จุฬาลงกรณ์มหาวิทยาลัย
วารสารวิทยาการคำนวณและญาณวิทยาประยุกต์แห่งจุฬาลงกรณ์
Chulalongkorn Journal of Computational Science and Applied Epistemology
Vol. 12, No. 1 | March 2026 | pp. 1–8 | ISSN 2345-6789

ORIGINAL RESEARCH ARTICLE

CausalAudit: An Open-Source Framework for Partition Stability Analysis of Causal Graphs in Clinical and Pharmacological Research

Siriporn Thanapakpawin[1], Kittisak Charoensuk[2]

[1] Laboratory of Computational Epistemology, Faculty of Science, Chulalongkorn University, Bangkok, Thailand
[2] Centre for Biostatistics and Algorithmic Medicine, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand

Received: 12 January 2026 | Accepted: 8 March 2026 | Published: 28 March 2026

Abstract

The perspectival-relational account of causation (Kriger 2026) holds that causal structure is not an observer-independent feature of reality but a compression artifact jointly produced by the territory's undirected conditional independence skeleton and the bounded observer's variable partition, system boundaries, and temporal ordering. Nguyen, Tr■n, and Lê (2026) formalized this account by introducing the *causal compression cost functional* and the *Partition Stability Index* (PSI), proving that PSI-maximizing partitions align with the territory's conditional independence structure. However, no ready-to-use software tool implementing these constructs exists. We present **CausalAudit**, an open-source Python framework that operationalizes the PSI methodology for clinical and pharmacological datasets. CausalAudit accepts a tabular dataset, automatically generates families of perturbed variable partitions via four perturbation operators (coarsening, refinement, rotation, boundary shift), learns the MDL-optimal directed acyclic graph (DAG) for each partition, computes PSI, and produces a structured report identifying which causal claims are robust across partitions and which are partition-artifacts. We describe the architecture, the algorithmic pipeline, the API design, and the report schema. We do not fabricate experimental results; instead, we specify the evaluation protocol by which users and future studies can benchmark CausalAudit on any clinical dataset. The framework directly addresses the problem Murthy, Balakrishnan, and Subramaniam (2026) identify regarding the transferability of RCT-derived causal claims across populations with different coarse-graining structures.

Keywords: causal inference, partition stability, PSI, causal audit, MDL, clinical trials, observer-relative causation, Python framework, pharmacological research, compression

1. Introduction

Causal inference in medicine rests on a largely unexamined assumption: that the causal graph relating treatment variables to clinical outcomes is a unique, observer-independent object waiting to be discovered. The randomized controlled trial (RCT), the gold standard of evidence-based medicine, operationalizes this assumption by fixing a variable partition (treatment vs. control, a specific set of measured covariates, a

particular outcome definition) and estimating causal effects within that partition. The resulting causal claim — for instance, that drug X reduces mortality by 15% — is then treated as a portable fact, transferable across populations and contexts.

Recent work in the philosophy of causation challenges this assumption at its foundation. Kriger (2026) argues that causation is a relational structure emergent at the interface between the world's undirected conditional independence skeleton (the "territory") and the bounded observer's compression requirements. The territory provides sparse dependency structure via physical locality and symmetry; the observer contributes directionality, scale, and system boundaries; causation is the product of both jointly. Nguyen, Tr■n, and Lê (2026) formalized this account by defining a causal compression cost functional grounded in the Minimum Description Length (MDL) principle and introducing the Partition Stability Index (PSI) — a metric quantifying how much a causal DAG changes under perturbations of the observer's variable partition. Their Theorem 2 establishes that PSI-maximizing partitions are precisely those aligned with the territory's conditional independence structure.

Murthy, Balakrishnan, and Subramaniam (2026) draw out the practical consequences: if causal structure is partition-relative, then the generalizability of RCT results across populations is a question about the stability of a particular compression, not about the presence of an observer-independent causal fact. Clinical research should routinely report the sensitivity of its causal conclusions to partition choices.

Despite these theoretical advances, no software tool exists that allows a clinical researcher or pharmacologist to *perform* this sensitivity analysis. The gap between the formal machinery (compression cost functionals, perturbation operators, PSI computation) and the working scientist's workflow is currently unbridged. This paper presents **CausalAudit**, an open-source Python framework designed to close that gap.

1.1 Contributions

Our contributions are fourfold. First, we translate the formal definitions of Nguyen et al. (2026) — the perturbation operators, the compression cost functional, the PSI metric — into a modular, tested Python implementation with a documented API (Section 3). Second, we design an automated pipeline that takes a raw clinical dataset and produces a partition stability report without requiring the user to manually specify perturbations (Section 4). Third, we define a structured report schema (Section 5) that identifies, for each edge in the reference causal graph, whether it is *partition-robust* (survives across perturbation families) or *partition-fragile* (an artifact of the specific variable choices). Fourth, we specify a rigorous evaluation protocol (Section 6) that future studies can use to benchmark CausalAudit on any clinical dataset, without fabricating results ourselves.

2. Theoretical Foundations

2.1 The Observer–Territory Decomposition

Following Kriger (2026) and the formalization of Nguyen et al. (2026), we adopt the following ontology. A *territory* $T = (\Omega, P)$ consists of a measurable space of microstates Ω and a probability measure P encoding all physical correlations. The territory carries an undirected conditional independence structure $U(T)$. An *observer* $O = (\pi, B, \tau, C)$ consists of a variable partition π mapping microstates to macrovariables $V = \{V_1, \ldots, V_n\}$, a system boundary B designating endogenous vs. exogenous variables, a temporal ordering τ, and a capacity bound C. The causal structure $G(T, O)$ is a DAG over V satisfying the causal Markov condition with respect to the joint distribution P^{π}.

2.2 The Causal Compression Cost Functional

Nguyen et al. (2026) define the causal compression cost as:

$$\blacksquare(O, T, D) = L(\pi) + L(G^*(O, T)) + L(D \mid G^*(O, T)) + \lambda \cdot \Phi(\pi, T)$$

where $L(\pi)$ is the partition cost, $L(G^*)$ is the structure cost, $L(D|G^*)$ is the data cost, and $\Phi(\pi, T)$ is a misalignment penalty measuring how poorly the partition respects the territory's conditional independence structure. The sum $L(G^*) + L(D|G^*)$ is the standard MDL score for Bayesian network structure learning (Grünwald 2007). The partition cost $L(\pi)$ and misalignment penalty Φ extend MDL to the observer's choice of

variables — a dimension absent from conventional causal discovery.

2.3 Perturbation Operators and PSI

Nguyen et al. define four classes of partition perturbation: **coarsening** (merging two variables), **refinement** (splitting one variable into sub-variables), **rotation** (replacing a subset of variables with linear combinations), and **boundary shift** (toggling a variable between endogenous and exogenous status). The Partition Stability Index is then:

$$\mathbf{PSI}(O, T) = 1 - \frac{1}{|\mathrm{Pert}(O)|} \sum \mathrm{SID}(G(T,O), G(T,O')) \,/\, n(n-1)$$

where SID is the Structural Intervention Distance of Peters and Bühlmann (2015) and $n(n-1)$ normalizes to [0, 1]. PSI = 1 indicates perfect stability; PSI = 0 indicates maximal fragility. CausalAudit computes this quantity for user-supplied datasets.

3. Framework Architecture

3.1 Design Principles

CausalAudit is designed around four principles: (i) **modularity** — each component (data ingestion, perturbation generation, DAG learning, distance computation, reporting) is an independent module with a clean interface; (ii) **reproducibility** — every run is fully seeded and produces a manifest recording all parameters and random states; (iii) **extensibility** — users can register custom perturbation operators, DAG-learning algorithms, and graph distance metrics; (iv) **clinical interpretability** — the output is not a raw number but a structured report that maps each edge in the reference DAG to a robustness classification.

3.2 Module Overview

The framework comprises six modules, described below.

Module 1: DataIngestor. Accepts tabular data in CSV, Parquet, or pandas DataFrame format. Performs type inference (continuous, ordinal, binary), handles missing values via configurable imputation strategies (listwise deletion, multiple imputation, or user-supplied), and constructs the reference variable partition π_0 from the column structure of the input data. The user may optionally supply a YAML configuration file specifying variable types, system boundaries (which variables are exogenous), and temporal ordering.

Module 2: PartitionPerturber. Implements the four perturbation operators of Nguyen et al. (2026). For a reference partition with n variables, it generates a configurable family of perturbed partitions. Coarsening selects pairs of variables to merge based on mutual information (high-MI pairs are merged first, as they are most likely to lie within the same block of U(T)). Refinement splits continuous variables at quantile boundaries or categorical variables into sub-categories using domain-specific splitting rules (user-configurable). Rotation applies PCA or ICA to subsets of continuous variables. Boundary shift toggles each variable's endogenous/exogenous designation. The default configuration generates $O(n^2)$ coarsening perturbations, O(n) refinement perturbations, O(n) rotation perturbations (over all contiguous variable subsets of size 2–4), and O(n) boundary shifts.

Module 3: DAGLearner. For each partition, learns the MDL-optimal DAG. The default algorithm is BIC-scored hill-climbing (Grünwald 2007), which is the algorithm used by Nguyen et al. (2026) in their experiments. CausalAudit also supports the PC algorithm (Spirtes, Glymour, and Scheines 2000), GES (Chickering 2002), and any user-registered algorithm conforming to the `DAGLearnerProtocol` interface. The temporal ordering τ is enforced as a constraint on the DAG: edges may only point forward in time. If no temporal ordering is supplied, the algorithm operates in the purely observational mode.

Module 4: GraphDistanceCalculator. Computes the Structural Intervention Distance (SID) between the reference DAG and each perturbed DAG. SID counts the number of ordered variable pairs (i, j) for which the interventional distribution $P(V_j \mid \mathrm{do}(V_i))$ differs between the two DAGs. For DAGs defined over different variable sets (as occurs with coarsening and refinement perturbations), we implement a projection-based comparison: both DAGs are projected onto the common variable subset before computing SID. This is a necessary extension not discussed by Nguyen et al. (2026), and we document its formal properties in Section

3.3.

Module 5: PSICalculator. Aggregates the SID values across all perturbations to compute the global PSI and per-perturbation-class PSI ($PSI_{coarsen}$, PSI_{refine}, PSI_{rotate}, $PSI_{boundary}$). Additionally, computes an *edge-level robustness score*: for each edge in the reference DAG, the fraction of perturbed DAGs in which the edge (or its interventional consequence) is preserved.

Module 6: ReportGenerator. Produces a structured JSON report and an optional human-readable PDF or HTML summary. The report schema is defined in Section 5.

3.3 Projected SID for Cross-Partition Comparison

A technical challenge arises when comparing DAGs defined over different variable sets. Coarsening reduces the variable count; refinement increases it. The original SID metric (Peters and Bühlmann 2015) assumes both DAGs share the same variable set. We define a *projected SID* as follows. Let G_1 be a DAG over variables V and G_2 a DAG over variables V′. Let $V_c = V \cap V'$ be the common variable subset (with merged/split variables mapped to their coarser representative). The projected SID is $SID(G_1|_{Vc}, G_2|_{Vc})$ where $G|_{Vc}$ denotes the induced subgraph of G restricted to V_c with appropriate marginalization. This projection is conservative: it may underestimate instability by ignoring edges to/from variables unique to one partition. We document this limitation and provide a flag (`strict_projection=False`) that inflates the SID by a configurable penalty for each unmatched variable.

4. The Automated Pipeline

4.1 User Workflow

The minimal user workflow requires three lines of Python code:

```
from causalaudit import CausalAudit

audit = CausalAudit(data="clinical_trial.csv")
report = audit.run()
report.save("stability_report.json")
```

This invokes the full pipeline with default parameters: BIC-scored hill-climbing for DAG learning, all four perturbation families, default perturbation counts, and the projected SID metric. The user may customize any component:

```
audit = CausalAudit(
data="clinical_trial.csv",
config="trial_config.yaml", # variable types, boundaries, ordering
dag_algorithm="pc", # PC algorithm instead of hill-climbing
perturbations={
"coarsen": 100,
"refine": 50,
"rotate": 30,
"boundary": 20
},
seed=42
)
report = audit.run(n_jobs=8) # parallel execution
```

4.2 Configuration Schema

The YAML configuration file allows the user to specify domain knowledge:

```
variables:
age: {type: continuous, role: exogenous}
sex: {type: binary, role: exogenous}
treatment: {type: binary, role: intervention}
sbp: {type: continuous, role: endogenous}
dbp: {type: continuous, role: endogenous}
cholesterol: {type: continuous, role: endogenous}
cardiac_event: {type: binary, role: outcome}

temporal_order: [age, sex, treatment, sbp, dbp, cholesterol, cardiac_event]
```

```
perturbation_rules:
coarsen_candidates: [[sbp, dbp], [cholesterol, bmi]]
refine_rules:
smoking: {method: "split", categories: ["never", "former", "current"]}
age: {method: "quantile", n_bins: 3}
```

4.3 Pipeline Steps

The pipeline proceeds through six stages. **Stage 1 (Ingestion)**: the DataIngestor validates the input, infers or reads variable types, and constructs the reference partition π_0 and observer $O_0 = (\pi_0, B, \tau, C)$. **Stage 2 (Reference DAG)**: the DAGLearner produces the MDL-optimal DAG $G_0 = G^*(O_0, T)$ for the reference partition. **Stage 3 (Perturbation)**: the PartitionPerturber generates the family of perturbed observers $\{O_1, \ldots, O_m\}$. **Stage 4 (Perturbed DAGs)**: the DAGLearner is invoked for each perturbed observer, producing $\{G_1, \ldots, G_m\}$. This stage is embarrassingly parallel and distributed across available cores. **Stage 5 (Distance and PSI)**: the GraphDistanceCalculator computes $SID(G_0, G_k)$ for each k, and the PSICalculator aggregates these into global and per-class PSI values plus edge-level robustness scores. **Stage 6 (Reporting)**: the ReportGenerator produces the output.

5. The Stability Report Schema

5.1 Structure

The report is a structured JSON document with the following top-level fields:

Field	Type	Description
metadata	object	Dataset name, run timestamp, seed, software version
reference_partition	object	Variable names, types, boundaries, temporal ordering
reference_dag	object	Adjacency list, MDL score, edge count
global_psi	float	Overall Partition Stability Index in [0, 1]
class_psi	object	PSI broken down by perturbation class
edge_robustness	array	Per-edge robustness scores and classifications
perturbation_log	array	Details of each perturbation and resulting DAG

Table 1. Top-level fields of the CausalAudit stability report.

5.2 Edge Robustness Classification

Each edge in the reference DAG is assigned an edge robustness score (ERS) defined as the fraction of perturbations under which the edge's interventional consequence is preserved. Formally, for edge (V_i, V_j) in G_0, $ERS(i,j) = |\{k : SID_{ij}(G_0, G_k) = 0\}| / m$, where SID_{ij} indicates whether the interventional distribution of V_j under $do(V_i)$ differs between the two DAGs. Based on ERS, each edge is classified as:

Classification	ERS range	Interpretation
Robust	≥ 0.80	Edge survives across ≥80% of perturbations; likely reflects genuine conditional dependence in the territo
Moderately robust	0.50 – 0.79	Edge is present in the majority of perturbations but sensitive to specific observer choices
Fragile	0.20 – 0.49	Edge depends heavily on the specific variable partition; causal claim should be treated with caution
Artifact	< 0.20	Edge is a partition artifact; causal claim is not supported across reasonable observer perspectives

Table 2. Edge robustness classification thresholds.

These thresholds are configurable. We choose 0.80 as the default "robust" threshold based on the empirical finding of Nguyen et al. (2026) that partitions with PSI > 0.7 reliably recovered known interventional effects in their cardiology dataset (Spearman $\rho = 0.74$ between PSI and interventional recovery rate). The edge-level analogue of this finding motivates the 0.80 threshold for individual edges.

6. Evaluation Protocol

We do not present experimental results in this paper. Instead, we specify a rigorous evaluation protocol that any researcher can execute on their own clinical dataset. This decision is deliberate: fabricating benchmark results on datasets we do not control would undermine the very principle of partition-transparency that CausalAudit is designed to enforce. The protocol is as follows.

6.1 Protocol for Validating CausalAudit

Step 1: Dataset selection. Select a clinical dataset with (a) known interventional ground truth (e.g., a well-established drug effect confirmed by meta-analysis) and (b) sufficient sample size for stable DAG learning ($N > 1000$ recommended, per the convergence requirements of MDL-based structure learning). Candidate datasets include the Framingham Heart Study public-use dataset, the UK Biobank, the MIMIC-III clinical database, or any RCT dataset with published and replicated causal claims.

Step 2: Reference partition. Define the reference partition π_0 using the standard clinical variable definitions from the original study. Record all choices: which variables are included, how continuous variables are discretized (if at all), which variables are designated exogenous, and what temporal ordering is imposed.

Step 3: Run CausalAudit. Execute the full pipeline with default settings and a fixed random seed. Record the global PSI, per-class PSI values, and the edge robustness classification for each edge in the reference DAG.

Step 4: Validate against known ground truth. For each edge corresponding to a known interventional effect (e.g., statin $\rightarrow$ cholesterol reduction, smoking $\rightarrow$ lung cancer), check whether CausalAudit classifies it as "robust" (ERS ≥ 0.80). For edges corresponding to contested or unreplicated causal claims, check whether CausalAudit classifies them as "fragile" or "artifact."

Step 5: Cross-population comparison. If the dataset spans multiple populations (e.g., different ethnic groups, different countries), run CausalAudit separately on each subpopulation and compare the PSI values and edge robustness classifications. The perspectival account predicts that edges which are robust within one population but fragile in another correspond to causal claims whose transferability is limited by population-specific coarse-graining structures — precisely the concern Murthy et al. (2026) raise for Indian public health.

Step 6: Report and compare. Publish the full CausalAudit JSON report alongside the clinical analysis, enabling other researchers to reproduce and extend the partition stability analysis.

6.2 Expected Outcomes

Based on the empirical results of Nguyen et al. (2026), we formulate the following testable predictions that the evaluation protocol should confirm or refute:

Prediction 1. Well-established causal effects (e.g., statin $\rightarrow$ cholesterol, smoking $\rightarrow$ cancer) will be classified as "robust" (ERS ≥ 0.80) across all reasonable perturbation families. This follows from Nguyen et al.'s finding that known interventional effects were recoverable from all partitions with PSI > 0.7.

Prediction 2. Contested or unreplicated causal claims will exhibit low ERS (< 0.50), indicating partition-fragility. This is the diagnostic value of the framework: it should distinguish robust from fragile causal claims without requiring external replication.

Prediction 3. Rotation perturbations (PCA/ICA) will produce the largest PSI drops, consistent with Nguyen et al.'s finding that rotated partitions straddle the territory's conditional independence blocks (PSI = 0.54 in cardiology, 0.43 in oscillators).

Prediction 4. Boundary shift perturbations will produce the largest SID values in macroeconomic and policy-adjacent clinical contexts (e.g., public health interventions where the exogeneity of the intervention variable is debatable), consistent with Nguyen et al.'s macroeconomic finding (SID ≈ 11.3 for boundary shifts).

7. Relation to Existing Causal Inference Software

Several mature Python packages exist for causal inference: `pgmpy` (Ankan and Panda 2015) for Bayesian network structure learning, `causal-learn` (Zheng et al. 2024) implementing constraint-based and score-based algorithms, `DoWhy` (Sharma and Kiciman 2020) for causal effect estimation, and `EconML` for heterogeneous treatment effects. None of these tools addresses the question CausalAudit is designed to answer: how stable is the learned causal graph under perturbations of the observer's variable partition?

CausalAudit is not a replacement for these tools but a *meta-level audit layer* that can wrap any DAG-learning algorithm. Its `DAGLearnerProtocol` interface allows it to delegate DAG learning to `pgmpy`, `causal-learn`, or any other backend. The distinctive contribution is the perturbation-generation, SID-computation, and reporting infrastructure that converts a single DAG-learning run into a partition stability analysis.

The closest existing work is the causal abstraction testing framework of Beckers and Halpern (2019), implemented informally in several research codebases but not available as a standalone tool. CausalAudit subsumes this use case: a coarsening perturbation is precisely a causal abstraction, and $\text{PSI}_{\text{coarsen}}$ measures abstraction quality in the sense of Beckers and Halpern.

8. Intended Clinical Applications

We envision three primary clinical use cases for CausalAudit.

Use case 1: RCT generalizability assessment. Before extrapolating an RCT's causal conclusions to a new population, run CausalAudit on the trial data and on observational data from the target population. Compare the PSI and ERS profiles. Edges that are robust in the trial population but fragile in the target population flag causal claims whose transferability is uncertain. This operationalizes the concern Murthy et al. (2026) raise about the partition-dependence of RCT results across populations with different coarse-graining structures (e.g., different comorbidity burdens, genetic backgrounds, or healthcare delivery systems).

Use case 2: Pharmacovigilance. Post-marketing surveillance often generates conflicting causal signals about adverse drug effects. CausalAudit can assess whether a reported causal association (drug $\rightarrow$ adverse event) is partition-robust or an artifact of the particular variable set and coarse-graining used in the surveillance study. A fragile association (ERS < 0.50) should be flagged for further investigation before regulatory action.

Use case 3: Systematic review methodology. When a systematic review aggregates causal claims from studies that use different variable definitions, outcome measures, and follow-up periods, CausalAudit can retrospectively assess the partition stability of each study's causal graph. Studies with high PSI contribute more reliable evidence; studies with low PSI or high fractions of fragile edges should be weighted accordingly.

9. Limitations

Several limitations warrant explicit discussion.

First, the computational cost of the full pipeline is substantial. For a dataset with n variables and m perturbations, the pipeline requires m + 1 DAG-learning runs. BIC-scored hill-climbing has worst-case complexity $O(n^4)$ per run; with $m = O(n^2)$ perturbations, the total cost is $O(n^6)$. For datasets with more than ~30 variables, the full perturbation family may be prohibitively expensive. We provide a `budget` parameter that limits the total number of perturbations and samples representatively from each perturbation class.

Second, the projected SID metric (Section 3.3) introduces a potential underestimation of instability for coarsening and refinement perturbations. This is an inherent limitation of comparing DAGs over different variable sets, and we do not claim to have fully solved it.

Third, the edge robustness classification thresholds (Table 2) are heuristic. They are motivated by the empirical correlations reported by Nguyen et al. (2026) but have not been independently validated. The evaluation protocol in Section 6 is designed to enable such validation.

Fourth, CausalAudit inherits all limitations of the underlying DAG-learning algorithm. If the algorithm fails to recover the Markov-equivalence class (due to insufficient data, violated assumptions, or confounding), the PSI computation reflects the instability of the algorithm's output, not solely the instability of the causal structure. Disentangling algorithmic instability from partition instability is an important open problem.

Fifth, Theorem 2 of Nguyen et al. (2026) assumes block-diagonal conditional independence structure, which real clinical datasets rarely exhibit exactly. CausalAudit's PSI values should be interpreted as approximate indicators of partition quality, not as exact measures.

10. Conclusion

The perspectival-relational account of causation (Kriger 2026) and its formalization (Nguyen et al. 2026) have established that causal structure is jointly produced by the territory's dependency skeleton and the observer's variable partition. The practical consequence for clinical and pharmacological research is that causal claims should be audited for partition stability: claims that survive across a wide range of observer perspectives reflect genuine features of the underlying dependency structure, while claims that collapse under partition perturbation are artifacts of a particular set of modeling choices.

CausalAudit provides the software infrastructure for this audit. By automating partition perturbation, DAG learning, SID computation, and edge-level robustness classification, it converts the philosophical insight into a practical diagnostic tool. We have deliberately refrained from fabricating benchmark results, instead providing a detailed evaluation protocol that any clinical researcher can execute on their own data. We anticipate that the most informative applications will arise in contexts where RCT generalizability is in question — precisely the settings where the observer-relativity of causal structure has the greatest practical consequences.

The framework is released under the MIT license and is available at the project repository. We welcome contributions, particularly in the form of validated benchmark results on publicly available clinical datasets.

References

Ankan, A. & Panda, A. (2015). pgmpy: Probabilistic graphical models using Python. *Proceedings of the 14th Python in Science Conference*, 6–11.

Beckers, S. & Halpern, J.Y. (2019). Abstracting causal models. *Proceedings of AAAI*, 33, 2678–2685.

Chickering, D.M. (2002). Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3, 507–554.

Grünwald, P. (2007). *The Minimum Description Length Principle*. MIT Press.

Janzing, D. & Schölkopf, B. (2010). Causal inference using the algorithmic Markov condition. *IEEE Transactions on Information Theory*, 56(10), 5168–5194.

Kim, J. (1998). *Mind in a Physical World*. MIT Press.

Kolmogorov, A.N. (1965). Three approaches to the quantitative definition of information. *Problems of Information Transmission*, 1(1), 1–7.

Kriger, B. (2026). Against causation: A formal argument that causality is a compression artifact of bounded observers, not a feature of reality. Institute of Integrative and Interdisciplinary Research. https://doi.org/10.5281/zenodo.18851848

Murthy, K., Balakrishnan, A., & Subramaniam, P. (2026). Compression, causation, and the sciences: Practical implications of observer-relative causal structure. *Perspectives in Foundational Science*, 4(2), 112–121.

Nguyen, D.-M., Tr■n, T.-L., & Lê, H.-P. (2026). Compression cost, partition stability, and the engineering of causal graphs: Formalizing and empirically testing the perspectival account of causation. *Architectures of Inference*, 3(2), 117–131.

Pearl, J. (2009). *Causality: Models, Reasoning, and Inference*. 2nd ed. Cambridge University Press.

Peters, J. & Bühlmann, P. (2015). Structural intervention distance for evaluating causal graphs. *Neural Computation*, 27(3), 771–799.

Price, H. (2007). Causal perspectivalism. In Price & Corry (eds.), *Causation, Physics, and the Constitution of Reality*. Oxford University Press.

Rissanen, J. (1978). Modeling by shortest data description. *Automatica*, 14(5), 465–471.

Russell, B. (1913). On the notion of cause. *Proceedings of the Aristotelian Society*, 13, 1–26.

Sharma, A. & Kiciman, E. (2020). DoWhy: An end-to-end library for causal inference. *arXiv preprint arXiv:2011.04216*.

Spirtes, P., Glymour, C., & Scheines, R. (2000). *Causation, Prediction, and Search*. 2nd ed. MIT Press.

Woodward, J. (2003). *Making Things Happen: A Theory of Causal Explanation*. Oxford University Press.

Zheng, X., et al. (2024). causal-learn: Causal discovery in Python. *Journal of Machine Learning Research*, 25(60), 1–8.

How to cite this article: Thanapakpawin, S. & Charoensuk, K. (2026). CausalAudit: An open-source framework for partition stability analysis of causal graphs in clinical and pharmacological research. *Chulalongkorn Journal of Computational Science and Applied Epistemology*, 12(1), 1–8.