Local Discovery by Partitioning: Polynomial-Time Causal Discovery Around Exposure-Outcome Pairs

Jacqueline R. M. A. Maasch
Cornell Tech

Weishen Pan
Weill Cornell Medicine

Shantanu Gupta
Carnegie Mellon University

Volodymyr Kuleshov
Cornell Tech

Kyra Gan
Cornell Tech

Fei Wang
Weill Cornell Medicine

Abstract

This work addresses the problem of automated covariate selection under limited prior knowledge. Given an exposure-outcome pair $\{X, Y\}$ and a variable set $\mathbf{Z}$ of unknown causal structure, the *Local Discovery by Partitioning* (LDP) algorithm partitions $\mathbf{Z}$ into subsets defined by their relation to $\{X, Y\}$. We enumerate eight exhaustive and mutually exclusive partitions of any arbitrary $\mathbf{Z}$ and leverage this taxonomy to differentiate confounders from other variable types. LDP is motivated by *valid adjustment set* identification, but avoids the pretreatment assumption commonly made by automated covariate selection methods. We provide theoretical guarantees that LDP returns a valid adjustment set for any $\mathbf{Z}$ that meets sufficient graphical conditions. Under stronger conditions, we prove that partition labels are asymptotically correct. Total independence tests is worst-case quadratic in $|\mathbf{Z}|$, with sub-quadratic runtimes observed empirically. We numerically validate our theoretical guarantees on synthetic and semi-synthetic graphs. Adjustment sets from LDP yield less biased and more precise average treatment effect estimates than baselines, with LDP outperforming on confounder recall, test count, and runtime for valid adjustment set discovery.

1 INTRODUCTION

Covariate selection is a central task in the design of observational studies (Guo et al., 2022). The objectives of covariate selection include eliminating bias and reducing variance in causal estimates, improving model fitting and interpretability through dimensionality reduction (Schnitzer et al., 2016), and increasing robustness to model misspecification (Guo et al., 2022). The primary goal of covariate selection is to obtain a *valid adjustment set* for an exposure-outcome pair that eliminates confounding bias by adjusting for confounders (Witte & Didelez, 2019). Confounding bias distorts the observed relationship between the exposure and outcome, leading to incorrect effect measures even under infinite data (Hernán & Robins, 2020).

Identifying valid adjustment sets is a challenging task in many real-world settings, even with the guidance of domain experts. A naive approach is to adjust for all measured variables. However, it is established that multiple variable types can induce bias when retained for adjustment (Lu et al., 2021; Schisterman et al., 2009). These include colliders, which induce selection bias (Hernán et al., 2004; Elwert & Winship, 2014; Holmberg & Andersen, 2022); mediators, which bias total effects by controlling for indirect effects (Pearl, 2001); and instrumental variables, which can amplify existing bias or introduce new bias in some settings (Pearl, 2012). Unnecessary adjustment (Schisterman et al., 2009) may increase the variance of causal effect estimates or undermine model fitting due to the curse of dimensionality (Schnitzer et al., 2016).

Accepted to the NeurIPS 2023 Workshop on Causal Representation Learning.

Data-driven approaches can automate the principled selection of covariates when manual identification is not feasible. Causal discovery can enable automated covariate selection by inferring the causal graphical structure surrounding the exposure-outcome pair (Häggström, 2018). While local discovery can avoid the computational inefficiencies of global structure inference, most local methods impose strong graphical assumptions that require prior knowledge to ensure the identifiability of valid adjustment sets. In this paper, we address the following question: *in the absence of prior knowledge, does there exist a polynomial-time algorithm that can select covariates in a principled, automated, and causality-based manner with theoretical guarantees on correctness?*

Related Works Various parametric and nonparametric approaches have been proposed for automated covariate selection (Witte & Didelez, 2019) and adjacent tasks, such as causal parent identification (Yu et al., 2021a,b). Most assume that input variables are non-descendants of the exposure (e.g., by taking the *pretreatment assumption*, which excludes the existence of colliders, mediators, and other descendants of the exposure) (Häggström et al., 2015; Entner et al., 2013; Gultchin et al., 2020; Soleymani et al., 2022; Shah et al., 2022). This requires prior knowledge of the causal graph, and overly simplifies the identification of confounders and other non-descendants. Regression-based covariate selection detects associations rather than causal relationships, and generally requires parametric and/or non-descendants assumptions (Shortreed & Ertefaie, 2017; Tian et al., 2018). Global causal discovery can bypass the non-descendants assumption, but is computationally expensive. Global structure inference over arbitrary causal *directed acyclic graphs* (DAGs) is NP-hard, with the search space growing superexponentially with node count (Chickering et al., 2004). Heuristic search methods are often exponential complexity (Spirtes et al., 2000), though sparsity constraints can enable polynomial runtimes (Claassen et al., 2013). With the pretreatment assumption and *local* causal discovery, runtimes for covariate selection can still scale exponentially with total nodes adjacent to exposure or outcome (Cheng et al., 2022). Thus, automated covariate selection methods tend to present tradeoffs between time complexity and prior knowledge requirements.

Contributions We introduce *Local Discovery by Partitioning* (LDP), a polynomial-time, constraint-based algorithm that infers the causal relations between an exposure-outcome pair $\{X, Y\}$ and a variable set $\mathbf{Z}$ of unknown causal structure. By forgoing common parametric and pretreatment assumptions, LDP addresses settings that are often overlooked by methods that identify causal ancestors. The exposure-outcome pair serves as a nucleus around which LDP assembles a partial causal graph, partitioning $\mathbf{Z}$ into subsets defined solely by their relation to $\{X, Y\}$. LDP addresses the setting of limited prior knowledge and compute by meeting the following desiderata: LDP 1) remains agnostic to the strength of the effect of X on Y, which may be null; 2) imposes minimal assumptions on the relations entailed by the joint distribution $p(X, Y, \mathbf{Z})$; 3) avoids parametric assumptions over causal functions and variable distributions; and 4) scales quadratically with the cardinality of $\mathbf{Z}$ ($|\mathbf{Z}|$).

To provide theoretical guarantees for LDP, we first enumerate an exhaustive taxonomy of eight mutually exclusive partitions defining any arbitrary dataset with respect to an exposure-outcome pair. We then prove that 1) LDP returns asymptotically correct partition labels under the sufficient condition that inter-partition active paths are fully mediated by the exposure-outcome pair, and 2) LDP returns valid adjustment sets even when the latter condition is violated. We provide an open-source Python implementation of LDP on GitHub.[1]

2 PARTITIONS OF Z

Preliminaries are presented in Appendix B. Here, we present the first theoretical results of this work.

Theorem 2.1. *Any* $\mathbf{Z}$ *can be partitioned into eight mutually exclusive subsets (of cardinality greater than or equal to zero) defined solely by their relation to exposure* X *and outcome* Y*. Thus, each* $Z \in \mathbf{Z}$ *uniquely belongs to a single partition defined in Table 1.*

When we assume that all paths among $\{X, Y, \mathbf{Z}\}$ are of length at most 1, these partitions reduce to the eight triple subgraphs in Table A.1. These triples arise from simple enumeration of the three possible relations that one variable can take with respect to another: cause, effect, or neither. This work generalizes Table A.1 to the setting of arbitrary cardinality and indirect active paths. Then, the

[1] https://anonymous.4open.science/r/ldp-0171/

Mutually Exclusive Partitions of Arbitrary $\mathbf{Z}$	
$\mathbf{Z}_1$	*Confounders*: Non-descendants of X that lie on an active backdoor path between X and Y (Definition B.5).
$\mathbf{Z}_2$	*Colliders*: Non-ancestors of $\{X, Y\}$ with at least one active path to X not mediated by Y and at least one active path to Y not mediated by X.
$\mathbf{Z}_3$	*Mediators*: Descendants of X that are ancestors of Y.
$\mathbf{Z}_4$	Non-descendants of Y that are marginally dependent on Y but marginally independent of X (Definition C.3).
$\mathbf{Z}_5$	*Instruments*: Non-descendants of X whose causal effect on Y is fully mediated by X, and that share no confounders with Y (Definition C.1).
$\mathbf{Z}_6$	Descendants of Y where all active paths shared with X are mediated by Y.
$\mathbf{Z}_7$	Descendants of X where all active paths shared with Y are mediated by X.
$\mathbf{Z}_8$	All nodes that share no active paths with X nor Y.

Table 1: Partition categories of any arbitrary $\mathbf{Z}$.

primitive relations of cause, effect, and neither map to the more complex relational combinations enumerated in Tables E.1 and E.2 (e.g., ancestor, non-ancestor, descendant, and non-descendant).

Proof of mutual exclusivity in our taxonomy follows from the fact that each cell of Table E.2 contains a single partition, implying that the pattern of allowable active path types from Z to $\{X, Y\}$ is unique for each partition. Proof of exhaustiveness follows from the fact that every cell that does not violate acyclicity contains a partition, as Table E.2 expresses all possible active path types relative to $\{X, Y\}$. Under mutual exclusivity, one ground truth label exists per variable.

Some partitions coincide with existing terminology while others do not. $\mathbf{Z}_1$ maps to *confounder* (VanderWeele & Shpitser, 2013), $\mathbf{Z}_2$ maps to *collider*, $\mathbf{Z}_3$ maps to *mediator*, $\mathbf{Z}_4$ maps to *pure prognostic variable* (Hahn & Herren, 2022), and $\mathbf{Z}_5$ maps to *instrumental variable* (Lousdal, 2018). To our knowledge, $\{\mathbf{Z}_6, \mathbf{Z}_7, \mathbf{Z}_8\}$ do not coincide with existing terms in the causal inference literature. We dedicate further attention to defining $\mathbf{Z}_4$ and $\mathbf{Z}_5$ in Appendix C, given their role in the sufficient conditions for identifiability. When referring to multiple partitions collectively, e.g., $\mathbf{Z}_5$ and $\mathbf{Z}_7$, we use notation of the form $\mathbf{Z}_{5,7}$.

Within a single partition, there can be arbitrarily many active paths among its members (e.g., $Z_1 \to \cdots \to Z_1$). Some partitions can share active paths with other partitions without violating acyclicity or Table 1 (e.g., $Z_4 \to \cdots \to Z_2$), while other paths are not permissible (e.g., $Z_8 \to \cdots \to Z_5$). When we assemble all partitions into a single DAG, reduce active paths with $\{X, Y\}$ to length-1 arrows, and abstract away *inter-partition active paths*, we obtain Figure 1.

Definition 2.2 (Inter-partition active path)**.** Any active path that is shared by members of at least two partitions, is *not* fully mediated by $\{X, Y\}$, and complies with acyclicity and Table 1.

3 LOCAL DISCOVERY BY PARTITIONING

The pseudocode for LDP is expressed in Algorithm 1. Given an exposure-outcome pair and background set $\mathbf{Z}$, LDP partitions $\mathbf{Z}$ into mutually exclusive subsets as defined in Theorem 2.1. LDP uses a series of constraint-based rules to differentiate and return 1) five partitions individually ($\mathbf{Z}_1$, $\mathbf{Z}_4$, $\mathbf{Z}_5$, $\mathbf{Z}_7$, and $\mathbf{Z}_8$), and 2) a superset $\mathbf{Z}_{\text{POST}}$, which aggregates the remaining three post-treatment partitions ($\mathbf{Z}_2$, $\mathbf{Z}_3$, and $\mathbf{Z}_6$). In the process of differentiating partitions, an intermediate superset ($\mathbf{Z}_{\text{MIX}}$) is temporarily aggregated. We provide a more extensive, high-level description of the algorithm in Appendix D.1. A visual schematic of Algorithm 1 is provided in Table D.1.

Independence Testing Algorithm 1 is compatible with any independence test, which should be selected in accordance with the kind of data in $\mathbf{Z}$. Here, we use the nonparametric chi-square

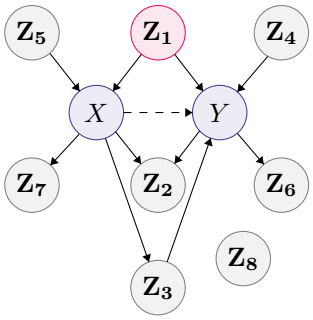

Figure 1: The partitions of $\mathbf{Z}$ (Table 1) reduce to a 10-node DAG surrounding $\{X, Y\}$ where nodes represent partition sets, arrows signify both direct adjacencies and indirect active paths (one or more), and inter-covariate paths are abstracted away. The dashed edge between X and Y indicates that the strength of this relation is unknown, and may be null. Conditioning on $\mathbf{Z}_1$ in red blocks all backdoor paths for $\{X, Y\}$.

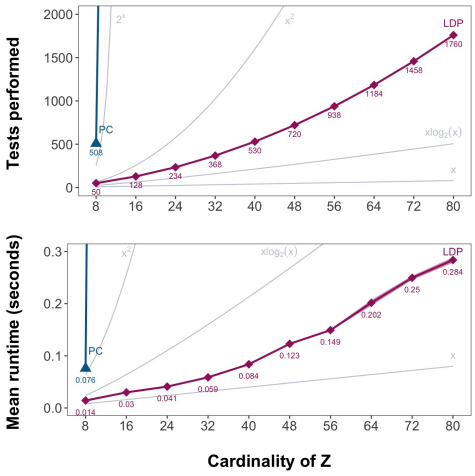

Figure 2: Total tests performed per $\mathbf{Z}$ under an independence oracle (top) and mean runtime over 100 replicates (bottom) as the cardinality of $\mathbf{Z}$ increases, with 95% confidence intervals in shaded regions. Each DAG resembles Figure 1 with equal cardinality per partition ($[1, 10]$). Results are reported for LDP and PC. LDECC and MB-by-MB curves overlapped with PC, with PC outperforming. Exponential, quadratic, $x \log_2(x)$, and linear curves (in tests and milliseconds) serve as comparison. Table G.1 reports raw data.

test for discrete data. We use the Fisher-z test for linear-continuous DAGs, as it is optimal for linear-Gaussian data. The maximum conditioning set size is in $O(|\mathbf{Z}_{1,2,3,5}|)$ (Step 5). All other conditioning sets are cardinality one or two.

Time Complexity We report Big O complexity in terms of total independence tests performed, as is conventional for constraint-based causal discovery (Spirtes et al., 2000; Tsamardinos et al., 2006). The first for-loop (Steps 1–3) requires a linear number of tests in $O(|\mathbf{Z}|)$, where Step 1 caches all marginal test results for every candidate relative to $\{X, Y\}$. Step 4 requires $O(|\mathbf{Z}|^2)$ tests. Step 5 requires $O(|\mathbf{Z}|)$ and Step 6 requires $O(|\mathbf{Z}|^2)$. Step 7 requires no tests, as it uses cached test results. Thus, total tests performed is in $O(|\mathbf{Z}|^2)$. Figure 2 indicates sub-quadratic runtimes, outperforming existing algorithms by large margins.

3.1 LDP For Covariate Selection

We define popular theoretical criteria for covariate selection in Section B.4, which are consistent with the backdoor criterion (Definition B.4). As LDP returns $\mathbf{Z}_1$, $\mathbf{Z}_4$, and $\mathbf{Z}_5$, LDP can be used as

an automated preprocessing step for covariate selection under the *common cause criterion* (which retains only $\mathbf{Z}_1$), the *disjunctive cause criterion* (which retains $\{\mathbf{Z}_1, \mathbf{Z}_4, \mathbf{Z}_5\}$) (VanderWeele & Shpitser, 2011), and the *outcome criterion* (which retains $\{\mathbf{Z}_1, \mathbf{Z}_4\}$) (Brookhart et al., 2006), all of which yield valid adjustment sets under the backdoor criterion and the *generalized adjustment criterion* (Perkovic et al., 2015).

3.2 Sufficient Conditions for Identifiability

This work presents three main theoretical results: 1) the existence of eight exhaustive and mutually exclusive partitions that define any arbitrary $\mathbf{Z}$ (Theorem 2.1); 2) LDP yields asymptotically correct partitions of $\mathbf{Z}$ under sufficient conditions (Theorem D.1); and 3) LDP returns valid adjustment sets under weakened sufficient conditions (Theorem D.2). In Appendix D.2, we describe the sufficient conditions for these results.

4 EXPERIMENTAL DESIGN

Experimental objectives were to demonstrate that 1) LDP correctly partitions $\mathbf{Z}$ under sufficient conditions, 2) LDP returns valid adjustment sets under weakened sufficient conditions, and 3) adjustment sets selected by LDP yield precise and unbiased *average treatment effect* (ATE) estimates relative to baselines. All baselines are constraint-based and do not take the pretreatment assumption. Data are custom synthetic DAGs and one benchmark from the `bnlearn` Bayesian Network Repository (Scutari, 2010).[2]

Baseline Methods We compare the performance of LDP against three baselines: 1) the PC Algorithm (PC), a classic global structure inference algorithm that provides asymptotic theoretical guarantees (Spirtes et al., 2000); 2) MB-by-MB, a local Markov blanket learner that infers the local structure around a target node to distinguish parents from children (Wang et al., 2014); and 3) Local Discovery using Eager Collider Checks (LDECC), a local discovery algorithm that leverages unshielded colliders to orient the edges around a target to differentiate parents from children (Gupta et al., 2023). Additional details of these methods and their performance evaluation are provided in Appendix F.

Synthetic Data Theoretical guarantees were validated for 18 data generating processes and four DAG structures. Discrete data simulations used 12 data generating processes for the 10-node DAG (Figure 1), four processes for both 13-node DAGs (Figure A.3), and two processes for the 17-node DAG (Figure A.4). Causal mechanisms were linear and nonlinear. Six linear-continuous data generating processes were simulated for the 10-node DAG (Figure 1). Structural equations are reported in Tables G.2 and G.3.

MILDEW Benchmark Data The MILDEW network models fungicide use against powdery mildew in winter wheat (Jensen & Jensen, 1996). We selected one exposure-outcome pair (MIKRO_1 $\rightarrow$ MELDUG_2) that meets sufficient conditions for LDP. All variables are categorical. $\mathbf{Z}$ contains 31 nodes in $\{\mathbf{Z}_1, \mathbf{Z}_2, \mathbf{Z}_4, \mathbf{Z}_5, \mathbf{Z}_8\}$, with a low proportion of confounders ($|\mathbf{Z}_1| = 2$) and high proportion of colliders ($|\mathbf{Z}_2| = 14$). Data were sampled using the `bnlearn` R package (Scutari, 2010). Figure A.5 further describes the DAG used for inference and evaluation.

5 EMPIRICAL RESULTS

Partition Label Correctness We measure partition accuracy as the percent of partition labels that are consistent with ground truth. Results on the 10-node DAG with one variable per partition (Figure 1) indicate that LDP correctly partitions $\mathbf{Z}$ under continuous, discrete, linear, and nonlinear data generating processes (Figure 3, Tables G.4, G.5). Figure 3 also supports the claim that LDP is agnostic to the strength of the direct effect of X on Y, as results are unharmed when X is not adjacent to Y. High partition accuracy on 13-node and 17-node DAGs containing M-structures with paths between $\mathbf{Z}_4$, $\mathbf{Z}_5$, and $\mathbf{Z}_2$ (Figures A.3, A.4) empirically demonstrate robustness to certain kinds of inter-partition active paths (Tables G.6, G.7). High partition accuracies on MILDEW ($\geq 90\%$;

[2] https://www.bnlearn.com/bnrepository/

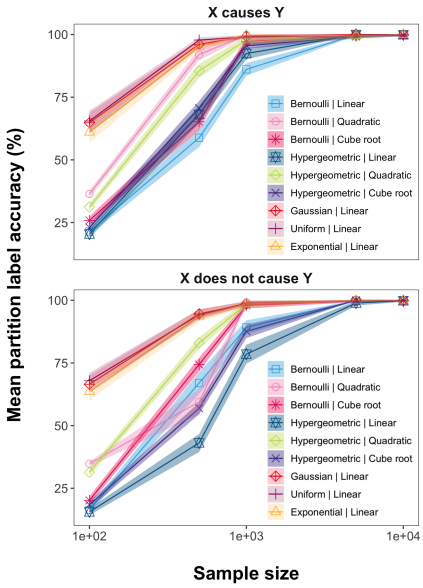

Figure 3: Partition label accuracy of LDP on a 10-node DAG with one node per partition (Figure 1). Accuracy is averaged over 100 DAGs (i.e., 800 variables total, excluding exposure-outcome pairs), with 95% confidence intervals in shaded regions. Independence was determined by chi-square tests for discrete data and Fisher-z for continuous data ($\alpha = 0.001$). Tables G.4 and G.5 report raw data.

Figure G.1) further corroborate the ability of LDP to handle certain forms of inter-partition active paths.

Valid Adjustment Sets First, we explore adjustment set quality for two graphs with small $\mathbf{Z}_1$: the MILDEW benchmark ($|\mathbf{Z}_1| = 2$) and a synthetic linear-Gaussian DAG ($|\mathbf{Z}_1| = 1$) (Figure 4). For MILDEW, LDP outperformed all baselines on confounder recall in $\mathbf{A}_{XY}$, test count, and runtime. Though PC sometimes obtained a greater percentage of valid $\mathbf{A}_{XY}$ for MILDEW, LDP achieved higher confounder precision and recall. High confounder recall for LDP is reflective of its ability to detect confounders that are not directly adjacent to either X nor Y, unlike local baselines. As expected, LDP displayed superior confounder precision under the CCC for both graphs but was comparable to other methods when $\mathbf{Z}_4$ and $\mathbf{Z}_5$ were intentionally retained under the DCC. Though global and local baselines should theoretically detect confounders that are directly adjacent to both X and Y, only LDP consistently returned valid $\mathbf{A}_{XY}$ for the linear-Gaussian DAG under both adjustment criteria.

A synthetic DAG with a complex backdoor path was constructed to illustrate a known failure mode of LDP partition labeling that still results in valid adjustment sets (Figure A.6). In this DAG, the confounder adjacent to Y is marginally dependent on $\mathbf{Z}_4$ and will be mislabeled as $\mathbf{Z}_{\text{POST}}$. Further, a collider that is 1) a non-descendent of X and 2) conditionally independent of $\{X, Y\}$ given $\mathbf{Z}_1$ is guaranteed to be placed in $\mathbf{Z}_1$. Despite these mislabelings, LDP returned a valid adjustment set for 99% (99/100) of replicates (sample size $n = 5k$). Figure A.6 describes further details.

Conditioning Set Size Local baselines faced challenges with chi-square independence tests on MILDEW for $n \geq 75k$. LDECC errored out on 2/10 and 10/10 replicates at $n = 75k$ and $n = 100k$, respectively, while MB-by-MB could not return results for 3/10 and 9/10. Independence test failures persisted even with resampling from the ground truth DAG, and are likely due to large conditioning sets resulting in low or no samples for some groups during binning. While the maximum conditioning set size for LDP on MILDEW was 4, this was 17 for LDECC and 19 for MB-by-MB.

Statistical Efficiency ATE estimate variance served as a measure of statistical efficiency across baselines (Figure 4, bottom). The ATE was estimated using linear regression[3] for linear-Gaussian

[3] https://scikit-learn.org/

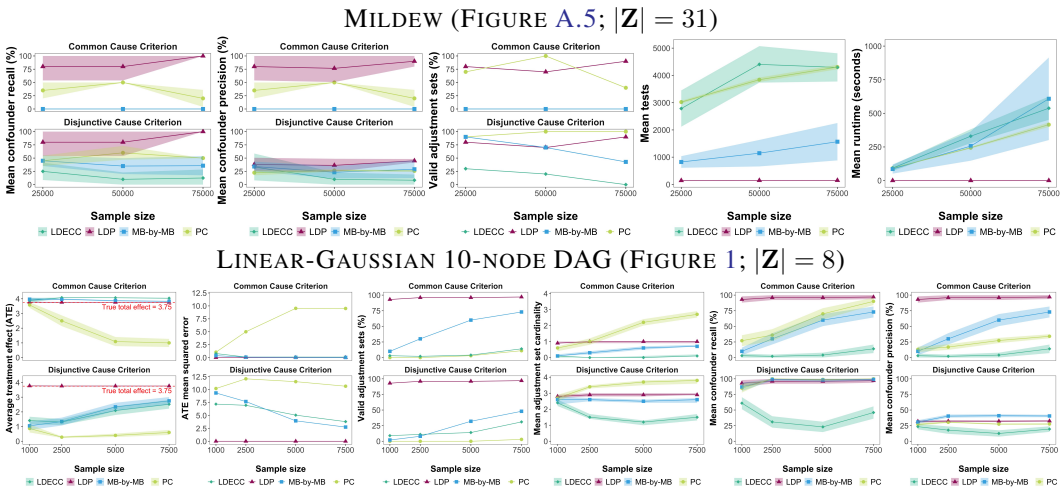

Figure 4: Baselines on MILDEW and 10-node DAGs with chi-square ($\alpha = 0.001$) and Fisher-z tests ($\alpha = 0.01$), respectively. Results are for 10 replicates of MILDEW and 100 replicates of the 10-node DAG per sample size (95% confidence intervals in shaded regions). Confounder precision and recall are computed per adjustment set, demonstrating the proportion of true confounders and non-confounders. Raw data are in Tables G.8 and G.9.

DAGs with a ground truth total effect of 3.75. LDP returned the highest quality adjustment sets in terms of ATE mean squared error (MSE), confounder recall, and percent valid, with baselines lagging even as sample size increased. LDP generally produced the least biased ATE estimates and lowest ATE variance, and was the only method to achieve unbiased estimates under the DCC. Rising ATE MSE for PC may be explained by the cardinality of $\mathbf{A}_{XY}$ increasing with sample size. Ground truth $|\mathbf{A}_{XY}| = 1$ under the CCC and 3 under the DCC, which LDP adhered to more closely than baselines.

Robustness to Latent Confounding We probed the robustness of LDP to specific forms of latent confounding in $\mathcal{G}_{XY\mathbf{Z}}$ that contain M-structures or butterfly structures (Ding & Miratrix, 2014). Each experiment tested 100 replicate 13-node, linear-Bernoulli DAGs (Figure A.3) using chi-square tests ($\alpha = 0.001$). In DAGs with M-structures where node $M_1 \in \mathbf{Z}_5$ is latent, partition accuracy was 99.8% (95% CI $[99.4, 100]$), $\mathbf{Z}_1$ precision and recall were 99.0% (95% CI $[97.0, 100]$), and M-colliders were correctly labeled. With $M_2 \in \mathbf{Z}_4$ latent, partition accuracy was 80.0% (95% CI $[80.0, 80.0]$), $\mathbf{Z}_1$ precision was 33.3% (95% CI $[33.3, 33.3]$), and $\mathbf{Z}_1$ recall was 100.0%, as the M-collider was placed in $\mathbf{Z}_1$. In such a case, $\mathbf{A}_{XY}$ could induce M-bias (Ding & Miratrix, 2014). Treating butterfly nodes $\{B_1, B_2\} \in \mathbf{Z}_1$ as latent had no effect on performance. With $B_3 \in \mathbf{Z}_1$ unobserved, partition accuracy was 80.0% (95% CI $[80.0, 80.0]$) and $\mathbf{Z}_1$ precision and recall were 66.7% (95% CI $[66.7, 66.7]$), leaving an unblocked backdoor path $X - B_3 - Y$. These results indicate that causal sufficiency in $\mathcal{G}_{XY\mathbf{Z}}$ is not a necessary condition, but certain forms of latent confounding are detrimental to both partition accuracy and valid adjustment set identification.

Limitations Performance of LDP will be constrained by the accuracy and runtime of the independence test that it employs. Theoretical guarantees in this work are based on asymptotic analyses, while our empirical results offer insights into performance under finite sample conditions. We leave the derivation of necessary conditions for identifiability to future inquiry, including more rigorous conclusions on the impacts of latent confounding.

Acknowledgments

The authors would like to acknowledge support from the NSF Graduate Research Fellowship; NSF 1750326, 2212175; and NIH R01AG080991, R01AG076234 for this research.

References

Brookhart, M. A., Schneeweiss, S., Rothman, K. J., Glynn, R. J., Avorn, J., and Stürmer, T. Variable Selection for Propensity Score Models. *American Journal of Epidemiology*, 163(12):1149–1156, June 2006. ISSN 1476-6256, 0002-9262. doi: 10.1093/aje/kwj149.

Cheng, D., Li, J., Liu, L., Zhang, J., Liu, J., and Le, T. D. Local search for efficient causal effect estimation. *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–14, 2022. ISSN 1041-4347, 1558-2191, 2326-3865. doi: 10.1109/TKDE.2022.3218131.

Chickering, D. M., Heckerman, D., and Meek, C. Large-Sample Learning of Bayesian Networks is NP-Hard. *Journal of Machine Learning Research*, 5:1287–1330, 2004.

Claassen, T., Mooij, J. M., and Heskes, T. Learning Sparse Causal Models is not NP-hard. *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence (UAI2013)*, 2013.

Ding, P. and Miratrix, L. To Adjust or Not to Adjust? Sensitivity Analysis of M-Bias and Butterfly-Bias, August 2014. arXiv:1408.0324 [math, stat].

Elwert, F. and Winship, C. Endogenous Selection Bias: The Problem of Conditioning on a Collider Variable. *Annual Review of Sociology*, 40(1):31–53, July 2014. ISSN 0360-0572, 1545-2115. doi: 10.1146/annurev-soc-071913-043455.

Entner, D., Hoyer, P. O., and Spirtes, P. Data-driven covariate selection for nonparametric estimation of causal effects. In *Proceedings of the 16th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2013.

Gultchin, L., Kusner, M. J., Kanade, V., and Silva, R. Differentiable Causal Backdoor Discovery. In *Proceedings of the 23rdInternational Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 108, Palermo, Italy, 2020. PMLR.

Guo, F. R., Lundborg, A. R., and Zhao, Q. Confounder Selection: Objectives and Approaches, August 2022. arXiv:2208.13871 [math, stat].

Gupta, S., Childers, D., and Lipton, Z. C. Local Causal Discovery for Estimating Causal Effects. In *Proceedings of the 2nd Conference on Causal Learning and Reasoning (CLeaR)*. arXiv, February 2023. URL http://arxiv.org/abs/2302.08070. arXiv:2302.08070 [cs, stat].

Hahn, P. R. and Herren, A. Feature selection in stratification estimators of causal effects: lessons from potential outcomes, causal diagrams, and structural equations, September 2022. URL http://arxiv.org/abs/2209.11400. arXiv:2209.11400 [stat].

Hernán, M. A. and Robins, J. M. Instruments for Causal Inference: An Epidemiologist's Dream? *Epidemiology*, 17(4):360–372, July 2006. ISSN 1044-3983. doi: 10.1097/01.ede.0000222409.00878.37. URL https://journals.lww.com/00001648-200607000-00004.

Hernán, M. A. and Robins, J. M. *Causal Inference: What If*. 2020.

Hernán, M. A., Hernández-Díaz, S., and Robins, J. M. A Structural Approach to Selection Bias. *Epidemiology*, 15(5):615–625, September 2004. ISSN 1044-3983. doi: 10.1097/01.ede.0000135174.63482.43.

Holmberg, M. J. and Andersen, L. W. Collider bias. *JAMA Guide to Statistics and Methods*, 327 (13), 2022.

Hoyer, P. O., Janzing, D., Mooij, J. M., Peters, J., and Schölkopf, B. Nonlinear causal discovery with additive noise models. *Advances in Neural Information Processing Systems 21 (NIPS 2008)*, 2008.

Häggström, J. Data-driven confounder selection via Markov and Bayesian networks: Data-Driven Confounder Selection. *Biometrics*, 74(2):389–398, June 2018. ISSN 0006341X. doi: 10.1111/biom.12788.

Häggström, J., Persson, E., Waernbaum, I., and De Luna, X. CovSel : An R Package for Covariate Selection When Estimating Average Causal Effects. *Journal of Statistical Software*, 68(1), 2015. ISSN 1548-7660. doi: 10.18637/jss.v068.i01.

Imbens, G. Instrumental variables: an econometrician's perspective. *National Bureau of Economic Research*, (No. w19983), 2014.

Jensen, A. L. and Jensen, F. V. MIDAS: An influence diagram for management of mildew in winter wheat. In *Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence*, pp. 349–356, 1996.

Kalisch, M. and Buhlmann, P. Estimating High-Dimensional Directed Acyclic Graphs with the PC-Algorithm. *Journal of Machine Learning Research*, (8):613–636, 2007.

Labrecque, J. and Swanson, S. A. Understanding the Assumptions Underlying Instrumental Variable Analyses: a Brief Review of Falsification Strategies and Related Tools. *Current Epidemiology Reports*, 5(3):214–220, September 2018. ISSN 2196-2995. doi: 10.1007/s40471-018-0152-1. URL http://link.springer.com/10.1007/s40471-018-0152-1.

Lousdal, M. L. An introduction to instrumental variable assumptions, validation and estimation. *Emerging Themes in Epidemiology*, 15(1):1, December 2018. ISSN 1742-7622. doi: 10.1186/s12982-018-0069-7. URL https://ete-online.biomedcentral.com/articles/10.1186/s12982-018-0069-7.

Lu, H., Cole, S. R., Platt, R. W., and Schisterman, E. F. Revisiting Overadjustment Bias. *Epidemiology*, 32(5):e22–e23, September 2021. ISSN 1044-3983. doi: 10.1097/EDE.0000000000001377.

Maathuis, M. H. and Colombo, D. A generalized back-door criterion. *The Annals of Statistics*, 43 (3), June 2015. ISSN 0090-5364. doi: 10.1214/14-AOS1295.

Neal, B. *Introduction to Causal Inference*. 2020.

Pearl, J. Causal diagrams for empirical research. *Biometrika*, 82(4):669–688, 1995.

Pearl, J. Direct and Indirect Effects. In *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*, 2001.

Pearl, J. Causal inference in statistics: An overview. *Statistics Surveys*, 3(none), January 2009. ISSN 1935-7516. doi: 10.1214/09-SS057. URL https://projecteuclid.org/journals/statistics-surveys/volume-3/issue-none/Causal-inference-in-statistics-An-overview/10.1214/09-SS057.full.

Pearl, J. On a Class of Bias-Amplifying Variables that Endanger Effect Estimates. 2012.

Perkovic, E., Textor, J., Kalisch, M., and Maathuis, M. H. A Complete Generalized Adjustment Criterion. *Proceedings of the Thirty-First Conference of Uncertainty in Artificial Intelligence (UAI)*, pp. 682–691, 2015.

Peters, J., Janzing, D., and Scholkopf, B. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, Cambridge, Massachusetts, 2017. ISBN 978-0-262-03731-0.

Rubin, D. B. For objective causal inference, design trumps analysis. *The Annals of Applied Statistics*, 2(3), September 2008. ISSN 1932-6157. doi: 10.1214/08-AOAS187.

Runge, J. Necessary and sufficient graphical conditions for optimal adjustment sets in causal graphical models with hidden variables. *Proceedings of the 35th Conference on Neural Information Processing Systems*, 2021.

Schisterman, E. F., Cole, S. R., and Platt, R. W. Overadjustment Bias and Unnecessary Adjustment in Epidemiologic Studies. *Epidemiology*, 20(4):488–495, July 2009. ISSN 1044-3983. doi: 10.1097/EDE.0b013e3181a819a1.

Schnitzer, M. E., Lok, J. J., and Gruber, S. Variable Selection for Confounder Control, Flexible Modeling and Collaborative Targeted Minimum Loss-Based Estimation in Causal Inference. *The International Journal of Biostatistics*, 12(1):97–115, May 2016. ISSN 2194-573X, 1557-4679. doi: 10.1515/ijb-2015-0017.

Scutari, M. Learning Bayesian Networks with the bnlearn R Package, July 2010. URL http://arxiv.org/abs/0908.3817. arXiv:0908.3817 [stat].

Shah, A., Shanmugam, K., and Ahuja, K. Finding Valid Adjustments under Non-ignorability with Minimal DAG Knowledge. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 151, Valencia, Spain, 2022. PMLR.

Shimizu, S., Hoyer, P. O., Hyvarinen, A., and Kerminen, A. A Linear Non-Gaussian Acyclic Model for Causal Discovery. *Journal of Machine Learning Research*, 7:2003–2030, 2006.

Shortreed, S. M. and Ertefaie, A. Outcome-adaptive lasso: Variable selection for causal inference. *Biometrics*, 73(4):1111–1122, December 2017. ISSN 0006-341X, 1541-0420. doi: 10.1111/biom.12679.

Soleymani, A., Raj, A., Bauer, S., Schölkopf, B., and Besserve, M. Causal Feature Selection via Orthogonal Search. *Transactions on Machine Learning Research*, September 2022. URL http://arxiv.org/abs/2007.02938. arXiv:2007.02938 [cs, math, stat].

Spirtes, P., Glymour, C., and Scheines, R. *Causation, Prediction, and Search*, volume 81 of *Lecture Notes in Statistics*. Springer New York, New York, NY, 2000. ISBN 978-1-4612-7650-0 978-1-4612-2748-9. doi: 10.1007/978-1-4612-2748-9.

Tian, Y., Schuemie, M. J., and Suchard, M. A. Evaluating large-scale propensity score performance through real-world and synthetic data experiments. *International Journal of Epidemiology*, 47(6): 2005–2014, December 2018. ISSN 0300-5771, 1464-3685. doi: 10.1093/ije/dyy120.

Tsamardinos, I., Aliferis, C. F., and Statnikov, A. Algorithms for Large Scale Markov Blanket Discovery. 2003.

Tsamardinos, I., Brown, L. E., and Aliferis, C. F. The max-min hill-climbing Bayesian network structure learning algorithm. *Machine Learning*, 65(1):31–78, October 2006. ISSN 0885-6125, 1573-0565. doi: 10.1007/s10994-006-6889-7.

VanderWeele, T. J. and Shpitser, I. A New Criterion for Confounder Selection. *Biometrics*, 67(4): 1406–1413, December 2011. ISSN 0006341X. doi: 10.1111/j.1541-0420.2011.01619.x.

VanderWeele, T. J. and Shpitser, I. On the definition of a confounder. *The Annals of Statistics*, 41 (1), February 2013. ISSN 0090-5364. doi: 10.1214/12-AOS1058.

Wang, C., Zhou, Y., Zhao, Q., and Geng, Z. Discovering and orienting the edges connected to a target variable in a DAG via a sequential local learning approach. *Computational Statistics & Data Analysis*, 77:252–266, September 2014. ISSN 01679473. doi: 10.1016/j.csda.2014.03.003. URL https://linkinghub.elsevier.com/retrieve/pii/S0167947314000802.

Witte, J. and Didelez, V. Covariate selection strategies for causal inference: Classification and comparison. *Biometrical Journal*, 61(5):1270–1289, September 2019. ISSN 0323-3847, 1521-4036. doi: 10.1002/bimj.201700294.

Yu, K., Guo, X., Liu, L., Li, J., Wang, H., Ling, Z., and Wu, X. Causality-based Feature Selection: Methods and Evaluations. *ACM Computing Surveys*, 53(5):1–36, September 2021a. ISSN 0360-0300, 1557-7341. doi: 10.1145/3409382.

Yu, K., Liu, L., and Li, J. A Unified View of Causal and Non-causal Feature Selection. *ACM Transactions on Knowledge Discovery from Data*, 15(4):1–46, August 2021b. ISSN 1556-4681, 1556-472X. doi: 10.1145/3436891.

Zhang, K. and Hyvarinen, A. On the Identifiability of the Post-Nonlinear Causal Model. *Uncertainty in Artificial Intelligence*, 2009.

A APPENDIX: SUPPLEMENTAL FIGURES

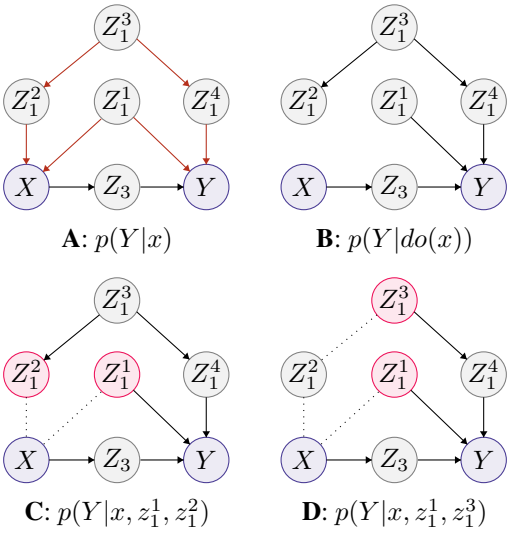

Figure A.1: Valid adjustment sets. Here, the effect of exposure X on outcome Y is mediated by Z_3. Let $\mathbf{Z}_1 = \{Z_1^1, Z_1^2, Z_1^3, Z_1^4\}$. (**A**) The conditional distribution $p(Y|x)$ fails to isolate the causal association between X and Y due to the open backdoor paths through $\mathbf{Z}_1$, pictured as red arrows. (**B**) We can isolate the causal association between X and Y by intervening on X such that edges $Z_1^2 \to X$ and $Z_1^1 \to X$ are removed. This blocks the non-causal association flowing through these backdoor paths. (**C**) We can identify the interventional distribution $p(Y|do(x))$ via a statistical quantity by conditioning on valid adjustment set $\{Z_1^1, Z_1^2\}$ (highlighted in red), which also blocks the flow of non-causal association. (**D**) Valid adjustment sets are often non-unique. An alternative valid adjustment set for this structure would be $\{Z_1^1, Z_1^3\}$, and still others exist. Figure adapted from Neal (2020).

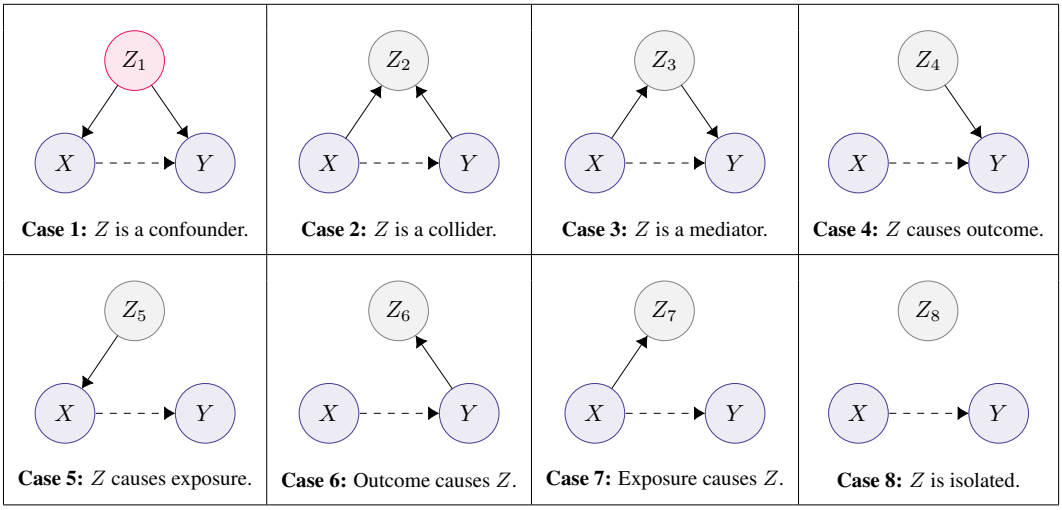

Table A.1: All potential acyclic triple subgraphs that can be induced by X, Y, and a single Z when paths are restricted to a length of 1. The dashed arrow from exposure X to outcome Y indicates that the strength of this relation is unknown. While the effect of X on Y might be null, it is known that $X \not\perp\!\!\!\perp Y$ and that Y does not cause X. The partition taxonomy proposed in this work (Table 1) generalizes these cases to more complex structures. In the more complex setting, edges represent both direct adjacencies and indirect active paths. Absence of a directed edge therefore indicates either an inactive path or no path at all.

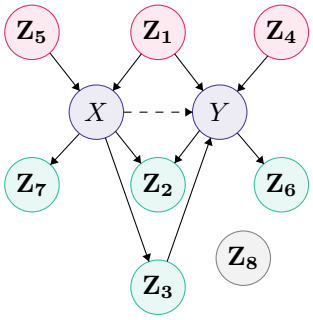

Figure A.2: Pretreatment variables (red) versus post-treatment variables (green). $\mathbf{Z}_1$ (confounders), $\mathbf{Z}_4$, and $\mathbf{Z}_5$ (instruments) are pretreatment variables, which causally precede exposure X. $\mathbf{Z}_2$ (colliders), $\mathbf{Z}_3$ (mediators), $\mathbf{Z}_6$, and $\mathbf{Z}_7$ are post-treatment variables, with X as their causal ancestor. $\mathbf{Z}_8$ is neither pre- nor post-treatment, as it is causally unrelated to X.

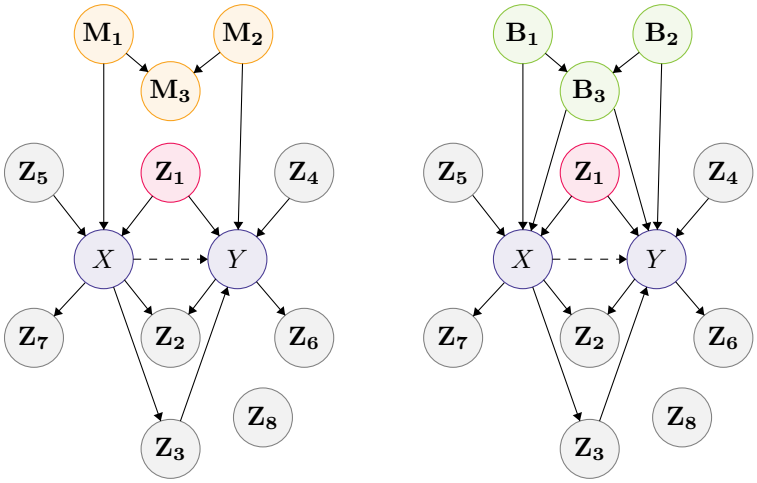

Figure A.3: M-structures and butterfly structures. Ten-node DAG plus M-structure (right) and ten-node DAG plus butterfly structure (left). Note that $\mathbf{M}_1 \in \mathbf{Z}_5$, $\mathbf{M}_2 \in \mathbf{Z}_4$, $\mathbf{M}_3 \in \mathbf{Z}_2$, and $\{\mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3\} \in \mathbf{Z}_1$. Performance of LDP on these structures is reported in Table G.6.

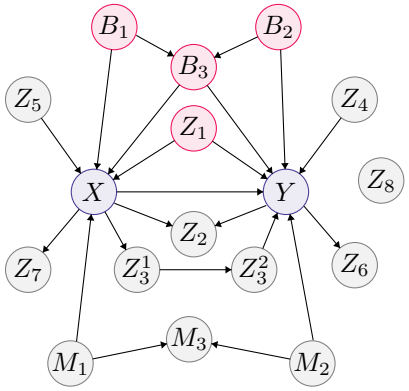

Figure A.4: Seventeen-node DAG with M-structure, butterfly structure, and mediator chain. Note that $\mathbf{M}_1 \in \mathbf{Z}_5$, $\mathbf{M}_2 \in \mathbf{Z}_4$, $\mathbf{M}_3 \in \mathbf{Z}_2$, and $\{\mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3\} \in \mathbf{Z}_1$. Nodes highlighted in red ($\{Z_1, B_1, B_2, B_3\}$) represent all confounders for $\{X, Y\}$. Performance of LDP on this structure is reported in Table G.7.

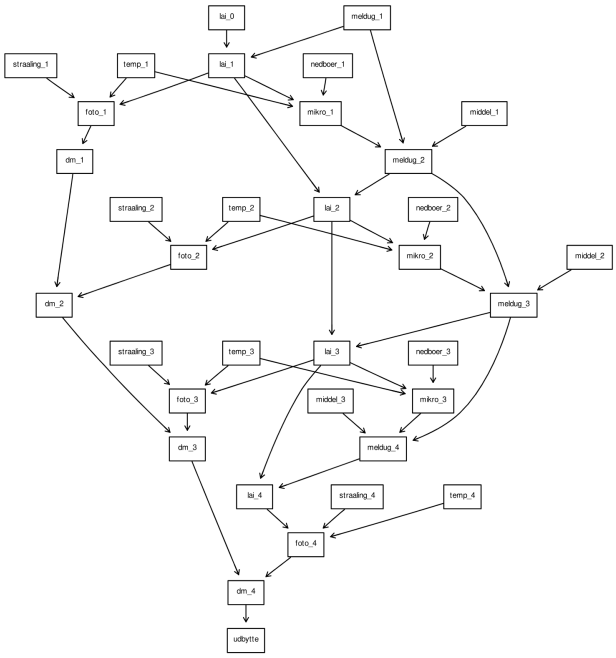

Figure A.5: The complete ground truth MILDEW DAG (Jensen & Jensen, 1996) obtained from `bnlearn` (Scutari, 2010). The ground truth DAG contains 35 nodes, 46 edges, and 540150 parameters. The average Markov blanket size, average degree, and maximum in-degree are 4.57, 2.63, and 3, respectively. Inference and evaluation omit variables DM_1 and FOTO_1 due to independence test challenges with LDP, MB-by-MB, and LDECC, including those described in Section 5 for MB-by-MB and LDECC (which were made more severe by inclusion of these nodes). Performance of LDP on this structure is reported in Table G.8 and Figure 4.

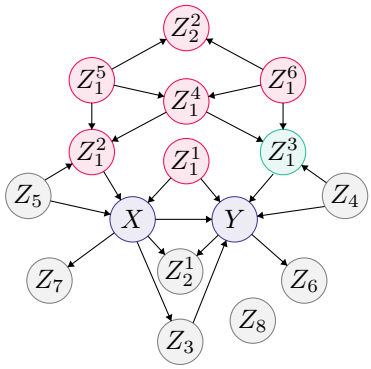

Figure A.6: A complex backdoor path illustrates a known failure mode of LDP partition labeling that is still successful for valid adjustment set identification. In theory, all nodes highlighted in red will be placed in $\mathbf{Z}_1$ by LDP. Even though Z_1^2 is adjacent to the only instrument in this DAG, this confounder will be discoverable due to its marginal independence with Z_1^1. Due to its marginal dependence on Z_4, confounder Z_1^3 will be mislabeled and placed in $\mathbf{Z}_{\text{POST}}$ by LDP. This mislabeling persists even under infinite data. Due to its marginal independence with Z_4, collider Z_2^2 will be mislabeled and placed in $\mathbf{Z}_1$. Despite these mislabelings, the red node set constitutes a valid adjustment set per the proofs in Section E.3. LDP returned a valid adjustment set for this structure for 99% (99/100) of replicates at $n = 5k$ samples and 98% (98/100) of replicates at $n = 10k$ samples. Noise was hypergeometric, causal mechanisms were quadratic, and $\alpha = 0.001$ with the chi-square independence test.

B APPENDIX: PRELIMINARIES

B.1 Notation

Univariate random variables are denoted by uppercase letters (e.g., X). Sets or multivariate random variables are denoted by bold uppercase (e.g., $\mathbf{Z}$), and graphs by calligraphic letters (e.g., $\mathcal{G}$). Let $\{X, Y\}$ be continuous or discrete random variables representing an exposure and outcome, respectively. Let $\mathbf{Z}$ be a set of discrete or continuous random variables of unknown relation to $\{X, Y\}$ and $\mathcal{G}_{XY\mathbf{Z}}$ be the graph induced by $\{X, Y, \mathbf{Z}\}$.

B.2 Causal Graphical Models

We restrict our attention to the set of causal graphs that are directed and acyclic. Causal DAGs are probabilistic graphical models that factorize joint distributions and impose a causal interpretation on directed edges, such that $X \rightarrow Y$ implies that X causes Y. In this work, we assume that any $\mathcal{G}_{XY\mathbf{Z}}$ is a causal DAG satisfying the causal Markov condition and faithfulness as defined by Spirtes et al. (2000). This implies that the conditional independence relations entailed by the joint distribution $p(X, Y, \mathbf{Z})$ correspond precisely to those entailed by the Markov condition applied to $\mathcal{G}_{XY\mathbf{Z}}$, i.e., $p(X, Y, \mathbf{Z})$ and $\mathcal{G}_{XY\mathbf{Z}}$ are *faithful* to each other. We define active and inactive paths in $\mathcal{G}_{XY\mathbf{Z}}$ following from the concept of d-separation.

Definition B.1 (D-separation, Spirtes et al. 2000)**.** Nodes X and Y in arbitrary causal DAG $\mathcal{G}$ are d-separated given node set $\mathbf{D}$ (where $\{X, Y\} \notin \mathbf{D}$) when there is no undirected path between X and Y that is *active* relative to $\mathbf{D}$.

Definition B.2 (Active paths, Spirtes et al. 2000)**.** An undirected path is *active* relative to node set $\mathbf{D}$ when every node on this path is active relative to $\mathbf{D}$. Node $V \in \mathcal{G}$ is active on a path relative to $\mathbf{D}$ if

1. $V \notin \mathbf{D}$ is not a collider,
2. $V \in \mathbf{D}$ is a collider, or
3. $V \notin \mathbf{D}$ is a collider and at least one of its descendants is in $\mathbf{D}$.

By extension, we take an *inactive* path to be one that does not meet Definition B.2 (e.g., due to existence of a collider $\notin \mathbf{D}$ on that path). As the definitions of active and inactive are with respect to $\mathbf{D}$, we assume $\mathbf{D} = \emptyset$ unless explicitly stated. We classify active paths between two nodes $\{Z, Z'\}$ following from Table E.1: 1) $Z \rightarrow \cdots \rightarrow Z'$, 2) $Z \leftarrow \cdots \leftarrow Z'$, or 3) $Z \leftarrow \cdots Z'' \cdots \rightarrow Z'$, where Z'' denotes a third node.

We say that causal association flows from exposure X to outcome Y through directed paths $X \rightarrow \cdots \rightarrow Y$. A non-causal association between X and Y due to a common cause also presents as statistical dependency, per Reichenbach's common cause principle (Peters et al., 2017). Such common causes lie along *backdoor paths* for $\{X, Y\}$ ($X \leftarrow \cdots Z \cdots \rightarrow Y$).

Definition B.3 (Backdoor path, Pearl 2009)**.** Any non-causal path between exposure X and outcome Y with an edge pointing into X ($\cdots \rightarrow X$).

B.3 Covariate Selection for Valid Adjustment

The primary focus of this work is the identification of valid adjustment sets under the backdoor criterion.

Definition B.4 (Valid adjustment under the backdoor criterion, Peters et al. 2017)**.** Let $\mathbf{A}_{XY}$ be an adjustment set for $\{X, Y\}$ that does not contain $\{X, Y\}$. $\mathbf{A}_{XY}$ is valid if

1. $\mathbf{A}_{XY}$ contains no descendants of X and
2. $\mathbf{A}_{XY}$ blocks all backdoor paths from X to Y.

Blocking all backdoor paths is equivalent to $\{X, Y\}$ achieving d-separation when all edges exiting X ($X \rightarrow \cdots$) are removed. Note that valid adjustment sets can be non-unique (Figure A.1) and that Definition B.4 does not consider minimality nor optimality, as explored elsewhere (e.g. Runge (2021)). In Section B.4, we define criteria that are consistent with Definition B.4 but provide additional guidance.

Confounding Given a valid adjustment set $\mathbf{A}_{XY}$, we eliminate confounding bias and obtain *unconfoundedness* (VanderWeele & Shpitser, 2013): the independence of exposure X and the potential outcomes of Y, factual or counterfactual. This condition is also commonly referred to as *ignorability*, *ignorable treatment assignment*, or *exchangeability*. LDP identifies a valid adjustment set by returning a set of confounders that blocks all backdoor paths in $\mathcal{G}_{XY\mathbf{Z}}$.

Definition B.5 (Confounder, VanderWeele & Shpitser 2013)**.** A confounder is a pretreatment covariate Z_1 for which there exists a set of other covariates S (which may be the empty set) such that the effect of exposure X on outcome Y is unconfounded given $\{Z_1, S\}$ but not given any proper subset of $\{Z_1, S\}$.

In this work, we refer to all ground truth confounders for $\{X, Y\}$ as partition $\mathbf{Z}_1 \in \mathbf{Z}$ (Table 1). Following from Definition B.5, conditioning on the set of all confounders $\mathbf{Z}_1$ blocks all backdoor paths and the flow of non-causal association from X to Y. Following from Figure A.1, there can exist multiple subsets of $\mathbf{Z}_1$ that block all backdoor paths (i.e., some members of $\mathbf{Z}_1$ might be redundant in their ability to block a given backdoor path). This redundancy permits LDP to return a valid adjustment that is a subset of $\mathbf{Z}_1$.

B.4 Covariate Selection Criteria

Pearl's *backdoor path criterion* dictates that a valid adjustment set contains no descendants of the exposure and blocks all backdoor paths (Definition B.4; Figure A.1) (Pearl, 1995). Additional covariate selection criteria have been proposed, which are consistent with the backdoor criterion but provide additional guidance. The *common cause criterion* advocates controlling only for confounders ($\mathbf{Z}_1$), and is popular in practice (Guo et al., 2022). The *pretreatment criterion* controls for all measured baseline variables, an approach previously defended by Donald Rubin (Rubin, 2008; Guo et al., 2022). This approach is at risk of overadjustment (Schisterman et al., 2009; Lu et al., 2021) as it allows instruments ($\mathbf{Z}_5$) and M-structure colliders ($M_3 \in \mathbf{Z}_2$; Figure A.3) to be included in the adjustment set (Ding & Miratrix, 2014). The *disjunctive cause criterion* is an intermediate approach between the common cause and pretreatment criteria (VanderWeele & Shpitser, 2011). This criterion retains covariates that are causal for exposure, outcome, or both (i.e., $\mathbf{Z}_1$, $\mathbf{Z}_4$, and $\mathbf{Z}_5$). Adjusting only for $\mathbf{Z}_1$ and $\mathbf{Z}_4$ has also been advocated (Brookhart et al., 2006), as 1) unnecessarily adjusting for $\mathbf{Z}_5$ raises risks of variance inflation and bias amplification while 2) adjusting for $\mathbf{Z}_4$ can improve causal estimate precision without impacting bias. We refer to this approach as the *outcome criterion*. The *generalized adjustment criterion* (Perkovic et al., 2015) extends the sufficient but not necessary *generalized backdoor criterion* (Maathuis & Colombo, 2015) to provide a unified criterion for necessary and sufficient adjustment sets that applies to DAGs, maximum ancestral graphs (MAGs), completed partially directed acyclic graphs (CPDAGs), and partial ancestral graphs (PAGs).

C APPENDIX: EXTENDED PARTITION DEFINITIONS

C.1 Partition $\mathbf{Z}_5$ (Instrumental Variables)

Instrumental variable methods have been used heavily in econometrics (Imbens, 2014) and epidemiology (Hernán & Robins, 2006; Labrecque & Swanson, 2018) for causal effect estimation in the presence of latent confounding. The present work explores an additional way to relate instrumental variables to the problem of confounding, where the marginal independence between instruments and confounders is exploited to detect confounders in unknown causal structures. We define an instrument as any variable that meets the criteria enumerated in Definition C.1. We then claim Proposition C.2 about the relations among $\mathbf{Z}_1$ and $\mathbf{Z}_5$, as a theoretical basis for sufficient condition C3. Proof of Proposition C.2 follows from Propositions E.17 and E.19.

Definition C.1 (Instrumental variable, Lousdal (2018))**.** Any instrument Z_5 meets the following criteria:

1. *Relevance assumption*: Z_5 is causal for exposure X.
2. *Exclusion restriction*: The effect of instrument Z_5 on outcome Y is fully mediated by X.
3. *Exchangeability assumption*: Z_5 and Y do not share a common cause.

Proposition C.2. *Any instrument Z_5 in $\mathbf{Z}$ will meet the following criteria with respect to at least one confounder Z_1 on every backdoor path in $\mathcal{G}_{XY\mathbf{Z}}$.*

1. *Z_5 and Z_1 are marginally independent.*
2. *Z_5 and Z_1 are conditionally dependent given X.*

C.2 Partition $\mathbf{Z}_4$

To our knowledge, partition $\mathbf{Z}_4$ has been significantly less characterized and less utilized in the causal inference literature than confounders ($\mathbf{Z}_1$), colliders ($\mathbf{Z}_2$), mediators ($\mathbf{Z}_3$), and instrumental variables ($\mathbf{Z}_5$). Limited reference has been made to this partition under the term *pure prognostic variables* (Hahn & Herren, 2022). We elaborate on our definition of $\mathbf{Z}_4$ below.

Definition C.3 (Partition $\mathbf{Z}_4$)**.** Partition $\mathbf{Z}_4$ encompasses all non-descendants of Y that are marginally dependent on Y but marginally independent of X (Table 1). Given this definition, we observe that any Z_4 participates in a v-structure $X \cdots \rightarrow Y \leftarrow \cdots Z_4$. This implies the following:

1. X cannot share active paths with any Z_4. Thus, X can share no common causes with any Z_4.
2. $\mathbf{Z}_4$ is conditionally dependent on X given Y. This implicitly requires that X and Y are marginally dependent, though they may not be directly adjacent in $\mathcal{G}_{XY\mathbf{Z}}$.

D APPENDIX: METHOD

D.1 High-level overview of Algorithm 1

Here, we present the pseudocode for Algorithm 1 and a high-level explanation of each step.

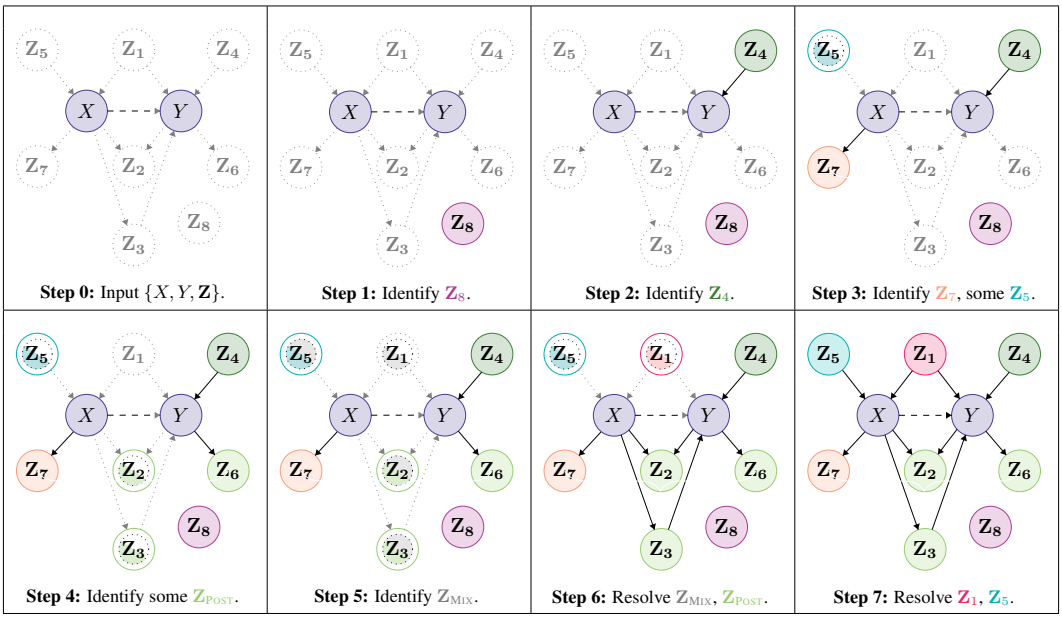

Table D.1: Schematic of Algorithm 1. The exposure-outcome pair $\{X, Y\}$ serves as a nucleus around which LDP assembles a partial causal graph. Each step reveals additional information about the partitions of $\mathbf{Z}$. Nodes that are fully colored are fully discovered by Algorithm 1. Partial coloring denotes partial knowledge, and no coloring denotes no knowledge.

Table D.1 provides a visual walk-through of Algorithm 1 as knowledge is progressively obtained at each step. In plain English, the steps of Algorithm 1 are as follows.

Step 1 $\mathbf{Z}_8$ is discovered using prior knowledge of $\{X, Y\}$ only.
Step 2 $\mathbf{Z}_4$ is discovered using prior knowledge of $\{X, Y\}$ only.
Step 3 $\mathbf{Z}_7$ is discovered using prior knowledge of $\{X, Y\}$ only. Some or all of $\mathbf{Z}_5$ can also be discovered at this step for some graphical structures. In theory, this occurs when $|\mathbf{Z}_1| = 0$.

In practice, we have occasionally observed this happening under small finite samples even when $|\mathbf{Z}_1| > 0$.

Step 4 A fraction of $\mathbf{Z}_{\text{POST}}$ is discovered, providing complete knowledge of $\mathbf{Z}_6$ and partial knowledge of $\mathbf{Z}_2$ and $\mathbf{Z}_3$. This step leverages prior knowledge of $\mathbf{Z}_4$ that was obtained programmatically at Step 2.

Step 5 $\mathbf{Z}_{\text{MIX}}$ is temporarily aggregated, providing partial knowledge of $\mathbf{Z}_1$, $\mathbf{Z}_2$, $\mathbf{Z}_3$, and $\mathbf{Z}_5$. $\mathbf{Z}_{\text{MIX}}$ is a transient superset that is used to differentiate $\mathbf{Z}_1$ and $\mathbf{Z}_5$ from $\mathbf{Z}_{\text{POST}}$ in Step 6. The partitions that can be represented in $\mathbf{Z}_{\text{MIX}}$ will depend on whether sufficient condition C1 is violated or not.

Step 6 Knowledge of $\mathbf{Z}_{\text{POST}}$ is complete. $\mathbf{Z}_{\text{MIX}}$ is fully disaggregated, providing final partition labels for a fraction of members and moving others to superset $\mathbf{Z}_{1,5}$. In the process, a fraction of $\mathbf{Z}_1$ is placed in $\mathbf{Z}_1$. At this step, we union the $\mathbf{Z}_{\text{MIX}}$ discovered in Step 5 with the $\mathbf{Z}_{5,7}$ discovered at Step 3. This is under the assumption that some $\mathbf{Z}_5$ might have been discovered at Step 3, and will need to be distinguished from $\mathbf{Z}_7$. Thus, this step serves as a final check on the set-purity of $\mathbf{Z}_7$, and knowledge of $\mathbf{Z}_7$ is complete.

Step 7 $\mathbf{Z}_1$ and $\mathbf{Z}_5$ are fully differentiated from each other. This step tests whether a member of the superset $\mathbf{Z}_{1,5}$ is marginally dependent on any known Z_1. All previously known Z_1 are those that are directly adjacent to Y. Z_1 that are left to be discovered are those with indirect active paths to Y. Even when sufficient condition C1 is violated, no Z_5 will ever be dependent on a Z_1 that is directly adjacent to Y. However, all Z_1 will be marginally dependent on at least one Z_1 that is adjacent to Y.

Graphical and Parametric Assumptions We assume the causal Markov condition, faithfulness, and acyclicity. Importantly, variables in $\mathbf{Z}$ are *not* assumed to be exclusively pretreatment (Figure A.2), and we do *not* place sparsity constraints on $\mathcal{G}_{XY\mathbf{Z}}$. We do not make assumptions about the distributional forms of variables nor the functional forms of their causal relations. No specific functional causal model is imposed, freeing LDP from the identifiability assumptions of the post-nonlinear additive noise model (PNL) (Zhang & Hyvarinen, 2009). Unlike the PNL and its special cases (e.g., Hoyer et al. 2008; Shimizu et al. 2006), LDP is identifiable in the linear-Gaussian case.

The Exposure-Outcome Pair The only prior knowledge of $\mathcal{G}_{XY\mathbf{Z}}$ that is required by LDP concerns the exposure-outcome relationship. While the causal effect of X on Y can be of arbitrary strength or null, we assume that 1) X and Y are marginally dependent and 2) Y cannot be a direct nor indirect cause of X due to the acyclicity assumption. All proofs and experiments assume univariate X and Y.

D.2 Suffcient Conditions for Identifiability

Sufficient Conditions for Partition Accuracy Given an independence oracle, we claim the following *sufficient* (but not necessary) conditions for asymptotically correct partitioning:

C1 The absence of inter-partition active paths that are not fully mediated by $\{X, Y\}$ (Definition 2.2).

C2 The existence of at least one Z_4. Given Condition C1, all Z_2 (if any exist) will be marginally dependent on such a Z_4 and will be identifiable by LDP. This in turn guarantees that all backdoor paths will be blocked by the conditioning set in Step 5 of Algorithm 1, which is used to discover $\mathbf{Z}_5$. This condition is testable at line 9 of Algorithm 1.

C3 Every true Z_1 forms a v-structure at X with at least one other variable $Z \in \mathbf{Z}$ ($Z \cdots \rightarrow X \leftarrow \cdots Z_1$) such that $Z \perp\!\!\!\perp Z_1 \wedge Z \not\perp\!\!\!\perp Z_1 | X$. By definition, variable Z can be either in $\mathbf{Z}_5$ or $\mathbf{Z}_1$. Given C1, $\mathbf{Z}_5$ shares no active paths with $\mathbf{Z}_1$ and thus all of $\mathbf{Z}_1$ is marginally independent of $\mathbf{Z}_5$. If $|\mathbf{Z}_5| = 0$, the existence of at least two non-overlapping backdoor paths in $\mathcal{G}_{XY\mathbf{Z}}$ can satisfy this condition.

C4 Causal sufficiency in $\mathcal{G}_{XY\mathbf{Z}}$.

These sufficient conditions can be weakened in some settings. For example, existence of Z_4 is unnecessary when $\mathbf{Z}$ is pretreatment. We provide theoretical and empirical results demonstrating robustness to certain violations of Condition C1, e.g., partition correctness on M-structures (Table G.6). Causal sufficiency may not be a necessary condition, as LDP is robust to some forms of latent confounding in $\mathcal{G}_{XY\mathbf{Z}}$ (Section 5).

Algorithm 1 *Local Discovery by Partitioning (LDP)*

input $\{X, Y\}, \mathbf{Z}$, independence test of choice.
output Partitions of $\mathbf{Z}$:

- $\mathbf{Z}_1$: Confounders for $\{X, Y\}$.
- $\mathbf{Z}_4$: Non-descendants of Y s.t. $Y \not\perp\!\!\!\perp Z_4 \wedge X \perp\!\!\!\perp Z_4$.
- $\mathbf{Z}_5$: Instrumental variables.
- $\mathbf{Z}_7$: Descendants of X where $Y \perp\!\!\!\perp Z_7 \mid X$.
- $\mathbf{Z}_8$: Variables with no active paths to $\{X, Y\}$.
- $\mathbf{Z}_{\text{POST}}$: Post-treatment subset $\{\mathbf{Z}_2, \mathbf{Z}_3, \mathbf{Z}_6\}$.

Copy $\mathbf{Z}' \leftarrow \mathbf{Z}$
for all $Z \in \mathbf{Z}'$ **do**
▷ STEP 1: TEST FOR $\mathbf{Z}_8$
  if $X \perp\!\!\!\perp Z$ and $Y \perp\!\!\!\perp Z$ **then**
    $Z \in \mathbf{Z}_8$, $\mathbf{Z}' \leftarrow \mathbf{Z}' \setminus Z$
▷ STEP 2: TEST FOR $\mathbf{Z}_4$
  if $X \perp\!\!\!\perp Z$ and $X \not\perp\!\!\!\perp Z|Y$ **then**
    $Z \in \mathbf{Z}_4$, $\mathbf{Z}' \leftarrow \mathbf{Z}' \setminus Z$
▷ STEP 3: TEST FOR $\mathbf{Z}_{5,7}$
  if $Y \not\perp\!\!\!\perp Z$ and $Y \perp\!\!\!\perp Z|X$ **then**
    $Z \in \mathbf{Z}_{5A,7}$, $\mathbf{Z}' \leftarrow \mathbf{Z}' \setminus Z$

▷ STEP 4: TEST FOR $\mathbf{Z}_{\text{POST}}$
if $|\mathbf{Z}_4| > 0$ **then**
  for all $Z \in \mathbf{Z}'$ **do**
    if $\exists\, Z_4$: $Z \not\perp\!\!\!\perp Z_4$ or $Z \perp\!\!\!\perp Z_4|X \cup Y$ **then**
      $Z \in \mathbf{Z}_{2,3,6} \in \mathbf{Z}_{\text{POST}}$
$\mathbf{Z}' \leftarrow \mathbf{Z}' \setminus \mathbf{Z}_{\text{POST}}$

▷ STEP 5: TEST FOR $\mathbf{Z}_{\text{MIX}}$
for all $Z \in \mathbf{Z}'$ **do**
  if $Y \not\perp\!\!\!\perp Z$ and $Y \perp\!\!\!\perp Z|X \cup \mathbf{Z}' \setminus Z$ **then**
    $Z \in \mathbf{Z}_{1,2,3,5} \in \mathbf{Z}_{\text{MIX}}$
$\mathbf{Z}' \leftarrow \mathbf{Z}' \setminus \mathbf{Z}_{\text{MIX}}$

▷ STEP 6: SPLIT $\mathbf{Z}_{\text{MIX}}$ BETWEEN $\mathbf{Z}_{1,5}$, $\mathbf{Z}_7$, $\mathbf{Z}_{\text{POST}}$
$\mathbf{Z}_{\text{MIX}} \leftarrow \mathbf{Z}_{\text{MIX}} \cup \mathbf{Z}_{5,7}$
if $|\mathbf{Z}_{\text{MIX}}| > 0$ **then**
  for all $Z \in \mathbf{Z}'$ **do**
    if $\exists\, Z_{\text{MIX}}$: $Z_{\text{MIX}} \perp\!\!\!\perp Z$ and $Z_{\text{MIX}} \not\perp\!\!\!\perp Z|X$ **then**
      $Z \in \mathbf{Z}_1$, $Z_{\text{MIX}} \in \mathbf{Z}_{1,5} \notin \mathbf{Z}_{\text{MIX}}$
    else
      $Z \in \mathbf{Z}_3 \in \mathbf{Z}_{\text{POST}}$
  for all $Z_{\text{MIX}} \in \mathbf{Z}_{\text{MIX}}$ **do**
    if $\exists\, Z_{1,5}$: $Z_{1,5} \perp\!\!\!\perp Z_{\text{MIX}}$ **then**
      $Z_{\text{MIX}} \in \mathbf{Z}_1$
    else
      $Z_{\text{MIX}} \in \mathbf{Z}_{2,3} \in \mathbf{Z}_{\text{POST}}$

▷ STEP 7: FINALIZE $\mathbf{Z}_1$ AND $\mathbf{Z}_5$
if $|\mathbf{Z}_{1,5}| > 0$ and $|\mathbf{Z}_1| > 0$ **then**
  for all $Z_{1,5} \in \mathbf{Z}_{1,5}$ **do**
    if $\exists\, Z_1 \in \mathbf{Z}_1$: $Z_{1,5} \not\perp\!\!\!\perp Z_1$ **then**
      $Z_{1,5} \in \mathbf{Z}_1$
    else
      $Z_{1,5} \in \mathbf{Z}_5$

{not identifiable} $\leftarrow \mathbf{Z}'$
return Partitions of $\mathbf{Z}$ and {not identifiable}.

Given these sufficient conditions, we obtain Theorem D.1. Proof is provided in Appendix E.2.

Theorem D.1 (Correctness of Algorithm 1)**.** *Given $\{X, Y, \mathbf{Z}\}$, an independence oracle, and Conditions C1-C4, Algorithm 1 is guaranteed to output a correct partition of $\mathbf{Z}$ that represents the local*

subgraph of $\mathcal{G}_{XY\mathbf{Z}}$ surrounding $\{X, Y\}$, where each $Z \in \mathbf{Z}$ is defined solely by its relation to $\{X, Y\}$.

Sufficient Conditions for Valid Adjustment Set Identification In Theorem D.2, we establish that Conditions C2-C4 are sufficient for valid adjustment set identification, eliminating the need for Condition C1. Proof is provided in Appendix E.3.

Theorem D.2 (LDP returns valid adjustment sets)**.** *Given $\{X, Y, \mathbf{Z}\}$, an independence oracle, and Conditions C2-C4, Algorithm 1 is guaranteed to return a valid adjustment set under the backdoor criterion.*

E APPENDIX: PROOFS

In this section, we prove the three main theorems presented in this work: Theorem 2.1 (Section E.1), Theorem D.1 (Section E.2), and Theorem D.2 (Section E.3). In Section D.1, we present a high-level discussion of how Algorithm 1 works. We assume access to an independence oracle for all proofs. We assume that sufficient conditions C1–C4 are met, unless it is explicitly stated that they can be weakened.

E.1 Partitions of Z

We prove Theorem 2.1, which states that any $\mathbf{Z}$ can be partitioned into eight mutually exclusive subsets (of cardinality greater than or equal to zero) defined solely by their relation to exposure X and outcome Y.

Proof. To prove Theorem 2.1, we first define every type of active path from a candidate $Z \in \mathbf{Z}$ to X or Y that can possibly arise in $\mathcal{G}_{XY\mathbf{Z}}$ (Table E.1). These paths can be direct adjacencies (i.e., length-1 paths) or indirect active paths of arbitrary length. In Table E.2, we express every possible combination of path types that can coincide for a single Z.

Definition E.1 (Active path types in $\mathcal{G}_{XY\mathbf{Z}}$)**.** We exhaustively enumerate the types of active paths that can lie between Z and $\{X, Y\}$ in Table E.1.

TYPE	ACTIVE PATH RELATIVE TO X	ACTIVE PATH RELATIVE TO Y
1	None (or none that do not pass through Y).	None (or none that do not pass through X).
2	$Z \to \cdots \to X$ path(s) and no other types.	$Z \to \cdots \to Y$ path(s) not passing through X and no other types.
3	$X \to \cdots \to Z$ path(s) not passing through Y and no other types.	$Y \to \cdots \to Z$ path(s) and no other types.
4	$Z \leftarrow \ldots Z' \cdots \to X$ path(s) and no other types.	$Z \leftarrow \ldots Z' \cdots \to Y$ path(s) and no other types.
5	Type 2 path(s) and Type 4 path(s).	Type 2 path(s) and Type 4 path(s).
6	Type 3 path(s) and Type 4 path(s).	Type 3 path(s) and Type 4 path(s).

Table E.1: Exhaustive enumeration of the types of active paths that can lie between any given Z and $\{X, Y\}$. In confounded paths, Z' denotes an additional variable in $\mathbf{Z}$ that may or may not belong to the same partition as Z. Note that Type 1 and Type 2 paths cannot coincide for a single Z, as this would induce a cycle.

Next, we prove the exhaustivity of the partition taxonomy proposed in Theorem 2.1 (Table 1). Every cell in Table E.2 that does not violate acyclicity contains a partition. As this table represents all possible path type combinations for a single Z in $\mathcal{G}_{XY\mathbf{Z}}$, the lack of empty cells indicates exhaustivity in our taxonomy. Finally, we prove the mutual exclusivity of partitions in our taxonomy. The mutual exclusivity of each partition is supported by the fact that each cell of Table E.2 contains a single partition, such that the pattern of allowable active path types from Z to X and Y is unique for each partition. Thus, we conclude that Table 1 presents an exhaustive and mutually exclusive taxonomy of partitions for any arbitrary $\mathbf{Z}$. □

E.2 Correctness of Algorithm 1

Here, we prove Theorem D.1. We provide theoretical guarantees that partitions returned by LDP are asymptotically correct for any arbitrary $\mathbf{Z}$ under sufficient conditions C1–C4. Proof of Theorem D.1 follows from proofs of Lemmas E.2–E.9, which prove correctness for each step of Algorithm 1

		Relative to X					
		Type 1	Type 2	Type 3	Type 4	Type 5	Type 6
Relative to Y	Type 1	$\mathbf{Z}_8$	$\mathbf{Z}_5$	$\mathbf{Z}_7$	$\mathbf{Z}_5$	$\mathbf{Z}_5$	$\mathbf{Z}_7$
	Type 2	$\mathbf{Z}_4$	$\mathbf{Z}_1$	$\mathbf{Z}_3$	$\mathbf{Z}_1$	$\mathbf{Z}_1$	$\mathbf{Z}_3$
	Type 3	$\mathbf{Z}_6$	$\emptyset$	$\mathbf{Z}_2$	$\mathbf{Z}_2$	$\emptyset$	$\mathbf{Z}_2$
	Type 4	$\mathbf{Z}_4$	$\mathbf{Z}_1$	$\mathbf{Z}_2$	$\mathbf{Z}_{2\in\mathbf{M}_3}$	$\mathbf{Z}_1$	$\mathbf{Z}_2$
	Type 5	$\mathbf{Z}_4$	$\mathbf{Z}_1$	$\mathbf{Z}_3$	$\mathbf{Z}_1$	$\mathbf{Z}_{1\in\mathbf{B}_3}$	$\mathbf{Z}_3$
	Type 6	$\mathbf{Z}_6$	$\emptyset$	$\mathbf{Z}_2$	$\mathbf{Z}_2$	$\emptyset$	$\mathbf{Z}_2$

Table E.2: Combinations of active path types relative to X and Y. Cells contain partitions that can participate in the given combination of active path types. The empty set ($\emptyset$) indicates that this combination of active path types is forbidden under the acyclicity constraint. A subscript of $\mathbf{M}_3$ indicates that this variable is an M-collider, while a subscript of $\mathbf{B}_3$ denotes a butterfly-type confounder (Figure A.3).

sequentially. In footnotes, we acknowledge certain partitioning behaviors that occur when condition C1 is violated. However, these acknowledgements are non-exhaustive.

Lemma E.2 (Step 1 of Algorithm 1)**.** $X \perp\!\!\!\perp Z \wedge Y \perp\!\!\!\perp Z \iff Z \in \mathbf{Z}_8$.

Proof. Step 1 of Algorithm 1 correctly identifies $\mathbf{Z}_8$. This subset of $\mathbf{Z}$ is the most trivial to identify, as it is does not share an active path with either exposure nor outcome in $\mathcal{G}_{XY\mathbf{Z}}$. By definition, any $Z_8 \in \mathbf{Z}_8$ is marginally independent of X and marginally independent of Y. Additionally, no candidate $Z \in \mathbf{Z} \setminus \mathbf{Z}_8$ is marginally independent of both X and Y. Thus, any $Z \in \mathbf{Z}$ satisfying $X \perp\!\!\!\perp Z \wedge Y \perp\!\!\!\perp Z$ belongs to $\mathbf{Z}_8$ and can be removed from further consideration. □

Lemma E.3 (Step 2 of Algorithm 1)**.** $X \perp\!\!\!\perp Z \wedge X \not\perp\!\!\!\perp Z|Y \iff Z \in \mathbf{Z}_4$.

Proof. Step 2 of Algorithm 1 correctly identifies $\mathbf{Z}_4$. Variables in $\mathbf{Z}_4$ share an active path with outcome Y in $\mathcal{G}_{XY\mathbf{Z}}$ but not exposure X. For any $Z_4 \in \mathbf{Z}_4$, this results in a v-structure $X \cdots \rightarrow Y \leftarrow \cdots Z_4$.[4] By definition, all such v-structures entail $X \perp\!\!\!\perp Z_4 \wedge X \not\perp\!\!\!\perp Z_4|Y$. Besides $\mathbf{Z}_4$, only $\mathbf{Z}_8$ is marginally independent of X. However, $\mathbf{Z}_8$ is not conditionally dependent on X given Y. Thus, no subset of $\mathbf{Z}$ entails $X \perp\!\!\!\perp Z \wedge X \not\perp\!\!\!\perp Z|Y$ except $\mathbf{Z}_4$. Any variable passing the test in Step 2 is unambiguously a member of $\mathbf{Z}_4$. Further, $\mathbf{Z}_4$ is correctly identified for downstream use in Step 4 to identify $\mathbf{Z}_{\text{POST}}$. □

Lemma E.4 (Step 3 of Algorithm 1)**.** $Y \not\perp\!\!\!\perp Z \wedge Y \perp\!\!\!\perp Z|X \iff Z \in \mathbf{Z}_{5,7}$ *and* X *blocks all backdoor paths between* Z *and* Y *in* $\mathcal{G}_{XY\mathbf{Z}}$.

Proof. Step 3 of Algorithm 1 correctly identifies $\mathbf{Z}_{5,7}$ *when all backdoor paths between* $\mathbf{Z}_{5,7}$ *and* Y *are blocked by* X. We prove both directions of the bidirectional statement by direct proof. There are two conditions under which exposure X will block all backdoor paths between $Z \in \mathbf{Z}_{5,7}$ and outcome Y: 1) $Z \in \mathbf{Z}_7$ for any $\mathcal{G}_{XY\mathbf{Z}}$ and 2) $Z \in \mathbf{Z}_5$ when $\mathcal{G}_{XY\mathbf{Z}}$ does not contain paths in $\mathbf{Z}$ on which X is a collider for $\mathbf{Z}_5$ (i.e., $\mathcal{G}_{XY\mathbf{Z}}$ contains no $\mathbf{Z}_1$). Thus, this test will capture $\mathbf{Z}_7$ under any circumstances but will capture $\mathbf{Z}_5$ only when $\mathcal{G}_{XY\mathbf{Z}}$ is structured such that exposure X blocks all backdoor paths from $\mathbf{Z}_5$ to outcome Y. Further, no subset of $\mathbf{Z}$ will pass the test in Step 3 but $\mathbf{Z}_{5,7}$. $\mathbf{Z}_1$, $\mathbf{Z}_2$, $\mathbf{Z}_3$, $\mathbf{Z}_4$, and $\mathbf{Z}_6$ are parents or effects of Y and thus X cannot block the flow of association between them. $\mathbf{Z}_8$ will not pass this test either, as it is not marginally dependent on Y. Therefore, $Y \not\perp\!\!\!\perp Z \wedge Y \perp\!\!\!\perp Z|X$ if and only if Z is in $\mathbf{Z}_{5,7}$. □

Lemma E.5 (Step 4 of Algorithm 1)**.** *Given execution of prior steps in Algorithm 1,* $\exists\, Z_4$*:* $Z \not\perp\!\!\!\perp Z_4$ *or* $Z \perp\!\!\!\perp Z_4|X \cup Y \iff Z \in \mathbf{Z}_{2,3,6} \in \mathbf{Z}_{\text{POST}}$.

[4]Note that this requires X and Y to be marginally dependent, an assumption made in Section 3.2. $X \not\perp\!\!\!\perp Y$ is true when at least one of the following conditions is true: 1) X is a direct cause of Y, 2) X is an indirect cause of Y through mediators $\mathbf{Z}_3$, and/or 3) X and Y share confounders $\mathbf{Z}_1$.

Proof. Step 4 of Algorithm 1 correctly identifies $\mathbf{Z}_{2,3,6} \in \mathbf{Z}_{\text{POST}}$. This test exploits prior knowledge of $\mathbf{Z}_4$ to identify all of $\mathbf{Z}_2$ and $\mathbf{Z}_6$ in any arbitrary $\mathcal{G}_{XY\mathbf{Z}}$ meeting sufficient conditions C1–C4. Under condition C1, no $\mathbf{Z}_3$ will pass this test by the same logic that $\{\mathbf{Z}_1, \mathbf{Z}_5\}$ will not (as proven below).[5] Note that $\mathbf{Z}_4$, $\mathbf{Z}_7$, and $\mathbf{Z}_8$ have already been identified and removed from further consideration, as has a subset of $\mathbf{Z}_5$ of cardinality greater than or equal to zero. Thus, this test must correctly identify $\mathbf{Z}_2$ and $\mathbf{Z}_6$ and must not incorrectly label these partitions as $\mathbf{Z}_1$ or $\mathbf{Z}_5$. We demonstrate correctness by direct proof of both directions of the bidirectional statement.

Under the assumption that X and Y are marginally dependent (Section 3.2), any member Z of $\{\mathbf{Z}_1, \mathbf{Z}_5\}$ will form a v-structure $Z_4 \cdots \rightarrow Y \leftarrow \cdots Z$, but members of $\{\mathbf{Z}_2, \mathbf{Z}_6\}$ will not (Figure 1). Such a v-structure implies that $Z \perp\!\!\!\perp Z_4$ and $Z \not\perp\!\!\!\perp Z_4 | X \cup Y$. As we seek to identify candidates Z that do not induce such a v-structure, we logically negate these independence statements to test for $\mathbf{Z}_2$ and $\mathbf{Z}_6$. According to De Morgan's Laws, the negation of a conjunction is the disjunction of the negations. This yields the logical equivalence

$$\neg\left[(Z \perp\!\!\!\perp Z_4) \wedge (Z \not\perp\!\!\!\perp Z_4 | X \cup Y)\right] \equiv (Z \not\perp\!\!\!\perp Z_4) \vee (Z \perp\!\!\!\perp Z_4 | X \cup Y). \quad \text{Per De Morgan's Laws.} \tag{1}$$

Thus, when $Z \not\perp\!\!\!\perp Z_4$ or $Z \perp\!\!\!\perp Z_4 | X \cup Y$ is true, we will identify $\{\mathbf{Z}_2, \mathbf{Z}_6\}$ but not $\{\mathbf{Z}_1, \mathbf{Z}_5\}$. Likewise, when $Z \in \mathbf{Z}_{2,6}$, a v-structure $Z_4 \rightarrow Y \leftarrow Z$ will never arise and thus $Z \not\perp\!\!\!\perp Z_4$ or $Z \perp\!\!\!\perp Z_4 | X \cup Y$. □

To support Lemmas E.7-E.9, we introduce Proposition E.6.

Proposition E.6. *For any Z_1 that has an indirect active path to outcome Y, there must exist another Z_1 that is directly adjacent to Y. This extends analogously to indirect active paths between $\mathbf{Z}_1$ and X.*

Lemma E.7 (Step 5 of Algorithm 1)**.** *Given execution of prior steps in Algorithm 1, if $Y \not\perp\!\!\!\perp Z \wedge Y \perp\!\!\!\perp Z | X \cup \mathbf{Z}' \setminus Z$ then $Z \in \mathbf{Z}_{1,2,3,5} \in \mathbf{Z}_{\text{MIX}}$, and all backdoor paths between $\mathbf{Z}_{\text{MIX}}$ and Y are blocked by X and the members of $\mathbf{Z}$ that have not yet been labeled.*

Proof. Step 5 of Algorithm 1 correctly identifies $\mathbf{Z}_{\text{MIX}}$. Here, will assume that $\mathbf{Z}_5$ was not yet discovered at Step 3. We will prove that the conditioning set used in Step 5 correctly blocks all backdoor paths between $\mathbf{Z}_{\text{MIX}}$ and Y. Given sufficient conditions C1–C4, $\mathbf{Z}_2$, $\mathbf{Z}_4$, $\mathbf{Z}_6$, and $\mathbf{Z}_8$ have been previously identified and removed from further consideration.[6] Thus, we assume that only $\mathbf{Z}_1$, $\mathbf{Z}_3$, and $\mathbf{Z}_5$ are remaining in $\mathbf{Z}'$. By conditioning on $\mathbf{Z}' \setminus Z$, confounding for $\{X, Y\}$ is blocked due to the inclusion of all $\mathbf{Z}_1 \in \mathbf{Z}'$. Thus, conditioning on $X \cup \mathbf{Z}' \setminus Z$ blocks all causal and non-causal association between Z and Y. For all $Z \in \mathbf{Z}_5$, $Y \perp\!\!\!\perp Z | X \cup \mathbf{Z}' \setminus Z$. For any $Z \in \mathbf{Z}_1$ or $Z \in \mathbf{Z}_3$ that is not directly adjacent to Y, $Y \perp\!\!\!\perp Z | X \cup \mathbf{Z}' \setminus Z$. All members of $\mathbf{Z}_1$ and $\mathbf{Z}_3$ that are adjacent to Y will proceed to be identified at Step 6. Thus, $\mathbf{Z}_{\text{MIX}}$ will consist of $\mathbf{Z}_5$, a fraction of $\mathbf{Z}_1$ (which may be the empty set), and a fraction of $\mathbf{Z}_3$ (which may be the empty set). □

Lemma E.8 (Step 6 of Algorithm 1)**.** *Let $\mathbf{Z}_{\text{MIX}} = \mathbf{Z}_{\text{MIX}} \cup \mathbf{Z}_{5,7}$. Given execution of prior steps in Algorithm 1, if $\exists\ Z_{\text{MIX}} \in \mathbf{Z}_{\text{MIX}}$ such that $Z_{\text{MIX}} \perp\!\!\!\perp Z$ and $Z_{\text{MIX}} \not\perp\!\!\!\perp Z | X$ then $Z \in \mathbf{Z}_1$ and $Z_{\text{MIX}} \in \mathbf{Z}_{1,5}$. Else, $Z \in \mathbf{Z}_3 \in \mathbf{Z}_{\text{POST}}$. After execution of these tests, we loop through the remaining $\mathbf{Z}_{\text{MIX}}$ again. If $\exists\ Z_{1,5} \in \mathbf{Z}_{1,5}$ such that $Z_{1,5} \perp\!\!\!\perp Z_{\text{MIX}}$ and $Z_{1,5} \not\perp\!\!\!\perp Z_{\text{MIX}} | X$, then $Z_{\text{MIX}} \in \mathbf{Z}_1$. Else, $Z_{\text{MIX}} \in \mathbf{Z}_{2,3} \in \mathbf{Z}_{\text{POST}}$.*

Proof. Step 6 of Algorithm 1 correctly differentiates $\mathbf{Z}_1$, $\mathbf{Z}_{1,5}$, $\mathbf{Z}_7$, *and* $\mathbf{Z}_{\text{POST}}$. This step relies on prior knowledge of $\mathbf{Z}_{\text{MIX}}$, which is gained programmatically through Steps 3 and 5. Under sufficient conditions C1–C4, $\mathbf{Z}_{\text{MIX}}$ initially contains $\mathbf{Z}_5$ and the members of $\mathbf{Z}_1$ and $\mathbf{Z}_3$ that are not adjacent to Y. At Step 6, we begin by unioning $\mathbf{Z}_{\text{MIX}}$ with $\mathbf{Z}_{5,7}$ as a safeguard in case any member of $\mathbf{Z}_5$ was lumped with $\mathbf{Z}_7$ at Step 3.

Step 6 exploits the presence of v-structures $Z \cdots \rightarrow X \leftarrow \cdots Z_1$ in $\mathcal{G}_{XY\mathbf{Z}}$. For any $\mathcal{G}_{XY\mathbf{Z}}$ (even when sufficient conditions are not met), the variables that can form such a v-structure with a Z_1 are

[5] If sufficient condition C1 is violated, a Z_3 may be captured at this step if it is marginally dependent on any Z_4. Further, this violation can cause Step 4 to miss members of $\mathbf{Z}_2$ that are not descendants of Y (as discussed throughout Section E.3).

[6] If sufficient condition C1 is violated, members of $\mathbf{Z}_2$ that were not marginally dependent on any Z_4 (and thus not identified at Step 4) could be placed in $\mathbf{Z}_{\text{MIX}}$ at Step 5 instead. We prove in Section E.3 that the presence of $\mathbf{Z}_2$ in $\mathbf{Z}_{\text{MIX}}$ does not undermine the validity of the adjustment set returned by Algorithm 1.

1) a Z_5 or 2) another Z_1 that does not share an active path with the first. The only other variables that are marginally independent of $\mathbf{Z}_1$ are $\mathbf{Z}_4$ and $\mathbf{Z}_8$, both of which were previously identified.

First, we prove the first phase of Step 6. Under sufficient condition C1, $\mathbf{Z}_5 \cdots \rightarrow X \leftarrow \cdots \mathbf{Z}_1$ for all $\{\mathbf{Z}_1, \mathbf{Z}_5\}$. This means that all of $\mathbf{Z}_5$ is marginally independent of $\mathbf{Z}_1$, but is conditionally dependent on $\mathbf{Z}_1$ given X. As described in sufficient condition C3, the existence of at least two non-overlapping backdoor paths in $\mathcal{G}_{XY\mathbf{Z}}$ can also enable some Z_1 to form a v-structure at X with another Z_1. Thus, when a v-structure $Z_{\text{MIX}} \cdots \rightarrow X \leftarrow \cdots Z$ is detected, then Z must be in $\mathbf{Z}_1$ and Z_{MIX} must be in $\mathbf{Z}_{1,5}$. By extension, Z_{MIX} is not in $\mathbf{Z}_{\text{POST}}$ nor $\mathbf{Z}_7$, and can be removed from the latter if it had been placed there at Step 3. Else, Z must be in $\mathbf{Z}_{\text{POST}}$.

Finally, we prove the second phase of Step 6. Variables still in $\mathbf{Z}_{\text{MIX}}$ must be tested to distinguish the remaining members in $\mathbf{Z}_1$ from those in $\mathbf{Z}_{\text{POST}}$. Any ground truth member of $\mathbf{Z}_1$ that remains in $\mathbf{Z}_{\text{MIX}}$ at this point must be marginally dependent on all previously discovered Z_1, otherwise these would have already been placed in $\mathbf{Z}_{1,5}$. By this point, all of $\mathbf{Z}_5$ is now contained in $\mathbf{Z}_{1,5}$. Under sufficient condition C1, $\mathbf{Z}_1 \perp\!\!\!\perp \mathbf{Z}_5$ but $\mathbf{Z}_{\text{POST}} \not\perp\!\!\!\perp \mathbf{Z}_5$. Thus, testing $\mathbf{Z}_{\text{MIX}}$ against $\mathbf{Z}_{1,5}$ for marginal independence will differentiate the remaining $\mathbf{Z}_1 \in \mathbf{Z}_{\text{MIX}}$ from the remaining $\mathbf{Z}_{\text{POST}} \in \mathbf{Z}_{\text{MIX}}$.[7] □

Lemma E.9 (Step 7 of Algorithm 1)**.** *Given execution of prior steps in Algorithm 1, if* $\exists\ Z_1 \in \mathbf{Z}_1$ *and* $Z_{1,5} \in \mathbf{Z}_{1,5}$ *such that* $Z_{1,5} \not\perp\!\!\!\perp Z_1$*, then* $Z_{1,5} \in \mathbf{Z}_1$*. Else,* $Z_{1,5} \in \mathbf{Z}_5$*.*

Proof. Step 7 of Algorithm 1 correctly differentiates $\mathbf{Z}_1$ *from* $\mathbf{Z}_5$*.* This step handles cases exemplified by node B_1 in the butterfly structure of Figure A.3, which can have arbitrarily long, indirect, yet active paths to Y. During Step 5, the conditioning set $\{\mathbf{Z}' \setminus Z\}$ contains all $\mathbf{Z}_1$, among other variables. For a B_1-type confounder, this conditioning set blocks all backdoor paths to Y, triggering the test to label the node as a member of $\mathbf{Z}_{\text{MIX}}$. To detect such a case, observe that B_1-type confounders have marginal dependence on the subset of $\mathbf{Z}_1$ that was discovered at Step 6. All Z_1 previously discovered at Step 6 are directly adjacent to Y. Under sufficient condition C1, all of $\mathbf{Z}_5$ is marginally independent of $\mathbf{Z}_1$. Even when sufficient condition C1 is violated, no Z_5 will ever be dependent on a Z_1 that is directly adjacent to Y. Therefore, any $Z_{1,5}$ that is marginally dependent on at least one member of $\mathbf{Z}_1$ discovered at Step 6 must be in $\mathbf{Z}_1$. If such marginal dependence is not detected between a given $Z_{1,5}$ and any member of $\mathbf{Z}_1$ discovered at Step 6, then $Z_{1,5} \in \mathbf{Z}_5$ instead. □

E.3 Validity of adjustment sets

Here we prove Theorem D.2, which states that LDP returns valid adjustment sets when sufficient conditions C2–C4 are satisfied. We will show that Theorem D.2 holds even when sufficient condition C1 is violated, i.e., $\mathcal{G}_{XY\mathbf{Z}}$ contains *inter-partition active paths* (Definition 2.2). Recall the definition of a valid adjustment set under the backdoor criterion (Definition B.4). Let $\mathbf{A}_{XY}$ be an adjustment set for $\{X, Y\}$ that does not contain $\{X, Y\}$. We say that $\mathbf{A}_{XY}$ is valid if

Item 1 $\mathbf{A}_{XY}$ contains no descendants of X; and
Item 2 $\mathbf{A}_{XY}$ blocks all backdoor paths from X to Y (Peters et al., 2017).

The set $\mathbf{A}_{XY}$ returned by LDP is synonymous with partition $\mathbf{Z}_1$. To prove that $\mathbf{Z}_1$ is a valid adjustment set even when sufficient condition C1 is violated, we must prove that both *Item 1* and *Item 2* always hold for the $\mathbf{Z}_1$ returned by Algorithm 1.

We begin by proving that the identification of $\mathbf{Z}_4$ and $\mathbf{Z}_5$ is guaranteed in this setting (Theorem E.10), as the existence of these partitions is sufficient for discovering $\mathbf{Z}_1$ under conditions C2 and C3. We prove *Item 1* with Theorem E.14 by demonstrating that all Z placed in $\mathbf{Z}_1$ are non-descendants of X, even if their partition label is incorrect. By extension, proving *Item 1* guarantees that no causal path from X to Y will be blocked by the $\mathbf{Z}_1$ returned by LDP. We prove *Item 2* with Theorem E.15, which states that LDP blocks all backdoor paths even when condition C1 is violated.

[7] Per the footnote above, some members of $\mathbf{Z}_2$ could eventually be placed in $\mathbf{Z}_1$ when sufficient condition C1 is violated. In Section E.3, we prove that this occurrence does not negate the validity of the adjustment set returned by LDP.

E.3.1 Identification of $\mathbf{Z}_4$ and $\mathbf{Z}_5$ is guaranteed

In order to catch $\mathbf{Z}_5$, LPD must first catch all $\mathbf{Z}_2$ that, when conditioned on, open an active path from X to Y. In order to catch such $\mathbf{Z}_2$ at Step 4, LDP must first catch $\mathbf{Z}_4$ at Step 2. Therefore, we must prove that each of these steps are unaffected by inter-partition active paths.

Theorem E.10 (Identification of $\mathbf{Z}_4$ and $\mathbf{Z}_5$ is guaranteed when sufficient condition C1 is violated)**.**

Proof of Theorem E.10 follows from Lemmas E.11 and E.13.

Lemma E.11 (Discovery of $\mathbf{Z}_4$ is guaranteed at Step 2)**.**

Proposition E.12. *If two variables are marginally or conditionally dependent and the conditioning set remains unchanged, the addition of a new active path in $\mathcal{G}_{XY\mathbf{Z}}$ cannot render them independent.*

Proof. The test at Step 2 of Algorithm 1 relies only on $\{X, Y, Z\}$ for a given candidate Z. No valid inter-partition active paths can cause $\mathbf{Z}_4$ to be marginally dependent on X, as this would violate the definition of $\mathbf{Z}_4$ (Definition C.3). Per Proposition E.12, no valid inter-partition active paths can negate the conditional dependence of $\mathbf{Z}_4$ and X as the conditioning set remains unchanged. Therefore, the discovery of $\mathbf{Z}_4$ remains unaffected by inter-partition active paths. □

Lemma E.13 (Discovery of $\mathbf{Z}_5$ is guaranteed at Step 5)**.**

Proof. To prove Lemma E.13, we will show that each phase to the discovery of $\mathbf{Z}_5$ at Steps 5, 6, and 7 is not affected by violations of sufficient condition C1.

First, we address potential impacts at Step 5. A subset of $\mathbf{Z}_2$ is the only subpartition whose inclusion in the conditioning set at Step 5 can prevent downstream detection of $\mathbf{Z}_5$. This problematic subset of $\mathbf{Z}_2$ must be

1. marginally independent of all of $\mathbf{Z}_4$, otherwise it would be identified at Step 4; and
2. able to render an inactive path between X and Y active by its inclusion in the conditioning set at Step 5.

We will show that no Z_2 that is undiscovered by Step 4 can meet the second condition. Observe that any Z_2 can either be descended from Y or share only a confounded path with Y. To meet the first condition above, a Z_2 cannot be descended from Y. Any Z_2 that is a descendant of Y will always be discovered at Step 4 regardless of the other paths it lies on, as it will always be marginally dependent on $\mathbf{Z}_4$.

We next consider the subset of $\mathbf{Z}_2$ that only shares confounded paths with Y. For a Z_2 to meet the second condition above, any path between X and Y that is opened by conditioning on this Z_2 *must not be re-blocked* by the rest of the conditioning set used in Step 5. Thus, we must prove that any Z_2 that is not identified at Step 4 must be on a path to Y that is *re-blocked* by the other members of the conditioning set at Step 5.

Consider the types of confounded paths that $\mathbf{Z}_2$ can share with Y. Any confounder for $\{Z_2, Y\}$ can only belong to $\mathbf{Z}_1$, $\mathbf{Z}_3$, $\mathbf{Z}_4$, or $\mathbf{Z}_6$. This follows from Table E.2, which states that the only partitions that can have an edge entering Y are:

1. Type 2 paths to Y: $\mathbf{Z}_1$, $\mathbf{Z}_3$, $\mathbf{Z}_4$.
2. Type 4 paths to Y: $\mathbf{Z}_1$, $\mathbf{Z}_2$, $\mathbf{Z}_4$.
3. Type 5 paths to Y: $\mathbf{Z}_1$, $\mathbf{Z}_3$, $\mathbf{Z}_4$.
4. Type 6 paths to Y: $\mathbf{Z}_2$, $\mathbf{Z}_6$.

If the confounder for $\{Z_2, Y\}$ is in $\mathbf{Z}_4$ (e.g., in the M-structure featured in Figure A.3), then this Z_2 is guaranteed to be discovered at Step 4. Likewise, as $\mathbf{Z}_6$ is descended from $\mathbf{Z}_4$ through Y, any Z_2 sharing an active path with $\mathbf{Z}_6$ will be discovered at Step 4. If the only confounders for $\{Z_2, Y\}$ are in $\mathbf{Z}_1$ or $\mathbf{Z}_3$, Z_2 will not be discovered at Step 4 and will be conditioned on during Step 5. However, all members of $\mathbf{Z}_1$ and $\mathbf{Z}_3$ that are adjacent to Y will also be included in the conditioning set at Step 5. Together with X, these variables will block all paths from $\mathbf{Z}_5$ to Y. Therefore, such a Z_2 will be on a path to Y that is *re-blocked* by the conditioning set at Step 5.

Finally, we address the resolution of $\mathbf{Z}_5$ at Steps 6 and 7 when sufficient condition C1 does not hold. Even when C1 is violated, no Z_5 will ever be dependent on a Z_1 that is directly adjacent to Y (as

described in Definition C.1 and Proposition E.17). Therefore, any Z_5 will be placed in $\mathbf{Z}_{1,5}$ at Step 6 and in $\mathbf{Z}_5$ at Step 7. □

E.3.2 LDP does not place descendants of X in $\mathbf{Z}_1$

Theorem E.14 (Algorithm 1 does not place direct nor indirect descendants of X in $\mathbf{Z}_1$ when sufficient condition C1 is violated)**.**

Proof. All descendants of X will be marginally dependent on all of $\mathbf{Z}_1$ and all of $\mathbf{Z}_5$ by definition, regardless of any inter-partition active paths that they participate in. Thus, any descendant of X will be placed in $\mathbf{Z}_{\text{POST}}$ at Step 6 if it was not previously eliminated. As any causal path from X to Y features an edge out of X ($X \rightarrow \cdots$), this also guarantees that no causal path from X to Y will be blocked by the $\mathbf{Z}_1$ returned by LDP. □

E.3.3 Adjustment sets returned by LDP block all backdoor paths for $\{X, Y\}$

Theorem E.15 (Adjustment sets returned by LDP block all backdoor paths for $\{X, Y\}$ when sufficient condition C1 is violated)**.**

Proof of Theorem E.15 proceeds from the following argument, as supported by Lemmas E.20 and E.21.

As illustrated in Figure A.1, not every Z_1 must be included in the adjustment set in order to block all backdoor paths. We will show that the adjustment set returned by LDP still blocks all backdoor paths even when 1) some ground truth Z_1 are not placed in $\mathbf{Z}_1$ and 2) some non-Z_1 are placed in $\mathbf{Z}_1$.

First, we address the latter claim that LDP returns valid adjustment sets even when some non-Z_1 are placed in $\mathbf{Z}_1$. Consider the partitions that could be incorrectly labeled as $\mathbf{Z}_1$ when sufficient condition C1 is violated. We have already proven that no descendant of X will ever be placed in $\mathbf{Z}_1$ (Theorem E.14). This implies that members of $\mathbf{Z}_3$, $\mathbf{Z}_6$, and $\mathbf{Z}_7$ can never be mislabeled as $\mathbf{Z}_1$. $\mathbf{Z}_8$ can also never be placed in $\mathbf{Z}_1$, as inter-partition active paths have no effect on its discovery at Step 1. This leaves $\mathbf{Z}_2$, $\mathbf{Z}_4$, and $\mathbf{Z}_5$. Per Theorem E.10, violating sufficient condition C1 does not impact discovery of $\mathbf{Z}_4$ and $\mathbf{Z}_5$. Further, these variables are permissible in valid adjustment sets and are intentionally retained under some criteria (e.g., the disjunctive cause criterion (VanderWeele & Shpitser, 2011)). Per Lemma E.13, $\mathbf{Z}_2$ can only share a confounded path with Y when the confounder is in $\mathbf{Z}_1$, $\mathbf{Z}_3$, $\mathbf{Z}_4$, or $\mathbf{Z}_6$. Further, any Z_2 that is marginally dependent on a Z_4 or Z_6 will be placed in $\mathbf{Z}_{\text{POST}}$ at Step 4. Per Theorem E.14, any member of $\mathbf{Z}_2$ that is descended from X will never be placed in $\mathbf{Z}_1$. Thus, when a Z_3 acts as a confounder for a Z_2 and Y, this Z_2 will never be placed in $\mathbf{Z}_1$. Therefore, the only members of $\mathbf{Z}_2$ that could be placed in $\mathbf{Z}_1$ are non-descendants of X whose only path to Y is confounded by members of $\mathbf{Z}_1$. This does not violate the validity of the returned adjustment set, as such Z_2 lie on paths that will already by blocked by the rest of $\mathbf{Z}_1$, preventing collider bias. This case is illustrated by node Z_2^2 in Figure E.1 (right-hand DAG).

Next, we address the former claim that the adjustment set returned by LDP still blocks all backdoor paths even when some ground truth Z_1 are mislabeled. We begin by introducing the concepts of *root*-Z_1 and *collider*-Z_1. We observe that every backdoor path features a Z_1 that acts as a *root* node for that path: i.e., it is a common cause for $\{X, Y\}$ and all Z_1 that are its descendants on the paths to X and Y. In Figure E.1, $\{Z_1^1, Z_1^3, Z_1^6\}$ are roots for backdoor paths in the left-hand DAG while $\{Z_1^2, Z_1^4, Z_1^5\}$ are roots for backdoor paths in the right-hand DAG. When multiple backdoor paths in $\mathcal{G}_{XY\mathbf{Z}}$ overlap (i.e., share subpaths), some Z_1 can behave as *colliders* for two parent Z_1. In Figure E.1, $\{Z_1^2, Z_1^4\}$ are *collider*-Z_1 on overlapping backdoor paths in the left-hand DAG while $\{Z_1^1, Z_1^2, Z_1^3\}$ are *collider*-Z_1 for backdoor paths in the right-hand DAG. Note that node Z_1^2 in the right-hand DAG simultaneously behaves as a *root*-Z_1 and a *collider*-Z_1 for different backdoor paths.

Let $\mathbf{A}_{XY}$ be an adjustment set for $\{X, Y\}$ that is returned by LDP. We claim that any $\mathbf{A}_{XY}$ that blocks a backdoor path $\mathcal{P}$ meets *at least one* of the following conditions with respect to $\mathcal{P}$:

Item 1 At least one non-*collider*-Z_1 on $\mathcal{P}$ is in $\mathbf{A}_{XY}$; or
Item 2 No *collider*-Z_1 on $\mathcal{P}$ nor any of its descendants is in $\mathbf{A}_{XY}$.

Note that if *Item 1* is met but *Item 2* is not (e.g., a *collider*-Z_1 is in $\mathbf{A}_{XY}$ but so is a non-*collider*-Z_1), $\mathcal{P}$ is still blocked by adjusting for $\mathbf{A}_{XY}$. To prove that $\mathbf{A}_{XY}$ satisfies either *Item 1* or *Item 2*, we introduce the following propositions.

Proposition E.16. *If a Z_4 shares an active path with any Z_1 on $\mathcal{P}$ such that $Z_4 \not\perp\!\!\!\perp Z_1$, that Z_4 must form a v-structure $Z_4 \cdots \rightarrow Z_1 \leftarrow \cdots Z_1'$, where Z_1' lies between Z_1 and X on $\mathcal{P}$. If not, Z_4 would share an active path with X, which violates the definition of Z_4 (Definition C.3). In Figure E.1 (right-hand DAG), examples include $Z_4 \rightarrow Z_1^3 \leftarrow Z_1^2$ and $Z_4 \rightarrow Z_1^3 \leftarrow Z_1^5$. Together with Definition C.3, this proposition implies that no Z_4 will ever be marginally dependent on a Z_1 that is directly adjacent to X.*

Proposition E.17. *If a Z_5 shares an active path with any Z_1 on $\mathcal{P}$ such that $Z_5 \not\perp\!\!\!\perp Z_1$, that Z_5 must form a v-structure $Z_5 \cdots \rightarrow Z_1 \leftarrow \cdots Z_1'$, where Z_1' lies between Z_1 and Y on $\mathcal{P}$. If not, Z_5 would share an active path with Y, which violates the definition of Z_5 (Definition C.1). In Figure E.1 (right-hand DAG), examples include $Z_5 \rightarrow Z_1^1 \leftarrow Z_1^2$ and $Z_5 \rightarrow Z_1^1 \leftarrow Z_1^4$. Together with Definition C.1, this proposition implies that no Z_5 will ever be marginally dependent on a Z_1 that is directly adjacent to Y.*

Proposition E.18 (A single Z_1 cannot be a collider for a Z_4 and a Z_5)**.** *If a single Z_1 was a collider for Z_4 and Z_5, then Z_4 would share an active path with X and Z_5 would share an active path with Y, violating the definitions of these partitions. This proposition justifies the forbidden causal path between Z_1^5 and Z_1^7 in Figure E.1 (left-hand DAG).*

Proposition E.19 (The *root*-Z_1 of a backdoor path will never be marginally dependent on a Z_4 nor a Z_5)**.** *As all root-Z_1 are causal for both X and Y, marginal dependence on either a Z_4 or a Z_5 would violate Propositions E.16, E.17, and E.18.*

Lemma E.20 (LDP is guaranteed to correctly label at least one Z_1 per backdoor path, i.e., the *root*-Z_1)**.**

Proof. Any Z_1 that is not marginally dependent on any Z_4 nor Z_5 will not be incorrectly placed in $\mathbf{Z}_{\text{POST}}$ at Step 4 and will be placed in $\mathbf{Z}_1$ at Step 6 or Step 7. Per Proposition E.19, the *root*-Z_1 of a backdoor path will never be marginally dependent on a Z_4 nor a Z_5. As all backdoor paths must have a *root*-Z_1, then LDP is guaranteed to correctly label at least one Z_1 per backdoor path. □

Proof of Lemma E.20 shows that LDP is guaranteed to block any backdoor path with only a single Z_1 that is adjacent to both X and Y (e.g., Z_1^1 in the left-hand DAG of Figure E.1). Proof of Lemma E.20 is almost proof of *Item 1* and *Item 2* for more complex backdoor paths, but with one missing link: a single Z_1 can act simultaneously as a *root*-Z_1 for one backdoor path and as a *collider*-Z_1 for an overlapping backdoor path. This case is exemplified by node Z_1^2 in Figure E.1 (right-hand DAG). To fully prove *Item 1* and *Item 2*, we conclude with Lemma E.21.

Lemma E.21 (Adjustment sets returned by LDP satisfy *Item 1* and *Item 2*)**.**

Proof. Let $\mathcal{P}$ be a single backdoor path in $\mathcal{G}_{XY\mathbf{Z}}$. To prove Lemma E.21, we will prove that if any *collider*-Z_1 on $\mathcal{P}$ is contained in $\mathbf{A}_{XY}$, then $\mathbf{A}_{XY}$ will also contain a non-*collider*-Z_1 on $\mathcal{P}$. To do so, it suffices to prove that LDP correctly labels at least one *root*-Z_1 on $\mathcal{P}$ that is not also a *collider*-Z_1 for an overlapping backdoor path. Observe that any *collider*-Z_1 Z_1^* must be a descendant of two *root*-Z_1 that are not *collider*-Z_1 for any other path, however long the indirect paths to these roots are. Even if Z_1^* is retained in $\mathbf{A}_{XY}$, so will its ancestors that are *root*-Z_1 (per Lemma E.20). Thus, even when *Item 2* goes unsatisfied, $\mathbf{A}_{XY}$ is guaranteed to satisfy *Item 1*. □

Figure E.1 provides an illustrative example in the right-hand DAG. There, Z_1^2 is a *root*-Z_1 for backdoor path $X - Z_1^1 - Z_1^2 - Z_1^3 - Y$ but is a *collider*-Z_1 for root Z_1^4 and root Z_1^5. Even though Z_1^2 is in $\mathbf{A}_{XY}$ (per Lemma E.20), so are Z_1^4 and Z_1^5 (also per Lemma E.20). Thus, *Item 1* and *Item 2* are satisfied by $\mathbf{A}_{XY}$.

F APPENDIX: EXTENDED EXPERIMENTAL DESIGN

F.1 Baseline Methods

PC Algorithm PC is a classic global structure inference algorithm that provides asymptotic theoretical guarantees (Spirtes et al., 2000). It assumes Causal Markov, faithfulness, and causal suf-

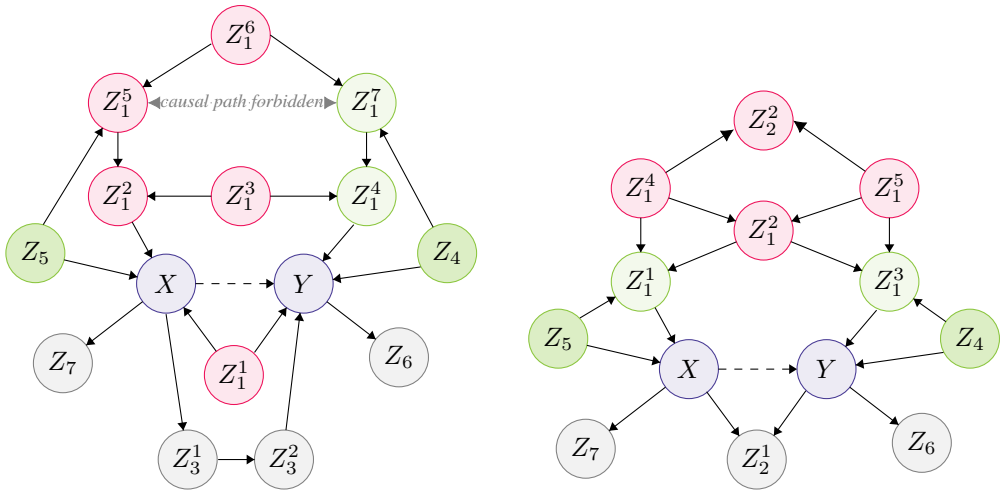

Figure E.1: Two DAGs that exemplify the behavior of LDP for valid adjustment set detection in the presence of inter-partition active paths. All red nodes will be placed in $\mathbf{Z}_1$ by LDP. All confounders for $\{X, Y\}$ that are colored green will be mislabeled due to their marginal dependence on Z_4 or Z_5. ***Left:*** Per Lemma E.20, Z_1^1, Z_1^3 and Z_1^6 will be placed in $\mathbf{Z}_1$. Despite their marginal dependence on the only Z_5 in this structure, Z_1^2 and Z_1^5 will never be placed in $\mathbf{Z}_{\text{POST}}$ due to the presence of Z_1^1, as $Z_1^2 \perp\!\!\!\perp Z_1^1$ and $Z_1^5 \perp\!\!\!\perp Z_1^1$. Together, the confounders highlighted in red ($\{Z_1^1, Z_1^2, Z_1^3, Z_1^5, Z_1^6\}$) constitute a valid adjustment set that blocks all backdoor paths and contains no descendents of X. No causal path of either directionality is permissible between Z_1^5 and Z_1^7 per Proposition E.18. If this path were to contain a confounder analogous to Z_1^3, this would be permissible and this node would be placed in $\mathbf{Z}_1$ by LDP.
Right: This DAG contains a modified butterfly structure, which will be partially retained in $\mathbf{Z}_1$ ($\{Z_1^2, Z_1^4, Z_1^5\}$) while still blocking all backdoor paths. As there is only one Z_5 in this structure and no backdoor path whose members are marginally independent of Z_1^1, this confounder will be mislabeled as $\mathbf{Z}_{\text{POST}}$ at Step 6. This DAG also illustrates a case where a member of $\mathbf{Z}_2$ (Z_2^2) is placed in $\mathbf{Z}_1$. Inclusion of Z_2^2 does not violate the validity of the adjustment set returned by LDP, as this node is not a descendent of X and adjusting for $\{Z_1^2, Z_1^4, Z_1^5\}$ prevents collider bias.

ficiency and returns a *Markov equivalence class* (MEC). Time complexity for PC is worst-case exponential in node count (Kalisch & Buhlmann, 2007). Experiments use the implementation by Kalisch & Buhlmann (2007).[8]

MB-by-MB MB-by-MB (Wang et al., 2014) infers the local structure around a target node to distinguish parents from children. It sequentially learns Markov blankets (MBs) and the local structures within these, starting from the target node, moving to its neighbors, and so on. It terminates when the parents and children of the target are discovered or if it is not possible to distinguish them, returning the induced *completed partially directed acyclic graph* (CPDAG) over the target and its neighbors. Experiments use an implementation that combines IAMB (Tsamardinos et al., 2003, Fig. 2) and PC for every sequential step. Like PC, time complexity is worst-case exponential in node count.

Local Discovery using Eager Collider Checks (LDECC) LDECC (Gupta et al., 2023) is a local discovery algorithm that infers the induced CPDAG over a given target node and its neighbors. Unlike MB-by-MB, LDECC does not proceed sequentially and runs conditional independence tests in a similar order as PC, leveraging discovered unshielded colliders to immediately orient the edges around the target node. LDECC is provably polynomial-time for certain categories of DAGs, but exponential for others.

Baseline Evaluation Let $\mathbf{A}_{XY}$ be any adjustment set for $\{X, Y\}$ returned by a method in this study. Let $\mathbf{A}_{CC} := \{\mathbf{Z}_1\}$ and $\mathbf{A}_{DC} := \{\mathbf{Z}_1, \mathbf{Z}_4, \mathbf{Z}_5\}$ be valid adjustment sets for $\{X, Y\}$ under the *common cause criterion* (CCC) and *disjunctive cause criterion* (DCC), respectively (Van-

[8] https://github.com/keiichishima/pcalg

derWeele & Shpitser, 2011). For PC, $\mathbf{A}_{CC} := \text{ancestors}(X) \cap \text{ancestors}(Y) = \{\mathbf{Z}_1\}$, and $\mathbf{A}_{DC} := \{\text{ancestors}(X) \cup \text{ancestors}(Y) \setminus \text{descendants}(X)\} = \{\mathbf{Z}_1, \mathbf{Z}_4, \mathbf{Z}_5\}$, where ancestors and descendants hold for all members of the MEC. As MB-by-MB and LDECC only return the direct parents and children of a single target, we run these baselines with X and Y as separate targets and cache intermediate results to prevent redundant independence testing. $\mathbf{A}_{DC} := \{\text{parents}(X) \cup \text{parents}(Y) \setminus \text{children}(X)\} = \mathbf{Z}'_1 \cup \mathbf{Z}_4 \cup \mathbf{Z}_5$, where $\mathbf{Z}'_1$ is directly adjacent to X, Y, or both (but not neither). $\mathbf{A}_{CC} := \{\text{parents}(X) \cap \text{parents}(Y)\}$, i.e., all confounders directly adjacent to both X and Y. Thus, $\mathbf{A}_{CC}$ under LDECC and MB-by-MB are not guaranteed to block all backdoor paths.

G APPENDIX: EXPERIMENTAL RESULTS

			Mean Runtime (seconds)				Tests Per Run			
$\lvert\mathbf{Z}\rvert$	$\lvert\mathbf{Z}_{-}\rvert$	LDP:$\lvert\mathbf{Z}\rvert^2$	LDP	LDECC	MB-by-MB	PC	LDP	LDECC	MB-by-MB	PC
8	1	0.781	0.0143 (0.0121-0.0165)	0.1144 (0.1098-0.119)	0.1205 (0.1163-0.1247)	0.076 (0.0734-0.0787)	50	641	513.6 (513.2-514.1)	508
16	2	0.500	0.0299 (0.0287-0.0311)	7.011 (6.7783-7.2437)	9.3711 (9.0831-9.6591)	8.7598 (8.5028-9.0169)	128	23344.3 (23331.9-23356.7)	29687.5 (29675.4-29699.5)	29556
24	3	0.406	0.0410 (0.0399-0.0421)	-	-	-	234	-	-	-
32	4	0.359	0.0587 (0.0574-0.0600)	-	-	-	368	-	-	-
40	5	0.331	0.0838 (0.0827-0.0849)	-	-	-	530	-	-	-
48	6	0.313	0.1230 (0.1215-0.1245)	-	-	-	720	-	-	-
56	7	0.299	0.1492 (0.1476-0.1509)	-	-	-	938	-	-	-
64	8	0.289	0.2016 (0.1963-0.2070)	-	-	-	1184	-	-	-
72	9	0.281	0.2495 (0.2458-0.2533)	-	-	-	1458	-	-	-
80	10	0.275	0.2836 (0.2784-0.2887)	-	-	-	1760	-	-	-

Table G.1: Mean runtime and total independence tests performed per DAG as cardinality of $\mathbf{Z}$ ($|\mathbf{Z}|$) increases. Values are averaged over 100 replicates for DAGs analogous to Figure 1 (sample size $n = 1k$ each), with 95% confidence intervals in parentheses. All data generating processes feature hypergeometric noise distributions with quadratic causal mechanisms, under the same structural equation as for the 10-node DAG reported in Table G.2. Independence was determined by an oracle. Cardinality of each partition is reported as $|\mathbf{Z}_{-}|$. The ratio of true total tests for LDP to expected quadratic count is reported as LDP:$|\mathbf{Z}|^2$. Baselines were only evaluated up to $|\mathbf{Z}| = 16$ due to very high test counts. All experiments were run on a 2017 MacBook with 2.9 GHz Quad-Core Intel Core i7. Growth curves are plotted in Figure 2.

DAG Structure	Causal Mechanism	Noise Distribution	$X \to Y$	Structural Equation
10-node (Figure 1)	Linear	Bernoulli	True	$V_i = \lfloor(0.3 * sum(\mathbf{Pa}_i))\rfloor + \epsilon_i$
10-node (Figure 1)	Linear	Bernoulli	False	$V_i = \lfloor(0.45 * sum(\mathbf{Pa}_i))\rfloor + \epsilon_i$
10-node (Figure 1)	Linear	Hypergeometric	True	$V_i = \lfloor(0.3 * sum(\mathbf{Pa}_i))\rfloor + \epsilon_i$
10-node (Figure 1)	Linear	Hypergeometric	False	$V_i = \lfloor(0.45 * sum(\mathbf{Pa}_i))\rfloor + \epsilon_i$
10-node (Figure 1)	Quadratic	Bernoulli	True	$V_i = \lfloor(-1.4 * sum(\mathbf{Pa}_i)^2)\rfloor + \epsilon_i$
10-node (Figure 1)	Quadratic	Bernoulli	False	$V_i = \lfloor(-1.4 * sum(\mathbf{Pa}_i)^2)\rfloor + \epsilon_i$
10-node (Figure 1)	Quadratic	Hypergeometric	True	$V_i = \lfloor(0.4 * sum(\mathbf{Pa}_i)^2)\rfloor + \epsilon_i$
10-node (Figure 1)	Quadratic	Hypergeometric	False	$V_i = \lfloor(0.4 * sum(\mathbf{Pa}_i)^2)\rfloor + \epsilon_i$
10-node (Figure 1)	Cube root	Bernoulli	True	$V_i = \lfloor(1.2 * \sqrt[3]{(\mathbf{Pa}_i)})\rfloor + \epsilon_i$
10-node (Figure 1)	Cube root	Bernoulli	False	$V_i = \lfloor(1.2 * \sqrt[3]{(\mathbf{Pa}_i)})\rfloor + \epsilon_i$
10-node (Figure 1)	Cube root	Hypergeometric	True	$V_i = \lfloor(0.7 * \sqrt[3]{(\mathbf{Pa}_i)})\rfloor + \epsilon_i$
10-node (Figure 1)	Cube root	Hypergeometric	False	$V_i = \lfloor(0.7 * \sqrt[3]{(\mathbf{Pa}_i)})\rfloor + \epsilon_i$
13-node with M (Figure A.3)	Linear	Bernoulli	True	$V_i = \lfloor(1.5 * sum(\mathbf{Pa}_i))\rfloor + \epsilon_i$
13-node with M (Figure A.3)	Quadratic	Hypergeometric	True	$V_i = \lfloor(1.5 * sum(\mathbf{Pa}_i)^2)\rfloor + \epsilon_i$
13-node with butterfly (Figure A.3)	Linear	Bernoulli	True	$V_i = \lfloor(1.9 * sum(\mathbf{Pa}_i))\rfloor + \epsilon_i$
13-node with butterfly (Figure A.3)	Quadratic	Hypergeometric	True	$V_i = \lfloor(2.8 * sum(\mathbf{Pa}_i)^2)\rfloor + \epsilon_i$

Table G.2: Structural equations for all discrete synthetic data generating processes. V_i denotes a random variable, $\mathbf{Pa}_i$ denotes the set of its direct causal parents, and ϵ_i denotes the random noise term associated with it. Fixed coefficients range across structural equations ($[-1.4, 2.8]$) to simulate varying effect sizes.

DAG Structure	Experiment	Causal Mechanism	Noise Distribution	$X \to Y$	Structural Equation
10-node (Figure 1)	Figure 3	Linear	Gaussian	True	$V_i = \sum(r * \mathbf{Pa}_i) + \epsilon_i$
10-node (Figure 1)	Figure 3	Linear	Gaussian	False	$V_i = \sum(r * \mathbf{Pa}_i) + \epsilon_i$
10-node (Figure 1)	Figure 3	Linear	Uniform	True	$V_i = \sum(r * \mathbf{Pa}_i) + \epsilon_i$
10-node (Figure 1)	Figure 3	Linear	Uniform	False	$V_i = \sum(r * \mathbf{Pa}_i) + \epsilon_i$
10-node (Figure 1)	Figure 3	Linear	Exponential	True	$V_i = \sum(r * \mathbf{Pa}_i) + \epsilon_i$
10-node (Figure 1)	Figure 3	Linear	Exponential	False	$V_i = \sum(r * \mathbf{Pa}_i) + \epsilon_i$
10-node (Figure 1)	Figure 4	Linear	Gaussian	True	$V_i = \sum(c * \mathbf{Pa}_i) + \epsilon_i$

Table G.3: Structural equations for all continuous synthetic data generating processes. V_i denotes a random variable, $\mathbf{Pa}_i$ denotes the set of its direct causal parents, and ϵ_i denotes the random noise term associated with it. Coefficient r is a float selected uniformly at random from the range $[1.0, 3.0)$. For the experiments reported in Figure 4, coefficient c is 1.0 for all parents except when $V_i = Y$ and $Pa_i = X$, in which case $c = 2.75$. For this DAG, the total effect of X on Y is 3.75, as the direct effect is 2.75 and the indirect effect through Z_3 is 1.0.

	10-node graph with bernoulli noise					
	Linear		Quadratic		Cube Root	
n	$X \to Y$	$X \not\to Y$	$X \to Y$	$X \not\to Y$	$X \to Y$	$X \not\to Y$
100	21.4 (20.1-22.6)	16.4 (15.2-17.5)	36.4 (34.9-37.8)	34.8 (33.7-35.8)	25.8 (24.1-27.4)	20.2 (18.7-21.8)
500	58.8 (55.2-62.3)	67.0 (62.7-71.3)	92.0 (89.7-94.3)	59.5 (54.5-64.5)	65.2 (62.0-68.5)	74.4 (71.6-77.1)
$1k$	86.1 (83.6-88.6)	89.2 (86.5-92.0)	99.8 (99.3-100)	99.2 (98.2-100)	97.2 (95.8-98.7)	98.0 (96.7-99.3)
$5k$	99.9 (99.6-100)	99.9 (99.6-100)	100 (100-100)	100 (100-100)	99.9 (99.6-100)	99.8 (99.3-100)
$10k$	100 (100-100)	99.9 (99.6-100)	100 (100-100)	100 (100-100)	99.6 (99.1-100)	99.9 (99.6-100)
	10-node graph with hypergeometric noise					
	Linear		Quadratic		Cube Root	
n	$X \to Y$	$X \not\to Y$	$X \to Y$	$X \not\to Y$	$X \to Y$	$X \not\to Y$
100	20.4 (18.8-22.0)	15.6 (14.3-17.0)	31.2 (29.9-32.6)	31.4 (29.9-32.9)	23.0 (21.7-24.3)	18.4 (16.9-19.8)
500	68.1 (64.7-71.6)	43.1 (39.5-46.7)	85.5 (82.7-88.3)	83.0 (80.7-85.3)	70.6 (67.7-73.5)	56.9 (54.5-59.2)
$1k$	92.4 (90.1-94.6)	78.5 (74.7-82.3)	97.8 (96.3-99.2)	98.5 (97.2-99.8)	95.5 (94.3-96.7)	87.6 (84.9-90.3)
$5k$	100 (100-100)	98.9 (97.9-99.8)	99.2 (98.0-100)	100 (100-100)	99.2 (98.5-100)	100 (100-100)
$10k$	99.9 (99.6-100)	99.8 (99.4-100)	99.8 (99.4-100)	100 (100-100)	99.8 (99.3-100)	100 (100-100)

Table G.4: Partition label accuracy of Algorithm 1 on a 10-node DAG (Figure 1) across discrete noise distributions, linear and nonlinar causal mechanisms, and sample sizes (n). All DAGs feature one node per partition ($\mathbf{Z}_1 - \mathbf{Z}_8$). Reported values are partition label accuracy averaged over 100 DAGs (i.e., 800 variables total, excluding all exposure-outcome pairs). The 95% confidence interval is reported in parentheses. Data generating processes where X is a direct cause of Y are denoted by $X \to Y$, with $X \not\to Y$ denoting no direct causal effect of X on Y. Independence was determined by chi-square tests ($\alpha = 0.001$). All experiments were run on a 2017 MacBook with 2.9 GHz Quad-Core Intel Core i7.

	10-node graph with continuous noise					
	Gaussian | Linear		Uniform | Linear		Exponential | Linear	
n	$X \to Y$	$X \not\to Y$	$X \to Y$	$X \not\to Y$	$X \to Y$	$X \not\to Y$
100	64.8 (60.7-68.8)	66.5 (63.0-70.0)	65.5 (61.4-69.6)	67.9 (64.6-71.1)	61.0 (56.9-65.1)	63.6 (60.2-67.0)
500	96.1 (94.4-97.8)	94.6 (92.8-96.5)	97.8 (96.6-98.9)	94.2 (92.2-96.3)	95.2 (93.5-97.0)	93.6 (91.3-95.9)
$1k$	99.2 (98.7-99.8)	98.5 (97.5-99.5)	99.0 (98.3-99.7)	98.8 (97.8-99.7)	99.8 (99.4-100)	98.5 (97.4-99.6)
$5k$	100 (100-100)	99.9 (99.6-100)	99.6 (98.9-100)	99.8 (99.3-100)	99.5 (98.7-100)	99.9 (99.6-100)
$10k$	99.9 (99.6-100)	99.9 (99.6-100)	100 (100-100)	100 (100-100)	100 (100-100)	100 (100-100)

Table G.5: Partition label accuracy of Algorithm 1 on a 10-node DAG (Figure 1) across continuous noise distributions, linear causal mechanisms, and sample sizes (n). All DAGs feature one node per partition ($\mathbf{Z}_1 - \mathbf{Z}_8$). Reported values are partition label accuracy averaged over 100 DAGs (i.e., 800 variables total, excluding all exposure-outcome pairs). The 95% confidence interval is reported in parentheses. Data generating processes where X is a direct cause of Y are denoted by $X \to Y$, with $X \not\to Y$ denoting no direct causal effect of X on Y. Independence was determined by Fisher-z tests ($\alpha = 0.001$). All experiments were run on a 2017 MacBook with 2.9 GHz Quad-Core Intel Core i7.

	M-Structure					
	BERNOULLI | LINEAR			HYPERGEOMETRIC | QUADRATIC		
n	$\mathbf{Z}$ ACC	$\mathbf{Z}_1$ PREC	$\mathbf{Z}_1$ REC	$\mathbf{Z}$ ACC	$\mathbf{Z}_1$ PREC	$\mathbf{Z}_1$ REC
500	73.5 (71.0-76.0)	26.2 (17.6-34.9)	27.0 (18.3-35.7)	75.3 (73.8-76.8)	16.0 (8.8-23.2)	16.0 (8.8-23.2)
$1k$	92.1 (90.4-93.8)	90.0 (84.1-95.9)	90.0 (84.1-95.9)	87.3 (85.8-88.7)	94.0 (89.3-98.7)	94.0 (89.3-98.7)
$5k$	97.1 (96.0-98.2)	97.0 (93.6-100)	97.0 (93.6-100)	99.8 (99.6-100)	100 (100-100)	100 (100-100)
$10k$	99.7 (99.4-100)	100 (100-100)	100 (100-100)	100 (100-100)	100 (100-100)	100 (100-100)
	Butterfly Structure					
	BERNOULLI | LINEAR			HYPERGEOMETRIC | QUADRATIC		
n	$\mathbf{Z}$ ACC	$\mathbf{Z}_1$ PREC	$\mathbf{Z}_1$ REC	$\mathbf{Z}$ ACC	$\mathbf{Z}_1$ PREC	$\mathbf{Z}_1$ REC
$1k$	60.4 (57.5-63.2)	16.8 (9.6-24.0)	12.5 (6.6-18.4)	61.5 (58.9-64.0)	28.9 (20.1-37.7)	16.0 (10.3-21.7)
$2.5k$	98.8 (97.1-100)	98.0 (95.2-100)	98.0 (95.2-100	99.9 (99.7-100)	100 (100-100)	100 (100-100)
$5k$	98.9 (97.4-100)	99.0 (97.0-100)	98.2 (95.8-100)	99.9 (99.7-100)	100 (100-100)	100 (100-100)
$10k$	99.7 (99.4-100)	100 (100-100)	99.2 (98.4-100)	99.8 (99.5-100)	100 (100-100)	99.5 (98.5-100)

Table G.6: Performance of Algorithm 1 on 13-node DAGs containing an M-structure structure or butterfly structure (Figure A.3) across noise distributions, causal mechanisms, and sample sizes (n). In all DAGs, exposure X is a direct cause of outcome Y. Metrics reported are accuracy of all labels ($\mathbf{Z}$ ACC), mean precision for partition $\mathbf{Z}_1$ ($\mathbf{Z}_1$ PRE), and mean recall for partition $\mathbf{Z}_1$ ($\mathbf{Z}_1$ REC). The 95% confidence interval is reported in parentheses. Independence was determined by chi-square tests with $\alpha = 0.001$. All experiments were run on a 2017 MacBook with 2.9 GHz Quad-Core Intel Core i7.

	Graph with m-structure, butterfly structure, and indirect mediators					
	BERNOULLI | LINEAR			HYPERGEOMETRIC | QUADRATIC		
n	$\mathbf{Z}$ ACC	$\mathbf{Z}_1$ PREC	$\mathbf{Z}_1$ REC	$\mathbf{Z}$ ACC	$\mathbf{Z}_1$ PREC	$\mathbf{Z}_1$ REC
$5k$	60.2 (59.0-61.4)	48.8 (38.9-58.6)	16.8 (12.4-21.1)	72.7 (70.2-75.3)	93.5 (88.7-98.3)	57.8 (51.4-64.1)
$10k$	85.8 (82.2-89.4)	66.5 (57.4-75.6)	66.2 (57.1-75.4)	97.9 (96.5-99.2)	96.9 (93.8-99.9)	97.0 (94.0-100.0)
$15k$	97.9 (96.5-99.2)	96.3 (93.3-99.4)	96.8 (93.6-99.9)	98.0 (96.7-99.3)	96.3 (93.3-99.4)	97.2 (94.3-100)
$20k$	98.7 (97.6-99.9)	97.4 (94.6-100)	98.0 (95.2-100)	98.7 (98.0-99.4)	99.1 (98.1-100.0)	99.5 (98.8-100)

Table G.7: Performance of Algorithm 1 on a 17-node DAG featuring an M-structure, butterfly structure, and mediator chain (Figure A.4). Data generating processes represent various discrete noise distributions, linear and nonlinar causal mechanisms, and sample sizes (n). Exposure X is a direct cause of outcome Y for all DAGs. Reported values are averaged over 100 DAGs. Metrics reported are mean accuracy of all labels ($\mathbf{Z}$ ACC), mean precision for partition $\mathbf{Z}_1$ ($\mathbf{Z}_1$ PRE), and mean recall for partition $\mathbf{Z}_1$ ($\mathbf{Z}_1$ REC). The 95% confidence interval is reported in parentheses. Independence was determined by chi-square tests with $\alpha = 0.005$. All experiments were run on a 2017 MacBook with 2.9 GHz Quad-Core Intel Core i7.

	Common Cause Criterion											
	Valid Adjustment Set (↑)				Confounder Precision (↑)				Confounder Recall (↑)			
n	LDP	PC	LDECC	MB-BY-MB	LDP	PC	LDECC	MB-BY-MB	LDP	PC	LDECC	MB-BY-MB
$25k$	**0.8**	0.7	0.0	0.0	**80.00 (53.87-100)**	35.00 (20.03-49.97)	0.0 (0.0-0.0)	0.0 (0.0-0.0)	**80.00 (53.87-100)**	35.00 (20.03-49.97)	0.0 (0.0-0.0)	0.0 (0.0-0.0)
$50k$	0.7	**1.0**	0.0	0.0	**76.67 (50.81-100)**	50.00 (50.00-50.00)	0.0 (0.0-0.0)	0.0 (0.0-0.0)	**80.00 (53.87-100)**	50.00 (50.00-50.00)	0.0 (0.0-0.0)	0.0 (0.0-0.0)
$75k$	**0.9**	0.4	0.0	0.0	**90.00 (80.02-99.98)**	20.00 (4.00-36.00)	0.0 (0.0-0.0)	0.0 (0.0-0.0)	**100 (100-100)**	20.00 (4.00-36.00)	0.0 (0.0-0.0)	0.0 (0.0-0.0)
	Disjunctive Cause Criterion											
	Valid Adjustment Set (↑)				Confounder Precision (↑)				Confounder Recall (↑)			
n	LDP	PC	LDECC	MB-BY-MB	LDP	PC	LDECC	MB-BY-MB	LDP	PC	LDECC	MB-BY-MB
$25k$	0.8	**0.9**	0.3	0.9	**38.00 (25.33-50.67)**	22.50 (17.60-27.40)	33.33 (8.03-58.64)	34.17 (24.91-43.42)	**80.00 (53.87-100)**	45.00 (35.20-54.80)	25.00 (8.67-41.33)	45.00 (35.20-54.80)
$50k$	0.7	**1.0**	0.2	0.7	**36.33 (23.92-48.75)**	26.67 (24.49-28.84)	10.00 (0-23.07)	23.33 (11.71-34.96)	**80.00 (53.87-100)**	60.00 (46.93-73.07)	10.00 (0-23.07)	35.00 (20.03-49.97)
$75k$	0.9	**1.0**	0.0	0.4	**45.00 (41.73-48.27)**	25.83 (24.2-27.47)	8.33 (0-19.03)	29.05 (12.20-45.90)	**100 (100-100)**	50.00 (50.00-50.00)	12.50 (0-28.54)	35.71 (17.64-53.79)
	Both Criteria											
	Independence Tests (↓)				Runtime (seconds) (↓)							
n	LDP	PC	LDECC	MB-BY-MB	LDP	PC	LDECC	MB-BY-MB				
$25k$	**142.9 (141.5-144.3)**	3021.9 (2975.2-3068.6)	2784.1 (2118.4-3449.8)	823.7 (610.1-1037.3)	**0.065 (0.061-0.069)**	92.08 (90.351-93.81)	99.485 (78.085-120.885)	86.292 (51.39-121.193)				
$50k$	**146.9 (145.2-148.6)**	3841.9 (3761.2-3922.6)	4405 (3734.8-5075.2)	1146.6 (660.5-1632.7)	**0.109 (0.101-0.116)**	243.973 (237.472-250.474)	310.255 (262.338-358.172)	263.322 (145.116-381.528)				
$75k$	**148.6 (147.3-149.9)**	4307.9 (4225.9-4389.9)	4615.2 (4049.2-5181.3)	1567.3 (881.9-2252.7)	**0.162 (0.145-0.178)**	415.874 (398.714-433.035)	473.107 (408.198-538.016)	582.672 (306.342-859.001)				

Table G.8: Baseline comparison on the MILDEW benchmark from `bnlearn` (Scutari, 2010), with MIKRO_1 as exposure and MELDUG_2 as outcome. Independence was determined by chi-square independence tests with $\alpha = 0.005$. Both the common cause criterion and disjunctive cause criterion are considered. Values are reported for 10 replicate DAGs with 95% confidence intervals in parentheses. Sample size is denoted by n. Adjustment set quality was measured by fraction that are valid under the backdoor criterion, confounder precision per adjustment set, and confounder recall per adjustment set. The method proposed in this work is highlighted in yellow. The most performant values per metric are bolded. All experiments were run on a 2017 MacBook with 2.9 GHz Quad-Core Intel Core i7. Results are visualized in Figure 4.

	Common Cause Criterion											
	Valid Adjustment Set (↑)				Average Treatment Effect (ATE)				ATE Mean Squared Error (↓)			
n	LDP	PC	LDECC	MB-by-MB	LDP	PC	LDECC	MB-by-MB	LDP	PC	LDECC	MB-by-MB
$1k$	**0.93**	0.00	0.03	0.10	**3.77 (3.75-3.79)**	3.58 (3.38-3.77)	3.88 (3.71-4.05)	3.97 (3.86-4.08)	**0.0096**	1.0509	0.7817	0.3703
$2.5k$	**0.96**	0.00	0.02	0.30	**3.76 (3.75-3.78)**	2.5 (2.14-2.87)	4.08 (4.07-4.09)	3.97 (3.93-4.01)	**0.0053**	4.9982	0.1088	0.0817
$5k$	**0.96**	0.03	0.04	0.60	**3.76 (3.75-3.77)**	1.09 (0.78-1.4)	4.07 (4.05-4.08)	3.87 (3.83-3.9)	**0.0046**	9.5287	0.1054	0.0473
$7.5k$	**0.97**	0.11	0.14	0.73	**3.76 (3.75-3.77)**	1 (0.72-1.27)	4.04 (4.01-4.06)	3.83 (3.8-3.86)	**0.0037**	9.5009	0.0950	0.0325
	Confounder Precision (↑)				Confounder Recall (↑)				Adjustment Set Cardinality			
n	LDP	PC	LDECC	MB-by-MB	LDP	PC	LDECC	MB-by-MB	LDP	PC	LDECC	MB-by-MB
$1k$	**93 (87.97-98.03)**	13.17 (8.88-17.45)	3 (0-6.36)	10 (4.09-15.91)	**93 (87.97-98.03)**	27 (18.25-35.75)	3 (0-6.36)	10 (4.09-15.91)	**0.9 (0.9-1)**	0.6 (0.4-0.7)	0.1 (0-0.1)	0.1 (0.1-0.2)
$2.5k$	**96 (92.14-99.86)**	16.83 (12.34-21.32)	2 (0-4.76)	30 (20.97-39.03)	**96 (92.14-99.86)**	36 (26.54-45.46)	2 (0-4.76)	30 (20.97-39.03)	**1 (0.9-1)**	1 (0.8-1.2)	0 (0-0)	0.3 (0.2-0.4)
$5k$	**96 (92.14-99.86)**	27.33 (23.19-31.48)	4 (0.14-7.86)	60 (50.35-69.65)	**96 (92.14-99.86)**	70 (60.97-79.03)	4 (0.14-7.86)	60 (50.35-69.65)	**1 (0.9-1)**	2.2 (2-2.4)	0 (0-0.1)	0.6 (0.5-0.7)
$7.5k$	**97 (93.64-100)**	34.08 (30.81-37.36)	14 (7.16-20.84)	73 (64.25-81.75)	**97 (93.64-100)**	90 (84.09-95.91)	14 (7.16-20.84)	73 (64.25-81.75)	**1 (0.9-1)**	2.7 (2.5-2.9)	0.1 (0.1-0.2)	0.7 (0.7-0.8)
	Disjunctive Cause Criterion											
	Valid Adjustment Set (↑)				Average Treatment Effect (ATE)				ATE Mean Squared Error (↓)			
n	LDP	PC	LDECC	MB-by-MB	LDP	PC	LDECC	MB-by-MB	LDP	PC	LDECC	MB-by-MB
$1k$	**0.93**	0.0	0.09	0.02	**3.77 (3.76-3.79)**	0.87 (0.6-1.14)	1.41 (1.15-1.67)	1.05 (0.77-1.34)	**0.0091**	10.2090	7.1930	9.3487
$2.5k$	**0.96**	0.0	0.11	0.08	**3.76 (3.75-3.78)**	0.28 (0.23-0.33)	1.31 (1.11-1.51)	1.34 (1.07-1.61)	**0.0055**	12.0875	6.9627	7.6885
$5k$	**0.96**	0.0	0.14	0.32	**3.76 (3.75-3.78)**	0.4 (0.29-0.52)	2.09 (1.79-2.38)	2.34 (2.07-2.62)	**0.0050**	11.5372	5.0590	3.9600
$7.5k$	**0.97**	0.03	0.31	0.48	**3.76 (3.75-3.77)**	0.6 (0.43-0.77)	2.52 (2.22-2.81)	2.75 (2.49-3.01)	**0.0037**	10.6568	3.8045	2.7495
	Confounder Precision (↑)				Confounder Recall (↑)				Adjustment Set Cardinality			
n	LDP	PC	LDECC	MB-by-MB	LDP	PC	LDECC	MB-by-MB	LDP	PC	LDECC	MB-by-MB
$1k$	**31.83 (29.97-33.7)**	27.08 (24.77-29.4)	23.4 (19.08-27.72)	30.83 (27.95-33.72)	**93 (87.97-98.03)**	85 (77.97-92.03)	61 (51.39-70.61)	87 (80.38-93.62)	**2.8 (2.7-2.9)**	2.7 (2.5-2.9)	2.4 (2.2-2.6)	2.6 (2.3-2.8)
$2.5k$	32.33 (30.95-33.71)	30.43 (29.51-31.36)	17.83 (11.86-23.8)	**40.33 (37.88-42.79)**	96 (92.14-99.86)	100 (100-100)	31 (21.89-40.11)	**99 (97.04-100)**	**2.9 (2.8-3)**	3.4 (3.3-3.5)	1.5 (1.4-1.7)	2.6 (2.5-2.7)
$5k$	32.17 (30.83-33.5)	27.5 (26.46-28.54)	12.5 (7.41-17.59)	**40.92 (38.57-43.26)**	96 (92.14-99.86)	99 (97.04-100)	23 (14.71-31.29)	**98 (95.24-100)**	**2.9 (2.8-3)**	3.7 (3.5-3.8)	1.2 (1-1.3)	2.5 (2.4-2.6)
$7.5k$	32.5 (31.33-33.67)	27.63 (26.49-28.77)	19.5 (15.05-23.95)	**40.5 (38.56-42.44)**	97 (93.64-100)	100 (100-100)	46 (36.18-55.82)	**99 (97.04-100)**	**2.9 (2.9-3)**	3.8 (3.6-3.9)	1.5 (1.3-1.7)	2.6 (2.4-2.7)

Table G.9: Average treatment effect (ATE) estimation with adjustment sets identified by LDP, PC, LDECC, and MB-by-MB for a 10-node linear-Gaussian DAG (Figure 1). Both the common cause criterion (CCC) and disjunctive cause criterion (DCC) are considered. Values are reported for 100 replicate DAGs with 95% confidence intervals in parentheses. Independence was determined by Fisher-z tests with $\alpha = 0.01$. Adjustment set quality was measured by fraction that are valid under the backdoor criterion, ATE (ground truth = 3.75), ATE mean squared error, confounder precision per adjustment set, confounder recall per adjustment set, and cardinality of the adjustment set (ground truth is 1 under the CCC and 3 under the DCC). The method proposed in this work is highlighted in yellow. The most performant values per metric are bolded. All experiments were run on a 2017 MacBook with 2.9 GHz Quad-Core Intel Core i7. Results are visualized in Figure 4.

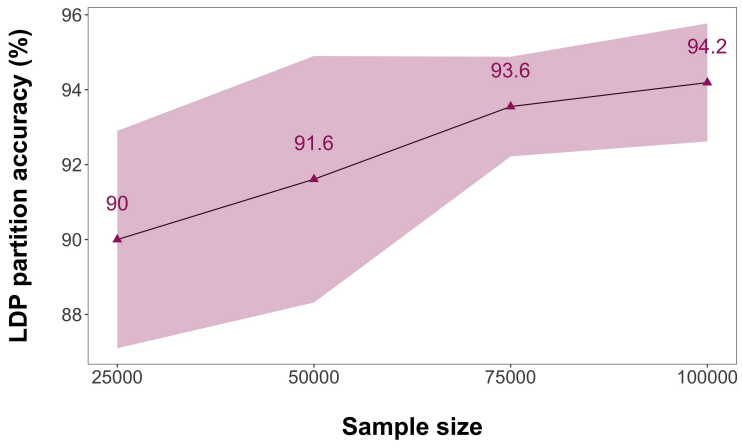

Figure G.1: LDP partition accuracy on the MILDEW benchmark. Mean accuracy was computed for 10 replicate samples from the ground truth DAG using `bnlearn` (Scutari, 2010). We measure partition accuracy as the percent of partition labels that are consistent with ground truth. Independence was determined by chi-square tests ($\alpha = 0.005$). Shaded regions represent the 95% confidence interval. All experiments were run on a 2017 MacBook with 2.9 GHz Quad-Core Intel Core i7.