Causal Discovery in the presence of Latent Confounders

A Stochastic Optimization Approach

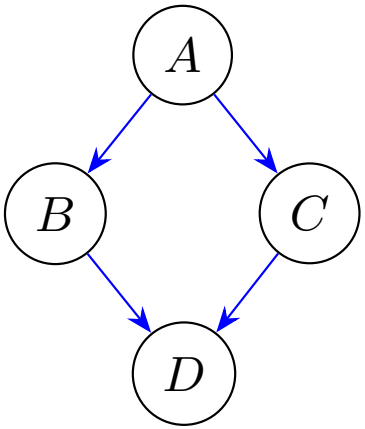
Danish Shakeel Mohammad

Queen Mary, University of London

December 2, 2025

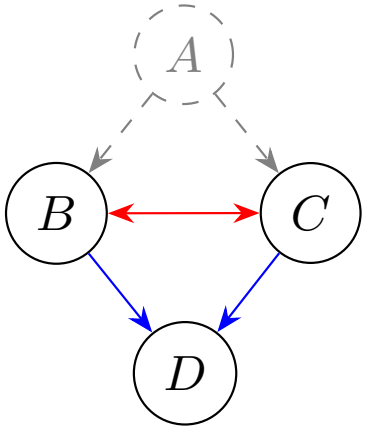
Introduction

Causal Discovery is the automated extraction of cause and effect relationships from data.



DAG

Causally Sufficient



ADMG

Causally Insufficient

Linear Gaussian SCMs

$$\mathbf{V} = \mathbf{\Theta V} + \mathbf{U}$$

$$\mathbf{U} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Omega})$$

$\mathbf{V}$: observable variables
$\mathbf{U}$: unobservable variables
$\mathbf{\Theta}$: structural coefficients matrix
$\mathbf{\Omega}$: error covariances matrix

In the presence of latent confounding, off-diagonal elements of $\mathbf{\Omega}$ are non-zero.

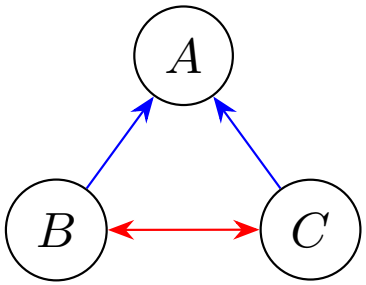
Definition (Causal effect — real world definition)

A variable X has a causal effect on the variable Y if forcing X to take some value x, the distribution of Y explicitly depends on x:

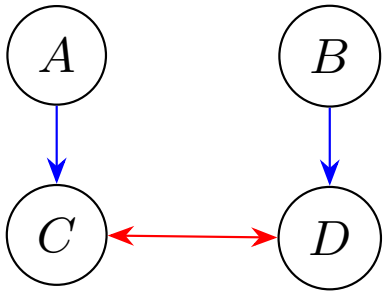
$$\exists\, x \in \mathcal{X} : P(Y|\text{do}(X = x)) \neq P(Y)$$

Graph Classes and Problem Statement

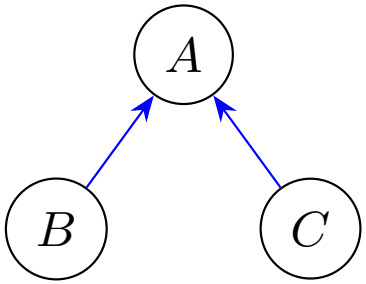
Bow-free



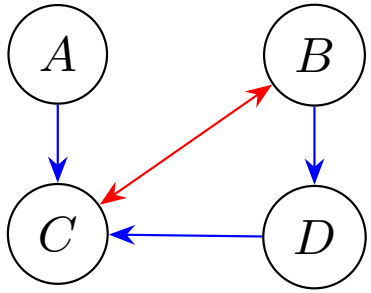
Ancestral



Non Bow-free



Non Ancestral



Bow-free ADMGs

1. In the case of Linear Gaussian SCMs are almost-everywhere identifiable, in the limit of infinite data
2. Can capture Verma Constraints

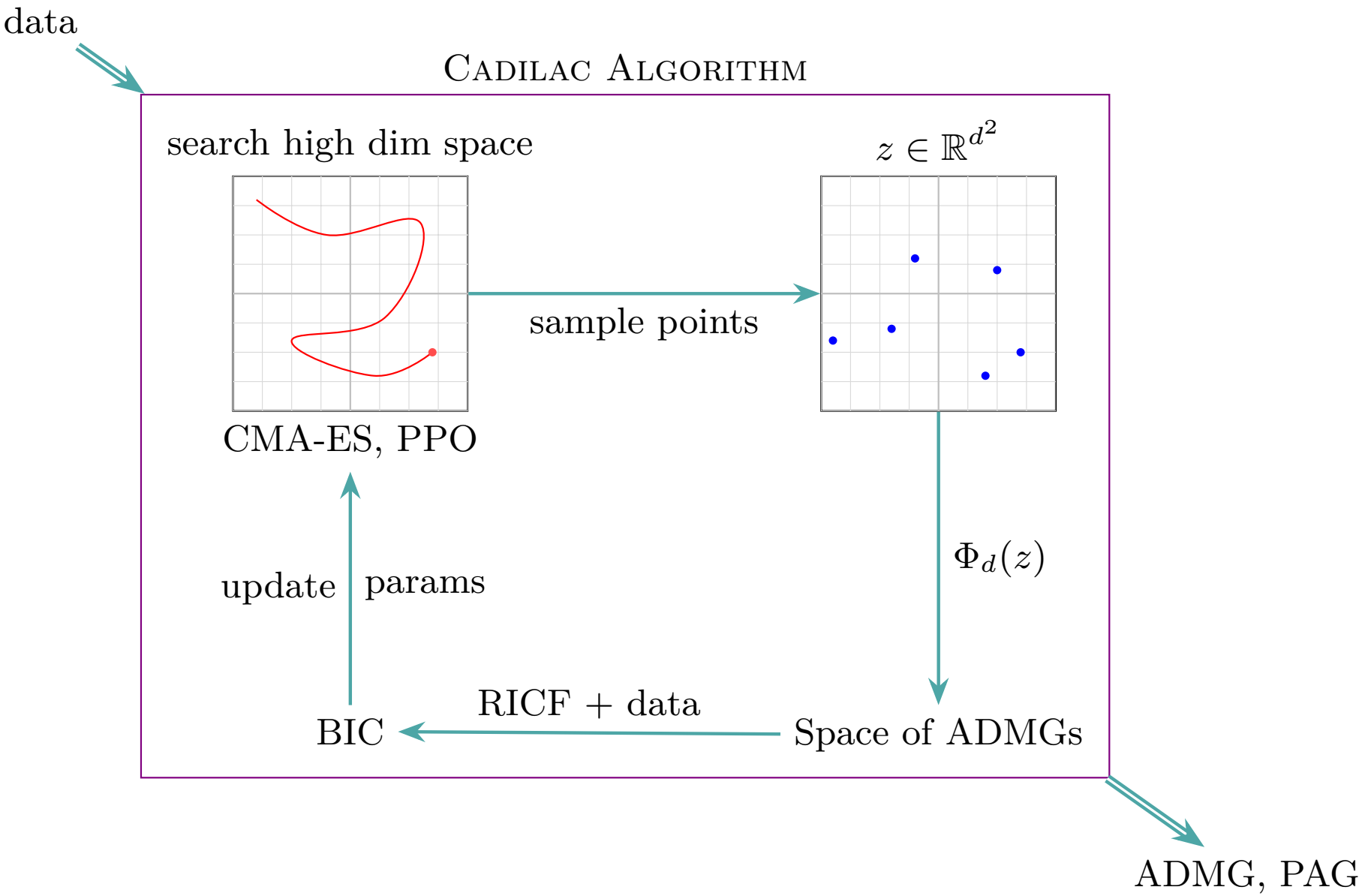
Ancestral ADMGs

1. In the case of Linear Gaussian SCMs are globally identifiable, in the limit of infinte data
2. Can't capture Verma Constraints
3. Are a subset of Bow-free ADMGs

Problem Statement

Causal Discovery on observational data for Linear Gaussian Structural Causal Models (SCMs), targeting ancestral and bow-free Acyclic Directed Mixed Graphs (ADMGs).

Methodology



Vec2ADMG Mappings

Definition

$\forall\ d \in \mathbb{N}^+$ and $z \in \mathbb{R}^{d^2}$, $p \in \mathbb{R}^d$ is the vector formed from the first d elements of z, $E_{\rightarrow}, E_{\leftrightarrow} \in \mathbb{R}^{d \times d}$ are strictly lower triangular matrices formed from the next $\frac{d(d-1)}{2}$ and final $\frac{d(d-1)}{2}$ elements of z, respectively:

$$\Phi_d^{BF}(z)[D] := H\left(E_{\rightarrow} + E_{\rightarrow}^{\top}\right) \odot H\left(\text{grad}(p)\right)$$
$$\Phi_d^{BF}(z)[B] := H\left(E_{\leftrightarrow} + E_{\leftrightarrow}^{\top}\right) \odot \Psi(D)$$
$$\Phi_d^{AN}(z)[D] := H\left(E_{\rightarrow} + E_{\rightarrow}^{\top}\right) \odot H\left(\text{grad}(p)\right)$$
$$\Phi_d^{AN}(z)[B] := H\left(E_{\leftrightarrow} + E_{\leftrightarrow}^{\top}\right) \odot \Psi(D^+)$$

where $\Psi(M) := (I - M) \odot (I - M^{\top})$,
$H(x) := \begin{cases} 1, & \text{if } x > 0 \\ 0, & \text{if } x \leq 0 \end{cases}$ is the Heaviside step function, $\odot$ is the element-wise product, $\text{grad}(u)_{ij} := u_j - u_i$, and A^+ is the transitive closure of A.

Properties

1. Φ_d^{BF} has time complexity $\mathcal{O}(d^2)$
2. Φ_d^{AN} has time complexity $\mathcal{O}(d^{2.8})$
3. Automatic acyclicity
4. No differentiable constraints needed
5. Surjective
6. Scale and Translation Invariance

Optimization Algorithms

PPO

1. Deep RL algorithm
2. Stochastic Gradient Descent based optimizer
3. Tries to find optimal policy π_θ which maximises $J(\pi_\theta) = \mathbb{E}_{\tau\sim\pi_\theta}[\sum_{t=0}^{\infty}\gamma^t r_t]$
4. Policy updates are limited to trust region to add stability to training
5. $\pi_\theta(z) = \mathcal{N}(z; \mu_\theta, \mathrm{diag}(\sigma_\theta^2))$
6. $R(z) = -\frac{1}{n}\mathrm{BIC}(X, \Phi_d(z))$
7. We do entropy annealing to help escape local minima
8. One-step environment
9. Designed to function well in high dimensional spaces
10. On-policy algorithm, sample in-efficient

CMA-ES

1. Evolutionary algorithm
2. Stochastic derivative-free black-box optimizer
3. Iterative algorithm: sample, rank, update loop
4. Tries to find samples from the search space that maximise an objective function
5. We restrict covariance matrix to be diagonal for scalability
6. Objective fn
   $f_d(z) = \mathrm{BIC}\left(X, \Phi_d(z)\right) - \gamma\Gamma(z)$
   $\Gamma(z) = \sum_{i<j}^{\{i,j\}<d} \min(|z_i - z_j|, \delta) + \sum_{k,k\geq d} \min(|z_k|, \delta)$
7. Designed to function even in non-convex or ill-behaved landscapes
8. Sample efficient

Comparison Algorithms and Data Generation

GFCI

1. Hybrid Algorithm: Has constraint based and score based phases
2. Only outputs a PAG.
3. Used as a baseline for comparison.

DCD

1. Differentiable constraints for acyclicity and to restrict search to bow-free/arid/ancestral graph classes
2. Uses modified RICF algorithm
3. Uses augmented Lagrangian to obtain unconstrained optimization problem
4. Solves the optimization problem with dual descent

Synthetic Data Generation

1. Uses modified version of Erdős-Rényi random graph generation model
2. Modification accounts for existence of bidirected edges, requirement for bow-free/ancestral graphs
3. Inputs: average degree of graph skeleton $\bar{\rho}$, fraction of directed edges $f^{\rightarrow}$
4. Guarantees: $|\hat{E}| = \bar{\rho}d/2 \pm 5$ and $\widehat{f^{\rightarrow}} = f^{\rightarrow} \pm 0.1$

Table: $\boldsymbol{\Theta}$ is the structural coefficients matrix and $\boldsymbol{\Omega}$ error covariances matrix

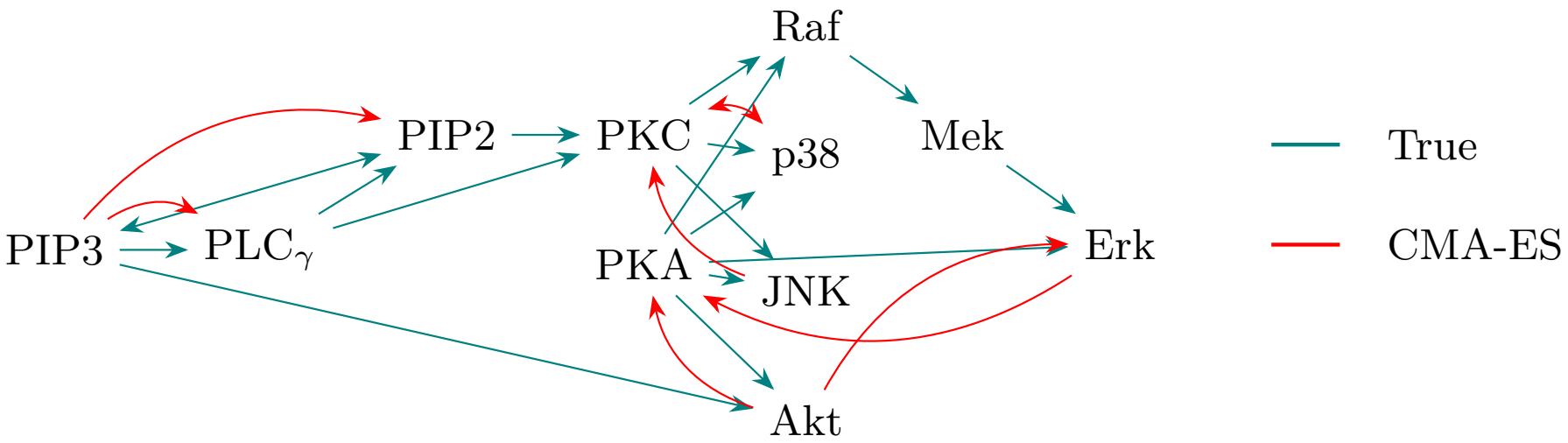
Matrix	**Distribution**
$\boldsymbol{\Theta}$	$\mathcal{U}(\pm[0.5, 2])$
Off-diag $\boldsymbol{\Omega}$	$\mathcal{U}(\pm[0.4, 0.7])$
Diag $\boldsymbol{\Omega}$	$\mathcal{U}([0.7, 1.2] + \sum(\lvert\boldsymbol{\Omega}_{i,-i}\rvert)$

Sachs Dataset

Table: Performance of Algorithms on Sachs Dataset

	SHD (↓)	$\lvert E \cap \hat{E} \rvert / \lvert \hat{E} \rvert$[1](↑)	**PAG** $\mathbf{F_1}$[3](↑)	τ[2](↓)
DCD	53	3 / 43	0.47	249.5
CMA-ES	22	1 / 8	0.58	105.7
Relcadilac	23	1 / 9	0.48	1029.2

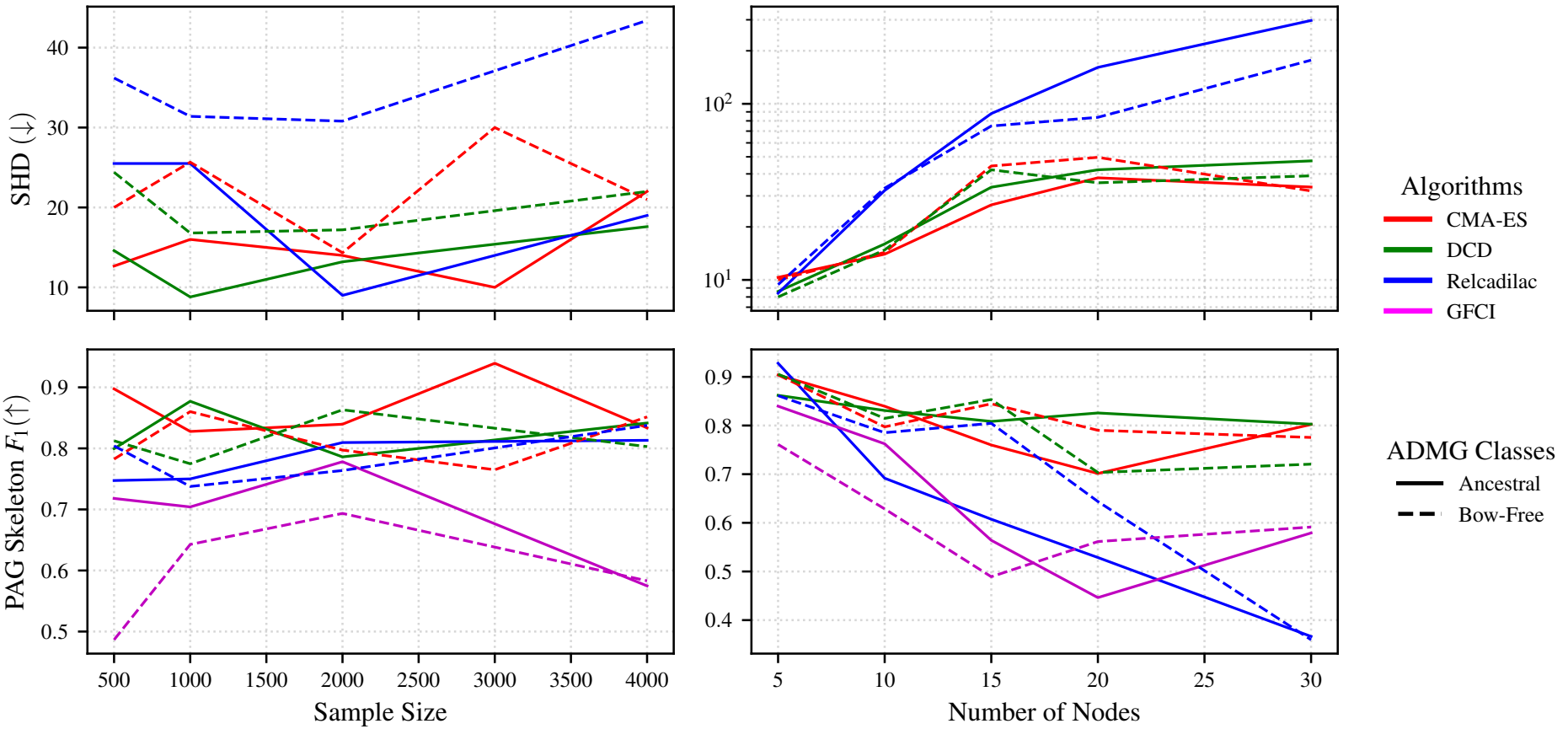
[1] $|E \cap \hat{E}|$ is the number of correct predicted edges, and $|\hat{E}|$ is the total number of predicted edges.
[2] τ is the runtime of the algorithm in seconds.
[3] The F_1 score is computed on the skeleton of the PAG.



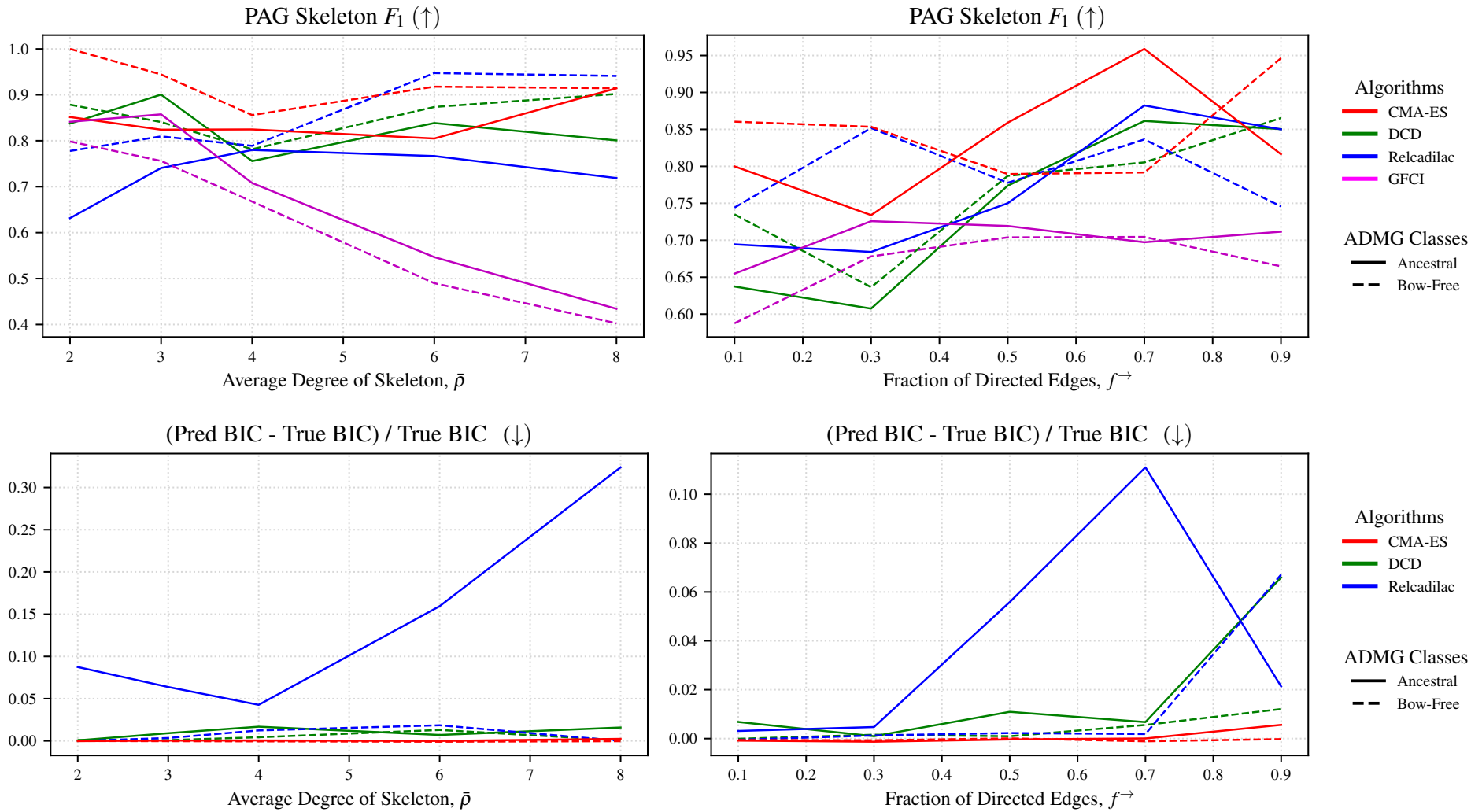
Performance Plots I

Varying Parameters: Sample Size, Number of Nodes, Average Degree of Skeleton, Fraction of Directed Edges

Metrics Captured: Structural Hamming Distance (lower is better), F_1 score of the edges of the PAG Skeleton (higher is better), Fractional Excess BIC score (lower is better)



Performance Plots II



Future Work

1. Explore the SAC RL algorithm for potentially better sample efficiency than PPO
2. RICF algorithm bi-directed connected components-based decomposition and caching for potential speedup in sparse graphs
3. Extend the Vec2ADMG mappings to include other graph classes like arid graphs