PRETRAINED CAUSAL DISCOVERY AT ANY SIZE

Anonymous authors
Paper under double-blind review

ABSTRACT

Pretrained causal discovery from observational data trains a model on simulated systems and applies it to new data in a single inference pass. In the temporal case, most such models read the raw time series, tying them to the scale of the training data and restricting flexibility. A different strategy turns discovery into supervised learning over features of the data rather than over the data itself. We present a deep learning reformulation of that strategy: a permutation-equivariant network over a single tensor of dependence statistics for every target, source and lag triplet. No weight depends on the number of variables or lags, addressing a clear gap of currently released pretrained models. The network is competitive with the per-edge classifier it reformulates, with other pre-trained models and with traditional approaches. We also investigate transfer to unseen domains and realistic datasets.

1 INTRODUCTION

Causal discovery from observational time series has traditionally meant running an algorithm from scratch on every dataset under study. Pretrained causal discovery proposes to train a model once on simulated systems and apply it to new data in a single inference pass (Lorch et al., 2022; Löwe et al., 2022; Stein et al., 2024; Kougioulis et al., 2026). Most of these models belong to a *raw-series* family: they learn directly from the observed values, which ties them to the variable counts, series lengths and scales seen in training. A *descriptor* family avoids this by learning from features of the data rather than from the data itself (Lopez-Paz et al., 2015; Guyon et al., 2019). Recent methods such as TD2C (Paldino & Bontempi, 2025) and its static predecessor D2C (Bontempi & Flauder, 2015) summarize each candidate edge by a vector of dependence statistics and classify it with a random forest. This buys robustness to the scale, length and marginal shape of the series, but computes a number of classifications that grows quadratically with variable counts, each with a conditioning set of its own to estimate.

In this paper, we reformulate the descriptor strategy as deep learning over a tensor of dependence statistics computed per each candidate target, source and lag triplets. The tensor does not change if a series is shifted or rescaled, its rank-based channels are also unchanged by any monotone map of a marginal, and its values stay comparable across series lengths. A permutation-equivariant network, i.e. a network whose output reorders in the same way as its input is reordered (Hartford et al., 2018; Maron et al., 2019) processes this tensor. Cells inform each other through the shared representation, and the statistics are computed once per system rather than once per edge. A single trained model serves any number of variables and lags, including counts at which the released pretrained models are structurally unable to run.

We make two contributions:

- We propose a permutation-equivariant architecture over tensors of dependence statistics whose structure does not depend on the number of variables and lags.
- We evaluate the architecture against its predecessor, against the statistics it takes as input, and against pretrained and traditional methods. We do so on held-out generated data, on synthetic benchmarks from four independent generators, and on two benchmarks derived from real measurements.

The rest of the paper is structured as follows. Section 2 reviews related work in amortized and descriptor-based causal discovery. Section 3 formalizes the problem setting, details the construction of the descriptor tensor, introduces our permutation-equivariant architecture, and contrasts it with its

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

per-edge predecessor. Section 4 describes the synthetic generator, the benchmark datasets, the baseline methods, and the evaluation protocol. Section 5 compares the model with the statistics it takes as input, with complete discovery algorithms and with the released pretrained models, across synthetic and real-world benchmarks. Finally, Section 6 concludes the paper and discusses limitations and directions for future work.

2 Related work

Amortized causal discovery replaces per-dataset search with a network trained once, on simulated systems whose graphs are known, to map a dataset to its structure; inference on new data is then a single forward pass, the pretrain-once, apply-anywhere strategy behind foundation models in language and vision (Bommasani, 2021). The first instances target static, i.i.d. data. CSIvA (Ke et al., 2022) reads observational and interventional samples with a transformer (Vaswani et al., 2017) and decodes the adjacency matrix autoregressively, one edge at a time. AVICI (Lorch et al., 2022) alternates attention over the sample and variable axes, pools over samples, and scores each edge from the inner product of two node embeddings; being equivariant over variables, one network serves graphs of any size, and an acyclicity penalty in training keeps its outputs close to DAGs. SEA (Wu et al., 2024) does not read raw samples at all: it feeds the estimates of classical algorithms on small variable subsets, plus global statistics, to a learned aggregator. CInA (Zhang et al., 2023) carries the recipe to effect estimation. What these models can learn is bounded by what they were trained on: Montagna et al. (2024) show on bivariate data that CSIvA falls to the level of a coin flip when the mechanism family changes between training and test, so the training distribution plays the role that explicit assumptions play for a classical method.

On time series the first amortized method, ACD (Löwe et al., 2022), amortizes in a different sense. An encoder infers a graph for each sample and a decoder shared across samples models the dynamics given that graph, trained by variational reconstruction on unlabelled series that share dynamics but differ in graph; it amortizes within a domain (one shared dynamics, many graphs), whereas in the setting studied here both the graph and the dynamics differ from system to system, and nothing at test time is assumed shared with training. Supervised transfer starts with Causal Pretraining (CP) (Stein et al., 2024), which maps a multivariate series directly to its lagged adjacency tensor, trained on synthetic nonlinear systems, and releases trained models for five backbones at five variables and three lags. (Section 5.2). Large Causal Models (LCM) (Kougioulis, 2025; Kougioulis et al., 2026) scale the approach, with a convolutional embedding and transformer encoder trained on 275k systems that mix synthetic generators with causal twins of real series, zero-padded to twelve variables and three lags. Both CP and LCM fix the output tensor at training time, so the variable count and lag window are frozen into the released model, and a larger system can be run only after discarding variables.

Closest to this work is the descriptor lineage, which learns from features of the data rather than raw values (Lopez-Paz et al., 2015; Guyon et al., 2019) and thereby buys invariance to scale, length and marginal shape. The method presented here builds on TD2C (Paldino & Bontempi, 2025) and its static predecessor D2C (Bontempi & Flauder, 2015), which classify information-theoretic descriptors of candidate lagged edges with a balanced random forest, trained on synthetic systems and applied to new data without any retraining.

3 Proposed method

Let $X = (X_1, \dots, X_V)$ be V jointly observed series of length T, with $X_j(t)$ the value of series j at time t. The task is to recover the lagged adjacency tensor $A \in \{0,1\}^{V \times V \times L}$, with $A[i,j,l] = 1$ iff $X_j(t-l)$ causes $X_i(t)$. We call one position (target i, source j, lag l) of this tensor a *cell*.

We frame recovery as supervised learning over simulated systems. The series enter through their lag embedding $Z \in \mathbb{R}^{(T-L)\times V(L+1)}$, whose row for time t is the window $(X(t), X(t-1), \dots, X(t-L))$. A fixed feature map ϕ computes, for every cell (i,j,l), dependence statistics between the columns $X_i(t)$ and $X_j(t-l)$ of Z, giving a descriptor tensor $D = \phi(Z) \in \mathbb{R}^{V \times V \times L \times C}$; the remaining columns enter only as conditioning sets. A network f_θ maps D to edge probabilities $\hat{A} = f_\theta(D) \in [0,1]^{V \times V \times L}$. One forward pass infers the full $\hat{A}$ for a system of any V and L.

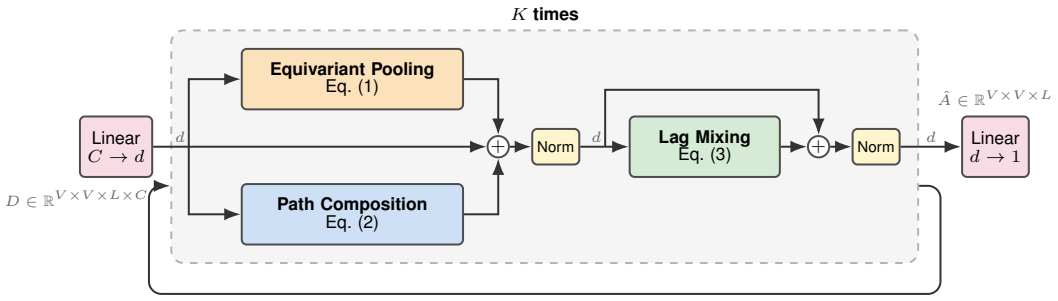

Figure 1: One residual block, applied K times. Numbers on arrows are channels per cell of the $V \times V \times L$ grid: every linear map acts on each cell independently, so the $V \cdot V \cdot L$ cells flow through as a batch, and only the three coloured operators couple them: the pooling views across the grid, the path composition, whose two legs are taken at lags summing to the output lag, and the depthwise convolution along the lag axis. The weights are independent of V and L.

We make four assumptions: (i) causal sufficiency, i.e. no latent confounders exist and all relevant variables are measured; (ii) no contemporaneous effects, i.e. every causal effect acts with a temporal delay; (iii) causal stationarity, i.e. mechanisms don't change over the observation window; and (iv) lag is bounded, i.e. no causal effects whose delay is greater than the explored observation window.

3.1 The descriptors tensor

Each *cell* receives $C = 8$ dependence statistics. Pearson correlation (Pearson, 1895) alone cannot separate a direct link from a mediated one or from shared history (C_1); partial correlation, conditioned on every other embedding column including the target's own past and estimated from a Ledoit-Wolf-shrunk covariance (Ledoit & Wolf, 2004), removes both (C_2). Both are also computed on Gaussian normal scores (Liu et al., 2009), which carry the same two readings for monotone nonlinear links (C_3, C_4). HSIC (Gretton et al., 2005), with Gaussian kernels at a shared median-heuristic bandwidth and normalised as centered kernel alignment, is included for mechanisms such as multiplicative gates and even functions (C_5). The binned correlation ratio (Rényi, 1959), is the fraction of the target's variance explained by any function of the source, estimated by averaging the target within bins of the source. It detects non-monotone dependence in the mean with a bounded, interpretable magnitude, and it is asymmetric, so it enters in both directions (C_6, C_7). The number of rows of the lagged sample (C_8), informs the network of how much data stands behind all the above. Every statistic is computed on at most 1024 rows per system, HSIC on at most 512, for efficiency.

Time enters through the columns of Z: every channel measures dependence at a specific delay, and the output is indexed by lag. The statistics treat the rows of Z as an exchangeable sample, which under stationarity discards no information about the mechanism. The resulting input is invariant to reordering the rows and to affine rescaling of any series; the null scaling makes values comparable, and the normal-score channels are further invariant to monotone reparameterisation of a marginal.

With $n \leq 1024$ rows, $m \leq 512$ for HSIC and embedding width $p = V(L+1)$, the descriptors cost $O(nV^2L^2 + V^3L^3 + m^2V^2L)$. The first term is the shrunk covariance of the p embedding columns, from which Pearson and partial correlation are read, together with the per-cell passes of the correlation ratio, which are dominated by it. The second is the single inversion of that covariance, which yields every partial correlation at once rather than one conditioning set per edge. The third is HSIC, whose kernel alignment is quadratic in the rows it uses, for each of the V^2L cells. .

3.2 Architecture

The network keeps one state vector $h_{ijl} \in \mathbb{R}^d$ per cell (target i, source j, lag l) of the $[V, V, L]$ tensor. Each input cell $x_{ijl} \in \mathbb{R}^C$ is standardised channel-wise and embedded by a shared linear map, $h^{(0)}_{ijl} = W_{\text{in}}\big((x_{ijl} - \mu) \oslash s\big)$, where $\mu, s \in \mathbb{R}^C$ are the per-channel mean and standard deviation computed once over all cells of the training archive, stored with the weights and never recomputed.

We propose three operators that couple cells of this tensor, each answering one question: how strong a dependence is relative to its context (*Equivariant pooling*), whether it is carried by a third variable (*Path composition*), and where along the lag axis it peaks (*Lag mixing*). Every other map acts on each cell independently; biases are omitted throughout. Figure 1 presents the structure visually.

Equivariant pooling. The strength of a dependence is informative only relative to its surroundings: the same correlation means little in a system where all pairs correlate and much in a sparse one. Direction is carried largely by the contrast between a cell and its mirrored partner (j, i, l), and the two variables' own histories are the first candidate explanation for any apparent link. The pooling operator supplies every cell with exactly these surroundings:

$$m_{ijl} = \mathrm{MLP}\big(\big[\, h_{ijl},\ h_{jil},\ \bar{h}_{i\cdot l},\ \bar{h}_{\cdot jl},\ \bar{h}_{\cdot\cdot l},\ h_{iil},\ h_{jjl}\,\big]\big). \tag{1}$$

Here $\bar{h}_{i\cdot l} = \frac{1}{V}\sum_j h_{ijl}$ is the target's row mean, $\bar{h}_{\cdot jl} = \frac{1}{V}\sum_i h_{ijl}$ the source's column mean and $\bar{h}_{\cdot\cdot l}$ the mean over all V^2 cells, each taken per lag and including the self-pairs; $[\cdot]$ concatenates to width $7d$, and the MLP is linear–GELU–linear back to width d. The seven terms are a subset of the basis of linear equivariant maps on 2-tensors (Maron et al., 2019), and the three means are the exchangeable-matrix layer of Hartford et al. (2018); here they are combined through an MLP.

Path composition. Separating a direct link from a mediated one requires information about third variables: the pair (i, j) is explained away when some variable k carries the dependence through the cells (i, k) and (k, j). In the temporal setting this test must also respect the arithmetic of delays. The path operator relates the pair (i, j) to every intermediate variable k:

$$p_{ijl} = W_o\Big(\tfrac{1}{V}\textstyle\sum_{k=1}^{V}\ \sum_{\substack{\lambda_1+\lambda_2=l\\ \lambda_1,\lambda_2\geq 1}} \big[U_a h_{ik\lambda_1}\big]_+ \odot \big[U_b h_{kj\lambda_2}\big]_+\Big). \tag{2}$$

For each intermediate variable k, the term pairs the evidence that k drives i at lag λ_1 (cell (i, k, λ_1)) with the evidence that j drives k at lag λ_2 (cell (k, j, λ_2)). The two cells play different roles in the chain, the first target and the second as source, so each is read through its own linear map, U_a and U_b. These maps select the features of a cell that matter for its role. The rectifier $[\cdot]_+$ keeps only positive evidence, and the elementwise product $\odot$ is large only in channels where both sides are present at once. A chain $X_j \to X_k \to X_i$ with these delays produces a dependence of X_i on X_j at lag $\lambda_1 + \lambda_2$, so the inner sum collects every pair of lags that adds up to the output lag l. The outer sum averages this evidence over all k, including i and j, and W_o maps the result back into the cell's state. At $l = 1$ the term is zero: no chain through an intermediate variable can act within one lag.

Lag mixing. Scores at neighbouring lags are correlated, because autocorrelated sources propagate dependence across adjacent delays. The lag operator mixes each cell with its two neighbouring lags:

$$g_{ijl} = \textstyle\sum_{\delta=-1}^{1} w_\delta \odot h_{ij(l+\delta)}. \tag{3}$$

This is a depthwise convolution along the lag axis with a kernel of width three and replicate padding; at $L = 1$ it reduces to a per-channel affine map.

Combining the operators. Pooling and path composition read the same state in parallel and are added to it residually; lag mixing then acts on the result, again residually:

$$\tilde{h}_{ijl} = \mathrm{LN}\big(h_{ijl} + m_{ijl} + p_{ijl}\big), \tag{4}$$

$$h^{+}_{ijl} = \mathrm{LN}\big(\tilde{h}_{ijl} + g_{ijl}[\tilde{h}]\big), \tag{5}$$

where $g[\tilde{h}]$ is Eq. (3) applied to $\tilde{h}$ and LN is layer normalisation over the channels of a cell. GELU and the rectifier of Eq. (2) are the block's only nonlinearities. Eqs. (4)–(5) form one block, which is applied K times with separate weights, $h^{(k+1)} = h^{(k)+}$. Each pass therefore updates every cell using neighbours that previous passes have already refined. A per-cell linear head then gives one logit per cell, $\hat{A}_{ijl} = \mathrm{sigmoid}(w_{\mathrm{out}}^{\top} h^{(K)}_{ijl} + b)$. Per layer, path composition costs $O(V^3L^2d)$, the second factor of L coming from the lag splits, and everything else $O(V^2Ld^2)$. The model does not decode jointly: cell probabilities carry no mutual-consistency or acyclicity constraint, unlike score-based search (Pamfil et al., 2020). Any joint structure in the output comes from the shared representation while the decision rule stays per-cell.

4 EXPERIMENTAL SETUP

4.1 SYNTHETIC DATA GENERATION

The generator draws each system's mechanisms from named pools: a linear pool and three nonlinear families, namely general autoregressions, mechanisms whose dependence is invisible at second order, and self-dominant resonators. Each nonlinear family is split into a training pool and a held-out pool; systems drawn from the held-out pools form the *internal* test set. Independently of the mechanisms we vary the parent topology (Bernoulli, hub, or a random recursive tree in which every variable has one parent), the weight scheme (uniform, Dirichlet or signed) and the noise scheme (shared or heteroscedastic); we call this triple the *scaffold*. The default scaffold is Bernoulli, uniform and shared; the randomised setting resamples each knob per system. The linear pool follows its own path and ignores the scaffold, so the randomisation reaches the 90% of systems drawn from the nonlinear families. Training uses a 100,000-system archive with $V \in [5, 40]$ and $T \in [60, 2000]$, lags $L \in [1, 5]$, edge density $\delta \in [0.03, 0.45]$ with explicit density regimes, three noise distributions (Gaussian, Laplace, uniform) at $\sigma \in [0.03, 0.30]$, the scaffold randomised over all three knobs per system, and a 20% slice of replicated systems.

4.2 BENCHMARK DATA

Synthetic CDML (Lawrence et al., 2020) provides 240 cases from an independent simulator, 5-12 variables, lags 1-3, prevalence 0.19 over 39,872 cells. We also include the following. *Kuramoto*: 200 simulations of five coupled phase oscillators (Kuramoto, 1975), from the generator (Löwe et al., 2022) bundled with the LCM release, 499 steps each, with a static coupling matrix as truth; we also regenerate it at coupling densities $p \in \{0.15, 0.30, 0.50\}$. *Synth_1*: 300 five-variable systems of 500 steps from Causal Pretraining's generator, reported only against the pretrained models (Table 3) because every method reaches its ceiling there. *Synth_230K*: the first 300 systems of the 230K set from the same generator, with 2-12 active variables padded to twelve, so that positives are rare: prevalence 0.019 over all 118,800 cells and 0.061 over the 36,198 cells between active variables. *TCDF finance* (Nauta et al., 2019): nine simulated market-return sets of 25 variables with lagged truths. Rivers and the finance sets exceed the twelve-variable cap of the released pretrained models.

Realistic *CausalTime* (Cheng et al., 2024) provides a realistic end: three subsets (20-36 variables) derived from real measured series, in 480 windows of only 40 steps each, far shorter than any training series, with dense static truths (prevalence 0.21-0.38); statistics are pooled over the windows. *Rivers* (Stein et al., 2025): a 42-variable river-discharge network of real gauge measurements whose truth is the drainage topology, at prevalence 0.024, sparser than anything in training. CausalTime and Rivers carry static truths and fix no orientation convention. We take each pair's maximum over lags as static score, and we take the orientation with higher lagged-correlation statistic and use it for every method alike.

4.3 BENCHMARK METHODS AND IMPLEMENTATION

A small-scale screening of operator variants fixed Eqs. (1) to (3); a one-factor-at-a-time sweep then showed that only depth pays, with a plateau from $K{=}4$, while width and weight decay are flat and a higher learning rate hurts at depth. The selected model has 1,633,921 parameters ($d = 192$, $K = 4$, MLP hidden width d, $C = 8$, learning rate 10^{-3}, weight decay 10^{-4}); three 30-epoch seeds reach 0.9351 ± 0.0006 validation AUROC. Width was chosen on a truncated-budget margin that does not survive full training, so the selected size is an upper bound on what the task needs (Section 5).

TD2C (Paldino & Bontempi, 2025) in a refactored implementation that reproduces the released one and runs roughly $25\times$ faster. PCMCI+ (Runge, 2020) with the ParCorr test and the maximum lag set to each case's maximum lag; DYNOTEARS (Pamfil et al., 2020); VAR-LiNGAM (Hyvärinen et al., 2010); and Lasso-Granger, all with default parameters. We also score seven lagged statistics directly as edge scores, the best of them on each set serves as the training-free reference (slightly unfair since post-hoc). Causal Pretraining (Stein et al., 2024), wherever its released form can be validly run (exactly five variables and three lags); LCM (Kougioulis et al., 2026) accepts at most twelve variables and three lags, but we could not obtain valid scores from its released code.

4.4 Evaluation protocol

We report AUROC and AUPRC over off-diagonal cells (Bradley, 1997); AUPRC is reported because prevalence is low and uneven across sets, and a method can hold its AUROC while collapsing in precision. Every method is scored on identical cells of the same systems, in one pass from one parse of the data, with the same masks, row budgets and orientation resolution; the few entries not measured are dashed. Learned models are reported as seed ranges, except on the paired internal selection, where a single seed was run; wherever a tie is claimed, the seed spread exceeds the gap. Head-to-head comparison with the per-edge predecessor and PCMCI+ uses a paired internal selection ($V \leq 12$, $L \leq 3$, 59,786 cells), where all methods run within budget. On Synth_230K, padding makes most cells trivially negative, so we score only the cells among active variables (36,198 cells, prevalence 0.061). Activity is defined from the labels, which favours every method alike. We do not score the self-lag cells (i, i, l): in the systems studied nearly every variable depends on its own past, so these cells are close to always-true, and benchmarks disagree on whether they belong to the ground truth at all. They remain in the input, because a variable's own history is the main thing to condition on.

5 Results

We read the results in two steps. Tables 1 and 2 compare the model with the statistics it takes as input and with complete discovery algorithms, on every set (Section 5.1). Table 3 compares it with the released pretrained models, on the few sets where they can run (Section 5.2). Throughout, our model is reported as a range over three training seeds, bold marks the best entry of a column with our range entering by its mean, and a dash marks a cell the corresponding run did not measure.

5.1 Against its inputs and classical algorithms

Table 1: Our model against every statistic in its input, the classical discovery algorithms and the per-edge predecessor (AUROC), on identical cells of every set. Bold marks the best entry in each column; our three-seed range enters by its mean. The discovery algorithms were not run on Synth_230K. †Paired internal selection, where PCMCI+ and the algorithms run; a single seed for ours. ‡Cells among active variables only (Section 4.4)

	internal	CDML	Kuramoto	Synth_230K‡	TCDF	CausalTime	Rivers
ours (3 seeds)	**.972**†	**.829-.832**	.724-.755	**.856-.858**	.993	.456-.544	.497-.688
Discovery algorithms and the per-edge predecessor							
TD2C (per-edge predecessor)	.787	.768	.641	-	.798	.605	.768
PCMCI+	.810	.813	.848	-	.810	.544	.551
DYNOTEARS	.841	.816	.751	-	.915	.637	.608
VAR-LiNGAM	.796	.794	.803	-	.990	.452	**.774**
Lasso-Granger	.830	.777	**.972**	-	.994	.526	.521
Statistics of the input, scored directly							
correlation	.769	.812	.868	.744	.824	.793	.574
partial correlation	.804	.802	.913	.708	**.997**	.566	.526
rank correlation	.767	.814	.854	.740	.851	.810	.622
rank partial corr.	.805	.806	.777	.715	.996	.528	.555
HSIC	.774	.778	.842	.828	.887	**.821**	.651
var. reduction, fwd	.744	.765	.787	.788	.830	.807	.626
var. reduction, bwd	.746	.757	.799	.768	.744	.810	.573

Tables 1 (AUROC) and 2 (AUPRC) share one layout. The first row is our model. The middle block holds complete discovery algorithms, including the per-edge predecessor TD2C. The bottom block holds each statistic of the descriptor tensor scored directly as an edge score; its best entry per column is the training-free reference, chosen post hoc and therefore favourable to the statistics. The tables answer two questions: does the network do more than select the best of its own inputs, and how does it compare with methods that search for a structure? The seven columns fall into four groups with different answers.

Table 2: AUPRC counterpart of Table 1. Same cells, same protocol; prevalence per set is stated in Section 4.2. Cells the corresponding run did not measure are dashed. ‡Cells among active variables only (Section 4.4).

	internal	CDML	Kuramoto	Synth_230K‡	TCDF	CausalTime	Rivers
ours (3 seeds)	**.898**†	**.715-.721**	.357-.386	**.650-.655**	.820-.887	.234-.289	.080-.211
Discovery algorithms and the per-edge predecessor							
TD2C (per-edge predecessor)	.437	.557	-	-	.106	.329	.083
PCMCI+	.632	.706	.514	-	.556	.283	.034
DYNOTEARS	.652	.700	-	-	.688	.369	.067
VAR-LiNGAM	.333	.571	-	-	.794	.289	.087
Lasso-Granger	.614	.650	-	-	.923	.293	.036
Statistics of the input, scored directly							
correlation	.303	.699	.423	.361	.216	.742	**.185**
partial correlation	.619	.682	**.535**	.466	**.935**	.333	.126
rank correlation	.311	.702	.409	.378	.420	.780	.116
rank partial corr.	.618	.686	.348	.478	.918	.294	.044
HSIC	.301	.637	.394	.365	.438	**.792**	.182
var. reduction, fwd	.295	.616	.343	.397	.276	.756	.138
var. reduction, bwd	.278	.589	.349	.343	.199	.750	.122

On the paired internal cells, the only selection where every algorithm runs, the model reaches 0.972 AUROC against 0.841 for DYNOTEARS, the strongest algorithm, 0.805 for the best statistic and 0.787 for TD2C; in AUPRC the gap widens (0.898 against 0.652 and 0.619). On the full internal split, too large for the algorithms, its three seeds land within 0.0002 of one another (0.9371-0.9373) against 0.685 for the best statistic. The lead survives an independent simulator: on CDML the worst seed (0.829) clears the best statistic (rank correlation, 0.814), DYNOTEARS (0.816) and PCMCI+ (0.813), and gains $+0.061$ to $+0.064$ over TD2C. The CDML margin is narrower in AUPRC (0.715-0.721 against 0.706 for PCMCI+), but the order is the same. On the active cells of Synth_230K, a third generator, the model (0.856-0.858) leads the best statistic, HSIC (0.828), and the AUPRC margin is large (0.650-0.655 against 0.478).

TCDF's market-return sets suit linear methods. Partial correlation (0.997), Lasso-Granger (0.994), the model (0.993) and VAR-LiNGAM (0.990) are indistinguishable in AUROC, and the model gains $+0.195$ over TD2C. AUPRC separates them slightly: partial correlation (0.935) and Lasso-Granger (0.923) rank the few true edges higher than the model (0.820-0.887). When the right statistic is linear and known, learning adds nothing.

Kuramoto is the one simulated set where the order reverses. The model (0.724-0.755) is below Lasso-Granger (0.972), partial correlation (0.913), PCMCI+ (0.848) and VAR-LiNGAM (0.803), level with DYNOTEARS (0.751), and ahead only of TD2C (0.641); AUPRC agrees (0.357-0.386 against 0.535 for partial correlation).

On CausalTime the model is at chance (0.456-0.544), and the structure-learning methods do little better, between 0.452 (VAR-LiNGAM) and 0.637 (DYNOTEARS). The marginal statistics, by contrast, hold up: HSIC (0.821), rank correlation (0.810) and both variance reductions (0.807-0.810), all of them channels of the model's own input. AUPRC makes the collapse sharper: the model's 0.234-0.289 does not exceed the set's prevalence (0.21-0.38), while HSIC reaches 0.792. On Rivers the model spans 0.497-0.688, a wide seed spread; VAR-LiNGAM (0.774), whose non-Gaussianity assumption suits heavy-tailed discharge data, and TD2C (0.768) lead, and the best statistic in the table is HSIC (0.651). In AUPRC, where prevalence is 0.024, the picture is less one-sided: the model's 0.080-0.211 is level with the best algorithm (VAR-LiNGAM, 0.087) at its weakest seed and above every algorithm otherwise, the leading AUROC methods included; only the marginal correlation (0.185) and HSIC (0.182) come close to its best seed. An earlier variant of the model sat well below chance on Rivers, which a transposed-truth check traced to a systematic orientation inversion; the lag-consistent path operator of Eq. (2) removes the inversion, although the weakest seed (0.497) remains at chance.

5.2 AGAINST RELEASED PRETRAINED MODELS

Table 3: Our model against the released Causal Pretraining (CP) models. Every row of a column is scored on identical cells: 2,240 on the 48 five-variable CDML cases, 18,000 on Synth_1, 4,000 on Kuramoto at coupling density 0.15. CP, PCMCI+ and statistic rows are the archived one-parse runs; ours is the selected model over three seeds, entering the bold comparison by its mean. LCM is absent because we could not obtain valid scores from its released code.

	CDML, 5 variables		Synth_1		Kuramoto	
	AUROC	AUPRC	AUROC	AUPRC	AUROC	AUPRC
ours (3 seeds)	.852-.873	.788-.794	.989-.993	.942-.946	.724-.755	.357-.386
CP, transformer	.859	.778	.999	.988	.644	.300
CP, GRU	.863	.787	.999	.987	.617	.282
CP, bidirectional GRU	.851	.779	.998	.981	.571	.219
PCMCI+	.870	.809	**1.000**	**.994**	.848	.514
best statistic	**.875**	**.813**	**1.000**	.993	**.913**	**.535**

The released Causal Pretraining models accept exactly five variables and three lags, so Table 3 is restricted to the three sets that fit: the 48 five-variable CDML cases, Synth_1 from CP's own generator, and five-oscillator Kuramoto. None of TCDF, CausalTime or Rivers can be scored, since their 20 to 42 variables exceed both CP's cap and LCM's cap of twelve. The table therefore compares the models only where the pretrained ones are strongest. On five-variable CDML all methods lie within 0.025 AUROC (best statistic 0.875, PCMCI+ 0.870, CP 0.851-0.863, ours 0.852-0.873), and AUPRC is just as bunched: small, data-rich graphs are where classical methods peak and an amortized model has least to add. On Synth_1, CP's training distribution, every method is at the ceiling. Kuramoto separates the learned models: all three CP models fall to 0.571-0.644, below every one of our seeds (0.724-0.755), with the same ranking in AUPRC. Neither learned family, however, reaches partial correlation (0.913) or PCMCI+ (0.848) on this set, consistent with Table 1. LCM does not appear: we could not obtain valid scores from its released code.

6 CONCLUSIONS AND FUTURE WORK

We reformulated descriptor-based causal discovery as a single permutation-equivariant pass over a tensor of dependence statistics indexed by target, source and lag. No weight depends on the number of variables or lags, so one trained model outputs the full lagged adjacency of systems with 20 to 42 variables, beyond the reach of the released pretrained models. The network does more than select among its own inputs. On the paired internal cells it reaches 0.972 AUROC, against 0.841 for DYNOTEARS, the strongest algorithm, 0.805 for the best input statistic and 0.787 for its per-edge predecessor. On CDML, from an independent simulator, every seed clears every classical algorithm and every input statistic. Against the released Causal Pretraining models the network is level on five-variable CDML and ahead on Kuramoto, where it still trails partial correlation, PCMCI+ and Lasso-Granger. The realistic gap remains open. On CausalTime the model is at chance and no structure-learning method exceeds 0.64 AUROC, although statistics in its own input reach 0.81–0.82. On Rivers it stays behind VAR-LiNGAM and the per-edge predecessor in AUROC. We hypothesise that the next lever is the data rather than the model: training on, or fine-tuning towards, realistic series. We leave this to future work, together with contemporaneous effects and latent confounding.

REPRODUCIBILITY STATEMENT

Code is provided upon acceptance.

AI USE STATEMENT

In this work, we used generative AI tools for AI-assisted coding and to improve readability. We have not used generative AI tools to propose or refine hypotheses. We have reviewed all AI-assisted work. Every reported number is regenerated by scripts in the repository and traced to a result file; AI-assisted code was tested by at least one author. We take responsibility for the final content of this work, including text, claims and artifacts produced with the aid of generative AI.

REFERENCES

Rishi Bommasani. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

Gianluca Bontempi and Maxime Flauder. From dependency to causality: A machine learning approach. *Journal of Machine Learning Research*, 16(74):2437–2457, 2015.

Andrew P Bradley. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159, 1997.

Yuxiao Cheng, Ziqian Wang, Tingxiong Xiao, Qin Zhong, Jinli Suo, and Kunlun He. Causaltime: Realistically generated time-series for benchmarking of causal discovery. In *International Conference on Learning Representations*, 2024.

Arthur Gretton, Olivier Bousquet, Alex Smola, and Bernhard Schölkopf. Measuring statistical dependence with Hilbert-Schmidt norms. In *Algorithmic Learning Theory*, volume 3734 of *Lecture Notes in Computer Science*, pp. 63–77. Springer, 2005.

Isabelle Guyon, Alexander Statnikov, and Berna Bakir Batu (eds.). *Cause Effect Pairs in Machine Learning*. The Springer Series on Challenges in Machine Learning. Springer, 2019.

Jason Hartford, Devon Graham, Kevin Leyton-Brown, and Siamak Ravanbakhsh. Deep models of interactions across sets. In *International Conference on Machine Learning*, pp. 1909–1918, 2018.

Aapo Hyvärinen, Kun Zhang, Shohei Shimizu, and Patrik O Hoyer. Estimation of a structural vector autoregression model using non-gaussianity. *Journal of Machine Learning Research*, 11(5), 2010.

Nan Rosemary Ke, Silvia Chiappa, Jane Wang, Jorg Bornschein, Theophane Weber, Anirudh Goyal, Matthew Botvinick, Michael Mozer, and Danilo Jimenez Rezende. Learning to induce causal structure. *arXiv preprint arXiv:2204.04875*, 2022.

Nikolaos Kougioulis. Large causal models for temporal causal discovery. Master's thesis, University of Crete, 2025.

Nikolaos Kougioulis, Nikolaos Gkorgkolis, MingXue Wang, Bora Caglayan, Dario Simionato, Andrea Tonon, and Ioannis Tsamardinos. Large causal models for temporal causal discovery. *arXiv preprint arXiv:2602.18662*, 2026. Venue unconfirmed: the source ships llncs.cls and splncs04.bst, which suggests a Springer LNCS venue rather than a bare preprint. Confirm before submission (OUTLINE.md, open item O15).

Yoshiki Kuramoto. Self-entrainment of a population of coupled non-linear oscillators. In *International Symposium on Mathematical Problems in Theoretical Physics: Kyoto University, Kyoto/Japan*. Springer, 1975.

Andrew R. Lawrence, Marcus Kaiser, Rui Sampaio, and Maksim Sipos. Data generating process to evaluate causal discovery techniques for time series data. *NeurIPS Causal Discovery & Causality-Inspired Machine Learning Workshop*, 2020.

Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88(2):365–411, 2004. ISSN 0047-259X. doi: https://doi.org/10.1016/S0047-259X(03)00096-4.

Han Liu, John Lafferty, and Larry Wasserman. The nonparanormal: Semiparametric estimation of high dimensional undirected graphs. *Journal of Machine Learning Research*, 10:2295–2328, 2009.

David Lopez-Paz, Krikamol Muandet, Bernhard Schölkopf, and Ilya Tolstikhin. Towards a learning theory of cause-effect inference. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *PMLR*, pp. 1452–1461, 2015.

Lars Lorch, Scott Sussex, Jonas Rothfuss, Andreas Krause, and Bernhard Schölkopf. Amortized inference for causal structure learning. *Advances in Neural Information Processing Systems*, 35: 13104–13118, 2022.

Sindy Löwe, David Madras, Richard Zemel, and Max Welling. Amortized causal discovery: Learning to infer causal graphs from time-series data. In *Conference on Causal Learning and Reasoning*, pp. 509–525. PMLR, 2022.

Haggai Maron, Heli Ben-Hamu, Nadav Shamir, and Yaron Lipman. Invariant and equivariant graph networks. In *International Conference on Learning Representations*, 2019.

Francesco Montagna, Max Cairney-Leeming, Dhanya Sridhar, and Francesco Locatello. Demystifying amortized causal discovery with Transformers. *arXiv preprint arXiv:2405.16924*, 2024.

Meike Nauta, Doina Bucur, and Christin Seifert. Causal discovery with attention-based convolutional neural networks. *Machine Learning and Knowledge Extraction*, 1(1):19, 2019.

Gian Marco Paldino and Gianluca Bontempi. Causal discovery in multivariate time series through mutual information featurization. *arXiv preprint arXiv:2508.01848*, 2025.

Roxana Pamfil, Nisara Sriwattanaworachai, Shaan Desai, Philip Pilgerstorfer, Konstantinos Georgatzis, Paul Beaumont, and Bryon Aragam. DYNOTEARS: Structure learning from time-series data. In *International Conference on Artificial Intelligence and Statistics*, pp. 1595–1605. Proceedings of Machine Learning Research, 2020.

Karl Pearson. Note on regression and inheritance in the case of two parents. *Proceedings of the Royal Society of London*, 58:240–242, 1895.

Alfréd Rényi. On measures of dependence. *Acta Mathematica Academiae Scientiarum Hungaricae*, 10:441–451, 1959.

Jakob Runge. Discovering contemporaneous and lagged causal relations in autocorrelated nonlinear time series datasets. In *Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI)*, volume 124, pp. 1388–1397, Online, 2020. PMLR. URL `https://proceedings.mlr.press/v124/runge20a.html`.

Gideon Stein, Maha Shadaydeh, and Joachim Denzler. Embracing the black box: Heading towards foundation models for causal discovery from time series data. *AAAI Workshop (AI4TS)*, 2024.

Gideon Stein, Maha Shadaydeh, Jan Blunk, Niklas Penzel, and Joachim Denzler. Causalrivers – scaling up benchmarking of causal discovery for real-world time-series. In *International Conference on Learning Representations*, 2025.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017.

Menghua Wu, Yujia Bao, Regina Barzilay, and Tommi Jaakkola. Sample, estimate, aggregate: A recipe for causal discovery foundation models. In *ICLR Workshop on Machine Learning for Genomics Explorations*, 2024.

Jiaqi Zhang, Joel Jennings, Agrin Hilmkil, Nick Pawlowski, Cheng Zhang, and Chao Ma. Towards causal foundation model: On duality between causal inference and attention. *arXiv preprint arXiv:2310.00809*, 2023.