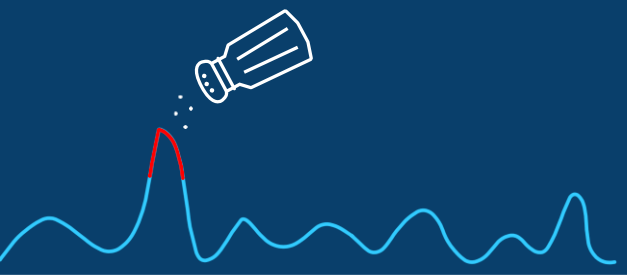


SALT: Streaming Anomaly Labeling for Time series

An approach inspired by SAND (2021)

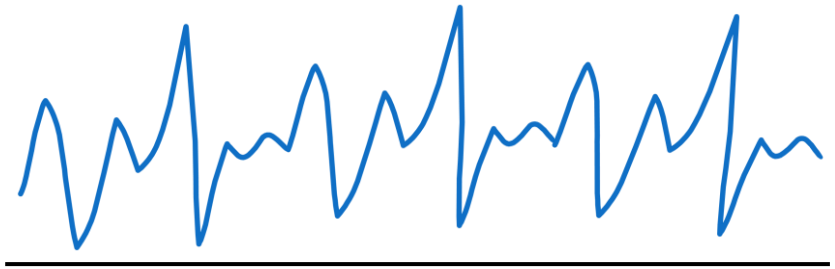
Vasileios Papastergios and Christos Balaktsis
{papster, balaktsis}@csd.auth.gr

ARISTOTLE UNIVERSITY OF THESSALONIKI

DATA&WEB SCIENCE LABORATORY

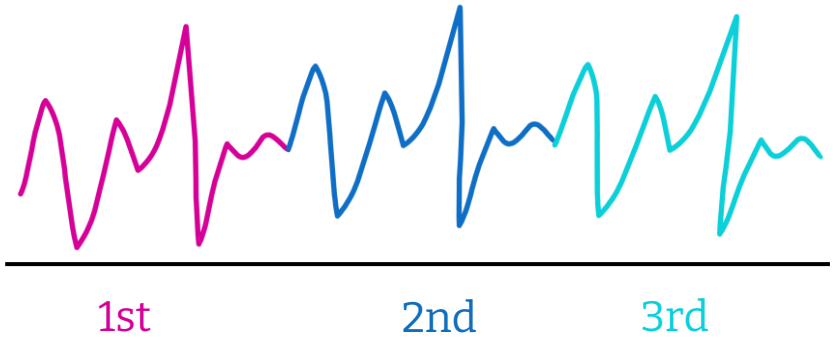
ONLINE ANOMALIES ARE CHALLENGING

Consider a time series stream.



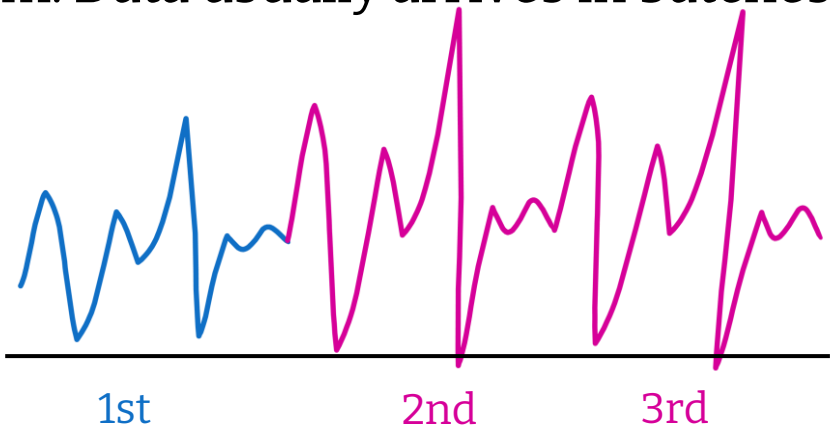
ONLINE ANOMALIES ARE CHALLENGING

Consider a time series stream. Data usually arrives in batches of a specific length.



ONLINE ANOMALIES ARE CHALLENGING

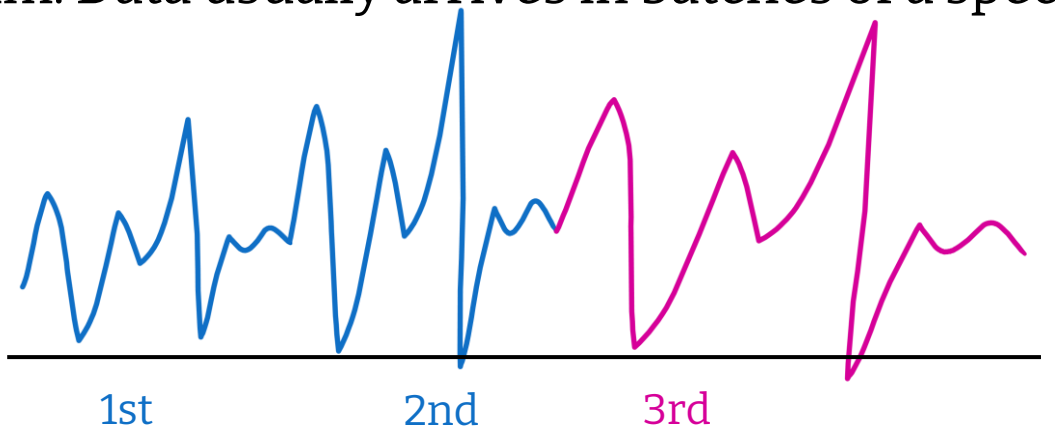
Consider a time series stream. Data usually arrives in batches of a specific length.



Concept drift (normality change)

ONLINE ANOMALIES ARE CHALLENGING

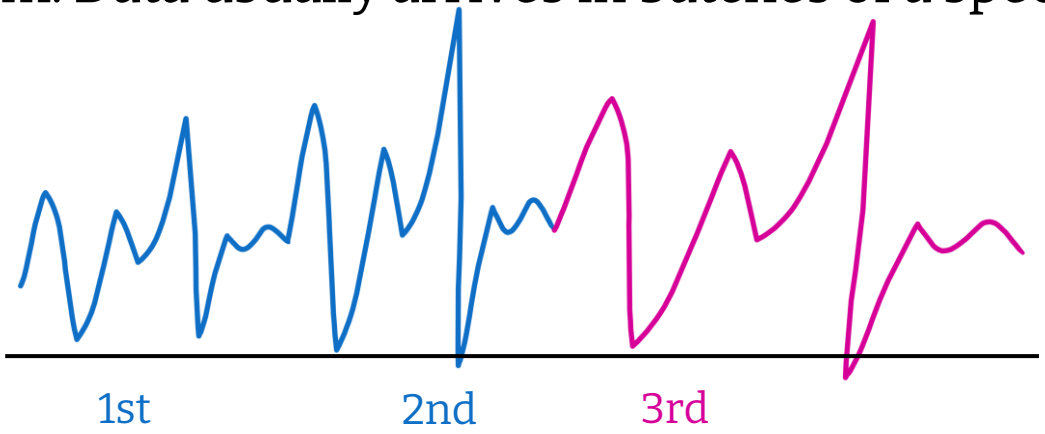
Consider a time series stream. Data usually arrives in batches of a specific length.



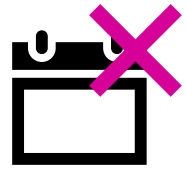
Concept drift (normality change)
Real-time constraints (e.g., shifting)

ONLINE ANOMALIES ARE CHALLENGING

Consider a time series stream. Data usually arrives in batches of a specific length.



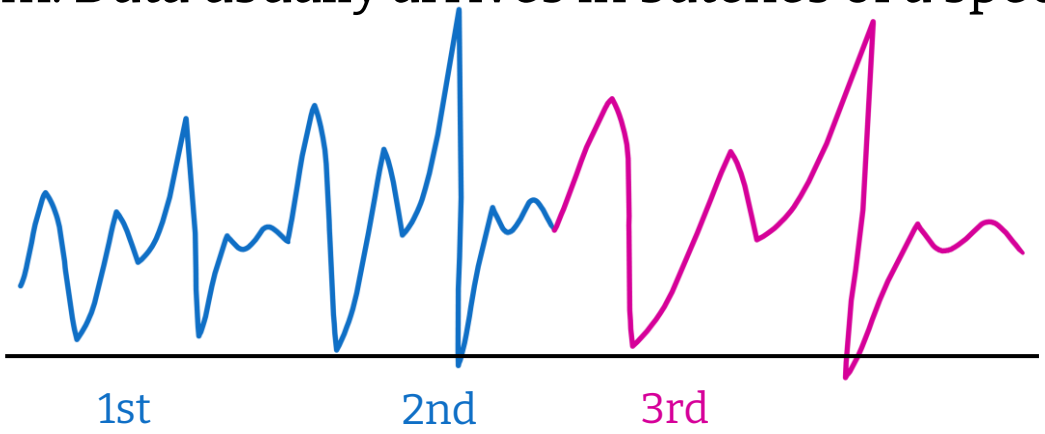
Concept drift (normality change)
Real-time constraints (e.g., shifting)



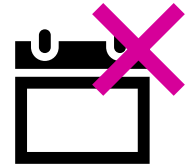
previously learned models become ***outdated***

ONLINE ANOMALIES ARE CHALLENGING

Consider a time series stream. Data usually arrives in batches of a specific length.



Concept drift (normality change)
Real-time constraints (e.g., shifting)

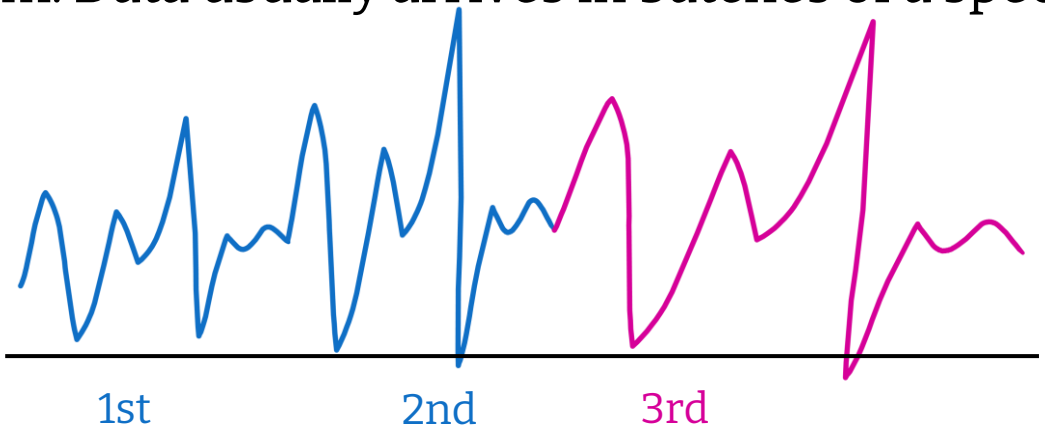


previously learned models become ***outdated***

Keep **more** historical information of the data!

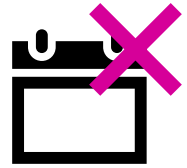
ONLINE ANOMALIES ARE CHALLENGING

Consider a time series stream. Data usually arrives in batches of a specific length.



Concept drift (normality change)
Real-time constraints (e.g., shifting)

→



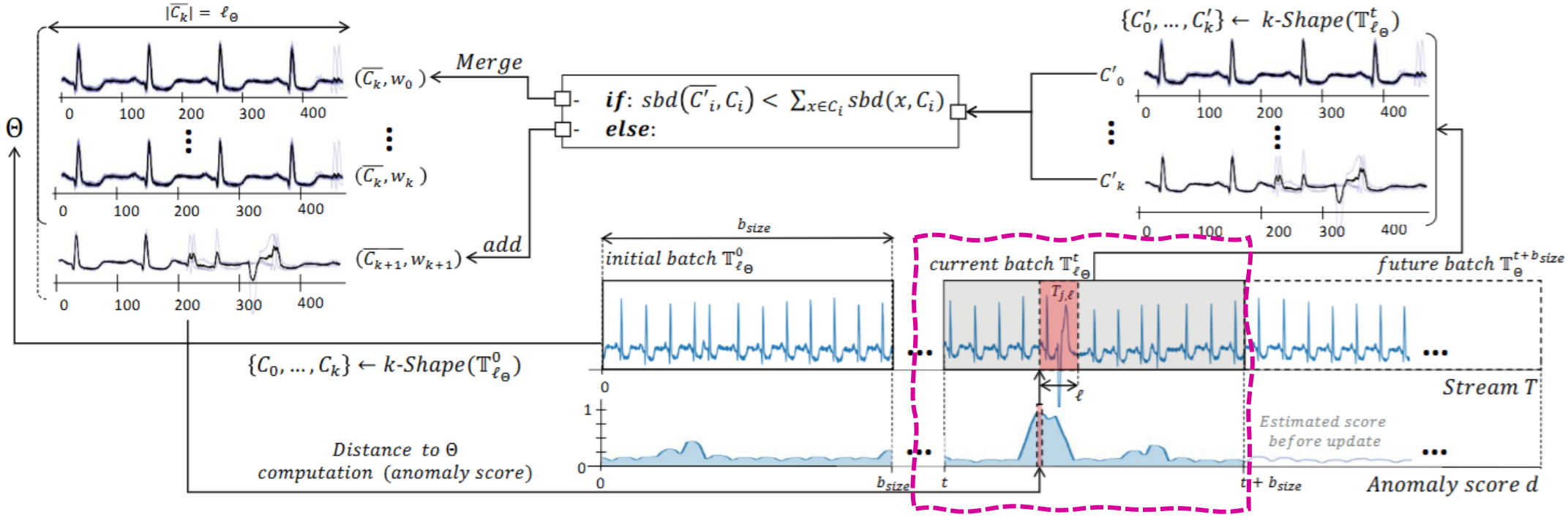
previously learned models become ***outdated***

Keep **more** historical information of the data!

→ *good* **representatives**

SAND: An inspirational Algorithm

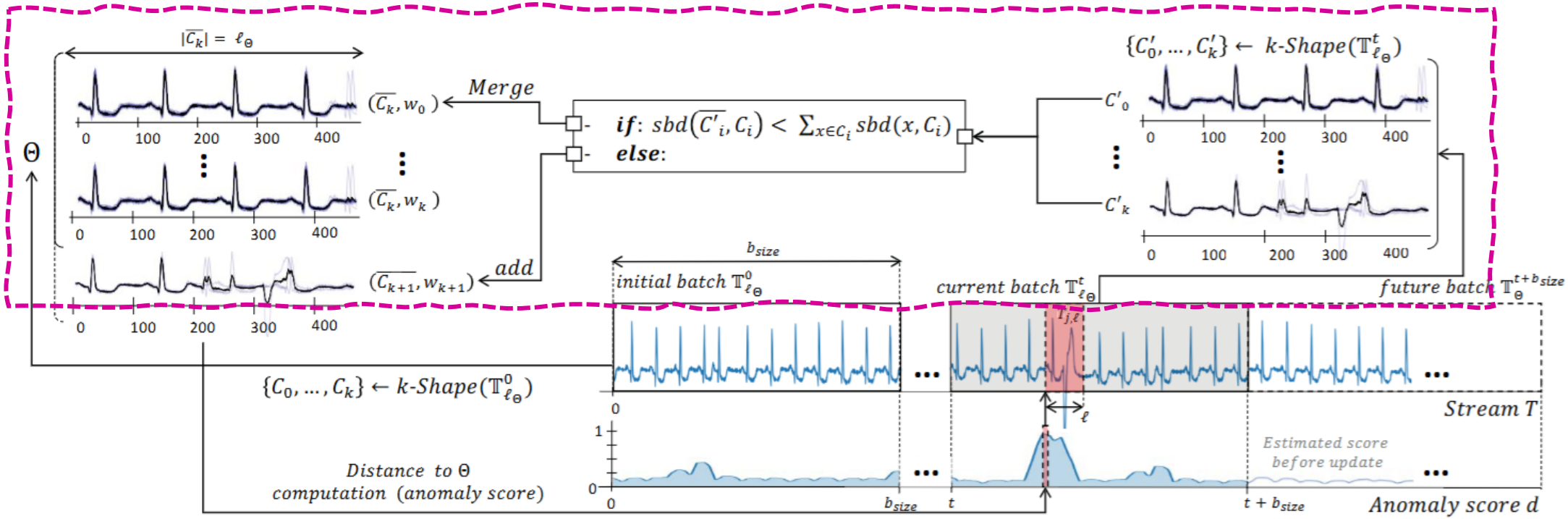
SAND (Boniol, 2021) is a clustering-based streaming approach for anomaly detection working in time series batches.



Paul Boniol, John Paparrizos, Themis Palpanas, and *Michael J. Franklin.* 2021. SAND: streaming subsequence anomaly detection. Proc. VLDB Endow. 14, 10 (June 2021).

SAND: An inspirational Algorithm

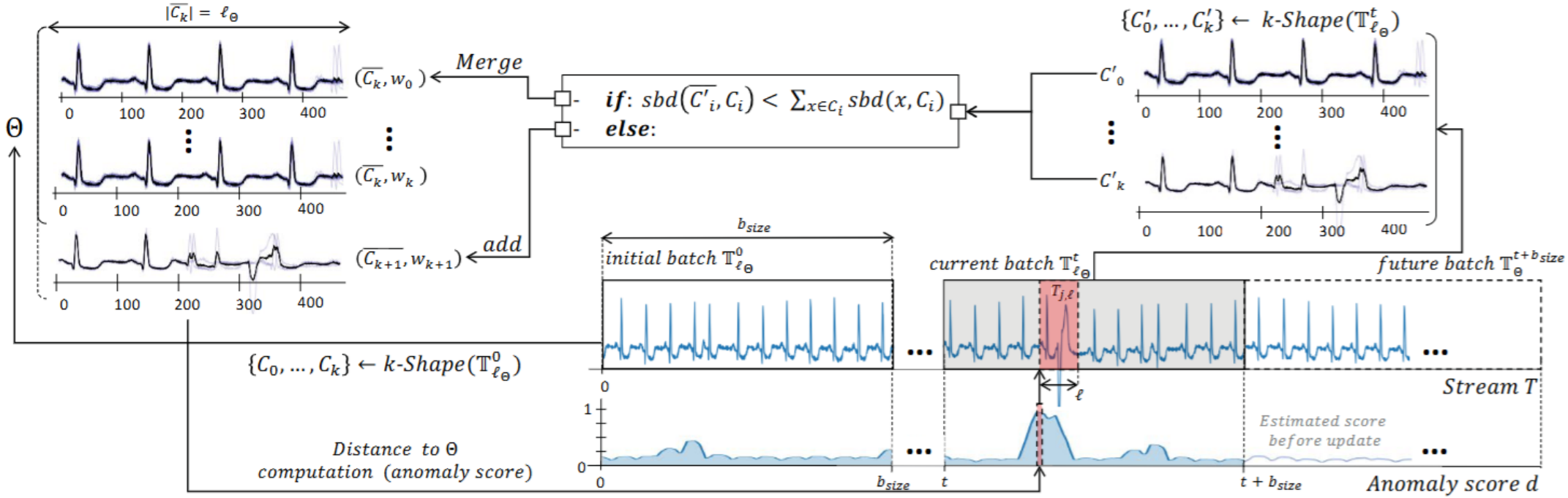
SAND (Boniol, 2021) is a clustering-based streaming approach for anomaly detection working in time series batches. **Each new batch is being clustered and combined with previous clusters.**



Paul Boniol, John Paparrizos, Themis Palpanas, and *Michael J. Franklin.* 2021. SAND: streaming subsequence anomaly detection. Proc. VLDB Endow. 14, 10 (June 2021).

SAND: An inspirational Algorithm

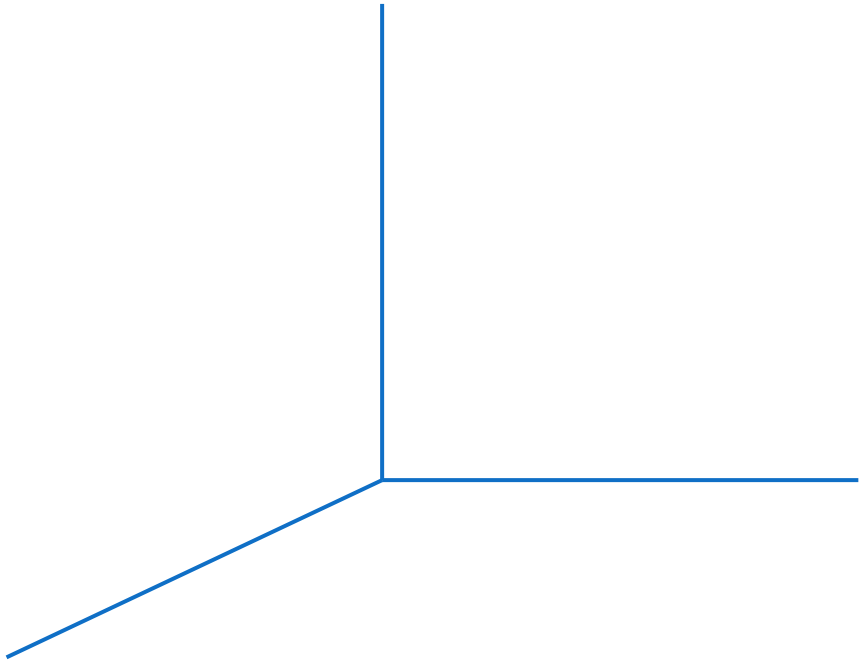
SAND (Boniol, 2021) is a clustering-based streaming approach for anomaly detection working in time series batches. **Each new batch is being clustered and combined with previous clusters.** Very distant clusters/points are considered **anomalous**.



Paul Boniol, John Paparrizos, Themis Palpanas, and *Michael J. Franklin.* 2021. SAND: streaming subsequence anomaly detection. Proc. VLDB Endow. 14, 10 (June 2021).

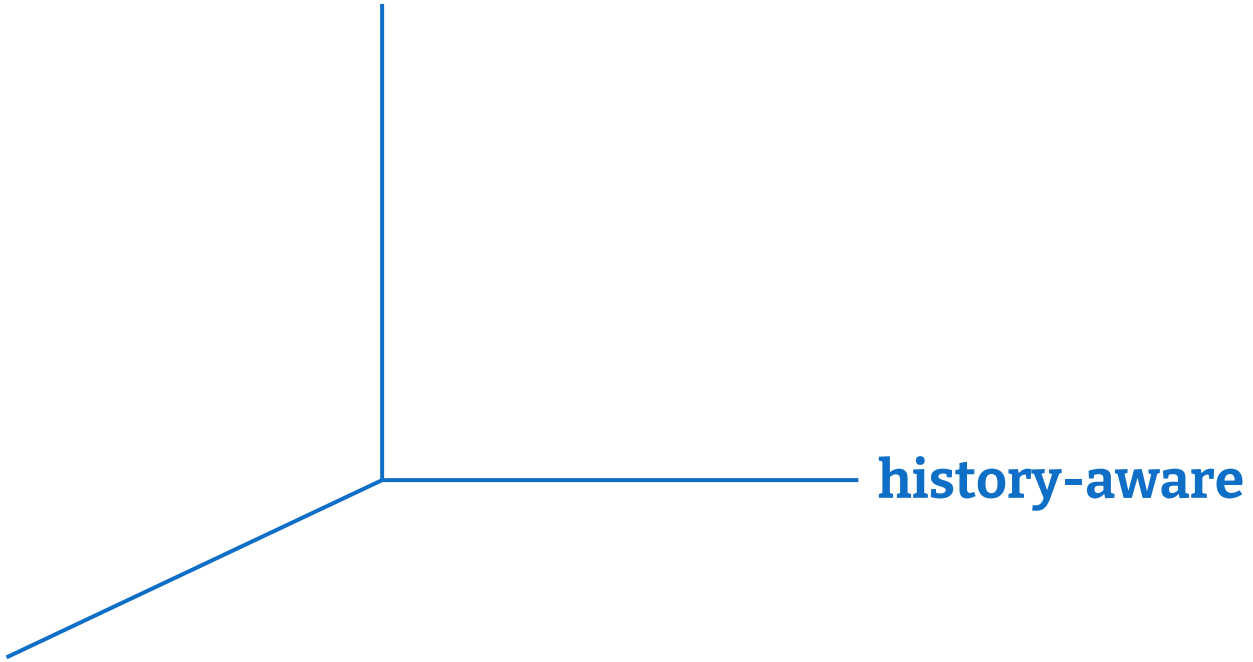
Three *Take-Away* Points from SAND

A new streaming series anomaly detection variant needs to be:



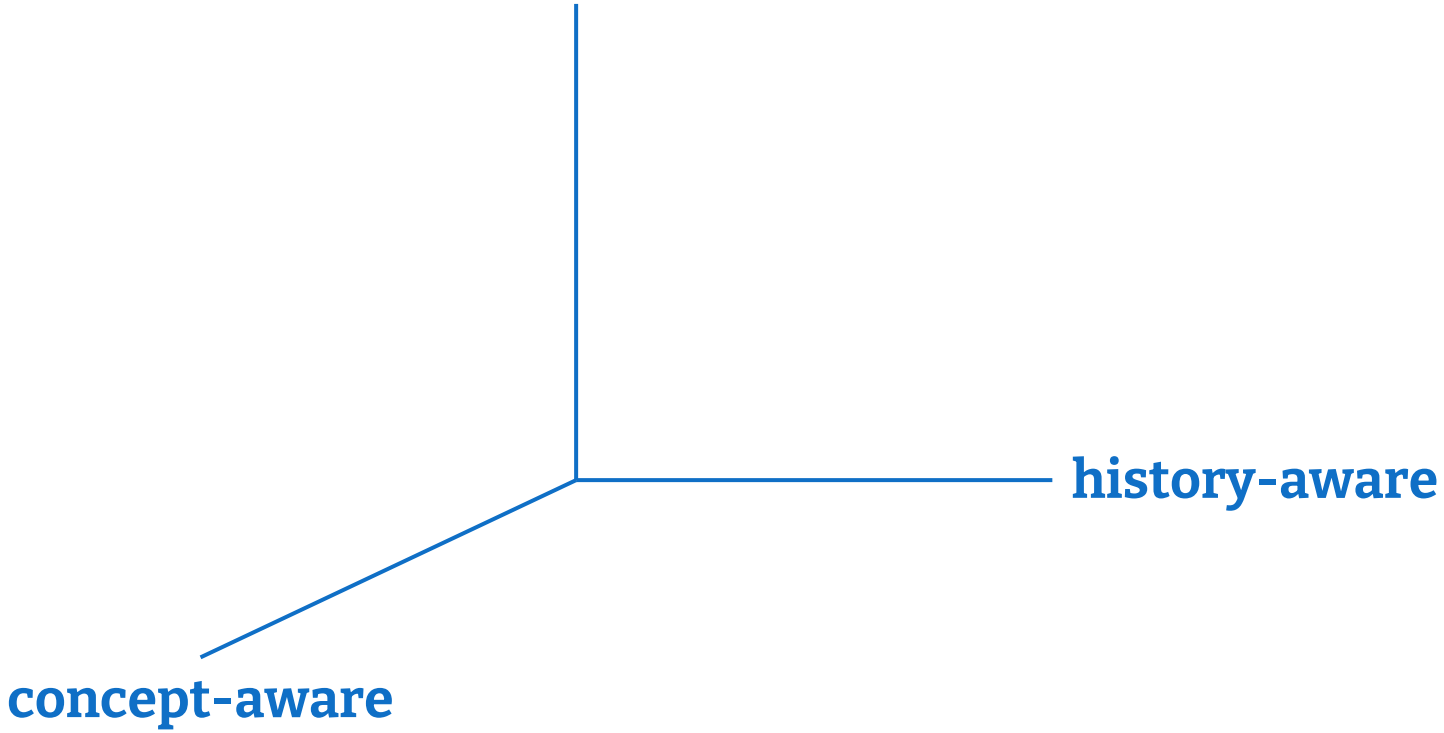
Three *Take-Away* Points from SAND

A new streaming series anomaly detection variant needs to be:



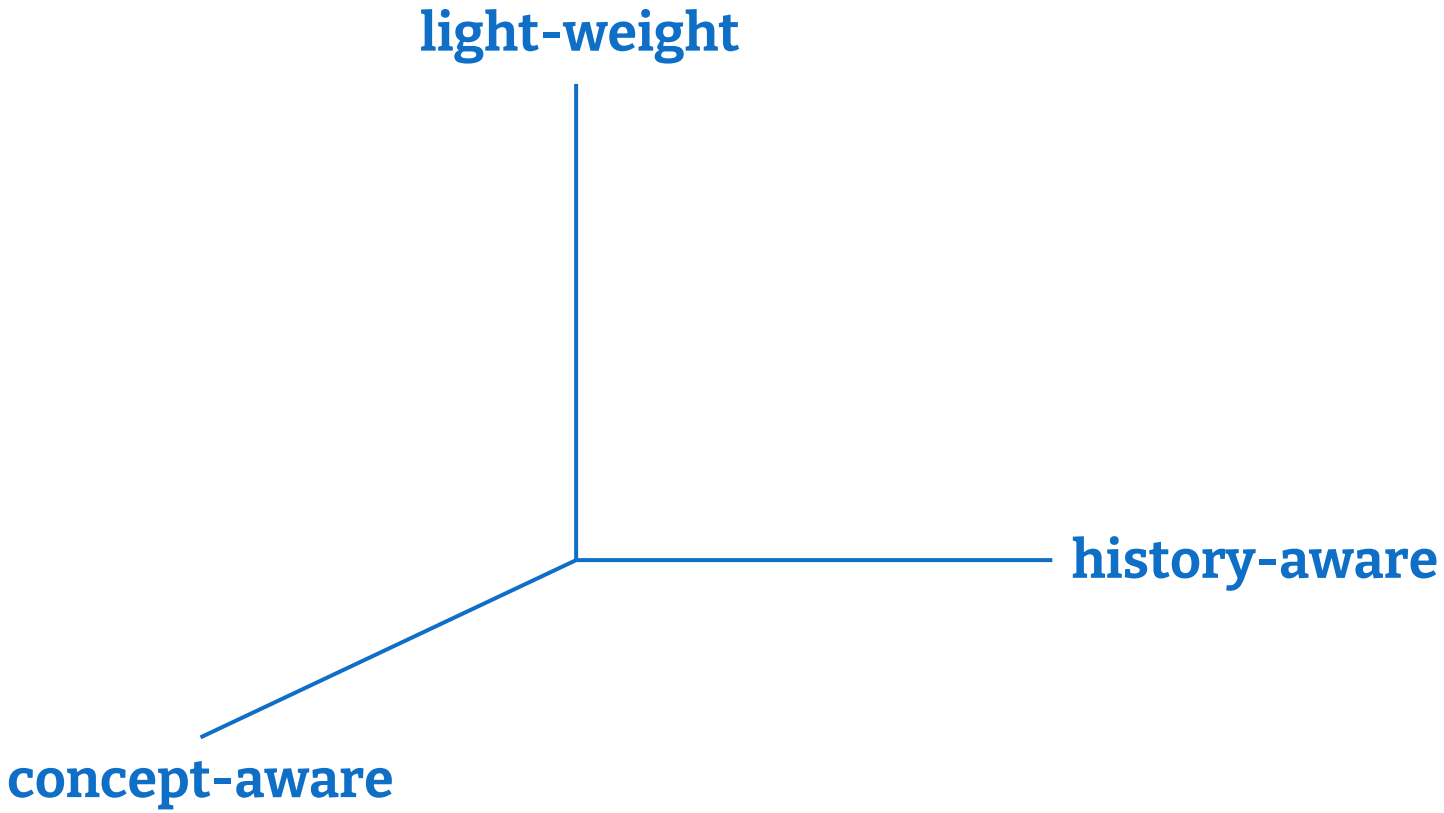
Three *Take-Away* Points from SAND

A new streaming series anomaly detection variant needs to be:



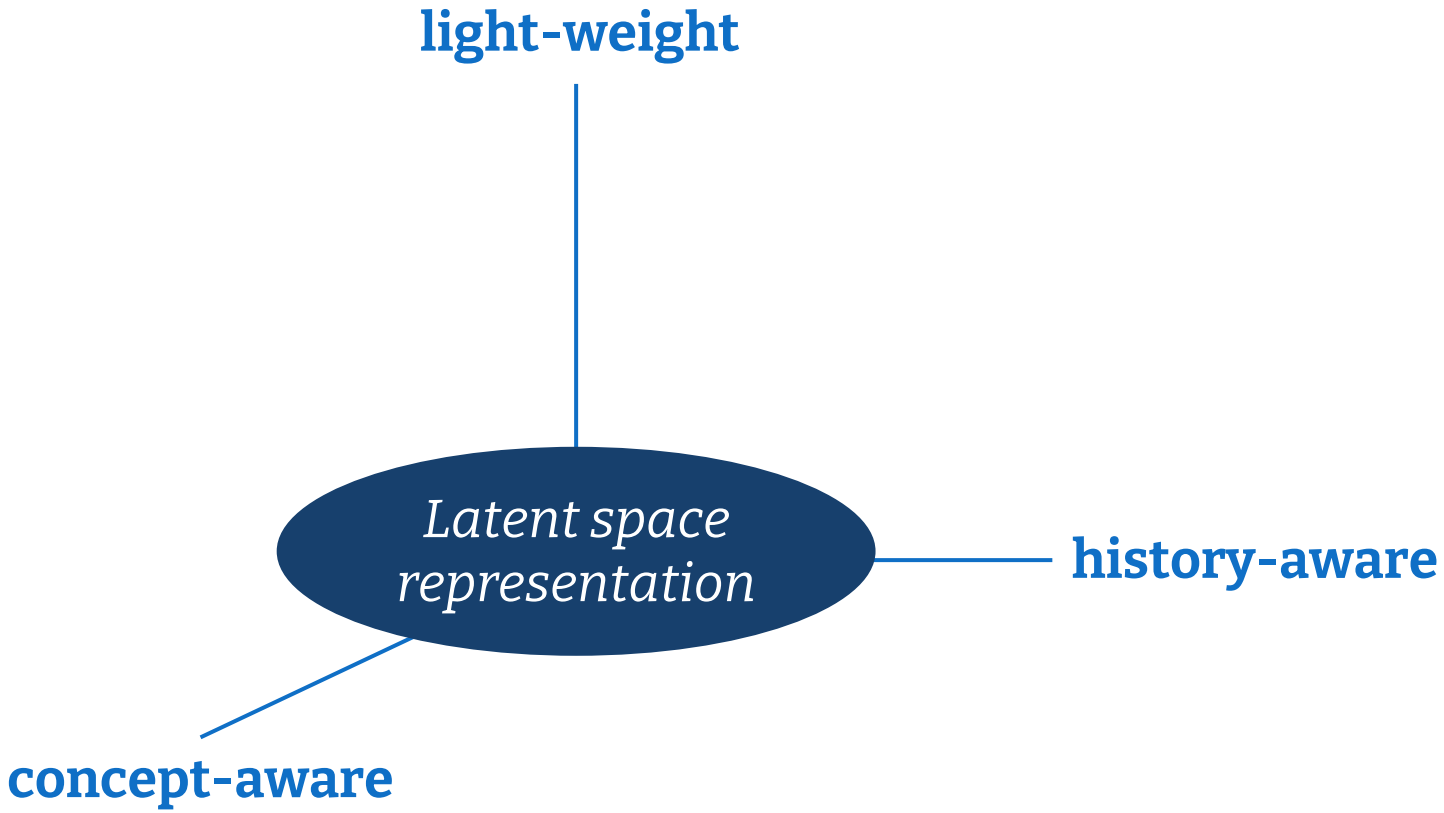
Three *Take-Away* Points from SAND

A new streaming series anomaly detection variant needs to be:

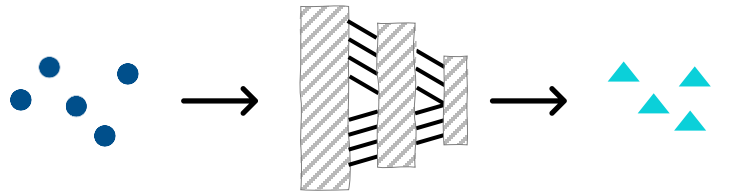


Three *Take-Away* Points from SAND

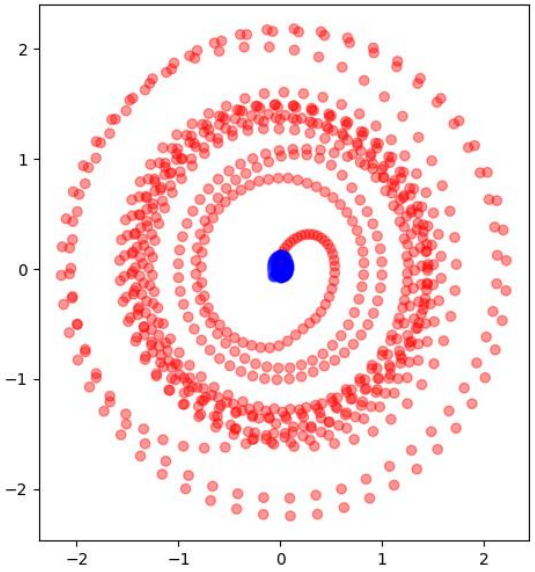
A new streaming series anomaly detection variant needs to be:



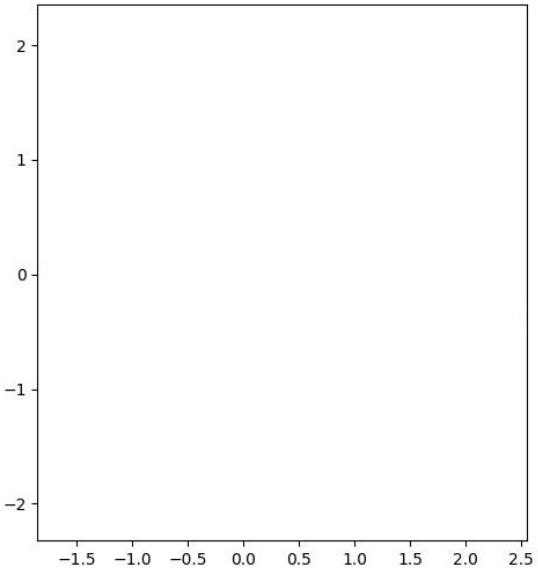
Better Separation in the Latent Space



PCA

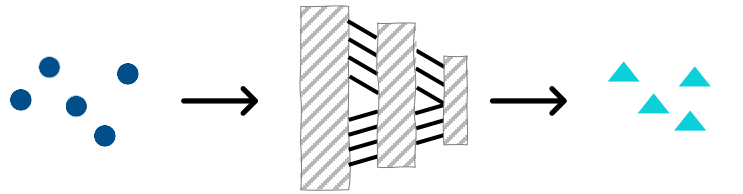


Encoder

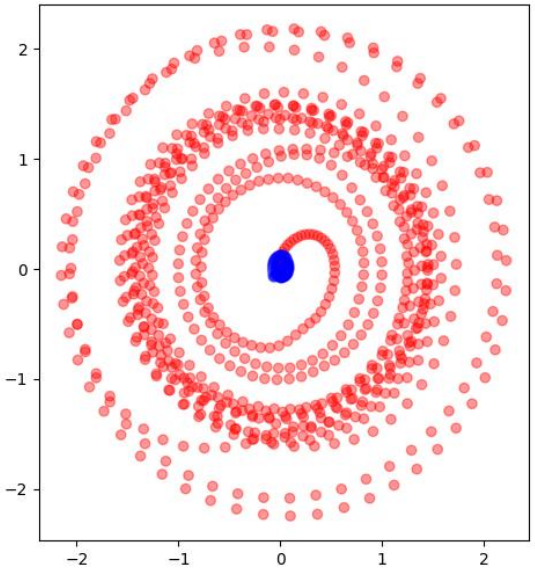


NAB dataset

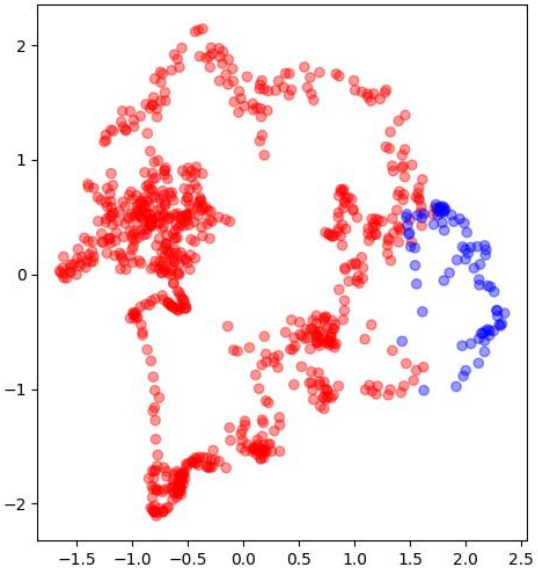
Better Separation in the Latent Space



PCA

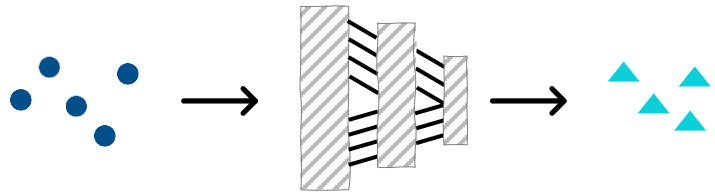


Encoder

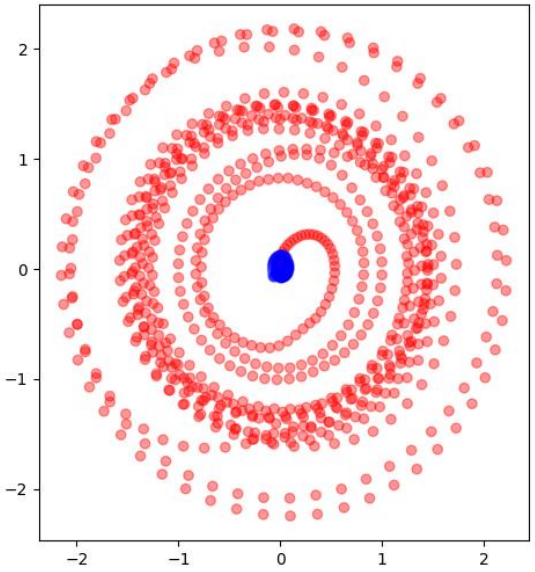


NAB dataset

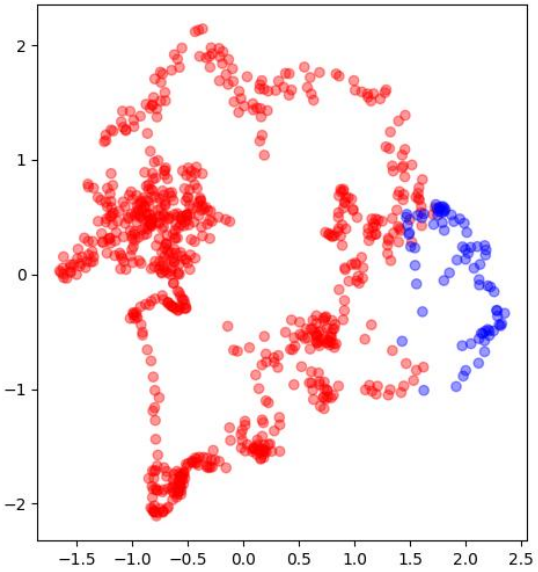
Better Separation in the Latent Space



PCA

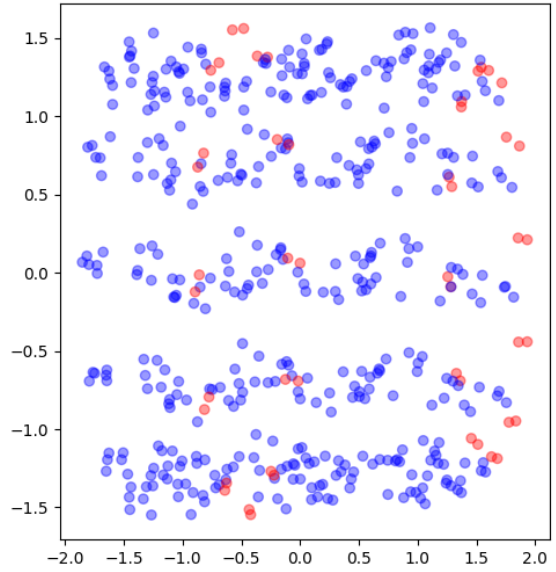


Encoder

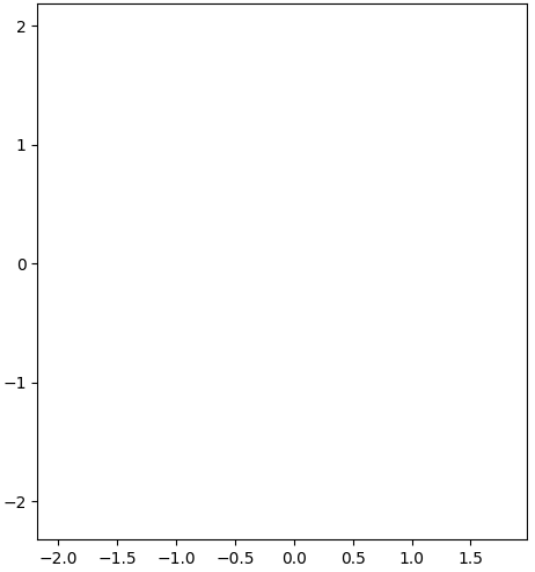


NAB dataset

PCA

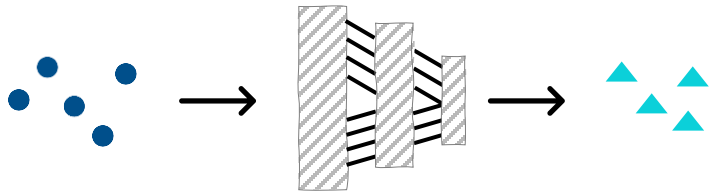


Encoder

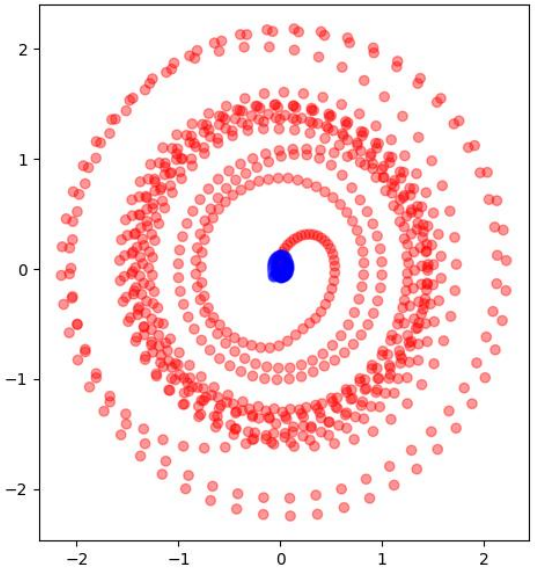


YAHOO dataset

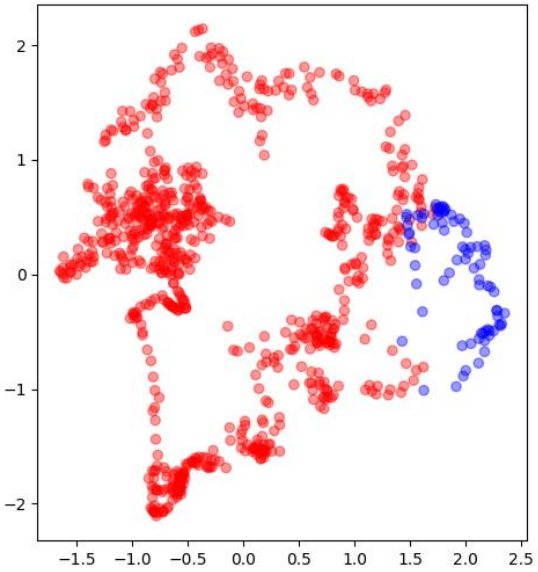
Better Separation in the Latent Space



PCA

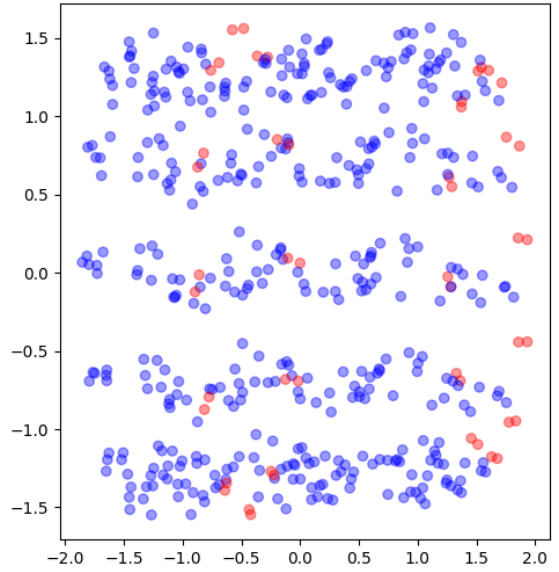


Encoder

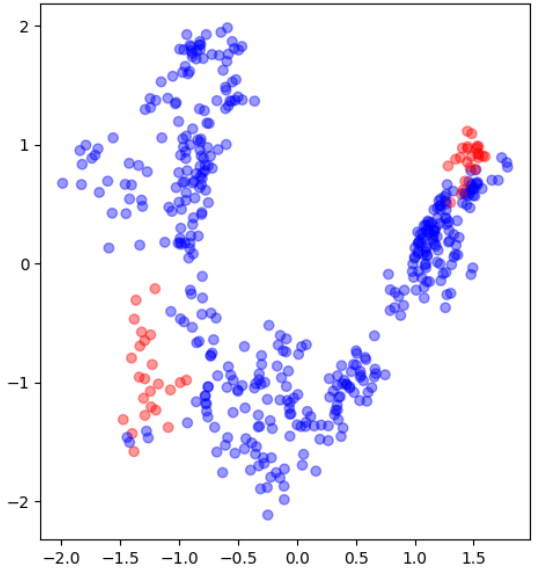


NAB dataset

PCA



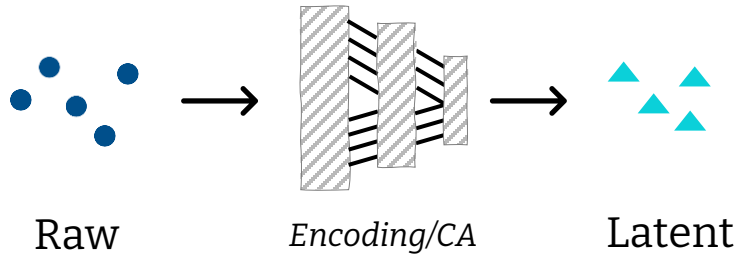
Encoder



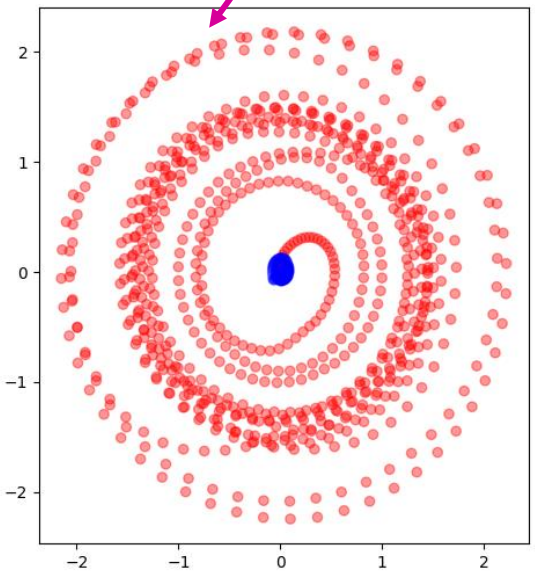
YAHOO dataset

Better Separation in the Latent Space

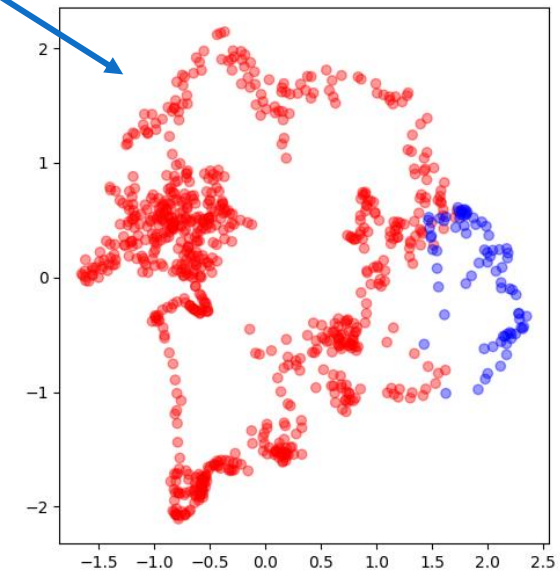
Notice the separation between anomalous and normal points in latent space.



PCA

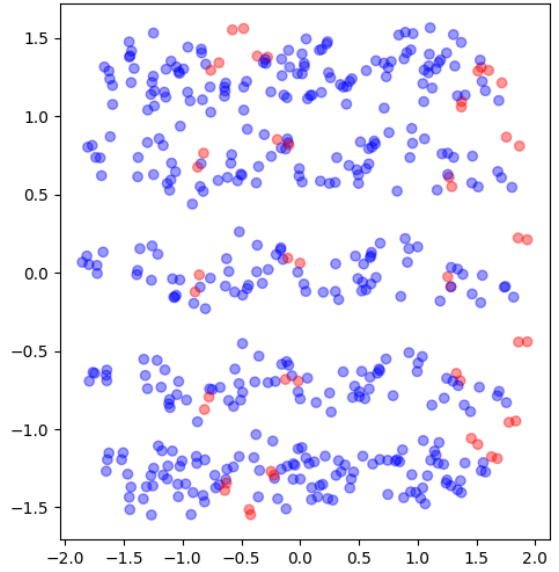


Encoder

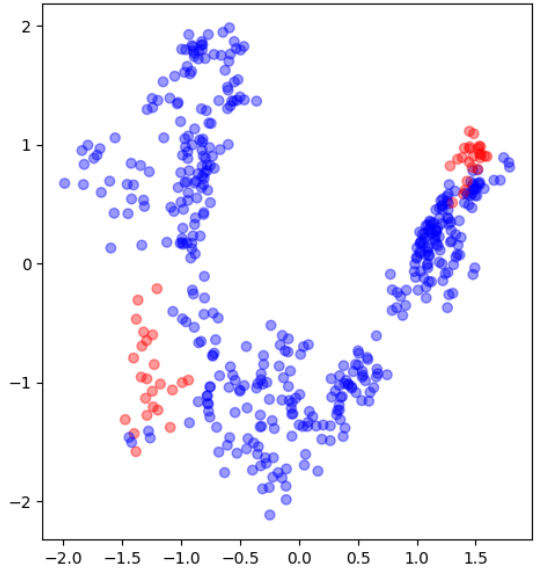


NAB dataset

PCA



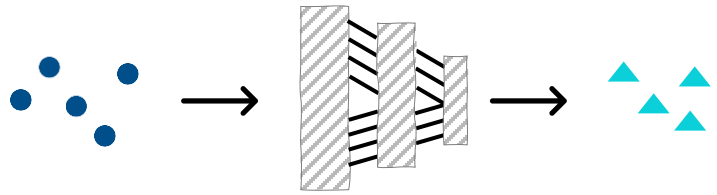
Encoder



YAHOO dataset

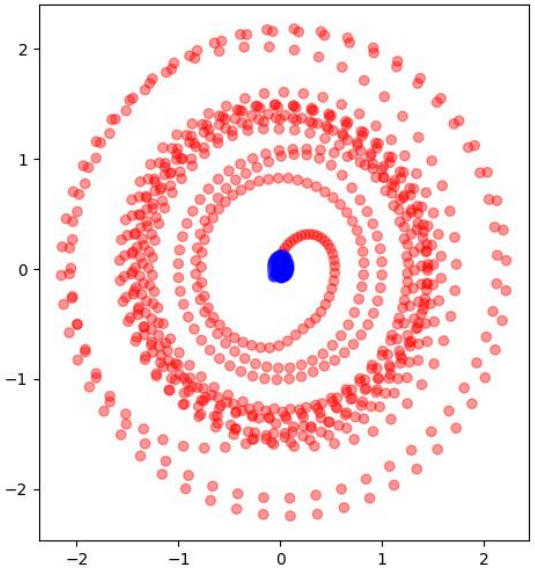
Better Separation in the Latent Space

Notice the separation between anomalous and normal points in latent space.

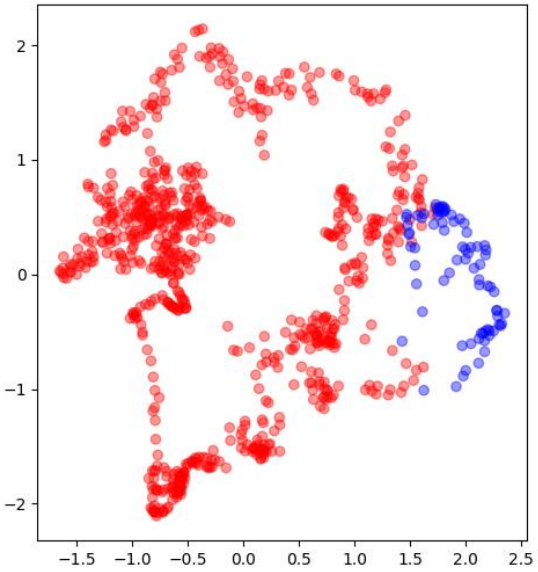


Clustering in smaller volume of data with better separation.

PCA

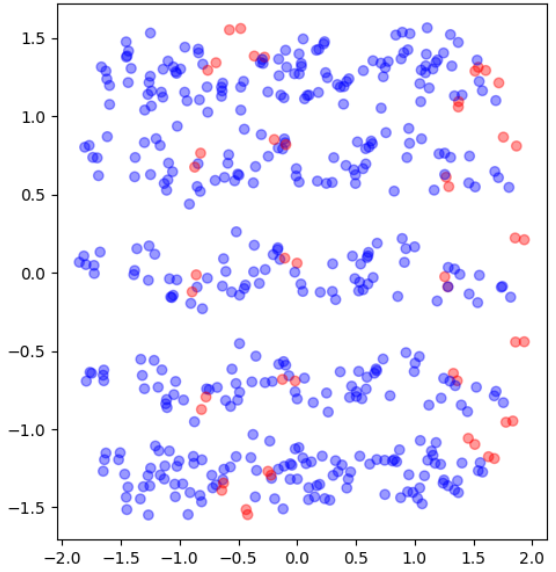


Encoder

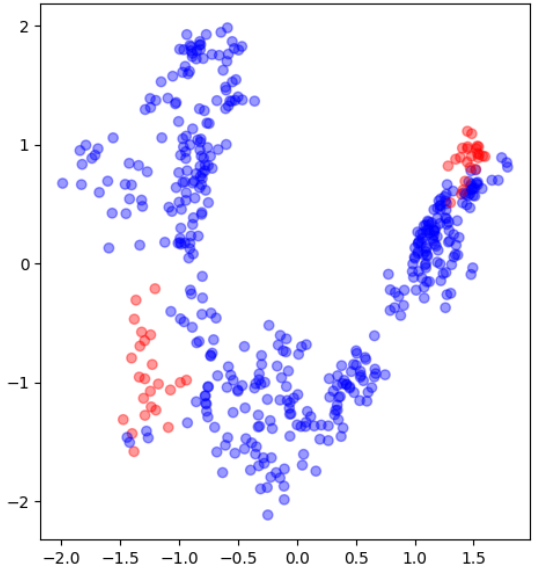


NAB dataset

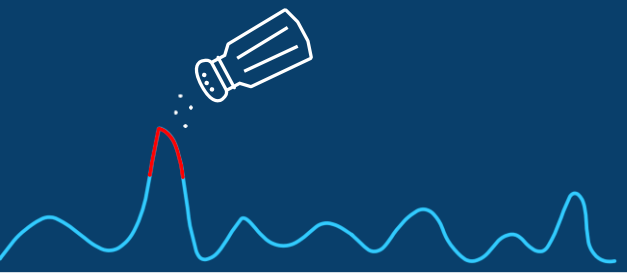
PCA



Encoder



YAHOO dataset



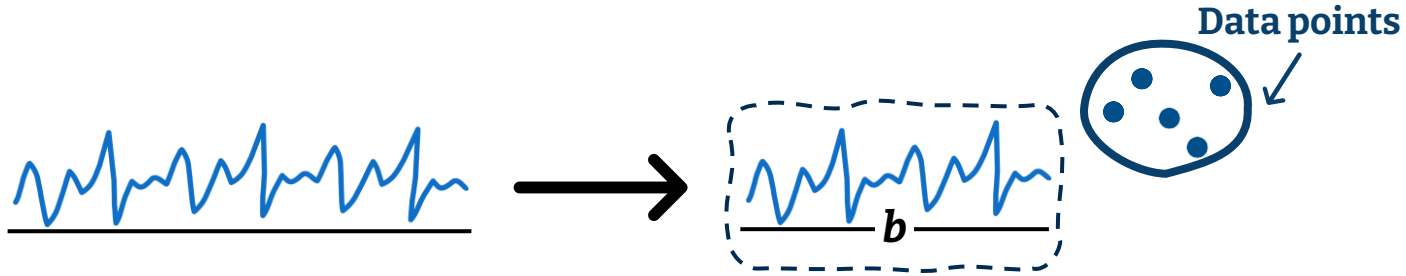
SALT

Our approach is demonstrated using the **Isolation Forest** but can be readily adapted to the **Local Outlier Factor** method.

SALT

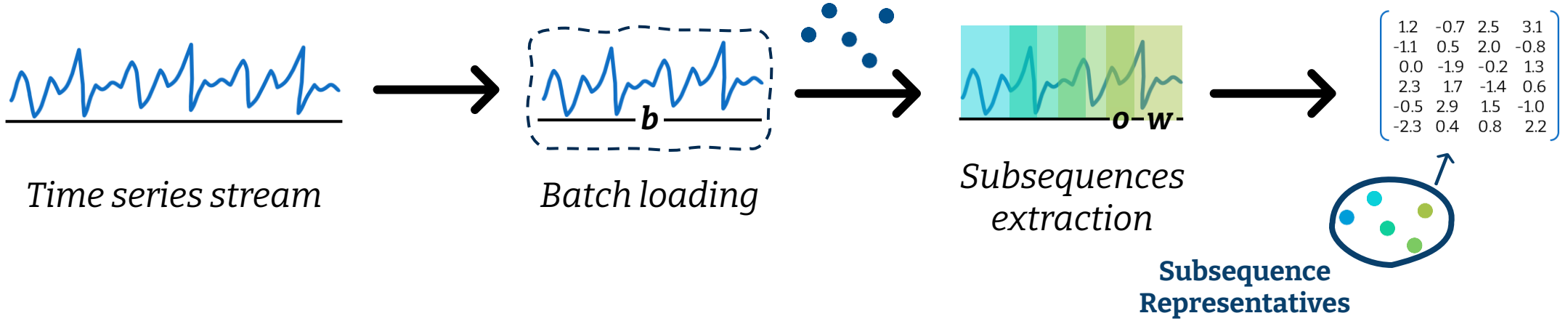
Time series stream

SALT

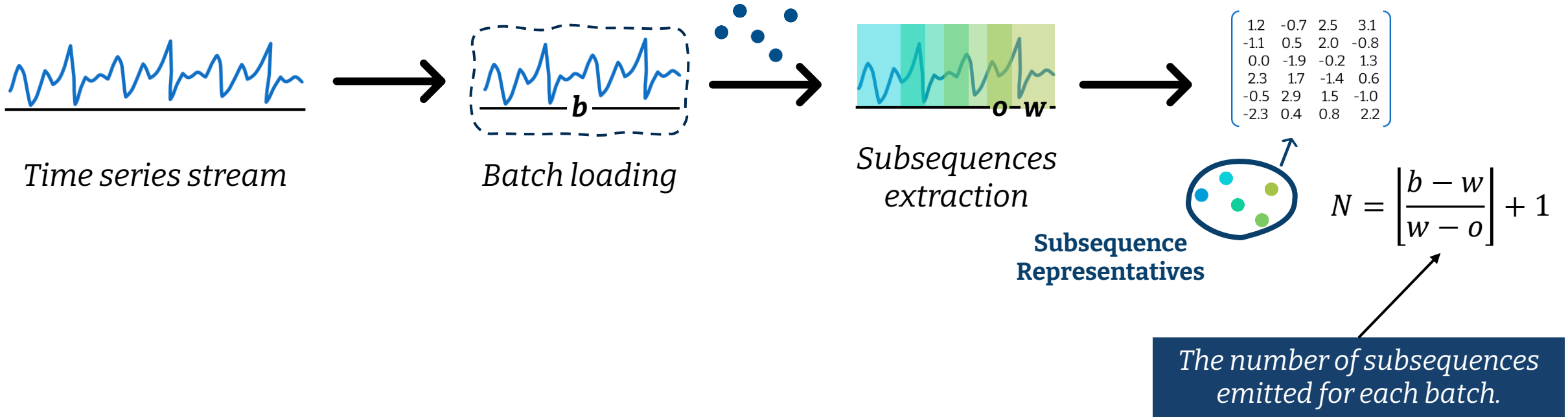


Time series stream *Batch loading*

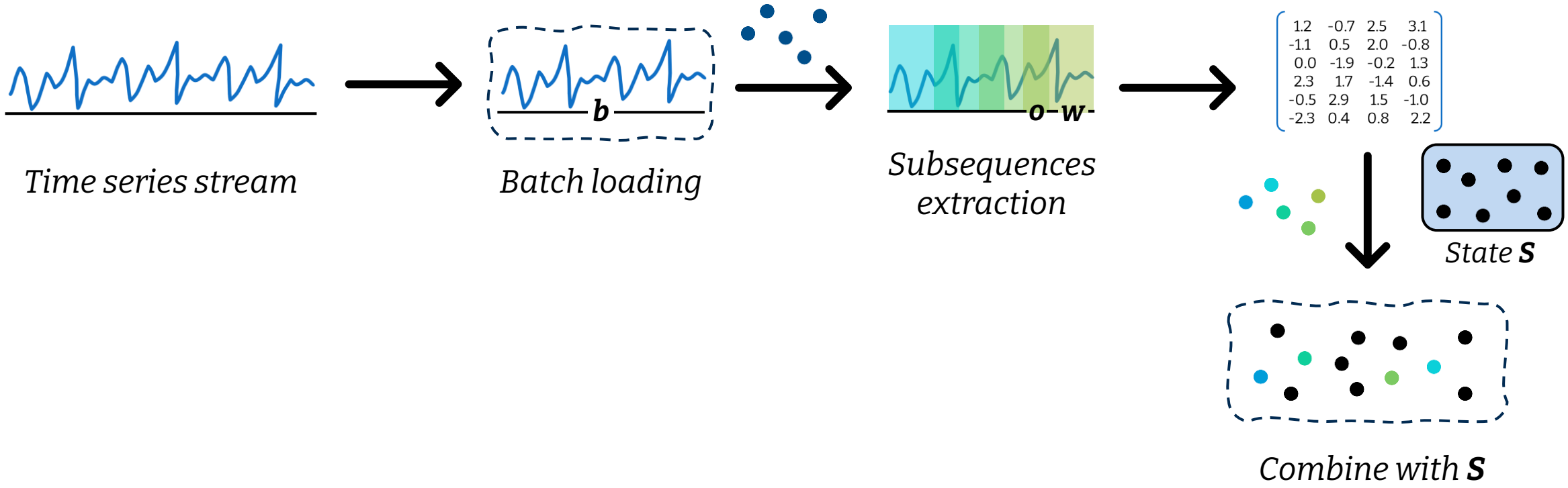
SALT


b
o–w-
1.2 -0.7 2.5 3.1
-1.1 0.5 2.0 -0.8
0.0 -1.9 -0.2 1.3
2.3 1.7 -1.4 0.6
-0.5 2.9 1.5 -1.0
-2.3 0.4 0.8 2.2
Time series stream
Batch loading
Subsequences extraction
Subsequence Representatives

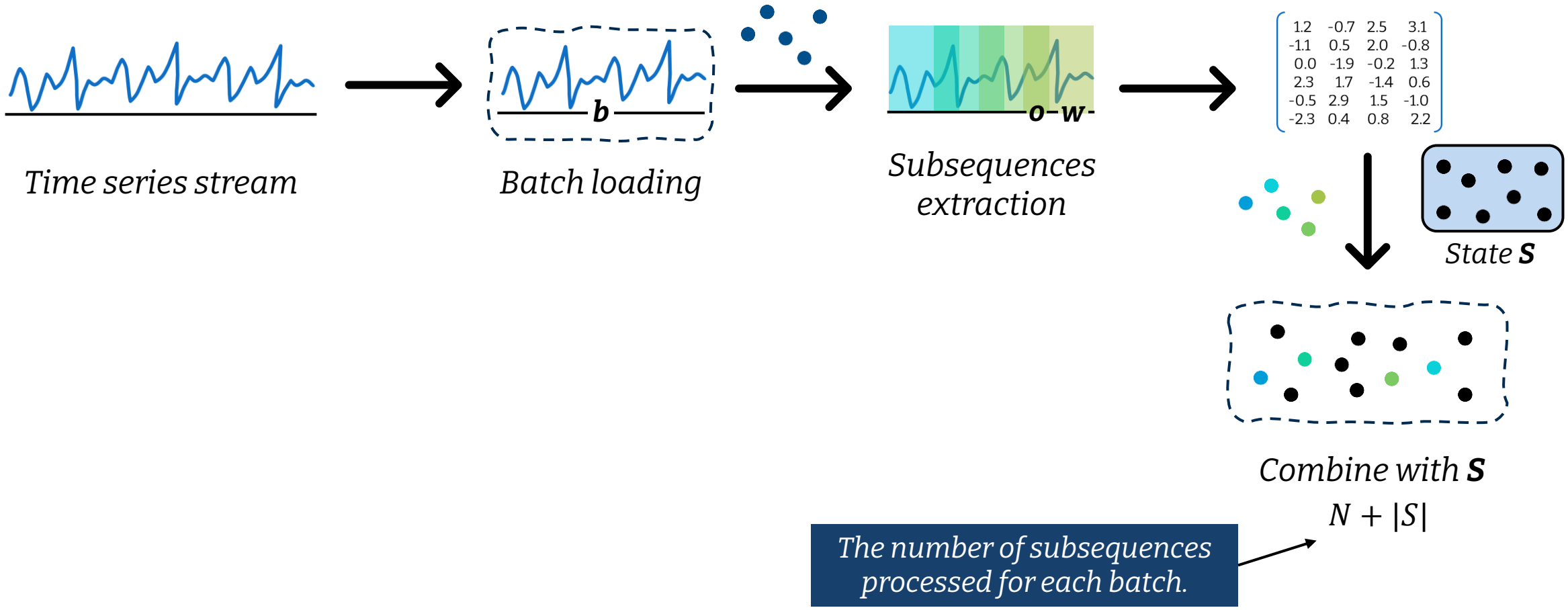
SALT


b
o–w-
1.2 -0.7 2.5 3.1
-1.1 0.5 2.0 -0.8
0.0 -1.9 -0.2 1.3
2.3 1.7 -1.4 0.6
-0.5 2.9 1.5 -1.0
-2.3 0.4 0.8 2.2
Time series stream
Batch loading
Subsequences extraction
Subsequence Representatives
$N = \left\lfloor \frac{b-w}{w-o} \right\rfloor + 1$
The number of subsequences emitted for each batch.

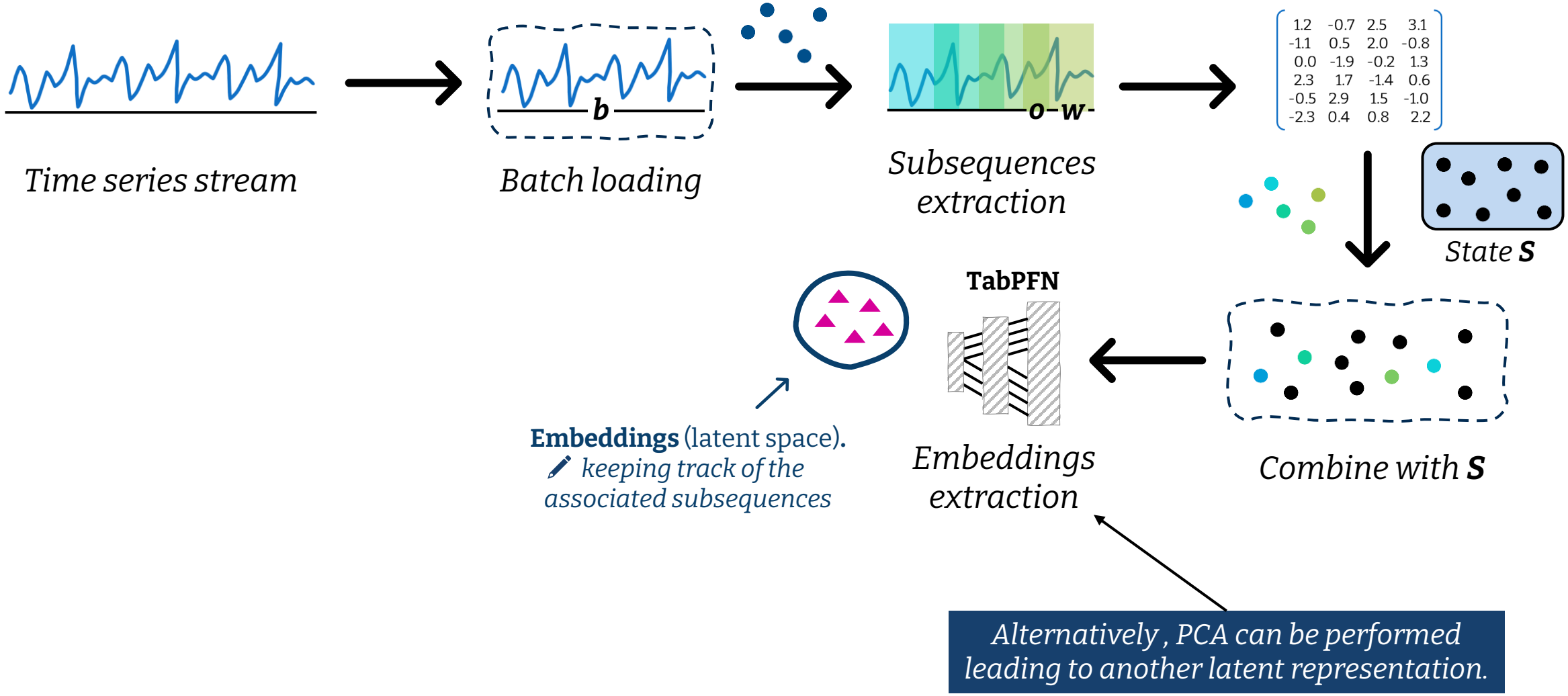
SALT


b
o–w-
1.2 -0.7 2.5 3.1
-1.1 0.5 2.0 -0.8
0.0 -1.9 -0.2 1.3
2.3 1.7 -1.4 0.6
-0.5 2.9 1.5 -1.0
-2.3 0.4 0.8 2.2
State S
Time series stream
Batch loading
Subsequences extraction
Combine with S

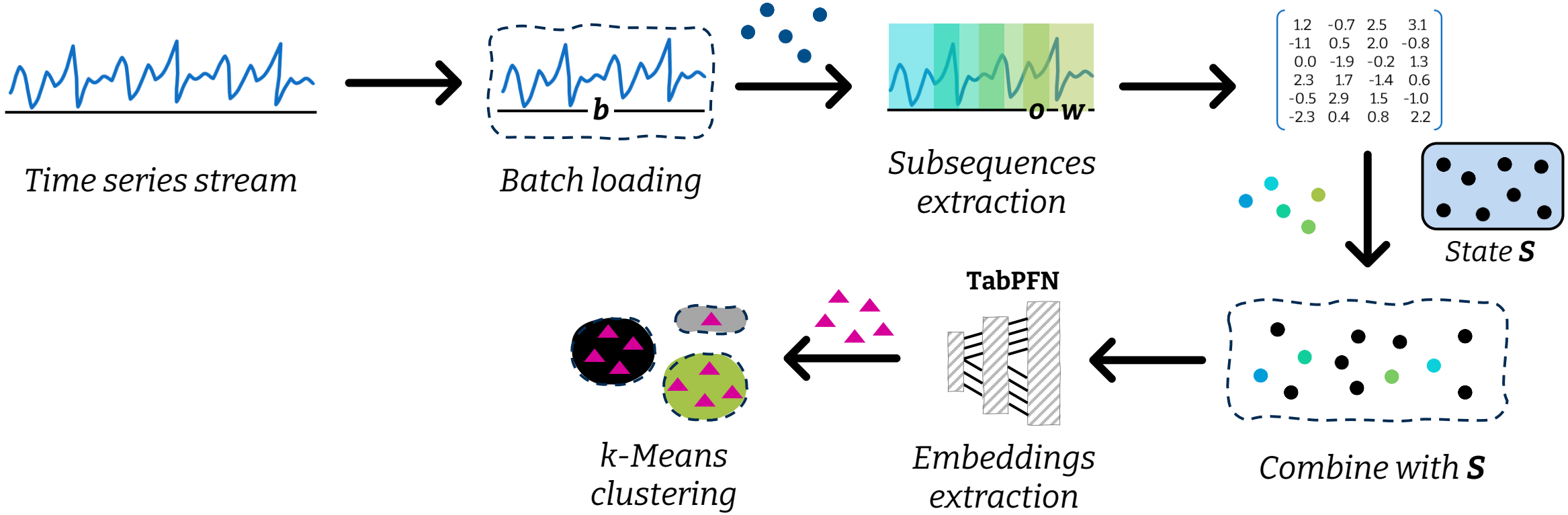
SALT


b
o–w-
1.2 -0.7 2.5 3.1
-1.1 0.5 2.0 -0.8
0.0 -1.9 -0.2 1.3
2.3 1.7 -1.4 0.6
-0.5 2.9 1.5 -1.0
-2.3 0.4 0.8 2.2
Time series stream
Batch loading
Subsequences extraction
State S
Combine with S
N + |S|
The number of subsequences processed for each batch.

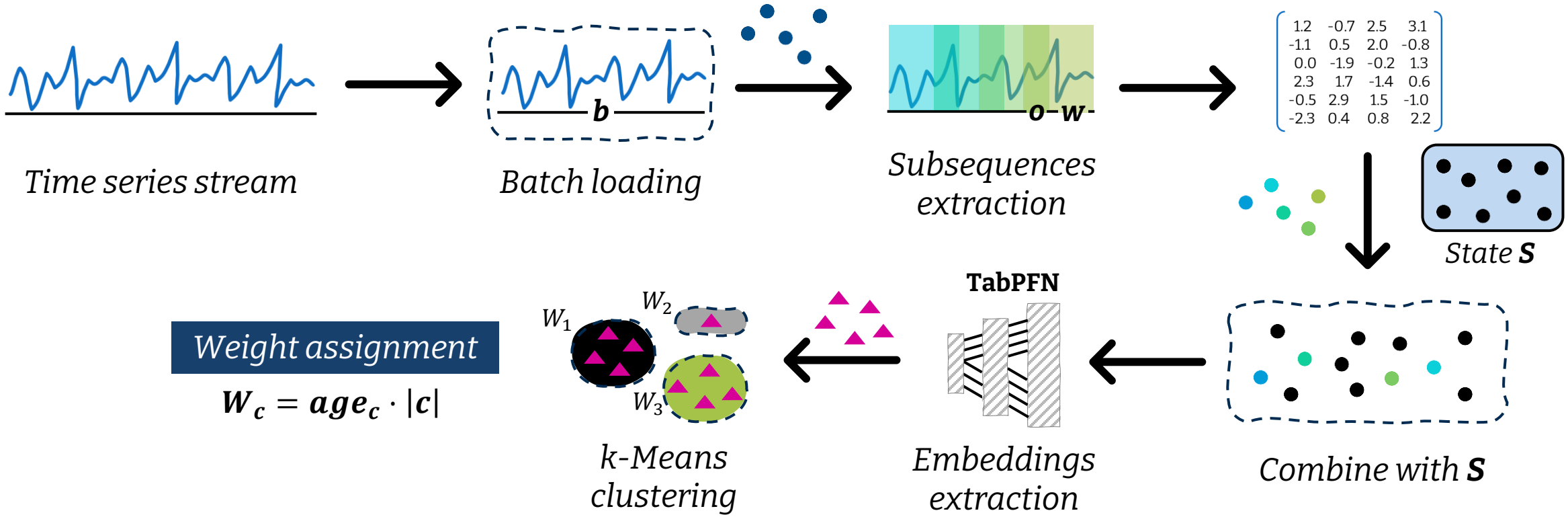
SALT


b
o–w-
1.2 -0.7 2.5 3.1
-1.1 0.5 2.0 -0.8
0.0 -1.9 -0.2 1.3
2.3 1.7 -1.4 0.6
-0.5 2.9 1.5 -1.0
-2.3 0.4 0.8 2.2
TabPFN
Time series stream
Batch loading
Subsequences extraction
State S
Combine with S
Embeddings extraction
Embeddings (latent space).
keeping track of the associated subsequences
Alternatively , PCA can be performed leading to another latent representation.

SALT


b
o–w
1.2 -0.7 2.5 3.1
-1.1 0.5 2.0 -0.8
0.0 -1.9 -0.2 1.3
2.3 1.7 -1.4 0.6
-0.5 2.9 1.5 -1.0
-2.3 0.4 0.8 2.2
TabPFN
Time series stream
Batch loading
Subsequences extraction
State S
Combine with S
Embeddings extraction
k-Means clustering

SALT



Time series stream

Batch loading

Subsequences extraction

State ***S***

Combine with ***S***

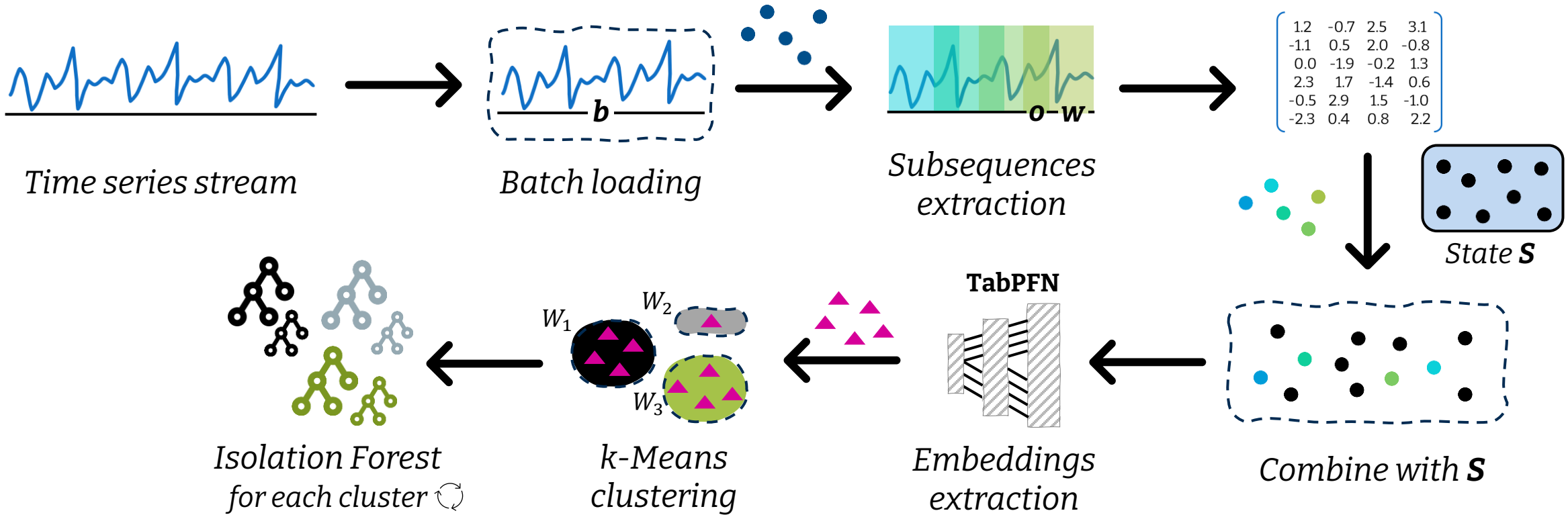
Embeddings extraction

k-Means clustering

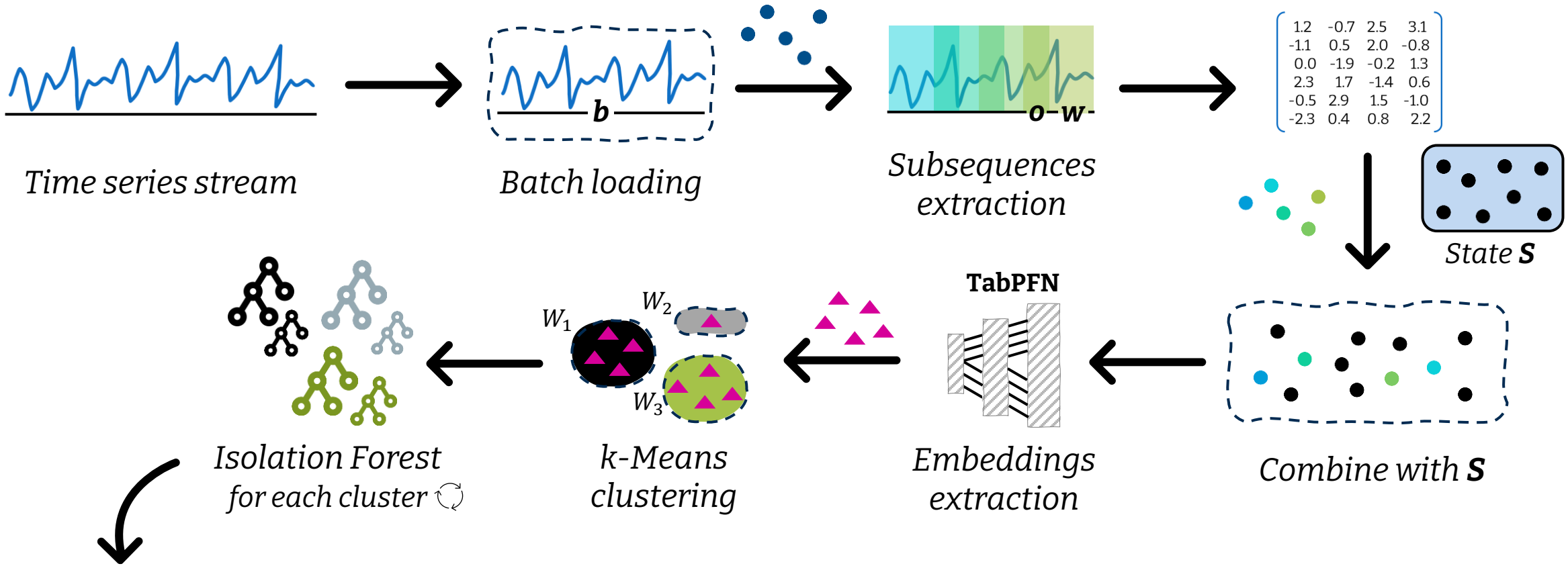
Weight assignment

$$\boldsymbol{W_c = age_c \cdot |c|}$$

SALT

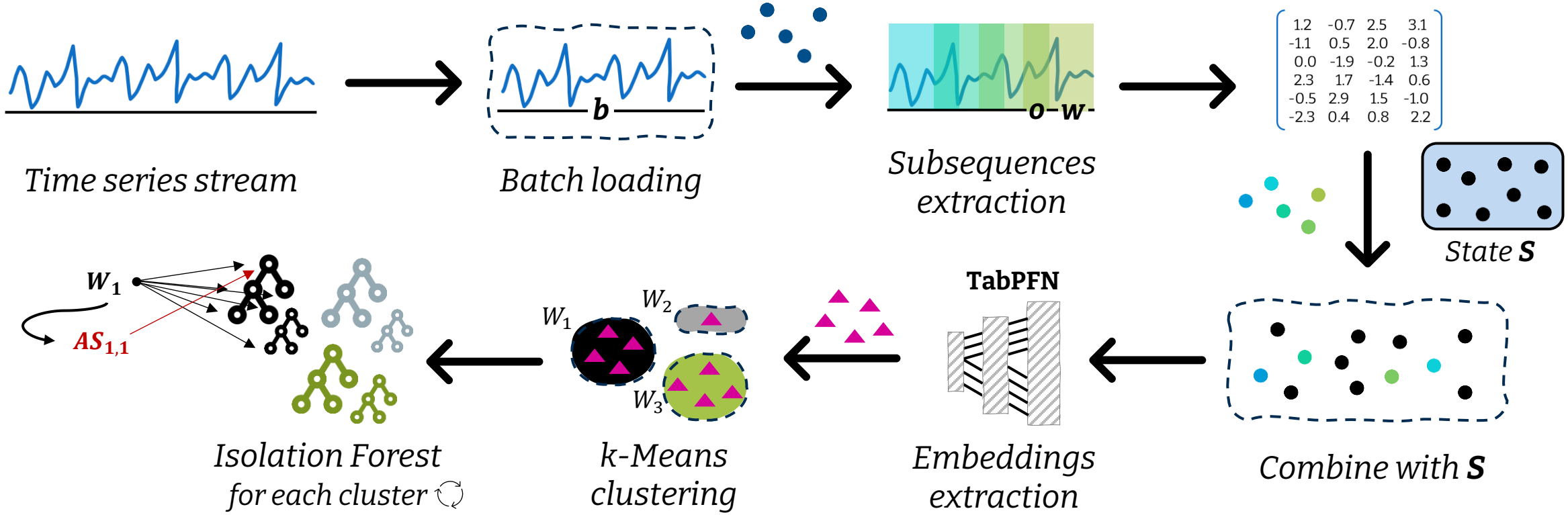

b
o–w-
1.2 -0.7 2.5 3.1
-1.1 0.5 2.0 -0.8
0.0 -1.9 -0.2 1.3
2.3 1.7 -1.4 0.6
-0.5 2.9 1.5 -1.0
-2.3 0.4 0.8 2.2
TabPFN
W_1
W_2
W_3
Time series stream
Batch loading
Subsequences extraction
State S
Combine with S
Embeddings extraction
k-Means clustering
Isolation Forest
for each cluster

SALT



* Trivial adaptation for LOF and any other subsequence-based AD method

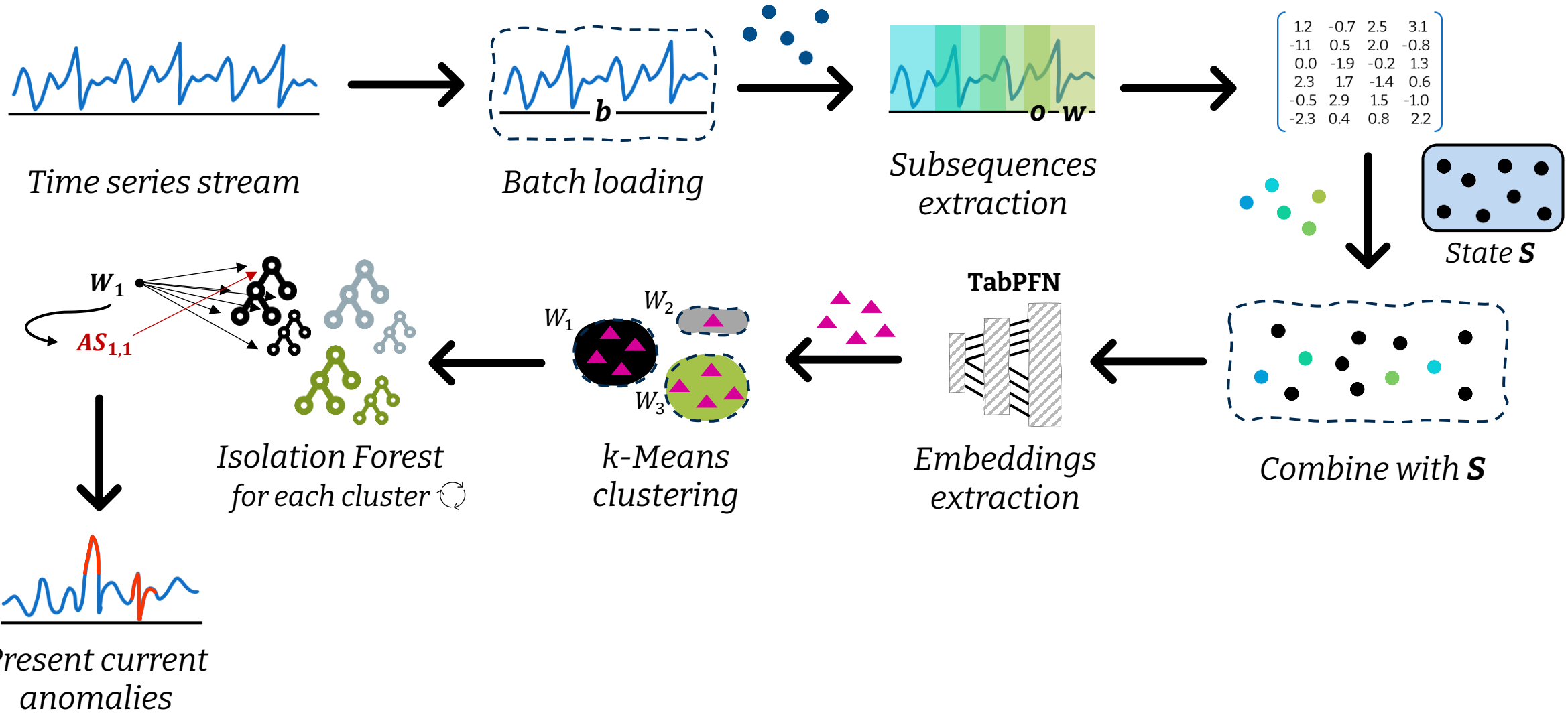
SALT



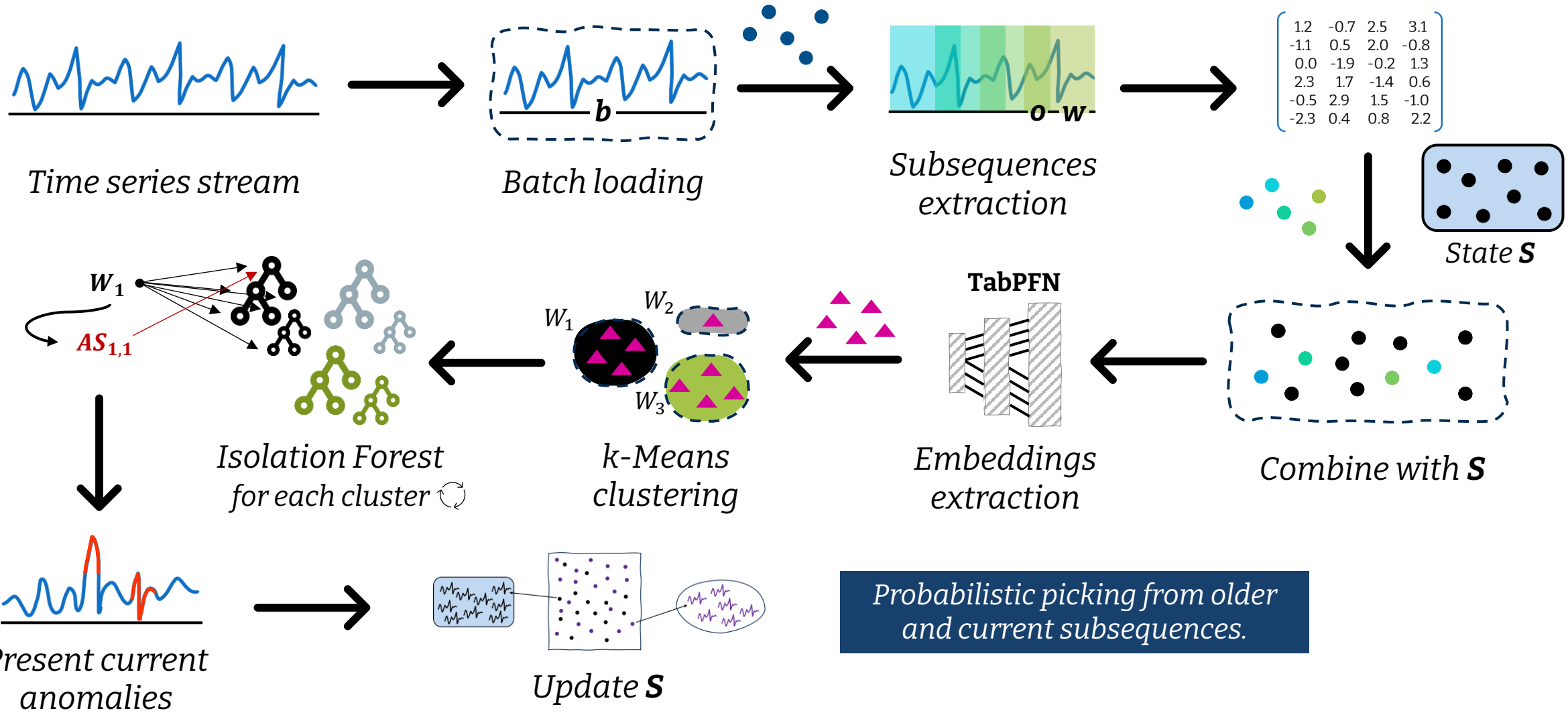
Time series stream → *Batch loading* → *Subsequences extraction* → *State* ***S*** → *Combine with* ***S*** → *Embeddings extraction* → *k-Means clustering* → *Isolation Forest for each cluster* ↻

Each subsequence has an ***anomaly score*** *weighted by its cluster's* ***weight****.*

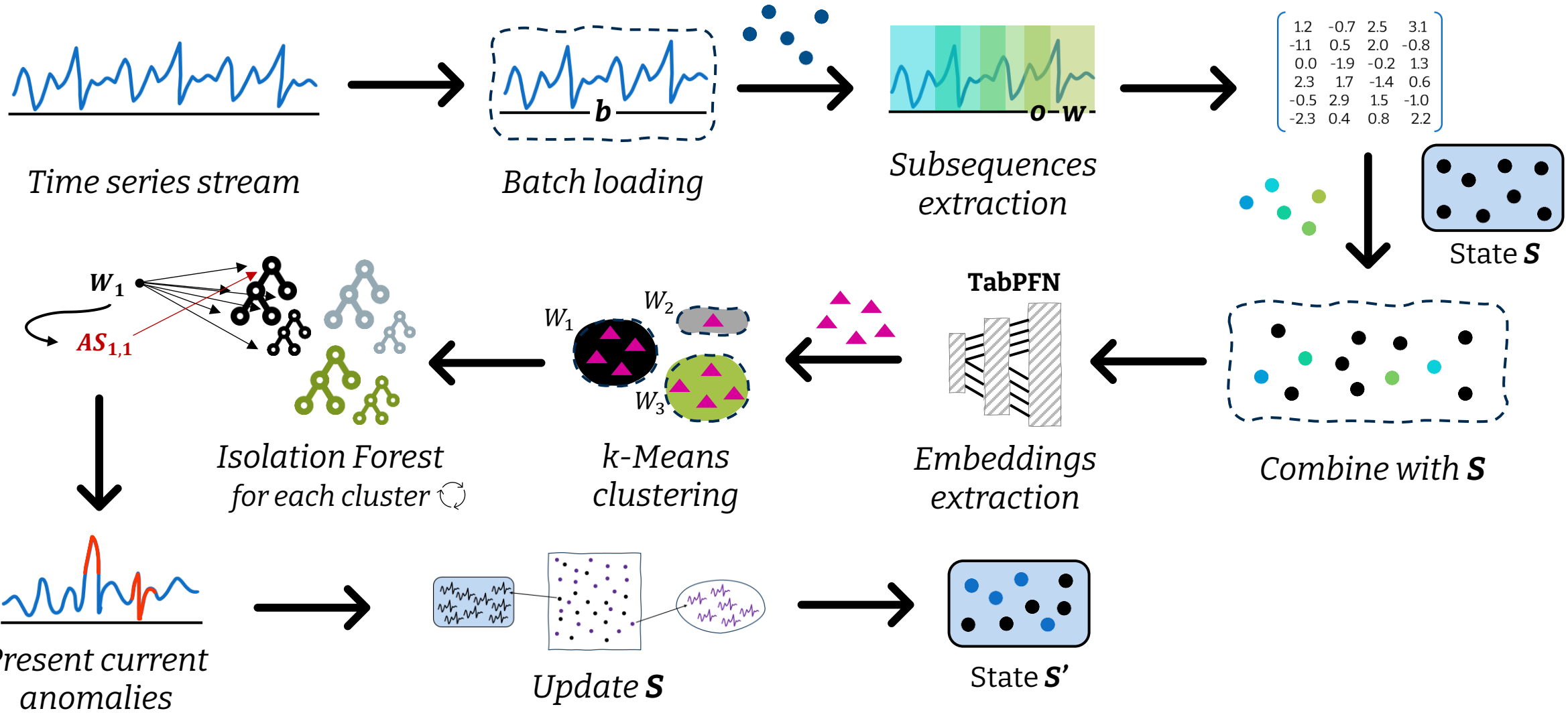
SALT


Time series stream
b
Batch loading
o-w-
Subsequences extraction
1.2 -0.7 2.5 3.1
-1.1 0.5 2.0 -0.8
0.0 -1.9 -0.2 1.3
2.3 1.7 -1.4 0.6
-0.5 2.9 1.5 -1.0
-2.3 0.4 0.8 2.2
State S
Combine with S
TabPFN
Embeddings extraction
W1
W2
W3
k-Means clustering
W1
AS1,1
Isolation Forest for each cluster
Present current anomalies

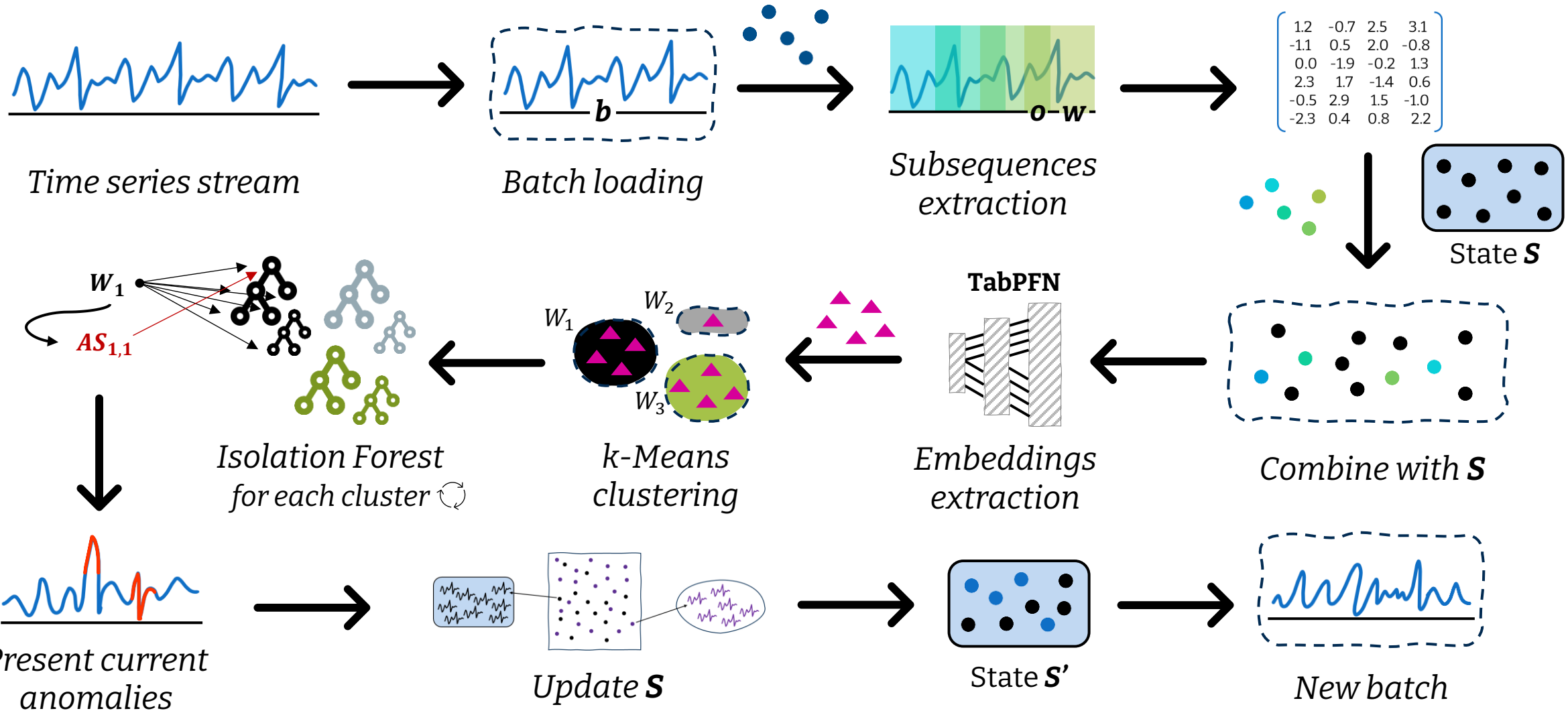
SALT


b
o–w-
1.2 -0.7 2.5 3.1
-1.1 0.5 2.0 -0.8
0.0 -1.9 -0.2 1.3
2.3 1.7 -1.4 0.6
-0.5 2.9 1.5 -1.0
-2.3 0.4 0.8 2.2
TabPFN
W_1
W_2
W_3
W_1
$AS_{1,1}$
Time series stream
Batch loading
Subsequences extraction
State S
Combine with S
Embeddings extraction
k-Means clustering
Isolation Forest for each cluster
Present current anomalies
Update S
Probabilistic picking from older and current subsequences.

SALT

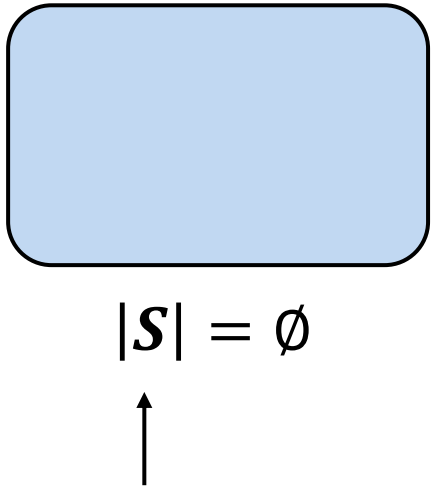

Time series stream
b
Batch loading
o-w-
Subsequences extraction
1.2 -0.7 2.5 3.1
-1.1 0.5 2.0 -0.8
0.0 -1.9 -0.2 1.3
2.3 1.7 -1.4 0.6
-0.5 2.9 1.5 -1.0
-2.3 0.4 0.8 2.2
State S
Combine with S
TabPFN
Embeddings extraction
W1
W2
W3
k-Means clustering
W1
AS1,1
Isolation Forest for each cluster
Present current anomalies
Update S
State S'

SALT


Time series stream
b
Batch loading
o–w-
Subsequences extraction
1.2 -0.7 2.5 3.1
-1.1 0.5 2.0 -0.8
0.0 -1.9 -0.2 1.3
2.3 1.7 -1.4 0.6
-0.5 2.9 -1.5 -1.0
-2.3 0.4 0.8 2.2
State S
TabPFN
Embeddings extraction
Combine with S
W1
W2
W3
k-Means clustering
W1
AS1,1
Isolation Forest
for each cluster
Present current anomalies
Update S
State S'
New batch

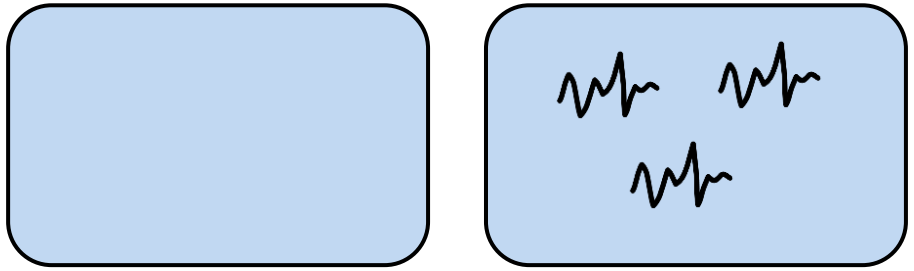
Evolution of State

State $\boldsymbol{S}$ is a **MEM-stored** and S^T**-upper-bounded-size** **list of subsequences** persistently pertained and updated in the streaming environment.



Evolution of State

State $\boldsymbol{S}$ is a **MEM-stored** and S^T**-upper-bounded-size** **list of subsequences** persistently pertained and updated in the streaming environment.



$|\boldsymbol{S}| = \emptyset$

Cardinality of $\boldsymbol{S}$

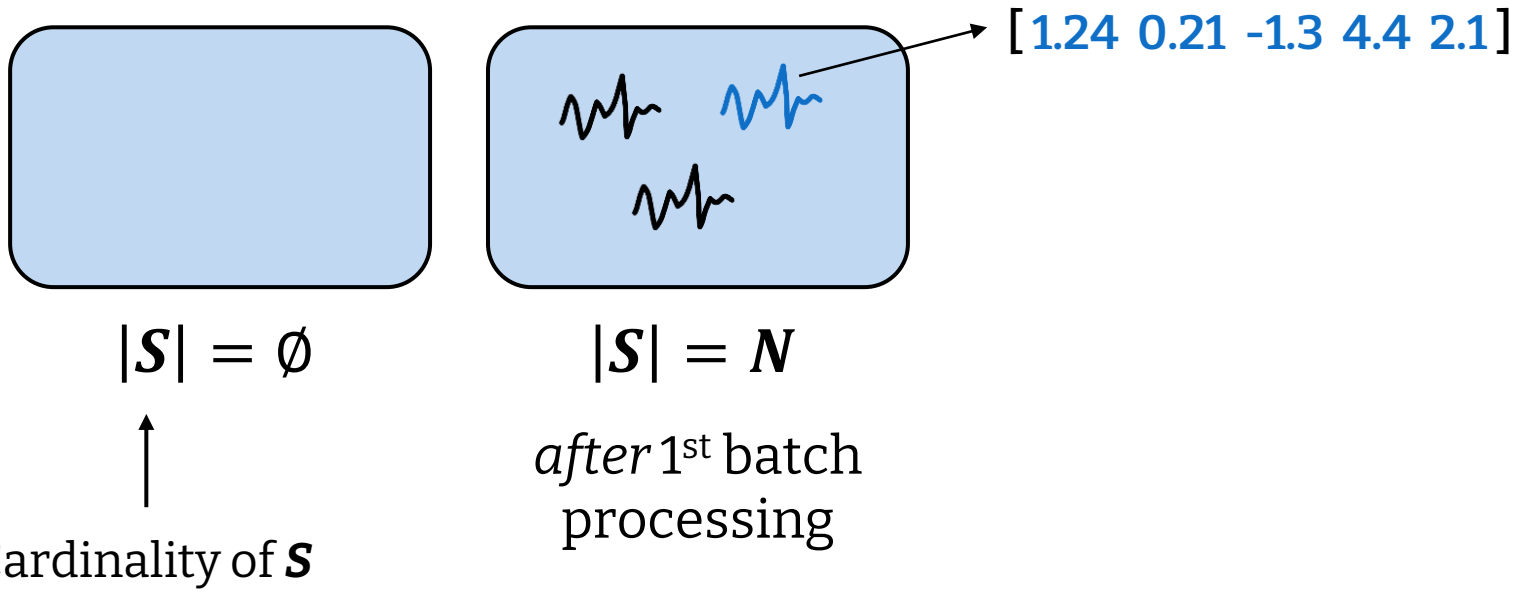
$|\boldsymbol{S}| = \boldsymbol{N}$

after 1st batch processing

* $N = \left\lfloor \frac{b-w}{w-o} \right\rfloor + 1$

Evolution of State

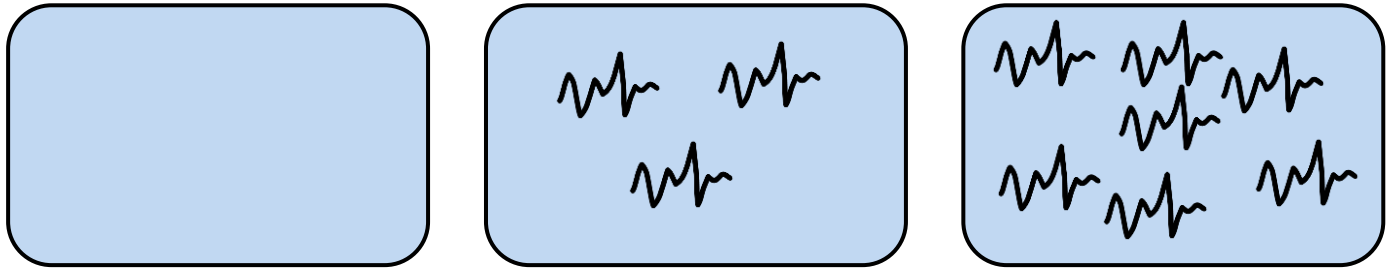
State $\boldsymbol{S}$ is a **MEM-stored** and S^T**-upper-bounded-size** **list of subsequences** persistently pertained and updated in the streaming environment.



* $N = \left\lfloor \frac{b-w}{w-o} \right\rfloor + 1$

Evolution of State

State $\boldsymbol{S}$ is a **MEM-stored** and $\boldsymbol{S^T}$**-upper-bounded-size** **list of subsequences** persistently pertained and updated in the streaming environment.



$|\boldsymbol{S}| = \emptyset$

Cardinality of $\boldsymbol{S}$

$|\boldsymbol{S}| = \boldsymbol{N}$

after 1st batch processing

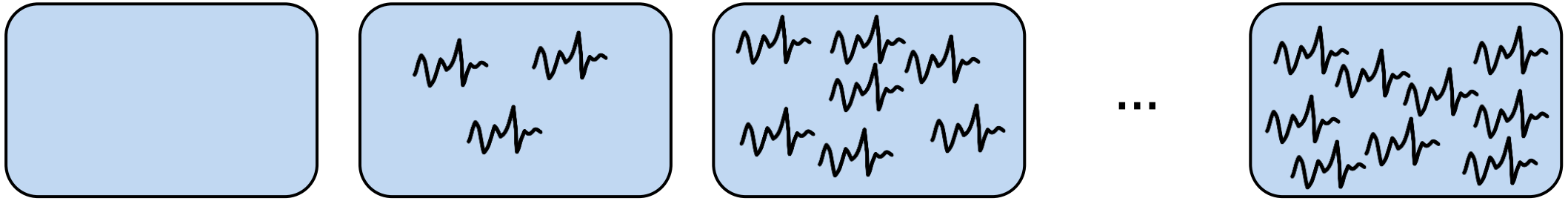
$|\boldsymbol{S}| = \boldsymbol{2N}$

after 2nd batch processing

* $N = \left\lfloor \frac{b-w}{w-o} \right\rfloor + 1$

Evolution of State

State $\boldsymbol{S}$ is a **MEM-stored** and $\boldsymbol{S^T}$**-upper-bounded-size** **list of subsequences** persistently pertained and updated in the streaming environment.



$|\boldsymbol{S}| = \emptyset$

Cardinality of $\boldsymbol{S}$

$|\boldsymbol{S}| = \boldsymbol{N}$

after 1st batch processing

$|\boldsymbol{S}| = \boldsymbol{2N}$

after 2nd batch processing

...

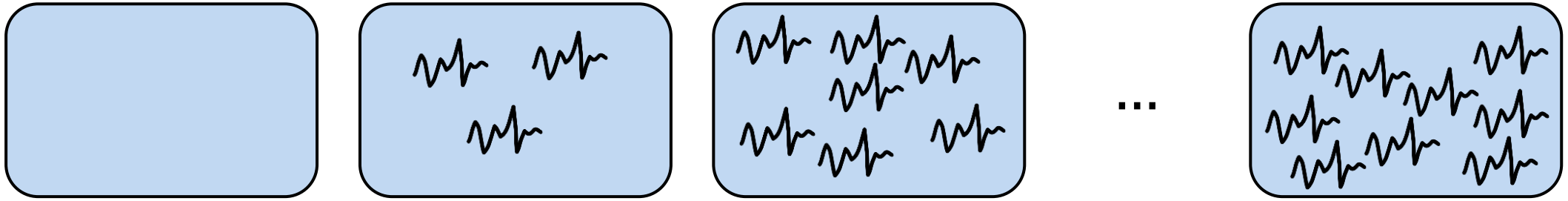
$|\boldsymbol{S}| = \boldsymbol{S^T}$

after the n^{th} *batch processing*

* $N = \left\lfloor \frac{b-w}{w-o} \right\rfloor + 1$

Evolution of State

State $\boldsymbol{S}$ is a **MEM-stored** and $\boldsymbol{S^T}$**-upper-bounded-size** **list of subsequences** persistently pertained and updated in the streaming environment.



$|\boldsymbol{S}| = \emptyset$

Cardinality of $\boldsymbol{S}$

$|\boldsymbol{S}| = \boldsymbol{N}$

after 1st batch processing

$|\boldsymbol{S}| = \boldsymbol{2N}$

after 2nd batch processing

...

$|\boldsymbol{S}| = \boldsymbol{S^T}$

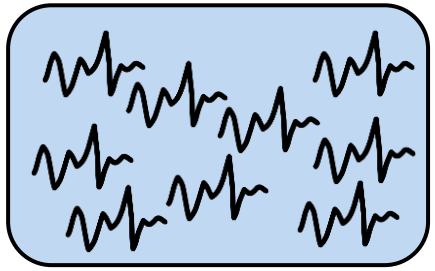
after the n^{th} *batch processing*

Low-resource in terms of memory.

* $N = \left\lfloor \frac{b-w}{w-o} \right\rfloor + 1$

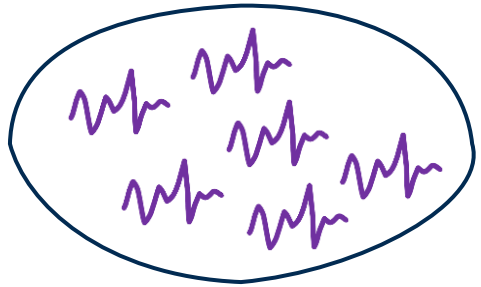
Evolution of State

Once the S^T threshold is met, we proceed to **weighted sampling** among $\boldsymbol{S}$'s data and current batch's ones.



$$|\boldsymbol{S}| = \boldsymbol{S^T}$$

after the n^{th} batch processing



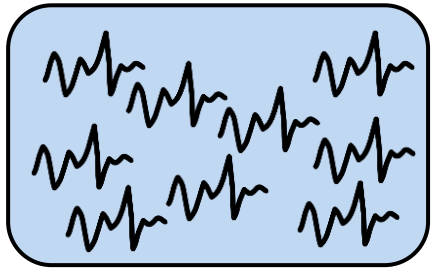
$$|\boldsymbol{B_{n+1}}| = \boldsymbol{N}$$

current subsequences

* $N = \left\lfloor \frac{b-w}{w-o} \right\rfloor + 1$

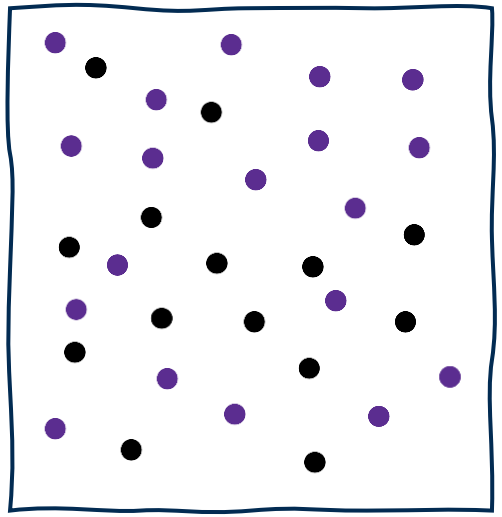
Evolution of State

Once the S^T threshold is met, we proceed to **weighted sampling** among ***S***'s data and current batch's ones.

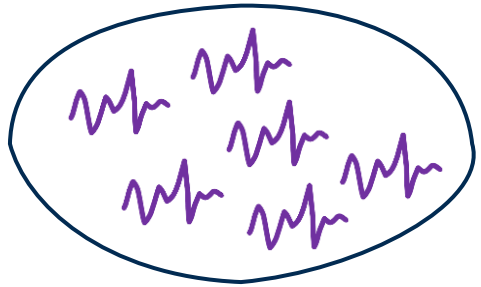


$$|S| = S^T$$

after the n^{th} batch processing



Probabilistic Sampling



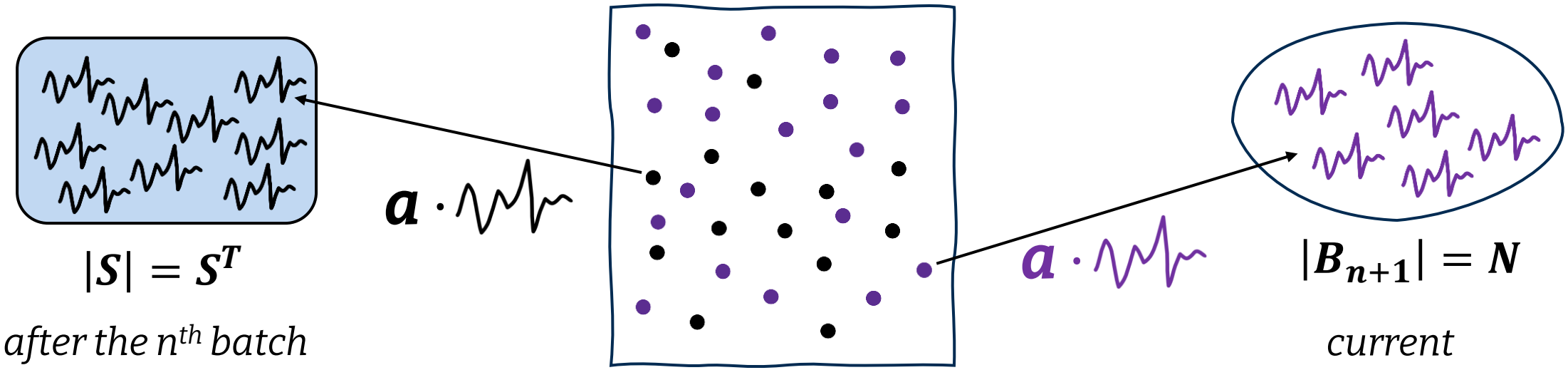
$$|B_{n+1}| = N$$

current subsequences

* $N = \left\lfloor \frac{b-w}{w-o} \right\rfloor + 1$

Evolution of State

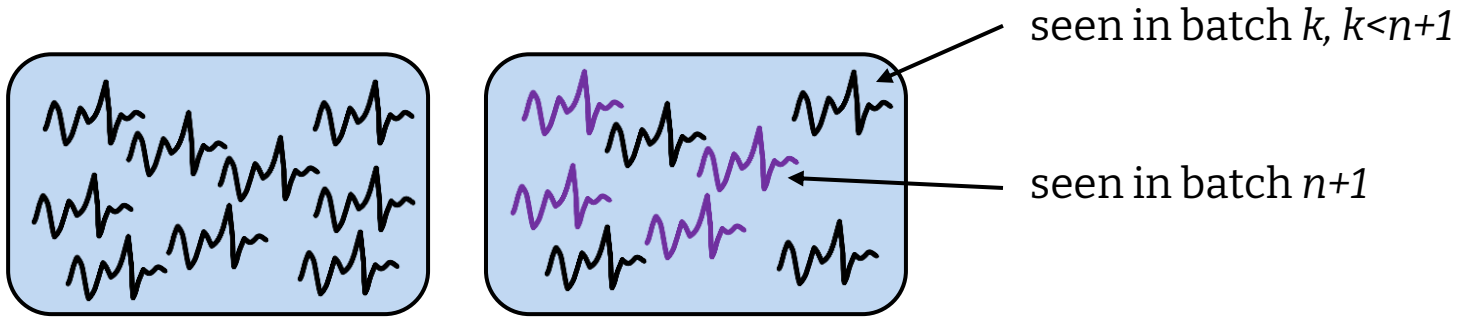
Once the S^T threshold is met, we proceed to **weighted sampling** among S's data and current batch's ones.



* $N = \left\lfloor \frac{b-w}{w-o} \right\rfloor + 1$

Evolution of State

The new state, ***S'***, consists of some subsequences from both former and latter batches.



$|S| = S^T$

after the n^{th} batch processing

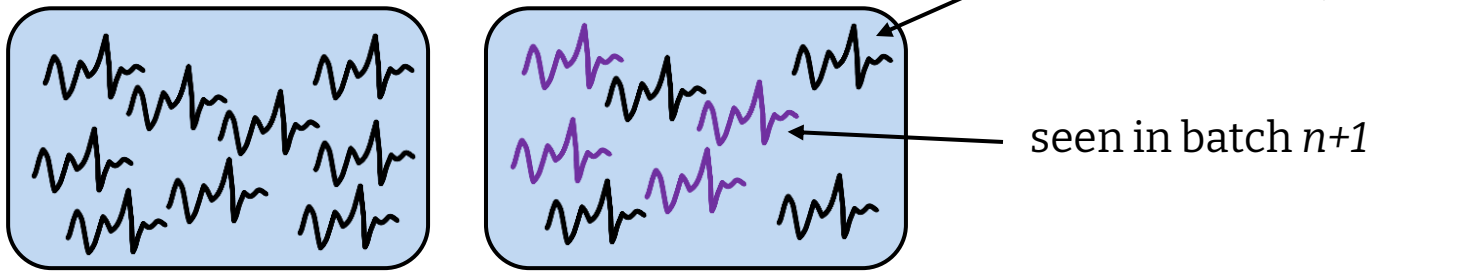
$|S| = S^T$

after $(n+1)^{th}$ batch processing

- ***S*** tries to pertain both older and newer behavior.
- Newer subsequences represent the current status of the series.

Evolution of State

The new state, ***S'***, consists of some subsequences from both former and latter batches.



$|S| = S^T$

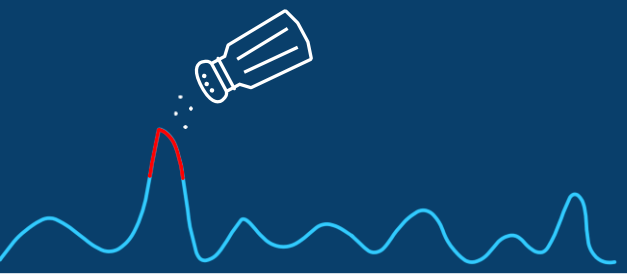
after the n^{th} batch processing

$|S| = S^T$

after $(n+1)^{th}$ batch processing

- ***S*** tries to pertain both older and newer behavior.
- Newer subsequences represent the current status of the series.

This ensures **concept drift** is considered in future clustering.



Evaluation

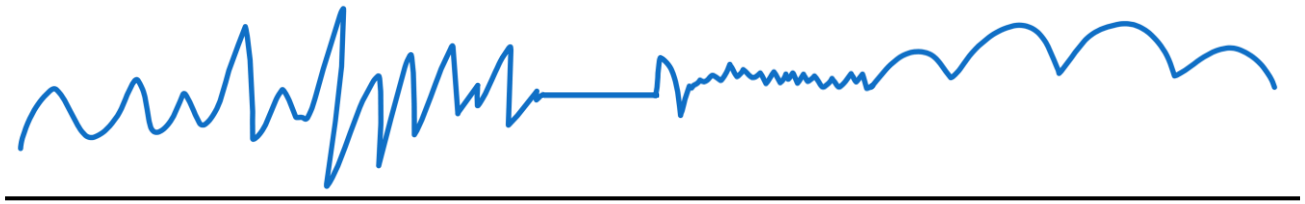
SALT with IF and LOF

SETTING

We pick time series from two datasets contained in TSB; NAB and YAHOO.

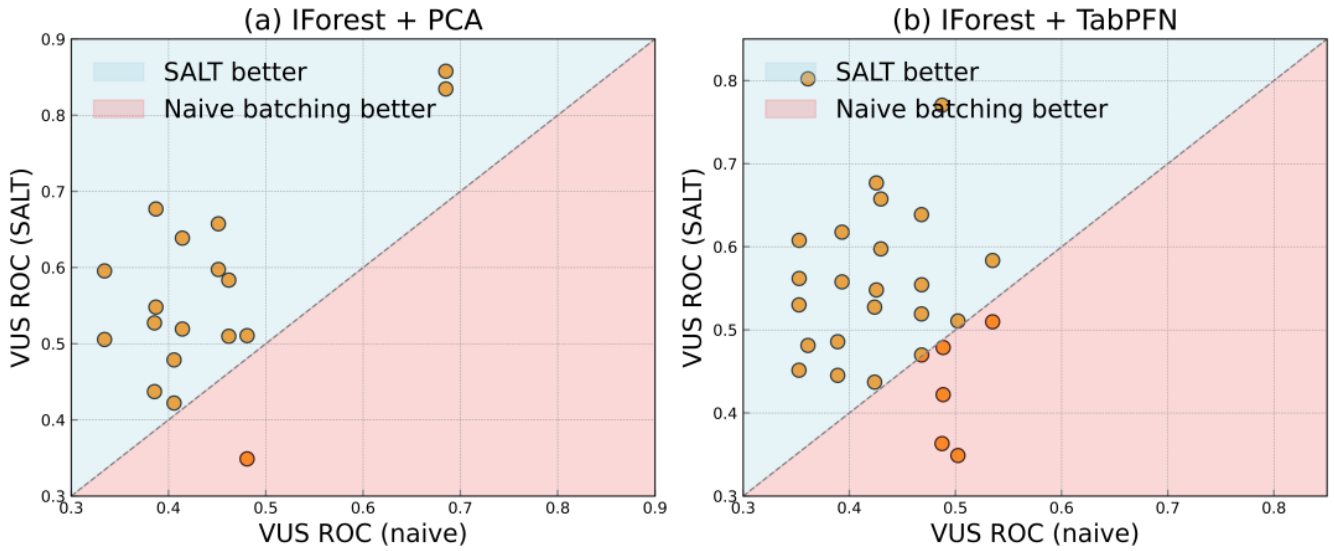
Dataset	Count	Average length	Average number of anomalies	Average number of abnormal points
NAB	58	6301.7	2.0	575.5
YAHOO	367	1561.2	5.9	10.7

We consider and evaluate **4 notions of normality** combining both from **same-** and **cross-domain** time series.



NOTABLE IMPACT OF LATENT REPRESENTATION

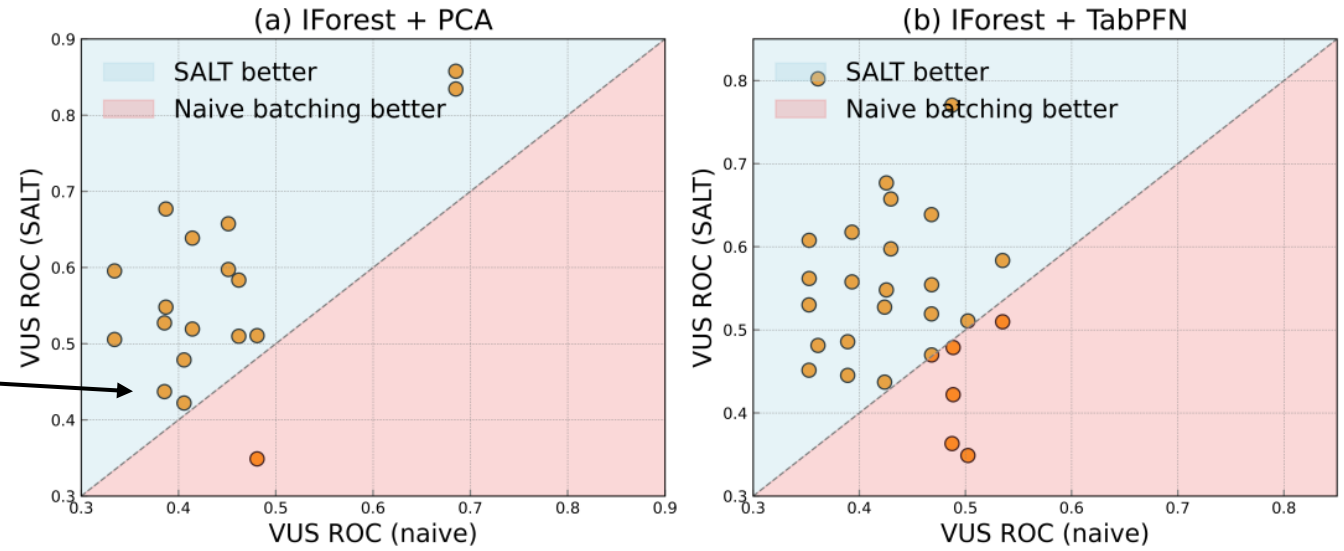
Applying PCA shows that the latent-space separation pays off during the batch AD, with k-Means clustering of smaller *k* values.



NOTABLE IMPACT OF LATENT REPRESENTATION

Applying PCA shows that the latent-space separation pays off during the batch AD, with k-Means clustering of smaller *k* values.

Each point represents another execution for a specific normality notion.

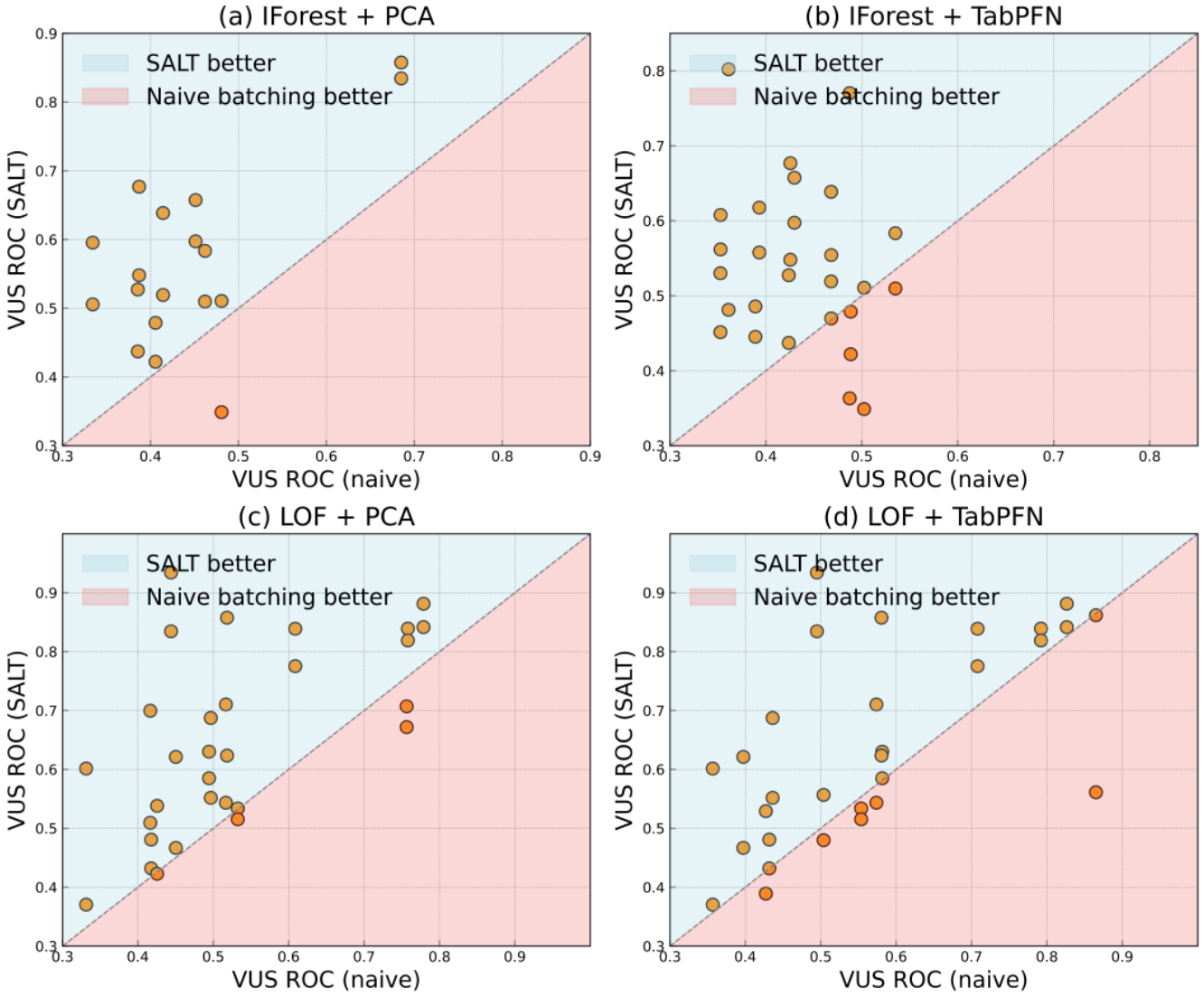


NOTABLE IMPACT OF LATENT REPRESENTATION

Applying PCA shows that the latent-space separation pays off during the batch AD, with k-Means clustering of smaller *k* values.

Each point represents another execution for a specific normality notion.

Both naïve IForest and naïve LOF are often outperformed by SALT in batching mode due to component analysis over the subsequences.



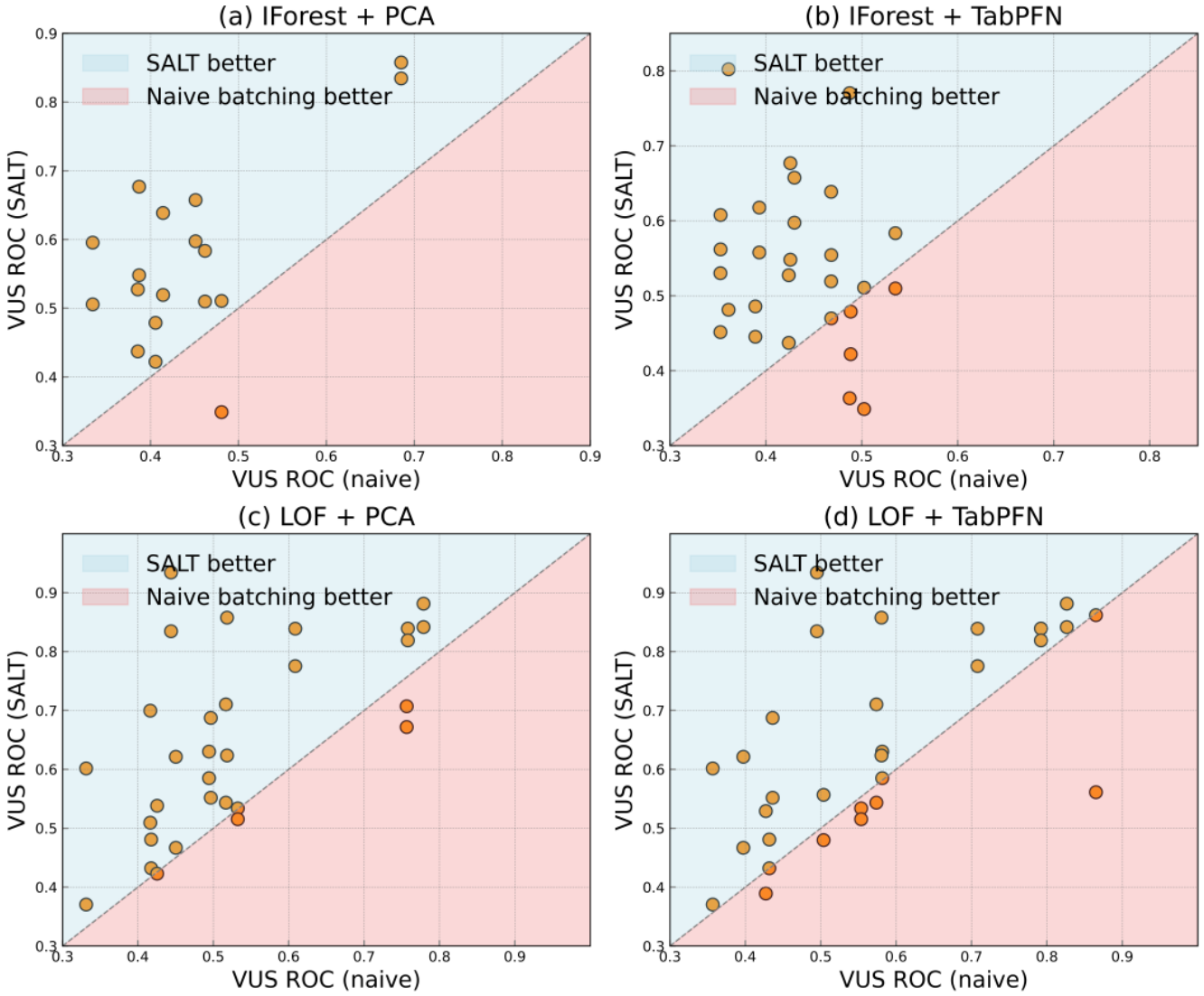
NOTABLE IMPACT OF LATENT REPRESENTATION

Applying PCA shows that the latent-space separation pays off during the batch AD, with k-Means clustering of smaller *k* values.

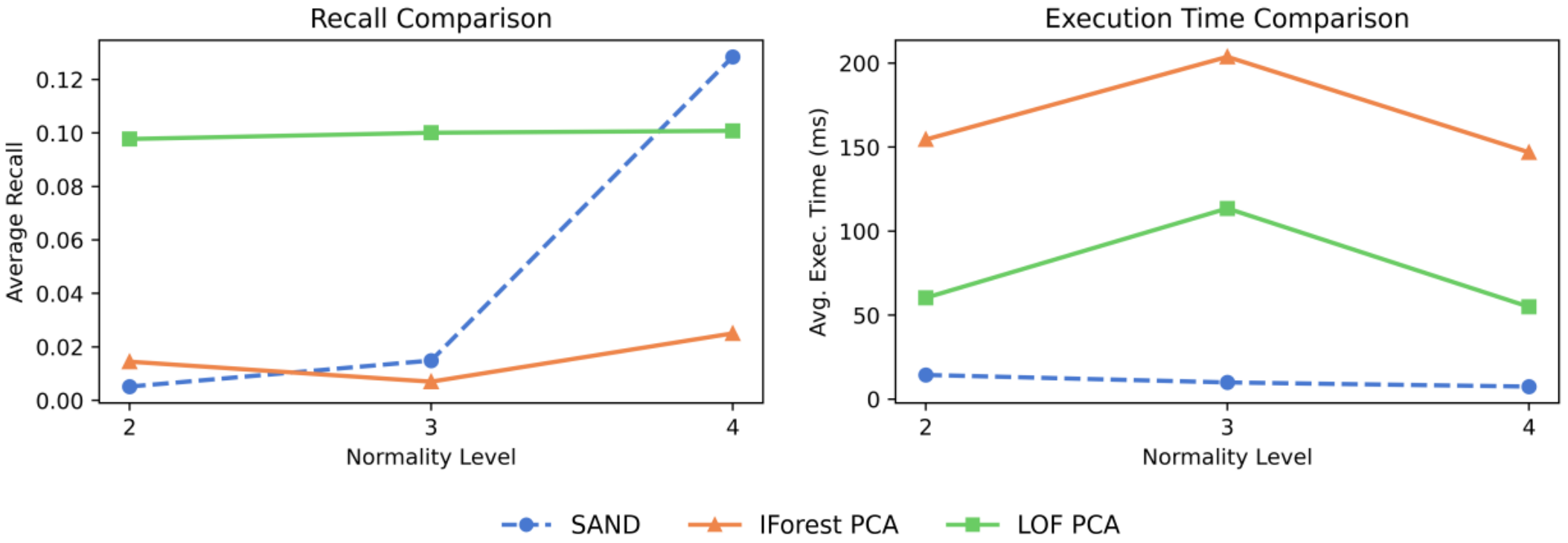
Each point represents another execution for a specific normality notion.

Both naïve IForest and naïve LOF are often outperformed by SALT in batching mode due to component analysis over the subsequences.

TabPFN embeddings seem less beneficial for the clustering step of SALT, compared to PCA.

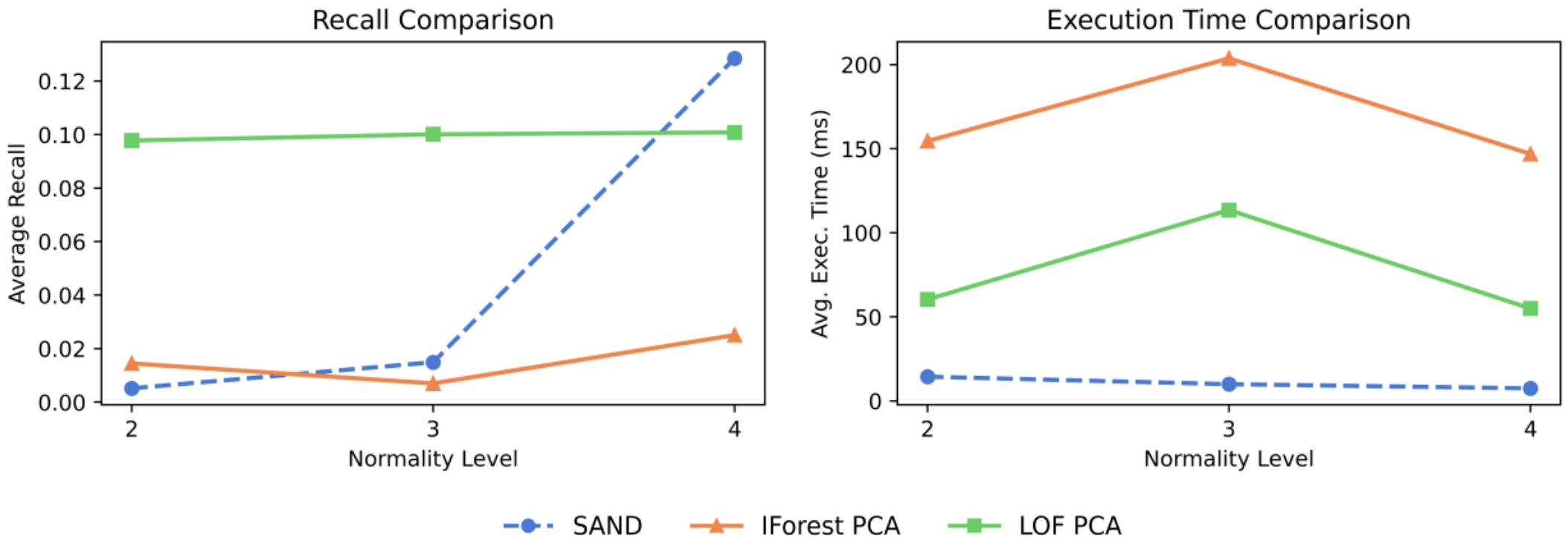


Sensitivity Analysis on Normalities



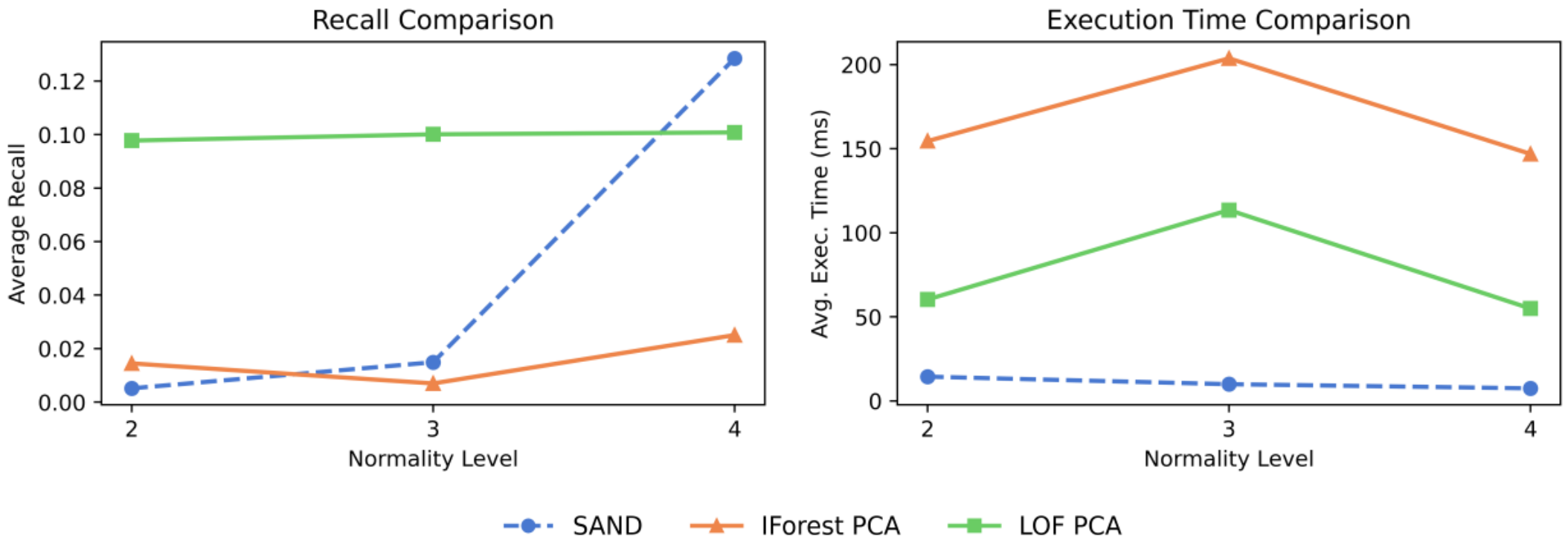
We compare SALT's best models with SAND.

Sensitivity Analysis on Normalities



- SAND seems to perform better in terms of recall and runtime.

Sensitivity Analysis on Normality



- SAND seems to perform better in terms of recall and runtime.
- SALT'S LOF with PCA performs better than its IForest variant.

Thank you :)

Any questions?

SALT

/balaktsis/StreamingAnomalyDetection

ARISTOTLE UNIVERSITY OF THESSALONIKI

DATA&WEB SCIENCE LABORATORY