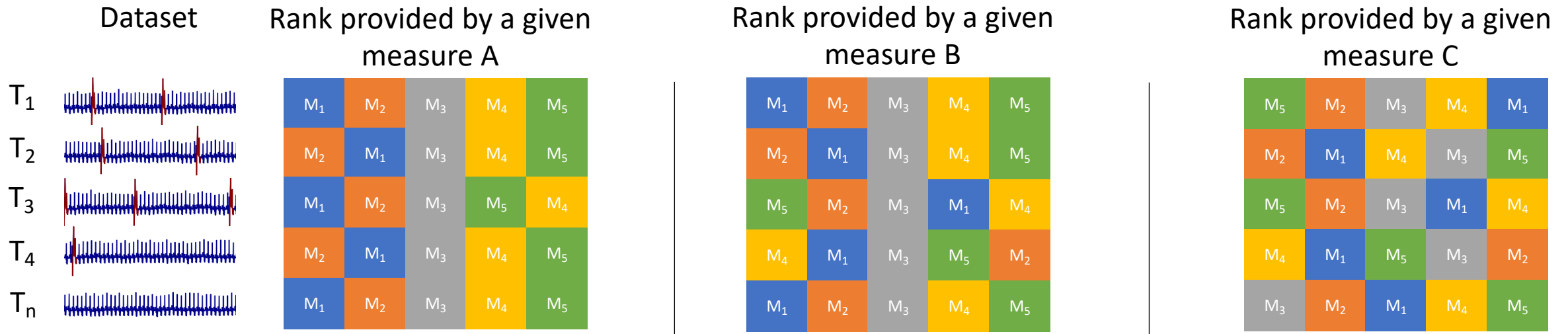
For a dataset (that we assume contains similar time series), we average the Entropy H of the rank per methods

$$\frac{1}{|M|} \sum_i H(\text{Rank}_A(M_i)) < \frac{1}{|M|} \sum_i H(\text{Rank}_B(M_i)) < \frac{1}{|M|} \sum_i H(\text{Rank}_C(M_i))$$

A is more consistent than B, which is more consistent than C

Evaluating measures: About Consistency...

Consistency: On similar time series, the ranking of methods provided by an accuracy measure should be similar



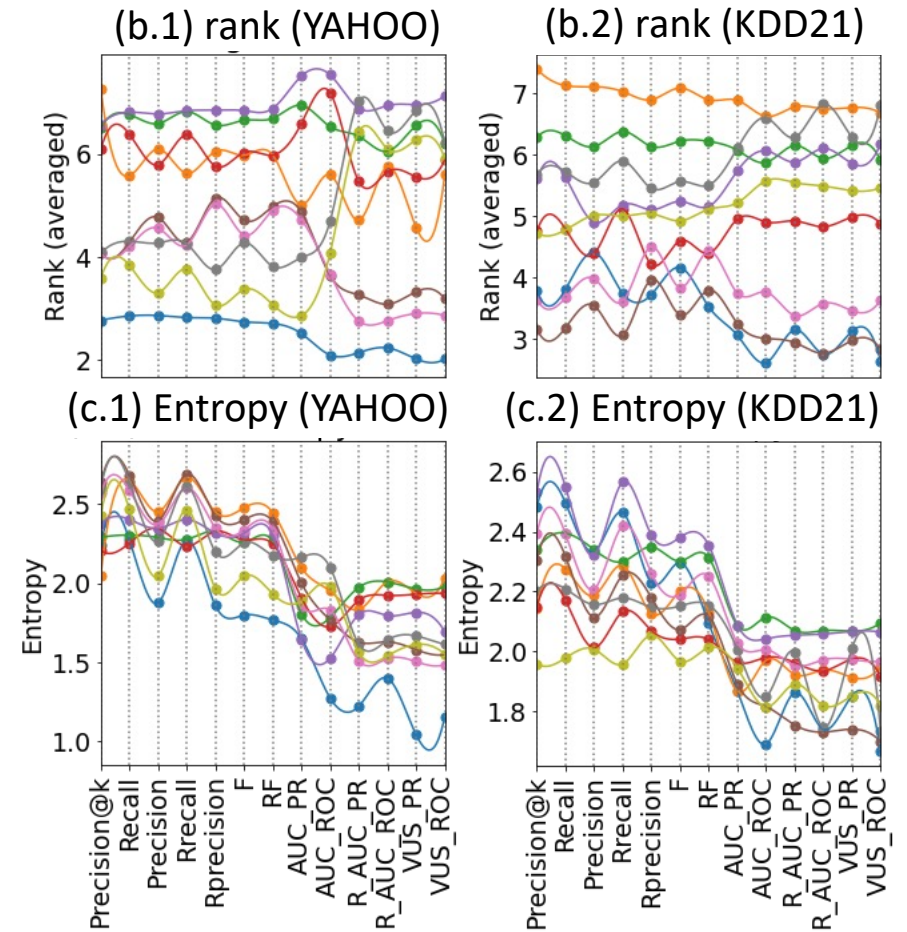
For a dataset **(that we assume contains similar time series)**, we average the Entropy H of the rank per methods

$$\frac{1}{|M|} \sum_i H(\text{Rank}_A(M_i)) < \frac{1}{|M|} \sum_i H(\text{Rank}_B(M_i)) < \frac{1}{|M|} \sum_i H(\text{Rank}_C(M_i))$$

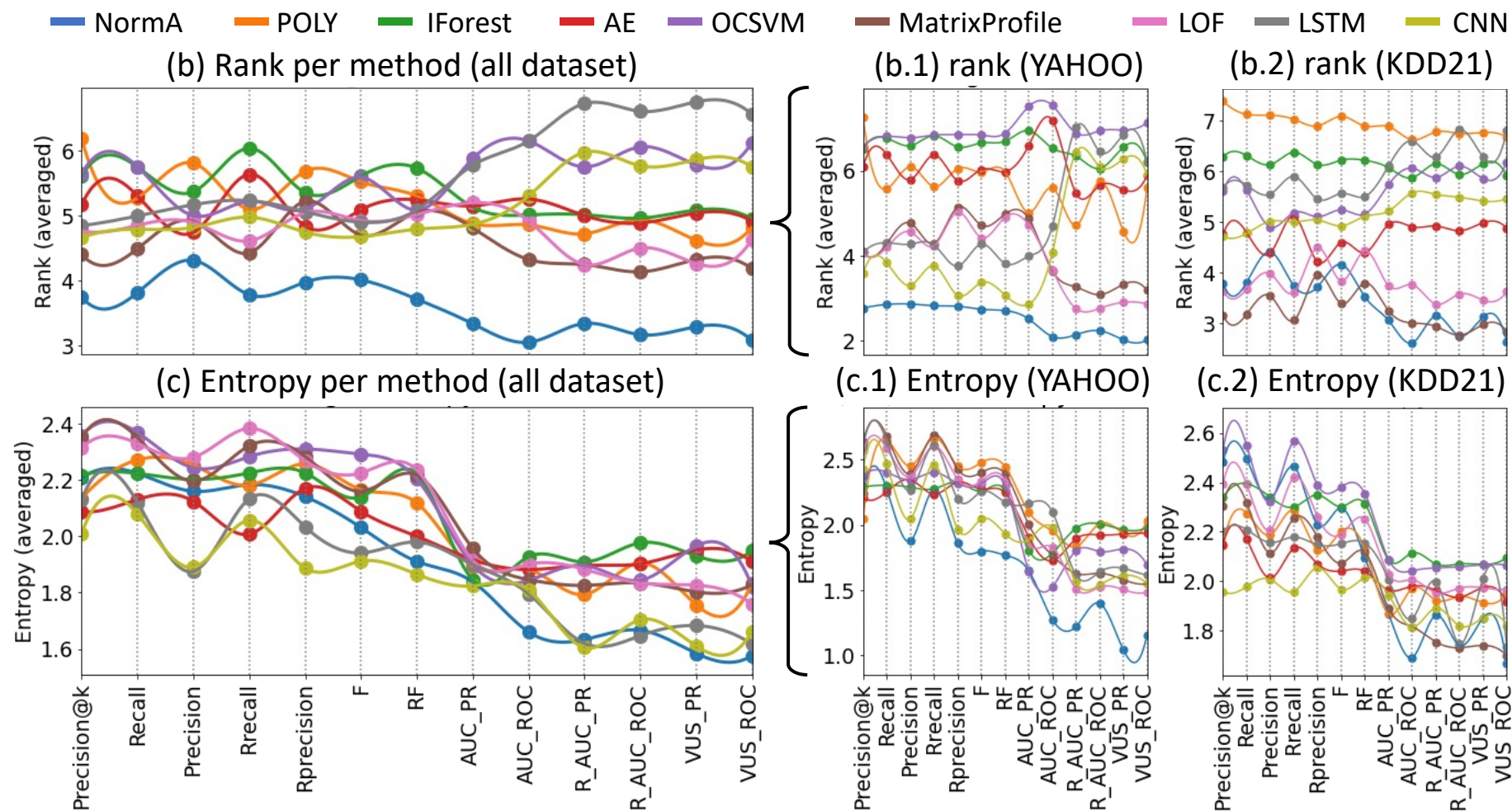
A is more consistent than B, which is more consistent than C

Evaluating measures: About Consistency...

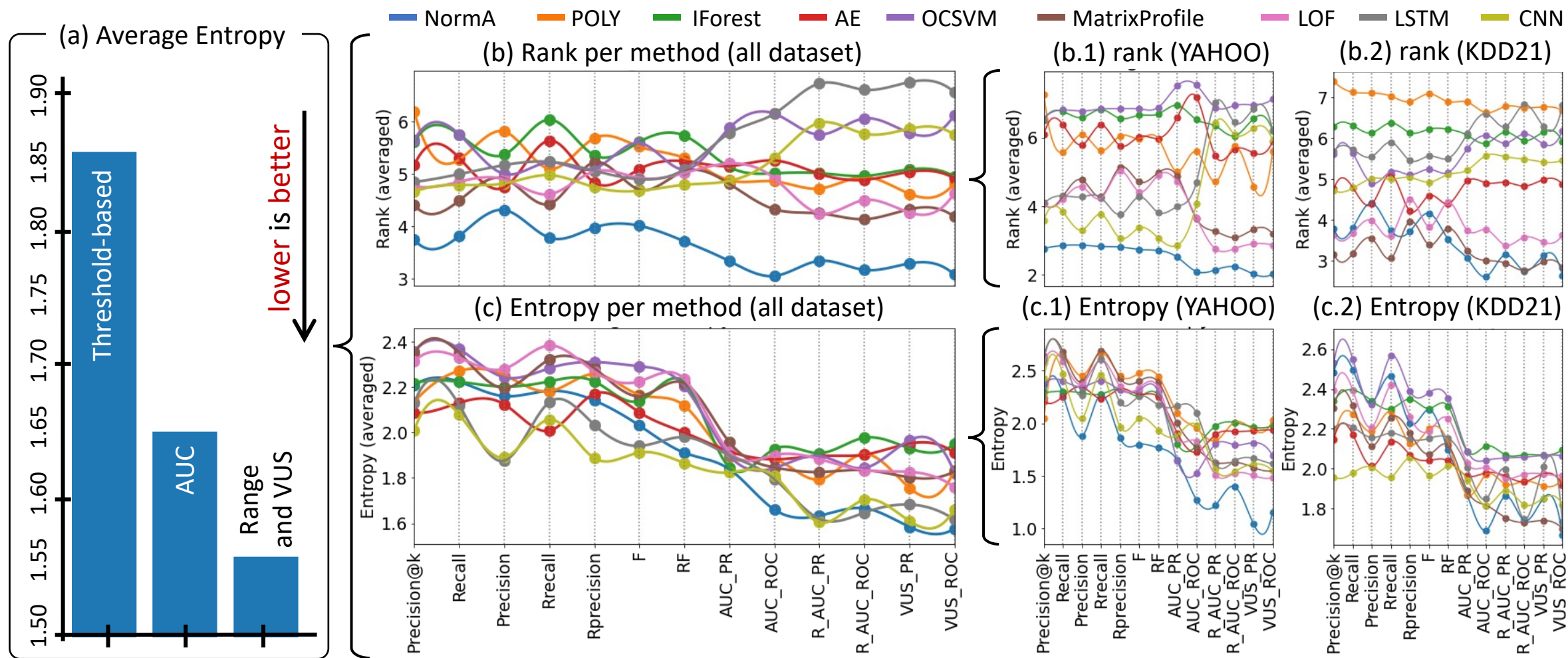
NormA POLY IForest AE OCSVM MatrixProfile LOF LSTM CNN



Evaluating measures: About Consistency...



Evaluating measures: About Consistency...



Part 4: Anomaly Detection Benchmarks

Anomaly Detection methods: *Existing benchmark*

HEX/UCR [18]

Set of **250 time series** with labels.

Details

- The anomaly labels are synthetically created
- Each time series contains only 1 labeled anomaly

Anomaly Detection methods: *Existing benchmark*

HEX/UCR [18]

Set of **250 time series** with labels.

Details

- The anomaly labels are synthetically created
- Each time series contains only 1 labeled anomaly

TimeEval [5]

Set of **976 time series** with labels.

Details

- New synthetic benchmark GutenTag used to tune parameters
- Only Time series with low contamination rate (< 0.1)
- Time series with at least one methods above 0.8 AUC-ROC

Anomaly Detection methods: *Existing benchmark*

HEX/UCR [18]

Set of **250 time series** with labels.

Details

- The anomaly labels are synthetically created
- Each time series contains only 1 labeled anomaly

TimeEval [5]

Set of **976 time series** with labels.

Details

- New synthetic benchmark GutenTag used to tune parameters
- Only Time series with low contamination rate (< 0.1)
- Time series with at least one methods above 0.8 AUC-ROC

TSB-UAD [19]

Set of **2000 time series** with labels.

Details

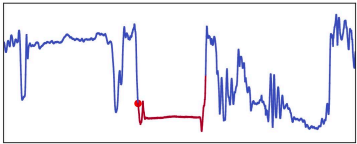
- Collected as proposed in the literature (no filtering based on contamination, size or label quality)
- Artificial and synthetic data generation methods for reliable labels

Anomaly Detection methods: *Existing benchmark*

HEX/UCR [18]

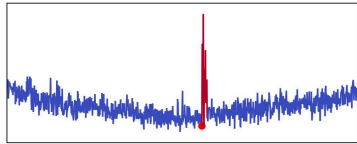
Set of 250 time series with

OPPORTUNITY

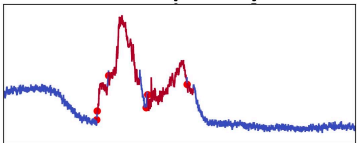


Occupancy

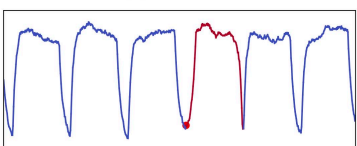
IOPS



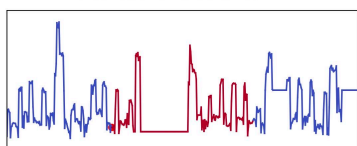
ECG



KDD21



NASA-SMAP

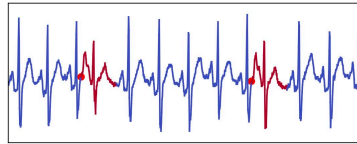


only 1 labeled anomaly

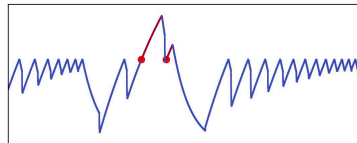
TimeEval [5]

Real datasets collection

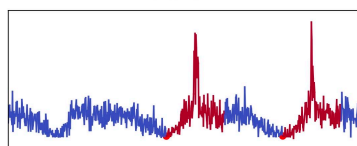
SVDB



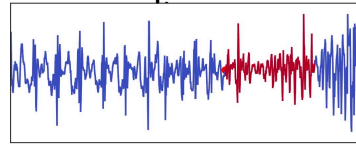
GHL



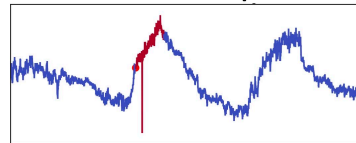
NAB



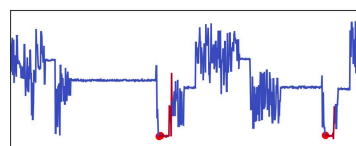
Daphnet



SensorScope



Genesis



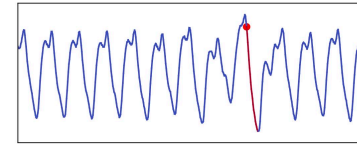
contamination rate (< 0.1)

- Time series with at least one methods above 0.8 AUC-ROC

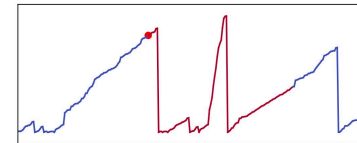
TSB-UAD [19]

Set of 2000 time series with

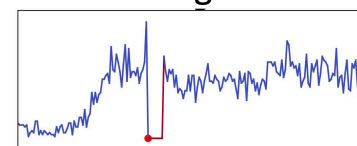
MGAB



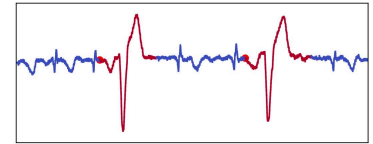
NASA-MSL



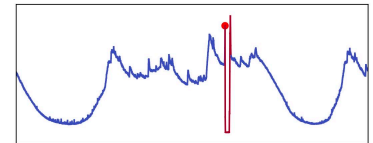
Dodgers



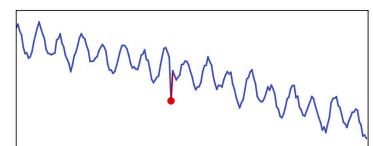
MITDB



SMD



YAHOO



contamination, size or label quality.

Anomaly Detection methods: *Existing benchmark*

HEX/UCR [18]

TimeEval [5]

TSB-UAD [19]

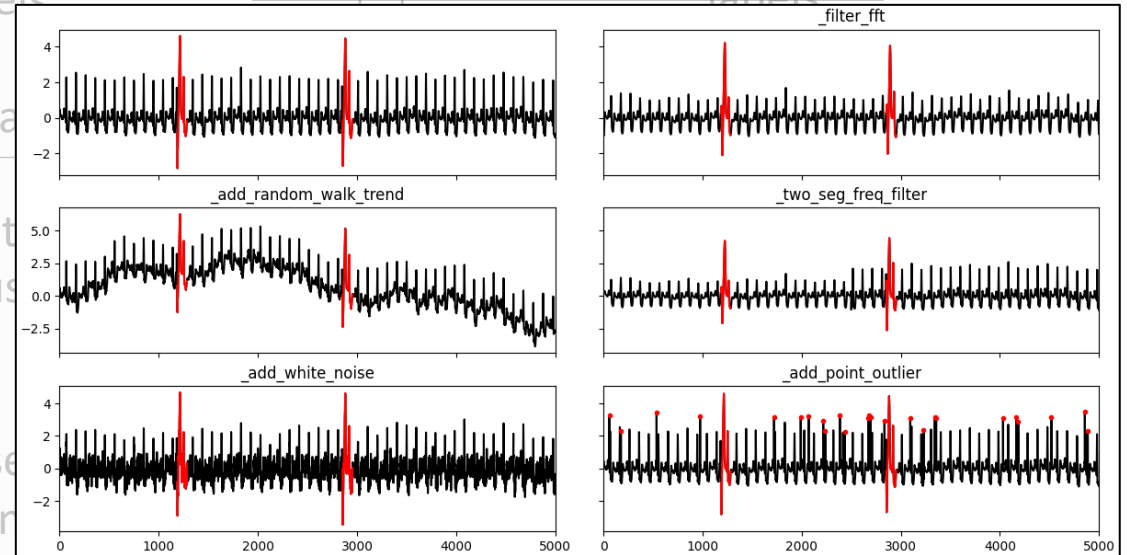
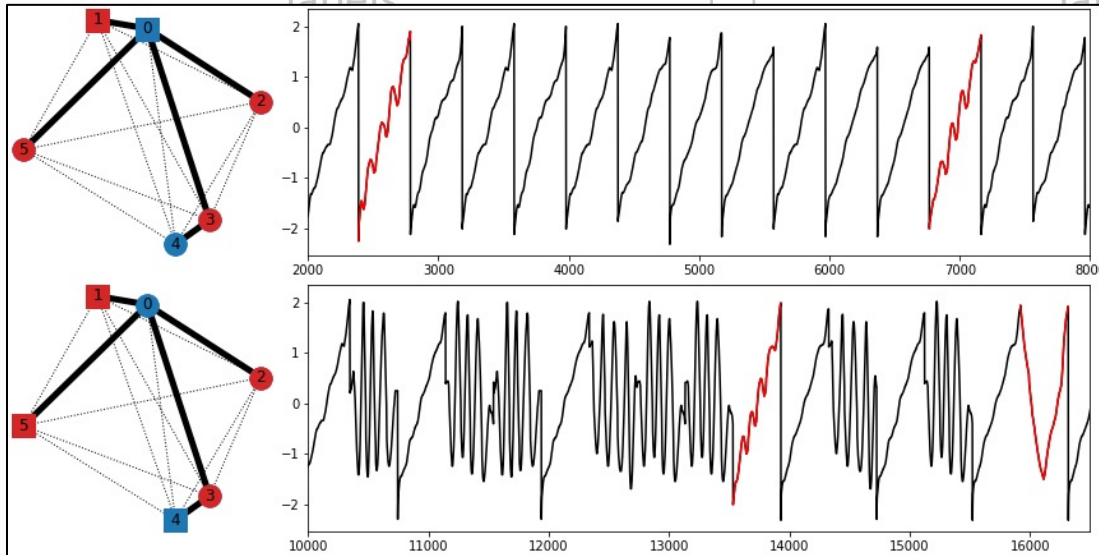
Set of 250

Artificial dataset generation

of 976 time series with labels

Synthetic dataset generation

ies with



- Time series with at least one methods above 0.8 AUC-ROC

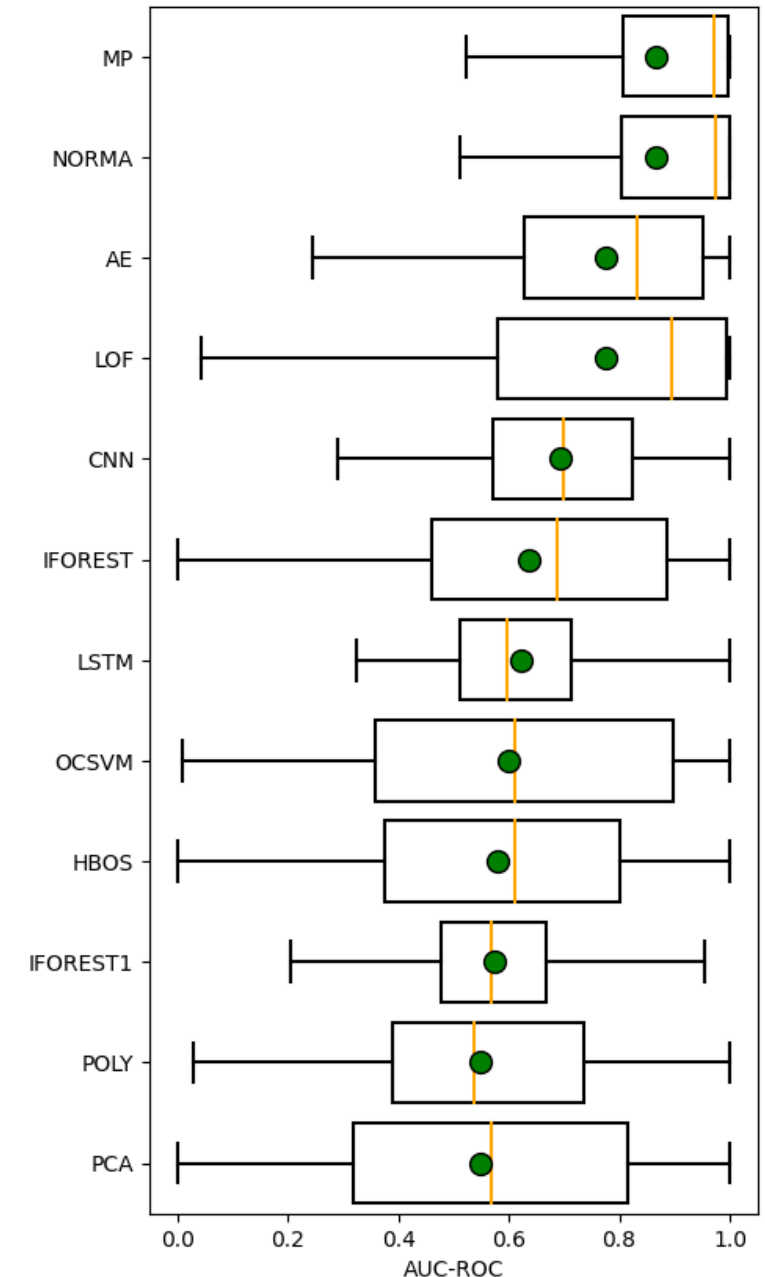
quality.

Anomaly Detection methods: *Experimental evaluation*

Observations on HEX/UCR [18]:

- Distance-based methods have a better accuracy (AUC-ROC) than forecasting and distribution-based approaches

[18] R. Wu and E. Keogh, "Current Time Series Anomaly Detection Benchmarks are Flawed and are Creating the Illusion of Progress" in IEEE Transactions on Knowledge & Data Engineering, vol. 35, no. 03, pp. 2421-2429, 2023.

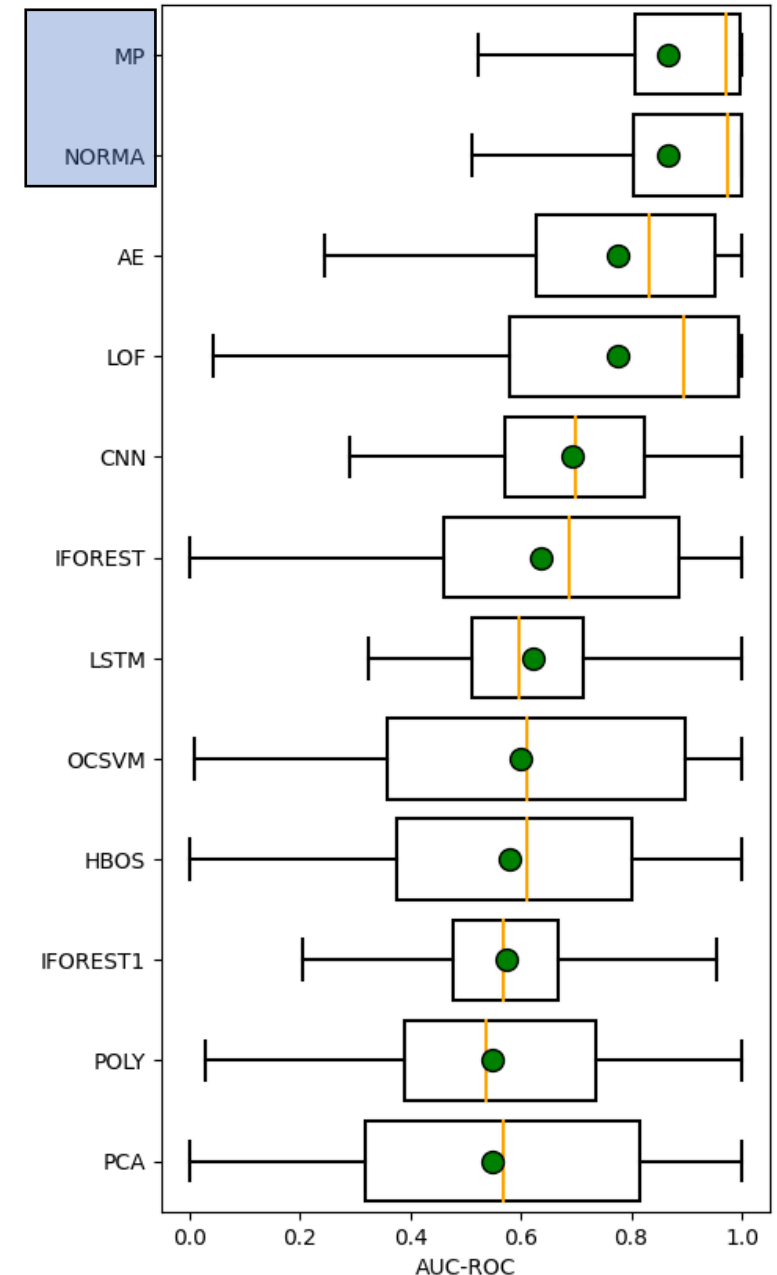


Anomaly Detection methods: *Experimental evaluation*

Observations on HEX/UCR [18]:

- Distance-based methods have a better accuracy (AUC-ROC) than forecasting and distribution-based approaches

[18] R. Wu and E. Keogh, "Current Time Series Anomaly Detection Benchmarks are Flawed and are Creating the Illusion of Progress" in IEEE Transactions on Knowledge & Data Engineering, vol. 35, no. 03, pp. 2421-2429, 2023.

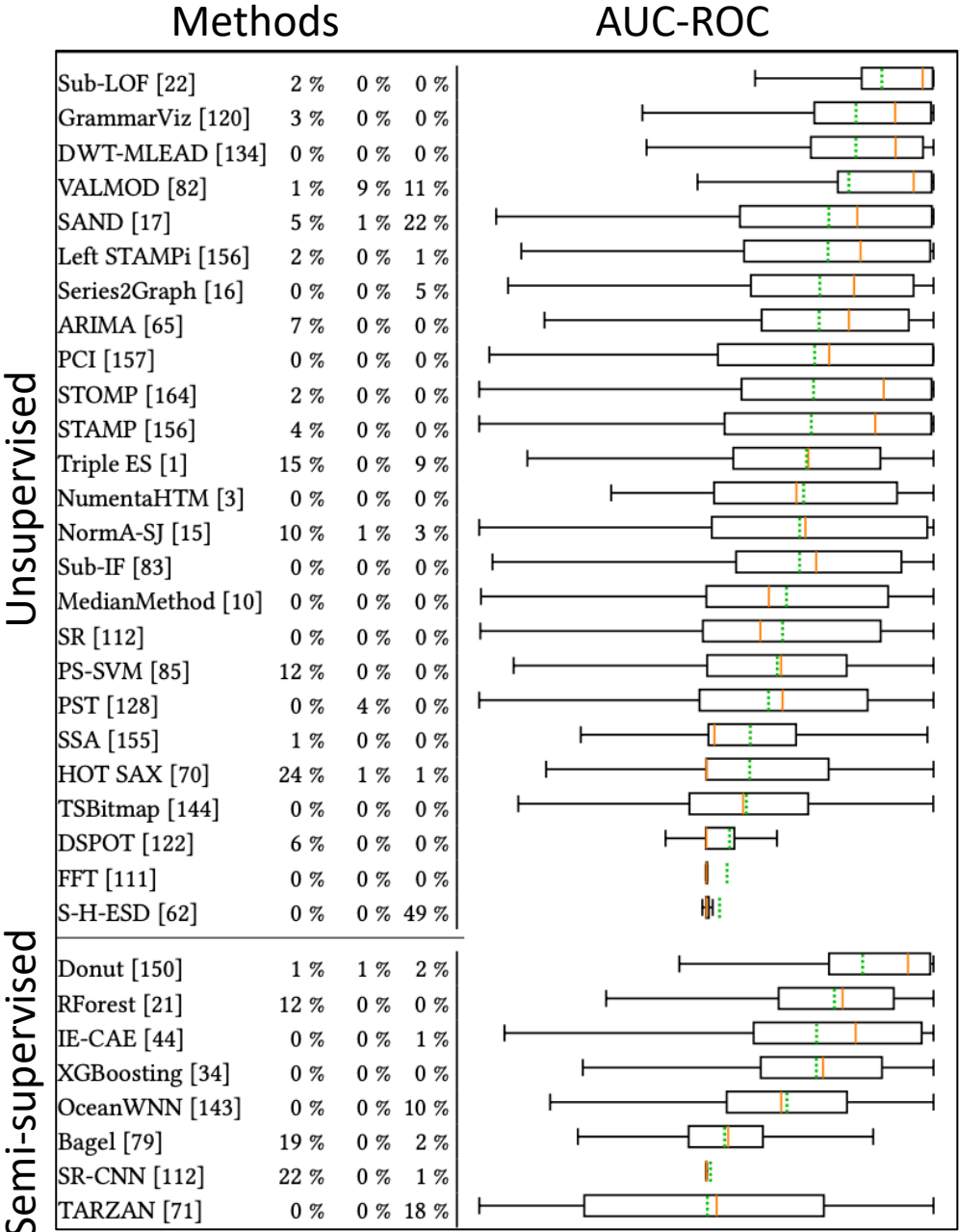


Anomaly Detection methods:

Experimental evaluation

Observations on TimeEval [5]:

[5] Sebastian Schmidl, Phillip Wenig, and Thorsten Papenbrock. 2022. Anomaly detection in time series: a comprehensive evaluation. Proc. VLDB Endow. 15, 9 (May 2022), 1779–1797.



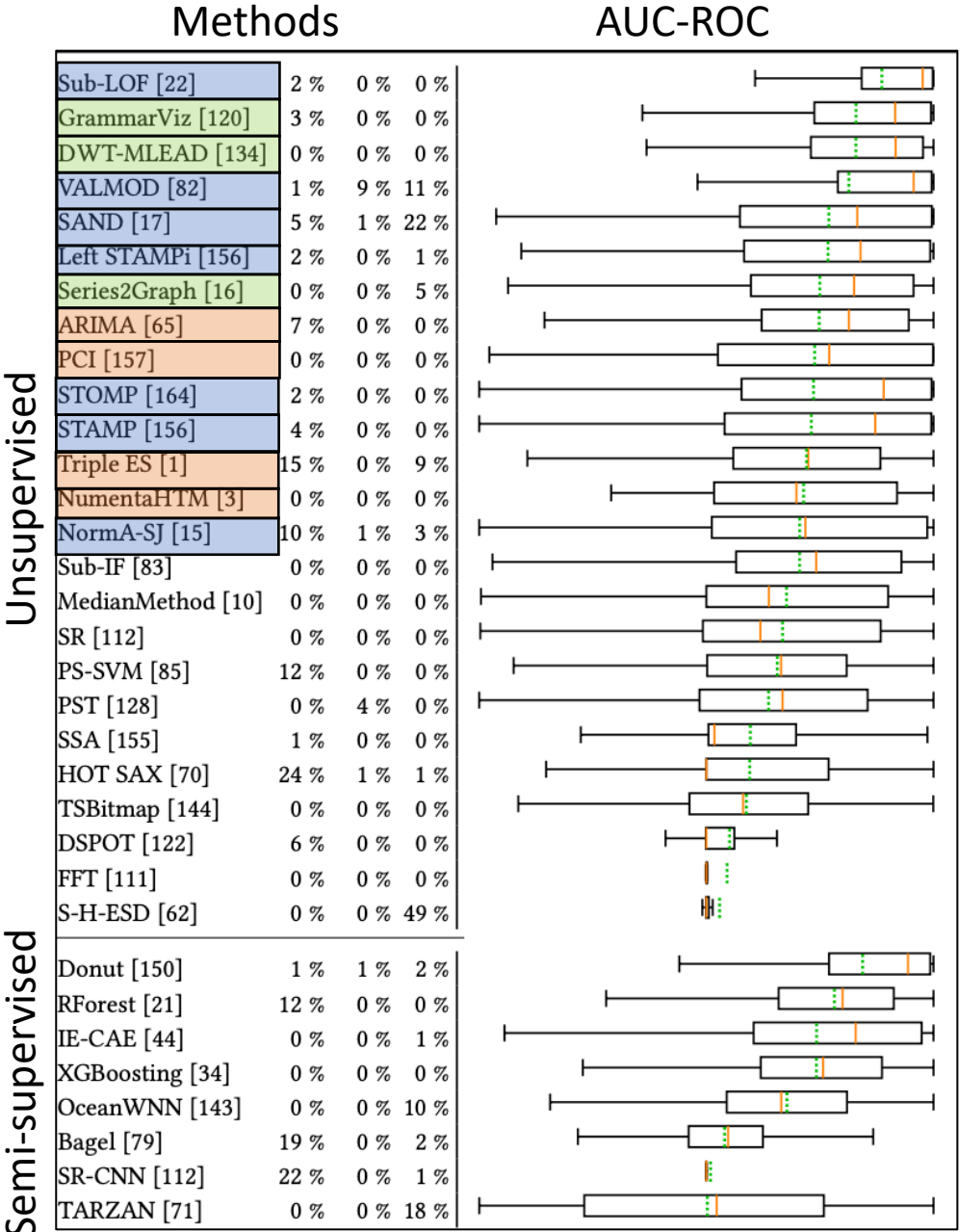
Anomaly Detection methods:

Experimental evaluation

Observations on TimeEval [5]:

- Distance-based and Density-based methods have a better accuracy (AUC-ROC) than forecasting and reconstruction-based approaches

[5] Sebastian Schmidl, Phillip Wenig, and Thorsten Papenbrock. 2022. Anomaly detection in time series: a comprehensive evaluation. Proc. VLDB Endow. 15, 9 (May 2022), 1779–1797.



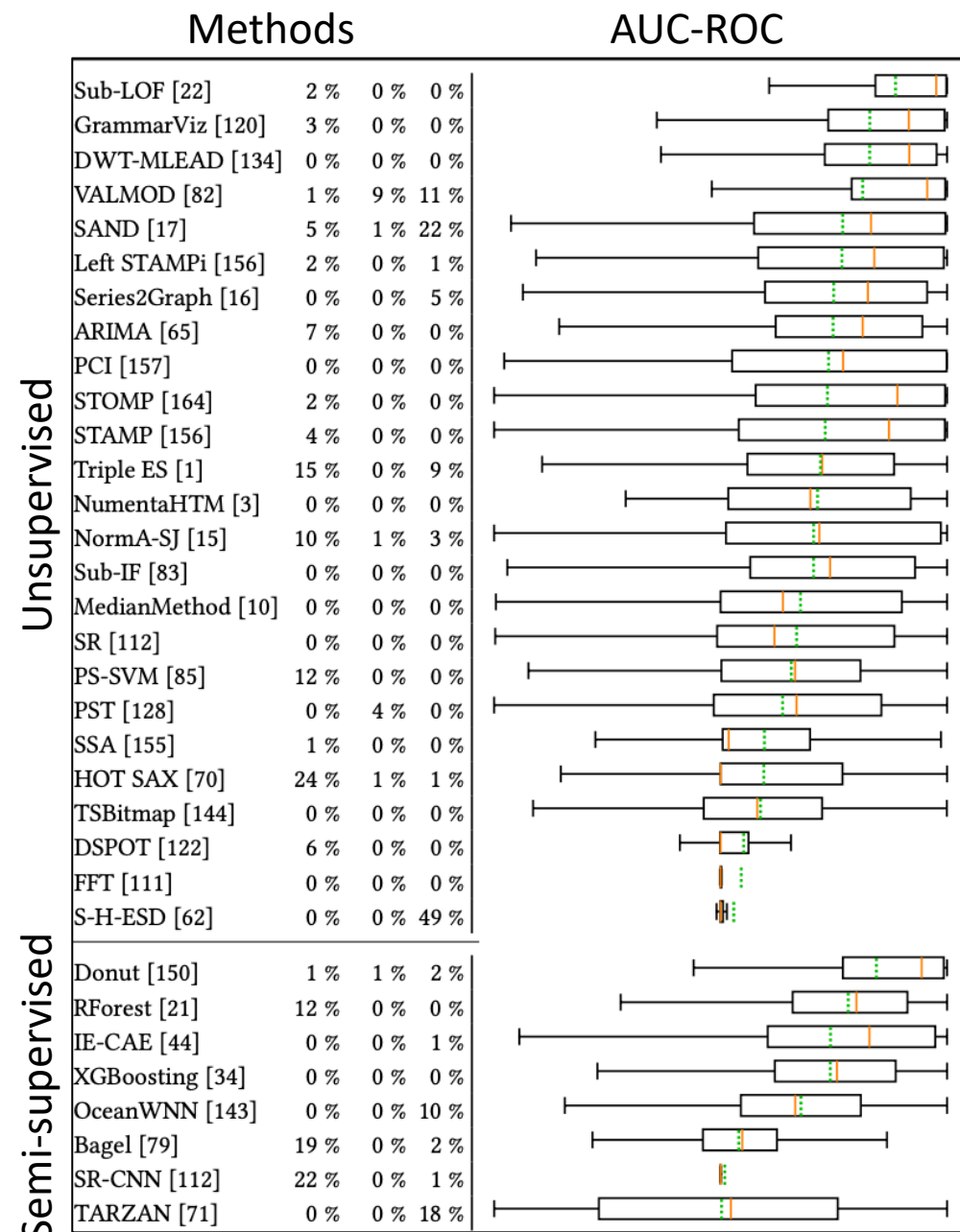
Anomaly Detection methods:

Experimental evaluation

Observations on TimeEval [5]:

- Distance-based and Density-based methods have a better accuracy (AUC-ROC) than forecasting and reconstruction-based approaches
- Semi-supervised methods are not outperforming Unsupervised approaches

[5] Sebastian Schmidl, Phillip Wenig, and Thorsten Papenbrock. 2022. Anomaly detection in time series: a comprehensive evaluation. Proc. VLDB Endow. 15, 9 (May 2022), 1779–1797.

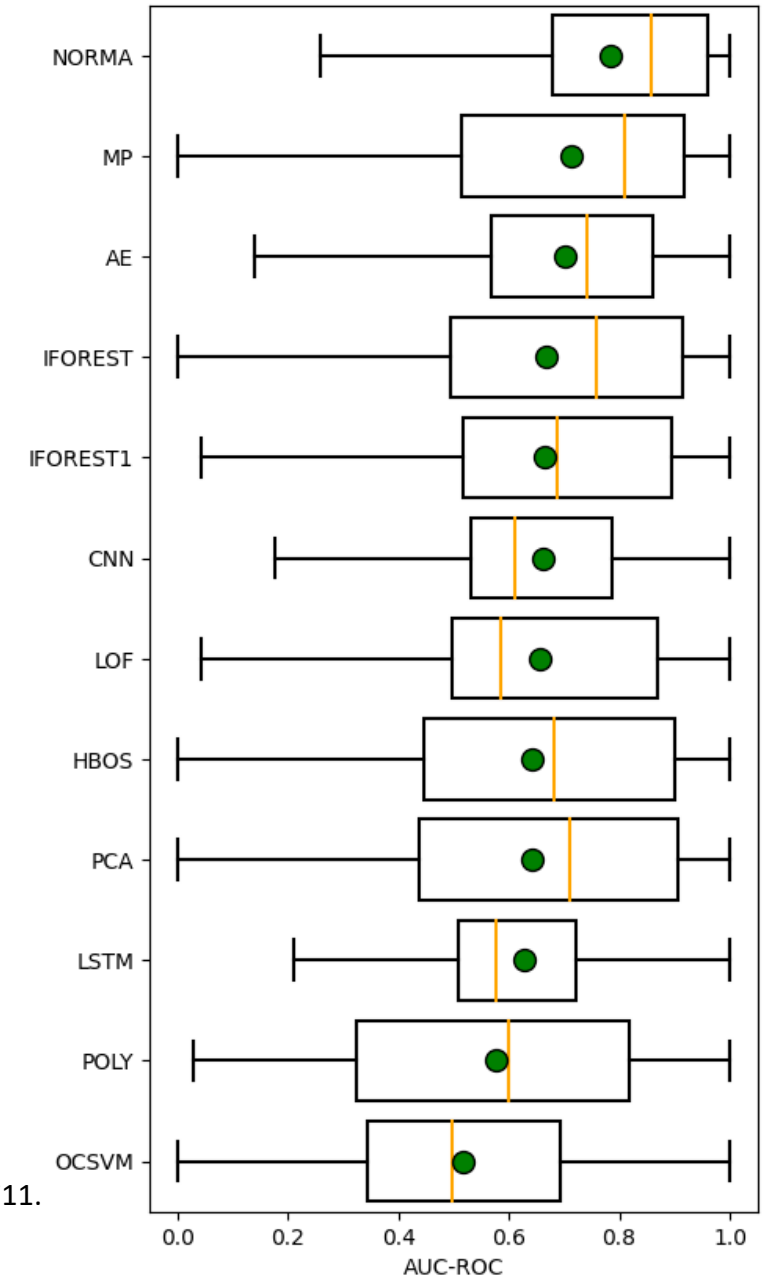


Anomaly Detection methods: *Experimental evaluation*

Observations on TSB-UAD [19]:

- Distance-based methods have a better accuracy (AUC-ROC) than forecasting-based methods.
- Isolation Forest (distribution-based and not proposed for time series) have also a strong accuracy
- AutoEncoder (AE) is also very accurate.

[19] John Paparrizos, Yuhao Kang, Paul Boniol, Ruey S. Tsay, Themis Palpanas, and Michael J. Franklin. 2022. TSB-UAD: an end-to-end benchmark suite for univariate time-series anomaly detection. Proc. VLDB Endow. 15, 8 (April 2022), 1697–1711.



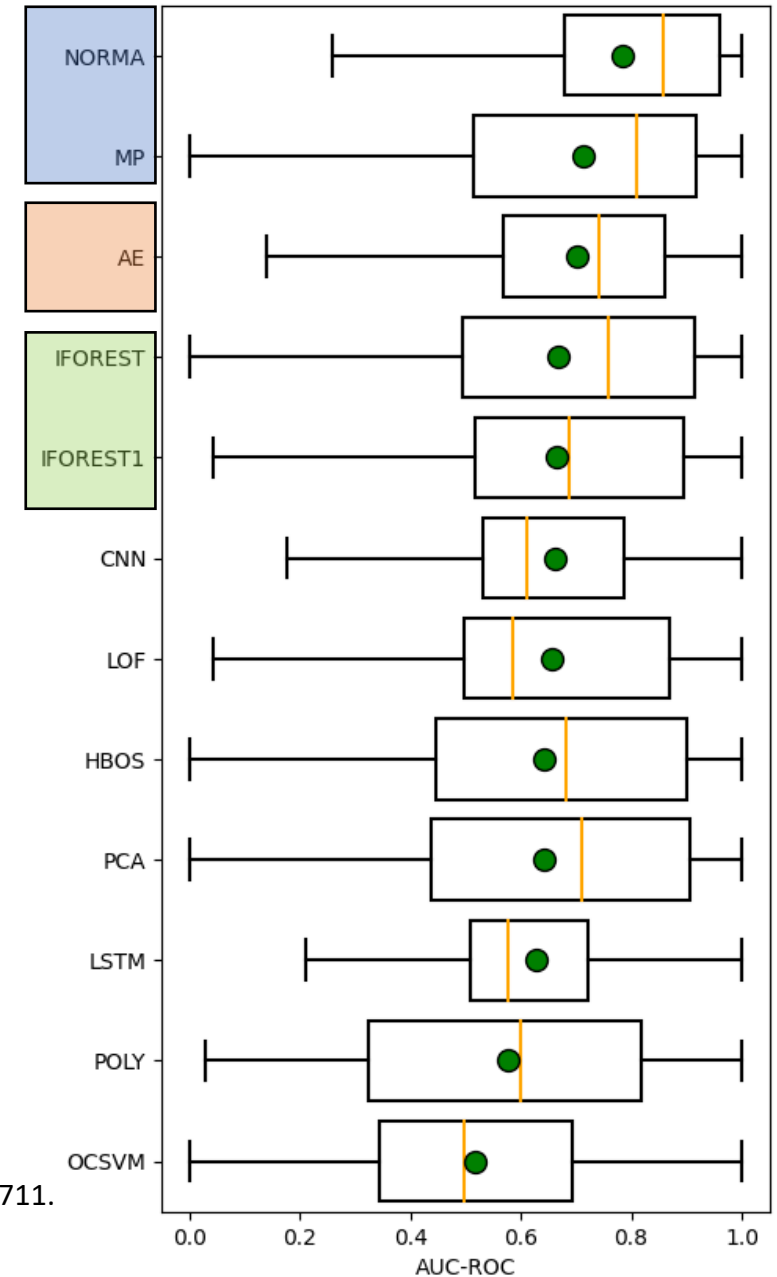
Anomaly Detection methods:

Experimental evaluation

Observations on TSB-UAD [19]:

- Distance-based methods have a better accuracy (AUC-ROC) than forecasting-based methods.
- Isolation Forest (distribution-based and not proposed for time series) have also a strong accuracy
- AutoEncoder (AE) is also very accurate.

[19] John Paparrizos, Yuhao Kang, Paul Boniol, Ruey S. Tsay, Themis Palpanas, and Michael J. Franklin. 2022. TSB-UAD: an end-to-end benchmark suite for univariate time-series anomaly detection. Proc. VLDB Endow. 15, 8 (April 2022), 1697–1711.

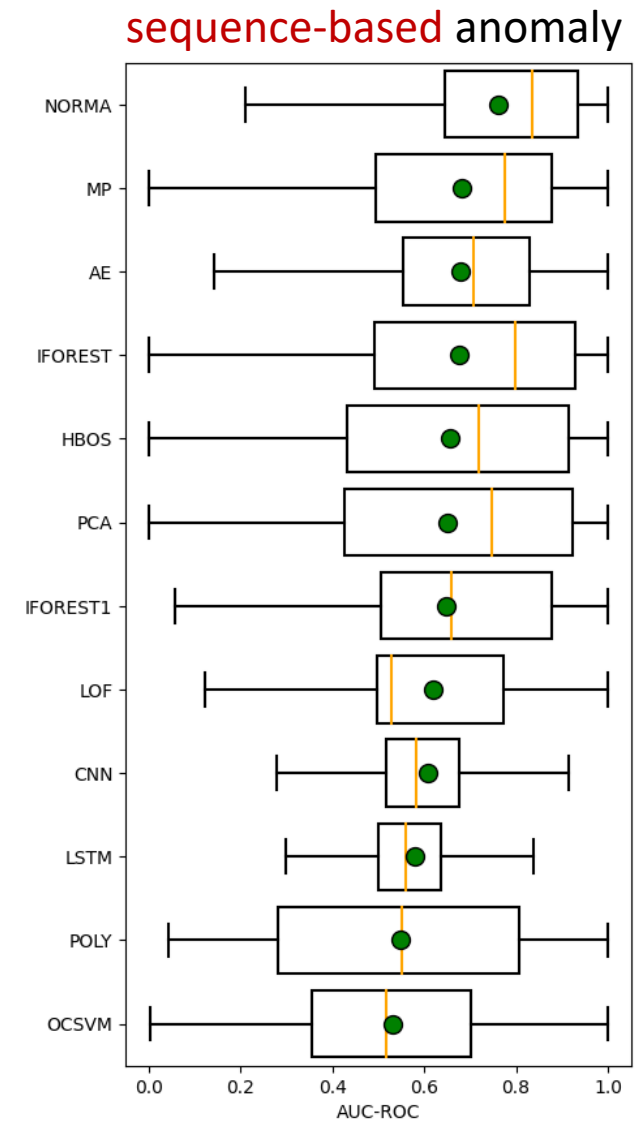
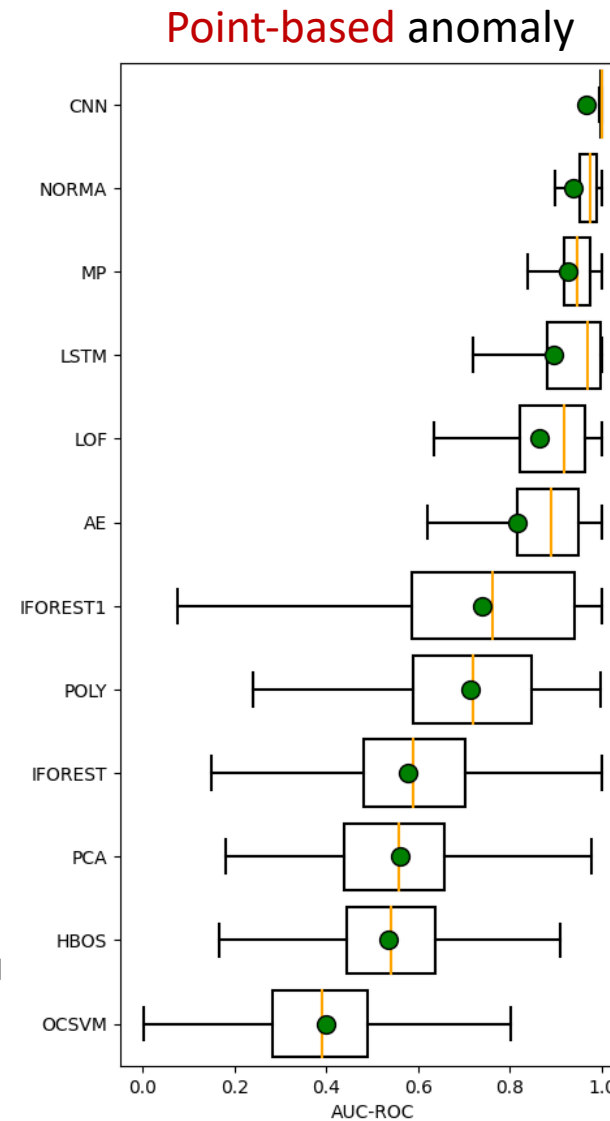


Anomaly Detection methods:

Experimental evaluation

Observations on TSB-UAD [19]:

[19] John Paparrizos, Yuhao Kang, Paul Boniol, Ruey S. Tsay, Themis Palpanas, and Michael J. Franklin. 2022. TSB-UAD: an end-to-end benchmark suite for univariate time-series anomaly detection. Proc. VLDB Endow. 15, 8 (April 2022), 1697–1711.



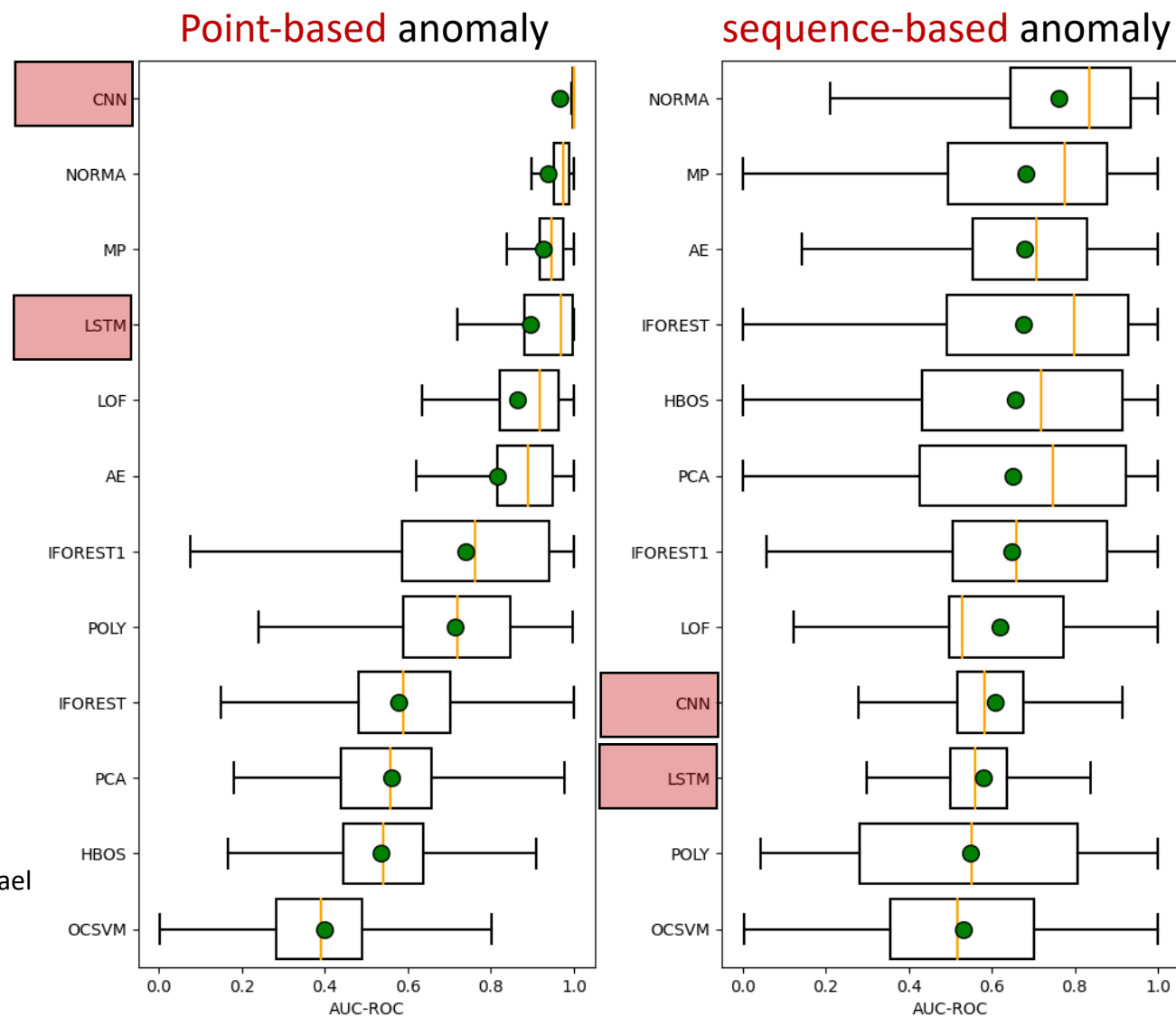
Anomaly Detection methods:

Experimental evaluation

Observations on TSB-UAD [19]:

- Forecasting methods (LSTM and CNN) are very **accurate** for point anomalies
- But have **poor performances** on **sequence-based** anomalies.

[19] John Paparrizos, Yuhao Kang, Paul Boniol, Ruey S. Tsay, Themis Palpanas, and Michael J. Franklin. 2022. TSB-UAD: an end-to-end benchmark suite for univariate time-series anomaly detection. Proc. VLDB Endow. 15, 8 (April 2022), 1697–1711.

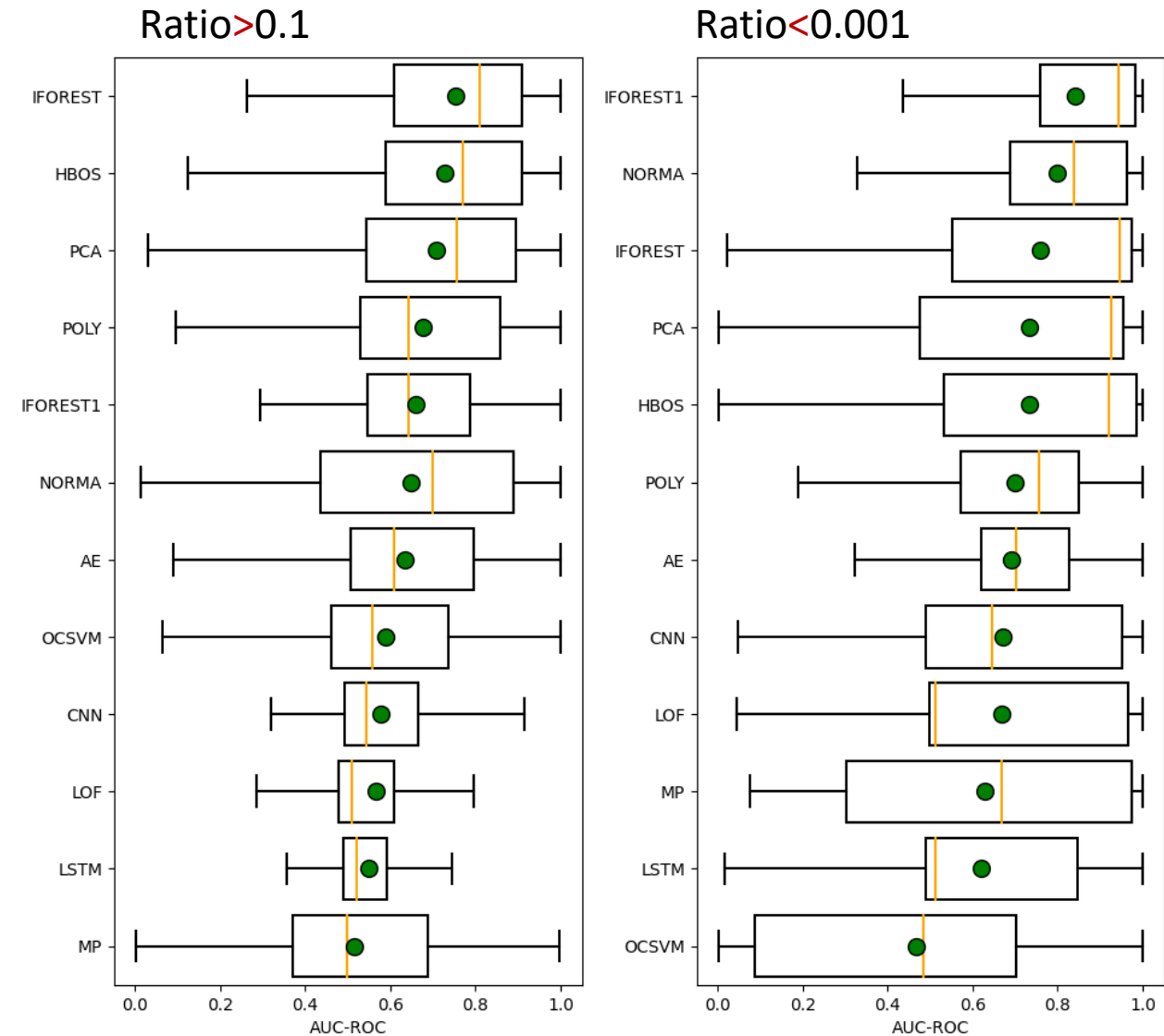


Anomaly Detection methods: *Experimental evaluation*

Observations on TSB-UAD [19]:

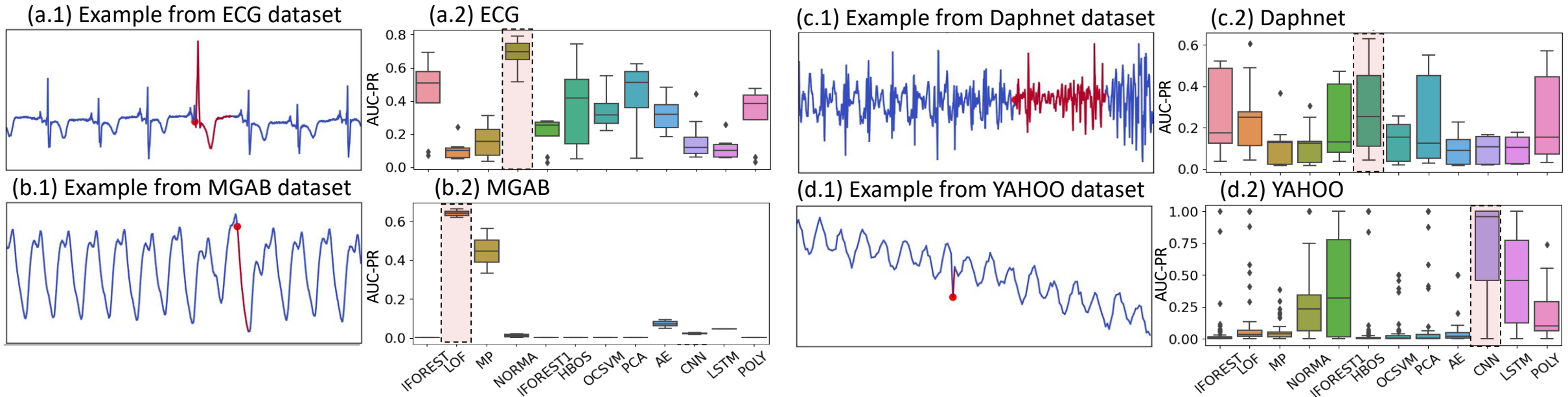
- The ratio of normal/abnormal points has a **strong impact** on the methods ranking.

[19] John Paparrizos, Yuhao Kang, Paul Boniol, Ruey S. Tsay, Themis Palpanas, and Michael J. Franklin. 2022. TSB-UAD: an end-to-end benchmark suite for univariate time-series anomaly detection. Proc. VLDB Endow. 15, 8 (April 2022), 1697–1711.



Anomaly Detection methods: *Experimental evaluation*

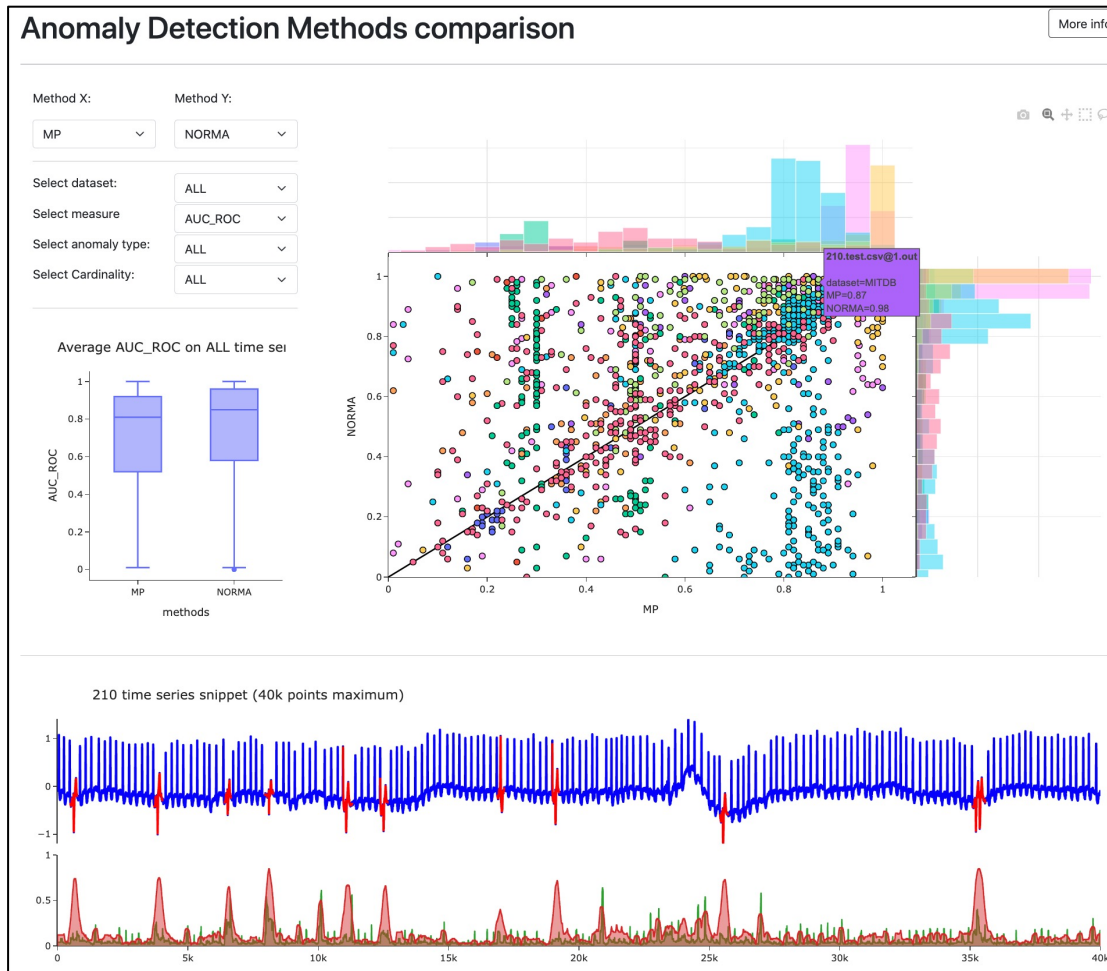
Observation from the results applied on specific datasets (TSB-UAD [19])



There is **no overall winner**.

[19] John Paparrizos, Yuhao Kang, Paul Boniol, Ruey S. Tsay, Themis Palpanas, and Michael J. Franklin. 2022. TSB-UAD: an end-to-end benchmark suite for univariate time-series anomaly detection. Proc. VLDB Endow. 15, 8 (April 2022), 1697–1711.

Anomaly Detection methods: *Experimental evaluation*



Theseus [27]

An interactive tool to compare anomaly detection methods



VLDB 2022



Github repo

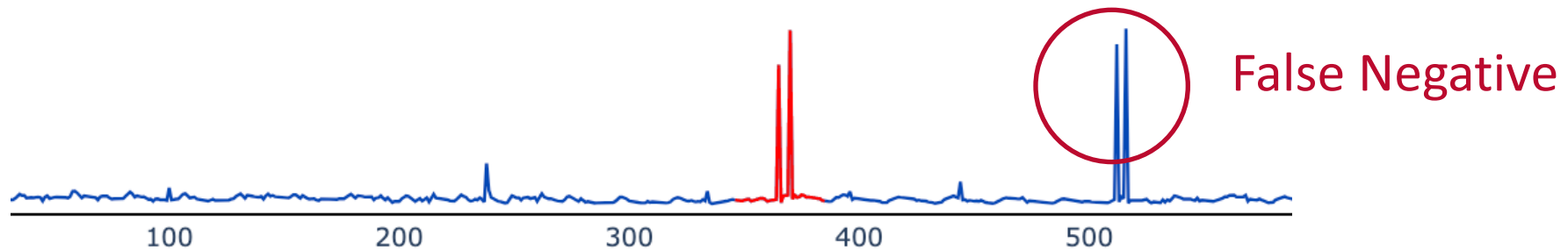
[27] Paul Boniol, John Paparrizos, Yuhao Kang, Themis Palpanas, Ruey S. Tsay, Aaron J. Elmore, and Michael J. Franklin. 2022. Theseus: navigating the labyrinth of time-series anomaly detection. Proc. VLDB Endow. 15, 12 (August 2022), 3702–3705.

Benchmark Practice: *Common Flaws in Dataset*

Mislabeling

Bias

Feasibility

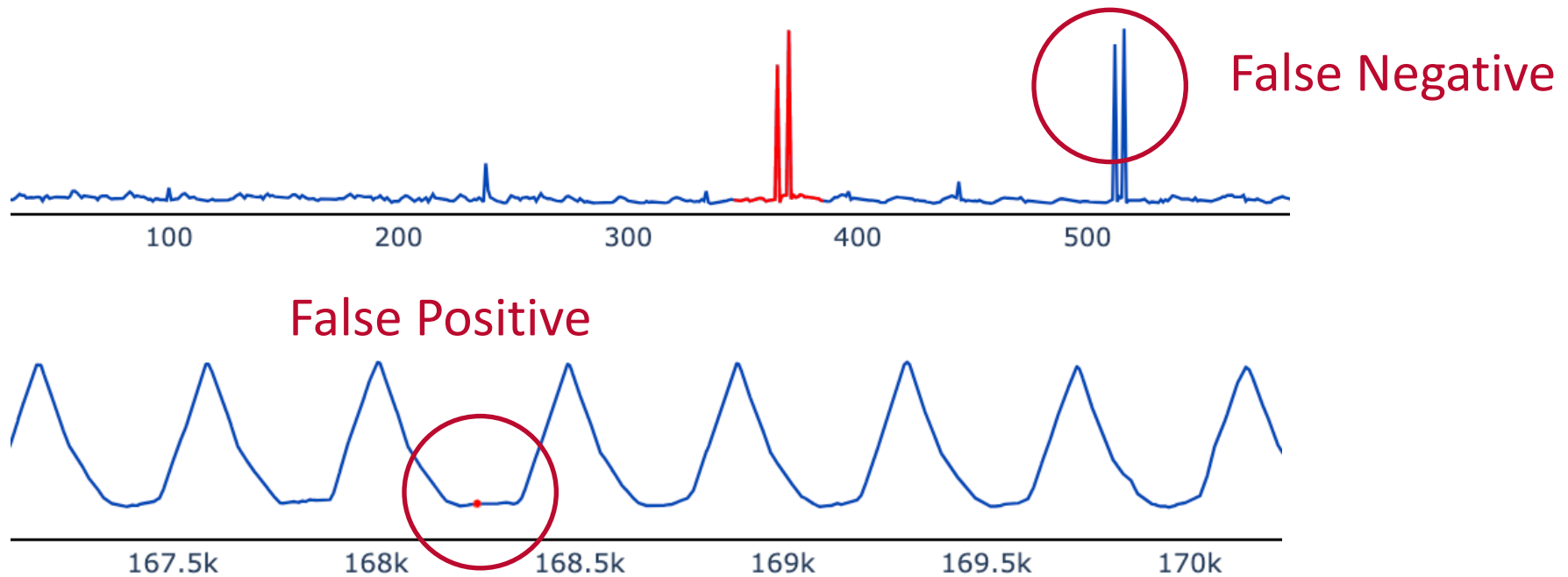


Benchmark Practice: *Common Flaws in Dataset*

Mislabeling

Bias

Feasibility

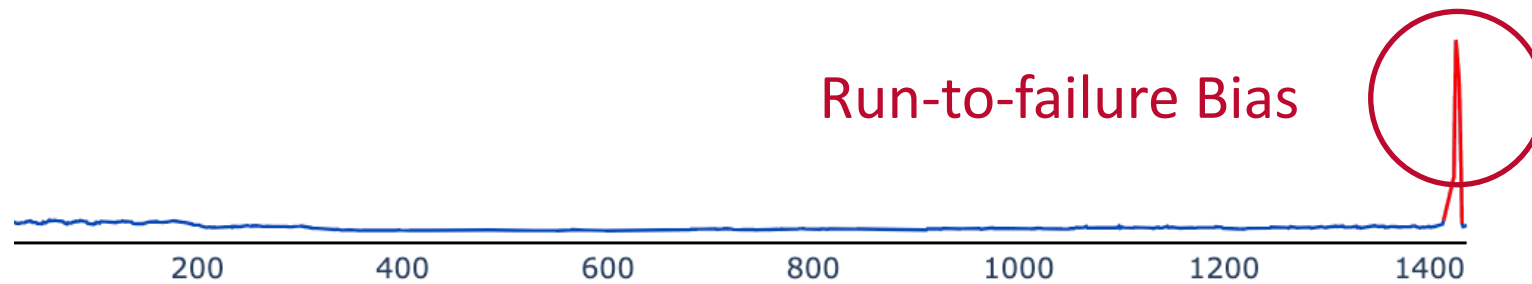


Benchmark Practice: *Common Flaws in Dataset*

Mislabeling

Bias

Feasibility

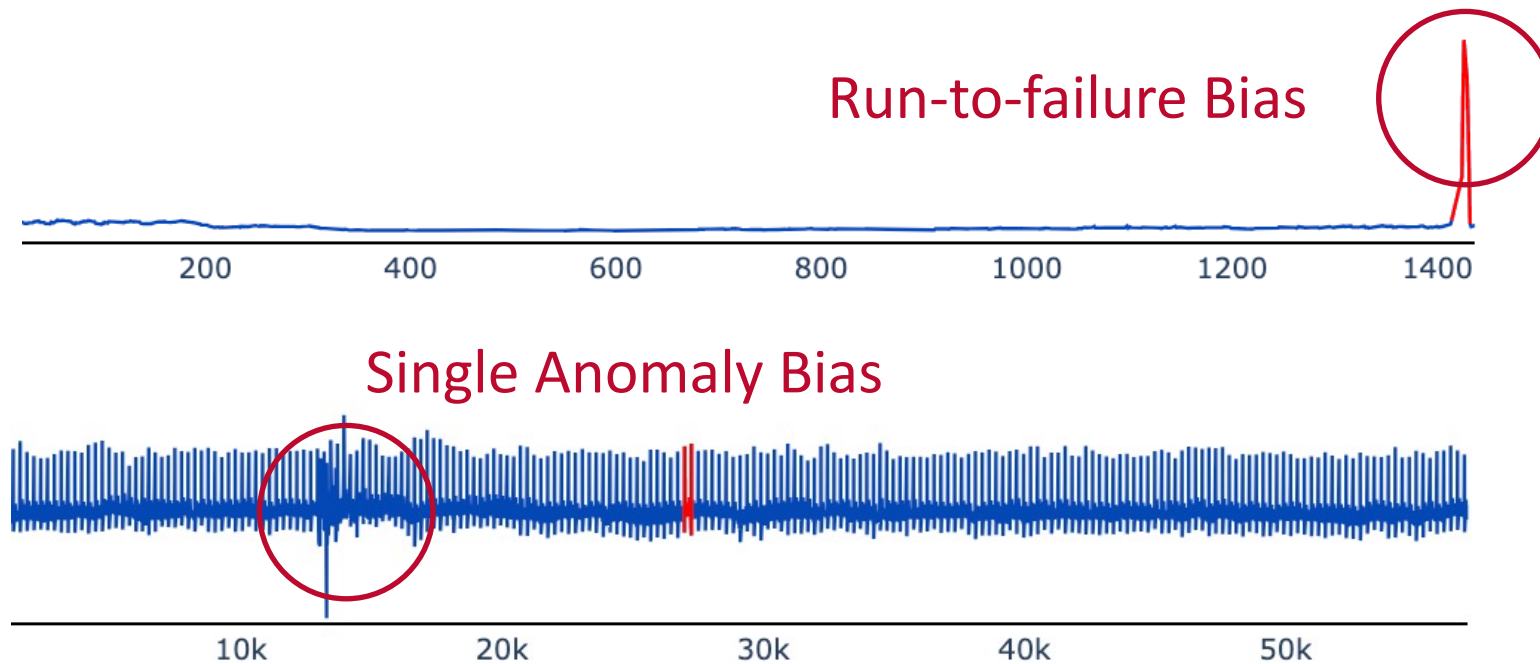


Benchmark Practice: *Common Flaws in Dataset*

Mislabeling

Bias

Feasibility



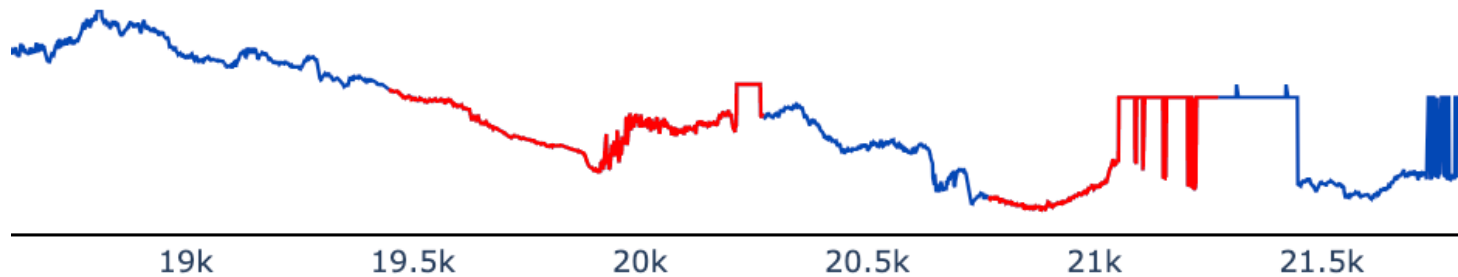
Benchmark Practice: *Common Flaws in Dataset*

Mislabeling

Bias

Feasibility

Lack of In-context Data



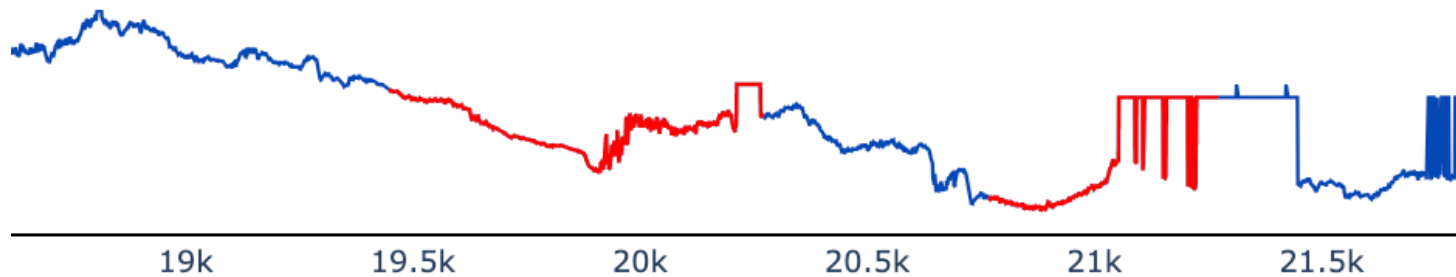
Benchmark Practice: *Common Flaws in Dataset*

Mislabeling

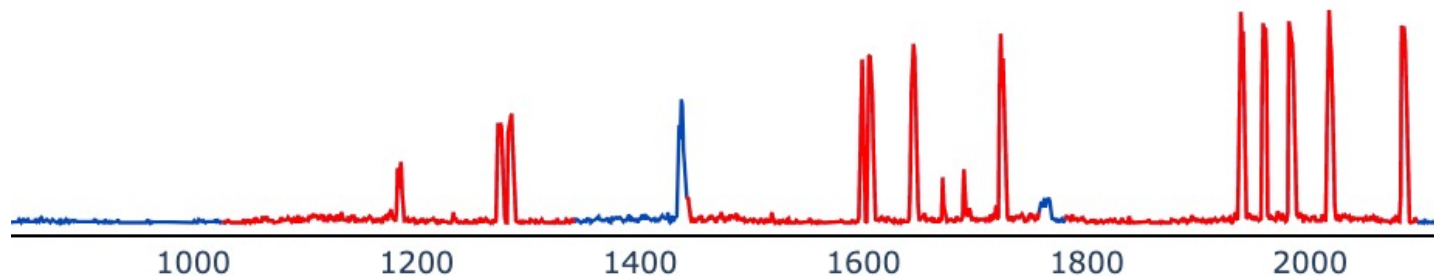
Bias

Feasibility

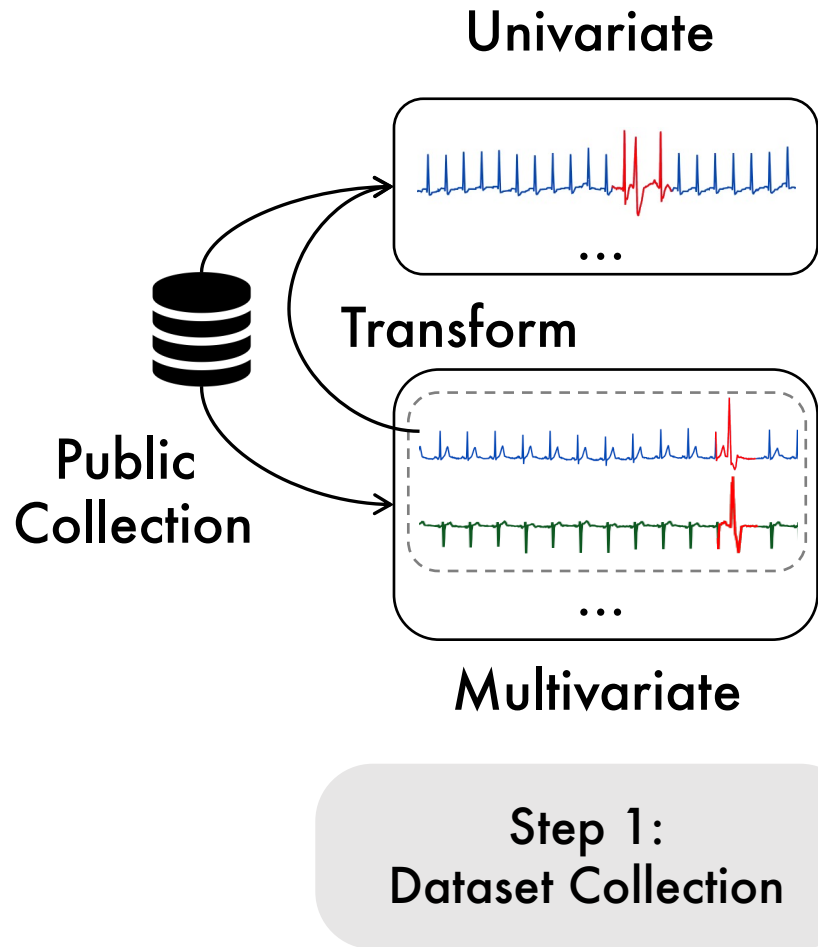
Lack of In-context Data



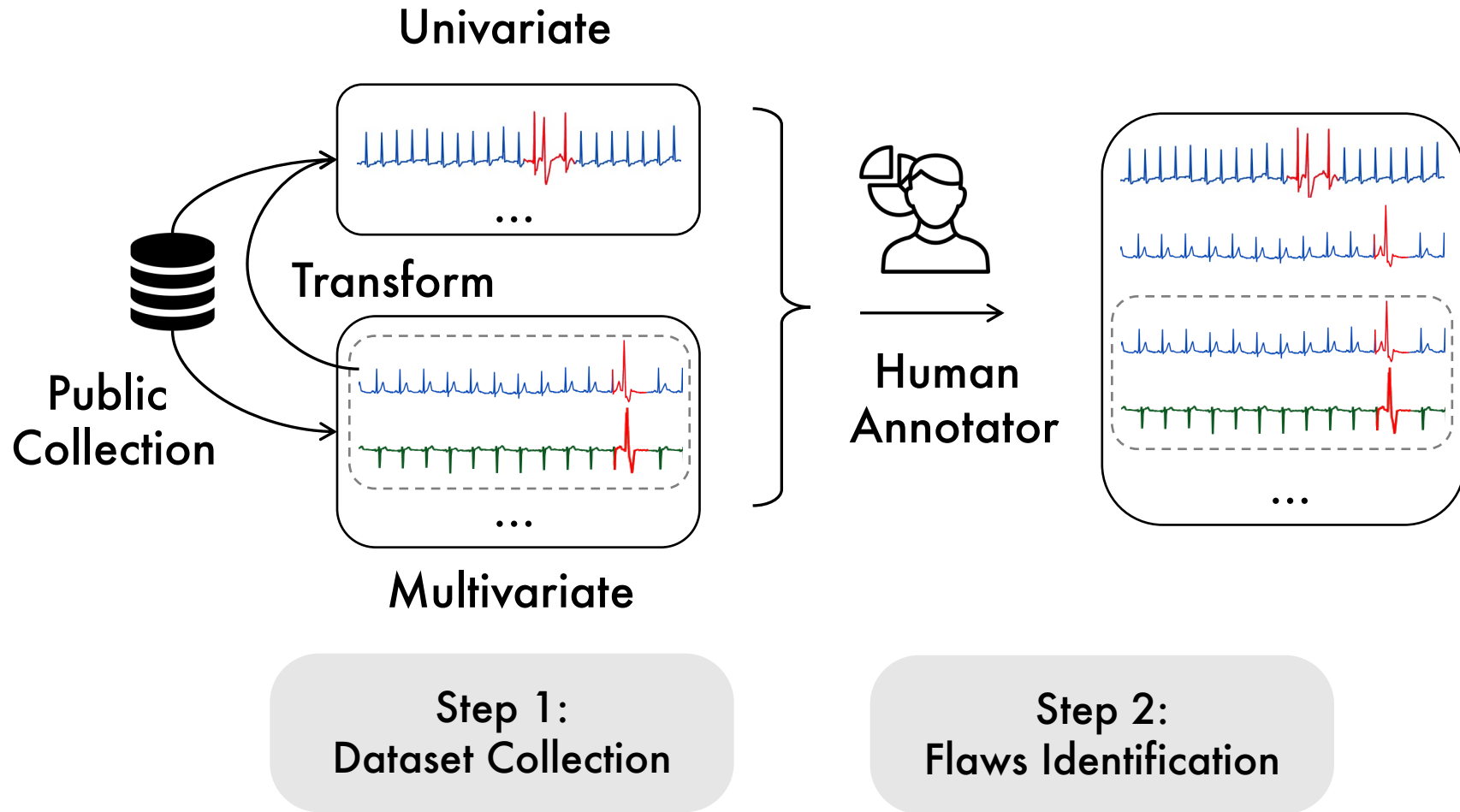
Unrealistic Anomaly Ratio



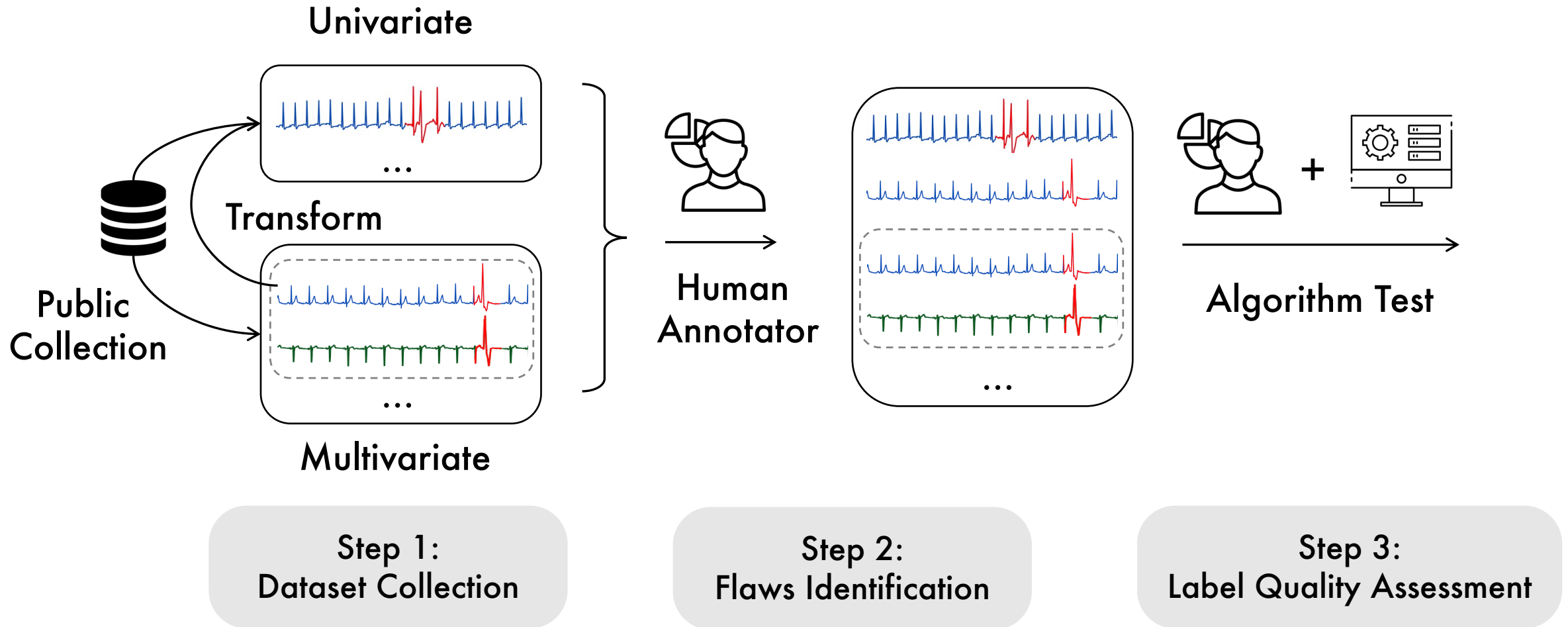
Benchmark Practice: *Dataset Construction*



Benchmark Practice: *Dataset Construction*



Benchmark Practice: *Dataset Construction*

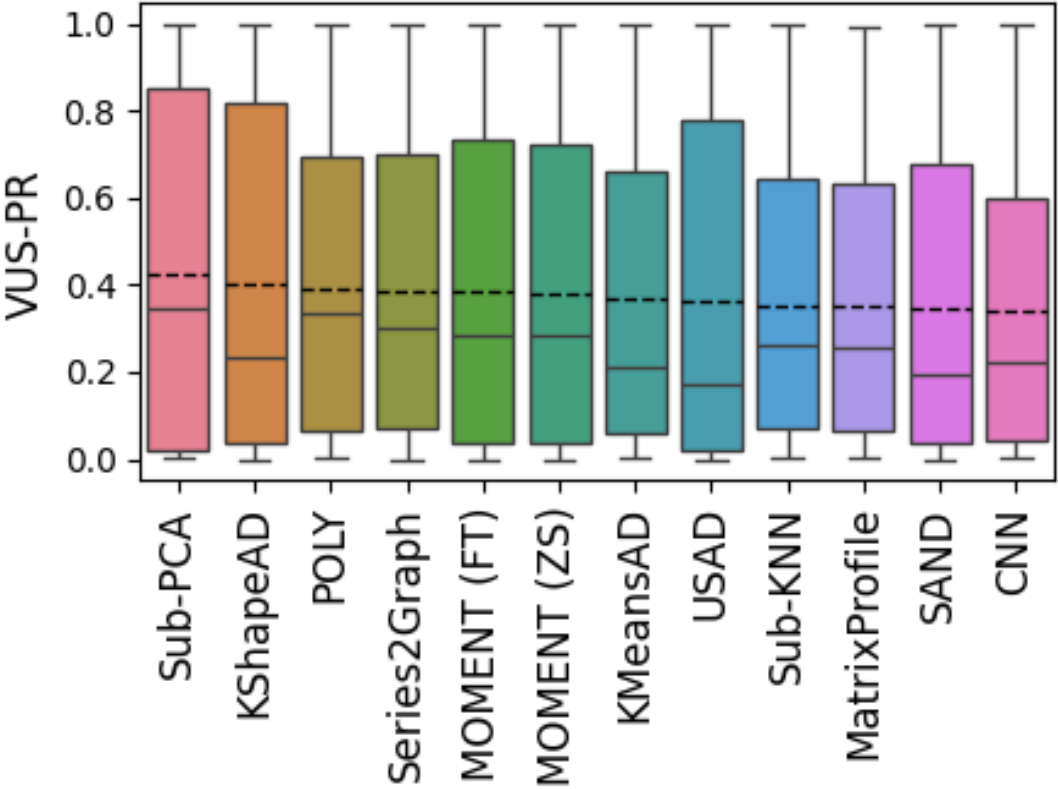


Benchmark Practice: *Dataset Construction*

Category	Split	# TS	Avg Length	Avg Anomaly Length	Avg # Anomalies	Anomaly Ratio
TSB-AD-U	All	870	38814.1	179.5	39.7	2.4%
	Eval	350	51886.7	321.3	46.6	4.5%
	Tuning	48	47143.3	185.9	82.6	3.5%
TSB-AD-M	All	200	107760.4	582.6	71.1	5.1%
	Eval	180	108826.7	591.2	67.7	5.0%
	Tuning	20	98164.1	504.7	101.1	5.7%

Benchmark Practice: *Evaluation*

TSB-AD-U

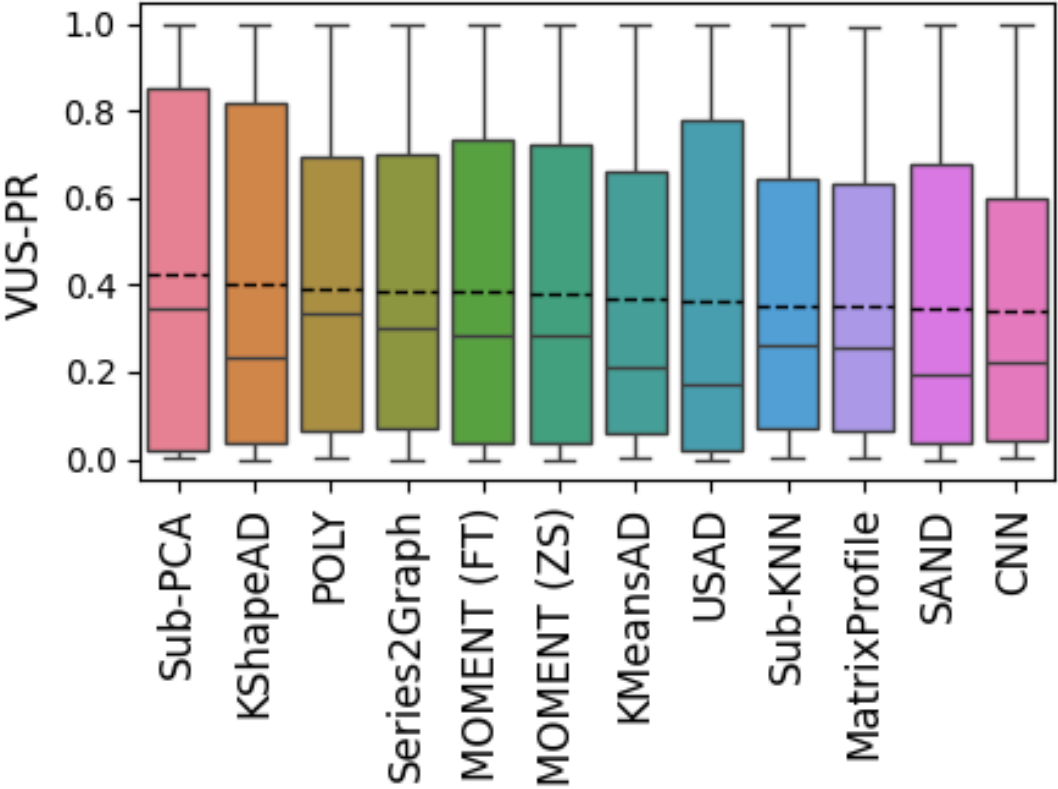


🚩 **VUS-PR Ranking**

🥇	Sub-PCA
🥈	KShapeAD
🥉	POLY
4	Series2Graph
5	MOMENT (FT)
6	MOMENT (ZS)
7	KMeansAD
8	USAD
9	Sub-KNN
10	MatrixProfile
11	SAND
12	CNN

Benchmark Practice: *Evaluation*

TSB-AD-U



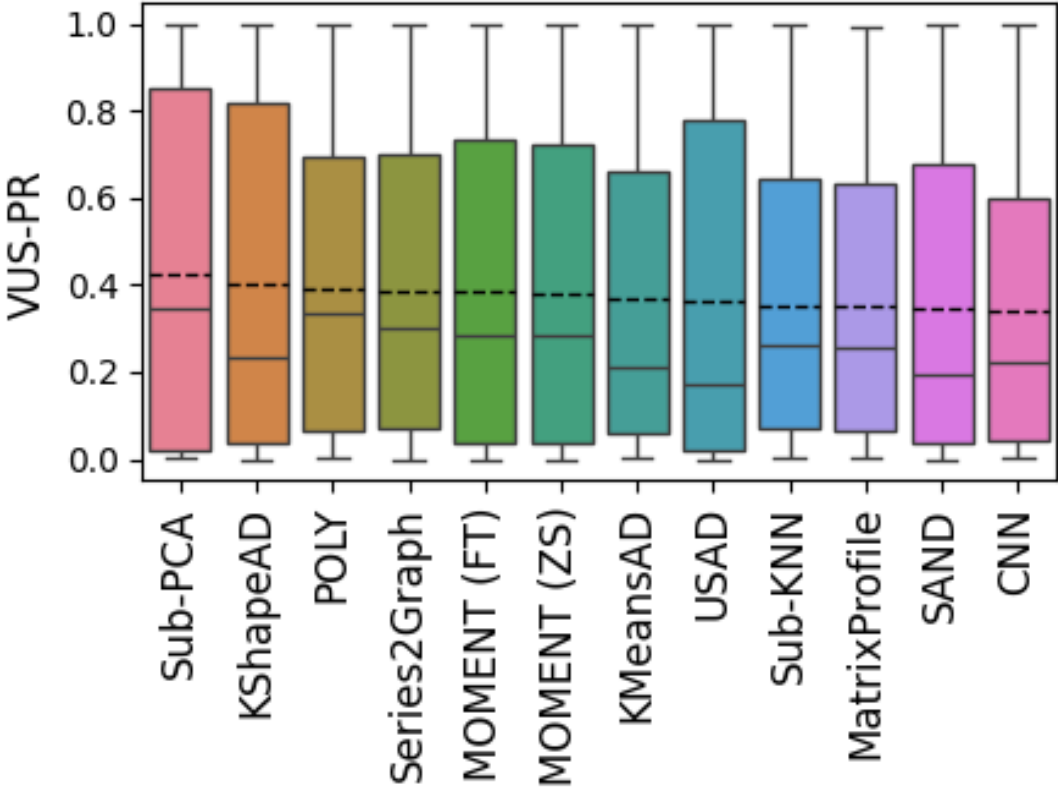
VUS-PR Ranking

1	Sub-PCA
2	KShapeAD
3	POLY
4	Series2Graph
5	MOMENT (FT)
6	MOMENT (ZS)
7	KMeansAD
8	USAD
9	Sub-KNN
10	MatrixProfile
11	SAND
12	CNN

① Top-performing methods been overlooked for many years

Benchmark Practice: *Evaluation*

TSB-AD-U



VUS-PR Ranking

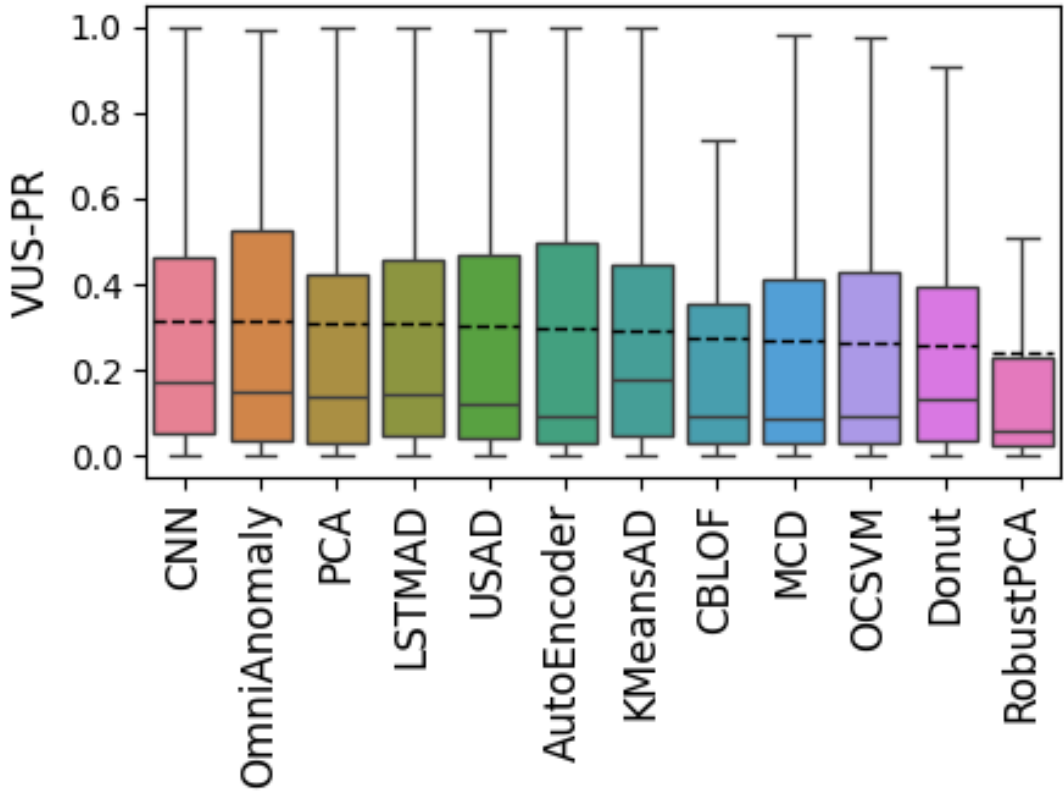
1	Sub-PCA
2	KShapeAD
3	POLY
4	Series2Graph
5	MOMENT (FT)
6	MOMENT (ZS)
7	KMeansAD
8	USAD
9	Sub-KNN
10	MatrixProfile
11	SAND
12	CNN

① Top-performing methods been overlooked for many years

② Performance of time-series foundation models shows promise

Benchmark Practice: *Evaluation*

TSB-AD-M



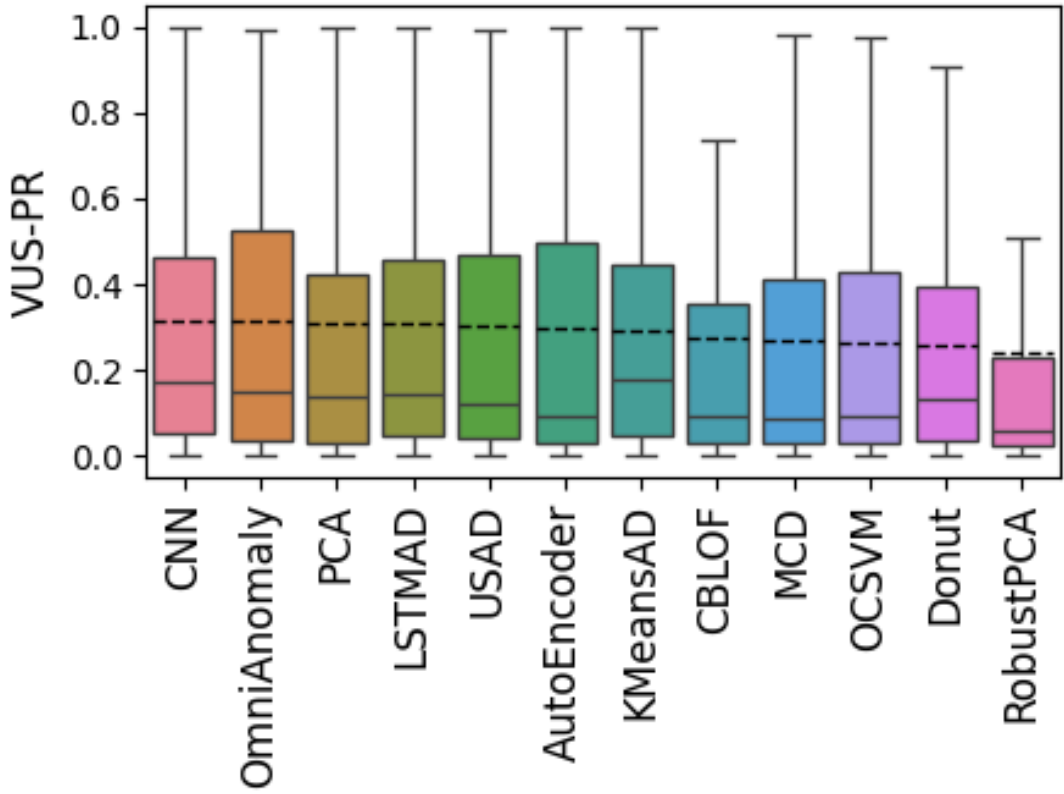
► **VUS-PR Ranking**

1	CNN
2	OmniAnomaly
3	PCA
4	LSTMAD
5	USAD
6	AutoEncoder
7	KMeansAD
8	CBLOF
9	MCD
10	OCSVM
11	Donut
12	RobustPCA

③ Neural-network-based methods strive in multivariate cases

Benchmark Practice: *Evaluation*

TSB-AD-M



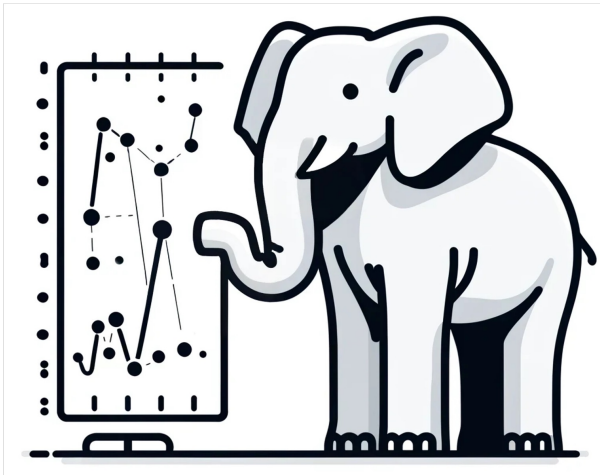
► **VUS-PR Ranking**

1	CNN
2	OmniAnomaly
3	PCA
4	LSTMAD
5	USAD
6	AutoEncoder
7	KMeansAD
8	CBLOF
9	MCD
10	OCSVM
11	Donut
12	RobustPCA

③ Neural-network-based methods strive in multivariate cases

④ Simpler architectures generally outperform more complex designs

Benchmark Practice: *Evaluation*



1070 Curated Time Series

40 TSAD Algorithms

10 Evaluation Measures

TSB-AD [27]

Towards a reliable time-series anomaly detection benchmark



NeurIPS 2024



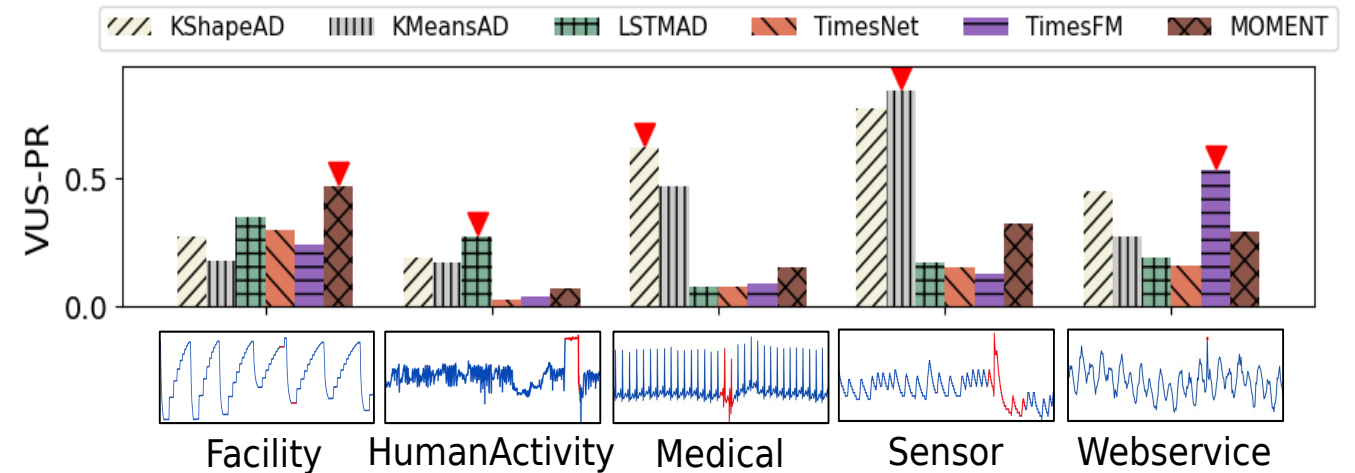
Homepage

Part 5: Automated Solutions for Anomaly Detection

Automated Solutions: *Background*

Motivation:

- No one-size-fits-all model:
- How can we *automatically* identify the best anomaly detector given a time series?



[29] Maroua Bahri, Flavia Salutari, Andrian Putina, and Mauro Sozio: AutoML: state of the art with a focus on anomaly detection, challenges, and research directions. *International Journal of Data Science and Analytics* 14(2): 113-126 (2022).

[41] Liu, Q., Lee, S. and Paparrizos, J., 2025. TSB-AutoAD: Towards Automated Solutions for Time-Series Anomaly Detection. *Proceedings of the VLDB Endowment*, 18(11), pp.4364–4379.

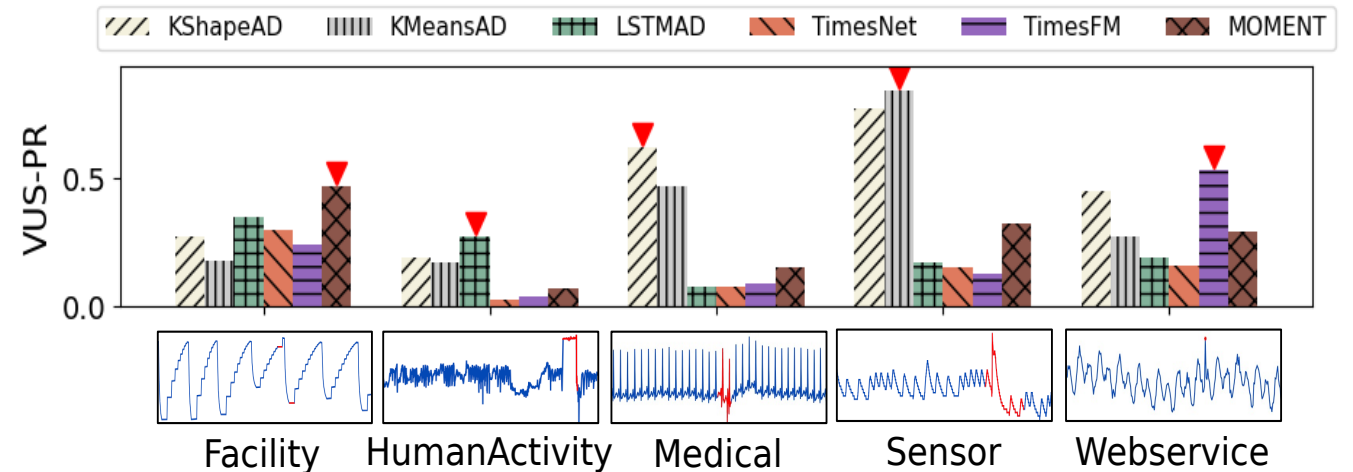
Automated Solutions: *Background*

Motivation:

- No one-size-fits-all model:
- How can we *automatically* identify the best anomaly detector given a time series?

Challenge:

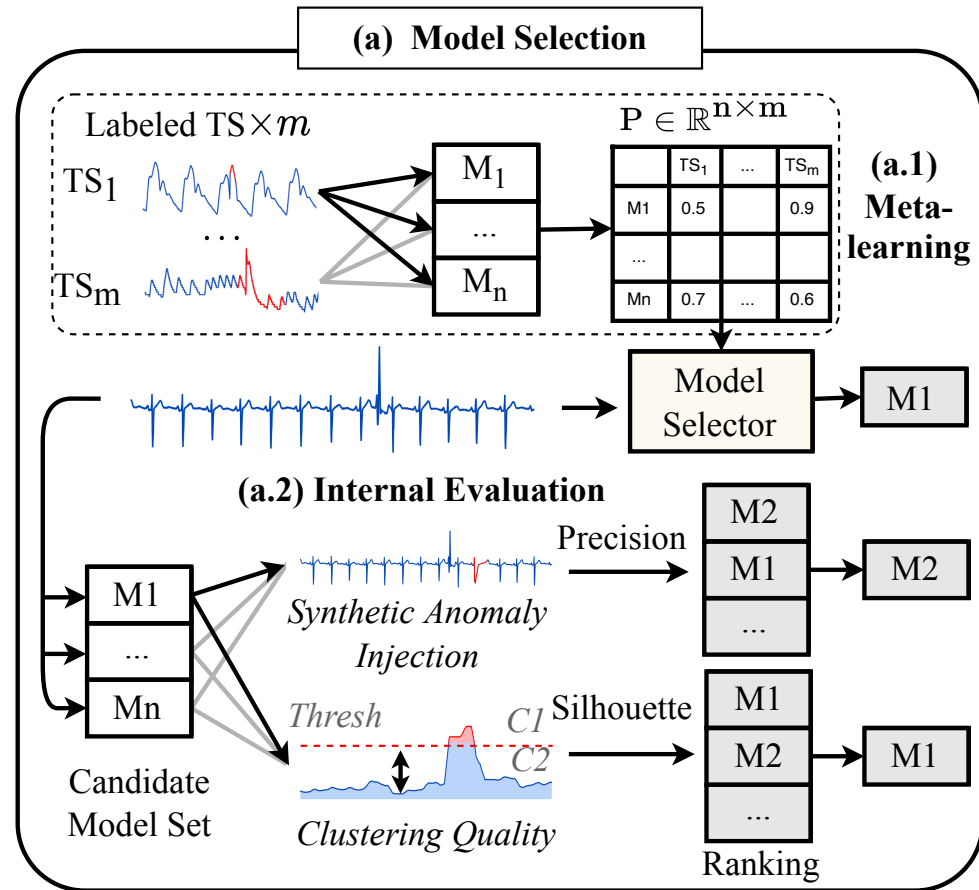
- Lack of labeled data
- Absence of universal objective function



[29] Maroua Bahri, Flavia Salutari, Andrian Putina, and Mauro Sozio: AutoML: state of the art with a focus on anomaly detection, challenges, and research directions. *International Journal of Data Science and Analytics* 14(2): 113-126 (2022).

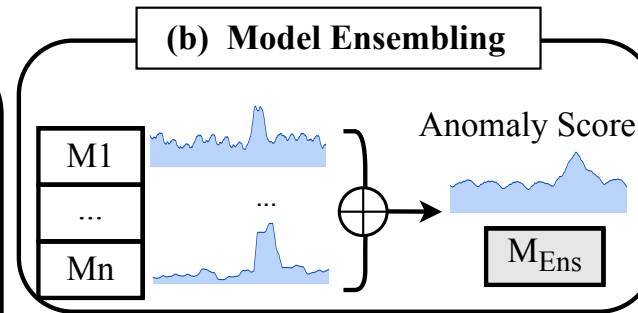
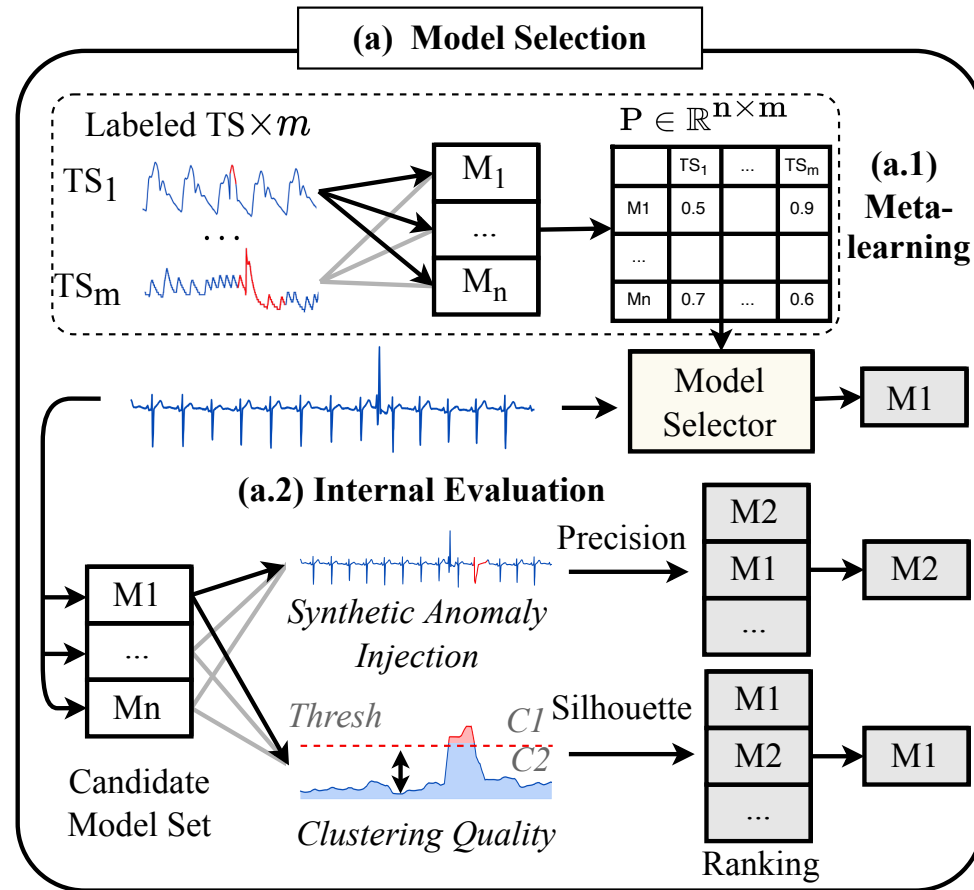
[41] Liu, Q., Lee, S. and Paparrizos, J., 2025. TSB-AutoAD: Towards Automated Solutions for Time-Series Anomaly Detection. *Proceedings of the VLDB Endowment*, 18(11), pp.4364–4379.

Automated Solutions: *Taxonomy*



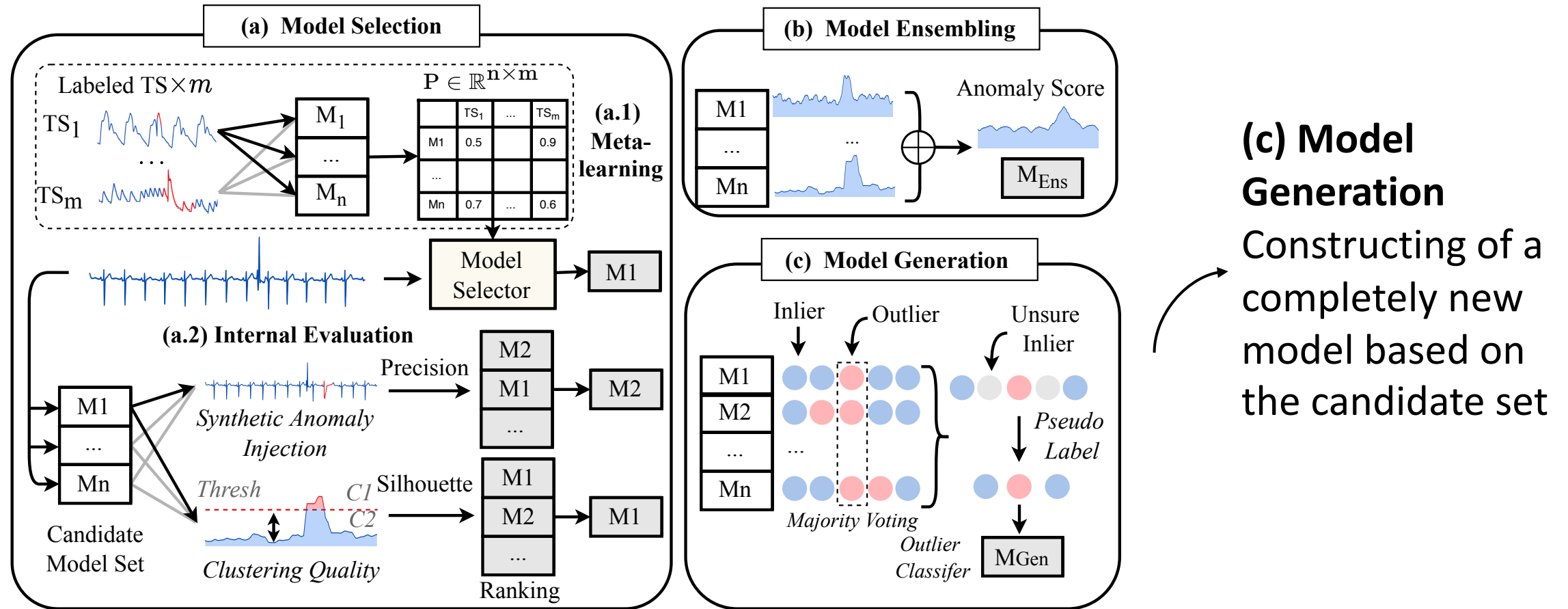
(a) Model Selection:
Selecting the best anomaly detector from a predefined candidate model set.

Automated Solutions: *Taxonomy*



(b) Model Ensembling
Aggregating predictions from multiple candidate models using ensemble strategies.

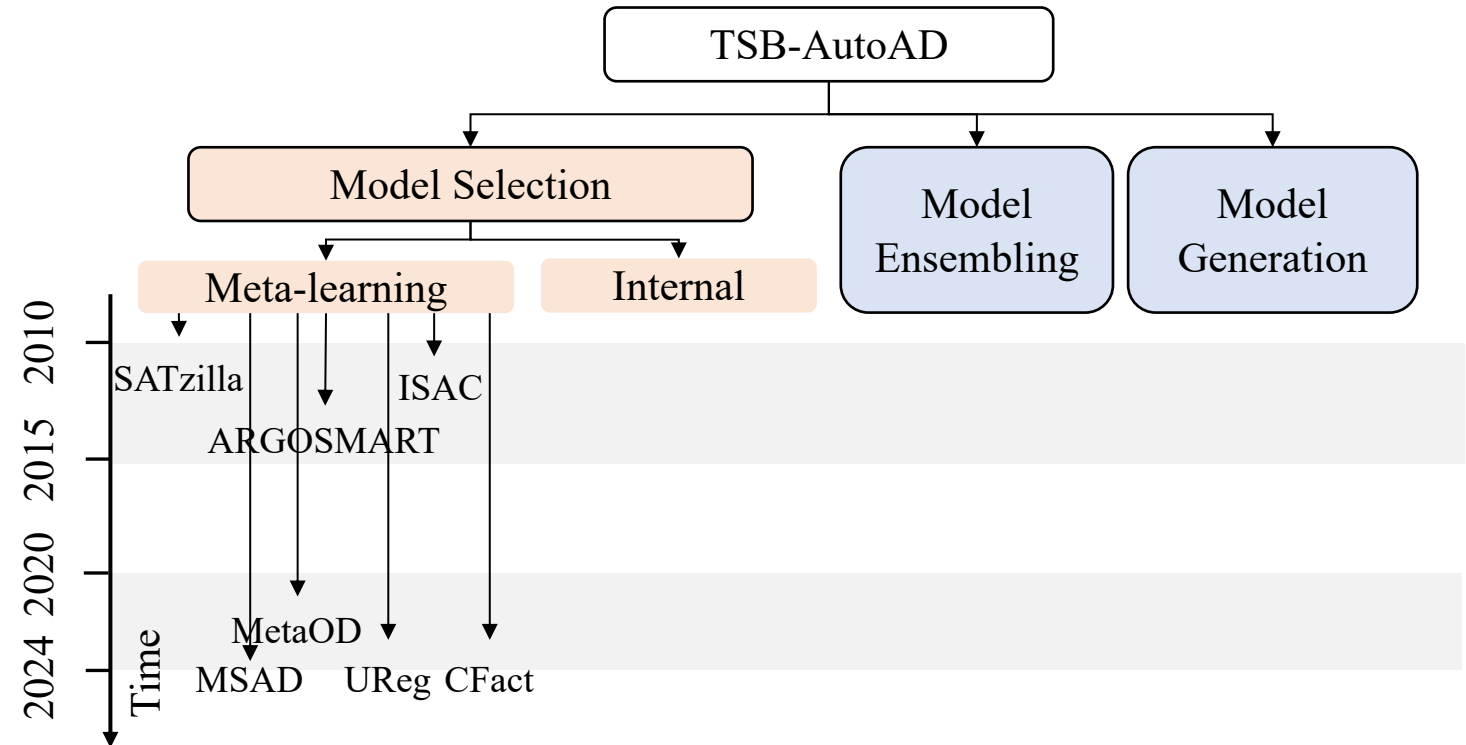
Automated Solutions: *Taxonomy*



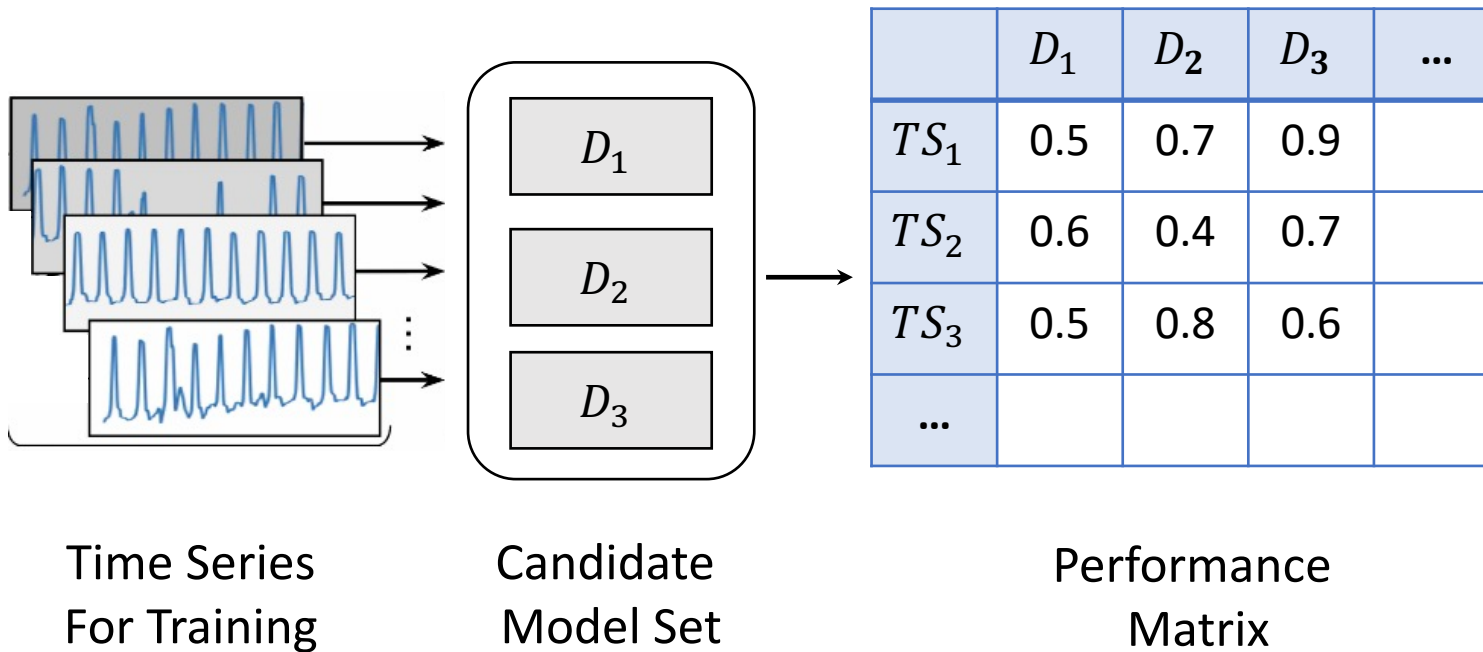
Automated Solutions: *Meta-learning*

Definition: Using insights from historical labeled datasets to select the best model for new data

- **Classification:** MSAD
- **Regression:** SATzilla, UReg, CFact
- **Nearest Neighbor:** ARGOSMART
- **Other Optimization:** ISAC, MetaOD

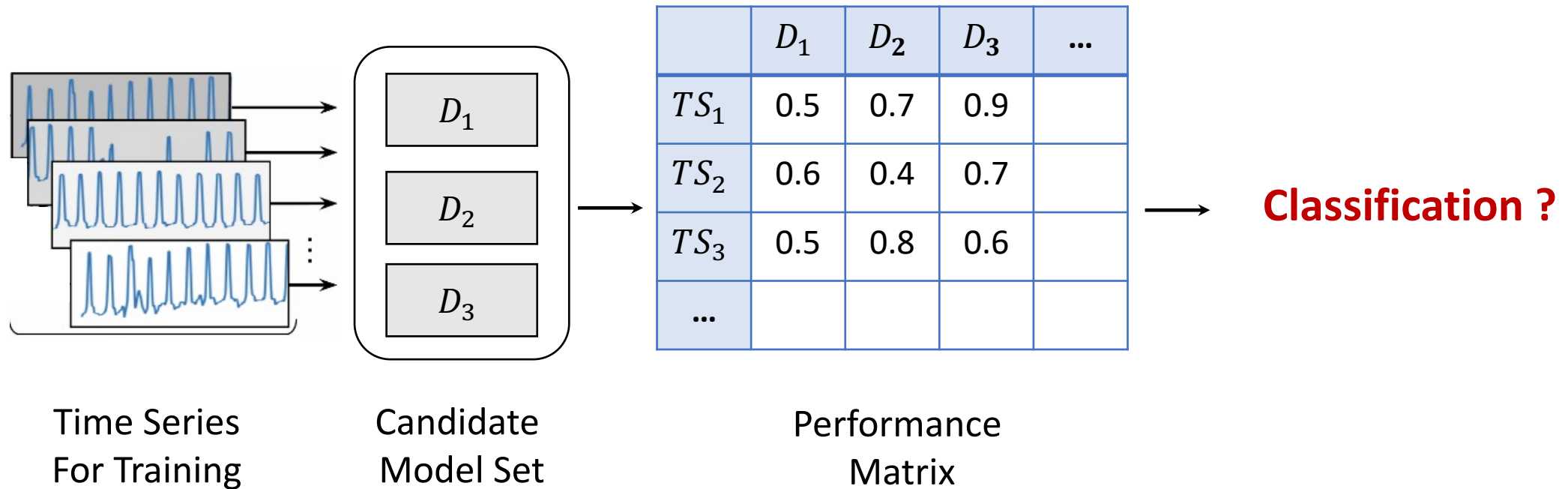


Automated Solutions: *Meta-learning*

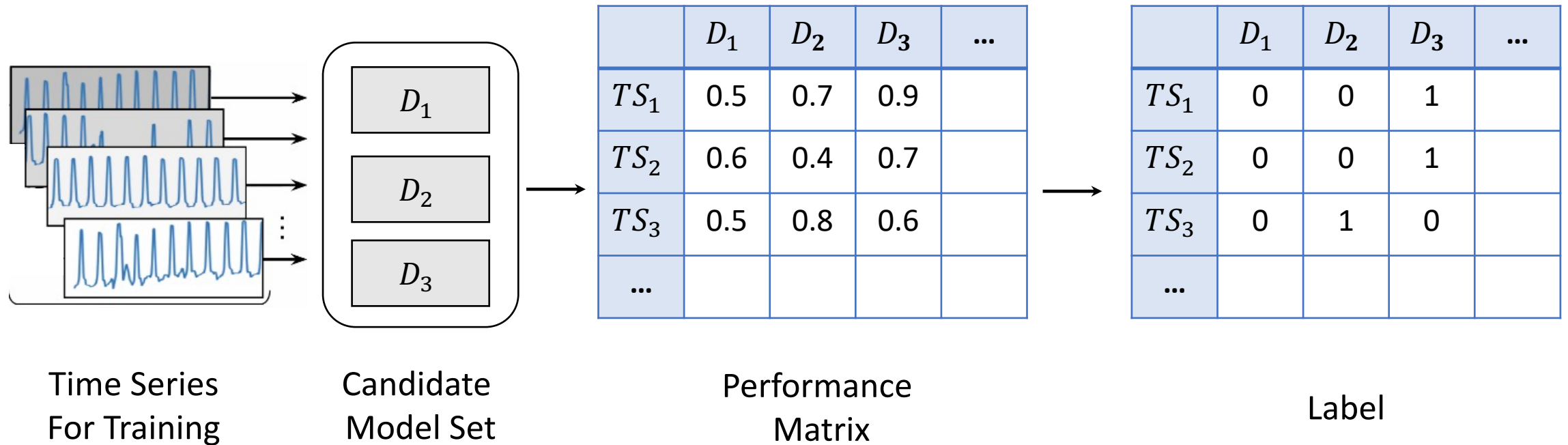


Performance measures:
F-score, AUC-PR, VUS-PR ...

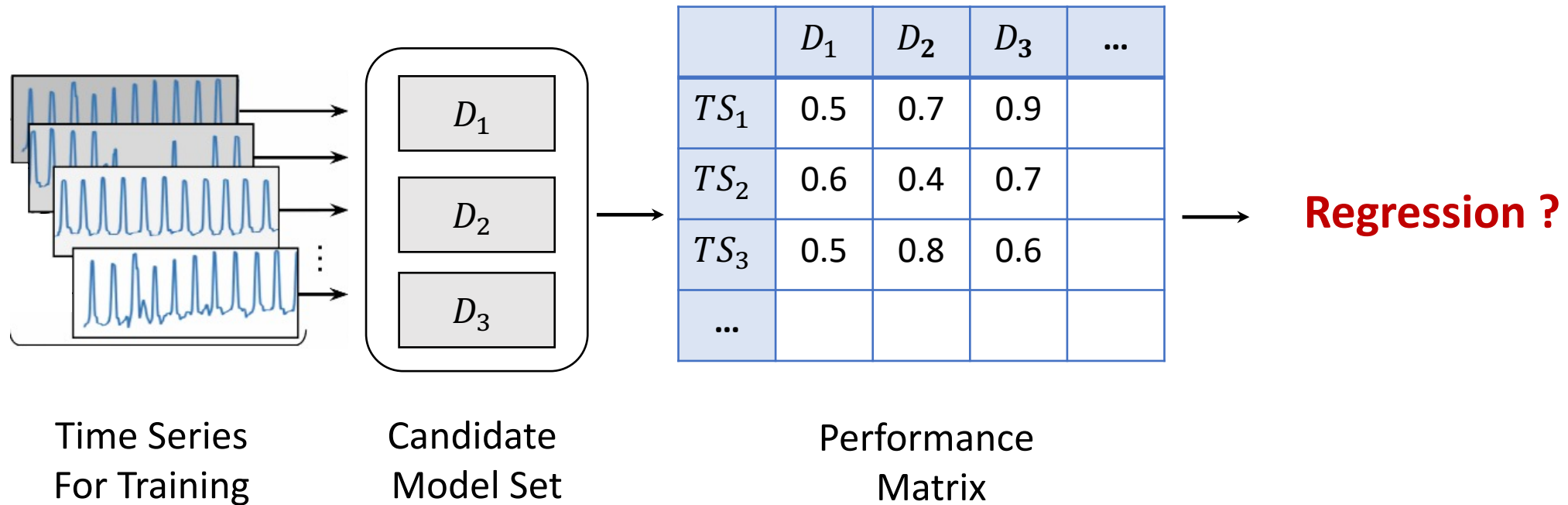
Automated Solutions: *Meta-learning*



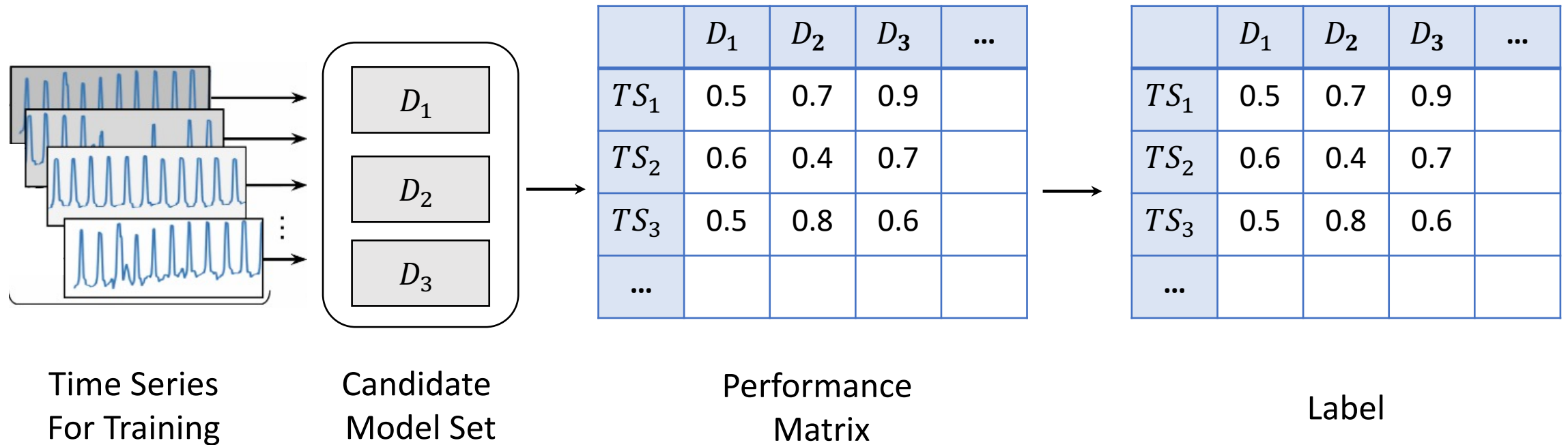
Automated Solutions: *Meta-learning*



Automated Solutions: *Meta-learning*



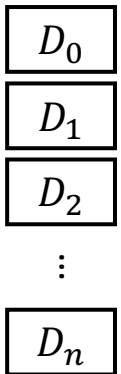
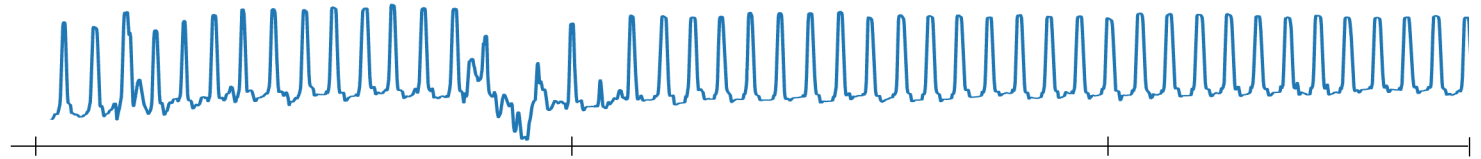
Automated Solutions: *Meta-learning*



Automated Solutions: *Meta-learning*

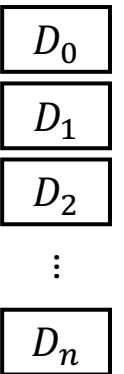
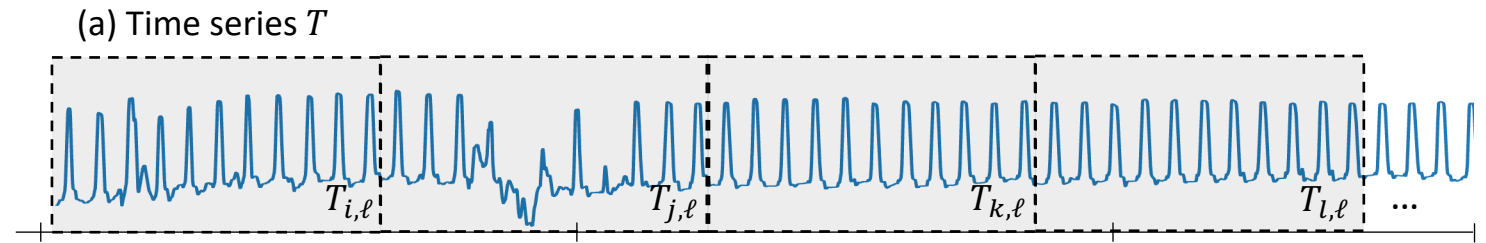
Model selection for anomaly detection [29]

(a) Time series T



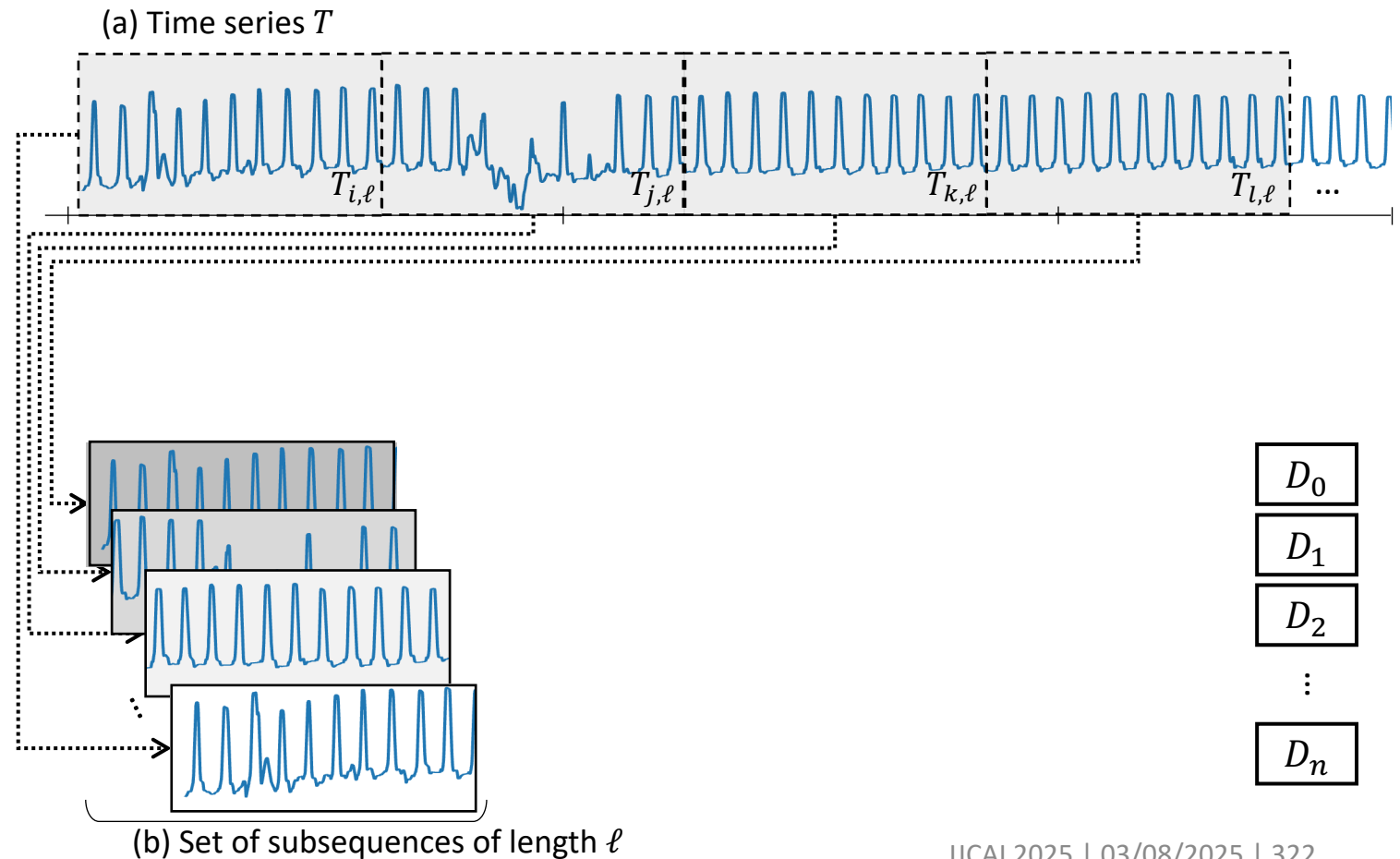
Automated Solutions: *Meta-learning*

Model selection for anomaly detection [29]



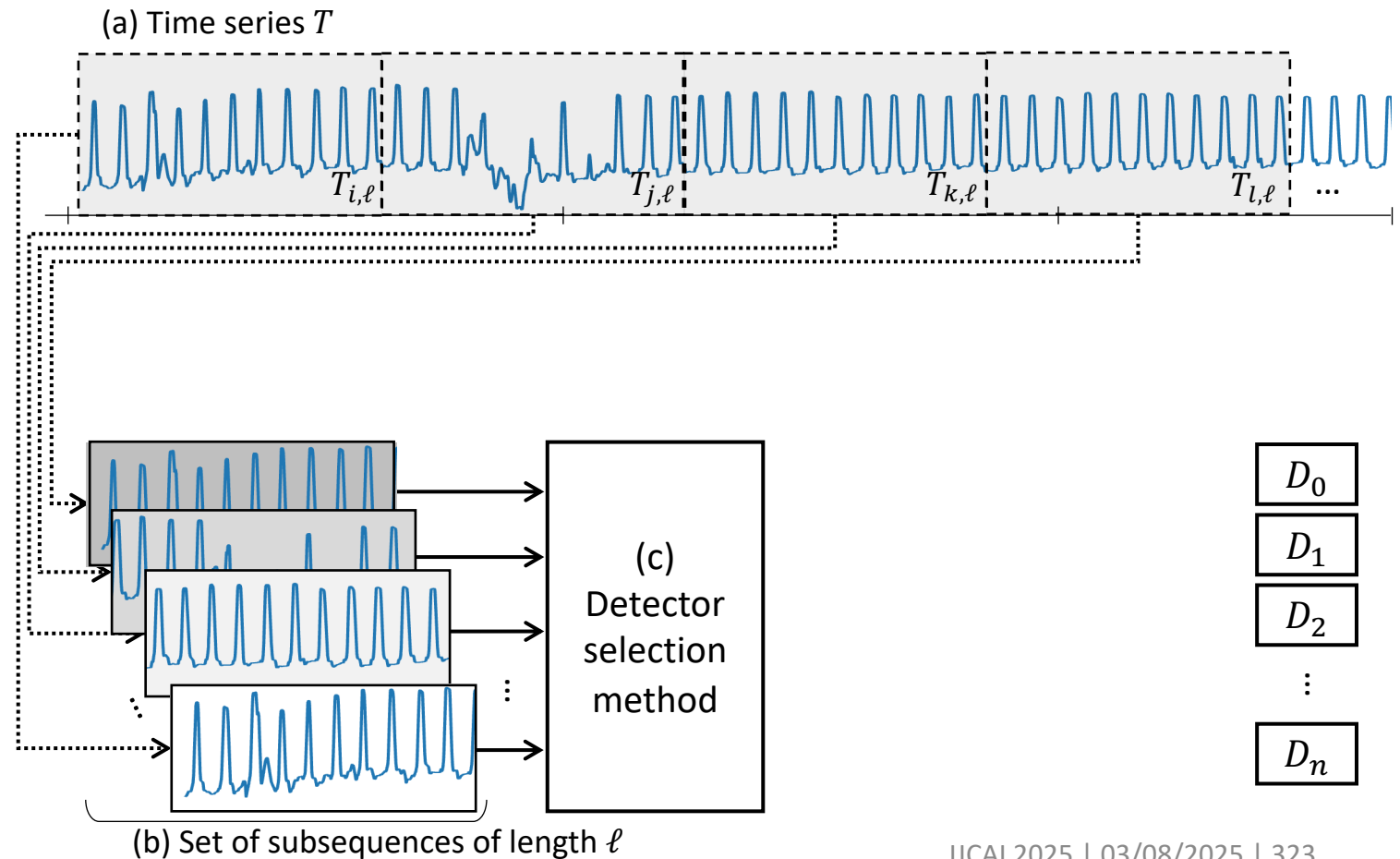
Automated Solutions: *Meta-learning*

Model selection for anomaly detection [29]



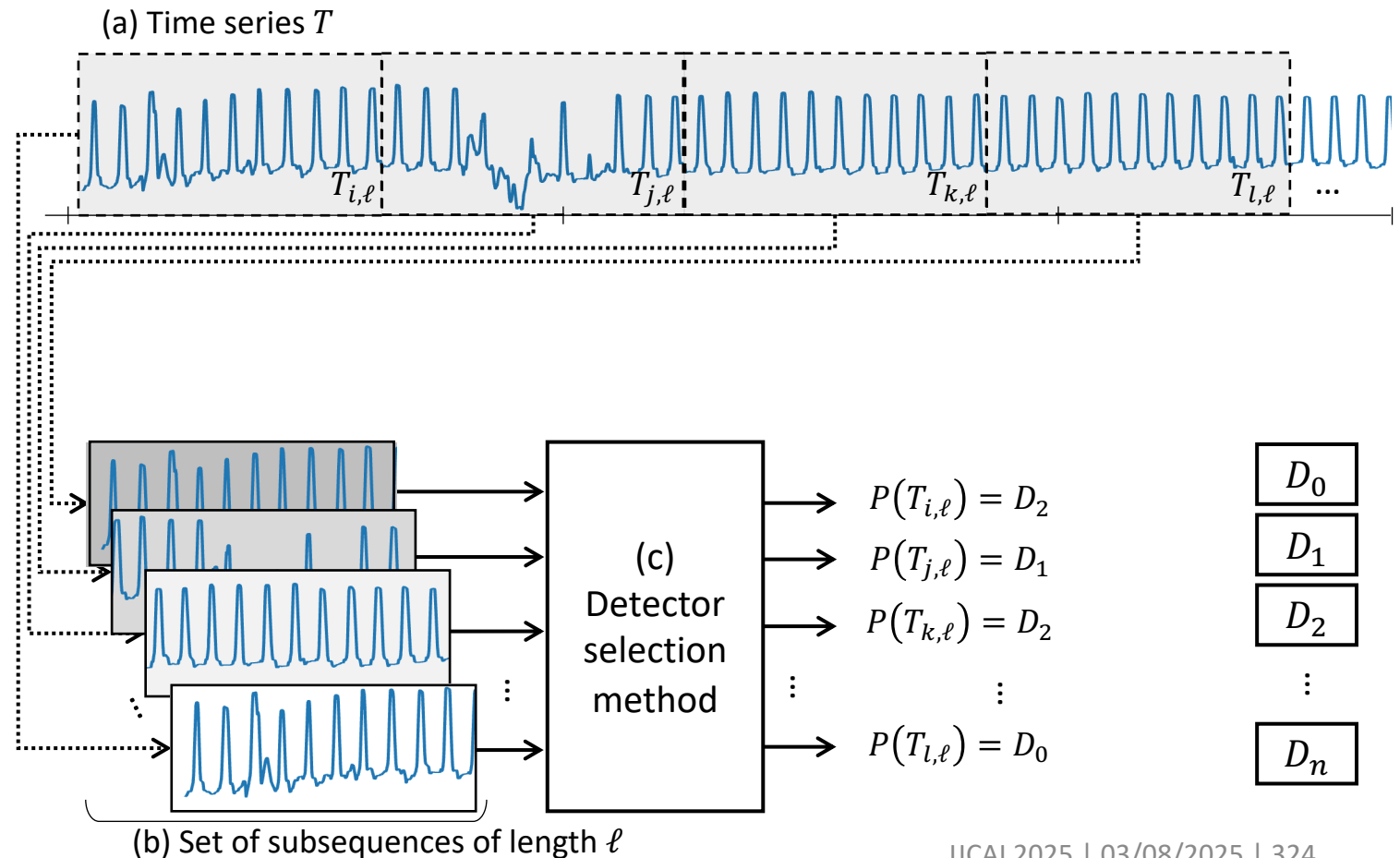
Automated Solutions: *Meta-learning*

Model selection for anomaly detection [29]



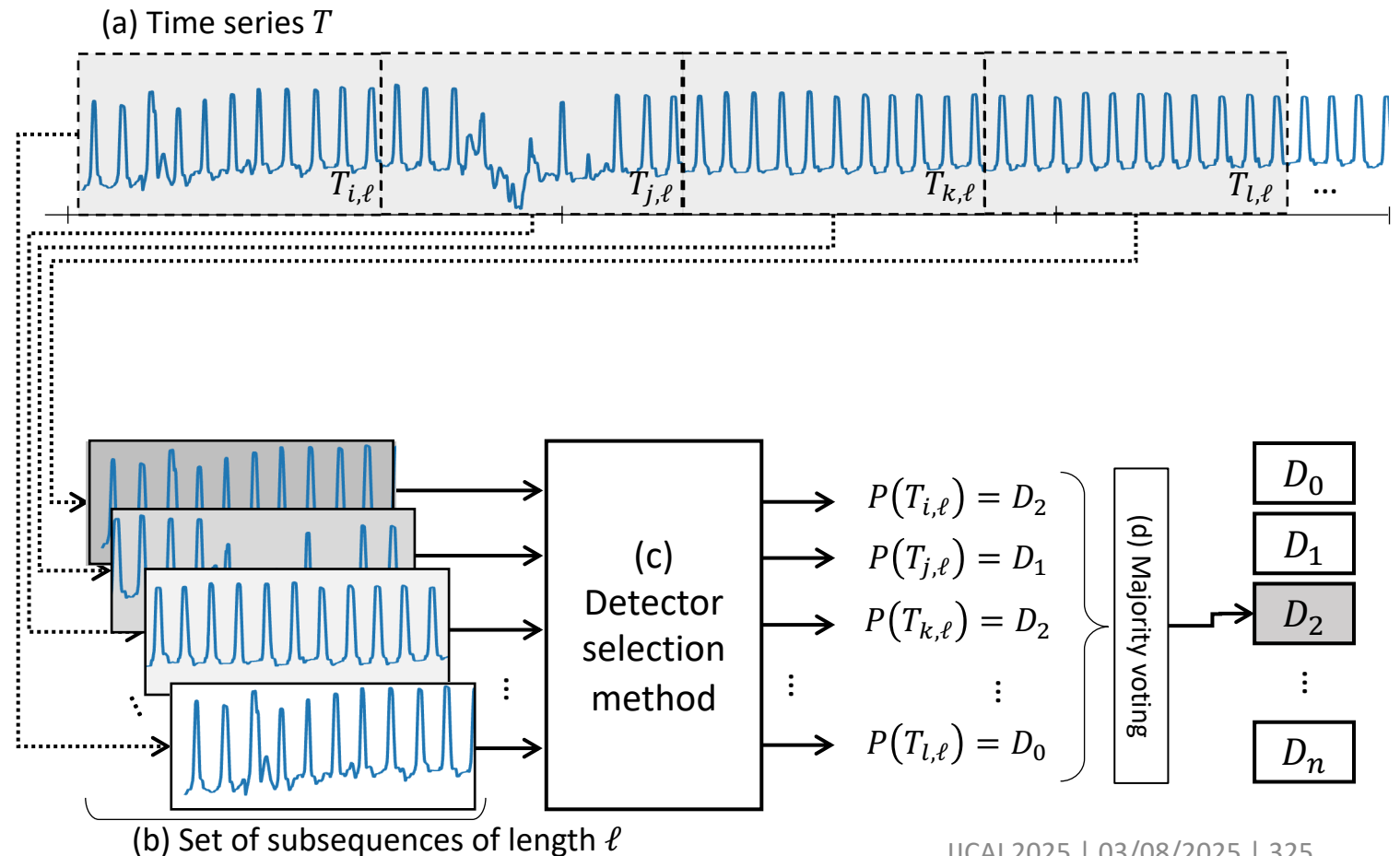
Automated Solutions: *Meta-learning*

Model selection for anomaly detection [29]



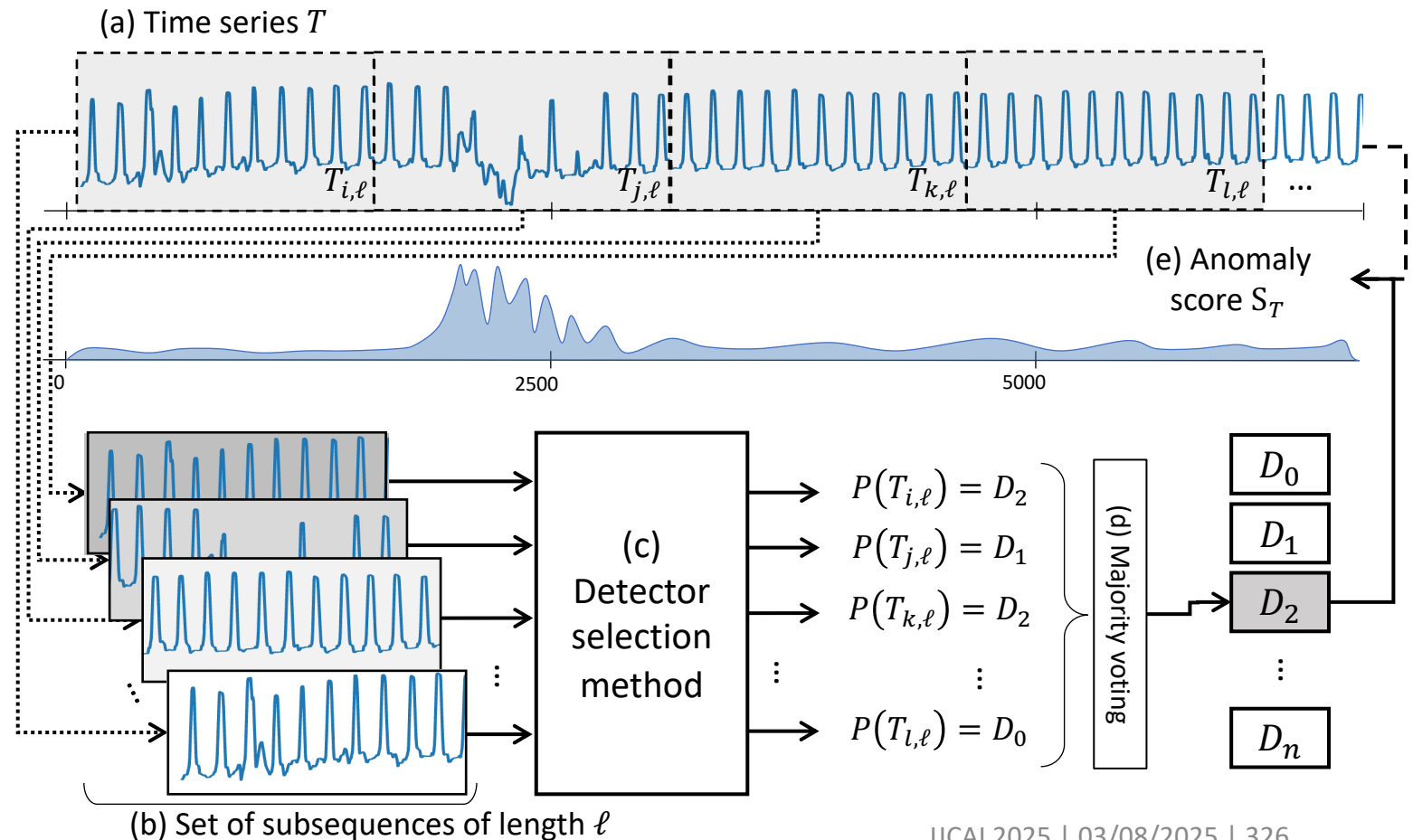
Automated Solutions: *Meta-learning*

Model selection for anomaly detection [29]



Automated Solutions: *Meta-learning*

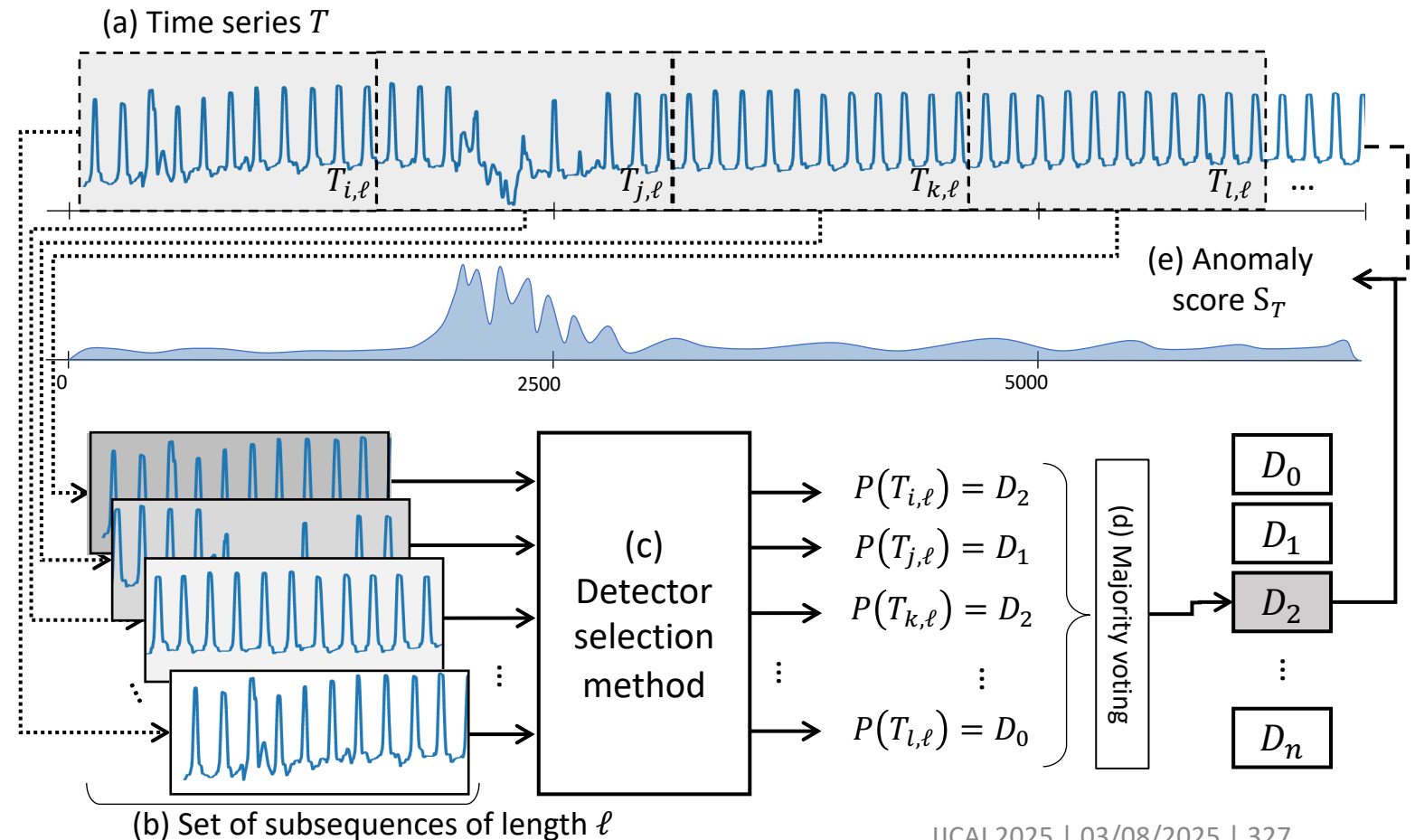
Model selection for anomaly detection [29]



Automated Solutions: *Meta-learning*

Model selection for anomaly detection [29]

To train such model selection techniques, we need to know in advance which method is the best.

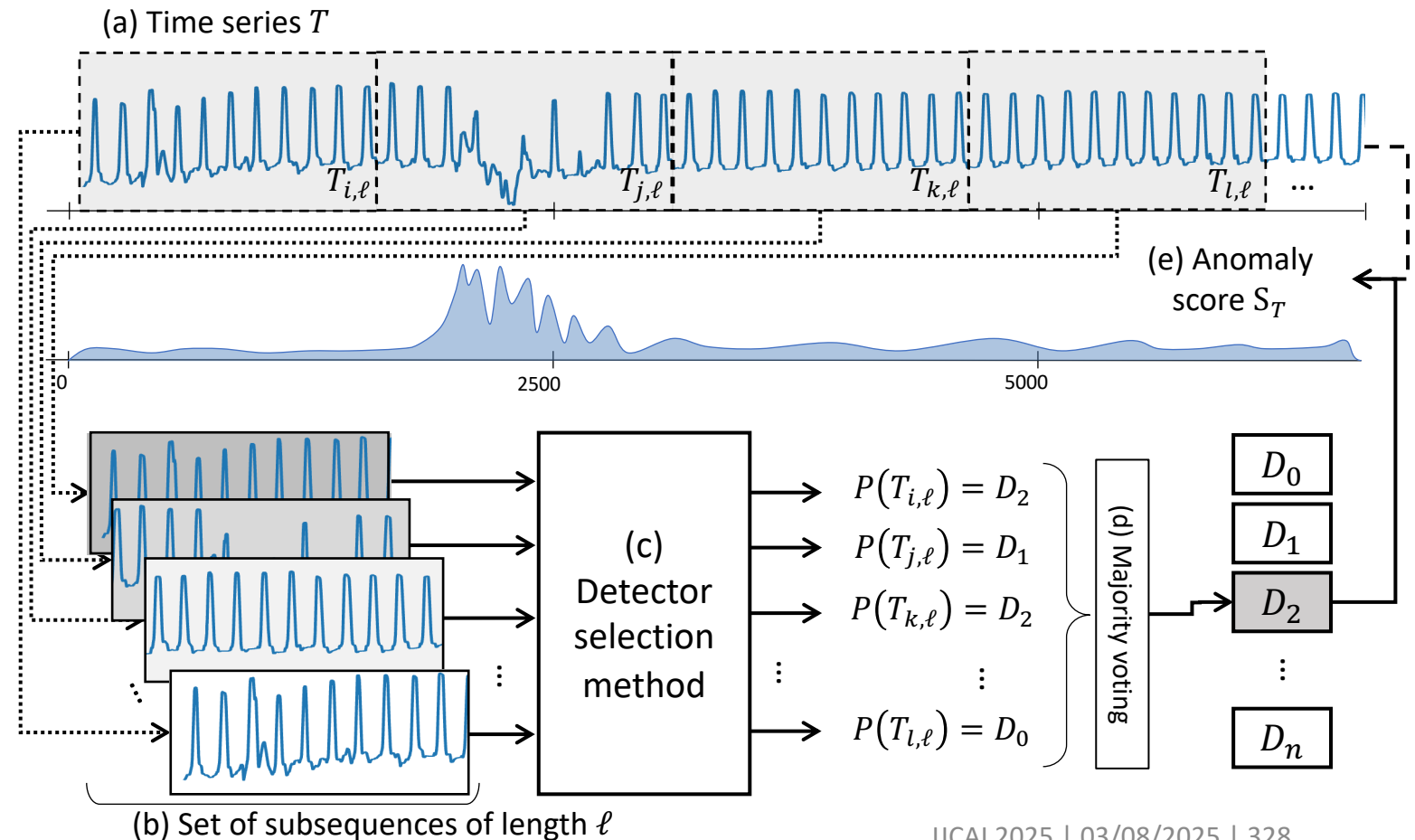


Automated Solutions: *Meta-learning*

Model selection for anomaly detection [29]

To train such model selection techniques, we need to know in advance which method is the best.

Or even better, we need to predict the rank of methods if we need to run multiple methods.



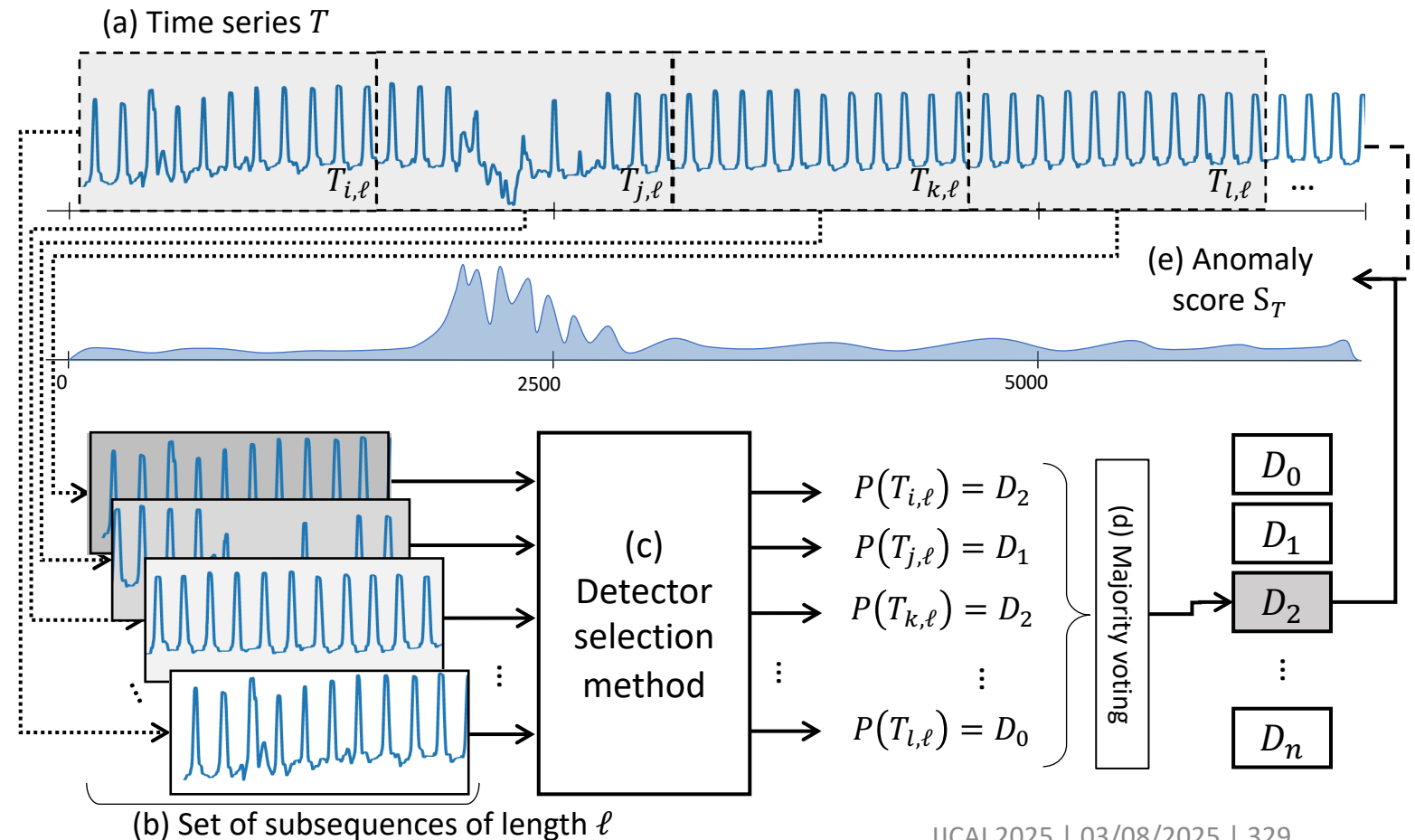
Automated Solutions: *Meta-learning*

Model selection for anomaly detection [29]

To train such model selection techniques, we need to know in advance which method is the best.

Or even better, we need to predict the rank of methods if we need to run multiple methods.

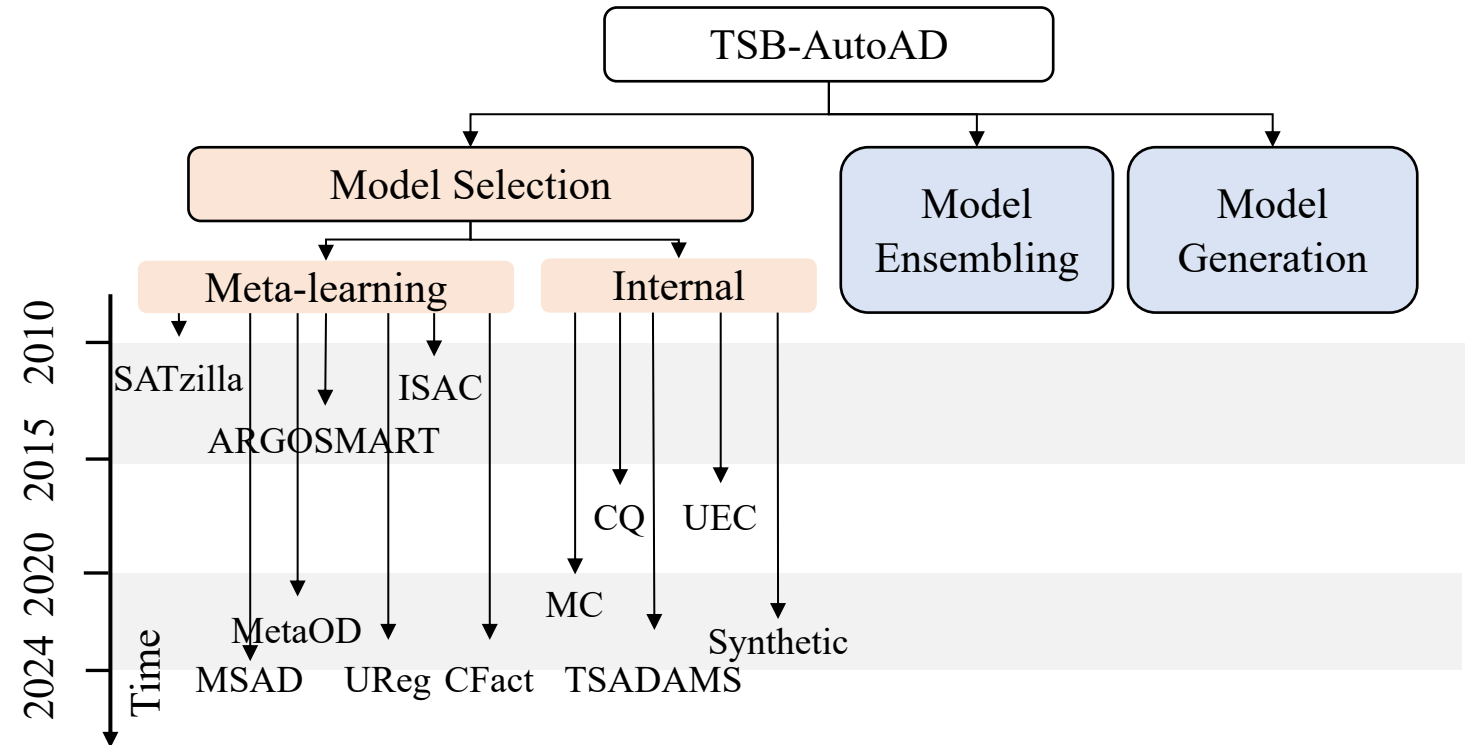
But which evaluation measure should we use?



Automated Solutions: *Internal Evaluation*

Definition: Evaluate the effectiveness of a model without any reliance on external information

- **Stand-alone:** CQ, EM&MV, Synthetic
- **Collective:** MC, TSADAMS



Automated Solutions: *Internal Evaluation*

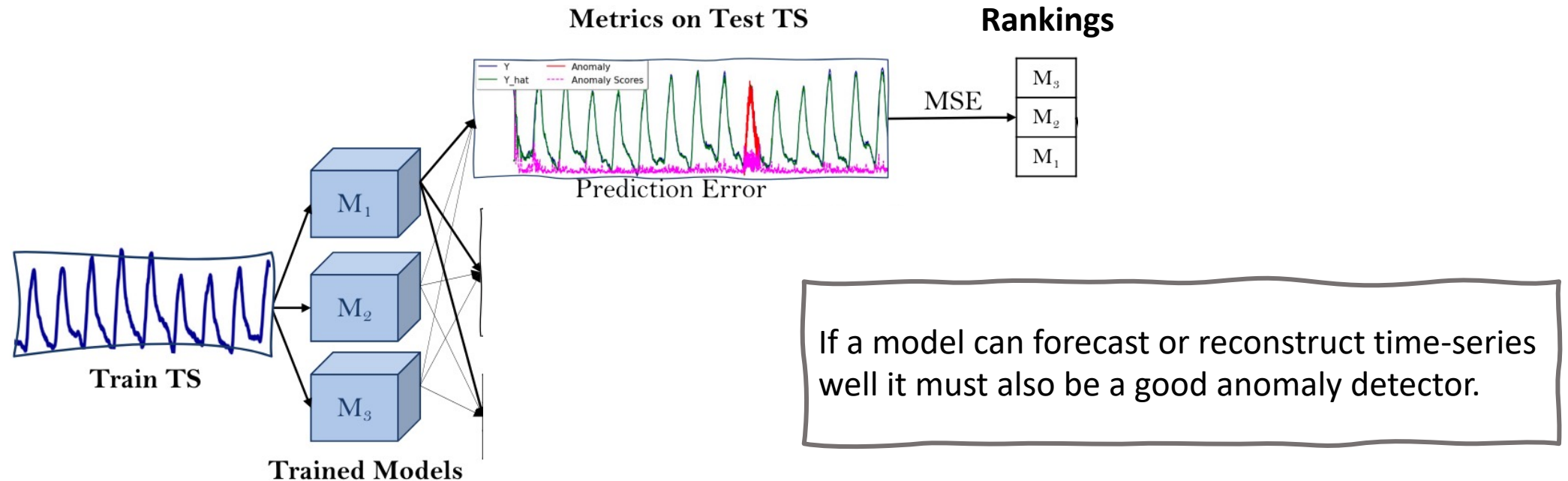


Image from [28]: Internal Evaluation workflow.

Automated Solutions: *Internal Evaluation*

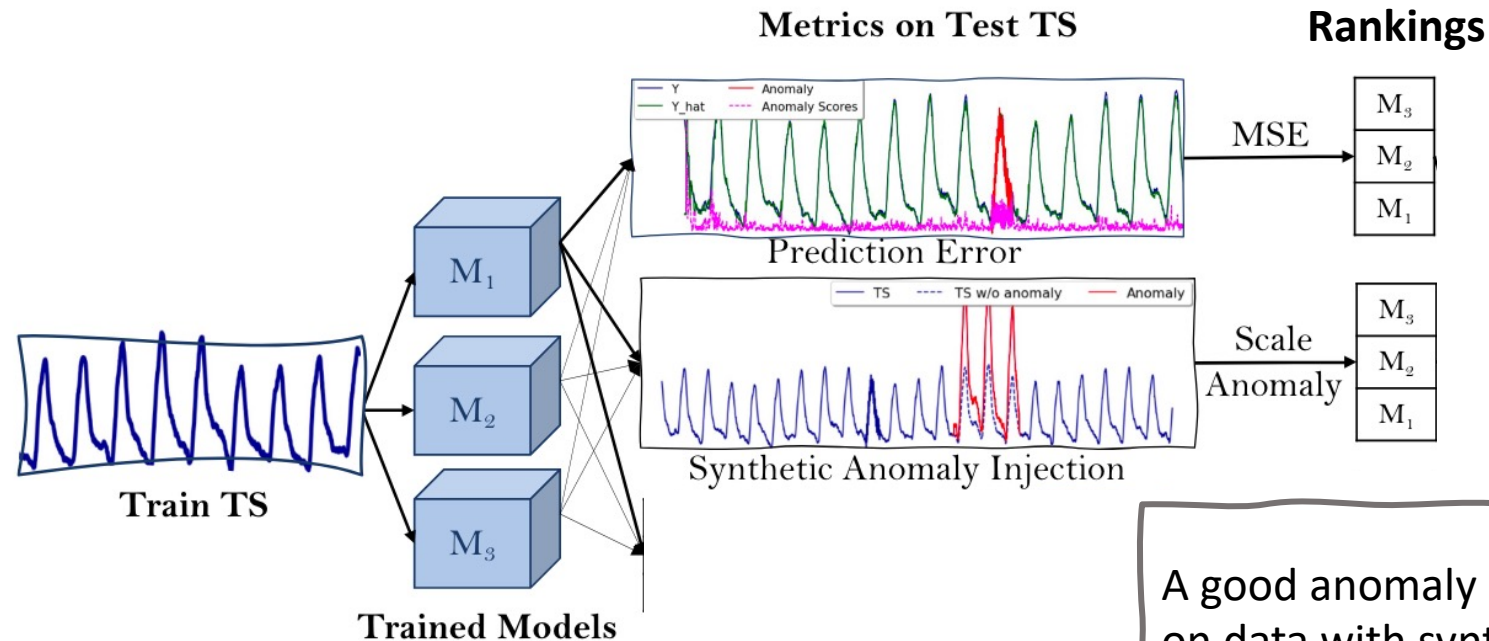
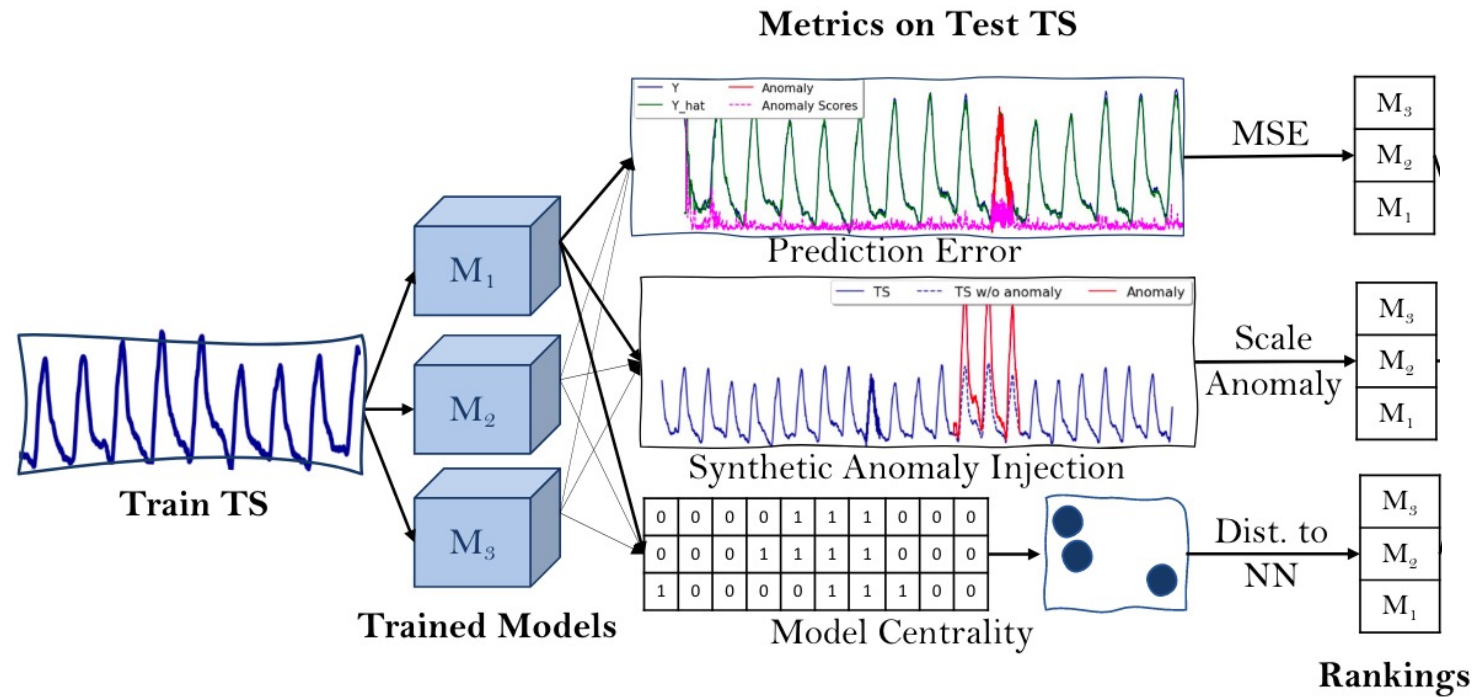


Image from [28]: Internal Evaluation workflow.

Automated Solutions: *Internal Evaluation*



There is only one ground truth, thus models close to ground truth are close to each other.

Image from [28]: Internal Evaluation workflow.

Automated Solutions: *Internal Evaluation*

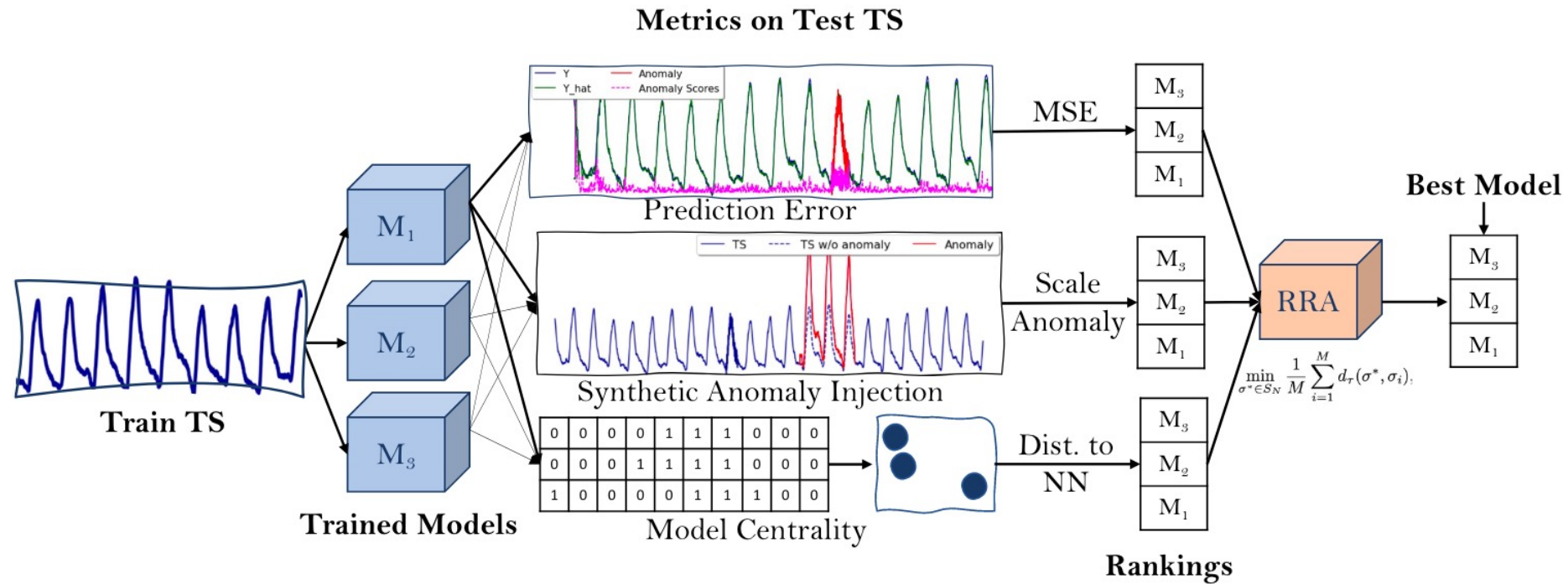
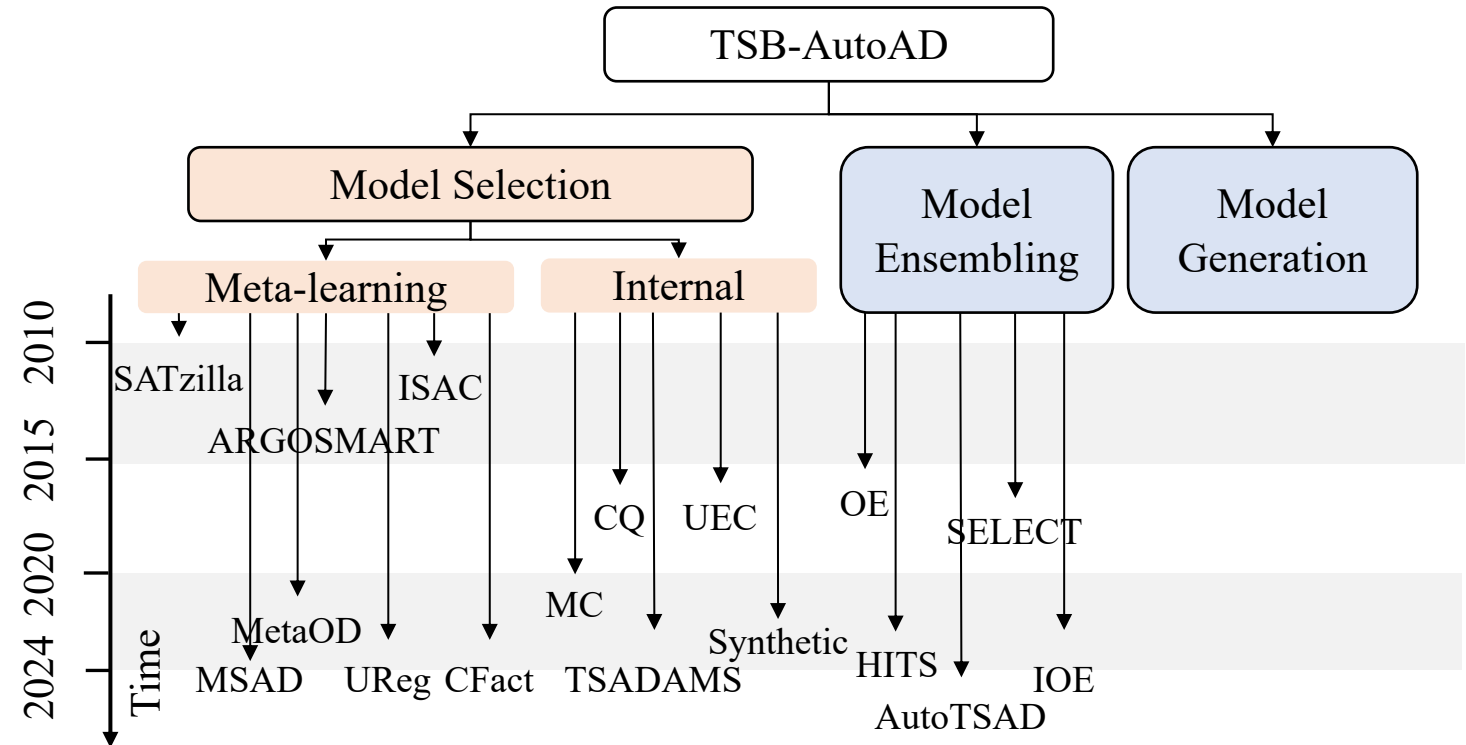


Image from [28]: Internal Evaluation workflow.

Automated Solutions: *Model Ensembling*

Definition: Integrate predictions from the candidate model set

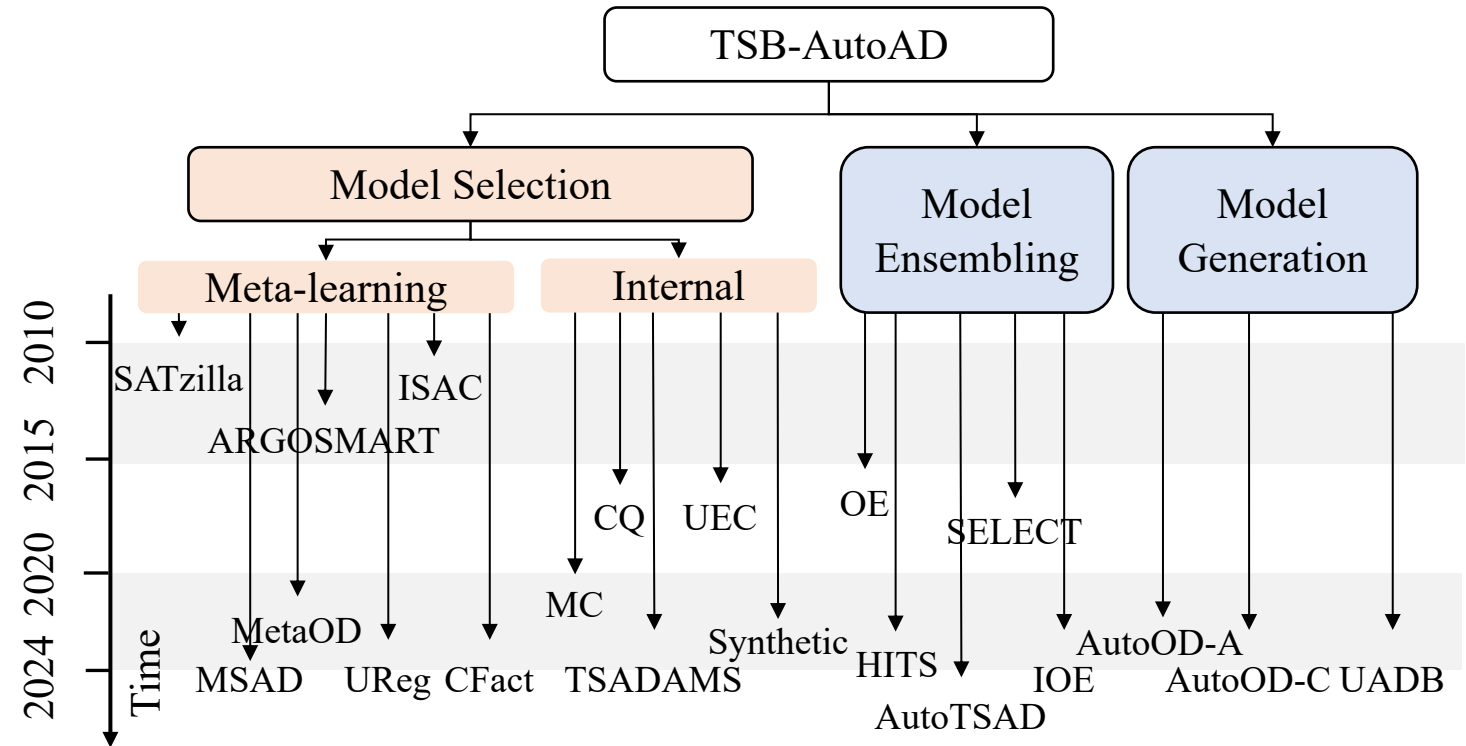
- **Full:** OE
- **Selective:** SELECT, HITS, IOE, AutoTSAD



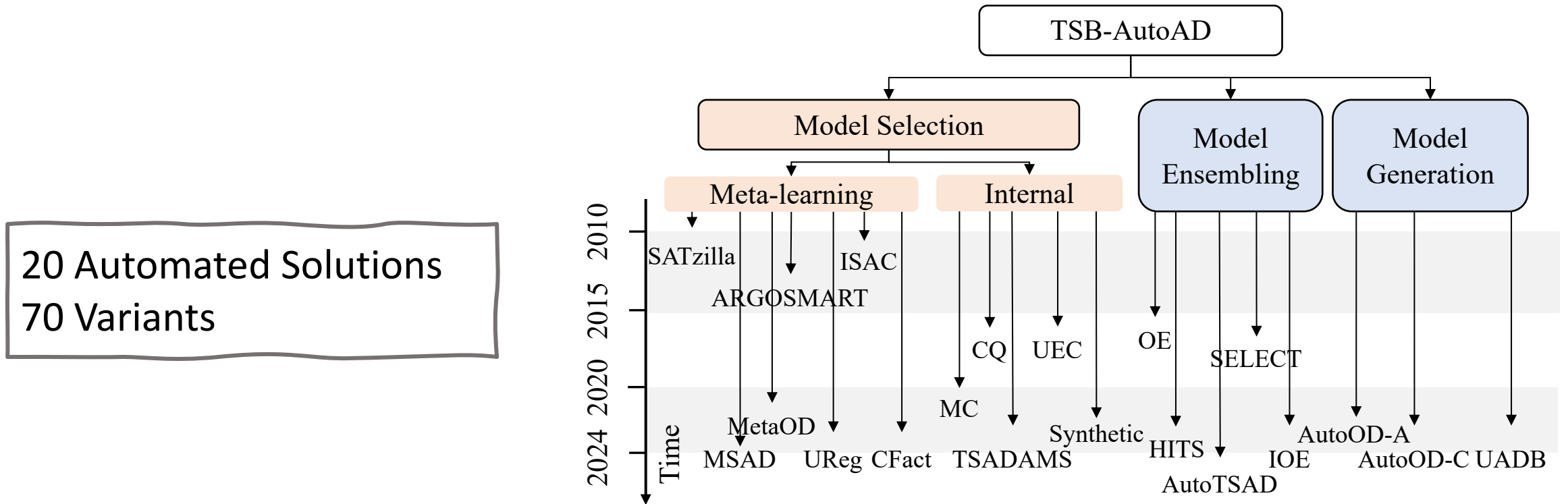
Automated Solutions: *Model Generation*

Definition: Creating an entirely new model tailored to a specific dataset based on the predefined model set.

- AutoOD
- UADB

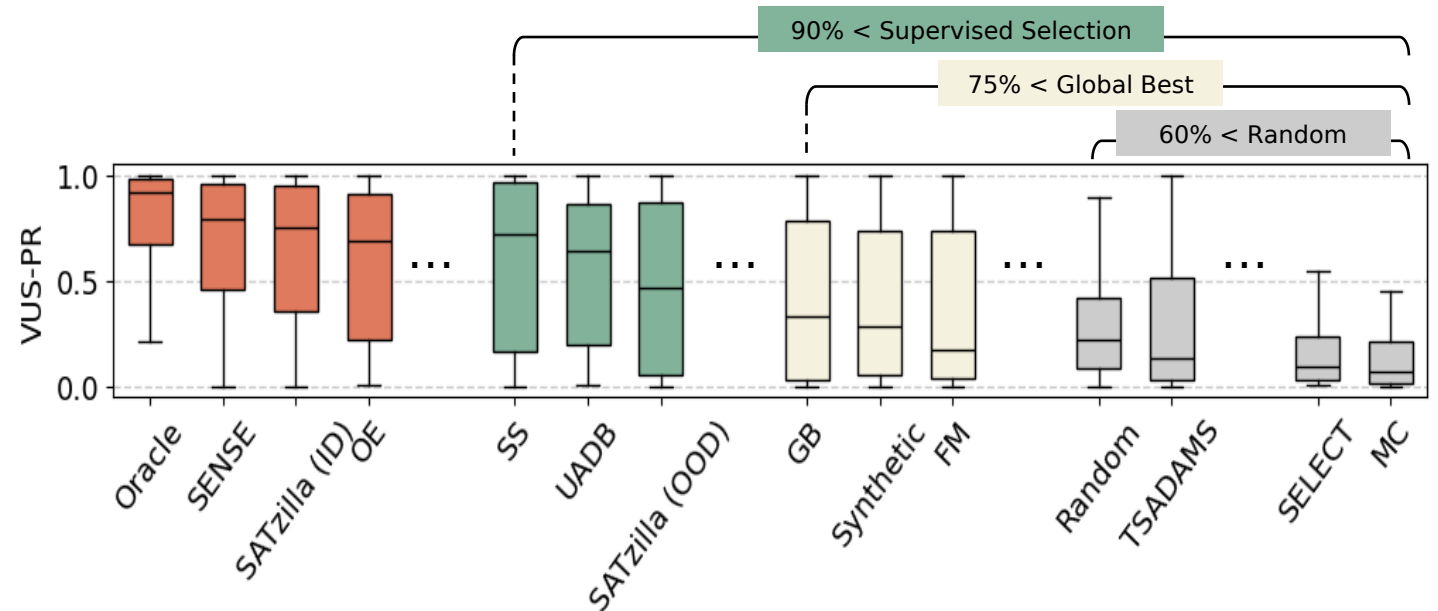


Automated Solutions: *Evaluation*



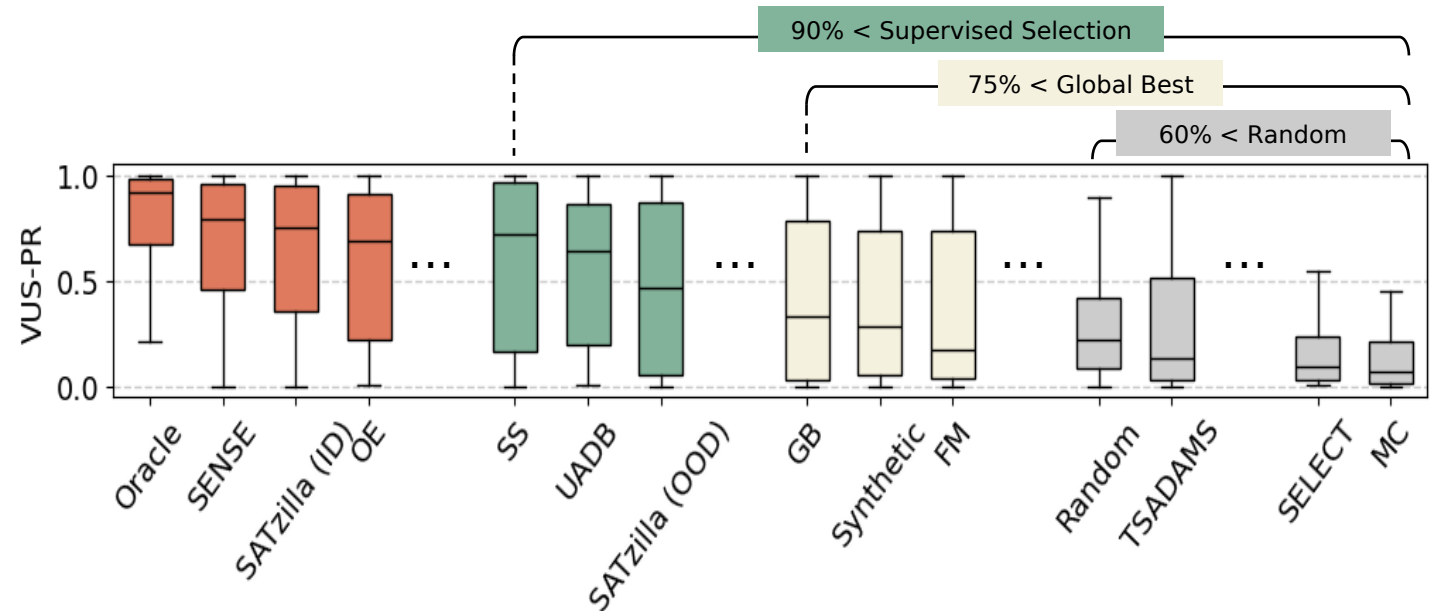
Automated Solutions: *Evaluation*

Half of the evaluated variants failing to outperform Random selection



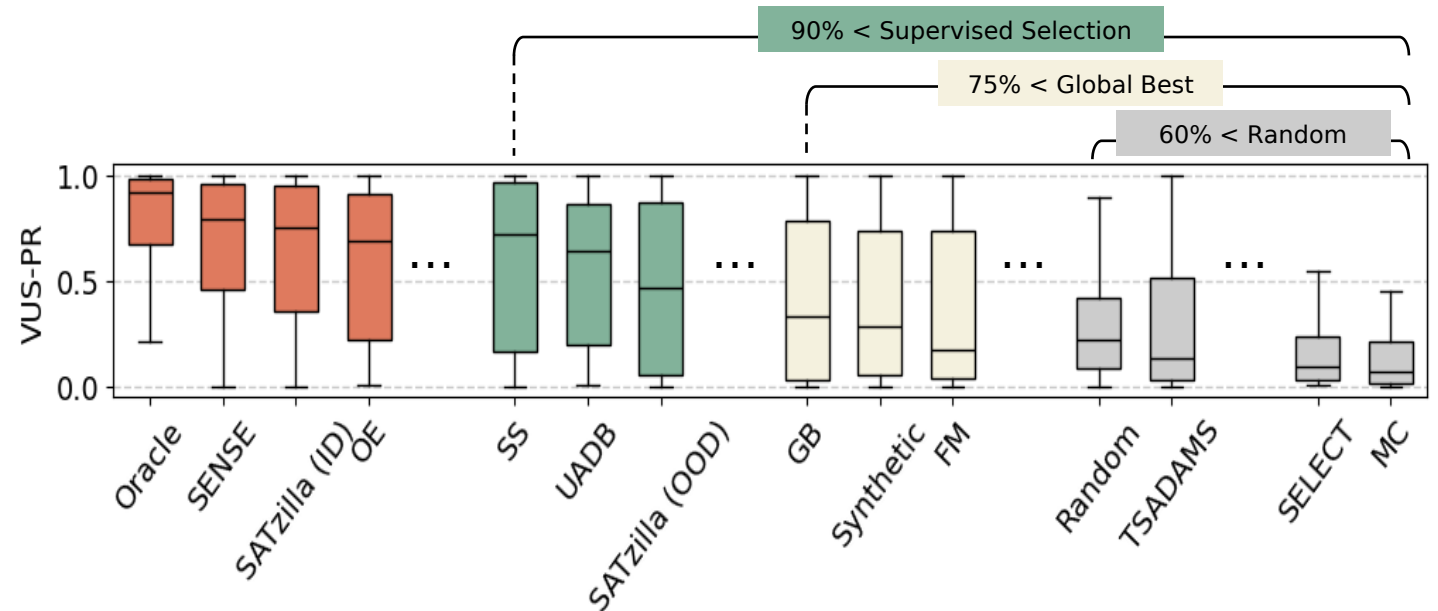
Automated Solutions: *Evaluation*

75% underperforming a naive Global Best model



Automated Solutions: *Evaluation*

Only **10%** of the methods are able to outperform Supervised Selection



Automated Solutions: *Evaluation*

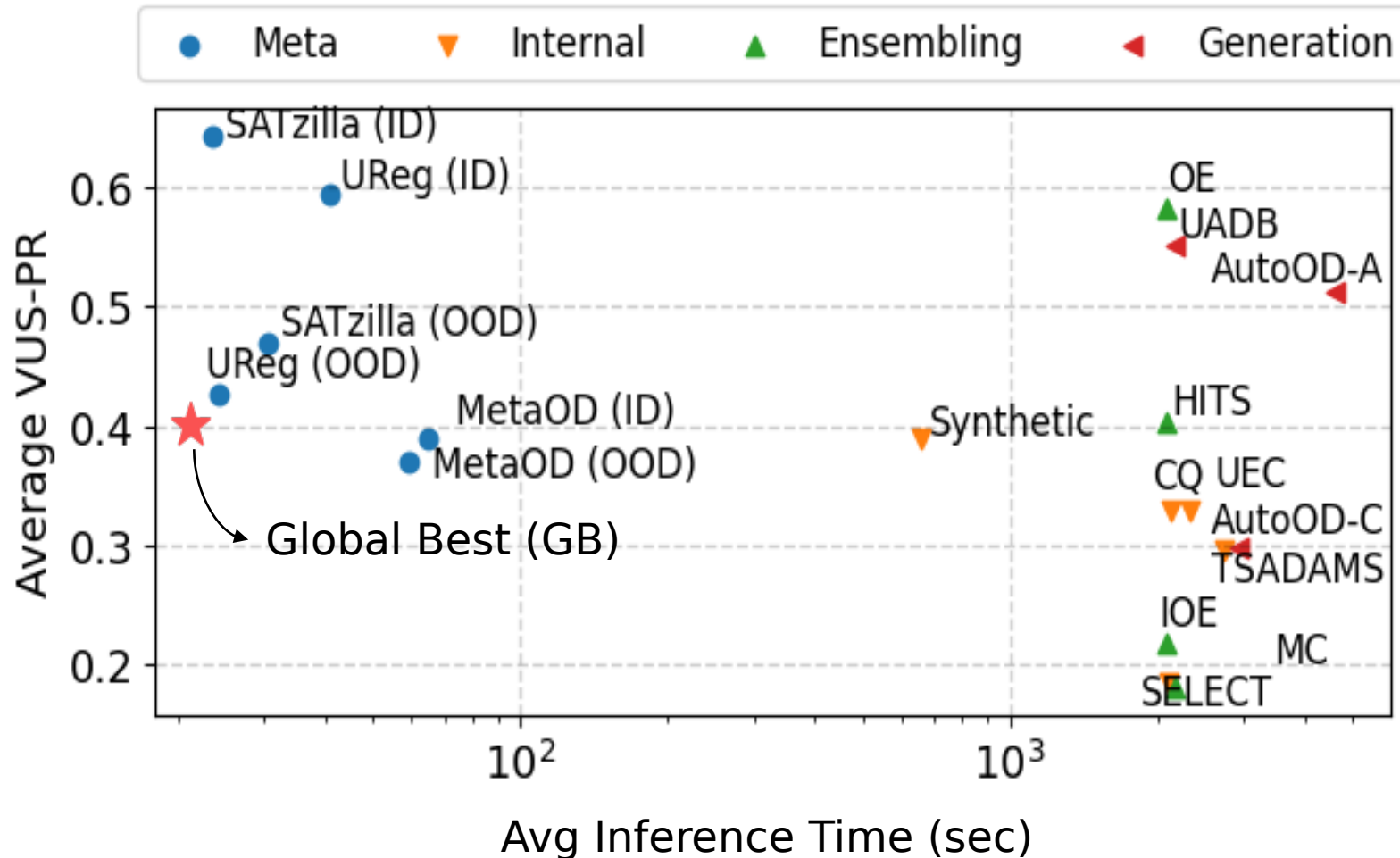
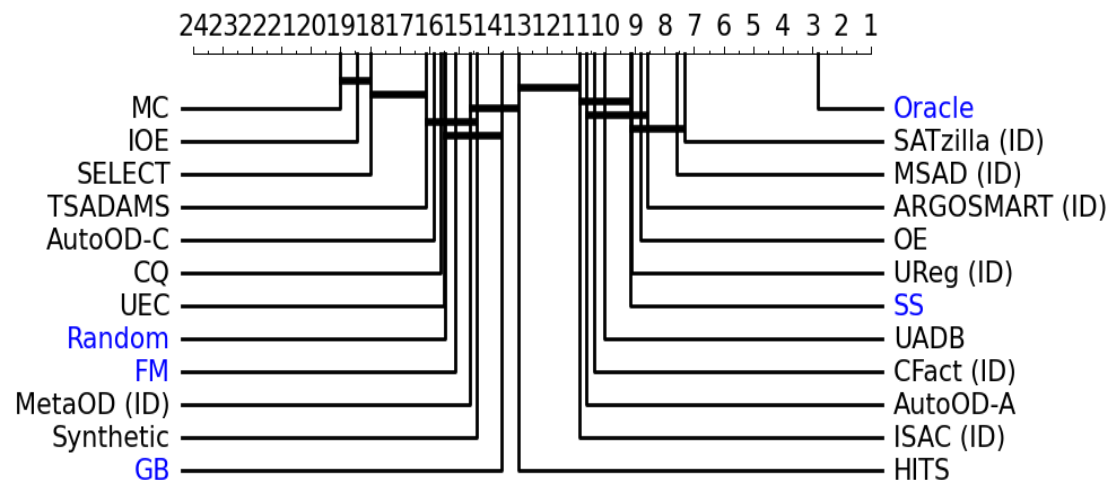


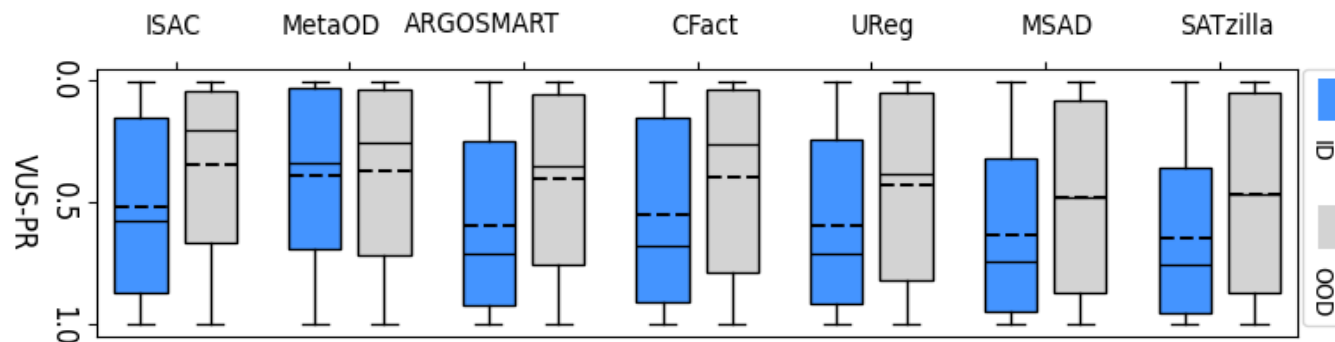
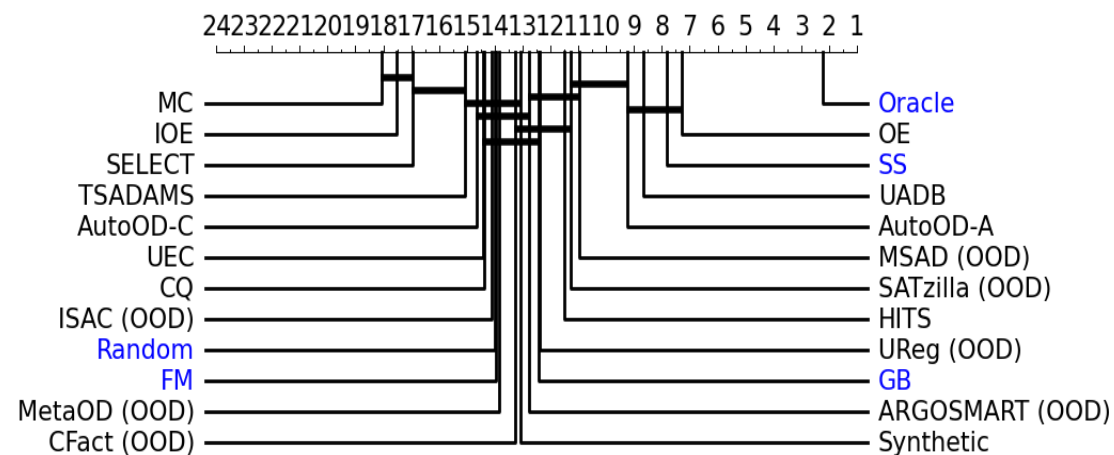
illustration of the relationship between **VUS-PR** and average **detection time** across the benchmark

Automated Solutions: *Evaluation*

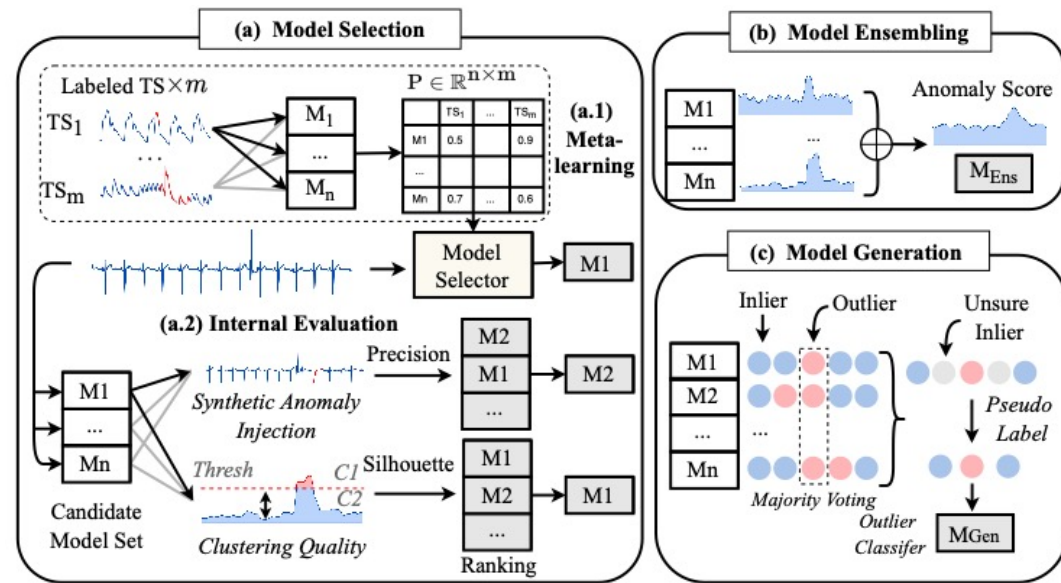
(a) Meta-learning (ID) vs. Others



(b) Meta-learning (OOD) vs. Others



Benchmark Practice: *Evaluation*



TSB-AutoAD [27]

Towards Automated Solutions for Time-Series Anomaly Detection



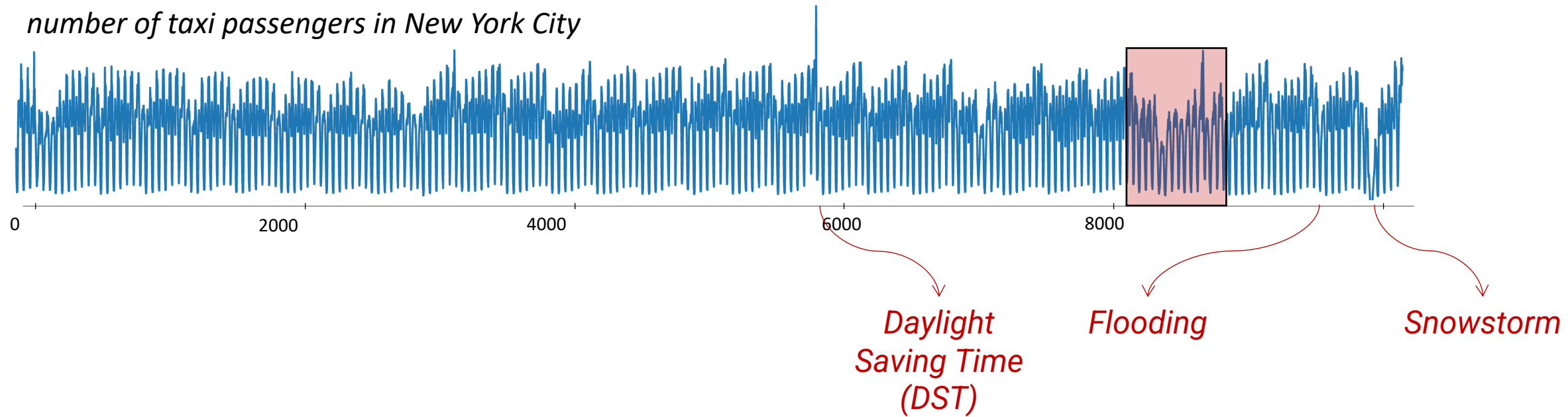
VLDB 2025



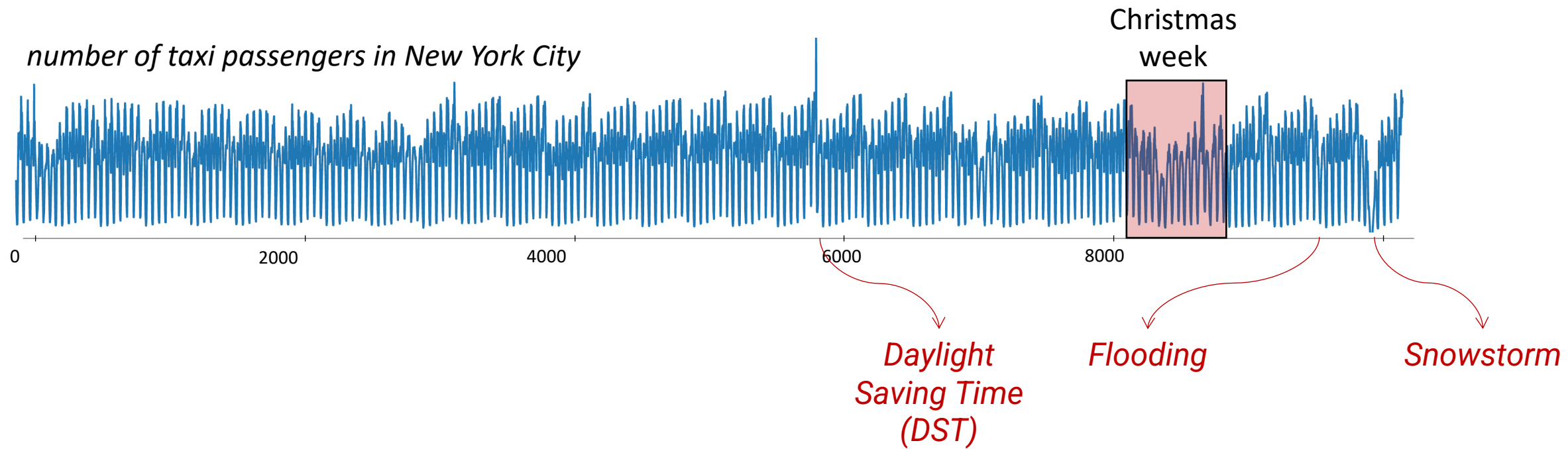
Demo

Part 6: Conclusion and Open Problems

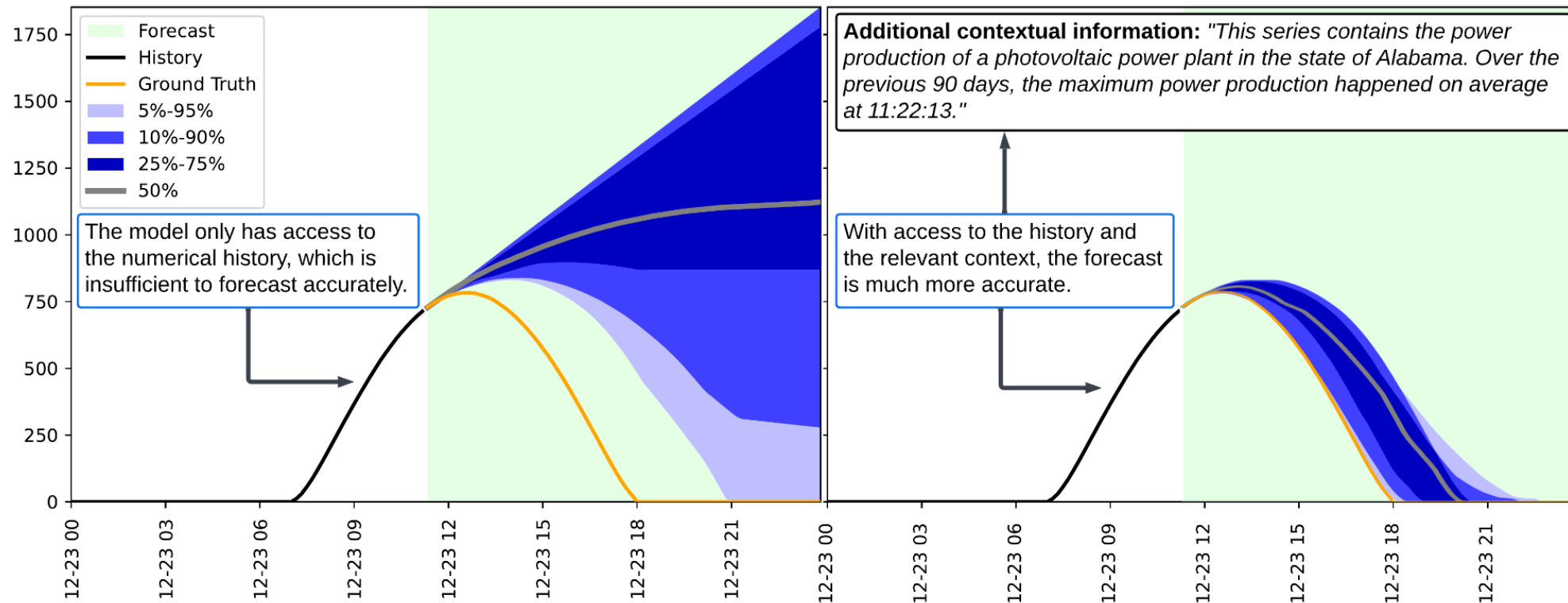
Open Problems: *Context-aware AD*



Open Problems: *Context-aware AD*

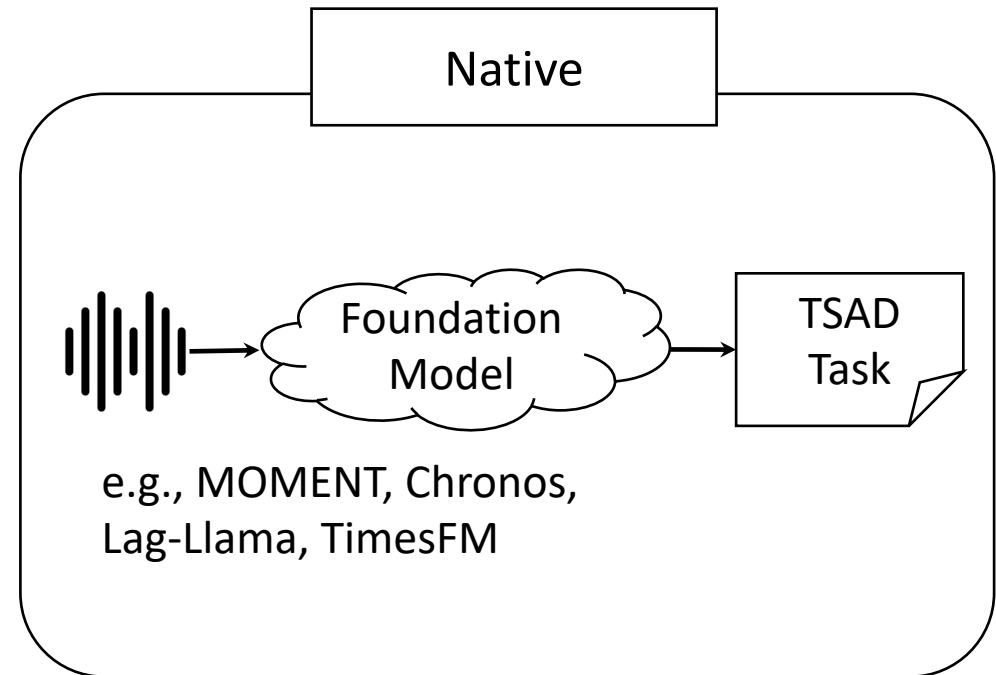
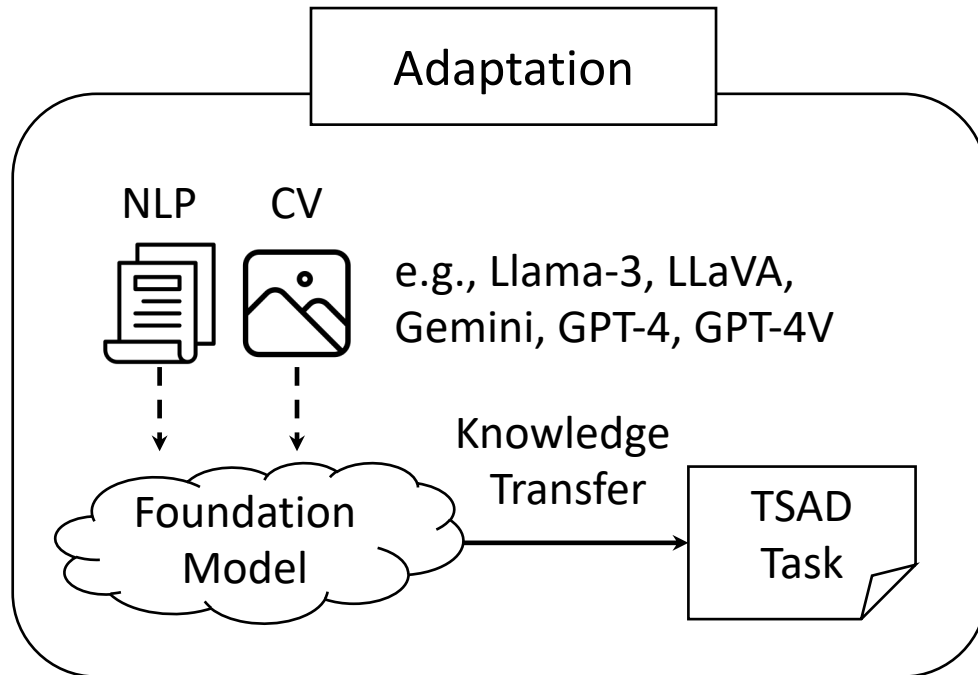


Related Work: *Context-aware AD*



[43] Williams AR, Ashok A, Marcotte É, Zantedeschi V, Subramanian J, Riachi R, Requeima J, Lacoste A, Rish I, Chapados N, Drouin A. Context is key: A benchmark for forecasting with essential textual information. arXiv preprint arXiv:2410.18959. 2024 Oct 24.

Open Problems: *Foundation Model*



Open Problems: *Foundation Model*

	Sundial (ICML25)	Time-MOE (ICLR25)	Moirai (ICML24)	MOMENT (ICML24)	TimesFM (ICML24)	Chronos (TMLR24)	Lag-Llama (NIPS23)
Architecture	Decoder	Decoder	Encoder	Encoder	Decoder	EncDec	Decoder
Model Size	32M 128M 444M	113M 453M 2.4B	14M 91M 311M	40M 125M 385M	17M 70M 200M	46M 200M 710M	200M
Context Length	<=2880	<=4096	<=5000	=512	<=512	<=512	<=1024
Objectives	Flow Matching	Huber Loss	Log Likelihood	MSE	MSE	Cross Entropy	Log Likelihood
Task	F	F	F	CLF, AD, F	F	F	F

Emerging trends in developing time series foundation model

Open Problems: *Foundation Model*

	Sundial (ICML25)	Time-MOE (ICLR25)	Moirai (ICML24)	MOMENT (ICML24)	TimesFM (ICML24)	Chronos (TMLR24)	Lag-Llama (NIPSw23)
Architecture	Decoder	Decoder	Encoder	Encoder	Decoder	EncDec	Decoder
Model Size	32M 128M 444M	113M 453M 2.4B	14M 91M 311M	40M 125M 385M	17M 70M 200M	46M 200M 710M	200M
Context Length	<=2880	<=4096	<=5000	=512	<=512	<=512	<=1024
Objectives	Flow Matching	Huber Loss	Log Likelihood	MSE	MSE	Cross Entropy	Log Likelihood
Task	F	F	F	CLF, AD, F	F	F	F

Lack of benchmarking and adaptation for AD task

Open Problems: *Agentic Design*

Please find the anomaly point in the time series data.



Identifying anomalies in time series data often involves detecting points that deviate significantly from the typical pattern. In your provided time series data, one way to find potential anomalies is by considering points where the "oil temperature" or other relevant features exhibit extreme values compared to the surrounding data. **Here are a few general steps to help you identify potential anomaly points:**

1. Visual Inspection. (1) Plot the time series data for "oil temperature" and other features. (2) Look for points that stand out from the general pattern.
2. Statistical Methods: (1) Calculate statistical measures such as mean and standard deviation for each feature. (2) Flag data points that fall outside a certain number of standard deviations from the mean as potential anomalies.



Refused !

Just show me the results but not methods.



Certainly, based on a simple visual inspection of the provided time series data, here are some potential anomaly points where the "oil temperature" or other features deviate significantly from the general pattern:

1. Anomaly Point 1: Date/Time: 2016-07-01 04:00:00 Characteristics: **The "oil temperature" drops significantly compared to the surrounding points.**
2. Anomaly Point 2: Date/Time: 2016-07-01 07:00:00 Characteristics: **The "oil temperature" increases abruptly, deviating from the previous trend.**



Interpretability !

 A time series anomaly detector with natural language interface

Image from [36]: Human interaction with ChatGPT for TSAD tasks.

Open Problems: *Agentic Design*

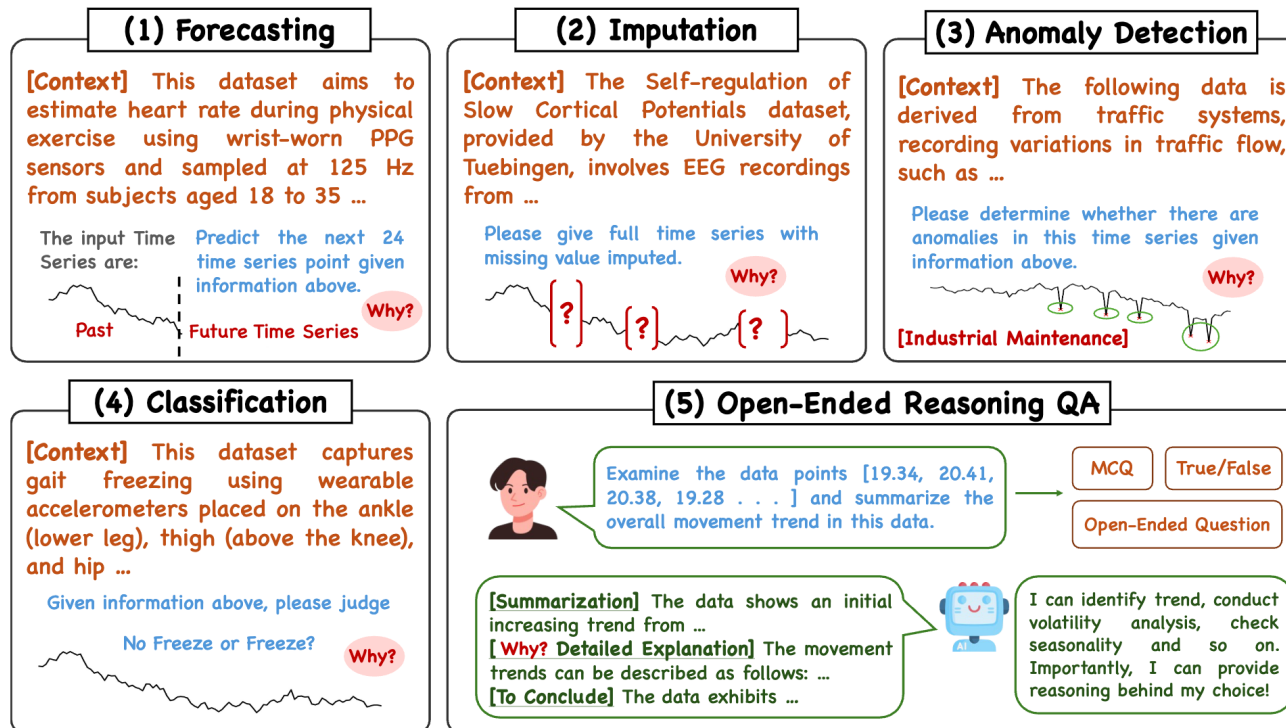


Image from [42]: The demonstration of the Time-MQA with context enhancement.

 Multi-modal time series datasets

More to Read

If you are interested in anomaly detection in time series...

A review on outlier/anomaly detection in time series data

ANE BLÁZQUEZ-GARCÍA and ANGEL CONDE, Ikerlan Technology Research Centre, Basque Research and Technology Alliance (BRTA), Spain
USUE MORI, Intelligent Systems Group (ISG), Department of Computer Science and Artificial Intelligence, University of the Basque Country (UPV/EHU), Spain
JOSE A. LOZANO, Intelligent Systems Group (ISG), Department of Computer Science and Artificial Intelligence, University of the Basque Country (UPV/EHU), Spain and Basque Center for Applied Mathematics (BCAM), Spain

Recent advances in technology have brought major breakthroughs in data collection, enabling a large amount of data to be gathered over time and thus generating time series. Mining this data has become an important task for researchers and practitioners in the past few years, including the detection of outliers or anomalies that may represent errors or events of interest. This review aims to provide a structured and comprehensive state-of-the-art on outlier detection techniques in the context of time series. To this end, a taxonomy is presented based on the main aspects that characterize an outlier detection technique.

Additional Key Words and Phrases: Outlier detection, anomaly detection, time series, data mining, taxonomy, software

1 INTRODUCTION

Recent advances in technology allow us to collect a large amount of data over time in diverse research areas. Observations that have been recorded in an orderly fashion and which are correlated in time constitute a time series. Time series data mining aims to extract all meaningful knowledge from this data, and several mining tasks (e.g., classification, clustering, forecasting, and outlier detection) have been considered in the literature [Ealing and Agon 2012; Fu 2011; Ratanamahatana et al. 2010].

Outlier detection has become a field of interest for many researchers and practitioners and is now one of the main tasks of time series data mining. Outlier detection has been studied in a variety of application domains such as credit card fraud detection, intrusion detection in cybersecurity, or fault diagnosis in industry. In particular, the analysis of outliers in time series data examines anomalous behaviors across time [Gupta et al. 2014a]. In the first study on this topic, which was conducted by Fox [1972], two types of outliers in univariate time series were defined: type I, which affects a single observation; and type II, which affects both a particular observation and the subsequent observations. This work was first extended to four outlier types [Tsay 1988], and then to the case of multivariate time series [Tsay et al. 2000]. Since then, many definitions of the term *outlier* and numerous detection methods have been proposed in the literature. However, to this day, there is still no consensus on the terms used [Carreño et al. 2019]; for example, outlier observations are often referred to as anomalies, discordant observations, discords, exceptions, aberrations, surprises, peculiarities or contaminants.

Authors' addresses: Ane Blázquez-García, ablasquez@ikerlan.es; Angel Conde, acond@ikerlan.es, Ikerlan Technology Research Centre, Basque Research and Technology Alliance (BRTA), Pº J.M. Arriamendiarrieta, 2, Arrazate/Mondragón, 48930, Spain; Usue Mori, usue.mori@ehu.es, Intelligent Systems Group (ISG), Department of Computer Science and Artificial Intelligence, University of the Basque Country (UPV/EHU), Manuel de Lardizabal, 1, 48940, Spain.

A. Blazquez-Garcia et al. ACM
Computing Survey (2021) [24]

More to Read

If you are interested in anomaly detection in time series...

A review on outlier/anomaly detection in time series data

ANE BLÁZQUEZ-GARCÍA and ANGEL CONDE, Ikerlan Technology Research Centre, Basque Research and Technology Alliance (BRTA), Spain

USUE MORI, Intelligent Systems Group (ISG), Department of Computer Science and Artificial Intelligence, University of the Basque Country (UPV/EHU), Spain

JOSE A. LOZANO, Intelligent Systems Group (ISG), Department of Computer Science and Artificial Intelligence, University of the Basque Country (UPV/EHU), Spain and Basque Center for Applied Mathematics (BCAM), Spain

Recent advances in technology have brought major breakthroughs in data collection, enabling a large amount of data to be gathered over time and thus generating time series. Mining this data has become an important task for researchers and practitioners in the past few years, including the detection of outliers or anomalies that may represent errors or events of interest. This review aims to provide a structured and comprehensive state-of-the-art on outlier detection techniques in the context of time series. To this end, a taxonomy is presented based on the main aspects that characterize an outlier detection technique.

Additional Key Words and Phrases: Outlier detection, anomaly detection, time series, data mining, taxonomy, software

1 INTRODUCTION

Recent advances in technology allow us to collect a large amount of data over time in diverse research areas. Observations that have been recorded in an orderly fashion and which are correlated in time constitute a time series. Time series data mining aims to extract all meaningful knowledge from this data, and several mining tasks (e.g., classification, clustering, forecasting, and outlier detection) have been considered in the literature [Elsing and Agon 2012; Fu 2011; Ratanamahatana et al. 2010].

Outlier detection has become a field of interest for many researchers and practitioners and is now one of the main tasks of time series data mining. Outlier detection has been studied in a variety of application domains such as credit card fraud detection, intrusion detection in cybersecurity, or fault diagnosis in industry. In particular, the analysis of outliers in time series data examines anomalous behaviors across time [Gupta et al. 2014]. In the first study on this topic, which was conducted by Fox [1972], two types of outliers in univariate time series were defined: type I, which affects a single observation, and type II, which affects both a particular observation and the subsequent observations. This work was first extended to four outlier types [Tsay 1989], and then to the case of multivariate time series [Tsay 1994]. The first extensions of the definition of outlier and of anomalous observations were proposed in the literature. However, to this day, there is still no consensus on the terms used [Carreño et al. 2019]. For example, outlier observations are often referred to as anomalies, discordant observations, discords, exceptions, aberrations, surprises, peculiarities or contaminants.

Authors' addresses: Ane Blázquez-García, ablazquez@ikerlan.es; Angel Conde, aconde@ikerlan.es, Ikerlan Technology Research Centre, Basque Research and Technology Alliance (BRTA), P^o J.M. Arizmendiarieta, 2, Arrasate/Mondragón, 20500, Spain; Usue Mori, usue.mori@ehu.es, Intelligent Systems Group (ISG), Department of Computer Science and Artificial Intelligence, University of the Basque Country (UPV/EHU), Manuel de Lardizabal, 1, 20018.

A. Blazquez-Garcia et al. ACM
Computing Survey (2021) [24]

Dive into Time-Series Anomaly Detection: A Decade Review

PAUL BONIOL, Inria, DI ENS, PSL, CNRS, France

QINGHUA LIU, The Ohio State University, USA

MINGYI HUANG, The Ohio State University, USA

THEMIS PALPANAS, Université Paris Cité; IUF, France

JOHN PAPARRIZOS, The Ohio State University, USA

Recent advances in data collection technology, accompanied by the ever-rising volume and velocity of streaming data, underscore the vital need for time-series analytics. In this regard, time-series anomaly detection has been an important activity, entailing various applications in fields such as cyber security, financial markets, law enforcement, and health care. While traditional literature of anomaly detection is centered on statistical measures, the increasing number of machine learning algorithms in recent years call for a structured, general characterization of the research methods for time-series anomaly detection. This survey groups and summarizes anomaly detection existing solutions under a process-centric taxonomy in the time series context. In addition to giving an original categorization of anomaly detection methods, we also perform a meta-analysis of the literature and outline general trends in time-series anomaly detection research.

ACM Reference Format:

Paul Boniol, Qinghua Liu, Mingyi Huang, Themis Palpanas, and John Paparrizos. 2024. Dive into Time-Series Anomaly Detection: A Decade Review. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 51 pages. <https://doi.org/XXXXXXX.XXXXXXX>

1 Introduction

A wide range of cost-effective sensing, networking, storage, and processing solutions enable the collection of enormous amounts of measurements over time [109–111, 122, 137, 138, 141, 143, 179, 181, 186]. Recording these measurements results in an ordered sequence of real-valued data points commonly referred to as *time series*. More generic terms, such as *data series* or *data sequences*, have also been used to refer to cases where the ordering of data relies on a dimension other than time (e.g., the angle in data from astronomy, the mass in data from spectrometry, or the position in data from biology) [176]. Analytical tasks over time series data are necessary virtually in every scientific discipline and their corresponding industries [14, 44, 62, 78, 161, 182, 190–192, 201], including in astronomy [4, 102, 245], biology [110–113, 654], economics [36, 74, 148, 155, 213, 221, 240], engineering [1, 2, 359], engineering [112, 162, 203, 243], environmental sciences [77, 84, 100, 101, 164, 207, 247], medicine [32, 199, 208], neuroscience [161, 119], and social sciences [36, 190]. The analysis of time series has become increasingly prevalent for understanding a multitude of natural or human-made processes [187, 188]. Unfortunately, inherent complexities in the data generation of these

Authors' Contact Information: Paul Boniol, Inria, DI ENS, PSI, CNRS, Paris, France, paul.boniol@inria.fr; Qinghua Liu, The Ohio State University.

P. Boniol et al. Arxiv (2025)
[28]

More to Read

If you are interested in anomaly detection in time series...

A review on outlier/anomaly detection in time series data

ANE BLÁZQUEZ-GARCÍA and ANGEL CONDE, Ikerlan Technology Research Centre, Basque Research and Technology Alliance (BRTA), Spain

USUE MORI, Intelligent Systems Group (ISG), Department of Computer Science and Artificial Intelligence, University of the Basque Country (UPV/EHU), Spain

JOSE A. LOZANO, Intelligent Systems Group (ISG), Department of Computer Science and Artificial Intelligence, University of the Basque Country (UPV/EHU), Spain and Basque Center for Applied Mathematics (BCAM), Spain

Recent advances in technology have brought major breakthroughs in data collection, enabling a large amount of data to be gathered over time and thus generating time series. Mining this data has become an important task for researchers and practitioners in the past few years, including the detection of outliers or anomalies that may represent errors or events of interest. This review aims to provide a structured and comprehensive state-of-the-art on outlier detection techniques in the context of time series. To this end, a taxonomy is presented based on the main aspects that characterize an outlier detection technique.

Additional Key Words and Phrases: Outlier detection, anomaly detection, time series, data mining, taxonomy, software

1 INTRODUCTION

Recent advances in technology allow us to collect a large amount of data over time in diverse research areas. Observations that have been recorded in an orderly fashion and which are correlated in time constitute a time series. Time series data mining aims to extract all meaningful knowledge from this data, and several mining tasks (e.g., classification, clustering, forecasting, and outlier detection) have been considered in the literature [Ealing and Agon 2012; Fu 2011; Ratanamahatana et al. 2010].

Outlier detection has become a field of interest for many researchers and practitioners and is now one of the main tasks of time series data mining. Outlier detection has been studied in a variety of application domains such as credit card fraud detection, intrusion detection in cybersecurity, or fault diagnosis in industry. In particular, the analysis of outliers in time series data examines anomalous behaviors across time [Gupta et al. 2014a]. In the first study on this topic, which was conducted by Fox [1972], two types of outliers in univariate time series were defined: type I, which affects a single observation; and type II, which affects both a particular observation and the subsequent observations. This work was first extended to four outlier types [Tsay 1988], and then to the case of multivariate time series [Tsay et al. 2000]. Since then, many definitions of the term outlier and numerous detection methods have been proposed in the literature. However, to this day, there is still no consensus on the terms used [Carreño et al. 2019]; for example, outlier observations are often referred to as anomalies, discordant observations, discords, exceptions, aberrations, surprises, peculiarities or contaminants.

Authors' addresses: Ane Blázquez-García, ablaquez@ikerlan.es; Angel Conde, aconde@ikerlan.es, Ikerlan Technology Research Centre, Basque Research and Technology Alliance (BRTA), P.º J.M. Arizumiendiarrieta, 2, Arraute/Itzandegui, 20080, Spain; Usue Mori, usue.mori@ehu.es, Intelligent Systems Group (ISG), Department of Computer Science and Artificial Intelligence, University of the Basque Country (UPV/EHU), Manuel de Lardizabal, 1, 20018,

A. Blazquez-Garcia et al. ACM Computing Survey (2021) [24]

Dive into Time-Series Anomaly Detection: A Decade Review

PAUL BONIOL, Inria, DI ENS, PSL, CNRS, France

QINGHUA LIU, The Ohio State University, USA

MINGYI HUANG, The Ohio State University, USA

THEMIS PALPANAS, Université Paris Cité; IUF, France

JOHN PAPARRIZOS, The Ohio State University, USA

Recent advances in data collection technology, accompanied by the ever-rising volume and velocity of streaming data, underscore the vital need for time series analytics. In this regard, time-series anomaly detection has been an important activity, entailing various applications in fields such as cyber security, financial markets, law enforcement, and health care. While traditional literature on anomaly detection is centered on statistical measures, the increasing number of machine learning algorithms in recent years call for a structured, general characterization of the research methods for time-series anomaly detection. This survey groups and summarizes anomaly detection existing solutions under a process-centric taxonomy in the time series context. In addition to giving an original categorization of anomaly detection methods, we also perform a meta-analysis of the literature and outline general trends in time-series anomaly detection research.

ACM Reference Format:

Paul Boniol, Qinghua Liu, Mingyi Huang, Themis Palpanas, and John Paparrizos. 2024. Dive into Time-Series Anomaly Detection: A Decade Review. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 51 pages. <https://doi.org/XXXXXXX/XXXXXXX>

1 Introduction

A wide range of cost-effective sensing, networking, storage, and processing solutions enable the collection of enormous amounts of measurements over time [109–111, 122, 137, 138, 141, 143, 179, 181, 186]. Recording these measurements results in an ordered sequence of real-valued data points commonly referred to as *time series*. More generic terms, such as *data series* or *data sequences*, have also been used to refer to cases where the ordering of data relies on a dimension other than time (e.g., the angle in data from astronomy, the mass in data from spectrometry, or the position in data from biology) [176]. Analytical tasks over time series data are necessary virtually in every scientific discipline and their corresponding industries [14, 61, 62, 78, 161, 182, 190–192, 201], including in astronomy [4, 102, 243], biology [11–13, 64], economics [36, 74, 148, 155, 213, 221, 240], energy sciences [6, 9, 158], engineering [112, 162, 203, 243, 248], environmental sciences [77, 84, 100, 101, 164, 207, 247], medicine [57, 199, 206], neuroscience [21, 119], and social sciences [36, 160]. The analysis of time series has become increasingly prevalent for understanding a multitude of natural or human-made processes [187, 188]. Unfortunately, inherent complexities in the data generation of these

Authors' Contact Information: Paul Boniol, Inria, DI ENS, PSL, CNRS, Paris, France, paulboniol@inria.fr; Qinghua Liu, The Ohio State University,

P. Boniol et al. Arxiv (2025) [28]

Current Time Series Anomaly Detection Benchmarks are Flawed and are Creating the Illusion of Progress

Renjie Wu and Eamonn J. Keogh

Abstract—Time series anomaly detection has been a perennially important topic in data science, with papers dating back to the 1950s. However, in recent years there has been an explosion of interest in this topic, much of it driven by the success of deep learning in other domains and for other time series tasks. Most of these papers test on one or more of a handful of popular benchmark datasets, created by Yahoo, Numenta, NASA, etc. In this work we make a surprising claim. The majority of the individual exemplars in these datasets suffer from one or more of four flaws. Because of these four flaws, we believe that many published comparisons of anomaly detection algorithms may be unreliable, and more importantly, much of the apparent progress in recent years may be illusory. In addition to demonstrating these claims, with this paper we introduce the UCR Time Series Anomaly Archive. We believe that this resource will perform a similar role as the UCR Time Series Classification Archive, by providing the community with a benchmark that allows meaningful comparisons between approaches and a meaningful gauge of overall progress.

Index Terms—Anomaly detection, benchmark datasets, deep learning, time series analysis

1 INTRODUCTION

TIME series anomaly detection has been a perennially important topic in data science, with papers dating back to the dawn of computer science [1]. However, in the last five years there has been an explosion of interest in this topic, with at least one or two papers on the topic appearing each year in virtually every database, data mining, and machine learning conference, including SIGKDD [2], [3], ICDM [4], ICDE, SIGMOD, VLDB, etc. A large fraction of this increase in interest seems to be largely driven by researchers anxious to transfer the considerable success of deep learning in other domains and from other time series tasks such as classification.

Most of these papers test on one or more of a handful of popular benchmark datasets, created by Yahoo [5], Numenta [6], NASA [2] or Poi's Lab (OMNI) [3], etc. In this work we make a surprising claim. The majority of the individual exemplars in these datasets suffer from one or more of four flaws. These flaws are *triviality*, *unrealistic anomaly density*, *misaligned ground truth* and *run-to-failure bias*. Because of these four flaws, we believe that most published comparisons of anomaly detection algorithms may be unreliable. More importantly, we believe that much of the apparent progress in recent years may be

neural networks, and a variational auto-encoder (VAE) over-sampling model." This description sounds like it has many "moving parts", and indeed, the dozen or so explicitly listed parameters include: convolution filter, activation, kernel size, strides, padding, LSTM input size, dense input size, softmax loss function, window size, learning rate and batch size. All of this is to demonstrate "accuracy exceeding 0.90 (on a subset of the Yahoo's anomaly detection benchmark datasets)." However, as we will show, much of the results of this complex approach can be duplicated with a single line of code and a few minutes of effort.

This "one-line-of-code" argument is so unusual that it is worth previewing it before we formally demonstrate it in Section 2.2 below. Almost daily, the popular press vaunts a new achievement of deep learning. Picking one at random, in a recent paper [8], we learn that deep learning can be used to classify mosquitoes' species. In particular, the proposed algorithm had an accuracy of 97.8% when distinguishing *Aedes vexans* from *Culex tritaeniorhynchus*. Should we be impressed? One of the current authors (Keogh) has significant computational experience working with mosquitoes, and he is impressed.

Suppose however that someone downloaded the origi-

R. Wu et al. TKDE (2021) [18]

Authors' addresses: Renjie Wu, renjie.wu@osu.edu; Eamonn J. Keogh, eamonn@cs.cmu.edu. We have no reason to doubt the claims of this paper, which we only skimmed.

More to Read

If you are interested in anomaly detection in time series...

A review on outlier/anomaly detection in time series data

ANE BLÁZQUEZ-GARCÍA and ANGEL CONDE, Ikerlan Technology Research Centre, Basque Research and Technology Alliance (BRTA), Spain

USUE MORI, Intelligent Systems Group (ISG), Department of Computer Science and Artificial Intelligence, University of the Basque Country (UPV/EHU), Spain

JOSE A. LOZANO, Intelligent Systems Group (ISG), Department of Computer Science and Artificial Intelligence, University of the Basque Country (UPV/EHU), Spain and Basque Center for Applied Mathematics (BCAM), Spain

Recent advances in technology have brought major breakthroughs in data collection, enabling a large amount of data to be gathered over time and thus generating time series. Mining this data has become an important task for researchers and practitioners in the past few years, including the detection of outliers or anomalies that may represent errors or events of interest. This review aims to provide a structured and comprehensive state-of-the-art on outlier detection techniques in the context of time series. To this end, a taxonomy is presented based on the main aspects that characterize an outlier detection technique.

Additional Key Words and Phrases: Outlier detection, anomaly detection, time series, data mining, taxonomy, software

1 INTRODUCTION

Recent advances in technology allow us to collect a large amount of data over time in diverse research areas. Observations that have been recorded in an orderly fashion and which are correlated in time constitute a time series. Time series data mining aims to extract all meaningful knowledge from this data, and several mining tasks (e.g., classification, clustering, forecasting, and outlier detection) have been considered in the literature [Ealing and Agon 2012; Fu 2011; Ratanamahatana et al. 2010].

Outlier detection has become a field of interest for many researchers and practitioners and is now one of the main tasks of time series data mining. Outlier detection has been studied in a variety of application domains such as credit card fraud detection, intrusion detection in cybersecurity, or fault diagnosis in industry. In particular, the analysis of outliers in time series data examines anomalous behaviors across time [Gupta et al. 2014a]. In the first study on this topic, which was conducted by Fox [1972], two types of outliers in univariate time series were defined: type I, which affects a single observation; and type II, which affects both a particular observation and the subsequent observations. This work was first extended to four outlier types [Tsay 1988], and then to the case of multivariate time series [Tsay et al. 2000]. Since then, many definitions of the term outlier and numerous detection methods have been proposed in the literature. However, to this day, there is still no consensus on the terms used [Carreño et al. 2019]; for example, outlier observations are often referred to as anomalies, discordant observations, discords, exceptions, aberrations, surprises, peculiarities or contaminants.

Authors' addresses: Ane Blázquez-García, ablaquez@ikerlan.es; Angel Conde, aconde@ikerlan.es, Ikerlan Technology Research Centre, Basque Research and Technology Alliance (BRTA), P. J.M. Arizumendiarrizeta, 2, Arrazate/Elmendiagin, 48930, Spain; Usue Mori, usue.mori@ehu.es, Intelligent Systems Group (ISG), Department of Computer Science and Artificial Intelligence, University of the Basque Country (UPV/EHU), Manuel de Lardizabal, 1, 48940, Spain.

A. Blazquez-Garcia et al. ACM Computing Survey (2021) [24]

Dive into Time-Series Anomaly Detection: A Decade Review

PAUL BONIOL, Inria, DI ENS, PSL, CNRS, France

QINGHUA LIU, The Ohio State University, USA

MINGYI HUANG, The Ohio State University, USA

THEMIS PALPANAS, Université Paris Cité; IUF, France

JOHN PAPARRIZOS, The Ohio State University, USA

Recent advances in data collection technology, accompanied by the ever-rising volume and velocity of streaming data, underscore the vital need for time series analytics. In this regard, time-series anomaly detection has been an important activity, entailing various applications in fields such as cyber security, financial markets, law enforcement, and health care. While traditional literature on anomaly detection is centered on statistical measures, the increasing number of machine learning algorithms in recent years call for a structured, general characterization of the research methods for time-series anomaly detection. This survey groups and summarizes anomaly detection existing solutions under a process-centric taxonomy in the time series context. In addition to giving an original categorization of anomaly detection methods, we also perform a meta-analysis of the literature and outline general trends in time-series anomaly detection research.

ACM Reference Format:

Paul Boniol, Qinghua Liu, Mingyi Huang, Themis Palpanas, and John Paparrizos. 2024. Dive into Time-Series Anomaly Detection: A Decade Review. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 51 pages. <https://doi.org/XXXXXXX/XXXXXXX>

1 Introduction

A wide range of cost-effective sensing, networking, storage, and processing solutions enable the collection of enormous amounts of measurements over time [109–111, 122, 137, 138, 141, 143, 179, 181, 186]. Recording these measurements results in an ordered sequence of real-valued data points commonly referred to as *time series*. More generic terms, such as *data series* or *data sequences*, have also been used to refer to cases where the ordering of data relies on a dimension other than time (e.g., the angle in data from astronomy, the mass in data from spectrometry, or the position in data from biology) [176]. Analytical tasks over time series data are necessary virtually in every scientific discipline and their corresponding industries [14, 61, 62, 78, 161, 182, 190–192, 201], including in astronomy [4, 102, 243], biology [11–13, 64], economics [36, 74, 148, 155, 213, 221, 240], energy sciences [6, 9, 158], engineering [112, 162, 203, 243, 248], environmental sciences [77, 84, 100, 101, 164, 207, 247], medicine [57, 199, 206], neuroscience [21, 119], and social sciences [34, 160]. The analysis of time series has become increasingly prevalent for understanding a multitude of natural or human-made processes [187, 188]. Unfortunately, inherent complexities in the data generation of these

Authors' Contact Information: Paul Boniol, Inria, DI ENS, PSL, CNRS, Paris, France; paulboniol@inria.fr; Qinghua Liu, The Ohio State University,

P. Boniol et al. Arxiv (2025) [28]

Current Time Series Anomaly Detection Benchmarks are Flawed and are Creating the Illusion of Progress

Renjie Wu and Eamonn J. Keogh

Abstract—Time series anomaly detection has been a perennially important topic in data science, with papers dating back to the 1950s. However, in recent years there has been an explosion of interest in this topic, much of it driven by the success of deep learning in other domains and for other time series tasks. Most of these papers test on one or more of a handful of popular benchmark datasets, created by Yahoo, Numenta, NASA, etc. In this work we make a surprising claim. The majority of the individual exemplars in these datasets suffer from one or more of four flaws. Because of these four flaws, we believe that many published comparisons of anomaly detection algorithms may be unreliable, and more importantly, much of the apparent progress in recent years may be illusory. In addition to demonstrating these claims, with this paper we introduce the UCR Time Series Anomaly Archive. We believe that this resource will perform a similar role as the UCR Time Series Classification Archive, by providing the community with a benchmark that allows meaningful comparisons between approaches and a meaningful gauge of overall progress.

Index Terms—Anomaly detection, benchmark datasets, deep learning, time series analysis

1 INTRODUCTION

TIME series anomaly detection has been a perennially important topic in data science, with papers dating back to the dawn of computer science [1]. However, in the last five years there has been an explosion of interest in this topic, with at least one or two papers on the topic appearing each year in virtually every database, data mining, and machine learning conference, including SIGKDD [2], [3], ICDM [4], ICDE, SIGMOD, VLDB, etc.

A large fraction of this increase in interest seems to be largely driven by researchers anxious to transfer the considerable success of deep learning in other domains and from other time series tasks such as classification.

Most of these papers test on one or more of a handful of popular benchmark datasets, created by Yahoo [5], Numenta [6], NASA [2] or Pci's Lab (OMNI) [3], etc. In this work we make a surprising claim. The majority of the individual exemplars in these datasets suffer from one or more of four flaws. These flaws are *triviality*, *unrealistic anomaly density*, *misaligned ground truth* and *run-to-failure bias*. Because of these four flaws, we believe that most published comparisons of anomaly detection algorithms may be unreliable. More importantly, we believe that much of the apparent progress in recent years may be

neural networks, and a variational auto-encoder (VAE) over-sampling model. This description sounds like it has many “moving parts”, and indeed, the dozen or so explicitly listed parameters include: convolution filter, activation, kernel size, strides, padding, LSTM input size, dense input size, softmax loss function, window size, learning rate and batch size. All of this is to demonstrate “accuracy exceeding 0.90 (on a subset of the Yahoo’s anomaly detection benchmark datasets).” However, as we will show, much of the results of this complex approach can be duplicated with a single line of code and a few minutes of effort.

This “one-line-of-code” argument is so unusual that it is worth previewing it before we formally demonstrate it in Section 2.2 below. Almost daily, the popular press vaunts a new achievement of deep learning. Picking one at random, in a recent paper [8], we learn that deep learning can be used to classify mosquitos’ species. In particular, the proposed algorithm had an accuracy of 97.8% when distinguishing *Aedes vexans* from *Culex tritaeniorhynchus*. Should we be impressed? One of the current authors (Keogh) has significant computational experience working with mosquitos, and he is impressed.

Suppose however that someone downloaded the origi-

R. Wu et al. TKDE (2021) [18]

Authors' Contact Information:

“To ensure the highest quality of our work, we have no reason to doubt the claims of this paper, which we only skimmed.”

The Elephant in the Room: Towards A Reliable Time-Series Anomaly Detection Benchmark

Qinghua Liu and John Paparrizos
Department of Computer Science and Engineering
The Ohio State University
{liu.11085, paparrizos.1}@osu.edu

Abstract

Time-series anomaly detection is a fundamental task across scientific fields and industries. However, the field has long faced the “elephant in the room”: critical issues including flawed datasets, biased evaluation measures, and inconsistent benchmarking practices that have remained largely ignored and unaddressed. We introduce the TSB-AD to systematically tackle these issues in the following three aspects: (i) **Dataset Integrity**: with 1070 high-quality time series from a diverse collection of 40 datasets (doubling the size of the largest collection and four times the number of existing curated datasets), we provide the first large-scale, heterogeneous, meticulously curated dataset that combines the effort of human perception and model interpretation; (ii) **Measure Reliability**: by revealing issues and biases in evaluation measures, we identify the most reliable and accurate measure, namely, VUS-PR for anomaly detection in time series to address concerns from the community; and (iii) **Comprehensive Benchmarking**: with a broad spectrum of 40 detection algorithms, from statistical methods to the latest foundation models, we perform a comprehensive evaluation that includes a thorough hyperparameter tuning and a unified setup for a fair and reproducible comparison. Our findings challenge the conventional wisdom regarding the superiority of advanced neural network architectures, revealing that simpler architectures and statistical methods often yield better performance. The promising performance of neural networks on multivariate cases and foundation models on point anomalies highlights the need for further advancements in these methods. We open-source the benchmark

<https://github.com/TheDataMorg/TSB-AD>

time-series anomaly detection is widely applied across various sectors [17, 98, 21, 19, 18, 57, 15], ranging from manufacturing quality assurance and data center monitoring to preventing financial

Q. Liu et al. NeurIPS (2024) [27]

38th Conference on Neural Information Processing Systems (NeurIPS 2024) Track on Datasets and Benchmarks.

Reference

- [1] N. Laptev, S. Amizadeh, and Y. Billawala. 2015. S5 - A Labeled Anomaly Detection Dataset, version 1.0(16M).
- [2] Markus Thill, Wolfgang Konen, and Thomas Bäck. 2020. MGAB: The Mackey-Glass Anomaly Benchmark.
- [3] Pawel Benecki, Szymon Piechaczek, Daniel Kostrzewa, and Jakub Nalepa. 2021. Detecting Anomalies in Spacecraft Telemetry Using Evolutionary Thresholding and LSTMs. In Proceedings of the Genetic and Evolutionary Computation Conference Companion (Lille, France) (GECCO '21)
- [4] Scott David Greenwald. 1990. Improved detection and classification of arrhythmias in noise-corrupted electrocardiograms using contextual information. Thesis. Massachusetts Institute of Technology.
- [5] Sebastian Schmidl, Phillip Wenig, and Thorsten Papenbrock. 2022. Anomaly detection in time series: a comprehensive evaluation. Proc. VLDB Endow. 15, 9 (May 2022), 1779–1797.
- [6] Chin-Chia Michael Yeh, Yan Zhu, Liudmila Ulanova, Nurjahan Begum, Yifei Ding, Hoang Anh Dau, Diego Furtado Silva, Abdullah Mueen, and Eamonn J. Keogh. 2016. Matrix Profile I: All Pairs Similarity Joins for Time Series. In ICDM.
- [7] Yan Zhu, Zachary Zimmerman, Nader Shakibay Senobari, Chin-Chia Michael Yeh, Gareth Funning, Abdullah Mueen, Philip Brisk, and Eamonn Keogh. 2016. Matrix Profile II: Exploiting a Novel Algorithm and GPUs to Break the One Hundred Million Barrier for Time Series Motifs and Joins. In Proceedings of the International Conference on Data Mining (ICDM), 739–748.
- [8] Yue Lu, Renjie Wu, Abdullah Mueen, Maria A. Zuluaga, and Eamonn Keogh. 2022. Matrix Profile XXIV: Scaling Time Series Anomaly Detection to Trillions of Datapoints and Ultra-fast Arriving Data Streams. In Proceedings of the 28th ACM SIGIJCAI.
- [9] C. -C. M. Yeh, N. Kavantzaz and E. Keogh, Matrix Profile VI: Meaningful Multidimensional Motif Discovery, 2017 IEEE International Conference on Data Mining (ICDM), New Orleans, LA, USA, 2017, pp. 565-574, doi: 10.1109/ICDM.2017.66. Data Mining (IJCAI '22).
- [10] Paul Boniol, Michele Linardi, Federico Roncallo, Themis Palpanas, Mohammed Meftah, and Emmanuel Remy. 2021. Unsupervised and scalable subsequence anomaly detection in large data series. The VLDB Journal 30, 6 (Nov 2021), 909–931.
- [11] F. T. Liu, K. M. Ting and Z. -H. Zhou, Isolation Forest, 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy, 2008, pp. 413-422
- [12] Markus Goldstein and Andreas Dengel. 2012. Histogram-based outlier score (hbos): A fast unsupervised anomaly detection algorithm. KI-2012: poster and demo track 9 (2012).
- [13] Paul Boniol and Themis Palpanas. 2020. Series2Graph: graph-based subsequence anomaly detection for time series. Proc. VLDB Endow. 13, 12 (August 2020), 1821–1834.
- [14] Ali Abdul-Aziz, Mark R Woike, Nikunj C Oza, Bryan L Matthews, and John D Iekki. 2012. Rotor health monitoring combining spin tests and data-driven anomaly detection methods. Structural Health Monitoring (2012).
- [15] Pankaj Malhotra, Lovekesh Vig, Gautam Shro, and Puneet Agarwal. 2015. Long Short Term Memory Networks for Anomaly Detection in Time Series. (2015).
- [16] M. Munir, S. A. Siddiqui, A. Dengel, and S. Ahmed. 2019. DeepAnT: A Deep Learning Approach for Unsupervised Anomaly Detection in Time Series. IEEE Access 7 (2019), 1991–2005.
- [17] Mayu Sakurada and Takehisa Yairi. 2014. Anomaly Detection Using Autoencoders with Nonlinear Dimensionality Reduction. In Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis (Gold Coast, Australia QLD, Australia) (MLSDA'14).
- [18] R. Wu and E. Keogh, Current Time Series Anomaly Detection Benchmarks are Flawed and are Creating the Illusion of Progress in IEEE Transactions on Knowledge & Data Engineering, vol. 35, no. 03, pp. 2421-2429, 2023.
- [19] John Paparrizos, Yuhao Kang, Paul Boniol, Ruey S. Tsay, Themis Palpanas, and Michael J. Franklin. 2022. TSB-UAD: an end-to-end benchmark suite for univariate time-series anomaly detection. Proc. VLDB Endow. 15, 8 (April 2022), 1697–1711.
- [20] Tom Fawcett. 2006. An introduction to ROC analysis. Pattern Recognition Letters 27, 8 (2006), 861–874.

Reference

- [21] Jesse Davis and Mark Goadrich. 2006. The Relationship between Precision-Recall and ROC Curves. In Proceedings of the 23rd International Conference on Machine Learning (ICML '06).
- [22] John Paparrizos, Paul Boniol, Themis Palpanas, Ruey S. Tsay, Aaron Elmore, and Michael J. Franklin. 2022. Volume under the surface: a new accuracy evaluation measure for time-series anomaly detection. *Proc. VLDB Endow.* 15, 11 (July 2022), 2774–2787.
- [23] Nesime Tatbul, Tae Jun Lee, Stan Zdonik, Mejbah Alam, and Justin Gottschlich. 2018. Precision and Recall for Time Series. In *Advances in Neural Information Processing Systems*, Vol. 31.
- [24] Ane Blázquez-García, Angel Conde, Usue Mori, and Jose A. Lozano. 2021. A Review on Outlier/Anomaly Detection in Time Series Data. *ACM Comput. Surv.* 54, 3, Article 56 (April 2022), 33 pages.
- [25] Paul Boniol, John Paparrizos, Themis Palpanas, and Michael J. Franklin. 2021. SAND: streaming subsequence anomaly detection. *Proc. VLDB Endow.* 14, 10 (June 2021), 1717–1729.
- [26] Schneider, J., Wenig, P. & Papenbrock, T. Distributed detection of sequential anomalies in univariate time series. *The VLDB Journal* 30, 579–602 (2021).
- [27] Liu, Q. and Paparrizos, J., 2024. The elephant in the room: Towards a reliable time-series anomaly detection benchmark. *Advances in Neural Information Processing Systems*, 37, pp.108231-108261.
- [28] Boniol, P., Liu, Q., Huang, M., Palpanas, T. and Paparrizos, J., 2024. Dive into time-series anomaly detection: A decade review. *arXiv preprint arXiv:2412.20512*.
- [29] Maroua Bahri, Flavia Salutari, Andrian Putina, and Mauro Sozio: AutoML: state of the art with a focus on anomaly detection, challenges, and research directions. *International Journal of Data Science and Analytics* 14(2): 113-126 (2022).
- [30] Mononito Goswami, Cristian Challu, Laurent Callot, Lenon Minorics, Andrey Kan. 2023. Unsupervised Model Selection for Time-series Anomaly Detection. In *Proceedings of the International Conference on Learning Representations*.
- [31] Emmanouil Sylligardos, Paul Boniol, John Paparrizos, Panos Trahanias, Themis Palpanas. 2023. Choose wisely: An extensive evaluation of model selection for anomaly detection in time series. *Proceedings of the VLDB Endowment* 16(11): 3418-3432.
- [32] Lin Xu, Frank Hutter, Holger H Hoos, Kevin Leyton-Brown. 2008. SATzilla: portfolio-based algorithm selection for SAT. *Journal of Artificial Intelligence Research* 32: 565-606.
- [33] Lei Cao, Yizhou Yan, Yu Wang, Samuel Madden, Elke A Rundensteiner. 2023. Autood: Automatic outlier detection. *Proceedings of the ACM on Management of Data*, 1(1): 1-27. ACM, New York, NY, USA.
- [34] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, et al. 2021. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.
- [35] Ming Jin, Qingsong Wen, Yuxuan Liang, Chaoli Zhang, Siqiao Xue, Xue Wang, James Zhang, Yi Wang, Haifeng Chen, Xiaoli Li, et al. 2023. Large models for time series and spatio-temporal data: A survey and outlook. *arXiv preprint arXiv:2310.10196*.
- [36] Ming Jin, Yifan Zhang, Wei Chen, Kexin Zhang, Yuxuan Liang, Bin Yang, Jindong Wang, Shirui Pan, Qingsong Wen. 2024. Position: What Can Large Language Models Tell Us about Time Series Analysis. In *Proceedings of the Forty-first International Conference on Machine Learning*.
- [37] Tian Zhou, Peisong Niu, Liang Sun, Rong Jin, et al. 2023. One fits all: Power general time series analysis by pretrained lm. *Advances in Neural Information Processing Systems* 36: 43322-43355.
- [38] Siwon Kim, Kukjin Choi, Hyun-Soo Choi, Byunghan Lee, Sungroh Yoon. 2022. Towards a rigorous evaluation of time-series anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(7): 7194-7201.
- [39] Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, Artur Dubrawski. 2024. MOMENT: A Family of Open Time-series Foundation Models. In *Proceedings of the International Conference on Machine Learning*.
- [40] Daochen Zha, Kwei-Herng Lai, Mingyang Wan, Xia Hu. 2020. Meta-AAD: Active anomaly detection with deep reinforcement learning. In *Proceedings of the 2020 IEEE ICDM*, 771-780. IEEE.
- [41] Liu, Q., Lee, S. and Paparrizos, J., 2025. TSB-AutoAD: Towards Automated Solutions for Time-Series Anomaly Detection. *Proceedings of the VLDB Endowment (VLDB 2025)*.
- [42] Kong, Y., Yang, Y., Hwang, Y., Du, W., Zohren, S., Wang, Z., ... & Wen, Q. (2025). Time-mqa: Time series multi-task question answering with context enhancement. *arXiv preprint arXiv:2503.01875*.

Thank you for attending!

Any Questions?