Data Driven Time series causal discovery through featurization

Gian Marco Paldino[1 [0000−0002−8680−9403]] and Gianluca Bontempi[1 [0000−0001−8621−316X]]

Machine Learning Group, Computer Science Department, Universite' Libre de Bruxelles, Brussels, Belgium {gpaldino,gbonte}@ulb.ac.be

Abstract. This paper introduces a novel approach for causal discovery in time series data, focusing on the computation of asymmetric conditional mutual information within the Markov blankets of variable pairs. We frame causal discovery as a supervised learning problem, utilizing featurization to capture the asymmetric nature of causal relationships. Our methodology includes a comprehensive set of features specifically designed to detect causal signals and is benchmarked against existing state-of-the-art techniques. Empirical results validate that our method not only competes with established benchmarks but also provides new insights into causal inference dynamics. To enhance reproducibility and encourage further research, we offer an open-source Python implementation of our approach along with a benchmark dataset.

Keywords: Causal Inference · Information Theory · Machine Learning

1 Introduction

The pursuit of understanding causal relationships within time series data represents a cornerstone of scientific inquiry across diverse disciplines, from economics and social sciences to biology and environmental studies. It offers profound insights into the underlying mechanisms governing dynamic systems, enabling predictions, interventions, and policy formulations based on not just correlations but on the causal architecture of the phenomena under study.

Traditionally, causal discovery in time series has been dominated by model-based approaches in the context of Granger causality [11]. Granger causality is based on the idea that if a time series X^i "Granger-causes" X^j, then past values of X^i should contain information that helps predict X^j above and beyond the information contained in past values of X^j alone. This method has several notable limitations: it operates under the assumption that relationships between variables are linear, it suffers from the "curse of dimensionality," where accurately modeling relationships requires exponentially more data as the number of variables increases, and it relies on temporal precedence, which complicates its ability to determine causality in nearly simultaneous relationships. Most importantly, it can only suggest potential causal links based on predictive ability and thus might misinterpret correlations as causality. Despite this limitations, it is widely used in multiple fields such as econometrics [12], neuroscience [5], and climate science [24]. Several variations have been proposed, to address nonstationarity [18], nonlinearity [19], and multivariate settings [22].

Building on the traditional approach of Granger causality, it's important to recognize that this method, along with others like the Additive Noise Model (ANM, [13]) and Information Geometric Causal Inference (IGCI, [10]), predominantly operates within a bivariate framework. Specifically, the ANM assumes the cause and effect relationship between variables can be modeled by one variable being a function of another plus some additive noise, and IGCI applies information geometry principles to compare the complexity of the transformation in both potential causal directions. It uses measures such as the uniformity and entropy of the distributions of the variables to determine which scenario is more likely- X^i causing X^j or X^j causing X^i. These methods typically analyze the causal relationship between two variables at a time. However, real-world systems are often multivariate and involve interactions among multiple variables. To truly discern the causal relationships within such systems, it is critical to consider the potential confounding effects of additional variables. For example, the causal influence of one variable X^i on another X^j cannot be accurately assessed without considering the influence of all other variables in the set $X \setminus \{X^i, X^j\}$ [30], where X is the set of all variables considered. This consideration is crucial because any of these additional variables could confound the relationship between X^i and X^j, leading to incorrect inferences if not properly accounted for. [17] and [4] have succesfully leveraged the *context* to improve causal discovery.

Time series data, characterized by its temporal dependencies, high dimensionality, and potential for latent confounding factors, poses unique challenges that bivariate models may not fully address. This has led to the development of more sophisticated methods [7] tailored to the intricacies of time series analysis. Recent advancements like VarLiNGaM [14], DYNOTEARS [23], and PCMCI [28] represent significant strides in this direction, extending the foundational principles of causality into the dynamic and often complex domain of time series. VarLiNGaM applies the principles of LiNGaM to time series data, incorporating temporal information to uncover causal relationships over time. DYNOTEARS, on the other hand, integrates dynamic Bayesian network principles with structure learning to model and infer causal dynamics in time-evolving data. PCMCI, combines conditional independence tests with momentary information criteria to effectively identify causal structures in high-dimensional time series datasets. Each of these methods addresses the limitations of traditional bivariate and static causal discovery techniques, offering more nuanced and applicable insights for time series analysis.

Our proposed method, rooted in [4,3], aims to address the challenges of causal discovery by reframing it as a supervised learning problem. Unlike previous data-driven approaches, such as the kernel-embedding method by [17], our method incorporates context by examining the estimated Markov blanket of the variables under study. This allows us to harness a broader spectrum of information and leverage the influence of adjacent variables. By computing asymmetric features (also called descriptors) based on information theory, we gain critical insights into the causal structure.

Data-driven approaches for causal discovery have not received extensive attention in the literature, likely due to the need for hard-coding features [17], the high dimensionality of the datasets involved, and the associated computational burden. However, we aim to demonstrate that interest in data-driven causal discovery can be revitalized.

By adequately crafting features and improving the estimation of quantities of interest, our method shows that it is possible to make these approaches efficient and effective. Specifically, we build on top of [4] by proposing several variations to better leverage the temporal dependencies of time series data, reduce the dimensionality of the problem at hand, and effectively reaching state of the art performance with limited computing.

The main contributions of this work are the following:

- We introduce a novel featurization technique leveraging estimated asymmetric conditional mutual information terms among variable pairs and their Markov blankets.
- We develop a method that utilizes temporal dynamics to estimate the Markov blanket, eliminating the need for preliminary estimates and demonstrating enhanced performance and efficiency.
- We adopt an alternative estimation method for mutual information from the literature, showing significant improvements over previous approaches.
- We conduct a comprehensive comparison of our approach against contemporary state-of-the-art methods in time series causal discovery.
- We provide an open-source Python implementation of our causal discovery method, along with a benchmark dataset [1]

The remaining of the manuscript is organized as follows: Section 2 introduces related works, while 3 provides the necessary theoretical background. Section 4 presents the contribution. Experiments can be found in 5, and are divided in settings (Section 5.1), and Results (Section 5.2). Section 6 concludes the paper.

2 State of the art

The literature on causal discovery from time series data is vast and encompasses a wide array of methodologies, each designed to tackle various aspects of the problem under different assumptions. These methodologies can be broadly classified into several families [1], each with distinct characteristics and applications:

- Granger-based Methods: this family of methods is based on the idea that past values of one variable help predict the current value of another variable if they are causally connected. The simplest implementation is the Pairwise Granger causality test [11], whose null hypothesis posits that X^i does not Granger cause X^j. A statistical test is performed to demonstrate that the inclusion of past values of X^i significantly enhances the prediction of X^j when past values of X^j are also used as regressors. If the resulting p-values from the test are below the threshold set for statistical significance, the null hypothesis is rejected.
- Constraint-Based Methods: these methods rely on statistical tests for conditional independence to infer causal relationships from the data, and build a causal graph by systematically adding or removing edges based on the independence tests. The Peter and Clark algorithm [31] performs conditional independence tests between pair of variables, using increasingly larger conditioning sets. In a large-variate time series context, when lagged variables need also to be tested, the problem becomes

[1] The code and data adopted in this paper are available at https://github.com/gmpal/TD2C

easily intractable, especially for all possible combinations q of conditioning sets. To tackle this problem, more advanced algorithms such as PCMCI (Peter and Clark Momentary Conditional Independence, [28]) have been proposed. This algorithm is divided into two phases: an initial PC_1 phase, where a variation of the PCstable [7] algorithm is applied, only testing the p parents with strongest dependency, that is, restricting the maximum number of combinations $q_{\max}$ per iteration to $q_{\max} = 1$. The output of PC_1 is a superset of parents $\widehat{\mathcal{P}}\left(X_t^j\right)$ for each variable X^j at each time-step t. This superset is refined in the MCI phase, where the following hypotheses are tested, for each possible couple:

$$\text{MCI: } X_{t-\tau}^i \not\perp\!\!\!\perp X_t^j \mid \widehat{\mathcal{P}}\left(X_t^j\right) \setminus \left\{X_{t-\tau}^i\right\}, \widehat{\mathcal{P}}_{p_X}\left(X_{t-\tau}^i\right) \tag{1}$$

This method is particularly effective in disentangling direct from indirect influences, even in the presence of latent confounders and over large sets of variables.

– Noise-Based Approaches: they exploit the non-Gaussian nature of the data to distinguish between cause and effect based on the independence of the residuals (noise) when one variable is regressed on another. Considering a system where each variable in X is linearly influenced by all other variables in X through the coefficients matrix B, plus an independent non-Gaussian noise component e, we can write $X = BX + e$. If we define $A = (I - B)^{-1}$, this can be written as $X = Ae$. If the available data X is a linear, invertible mixture of non-Gaussian independent disturbance variables, it can be shown that the mixing matrix A is identifiable and this process defines the standard linear Independent Component Analysis (ICA, [8]) model. Solving for A through ICA is the core aspect of the LiNGAM [29] family of methods, whose extensions to time series data include VARLiNGAM [14]. In [14], the temporal dynamics are modeled as

$$X_t = \sum_{\tau=1}^{k} B_\tau X_{t-\tau} + e_t$$

where k is the number of time-delays used, that is, the order of the autoregressive model, B_τ, $\tau = 1, \ldots, k$ are $n \times n$ matrices, and $e(t)$ is the non-gaussian disturbance.

– Score-Based Strategies: these techniques use a scoring criterion to evaluate and compare different causal models. For example, DyNOTEARS [23] frames the problem of causal discovery as the following optimization problem:

$$\min_{W,A} \frac{1}{2n} \|X_t - X_t W - Y W^*\|_F^2. \text{ s.t. } W \text{ is acyclic}$$

Where X_t, represents the current values of the variables under study at time t, while Y encompasses the values of these variables at previous time points, up to a defined lag k. Essentially, X_t captures the present, while Y captures the past, and both are used to estimate the causal relationships between time series data points through the matrices W and W^*. The matrix W represents the contemporaneous causal relationships among variables, with each entry W_{ij} indicating the direct influence of variable j on variable i at the same time point. The requirement that W is acyclic is crucial because it ensures that there are no cyclic dependencies among

the variables, which is a fundamental assumption for valid causal inference. This acyclicity implies that it is possible to order the variables such that no variable is influenced by another variable that comes later in the order, effectively ruling out any possibility of a variable causing itself either directly or through a cycle of other variables. Mathematically, acyclicity is enforced in the optimization by using the constraint that the trace of the exponential of the Hadamard product $W \circ W$ (which represents element-wise multiplication) minus the dimension of W equals zero, $\text{tr}(e^{W \circ W}) - d = 0$. This condition is derived from graph theory, where the trace of the exponential of a matrix can be used to count cycles in a graph [33]. Specifically, if W contains cycles, the exponential will produce non-zero off-diagonal terms, leading to a non-zero trace. On the other hand, the edges in W^* go only forward in time and thus they do not create cycles. It is important to remark the difference between W^* and $B_{t-\tau}$: the former represents the complete mixing matrix which encapsulates all influences among the observed variables over all time lags $B_{t-\tau}$. Where X_t is the input time series vector, Y is the lagged time series vector, and W and A are the contemporaneous and time-lagged weighted adjacency matrices, respectively, to be estimated. The aciclicity constraint is transformed into an equality constraint since the function $h(W) = \text{tr}\, e^{W \circ W} - d$ satisfies $h(W) = 0$ if and only if W is acyclic . Furthermore, sparsity of W and W^* is enforced by introducing ℓ_1 penalties in the objective function and the final problem is solved using the augmented Lagrangian method.

It is essential to also highlight recent algorithmic developments that have demonstrated strong performances in public benchmarks, such as SLARAC (Subsampled Linear Auto-Regression Absolute Coefficients), QRBS (Quantiles of Ridge regressed Bootstrap Samples), LASAR (Lasso Auto-Regression), and SELVAR (Selective autoregressive model) [32]. However, as pointed out by [26], data scale and marginal variance may carry information about the data generating process. This information can dominate benchmarking results, such as, for example, the outcome of the NeurIPS Causality 4 Climate competition [Runge et al., 2020]. Here, the magnitude of regression coefficients was informative about the existence of causal links such that ordinary regression-based methods on raw data outperformed causal discovery algorithms.

For a more comprehensive review of these methods and to understand the full scope of the field, we invite readers to consult the detailed discussions available in the literature [1].

3 Background

In the following, we consider $\mathbf{z}_i$, $\mathbf{z}_j$, $\mathbf{z}_k$ three continuous random variables from a n-variate distribution $\mathbf{Z} = [\mathbf{z}_1, \ldots, \mathbf{z}_n]$ with a joint Lebesgue density. (2) defines the mutual information between $\mathbf{z}_i$ and $\mathbf{z}_j$, while (3) defines the conditional mutual information between $\mathbf{z}_i$ and $\mathbf{z}_j$ given $\mathbf{z}_k$. Both are written in terms of the respective probabilistic density functions $p(z_i)$, $p\left(z_j\right)$, $p\left(z_i, z_j\right)$, etc. We adopt the convention $0 \log \frac{0}{0} = 0$.

$$I\left(\mathbf{z}_i; \mathbf{z}_j\right) = \iint \log \frac{p\left(z_i, z_j\right)}{p\left(z_i\right) p\left(z_j\right)} p\left(z_i, z_j\right) dz_i dz_j \tag{2}$$

$$I\left(\mathbf{z}_i;\mathbf{z}_j \mid \mathbf{z}_k\right)=\iiint \log \frac{p\left(z_i,z_j \mid z_k\right)}{p\left(z_i \mid z_k\right) p\left(z_j \mid z_k\right)} p\left(z_i,z_j,z_k\right) dz_i dz_j dz_k \tag{3}$$

Mutual information quantifies the amount of information obtained about one random variable through observing another random variable. Formally, the mutual information between two variables, $\mathbf{z}_i$ and $\mathbf{z}_j$, is zero if and only if these variables are statistically independent, i.e., $I(\mathbf{z}_i;\mathbf{z}_j)=0$ if $\mathbf{z}_i \perp\!\!\!\perp \mathbf{z}_j$. This relationship underscores the utility of mutual information as a measure of dependence.

The Markov Blanket $\mathbf{M}_i$ of a variable $\mathbf{z}_i$ is the set of variables that *shield* the rest of the system from that specific variable. In other words, when conditioning on $\mathbf{M}_i$, $\mathbf{z}_i$ becomes conditionally independent of all the remaining variables. This means that it is possible to describe this structural notion in terms of conditional mutual information as follows: $\mathbf{M}_i$ is the smallest subsets of variables from $\mathbf{Z} \setminus \mathbf{z}_i$ such that:

$$I\left(\mathbf{z}_i;(\mathbf{Z}\setminus(\mathbf{M}_i \cup \mathbf{z}_i)) \mid \mathbf{M}_i\right)=0$$

This formulation is equivalent to $\mathbf{M}_i=\mathbf{C}_i \cup \mathbf{E}_i \cup \mathbf{S}_i$ where $\mathbf{C}_i=\{\mathbf{z}_c|\mathbf{z}_c \rightarrow \mathbf{z}_i\}$ is the set of variables that have direct arrows pointing to $\mathbf{z}_i$ (causes), $\mathbf{E}_i=\{\mathbf{z}_e|\mathbf{z}_i \rightarrow \mathbf{z}_e\}$, are the set of variables to which $\mathbf{z}_i$ has a direct causal influence, i.e., where $\mathbf{z}_i$ is the parent (effects) and $\mathbf{S}_i=\{\mathbf{z}_s|\exists \mathbf{z}_j : \mathbf{z}_s \rightarrow \mathbf{z}_j \leftarrow \mathbf{z}_i \text{ and } \mathbf{z}_s \neq \mathbf{z}_i\}$, are the variables that are not direct causes or effects of $\mathbf{z}_i$ but are connected to $\mathbf{z}_i$ through a common effect (spouses). Members of $\mathbf{C}_i$, $\mathbf{E}_i$, $\mathbf{S}_i$ are denoted as $\mathbf{c}_i^{(k_c)}$, $\mathbf{e}_i^{(k_e)}$, $\mathbf{s}_i^{(k_s)}$, respectively, and k_c, k_e, k_s identify multiple elements from each set.

These sets help us understand the conditional independence relationships within a directed acyclic graph (DAG) [25]. As illustrated in Figure 1:

- $c_i^{(1)} \rightarrow z_i \rightarrow z_j$ is called a *chain*: when conditioning on z_i, $c_i^{(1)}$ and z_j become independent, because the mediator z_i is observed.
- The structure $e_i^{(1)} \leftarrow z_i \rightarrow z_j$ is called a *fork*: when we condition on z_i, $e_i^{(1)}$ and z_j become independent due to the common cause z_i being observed.
- The structure $z_j \rightarrow z_i \leftarrow c_j^{(1)}$ is called a *collider*: conditioning on z_j (or on a descendant of z_j) introduces a dependence between z_i and $c_j^{(1)}$ because it opens a path through the common effect z_j .

Authors in [4] define a *dependency descriptor* of the pair $(\mathbf{z}_i,\mathbf{z}_j)$ as a function $d(i,j)$ of the distribution of $\mathbf{Z}$, symmetric if $d(i,j)=d(j,i)$, asymmetric otherwise. Examples of symmetric descriptors include the correlation $\rho(i,j)$ or the mutual information $I(i,j)$. Conditional mutual information, on the other hand, allows to create several asymmetric descriptors. In the context of Figure 1, consider the structural configuration given by $z_i \rightarrow z_j \leftarrow c_j^{(1)}$ and $c_i^{(2)} \rightarrow z_i \rightarrow z_j$. When conditioning on z_j, a dependency between z_i and $c_j^{(1)}$ emerges, whereas conditioning on z_i results in the independence of z_j and $c_i^{(2)}$. This can be generalized as:

$$\mathbf{z}_i \rightarrow \mathbf{z}_j \iff \mathbf{z}_i \not\perp\!\!\!\perp \mathbf{c}_j^{(k)} \mid \mathbf{z}_j \text{ and } \mathbf{z}_j \perp\!\!\!\perp \mathbf{c}_i^{(k)} \mid \mathbf{z}_i \quad \forall k \tag{4}$$

From this approach, it is possible to derive a collection of (conditional) mutual information terms, such as $I\left(\mathbf{z}_i;\mathbf{c}_j^{(k)} \mid \mathbf{z}_j\right)$, or $I\left(\mathbf{e}_i^{(k)};\mathbf{c}_j^{(k)} \mid \mathbf{z}_j\right)$. These quantities are greater

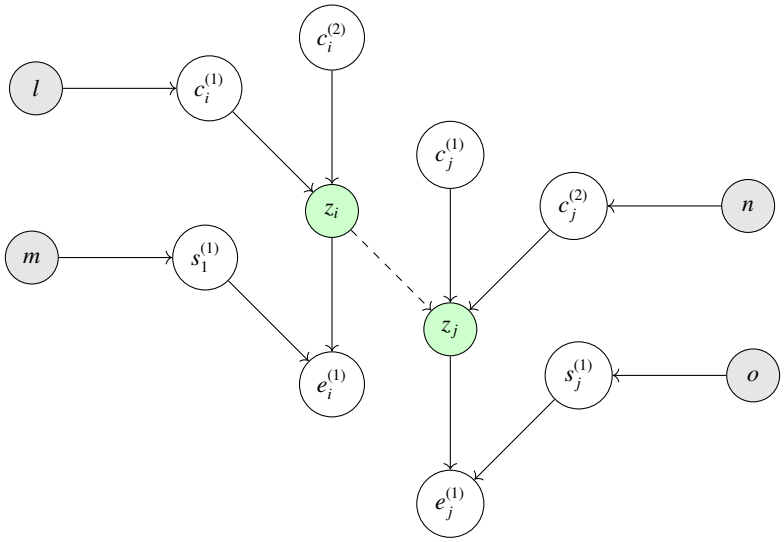

Fig. 1. The figure illustrates the concept of a Markov Blanket and its components within a Directed Acyclic Graph (DAG). The DAG depicted here showcases a section of a more extensive network, highlighting the variables z_i and z_j, where z_i is a direct cause of z_j. The Markov Blankets for both z_i and z_j are visualized, revealing the variables that insulate the rest of the system from these specific variables. Parents $c_i^{(1)}, c_i^{(2)}$ and $c_j^{(1)}, c_j^{(2)}$, children $e_i^{(1)}$ and $e_j^{(1)}$, and spouses $s_i^{(1)}$ and $s_j^{(1)}$, are specified. The number of variables in the Markov blanket can vary. Variables l, m, n, o are additional variables that represent the remaining network.

then zero, while their counterparts computed replacing i and j are null. Additional descriptors can be computed with similar reasoning:

$$\begin{aligned} d_a^{(k)}(i,j) &= I\left(\mathbf{z}_i; \mathbf{c}_j^{(k)} \mid \mathbf{z}_j\right) > 0, \quad d_a^{(k)}(j,i) = I\left(\mathbf{z}_j; \mathbf{c}_i^{(k)} \mid \mathbf{z}_i\right) = 0 \\ d_b^{(k)}(i,j) &= I\left(\mathbf{e}_i^{(k)}; \mathbf{c}_j^{(k)} \mid \mathbf{z}_j\right) > 0, \quad d_b^{(k)}(j,i) = I\left(\mathbf{e}_j^{(k)}; \mathbf{c}_i^{(k)} \mid \mathbf{z}_i\right) = 0 \\ d_c^{(k)}(i,j) &= I\left(\mathbf{c}_i^{(k)}; \mathbf{c}_j^{(k)} \mid \mathbf{z}_j\right) > 0, \quad d_c^{(k)}(j,i) = I\left(\mathbf{c}_j^{(k)}; \mathbf{c}_i^{(k)} \mid \mathbf{z}_i\right) = 0 \\ d_d^{(k)}(i,j) &= I\left(\mathbf{z}_j; \mathbf{c}_i^{(k)}\right) > 0, \quad d_d^{(k)}(j,i) = I\left(\mathbf{z}_i; \mathbf{c}_j^{(k)}\right) = 0 \end{aligned}$$

As clearly stated in [4] and [3], these descriptors already presuppose the ability to differentiate between causes and effects within the Markov Blanket of a given variable, which is the causal discovery process itself. However, it is possible to show that for any generic component $\mathbf{m}^{(k)}$ of the Markov Blanket, the overall distribution of terms related to $\mathbf{m}^{(k)}$ forms a mixture of three subpopulations: causes, effects, and spouses. The properties of these subpopulations influence the overall mixture's characteristics. For example, consider the quantity $I(\mathbf{z}_i; \mathbf{m}_j^{(k_j)} \mid \mathbf{z}_j)$, where $\mathbf{m}_j^{(k_i)}$ is a member of the set $M_j \backslash \mathbf{z}_i$, and call D_{zmz} the corresponding population. The distributions of the populations $D_{zmz}(i,j)$ and $D_{zmz}(j,i)$ will differ as follows:

$$\begin{cases} I\left(\mathbf{z}_i; \mathbf{m}_j^{(k_j)} \mid \mathbf{z}_j\right) > I\left(\mathbf{z}_j; \mathbf{m}_i^{(k_i)} \mid \mathbf{z}_i\right), & \text{if } \mathbf{m}_j^{(k_j)} = \mathbf{c}_j^{(k_j)} \wedge \mathbf{m}_i^{(k_i)} = \mathbf{c}_i^{(k_i)} \\ I\left(\mathbf{z}_i; \mathbf{m}_j^{(k_j)} \mid \mathbf{z}_j\right) = I\left(\mathbf{z}_j; \mathbf{m}_i^{(k_i)} \mid \mathbf{z}_i\right), & \text{else} \end{cases} \tag{5}$$

These properties are valid when considering the entire Markov Blanket; however, it still needs to be estimated. In [4], a preliminary causal variable ranking was used to facilitate this estimation. This ranking was based on measures such as minimum interaction maximum relevance (mIMR), minimum redundancy maximum relevance (MRMR), or correlation. However, these is prone to errors and increases computational time [3].

A set of quantiles of the empirical distributions of the terms computed in the steps before is used to represent the overall mixture's characteristics. The rationale behind the proposed approach is that asymmetric descriptors might be informative for predicting the directed causal link $\mathbf{z}_i \rightarrow \mathbf{z}_j$. Their values (or the corresponding quantiles, in the case of populations of descriptors) could be used to learn a classifier (e.g. a Random Forest) from synthetic data with known causal structure. The classifier would return the probability of a causal link given the descriptors value computed from new, unseen, data.

In [4] and [3], conditional mutual information is estimated as in (6).

$$I(\mathbf{z}_1; \mathbf{z}_2 \mid \mathbf{z}_3) = H(\mathbf{z}_1 \mid \mathbf{z}_3) - H(\mathbf{z}_1 \mid \mathbf{z}_2, \mathbf{z}_3) \tag{6}$$

The entropy terms are inferred from the normalised mean squared error (NMSE) of predictions made by a Ridge regression model, which indirectly measures how much information the predictor variables contain about the outcome.

[3] applies the concepts from [4] to the time series case, by including an additional descriptor based on information interaction [20], which may provide some insight about

the nature of the elements belonging to the Markov blankets. Given three random variables $\mathbf{m}_i^{(1)}, \mathbf{m}_i^{(2)} \in \mathbf{M}_i$ and $\mathbf{z}_i$ the interaction information is

$$I\left(\mathbf{m}_i^{(1)}; \mathbf{m}_i^{(2)}; \mathbf{z}_i\right) = I\left(\mathbf{m}_i^{(1)}; \mathbf{m}_i^{(2)}\right) - I\left(\mathbf{m}_i^{(1)}; \mathbf{m}_i^{(2)} \mid \mathbf{z}_i\right)$$

This quantity sheds a light on the possible causal patterns and is proven to reduce the uncertainty about the existence of this link. The complete list of descriptors used in [3] can be found in Appendix B and it amounts to 28 terms.

Assumptions We assume the absence of confounding, selection bias, and feedback configurations. We assume that the set of causal relationships existing between the variables of interest can be described by a Markov and faithful Directed Acyclic Graphs (DAG).

4 Contribution

Additional Assumptions -stationarity

In this work, we focus on the time series case, building upon the concepts introduced by [4] and [3]. In this context, we introduce a new notation to better capture the temporal dependencies between variables. With $z_i^{(t)}$ representing the value of variable $\mathbf{z}_i$ at time t, we denote the causal relationship from variable $\mathbf{z}_j$ at time $t-\tau$ in the past to variable $\mathbf{z}_k$ at time t as $z_k^{(t-\tau)} \rightarrow z_i^{(t)}$. A key assumption we make is that the past values of a variable always causally influence its own future values. Formally, $z_i^{(t-1)} \rightarrow z_i^{(t)} \forall (i,t)$. This assumption, which might be valid for many real world scenarios where past states affect future states, allows us to shed clarity on certain members of the MB, specifically the parents and children of a given variable. By identifying these members, we can more precisely target specific descriptors relevant to the causal relationships within the MB, reduce the dimensionality of the problem by focusing on the most pertinent descriptors, which in turn decreases the computational time required for the featurization and for model training. This efficiency is particularly important in the context of time series data, where the number of potential causal links can be substantial due to the temporal aspect. Furthermore, we can skip the MB estimation phase from [4] explained in Section 3.

Looking at Figure 2, we can extract a subset of the MB of $z_i^{(t)}$ as $\{z_i^{(t-1)}, z_i^{(t+1)}\}$ (cause, effect) even though additional MB members (spouses) could be missing. Nevertheless, the certainty about the presence of these two elements in the MB allows to compute more accurately a subset of descriptors from [3]. In particular, we only consider the following ones:

$$I(\mathbf{z}_i; \mathbf{m}_j^{(k)} | \mathbf{z}_j) \forall \mathbf{m}_j^{(k)} \in \mathbf{MB}_j \tag{7}$$

$$I(\mathbf{z}_j; \mathbf{m}_i^{(k)} | \mathbf{z}_i) \forall \mathbf{m}_i^{(k)} \in \mathbf{MB}_i \tag{8}$$

$$I(\mathbf{m}_i^{(k_i)}; \mathbf{m}_j^{(k_j)} | \mathbf{z}_i) \forall (\mathbf{m}_i^{(k_i)}, \mathbf{m}_j^{(k_j)}) \in \mathbf{MB_i} \times \mathbf{MB_j} \tag{9}$$

$$I(\mathbf{m}_j^{(k_j)}; \mathbf{m}_i^{(k_i)} | \mathbf{z}_j) \forall (\mathbf{m}_i^{(k_i)}, \mathbf{m}_j^{(k_j)}) \in \mathbf{MB_i} \times \mathbf{MB_j} \tag{10}$$

Given that, for the case in Figure 2, investigating the causal relationship $z_i^{(t)} \rightarrow z_j^{(t+1)}$, we estimate $MB_{z_i^{(t)}} = \{z_i^{(t-1)}, z_i^{(t+1)}\}$, and $MB_{z_j^{(t+1)}} = \{z_j^{(t)}, z_j^{(t+2)}\}$, we call D_{zmz} the family of terms from (7) and (8), and D_{mmz} the family of terms from (9) and (10). We derive:

$$D_{zmz}(i,j) = \{I(z_i^{(t)}; z_j^{(t+2)} | z_j^{(t+1)}), I(z_i^{(t)}; z_j^{(t)} | z_j^{(t+1)})\} \tag{11}$$

$$D_{zmz}(j,i) = \{I(z_j^{(t+1)}; z_i^{(t-1)} | z_i^{(t)}), I(z_j^{(t+1)}; z_i^{(t+1)} | z_i^{(t)})\} \tag{12}$$

$$\begin{aligned} D_{mmz}(i,j) = \{&I(z_i^{(t-1)}; z_j^{(t)} \mid z_i^{(t)}), I(z_i^{(t-1)}; z_j^{(t+2)} \mid z_i^{(t)}), \\ &I(z_i^{(t+1)}; z_j^{(t)} \mid z_i^{(t)}), I(z_i^{(t+1)}; z_j^{(t+2)} \mid z_i^{(t)})\} \end{aligned} \tag{13}$$

$$\begin{aligned} D_{mmz}(j,i) = \{&I(z_j^{(t)}; z_i^{(t-1)} \mid z_j^{(t+1)}), I(z_j^{(t)}; z_i^{(t+1)} \mid z_j^{(t+1)}), \\ &I(z_j^{(t+2)}; z_i^{(t-1)} \mid z_j^{(t+1)}), I(z_j^{(t+2)}; z_i^{(t+1)} \mid z_j^{(t+1)})\} \end{aligned} \tag{14}$$

These populations are asymmetric since only one of each counterparts contains a zero term. Specifically, $I\left(z_i^{(t)}; z_j^{(t)} \mid z_j^{(t+1)}\right)$ from (11) and $I\left(z_i^{(t+1)}; z_j^{(t)} \mid z_i^{(t)}\right)$ from (13) are zero, while no null term can be found in their counterparts.

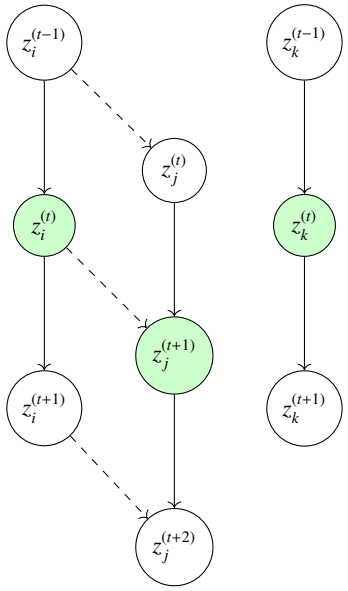

Fig. 2. A time series toy example with three variables (z_i, z_j, z_k): we study the causal link $z_i^{(t)} \rightarrow z_j^{(t+1)}$ under the hypothesis that $z_{t-1}^{(i)} \rightarrow z_t^{(i)} \forall (i,t)$. We show that it is possible to significantly reduce the number of considered descriptors w.r.t. [3] for detecting asymmetries.

Differently from [3], where the prediction error from a regression model was informing the entropy estimation under the Gaussian assumption [9], we adopt the kn-

nCMI method [21] to estimate conditional mutual information using a nearest neighbors approach. An alternative method based on the same principle is proposed by [27].

We also keep a few more descriptors that are not based on mutual information but appeared to be useful from previous experiments. In (15) and (16), $\oplus$ indicates vector concatenation

$$b : \mathbf{z}_j = b \cdot (\mathbf{z}_i \oplus \mathbf{MB}_j) \tag{15}$$

$$b : \mathbf{z}_i = b \cdot (\mathbf{z}_j \oplus \mathbf{MB}_i) \tag{16}$$

$$\text{kurt}(\mathbf{z}_i) = \mathbb{E}[(\mathbf{z}_i - \mathbb{E}[\mathbf{z}_i])^4] / \left(\mathbb{E}[(\mathbf{z}_i - \mathbb{E}[\mathbf{z}_i])^2]\right)^2 - 3 \tag{17}$$

$$\text{kurt}(\mathbf{z}_j) = \mathbb{E}[(\mathbf{z}_j - \mathbb{E}[\mathbf{z}_j])^4] / \left(\mathbb{E}[(\mathbf{z}_j - \mathbb{E}[\mathbf{z}_j])^2]\right)^2 - 3 \tag{18}$$

$$\text{HOC}_{1,2}(\mathbf{z}_i, \mathbf{z}_j) = \mathbb{E}\left[(\mathbf{z}_i - \mathbb{E}[\mathbf{z}_i])^1 \cdot (\mathbf{z}_j - \mathbb{E}[\mathbf{z}_j])^2\right] \tag{19}$$

$$\text{HOC}_{2,1}(\mathbf{z}_i, \mathbf{z}_j) = \mathbb{E}\left[(\mathbf{z}_i - \mathbb{E}[\mathbf{z}_i])^2 \cdot (\mathbf{z}_j - \mathbb{E}[\mathbf{z}_j])^1\right] \tag{20}$$

$$\text{HOC}_{1,3}(\mathbf{z}_i, \mathbf{z}_j) = \mathbb{E}\left[(\mathbf{z}_i - \mathbb{E}[\mathbf{z}_i])^1 \cdot (\mathbf{z}_j - \mathbb{E}[\mathbf{z}_j])^3\right] \tag{21}$$

$$\text{HOC}_{3,1}(\mathbf{z}_i, \mathbf{z}_j) = \mathbb{E}\left[(\mathbf{z}_i - \mathbb{E}[\mathbf{z}_i])^3 \cdot (\mathbf{z}_j - \mathbb{E}[\mathbf{z}_j])^1\right] \tag{22}$$

For each selected pair of variables, we follow a systematic procedure to determine the probability of causal relationships, as outlined in Algorithm 3. This procedure involves the following steps:

- For each variable, we estimate its Markov Blanket (MB) by selecting its lagged versions from one time step before and one after. This approach helps to identify the most relevant variables influencing the current variable.
- Using the identified MB, we compute a set of descriptors that characterize the causal relationship between the variable pairs. These descriptors include conditional mutual information terms and other statistical properties that provide insights into the dependencies and interactions between the variables.
- For families of descriptors, we compute the quantiles of their empirical distributions. This step captures the distributional characteristics and aids in feature representation for the classifier.
- The computed descriptors and their quantiles are compiled into an input feature vector. This vector encapsulates the essential characteristics of the causal relationships and serves as the input for the classifier.
- For training data, each input vector is labeled as causal (1) or noncausal (0) based on the original selection criteria from the synthetic data's Directed Acyclic Graph (DAG). This labeling is crucial for supervised learning and model training.
- The labeled dataset, comprising the feature vectors, is used to train a classifier. The classifier learns to predict the likelihood of causal relationships based on the descriptors.
- For unseen time series data, the trained classifier predicts the probability of causal links for each pair of variables. The predictions are based on the computed descriptors for the test data.

Algorithm 1 GenerateSyntheticData

Input: Generative Process p, Number of variables N, Noise standard deviations σ, Maximum neighborhood size η_{max}, Number of observations O, Number of timesteps L
Output: Collection of time series S, corresponding directed acyclic graphs $DAGS_S$
for each generative process p **do**
  for each combination of N, σ, η_{max} **do**
    for each observation o from 1 to O **do**
      Initialize time series s_t with N variables
      for each variable j from 1 to N **do**
        Initialize j's value with uniform random numbers from -1 to 1 (for maxlags)
      end for
      for each variable j from 1 to N **do**
        Determine neighborhood size η_j randomly up to $\eta_{\max}$
        Fix neighborhood $\mathcal{N}_j$ of size η_j past variables influencing j
        for each timestep l from 1 to L **do**
          Compute j's value at timestep l from η_j and p
        end for
      end for
      Add generated time series s_t to collection S
    end for
  end for
end for
return the collection of time series S

Algorithm 2 ComputeDescriptors

Input: Raw Time Series Data S, maximum lag τ_{max}
Output: Dataset D with vectors of descriptors for each variable pair
Standardize data to avoid *varsortability* [26].
$S' \leftarrow$ CreateLagged(S, τ_{max}) ▷ Include shifted features to account for temporal precedence
for each $(x_i^{(t-\tau)}, x_j^{(t)}) \in S_{\text{test}}$ with $\tau \in [1...\tau_{max}]$ **do**
  Identify $MB_{x_i^{(t-\tau)}} = x_i^{(t-\tau-1)}, x_i^{(t-\tau+1)}$ ▷ Fixed MB elements for $x_i^{(t-\tau)}$
  Identify $MB_{x_j^{(t)}} = x_j^{(t-1)}, x_j^{(t+1)}$ ▷ Fixed MB elements for $x_j^{(t)}$
  Compute family of descriptors D_{zmz} from (7), (8) using KMICNN to estimate MI
  Compute family of descriptors D_{mmz} from (9), (10) using KMICNN to estimate MI
  Compute quantiles from D_{zmz} and D_{mmz},
  Compute additional descriptors (15), (16), (17),(18),(19),(20),(21), (22)
  Compile computed descriptors (and quantiles for families) into an input vector
  if Label is known (training data) **then**
    Label the vector as causal (1) or noncausal (0) based on the corresponding DAG
  end if

end for
Aggregate all vectors into a dataset for the classifier
return the prepared dataset

Algorithm 3 TD2C Algorithm - Causal Discovery

Input: Unseen time series S_{test}, maximum lag τ_{max}, classifier $\mathcal{K}$
Output: $\hat{y}$ vector of probabilities of causal link $\forall(x_i^{(t-\tau)}, x_j^{(t)}) \in S_{\text{test}}$ with $\tau \in [1...\tau_{max}]$
$S_{\text{train}}, DAGS_{train} \leftarrow$ GenerateSyntheticData$(p, N, \sigma, \eta_{max}, O, L)$ ▷ training data generation
$D_{train} \leftarrow$ ComputeDescriptors$(S_{\text{train}}, \tau_{max})$ ▷ training descriptors
$\mathcal{K}_{trained} \leftarrow$ train$(\mathcal{K}, S_{\text{train}}, DAGS_{train})$ ▷ classifier training
$D_{\text{test}} \leftarrow$ ComputeDescriptors$(S_{\text{test}}, \tau_{max})$ ▷ testing descriptors
$\hat{y} \leftarrow$ predict$(\mathcal{K}_{trained}, D_{\text{test}})$ ▷ prediction
return $\hat{y}$

5 Experiments

This section presents a comprehensive evaluation of our causal discovery methodology, described through various experimental setups and rigorous comparative analysis. It is organized into multiple subsections, each catering to specific aspects of our experimental design and findings. Section 5.1 outlines the settings and the nature of the data used for the experiments, including the synthetic data generation process and the detailed parameters that shape our tests. This is crucial for understanding the basis on which our methodologies are evaluated. We then delve into the specifics of how the causal discovery methods are assessed, explained in the benchmarking process, and the results of these evaluations are systematically presented in Section 5.2. A sample generated time series can be found in

5.1 Experimental Settings

Synthetic Data Generation To validate our methodology, we created a comprehensive dataset from 18 unique multivariate Nonlinear Autoregressive (NAR) processes (see Table 4), with variations in the number of variables N (5, 10, 25). We set the noise standard deviation at 0.01. In our generative processes, a variable is influenced by a collection of other variables (parents) that we refer to as *neighborhood* $\mathcal{N}_j$. The size of the neighborhood η is randomly chosen from $[1...\eta_{max}]$ for each observation and kept constant for all the required timesteps. A variable is always part of its own neighborhood, to respect the hypothesis $z_{t-1}^{(i)} \rightarrow z_t^{(i)} \forall(i,t)$. Remaining members of neighborhood are chosen randomly and kept fixed for the current time series. We set $\eta_{max} = 2$. For each of these parameters combination and each generative process, we generate $O = 40$ different time series observations. We remark that each variable n from each observation o has a different neighborhood $\mathcal{N}_j$, ensuring variability within the generated time series. The generative processes used are based on those described in [3], with modifications to processes (24), (29), removal of (27), (39) addition of (40),(41), (42). Each time series is designed to reveal dynamic dependencies and capture complex nonlinear relationships and lag structures. The initial state for each series is set by a uniform random distribution, with variable values ranging from -1 to 1. Each variable, indexed by j, evolves over time influenced by its neighborhood, $\mathcal{N}_j$. Each time series consists of 250 timesteps to create a finite-sample size scenario (after removing the initialization timesteps). A summary of the parameters of our generation process can be found in

Table 3. An example of a generated time series with $N = 5$ can be found in Table 1 and its corresponding DAG in Table 2. In the following, we will refer to processes with an index from 1 to 20, according to the order they are presented, from (23) to (42). Since we removed (27), (39), the corresponding indices 5 and 17 will be missing.

0	1	2	3	4
0.712569	0.601736	1.220099	0.689470	0.387217
0.856346	1.042727	1.526494	1.115895	0.909992
1.380457	1.354174	1.161862	1.480075	1.408838
1.515645	1.519205	0.264640	1.348142	1.488786
0.628601	0.813913	0.269340	0.473941	0.589428
-0.078369	-0.086761	0.710499	0.037373	-0.066524
0.599358	0.600255	1.137044	0.600913	0.600516
0.980150	0.943964	1.492956	1.002143	0.981254
1.391383	1.367835	1.317824	1.401253	1.390770

Table 1. Example of generated Time Series

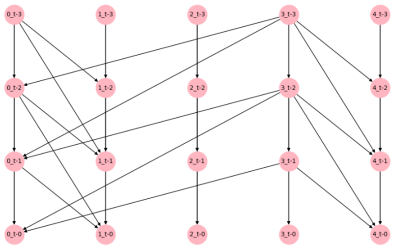

Table 2. Corresponding DAG

Parameter	Description
Generative Process p	$p \in \{1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 18, 19, 20\}$
Variables per Series N	$N \in \{5, 10, 25\}$
Observations per combination o	$o = 40$
Noise standard deviations σ	$\sigma = 0.01$
Interactions	Dynamic, capturing linear and nonlinear relationships
Lag Structures	Complex, with temporal dependencies up to 4 time steps
Max Neighborhood Size η_j^{max}	$\eta_j^{max} = 2$
Neighborhood Mechanism	Stochastic selection, different for each variable
Initial State Distribution	Uniform random distribution between -1 and 1
Evolution Over Time	Each variable influenced by its neighborhood, N_j (random)
Timesteps per Series L	$L = 250$, to observe variable evolution and causal relationships

Table 3. Summary of the parameters of our time series generation

Descriptors computation From the DAGs associated with the observations generated here, we extract the ground truth of causal relationships within the available data. We compute descriptors for all possible causal pairs (i.e., $t - \tau \rightarrow t$). When computing the descriptors, our data is standardized to avoid *varsortability*. [26]. In practice, this means computing descriptors for 125 pairs for N=5, 500 pairs for N=10, 3125 pairs for N=25. This is done for each of the 40 time series generated from each of the 18 generative processes. Notice that one could choose to limit the number of considered pairs for training, which could be useful for cases when we are interested in predicting the existence of a causal link between a specific subset of couples.

$$Y_{t+1}[j] = -0.4\frac{\left(3-\bar{Y}_t\left[\mathcal{N}_j\right]^2\right)}{\left(1+\bar{Y}_t\left[\mathcal{N}_j\right]^2\right)} + 0.6\frac{3-\left(\bar{Y}_{t-1}\left[\mathcal{N}_j\right]-0.5\right)^3}{1+\left(\bar{Y}_{t-1}\left[\mathcal{N}_j\right]-0.5\right)^4} + W_{t+1}[j] \tag{23}$$

$$Y_{t+1}[j] = \left(0.4-2\cos\left(40\bar{Y}_{t-2}\left[\mathcal{N}_j\right]\right)\exp\left(-30\bar{Y}_{t-2}\left[\mathcal{N}_j\right]^2\right)\right)\bar{Y}_{t-2}\left[\mathcal{N}_j\right] + \left(0.5-0.5\exp\left(-50\bar{Y}_{t-1}\left[\mathcal{N}_j\right]^2\right)\right)\bar{Y}_{t-1}\left[\mathcal{N}_j\right] + W_{t+1}[j] \tag{24}$$

$$Y_{t+1}[j] = 1.5\sin\left(\pi/2\bar{Y}_{t-1}\left[\mathcal{N}_j\right]\right) - \sin\left(\pi/2\bar{Y}_{t-2}\left[\mathcal{N}_j\right]\right) + W_{t+1}[j] \tag{25}$$

$$Y_{t+1}[j] = 2\exp\left(-0.1\bar{Y}_t\left[\mathcal{N}_j\right]^2\right)\bar{Y}_t\left[\mathcal{N}_j\right] - \exp\left(-0.1\bar{Y}_{t-1}\left[\mathcal{N}_j\right]^2\right)\bar{Y}_{t-1}\left[\mathcal{N}_j\right] + W_{t+1}[j] \tag{26}$$

$$Y_{t+1}[j] = -2\bar{Y}_t\left[\mathcal{N}_j\right]I\left(\bar{Y}_t\left[\mathcal{N}_j\right]<0\right) + 0.4\bar{Y}_t\left[\mathcal{N}_j\right]I\left(\bar{Y}_t\left[\mathcal{N}_j\right]<0\right) + W_{t+1}[j] \tag{27}$$

$$Y_{t+1}[j] = 0.8\log\left(1+3\bar{Y}_t\left[\mathcal{N}_j\right]^2\right) - 0.6\log\left(1+3\bar{Y}_{t-2}\left[\mathcal{N}_j\right]^2\right) + W_{t+1}[j] \tag{28}$$

$$Y_{t+1}[j] = \left(0.4-2\cos\left(40\bar{Y}_{t-2}\left[\mathcal{N}_j\right]\right)\exp\left(-30\bar{Y}_{t-2}\left[\mathcal{N}_j\right]^2\right)\right)\bar{Y}_{t-2}\left[\mathcal{N}_j\right] + \left(0.5-0.5\exp\left(-50\bar{Y}_{t-1}\left[\mathcal{N}_j\right]^2\right)\right)\bar{Y}_{t-1}\left[\mathcal{N}_j\right] + W_{t+1}[j] \tag{29}$$

$$Y_{t+1}[j] = \left(0.5-1.1\exp\left(-50\bar{Y}_t\left[\mathcal{N}_j\right]^2\right)\right)\bar{Y}_t\left[\mathcal{N}_j\right] + \left(0.3-0.5\exp\left(-50\bar{Y}_{t-2}\left[\mathcal{N}_j\right]^2\right)\right)\bar{Y}_{t-2}\left[\mathcal{N}_j\right] + W_{t+1}[j] \tag{30}$$

$$Y_{t+1}[j] = 0.3\bar{Y}_t\left[\mathcal{N}_j\right] + 0.6\bar{Y}_{t-1}\left[\mathcal{N}_j\right] + \frac{\left(0.1-0.9\bar{Y}_t\left[\mathcal{N}_j\right]+0.8\bar{Y}_{t-1}\left[\mathcal{N}_j\right]\right)}{\left(1+\exp\left(-10\bar{Y}_t\left[\mathcal{N}_j\right]\right)\right)} + W_{t+1}[j] \tag{31}$$

$$Y_{t+1}[j] = \text{sign}\left(\bar{Y}_t\left[\mathcal{N}_j\right]\right) + W_{t+1}[j] \tag{32}$$

$$Y_{t+1}[j] = 0.8\bar{Y}_t\left[\mathcal{N}_j\right] - \frac{0.8\bar{Y}_t\left[\mathcal{N}_j\right]}{\left(1+\exp\left(-10\bar{Y}_t\left[\mathcal{N}_j\right]\right)\right)} + W_{t+1}[j] \tag{33}$$

$$Y_{t+1}[j] = 0.3\bar{Y}_t\left[\mathcal{N}_j\right] + 0.6\bar{Y}_{t-1}\left[\mathcal{N}_j\right] + \frac{\left(0.1-0.9\bar{Y}_t\left[\mathcal{N}_j\right]+0.8\bar{Y}_{t-1}\left[\mathcal{N}_j\right]\right)}{\left(1+\exp\left(-10\bar{Y}_t\left[\mathcal{N}_j\right]\right)\right)} + W_{t+1}[j] \tag{34}$$

$$Y_{t+1}[j] = 0.38\bar{Y}_t\left[\mathcal{N}_j\right]\left(1-\bar{Y}_{t-1}\left[\mathcal{N}_j\right]\right) + W_{t+1}[j] \tag{35}$$

$$Y_{t+1}[j] = \begin{cases} -0.5\bar{Y}_t\left[\mathcal{N}_j\right] & \text{if} \quad \bar{Y}_t\left[\mathcal{N}_j\right] < 1 \\ 0.4\bar{Y}_t\left[\mathcal{N}_j\right] \end{cases} \tag{36}$$

$$Y_{t+1}[j] = \begin{cases} 0.9\bar{Y}_t\left[\mathcal{N}_j\right] + W_{t+1}[j] \text{ if} & \left|\bar{Y}_t\left[\mathcal{N}_j\right]\right| < 1 \\ -0.3\bar{Y}_t\left[\mathcal{N}_j\right] + W_{t+1}[j] \end{cases} \tag{37}$$

$$Y_{t+1}[j] = \begin{cases} -0.5\bar{Y}_t\left[\mathcal{N}_j\right] + W_{t+1}[j] & \text{if} \quad x_t = 1 \\ 0.4\bar{Y}_t\left[\mathcal{N}_j\right] + W_{t+1}[j] \end{cases} \tag{38}$$

$$x_{t+1} = 1 - x_t, x_0 = 1$$

$$Y_{t+1}[j] = \sqrt{0.000019 + 0.846 * \left(\bar{Y}_t\left[\mathcal{N}_j\right]^2 + 0.3\bar{Y}_{t-1}\left[\mathcal{N}_j\right]^2 + 0.2\bar{Y}_{t-2}\left[\mathcal{N}_j\right]^2 + 0.1\bar{Y}_{t-3}\left[\mathcal{N}_j\right]^2\right)}W_{t+1}[j] \tag{39}$$

$$Y_{t+1}[j] = 0.9\cdot\bar{Y}_t\left[\mathcal{N}_j\right] + W_{t+1}[j] \tag{40}$$

$$Y_{t+1}[j] = 0.4\cdot\bar{Y}_{t-1}\left[\mathcal{N}_j\right] + 0.6\cdot\bar{Y}_{t-2}\left[\mathcal{N}_j\right] + W_{t+1}[j] \tag{41}$$

$$Y_{t+1}[j] = 0.5\cdot\bar{Y}_{t-3}\left[\mathcal{N}_j\right] + W_{t+1}[j] \tag{42}$$

(43)

Table 4. Cross-sectional and temporal series from [3]: $\mathcal{N}_j$ denotes the indices of the set of time series which are neighbors of the j^{th} component. $\bar{y}_t\left[\mathcal{N}_j\right]$ stands for the average of the value of the neighboring series at time t. Generative processes (27) and (39) have been removed because of their instability with the adopted initial conditions.

Benchmark A comprehensive comparison against three state-of-the-art causal discovery methodologies is performed. Each selected method exemplifies a distinct category within the spectrum of causal inference approaches. From the constraint-based family, we select the PCMCI algorithm developed by Runge [28], which is implemented in the Tigramite Python package. The Varlingam method, as proposed by Hyvärinen et al. [14], is our choice for the noise-based category. It is implemented in Python via the LiNGAM library [15]. Finally, for the score-based category, we incorporate DYNOTEARS, a method introduced by Pamfil et al. [23] and implemented in the CausalNex Python library [2]. DYNOTEARS is notable for its application of a differentiable programming approach to learn both static and dynamic causal networks from observational data. It is important to note that each method has its own underlying assumptions, which might not always be respected in practical scenarios. For example, VarLiNGAM assumes non-Gaussian errors, which is not the case in our experiments. Similarly, we use PCMCI with the ParCorr independence test; while a nonlinear test would be more appropriate. We faced computational challenges with PCMCI when attempting to use CMIknn. Our goal is not to demonstrate that our approach outperforms all others under all conditions. Instead, we aim to show that our method can be a valuable addition to the toolbox for causal discovery in time series, offering unique insights and potentially complementing existing techniques. The adopted conditions for each method might be suboptimal for the given dataset, yet they provide a robust benchmark to evaluate the relative strengths and potential applications of our proposed approach.

Evaluation For validation purposes, when evaluating the performance on data from a specific process p, all descriptor pairs originating from process p are excluded from the training set to prevent data leakage. This ensures that the model is tested on unseen data, maintaining the integrity of the validation process, and that all the remaining data is used for training.

The subsequent analysis is conducted within the framework of binary classification, focusing on the minority class. In this context, we define the following terms:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{44}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{45}$$

$$\text{F1-Score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2 * \text{TP}}{2 * \text{TP} + \text{FP} + \text{FN}} \tag{46}$$

Precision (44) calculates the proportion of actual positives among the predicted positives. Recall (45) quantifies the proportion of actual positives that were correctly identified. The F1-Score (46) is the harmonic mean of Precision and Recall. These metrics are influenced by the classification threshold, typically set at 0.5. This standard threshold assigns a class label based on whether the predicted probability surpasses 0.5. Raising this threshold increases precision by reducing false positives but may increase false negatives, reducing recall. This trade-off might be preferred in situations where the cost of a false positive is particularly high. Conversely, lowering the threshold increases recall but reduces precision. In our case, for precision, recall, and F1 score evaluations,

we set a threshold at 0.8 after analyzing the precision-recall versus threshold curve on a holdout set. While our recall remained stable across various thresholds, precision significantly increased at the chosen threshold, indicating reliable identification of true causal relationships. The performance of a model at varying thresholds is effectively assessed using tools like the ROC curve, which plots Recall against FPR for different thresholds. The area under the curve (AUC) indicates model quality, with a perfect classifier achieving an AUC of 1, corresponding to a Recall of 1 and an FPR of 0 at all thresholds. In the following, we will plot ROC-AUC when possible, since not all methods benchmarked provide probabilistic outputs necessary for calculating ROC-AUC. PCMCI returns a p-value for the independence test; for the sake of comparison via ROC-AUC, we use 1 - p-value as a proxy for confidence in the existence of a causal link. While this approach may not be methodologically ideal, it allows for a more consistent comparison. For a more rigorous comparison, we can refer to metrics such as the F1-Score, which remains unaffected by this issue due to its binary nature.

5.2 Results

In this section, we present the results of our causal discovery experiments, focusing on a comprehensive evaluation of various methods. Detailed results can be found in Appendix A. We prioritize the ROC-AUC metric as it provides a robust measure of overall performance without relying on a specific threshold.

In terms of ROC-AUC, TD2C demonstrates the highest median ROC AUC with a significantly lower spread with respect to other methods, for all the 3 dimensions tested. The lowest spread is achieved at N=10, likely due to the balance between the amount of data available for training and the complexity of the causal discovery task. TD2C is followed closely by PCMCI, which shows a higher spread. VARLINGAM follows with a slightly lower median AUC. DYNOTEARS does not provide any probabilistic measure and is therefore not considered for this metric.

For threshold-depended metrics, we observed that experiments with $N = 5$ variables resulted in significantly degraded performance for TD2C. This is likely due to insufficient data to capture the complexity of causal relationships, underscoring the necessity of adequate data for reliable causal discovery. TD2C dominates in terms of precision but performs worse than the other approaches in terms of recall. This can be explained by the process-specific results presented in Appendix A, since TD2C fails dramatically for a few specific processes, reducing the aggregating performances.

DYNOTEARS consistently demonstrated the lowest performance across all metrics. This underperformance can be attributed to the variance structure of the simulated data, which is misaligned with the topological order of the causal graph, as highlighted in [6]. VarLiNGAM showed competitive performances even in the suboptimal case of gaussian noise. Furthermore, VarLiNGAM implementation [15] occasionally fails to return scores for all edges, and in such cases, we complete the missing edges by assuming a non-detection, which further impacts its accuracy. This shows that VarLiNGAM remaining a competitive approach for causal discovery. This demonstrates that VarLiNGAM remains a viable and competitive approach for causal discovery. PCMCI, when used with partial correlation (ParCorr), showed competitive performance. This can be explained by the fact that linear methods may be employed for edge detection in

non-linear settings [32]. Comparison with PCMCI with nonlinear independe testing is left for future work.

We validate our results via statistical test. Specifically, we show critical difference diagrams based on the Wilcoxon-Holm method with a python implementation from [16]. Figures 4 to 7 provide a clear visual representation of the relative performance of the causal discovery methods. These tests confirm the previous considerations.

6 Conclusion

In this paper, we introduced a novel approach for causal discovery in time series data, emphasizing the computation of asymmetric conditional mutual information within the Markov blankets of variable pairs. By framing causal discovery as a supervised learning problem, our methodology leverages featurization to capture the asymmetric nature of causal relationships, providing a comprehensive set of features specifically designed to detect causal signals. Our empirical results demonstrate that our method not only competes with established benchmarks but also offers new insights into causal inference dynamics.

Our proposed method, TD2C, showed promising results across various experimental setups, outperforming other state-of-the-art methods in terms of precision and overall performance as measured by ROC-AUC. However, it exhibited lower recall in certain scenarios, particularly for smaller datasets with fewer variables, highlighting the necessity for adequate data to capture complex causal relationships accurately. This limitation underscores the importance of considering the specific characteristics of the dataset when applying our method.

For future work, we aim to explore the integration of nonlinear independence tests with PCMCI to further enhance its performance. Additionally, we plan to investigate the application of our method to real-world datasets, where the assumptions and conditions of the synthetic data may not hold, to validate its robustness and generalizability. Enhancing the efficiency of our method for larger datasets and exploring its application in different domains will also be key areas of focus.

In conclusion, our method provides a valuable addition to the toolbox for causal discovery in time series, offering unique insights and complementing existing techniques. The open-source Python implementation and benchmark dataset we provide aim to enhance reproducibility and encourage further research in this area, contributing to the ongoing development of effective causal inference methodologies.

References

1. Assaad, C.K., Devijver, E., Gaussier, E.: Survey and evaluation of causal discovery methods for time series. Journal of Artificial Intelligence Research **73**, 767–819 (2022)
2. Beaumont, P., Horsburgh, B., Pilgerstorfer, P., Droth, A., Oentaryo, R., Ler, S., Nguyen, H., Ferreira, G.A., Patel, Z., Leong, W.: CausalNex (Oct 2021), `https://github.com/quantumblacklabs/causalnex`
3. Bontempi, G.: Learning causal dependencies in large-variate time series. In: 2020 International Joint Conference on Neural Networks (IJCNN). pp. 1–7. IEEE (2020)

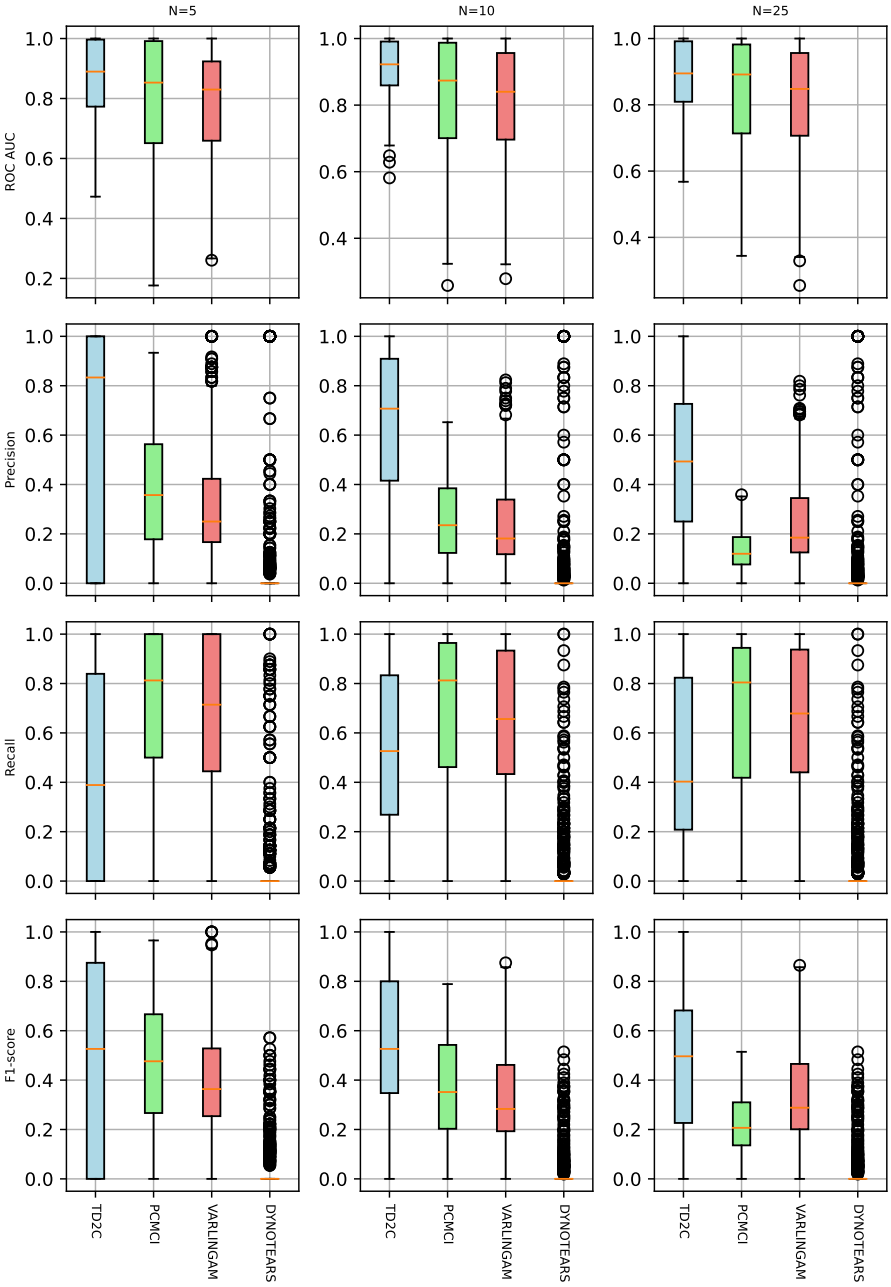

Fig. 3. Scores are computed across datasets generated by various generative processes as described in Table 4, grouped by number of variables.

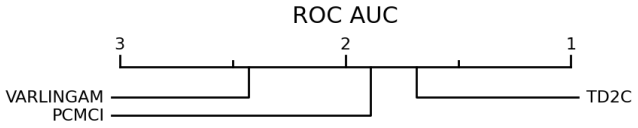

Fig. 4. Critical difference diagrams based on the Wilcoxon-Holm method - for ROC-AUC and N=10

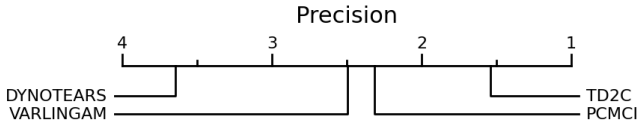

Fig. 5. Critical difference diagrams based on the Wilcoxon-Holm method - for precision and N=10

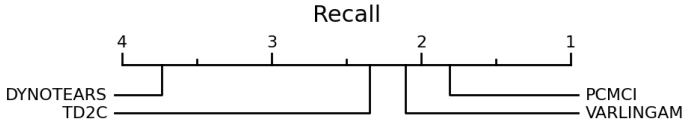

Fig. 6. Critical difference diagrams based on the Wilcoxon-Holm method - for recall and N=10

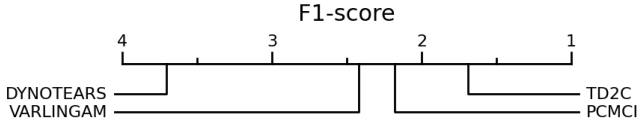

Fig. 7. Critical difference diagrams based on the Wilcoxon-Holm method - for F1-Score and N=10

4. Bontempi, G., Flauder, M.: From dependency to causality: a machine learning approach. J. Mach. Learn. Res. **16**(1), 2437–2457 (2015)
5. Brovelli, A., Ding, M., Ledberg, A., Chen, Y., Nakamura, R., Bressler, S.L.: Beta oscillations in a large-scale sensorimotor cortical network: directional influences revealed by granger causality. Proceedings of the National Academy of Sciences **101**(26), 9849–9854 (2004)
6. Bystrova, D., Assaad, C., Arbel, J., Devijver, E., Gaussier, É., Thuiller, W.: Causal discovery from time series with hybrids of constraint-based and noise-based algorithms. Transactions on Machine Learning Research Journal (2024)
7. Colombo, D., Maathuis, M.H., et al.: Order-independent constraint-based causal structure learning. J. Mach. Learn. Res. **15**(1), 3741–3782 (2014)
8. Comon, P.: Independent component analysis, a new concept? Signal processing **36**(3), 287–314 (1994)
9. Cover, T.M.: Elements of information theory. John Wiley & Sons (1999)
10. Daniusis, P., Janzing, D., Mooij, J., Zscheischler, J., Steudel, B., Zhang, K., Schölkopf, B.: Inferring deterministic causal relations. arXiv preprint arXiv:1203.3475 (2012)
11. Granger, C.W.: Testing for causality: A personal viewpoint. Journal of Economic Dynamics and control **2**, 329–352 (1980)
12. Hiemstra, C., Jones, J.D.: Testing for linear and nonlinear granger causality in the stock price-volume relation. The Journal of Finance **49**(5), 1639–1664 (1994)
13. Hoyer, P., Janzing, D., Mooij, J.M., Peters, J., Schölkopf, B.: Nonlinear causal discovery with additive noise models. Advances in neural information processing systems **21** (2008)
14. Hyvärinen, A., Zhang, K., Shimizu, S., Hoyer, P.O.: Estimation of a structural vector autoregression model using non-gaussianity. Journal of Machine Learning Research **11**(5) (2010)
15. Ikeuchi, T., Ide, M., Zeng, Y., Maeda, T.N., Shimizu, S.: Python package for causal discovery based on lingam. Journal of Machine Learning Research **24**(14), 1–8 (2023)
16. Ismail Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., Muller, P.A.: Deep learning for time series classification: a review. Data Mining and Knowledge Discovery **33**(4), 917–963 (2019)
17. Lopez-Paz, D., Muandet, K., Schölkopf, B., Tolstikhin, I.: Towards a learning theory of cause-effect inference. In: Bach, F., Blei, D. (eds.) Proceedings of the 32nd International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 37, pp. 1452–1461. PMLR, Lille, France (07–09 Jul 2015), https://proceedings.mlr.press/v37/lopez-paz15.html
18. Luo, L., Liu, W., Koprinska, I., Chen, F.: Discovering causal structures from time series data via enhanced granger causality. In: AI 2015: Advances in Artificial Intelligence: 28th Australasian Joint Conference, Canberra, ACT, Australia, November 30–December 4, 2015, Proceedings 28. pp. 365–378. Springer (2015)
19. Marinazzo, D., Pellicoro, M., Stramaglia, S.: Kernel method for nonlinear granger causality. Physical review letters **100**(14), 144103 (2008)
20. McGill, W.: Multivariate information transmission. Transactions of the IRE Professional Group on Information Theory **4**(4), 93–111 (1954)
21. Mesner, O.C., Shalizi, C.R.: Conditional mutual information estimation for mixed discrete and continuous variables with nearest neighbors. arXiv preprint arXiv:1912.03387 (2019)
22. Nauta, M., Bucur, D., Seifert, C.: Causal discovery with attention-based convolutional neural networks. Machine Learning and Knowledge Extraction **1**(1), 19 (2019)
23. Pamfil, R., Sriwattanaworachai, N., Desai, S., Pilgerstorfer, P., Georgatzis, K., Beaumont, P., Aragam, B.: Dynotears: Structure learning from time-series data. In: International Conference on Artificial Intelligence and Statistics. pp. 1595–1605. PMLR (2020)
24. Papagiannopoulou, C., Miralles, D.G., Decubber, S., Demuzere, M., Verhoest, N.E., Dorigo, W.A., Waegeman, W.: A non-linear granger-causality framework to investigate climate–vegetation dynamics. Geoscientific Model Development **10**(5), 1945–1960 (2017)

25. Pearl, J.: Causality. Cambridge university press (2009)

26. Reisach, A., Seiler, C., Weichwald, S.: Beware of the simulated dag! causal discovery benchmarks may be easy to game. Advances in Neural Information Processing Systems **34**, 27772–27784 (2021)

27. Runge, J.: Conditional independence testing based on a nearest-neighbor estimator of conditional mutual information. In: International Conference on Artificial Intelligence and Statistics. pp. 938–947. Pmlr (2018)

28. Runge, J., Nowack, P., Kretschmer, M., Flaxman, S., Sejdinovic, D.: Detecting and quantifying causal associations in large nonlinear time series datasets. Science advances **5**(11), eaau4996 (2019)

29. Shimizu, S., Hoyer, P.O., Hyvärinen, A., Kerminen, A., Jordan, M.: A linear non-gaussian acyclic model for causal discovery. Journal of Machine Learning Research **7**(10) (2006)

30. Spirtes, P., Glymour, C., Scheines, R., Kauffman, S., Aimale, V., Wimberly, F.: Constructing bayesian network models of gene expression networks from microarray data (2000)

31. Spirtes, P., Glymour, C., Scheines, R.: Causation, prediction, and search. MIT press (2001)

32. Weichwald, S., Jakobsen, M.E., Mogensen, P.B., Petersen, L., Thams, N., Varando, G.: Causal structure learning from time series: Large regression coefficients may predict causal links better in practice than small p-values. Proceedings of the NeurIPS 2019 Competition and Demonstration Track, Proceedings of Machine Learning Research, vol. 123, pp. 27–36. PMLR (2020), http://proceedings.mlr.press/v123/weichwald20a.html

33. Zheng, X., Aragam, B., Ravikumar, P.K., Xing, E.P.: Dags with no tears: Continuous optimization for structure learning. Advances in neural information processing systems **31** (2018)

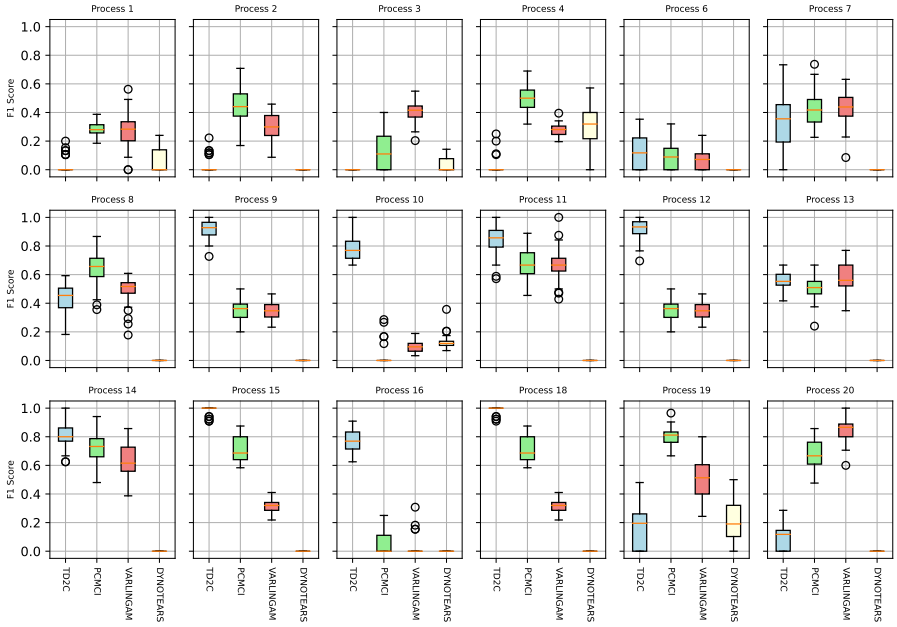

Fig. 8. F1 Scores N=5

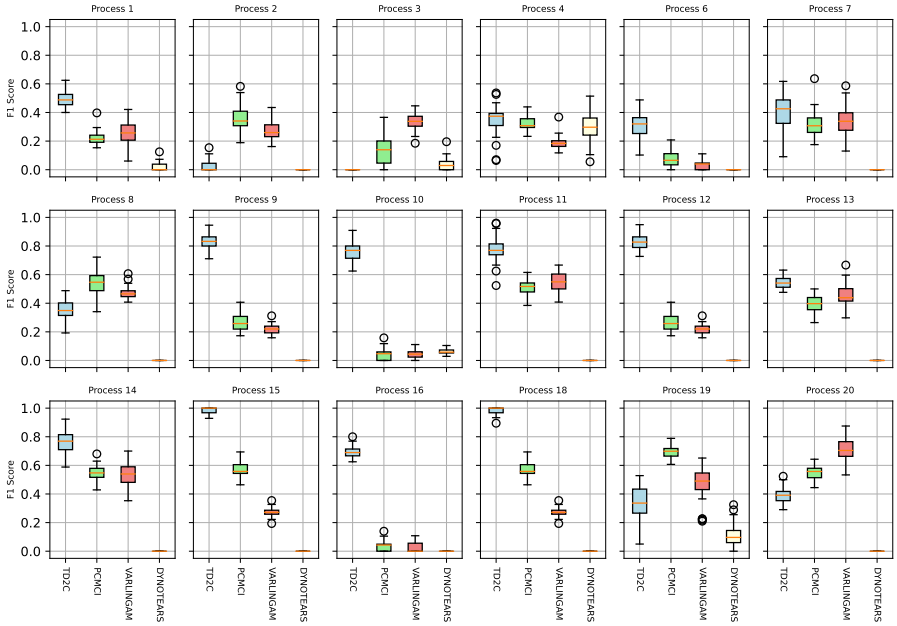

Fig. 9. F1 Scores N=10

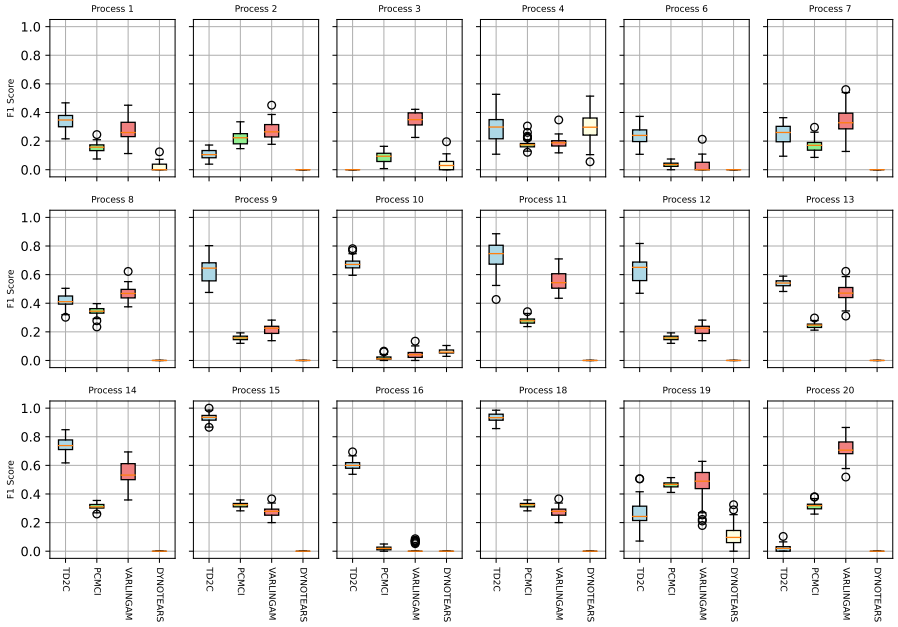

Fig. 10. F1 Scores N=25

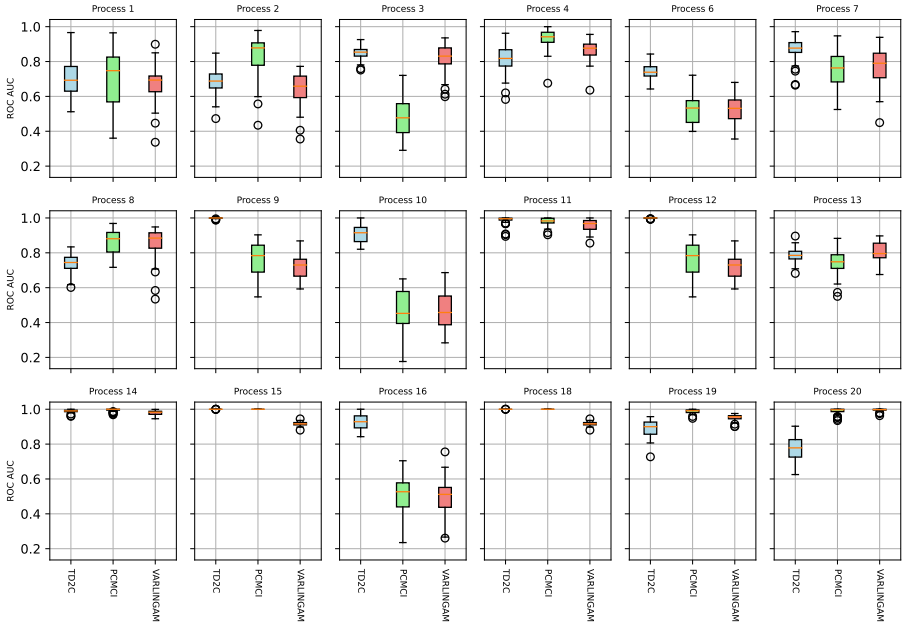

Fig. 11. ROC-AUC Scores N=5

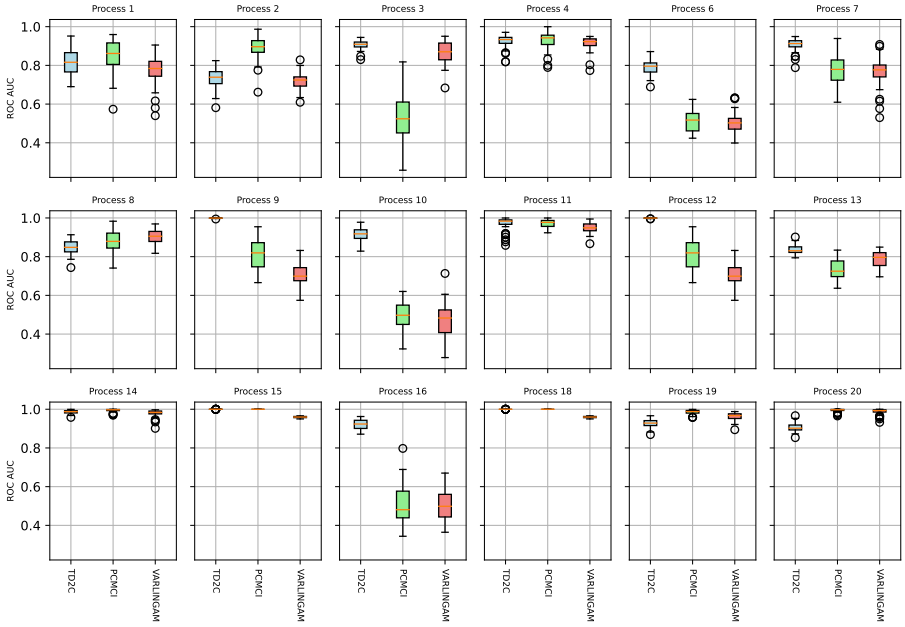

Fig. 12. ROC-AUC Scores N=10

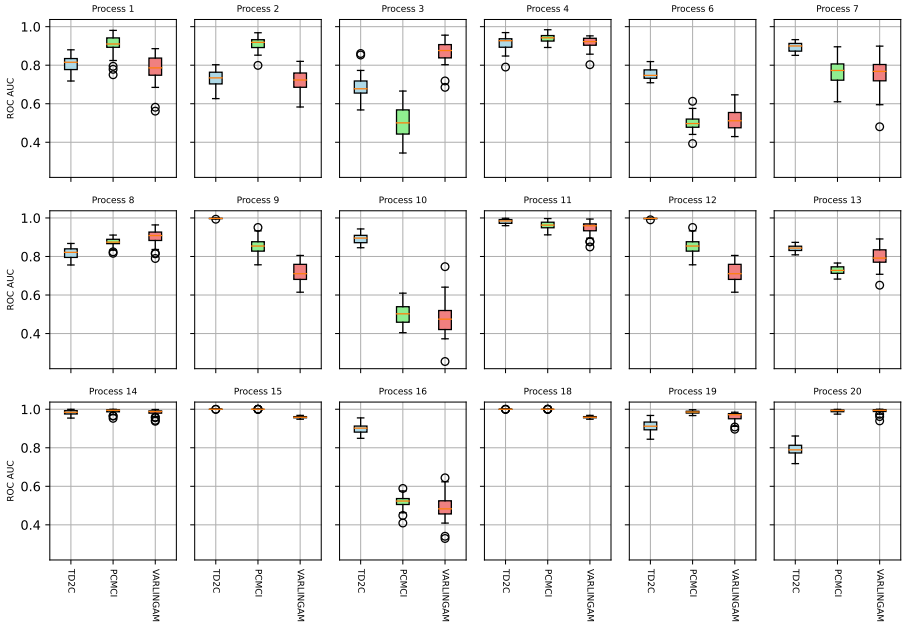

Fig. 13. ROC-AUC Scores N=25

A Detailed Results

B List of full descriptors from [3]

$$n \tag{47}$$

$$m \tag{48}$$

$$m/n \tag{49}$$

$$b : \mathbf{z}_j = b \cdot (\mathbf{z}_i \oplus \mathbf{MB}_j), \text{ where } \oplus \text{ indicates vector concatenation} \tag{50}$$

$$b : \mathbf{z}_i = b \cdot (\mathbf{z}_j \oplus \mathbf{MB}_i), \text{ where } \oplus \text{ indicates vector concatenation} \tag{51}$$

$$\text{kurt}(\mathbf{z}_i) = \mathbb{E}[(\mathbf{z}_i - \mathbb{E}[\mathbf{z}_i])^4] / \left(\mathbb{E}[(\mathbf{z}_i - \mathbb{E}[\mathbf{z}_i])^2]\right)^2 - 3 \tag{52}$$

$$\text{kurt}(\mathbf{z}_j) = \mathbb{E}[(\mathbf{z}_j - \mathbb{E}[\mathbf{z}_j])^4] / \left(\mathbb{E}[(\mathbf{z}_j - \mathbb{E}[\mathbf{z}_j])^2]\right)^2 - 3 \tag{53}$$

$$\text{skew}(\mathbf{z}_i) = \mathbb{E}[(\mathbf{z}_i - \mathbb{E}[\mathbf{z}_i])^3] / \left(\mathbb{E}[(\mathbf{z}_i - \mathbb{E}[\mathbf{z}_i])^2]\right)^{3/2} \tag{54}$$

$$\text{skew}(\mathbf{z}_j) = \mathbb{E}[(\mathbf{z}_j - \mathbb{E}[\mathbf{z}_j])^3] / \left(\mathbb{E}[(\mathbf{z}_j - \mathbb{E}[\mathbf{z}_j])^2]\right)^{3/2} \tag{55}$$

$$\text{HOC}_{1,2}(\mathbf{z}_i, \mathbf{z}_j) = \mathbb{E}\left[(\mathbf{z}_i - \mathbb{E}[\mathbf{z}_i])^1 \cdot (\mathbf{z}_j - \mathbb{E}[\mathbf{z}_j])^2\right] \tag{56}$$

$$\text{HOC}_{2,1}(\mathbf{z}_i, \mathbf{z}_j) = \mathbb{E}\left[(\mathbf{z}_i - \mathbb{E}[\mathbf{z}_i])^2 \cdot (\mathbf{z}_j - \mathbb{E}[\mathbf{z}_j])^1\right] \tag{57}$$

$$\text{HOC}_{1,3}(\mathbf{z}_i, \mathbf{z}_j) = \mathbb{E}\left[(\mathbf{z}_i - \mathbb{E}[\mathbf{z}_i])^1 \cdot (\mathbf{z}_j - \mathbb{E}[\mathbf{z}_j])^3\right] \tag{58}$$

$$\text{HOC}_{3,1}(\mathbf{z}_i, \mathbf{z}_j) = \mathbb{E}\left[(\mathbf{z}_i - \mathbb{E}[\mathbf{z}_i])^3 \cdot (\mathbf{z}_j - \mathbb{E}[\mathbf{z}_j])^1\right] \tag{59}$$

$$I(\mathbf{z}_i; \mathbf{z}_j) \tag{60}$$

$$I(\mathbf{z}_j; \mathbf{z}_i) \tag{61}$$

$$I(\mathbf{m}_j^{(k)}; \mathbf{z}_i) \forall \mathbf{m}_j^{(k)} \in \mathbf{MB}_j \tag{62}$$

$$I(\mathbf{m}_i^{(k)}; \mathbf{z}_j) \forall \mathbf{m}_i^{(k)} \in \mathbf{MB}_i \tag{63}$$

$$I(\mathbf{z}_i; \mathbf{z}_j | \mathbf{MB}_i \cap \mathbf{MB}_j) \tag{64}$$

$$I(\mathbf{z}_j; \mathbf{z}_i | \mathbf{MB}_j) \tag{65}$$

$$I(\mathbf{z}_i; \mathbf{z}_j | \mathbf{MB}_i) \tag{66}$$

$$I(\mathbf{z}_j; \mathbf{z}_i | \mathbf{MB}_i \cup \mathbf{m}_j^{(k)}) \forall \mathbf{m}_j^{(k)} \in \mathbf{MB}_j \tag{67}$$

$$I(\mathbf{z}_i; \mathbf{z}_j | \mathbf{MB}_j \cup \mathbf{m}_i^{(k)}) \forall \mathbf{m}_i^{(k)} \in \mathbf{MB}_i \tag{68}$$

$$I(\mathbf{z}_i; \mathbf{m}_j^{(k)} | \mathbf{z}_j) \forall \mathbf{m}_j^{(k)} \in \mathbf{MB}_j \tag{69}$$

$$I(\mathbf{z}_j; \mathbf{m}_i^{(k)} | \mathbf{z}_i) \forall \mathbf{m}_i^{(k)} \in \mathbf{MB}_i \tag{70}$$

$$I(\mathbf{m}_i^{(k_i)}; \mathbf{m}_j^{(k_j)} | \mathbf{z}_i) \forall (\mathbf{m}_i^{(k_i)}, \mathbf{m}_j^{(k_j)}) \in \mathbf{MB_i} \times \mathbf{MB_j} \tag{71}$$

$$I(\mathbf{m}_j^{(k_j)}; \mathbf{m}_i^{(k_i)} | \mathbf{z}_j) \forall (\mathbf{m}_i^{(k_i)}, \mathbf{m}_j^{(k_j)}) \in \mathbf{MB_i} \times \mathbf{MB_j} \tag{72}$$

$$I(\mathbf{m}_i^{(k_i^1)}; \mathbf{m}_i^{(k_i^2)} | \mathbf{z}_i) - I(\mathbf{m}_i^{(k_i^1)}; \mathbf{m}_i^{(k_i^2)}) \forall (\mathbf{m}_i^{(k_i^1)}, \mathbf{m}_i^{(k_i^2)}) \in \mathbf{MB_i} \times \mathbf{MB_i} \tag{73}$$

$$I(\mathbf{m}_j^{(k_j^1)}; \mathbf{m}_j^{(k_j^2)} | \mathbf{z}_j) - I(\mathbf{m}_j^{(k_j^1)}; \mathbf{m}_j^{(k_j^2)}) \forall (\mathbf{m}_j^{(k_j^1)}, \mathbf{m}_j^{(k_j^2)}) \in \mathbf{MB_j} \times \mathbf{MB_j} \tag{74}$$