DIGITAL FINANCE

This project has received funding from the Horizon Europe research and innovation programme under the Marie Skłodowska-Curie Grant Agreement No. 101119635

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

State Secretariat for Education, Research and Innovation SERI

Funded by the European Union

MARIE CURIE ACTIONS

DIGITAL

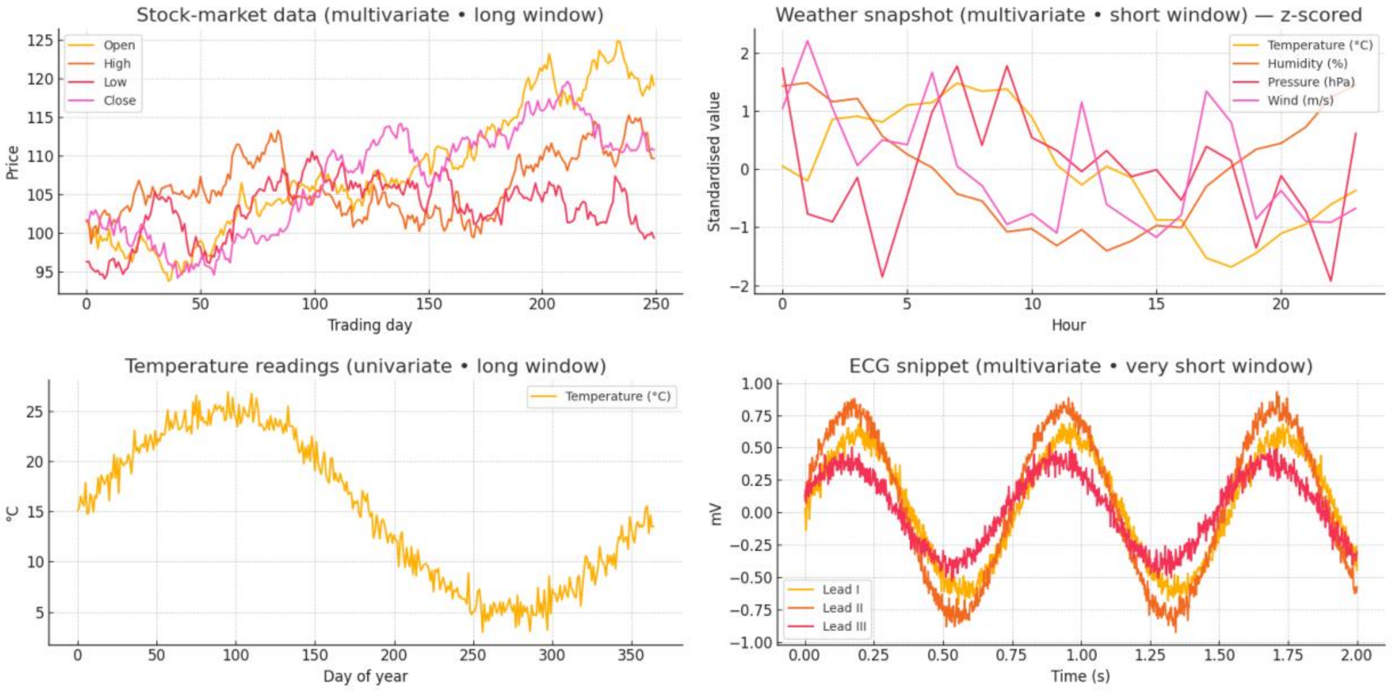
Time-Series xAI Methods

Faizan Ahmed

DIGITAL

Funded by
the European Union

Time series

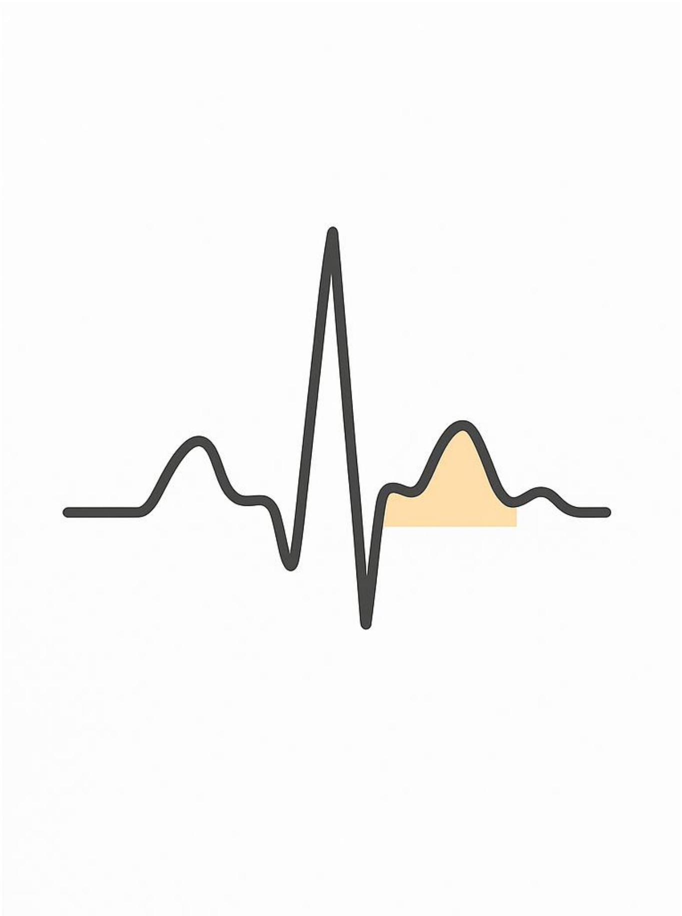


Number of observations collected over a successive period of time.

- Variable — *anything that changes over time*
- Time periods — *Can be daily, weekly, monthly, yearly*
- Variable Behaviour — *Quantifiable value*

DIGITAL

Image vs TS



DIGITAL

Why Special methods for XAI?

Time series lack the spatial structure—no pixels, colours or shapes to **rely on**.

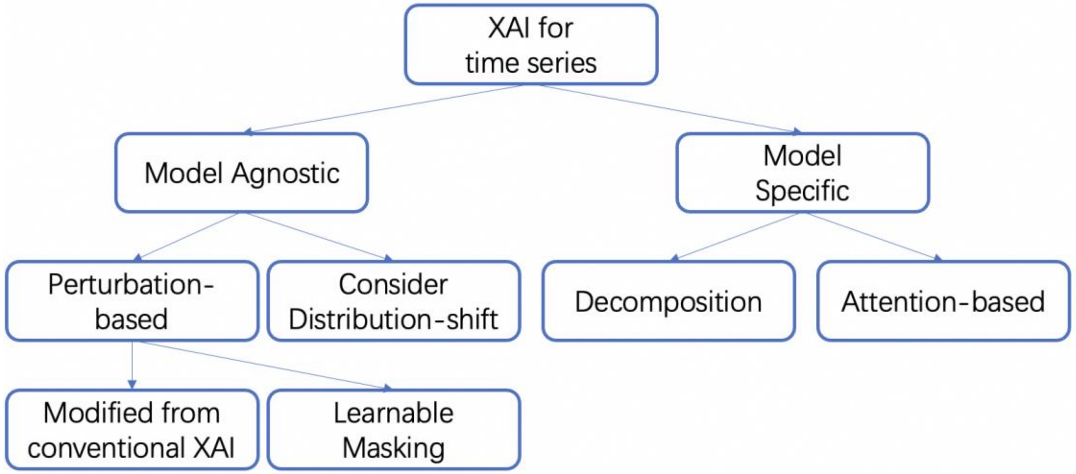
- Image saliency is visually intuitive; temporal signals rarely offer such anchors.
- Peaks and valleys alone seldom explain domain meaning; context over time is key.
- Example: In ECG subtle disturbances become clear only across several cardiac cycles.

Explanation methods for sequences must account for temporal dynamics.

Challenges

- choosing an in-distribution baseline
- Interpretable temporal representation
- Capturing temporal interactions
- Managing computational cost

DIGITAL



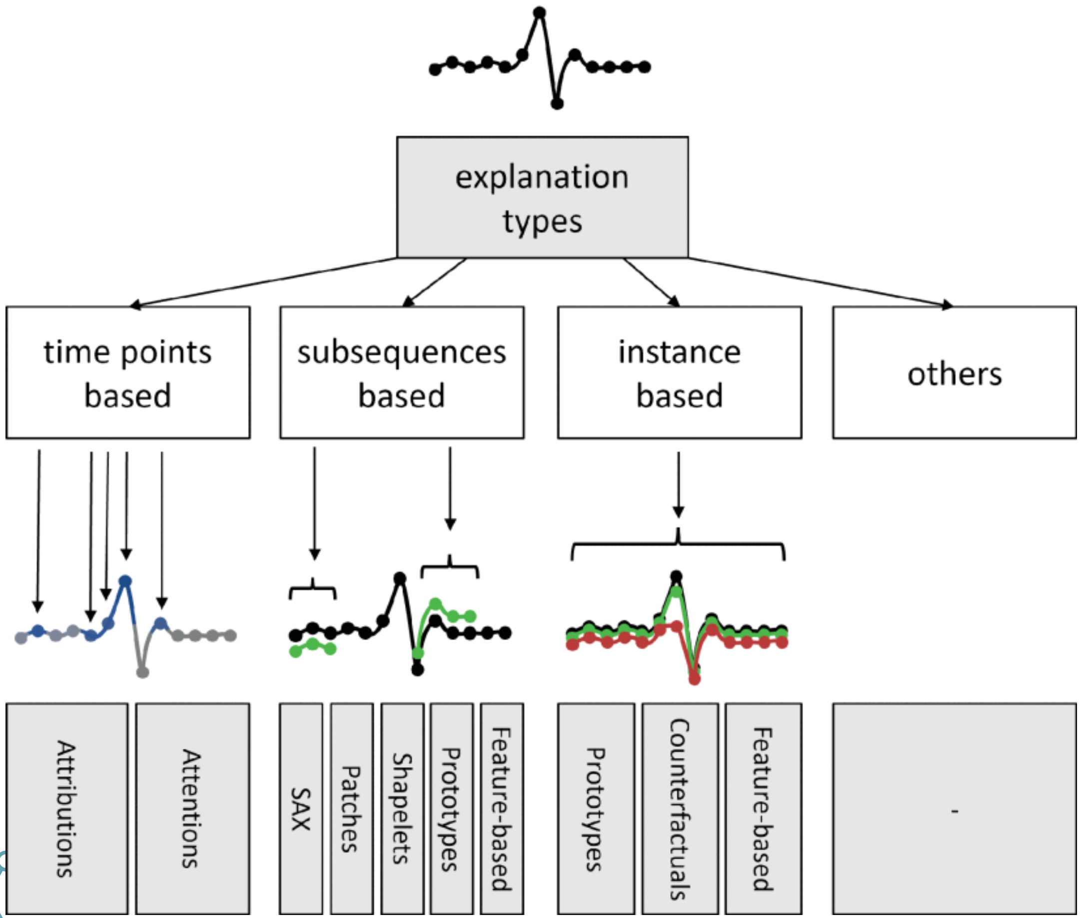
Family	Bucket	One-liner intuition
Model-agnostic	**Perturbation-based**	Mask or alter inputs and see how the prediction shifts (e.g., TimeSHAP, LIME-Segment, Dynamask).
	Distribution-aware	Replace inputs with *in-distribution* counterfactual samples; measure distributional shift (e.g., FIT).
Model-specific	**Decomposition-based**	Algebraically split the model into additive parts (e.g., Contextual Decomposition, ACD, REAT).
	Attention-score	Treat attention weights in RNNs/Transformers as importance indicators (e.g., RETAIN, TFT).

XAI for time series

TAXONOMY MODEL TYPE: GU, XINYUE & YANG, LINXIAO & SUN, LIANG. (2024). EXPLAINABLE ARTIFICIAL INTELLIGENCE FOR TIME SERIES: A SYSTEMATIC SURVEY. 10.13140/RG.2.2.23062.56642.

DIGITAL

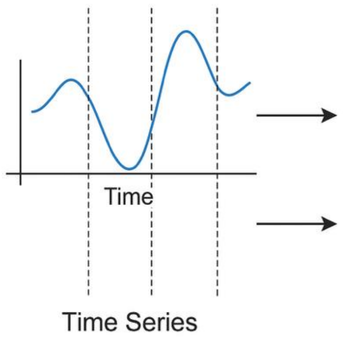
XAI for time series



- Taxonomy: Explanation Type
- See: https://ieeexplore.ieee.org/document/9895252

DIGITAL

XAI for TS: Windowing and Tabeling



Feature Engineering

Mean	Std. Dev.	Max
10.2	1.5	12,8
9.8	2.1	13,6
10.6	1.8	12,2
⋮	⋮	5.9
10.6	1.8	6,3
⋮	⋮	⋮

- **Window the raw signal** – slice it into fixed-length segments so each row in the future table corresponds to a "view" of the sequence.
- **Compute hand-crafted features** for each window
  - *Time-domain*: mean, variance, max/min, zero-crossings, slope, etc.
  - *Frequency-domain*: dominant frequency, band-power, spectral entropy, etc.
- **Result: a tabular matrix** (rows = windows, columns = features)
  - Now you can apply any tabular XAI tool (SHAP, LIME, feature permutation, global surrogate trees, etc.).
- **Pros & Cons**
  - **Pros:** interpretable features, fast inference, mature XAI support.
  - **Cons:** may discard fine-grained temporal patterns; requires domain knowledge to choose features.

Time Series Grad-CAM

Algorithm 3: Gradient-weighted Class Activation Mapping

Input: (Multi/single variate) time series t, trained CNN, target class c
Output: Heat-map L_{GradCAM}

; /* Forward pass */
$A^k \leftarrow$ feature-maps of the *last conv* layer;
$S_c \leftarrow$ predicted score for class c;
; /* Backward pass */
$\frac{\partial S_c}{\partial A^k} \leftarrow$ gradients w.r.t. each map;
; /* Channel importance */
$\alpha_k = \frac{1}{T}\sum_t \frac{\partial S_c}{\partial A_t^k}$
* average only along the time axis
; /* Linear combination & ReLU */
$L_{\mathrm{GradCAM}} = \mathrm{ReLU}(\sum_k \alpha_k A^k)$;
; /* Upsample */
Resize L_{GradCAM} to the resolution of T and overlay;
Generate Heatmap?

Mandatory:

1. Assaf, R., & Schumann, A. (2019, August). Explainable deep neural networks for multivariate time series predictions. In *IJCAI* (pp. 6488-6490).

2. *J. Van Der Westhuizen and J. Lasenby. Techniques for visualizing lstms applied to electrocardiograms. arXiv preprint arXiv:1705.08153, 2017.*

3. L. Tronchin *et al.*, "Translating Image XAI to Multivariate Time Series," in *IEEE Access*, vol. 12, pp. 27484-27500, 2024, doi: 10.1109/ACCESS.2024.3366994

DIGITAL

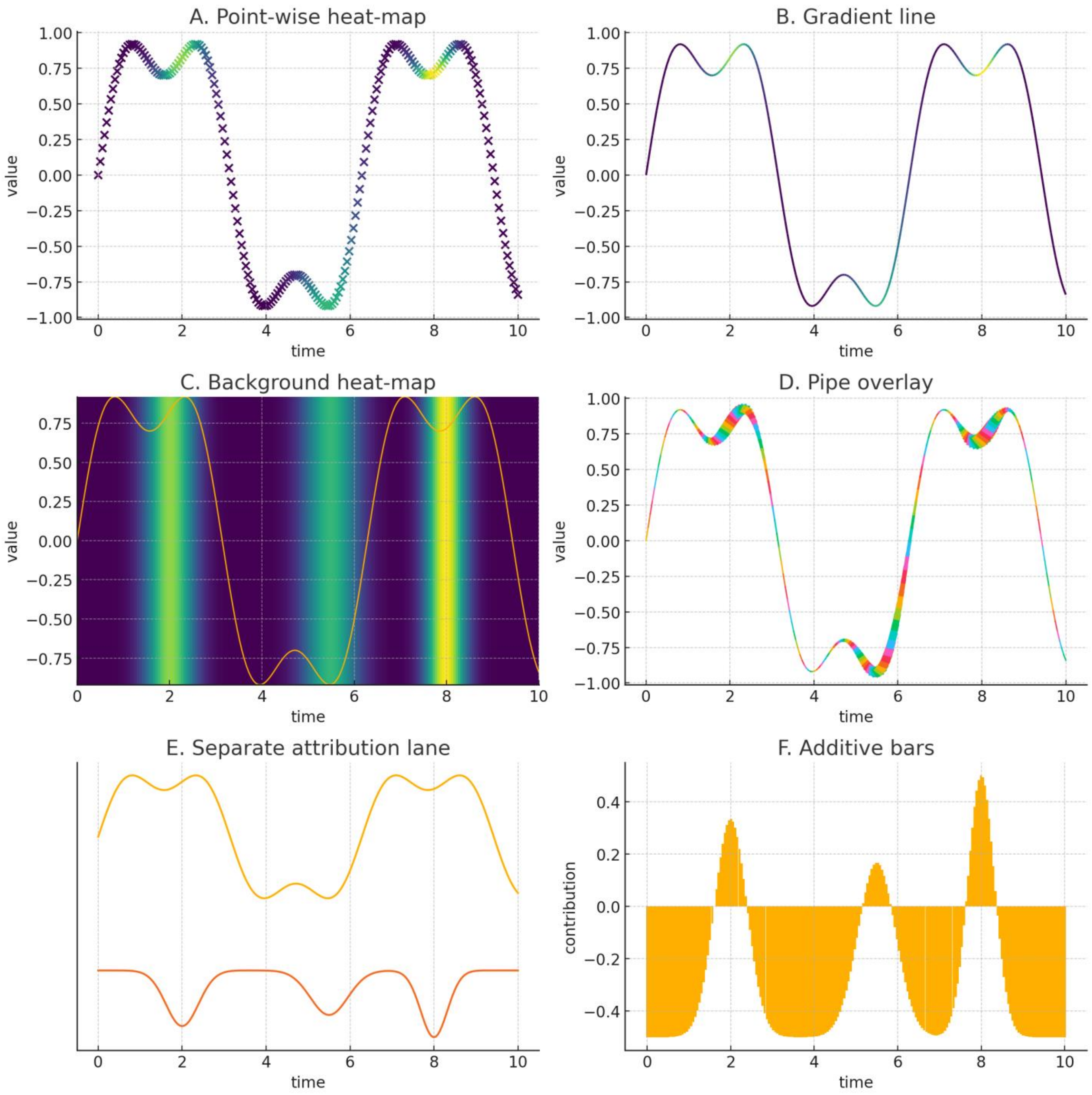
Family	Typical design	Strengths	Pain-points called out by the authors
A. Point-wise heat-maps	Put colour directly on the data marks (circles, dots).	extremely compact; preserves raw signal shape	heavy over-plotting, colour perception issues, later points over-draw earlier ones, hard beyond toy sequences
B. Gradient line segments	Encode relevance as a colour ramp along the poly-line.	keeps temporal ordering; no extra screen space	if the ramp is poorly chosen the signal itself becomes illegible; still juxtaposes high & low values without structure
C. Line-over-background heat-maps	Original curve in front, rectangular heat-map behind.	separates data & attribution layers; easy to add multiple attribution rows	overwhelms non-experts; small line vs huge heat-map; adjacent extremes visually clash
D. Dense pixel heat-maps (no signal)	Drop the line, show only a colour grid of relevance.	exposes recurrent patterns; good when the signal itself is distracting	eliminates temporal reference—experts struggle to relate colours back to real values
E. Pipe / tube overlays	Draw a variable-width, colour-coded “pipe” around the line.	width + colour jointly guide attention; low-relevance still visible	needs careful scale setting; may hide small fluctuations of the original series
F. Separate attribution lanes	Stack a second line plot (or small multiples) for relevance.	clean split—data intact, attribution legible; easy brushing & linking	relation between series & attribution requires eye jumps; less screen-efficient
G. Additive bar / arrow charts	Use SHAP additivity to draw positive/negative bars above & below the series, sometimes with arrows to compare models.	communicates direction (↑ helpful, ↓ harmful); supports multi-model comparison	only works for additive attributions; unsuitable for uni-variate series
H. Counter-factual-first workflows	Show “what would flip the prediction” examples first; drill down with attributions + what-if sliders.	aligns with Shneiderman mantra (overview → zoom → details); empirically easier for lay users	still a research vision; needs interactive tooling

Visualizing TS explanation

Time Series Model Attribution Visualizations as Explanations

DIGITAL

Visualizing TS expl



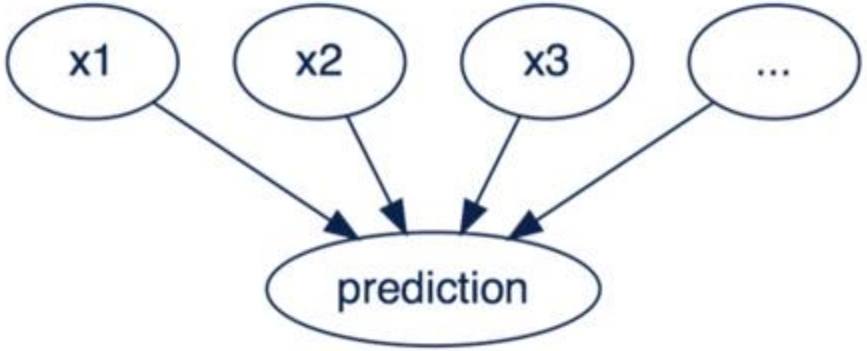
DIGITAL

A de tour...

DIGITAL

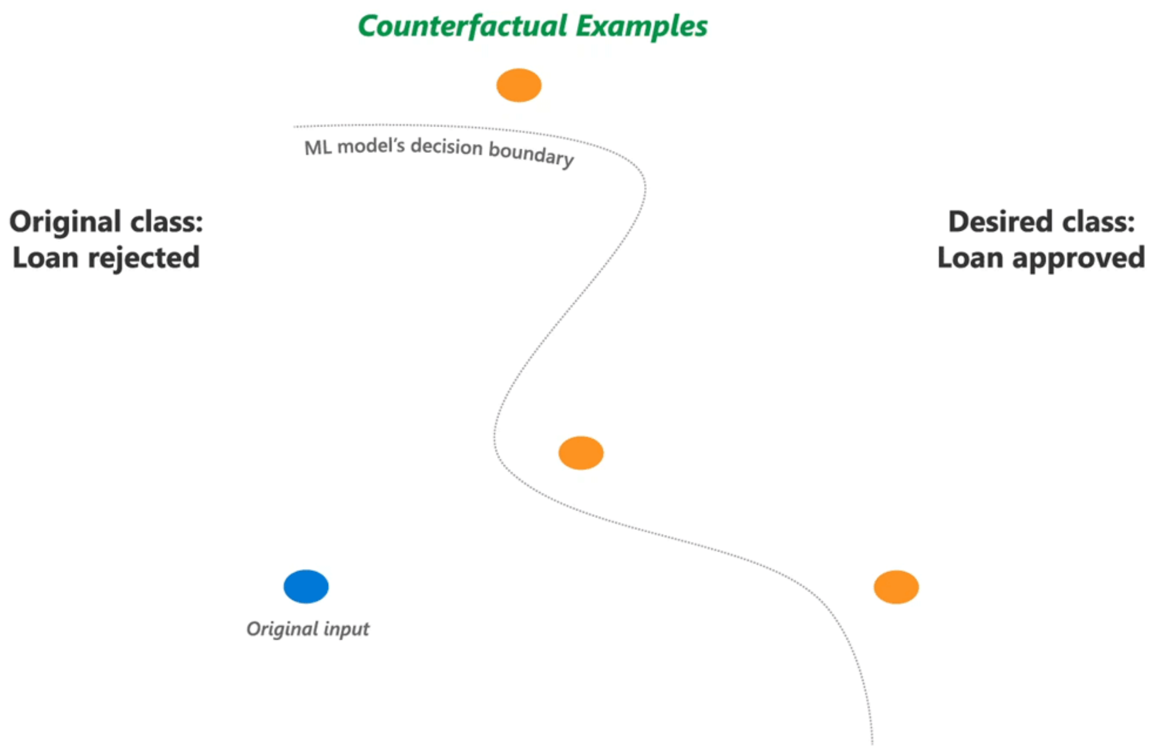
Counterfactual (CF) Explanations

- Considers the causal relationships between inputs of a machine learning model and the predictions, when the model is merely seen as a black box.
- Assumes the inputs cause the prediction (not necessarily reflecting the real causal relation of the data)



https://christophm.github.io/interpretable-ml-book/

Counterfactual (CF) Explanations



Visual representation of how counterfactuals work in a model trained for classifying loan approval status. Orange circle represent a counterfactual instance. Source - github.com/interpretml/DiCE

DIGITAL

Counterfactual (CF) Explanations

	Gender	Income	Education	…	Loan prediction
Query unit	F	$100,000	Bachelor's	…	0

DIGITAL

Counterfactual (CF) Explanations

	Gender	Income	Education	…	Loan prediction
Query unit	F	$100,000	Bachelor's	…	0
CF1	M	$100,000	Bachelor's	…	1
CF2	M	$1,100,000	Bachelor's	…	1
CF3	M	$100,000	Master's	…	1

DIGITAL

Counterfactual (CF) Explanations

	Gender	Income	Education	…	Loan prediction
Query unit	F	$100,000	Bachelor's	…	0
CF1	M	$100,000	Bachelor's	…	1
CF2	M	$1,100,000	Bachelor's	…	1
CF3	M	$100,000	Master's	…	1
CF4	F	$110,000	Master's	…	1

What if we also saw CF4?

DIGITAL

Counterfactual (CF) Explanations

- Predictive classifier *f*
- Instance ***x*** (observation), *y* (outcome)
- Goal: create counterfactuals {c1, …, ck} that are
  - Diverse : different from one another

	Gender	Income	Education	…	Loan prediction
Query unit	F	\$100,000	Bachelor's	…	0
Bad CF	M	\$100,000	Bachelor's	…	1
Good CF	F	\$100,100	Bachelor's	…	1

DIGITAL

Counterfactual (CF) Explanations

- Predictive classifier *f*
- Instance ***x*** (observation), *y* (outcome)
- Goal: create counterfactuals {c1, …, ck} that are
  - Sparse : do not involve too many features

	Gender	Income	Education	…	Loan prediction
Query unit	F	$100,000	Bachelor’s	…	0
Bad CF	M	$100,100	Master’s	…	1
Good CF	F	$100,100	Bachelor’s	…	1

DIGITAL

Counterfactual Explanations Evaluation

- *Validity*: the counterfactuals' predicted outcome is different than original outcome
- *Proximity*: the counterfactuals should be similar to the query instance
- *Sparsity*: the counterfactuals should not require changing too many covariates
- *Diversity*: the counterfactuals should be different from one another

DIGITAL

Counterfactual (CF) Explanations

$$\arg\min_{x'} \max_{\lambda} L(x, x', y', \lambda)$$

The loss measures how far the predicted outcome of the counterfactual is from the predefined outcome and how far the counterfactual is from the instance of interest

$$L(x, x', y', \lambda) = \lambda \cdot (\hat{f}(x') - y')^2 + d(x, x')$$

quadratic distance between the model prediction for the counterfactual x' and the desired outcome y'

the distance d between the instance x to be explained and the counterfactual x'

$$d(x, x') = \sum_{j=1}^{p} \frac{|x_j - x'_j|}{MAD_j}$$

Manhattan distance weighted with the inverse median absolute deviation (MAD) of each feature

$$MAD_j = \text{median}_{i \in \{1,...,n\}}(|x_{i,j} - \text{median}_{l \in \{1,...,n\}}(x_{l,j})|)$$

The total distance is the sum of all p feature-wise distances, that is, the absolute differences of feature values between instance x and counterfactual x'. The feature-wise distances are scaled by the inverse of the median absolute deviation of feature j over the dataset

DIGITAL

Counterfactual (CF) Explanations

$$\arg\min_{x'}\max_{\lambda} L(x, x', y', \lambda)$$

instead of selecting a value for λ to select a tolerance ϵ for how far away from y′ the prediction of the counterfactual instance is allowed to be.

$$|\hat{f}(x') - y'| \leq \epsilon$$

$$L(x, x', y', \lambda) = \boxed{\lambda} \cdot (\hat{f}(x') - y')^2 + d(x, x')$$

A higher value of λ means that we prefer counterfactuals with predictions close to the desired outcome y', a lower value means that we prefer counterfactuals x' that are very similar to x in the feature values

quadratic distance between the model prediction for the counterfactual x' and the desired outcome y'

the distance d between the instance x to be explained and the counterfactual x'

$$d(x, x') = \sum_{j=1}^{p} \frac{|x_j - x'_j|}{MAD_j}$$

$$MAD_j = \text{median}_{i\in\{1,\ldots,n\}}(|x_{i,j} - \text{median}_{l\in\{1,\ldots,n\}}(x_{l,j})|)$$

DIGITAL

Counterfactual (CF) Explanations

1. Select an instance x to be explained, the desired outcome y', a tolerance ϵ and a (low) initial value for λ.
2. Sample a random instance as initial counterfactual.
3. Optimize the loss with the initially sampled counterfactual as starting point.
4. While $|\hat{f}(x') - y'| > \epsilon$:
   - Increase λ.
   - Optimize the loss with the current counterfactual as starting point.
   - Return the counterfactual that minimizes the loss.
5. Repeat steps 2-4 and return the list of counterfactuals or the one that minimizes the loss.

DIGITAL

Coming back to time series…

Given a trained classifier f and an input multivariate time series X that is predicted as class c, find an alternative X' such that:

$$f(X') = c_{target} \quad X' \; is\ as\ close\ as\ possible\ to\ X$$

- Instead of simply flipping the output, CoMTE maximizes the probability of the **target class** $f_c(X')$ while minimizing the number of variables replaced.

$$L(f, c, A, X) = \left(1 - f_c(X')\right)^2 + \lambda \|A\|_1 \quad where\ X' = (I_m - A)X + AX_{dist}$$

$$\min_{A, X_{dist}} L(f, c, A, X)$$

A is a **binary diagonal matrix**:
$A_{jj} = 1 \rightarrow$ variable j replaced by the corresponding variable from X_{dist}

I_m is an identity matrix

DIGITAL

Challanges

Temporal dependency: you can’t arbitrarily perturb a single time step without considering neighbors

High dimensionality: many time points × variables

Realism / data manifold constraints: the counterfactual should “look like” a valid time series

Granularity & interpretability: we want sparse, meaningful changes (e.g. on subsequences, motifs)

Multiple objectives: validity (flip the output), proximity (stay close), sparsity (few changes), plausibility / domain constraints

DIGITAL

Counterfactual for Time Series

- Select X_{dist} from the given data[choose sample that is nearest to the given sample]
  - Need to maintain list of nearest neighbours.
- Heuristic search. –see paper.
- Ates, M., Aksar, B., Leung, V. J., & Coskun, A. K. (2021). *Counterfactual Explanations for Multivariate Time Series*. IEEE ICAPAI 2021. doi: 10.1109/ICAPAI49758.2021.9462048

DIGITAL

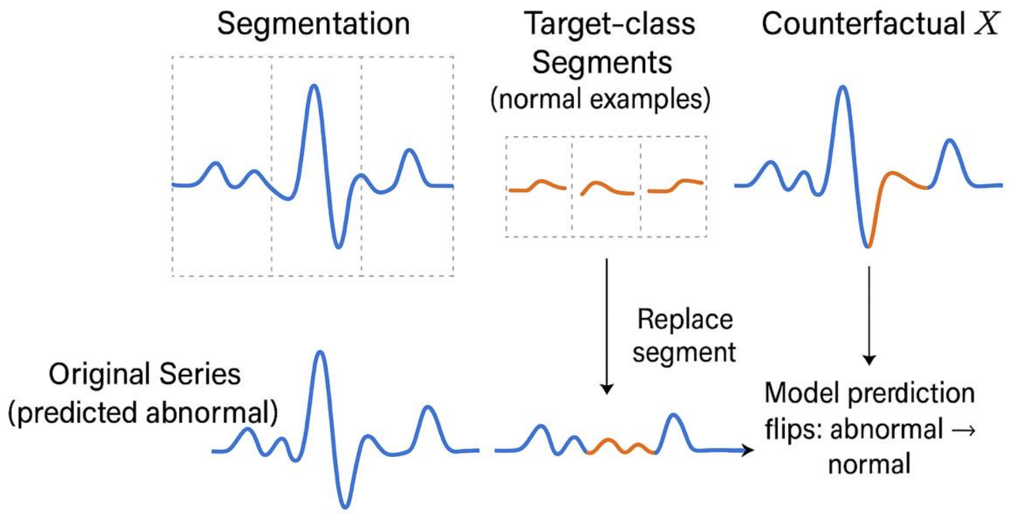
CoMTE: Counterfactual Explanations for Multivariate

- Instead of optimizing each data point (which often breaks temporal realism), CoMTE generates *plausible counterfactuals* by **replacing subsequences** (windows) of the original time series with *real segments* taken from examples of the target class.

Find

$$X' = Replace(X, S_{target}) \quad s.t.\, f(X') = y_{target}$$

where S_{target} are time windows from sequences in the target class chosen to minimize the distance $D(X, X')$ (often via Dynamic Time Warping).



Idea: Replace small subsequences of the original signal with replistic fragments from the target class until the model changes its decision.

DIGITAL

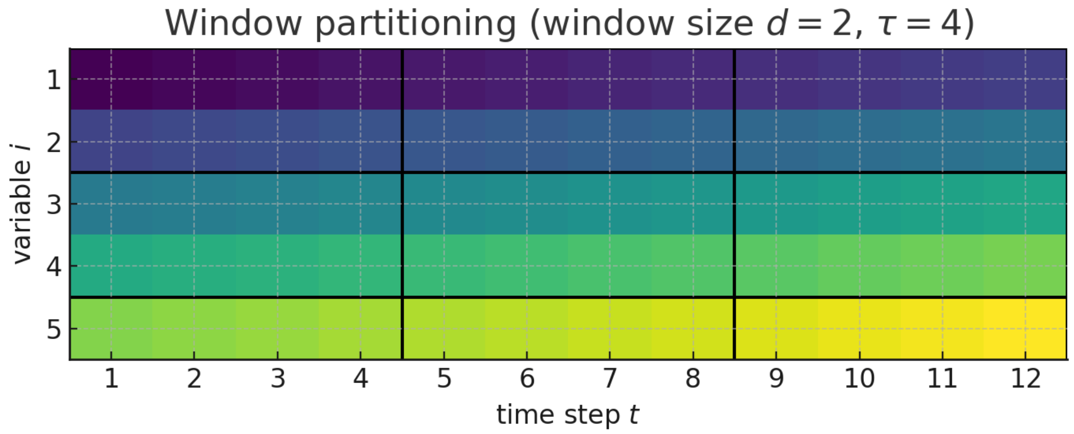
Explaining LSTM

TimeSHAP, WindowSHAp and C-SHAP

DIGITAL

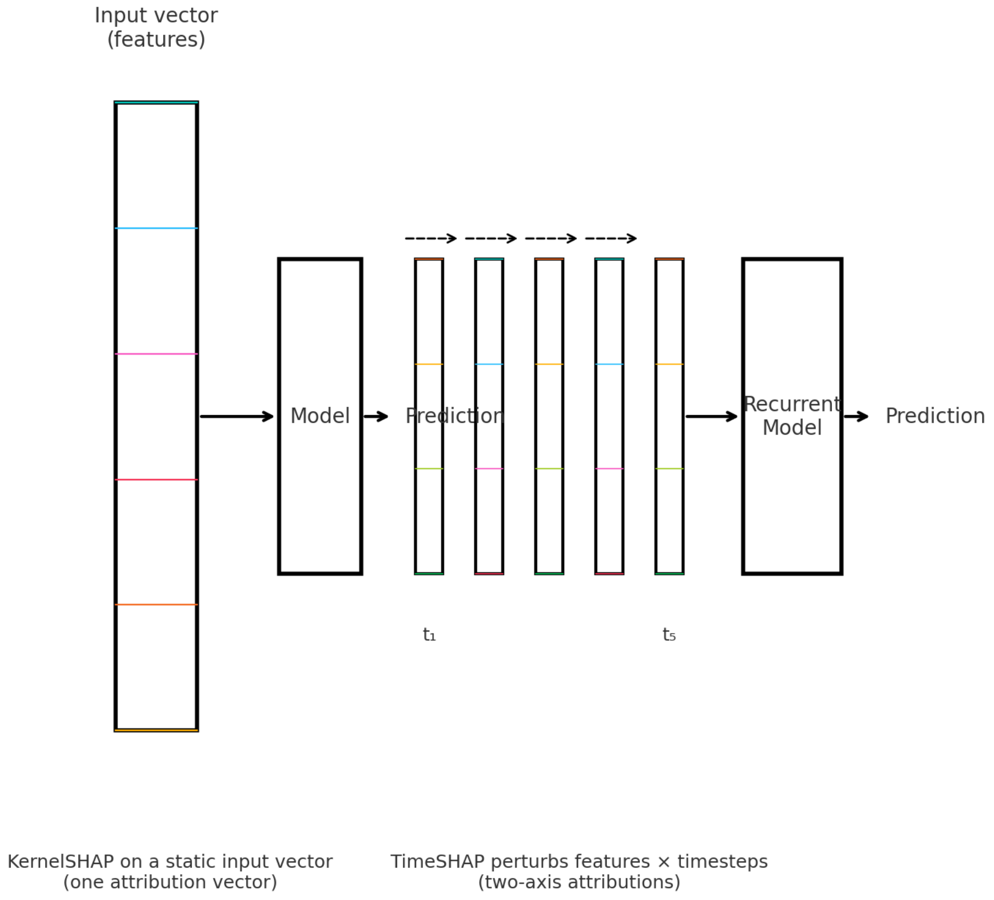
Shapley Values for Time Series

- Consider the multivariate time series $X \in \Re^{D \times L}$
- - D is the number of variables
- - L is the length of time series
- $\boldsymbol{\Delta} = \{(i,t) : 1 \leq i \leq D, 1 \leq t \leq T\}\}$ – set of all combination of time and variables.
- $\phi_{i,t} = \sum_{(S \subset \Delta \backslash \{(\mathrm{i,j})\}} \frac{|S|!(D \times L - |S| - 1)!}{(D \times L)!} [v_{X^*}(S \cup \{(i,t)\}) - v_{X^*}(S)]$



DIGITAL

TimeSHAP

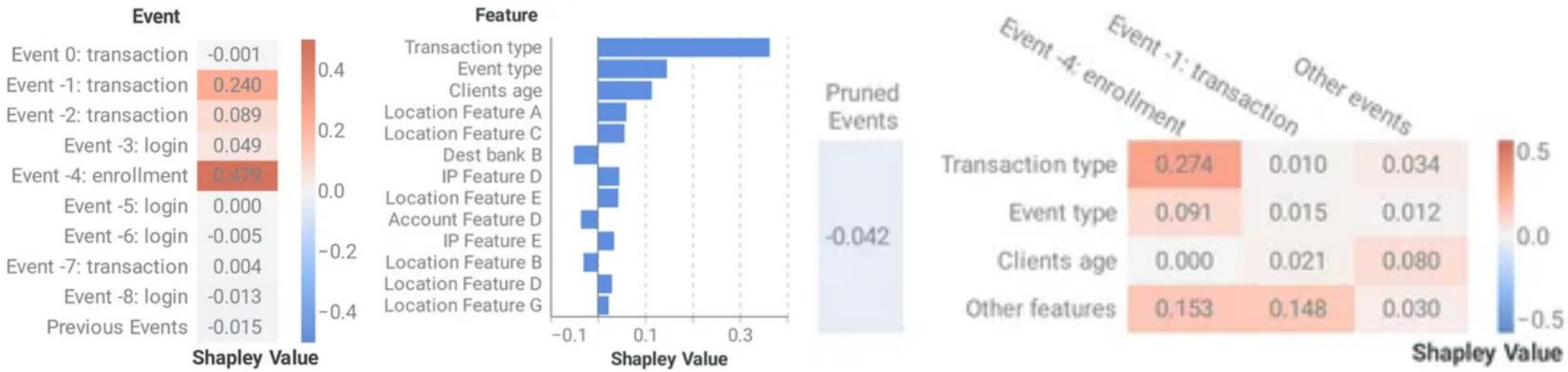


- Tabular KernelSHAP treats the whole history as **one** feature vector → loses temporal context.
- RNNs, TCNs, Transformers output predictions **because of specific features at specific timesteps**.
- **Question:** “Which past events actually drove the prediction?”
- Requires attributions on **two axes** → *variables* × *timesteps*.

TimeSHAP

Computes three levels of attribution (explanations) for a prediction:

1. **Feature-wise** (Importance of each input feature, across all events).
2. **Event-wise (Timestep)** (Importance of each sequential event).
3. **Cell-wise** (Importance of a specific feature at a specific event/timestep).



DIGITAL

https://medium.com/feedzaitech/timeshap-explaining-recurrent-models-through-sequence-perturbations-41f2324bfe5f

TimeSHAP

João Bento, Pedro Saleiro, André F. Cruz, Mário A.T. Figueiredo, and Pedro Bizarro. 2021. TimeSHAP: Explaining Recurrent Models through Sequence Perturbations. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (KDD '21). Association for Computing Machinery, New York, NY, USA, 2565–2573. https://doi.org/10.1145/3447548.3467166

- **TimeSHAP** extends KernelSHAP to sequences, producing **feature-, event- and cell-level Shapley attributions**
  - **Sequence-wide Shapley perturbations:** Extends KernelSHAP to *two* axes—features *and* timesteps—so you can ask "which past events and which variables actually drove the RNN's output?"
  - **Temporal-coalition pruning:** Groups the oldest, low-impact events into a single "background" coalition once their combined attribution falls below a tolerance η, slashing the exponential search space and runtime without losing Shapley guarantees.
  - **Cell-level zoom-in:** After isolating the few critical rows (features) and columns (events), it perturbs the individual cells at their intersections, yielding fine-grained attributions like "this unusually large transfer amount in event k triggered the fraud alert."

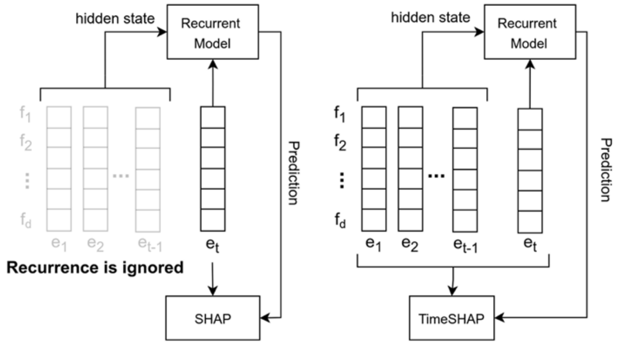


Figure 1: Current SHAP-based methods (left) only calculate attributions for a single input vector. TimeSHAP (right) applies perturbations throughout the input sequence.

DIGITAL

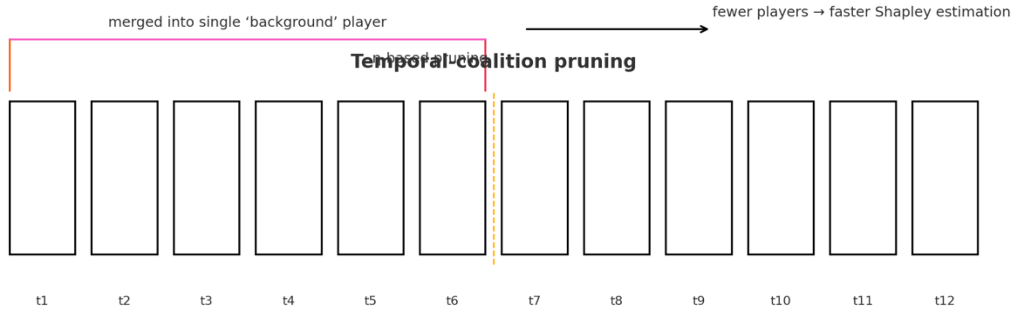
TimeSHAP

- Events are sorted by recency.
- Starting from the far past, merge events into one **background player** until

$$\sum_{j \in merged} \phi_{event_j} \leq \eta$$

where η is a user-set tolerance (e.g. 0.05).

- Reduces the exponential coalition space without violating Shapley axioms.



DIGITAL

TimeSHAP

Keep top-*k* rows (important features) and top-*k* columns (important timesteps).

Perturb only the **k^2 intersection cells** → quadratic, not exponential.

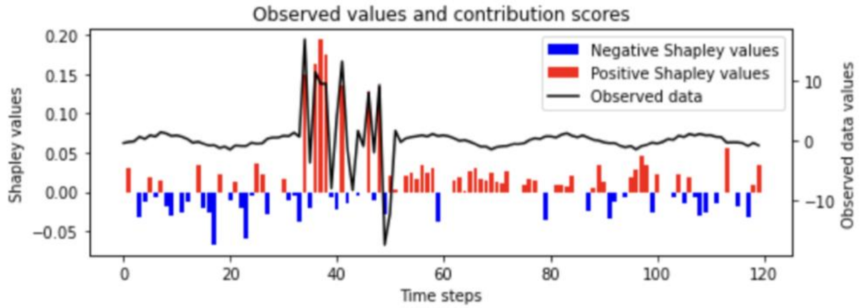
Produces fine-grained insights:

“This unusually large transfer amount at t = k triggered the fraud alert.”

DIGITAL

WindowSHAP

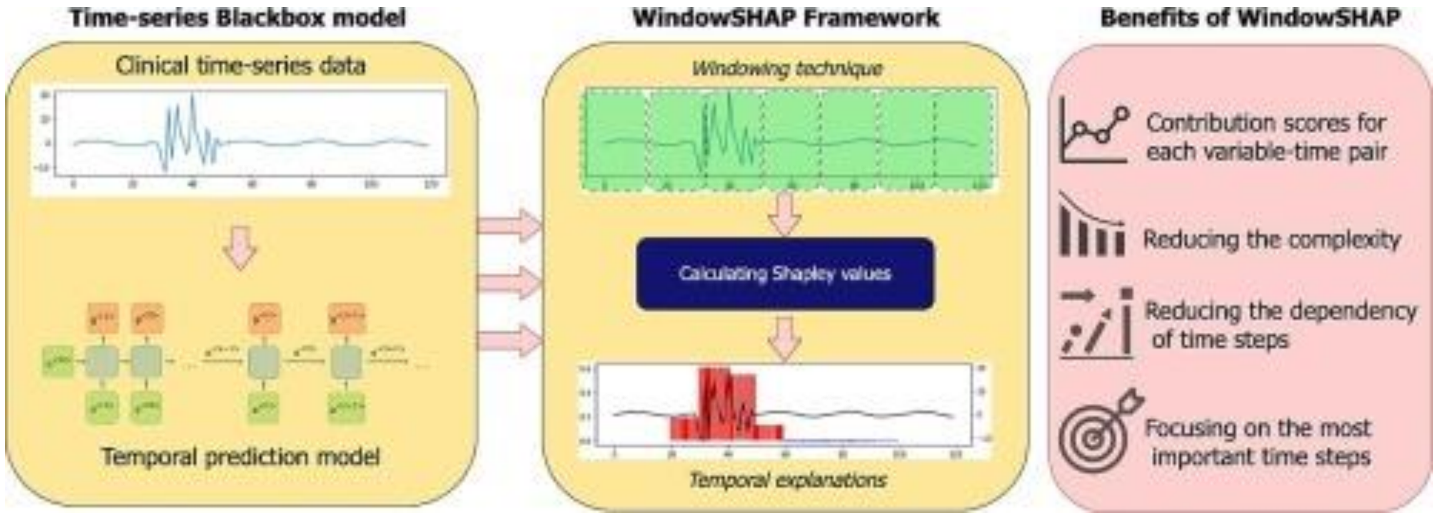
KernelSHAP



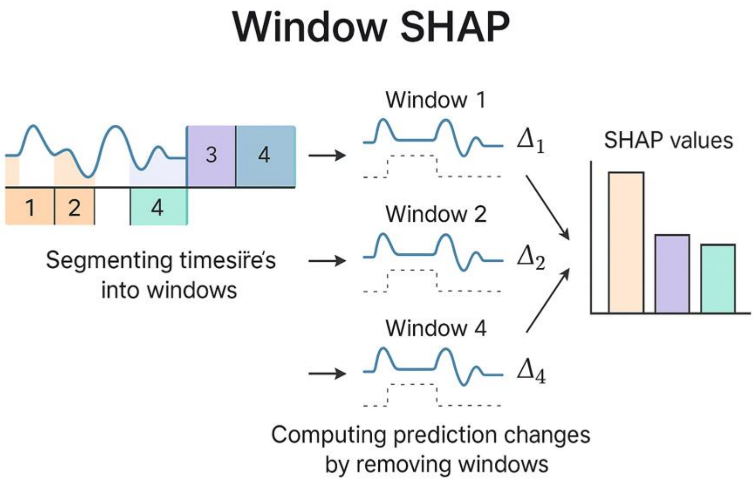
- Drawbacks (Kernel)SHAP for time series:
- - Not originally intended to be used with time-series data.
- - KernelSHAP approximates Shapley values and reduces computational time, but is still computationally expensive for high-dimensional data.
- - Sequential data points are often highly dependent. For dependent features, their joint contribution is distributed among them, resulting in many small Shapley values.
    - Difficult to draw conclusions

DIGITAL

WindowSHAP

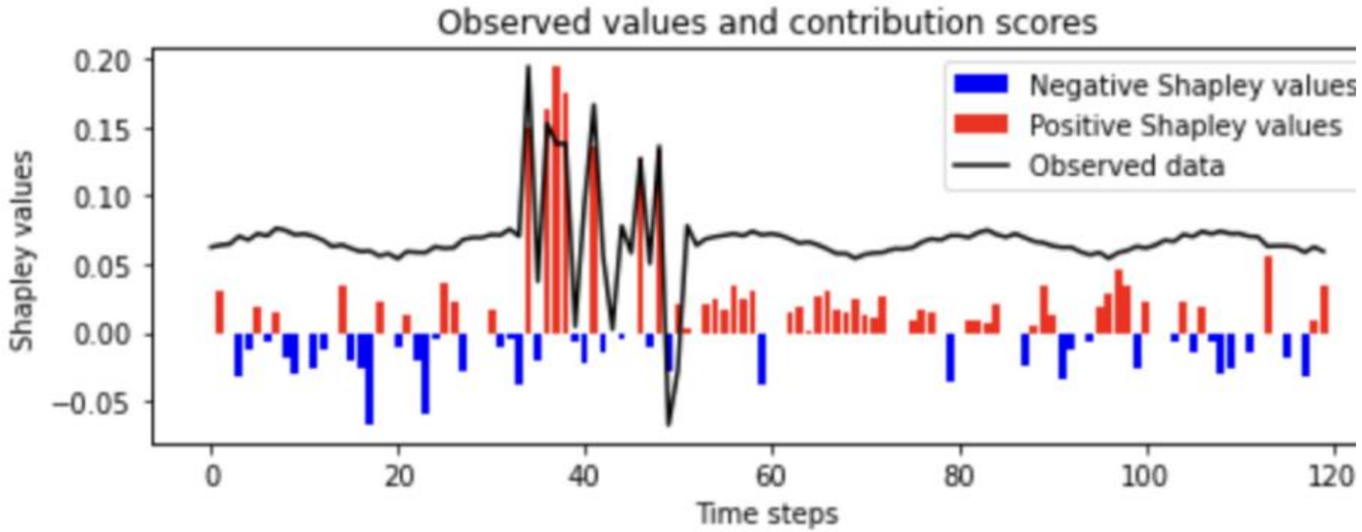


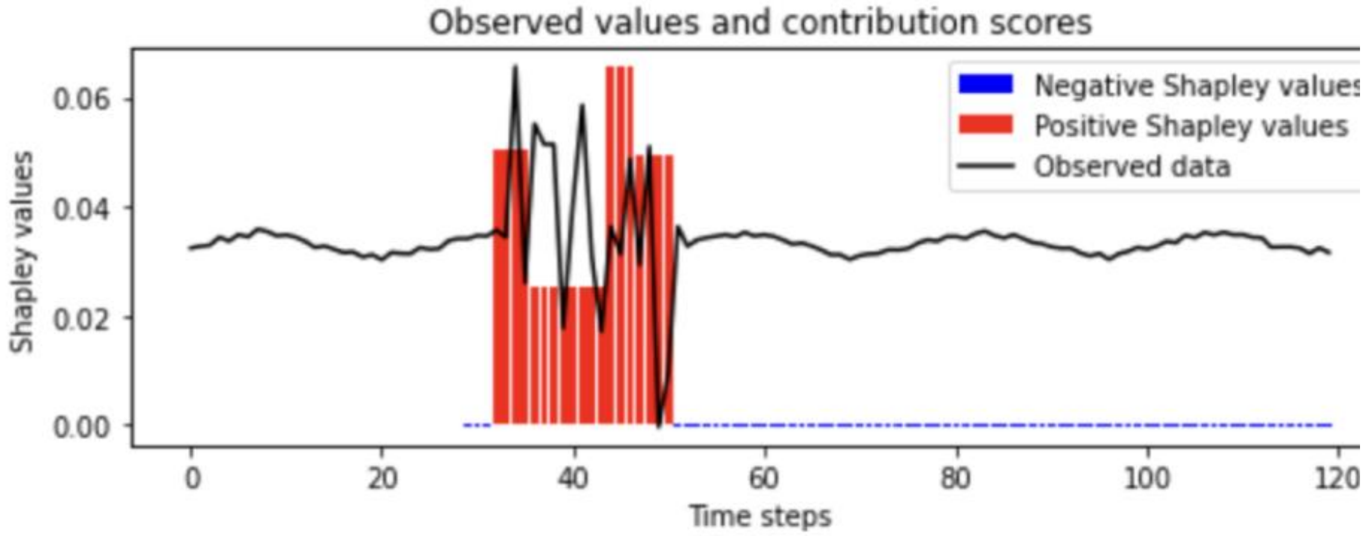
- Nayebi, A., Tipirneni, S., Reddy, C. K., Foreman, B., & Subbian, V. (2023). WindowSHAP: An efficient framework for explaining time-series classifiers based on Shapley values. *Journal of biomedical informatics*, *144*, 104438.



DIGITAL

WindowSHAP





DIGITAL

WindowSHAP

KernelSHAP:

- To mask a feature (=data point) replace it by an uninformative value
  - For example: zero, sampling from training data, …

WindowSHAP

- To mask a feature (=partition of data points) replace them all by an uninformative value
  - For example: zero, sampling from training data (subsequences), …

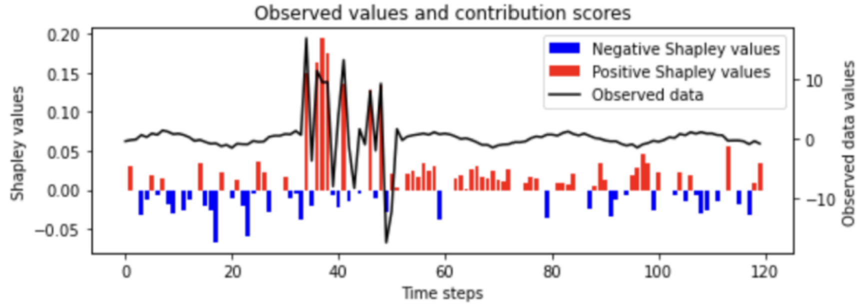
DIGITAL

WindowSHAP

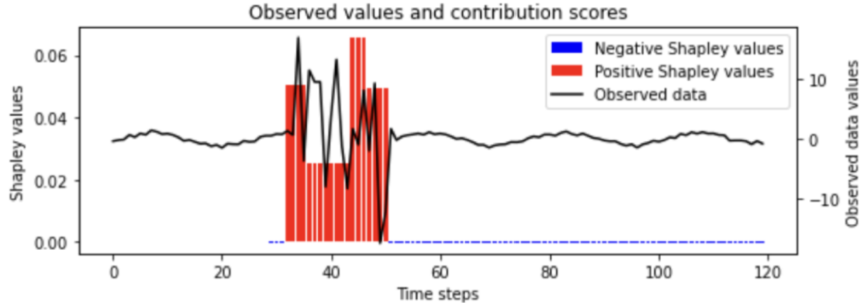
WindowSHAP solves these issues by partitioning data points and treating the partitions as features for SHAP

- Partitioning means less features, so lower computational complexity
- Partitioning balances out small SHAP values and instead leads to more meaningful explanations

KernelSHAP



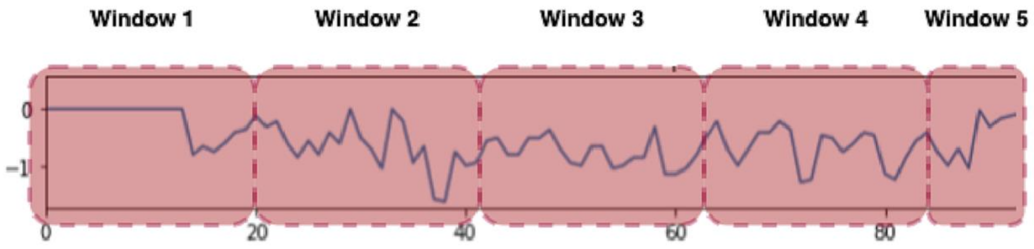
WindowSHAP



DIGITAL

WindowSHAP – partitioning

- Stationary WindowSHAP
    - Segment time series into adjacent fixed length windows



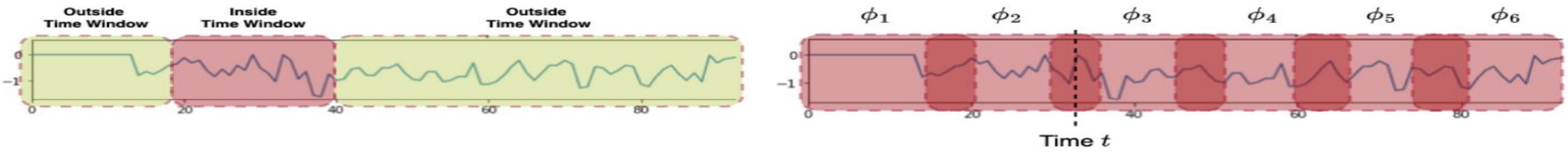
DIGITAL

WindowSHAP – partitioning

Stationary WindowSHAP

Sliding WindowSHAP

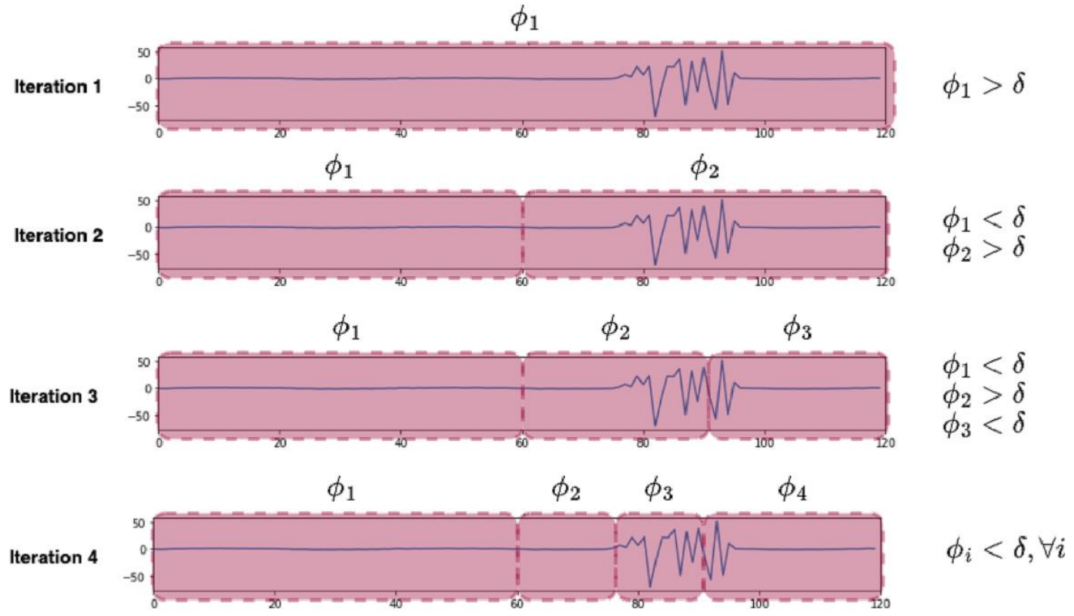
- Segment time series into overlapping fixed length windows to mitigate boundary issues
- SHAP is repeatedly applied for each segment
- Average out SHAP values for overlapping windows



DIGITAL

WindowSHAP – partitioning

- Stationary WindowSHAP
- Sliding WindowSHAP
- Dynamic WindowSHAP
  - Flexible length windows
  - Repeatedly apply WindowSHAP and split partitions with high SHAP values



DIGITAL

WindowSHAP

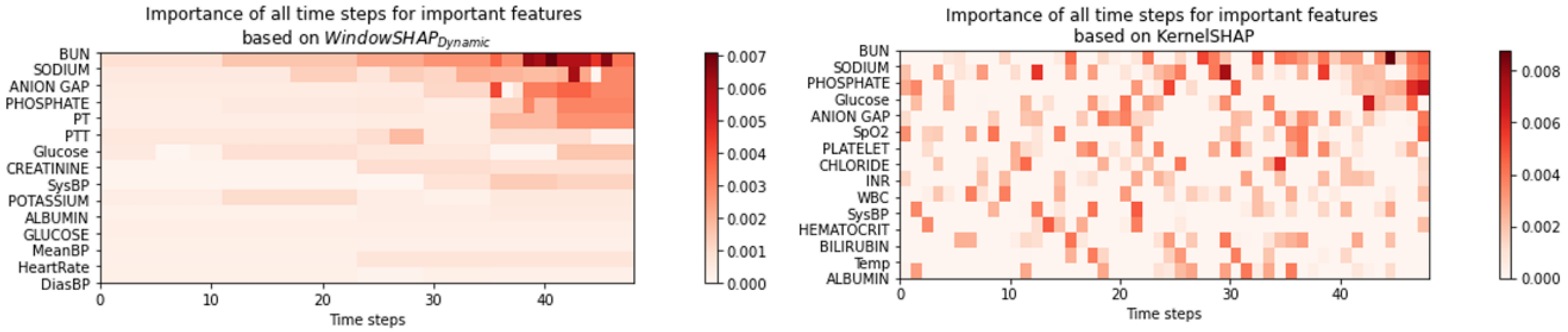
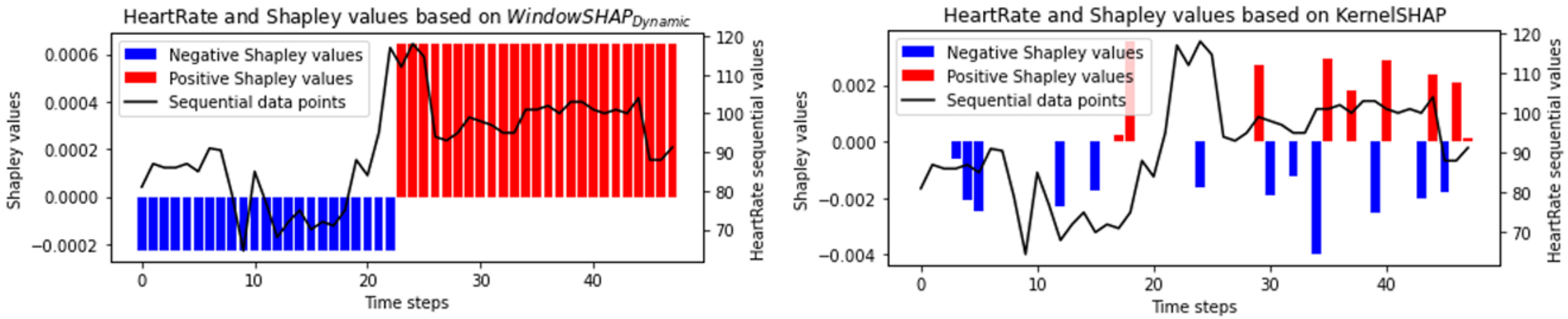


Figure 7. *Heatmaps depicting the importance of all time steps for the important features for a certain patient record from the MIMIC-III dataset. The top 15 variables depicted on the y axis are ranked according to their importance. The darker the color is, the higher the absolute value of the assigned Shapley value is.*



DIGITAL

XAI In Fraud Detection: A Causal Perspective

https://purl.utwente.nl/essays/105290

DIGITAL

The story..

- **Problem Statement:** Advanced AI fraud detection models are often too opaque
- **Challenge:** Existing explainable artificial intelligence (XAI) techniques are often not evaluated well

- **How can causal discovery techniques improve the explainability of fraud detection models?**

DIGITAL

Research Background - Causal Discovery in AI

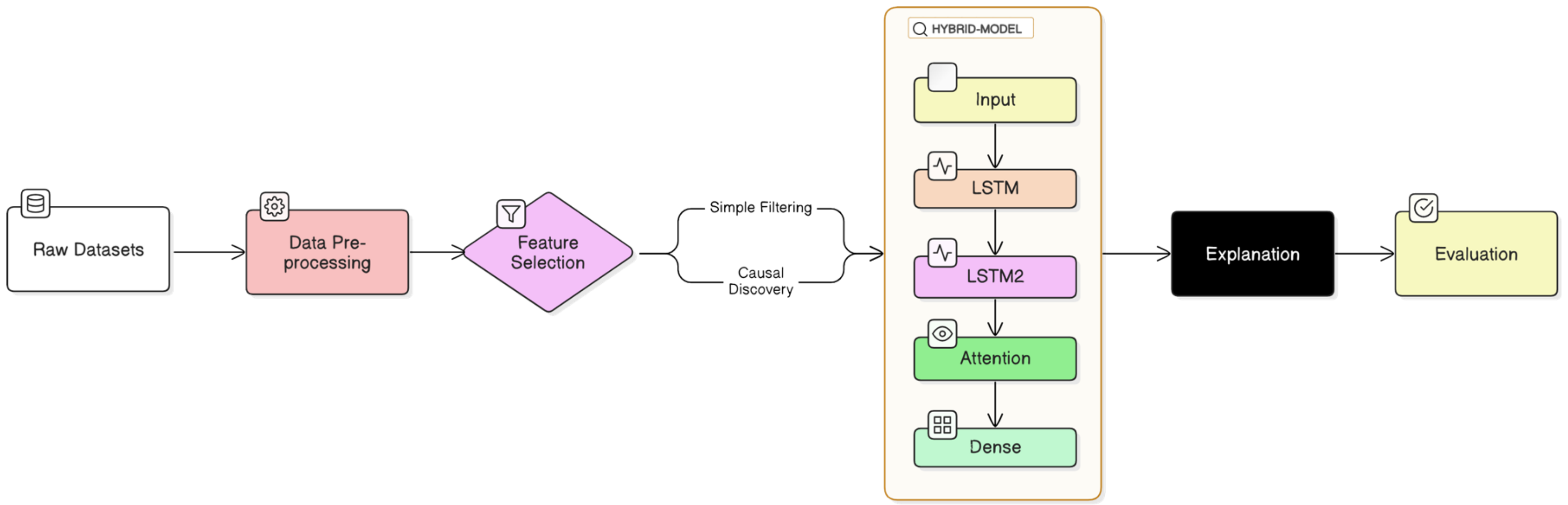
— **Why Causal Discovery?** Aims to identify true cause-effect relationships beyond correlation

Benefits:

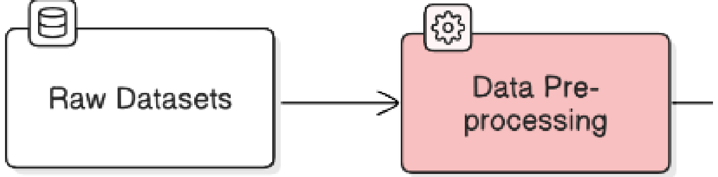
- ► Reduces spurious correlations
- ► Identifies root causes of fraud
- ► Increases model robustness

Method of choice: Constraint-based causal Discovery from heterogeneous/NOnstationary Data (CD-NOD)

Pipeline Architecture



DIGITAL

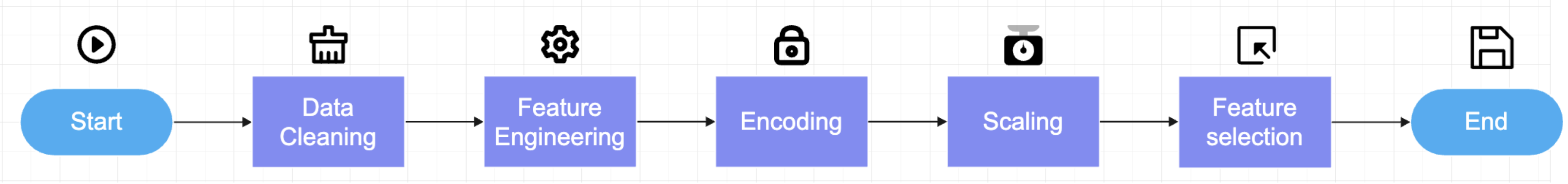


Synthetic

- Large scale, labeled transactions
- Detailed information about clients, transactions, and merchants
- Transactions over a decade

Real

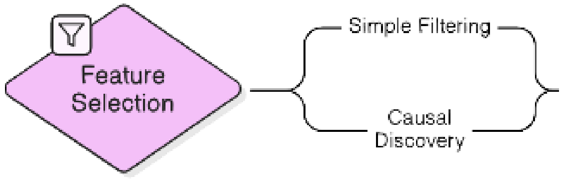
- PCA transformed features
- Anonymized - original features are unknown
- Transactions over a two-day period



DIGITAL

OF

Feature Selection



Simple filtering

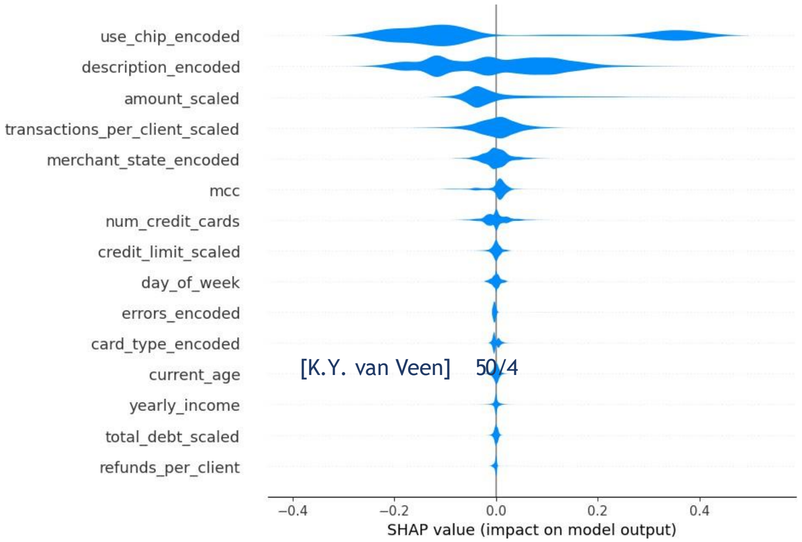
- **Techniques**: Chi-squared, ANOVA
- **Process**:
  - Select top-ranked features by the highest significance

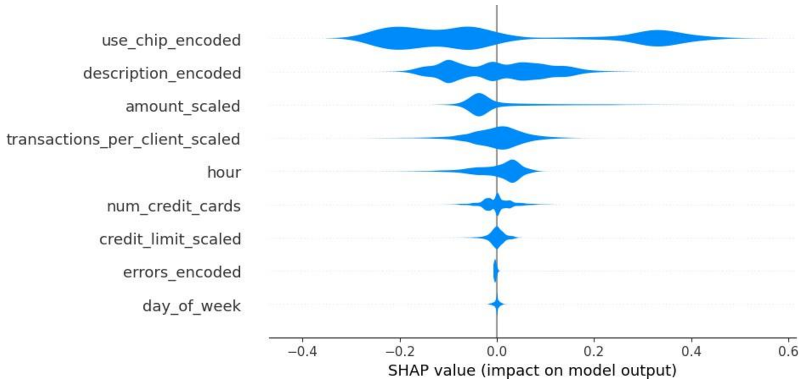
Causal (CD-NOD)

- **Process**:
  - Learns causal graph from transaction data
  - Identifies features with direct causal relationships to the fraud label
  - Takes the temporal dimension into account

Synthetic Dataset

XAI Performance:

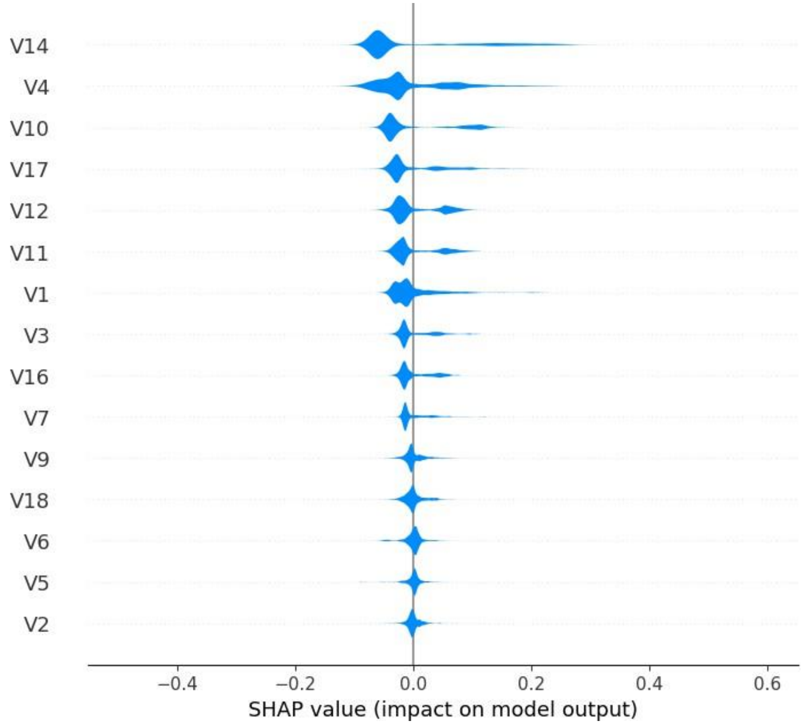


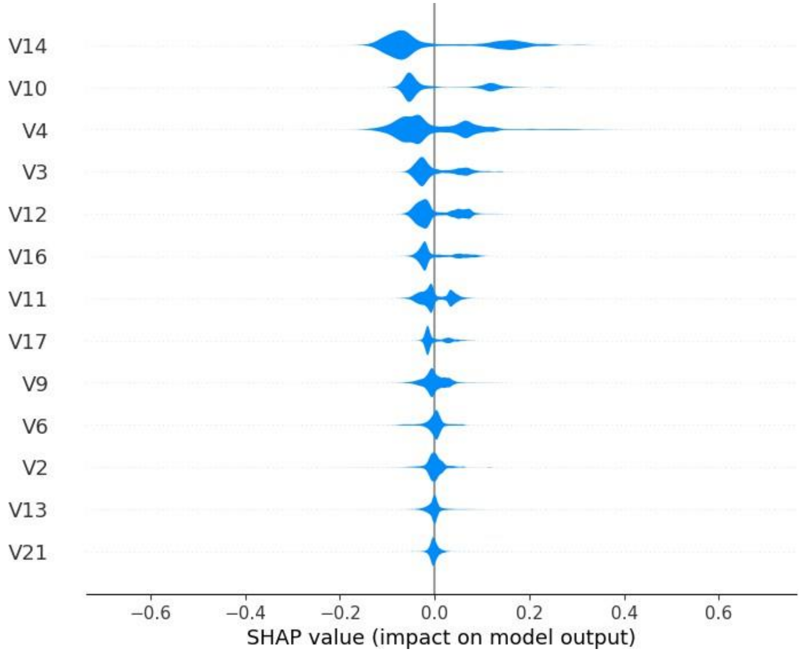


	Non-causal	Causal
Correctness	Most influential feature: 0.9216 Least influential feature: 0.7145	Most influential feature: 0.9305 Least influential feature: 0.8626
Contrasivity	Large SHAP differences for top features	Large SHAP differences for same features
Compactness	High mutual information (MI) among top features	Lower MI between features
Confidence	Wide confidence intervals for top features	Wide confidence intervals for same features
Coherence	Rank correlation: 0.4242	Rank correlation: 0.4848

Real Dataset

XAI Performance:

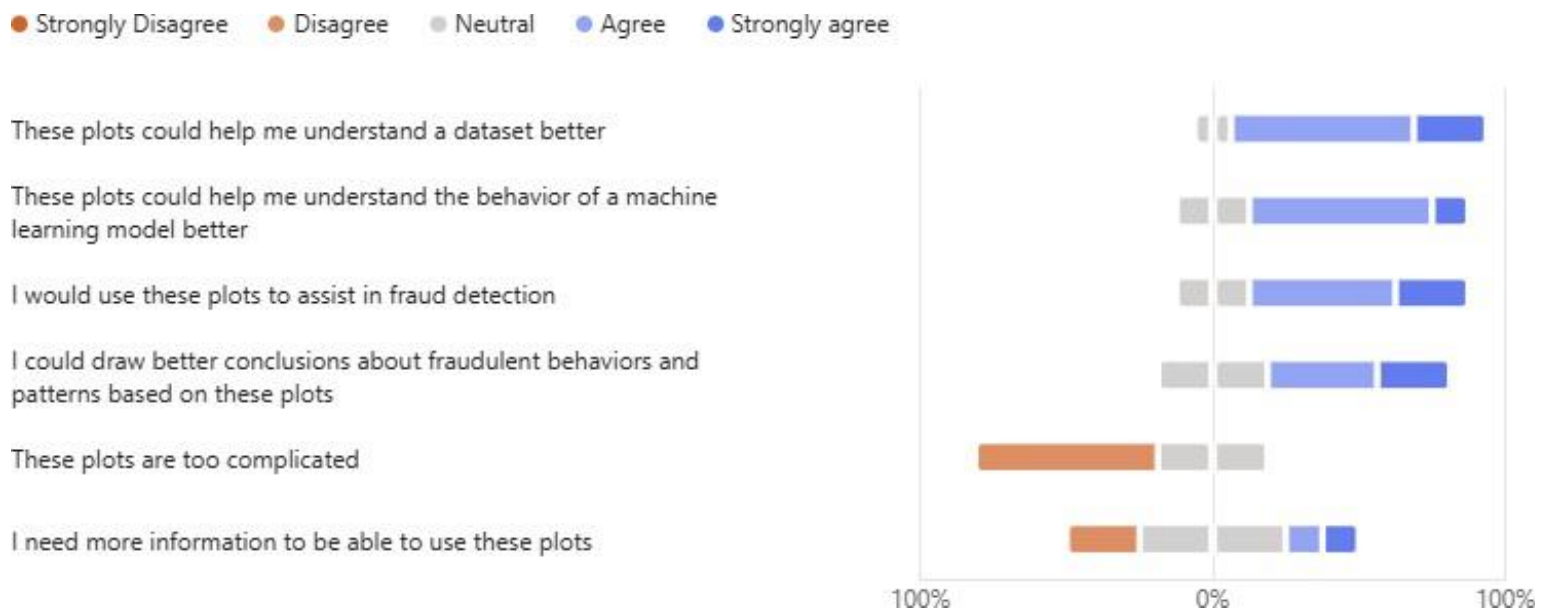




	Non-causal	Causal
Correctness	Most influential correlation: 0.2093 . Least influential correlation: 0.5698	Most influential correlation: 0.26 Least influential correlation: 0.6612
Contrasivity	SHAP differences mostly align with feature importance	Mismatch between SHAP value differences and feature importance
Compactness	Very high mutual information (MI) among features	Slightly lower MI between features
Confidence	Moderate confidence intervals	Wider intervals for most features

DIGITAL

Stakeholder Feedback



38% of respondents were not familiar with XAI explanations

DIGITAL

Interpretation of Findings

- Causal feature selection improved certain XAI metrics on synthetic data but faced limitations on anonymized real data
- The pipeline is able to integrate causal discovery with fraud detection
- Causal methods reduced feature redundancy,
- Current evaluation methods do not measure causal relationships directly
- Stakeholders indicated an interest in clear explanations

DIGITAL

Limitations & Future Directions

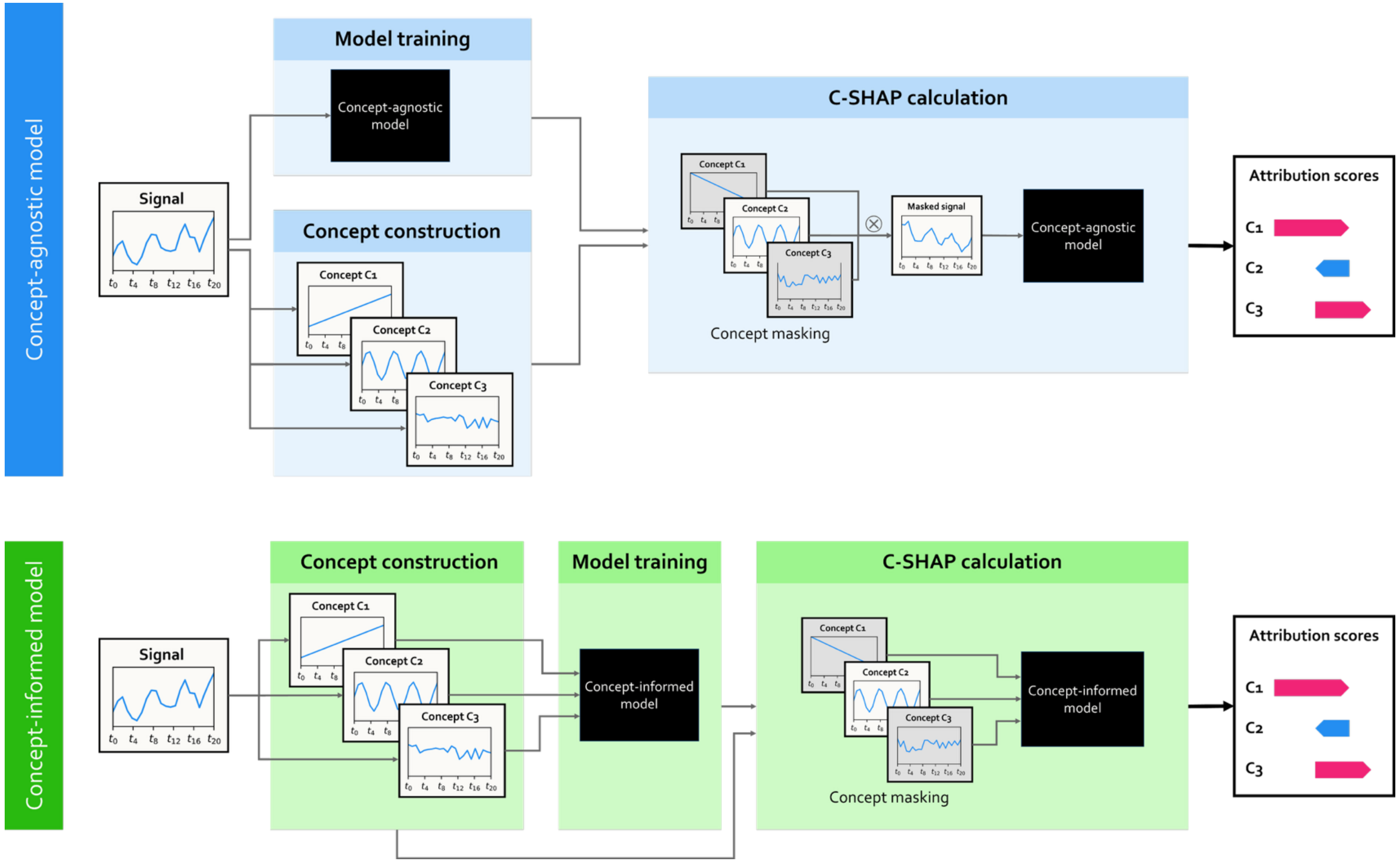
Limitations

- Limited techniques tested
- No causal ground truth
- Metrics do not measure causality
- One transaction at a time
- Limited by data quality

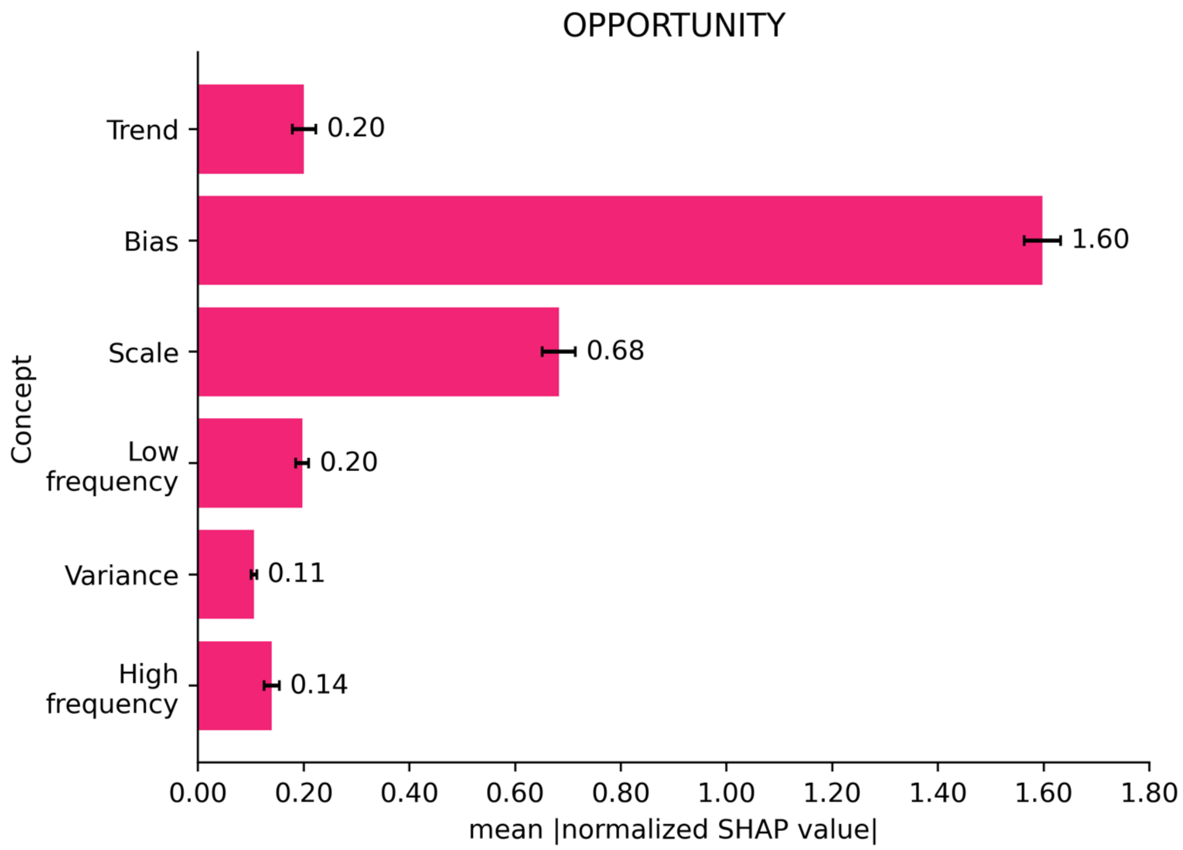
Future Directions

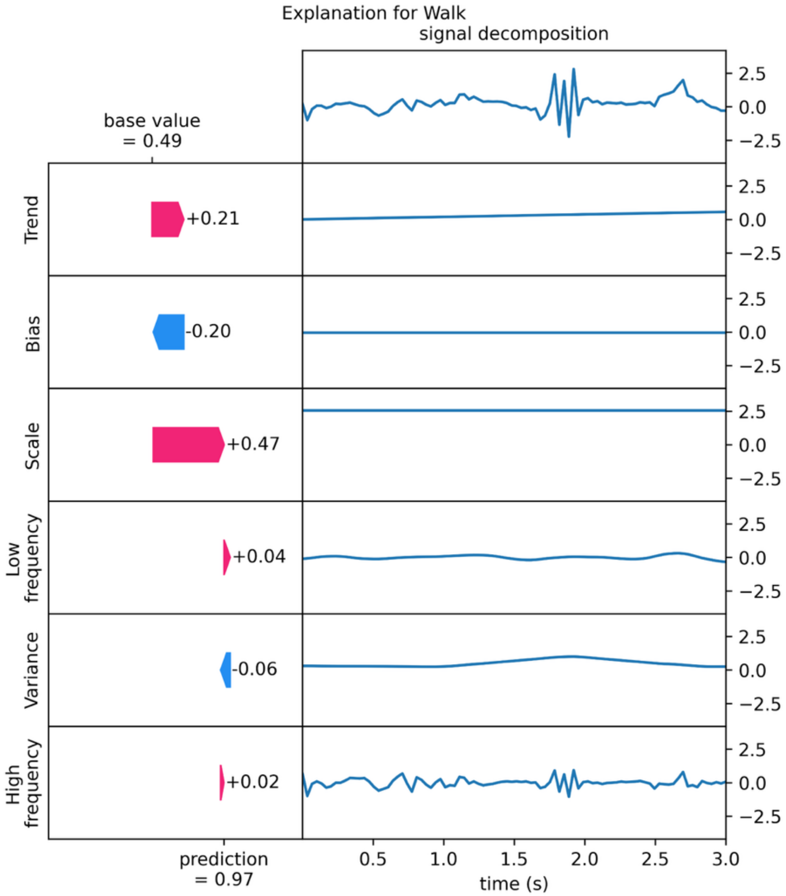
- Explore more methods
- Develop causal metrics
- Sequences of transactions
- Real-time analysis
- More types of fraud

C-SHAP
(an early version: https://arxiv.org/abs/2504.11159)



C-SHAP- some early results…





DIGITAL

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or Horizon Europe: Marie Skłodowska-Curie Actions. Neither the European Union nor the granting authority can be held responsible for them.

DIGITAL

This project has received funding from the European Union's Horizon Europe research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 101119635