Nonparametric Greedy Equivalence Search with Prior-Fitted Networks

Mateusz Gajewski* MG96272@GMAIL.COM
AKCES NCBR & Poznan University of Technology

Mateusz Olko* MATEUSZ.OLKO@GMAIL.COM
AKCES NCBR & University of Warsaw

Editors: Bijan Mazaheri and Niels Richard Hansen

Abstract

Greedy equivalence search is among the most widely used methods for causal discovery. Recent work has established a theoretical foundation for extending GES to nonparametric models, an approach that relies on Bayesian likelihood estimation. In parallel, the prior–data fitted network paradigm was introduced, demonstrating superior accuracy and computational efficiency over standard tabular models across a wide range of predictive tasks, while naturally providing Bayesian predictive posteriors. In this paper, we integrate TabPFN as a Bayesian likelihood estimator within nonparametric GES and conduct an extensive empirical evaluation of the resulting approach. The proposed method consistently outperforms state-of-the-art nonparametric causal discovery methods on a range of synthetic, simulated, and real-world datasets. These results highlight the PFN paradigm as a natural and promising direction for advancing causal discovery in complex real-world applications.

1. Introduction

Causal discovery is the central problem in scientific inquiry, as it aims to uncover the underlying mechanisms governing complex systems from observational data. By learning causal structure, scientists can move beyond correlation to enable principled causal reasoning, robustness to distribution shifts, and generalization across domains. To address real-world problems, causal discovery methods must function in nonparametric settings. Parametric assumptions, while convenient for theoretical analysis, often fail to capture the complexity of natural systems, limiting the reliability of the resulting causal models. Consequently, there is a growing need for causal discovery algorithms that remain valid under minimal modeling assumptions.

Greedy Equivalence Search (GES) is one of the most theoretically well-understood methods for causal discovery, but until recently, its guarantees were largely limited to linear parametric models. Recent work established a theoretical foundation for GES with nonparametric likelihoods (Aragam, 2024), requiring only modest modifications to the original algorithm and minimal assumptions on the data-generating process. Crucially, this extension relies on a Bayesian treatment of likelihood estimation, in which uncertainty over unknown functional relationships must be integrated out through a posterior predictive score rather than approximated by point estimates. While Bayesian neural networks offer a natural instantiation of this principle, they are often computationally expensive, sensitive to hyperparameters, and difficult to optimize, motivating the need for more practical Bayesian likelihood models.

In parallel, the tabular data community has developed a class of foundational models based on the prior–data fitted network (PFN) paradigm. These models are pretrained on data generated from a

* Equal contribution

specified prior over data-generating processes and perform Bayesian inference at test time (Reuter et al., 2025). A prominent example is TabPFN, which outperforms standard tabular models across a wide range of predictive tasks in both accuracy and computational efficiency (Hollmann et al., 2025; Prior Labs, 2025; Hollmann et al., 2023). Importantly for our setting, TabPFN yields uncertainty-aware predictive likelihoods by implicitly marginalizing over unknown functional relationships, making it a natural and practical likelihood model for nonparametric GES.

In this paper, we incorporate TabPFN into the nonparametric GES algorithm and provide an extensive empirical evaluation of the effectiveness of nonparametric GES paired with TabPFN as a Bayesian likelihood estimator. We compare the proposed approach against a range of state-of-the-art nonparametric causal discovery methods, assessing both structural accuracy and computational efficiency across synthetic, simulated, and real-world datasets. The results show that our approach is consistently more accurate, decreasing structural error by up to 92%, while remaining computationally competitive with methods based on continuous optimization. These findings point toward the PFN paradigm as a natural and promising direction for advancing causal discovery in complex real-world applications.

Our contributions include:

- We design PF-GES, a principled and efficient nonparametric causal discovery method by linking recent advances in causal discovery with equivalence search and Bayesian inference for tabular data.
- The proposed approach outperforms state-of-the-art nonparametric causal discovery methods on a series of synthetic, simulated, and real-world datasets.
- We address key practical challenges in applying nonparametric GES efficiently and demonstrate the impact of our solutions through ablation studies.

The remainder of the paper is structured as follows. In Section 2 we introduce previous work that facilitates our contribution. In Section 3 we provide our technical contribution together with the description of the proposed approach. Further in Section 4, we discuss existing approaches for causal discovery in a nonparametric setting and applications of the PFN paradigm in causality. Finally, we provide an extensive empirical evaluation of our method in Section 5.

2. Background

2.1. Prior-Fitted Networks and TabPFN

Prior-Fitted Networks are a paradigm for training Bayesian models that has gained recognition and positive community feedback (Müller et al., 2025). PFNs are pre-trained models trained on synthetic data generated from a specified prior that are designed to approximate the posterior predictive distribution (PPD). The prior $\pi(f)$ defines a space of hypotheses $\mathcal{F}$ on the relationship of a set of inputs to the outputs. Each hypothesis can be understood as a mechnism that define a data distribution from which one can sample a dataset. The PPD can be expressed as integral over the space of the hypotheses

$$p(y|x, D) \propto \int_{\mathcal{F}} p(y|x, f)p(D \mid f)\pi(f)\, df. \tag{1}$$

The PFNs perform Bayesian prediction of PPD under a mechanism prior $\pi(f)$ (Müller et al., 2022). In practice, the PFN training objective is defined on synthetic datasets sampled from a prior

$p(D) = \mathbb{E}_{f\sim\pi(f)}p(D|f)$. For a single test point $\{(x_{\text{test}}, y_{\text{test}})\} = D_{\text{test}}$, the loss minimized during training is

$$\mathcal{L}_{\text{PFN}} = \mathbb{E}_{\{(x_{\text{test}},y_{\text{test}})\}\cup D_{\text{train}}\sim p(D)}\left[-\log q_\theta(y_{\text{test}} \mid x_{\text{test}}, D_{\text{train}})\right], \tag{2}$$

where θ are parameters of the PFN model. During inference, PFNs perform in-context prediction of the PPD given input data. This paradigm is was also described under the name Neural Processes (Garnelo et al., 2018).

TabPFN is a pre-trained transformer for classification and regression on tabular data (Hollmann et al., 2023) that builds on the PFN idea. The authors used a synthetic prior based on Structural Causal Models and Bayesian Networks. The method demonstrates excellent accuracy and speed, outperforming classical approaches like boosted trees at regression and classification tasks. Research in transformer-based models for tabular data remains highly active, with ongoing improvements focusing on computational efficiency, scaling to larger datasets, and handling higher feature counts (Prior Labs, 2025; Kolberg et al., 2025).

2.2. Causal Graphical Models

A causal graphical model is a directed acyclic graph (DAG) $G = (V, E)$ where nodes $V = \{X_1, X_2, \ldots, X_n\}$ represent random variables and directed edges E represent direct causal relationships. The joint probability distribution over the variables factorizes according to the graph structure as:

$$P(X_1, X_2, \ldots, X_n) = \prod_{i=1}^{n} P(X_i \mid \text{Pa}(X_i)) \tag{3}$$

where $\text{Pa}(X_i)$ denotes the set of parent nodes of X_i in G.

2.3. Causal discovery

The goal of causal discovery is to recover the underlying graph structure of the data-generating process given data sampled from a joint distribution P_X. Without further assumptions, this is only possible up to a class of graphs that encode the same conditional independence statements called the Markov Equivalence Class (MEC).

Existing approaches to causal discovery can be broadly divided into two classes. Constraint-based methods, such as the PC algorithm, infer causal structure by performing a sequence of conditional independence tests. Likelihood-based (or score-based) methods formulate causal discovery as an optimization problem, seeking the graph that maximizes a likelihood-based score.

Greedy Equivalence Search The Greedy Equivalence Search takes the score-based approach to causal discovery (Meek, 1997). It is a well-established algorithm that has an efficient way of searching through the space of possible structures. The algorithm begins with an empty graph and at each step modifies the graph by edge insertions or deletions to improve the score.

In the forward phase, GES begins with an empty graph, considers all graphs in the neighborhood $\text{nb}_+(G)$ (graph G with an additional edge inserted) at each step, and picks the highest scoring graph until no further improvements in score are possible. In the backward phase, it considers graphs from $\text{nb}_-(G)$ (G with one edge deleted) and picks the graph with the highest score until no deletion increases the score. Later, Hauser and Bühlmann (2012) introduced an additional turning phase, which can improve results in the finite samples regime. In this phase, the GES algorithm

identifies all possible edge reversals and applies them until no further score improvement is possible. The algorithm was shown to be consistent (Chickering, 2002) when the score is locally consistent. Formally, the forward and backward neighborhoods of a graph G with respect to its MEC $\mathcal{C}(G)$ are defined as:

$$\text{nb}_+(G) := \bigcup_{H \in \mathcal{C}(G)} \{H_{k,j} : (k,j) \notin E(H)\}, \tag{4}$$

$$\text{nb}_-(G) := \bigcup_{H \in \mathcal{C}(G)} \{H_{-k,j} : (k,j) \in E(H)\}, \tag{5}$$

where $E(H)$ denotes set of edges of graph H, $H_{k,j}$ denotes the graph obtained by adding edge (k,j) to H, and $H_{-k,j}$ denotes the graph obtained by removing edge (k,j) from H.

The score used with the GES algorithm is the Bayesian Information Criterion (BIC) (Schwarz, 1978). BIC is a model selection criterion that trades off log-likelihood and model complexity, which was proven to be consistent for curved exponential families (Chickering, 2002). Models with higher BIC are preferred. A common model class used for continuous data with the BIC is the class of Gaussian linear models, where BIC is consistent under mild assumptions (Chickering, 2002; Haughton, 1988). The key challenge in applying BIC to more complex datasets is the assumptions that must be made to calculate BIC like additivity constraints, independent and/or Gaussian noise, linearity, or other semi-parametric assumptions (Aragam, 2024).

Nonparametric GES Aragam (2024) established a theoretical foundation for applying GES to nonparametric settings by modifying the standard algorithm. Rather than selecting the highest-scoring graph in terms of BIC score from the neighborhood, their approach uses a statistical test to determine whether a candidate graph H is preferred over the current graph G, and accepts any graph that passes this test with respect to some threshold λ. In the forward phase, the algorithm iterates over all preferred graphs in $\text{nb}^+(G)$, whereas in the backward phase it considers preferred graphs in $\text{nb}^-(G)$.

The test is constructed using the posterior odds ratio. Given two competing DAGs G and H, we define the test statistic

$$\text{PR}(G,H) = \frac{P(G \mid X)}{P(H \mid X)}, \tag{6}$$

where the model posterior is given by

$$P(G \mid X) \propto \int_{\mathcal{F}(G)} P(X \mid f)\pi(f \mid G)\pi(G)\, df. \tag{7}$$

Here, $\mathcal{F}(G)$ denotes the space of all functions $f = (f_1, \ldots, f_d)$ compatible with graph G, where each f_j represents a conditional density function for node j given its parents in G. The framework assumes only that these conditional densities are Lipschitz continuous, imposing no parametric or additive structure (Aragam, 2024). The prior $\pi(f \mid G)$ is a structural prior that factorizes over the conditional densities:

$$\pi(f \mid G) = \prod_{\ell=1}^{d} \pi_\ell(f_\ell \mid \text{pa}_G(\ell)), \tag{8}$$

where each $\pi_\ell(f_\ell \mid \text{pa}_G(\ell))$ is a prior over the conditional density f_ℓ given its parent set $\text{pa}_G(\ell)$. The test $\varphi_\lambda(X; G, H)$ with threshold $\lambda > 0$ is then defined as

$$\varphi_\lambda(X; G, H) = 1\left(\text{PR}(G,H) > \lambda\right), \tag{9}$$

where $\varphi_\lambda(X; G, H) = 1$ indicates that G is preferred over H. The graph prior $\pi(G)$ appropriately penalizes graph complexity in nonparametric models.

3. Nonparametric GES with PFN

Implementing the nonparametric GES approach requires solving two practical challenges. First, to conduct the test φ_λ, one needs access to full Bayesian posterior $P(G \mid X)$, as expressed by Equation 7. However, nonlinear and flexible models such as Bayesian Neural Networks (BNNs) are computationally expensive, both during training and inference, as they require multiple samples to approximate the posterior (Jia et al., 2020; Duvenaud et al., 2020; Jacobs et al., 2023). Selecting priors for these networks is also challenging (Nalisnick, 2018; Fortuin, 2022). The beliefs about how functions should behave have a highly non-trivial relationship to the distributions over the weights of neural networks (Fortuin, 2022). Therefore, the simplest priors, such as isotropic Gaussians, are typically used; however, they do not reflect true prior beliefs and can lead to suboptimal performance (Delaunoy et al., 2021; Fortuin et al., 2022).

Second, the evaluation of the posterior $P(G \mid X)$ on various graphs needs to be computationally efficient. Thanks to DAG factorization (recall Equation 3), one does not need to fit a posterior for each graph separately, but rather for each variable and each possible parent set. The simplest solution would be to train a separate model from scratch for each parent set configuration. However, this approach is computationally prohibitive and scales poorly, as the number of possible parent sets grows exponentially with the number of nodes. Nevertheless, we implemented and tested this approach in Section 5.4 to showcase its limitations. We believe both challenges can be resolved using PFN likelihood estimators and describe our solution below.

3.1. Method

Assumptions The proposed approach does not make any parametric assumptions about the data geneating mechanisms, however standard causal assumptions are needed to ensure recoverabilty of the Markov Equivalence Class. Our method relies on the data being faithful, causally Markov and causally sufficient.

Likelihood evaluation Evaluating the posterior ratio test (Equation 6) requires computing the posterior predictive likelihood $p(X|G)$. This decomposes accordign to the graph structure into:

$$p(X|G) = \sum_{j \in V} p(X_j | X_{pa(j)}, G) \tag{10}$$

for each node j under varying parent set configurations. Throughout the search, the algorithm may need to evaluate a substantial number of different parent sets per node, for a d-node graph, up to 2^{d-1} configurations per node.

We compute $p(X|G)$ using PFN with the following procedure.

For each parent set $\text{pa}(j)$, we construct training data from the observed dataset X by treating X_j as the target and $X_{\text{pa}(j)}$ as features. TabPFN then approximates the posterior predictive distribution via in-context learning: it takes these training samples as input and outputs the predictive distribution in a single forward pass, without model training or sampling. This amortized inference is essential for the computational efficiency of our method.

To avoid overfitting, we employ cross-validation: we partition the data into five folds of equal size , use each fold in turn for computing the log-likelihood in the posterior ratio while providing the remaining folds as context to TabPFN.

Regarding the graph prior, we used priors proposed in (Aragam, 2024), that have two parameters γ - controlling the scale of prior and Γ controlling how much do we regularize the structures complexity.

Algorithm 1 Likelihood Evaluation with PFN

Require: Dataset X, PFN likelihood model M, target variable X_j, feature columns $X_{\mathrm{pa}(j)}$
Ensure: Concatenated likelihood estimates $\mathcal{L}$

```
Split dataset X into five folds: {X^1, X^2, X^3, X^4, X^5}
for k = 1 to 5 do
    Training set X^train ← ⋃_{i≠k} X^i
    Test set X^test ← X^k
    Predict likelihoods L^k = p_M( X_j^test | X_pa(j)^test, X^train )
    Append L^k to L
end for
return L
```

Practical implementation of graph selection The nonparametric GES algorithm of Aragam (2024) uses a statistical test to determine whether a candidate graph H should be accepted over the current graph G based on the statistical test from Equation 9). The original formulation accepts *any* graph that passes this test, without specifying how to choose among multiple candidates that may simultaneously satisfy the criterion, which is a common occurrence in practice. In the simplest implementation, the algorithm iterates over candidate operators in a fixed order and accepts the first graph that passes the test.

Selecting graphs arbitrarily or in enumeration order can be problematic: in the forward phase, this may lead to accepting edges that create unnecessarily dense intermediate graphs, which both degrade structural accuracy and substantially increase computation time for subsequent search steps. We propose a simple but effective modification: rather than accepting the first passing candidate, we evaluate *all* graphs in the neighborhood and select the one with the highest posterior ratio.

This greedy selection strategy is consistent with the nonparametric framework; we accept only graphs that are statistically preferred, while providing a principled way to select the most promising candidate when multiple options exist. As shown in Table 2, this modification yields substantial improvements in structural accuracy and enables better scaling to larger graphs. We attribute this improvement to more confident structural decisions during the search: by selecting graphs with the strongest statistical evidence at each step, the algorithm makes more reliable progress through the search space. Importantly, this modification preserves all theoretical guarantees of nonparametric GES since we maintain the requirement that selected graphs must pass the statistical test.

4. Related work

Nonparametric constraint-based causal discovery The constraint-based methods of causal discovery rely on conducting a conditional independence test to identify the graph G. A classic and widely recognized method in this class is the PC algorithm (Spirtes et al., 2000). It performs

conditional independence tests, increasing the conditioning set in each step, to eliminate non-existent edges. In the simplest case, the test involves using the Fisher-Z test with the partial linear correlation coefficient, but can be easily extended by using more general conditional independence tests like the Kernel Conditional Independence test (KCI) (Zhang et al., 2012). The significant drawback of this class of methods is the complexity rapidly growing with the number of nodes (Spirtes et al., 2000; Le et al., 2016).

Neural score-based causal discovery The score optimization problem can be framed as a continuous non-convex optimization problem, enabling the application of gradient-based methods to causal discovery. This approach was first introduced in NOTEARS (Zheng et al., 2018). The method uses neural networks for density estimation and continuous optimization instead of search. The main component is a constrained optimization of the continuous adjacency matrix with respect to a differentiable acyclicity constraint. Brouillard et al. (2020) extended the idea to interventional data. Further, improved acyclicity constraints were proposed, and the optimization procedure has been adjusted to be even more computationally affordable (Nazaret et al., 2024; Bello et al., 2022; Yu et al., 2021; Lee et al., 2020).

NOTEARS and follow-up methods use a common scoring function. Namely, the penalized likelihood score:

$$s(G, D) = \log p(D \mid G) - \lambda |G|, \tag{11}$$

where G denotes the candidate graph, D the observed dataset, $|G|$ the number of edges in G, and $\lambda > 0$ a regularization coefficient that penalizes overly dense structures. Optimizing the score in Equation 11 with sufficiently small λ is guaranteed to recover a member of the MEC of the true structure (Brouillard et al., 2020).

In contrast to these approaches, nonparametric GES preserves a discrete, score-consistent search over Markov equivalence classes while enabling nonparametric likelihoods, avoiding nonconvex optimization and offering stronger theoretical grounding with competitive empirical efficiency.

Amortized causal discovery Some methods learn to predict causal graphs directly from data, allowing fast inference on new datasets without solving each problem from scratch. For example, AVICI (Lorch et al., 2022) uses neural networks trained on simulated data, while CSIvA (Ke et al., 2023) incorporates variational learning and structural biases. BCNP (Dhir et al.) further improves this approach by modeling uncertainty more explicitly. These methods trade higher training cost for efficient and reusable inference.

PFN for causal discovery and inference The PFN paradigm has recently been applied to causal inference tasks. Some works proposed amortized solutions for causal effect estimation by training PFN models on synthetic priors, achieving substantial gains in speed and comparable accuracy when compared to classical methods (Robertson et al., 2025; Meresht et al., 2025; Dhir et al., 2025a).

Concurrently to our work[1], PFN models have also been explored for causal discovery. Swelam et al. (2025) showed that graph structure can be partially recovered from intermediate activations of a PFN model, yielding an amortized causal discovery approach similar in spirit to AVICI (Lorch et al., 2022). An alternative approach employs TabPFN as a likelihood model and optimizes a continuous graph parameterization using policy gradients (Sypniewski et al., 2025).

1. These works have just been presented during Eurips Workshops on Dec 6-7, 2025.

In contrast to these approaches, our method leverages TabPFN within nonparametric GES, retaining a discrete, score-consistent search procedure with strong theoretical guarantees for the optimization process.

5. Experiments

We conduct a thorough evaluation of the proposed approach and compare it against state-of-the-art nonparametric causal discovery methods. In Section 5.1 we compare the method on synthetic and simulated data. In Section 5.2, we discuss results on a real-world dataset. Further, we compare the computational efficiency of the methods in Section 5.3. Finally, in Section 5.4 we provide ablation studies which validate our design choices described in Section 3.

Baselines We compare the proposed approach with state-of-the-art causal discovery methods: PC (Spirtes et al., 2000), GES (Meek, 1997), DCDI (Brouillard et al., 2020), SDCD (Nazaret et al., 2024), and AVICI (Lorch et al., 2022). We use a PC with the KCI test to handle a nonparametric setting. For GES, we used the implementation provided by Gamella (2024), facilitating an additional turning phase introduced in (Hauser and Bühlmann, 2012) that improves the performance in the finite sample regime. We denote this baseline as GES(BIC) to distinguish it from our approach, which we denote PF-GES. DCDI and SDCD are continuous neural methods that leverage expressive neural architectures and continuous optimization for causal discovery. We used implementations of these methods that were provided by the authors.

Metrics. We report performance using Structural Hamming Distance (SHD), Structural Intervention Distance (SID) and F1 metrics. Since causal discovery in our benchmark is not identifiable beyond the Markov equivalence class, we evaluate distances between the completed partially directed acyclic graphs (CPDAGs) of the estimated and ground-truth structures. In the main text we focus on reporting the SHD results, full set of metrics can be found in Appendix B.

5.1. Synthetic and simulated data

Datasets. We consider synthetic settings where graphs are sampled from the Erdős–Rényi(ER) or Scale-Free(SF) class, functional relations are modeled using randomly initialized neural networks, and additive Gaussian noise with different variances is assumed. The data is normalized during generation process to avoid Var-sortability Reisach et al. (2021). This setup follows standard practice in the literature (Brouillard et al., 2020; Lorch et al., 2021; Nazaret et al., 2024; Annadani et al., 2023). The details of the data generation procedure are provided in Appendix C. We consider graphs of varying sizes and densities, indicated in parentheses in the graph name by first listing the number of nodes and then the number of edges (e.g., ER(10, 20) denotes an Erdős–Rényi graph with 10 nodes and 20 edges).

Second, we evaluate on datasets generated by the SERGIO simulator, which models realistic biological relations (Dibaeinia and Sinha, 2020), with underlying graph structures sampled from the Scale-Free class and functional relationships generated using expert-designed differentiable equations. For each graph size, density, and type, we evaluate the methods on 10 datasets of size 1000 samples. We report the mean and 95% confidence intervals for the evaluation metric.

Experimental setup We report hyperparameters used by each method in the Appendix A. Depending on the quality of default values, we conducted additional searches in hyperparameter spaces. All

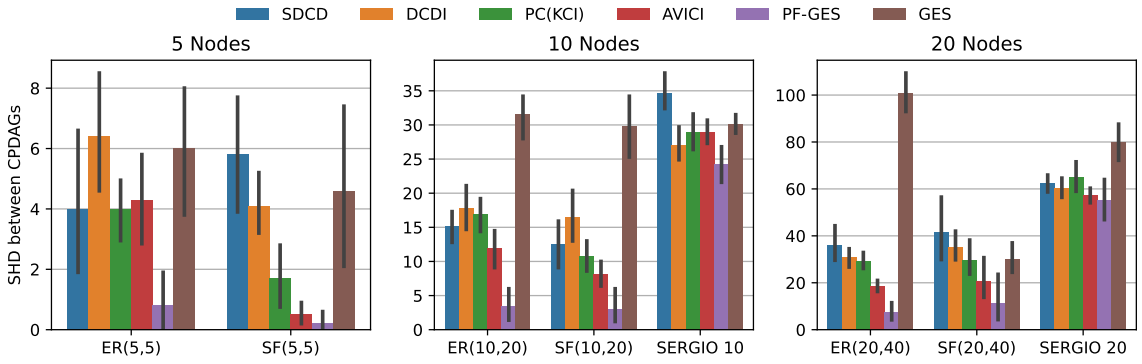

Figure 1: Benchmark results on synthetic and simulated (SERGIO) datasets. Error bars show 95% bootstrap confidence intervals.

methods have one hyperparameter that regulates the density of the solution. For fair comparison, when all other parameters were selected, we conducted an additional grid search over the sparsity parameter. We report the results for the value of the sparsity parameter that achieved the best average performance across all considered datasets of a certain type, size, and density.

Results. Figure 1 summarizes the benchmark results. The corresponding numerical values are reported in Table 6 in the Appendix. Our method, PF-GES, substantially outperforms all baselines on synthetic datasets, achieving the lowest SHD across all eight dataset variants. Remarkably, PF-GES almost perfectly recovers graphs with up to 10 nodes, attaining an SHD of 1.1 for ER(10,20) and 3.3 for SF(10,20) using only 1,000 samples. To the best of our knowledge, such performance has not been reported previously and highlights the strong potential of PFN-based models for causal discovery.

On the SERGIO simulated data, the performance gap between our approach and the baselines is smaller. We hypothesize that this is due to the PFN training prior being well aligned with synthetic data, whereas SERGIO exhibits systematically different characteristics. Notably, PF-GES does not fail to estimate the data likelihood in the simulated setting; rather, it is only less efficient than in the purely synthetic case. In terms of SID and F1 metrics PF-GES also outperforms other method often by a large margin, with the exception for SERGIO data with 20 nodes, where GES method is slightly better (see Appendix B).

Additionally, we include experiments with the number of samples varying from 100 to 1000 on graphs with 10 nodes, where we show that PF-GES performs better than or on par with all compared methods in terms of SHD. The results are presented and discussed in the Appendix B.2.

Results on additional mechanisms We evaluate the methods on two additional data-generating mechanism, linear functions with uniform noise and neural networks with non-additive noise (see Appendix C for details), using ER(20,40) graphs with 1000 samples. Results are presented in Figure 2.

On linear data with uniform noise, PF-GES achieves the best performance across all three metrics. Compared to AVICI, the second-best nonparametric method, PF-GES offers a modest improvement in SHD but substantially outperforms it on SID and F1. Interestingly, GES (BIC) achieves SID values comparable to PF-GES on this dataset, however, it exhibits considerably worse SHD and F1 scores.On data generated by neural networks with non-additive noise, PF-GES again achieves the best results across all metrics. AVICI is the second-best method, followed by DCDI and SDCD,

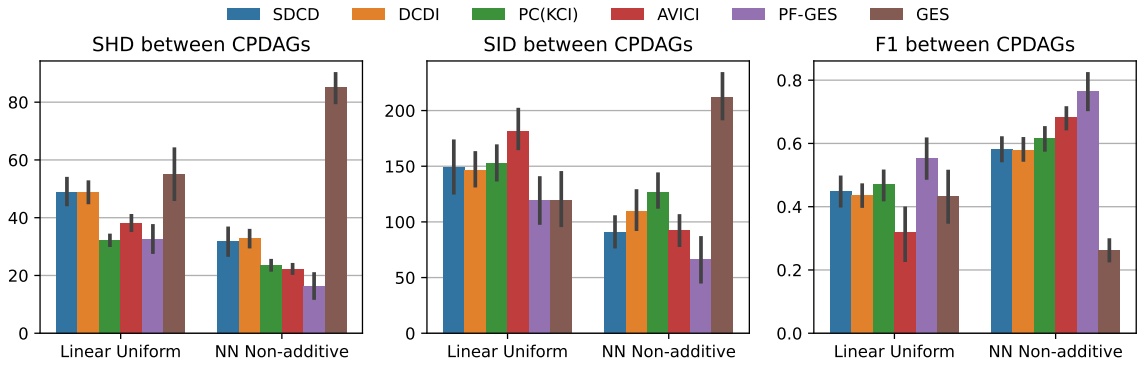

Figure 2: Comparison of causal discovery methods on ER(20,40) graphs with 1000 samples using two additional data-generating mechanisms: linear functions with uniform noise and neural networks with non-additive noise. Error bars represent 95% bootstrap confidence intervals.

DCDI	SDCD	PC (KCI)	PF-GES	GES (BIC)
52	98	41	**28**	34

Table 1: Benchmark results on the Causal Chambers dataset.

which perform comparably to each other. Notably, unlike the linear case, the gap between PF-GES and AVICI remains consistent across all three metrics, with no significant variation in the relative improvement on SHD, SID, or F1.

5.2. Real-world data

We evaluate the methods on Causal Chambers dataset (Gamella et al., 2025) a real-world physical system where a ground-truth DAG is known and consists of 20 nodes. To make the comparison fair, in this setting, we again conduct a grid search over the sparsity parameter, but this time select the best solution based on the likelihood of the held-out sample.

We compare the SHD of the obtained solutions in the Table 1. Our approach improves the structural accuracy by a large margin. This demonstrated that even with synthetic priors on the PFN training data, we can obtain substantial improvements in a real-world setting.

5.3. Computational cost comparison

To provide additional insight into the computational efficiency of our approach, we compare execution times across methods on standardized hardware consisting of 16 CPU cores and a single NVIDIA A100-SXM4-40GB GPU. Execution times are measured as wall-clock runtime. Note that the GES(BIC) and PC baselines does not utilize GPU acceleration.

Figure 3 compares the benchmarked methods in terms of runtime (x-axis) and accuracy (y-axis), illustrating the trade-off between computational cost and solution quality. In some settings, such as the SF(10,20) graphs (triangular markers), our method achieves a favorable Pareto position, being both faster and substantially more accurate than competing approaches.

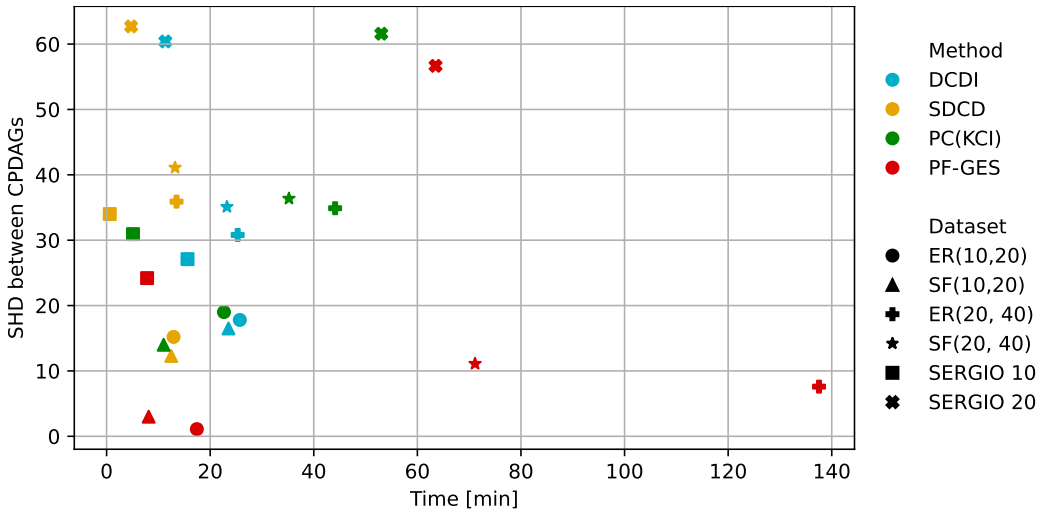

Figure 3: Comparison of causal discovery accuracy and running time. The closer to the bottom left corner, the better. The various shapes describe different datasets used in the evaluation. The graph is based on synthetic and simulated data from graphs of sizes 10 and 20.

Overall, our approach remains competitive in runtime with continuous-optimization baselines on smaller graphs (10 nodes), but its execution time increases noticeably with graph size. Moreover, runtime is highly sensitive to the underlying graph structure: Erdős–Rényi and scale-free graphs with the same number of nodes and expected density can exhibit substantially different execution times. This additional computational cost is, however, consistently compensated by markedly improved accuracy.

Runtime differences between graph types. We observed that for PF-GES, causal discovery on Erdős–Rényi graphs consistently takes more time than on Scale-Free graphs of the same size and expected density. We hypothesize that this is due to structural characteristics of ER graphs: during the forward phase, the search tends to reach more saturated intermediate graphs compared to SF graphs, resulting in larger neighborhoods to evaluate at each step. While GES is known to have exponential worst-case complexity, from our observation this manifests more strongly in ER graphs than in SF graphs. This behavior could potentially be mitigated by incorporating algorithmic improvements to GES, such as those proposed in Selective Greedy Equivalence Search (Maxwell Chickering and Meek, 2015), FGES (Ramsey et al., 2017) or XGES (Nazaret and Blei, 2025), which we leave for future work.

5.4. Ablations on the likelihood model

Likelihood model comparison TabPFN has demonstrated strong performance on regression and classification tasks, often matching or outperforming established methods such as gradient-boosted trees across diverse tabular datasets (Hollmann et al., 2025). However, the original work devoted relatively little attention to evaluating the quality of the posterior predictive distribution, the key quantity required for our application. Since accurate posterior estimation is critical for the reliability of the posterior ratio test, we conducted ablation studies comparing TabPFN against Bayesian Neural Networks, a natural alternative for nonparametric Bayesian inference.

SCM type	GES (BNN)	PF-GES
ER(5,5)	6.5 (5.48, 7.4)	**0.8** (0.0, 2.4)
SF(5,5)	7.4(6.4, 9.0)	**0.2** (0.0, 1.0)
ER(10,20)	25.9 24.1, 28.3	**1.1** (0.4, 2.2)
SF(10,20)	21.4 (18.8, 24.9)	**3.0** (1.4, 8.6)
SERGIO 10	**29.1** (26.3, 31.3)	**24.2** (21.6, 26.8)

Table 2: Ablation results on synthetic and simulated (SERGIO) datasets comparing PF-GES with GES with BNNs. Values in brackets describe 95% bootstrap confidence intervals.

SCM type	PF-GES	PF-GES (RANDOM)
ER(5,5)	**0.85** (0.3, 1.8)	**1.2** (0.6, 2.4)
SF(5,5)	**0.4** (0.0, 1.6)	**1.27** (0.67, 2.3)
ER(10,20)	**3.45** (1.9, 5.5)	8.4 (6.2, 10.9)
SF(10,20)	**5.25** (3.6, 7.3)	**8.3** (5.5, 11.0)
SERGIO 10	**24.95** (22.8, 27.4)	**26.75** (25.0, 28.9)
ER(20,40)	**5.6** (3.6, 8.9)	-
SF(20,40)	**11.05** (8.1, 14.8)	30.2 (25.6, 37.1)
SERGIO 20	**53.75** (49.6, 58.9)	**60.8** (55.6, 65.2)

Table 3: Ablation results on synthetic and simulated(SERGIO) datasets comparing the described graph selecting methods. Values in brackets describe 95% bootstrap confidence intervals.

For the BNN baseline, we trained a separate network for each parent set configuration evaluated during the search. We first performed a grid search over architectural and training hyperparameters (network depth, width, learning rate, and prior variance) on a held-out validation set to identify the best configuration. We then used these optimized hyperparameters within our GES method, conducting the same sparsity parameter grid search as described in the benchmark section.

Table 2 presents the results of this comparison. TabPFN substantially outperforms the BNN baseline across all graph types and sizes, achieving a dramatically lower SHD metric. For instance, on ER(10,20) graphs, TabPFN achieves an SHD of 1.1 compared to 21.4 for BNNs. These results demonstrate that TabPFN's pretrained prior provides significantly more reliable posterior estimates than BNNs trained from scratch on limited data, strengthening our choice of likelihood model.

Graph selection strategy We also evaluated the practical benefit of our greedy graph selection strategy described in Sec. 3.1 against the baseline approach of accepting any graph that passes the statistical test in enumeration order.

Table 3 shows that the greedy selection strategy yields substantially better results across most settings. On ER(10,20) graphs, greedy selection achieves an SHD of 1.1 compared to 9.4 for the enumeration baseline. Similarly, substantial improvements are observed on SF(10,20) graphs and bigger graphs. The only exception is SERGIO with 10 nodes, where both methods perform comparably (24.2 vs 24.0 SHD), with overlapping confidence intervals indicating no significant difference.

Furthermore, the naive enumeration strategy is severely limited in scalability. On ER(20,40) graphs, the enumeration approach exceeded our 24-hour time limit during the forward phase and could not complete, while the greedy strategy finished successfully.

6. Conclusions and discussion

In this work, we introduced new causal discovery approach that incorporates PFN paradigm into greedy equivalence search framework. The proposed approach outperforms state-of-the-art nonparametric causal discovery methods, often by a large margin, across synthetic, simulated, and real-world

datasets. These findings point toward the PFN paradigm as a natural and promising direction for advancing causal discovery in complex real-world applications.

On data-prior selection. TabPFN is trained entirely on synthetic data drawn from a prior defined by randomly initialized neural networks. Given the strong performance of our method on standard synthetic benchmarks, we believe that the training prior of TabPFN is well aligned with the assumptions underlying these benchmarks. Importantly, this prior is sufficiently general to improve DAG recovery even in settings involving more complex functional relationships, such as SERGIO and Causal Chambers. Recently, Olko et al. (2025) advanced the thesis that limitations in likelihood estimation constitute a primary bottleneck for modern causal discovery methods. Since our approach does not substantially modify the structure search procedure, we attribute its strong performance primarily to the quality of likelihood estimation provided by TabPFN, consistent with the findings of Sypniewski et al. (2025). These results suggest that designing application-specific priors can substantially improve causal discovery, making this a promising direction for future work. Preliminary evidence indicates that adapting PFNs to specific domains may be straightforward (Bühler et al., 2025), and that even imperfectly aligned priors can yield substantial gains (Ma et al., 2025). Finally, our approach relies on the assumption that the PFN prior is well aligned with the data-generating process. However, existing evaluation protocols do not provide diagnostic tools for assessing prior–data alignment or determining whether a given PFN is appropriate for a specific causal task. Developing principled evaluations of prior–data compatibility would be valuable for validating PFN-based causal discovery in real-world applications.

On computational efficiency. While our method demonstrates superior structural accuracy, the basic GES implementation employed here can become computationally demanding on larger graphs. Fortunately, a rich line of research on algorithmic improvements to GES could potentially be integrated into the nonparametric framework, including Fast GES (Ramsey et al., 2017), which scales to graphs with a million variables, eXtremely Greedy Equivalence Search (Nazaret and Blei, 2025), and the approach of Andrews et al. (2023) using best order score search with grow-shrink trees. Investigating whether these innovations can be combined with nonparametric GES to achieve both accuracy and scalability is an exciting direction for future work, particularly for domains such as genomics where nonparametric flexibility and scaling to thousands of variables are both essential.

Acknowledgments

Research of MO was partly supported by the Preludium project, funded by the National Science Centre, Poland, under grant no. 2025/57/N/ST6/04974. Mateusz Gajewski would like to acknowledge the National Science Centre, Poland, for the financial support in the framework of the project 2025/57/N/ST6/03567.

We gratefully acknowledge Polish high-performance computing infrastructure PLGrid (HPC Center: ACK Cyfronet AGH) for providing computer facilities and support within computational grant no. PLG/2024/017740. We thank Dominik Bogucki, Alicja Ziarko, and our Reviewers for thoughtfull feedback on the manuscript.

References

Bryan Andrews, Joseph Ramsey, Ruben Sanchez Romero, Jazmin Camchong, and Erich Kummerfeld. Fast scalable and accurate discovery of dags using the best order score search and grow shrink trees. *Advances in neural information processing systems*, 36:63945–63956, 2023.

Yashas Annadani, Nick Pawlowski, Joel Jennings, Stefan Bauer, Cheng Zhang, and Wenbo Gong. Bayesdag: Gradient-based posterior inference for causal discovery. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/05cf28e3d3c9a179d789c55270fe6f72-Abstract-Conference.html.

Bryon Aragam. Greedy equivalence search for nonparametric graphical models. *CoRR*, abs/2406.17228, 2024. doi: 10.48550/ARXIV.2406.17228. URL https://doi.org/10.48550/arXiv.2406.17228.

Kevin Bello, Bryon Aragam, and Pradeep Ravikumar. Dagma: Learning dags via m-matrices and a log-determinant acyclicity characterization. *Advances in Neural Information Processing Systems*, 35:8226–8239, 2022.

Eli Bingham, Jonathan P Chen, Martin Jankowiak, Fritz Obermeyer, Neeraj Pradhan, Theofanis Karaletsos, Rohit Singh, Paul Szerlip, Paul Horsfall, and Noah D Goodman. Pyro: Deep universal probabilistic programming. *Journal of machine learning research*, 20(28):1–6, 2019.

Philippe Brouillard, Sébastien Lachapelle, Alexandre Lacoste, Simon Lacoste-Julien, and Alexandre Drouin. Differentiable causal discovery from interventional data. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/f8b7aa3a0d349d9562b424160ad18612-Abstract.html.

Magnus Bühler, Lennart Purucker, and Frank Hutter. Causal data augmentation for robust fine-tuning of tabular foundation models. In *EurIPS 2025 Workshop: AI for Tabular Data*, 2025. URL https://openreview.net/forum?id=EfQv02muYG.

David Maxwell Chickering. Optimal structure identification with greedy search. *J. Mach. Learn. Res.*, 3:507–554, 2002. URL https://jmlr.org/papers/v3/chickering02b.html.

Arnaud Delaunoy, Florent Leclercq, and Bruno Regaldo-Saint Blancard. Bnnpriors: A library for bayesian neural network inference with different prior distributions. *Software Impacts*, 9:100089, 2021. doi: 10.1016/j.simpa.2021.100089.

Anish Dhir, Matthew Ashman, James Requeima, and Mark van der Wilk. A meta-learning approach to bayesian causal discovery. In *The Thirteenth International Conference on Learning Representations*.

Anish Dhir, Samuel Power, and Mark van der Wilk. Bivariate causal discovery using bayesian model selection. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=twm7qPVX1F.

Anish Dhir, Cristiana Diaconu, Valentinian Mihai Lungu, James Requeima, Richard E. Turner, and Mark van der Wilk. Estimating interventional distributions with uncertain causal graphs through meta-learning. *CoRR*, abs/2507.05526, 2025a. doi: 10.48550/ARXIV.2507.05526. URL https://doi.org/10.48550/arXiv.2507.05526.

Anish Dhir, Ruby Sedgwick, Avinash Kori, Ben Glocker, and Mark van der Wilk. Continuous bayesian model selection for multivariate causal discovery. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025b. URL https://proceedings.mlr.press/v267/dhir25a.html.

Payam Dibaeinia and Saurabh Sinha. Sergio: a single-cell expression simulator guided by gene regulatory networks. *Cell systems*, 11(3):252–271, 2020.

David Duvenaud et al. Bayesian neural networks. https://www.cs.toronto.edu/~duvenaud/distill_bayes_net/public/, 2020. Accessed: 2024.

Vincent Fortuin. Priors in bayesian deep learning: A review. *International Statistical Review*, 90(3): 563–591, 2022. doi: 10.1111/insr.12502.

Vincent Fortuin, Adrià Garriga-Alonso, Sebastian W Ober, Florian Wenzel, Gunnar Rätsch, Richard E Turner, Mark van der Wilk, and Laurence Aitchison. Bayesian neural network priors revisited. *arXiv preprint arXiv:2102.06571*, 2022.

Juan L. Gamella. GES: Python implementation of the GES algorithm for causal discovery. https://github.com/juangamella/ges, 2024. Python implementation of the Greedy Equivalence Search algorithm from Chickering (2002). BSD-3-Clause license.

Juan L Gamella, Jonas Peters, and Peter Bühlmann. Causal chambers as a real-world physical testbed for ai methodology. *Nature Machine Intelligence*, pages 1–12, 2025.

Marta Garnelo, Dan Rosenbaum, Christopher Maddison, Tiago Ramalho, David Saxton, Murray Shanahan, Yee Whye Teh, Danilo Jimenez Rezende, and S. M. Ali Eslami. Conditional neural processes. In Jennifer G. Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, Proceedings of Machine Learning Research, pages 1690–1699. PMLR, 2018. URL http://proceedings.mlr.press/v80/garnelo18a.html.

Dominique MA Haughton. On the choice of a model to fit data from an exponential family. *The annals of statistics*, pages 342–355, 1988.

Alain Hauser and Peter Bühlmann. Characterization and greedy learning of interventional markov equivalence classes of directed acyclic graphs. *The Journal of Machine Learning Research*, 13(1): 2409–2464, 2012.

Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. Tabpfn: A transformer that solves small tabular classification problems in a second. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=cp5PvcI6w8_.

Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nat.*, 637(8044):319–326, 2025. doi: 10.1038/S41586-024-08328-6. URL https://doi.org/10.1038/s41586-024-08328-6.

Jacob Jacobs et al. Lowering the computational barrier: Partially bayesian neural networks for transparency in medical imaging ai. *Frontiers in Computer Science*, 5:1071174, 2023. doi: 10.3389/fcomp.2023.1071174.

Xiaotao Jia, Bichen Song, Jianlei Yang, Yujin Zhu, Rui Li, and Yujiu Shen. Efficient computation reduction in bayesian neural networks through feature decomposition and memorization. *arXiv preprint arXiv:2005.03857*, 2020.

Nan Rosemary Ke, Silvia Chiappa, Jane X. Wang, Jörg Bornschein, Anirudh Goyal, Mélanie Rey, Theophane Weber, Matthew M. Botvinick, Michael Curtis Mozer, and Danilo Jimenez Rezende. Learning to induce causal structure. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=hp_RwhKDJ5.

Christopher Kolberg, Katharina Eggensperger, and Nico Pfeifer. Tabpfn-wide: Continued pre-training for extreme feature counts. *arXiv preprint arXiv:2510.06162*, 2025.

Thuc Duy Le, Tao Hoang, Jiuyong Li, Lin Liu, Huawen Liu, and Shu Hu. A fast pc algorithm for high dimensional causal discovery with multi-core pcs. *IEEE/ACM transactions on computational biology and bioinformatics*, 16(5):1483–1495, 2016.

Hao-Chih Lee, Matteo Danieletto, Riccardo Miotto, Sarah T. Cherng, and Joel T. Dudley. Scaling structural learning with NO-BEARS to infer causal transcriptome networks. In *Pacific Symposium on Biocomputing 2020, Fairmont Orchid, Hawaii, USA, January 3-7, 2020*, pages 391–402, 2020. URL https://psb.stanford.edu/psb-online/proceedings/psb20/Lee.pdf.

Lars Lorch, Jonas Rothfuss, Bernhard Schölkopf, and Andreas Krause. Dibs: Differentiable bayesian structure learning. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 24111–24123, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/ca6ab34959489659f8c3776aaf1f8efd-Abstract.html.

Lars Lorch, Scott Sussex, Jonas Rothfuss, Andreas Krause, and Bernhard Schölkopf. Amortized inference for causal structure learning. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/54f7125dee9b8b3dc798bb9a082b09e2-Abstract-Conference.html.

Junwei Ma, Nour Shaheen, Alex Labach, Amine Mhedhbi, Frank Hutter, Anthony L. Caterini, and Valentin Thomas. Generalization can emerge in tabular foundation models from a single table. In *EurIPS 2025 Workshop: AI for Tabular Data*, 2025. URL https://openreview.net/forum?id=jdM9Pa8she.

David Maxwell Chickering and Christopher Meek. Selective greedy equivalence search: Finding optimal bayesian networks using a polynomial number of score evaluations. *arXiv e-prints*, pages arXiv–1506, 2015.

Christopher Meek. Graphical Models: Selecting causal and statistical models. 1 1997. doi: 10.1184/R1/22696393.v1. URL https://kilthub.cmu.edu/articles/thesis/Graphical_Models_Selecting_causal_and_statistical_models/22696393.

Vahid Balazadeh Meresht, Hamidreza Kamkari, Valentin Thomas, Benson Li, Junwei Ma, Jesse C. Cresswell, and Rahul G. Krishnan. Causalpfn: Amortized causal effect estimation via in-context learning. *CoRR*, abs/2506.07918, 2025. doi: 10.48550/ARXIV.2506.07918. URL https://doi.org/10.48550/arXiv.2506.07918.

Samuel Müller, Noah Hollmann, Sebastian Pineda-Arango, Josif Grabocka, and Frank Hutter. Transformers can do bayesian inference. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL https://openreview.net/forum?id=KSugKcbNf9.

Samuel Müller, Arik Reuter, Noah Hollmann, David Rügamer, and Frank Hutter. Position: The future of bayesian prediction is prior-fitted, 2025. URL https://arxiv.org/abs/2505.23947.

Eric Thomas Nalisnick. *On Priors for Bayesian Neural Networks*. PhD thesis, UC Irvine, 2018. URL https://escholarship.org/uc/item/1jq6z904.

Achille Nazaret and David Blei. Extremely greedy equivalence search. *arXiv preprint arXiv:2502.19551*, 2025.

Achille Nazaret, Justin Hong, Elham Azizi, and David M. Blei. Stable differentiable causal discovery. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=JJZBZW28Gn.

Mateusz Olko, Mateusz Gajewski, Joanna Wojciechowska, Mikolaj Morzy, Piotr Sankowski, and Piotr Milos. Since faithfulness fails: The performance limits of neural causal discovery. *CoRR*,

abs/2502.16056, 2025. doi: 10.48550/ARXIV.2502.16056. URL https://doi.org/10.48550/arXiv.2502.16056.

Prior Labs. Tabpfn-2.5 model report. Technical report, 2025.

Joseph Ramsey, Madelyn Glymour, Ruben Sanchez-Romero, and Clark Glymour. A million variables and more: the fast greedy equivalence search algorithm for learning high-dimensional graphical causal models, with an application to functional magnetic resonance images. *International journal of data science and analytics*, 3(2):121–129, 2017.

Alexander G. Reisach, Christof Seiler, and Sebastian Weichwald. Beware of the simulated dag! causal discovery benchmarks may be easy to game. In *Neural Information Processing Systems*, 2021. URL https://api.semanticscholar.org/CorpusID:239998404.

Alexander G. Reisach, Myriam Tami, Christof Seiler, Antoine Chambaz, and Sebastian Weichwald. A scale-invariant sorting criterion to find a causal order in additive noise models. In *Neural Information Processing Systems*, 2023. URL https://api.semanticscholar.org/CorpusID:257901170.

Arik Reuter, Tim G. J. Rudner, Vincent Fortuin, and David Rügamer. Can transformers learn full bayesian inference in context? *CoRR*, abs/2501.16825, 2025. doi: 10.48550/ARXIV.2501.16825. URL https://doi.org/10.48550/arXiv.2501.16825.

Jake Robertson, Arik Reuter, Siyuan Guo, Noah Hollmann, Frank Hutter, and Bernhard Schölkopf. Do-pfn: In-context learning for causal effect estimation, 2025.

Gideon Schwarz. Estimating the dimension of a model. *The annals of statistics*, pages 461–464, 1978.

Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, Prediction, and Search, Second Edition*. Adaptive computation and machine learning. MIT Press, 2000. ISBN 978-0-262-19440-2.

Omar Swelam, Lennart Purucker, Jake Robertson, Hanne Raum, Joschka Boedecker, and Frank Hutter. Does tabpfn understand causal structures?, 2025. URL https://arxiv.org/abs/2511.07236.

Mateusz Sypniewski, Mateusz Olko, Mateusz Gajewski, and Piotr Miłoś. Amortized causal discovery with prior-fitted networks, 2025. URL https://arxiv.org/abs/2512.11840.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

Yue Yu, Tian Gao, Naiyu Yin, and Qiang Ji. Dags with no curl: An efficient dag structure learning approach. In *International Conference on Machine Learning*, pages 12156–12166. Pmlr, 2021.

Kun Zhang, Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. Kernel-based conditional independence test and application in causal discovery. *arXiv preprint arXiv:1202.3775*, 2012.

Xun Zheng, Bryon Aragam, Pradeep Ravikumar, and Eric P. Xing. Dags with NO TEARS: continuous optimization for structure learning. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 9492–9503, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/e347c51419ffb23ca3fd5050202f9c3d-Abstract.html.

Appendix A. Details of parameter tuning.

PF-GES (our method) We performed a grid search over two key hyperparameters: the prior regularization coefficient $\pi(G)$ (which corresponds to the graph prior in Equation 8) and the threshold λ used in the statistical test (Equation 9). For the threshold, we considered differences in logarithms of the posterior ratio.

The grid search ranges were:

- Prior regularization: $\{0.00001, 0.0001, 0.001, 0.01\}$
- Threshold λ: $\{-0.1, -0.01, -0.001, -0.0001, 0.0001, 0.001\}$

To balance computational efficiency with generalizability, we tuned hyperparameters on smaller graphs and applied the selected values to all graph sizes within each data type. Specifically, parameters were selected based on performance on 5-node graphs (ER(5,5) and SF(5,5)) and 10-node SERGIO graphs, then used for all larger instances of the respective graph class.

The final selected parameters were:

- Erdős-Rényi (ER): prior regularization = 0.00001, $\lambda = -0.01$
- Scale-Free (SF): prior regularization = 0.001, $\lambda = -0.01$
- SERGIO: prior regularization = 0.0001, $\lambda = -0.01$

These parameters were used consistently across all experiments reported in Section 5.

GES(BNN) We implemented Bayesian Neural Networks using the Pyro probabilistic programming library (Bingham et al., 2019). The hyperparameter tuning was conducted in two stages.

First, we performed a grid search over neural network architectures and training hyperparameters. To reduce computational cost, this search was conducted on 10-node graphs, allowing us to train more networks for comparison. For each dataset, we selected the architecture that achieved the lowest negative log-likelihood on held-out validation data. The grid included:

- Layer widths: $\{[4, 4], [8, 8], [16, 16]\}$
- Learning rate: $\{0.003, 0.01\}$
- Prior scale: $\{0.5, 1.0, 2.0\}$

Second, using the selected neural network configurations, we performed a grid search over the GES method hyperparameters (prior regularization and threshold λ) as described for PF-GES above.

The final selected parameters were:

- Erdős-Rényi (ER): layers $[16, 16]$, learning rate $= 0.003$, prior scale $= 2.0$, prior regularization $= 0.001$, $\lambda = -0.001$
- Scale-Free (SF): layers $[16, 16]$, learning rate $= 0.003$, prior scale $= 2.0$, prior regularization $= 0.001$, $\lambda = -0.01$
- SERGIO: layers $[4, 4]$, learning rate $= 0.01$, prior scale $= 0.5$, prior regularization $= 0.001$, $\lambda = -0.01$

AVICI For SERGIO dataset, the model pretrained for this dataset was used "neurips-grn" model. For all other datasets we used "scm-v0" model trained for the most broad settings.

DCDI We found the default parameters of DCDI to be stable and performing well across various settings. For the grid search over sparsity coefficient, we considered the following values: 5.5, 3.0, 1.7, 1.0, 0.3, 0.1.

SDCD We observed that the default hyperparameters provided with the original SDCD implementation did not reproduce the performance reported in the paper. To obtain competitive results, we performed an extensive random search over the optimization hyperparameters. The search ranges are summarized in Table 4. Hyperparameter tuning was conducted on the ER(10, 20) synthetic graph class. Final values were selected manually based on a trade-off between accuracy, measured by the lowest structural Hamming distance (SHD), and stability, defined as robustness of SHD to small perturbations of the hyperparameters. The selected configuration is reported in Table 5 and was used across all benchmark settings, including SERGIO.

SDCD includes two sparsity-controlling hyperparameters: one applied during the preliminary optimization stage without the acyclicity constraint, and another used in the main optimization stage. To simplify usage and reduce the dimensionality of the search space, we coupled these parameters using the relation $\alpha_1 = \alpha_2/3$. Consequently, sparsity tuning was performed only over α_2, with the following values considered: 1e-3, 3e-4, 1e-4, 3e-5, and 1e-5.

learning rate$_1$	[6e-5, 6e-3]
learning rate$_2$	[3e-5, 3e-3]
#epochs	[2000, 32000]
α_1	[1e-7, 3e-2]
α_2	[3e-6, 3e-3]
β_1	[0, 3e-3]
β_2	[0, 3e-3]
γ	[0.0001, 0.5]

Table 4: Random grid search ranges for SDCD method.

GES and PC There was no grid search nor sparsity tuning conducted for those algorithms.

learning rate$_1$	6e-3
learning rate$_2$	3e-3
#epochs	32 000
β_1	1e-4
β_2	3e-4
γ	0.001

Table 5: Parameters selected for evaluation.

Appendix B. Additional experimental results

In this section we include additional evaluations. First, we provide comparison of methods under full set of metrics: SHD, SID, F1. Than, we provide evaluations under varying sample size. Finally, we provide comparison with DiBS method (Lorch et al., 2021).

B.1. Additional metrics for main result

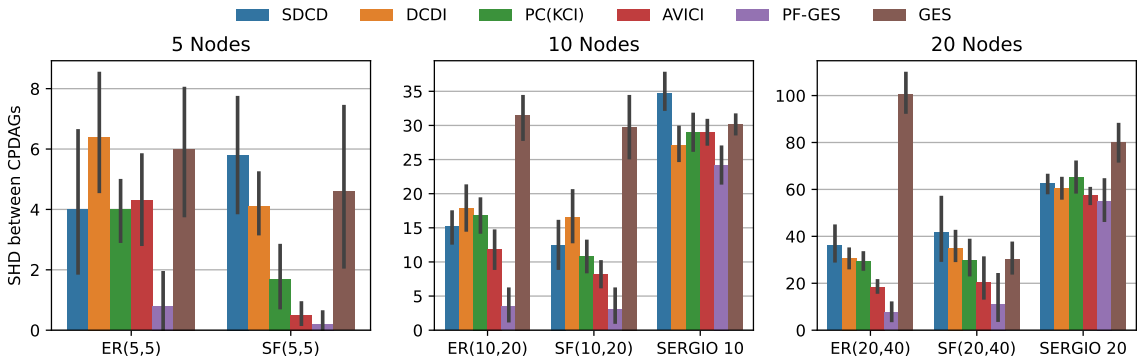

Figure 4: Comparison of causal discovery methods on synthetic (ER and SF) and simulated (SERGIO) datasets with varying graph sizes (5, 10, and 20 nodes) for SHD metric. Error bars represent 95% bootstrap confidence intervals calculated across 10 seeds.

We present the results of the main benchmark evaluated under three complementary metrics—SHD (Figure 4), SID (Figure 5), and F1 (Figure 6).

Considering first the SHD metric, PF-GES achieves the best or comparable-to-best results across all dataset configurations. The only setting where another method reaches similar performance is SERGIO with 20 nodes, where AVICI attains a comparable SHD, however the difference on F1 metric is significant and favourable for PF-GES method. We note, however, that AVICI uses a pretrained model specifically designed for the SERGIO simulator, whereas PF-GES relies on the standard, off-the-shelf TabPFN without any domain-specific adaptation. Importantly, even on this dataset, PF-GES outperforms AVICI on both the SID and F1 metrics.

Across all three metrics, the results consistently confirm the findings from the main text. PF-GES dominates on synthetic datasets (ER and SF graphs), where the advantage is particularly pronounced for 10- and 20-node graphs. On SERGIO data the margins are narrower, consistent with our hypothesis about prior alignment, yet PF-GES remains competitive or superior under every

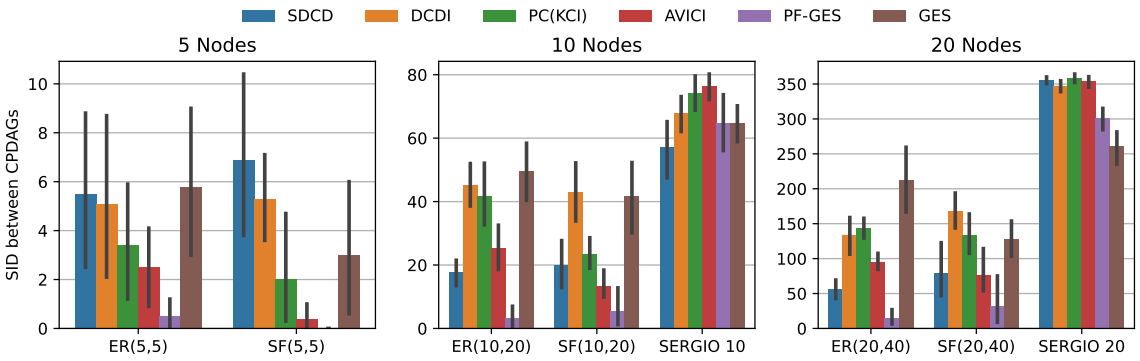

Figure 5: Comparison of causal discovery methods on synthetic (ER and SF) and simulated (SERGIO) datasets with varying graph sizes (5, 10, and 20 nodes) for SID metric. Error bars represent 95% bootstrap confidence intervals calculated across 10 seeds.

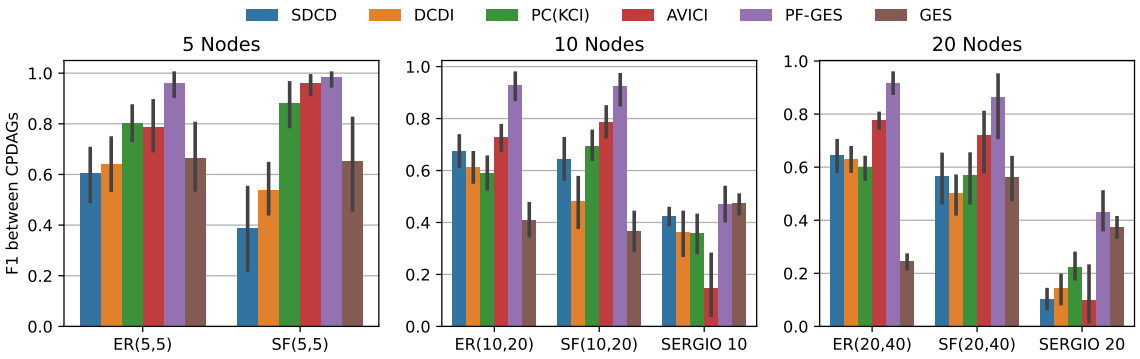

Figure 6: Comparison of causal discovery methods on synthetic (ER and SF) and simulated (SERGIO) datasets with varying graph sizes (5, 10, and 20 nodes) for F1 score metric. Error bars represent 95% bootstrap confidence intervals calculated across 10 seeds.

metric. The SID results further highlight the practical relevance of our method: lower structural intervention distances indicate that the recovered graphs better preserve the causal ordering, which is critical for applications involving interventional reasoning. The F1 scores reinforce these conclusions, showing that PF-GES achieves high precision and recall simultaneously, whereas several baselines trade off one for the other.

B.2. Evaluation under varying sample size

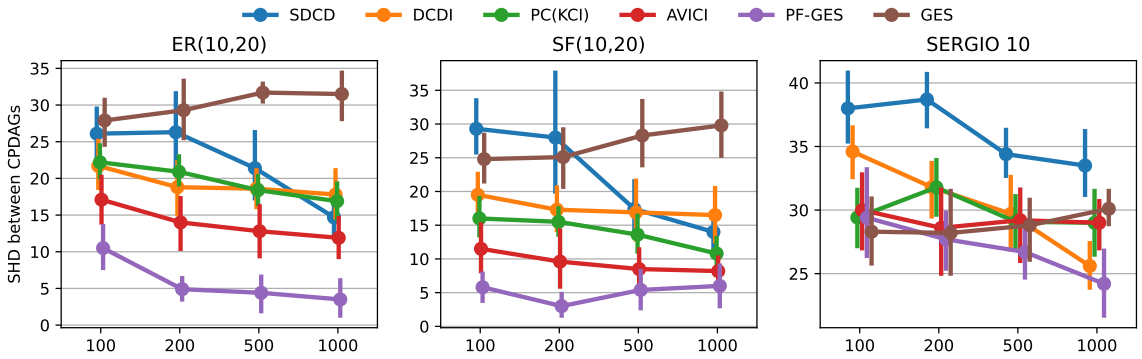

Figure 7: Comparison of causal discovery methods on 10-node graphs (ER, SF, and SERGIO) across varying sample sizes (100, 200, 500, 1000) for SHD metric. Error bars represent 95% bootstrap confidence intervals calculated across 10 seeds.

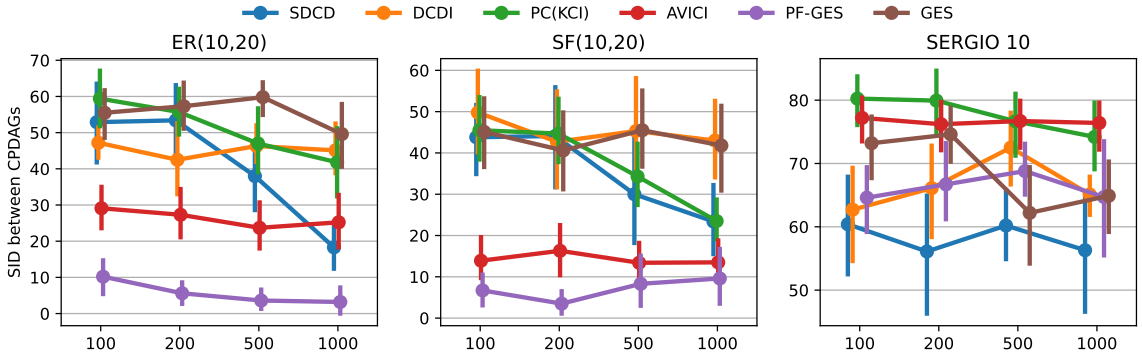

Figure 8: Comparison of causal discovery methods on 10-node graphs (ER, SF, and SERGIO) across varying sample sizes (100, 200, 500, 1000) for SID metric. Error bars represent 95% bootstrap confidence intervals calculated across 10 seeds.

We evaluate all methods on 10-node graphs across sample sizes of 100, 200, 500, and 1000, reporting SHD (Figure 7), SID (Figure 8), and F1 (Figure 9).

On ER(10,20) graphs, most methods exhibit consistent improvement with increasing sample size across all metrics, though the rate of improvement tends to stagnate at larger sample sizes. The exception is GES (BIC), which does not follow this trend, an expected outcome given its misspecified parametric assumptions on nonlinear data. Among the remaining methods, PF-GES achieves the

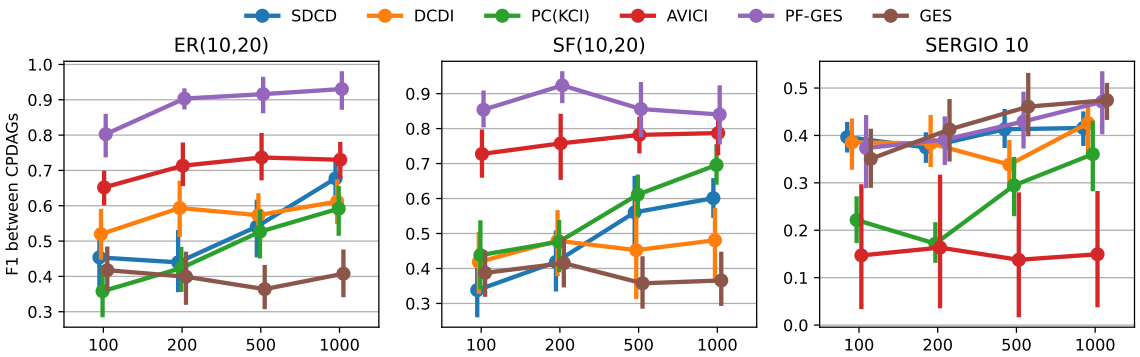

Figure 9: Comparison of causal discovery methods on 10-node graphs (ER, SF, and SERGIO) across varying sample sizes (100, 200, 500, 1000) for F1 score metric. Error bars represent 95% bootstrap confidence intervals calculated across 10 seeds.

best performance across all sample sizes on SHD and F1, with F1 improving with sample size across methods. The SID results follow a similar pattern, although the improvement with additional samples is less pronounced or absent for some baselines.

On SF(10,20) graphs, the performance of PF-GES, AVICI, and GES (BIC) becomes largely flat across sample sizes, suggesting that these methods extract most of the recoverable structure already from 100 samples. In contrast, DCDI and SDCD show continued improvement as more data becomes available. Nevertheless, PF-GES achieves the best or comparable performance across all sample sizes and metrics.

The most nuanced results emerge on the SERGIO dataset. For SHD, AVICI and GES (BIC) do not improve with increasing sample size—initially matching PF-GES but falling behind as the sample size grows. Meanwhile, PF-GES, DCDI, and SDCD all benefit from additional samples. For SID, there is little to no improvement across sample sizes for most methods, with DCDI, SDCD, and GES closely matching or slightly outperforming PF-GES, while AVICI performs the worst. For F1, SDCD initially outperforms PF-GES at the smallest sample sizes but is later surpassed by both PF-GES and GES (BIC). Overall on SERGIO, GES (BIC) matches the performance of PF-GES across most settings, while AVICI remains significantly worse across all sample sizes and metrics.

B.3. Comparison with DiBS

We compare agains the DiBS approach (Lorch et al., 2021), the results are presented in Figure 10. DiBS performs comparably to DCDI and SDCD. Our approach achieves better results in all compared cases under all metrics, often by a large margin.

B.4. Numerical experimental results

We provide numerical results for Figure 1 in the Table 6.

Appendix C. Data Generation Details

We generate data according to SCMs with varying parametric constraints, which are described in detail below. During generation, the data are normalized to prevent the creation of evaluation datasets

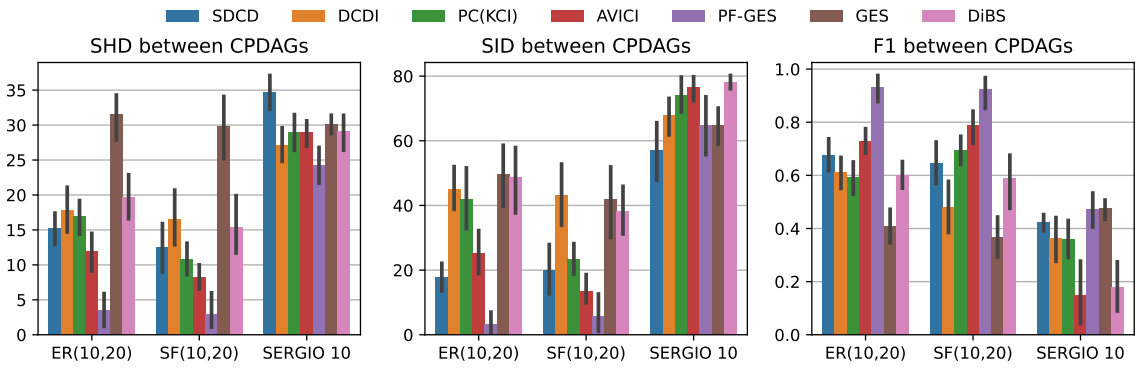

Figure 10: Comparison with DiBS(pink bar) on synthetic and simulated (SERGIO) datasets. Error bars show 95% bootstrap confidence intervals.

SCM type	DCDI	SDCD	PC (KCI)	AVICI	PF-GES	GES (BIC)
ER(5,5)	4.0 (2.1, 7.1)	6.4 (4.5, 8.6)	4.0 (3.0, 5.0)	4.3 (2.9, 5.8)	0.8 (0.0, 2.5)	6.0 (3.6, 7.9)
SF(5,5)	5.8 (4.0, 7.8)	4.1 (3.2, 5.4)	1.7 (0.8, 2.9)	0.5 (0.2, 0.9)	0.2 (0.0, 1.0)	4.6 (2.3, 7.7)
ER(10,20)	15.2 (12.9, 17.3)	17.8 (14.8, 21.2)	16.9 (14.1, 19.0)	11.9 (9.1, 14.6)	3.5 (1.5, 6.3)	31.5 (26.8, 33.9)
SF(10,20)	12.5 (9.1, 16.0)	16.5 (13.3, 21.5)	10.8 (8.5, 13.2)	8.2 (6.5, 10.2)	3.0 (1.4, 7.7)	29.8 (25.4, 34.5)
SERGIO 10	34.7 (32.6, 37.8)	27.1 (24.9, 29.6)	29.0 (26.3, 31.5)	29.0 (26.5, 30.4)	24.2 (21.6, 26.8)	30.1 (28.9, 31.6)
ER(20,40)	36.2 (29.8, 44.7)	30.8 (27.0, 34.7)	29.3 (26.1, 33.2)	18.4 (16.5, 21.3)	7.6 (4.7, 12.8)	100.8 (93.6, 110.4)
SF(20,40)	41.6 (31.9, 58.4)	35.1 (30.4, 44.0)	29.6 (24.8, 42.0)	20.5 (14.7, 39.6)	11.1 (4.8, 34.4)	30.2 (25.3, 38.2)
SERGIO 20	62.4 (59.2, 66.0)	60.4 (56.4, 64.6)	65.0 (59.9, 72.4)	57.4 (53.8, 60.3)	55.0 (47.3, 63.9)	80.0 (72.6, 88.5)

Table 6: Benchmark results on synthetic and simulated (SERGIO) datasets in terms of SHD metric. Values in brackets describe 95% bootstrap confidence intervals.

that can be solved using simple heuristics (Reisach et al., 2023, 2021). Variable values are generated in topological order. After each variable is generated, it is normalized before being used as input to the mechanisms generating its children. All mechanisms are aligned with previous work Brouillard et al. (2020).

Neural networks with additive noise This mechanism is implemented using a two-layer neural network with width 8 and a ReLU nonlinearity. Network weights are initialized from a standard normal distribution, while biases are initialized from a uniform distribution $\mathcal{U}(-\frac{1}{\sqrt{8}}, \frac{1}{\sqrt{8}})$. The inputs to the network consist of the parent variables, and noise is incorporated additively. Noise is sampled from a Gaussian distribution with zero mean and variable-specific variance, where $\sigma^2 \sim \mathcal{U}(0.02, 0.1)$ independently for each variable.

Neural networks with non-additive noise This mechanism uses a two-layer neural network with width 8 and a `tanh` nonlinearity. As in the additive case, weights are initialized from a standard normal distribution and biases from $\mathcal{U}(-\frac{1}{\sqrt{8}}, \frac{1}{\sqrt{8}})$. The inputs to the network include both the parent variables and a noise term. Noise is sampled from a zero-mean Gaussian distribution with independent, variable-specific variances drawn from $\sigma^2 \sim \mathcal{U}(0.02, 0.1)$.

Linear data with uniform noise This mechanism is based on linear functions whose coefficients are sampled from $\mathcal{U}(0.25, 1)$. The sign of each coefficient is sampled independently, with equal probability of being positive or negative. Additive noise is applied, where the noise is sampled from a variable-specific uniform distribution.

Appendix D. Bootsrapping details

The 95% confidence intervals reported throughout the paper are computed using the BCa (bias-corrected and accelerated) bootstrap method as implemented in `scipy.stats.bootstrap` (Virtanen et al., 2020) with default parameters (9999 resamples, 95% confidence level). For each dataset configuration, we compute the bootstrap confidence interval of the mean metric (e.g., SHD) across the evaluation datasets.

Appendix E. Marginal-Likelihood-Based Causal Discovery: Guarantees and Discussion

PF-GES leverages Bayesian hypothesis tests to compare candidate graphs. The comparison between candidate graphs relies on the computation of the *marginal likelihood*. Recent work has shown that, when used for Bayesian model selection, marginal likelihood, as opposed to maximum likelihood, can yield full identification of the data-generating graph without additional assumptions. In this section, we clarify the differences between our method and these approaches, and explain why PF-GES does not fall under the scope of these stronger identification results. PF-GES is not a fully Bayesian causal discovery method, as it does not estimate a posterior distribution over graph structures.

Theoretical guarantees PF-GES and the methods of Dhir et al. (2024, 2025b) operate under fundamentally different paradigms, leading to distinct theoretical guarantees and trade-offs. PF-GES (and nonparametric GES methods more broadly) enjoys *pointwise consistency* for Markov equivalence class recovery: given a sufficiently large sample size, the algorithm is guaranteed to identify the correct MEC. This is the same type of guarantee provided by other established causal

discovery methods, such as PC (assuming a valid conditional independence test) and DCDI (under sufficiently small regularization).

In contrast, Dhir et al. (2024, 2025b) aim to identify the full directed acyclic graph, including distinctions within a Markov equivalence class, by leveraging Bayesian model selection together with the independent causal mechanisms principle. Their guarantees are probabilistic rather than asymptotic: for example, Theorem B.7 in Dhir et al. (2025b) shows that the probability of error is strictly smaller than that of random guessing, and empirical results suggest that this error is typically low. This form of probabilistic performance guarantee is fundamentally different from pointwise consistency. How these differing paradigms compare in real-world causal discovery tasks remains largely unexplored and constitutes an interesting direction for future work.