Integer Programming for Causal Structure Learning in the presence of Latent Variables

Sanjeeb Dash
IBM Research AI

Joint work with Tian Gao (IBM), Rui Chen (U. Wisconsin-Madison)

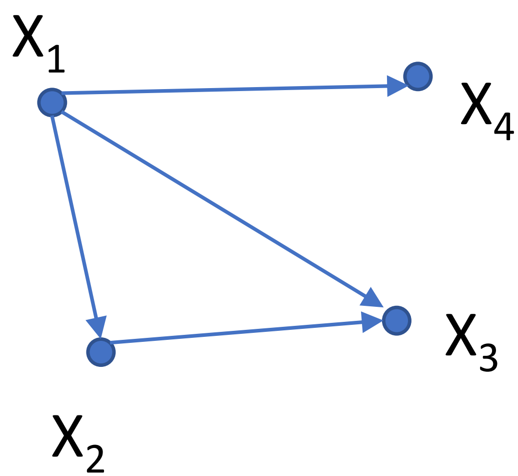
ICML 2021

Outline

- Bayesian Network Structure Learning
- Modeling Latent Variables
- Integer Programming Formulation to find optimal score
- Numerical Experiments

Bayesian Network Structure Learning

Bayesian Network: Directed acyclic graph (DAG) representing conditional probability relationships between variables.



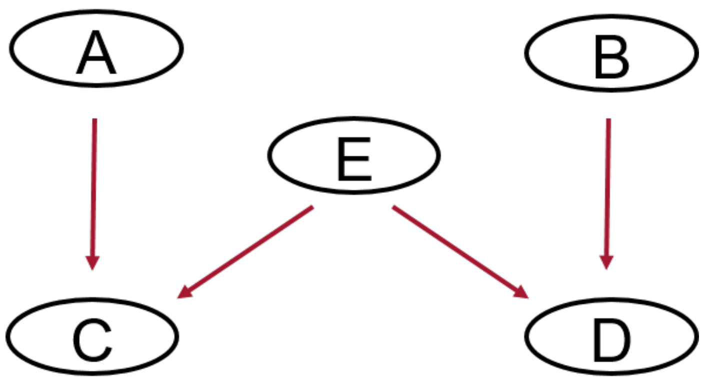
$$P(X_1, X_2, X_3, X_n) = P(X_4|X_1)P(X_3|X_1, X_2)P(X_2|X_1)P(X_1)$$

BNSL Problem - Learn DAG from data:
DP methods: Koivisto, Sood '04, Silander, Myllymäki '06
A* search: Yuan, Malone '13
Branch-and-bound: Campos, Ji '11
IP based solver GOBNILP: Bartlett, Cussens '13, '17
GOBNILP is a state-of-the-art method: Malone et. al. '17

Causal Bayesian Networks

- Graphical Models where directed edges represent causal relationships
- DAG encodes *structural equations*

Directed Acyclic Graph

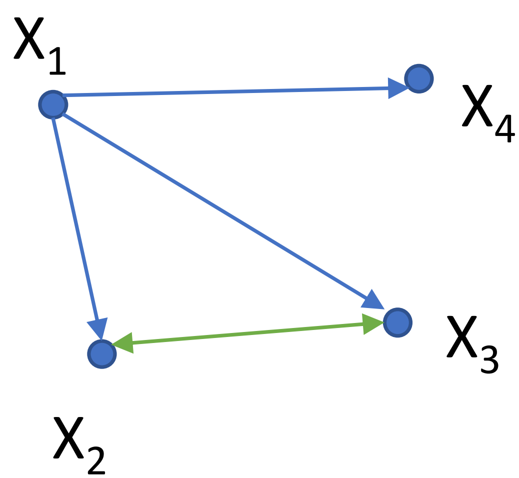


(Linear) Structural equations

$$\Leftrightarrow \begin{cases} x_A = \epsilon_A \\ x_B = \epsilon_B \\ x_C = b_{CA}x_A + b_{CE}x_E + \epsilon_C \\ x_D = b_{DB}x_B + b_{DE}x_E + \epsilon_D \\ x_E = \epsilon_E \end{cases}$$

Latent Variables

Goal: Learn causal network structures in the presence of latent vars.

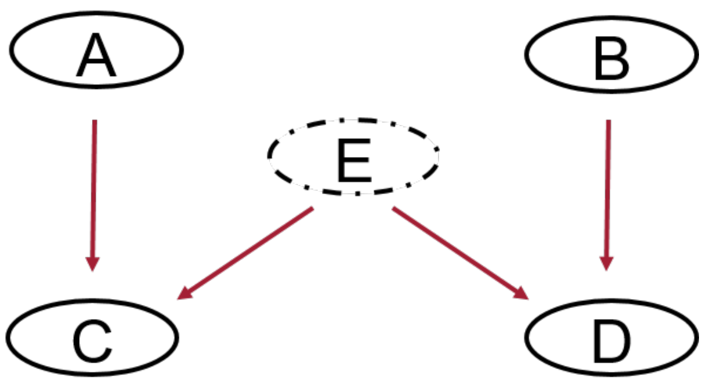


We use **ancestral acyclic directed mixed graphs** (with directed + bidirected edges) as models of data with latent confounders.

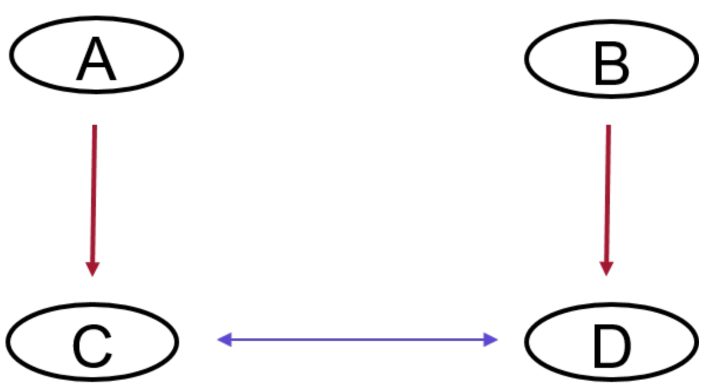
Chen, Dash, Gao ’21: MIP formulation & first exact score-based method to find optimal AADMG for continuous Gaussian variables.

Ancestral graphs (AGs)

- DAGs are not closed under marginalization!



Ancestral graphs (Richardson and Spirtes '02)



- Include all DAGs and are closed under marginalization
- Properties:

No directed cycles
$(a \to b \to \ldots \to a)$
No almost directed cycles
$(a \leftrightarrow b \to c \to \ldots \to a)$

Learning methods

Constraint-based methods:

► Apply conditional independence test on the data to infer the graph structure: FCI (Sprites et al., '00), cFCI (Ramsey et al., '12)

Score-based methods:

► Optimize a scoring criterion that measures the likelihood of the data: GSMAG (Triantafillou and Tsamardinos, '16)

Hybrid methods:

► Use both a scoring criterion and conditional independence tests: M^{3}HC (Tsirlis et al., '18), SPo (Bernstein et al., '20), CCHM (Chobtham and Constantinou, '20)

Current score-based and hybrid methods are all greedy or local search algorithms!

Scoring a graph

- The BIC score (Schwarz '78) for graph $\mathcal{G}$ is given by

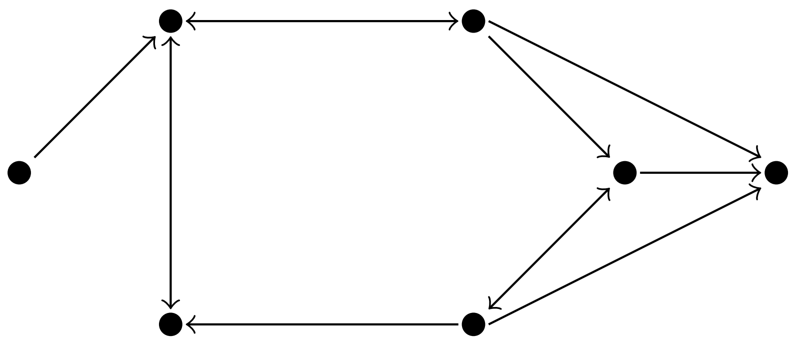
$$\text{BIC}_{\mathcal{G}} = 2\ln(l_{\mathcal{G}}(\hat{\Sigma})) - \ln(N)(2|V| + |E|)$$

- The maximum log-likelihood $\ln(l_{\mathcal{G}}(\hat{\Sigma}))$ can be decomposed by c-components in $\mathcal{G}$ (Nowzohour et al., '17)

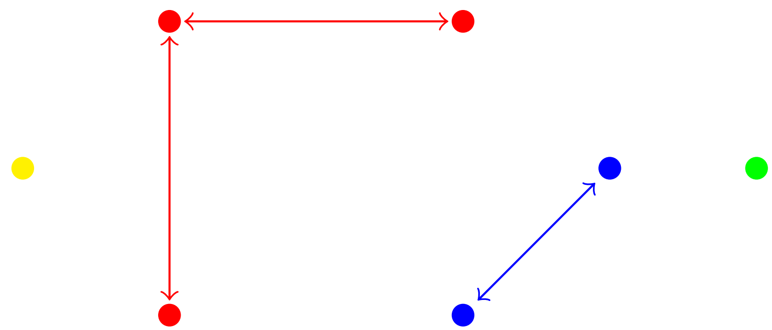
$$\ln(l_{\mathcal{G}}(\hat{\Sigma})) = -\frac{N}{2}\sum_{D\in\mathcal{D}}\Big[|D|\ln(2\pi) + \log(\frac{|\hat{\Sigma}_{\mathcal{G}_D}|}{\prod_{j\in pa_{\mathcal{G}}(D)}\hat{\sigma}^2_{Dj}}) + \frac{N-1}{N} tr(\hat{\Sigma}^{-1}_{\mathcal{G}_D} S_D - |pa_{\mathcal{G}}(D) \setminus D|)\Big]$$

district = component defined by bidirected edges
c-component = district + in-edges per node in district

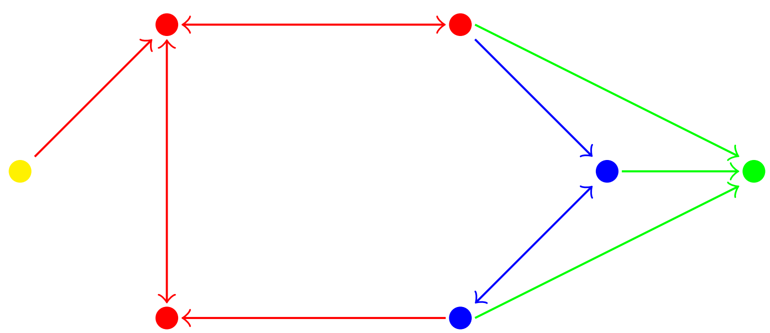
Decomposition into c-components



Ancestral ADMG



Districts

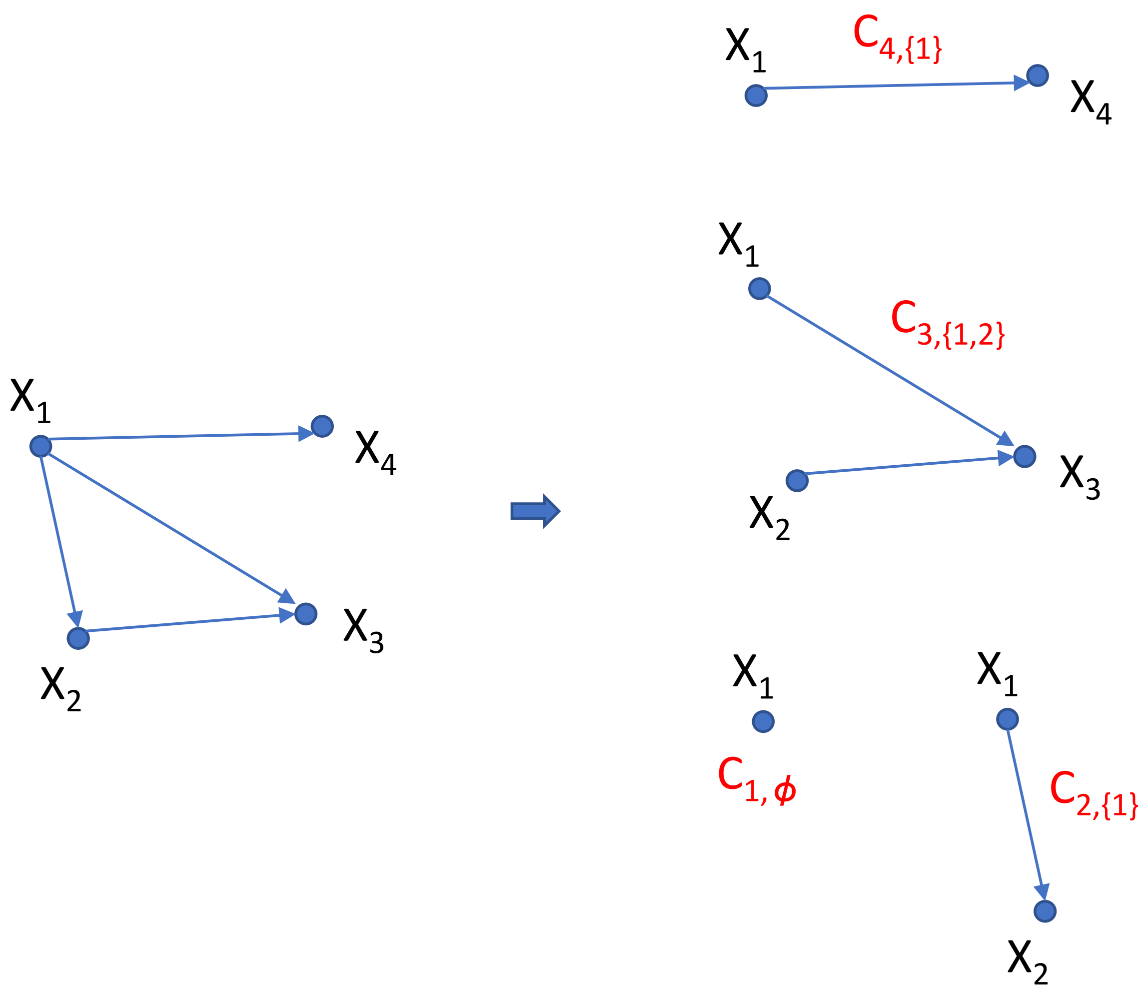


c-components

- We obtain a (BIC) score-maximizing ancestral ADMG for a set of continuous variables that follow a multivariate Gaussian distribution.

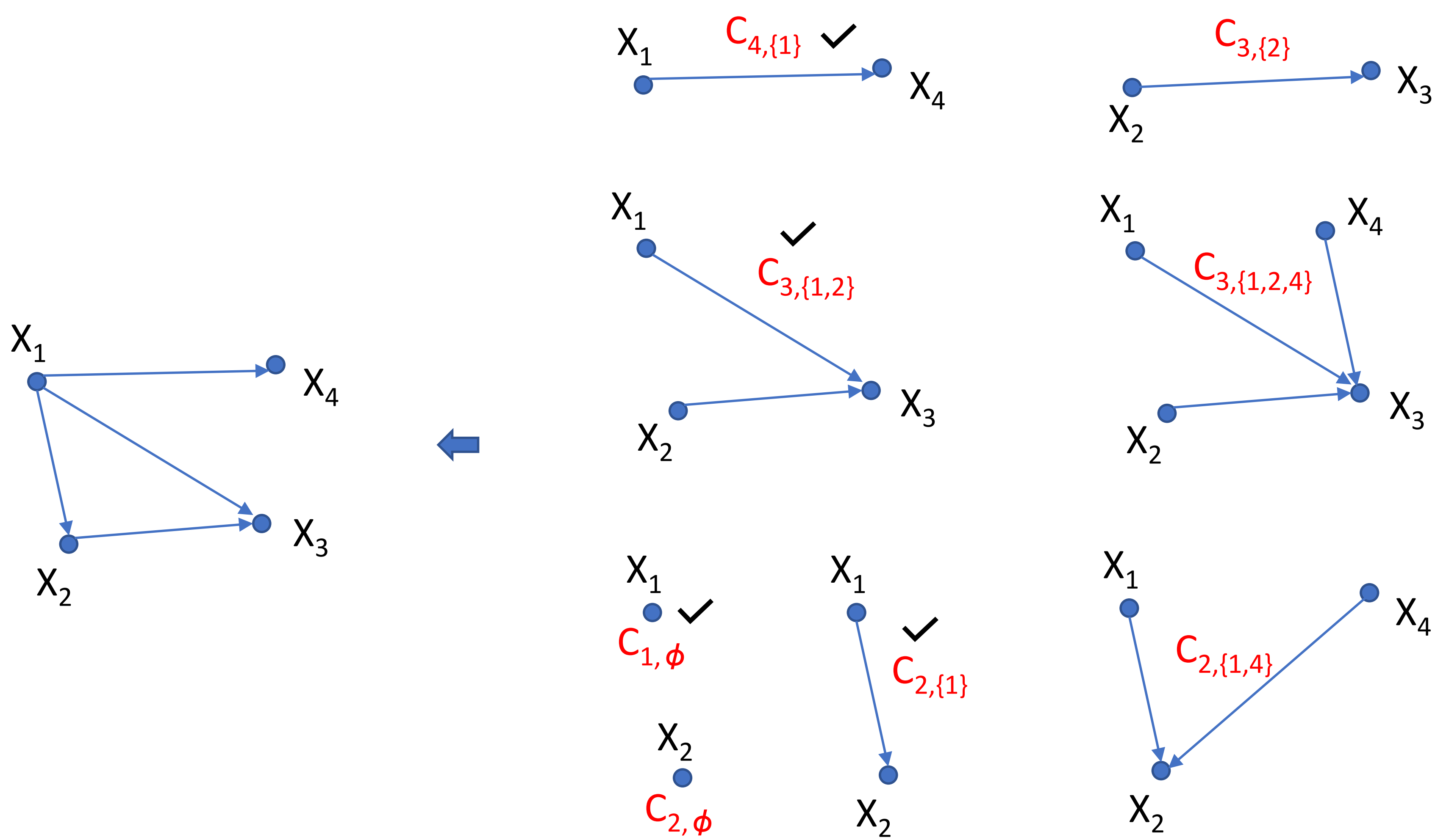
Score decompositions for BNSL

Score of DAG is sum of scores of "in-stars" (inward directed star)



MIP for score based approach

MIP has one variable per in-star, equations choosing one in-star per node, and *cluster inequalities* preventing cycles.



Opt. formulations

Notation: Node set - $V = \{1, \ldots, n\}$, $P(i)$ = set of parent sets of i.

MIP (parent set variables):

$$\max \sum_{i \in V} \sum_{P \in P(i)} c_{i,P} z_{i,P}$$

$$\sum_{P \in P(i)} z_{i,P} = 1, \;\; \forall i \in V$$

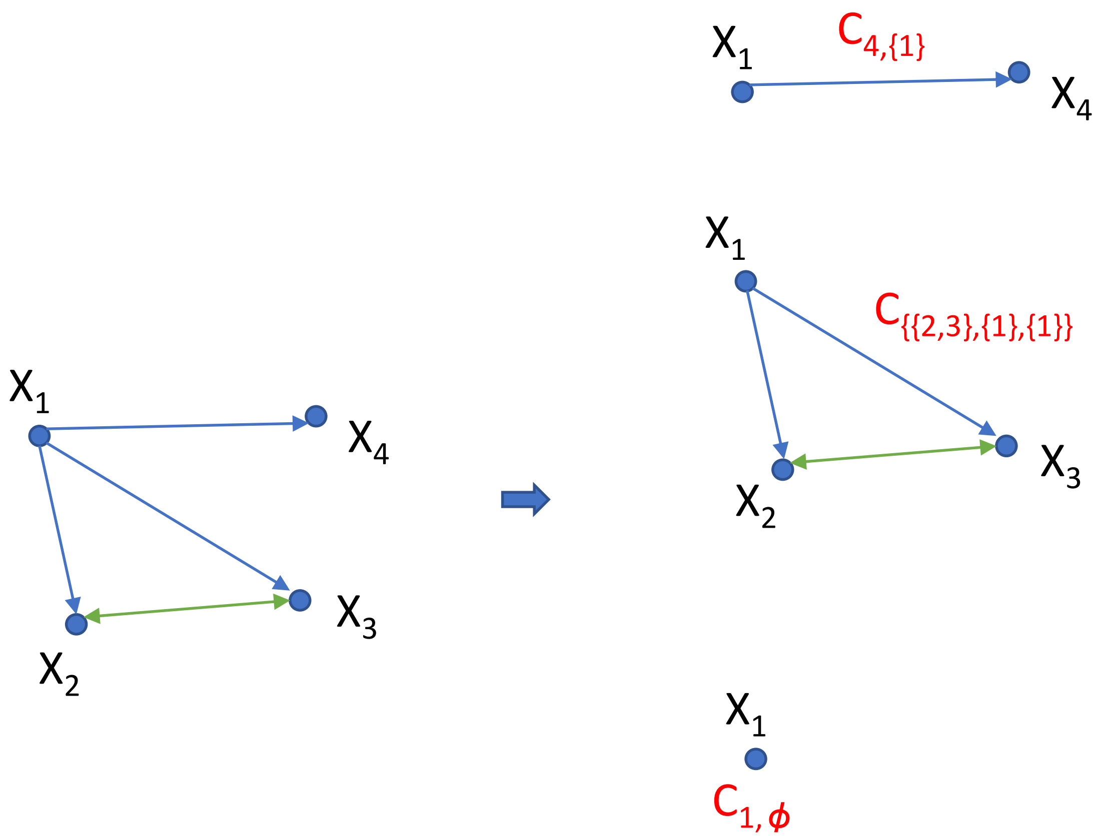
$$\sum_{i \in S, P \cap S = \emptyset} z_{i,P} \geq 1, \;\; \forall S \subseteq V \;\; *$$

$$z_{i,P} \in \{0, 1\}$$

Jaakkola, Sontag, Globerson, Meila ’10: cluster constraints(*)
Bartlett, Cussens ’13, 17: IP + software (GOBNILP)
Grotschel, Junger, Reinelt ’85: Acyclic subgraph polytope

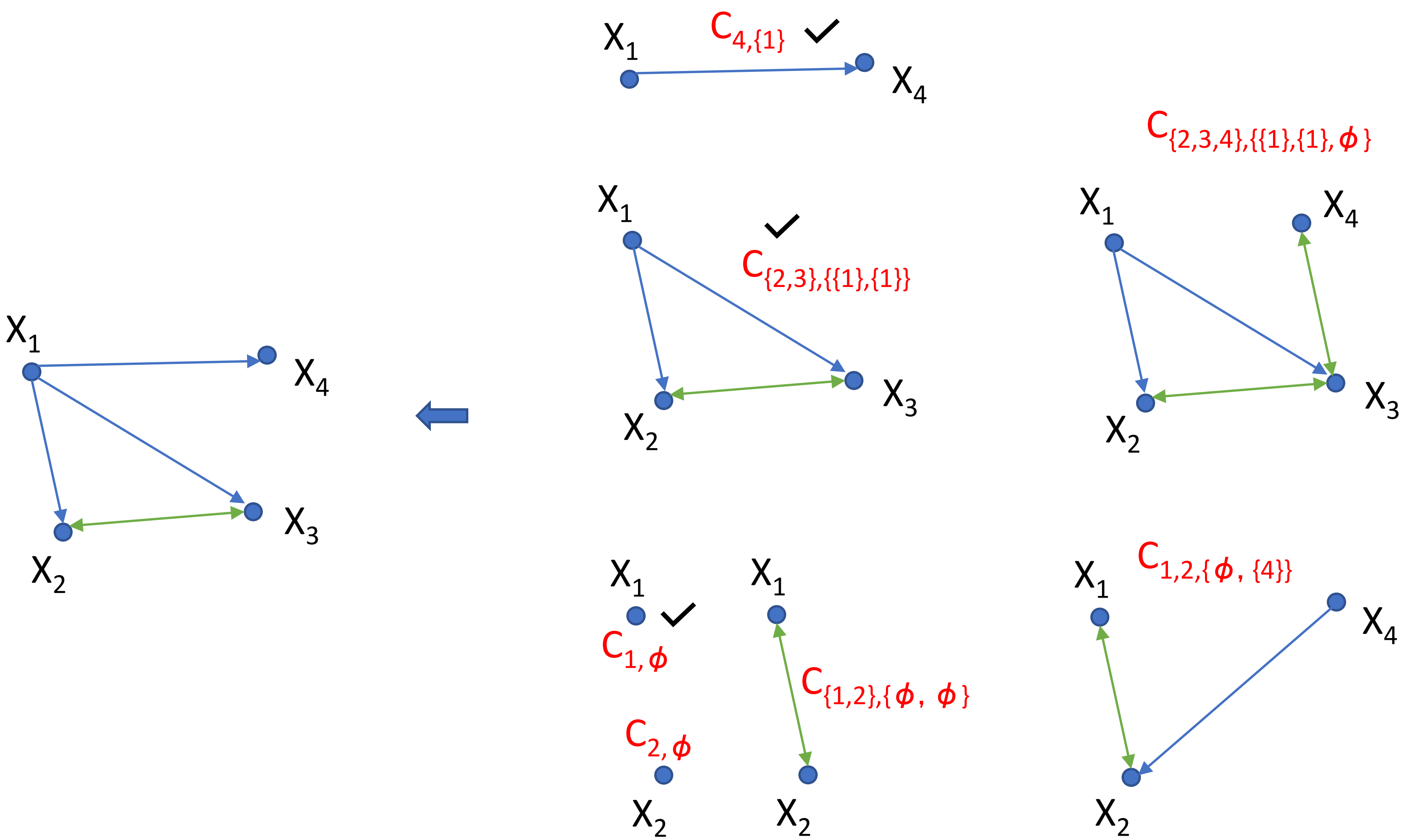
Score decomposition for AADMG

Score of AADMG is sum of scores of c-components



Approach

Our work: Learn an AADMG with maximum score from c-components



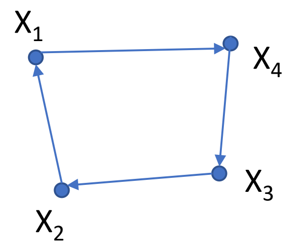
MIP formulation

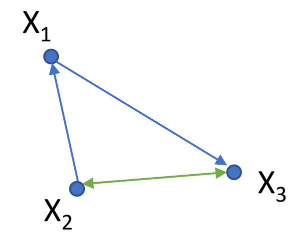
Let $\mathcal{C}$ be set of all c-components, and let $D(C)$ be the district of a c-component C.

MIP to find optimal AADMG:

$$\begin{aligned} \text{max} \quad & \sum_{c \in \mathcal{C}} s_C z_C \\ & \sum_{C : i \in D(C)} z_C = 1, \ \ \forall i \in V \\ & G(z) \text{ has no directed and almost directed cycles} \\ & z_C \in \{0, 1\} \end{aligned}$$

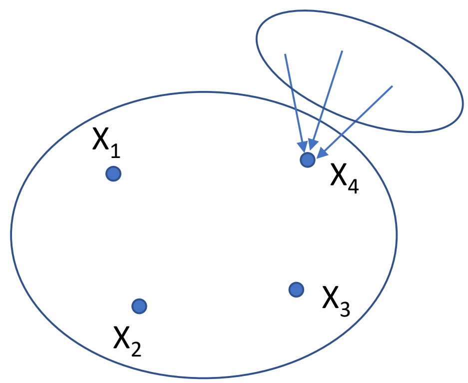
Cutting planes to avoid cycles

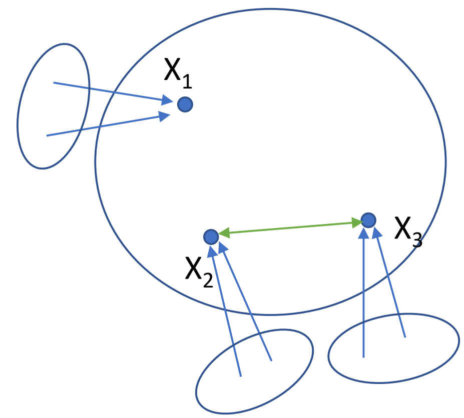




Cluster Inequalities:

$$\sum_{i\in S, P\cap S=\emptyset} z_{i,P} \geq 1, \ \ \forall S \subseteq V$$





Biclusterer inequalities: $(w_{i,j} = \sum_{C: i\leftrightarrow j \in D(C)} z_C)$

$$\sum_{v\in S\setminus\{i,j\}} \sum_{P:P\cap S=\emptyset} z_{v,P} + \sum_{P^1:P^1\cap S=\emptyset} \sum_{P^2:P^2\cap S=\emptyset} z_{i,j,P^1,P^2} \geq w_{i,j}$$

Cutting planes generation

► Karger's ('93) random contraction algorithm for min-cut problems: Randomly contract edge ij with probability $\propto$ edge weight

► *Separation heuristic* for cluster inequalities:
- Let $\mu^k(S)$ denote the LHS of the cluster inequality at iteration k and

$$w_{ij}^k = \mu^k(\{i\}) + \mu^k(\{j\}) - \mu^k(\{i,j\}),\ \forall i,j$$

- At iteration k, randomly contract edge ij with probability $\propto w_{ij}^k$
- Remove nodes i and j, create a pseudo-node i' and replace all occurrences of i and j in the original graph by the pseudo-node
- Repeat until $\mu^k(\{i\}) < 1$ for some $i \Rightarrow$ a violated cluster inequality

► Similar separation heuristic for bi-cluster inequalities

Numerical Experiments

- Test set 1:
  1. Randomly generated DAGs with 20 nodes
  2. $l =$ 2,4,6 variables set to be latent
  3. $d =$ remaining observed variables
  4. A sample of $N = 1000/10,000$ realizations of observed variables per instance

- Candidate c-components:
  1. Single-node districts with up to three parents
  2. Two-node districts with up to one parent each node

- Compared methods:
  1. AGIP: our IP model
  2. DAGIP: our IP model with only single-node districts
  3. M^{3}HC: a greedy hybrid method by Tsirlis et al. (2018)
  4. FCI: an exact constraint-based method by Sprites et al. (2000)
  5. cFCI: an exact constraint-based method by Ramsey et al. (2012)

Quality of formulation

20-node graphs; d = number of observed nodes, l = number of latent variables (removed from graph), N = number of samples.

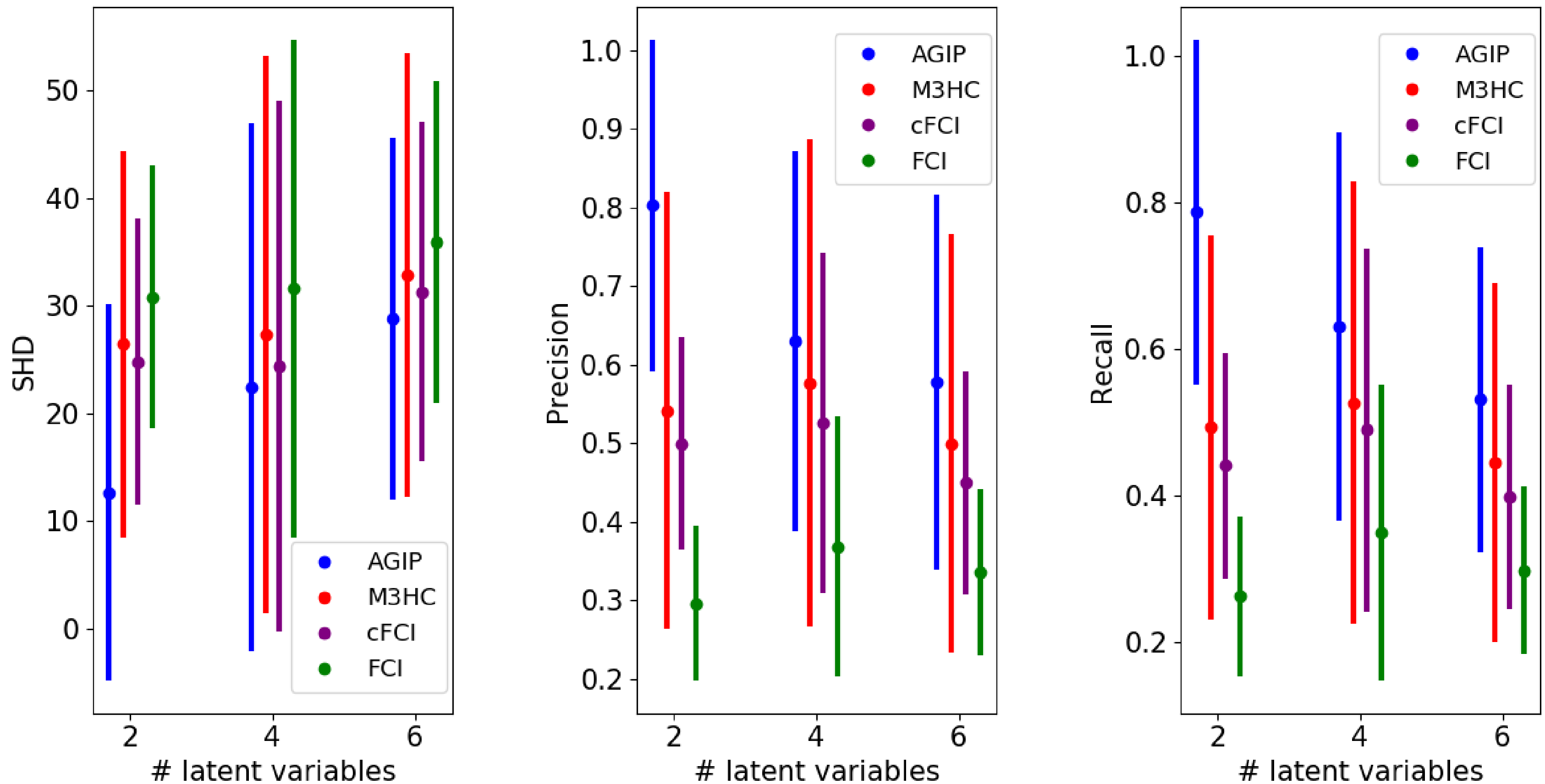
(d, l, N)	Avg # bin vars before pruning	Avg # bin vars after pruning	Avg pruning time (s)	Avg root gap (%)	Avg soln. time (s)
$(18, 2, \ 1000)$	59229	4116	19.1	0.65	60.4
$(16, 4, \ 1000)$	39816	3590	13.6	0.43	41.0
$(14, 6, \ 1000)$	20671	1788	3.9	0.54	8.9
$(18, 2, 10000)$	59229	9038	33.0	0.67	323.2
$(16, 4, 10000)$	39816	7378	21.4	0.53	215.4
$(14, 6, 10000)$	20671	3786	6.4	0.56	47.2

Comparison with a heuristic method

(d, l, N)	Avg improvement in score compared with M^{3}HC		# AGIP score $>$ DAGIP score
	AGIP	DAGIP	
$(18, 2, 1000)$	82.75	82.32	3/10
$(16, 4, 1000)$	90.03	89.33	5/10
$(14, 6, 1000)$	34.84	34.68	3/10
$(18, 2, 10000)$	373.44	373.44	0/10
$(16, 4, 10000)$	147.96	147.54	1/10
$(14, 6, 10000)$	150.52	150.44	1/10

Results for varying number of latent vars.

$d = 18$, $l = 2, 4, 6$, $N = 10,000$,



Results on non DAG-representable graphs

$d = 10$, $l = 10$, $N = 10,000$,

Graph index	Avg SHD		Avg precision (%)		Avg recall (%)		# AGIP score $>$ DAGIP score
	AGIP	DAGIP	AGIP	DAGIP	AGIP	DAGIP	
1	6.7	**6.6**	**63.7**	59.5	**64.4**	60.0	10/10
2	**9.2**	10.5	**59.4**	50.5	**63.0**	52.0	7/10
3	**8.0**	8.8	**67.3**	64.8	**63.8**	60.0	5/10
4	**29.8**	**29.8**	27.4	**29.2**	17.6	**19.0**	4/10
5	**21.7**	23.0	**30.0**	27.6	**27.3**	24.7	2/10
overall	**15.1**	15.7	**49.6**	46.3	**47.2**	43.1	28/50