HUAWEI Database and Data Processing Workshop 2022
16 December 2022, Moscow, Russia

Databases and time series: friends or foes?

情义无价
Friendship has no price.
Chinese proverb

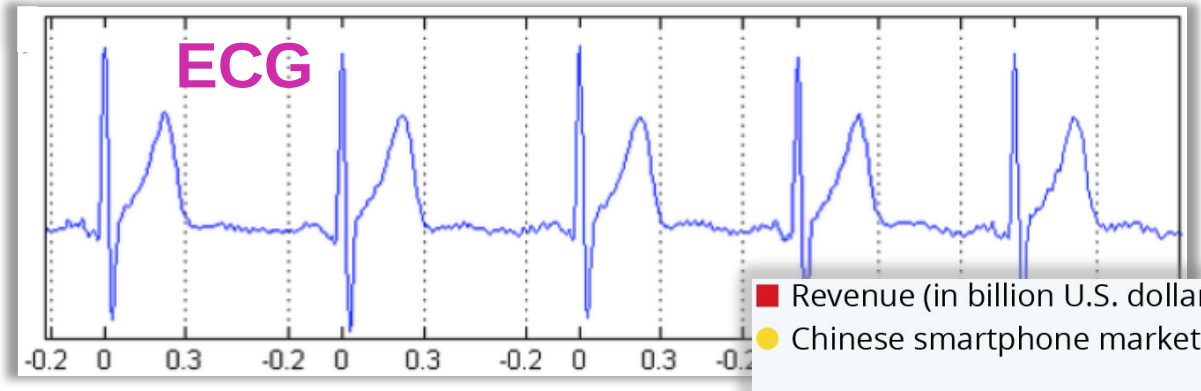
Mikhail Zymbler, Elena Ivanova,
Yana Kraeva, Andrey Goglachev, Alexey Yurtin

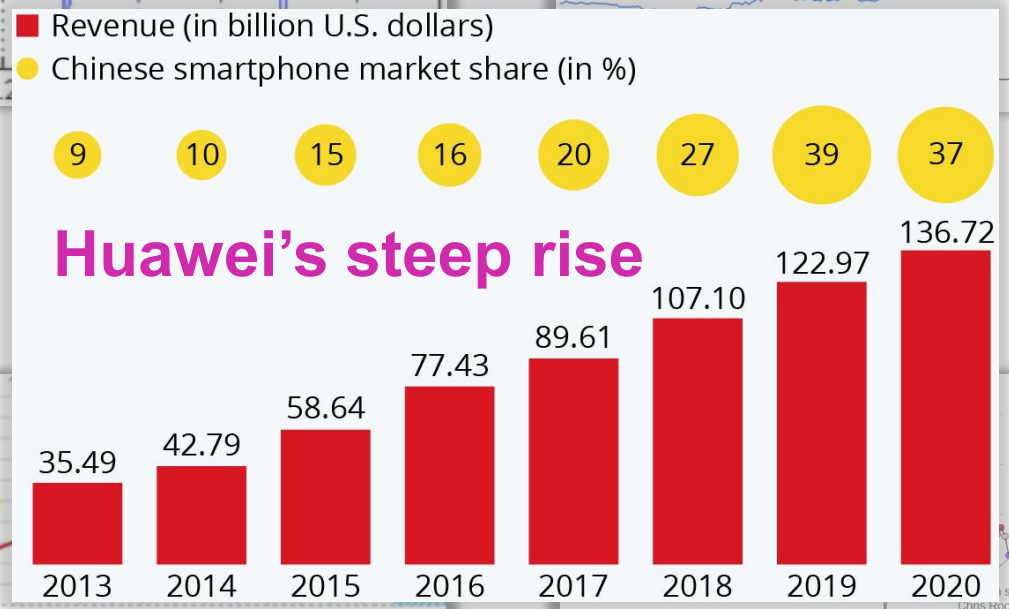
{mzym, elena.ivanova, kraevaya, goglachevai, iurtinaa}@susu.ru

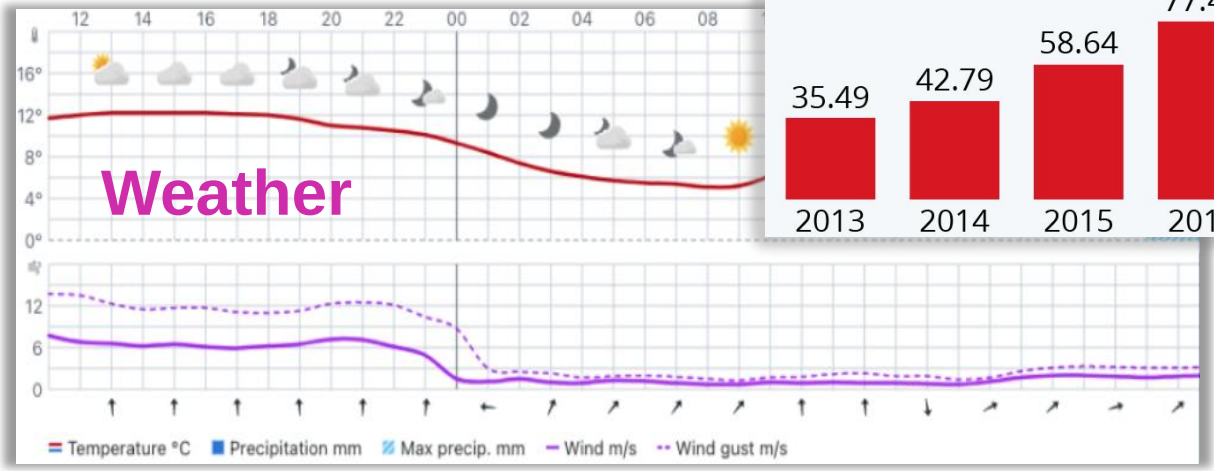
Big Data and Machine Learning Lab, South Ural State University, Chelyabinsk, Russia

This work was financially supported by the Russian Foundation for Basic Research (grant No. 17-07-00463)
and by the Ministry of Science and Higher Education of the Russian Federation (government order FENU-2020-0022)

People like to measure everything over time, …

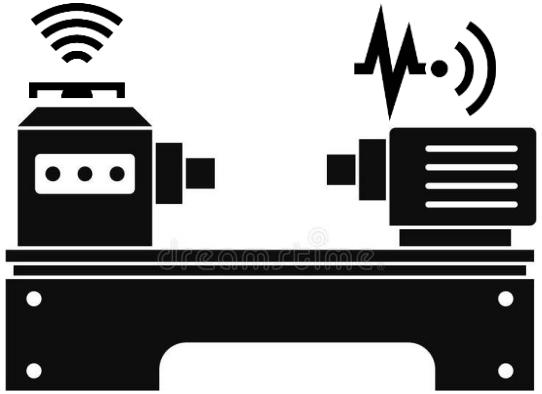




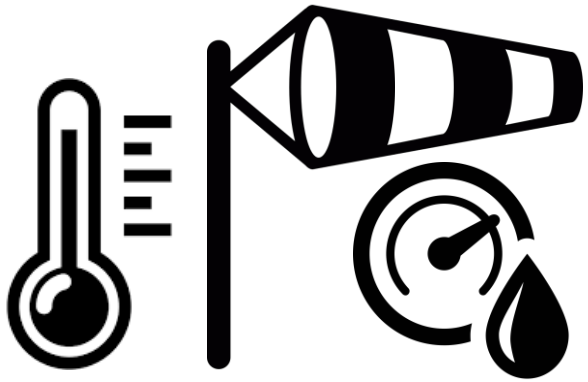


… in every sphere of our life, …

Internet
of Things



Smart manufacturing,
Predictive maintenance



Weather forecasting,
Climate modelling

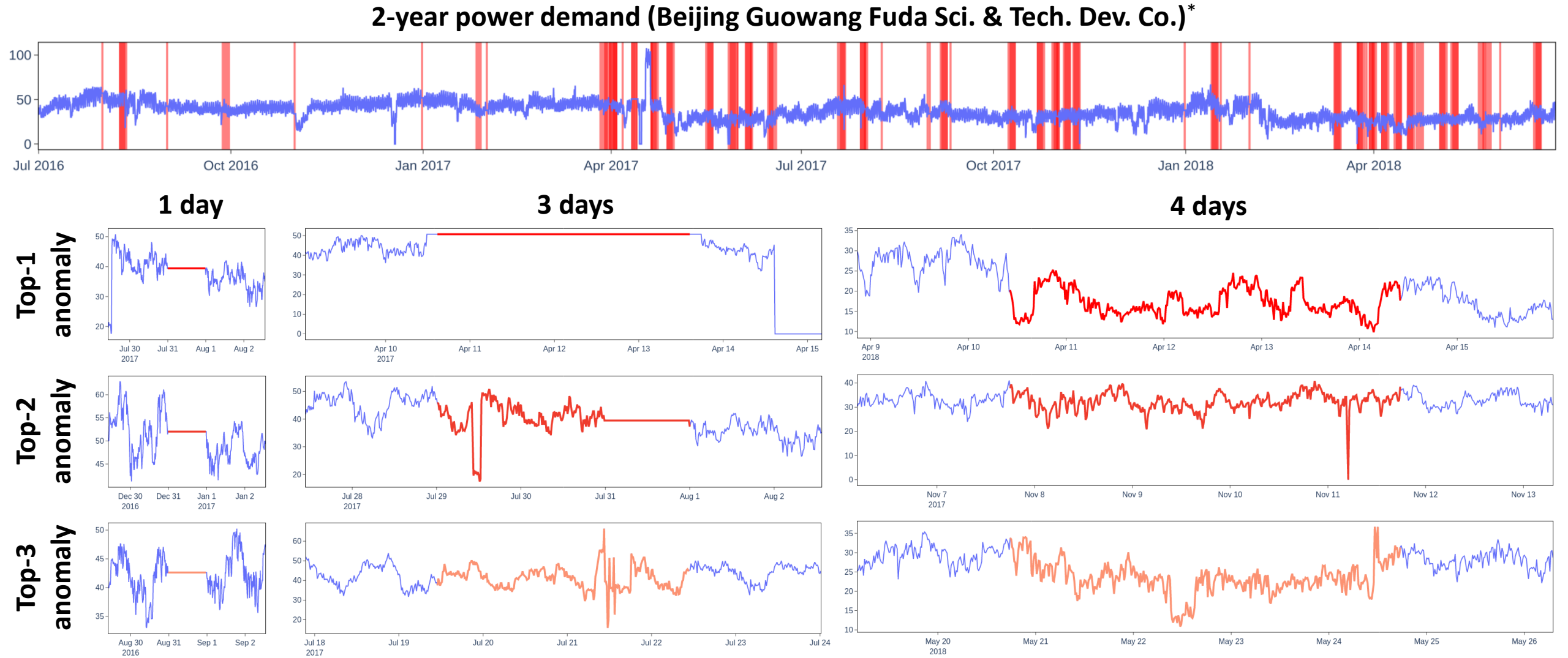
Business
and economics

Bioinformatics,
Cheminformatics

Personal
healthcare

Learning management systems,
Personal educational trajectories

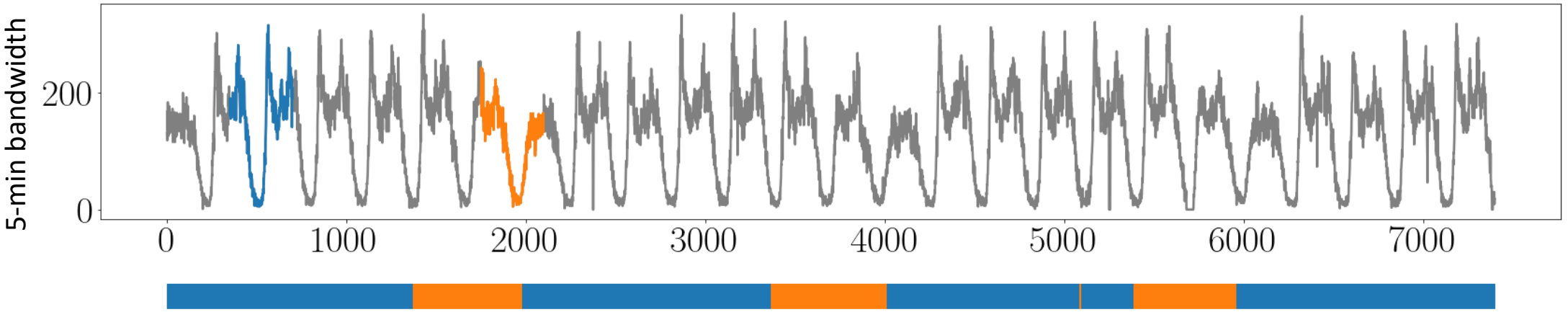
… to get insights from time series, e.g., **anomalies**



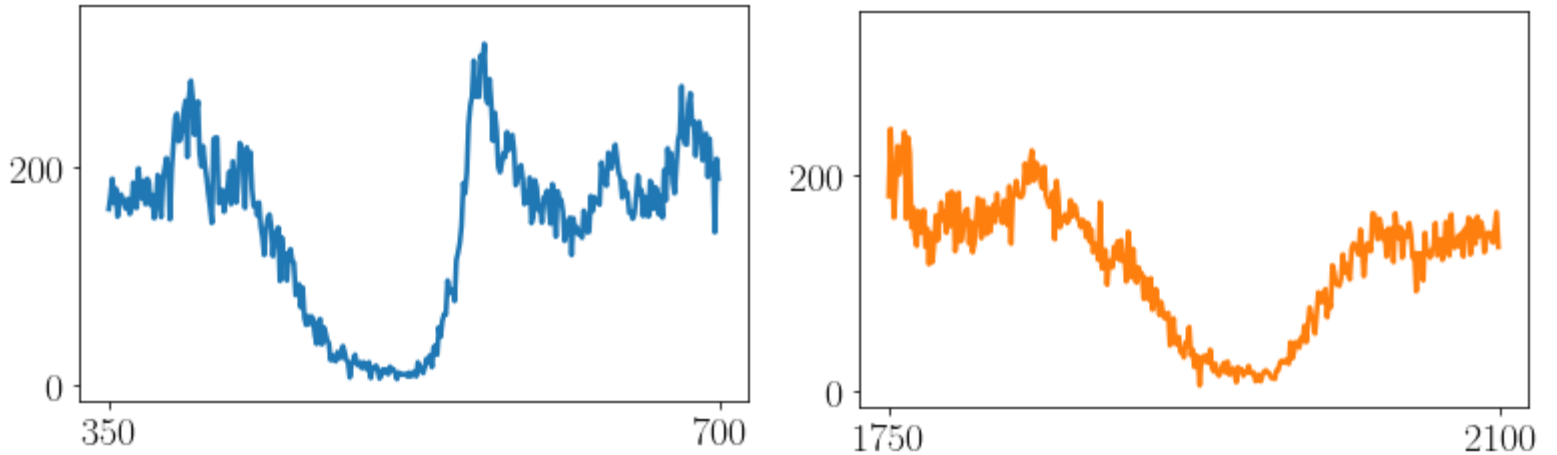
* Zhou H. *et al.* Informer: beyond efficient transformer for long sequence time-series forecasting. AAAI 2021: 11106-11115. DOI: [10.1609/aaai.v35i12.17325](https://doi.org/10.1609/aaai.v35i12.17325).

... to get insights from time series, e.g., **patterns**

One-month urban traffic in Munich (gathered by the Huawei Research Center)*

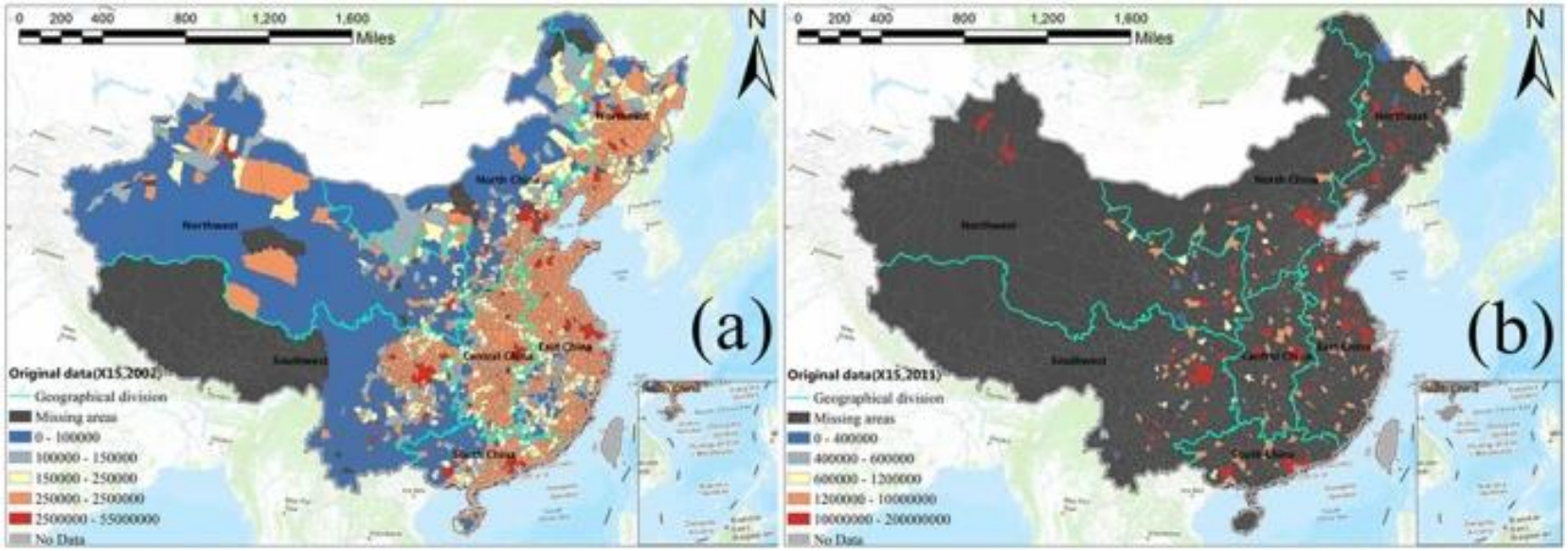


Patterns found



* Public (anonymized) road traffic prediction datasets from Huawei Munich Research Center. URL: https://zenodo.org/record/3653880#.Y0zZi3ZBxPa

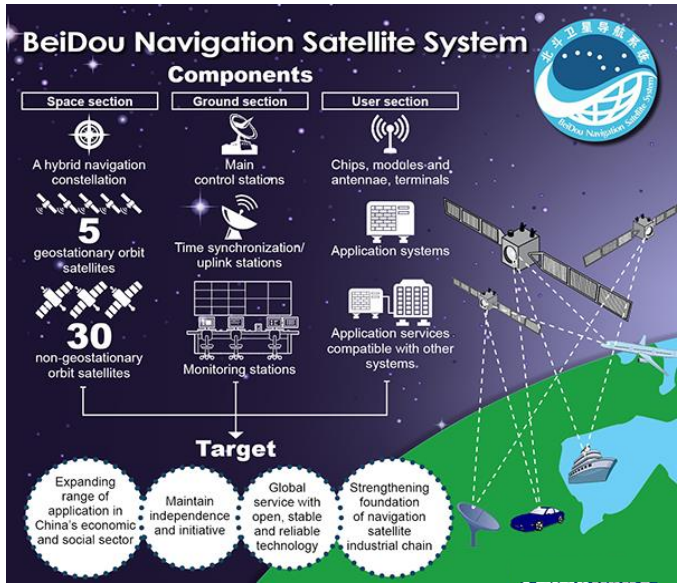
… to get insights from time series, despite **missing values**



China counties with missing official statistical data (one attribute)*

a) 2002: less than 15% data are missing

b) 2011: more than 85% data are missing



BeiDou satellite transmission link suffers the packet loss**

* Song C. *et al.* Estimating missing values in China's official socioeconomic statistics using progressive spatiotemporal Bayesian hierarchical modeling. Sci. Rep. 2018. Vol. 8, article 10055. DOI: 10.1038/s41598-018-28322-z

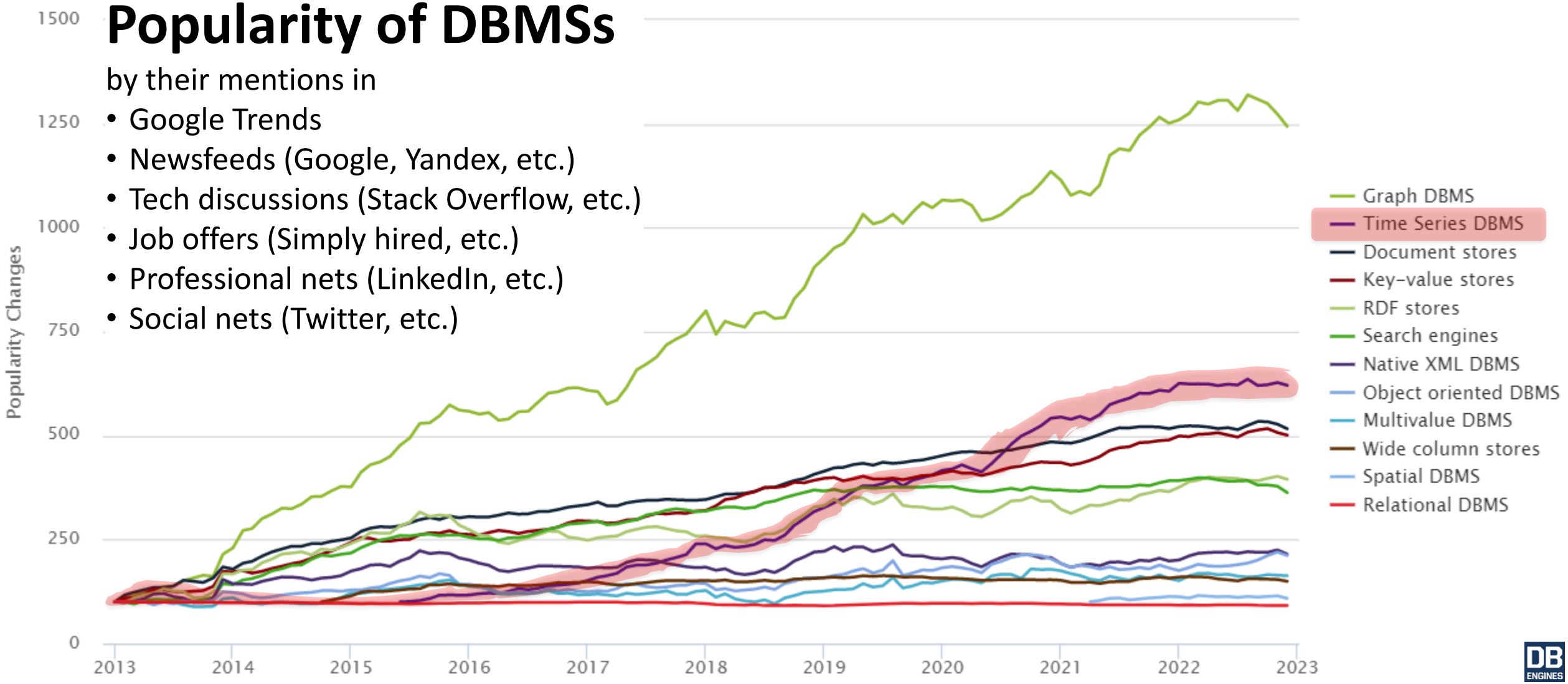
** Liu S. *et al.* A novel BeiDou satellite transmission framework with missing package imputation applied to smart ships. IEEE Sensors Journal. 2022. Vol. 22, no. 13. P. 13162-13176. DOI: 10.1109/JSEN.2022.3177167.

So, we need DBMSs to store time series, …

Popularity of DBMSs

by their mentions in

- Google Trends
- Newsfeeds (Google, Yandex, etc.)
- Tech discussions (Stack Overflow, etc.)
- Job offers (Simply hired, etc.)
- Professional nets (LinkedIn, etc.)
- Social nets (Twitter, etc.)



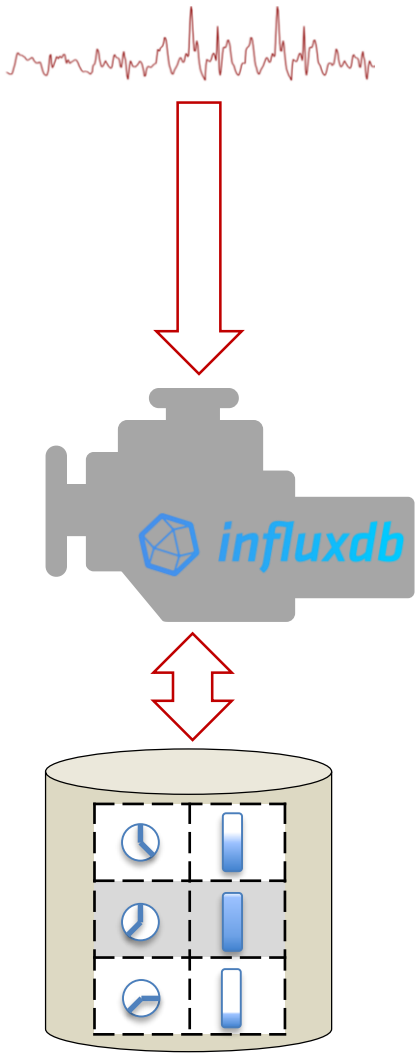
… and, we need systems to analyze time series

Modern Time Series DBMSs

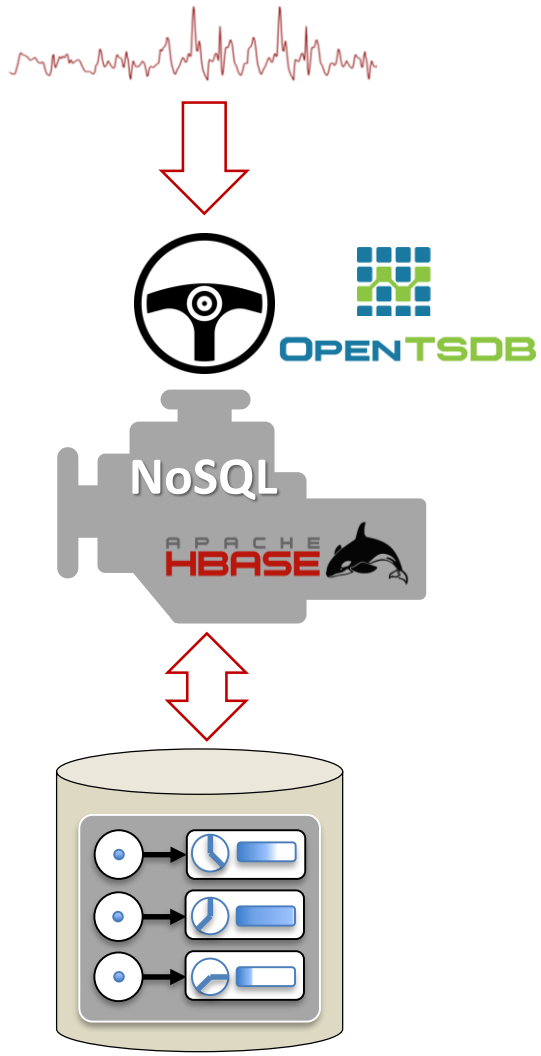
TOP-10 TS-DBMSs

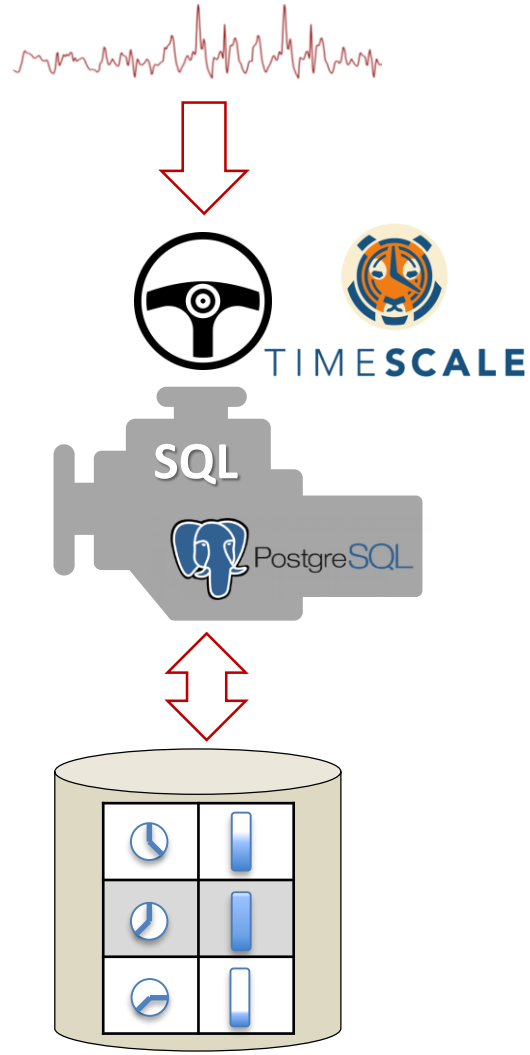
Rank Dec 2022	TS-DBMS
1.	influxdb
2.	Kdb+
3.	Graphite
4.	Prometheus
5.	TIMESCALE
6.	RRDtool
7.	OPENTSDB
8.	DolphinDB
9.	Apache Druid
10.	TDengine

Native TS-DBMSs



Add-on TS-DBMSs

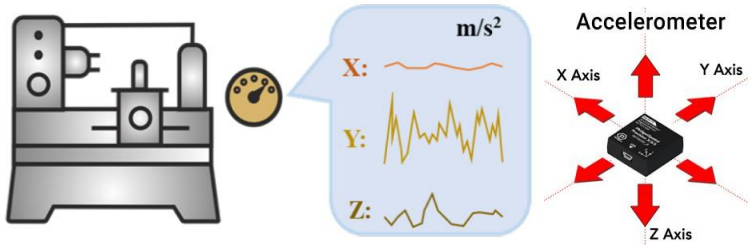




influxdb, native TS-DBMS

Data storage

SensorDB



Measurement: accelerometer

stamp	**val** (field)	**Axis** (tag)
2022-09-15T00:00:00Z	34.7	'x'
2022-09-15T00:00:01Z	5.0	'y'
2022-09-15T00:00:02Z	134.4	'z'
...	...	...

CREATE DATABASE SensorDB; -- Schemaless
INSERT accelerometer, axis='x' val=34.7
INSERT accelerometer, axis='y' val=5.0
INSERT accelerometer, axis='z' val=134.4

Data processing

- InfluxQL is SQL-alike

  SELECT MIN(val)
  FROM accelerometer
  GROUP BY axis

- Continuous query: runs automatically at a specified frequency

  CREATE CONTINUOUS QUERY cq_minimum
  ON SensorDB **BEGIN**
  SELECT MIN(val) **INTO** minX
  FROM accelerometer **WHERE** axis='x'
  GROUP BY time (1 h) **END**
  SELECT * FROM minX

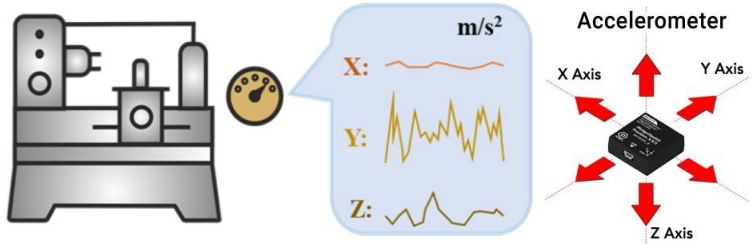
Built-in analytics

- Prediction through the Holt–Winters model

OpenTSDB, NoSQL-based TS-DBMS

Data storage

SensorDB



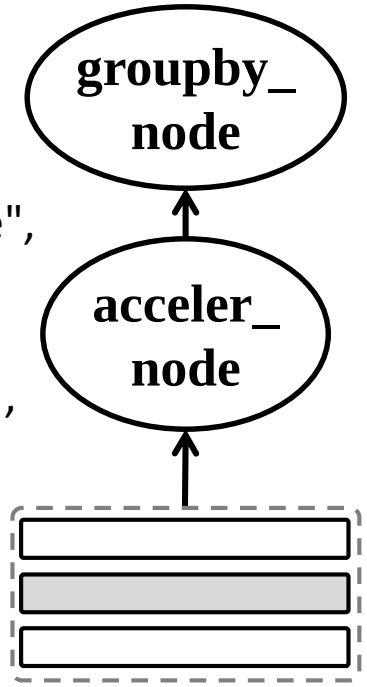
accelerometer	2022-09-15T00:00:00Z	34.7	axis='x'
accelerometer	2022-09-15T00:00:01Z	5.0	axis='y'
accelerometer	2022-09-15T00:00:02Z	134.4	axis='z'
...	...	...	...

env ./src/create_table.sh; -- Create structures for data storage
put accelerometer 2022-09-15T00:00:00Z 34.7 axis='x'
put accelerometer 2022-09-15T00:00:01Z 5.0 axis='y'
put accelerometer 2022-09-15T00:00:02Z 134.4 axis='z'

Data processing

- JSON queries (graphs):

```
{ "start": "1h-ago " ,
  "executionGraph": [
    { "id": "acceler_node",
      "type": "TimeSeriesDataSource",
      "metric": {
        "type": "MetricLiteral",
        "metric": "accelerometer" } } ,
    { "id": "groupby_node",
      "type": "groupby",
      "aggregator": "min",
      "tagKeys": ["axis"] ,
      "sources": ["acceler_node"]
    } ] }
```



Built-in analytics

- Down-sampling, interpolation

TIMESCALE, SQL-based TS-DBMS

Data storage

SensorDB

Hypertable: accelerometer

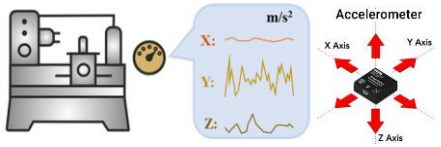


Table **x_January**

stamp	val
2022-01-01 00:00:00	4.7
…	

Table **x_January**

stamp	val
2022-01-01 00:00:01	52.1
…	

Table **z_January**

stamp	val
2022-01-01 00:00:03	34.0
…	

Table **x_December**

stamp	val
2022-12-15 00:00:00	34.7
…	

Table **y_December**

stamp	val
2022-12-15 00:00:01	5.0
…	

Table **z_December**

stamp	val
2022-12-15 00:00:02	134.4
…	

Data processing

- Full SQL compatibility

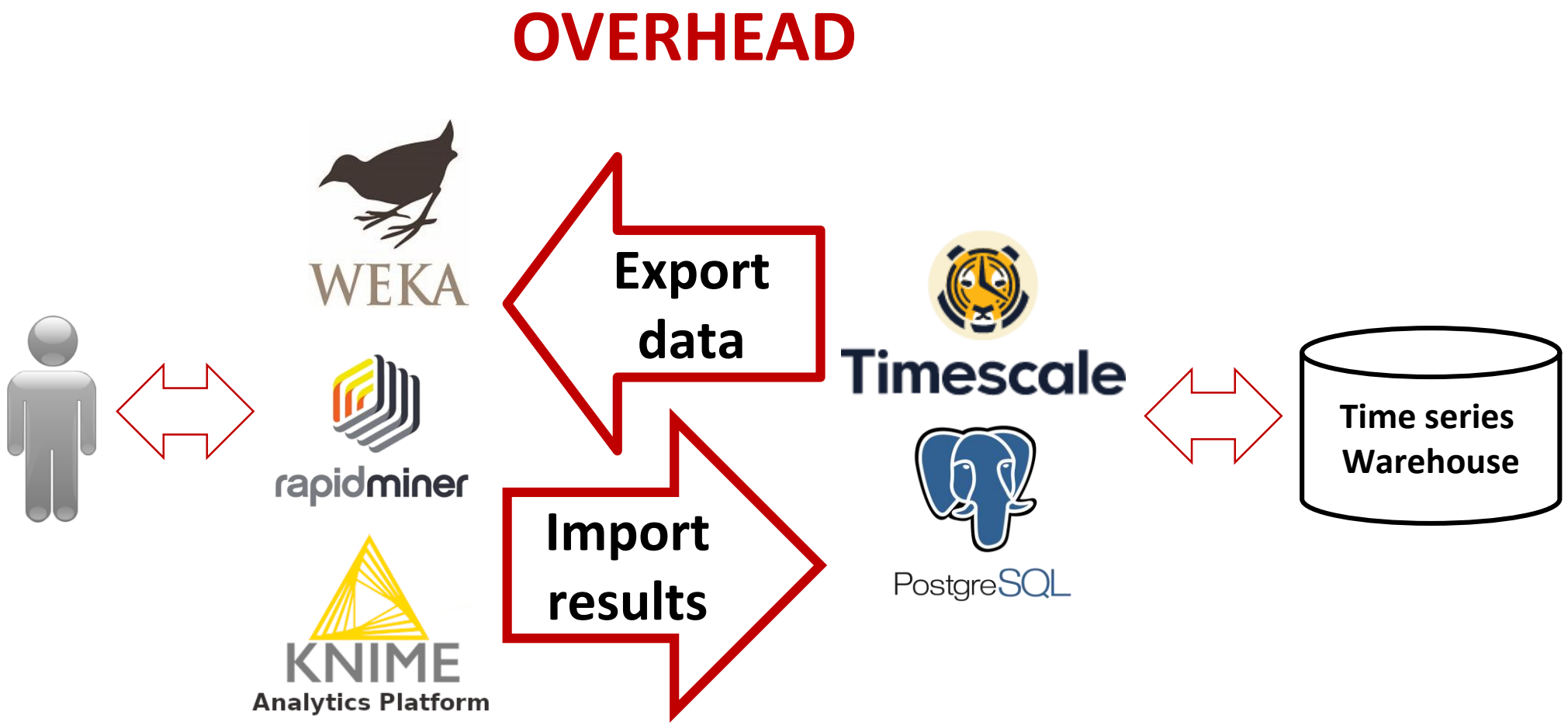
Built-in analytics

- Third-party libraries (e.g., Apache MADlib*)

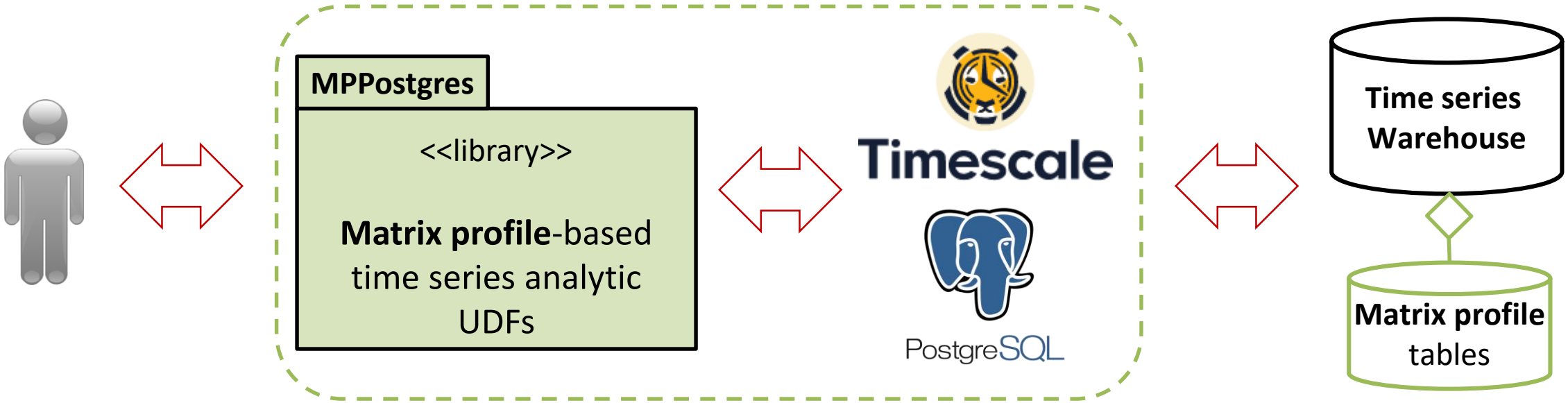
CREATE DATABASE SensorsDB; **CREATE EXTENSION** TimescaleDB
CREATE TABLE accelerometer(stamp **TIMESTAMP**, val **REAL**, axis **CHAR**)
SELECT create_hypertable ('accelerometer', 'stamp', 'axis', 3, chunk_time_interval => **INTERVAL** '1 month')
INSERT INTO accelerometer **VALUES** (NOW(), 34.7, 'x');
INSERT INTO accelerometer **VALUES** (NOW(), 5.0, 'y')

* Hellerstein J.M. *et al*. MADlib analytics library or MAD Skills, the SQL. Proc. VLDB Endow. 2012. DOI: 10.14778/2367502.2367510.

What's wrong with time series analytics?

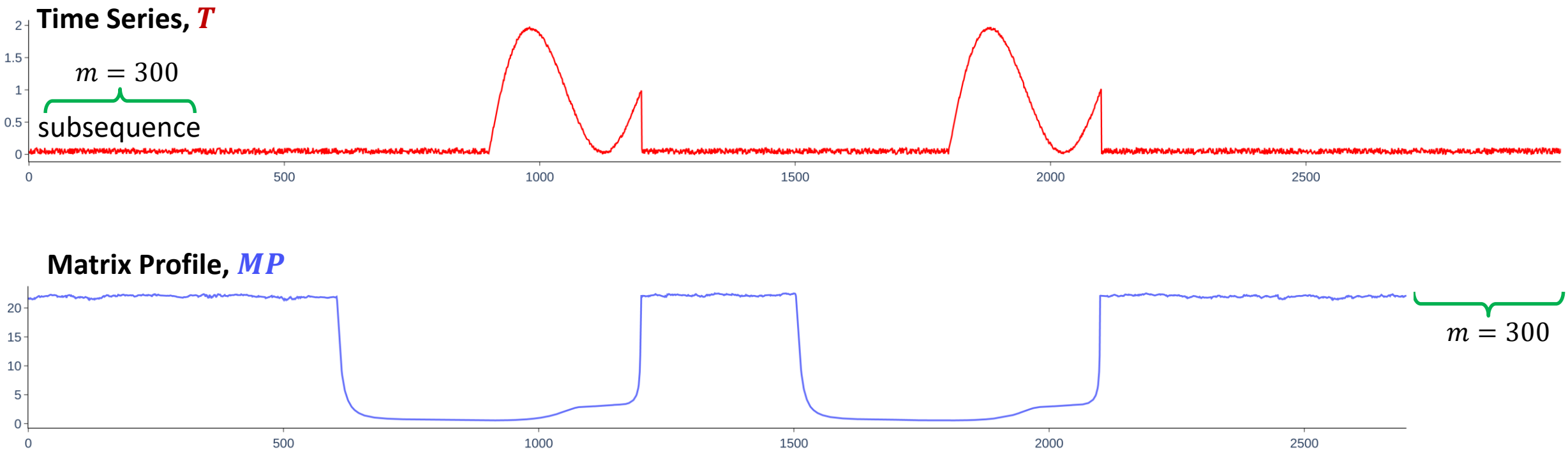


Can we analyze time series **inside** a relational DBMS?



骑驴找马
Looking for a horse, ride a donkey.
Chinese proverb

Matrix profile* of a time series

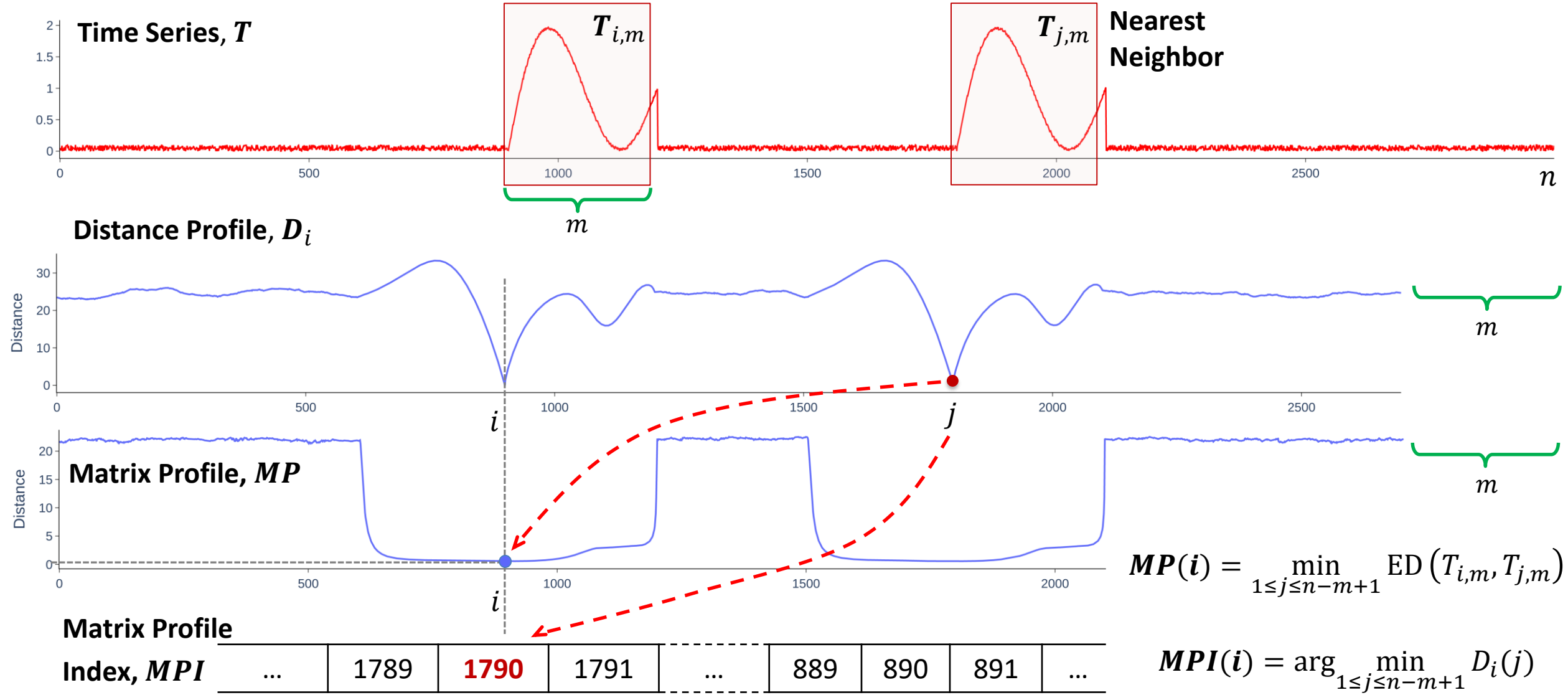


$$MP(i) = \min_{1 \le j \le n-m+1} \mathbf{ED}\left(T_{i,m}, T_{j,m}\right)$$

Matrix profile's i-th element is the **Euclidean distance** between i-th subsequence and its nearest non-overlapping neighbor

* Yeh C.M. *et al.* Time series joins, motifs, discords and shapelets: A unifying view that exploits the matrix prole. Data Min. Knowl. Discov. 32(1), 83-123 (2018). DOI: 10.1007/s10618-017-0519-9
* Zimmerman Z. *et al.* Matrix Profile XIV: Scaling time series motif discovery with GPUs to break a quintillion pairwise comparisons a day and beyond. SoCC 2019. DOI: 10.1145/3357223.3362721

Matrix profile of a time series

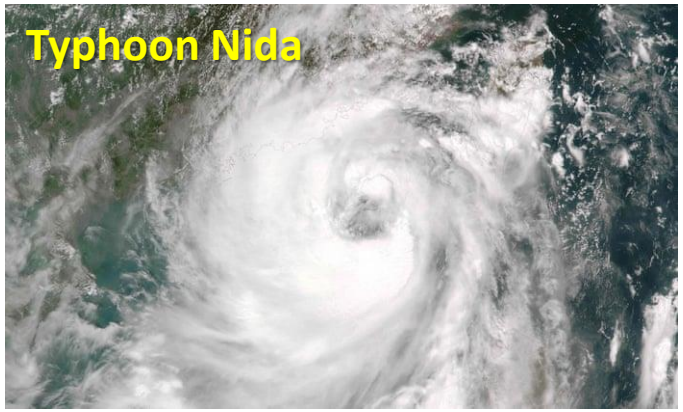


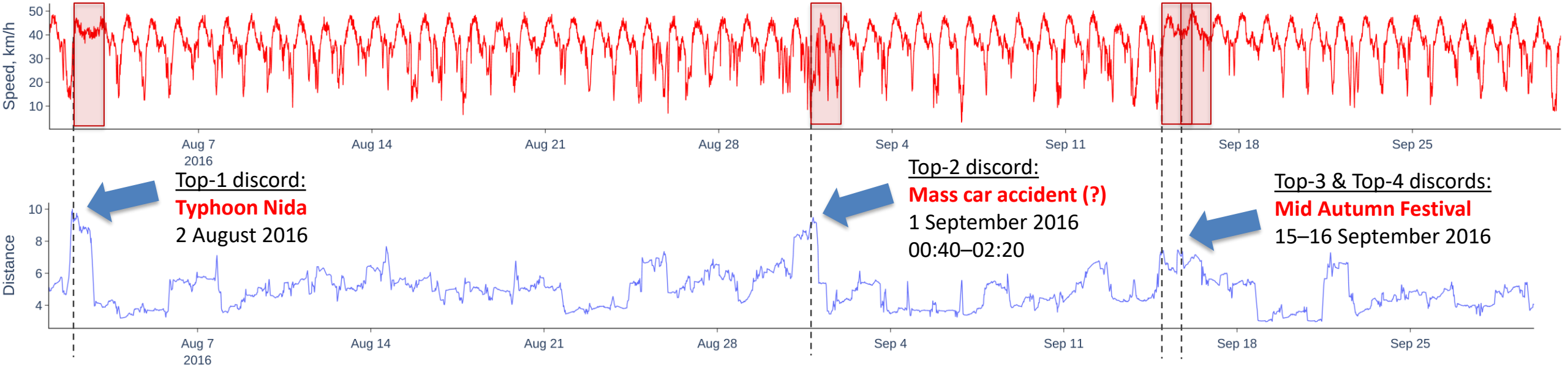
$$\boldsymbol{MP}(\boldsymbol{i}) = \min_{1 \le j \le n-m+1} \mathrm{ED}\left(T_{i,m}, T_{j,m}\right)$$

$$\boldsymbol{MPI}(\boldsymbol{i}) = \arg\min_{1 \le j \le n-m+1} D_i(j)$$

Insights from the Matrix profile: Discords

Urban traffic speed in Guangzhou*

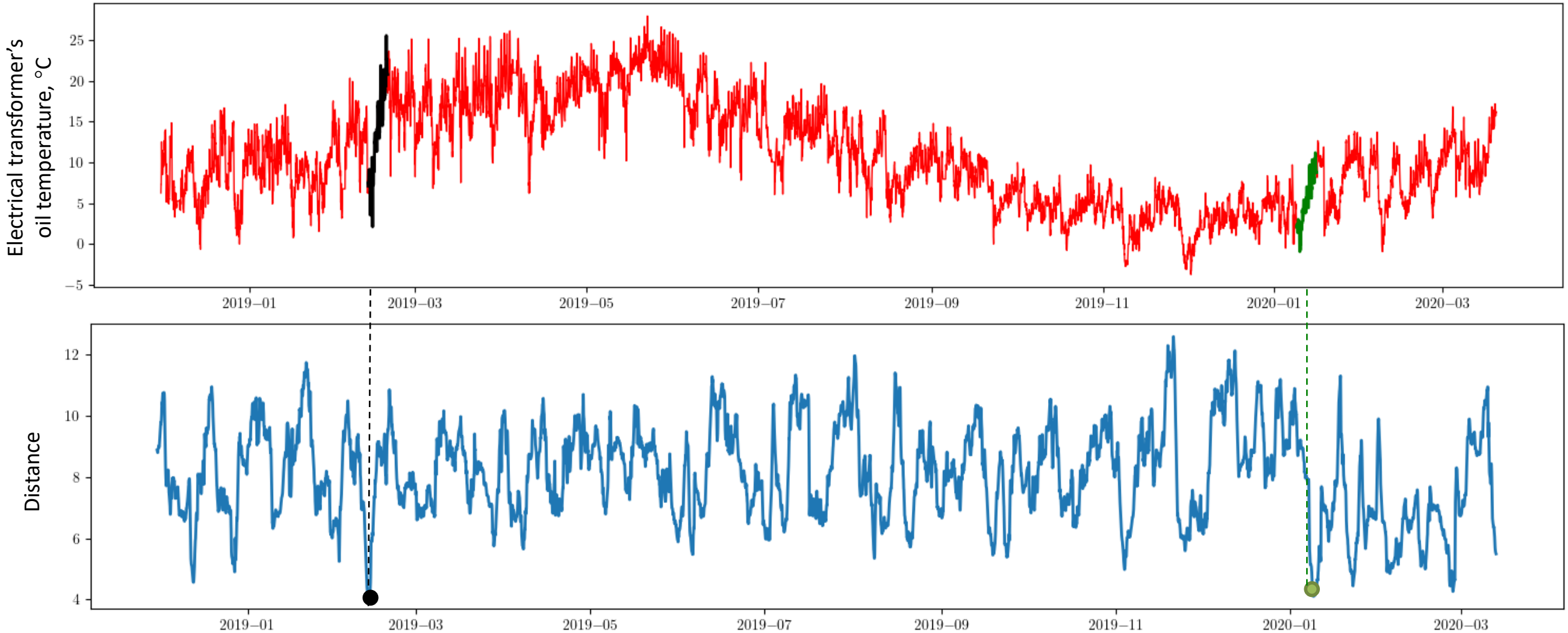




* Chen X, Chen Y, He Z. Urban traffic speed dataset of Guangzhou, China. 2018. DOI: 10.5281/zenodo.1205229.

Insights from the Matrix profile: Motifs

2-year power demand (Beijing Guowang Fuda Sci. & Tech. Dev. Co.)*



* Zhou H. *et al.* Informer: beyond efficient transformer for long sequence time-series forecasting. AAAI 2021: 11106-11115. DOI: 10.1609/aaai.v35i12.17325.

MPPostgres: Matrix profile-oriented warehouse

Time series data

Time Series Directory	
ID	BIGSERIAL
name	TEXT
len	BIGINT

ts_name (a time series)	
num	BIGSERIAL
stamp	TIMESTAMP
val	REAL

Matrix Profile data

Matrix Profile Directory	
ID	BIGSERIAL
tsID*	BIGINT
subseqLen	INT

mp_name_subseqLen (a matrix profile)	
num	BIGSERIAL
nnDist	REAL
nnIdx	BIGINT

Time series analytics API

discoverDiscords (**name** TEXT, **subseqLen** INT)

nnDist	REAL
idx	BIGINT
discord	REAL[subseqLen]

discover … (**name** TEXT, **subseqLen** INT)

…	…

…

discoverMotifs (**name** TEXT, **subseqLen** INT)

nnDist	REAL
idxLeft	BIGINT
motifLeft	REAL[subseqLen]
idxRight	BIGINT
motifRight	REAL[subseqLen]

matrixProfile (**name** TEXT, **subseqLen** INT)

num	BIGSERIAL
nnDist	REAL
nnIdx	BIGINT

MPPostgres: Time series representation

TIME SERIES DATA

Time Series Directory	
ID	BIGSERIAL
name	TEXT
len	BIGINT

ts_name (a time series)	
num	BIGSERIAL
stamp	TIMESTAMP
val1	REAL

Time Series Directory

ID	name	len
1	DailyExchangeRate	5000
2	MinutelyPulse	9000

ts_DailyExchangeRate

num	stamp	val
1	10.02.2008	23.35
2	11.02.2008	24.05
…	…	…
5000	19.10.2022	63.63

ts_MinutelyPulse

num	stamp	val
1	10.10.2022 00:01	135
2	10.10.2022 00:02	105
…	…	…
9000	16.10.2022 06:00	139

MPPostgres: Matrix profile representation

MATRIX PROFILE DATA

Matrix Profile Directory	
ID	BIGSERIAL
tsID*	BIGINT
subseqLen	INT

mp_name_subseqLen (Matrix Profile)	
num	BIGSERIAL
nnDist	REAL
nnIdx	BIGINT

mp_DailyExchangeRate_7

num	nnDist	nnIdx
1	0.02	2071
2	0.05	2078
…	…	…
4994	12.22	184

mp_DailyExchangeRate_30

num	nnDist	nnIdx
1	346.32	3546
2	358.15	4278
…	…	…
4971	0.10	1286

Time Series Directory

ID	name	len
1	DailyExchangeRate	5000
2	MinutelyPulse	9000

Matrix Profile Directory

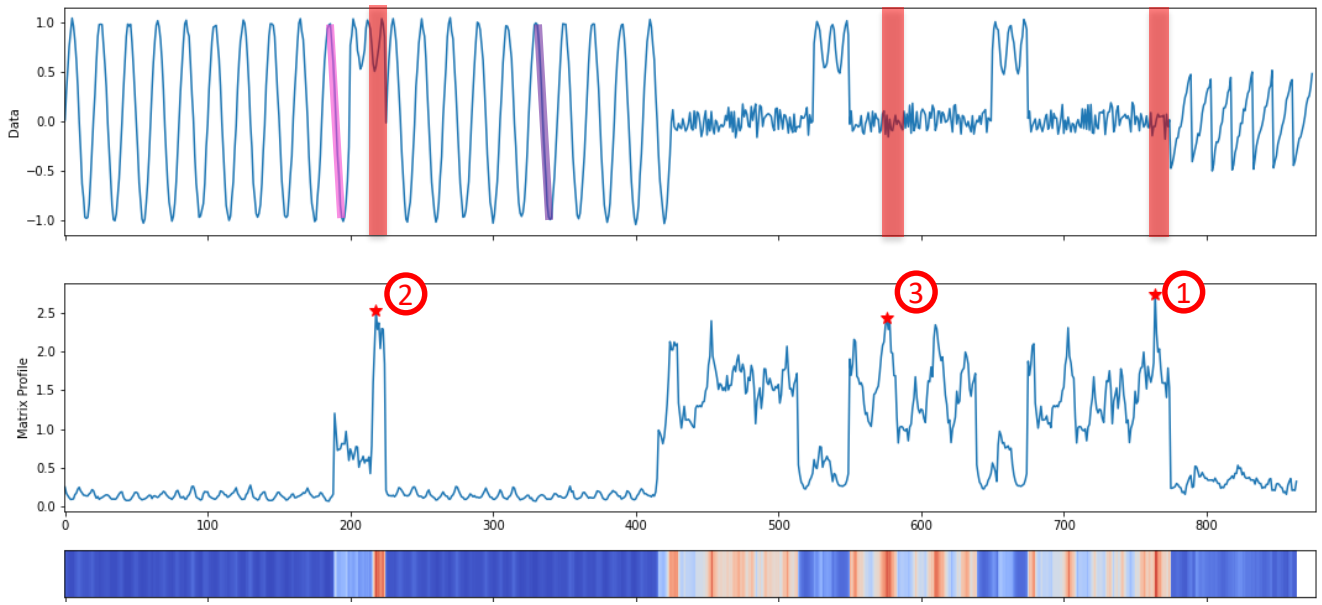
ID	tsID	subseqLen	
1	1	7	weekly
2	1	30	monthly
3	2	60	hourly

mp_MinutelyPulse_60

num	nnDist	nnIdx
1	23.04	145
2	39.78	7658
…	…	…
8941	11.65	7856

MPPostgres: Time series analytics API

SELECT matrixProfile (sensorData, 12);



SELECT discoverDiscords (sensorData, 12, 3);

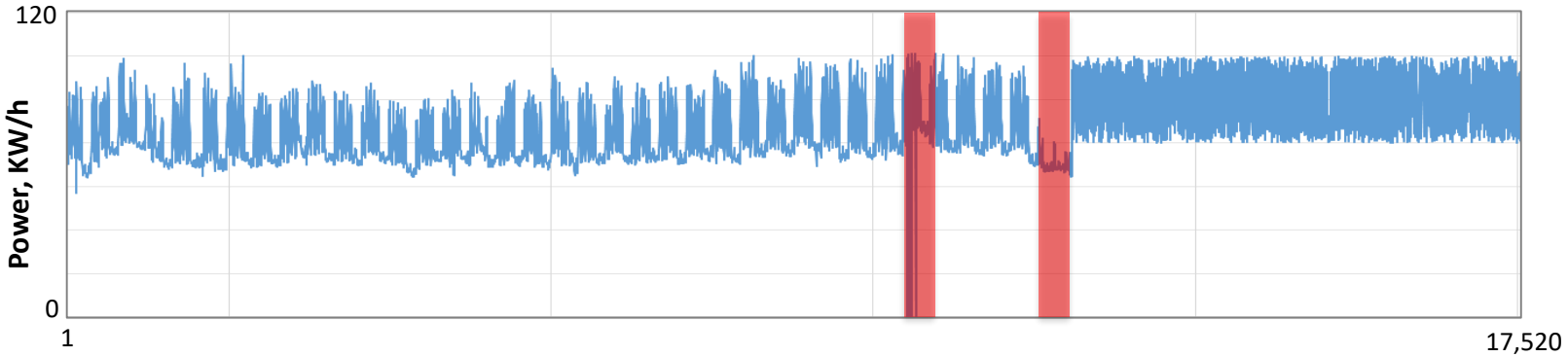
NNDIST	IDX	DISCORD
2.75 ①	764	[-0.05, 0.02, 0.06, 0.04, 0.04, 0.04, 0.05, -0.18, 0.06, -0.05, 0.01, -0.47]
2.53 ②	218	[0.55, 0.64, 0.74, 0.87, 1.03, 0.98, 0.85, -0.01, 0.35, 0.61, 0.81, 0.96]
2.43 ③	576	[0.05, -0.16, 0.01, -0.08, -0.09, -0.02, 0.05, -0.03, 0.01, -0.01, -0.06, -0.08]

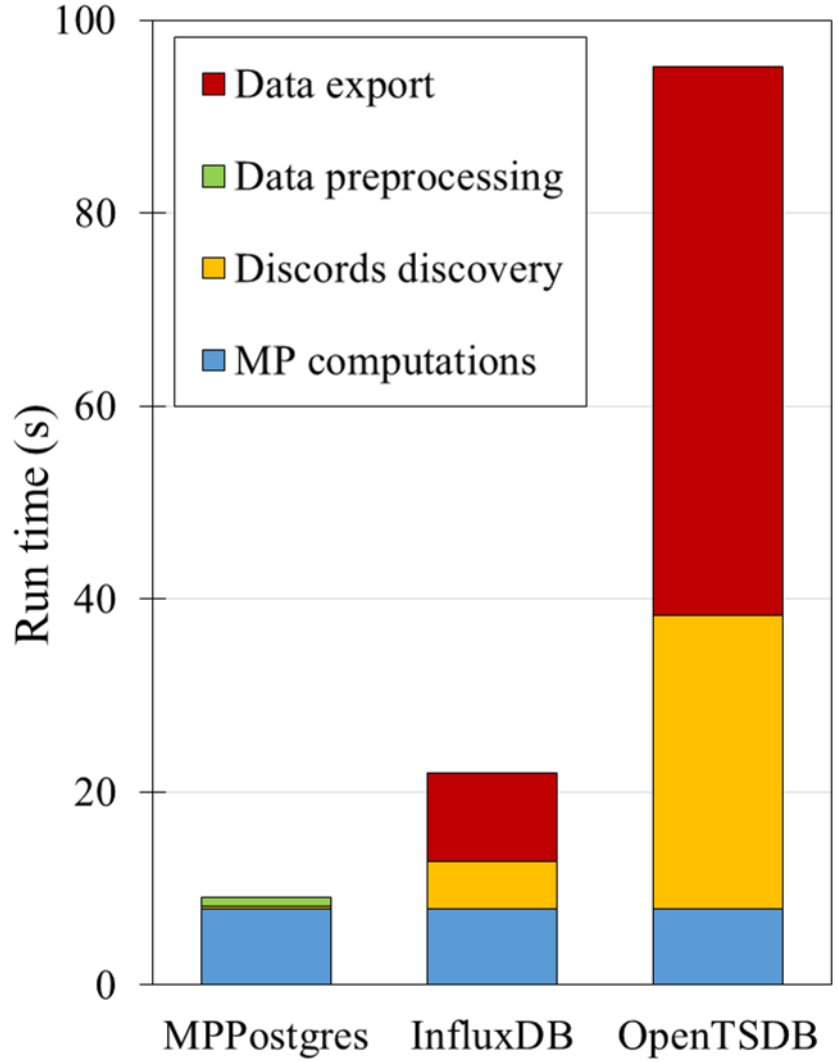
SELECT discoverMotifs (sensorData, 12, 1);

NNDIST	IDXLEFT	MOTIFLEFT	IDXRIGHT	MOTIFRIGHT
0.069	185	[0.97,0.98,0.79,0.59,0.31,-0.01, -0.34,-0.54,-0.85,-0.97,-1.01,-0.96]	330	[0.99,0.97,0.82,0.60,0.32,-0.02, -0.32,-0.56,-0.84,-0.97,-0.99,-0.98]

MPPostgres, case 1: Detection of active power consumption*

- **Data:** power consumption in a campus buildings (Zurich, Switzerland)
- **Goal:** determine the periods of active power consumption through top discords discovery

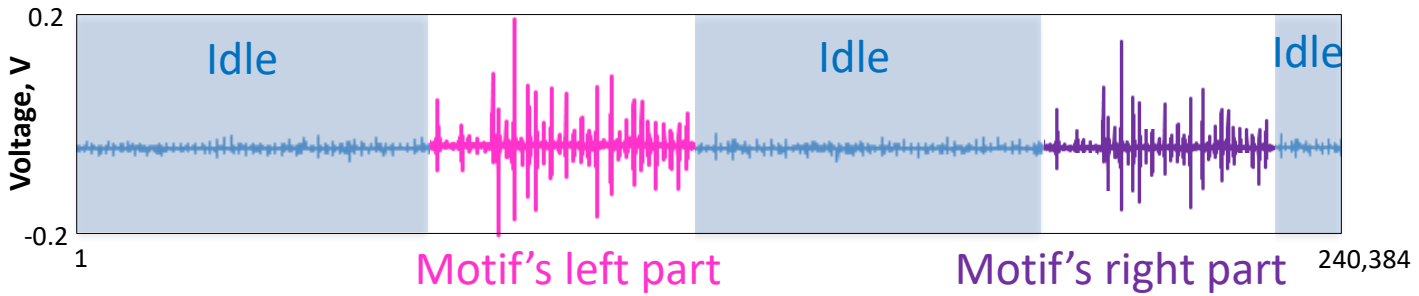


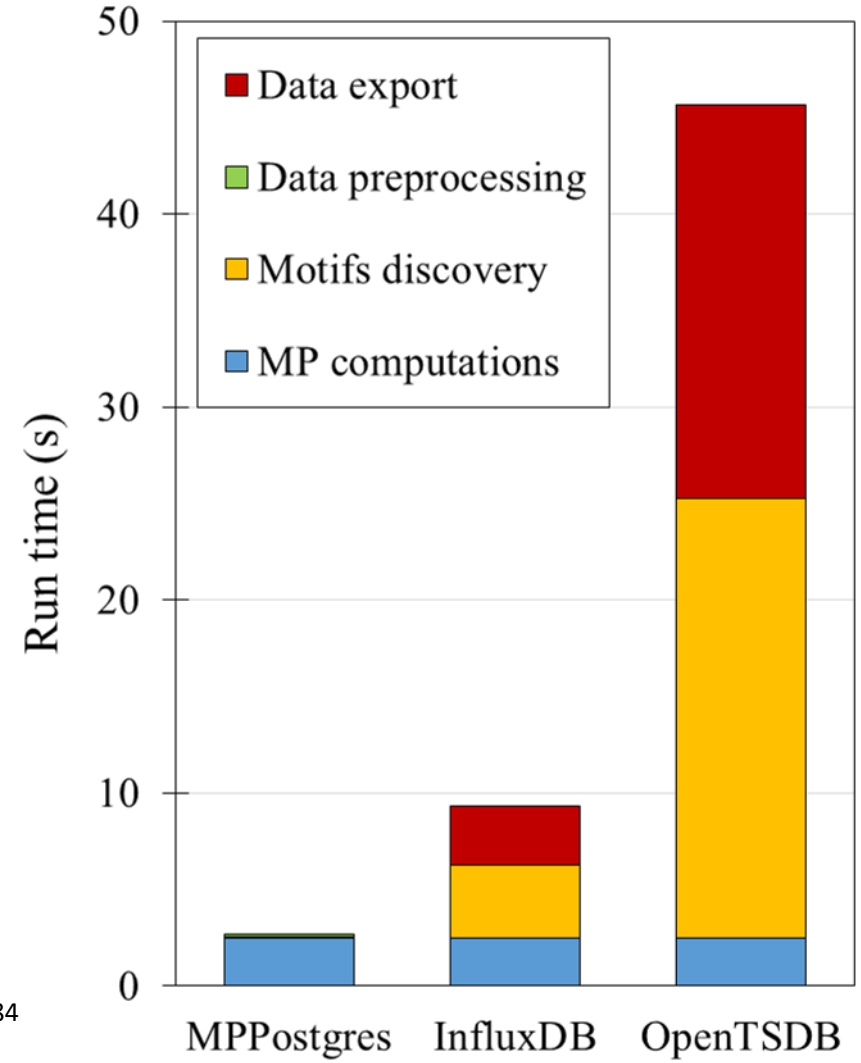


* Nichiforov C. *et al*. Information extraction approach for energy time series modelling. ICSTCC 2020. DOI: 10.1109/ICSTCC50638.2020.9259635.

MPPostgres, case 2: Tracking an industrial machine's status*

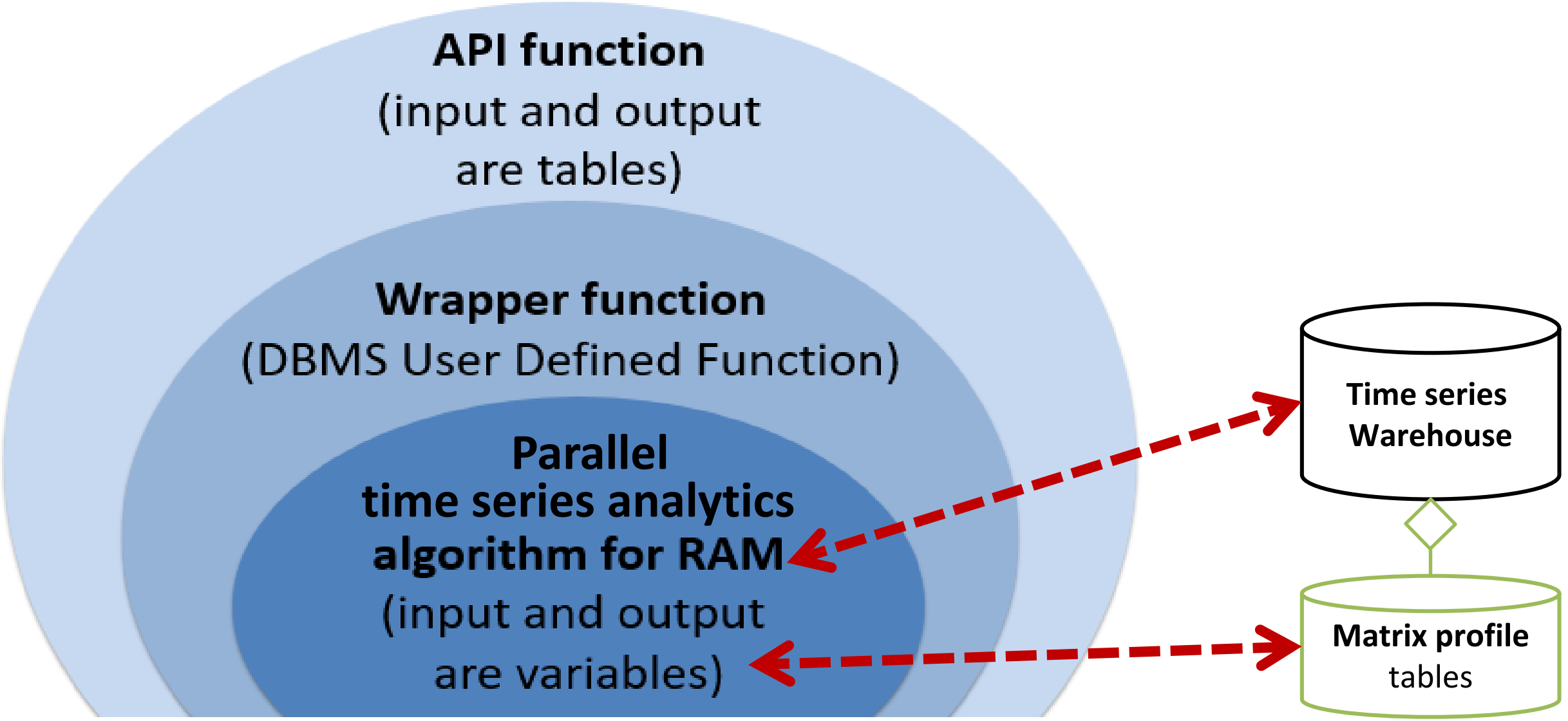
- **Data:** readings of a vibration sensor installed on a crushing machine
- **Goal:** determine the total duration of idle vs. action periods (to further predict the machine's residual life) through summing length of gaps between left and right parts of motifs





* Zymbler M. *et al*. Matrix profile-based approach to industrial sensor data analysis inside RDBMS. Mathematics. 9(17), 2146 (2021). DOI: 10.3390/math9172146.

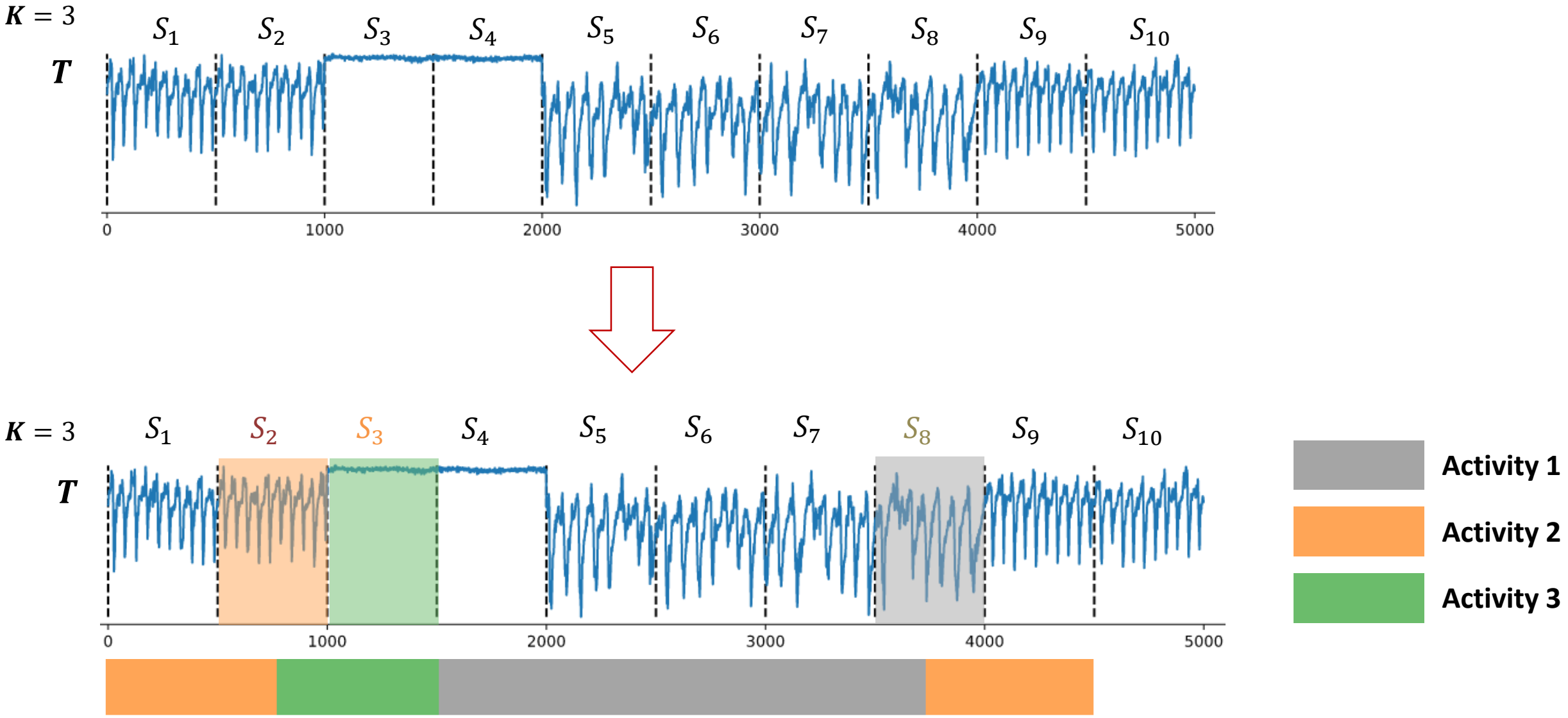
Can TS-DBMSs and parallel algorithms be friends?



Our developments: parallel time series analytics

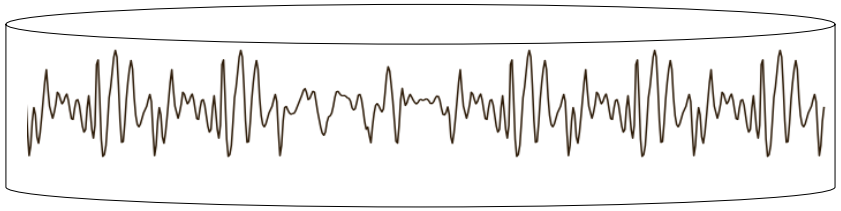
- **Parallel pattern discovery**
  - Zymbler M. *et al.* Fast summarization of long time series with graphics processor. Mathematics. 10(10), 1781 (2022). DOI: 10.3390/math10101781
- **Parallel anomaly discovery**
  - ***Many-core:*** Zymbler M. *et al.* A parallel approach to discords discovery in massive time series data. CMC. 66(2), 1867-1876 (2021). DOI: 10.32604/cmc.2020.014232
  - ***HPC cluster:*** Zymbler M. *et al.* Time series discord discovery on Intel many-core systems. PCT 2019. CCIS, Springer. 1063, 168-182 (2019). DOI: 10.1007/978-3-030-28163-2_12
- **Parallel motif discovery**
  - ***Intel:*** Zymbler M. *et al.* Discovery of time series motifs on Intel many-core systems. LJM. 40(12), 2124-2132 (2019). DOI: 10.1134/S199508021912014X
  - ***GPU:*** Zymbler M. *et al.* Parallel algorithm for time series motif discovery on graphics processor. SUSU Bulletin, CMSE. 9(3), 17-34 (2020). DOI: 10.14529/cmse200302
- **Parallel subsequence matching**
  - Zymbler M. *et al.* Scalable algorithm for subsequence similarity search in very large time series data on cluster of Phi KNL. DAMDID 2018. CCIS, Springer. 1003, 149-164 (2019). DOI: 10.1007/978-3-030-23584-0_9

Parallel pattern discovery



Our developments: ANN model for online anomaly detection

Representative fragment of time series



Parallel Labeling

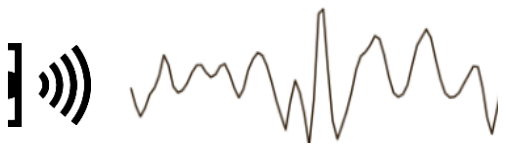
Discovers snippets

Parallel Discord Discovery

Removes anomalies

Training set

Potentially anomalous subsequence



Anomaly detector ANN

Norm

OR

Anomaly

Can TS-DBMSs and ANNs be friends?

ts_SensorData

num	stamp	val
1	15.09.2020	86.4
...	...	...
366	15.09.2021	85.57
...	...	...
732	15.09.2022	**NULL**
...	...	...
823	16.12.2022	**NULL**

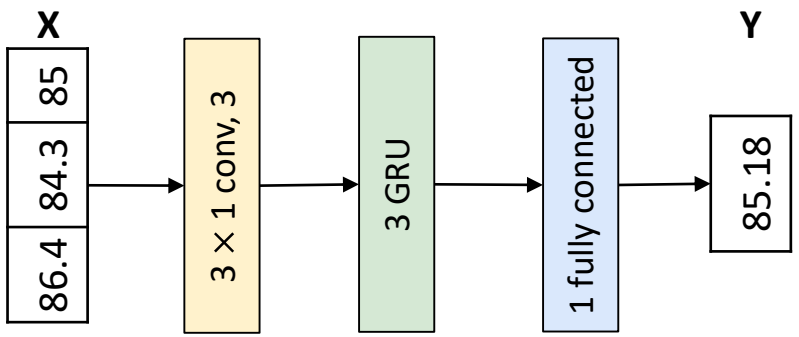
-- 1. Make a training set

```
create table trainTab as
  select * from ts_SensorData
  where stamp in
    [15.09.2020..14.09.2021]
```

trainTab

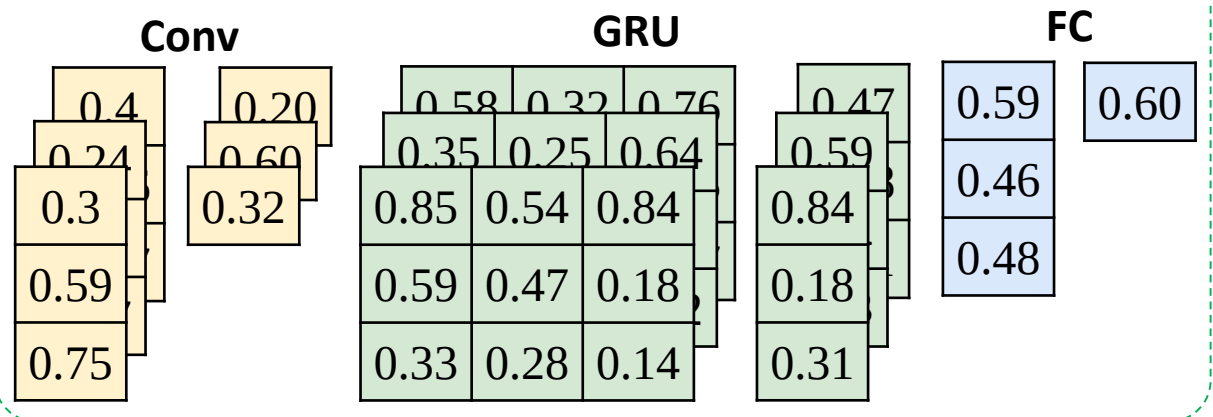
num	stamp	val
1	15.09.2020	86.4
...	...	...
365	14.09.2021	85.57

-- 2. Train ANN-based imputation model



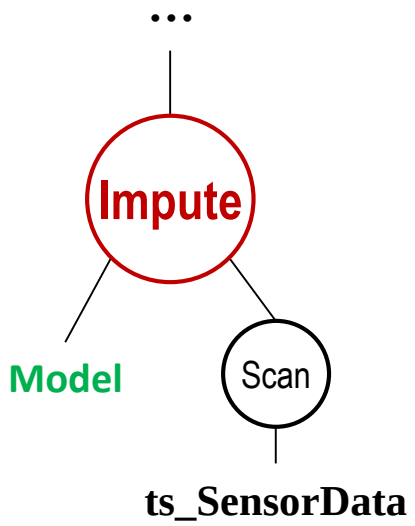
select fit_ImputeModel (trainTab, args_fit,...)

Model (matrices of weights and biases)



-- 3. Apply the model

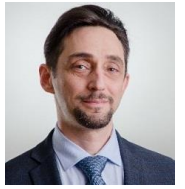
```
select * from ts_SenorData
```



Not a conclusion, but a proposal for further joint work

- Applying time series analytics to Smart DBMS
  - Anomaly detection to monitor database activities
  - Hardware lifecycle prediction
  - Workload prediction based on resource usage patterns
- Embedding time series analytics into DBMS
  - In-DBMS matrix profile support
  - Table data imputation: on the fly and/or in background
- Online time series analytics of mobile users
  - Activity recognition
  - Anomaly detection
  - Data imputation

Big Data and Machine Learning Laboratory



Mikhail Zymbler
Dr.Sci.
Assist. Prof.

Elena Ivanova
Cand.Sci.

Yana Kraeva
MSc,
2yr PhD student

Andrey Goglachev
MSc,
1yr PhD student

Alexey Yurtin
MSc,
1yr PhD student