Paper

Source code

Modeling Time-evolving Causality over Data Streams

Naoki Chihara, Yasuko Matsubara, Ren Fujiwara, Yasushi Sakurai

SANKEN, The University of Osaka

August 3–7, 2025
KDD2025
TORONTO

THE UNIVERSITY OF OSAKA

SANKEN
THE UNIVERSITY OF OSAKA

IST

Outline

- ☐ Background
- ☐ Proposed Model
- ☐ Optimization Algorithm
- ☐ Experiments
- ☐ Conclusion

Outline

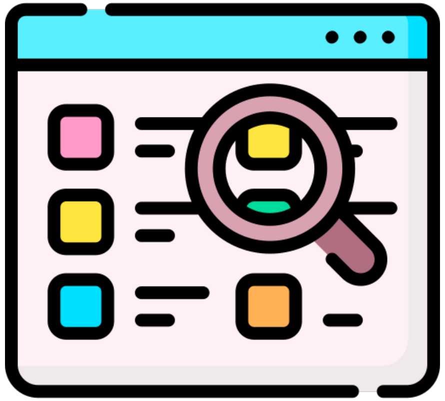
- ☐ **Background**
- ☐ Proposed Model
- ☐ Optimization Algorithm
- ☐ Experiments
- ☐ Conclusion

Multivariate Time Series

- Time series data has been collected from various domains

Motion analysis

Epidemiology

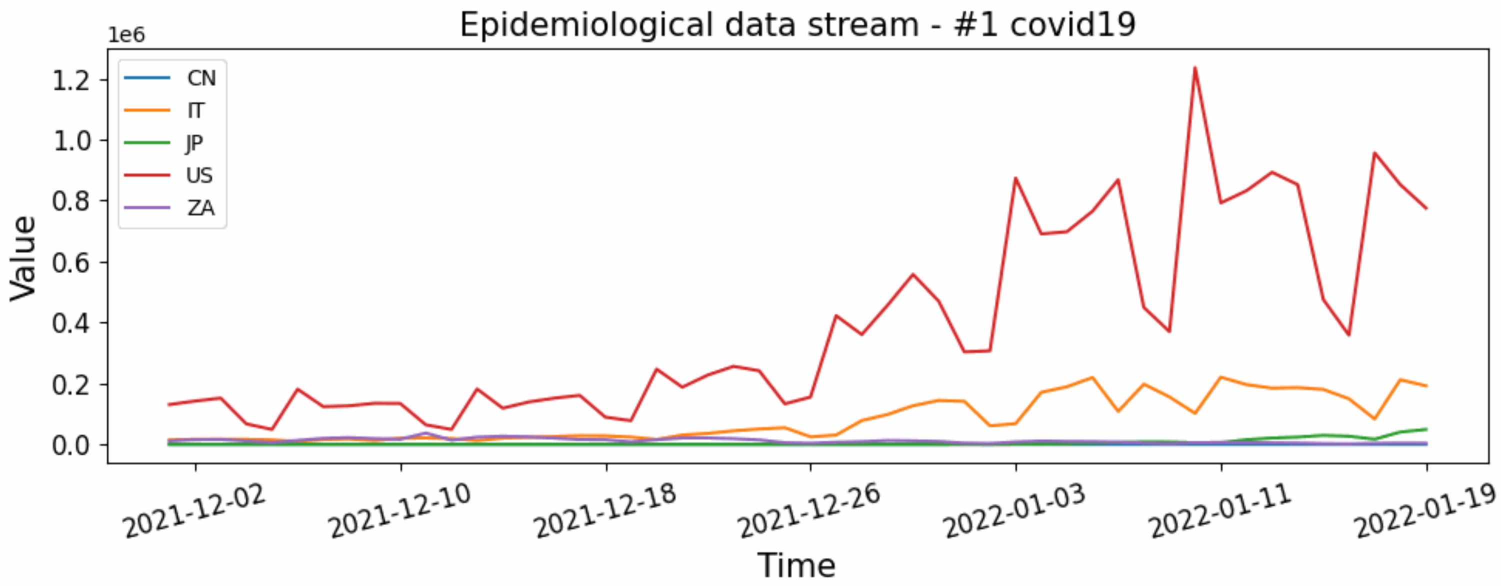


Web activity

Multivariate Time Series

- Time series data has been collected from various domains
- In real-world scenarios, these data are generated quickly and continuously

Epidemiology



Relationships between Observations

- Relationships between observations are critical for a wide range of time series analysis
  - E.g., Correlation, **Causality**, Independency
- Causality describes the relationship between cause and effect
  - Discovering causal relationships in time series data has been a long-standing challenge across many fields

Challenges: Time-evolving Causality

- However, most methods assume that causal relationships do not evolve over time 😢
  - Such approaches fall short in real-world applications
  - We refer to such relationships as **time-evolving causality**

Example. Spread of infectious diseases

- The emergence of a new virus strain leads to an increase in the number of infections in other countries
- Causative countries change over time

Challenges: Time-evolving Causality

- However, most methods assume that causal relationships do not evolve over time 😢
  - Such approaches fall short in real-world applications

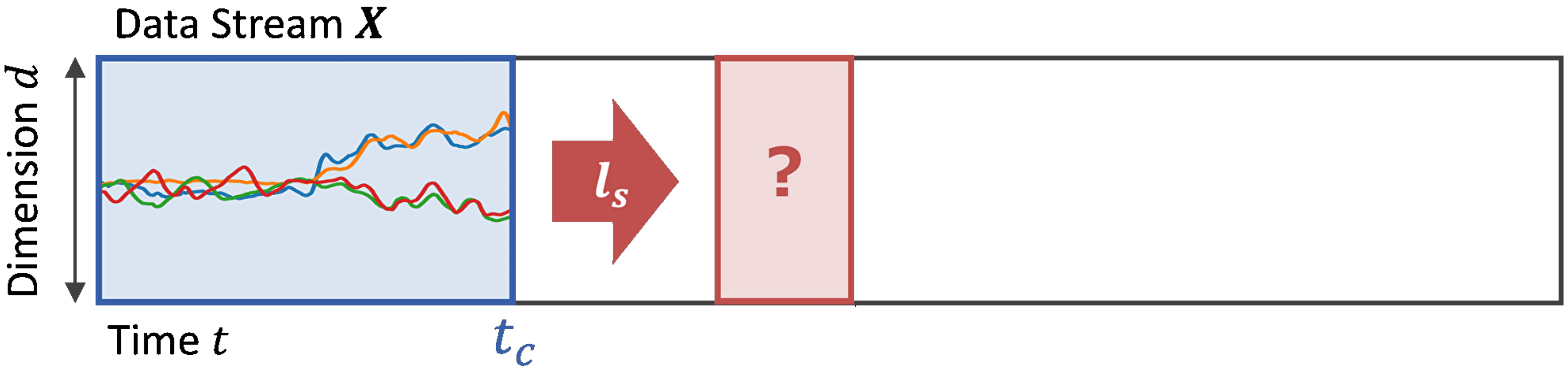
increase in the number of infections in other countries

- Causative countries change over time

We propose a novel **streaming** method **ModePlait** for modeling **time-evolving causality** and **forecasting**.

Problem Definition

- **Given:** Semi-infinite multivariate data stream $\boldsymbol{X} = \{\boldsymbol{x}(1), \ldots, \boldsymbol{x}(t_c), \ldots\}$
- **Goals:** Achieve all of the following requirements: (t_c: Current time point)
  - **Find** distinct dynamical patterns (i.e., regimes)
  - **Discover** time-evolving causality
  - **Forecast** an l_s-steps-ahead future value



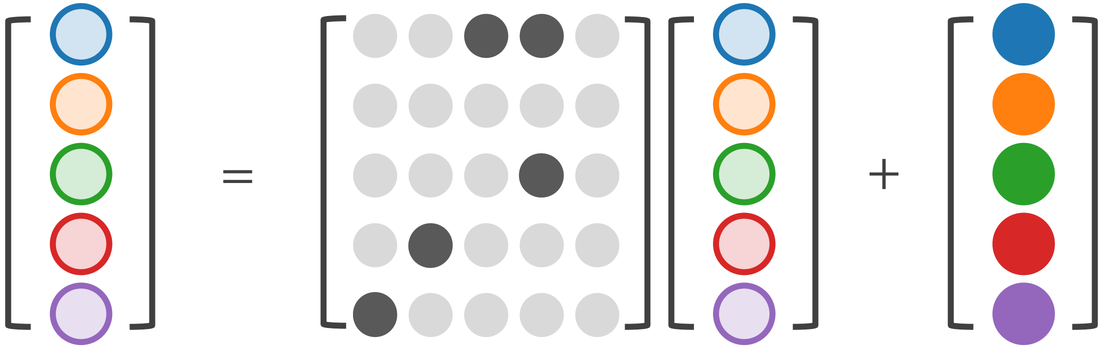
Outline

- ☐ Background
- ☐ **Proposed Model**
- ☐ Optimization Algorithm
- ☐ Experiments
- ☐ Conclusion

Principles and Concepts

- We design our proposed model based on the structural equation model (SEM) [Pearl 2009]

$$\underset{\textbf{Observed variables}}{\underline{X_{\text{sem}}}} = \underset{\textbf{Causal adjacency matrix}}{\underline{B_{\text{sem}}}} X_{\text{sem}} + \underset{\textbf{Exogenous variables}}{\underline{E_{\text{sem}}}}$$



Illusration of structural equation model (SEM)

Principles and Concepts

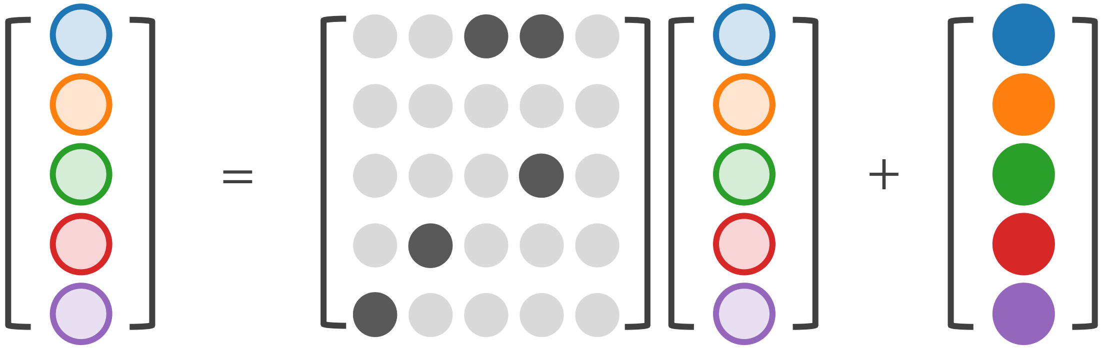
- We design our proposed model based on the structural equation model (SEM) [Pearl 2009]

$$\underline{X_{\text{sem}}} = \underline{B_{\text{sem}}}\, X_{\text{sem}} + \underline{E_{\text{sem}}}$$

Observed variables

Causal adjacency matrix

Exogenous variables



● : related
○ : not related

Unique component of each variable

Principles and Concepts

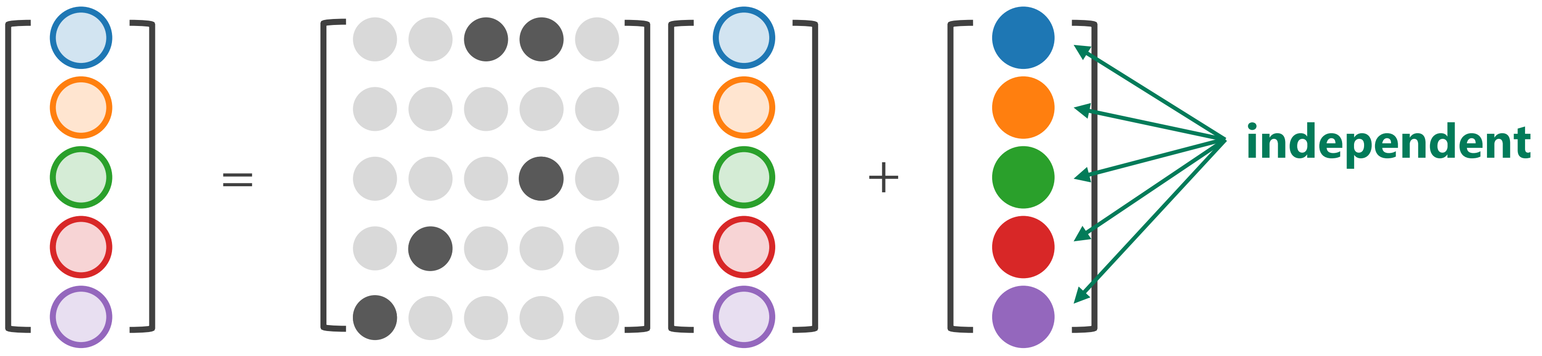
- We design our proposed model based on the structural equation model (SEM) [Pearl 2009]

$$\underline{X_{\text{sem}}} = \underline{B_{\text{sem}}}\, X_{\text{sem}} + \underline{E_{\text{sem}}}$$

Observed variables

Causal adjacency matrix

Exogenous variables



● : related

○ : not related

Unique component of each variable

Principles and Concepts

- We need to resolve the following questions to achieve our goal
  - How can we represent the inherent signals?
  - What is the best model for a single regime?
  - How can we handle multiple regimes in a data stream?

Principles and Concepts

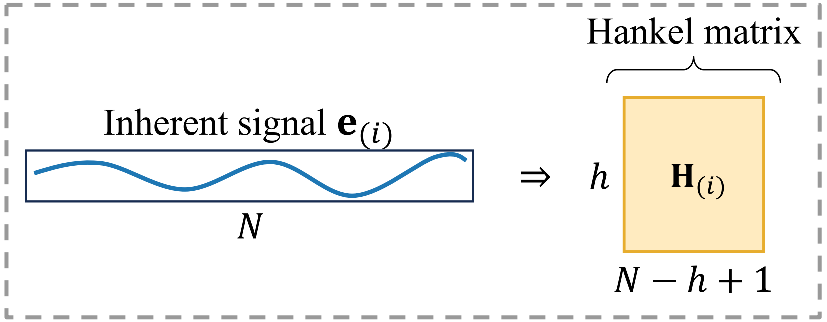
- We need to resolve the following questions to achieve our goal
  - How can we represent the inherent signals?
  - What is the best model for a single regime?
  - How can we handle multiple regimes in a data stream?

▼

1. **Latent temporal dynamics** of **inherent signals**
2. **Dynamical patterns** in a single regime
3. **Transitions of regimes** in a multivariate data stream

Latent temporal dynamics of inherent signal

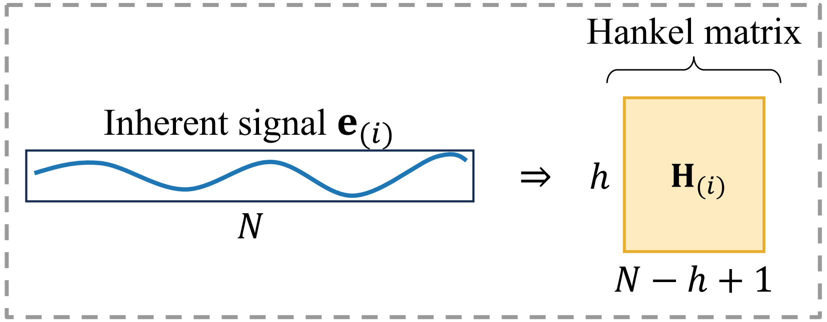
- We need to capture latent dynamics in univariate time series
  - Single dimension is inadequate for modeling the system 😢
  - We adopt the time-delay embedding to augment a state



$$\boldsymbol{H}_{(i)} = \begin{bmatrix} | & | & & | \\ g(e_{(i)}(h)) & g(e_{(i)}(h+1)) & \cdots & g(e_{(i)}(t)) \\ | & | & & | \end{bmatrix}$$

Latent temporal dynamics of inherent signal

- We need to capture latent dynamics in univariate time series
  - Single dimension is inadequate for modeling the system 😢
  - We adopt the time-delay embedding to augment a state

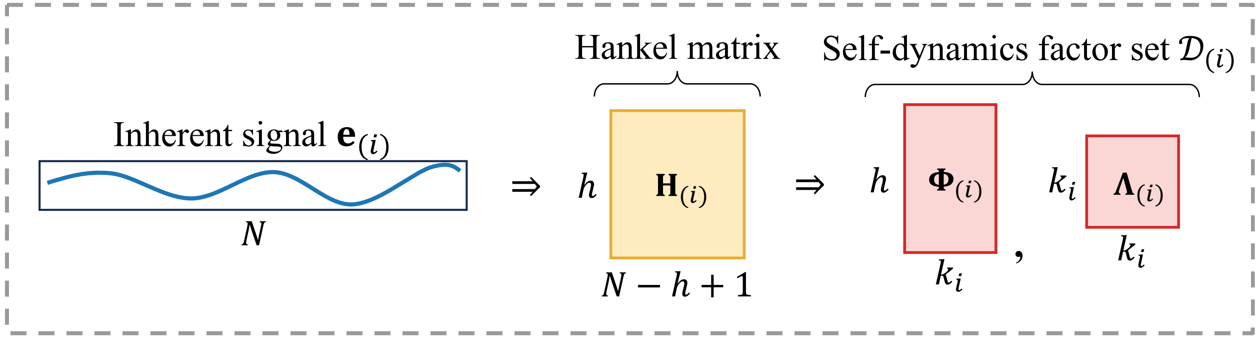


$$\boldsymbol{H}_{(i)} = \begin{bmatrix} | & | & & | \\ g(e_{(i)}(h)) & g(e_{(i)}(h+1)) & \cdots & g(e_{(i)}(t)) \\ | & | & & | \end{bmatrix}$$

$$g(e_{(i)}(t)) := (e_{(i)}(t), \underbrace{e_{(i)}(t-1), \ldots, e_{(i)}(t-h+1)}_{\text{Past history}}) \in \mathbb{R}^h$$

Latent temporal dynamics of inherent signal

- The i-th inherent signal $\boldsymbol{e}_{(i)}$ is given by the following equations



Self-dynamics factor set

$$\boxed{\mathcal{D}_{(i)} = \{\boldsymbol{\Phi}_{(i)}, \boldsymbol{\Lambda}_{(i)}\}}$$

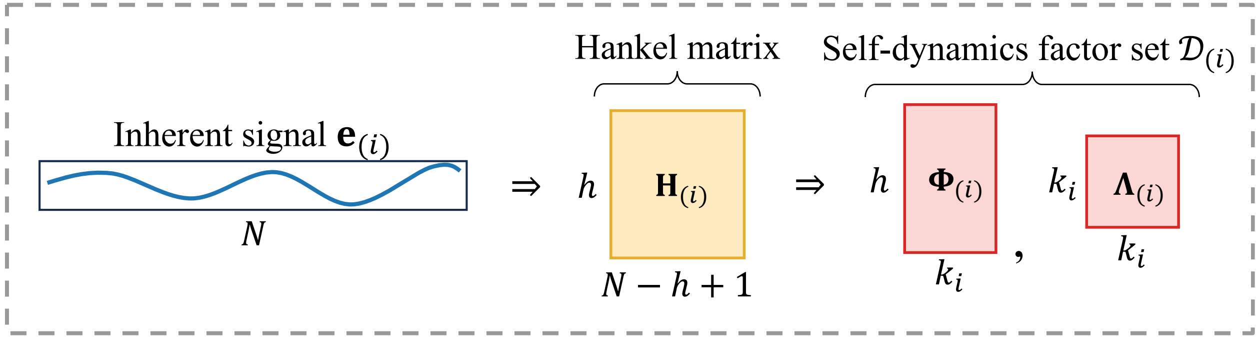
$$\underbrace{\boldsymbol{s}_{(i)}(t+1)}_{\textbf{Latent vector}} = \underbrace{\boldsymbol{\Lambda}_{(i)}}_{\text{Eigenvalues}} \boldsymbol{s}_{(i)}(t)$$: k_i-dimensional space

$$\underbrace{e_{(i)}(t)}_{\textbf{Inherent signal}} = \underbrace{g^{-1}}_{\text{Time-delay embedding}} (\underbrace{\boldsymbol{\Phi}_{(i)}}_{\text{Mode}} \boldsymbol{s}_{(i)}(t))$$: Projection ($\mathbb{C}^{k_i} \rightarrow \mathbb{R}$)

augmentation

Latent temporal dynamics of inherent signal

- The i-th inherent signal $\boldsymbol{e}_{(i)}$ is given by the following equations



Self-dynamics factor set

$$\boxed{\mathcal{D}_{(i)} = \{\boldsymbol{\Phi}_{(i)}, \boldsymbol{\Lambda}_{(i)}\}}$$

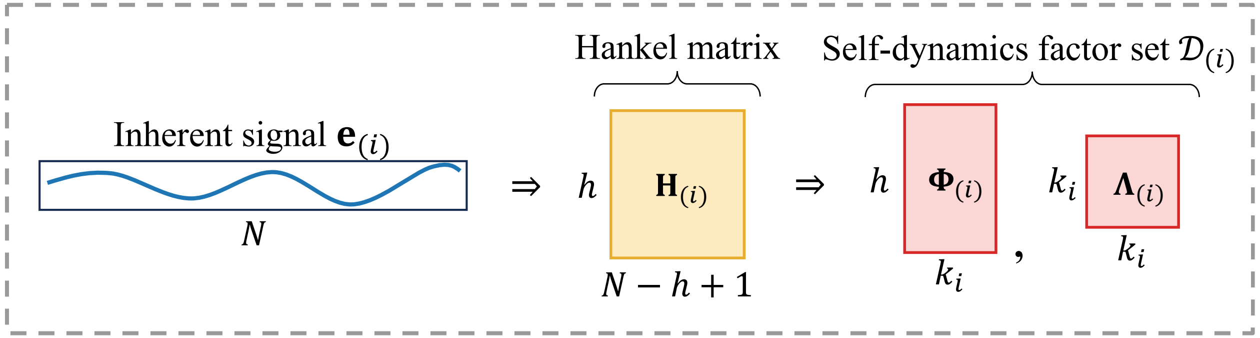
$$\underbrace{\boldsymbol{s}_{(i)}(t+1)}_{\text{Latent vector}} = \underbrace{\boldsymbol{\Lambda}_{(i)}}_{\text{Eigenvalues}} \boldsymbol{s}_{(i)}(t) \;:\; k_i\text{-dimensional space}$$

$$\underbrace{e_{(i)}(t)}_{\text{Inherent signal}} = \underbrace{g^{-1}}_{\text{Time-delay embedding}} (\underbrace{\boldsymbol{\Phi}_{(i)}}_{\text{Mode}} \boldsymbol{s}_{(i)}(t)) \;:\; \text{Projection } (\mathbb{C}^{k_i} \to \mathbb{R})$$

augmentation

Latent temporal dynamics of inherent signal

- The i-th inherent signal $\boldsymbol{e}_{(i)}$ is given by the following equations



Self-dynamics factor set

$$\boxed{\mathcal{D}_{(i)} = \{\boldsymbol{\Phi}_{(i)}, \boldsymbol{\Lambda}_{(i)}\}}$$

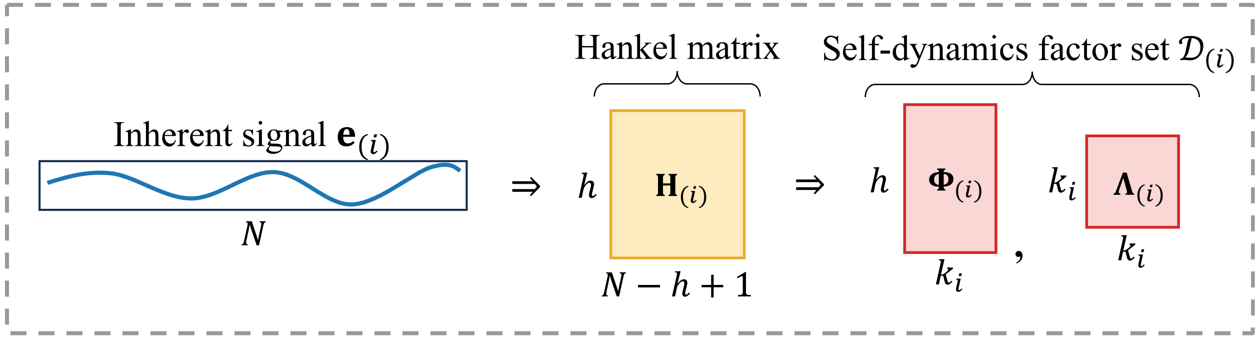
$$\underbrace{\boldsymbol{s}_{(i)}(t+1)}_{\text{Latent vector}} = \underbrace{\boldsymbol{\Lambda}_{(i)}}_{\text{Eigenvalues}} \boldsymbol{s}_{(i)}(t) \;:\; k_i\text{-dimensional space}$$

augmentation

$$\underbrace{e_{(i)}(t)}_{\text{Inherent signal}} = \underbrace{g^{-1}}_{\text{Time-delay embedding}} (\underbrace{\boldsymbol{\Phi}_{(i)}}_{\text{Mode}} \boldsymbol{s}_{(i)}(t)) \;:\; \text{Projection } (\mathbb{C}^{k_i} \rightarrow \mathbb{R})$$

Latent temporal dynamics of inherent signal

- The i-th inherent signal $\boldsymbol{e}_{(i)}$ is given by the following equations



Self-dynamics factor set

$$\boxed{\mathcal{D}_{(i)} = \{\mathbf{\Phi}_{(i)}, \mathbf{\Lambda}_{(i)}\}}$$

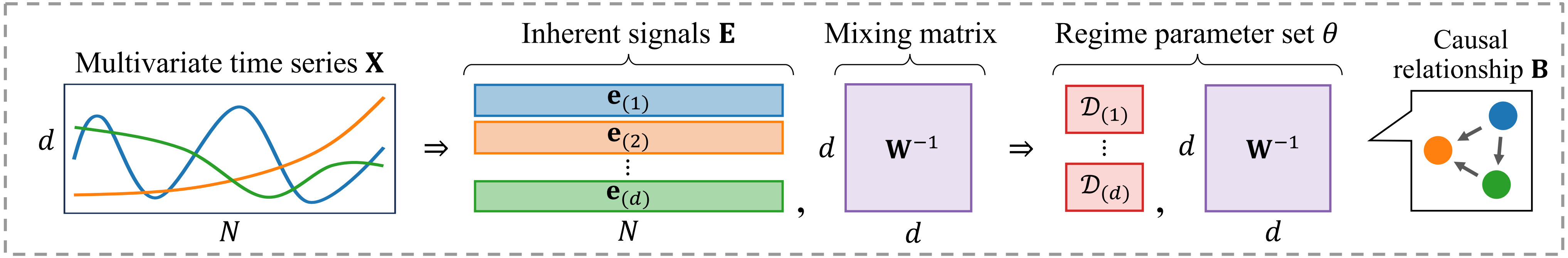
$$\underbrace{\boldsymbol{s}_{(i)}(t+1)}_{\text{Latent vector}} = \underbrace{\mathbf{\Lambda}_{(i)}}_{\text{Eigenvalues}} \boldsymbol{s}_{(i)}(t) \;:\; k_i\text{-dimensional space}$$

$$\underbrace{e_{(i)}(t)}_{\text{Inherent signal}} = \underbrace{g^{-1}}_{\text{Time-delay embedding}} (\underbrace{\mathbf{\Phi}_{(i)}}_{\text{Mode}} \boldsymbol{s}_{(i)}(t)) \;:\; \text{Projection } (\mathbb{C}^{k_i} \to \mathbb{R})$$

augmentation

Dynamical pattern in a single regime

- The single regime is governed by the following equations



$$\boxed{\begin{aligned} s_{(i)}(t+1) &= \Lambda_{(i)} s_{(i)}(t) \quad (1 \le i \le d) \\ e_{(i)}(t) &= g^{-1}(\Phi_{(i)} s_{(i)}(t)) \quad (1 \le i \le d) \end{aligned}}$$

A collection of d self-dynamics factor sets

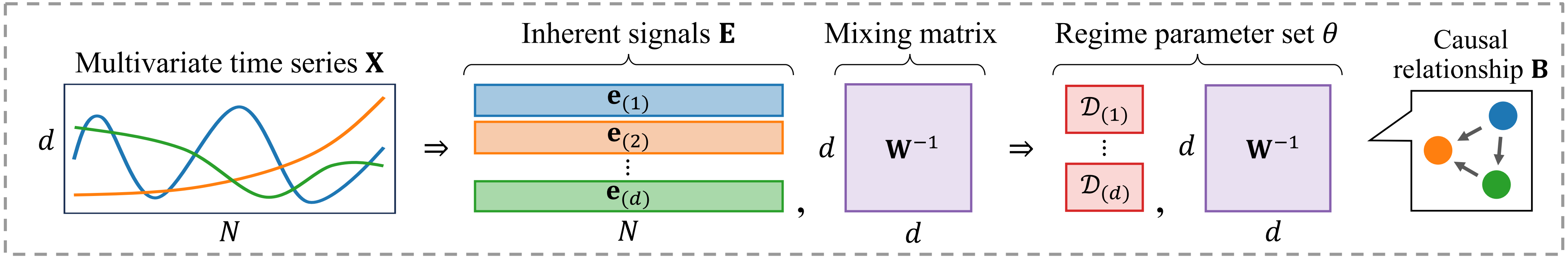
$$\underbrace{v(t)}_{\text{Estimated vector}} = \underbrace{W^{-1}}_{\text{Mixing matrix}} e(t) \quad \left(e(t) = \{e_{(i)}(t)\}_{i=1}^{d} \right)$$

Single regime

$$\boxed{\theta = \{W, \mathcal{D}_{(1)}, \dots, \mathcal{D}_{(d)}\}}$$

Dynamical pattern in a single regime

- The single regime is governed by the following equations



$$\boxed{\begin{aligned} s_{(i)}(t+1) &= \Lambda_{(i)} s_{(i)}(t) \quad (1 \le i \le d) \\ e_{(i)}(t) &= g^{-1}(\Phi_{(i)} s_{(i)}(t)) \quad (1 \le i \le d) \end{aligned}}$$

A collection of d self-dynamics factor sets

$$\underline{v(t)} = \underline{W^{-1}} \; e(t) \quad \left(e(t) = \{e_{(i)}(t)\}_{i=1}^{d}\right)$$

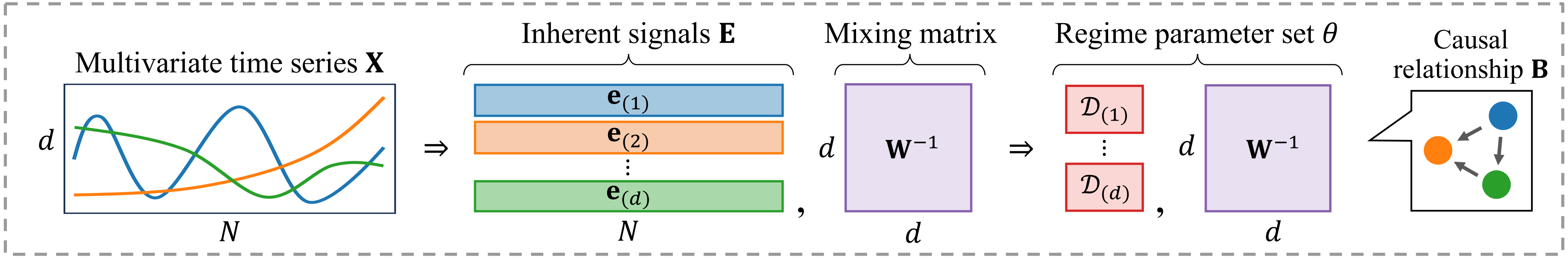
Estimated vector **Mixing matrix**

Single regime

$$\boxed{\theta = \{W, \mathcal{D}_{(1)}, \ldots, \mathcal{D}_{(d)}\}}$$

Dynamical pattern in a single regime

- The single regime is governed by the following equations



$$\boldsymbol{s}_{(i)}(t+1) = \Lambda_{(i)} \boldsymbol{s}_{(i)}(t) \quad (1 \le i \le d)$$

$$e_{(i)}(t) = g^{-1}(\Phi_{(i)} \boldsymbol{s}_{(i)}(t)) \quad (1 \le i \le d)$$

A collection of $\boldsymbol{d}$ self-dynamics factor sets

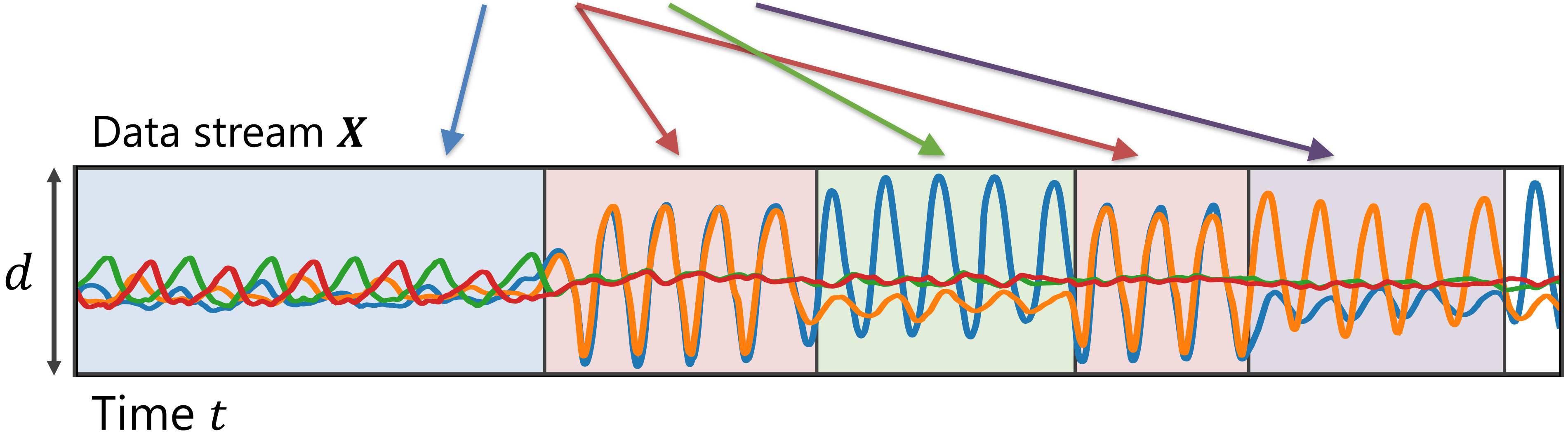
$$\underbrace{\boldsymbol{v}(t)}_{\text{Estimated vector}} = \underbrace{\boldsymbol{W}^{-1}}_{\text{Mixing matrix}} \boldsymbol{e}(t) \quad \left(\boldsymbol{e}(t) = \{e_{(i)}(t)\}_{i=1}^{d}\right)$$

Single regime

$$\boldsymbol{\theta} = \{\boldsymbol{W}, \mathcal{D}_{(1)}, \dots, \mathcal{D}_{(d)}\}$$

Transitions of regimes

- The transitions of regimes in a multivariate data stream
  - Regime set $\boldsymbol{\Theta} = \{\theta^1, \theta^2, \dots, \theta^R\}$ $\left(\theta^i = \left\{\boldsymbol{W}, \mathcal{D}_{(1)}, \dots, \mathcal{D}_{(d)}\right\}\right)$



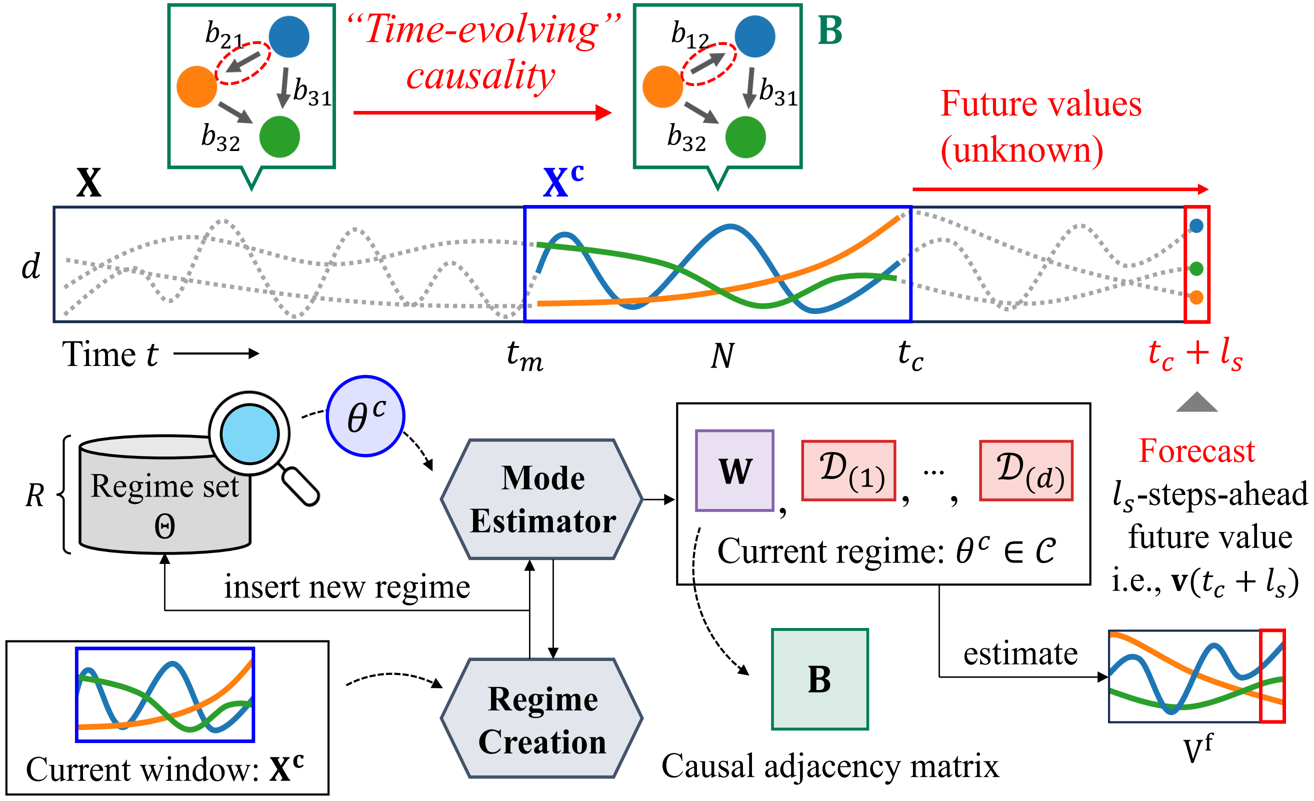
Outline

- ☐ Background
- ☐ Proposed Model
- ☐ **Optimization Algorithm**
- ☐ Experiments
- ☐ Conclusion

Optimization Algorithm

Proposed algorithm consists of the following components

- ModeEstimator
- RegimeCreation
- ModeGenerator
- RegimeUpdater



Update parameter:

$$\boldsymbol{\omega} = \{\{\boldsymbol{P}_{(i)}\}_{i=1}^{d}, \{\boldsymbol{\epsilon}_{(i)}\}_{i=1}^{d}\}$$

Full parameter set:

$$\mathcal{F} = \{\Theta, \Omega\}$$

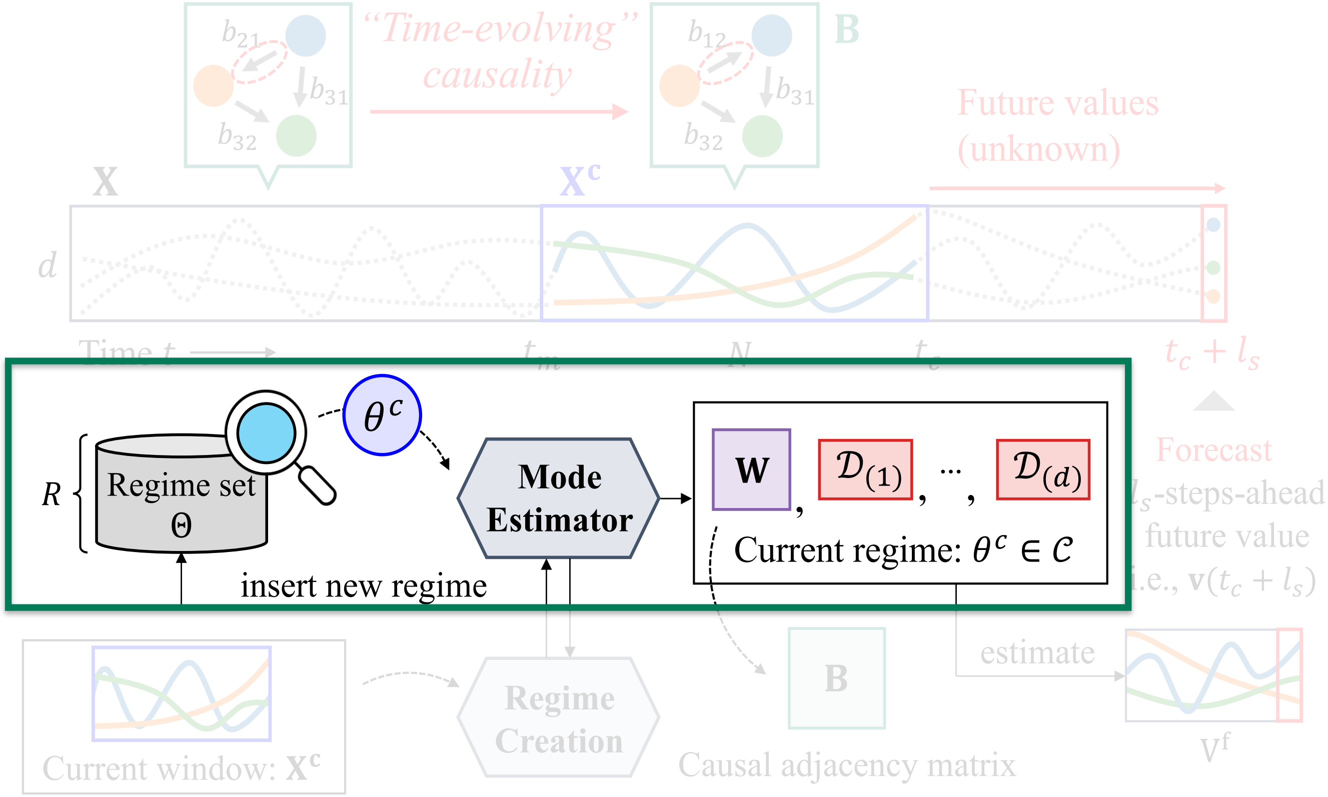
Model candidate:

$$\mathcal{C} = \{\boldsymbol{\theta}^c, \boldsymbol{\omega}^c, \boldsymbol{S}_{en}^c\}$$

Optimization Algorithm

Proposed algorithm consists of the following components

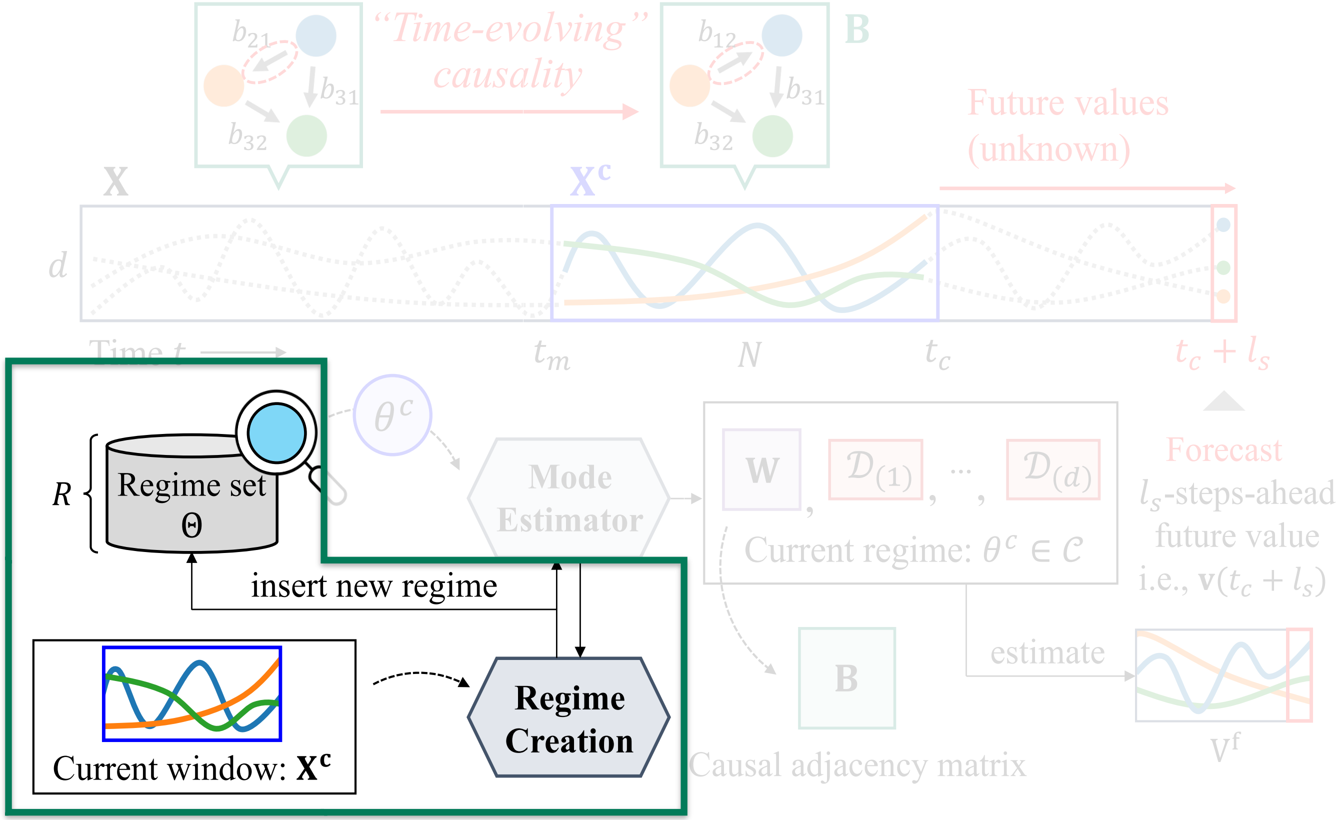
- **ModeEstimator**
  - Estimate $\mathcal{F}$ and $\mathcal{C}$ which appropriately describes the current dynamical pattern
- RegimeCreation
- ModeGenerator
- RegimeUpdater



Optimization Algorithm

Proposed algorithm consists of the following components

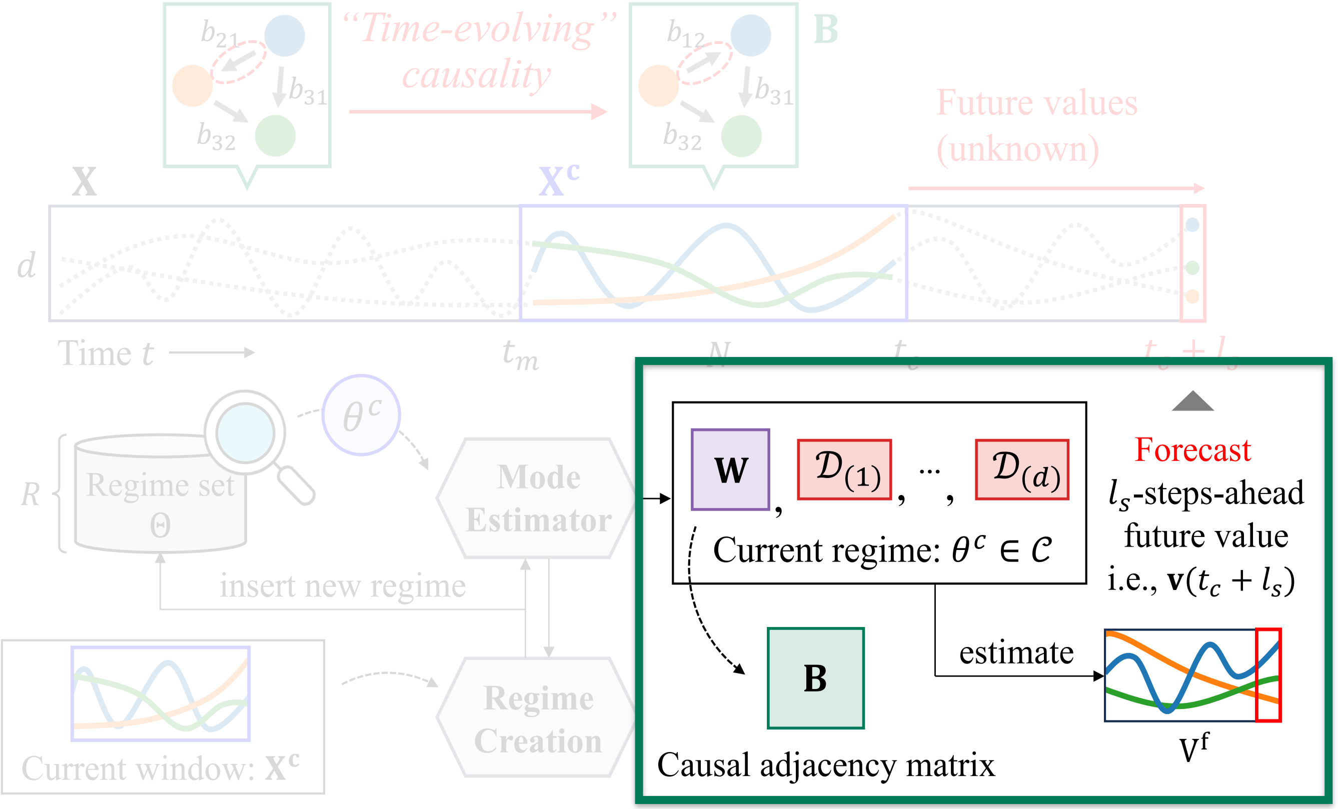
- ModeEstimator
- **RegimeCreation**
  - When it encounters an unknown pattern in $\boldsymbol{X}^c$, it estimates a new regime $\boldsymbol{\theta}$
- ModeGenerator
- RegimeUpdater



Optimization Algorithm

Proposed algorithm consists of the following components

- ModeEstimator
- RegimeCreation
- **ModeGenerator**
  - it identifies $\boldsymbol{B}$ and forecasts l_s-steps-ahead future value using $\mathcal{C}$
- RegimeUpdater



Optimization Algorithm

Proposed algorithm consists of the following components

- ModeEstimator
- RegimeCreation
- ModeGenerator
- **RegimeUpdater**
    - it updates $\boldsymbol{\theta}^c$ using $\boldsymbol{\omega} \in \mathcal{C}$ and the most recent value $\boldsymbol{x}(t_c)$

Details in paper

- Update demixing matrix $\boldsymbol{W}$
    - It is based on adaptive filtering
    - Ensure time and memory efficiency
- Update self-dynamics factor set $\mathcal{D}_{(i)}$

$$\boldsymbol{A}^{new}_{(i)} = \boldsymbol{A}^{prev}_{(i)} + (g(e_{(i)}(t_c)) - \boldsymbol{A}^{prev}_{(i)} g(e_{(i)}(t_c - 1)))\boldsymbol{\gamma}_{(i)}$$

$$\boldsymbol{\gamma}_{(i)} = \frac{g(e_{(i)}(t_c - 1))^\top \boldsymbol{P}^{prev}_{(i)}}{\mu + g(e_{(i)}(t_c - 1))^\top \boldsymbol{P}^{prev}_{(i)} \, g(e_{(i)}(t_c - 1))}$$

$$\boldsymbol{P}^{new}_{(i)} = \frac{1}{\mu}(\boldsymbol{P}^{prev}_{(i)} - \boldsymbol{P}^{prev}_{(i)} \, g(e_{(i)}(t_c - 1))\boldsymbol{\gamma}_{(i)})$$

Theoretical Analysis

- LEMMA 2 (CAUSAL IDENTIFIABILITY).
  *Causal discovery in MODEPLAIT is equivalent to finding the causal adjacency matrix **B** in MODEGENERATOR.*
  - It theoretically discovers causal relationships
- LEMMA 3 (TIME COMPLEXITY OF MODEPLAIT).
  The time complexity of MODEPLAIT is at least $O(N\sum_i k_i + dh^2)$ and at most $O(RN\sum_i k_i + N(d^2 + h^2) + k^2)$ per process.
  - It requires only constant time w.r.t. the entire data stream length
  - It is practical for semi-infinite data streams

Details in paper

Outline

- ☐ Background
- ☐ Proposed Model
- ☐ Optimization Algorithm
- ☐ **Experiments**
- ☐ Conclusion

Experiments

We aim to evaluate that **ModePlait** has ...

- **Q1. Effectiveness**
  How well does it find the time-evolving causality?
- **Q2. Accuracy**
  How accurately does it discover time-evolving causality and forecast future values?
- **Q3. Scalability**
  How does it scale in terms of computational time?

Experimental Setup

- 5 datasets
- Synthetics
  - We used it for the quantitative evaluation of causal discovery
  - 5 different temporal sequences
- Real-world datasets
  - Various domains datasets
    - Number of COVID-19 infections
    - Web-search counts
    - Sensor data from motion captures

- 12 baselines
  - CASPER
  - DARING
  - NoCurl
  - NO-MLP
  - NOTEARS
  - LiNGAM
  - GES
  - TimesNet
  - PatchTST
  - DeepAR
  - OrbitMap
  - ARIMA

7 models for **causal discovery**

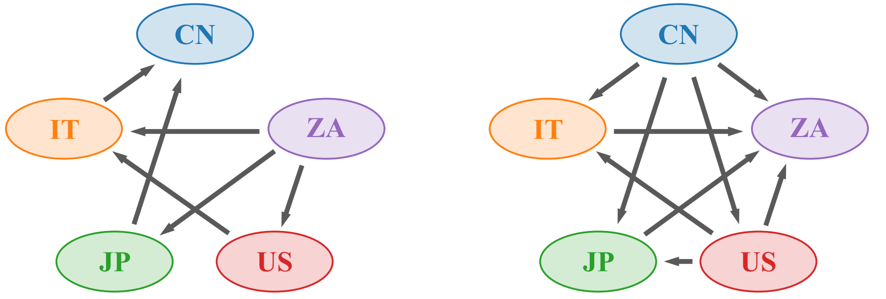
5 models for **time series forecasting**

Q1. Effectiveness

- Preview of our results from an epidemiological data stream
  - It consists of the number of COVID-19 infections in five countries

Base of arrows is cause, head is effect

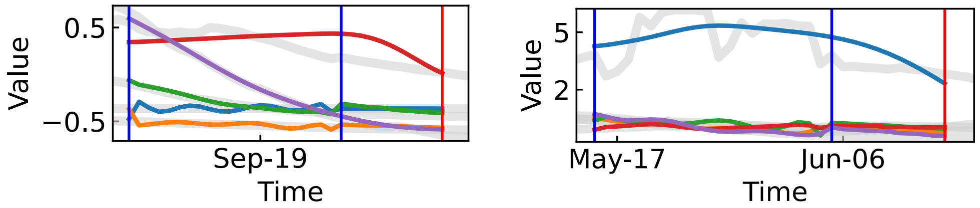
Health officials report a new lineage of the coronavirus in South Africa



(a-i) January 8, 2021 (a-ii) May 19, 2022

(a) Causal relationships at different time points

longest and toughest lockdowns in Shanghai



(c-i) September 27, 2021 (c-ii) June 5, 2022

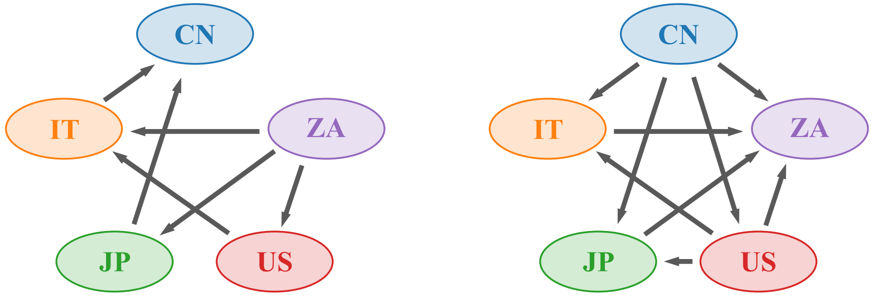
(c) Snapshots of 10 days-ahead future value forecasting

Accurate forecast based on the current distinct dynamical patterns

Q1. Effectiveness

- Preview of our results from an epidemiological data stream
  - It consists of the number of COVID-19 infections in five countries

Base of arrows is cause, head is effect



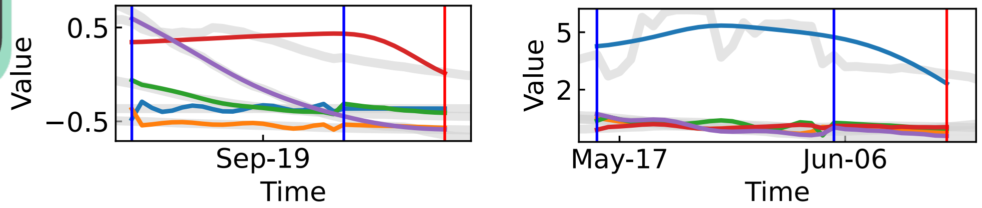
(a-i) January 8, 2021

(a-ii) May 19, 2022

(a) Causal relationships at different time points

longest and toughest lockdowns in Shanghai

Health officials report a new lineage of the coronavirus in South Africa



(c-i) September 27, 2021

(c-ii) June 5, 2022

(c) Snapshots of 10 days-ahead future value forecasting

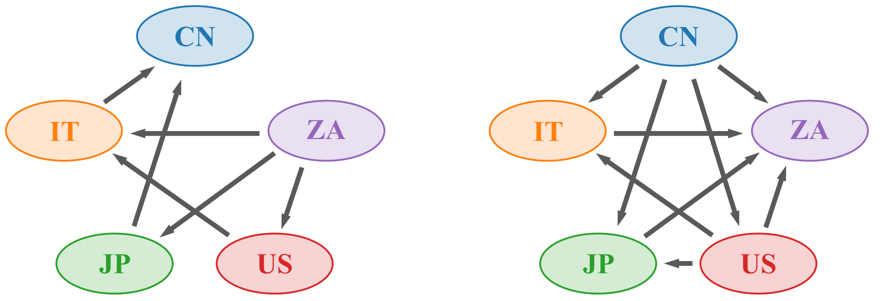
Accurate forecast based on the current distinct dynamical patterns

Q1. Effectiveness

- Preview of our results from an epidemiological data stream
  - It consists of the number of COVID-19 infections in five countries

Base of arrows is cause, head is effect

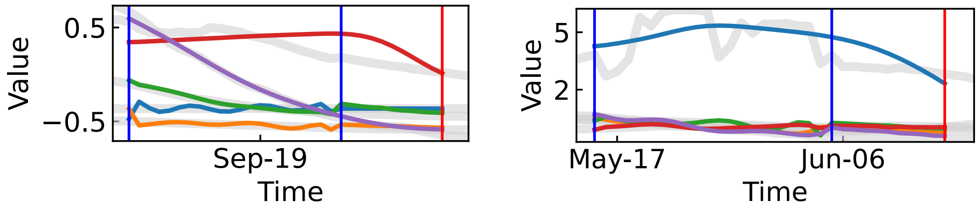
Health officials report a new lineage of the coronavirus in South Africa



(a-i) January 8, 2021 (a-ii) May 19, 2022

(a) Causal relationships at different time points

longest and toughest lockdowns in Shanghai



(c-i) September 27, 2021 (c-ii) June 5, 2022

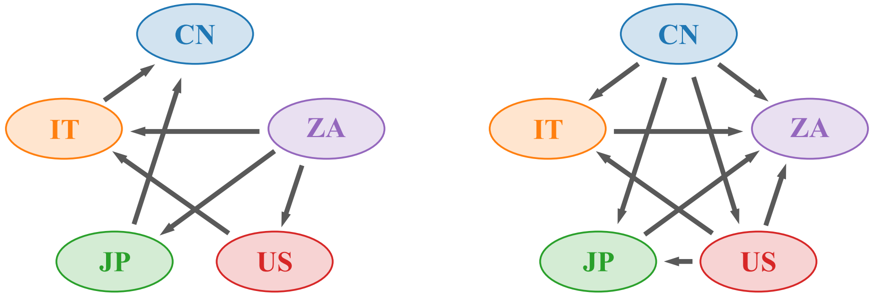
(c) Snapshots of 10 days-ahead future value forecasting

Accurate forecast based on the current distinct dynamical patterns

Q1. Effectiveness

- Preview of our results from an epidemiological data stream
  - It consists of the number of COVID-19 infections in five countries

Base of arrows is cause, head is effect



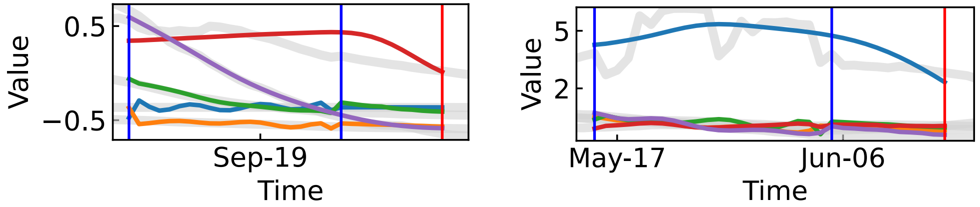
(a-i) January 8, 2021

(a-ii) May 19, 2022

(a) Causal relationships at different time points

longest and toughest lockdowns in Shanghai

Health officials report a new lineage of the coronavirus in South Africa



(c-i) September 27, 2021

(c-ii) June 5, 2022

(c) Snapshots of 10 days-ahead future value forecasting

Accurate forecast based on the current distinct dynamical patterns

Q2. Accuracy: Causal Discovery

"How accurately does ***ModePlait*** *discover time-evolving causality in a data stream?"*

Table 3: Causal discovering results with multiple temporal sequences to encompass various types of real-world scenarios.

Models	ModePlait		CASPER		DARING		NoCurl		NO-MLP		NOTEARS		LiNGAM		GES	
Metrics	SHD	SID	SHD	SID	SHD	SID	SHD	SID	SHD	SID	SHD	SID	SHD	SID	SHD	SID
1, 2, 1	**3.82**	**4.94**	5.58	7.25	5.75	8.58	6.31	9.90	6.36	8.74	5.03	9.95	7.13	8.23	7.49	11.7
1, 2, 3	**4.48**	**6.51**	5.97	8.44	5.81	9.17	6.13	9.51	6.44	8.77	5.69	9.56	6.79	7.33	7.03	10.1
1, 2, 2, 1	**4.32**	**5.88**	5.41	8.41	6.54	9.17	6.69	10.0	6.55	8.72	5.23	9.54	7.12	8.65	7.08	9.77
1, 2, 3, 4	**4.21**	**5.76**	6.22	8.33	6.12	9.58	6.10	9.61	6.62	8.87	5.73	10.1	7.10	8.50	7.29	11.3
1, 2, 3, 2, 1	**4.50**	**6.11**	6.02	8.28	5.45	7.77	6.20	9.83	6.56	8.83	5.57	9.11	7.46	8.05	7.74	12.1

Q2. Accuracy: Time Series Forecasting

*"How well does **ModePlait** forecast in a streaming fashion?"*

Table 4: Multivariate forecasting results for both synthetic and real-world datasets. We used forecasting steps $l_s \in \{5, 10, 15\}$.

Models		ModePlait		TimesNet		PatchTST		DeepAR		OrbitMap		ARIMA	
Metrics		RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
#0 synthetic	5	**0.722**	**0.528**	0.805	0.578	0.768	0.581	1.043	0.821	0.826	0.567	0.962	0.748
	10	**0.829**	**0.607**	0.862	0.655	0.898	0.649	1.073	0.849	0.896	0.646	0.966	0.752
	15	**0.923**	**0.686**	0.940	0.699	0.973	0.706	1.137	0.854	0.966	0.710	0.982	0.765
#1 covid19	5	**0.588**	**0.268**	0.659	0.314	0.640	0.299	1.241	0.691	1.117	0.646	1.259	0.675
	10	**0.740**	**0.361**	0.841	0.410	1.053	0.523	1.255	0.693	1.353	0.784	1.260	0.687
	15	**0.932**	**0.461**	1.026	0.516	1.309	0.686	1.265	0.690	1.351	0.792	1.277	0.718
#2 web-search	5	**0.573**	**0.442**	0.626	0.469	0.719	0.551	1.255	1.024	0.919	0.640	1.038	0.981
	10	**0.620**	**0.481**	0.697	0.514	0.789	0.604	1.273	1.044	0.960	0.717	1.247	1.037
	15	**0.646**	**0.505**	0.701	0.527	0.742	0.571	1.300	1.069	0.828	0.631	1.038	0.795
#3 chicken-dance	5	**0.353**	**0.221**	0.759	0.490	0.492	0.303	0.890	0.767	0.508	0.316	2.037	1.742
	10	**0.511**	**0.325**	0.843	0.564	0.838	0.535	0.886	0.753	0.730	0.476	1.863	1.530
	15	**0.653**	**0.419**	0.883	0.592	0.972	0.654	0.862	0.718	0.903	0.565	1.792	1.481
#4 exercise	5	**0.309**	**0.177**	0.471	0.275	0.465	0.304	0.408	0.290	0.424	0.275	1.003	0.748
	10	**0.501**	**0.309**	0.630	0.381	0.789	0.518	0.509	0.382	0.616	0.377	1.104	0.814
	15	0.687	**0.433**	0.786	0.505	1.147	0.758	**0.676**	0.475	0.691	0.434	1.126	0.901

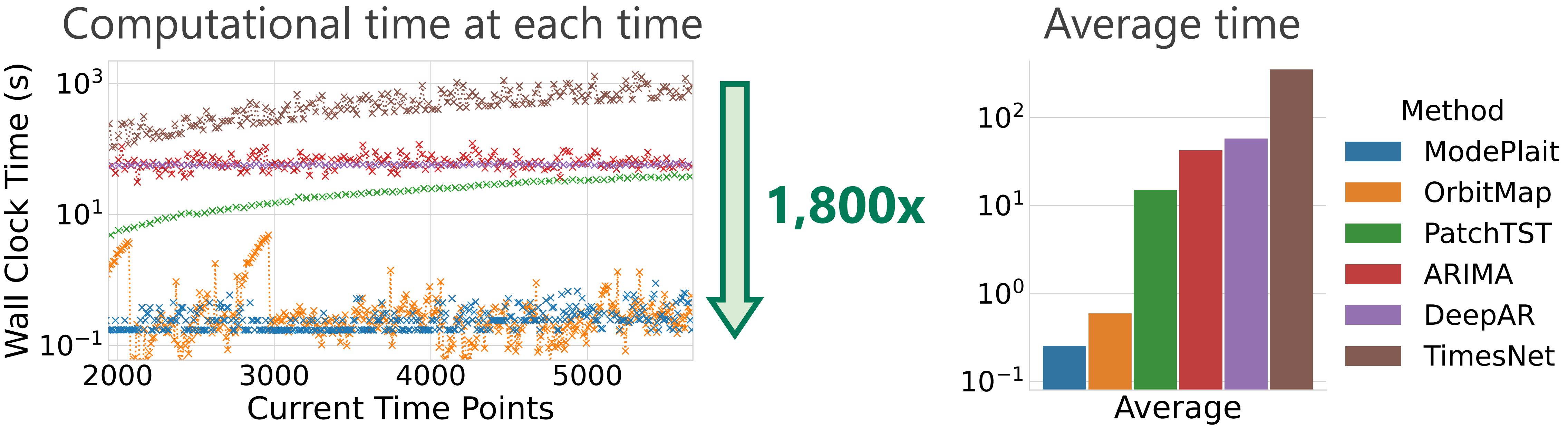
Q2. Accuracy: Ablation Study

“How substantially does causal discovery in a data stream enhance forecasting accuracy?”

Table 5: Ablation study results with forecasting steps $l_s \in \{5, 10, 15\}$ for both synthetic and real-world datasets.

Datasets		*#0 synthetic*		*#1 covid19*		*#2 web-search*		*#3 chicken-dance*		*#4 exercise*	
Metrics		RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
ModePlait (full)	5	0.722	0.528	0.588	0.268	0.573	0.442	0.353	0.221	0.309	0.177
	10	0.829	0.607	0.740	0.361	0.620	0.481	0.511	0.325	0.501	0.309
	15	0.923	0.686	0.932	0.461	0.646	0.505	0.653	0.419	0.687	0.433
w/o causality	5	0.759	0.563	0.758	0.374	0.575	0.437	0.391	0.262	0.375	0.218
	10	0.925	0.696	0.848	0.466	0.666	0.511	0.590	0.398	0.707	0.433
	15	1.001	0.760	1.144	0.583	0.708	0.545	0.821	0.537	0.856	0.533

Q3. Scalability


Computational time at each time

Average time

1,800x

*It requires only **constant computational time** with regard to the entire data stream length*

Outline

- ☐ Background
- ☐ Proposed Model
- ☐ Optimization Algorithm
- ☐ Experiments
- ☐ **Conclusion**

Conclusion

KDD Paper Source code

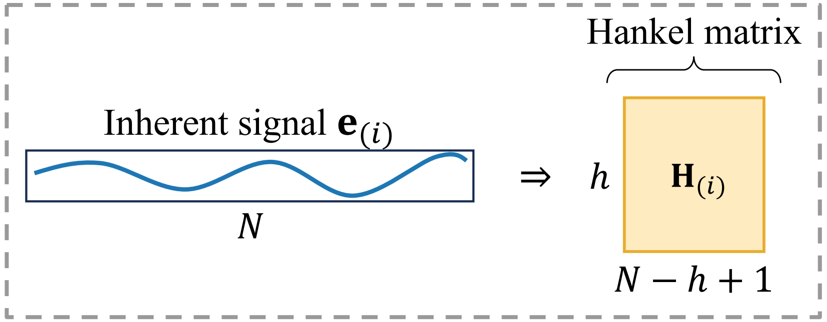
ModePlait has all of the following desirable properties

- **Effective**
  - It provides the time-evolving causality in a data stream based on monitoring regimes
- **Accurate**
  - It theoretically discovers time-evolving causality and precisely forecasts
  - Our experiments demonstrated that it outperforms its competitors
- **Scalable**
  - Our algorithm does not depend on data stream length

Appendix

Latent temporal dynamics of inherent signal

- We need to capture latent dynamics in univariate time series
  - Single dimension is inadequate for modeling the system 😢
  - We adopt the time-delay embedding to augment a state



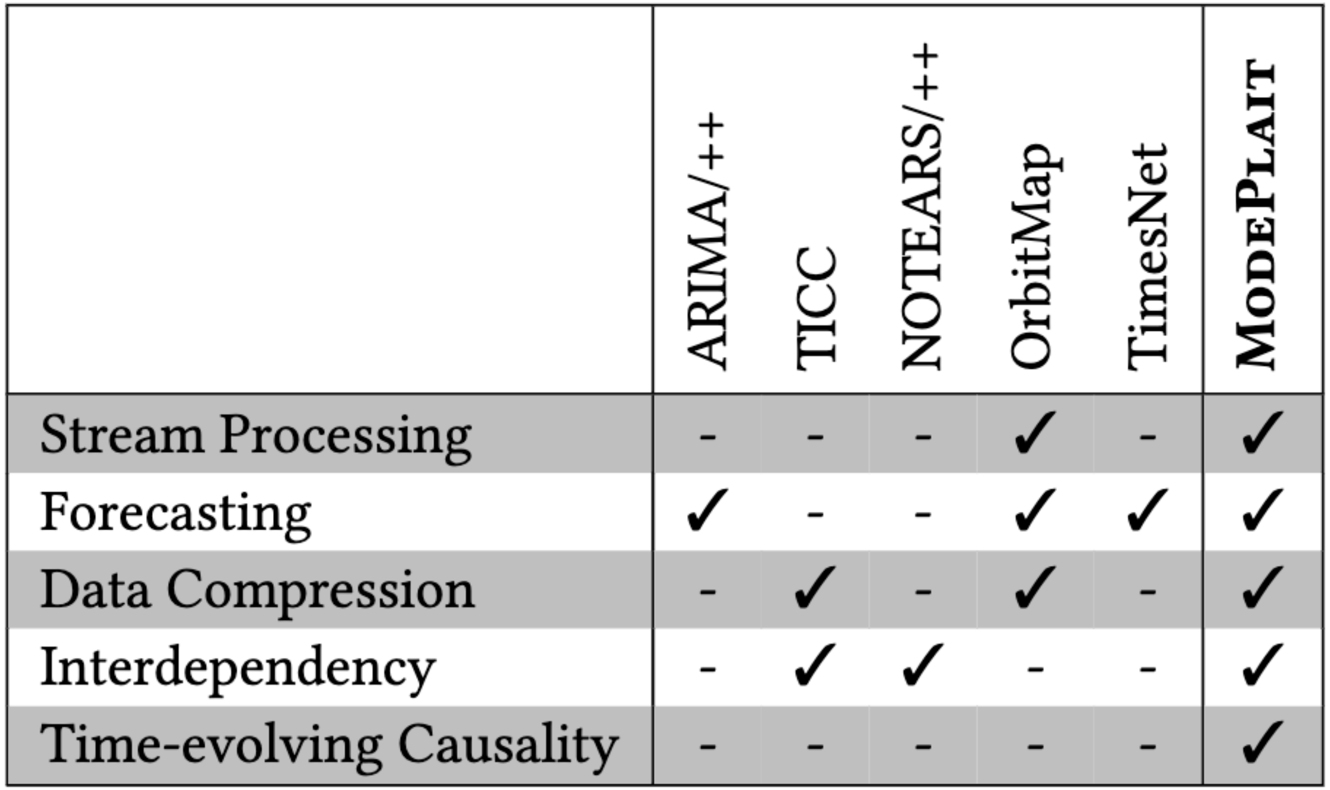
$$\boldsymbol{H}_{(i)} = \begin{bmatrix} | & | & & | \\ g(e_{(i)}(h)) & g(e_{(i)}(h+1)) & \cdots & g(e_{(i)}(t)) \\ | & | & & | \end{bmatrix}$$

According to Takens' embedding theorem

$$g(e_{(i)}(t)) \coloneqq (e_{(i)}(t), \underbrace{e_{(i)}(t-1), \ldots, e_{(i)}(t-h+1)}_{\textbf{Past history}}) \in \mathbb{R}^h$$

Related work

- ModePlait has the relative advantages with regard to five aspects.



	ARIMA/++	TICC	NOTEARS/++	OrbitMap	TimesNet	**ModePlait**
Stream Processing	-	-	-	✓	-	✓
Forecasting	✓	-	-	✓	✓	✓
Data Compression	-	✓	-	✓	-	✓
Interdependency	-	✓	✓	-	-	✓
Time-evolving Causality	-	-	-	-	-	✓

Related work

- ARIMA [Box and Jenkins 1976]
  - Classical method for time series forecasting
  - It assumes linear relationships between time series data 😢

- OrbitMap [Matsubara and Sakurai 2019]
  - Latest general method focusing on stream forecasting
  - It cannot discover the time-evolving causality 😢

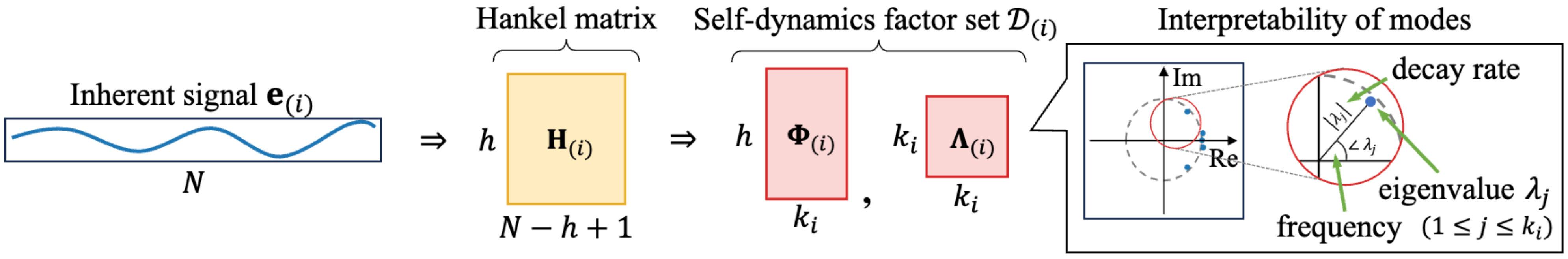
Related work

- Most methods for causal discovery
  - CASPER [Liu et al. 2023]etc.
  - It cannot handle time series data/data streams 😢

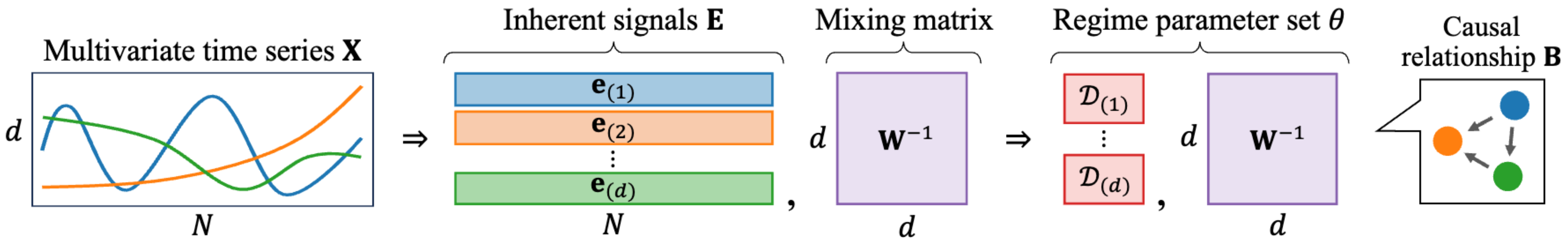
- Deep learning-based method for time series forecasting
  - TimesNet [Wu et al. 2023] etc.
  - The high computational costs associated with time series analysis hinders continuous model updating 😢

Proposal: Illustration of ModePlait

- Illustration of ModePlait is as follows



(a) Self-dynamics factor set (i.e., $\mathcal{D}_{(i)} = \{\mathbf{\Phi}_{(i)}, \mathbf{\Lambda}_{(i)}\}$)



(b) Single regime parameter set (i.e., $\theta = \{W, \mathcal{D}_{(1)}, \ldots, \mathcal{D}_{(d)}\}$)

Experiments: Metrics

We adopted SHD and SID to evaluate causal discovery accuracy

- structural Hamming distance (SHD)
  - It quantifies the difference in the causal adjacency matrix
  - It counts missing, extra, and reversed edges
- structural intervention distance (SID)
  - It is particularly suited to evaluate causal discovering accuracy
  - It counts the number of couples (i, j) such that the interventional distribution $p\left(x_j \mid \mathrm{do}(X_i = \bar{x})\right)$ would be miscalculated if we used the estimated causal adjacency matrix

Experiments: Metrics

We used RMSE and MAE to evaluate time series forecasting accuracy

- root mean square error (RMSE) ⋯ emphasizes large deviations

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \widehat{y}_i)}$$

- mean absolute error (MAE) ⋯ measures the overall errors

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \widehat{y}_i|$$

Experiments: Synthetics

Example of synthetics, “1, 2, 3“

- We generated synthetic datasets containing multiple clusters
  - Each cluster corresponds to one causal relationship
  - The causal adjacency matrix $\boldsymbol{B}$ is created based on Eröds-Rényi
  - Edge density $p = 0.5$, Number of observations $d = 5$

Proportion of actual edges