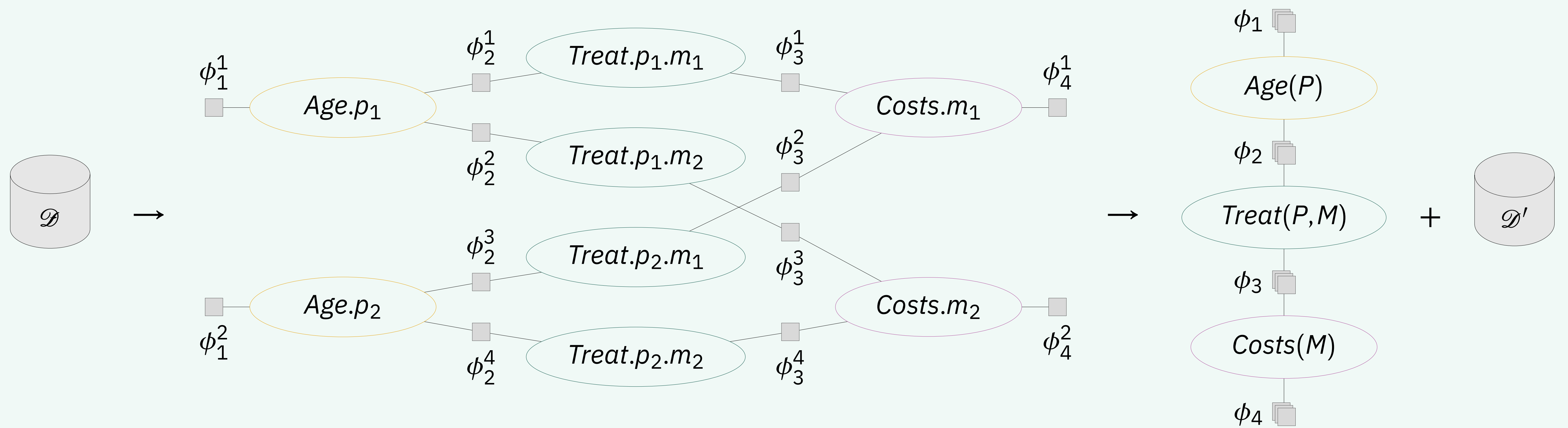


Towards Privacy-Preserving Relational Data Synthesis via Probabilistic Relational Models

Malte Luttermann, Ralf Möller, and Mattis Hartwig

1. Motivation and Overview

Goal: Synthesise data to make it openly available without revealing sensitive information



2. Approach

- (1) Cluster each entity in the database
- (2) Add a node for every attribute and every relation to the graph
- (3) Use conditional independence tests to add edges between nodes
- (4) Count values in the database to obtain potential values for the factors
- (5) Detect and exploit symmetries in the resulting factor graph

3. Step 1: Clustering

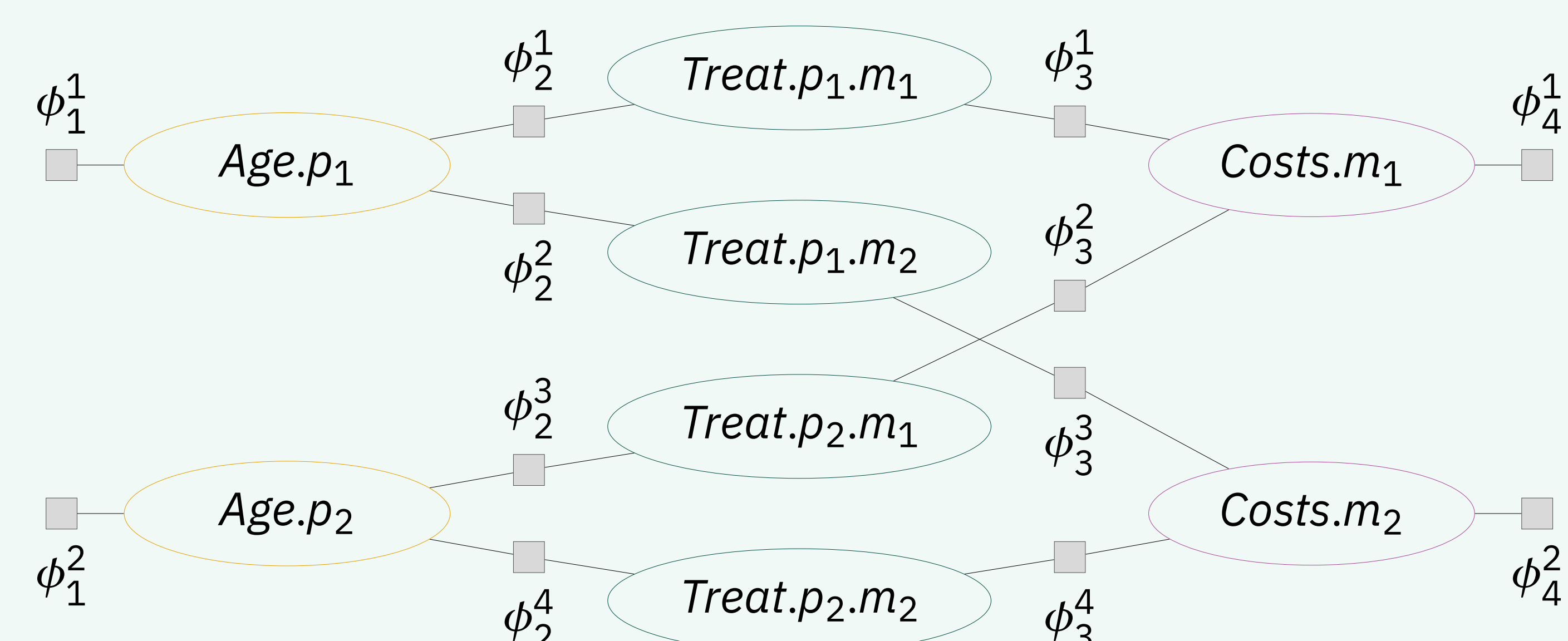
- Compute clusters for all entities using an arbitrary clustering algorithm
- Example:

PatientId	Age	C	PatientId	MedicationId	MedicationId	Costs	C
alice	≥ 18	p_1	alice	myalept	myalept	high	m_1
bob	≥ 18	p_2	alice	danyelza	danyelza	high	m_1
charlie	≥ 18	p_2	bob	paracetamol	paracetamol	low	m_2
dave	< 18	p_2	charlie	ibuprofen	ibuprofen	low	m_2
eve	< 18	p_1	eve	eliquis	eliquis	high	m_1

- Patient: $p_1 = \{\text{alice}, \text{eve}\}$, $p_2 = \{\text{bob}, \text{charlie}, \text{dave}\}$
- Medication: $m_1 = \{\text{myalept}, \text{danyelza}, \text{eliquis}\}$, $m_2 = \{\text{paracetamol}, \text{ibuprofen}\}$

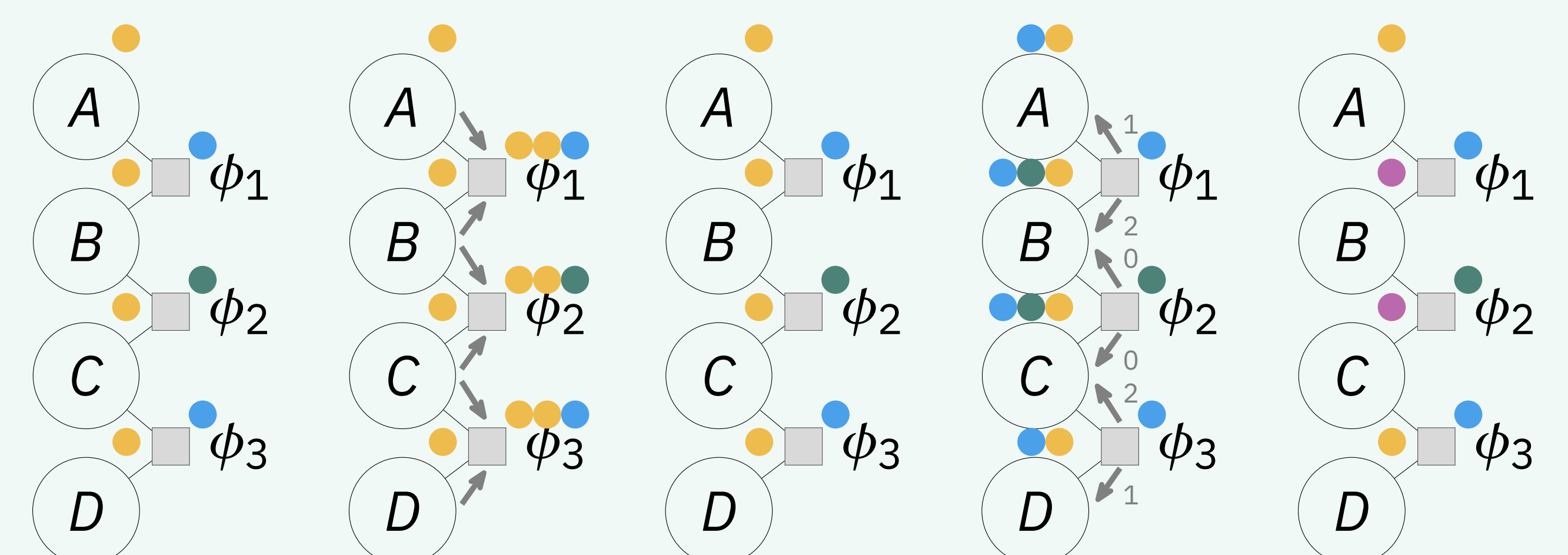
4. Steps 2-4: Constructing a Propositional Model

- Use clusters to add nodes for every attribute and every relation
- Use conditional independence tests to add edges
- For each factor, count occurrences of values by cluster in the database to obtain potentials

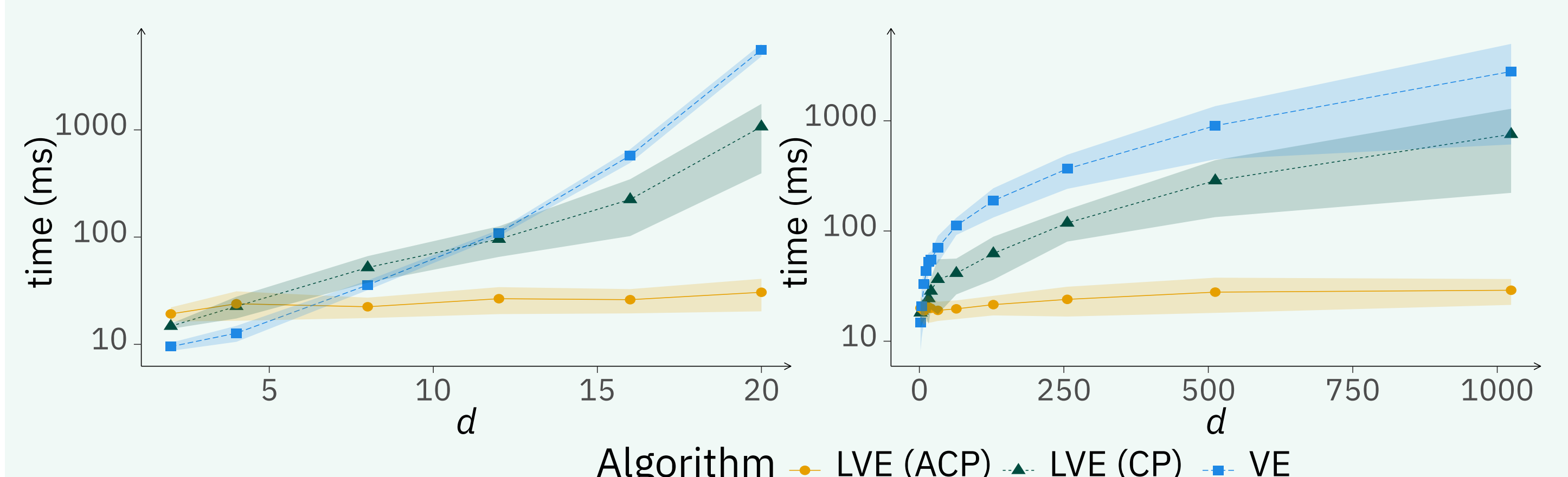


5. Step 5: Constructing a Lifted Model

- Advanced Colour Passing to lift a propositional model
- Pass colours around to detect symmetries in a graph



- Reasoning over cohorts also speeds up probabilistic inference



6. Sampling to Generate Synthetic Data

- A probabilistic relational model compactly encodes a full joint probability distribution over a set of random variables
- Sample from the distributions of the cohorts
- Release data sets for further use without privacy leakage

