# Verifiable Parameterization of Bayesian Networks from Scientific Literature: Unlocking Unstructured Empirical Evidence

**Jonas Gottal and Florian Matthes**
Technical University of Munich
School of Computation, Information and Technology
Department of Computer Science
{jonas.gottal, matthes}@tum.de

## Abstract

Learning Bayesian Networks typically requires access to raw tabular data to estimate conditional probabilities. However, in many scientific domains, raw data is unavailable due to privacy concerns or general lack of access, while structured statistical summaries are increasingly accessible through large language models and published literature. We propose and evaluate five distinct strategies to reconstruct local conditional probability tables solely from statistical summaries in order to parameterize Bayesian Networks. Our comprehensive evaluation across mixed-type synthetic networks demonstrates that copula-based methods significantly outperform standard baselines, offering a viable path for knowledge integration from heterogeneous sources – unlocking the wealth of published knowledge for causal modeling while ensuring transparency and verifiability.

## 1 Introduction

Causal reasoning is the foundation for reliable algorithmic decision-making (Kern et al., 2025). Unlike purely associative models, causal Bayesian Networks (BNs) allow us to predict the effects of interventions and reason counterfactually about complex systems. However, constructing these models requires not only a structural understanding of dependencies but also precise quantitative parameters – the Conditional Probability Tables (CPTs).

The richest and most reliable source of this causal knowledge lies in empirical studies. Yet, as illustrated in Figure 2, the scientific process typically analyzes complex causal mechanisms in fragments rather than as whole systems. Researchers rarely capture a massive, unified dataset covering all variables; instead, they isolate specific relationships, rendering statistical results in the form of t-tests, ANOVA, regression coefficients, or correlations. Consequently, valuable causal knowledge is disseminated as aggregated statistics scattered across multiple publications. With raw data rarely available, there is currently no standard methodology to merge these heterogeneous, isolated studies into a coherent BN.

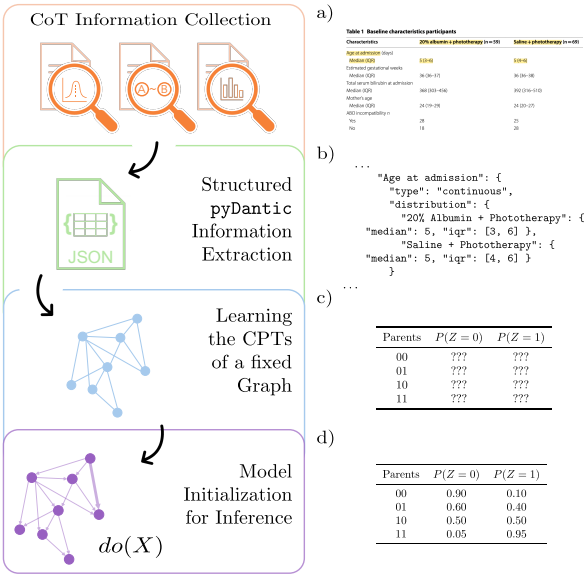


Figure 1: Learning local CPTs from statistical summaries: Given one of the real world examples (Magai et al., 2019b, p. 3) the relevant information is collected (a), and extracted into a structured JSON report (b). This report is then used to learn the local CPTs of a fixed graph (c), rendering a Bayesian Network (d) that can be used for inference and decision making.

This fragmentation creates a critical gap in causal modelling. While methods exist to learn causal structures from unstructured data or expert annotations, there are no established methods to learn local CPTs directly from statistical summaries. Current parameter learning algorithms overwhelmingly rely on raw tabular data, leaving the vast wealth of untapped knowledge in empirical literature inaccessible. Without accurate CPTs, the network remains a static graph, incapable of supporting probabilistic inference or decision-making.



*Findings of the Association for Computational Linguistics: ACL 2026*, pages 3400–3414
July 2-7, 2026 

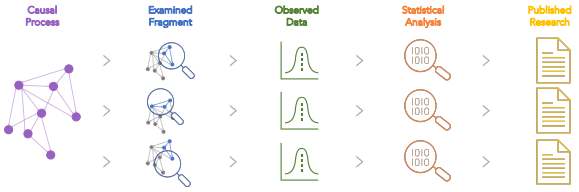


Figure 2: The Empirical Scientific Pipeline: From latent causal mechanisms to observed data and finally to codified knowledge in published research.

The challenge of reconstructing CPTs from statistical summaries is significant. A single statistical report (e.g., "variable $A$ is positively correlated with $B$, $r = 0.5$") or a marginal description (e.g., "variable $A$ has a mean of 24.3 with std of 2.7") imposes constraints on the joint distribution. This shift from raw data to summary-based reconstruction is essential for enabling robust, trustworthy, and transparent modeling in domains where data is scarce or private.

As illustrated in Figure 1 we present an approach to reconstruct the CPTs from unstructured scientific literature. To do so, we propose and rigorously evaluate five strategies to bridge this gap: direct Markov-Chain Monte-Carlo (MCMC) sampling, constrained entropy maximization, Iterative Proportional Fitting (IPF) with proxy targets, Expectation Maximization (EM) on synthetic samples, and a structural copula-based optimization. Additionally, we introduce Large Language Model (LLM) baselines to assess whether general-purpose foundation models can directly infer consistent CPTs from statistical reports. We compare these methods against an upper-bound oracle and a lower-bound random sampling baseline, providing a roadmap for reconstructing functional BNs from verifiable statistical evidence.

We make our code and data available in a public GitHub repository. [1]

## 2 Related Work

Causal reasoning forms the foundation for reliable algorithmic decision-making (Kern et al., 2025), distinguishing purely predictive models from those capable of planning interventions and counterfactual analysis, outlining the prescriptive perspective. Structural Causal Models (SCMs) (Pearl, 2009) provide the necessary formal language to represent these relationships. However, a functional SCM requires two distinct components: the causal graph (structure) and the CPTs (parameters). While the former has seen rapid advancements in text-based discovery, the latter remains significantly constrained by data availability.

### Causal Structure Discovery from Text

While most research has focused on discovering causal structures from tabular data using constraint-based or score-based algorithms, expert-annotated causal graphs – derived from domain knowledge or systematic literature review – offer an alternative source of high-quality structures without requiring raw datasets. Recently, text-based causal discovery has further expanded these capabilities by leveraging scientific literature (Ban et al., 2023b; Wan et al., 2025). Research in this space has followed two primary directions.

The first attempts to identify complete causal networks through direct prompting, asking LLMs to construct graphs from scratch. This remains an extremely challenging task, exceeding current model capabilities due to the complexity of reasoning across multiple variables simultaneously (Gendron et al., 2024). The second direction uses LLMs to either refine graphs initially generated from tabular data, restricting applicability to domains where such data already exists (Long et al., 2022, 2023; Ban et al., 2023a; Khatibi et al., 2024; Jiralerspong et al., 2024), or to use Retrieval Augmented Generation to extract associational relationships from literature to induce causal structures (Zhang et al., 2024).

However, none of these approaches address the critical challenge of learning the local CPTs. Without accurate parameterization, even a perfect causal graph remains a static diagram, incapable of inference and as such decision-making in complex domains.

### The Dependency on Raw Tabular Data

Conventionally, parameter learning for BNs depends on complete tabular datasets to derive parameters via Maximum Likelihood Estimation (MLE) or Bayesian methods (Russell and Norvig, 2021; Friedman and Koller, 2003). This reliance creates a bottleneck: in many high-stakes domains, raw data is inaccessible due to privacy regulations (e.g., GDPR, HIPAA), competitive constraints, or simply because the data was never aggregated into a single repository.

This scarcity stands in stark contrast to the abundance of statistical findings – t-tests, correlations,

[1] https://github.com/jonasgottal/CPTs

and effect sizes – published in peer-reviewed research. This literature represents a massive, untapped vault of knowledge that current parameter learning methods cannot access.

### Expert Elicitation Approaches

When empirical data are scarce, a possible paradigm involves eliciting CPT parameters directly from domain experts. (Barons et al., 2022) developed methods to reduce elicitation burden by having experts specify boundary distributions, then interpolating intermediate probabilities. (Guo et al., 2022) encode expert knowledge as qualitative constraints and convert these to virtual samples. (Huang et al., 2023) quantify verbal expert assessments via fuzzy logic, applying maximum entropy to balance expert judgments with limited case data. These methods formalize expert beliefs about probability distributions when empirical data are scarce, but available and domain expertise accessible. In contrast, our work targets a different evidence source: reconstructing local CPTs directly from published statistical summaries when both raw tabular data and access to domain experts are unavailable. Leveraging the abundance of published literature, this approach scales automatically via LLM-based extraction.

### Shortcomings of Direct LLM Parameterization

Recent attempts have explored using LLMs to directly parameterize BNs by bypassing data altogether. Nafar et al. (2025) introduced the "Extracting Probabilistic Knowledge" (EPK) framework, which prompts LLMs (e.g., GPT-4) to generate CPTs based on the model's internal pre-trained weights.

While promising for common-sense domains, this approach has fundamental limitations for scientific modeling. It relies on "implicit knowledge", treating the LLM as a black box; it is impossible to verify whether the estimates are derived from outdated, biased, or hallucinated sources (Xiong et al., 2023). Furthermore, these models struggle with complex topologies, such as nodes with many parents or high-cardinality state spaces, where the number of required parameters grows rapidly.

### Our Contribution: Evidence-Based Reconstruction

To bridge the gap between text-derived structures and functional models, we propose a verifiable, evidence-based reconstruction framework. In contrast to direct parameterization, we use the LLM solely as an information extraction tool to parse explicit statistical findings (e.g., "Correlation $r = 0.3$") into structured constraints. We then employ rigorous mathematical optimization to reconstruct CPTs that are statistically consistent with these reported statistics.

This approach ensures that every parameter is grounded in specific, citable evidence rather than the opaque intuition of a language model. In high-risk domains like healthcare and finance, where transparency is paramount, our framework provides a clear audit trail from published findings to learned parameters. Additionally, we introduce a benchmark framework that simulates diverse scenarios of statistical reporting – varying by node size, statistical test, and variable type – allowing for the systematic evaluation of summary-based parameter learning strategies.

## 3 Foundations

To reason based on observations and deal with data uncertainty, we apply statistical methods to learn probabilistic models. This work is founded on Bayesian statistics, where degrees of belief are represented by probabilities stored in Bayesian Networks (BNs).

A BN consists of a directed acyclic graph (DAG) $\boldsymbol{G} = \langle \boldsymbol{V}, \boldsymbol{E} \rangle$ with a finite set of vertices $\boldsymbol{V}$, representing discrete random variables, and a set of directed (no undirected/bidirected) edges $\boldsymbol{E}$ between them. Each node $V_i$ holds some local probability information – represented as a CPT $P(V_i \mid Parents(V_i))$, illustrated in Figure 6. This graphical representation assumes the *Markov assumption* – conditional independence from non-descendants given parents. Thus, the joint distribution factorizes as (Russell and Norvig, 2021; Hastie et al., 2009; Pearl, 2009):

$$P(\boldsymbol{V}) = \prod_{i=1}^{n} P(V_i \mid Parents(V_i)). \quad (1)$$

We assume a fixed BN structure $G$ (expert-annotated or text-derived) and focus on parameter learning from statistical summaries. Continuous variables are discretized via quantile-based binning to enable `TabularCPD` representations in `pgmpy` (Ankan and Panda, 2015). All methods operate within this discrete framework.

The input comprises JSON-formatted statistical summaries extracted from empirical studies, containing marginal distributions and edge-wise constraints (e.g., t-tests, correlations, ANOVA, $\chi^2$). Figure 5 visualizes the prevalence of these tests across scientific literature (Armitage et al., 2013; Mishra et al., 2019; Yan et al., 2017). Due to the concentration on few tests, our methods assess reconstruction quality by simulating data from candidate CPTs and verifying whether the induced statistics match the reported targets.

Structured information extraction to produce these JSONs is well-studied (Dagdelen et al., 2024; Shamsabadi et al., 2024; Odobesku et al., 2025). We demonstrate feasibility via CoT prompting with Pydantic validation on real-world studies (Appendix 3, 2), but make no contribution in this area.

## 4 Methodology

Our main contribution comprises five distinct strategies to reconstruct local CPTs from statistical summaries, plus baselines. All strategies operate on a fixed BN skeleton and JSON reports containing edge-wise statistical constraints and marginal summaries. To score candidate CPTs, the core mechanism shared by all optimization-based methods is simulation-based evaluation: given a candidate BN, we simulate $n = 500$ samples, (re-)compute the same family of statistical tests used to generate the JSON reports, and define an objective as the discrepancy between simulated and target statistics.

We evaluate three complementary settings: (i) a controlled synthetic benchmark for parameter learning (maximal control and ablations), (ii) a standalone extraction evaluation on real papers, and (iii) an end-to-end real-world evaluation of our pipeline. Because no publication-to-causal-BN gold standard is available, we use paired evidence (each paper PDF and its released raw dataset) and treat BNs learned from the released raw data as proxy references rather than expert-annotated causal graphs.

### 4.1 Direct CPT Optimization via MCMC

As a direct, model-free baseline for BN parameter reconstruction, we implement Metropolis-Hastings MCMC over the CPT probability masses. The Markov chain state is the full set of CPT entries for the fixed network structure, initialized from a random Dirichlet baseline ("zero knowledge") (Metropolis et al., 1953; Viinikka and Koivisto, 2020).

At each step, the proposal distribution perturbs a single local conditional distribution: it samples a random node and (if the node has parents) a random parent-configuration column of its CPT, adds small Gaussian noise to the probability vector, and renormalizes. This yields localized moves that can explore the parameter space without re-sampling an entire CPT from scratch.

The acceptance rule treats the simulation-based discrepancy as an energy function. To score a candidate BN, we draw $n = 500$ samples via forward simulation and re-compute the same families of summary statistics encoded in the JSON constraints (e.g., Pearson correlation, ANOVA, t-test, ...). The loss $L$ is defined as the discrepancy between simulated and target summaries, so each MCMC proposal is evaluated by re-simulating data and re-scoring its induced statistics. Given the current state $\mathcal{B}$ and a proposal $\mathcal{B}'$, we accept $\mathcal{B}'$ with probability $\alpha = \min\{1, \exp(L(\mathcal{B}) - L(\mathcal{B}'))\}$, which always accepts improvements and can accept worse proposals to escape local minima.

For stability, we do not return the final chain state. Instead, we average CPT parameters over the last 50 iterations of the chain and renormalize each CPT column, yielding a smoothed estimate that is less sensitive to simulation noise. Conceptually, the Metropolis-Hastings acceptance rule induces an energy-based pseudo-target proportional to $\exp(-L)$ rather than a likelihood-based Bayesian posterior.

### 4.2 Constrained Optimization Strategy

We formulate CPT reconstruction for a node $C$ and its parents $\mathrm{Pa}(C)$ as a constrained maximum-entropy problem over the *local joint* distribution. Instead of optimizing the CPT directly, we solve for the flattened joint probability vector $x = P(C, \mathrm{Pa}(C))$ by minimizing negative entropy $\sum x_i \log x_i$, subject to the probability axioms $\sum x_i = 1$ and $x_i \geq 0$. The final CPT is recovered by conditioning: $P(C \mid \mathrm{Pa}(C)) = P(C, \mathrm{Pa}(C))/P(\mathrm{Pa}(C))$.

**Encoding statistical constraints.** We translate reported summary statistics into linear or nonlinear equality constraints on $x$. Crucially, since most reports describe pairwise relationships (e.g., $P_i \rightarrow C$), we enforce these constraints on the *marginal* distributions of $x$. For instance, a t-test difference in means becomes a linear constraint on the expected value of $C$ given specific states

of $P_i$ (marginalizing out other parents). Similarly, Pearson correlations and ANOVA $\eta^2$ are enforced as constraints on the covariance and variance of the corresponding pairwise marginals, utilizing the bin-center mapping for discretized variables.

**Solver and integration into a BN.** We solve the resulting nonlinear program using SLSQP (`scipy.optimize.minimize` (Virtanen et al., 2020)), which supports bound constraints and equality constraints. This approach yields the most uninformative (maximum entropy) joint distribution that perfectly satisfies the reported pairwise statistics. In the full BN, we apply this optimization independently for each node $C$ given its parents $\mathrm{Pa}(C)$.

### 4.3 Iterative Proportional Fitting (IPF) Strategy

Our IPF strategy reconstructs CPTs by treating reported pairwise statistics as marginal constraints on the full local joint distribution $P(C, \mathrm{Pa}(C))$ (Deming and Stephan, 1940; Pukelsheim, 2014). The core insight is that a high-dimensional CPT can be approximated by finding the distribution closest to uniformity that matches all available pairwise "proxy targets" derived from the literature.

First, we synthesize **proxy target marginals** $P(C, P_i)$ for each reported parent-child edge. For continuous interactions, we sample from a bivariate Gaussian distribution parameterized by the reported means, variances, and correlation coefficient, and then discretize these samples into the network's state space to form a target contingency table. For binary-to-continuous relations (e.g., t-tests), we simulate data from conditional normal distributions derived from the reported group means and effect size. For categorical interactions, we generate contingency tables by perturbing independent marginal products with stochastic noise scaled by the reported Cramér's $V$.

The algorithm initializes the full local joint probability tensor for a node and its parents with ones (uniform distribution). It then cyclically iterates through each parent constraint, scaling the tensor entries along the corresponding axes to match the proxy target marginals while preserving the interaction structure established by previous steps. Convergence is reached when the maximum parameter change between iterations falls below $10^{-4}$. The converged joint tensor is normalized column-wise to yield the conditional probability table.

### 4.4 Expectation-Maximization (EM) on Synthetic Data

This strategy transforms the summary-based learning problem into a missing-data imputation problem. We assume that while we lack a complete dataset, the available statistics allow us to generate high-quality "partial" datasets for individual edges, which can then be fused to learn the global model. The process consists of two stages: local pairwise synthesis followed by global parameter learning via Expectation-Maximization (EM) (Koller and Friedman, 2010; Dempster et al., 1977).

**Local Pairwise Synthesis.** For each reported relationship $P_i \rightarrow C$, we optimize a pairwise Gaussian copula to generate a synthetic dataset ($n = 500$) that reproduces the reported statistical constraints (Haugh, 2016; Nelsen, 2006). Starting from an initialization based on the reported correlation sign ($r = \pm 0.3$), we repeatedly: (i) perturb the current correlation parameter $r$ with Gaussian noise, (ii) sample latent vectors from the copula and map them to observed domains using the marginal distributions (means/variances or categorical probabilities) specified in the JSON report, (iii) discretize continuous variables into the network's state space via quantile-based binning, and (iv) evaluate a loss function that measures the discrepancy between the synthetic sample's statistics and the reported targets (e.g., difference in group means, Pearson/Spearman correlation, or a signed proxy for $\chi^2$/ANOVA). Proposals are greedily accepted when they reduce this loss, yielding an optimized pairwise sample for each edge.

**Global Learning via EM.** The optimized pairwise datasets are "stacked" into a single sparse global dataframe. In this matrix, each row corresponds to a synthetic sample from a specific edge optimization, where only the relevant parent and child columns are observed, and all other network variables are marked as missing ($\mathrm{NaN}$). We then employ the EM algorithm to learn the parameters of the full BN from this incomplete data. We utilize a modified version of the `pgmpy` EM estimator that handles arbitrary missing data patterns. The E-step computes expected sufficient statistics for the missing entries based on the current CPT estimates, and the M-step updates the CPTs to maximize the likelihood of the completed sufficient statistics. This effectively allows the model to "fill in" the unobserved interactions by propagating in-

formation from the locally synthesized pairwise constraints.

### 4.5 Copula-Based Structural Optimization

To address the high dimensionality of optimizing CPTs directly, we developed a secondary approach that operates in a reduced parameter space. Instead of optimizing probability masses, this method optimizes the correlation parameters of a global Gaussian Copula that links all network variables. The network is parameterized by fixed univariate marginals (derived directly from reported means/frequencies) and learnable edge correlations.

In each optimization step, we construct a global correlation matrix $\Sigma$ from the current edge parameters, ensuring positive semi-definiteness. We then generate a synthetic dataset ($n = 500$) by sampling latent vectors from the multivariate normal distribution and mapping them to the target domains using the inverse cumulative distribution functions (CDFs) of the node marginals. This synthetic data is discretized using quantile binning and evaluated against the reported statistics: the loss is defined as the sum of squared differences between the empirical statistics of the synthetic sample (e.g., group mean differences, correlations) and the reported targets.

The optimization loop performs a greedy hill-climbing search, iteratively perturbing the correlation parameters and accepting changes that reduce this statistical loss. Once the optimal correlations are found, we use the final synthetic dataset to learn the BN structure and parameters via MLE.

By optimizing in the continuous copula space, this method maintains a coherent global dependence structure with drastically fewer free parameters than the full CPT space.

### 4.6 LLM-based Baselines

To benchmark the capabilities of modern generative AI in this domain, we introduced a baseline utilizing GPT-4o-mini, GPT-5.1 and o4-mini. This strategy tests whether a general-purpose language model can few-shot the translation of statistical text into valid probability tables without explicit mathematical modeling. All LLM baselines used `temperature` = 0, 5 independent runs per network (matching other methods), and a prompt template, including few shot templates, iteratively refined during pilot runs with manual inspection.

The method processes the JSON statistical summaries directly. We constructed a prompt template that provides the model with the variable definitions, their states, and the statistical constraints. The model is instructed to output the CPT. To ensure structural validity, we constrain the output using a Pydantic model, which enforces that the output is a valid multidimensional array. This baseline represents a "black box" approach, relying entirely on the LLM's internal representation of statistical relationships.

## 5 Experimental Setup and Evaluation

To validate the efficacy of the proposed reconstruction strategies, we developed a controlled synthetic evaluation pipeline. The pipeline generates a ground-truth BN, synthesizes JSON-style statistical summaries from simulated data, learns CPTs from these summaries via the proposed methods, and evaluates the reconstructed model against the ground truth across multiple dimensions.

Additionally, we evaluate the full pipeline on a set of real-world publications to demonstrate practical feasibility beyond synthetic benchmarks. We select studies that provide access to raw data, which allows us to validate the extracted summaries against statistics computed directly from the source data. For each study, we learn a Bayesian network from the raw data using BIC-scored hill-climbing and treat the resulting DAG as the ground-truth structure. We then extract statistical summaries from the corresponding PDF, reconstruct CPTs from these JSON constraints using our methods, and evaluate the resulting Bayesian networks against the ground-truth models using the same metrics as in the synthetic experiments.

### 5.1 Information Extraction

The information extraction pipeline follows a hierarchical strategy that employs chain-of-thought (CoT) prompting with few-shot examples and advanced table recognition. The process begins with TATR (Smock et al., 2022) to convert PDF documents into markdown, thereby preserving the structural integrity of tables essential for accurate extraction.

After markdown conversion, GPT-4.1 (`gpt-4.1-2025-04-14`) is prompted to identify core dataset characteristics such as sample size ($n$), variable names, and data types (continuous, ordinal, categorical, binary). This stage establishes the dataset's context and foundational schema. Detailed statistical information is then extracted,

including measures of central tendency (mean, median, mode), dispersion (standard deviation, variance, range, interquartile range (IQR)), and distribution (percentiles, confidence intervals). Subsequently, the pipeline extracts variable relationships by identifying correlation coefficients and statistical test outcomes. All extracted information is encoded as JSON and validated with Pydantic (Colvin et al., 2025) to ensure compliance with the further processing pipeline.

This workflow produces robust and reliable data for downstream processing. The use of TATR – with F1 scores of 0.89 - 0.91 in table recognition versus $< 0.34$ for traditional OCR (Adhikari and Agarwal, 2024) – enhances extraction accuracy and is extended by the `gmft` package (Wei, 2025), which optimizes native PDF parsing for scientific literature.

### 5.2 Data Generation and Protocol

We use a randomized but reproducible procedure to generate ground-truth networks $BN_{GT}$ and corresponding constraint sets.

**Ground Truth Generation.** For each experiment, we sample a random directed acyclic graph (DAG) with $N$ nodes (default $N = 5$) and edge probability $p = 0.3$, using a fixed seed for the ground-truth generator to ensure comparability across strategies. Nodes are assigned mixed variable types (`binary`, `ordinal`, `nominal`, `continuous`) to reflect heterogeneous settings. Ground-truth CPTs are sampled from a Dirichlet distribution with $\alpha = 1.0$; continuous variables are represented via a discretized $n$-state proxy with an associated continuous inflation mapping used during summary generation.

**Constraint Synthesis (JSON Reports).** From each $BN_{GT}$ we generate a synthetic population of $n = 10{,}000$ samples and convert it into mixed-type data when continuous mappings are present. We then synthesize JSON reports for each parent-child relation by computing commonly used statistical tests (e.g., Pearson correlation, t-tests, ANOVA, and $\chi^2$/Cramér's $V$) alongside marginal summaries (means/variances or category frequencies) and trend descriptions (e.g., "positive" for correlation). These reports serve as the sole input to all reconstruction strategies (except the MLE baseline), emulating the setting where only aggregated evidence is available rather than raw tabular data.

**Execution Protocol and Baselines.** Each strategy is evaluated over 5 independent data seeds, and we report mean $\pm$ standard deviation across seeds. We include two reference baselines: (i) a lower-bound **Random Guess** baseline that samples CPTs at random (Dirichlet), and (ii) an upper-bound **Baseline MLE** that learns CPTs directly from the full synthetic dataset using MLE, acting as oracle. To study compute-quality trade-offs, the full evaluation is repeated for multiple iteration budgets (e.g., 500/1000/2000/5000), depending on the strategy.

### 5.3 Evaluation Dimensions

We assess reconstruction quality at three complementary levels. The results are grouped accordingly in Table 1.

#### 5.3.1 Parameter-Level Fidelity (CPT Quality)

We compare learned CPTs $\hat{P}$ against the ground-truth CPTs $P$ using distributional distances/divergences aggregated over nodes. In addition to unweighted distances (e.g., Total Variation Distance (**TVD**)), we emphasize frequency-weighted metrics (e.g., Weighted KL (**W-KL**) divergence and Weighted Hellinger (**W-Hell**) distance) that down-weight rare parent configurations using empirical configuration frequencies estimated from the ground-truth samples. The **JSON** TVD metric measures the Total Variation Distance between target JSON statistics and those recomputed from samples of the reconstructed BN.

#### 5.3.2 Inference-Level Accuracy (Predictive Power)

We evaluate probabilistic inference using $N = 500$ held-out test samples drawn from $BN_{GT}$. For each test case we generate queries by selecting leaf variables as targets and treating all remaining variables as evidence, and we compare posteriors under the learned model against the ground truth. We report **Log-Loss** to assess calibration, Posterior Hellinger **(Hell)** distance to quantify the shape similarity of the true and predicted posterior distributions, and prediction Accuracy **(Acc)**, measuring the proportion of cases where the most probable predicted state matches the ground truth state.

#### 5.3.3 Decision-Level Utility (Clinical Value)

To quantify downstream utility, we define a synthetic decision problem to select interventions based on the network's predictions (analog to Section 5.3.2: $N = 500$). We report **Regret**, defined

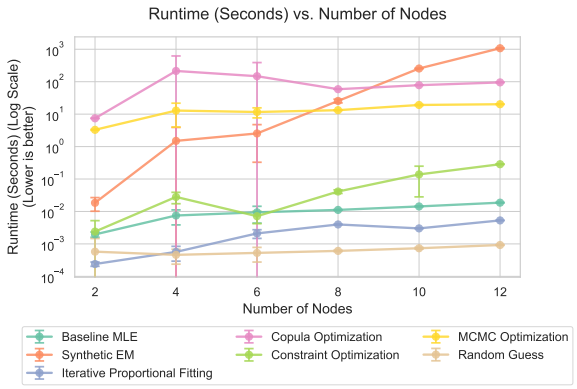


Figure 3: Runtime in seconds for increasing number of nodes in the BN.

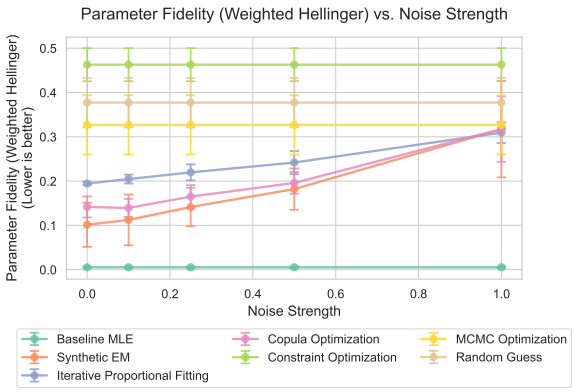


Figure 4: Change in W-Hell under pre-extraction noise (applied before JSON summary generation). Noise strength $s$ scales perturbation: continuous means are shifted by $\mathcal{N}(0, \sigma s)$ and variances by $\exp(\mathcal{N}(0, s))$; discrete probabilities are mixed with a random Dirichlet sample with weight $\alpha = \min(s, 1)$.

as the loss in expected utility when acting under the learned model compared to acting under the ground-truth model, and Decision Accuracy **(Acc)**, the proportion of cases where both models recommend the same action. Decision accuracy can exceed inference accuracy because multiple latent states may map to the same optimal action, making policies robust to certain posterior errors.

## 6 Results

The experiments were run on a MacBook Pro (24 GB RAM, M4 Pro), with dependencies managed via `Poetry` (Eustace, 2018) for reproducibility. Table 1 reports reconstruction performance across parameter fidelity, inference quality, and decision utility, averaged over five random data seeds in our synthetic evaluation pipeline for 500 iterations and mixed variables. The separate information extraction evaluation in Table 3 reveals high precision in extracting variable names and contents. Across all dimensions, the strongest performance is achieved by the two copula-based pipelines (*Synthetic EM* and *Copula Optimization*) and IPF, which form a clear performance cluster separated from direct CPT optimization, constraint optimization, and the LLMs. When noise is added prior to the JSON generation (Figure 4), several strategies show little degradation, suggesting limited use of fine-grained information in the summaries.

### Copula-based strategies best integrate heterogeneous constraints

Both copula-based strategies consistently outperform the non-copula approaches on weighted parameter metrics and downstream utility, indicating that optimizing dependence structure in a continuous latent space and only then projecting to discrete CPTs is an effective way to reconcile mixed-type statistical summaries. In particular, *Copula Optimization* achieves inference accuracy and decision utility substantially closer to the MLE upper bound than direct CPT-space optimization, despite not accessing raw data.

### A clear quality-runtime trade-off

The empirical gains of copula-based learning come with a significant runtime cost. This is expected because for the *Synthetic EM* each copula is optimized separately before sampling synthetic data and then learning from it, while the *Copula Optimization* aims to optimize the joint copula instead which is less efficient. In its current implementation, *Copula Optimization* is substantially slower than the other methods, suggesting that feasibility is demonstrated but computational efficiency remains a key engineering bottleneck.

*Synthetic EM* is comparatively stable, while *Copula Optimization* appears to benefit from larger iteration budgets (Figure 7). The runtime in Figure 3 confirms that the EM algorithm is highly sensitive to increasing node counts, which is expected given the combinatorial growth of the parameter space. The sensitivity analysis in Figure 9 confirms that binary-only settings are easiest, while mixed-type settings remain the most challenging.

For the real-world evaluation, performance is more heterogeneous (Tables 4, 5, and 6). Compared with the synthetic setting, extracted JSON constraints are noisier and often incomplete, and pub-

| | Parameter Fidelity | | | | Inference Power | | | | Decision Utility | | Resource Usage |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Strategy** | **W-Hell** | **W-KL** | **TVD** | **JSON** | **LogLoss** | **Hell** | **KL** | **Acc** | **Regret** | **Acc** | **Time (s)** |
| Baseline MLE | 0.004 | 0.000 | 0.006 | 0.005 | 0.640 | 0.006 | 0.000 | 0.717 | 0.004 | 0.976 | 0.006 |
| | (0.00) | (0.00) | (0.00) | (0.00) | (0.03) | (0.00) | (0.00) | (0.02) | (0.01) | (0.03) | (0.00) |
| Copula Optimization | 0.176 | 0.158 | 0.116 | **0.014** | **0.825** | **0.160** | 0.189 | **0.615** | **1.434** | 0.778 | 5.825 |
| | (0.06) | (0.09) | (0.05) | (0.00) | (0.04) | (0.01) | (0.06) | (0.06) | (0.66) | (0.07) | (0.03) |
| Synthetic EM | **0.133** | **0.116** | **0.102** | 0.059 | 0.831 | 0.176 | **0.187** | 0.569 | 2.121 | 0.520 | 0.394 |
| | (0.09) | (0.10) | (0.06) | (0.00) | (0.03) | (0.02) | (0.02) | (0.03) | (0.20) | (0.02) | (0.21) |
| Iterative Proportional Fitting | 0.306 | 0.408 | 0.301 | 0.132 | 0.871 | 0.213 | 0.238 | 0.574 | 2.115 | 0.635 | 0.001 |
| | (0.00) | (0.00) | (0.00) | (0.00) | (0.03) | (0.01) | (0.02) | (0.02) | (0.11) | (0.01) | (0.00) |
| MCMC Optimization | 0.217 | 0.323 | 0.290 | 0.261 | 1.020 | 0.262 | 0.392 | 0.439 | 4.202 | 0.441 | 1.770 |
| | (0.13) | (0.29) | (0.08) | (0.11) | (0.14) | (0.05) | (0.14) | (0.13) | (1.87) | (0.25) | (0.02) |
| Constraint Optimization | 0.251 | 0.303 | 0.324 | 0.205 | 1.012 | 0.302 | 0.374 | 0.361 | 4.880 | 0.300 | 0.023 |
| | (0.03) | (0.10) | (0.03) | (0.01) | (0.07) | (0.03) | (0.07) | (0.02) | (1.03) | (0.11) | (0.01) |
| gpt-4o-mini | 0.247 | 0.314 | 0.161 | 0.071 | 0.838 | 0.196 | 0.202 | 0.604 | 1.787 | **0.784** | 21.084 |
| | (0.02) | (0.03) | (0.01) | (0.01) | (0.05) | (0.02) | (0.05) | (0.03) | (0.28) | (0.01) | (1.36) |
| gpt-5.1 | 0.281 | 0.446 | 0.176 | 0.036 | 0.853 | 0.199 | 0.230 | 0.483 | 2.295 | 0.718 | 18.742 |
| | (0.08) | (0.15) | (0.05) | (0.01) | (0.07) | (0.05) | (0.06) | (0.15) | (0.47) | (0.07) | (3.52) |
| o4-mini | 0.403 | 0.777 | 0.249 | 0.038 | 0.904 | 0.233 | 0.262 | 0.466 | 2.449 | 0.715 | 80.423 |
| | (0.01) | (0.06) | (0.01) | (0.03) | (0.04) | (0.03) | (0.05) | (0.11) | (1.50) | (0.18) | (15.85) |
| Random Guess | 0.283 | 0.414 | 0.348 | 0.258 | 1.054 | 0.267 | 0.427 | 0.469 | 3.691 | 0.506 | 0.001 |
| | (0.08) | (0.19) | (0.11) | (0.13) | (0.31) | (0.09) | (0.32) | (0.10) | (1.67) | (0.20) | (0.00) |

Table 1: Reconstruction Performance (Mean ± Std). **Bold** indicates best result (excluding Baseline). Ordered by average rank.

lished descriptions do not always perfectly match the released raw data. In addition, many BN edges are not supported by explicit reported statistics, leaving large parts of the parameter space weakly constrained, which limits the observable impact.

### Transparency vs. implicit knowledge

A key distinction from recent LLM-centric BN parameterization approaches is the source of probabilistic information. Nafar et al. (2025) estimate CPT rows by querying an LLM for probabilities and optionally refine them with data by treating the LLM output as an expert prior (pseudocounts). While effective as a general-purpose prior in some domains, this mechanism remains fundamentally opaque: the resulting numbers are not directly attributable to any specific empirical measurement and may reflect memorization, dataset artifacts, or hallucinated numerical estimates.

Our framework instead decouples *extraction* from *inference*: LLMs are used only to extract explicit numeric constraints from empirical reports, and CPTs are then reconstructed by transparent optimization to satisfy these constraints. This design is advantageous in three ways. First, it is verifiable: each learned parameter can be traced back to the corresponding reported statistic and the objective term that enforces it. Second, it supports causal validity in the practical sense that the model is grounded in study-derived constraints (e.g., Randomized Controlled Trials) rather than correlations implicitly absorbed from web-scale text, which can otherwise encode confounding patterns. Third, it is domain-agnostic: the approach does not require the LLM to "know" the domain in advance, since the numerical constraints are provided explicitly via summaries.

## 7 Conclusion and Future Work

This work studies the feasibility of turning abundant published statistical evidence into functional BNs by reconstructing CPTs from JSON-formatted summaries rather than raw tabular data. Across a controlled synthetic benchmark, copula-based strategies provide the strongest overall reconstruction quality and downstream decision utility under the evaluation protocol based on held-out inference queries and utility-based regret.

Real-world evaluation is more challenging than the controlled synthetic setting. A central limitation is that the structures learned from raw data often include edges that are counterintuitive and thus not discussed in the resulting paper. Building an expert-annotated benchmark of pairs of publication and causal BN would provide a stronger foundation for evaluation and is a key direction for future work.

Future work should focus on extending beyond purely discrete CPT parameterizations toward native hybrid models when continuous variables are central.

## 8 Limitations

**Non-identifiability from summaries.** A fundamental limitation of learning from statistical summaries is that most reported statistics constrain the space of compatible joint distributions, but do not uniquely identify a single CPT. Consequently, any method that reconstructs CPTs from summaries can only produce one plausible instantiation that matches the reported evidence, not the unique data-generating parameters.

**Bias, confounding, and reporting error.** All biases present in the underlying empirical studies are inherited by the reconstructed model, and unmeasured confounders or selection mechanisms cannot be recovered if they are not reported. Moreover, our reconstruction assumes the extracted summary statistics are accurate; any misreporting, selective reporting, or extraction/transcription error propagates directly into the learned CPTs because the optimization is driven to satisfy the provided targets.

**Discrete approximation.** Our current framework evaluates all strategies in a discrete BN setting; continuous variables are handled via discretization and remapping during summary generation and evaluation. While this enables unified benchmarking across variable types in `pgmpy`, it can introduce discretization artifacts that disproportionately affect mixed-type settings.

**Scalability across publications.** Our current framework is designed for single-paper reconstruction and does not yet scale to multi-study evidence fusion. A key challenge is harmonizing semantically equivalent variables reported at different granularities (e.g., a binary indicator *smoker/non-smoker*, a continuous exposure *cigarettes per day*, or an ordinal duration *Smoked for 1/5/10/20+ years*). Without such harmonization, constraints from different studies cannot be integrated consistently into one BN. Building robust cross-study alignment and fusion methods is therefore a central direction for future work.

## Acknowledgements

This research has been supported by the German Federal Ministry of Education and Research (BMBF) grant 16IS23069 Software Campus 3.0 (TU München). We would like to thank the anonymous reviewers for their valuable feedback.

## References

2022. How to choose an appropriate statistical test.

Narayan S. Adhikari and Shradha Agarwal. 2024. A Comparative Study of PDF Parsing Tools Across Diverse Document Categories. *Preprint*, arXiv:2410.09871.

Ankur Ankan and Abinash Panda. 2015. Pgmpy: Probabilistic Graphical Models using Python. In *Python in Science Conference*, pages 6–11, Austin, Texas.

Peter Armitage, Geoffrey Berry, and J. N. S. Matthews. 2013. *Statistical Methods in Medical Research*. John Wiley & Sons.

Taiyu Ban, Lyuzhou Chen, Derui Lyu, Xiangyu Wang, and Huanhuan Chen. 2023a. Causal Structure Learning Supervised by Large Language Model. *Preprint*, arXiv:2311.11689.

Taiyu Ban, Lyvzhou Chen, Xiangyu Wang, and Huanhuan Chen. 2023b. From Query Tools to Causal Architects: Harnessing Large Language Models for Advanced Causal Discovery from Data. *Preprint*, arXiv:2306.16902.

Martine J. Barons, Steven Mascaro, and Anca M. Hanea. 2022. Balancing the Elicitation Burden and the Richness of Expert Input When Quantifying Discrete Bayesian Networks. *Risk Analysis*, 42(6):1196–1234.

Samuel Colvin, Eric Jolibois, and Hasan Ramezani. 2025. Pydantic/pydantic: Data validation using Python type hints.

John Dagdelen, Alexander Dunn, Sanghoon Lee, Nicholas Walker, Andrew S. Rosen, Gerbrand Ceder, Kristin A. Persson, and Anubhav Jain. 2024. Structured information extraction from scientific text with large language models. *Nature Communications*, 15(1):1418.

W. Edwards Deming and Frederick F. Stephan. 1940. On a Least Squares Adjustment of a Sampled Frequency Table When the Expected Marginal Totals are Known. *The Annals of Mathematical Statistics*, 11(4):427–444.

A. P. Dempster, N. M. Laird, and D. B. Rubin. 1977. Maximum Likelihood from Incomplete Data Via the EM Algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22.

Anthony O. Etyang, Sailoki Kapesa, Emily Odipo, Evasius Bauni, Catherine Kyobutungi, Marwah Abdalla, Paul Muntner, Solomon K. Musani, Alex Macharia, Thomas N. Williams, J. Kennedy Cruickshank, Liam Smeeth, and J. Anthony G. Scott. 2019. Effect of Previous Exposure to Malaria on Blood Pressure in Kilifi, Kenya: A Mendelian Randomization Study. *Journal of the American Heart Association: Cardiovascular and Cerebrovascular Disease*, 8(6):e011771.

Anthony O. Etyang, Sailoki Kapesa, Emily Odipo, Evasius Bauni, Catherine Kyobutungi, Marwah Abdalla, Paul Mutner, Solomon K. Musani, Alex Macharia, Thomas N. Williams, J Kennedy Cruickshank, Liam Smeeth, and J Anthony G. Scott. 2023. Data from: Effect of previous exposure to malaria on blood pressure in kilifi, kenya: A mendelian randomization study.

Sébastien Eustace. 2018. Poetry: Python packaging and dependency management made easy.

Nir Friedman and Daphne Koller. 2003. Being Bayesian About Network Structure. A Bayesian Approach to Structure Discovery in Bayesian Networks. *Machine Learning*, 50(1):95–125.

Gaël Gendron, Jože M. Rožanec, Michael Witbrock, and Gillian Dobbie. 2024. Counterfactual Causal Inference in Natural Language with Large Language Models. *Preprint*, arXiv:2410.06392.

Wenqiang Guo, Lei Hao, Jianwang Li, Yongyan Hou, Qinkun Xiao, Zixuan Huang, and Wei Li. 2022. A Novel Algorithm for Bayesian Network Parameter Learning with Informative Range Constraints and Sorting Model. In *2022 41st Chinese Control Conference (CCC)*, pages 4199–4204, Hefei, China. IEEE.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. 2009. *The Elements of Statistical Learning*, 2 edition. Springer New York, New York, US.

Martin Haugh. 2016. An Introduction to Copulas.

Lida Huang, Tao Chen, Qing Deng, and Yuli Zhou. 2023. Reasoning Disaster Chains with Bayesian Network Estimated Under Expert Prior Knowledge. *International Journal of Disaster Risk Science*, 14(6):1011–1028.

Margaret A Hull, Elizabeth A Nunamaker, and Penny S Reynolds. 2024. Effects of Refined Handling on Reproductive Indices of BALB/cJ and CD-1 IGS Mice. *Journal of the American Association for Laboratory Animal Science : JAALAS*, 63(1):3–9.

Thomas Jiralerspong, Xiaoyin Chen, Yash More, Vedant Shah, and Yoshua Bengio. 2024. Efficient Causal Graph Discovery Using Large Language Models. *Preprint*, arXiv:2402.01207.

Mark J. D. Jordans, Brandon A. Kohrt, Manaswi Sangraula, Elizabeth L. Turner, Xueqi Wang, Pragya Shrestha, Renasha Ghimire, Edith van't Hof, Richard A. Bryant, Katie S. Dawson, Kedar Marahatta, Nagendra P. Luitel, and Mark van Ommeren. 2021a. Data from: Effectiveness of Group Problem Management Plus, a brief psychological intervention for adults affected by humanitarian disasters in Nepal: A cluster randomized controlled trial.

Mark J. D. Jordans, Brandon A. Kohrt, Manaswi Sangraula, Elizabeth L. Turner, Xueqi Wang, Pragya Shrestha, Renasha Ghimire, Edith van't Hof, Richard A. Bryant, Katie S. Dawson, Kedar Marahatta, Nagendra P. Luitel, and Mark van Ommeren. 2021b. Effectiveness of Group Problem Management Plus, a brief psychological intervention for adults affected by humanitarian disasters in Nepal: A cluster randomized controlled trial. *PLOS Medicine*, 18(6):e1003621.

Christoph Kern, Unai Fischer-Abaigar, Jonas Schweisthal, Dennis Frauen, Rayid Ghani, Stefan Feuerriegel, Mihaela Van Der Schaar, and Frauke Kreuter. 2025. Algorithms for reliable decision-making need causal reasoning. *Nature Computational Science*, 5(5):356–360.

Elahe Khatibi, Mahyar Abbasian, Zhongqi Yang, Iman Azimi, and Amir M. Rahmani. 2024. ALCM: Autonomous LLM-Augmented Causal Discovery Framework. *Preprint*, arXiv:2405.01744.

Daphne Koller and Nir Friedman. 2010. *Probabilistic Graphical Models: Principles and Techniques*, nachdr. edition. Adaptive Computation and Machine Learning. MIT Press, Cambridge, Mass.

Stephanie Long, Alexandre Piché, Valentina Zantedeschi, Tibor Schuster, and Alexandre Drouin. 2023. Causal discovery with language models as imperfect experts. In *ICML 2023 Workshop on Structured Probabilistic Inference & Generative Modeling*.

Stephanie Long, Tibor Schuster, and Alexandre Piché. 2022. Can Large Language Models Build Causal Graphs? In *NeurIPS 2022 Workshop on Causality for Real-world Impact*.

Dorcas N. Magai, Michael Mwaniki, Amina Abubakar, Shebe Mohammed, Anne L. Gordon, Raphael Kalu, Paul Mwangi, Hans M. Koot, and Charles R. Newton. 2019a. Data from: A Randomized Control Trial of Phototherapy and 20\% Albumin Versus Phototherapy and Saline in Kilifi, Kenya.

Dorcas N. Magai, Michael Mwaniki, Amina Abubakar, Shebe Mohammed, Anne L. Gordon, Raphael Kalu, Paul Mwangi, Hans M. Koot, and Charles R. Newton. 2019b. A randomized control trial of phototherapy and 20% albumin versus phototherapy and saline in Kilifi, Kenya. *BMC Research Notes*, 12:617.

Mhairi Maskew, Sydney Rose, and Matthew Fox. 2020. Data from: Initiating Antiretroviral Therapy for HIV at a Patient's First Clinic Visit: The RapIT Randomized Controlled Trial.

Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. 1953. Equation of State Calculations by Fast Computing Machines. *The Journal of Chemical Physics*, 21(6):1087–1092.

Prabhaker Mishra, Chandra Mani Pandey, Uttam Singh, Amit Keshri, and Mayilvaganan Sabaretnam. 2019. Selection of Appropriate Statistical Methods for Data Analysis. *Annals of Cardiac Anaesthesia*, 22(3):297.

Aliakbar Nafar, Kristen Brent Venable, Zijun Cui, and Parisa Kordjamshidi. 2025. Extracting Probabilistic Knowledge from Large Language Models for Bayesian Network Parameterization. *Preprint*, arXiv:2505.15918.

Roger B. Nelsen. 2006. *An Introduction to Copulas*, second edition. Springer Series in Statistics. Springer, New York, NY.

R. Odobesku, K. Romanova, S. Mirzaeva, O. Zagorulko, R. Sim, R. Khakimullin, J. Razlivina, A. Dmitrenko, and V. Vinogradov. 2025. Agent-based multimodal information extraction for nanomaterials. *npj Computational Materials*, 11(1):194.

Judea Pearl. 2009. *Causality: Models, Reasoning, and Inference*, second edition, reprinted with corrections edition. Cambridge University Press, Cambridge New York, NY Port Melbourne New Delhi Singapore.

Kyle Peyton. 2020. Data from: Effect of physician gender and race on simulated patients' ratings and confidence in their physicians: A randomized clinical trial.

Friedrich Pukelsheim. 2014. Biproportional scaling of matrices and the iterative proportional fitting procedure. *Annals of Operations Research*, 215(1):269–283.

Penelope Reynolds. 2023. Data from: Effects of refined handling on reproductive indices of BALB/cJ and CD-1 IGS mice.

Sydney Rosen, Mhairi Maskew, Matthew P. Fox, Cynthia Nyoni, Constance Mongwenyana, Given Malete, Ian Sanne, Dorah Bokaba, Celeste Sauls, Julia Rohr, and Lawrence Long. 2016. Initiating Antiretroviral Therapy for HIV at a Patient's First Clinic Visit: The RapIT Randomized Controlled Trial. *PLOS Medicine*, 13(5):e1002015.

Stuart Jonathan Russell and Peter Norvig. 2021. *Artificial Intelligence – a Modern Approach, Global Edition*, 4 edition. Pearson Education. Prentice Hall, Harlow, United Kingdom.

Mahsa Shamsabadi, Jennifer D'Souza, and Sören Auer. 2024. Large Language Models for Scientific Information Extraction: An Empirical Study for Virology. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 374–392, St. Julian's, Malta. Association for Computational Linguistics.

Brandon Smock, Rohith Pesala, and Robin Abraham. 2022. PubTables-1M: Towards comprehensive table extraction from unstructured documents. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4624–4632, New Orleans, LA, USA. IEEE.

Rachel E. Solnick, Kyle Peyton, Gordon Kraft-Todd, and Basmah Safdar. 2020. Effect of Physician Gender and Race on Simulated Patients' Ratings and Confidence in Their Physicians: A Randomized Trial. *JAMA Network Open*, 3(2):e1920511.

Jussi Viinikka and Mikko Koivisto. 2020. Layering-MCMC for Structure Learning in Bayesian Networks. In *Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 839–848. PMLR.

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, and 15 others. 2020. SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods*, 17(3):261–272.

Guangya Wan, Yunsheng Lu, Yuqi Wu, Mengxuan Hu, and Sheng Li. 2025. Large Language Models for Causal Discovery: Current Landscape and Future Directions. *Preprint*, arXiv:2402.11068.

Galen Wei. 2025. Gmft.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2023. Can LLMs Express Their Uncertainty? An Empirical Evaluation of Confidence Elicitation in LLMs. In *The Twelfth International Conference on Learning Representations*.

Dong Xu. 2018. Data from: Lay health supporters aided by mobile text messaging to improve adherence, symptoms, and functioning among people with schizophrenia in a resource-poor community in rural China (LEAN): A randomized controlled trial.

Dong (Roman) Xu, Shuiyuan Xiao, Hua He, Eric D. Caine, Stephen Gloyd, Jane Simoni, James P. Hughes, Juan Nie, Meijuan Lin, Wenjun He, Yeqing Yuan, and Wenjie Gong. 2019. Lay health supporters aided by mobile text messaging to improve adherence, symptoms, and functioning among people with schizophrenia in a resource-poor community in rural China (LEAN): A randomized controlled trial. *PLOS Medicine*, 16(4):e1002785.

Fengxia Yan, Mayberry Robert, and Yonggang Li. 2017. Statistical methods and common problems in medical or biomedical science research. *International Journal of Physiology, Pathophysiology and Pharmacology*, 9(5):157–163.

Yuzhe Zhang, Yipeng Zhang, Yidong Gan, Lina Yao, and Chen Wang. 2024. Causal Graph Discovery with Retrieval-Augmented Generation based Large Language Models. *Preprint*, arXiv:2402.15301.

# A Additional Illustrations

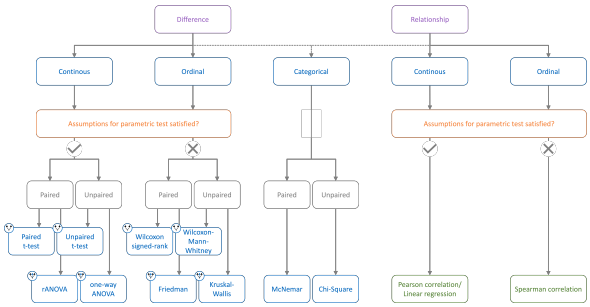


Figure 5: Overview of the statistical tests used in empirical scientific publications. Inspired by (Til, 2022).

| Parents | $P(Z=0)$ | $P(Z=1)$ |
|---|---|---|
| 00 | 0.90 | 0.10 |
| 01 | 0.60 | 0.40 |
| 10 | 0.50 | 0.50 |
| 11 | 0.05 | 0.95 |

Figure 6: Binary CPTs for fictional parent configurations.

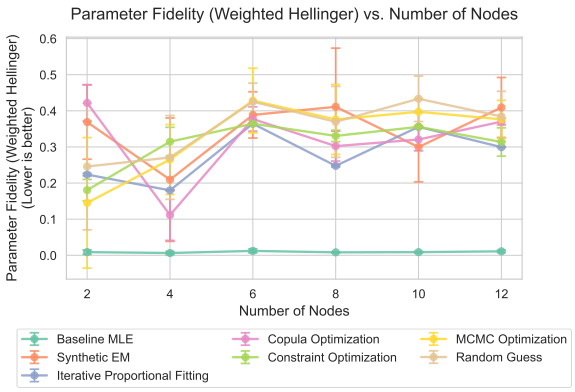


Figure 8: Change in W-Hell for increasing number of nodes in the BN.

# B Additional Plots

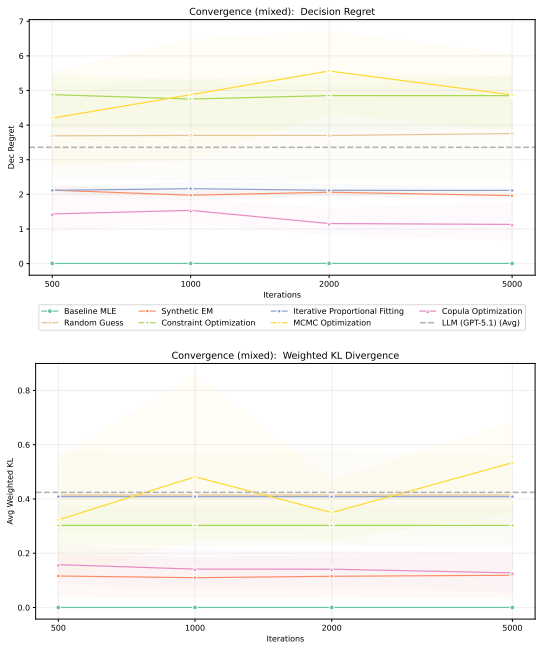


Figure 7: Convergence plots for Decision Regret and Average Weighted KL Divergence over iterations with (Confidence Interval) CI = 95 error bands.

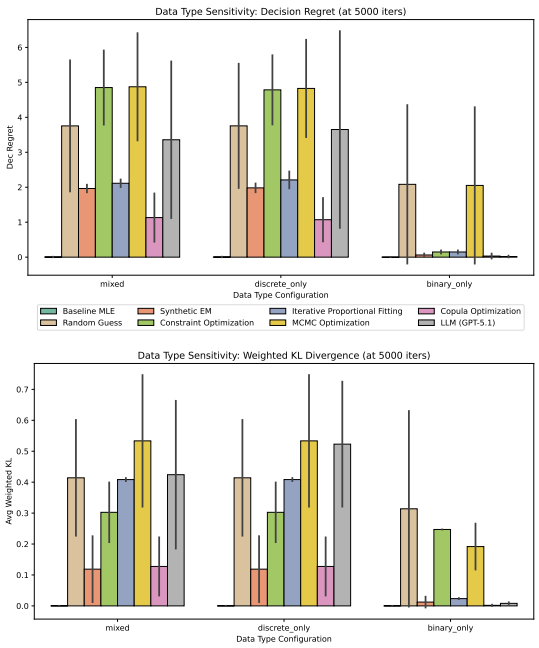


Figure 9: Sensitivity analysis of data types for Decision Regret and Average Weighted KL Divergence with increased iterations (5000) with CI = 95 error bars.

## C LLM Prompt Template

Listing 1: LLM prompt template for baselines.

```
prompt = f"""
You are an expert Bayesian Statistician. Your
    task is to estimate the Conditional
    Probability Table (CPT) for a node in a
    Bayesian Network based on statistical
    summary reports.

### CONTEXT
Target Node: '{node}'
States: {node_states}

Parents: {parents}
Parent States: {parent_states}

### STATISTICAL REPORTS
{relevant_reports}

### INSTRUCTIONS
1. Analyze the reports to determine
    correlations (positive, negative, or
    neutral).
2. Construct a probability distribution for
    '{node}' for **every possible combination**
    of parent states.
3. If a positive correlation exists (e.g.,
    Parent=High -> Child=High), assign higher
    probability to matching states.
4. If no clear correlation is reported, assume
    a uniform or weak prior.
5. Probabilities in each row must sum to 1.0.

### FORMAT EXAMPLE
{few_shot_text}
"""
```

## D Extraction Performance on Real-World Papers

| Category | Avg F1 | Avg Prec | Avg Rec |
|---|---|---|---|
| **Total** | 0.880 | 0.874 | 0.887 |
| Variable Names | 0.913 | 0.911 | 0.920 |
| Variable Content | 0.874 | 0.867 | 0.884 |
| Statistical Tests | 0.857 | 0.857 | 0.857 |

Table 2: Overall performance of extraction.

| DOI | Category | F1 | Prec | Rec |
|---|---|---|---|---|
| ...920511[1] | **Total** | 0.774 | 0.774 | 0.774 |
| | Variable Names | 0.800 | 0.769 | 0.833 |
| | Variable Content | 0.767 | 0.767 | 0.767 |
| | Statistical Tests | 1.000 | 1.000 | 1.000 |
| ...011771[2] | **Total** | 0.631 | 0.612 | 0.651 |
| | Variable Names | 0.875 | 0.933 | 0.824 |
| | Variable Content | 0.673 | 0.651 | 0.696 |
| | Statistical Tests | 0.000 | 0.000 | 0.000 |
| ...946322[3] | **Total** | 1.000 | 1.000 | 1.000 |
| | Variable Names | 1.000 | 1.000 | 1.000 |
| | Variable Content | 1.000 | 1.000 | 1.000 |
| | Statistical Tests | 1.000 | 1.000 | 1.000 |
| ...002015[4] | **Total** | 0.918 | 0.893 | 0.944 |
| | Variable Names | 0.906 | 0.857 | 0.960 |
| | Variable Content | 0.905 | 0.877 | 0.934 |
| | Statistical Tests | 1.000 | 1.000 | 1.000 |
| ...002785[5] | **Total** | 0.927 | 0.981 | 0.879 |
| | Variable Names | 0.917 | 1.000 | 0.846 |
| | Variable Content | 0.900 | 0.973 | 0.837 |
| | Statistical Tests | 1.000 | 1.000 | 1.000 |
| ...003621[6] | **Total** | 0.950 | 0.938 | 0.962 |
| | Variable Names | 0.936 | 0.898 | 0.978 |
| | Variable Content | 0.939 | 0.925 | 0.954 |
| | Statistical Tests | 1.000 | 1.000 | 1.000 |
| ...000028[7] | **Total** | 0.957 | 0.918 | 1.000 |
| | Variable Names | 0.957 | 0.917 | 1.000 |
| | Variable Content | 0.933 | 0.875 | 1.000 |
| | Statistical Tests | 1.000 | 1.000 | 1.000 |

Table 3: Detailed performance of extraction per DOI.

## E Additional Results

[1] (Solnick et al., 2020; Peyton, 2020)
[2] (Etyang et al., 2019, 2023)
[3] (Magai et al., 2019b,a)
[4] (Rosen et al., 2016; Maskew et al., 2020)
[5] (Xu et al., 2019; Xu, 2018)
[6] (Jordans et al., 2021b,a)
[7] (Hull et al., 2024; Reynolds, 2023)

| | Parameter Fidelity | | | | Inference Power | | | | Decision Utility | | Resource Usage |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Strategy** | **W-Hell** | **W-KL** | **TVD** | **JSON** | **LogLoss** | **Hell** | **KL** | **Acc** | **Regret** | **Acc** | **Time (s)** |
| Baseline MLE | 0.000 | 0.000 | 0.000 | 0.157 | 0.258 | 0.000 | 0.000 | 0.894 | 0.000 | 1.000 | 0.007 |
| | (0.00) | (0.00) | (0.00) | (0.00) | (0.03) | (0.00) | (0.00) | (0.02) | (0.00) | (0.00) | (0.00) |
| Copula Optimization | 0.249 | 0.351 | 0.311 | **0.015** | 0.529 | 0.190 | 0.245 | 0.756 | 1.953 | 0.765 | 31.271 |
| | (0.01) | (0.01) | (0.01) | (0.00) | (0.09) | (0.01) | (0.04) | (0.07) | (1.25) | (0.11) | (0.70) |
| Synthetic EM | **0.233** | 0.296 | 0.303 | **0.015** | **0.422** | **0.175** | **0.164** | **0.837** | 0.843 | 0.851 | 0.923 |
| | (0.01) | (0.01) | (0.01) | (0.00) | (0.02) | (0.01) | (0.03) | (0.01) | (0.11) | (0.01) | (0.03) |
| LLM Baseline | 0.234 | **0.273** | **0.281** | 0.036 | 0.443 | 0.202 | 0.187 | 0.686 | **0.818** | **0.853** | 23.691 |
| | (0.01) | (0.03) | (0.01) | (0.02) | (0.03) | (0.01) | (0.03) | (0.02) | (0.13) | (0.01) | (6.00) |
| Constraint Optimization | 0.413 | 0.559 | 0.416 | 0.107 | 0.832 | 0.451 | 0.574 | 0.696 | 11.494 | 0.085 | 0.002 |
| | (0.00) | (0.00) | (0.00) | (0.00) | (0.01) | (0.01) | (0.01) | (0.01) | (0.15) | (0.01) | (0.00) |
| MCMC Optimization | 0.482 | 1.048 | 0.492 | 0.216 | 1.202 | 0.477 | 0.949 | 0.487 | 5.891 | 0.514 | 13.972 |
| | (0.04) | (0.10) | (0.05) | (0.02) | (0.43) | (0.11) | (0.45) | (0.21) | (3.01) | (0.22) | (0.17) |
| Random Guess | 0.430 | 0.784 | 0.444 | 0.177 | 1.094 | 0.471 | 0.843 | 0.400 | 7.233 | 0.395 | **0.000** |
| | (0.06) | (0.12) | (0.03) | (0.03) | (0.47) | (0.15) | (0.49) | (0.37) | (5.22) | (0.39) | (0.00) |

Table 4: Real-World Benchmark Results (Mean ± Std) for . . . 946322.

| | Parameter Fidelity | | | | Inference Power | | | | Decision Utility | | Resource Usage |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Strategy** | **W-Hell** | **W-KL** | **TVD** | **JSON** | **LogLoss** | **Hell** | **KL** | **Acc** | **Regret** | **Acc** | **Time (s)** |
| Baseline MLE | 0.000 | 0.000 | 0.000 | 0.120 | 0.109 | 0.000 | 0.000 | 0.968 | 0.000 | 1.000 | 0.009 |
| | (0.00) | (0.00) | (0.00) | (0.00) | (0.01) | (0.00) | (0.00) | (0.01) | (0.00) | (0.00) | (0.00) |
| Synthetic EM | 0.342 | 0.434 | 0.325 | 0.017 | **0.645** | 0.444 | **0.544** | 0.808 | 6.727 | 0.473 | 1.183 |
| | (0.00) | (0.00) | (0.00) | (0.01) | (0.02) | (0.01) | (0.02) | (0.07) | (1.03) | (0.07) | (0.02) |
| Copula Optimization | 0.321 | 0.393 | 0.304 | 0.019 | 0.693 | 0.465 | 0.593 | 0.656 | 5.944 | 0.548 | 27.763 |
| | (0.01) | (0.01) | (0.01) | (0.01) | (0.02) | (0.01) | (0.02) | (0.07) | (1.54) | (0.12) | (0.25) |
| LLM Baseline | 0.318 | **0.376** | 0.300 | 0.060 | 0.680 | 0.457 | 0.578 | 0.608 | 8.091 | 0.376 | 33.455 |
| | (0.01) | (0.01) | (0.01) | (0.01) | (0.02) | (0.01) | (0.03) | (0.11) | (1.26) | (0.09) | (7.20) |
| Constraint Optimization | **0.309** | 0.381 | **0.294** | **0.013** | 0.693 | 0.467 | 0.592 | **0.845** | 11.622 | 0.122 | 0.000 |
| | (0.00) | (0.00) | (0.00) | (0.00) | (0.00) | (0.00) | (0.01) | (0.02) | (0.23) | (0.02) | (0.00) |
| MCMC Optimization | 0.373 | 0.605 | 0.370 | 0.155 | 0.718 | **0.398** | 0.617 | 0.616 | **5.249** | **0.603** | 7.719 |
| | (0.05) | (0.18) | (0.07) | (0.04) | (0.26) | (0.07) | (0.25) | (0.20) | (2.74) | (0.19) | (0.08) |
| Random Guess | 0.367 | 0.700 | 0.380 | 0.138 | 0.760 | 0.414 | 0.659 | 0.558 | 6.541 | 0.518 | **0.000** |
| | (0.05) | (0.23) | (0.04) | (0.02) | (0.17) | (0.07) | (0.16) | (0.16) | (2.49) | (0.18) | (0.00) |

Table 5: Real-World Benchmark Results (Mean ± Std) for . . . 011771.

| | Parameter Fidelity | | | | Inference Power | | | | Decision Utility | | Resource Usage |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Strategy** | **W-Hell** | **W-KL** | **TVD** | **JSON** | **LogLoss** | **Hell** | **KL** | **Acc** | **Regret** | **Acc** | **Time (s)** |
| Baseline MLE | 0.000 | 0.000 | 0.000 | 0.055 | 0.464 | 0.000 | 0.000 | 0.805 | 0.000 | 1.000 | 0.009 |
| | (0.00) | (0.00) | (0.00) | (0.00) | (0.04) | (0.00) | (0.00) | (0.02) | (0.00) | (0.00) | (0.00) |
| Copula Optimization | 0.132 | 0.224 | 0.277 | **0.012** | 0.611 | 0.145 | 0.153 | 0.676 | 1.005 | 0.826 | 34.196 |
| | (0.00) | (0.06) | (0.01) | (0.00) | (0.02) | (0.01) | (0.02) | (0.05) | (0.63) | (0.13) | (0.17) |
| Synthetic EM | **0.122** | 0.201 | 0.268 | 0.055 | 0.761 | 0.255 | 0.302 | 0.739 | 3.901 | 0.688 | 4.181 |
| | (0.01) | (0.09) | (0.01) | (0.00) | (0.02) | (0.01) | (0.02) | (0.05) | (0.64) | (0.09) | (0.09) |
| LLM Baseline | 0.115 | **0.086** | **0.220** | 0.035 | **0.567** | **0.137** | **0.108** | **0.780** | **0.039** | **0.997** | 25.275 |
| | (0.00) | (0.01) | (0.01) | (0.01) | (0.02) | (0.01) | (0.01) | (0.02) | (0.03) | (0.00) | (2.00) |
| Constraint Optimization | 0.308 | 0.363 | 0.350 | 0.246 | 0.894 | 0.349 | 0.431 | 0.656 | 7.994 | 0.144 | 0.001 |
| | (0.00) | (0.00) | (0.00) | (0.00) | (0.01) | (0.01) | (0.01) | (0.02) | (0.10) | (0.02) | (0.00) |
| MCMC Optimization | 0.362 | 0.646 | 0.414 | 0.313 | 1.475 | 0.447 | 1.015 | 0.326 | 6.817 | 0.338 | 10.605 |
| | (0.04) | (0.20) | (0.06) | (0.04) | (0.46) | (0.08) | (0.47) | (0.13) | (1.86) | (0.16) | (0.05) |
| Random Guess | 0.337 | 0.543 | 0.396 | 0.356 | 1.419 | 0.430 | 0.943 | 0.320 | 6.903 | 0.334 | **0.000** |
| | (0.06) | (0.20) | (0.06) | (0.09) | (0.42) | (0.08) | (0.44) | (0.13) | (1.79) | (0.17) | (0.00) |

Table 6: Real-World Benchmark Results (Mean ± Std) for . . . 002785.