Journal of Machine Learning Research 24 (2023) 1-144 Submitted 2/22; Revised 7/23; Published 7/23

# Distinguishing Cause and Effect in Bivariate Structural Causal Models: A Systematic Investigation

**Christoph Käding** MAIL@CKAEDING.NET
*Institute of Data Science*
*German Aerospace Center (DLR)*
*07745 Jena, Germany*

**Jakob Runge*** JAKOB.RUNGE@DLR.DE
*Institute of Data Science*
*German Aerospace Center (DLR)*
*07745 Jena, Germany*
*and*
*Technische Universität Berlin*
*10623 Berlin, Germany*

**Editor:** Silvia Chiappa

## Abstract

Distinguishing cause and effect from purely observational data is a fundamental problem in science. Even the atomic bivariate case, seemingly the simplest, is challenging and requires assumptions about the underlying data generating process to identify the direction of cause and effect. In recent years, a variety of approaches have been developed to address this problem, each with its own assumptions, strengths, and weaknesses. In machine learning, common benchmarks with real and synthetic data have been a main driver of innovation. For cause-effect identification, real data as well as synthetic benchmarks have been seminal to inspire the development of new causal methods. In contrast to real-world data, synthetic data can explicitly model data characteristics such as the underlying functional relations and underlying data distributions to assess in detail how methods perform. Currently, a systematic assessment of the state-of-the-art of methods on the latest set of real-world data and comprehensive synthetic data is missing. We provide a detailed and systematic comparison of a range of methods on current real-world data and a novel collection of data sets that systematically models individual data challenges. Our evaluation also covers more recent methods missing in previous studies. The aim is to assist users in finding the most suitable methods for their problem setting and for method developers to identify weaknesses of current methods to improve them or develop new methods. The novel suite of data sets will be contributed to the `causeme.net` benchmark platform to provide a continuously updated and searchable causal discovery method intercomparison database.

**Keywords:** causal discovery, cause and effect, systematic analysis, data set, benchmarking

*. Corresponding author.

## 1. Introduction

Discovering cause and effect from purely observational data is a fundamental problem in science (Pearl, 2000b; Spirtes et al., 2000b; Peters et al., 2017). Causal information not only enriches the body of scientific knowledge, it also enables the prediction of the effects of potential interventions and allows to estimate the behavior of the system under yet unseen conditions. The gold standard of inferring cause-effect relations are controlled experiments, *i.e.,* to intervene into the system and observe the outcome. However, such experiments are often expensive, unethical, or practically impossible. Examples of this type occur in economics, medicine, or the Earth's climate system (Runge et al., 2019).

In recent years machine learning methods, such as probabilistic modeling (*e.g.,* Ghahramani, 2015), kernel machines (*e.g.,* Schölkopf and Smola, 2008), and in particular deep learning (*e.g.,* Goodfellow et al., 2016), have delivered astonishing results in a variety of fields in natural-, life- and social sciences, as well as engineering. However, at their core, many machine learning methods learn from correlations inherent in the data sets used for training. There is growing interest in utilizing causal relations instead in the machine learning community. The integration of machine learning techniques and causal inference methods is approached from either side. On the one hand, machine learning methods are used to improve causal inference in dealing with complex data, while on the other hand causal inference can help build better machine learning methods or to improve their performance by guiding them to learn from causal relations rather than from mere correlation (Schölkopf, 2019).

Causal inference and discovery may be regarded as relatively young fields of science, pioneered by Pearl (Pearl, 1995, 2000b), Spirtes, Glymour, and Scheines (Spirtes et al., 2000b), Dawid (Dawid, 2015, 1979), Imbens and Rubin (Imbens and Rubin, 2015), and Schölkopf, Janzing, and Peters (Peters et al., 2017), among others. Causal inference provides a rigorous mathematical formalism to relate statistical with causal dependencies. A predecessor of this is Reichenbach's common cause principle (Reichenbach, 1991) which, in its generalized form, is called the causal Markov condition (see, *e.g.,* Peters et al., 2017): If two variables $X$ and $Y$ are statistically dependent ($X \not\perp\!\!\!\perp Y$), then $X$ (potentially indirectly) causes $Y$, $Y$ (potentially indirectly) causes $X$, a common cause $Z$ (potentially indirectly) causes both $X$ and $Y$, or $X$ and $Y$ have a spurious connection due to implicit conditioning by selection bias.

However, *learning* causal relations from data is not possible without further assumptions on the data generation process, *i.e.,* the problem is not identifiable (see, *e.g.,* Spirtes et al., 2000b; Peters et al., 2017; Pearl, 2000a; Peters et al., 2014; Goudet et al., 2019; Quinn et al., 2011). Therefore, causal discovery algorithms rely on different further assumptions to identify causal relations. For example, the *constraint-based* multivariate structure learning algorithms PC or FCI by Spirtes et al. (2000b) rely on the Faithfulness assumption and use conditional independence tests. The Markov condition and Faithfulness provide a connection between statistical (conditional) dependence and causal relations that lays the foundation of unraveling causal graphs from observational data. However, conditional-independence based methods require more than two variables and cannot distinguish within Markov equivalence classes of graphs (see, *e.g.,* Spirtes et al., 2000b; Peters et al., 2017). An alternative to constraint-based algorithms are *score-based causal discovery* (Chicker-

ing, 2002a) methods. These require a likelihood function that specifies the likelihood of the observed data given a particular graph with an assumed parametric statistical model. Score-based methods like GES (Chickering, 2002a) still only recover Markov equivalence classes of graphs.

Recently, novel approaches make further more direct assumptions about the data generating process to move beyond Markov equivalence, *e.g.,* types of functional dependencies and noise distributions (see Appendix D for a more comprehensive list). By leveraging these assumptions, those approaches are also able to handle the bivariate case. Peters et al. (2017) summarize established approaches for these concepts. Most methods are based on the following principles (Janzing, 2019; Peters et al., 2017): (1) approaches using the complexity of the marginal and conditional distribution, (2) approaches relying on the independence of the cause and the mechanism, and (3) supervised approaches (see Section 2.2 for more details). Category (1) may also be called *asymmetry-based*. Note that this classification is not strict. Another type not considered here are methods for the time series case (Runge et al., 2019, 2023).

As is typical in machine learning, a main driver of innovation have been common data benchmarks. Seminal benchmark studies that have had an immense impact on bivariate causal discovery are those by Mooij et al. (2016) and Guyon (2013). These data collections are based on a wide range of real data from fields such as economics, medicine, and climate as well as synthetic data. Such data collections have been a particular driving force for supervised method innovations. Further, these data collections have become widely cited and have become a major resource for users that look for the best methods for their data set and problem setting.

However, as the authors of these benchmarks also point out, the real data benchmarks have limitations such as: (1) a limited number of data sets leads to a questionable significance of the results and may lead to biases, (2) methods have been applied to different benchmark data set versions over the years leading to inconsistent results, and (3) ranked method tables across a large number of data sets give the impression that one method outperforms all others, while the performances may actually vary widely. In particular, depending on method inherent assumptions, results vary for different data set properties such as functional dependence, noise distributions, sample sizes, *etc.* Importantly, in some fields those challenges are known and users are aware of them, from the structure of non-linearities to the type of noise terms. To compare methods and to estimate their applicability, it is essential to assess how methods perform under certain data characteristics. In contrast, method papers are often of limited use because new methods are evaluated on synthetic data sets created by the authors and compared to a suitable but restricted set of baselines. Furthermore, users will typically download code and use it "as is" due to a lack of knowledge about the method. This can be especially crucial since hyper-parameter adaptation and optimization is often left to the user rather than being implemented in the code.

Our contributions are as follows:

(1) a new synthetic data collection with a fine-grained differentiation of the characteristics of the data generation process, in particular functional dependence, cause and noise distributions, strength of dependence between cause and effect, and sample size,
(2) a brief "user-friendly" conceptual overview of a large number of older and more recent methods,

(3) comparative evaluations of methods on established data collections with methods used "as is" since this is how they are typically applied by users,
(4) a comprehensive analysis using our newly proposed data collection *w.r.t.* all factors relevant for the data generation process and their impact on methods performance as well as a meta-analysis, visualizations and a synthesis, and, finally,
(5) the data collection and results will be published and further extended on `causeme.net`.

In short, our main findings are that no method exists that outperforms all others in all investigated settings, but some methods have clear advantages over others for specific settings. We also identify characteristics where no current method delivers satisfactory results. Further, we found through a Shapley analysis that in our extensive experiments the functional dependence between cause and effect is, after the method itself, the most influential factor when it comes to prediction accuracy. The fact that the method itself is almost always the main contributor from a Shapley perspective suggests that choosing the right method for a particular set of data is an important decision, with no jack-of-all-trades being currently available. Finally, statistical analyses show that every single characteristic of the data generation process does have a significant impact onto the method's performance in general, while this might be different for individual methods. We hope that our study will help (1) users to find which method works best for their data set, and (2) method developers to improve existing and develop new methods that address current challenges.

Our focus is on the bivariate case to keep the scope of this investigation within reasonable bounds. But the idea is to initialize a growing database on the `causeme.net` platform which already contains benchmark data sets for the multivariate time series case featuring different challenges. Furthermore, some of the evaluated methods are multivariate and their advantages and deficiencies in the bivariate setting may typically also apply to the multivariate case.

The remainder of this manuscript is structured as follows. We introduce the problem setting of bivariate causal discovery in Section 2, followed by an introduction to established data collections for this problem scenario in Section 3. The following Section 4 introduces our proposed data collection, which includes a detailed explanation of the individual building blocks and the data generation protocol. Section 4 provides an overview of the used evaluation measures and result visualizations. We briefly introduce the considered methods in Section 6 before we present the broad structured investigation of the obtained results in the form of a comparative study in Section 7. The main manuscript is concluded in Section 8. Due to the huge amount of obtained results and to maintain a certain level of compactness, we postponed a number of additional results to the appendix. While we provide further comparative results from a data collection perspective in Appendices A to C, an additional introduction as well as in-depth evaluation of each considered method is presented in Appendix D. Further, we quickly touch additional methods related to bivariate causal discovery in Appendix E before we conclude the appendix with a visualization of our proposed data collection as density plots in Appendix F.

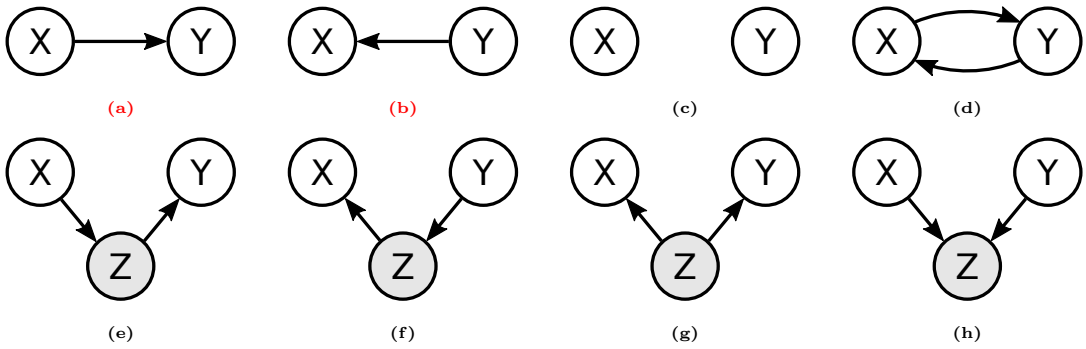



Figure 1: Possible relations of two observed variables $X$ and $Y$ including a (potentially unobserved) variable $Z$. The focus of this work is to evaluate methods that distinguish case **(a)** from case **(b)**.

## 2. Bivariate Causal Discovery

In the following, we briefly expand on the theoretical background of bivariate cause-effect-relationships and concepts of causal discovery. A comprehensive introduction is presented, *e.g.,* by Peters et al. (2017).

### 2.1 Problem Setup

The theory of causal inference lays out foundations of relating statistical dependencies with causal dependencies. In the atomic case given just two variables $X$ and $Y$, there are essentially five different ways how these two variables can be related (or unrelated) to each other: (1) $X$ (potentially indirectly) causes $Y$ (Fig. 1.a,e), (2) $X$ is (potentially indirectly) caused by $Y$ (Fig. 1.b,f), (3) $X$ and $Y$ are confounded, *i.e.,* a third variable $Z$ (potentially indirectly) causes $X$ and $Y$ while $X$ and $Y$ are not directly connected (Fig. 1.g), (4) $X$ and $Y$ have a spurious connection due to implicit conditioning by selection bias (Fig. 1.h), or (5) $X$ and $Y$ have no causal connection at all (Fig. 1.c). Further, any combination of (1)-(5) as well as cyclic connections (Fig. 1.d) can occur.

While these connections are possible, causal discovery methods will typically assume away some of these. In this study we focus on the dependent bivariate acyclic case. By the Markov assumption, we assume that the statistical dependence comes from a causal mechanism and by the causal sufficiency assumption we assume that no confounder of $X$ and $Y$ exists and, hence, either $X$ causes $Y$ or $Y$ causes $X$, *i.e.,* we consider the causal graphs shown in Fig. 1.a,b.

In case $X$ causes $Y$, this causal mechanism is defined by the following bivariate structural causal model (*SCM*) (Pearl, 1995, 2000b):

$$\begin{aligned} X &:= \epsilon_X \ , \\ Y &:= f_X\left(X, \epsilon_Y\right) \ , \end{aligned} \tag{1}$$

with some function $f_X\left(\cdot,\cdot\right)$ representing the causal mechanism by which $Y$ is set given the cause $X$ and a noise term $\epsilon_Y$ with $X \perp\!\!\!\perp \epsilon_Y$. $\epsilon_X$ and $\epsilon_Y$ follow some noise distributions $\mathcal{D}_X$

and $\mathcal{D}_Y$, respectively. A causal dependence $Y \to\!\!\blacktriangleright X$ would be defined vice versa. This SCM entails a joint distribution $P(X, Y)$ and corresponding conditional distributions $P(Y|X = x)$ and $P(X|Y = y)$. Pearl defines causal effects in this framework by considering *intervened* SCMs. Intervening on $X$ and forcing it to the value $x$, denoted as $do\,(X = x)$, implies to replace the assignment for $X$ in Eq. (1) by $X := x$. The resulting distribution is called an interventional distribution and denoted as $P(Y|do(X = x))$, different from $P(Y|X = x)$. Based on the interventional distribution a causal effect of $X$ on $Y$ is present if there exist values $x_1 \neq x_2$ such that:

$$P(Y|do(X = x_1)) \neq P(Y|do(X = x_2))$$

This is the case for the model $X \to\!\!\blacktriangleright Y$ above. On the other hand, intervening on $Y$ would not impact $X$.

The goal of bivariate acyclic causal discovery for dependent data is to infer whether $X \to\!\!\blacktriangleright Y$ or $Y \to\!\!\blacktriangleright X$ from a given sample $\{(x_i, y_i)\}_{i=1}^N$ of the distribution $P(X, Y)$. In addition, we assume the data to meet the following conditions: First, we restrict ourselves to the case of one-dimensional real-valued variables, *i.e.,* $x_i \in \mathbb{R}^{1\times 1}$ and $y_i \in \mathbb{R}^{1\times 1}$. Second, we also assume that the data is available in $N$ ordered pairs $\{(x_i, y_i)\}_{i=1}^N$, which are independent and identically distributed (*i.i.d*), drawn from the joint distribution $P(X, Y)$. That means there is no dependence between the data points which might be the case, *e.g.,* in time dependent processes. Further, we assume there is no selection bias.

In general, the question of which parts of a causal structure can in principle be inferred from the joint distribution is called structure identifiability. It is a known theoretical result (see, *e.g.,* Peters et al., 2017) that this is not possible for general SCMs, *i.e.,* SCMs with no restrictions *w.r.t.* the data generation process. Thus, assumptions on the underlying data generation process are key for the identifiability of cause-effect-relationships (see, *e.g.,* Peters et al., 2017; Pearl, 2000a; Peters et al., 2014; Goudet et al., 2019; Quinn et al., 2011).

As mentioned earlier, constraint-based (or conditional-independence based) causal discovery algorithms use the Markov and Faithfulness assumptions. An established example is the FCI algorithm (Spirtes et al., 2000a) or the PC algorithm (Spirtes et al., 2000b), the latter of which further uses the causal sufficiency assumption. An important restriction of constraint-based algorithms is that they require more than two variables and are restricted to Markov equivalence classes of graphs. In the bivariate case involving just two variables $X$ and $Y$, if they are dependent (*i.e.,* $X \not\perp\!\!\!\perp Y$), these algorithms will output $X \bullet\!\!-\!\!\bullet Y$, meaning that they conclude on an adjacency, but cannot determine its causal orientation. In fact, without any assumptions on the data generation process beyond the mentioned Markov and Faithfulness assumptions, or assuming time-ordered data (Runge et al., 2019), the bivariate problem is not identifiable (Peters et al., 2017), *i.e.,* the causal direction cannot be decided.

### 2.2 General Concepts and Methods

Assumptions are essential for the identifiability of the bivariate causal discovery problem. In the following we will introduce three main directions considered in the literature on bivariate causal discovery, following the taxonomy given by Janzing (2019) and Peters et al. (2017). A more detailed introduction for each individual method considered in our comparison is given in Appendix D. Note that the scope of our work is to provide insight into the performance of

the evaluated methods under various assumptions rather than to provide a comprehensive literature review.

We also do not further discuss causal discovery from interventional data, as proposed, for example, in Gao et al. (2022). While an interesting direction of research for multivariate causal disocvery, in the bivariate case interventional data trivially immediately provides identifiability.

#### 2.2.1 Using the Complexity of the Marginal and Conditional

The main idea is to factorize the joint distribution $P(X, Y)$ for both directions into the marginal and conditional, *i.e.,* $P(X)P(Y|X)$ and $P(Y)P(X|Y)$, and compare them according to an asymmetry in their complexities, hence this category can also be called *asymmetry-based*. Hence, a proper definition of the complexity is necessary. Having this definition at hand, the direction which yields the simpler model for $P(X)P(Y|X)$ or $P(Y)P(X|Y)$ is chosen as the causal direction *w.r.t.* the input data. This concept can also be linked to Occam's razor principle, which states that the simpler model should be favored (see, *e.g.,* Smith and Spiegelhalter, 1980). However, deciding about the causal direction by the simpler model can be implemented in various ways. For example, some approaches rely on a defined class of marginals and/or conditionals, which can be seen as a fixed complexity. This class covers, under certain class-defining assumptions, the joint distribution only for one direction and thus, the problem becomes identifiable. A member of this type of approach is ANM (see Appendix D.1 for details). Another interpretation is to decide the direction according to minimum regression errors. Here, the complexity is also kept constant in both directions by choosing a regression model class beforehand while the error decides which direction can be modeled better using the chosen complexity level (*e.g.,* RECI, see Appendix D.12). Further approaches rely on the minimum description length (*MDL*) principle (Rissanen, 1978). MDL states that there is an asymmetry inherent in the assumed data generation processes for both potential causal directions. To derive a decision, the true causal direction is assumed to be less complex. This asymmetry is often assessed by Kolmogorov complexity (see, *e.g.,* Chaitin, 1977), rather a theoretic construct than a practical or directly implementable measure. Thus, how the complexity is measured to decide according to the MDL principle is up to the individual methods. A more direct interpretation of measuring complexity is to fit different regression model types to the given data and evaluate which direction offers the simpler fit (*e.g.,* SLOPE, see Appendix D.13).

Multivariate score-based causal discovery can also be cast in this category. Here one chooses one or multiple graphs with respect to optimizing a scoring function, for example, built on the likelihood of the observed data given a particular graph and an assumed parametric statistical model (Peters et al., 2017). Searching over the space of graphs, however, grows super-exponentially, *e.g.,* (Chickering, 2002b) and greedy search techniques are often used (Chickering, 2002b,a). Our focus in this investigation is on the bivariate case.

#### 2.2.2 Relying on the Independence of Cause and Mechanisms

The core of this attempt to identify the causal direction is the assumption that the causal mechanism is "independent" of the cause, *i.e.,* mechanism and cause do not contain any information about each other. How this independence is assessed is, like the complexity in

Section 2.2.1, not straight-forward to define and does not imply any kind of statistical independence in general. Independence in that sense can be interpreted here in such a way that having information about $P(X)$ does not enable a shorter description of $P(Y|X)$ since the function or algorithm that maps $X$ to $Y$ is independent of $X$. Further, having information about the causal mechanism does not disclose any information about $P(X)$. Generally, relying on the independence of cause and mechanisms is a more theoretical handle on deciding upon a causal direction with fewer practical implementations being available. Nevertheless, a well known implementation is the IGCI method (see Appendix D.5). A further example is the formalization as algorithmic independence as described by Janzing and Schölkopf (2010) or Lemeire and Janzing (2013), an approach that is based on description length (compare Section 2.2.1). However, note that there is a relationship between the independence of cause and mechanisms, and the complexity of marginals and conditionals (see Section 2.2.1), as *e.g.,* detailed by Janzing (2019) and Peters et al. (2017).

#### 2.2.3 Supervised Approaches

While the approaches summarized in Sections 2.2.1 and 2.2.2 usually rely on a strict set of assumptions, algorithms belonging to this category treat the problem as a standard supervised learning problem. This means that they use exemplary data from pairs of $X$ and $Y$ in connection with knowledge about the causal directions of the training data to build supervised models. Related approaches might range from simpler models that are trained on more basic extracted features given an actual label of the causal direction to methods involving sophisticated neural network architectures to model the structure or functional dependence of given data (see Appendix E for examples). In this study we consider, *e.g.,* Jarfo and RCC (see Appendices D.6 and D.11), supervised methods that are trained on extracted features to decide for the causal direction. However, approaches belonging to this type face the usual problems of supervised learning algorithms, *i.e.,* they need sufficient amounts of representative training data, the complexity of model functions must be appropriate, *etc.* To approach this, models can be pre-trained using synthetic data that can easily be generated on a large scale. Those models might then be transferable to real-world data or even to out-of-distribution settings (see, *e.g.,* Li et al., 2020).

## 3. Data Collections

Evaluating methods in competing benchmarks has been a major driving force in the machine learning community. For example, the ImageNet challenges (Berg et al., 2010) took a great part in the success of deep neural networks. Neural nets were brought back to the attention of a wider audience in the form of convolutional networks through the work of Krizhevsky et al. (2011), which was able to win the ImageNet Large Scale Visual Recognition Challenge (*ILSVRC*) 2012 by a large margin. Further examples that drove the development in a certain field are the data sets (*e.g.,* Barrault et al., 2019) collected under the scope of the Workshop on Statistical Machine Translation (*WMT*) or the Protein Data Bank (*PDB*) library for protein folding (Berman et al., 2000). Thus, having a common benchmark to evaluate and compare methods can be expected to be very beneficial for the in this manuscript studied bivariate causal discovery problem.

Valuable works into this direction were already made, for example, by Mooij et al. (2016) with the Tübingen Cause Effect Pairs data set (*TCEP*, detailed introduction in Section 3.1.1) or by Guyon (2013) with the Cause Effect Pairs Challenge (*CEC13*, detailed introduction in Section 3.1.2). In the latter challenge, a method that decides according to collected data set features (*i.e.,* Jarfo, see Appendix D.6) was able to outperform methods with a much more thorough theoretical background[1]. Hence, investigating the practical performance in a comparison study can reveal the true strengths and weaknesses of methods rather than relying on theoretical assumptions alone.

Both data collections (*i.e.,* TCEP and CEC13) share the advantage that they comprise real as well as synthetic data. In contrast to, *e.g.,* the ImageNet challenge, where it is more straight-forward to collect and label images of real objects (*e.g.,* through social media or meta tags), it is much harder to collect measurements from diverse real-world problems with known underlying causal structure on a large scale. However, even if the performances of methods on these real-data problems are very interesting, the actual nature of the data (*e.g.,* noise distributions or true functional dependencies) is often unknown. Additionally, real-world data might suffer from biases and other uncertainties. Synthetic benchmark data have the advantage that underlying causal structures and functional dependencies are inherently known and the sample correctly reflects the underlying distribution.

Besides these bivariate data collections already mentioned, there are also multivariate data collections available in the literature (see, *e.g.,* Smith et al., 2011; Statnikov et al., 2012; Vowels et al., 2021). However, we want to focus on the bivariate causal setting data, and reconstructing bivariate causal structures from multivariate causal graphs is not straightforward. In essence, one would have to solve a more general problem, *i.e.,* unraveling more complex causal structures among multiple variables, to extract isolated bivariate problems. Although such methods exist, *e.g.,* (Spirtes et al., 2000b; Janzing et al., 2012b,c), they come with their own restrictions and inaccuracies that would eventually lead to less valuable bivariate data sets. See also (Peters et al., 2017) for further insight on this. Thus, we refrain from extracting bivariate problems from more complex data by ourselves and rely on established bivariate real-world data.

Our approach will be to build synthetic data sets in a systematic and detailed way, *i.e.,* we want to be able to identify the influence of the different building blocks of bivariate SCMs such as functional dependencies, noise distributions, dependency strengths, and sample sizes.

In Section 3.1 we will list our selection of current data collections, in Section 3.2 we discuss issues with these current data sources, and in Section 4 we introduce our proposed data collection.

### 3.1 Current Data Collections

In the following, we introduce three established data collections that we also include into our evaluation.

We use the following terms and notation: We refer to a single set of data, *i.e.,* a set of $\{(x_i, y_i)\}_{i=1}^{N}$ drawn from $P(X, Y)$, as a data set or realization of some underlying SCM. Each data set has a single label (*i.e.,* $X \to Y$ or $X \leftarrow Y$). Moreover, we will refer to $(x_i, y_i)$

1. However, Jarfo uses the result of some other approaches as features.

as a sample in the remainder (*i.e.,* $N$ is the sample size). We further refer to a collection for those data sets as data collection. Hence, *e.g.,* TCEP or CEC13 are data collections that contain several data sets. For synthetic data, a data collection might contain several configurations of data generating SCMs while a sample of sample size $N$ of each of these configurations is referred to as a realization (or data set). For example, suppose we consider two different configurations of a data generation process while we sample 10 realizations from each. We would obtain a data collection with 20 data sets (each of sample size $N$).

#### 3.1.1 Tübingen Cause Effect Pairs

The Tübingen Cause Effect Pairs data set (*TCEP*) is a growing collection of dependent real-world data with known causal structure (Mooij et al., 2016). The data originates from various problem domains, such as climate or medicine. It also includes a subset (*i.e.,* 41 sets) of the well known UCI machine learning repository (Dheeru and Karra Taniskidou, 2017). Thus, it comprises a collection of different functional dependencies, distributions, and sample-sizes. Due to the growing nature of the data collection, we pick the version 1.0 to operate with a fixed and defined state. We exclude five sets (*i.e.,* set 52, 53, 54, 55, and 71) due to different criteria (*i.e.,* multiple or categorical variables) as also done by Mooij et al. (2016). Hence, we end up with 95 valid data sets with 94 to 16 382 samples per set. Further, some data sets contained in the TCEP data are obtained from the same data sources. Thus, they can not be guaranteed to be independent, which might introduce biased results. To handle this, the authors introduce a weighting scheme where data sets from the same data source have the same weights while the weight of all corresponding sets sum up to one. Thus, we will consider the TCEP data twice: either with the proposed weighting (denoted as wTCEP) or without weighting (*i.e.,* each data set counts the same, denoted as TCEP). The data can be found at `webdav.tuebingen.mpg.de/cause-effect/`.

#### 3.1.2 Cause Effect Pairs Challenge Data

The well-known challenge for distinguishing cause from effect given data was last held in 2013 (Guyon, 2013). Later, a summarizing book was presented by Guyon et al. (2019). In the context of this challenge, a large collection of benchmark data was gathered and used for public training and validation as well as internal benchmarking purposes. The whole data is now freely available at `www.causality.inf.ethz.ch/cause-effect.php` and originates from various sources while comprising a collection of functional dependencies and data structures. For our data collection we rely on the final test data, but use only data with binary (treated as continuous) and continuous variables, real and synthetic data, and data with actual causal direction (leaving out independent and correlated), such that every method in the field of competitors can be treated equally. We refer to this data as *CEC13*. Thus, we use 1 584 data sets with 53 to 7 995 samples per set. Please note, there is at least some overlap between TCEP and CEC13 since both contain the data of the Pot-Luck competition held in 2008 (*i.e.,* eight data sets, see Mooij and Janzing, 2010).

#### 3.1.3 Cause Effect Data

The last data collection we consider from the literature is a subset of the data sets used by Goudet et al. (2018). While the authors use five different data collections in total,

we use the following three. First *CE-Net* that contains data generated by randomly initialized neural networks while the cause is generated with randomly picked distributions (*e.g.,* exponential, gamma, log normal, Laplace). Further, we use the *CE-Gauss* data which is sampled with the data generator provided by Mooij et al. (2016). This mechanism is realized by random Gaussian processes while the cause data originates from randomly generated Gaussian mixtures. Finally, we employ *CE-Multi*, a data set generated by employing post additive noise ($Y = f_X(X) + \epsilon_Y$), post multiplicative noise ($Y = f_X(X) \cdot \epsilon_Y$), pre-additive noise ($Y = f_X(X + \epsilon_Y)$), or pre-multiplicative noise ($Y = f_X(X \cdot \epsilon_Y)$) models with functions $f_X(\cdot)$ realized by linear and polynomial mechanisms. Each of these three data sets consists of 300 realizations with 1 500 samples each. The data can be found at `dx.doi.org/10.7910/DVN/3757KX`. We do not use their version of the CEC13 data since we derive our own more comprehensive version of it.

### 3.2 Problems and Challenges of Current Benchmarks

As pointed out by most authors, current data collections for benchmarking purposes suffer from several limitations, which will be summarized in the following.

*Comparison studies in method papers:* Firstly, comparison studies within method papers are often of limited use regarding an assessment of suitability for an application case because the new methods are evaluated on synthetic data sets that are tailored to the considered challenge and thus tend to favor the new method and are compared to only a limited set of baselines. What we refer to here are hypothetical cases like the following. Suppose we have method $A$, which is designed to operate in settings with very little data. A study presenting this work would thus likely include experiments with reasonable-sized data to show a fair comparison *w.r.t.* its baselines as well as a comparison on very small data sets to show its advantages there. If now a method $B$ is able to handle very noisy data, a study presenting this would most likely present reasonable-sized data with small and extreme amounts of noise. Thus, when we are faced with very little data that is also very noisy, it is not that clear if we should go as a user for method $A$ or method $B$ without further investigation.

*Limited data collection size:* This is especially apparent for real-world data where collecting sufficiently many data sets including necessary ground truth information requires a significant effort. Next to the difficulty of assessing the significance of comparison results, a small number of such labeled data sets also may be insufficient for supervised causal discovery approaches.

*Unknown biases:* It is known that large data collections such as ImageNet contain biases that come with implications for the resulting algorithms. For example, 45.4% of the images contained in ImageNet come from the US (Shankar et al., 2017). Given the manual selection process, also the current bivariate cause-effect benchmarks may come with significant biases leading to non-representative comparison studies.

*Uncertainty about underlying causal truth:* Further, real data might suffer from measurement noise, errors such as missing data due to sensor fault, or just the wrong measurement frequency or resolution. Then the issue is that the labeled ground truth for such a data set

may not actually reflect the actual ground truth of that contaminated data set. For example, cyclic feedback in time-dependent processes may imply different causal directions for different sampling frequencies. Measurement noise, especially when correlated, may induce a latent confounding or again a cyclic dependency.

*Unknown data generating process characteristics:* Most relevant to our work is that the true nature of the data generating process, *i.e.,* the data distributions and particular functional dependencies, are often not known. However, in some fields those data characteristics are actually known and users are aware of them, from the structure of non-linearities to the type of noise terms. Some well known examples are found in climate sciences, where *e.g.,* noise distributions can often reasonably assumed to be Gaussian or, in the case of precipitation, to follow an extreme value distribution. Given that no method outperforms all others for all cases (see Section 7.3.1), it is essential to assess how methods perform under certain data characteristics.

*Benchmarks are outdated:* Since methods are sometimes applied to different versions of data collections over the years, inconsistent results can appear. For example, TCEP is a collection of data that is growing due to newly data being added over time. Thus, there exist different versions of the data set, which lead to different challenges and thus different reported results. If we take the well known LiNGAM method as an example (see Appendix D.7 for further details), this method was able to achieve an accuracy of $\sim 62\%$ on average in the work of Mooij et al. (2010) but could only reach $\sim 45\%$ in a comparison done by Tagasovska et al. (2018). These differences emerged since the first result was obtained with 68 pairs while the second one uses 99 pairs. Further, authors sometimes do not state if they rely on the weighted evaluation proposed by Mooij et al. (2016) or if they count plain true and false predictions. Additionally, in the case of LiNGAM, there exist different versions of the algorithm (direct and ICA, see Appendix D.7), which leads to further differences. In contrast to TCEP, the data collected for CEC13 has, to the best of our knowledge, rarely been used in recent studies. Thus, even though both data collections, TCEP and CEC13, are helpful, a recent and broad comparison analysis covering the newest state-of-the-art methods is missing.

For these reasons the ranking tables in current benchmark papers may give the wrong impression that one method outperforms all others while the actual performance varies widely across individual data sets depending on the inherent data characteristics. Our study addresses this problem.

## 4. Our Data Collection

Current issues of available data collections for bivariate causal discovery and the benefits that benchmarks offer to the community motivated us to propose a novel collection of data sets. To study the performance of methods under different data characteristics, *i.e.,* functional dependencies, distribution types, and sample size, we propose an extensive new synthetic data collection. We do not model any time-dependence, confounding, or selection bias explicitly. Given that some methods come with inherent assumptions, our approach allows us to assess methods' performances under these ideal conditions as well as condi-

| Name | Function |
|---|---|
| lin_a | $Y = X + \epsilon_Y$ |
| add_a | $Y = X^2 + \epsilon_Y$ |
| add_b | $Y = 5.0 \cdot X^3 - 3.0 \cdot X + \epsilon_Y$ |
| add_c | $Y = sin(10.0 \cdot X) + e^{3.0 \cdot X} + \epsilon_Y$ |
| mul_a | $Y = 3.0 \cdot X^3 + X \cdot \epsilon_Y$ |
| mul_b | $Y = log(X + 1.01) + sin(4.0 \cdot X) \cdot \epsilon_Y$ |
| mul_c | $Y = sin(10.0 \cdot X) + e^{3.0 \cdot X} \cdot \epsilon_Y$ |
| com_a | $Y = 3.0 \cdot (X^7 - X^3) + (X + 0.75) \cdot \epsilon_Y + \epsilon_Y$ |
| com_b | $Y = 5.0 \cdot X^3 - 5.0 \cdot X + (5.0 \cdot X^3 - 5.0 \cdot X) \cdot e^{\epsilon_Y} + 0.5 \cdot (3.0 \cdot X) \cdot \epsilon_Y$ |
| com_c | $Y = log(X + 10.0) - \lvert X \rvert^{\epsilon_Y} + \epsilon_Y$ |

Figure 2: Overview of functional dependencies to generate synthetic data. For simplicity of display, we use $X$ as cause and $Y$ as effect variable while $\epsilon_\cdot$ denotes corresponding noise terms.

tions that do not meet these assumptions. Our hope is that this systematic assessment will help (1) users to find which method works best for certain characteristics of the underlying process, and (2) method developers to overcome deficiencies by studying for which characteristics methods tend to fail.

Section 7 provides a comprehensive evaluation of the impact of different aspects of the data generation process in form of a meta-analysis and synthesis that is further extended in Appendices A to C. Additionally, in Appendix D we provide an in-depth study on how individual methods perform for different characteristics of the data.

### 4.1 Building Blocks

We consider the following five different data set characteristics and refer to them as building blocks in the remainder of this manuscript (*i.e.,* we use building blocks as synonym for data set characteristics). These characteristics emerge from the assumption of an underlying SCM (see Section 2.1) and are systematically modified in a number of ways. Thus, we refer to this data collection as the "bivariate structural causal model characteristics" (*bSCMC*) data collection.

*(1) Functional dependence:* We consider four families of functional dependencies: linear, additive, multiplicative, and a more complex type. Since the space of possible realizations for each of these functional families is huge, we select a few inspired by Mitrovic et al. (2018) and list these in Fig. 2. Note, the color coding of the individual functions is consistent over the remainder of the manuscript.

*(2 & 3) Cause and noise distributions:* We consider five different distributions: uniform, normal, skewed normal, bimodal normal, and exponential. These types of noise distributions are considered either for the cause variable (*i.e.,* $X = \epsilon_X$), or as the noise term $\epsilon_Y$ of $Y$ in Fig. 2. The distributional parameters of these distributions were chosen as discussed in the detailed data generation protocol (see Section 4.2).

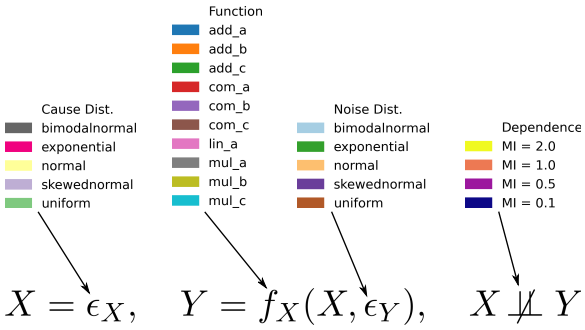



Figure 3: The structural causal model used for bivariate causal discovery with the considered configurations.

*(4) Mutual information between cause and effect:* The former three characteristics, *i.e.,* functional dependence and distribution types, have a strong impact on the dependency strength between cause and effect. We would argue that a categorization according to different dependency levels is much more consistent than a categorization using distributional parameters (*e.g.,* the standard deviation $\sigma$ for normal distributions). To differentiate the influence of functional and distributional characteristics from the strength of the resulting dependency, we characterize the data sets by different levels of mutual information ($MI$). We choose the levels $MI = \{0.1, 0.5, 1.0, 2.0\}$ that roughly match the values that we observed in established real data collections. Details on how the mutual information is computed and how the data is tuned to reach these levels of MI will be discussed in Section 4.2. Note in this manuscript, MI measures are given in natural units.

*(5) Number of available examples:* We include two different data set sample sizes, *i.e.,* 100 and 1 000 samples, to account for the influence of the number of available samples from the joint distribution $P(X, Y)$. Due to the huge runtime requirements of some algorithms, we were not able to use an even higher number of examples.

We illustrate these building blocks in Fig. 3 based on the underlying SCM (compare Section 2.1). Here, it can be seen that the cause variable $X = \epsilon_X$ is distributed in a certain way. Then, there is a certain functional dependence between $X$ as well as $\epsilon_Y$, and $Y$ depicted as $f_X(X, \epsilon_Y)$. Finally, due to this functional dependence, $X$ and $Y$ have a certain dependence strength measured by mutual information. We list the considered choices for each of these building blocks above while we identify each of them by a unique color code. This color code is consistent over the remainder of this manuscript.

### 4.2 Detailed Data Generation Protocol

The detailed protocol for the data set generation itself is divided into two phases as described in the following. The choices for functional dependence, as well as cause and noise distribution types are categorical while we want to consider every possible combination of those. In contrast, the adjustment of distribution inherent parameters to reach the four desired mutual information levels has to be performed by optimization.

*Phase (1):* We generate data for all ten function types, all five noise distribution types, and all four mutual information levels as initial data sets while we only consider the uniform cause distribution for now. To account for the different levels of MI, we adjust the distributional parameters of the noise distributions accordingly (*i.e.,* leaving the uniform cause distribution untouched). In detail, we build each possible configuration by successively selecting every combination out of the ten function types and the five noise distribution types. We then optimize the distributional parameters of the noise distribution from each configuration to successively meet the four desired MI levels. Therefore we conduct a fine grid search over different values of the distributional parameters (*e.g.,* over different values of $\sigma$ for normal noise distributions) while we use 100 realizations per grid value to estimate the mutual information. Each realization contains 10 000 data points sampled utilizing the configured SCM. The MI itself is estimated by the kNN estimator described in (Kraskov et al., 2004) with $k = 3$. We search for the MI values closest to the desired levels $MI = \{0.1, 0.5, 1.0, 2.0\}$ (see Section 4.1) and use the corresponding distributional parameters of the noise distributions in our configurations. If the closest MI value differs more than 0.1 from the desired levels, we consider this configuration as invalid. Thus, we obtain the first portion of our data collection in this first phase, *i.e.,* 200 configurations build from 10 functions $\times$ 1 cause distribution $\times$ 5 noise distributions $\times$ 4 MI levels.

*Phase (2):* In a second phase, we keep the distributional parameters of the noise distributions found in the first phase constant while we consider the remaining cause distribution types (*i.e.,* normal, bimodal normal, skewed normal, and exponential). Suppose we found a configuration represented by the tuple (functional dependency: add_a, cause distribution: uniform, noise distribution: normal, dependency strength: MI= 0.5) in the first phase. In this second phase we keep the $\sigma$ of the normal noise distribution fixed at its value determined in the first phase, while we successively replace the uniform cause distribution with the other four distribution types and optimize the corresponding distributional parameters until the resulting mutual information meets the four desired levels. Therefore, we conduct another grid search similar as done in phase (1). Eventually we obtain the remainder of our data collection consisting of 800 configurations (*i.e.,* 10 functions $\times$ 4 cause distribution $\times$ 5 noise distributions $\times$ 4 MI levels). Note, there are again emerging some invalid configurations here.

As mentioned, certain MI values were not possible to achieve under some combinations of distribution types and functional dependencies (see Fig. 4 for examples). Hence, we had to omit some invalid configurations and ended up with 828 valid out of 1 000 possible setups (*i.e.,* 10 functions $\times$ 5 cause distribution $\times$ 5 noise distribution $\times$ 4 MI dependency levels).

Then, each of these 828 valid setups is sampled with a sample size of 100 and 1 000. Finally, to reliably estimate the accuracy for a given configuration and sample size, we generate 100 realizations each. Thus, we end up with 165 600 generated data sets on which each method is evaluated while we obtain method accuracies for 1 656 different configurations and sample sizes. To avoid potential variable order effects, the underlying order, *i.e.,* $\{x_i, y_i\}_{i=1}^N$ or $\{y_i, x_i\}_{i=1}^N$, for each realized data set is chosen at random.

In Fig. 4 we visualize the resulting composition of our bSCMC data given the considered building blocks. There are two sets of rings. The outer three rings indicate the configuration

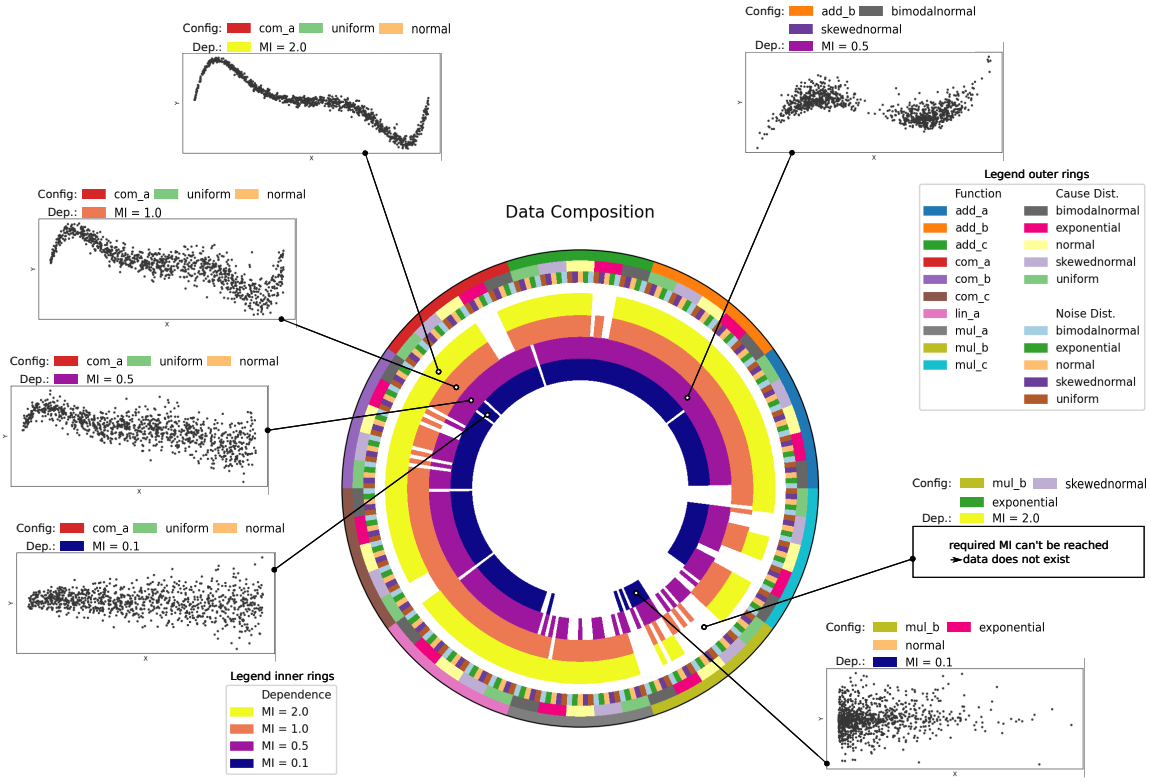



Figure 4: Data composition overview with examples.

of the data generation process regarding functional dependencies, cause distribution, and noise distribution (from outside to inside). The color coding is explained in the legend on the top right. Hence, each configuration given by the outer three rings leads to a sector of the inner circle with a given configuration defined by the tuple (function, cause distribution, noise distribution). The inner four rings now indicate the MI dependency strength of that configuration with the corresponding legend on the lower left. The white gaps in these inner four rings symbolize that the corresponding mutual information could not be achieved with the configuration defined by the outer three rings. In sum, for each colored circle element in the inner four rings, we have a valid configuration where we can sample our realizations from, *i.e.*, 100 realizations for $N = 100$ and $N = 1000$ samples. Some data set examples are given as scatter plots (with additionally indicated configurations, the true model is here always $X \dashrightarrow Y$ for visualization purposes). We additionally show the individual densities our data is sampled from in Appendix F. Note that the color codes in Fig. 4 are consistent in the whole manuscript.

## 5. Evaluation Measures and Result Visualization

In the following, we explain the used metrics to evaluate the considered methods and the influence of individual characteristics.

Our approach is to evaluate the methods "as is", *i.e.,* as they are provided in the method papers, either by the code supplied by the authors or by established publicly available implementations. We only ensure that the data is preprocessed as suggested by the authors, *i.e.,* centered, standardized, *etc.* The reason for this approach is that users will typically download code and use it as implemented due to a lack of knowledge about the method or because hyper-parameter choices and optimization is often left to the user rather than being implemented in the code.

Please note that we had to remove the detailed analysis of the individual methods from the main manuscript and move it to the appendix in order to maintain a certain level of compactness, as the whole study comprises a large number of results. Therefore, some of the following introductions to evaluation measures might be only relevant to the appendix, but in order to keep the structure of the manuscript reasonable, are presented here anyway.

### 5.1 Accuracy

The compared methods come with different design philosophies making an intercomparison of methods more challenging. We need to consider the following facts: Firstly, some algorithms only give a hard decision about the assumed causal direction (even though they might be based on real-valued numbers hidden in the author's code in some cases) and other algorithms might provide a real-valued decision. This makes an evaluation using, *e.g.,* ROC-AUC as typically done in literature in this setting impossible since the results can not be sorted properly to obtain the ROC curves. Secondly, while some algorithms predict two classes, *i.e.,* $X \to Y$ or $X \leftarrow Y$, others are more potent in that they predict among up to four cases, *i.e.,* $X \to Y$, $X \leftarrow Y$, $X \perp\!\!\!\perp Y$, or $X \not\perp\!\!\!\perp Y$. Further, the execution of some algorithms might fail due to technical issues. Hence, comparing those algorithms among each other is hard. To solve this, we follow Mooij et al. (2016) and "force" any algorithm into predicting $X \to Y$ versus $X \leftarrow Y$ by counting every decision to the contrary as wrong (*e.g.,* $X \perp\!\!\!\perp Y$ or $X \not\perp\!\!\!\perp Y$ are wrong).

With this approach we rely on accuracy for evaluating the performance of the considered methods, *i.e.,* the number of correctly identified data sets divided by the number of data sets in total, as follows:

$$ACC = \frac{\sum_{m=1}^{M} \delta_{d_m, \hat{d}_m}}{M} \ .$$

Here, $M$ is the total number of considered data sets and $\delta_{i,j}$ the Kronecker delta that equals one if $i = j$ and zero if $i \neq j$. Further, $d_m$ is the ground truth direction of the $m$th data set (*i.e.,* $X \to Y$ or $X \leftarrow Y$) while $\hat{d}_m$ is the prediction obtained from an algorithm (*e.g.,* $X \to Y$, $X \leftarrow Y$, $X \perp\!\!\!\perp Y$, or $X \not\perp\!\!\!\perp Y$). Invalid configurations are ignored in the accuracy computation.

Note that accuracy can be calculated for different scenarios, *e.g.,* a single accuracy for a method over all available configurations in the bSCMC collection (*i.e.,* performance of a method over the entire bSCMC data set). The accuracy can also be computed for subsets

of configurations, *e.g.,* only considering those data sets built with a linear functional dependence. In consequence, the number of considered data sets $M$ does change according to the scenario, but is at least $M = 100$ since we sample 100 realizations per configuration. Assuming Bernoulli trials, the estimation error $\sqrt{\mathrm{ACC}(1-\mathrm{ACC})/M}$ for an accuracy estimate of $ACC = 95\%$ with $M = 100$ is about 2% and for $ACC = 50\%$ with $M = 100$ it is 5%.

### 5.2 Invalid Decisions

A result of the used accuracy measure is that algorithms that are able to predict more than two classes as well as those that fail during execution tend to have a lower accuracy than those that only predict two causal directions. More precisely, an algorithm that can only predict two classes (*i.e.,* $X \to Y$ or $X \leftarrow Y$) has a 50% chance to be correct. In contrast, two of the possible predictions of the four-class-algorithms (*e.g.,* $X \to Y$, $X \leftarrow Y$, $X \perp\!\!\!\perp Y$, or $X \not\perp\!\!\!\perp Y$) are counted as wrong in any case implying that the random baseline is at 25%. To account for this and to make this effect more apparent, in the analysis we give, in addition to the accuracy numbers, the number of invalid decisions to put the accuracy into context for every method.

### 5.3 Visualization as Footprints

For presenting our results, we refined the technique from Tagasovska et al. (2018) and visualize individual results as circular *footprints*. In detail, obtained accuracies are presented in a circular form while the corresponding legend is depicted by the color coding on the outer rings (compare Section 4.2 and Fig. 4). We use this form of presentation for all considered data collections as well as for detailed analysis of our proposed data collection. The corresponding detailed result footprints can generally be read as follows. As mentioned, the outer ring depicts the building block (*i.e.,* dimension in the multi-dimensional array of results) the results are split along. For example, in Fig. 6.e, we see results for our proposed data collection for every involved method. Thus, we consider the results for individual building blocks to be aggregated (*e.g.,* averaged over all involved functional dependencies, *etc.*; *i.e.,* $M = 165\,600$, see Section 5.1). In contrast, in Fig. 28.a we see the detailed results for ■ ANM on data generated with the ■ add_a function. Here we have all the remaining building blocks either represented in the three outer rings or in the overlaid plotted curve (*i.e.,* $M = 100$, see Section 5.1). Thus, there is no dimension being aggregated and we present the obtained accuracies for every single valid configuration. Since there is no data available for some configurations (see Section 4.2 for details), some accuracies cannot be plotted which might result in curvy structures in the result footprints (*e.g.,* plot for ■ mul_b in Fig. 28.f). Please note, color coding of each legend entry is unique and consistent over the whole manuscript.

### 5.4 Visualization as Violins

In addition to plotting the obtained accuracies as footprints (see Section 5.3), we are also presenting the actual distribution of accuracies across different aggregations of results as violin plots.

For example, if we are interested in the distribution of results *w.r.t.* individual methods such as in Fig. 8, we aggregate the results over all the data set characteristics and represent the obtained distribution as violin. In contrast, if we are interested in a particular characteristic as for example in Fig. 9, we split the results in the individual configurations for this particular characteristic and aggregate the results for all other characteristics (or even methods) to obtain the presented violins.

Please note, the width of the violins is scaled to unit width individually and thus the area of individual violins might appear different.

### 5.5 Relative Importance of Characteristics

Our data collection comprises various conditions to challenge the methods evaluated. We employ Shapley values[2], introduced by Shapley (1951), to obtain insight into the importance *w.r.t.* methods performance of each of these characteristics.

Shapley values are a concept from cooperative game theory. It was developed for the purpose of fairly distributing the total surplus value created by the coalition of all players involved, depending on the contribution made. To do so, the marginal contribution of each player is computed. To apply this concept to explainability of model predictions, the prediction itself can be seen as the game played while the predicted value is the outcome of the game. The input features used to make the prediction are the players. By applying this concept, a contribution can be assigned to how much each individual input feature value contributes to the actual prediction of a particular model. In the context of explaining the prediction of a model, Shapley values depict how much a particular input changes the actually obtained output away from the expected mean output. However, to gain insight into the importance of the data set characteristics *w.r.t.* individual methods performance, we have to consider the following two aspects.

First, we are interested in the impact of characteristics on the performance of individual bivariate causal discovery methods. Applying Shapley's analysis to these methods directly would, if possible at all, result in an explanation of the estimated causal direction *w.r.t.* the sampled realization. To circumvent this, we translate the problem to explaining a hidden model that takes the characteristics as input (*i.e.,* the particular data set configuration such as, *e.g.,* described in Section 4.1 including the causal discovery method used) and outputs the obtained accuracy of the particular bivariate causal discovery method. Please note, accuracies are obtained by applying the actual bivariate causal discovery method to 100 sampled realizations (see Section 5.1). We realize this hidden model by gradient boosting regression trees[3] that are known to be a reasonable choice for tabular data (Grinsztajn et al., 2022). In particular, we translate each characteristic into a one-hot encoding (even mutual information and sample size for consistency reasons) and train an auxiliary regressor to estimate the obtained real-valued accuracy from the individual data configuration and the used bivariate causal discovery method as categorical inputs. The upper bound of regression

2. We use the *SHAP* python library.
3. We use the *XGBoost* python library.

error as root mean squared error (*RMSE*) of the auxiliary regressor lies at 0,82 % accuracy[4] and thus we are confident that it is able to capture the hidden model well[5].

Second, Shapley values can explain individual model outputs by assigning importance to its particular input features. Since we are interested in the general importance of the characteristics, we have to aggregate the individual values. To do so, we use the mean absolute Shapley values aggregated over every corresponding Shapley value obtained for each individual regression of accuracy. Thus, we obtain the mean absolute change a certain characteristic causes in comparison to the expected mean output of the model (*i.e.,* larger numbers imply more change and thus higher importance).

As an example, suppose we are interested in the importance of the characteristics for the bivariate causal discovery method ANM (see Appendix D.1 and Fig. 26). First, we need to train an auxiliary XGBoost regressor with every valid configuration of characteristics (see Section 4.1) as categorical inputs and by the method ANM obtained accuracies as outputs (see, *e.g.,* Fig. 28). This results in a regression problem with 1 656 data points (*i.e.,* 10 functions × 4 cause distribution × 5 noise distributions × 4 MI levels × 2 sample sizes − invalid cconfigurations) and thus 1 656 individual explanations where each characteristic in every explanation has an assigned Shapley value. We report the mean absolute values of these individual values to aggregate the information and thus obtain general importance of characteristics. Please note, to obtain results such as in Fig. 12, we treat the method as another categorical input to the auxiliary regressor.

### 5.6 Statistical Analysis of Differences Between Individual Configurations

The goal of our data collection is to assess how methods perform under various data set characteristics. To evaluate whether characteristics have an impact on method performances, we perform significance testing as follows.

Every single configuration given by a tuple of (functional dependency, cause distribution, noise distribution, dependency strength, sample size), yields an accuracy evaluated on 100 realizations (*i.e.,* $M = 100$, see Section 5.1). To evaluate whether a certain building block leads to significantly different results, we divide those accuracies into sets according to the single configurations contained in this building block.

Suppose we are interested in the results *w.r.t.* different values of mutual information. Then we would obtain four sets of accuracies, one for each mutual information value in $MI = \{0.1, 0.5, 1.0, 2.0\}$. Each of these sets contains a maximum of 500 accuracy numbers (*i.e.,* 10 functions × 5 cause distributions × 5 noise distributions × 2 sample sizes) minus the invalid configurations (see Section 4.2).

4. Please note that we regress the accuracy of the actual bivariate causal discovery methods. Thus, a RMSE of 0,82 % accuracy means that the auxiliary regressor miss-predicts the performance of the bivariate causal discovery methods by less than 1 % given a certain data configuration. We also do report the Shapley values on the data used to train the auxiliary regressor (*i.e.,* there is no train-test-split).
5. Regressing the accuracy from nicely defined features (*e.g.,* known function type, *etc.*) is far easier than regression using raw input data (and infer *e.g.,* function type first). Thus, this can not be seen as a recommender system.

The individual sets are then compared pairwise using the Mann-Whitney U test[6]. The Mann-Whitney U test is a non-parametric test that compares if samples (*i.e.,* the accuracies in our case) are drawn from the same distribution. The test does not assume that the inputs are normally distributed, which is most likely not true in our case. Further, the Mann-Whitney U test is more robust to outliers in comparison to the commonly used t-test (Fay and Proschan, 2010). The test's null-hypothesis is that both samples examined have the same underlying distribution. Thus, when the obtained p-value is lower than the test's significance level (5% in our case), we can reject this null-hypothesis and conclude that there is a significant difference in results for the tested characteristics.

Please note that the analysis presented here is meant to assess whether two tested configurations lead to significantly different results. Whether a single building block has a generally significant influence on the resulting accuracy is measured with conditional independence tests as presented in Section 5.7. Note further that the significances presented here do not measure how good an algorithm is suited for a certain setting, but only measure that two configurations yield different accuracies for the algorithm. Further, our evaluation does not consider compound effects due to a combination of different characteristics, *i.e.,* we can only conclude if a tested building block does have a significant impact on its own, or not.

### 5.7 Statistical Analysis of Dependence of Building Blocks and Accuracy

In addition to Section 5.6, where we are interested in the pairwise independence between two groups of results (*e.g.,* are the results significantly different for data with ■ 100 and ■ 1000 samples), we are further investigating the conditional independence between the configuration of data set characteristics and the obtained accuracies.

To do so, we apply a regression-based conditional independence test[7] as follows: To test $A \perp\!\!\!\perp B|C$, where $B$ is the target (accuracy in our case), $A$ the tested feature of interest (a certain building block in our case), and $C$ the remaining features (the other building block in our case), the regressions $B|AC$ versus $B|C$ are compared. For that, a goodness-of-fit statistic (deviance), here the sum of squares of residuals, is employed. If the fits of the respective regressions do not differ significantly (measured using the deviance), the null hypothesis of conditional independence is "accepted". Through a one-hot encoding of categorical features, this approach can be conducted with a multivariate linear regression.

To obtain conditional independencies of building blocks and accuracy, we apply the following pipeline. We have accuracies obtained for each of the 14 methods on every of the 828 valid combinations of data set characteristics with either 100 or 1 000 samples per realization (see Section 4). First, we transform every characteristics into categorical representations by encoding each individual characteristics with class numbers (not into one-hot encoding, compare Section 5.5; this is done by the employed test internally as described above). Then we perform conditional independence tests among those categorical inputs *w.r.t.* the real-valued accuracy (*e.g.,* Func. Type $\perp\!\!\!\perp$ Accuracy | [Cause Dist., Noise Dist.,

6. We use *scipy.stats.mannwhitneyu* to perform the tests. Please note, we can not use a paired sample test such as the Wilcoxon signed-rank test due to the nature of our proposed data collection. In detail, invalid configurations lead to cases where accuracies might be present only in one sample (*e.g.,* a certain MI could not be reached under particular circumstances, see Section 4.2).
7. We use *RegressionCI* from the *Tigramite* python package.

| Method | Short Description | Introduction |
|---|---|---|
| ANM (Hoyer et al., 2009) | independence of residual and cause | Appendix D.1 |
| CGNN (Goudet et al., 2018) | fitting of data generation process using neural networks | Appendix D.2 |
| EMD (Chen et al., 2014) | complexity as distance of marginal and conditional to uniform reference | Appendix D.3 |
| GPI (Mooij et al., 2010) | model selection | Appendix D.4 |
| IGCI (Daniusis et al., 2012) | differential entropy | Appendix D.5 |
| Jarfo (Fonollosa, 2016) | supervised using ensemble of features | Appendix D.6 |
| LiNGAM (Shimizu et al., 2006) | ICA to unravel causal order | Appendix D.7 |
| NLME (Hyvärinen and Smith, 2013) | likelihood | Appendix D.8 |
| PNL (Zhang and Hyvärinen, 2008) | independence of residual and cause | Appendix D.9 |
| QCCD (Tagasovska et al., 2018) | quantile scoring | Appendix D.10 |
| RCC (Lopez-Paz et al., 2015) | supervised using obtained features | Appendix D.11 |
| RECI (Blöbaum et al., 2018) | lower regression error | Appendix D.12 |
| SLOPE (Marx and Vreeken, 2017) | model complexity and residuals measured via bit length | Appendix D.13 |
| SLOPPY (Marx and Vreeken, 2019) | model complexity and residuals measured via scoring function | Appendix D.14 |

Figure 5: Method overview.

MI, #Samples] or Cause Dist. $\perp\!\!\!\perp$ Accuracy | [Func. Type, Noise Dist., MI, #Samples], *etc.*). The null-hypothesis of the employed test is conditional independence. If the obtained p-value is below the set significance level (5% in our case), we can reject this null-hypothesis and conclude dependence for the tested characteristic and the obtained accuracy. Thus, we can obtain information if a certain building block influences the performance of the methods in general. Please note, to obtain results such as in Fig. 13, we treat the method as just another categorical feature.

## 6. Brief Overview of Competing Methods

The main goal of this work is to investigate the implications of various data set challenges onto the performance of different state-of-the-art bivariate causal discovery methods. We concentrate on a comparative analysis and overview of obtained results in the main manuscript. However, due to the huge number of results obtained during our experiments, an explanation of individual methods and in-depth evaluation of their obtained results is given in Appendix D.

A very brief overview of the methods considered including a brief description can additionally be found in Fig. 5, while the interested reader is referred to Appendix D for more details. In total, we employ 14 methods for our systematic investigation that cover a range of commonly applied methods as well as various concepts. Although we have tried to cover as many methods as possible, there are many more methods available in the literature that could be considered for our study. A non-exhaustive list of missing methods including some elaboration is given in Appendix E. Further, we plan to host our data collection on `causeme.net` for public usage, which will probably lead to more and more different methods with results being available in our proposed data collection over time.

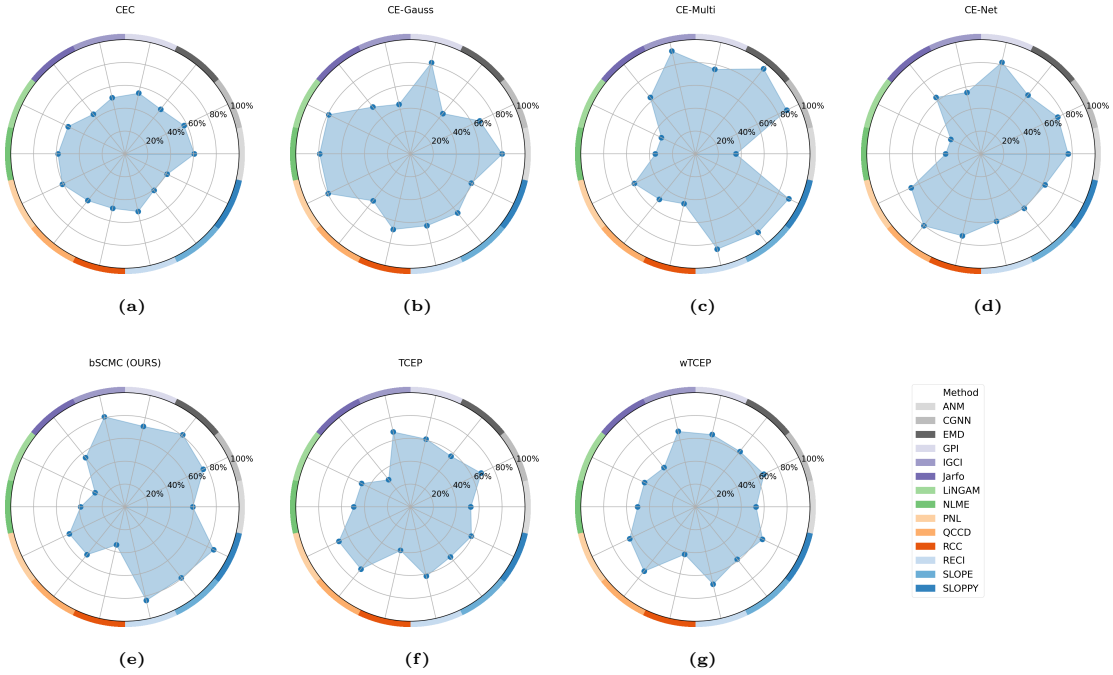



Figure 6: Data-collection-wise comparison of the individual methods as footprints.

## 7. Comparative Analysis and Synthesis

In the following we present a comprehensive comparison and synthesis of the obtained results. We present results on each considered established data collection (see Section 3.1) as well as an in-depth evaluation on our proposed data collection (see Section 4). By synthesizing different visualizations and performing a meta-analysis of individual data set characteristics, we can determine which aspects have the greatest impact on the accuracy of the methods. We are able to identify cases that are well covered by the investigated methods with high accuracy. However, we also unveil scenarios that cannot be reliably solved. Please note, this study is extended in Appendices A to C due to the number of results.

Section 7.1 presents a general overview of the obtained results as footprints. Then, we present in Section 7.2 the average accuracies and rankings of methods for different scenarios. Finally, we take a detailed look into various aspects using the detailed building blocks of our proposed data collection bSCMC.

In addition to the aspects we explore in this manuscript, another missing but nevertheless important aspect for users is the runtime of algorithms. However, we are not able to give a fair comparison *w.r.t.* runtimes. We obtained our results over a long period of time using a diverse set of hardware. Further, the code obtained from the authors is written in different languages and may or may not be optimized for multi-threading or usage of GPUs. Nevertheless, some of the authors provide runtime analyses.

### 7.1 General Overview of Results

We compare the individual results as footprints (see Section 5.3) of each method considered in this manuscript in Fig. 6. Here we organize the results *w.r.t.* the data collections (note, accuracies match those presented in the corresponding figures in Appendix D). As can be seen from the very different result shapes for each data collection, there is no clear winner method that is able to perform well in any collection. Each individual data collection has its own typical set of top scoring methods as well as a set of methods that performs poorly. It can further be seen that no method is able to solve any of the problem settings with 100% accuracy, while the actual range of achieved results range from very low percentages around the 25% mark to very high percentages around the 95% mark across all data sets. Please note, the random baseline of a binary decision is naturally at 50% while lower accuracies usually are a sign for systematic errors. Some algorithms considered here are more potent than others or fail during the evaluation and thus can cause invalid decisions (see Section 5.2). We set the individual methods into context in Appendix D. However, we can already spot methods that appear to achieve good results overall, while others seem to perform poorly in general. An additional ranking across all data collections is given in Section 7.2. Further, we highly recommend examining the individual evaluations given in Appendix D before judging methods' applicability.

As a side note, the differences in results caused by the weighting of wTCEP in comparison to TCEP become apparent here. Some methods benefit from this weighting (*e.g.,* RECI or Jarfo) while the performance of others are down-weighted (*e.g.,* PNL).

### 7.2 Average Accuracies

In addition to dissecting accuracies on individual data collections, we want to also present average accuracies of the considered methods in Fig. 7. However, please keep in mind that this averaging is hiding some individual aspects we want to unveil with this study, *i.e.,* no method is suitable for all settings. Thus, the averaged results presented have to be seen in the context of the composition of the corresponding data collections.

We treat TCEP and wTCEP as separate data sets during our evaluation. However, for averaging the results, we omit TCEP and take only wTCEP into account to avoid any bias. We provide four different averaged results: with or without our proposed bSCMC data as well as with or without weighting according to the number of single data sets in the individual data collections. The un-weighted average corresponds to the average of collection-wise results presented in Fig. 6 or in sub-figures (a) of individual footprint plots in Appendix D. Further, the weighted average corresponds to the computation of a single accuracy number from concatenated individual predictions over all data sets within each collection.

If we consider only the currently established available data collections (*i.e.,* Avg Acc Est and Wavg Acc Est), the methods CGNN and GPI appear to be close competitors and overall good performing methods. However, consider the individual analysis in Appendix D.2 and in Appendix D.4, where we examine the results of both methods. Comparing the results shows that both approaches perform well or poorly in different settings. For example, CGNN fails systematically on data that is built with a mul_a function

| Method | Avg Acc All | Wavg Acc All | Avg Acc Est | Wavg Acc Est |
|---|---|---|---|---|
| ANM | 60.78 % | 59.39 % | 61.07 % | 61.55 % |
| CGNN | 71.65 % | 76.00 % | 70.74 % | **64.44 %** |
| EMD | 66.67 % | 80.64 % | 63.80 % | 57.12 % |
| GPI | **72.06 %** | 72.26 % | **71.99 %** | 63.89 % |
| IGCI | 65.29 % | 80.61 % | 62.15 % | 55.93 % |
| Jarfo | 53.91 % | 55.22 % | 53.63 % | 49.94 % |
| LiNGAM | 45.94 % | 29.43 % | 49.31 % | 52.35 % |
| NLME | 48.94 % | 39.27 % | 50.92 % | 54.87 % |
| PNL | 64.32 % | 54.31 % | 66.35 % | 63.85 % |
| QCCD | 60.11 % | 53.32 % | 61.48 % | 56.00 % |
| RCC | 51.81 % | 34.30 % | 55.38 % | 53.13 % |
| RECI | 69.10 % | 83.47 % | 66.15 % | 58.62 % |
| SLOPE | 65.46 % | 78.80 % | 62.71 % | 52.10 % |
| SLOPPY | 67.29 % | **85.72 %** | 63.50 % | 52.14 % |

Figure 7: Overall result comparison. We show the average accuracies (*Avg Acc*) over all data collections (except wTCEP). We also show the average accuracies weighted *w.r.t.* the number of data sets per collection (*Wavg Acc*). Further, we give the accuracies for all in this manuscript considered collections (*All*) as well as for the established collections only (*Est*), *i.e.,* we omit our proposed data collection bSCMC from the accuracy computation. Highest results are marked **bold** and lowest are marked with an underline.

and a bimodal normal cause distribution, while GPI reaches very good accuracies in this setting (compare Fig. 35.e and Fig. 49.e).

On the other hand, if we include our data set collection, we see that SLOPPY is able to achieve the best result when it comes to the example-weighted average (*i.e.,* Wavg Acc All). The reason is that our data set is quite large in comparison to the remaining data collections while SLOPPY achieves good results on the bSCMC data (consider the individual analysis in Appendix D.14). If we consider the plain average over all data collections (*i.e.,* Avg Acc All), GPI is still the top performing method closely followed by CGNN. Thus we can conclude that it is not only important how the data is built, but also how the results are averaged.

In contrast, LiNGAM achieves poor results overall, which is not surprising given the underlying assumptions. The method is not designed for a broad applicability as required for this kind of evaluation presented here (*i.e.,* just averaging all settings). However, it is able to excel in scenarios it is made for (consider individual analysis in Appendix D.7).

Finally, please note that the random baseline for this setting is 50%. Thus, results around 50% are just as good as guessing while lower results are hinting towards systematic errors. However, this has to be seen in the context of invalid decisions (see Section 5).

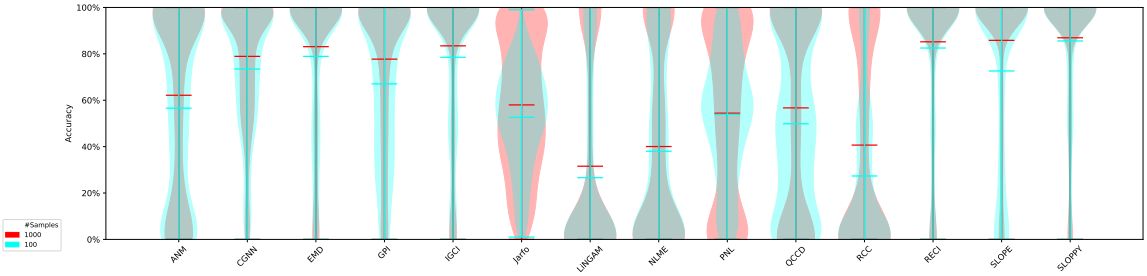



Figure 8: Distribution of accuracies *w.r.t.* individual configurations on our proposed data collection per method as violins. Please note, the width of the violins is scaled to unit width individually.

### 7.3 Overview of Results on Proposed Data Collection

One of the major goals of this study is to provide insight into the performances of the individual methods *w.r.t.* different characteristics of the data generating process. We want to give an overview from a broader perspective in addition to the individual evaluations presented in Appendix D. In the following, we examine various aspects and dive into details of our bSCMC including interactions, significances, importance of the individual building blocks our data collection is comprised of. This section is also helpful to identify setups where all methods tend to perform poorly.

#### 7.3.1 Comparative Overview of Methods

We want to investigate the actual distribution of accuracies across all configurations for each method. Thus, we employ 828 single accuracies per method computed over 100 realizations each (see Section 4), and plot these as violin plots (see Section 5.4) in Fig. 8. Please note, the sample sizes are overlayed.

Generally, we can see three different types of algorithms: Firstly, methods that perform quite good overall, *i.e.,* the vast majority of the distribution is located near the top (*e.g.,* EMD, SLOPPY or RECI). Secondly, methods that deliver a mediocre average performance with a violin that is spread out over the whole spectrum (*e.g.,* Jarfo or PNL) and a "belly" near the 50% mark. The latter being caused by the algorithm being not certain and delivering alternating true and false predictions. Thirdly, the range of methods also includes specialized approaches that work well on some kinds of data and systematically fail (potentially) per design on others. This behavior is visible in the corresponding violins when the majority of the distribution mass is located at the bottom, and only a small part which corresponds to the portion of data, which the method is suitable for, is located at the top (*i.e.,* LiNGAM or NLME). The fact that these methods reach an accuracy below the random benchmark of 50% implies that they are systematically biased. However, these results have to be seen in the context of invalid decisions (see Section 5).

Finally, this visualization also reveals that more data in general leads to better accuracy (see horizontal bars, compare to Fig. 10) even though the difference is sometimes not significant (see individual evaluations in Appendix D or Section 7.3.6).

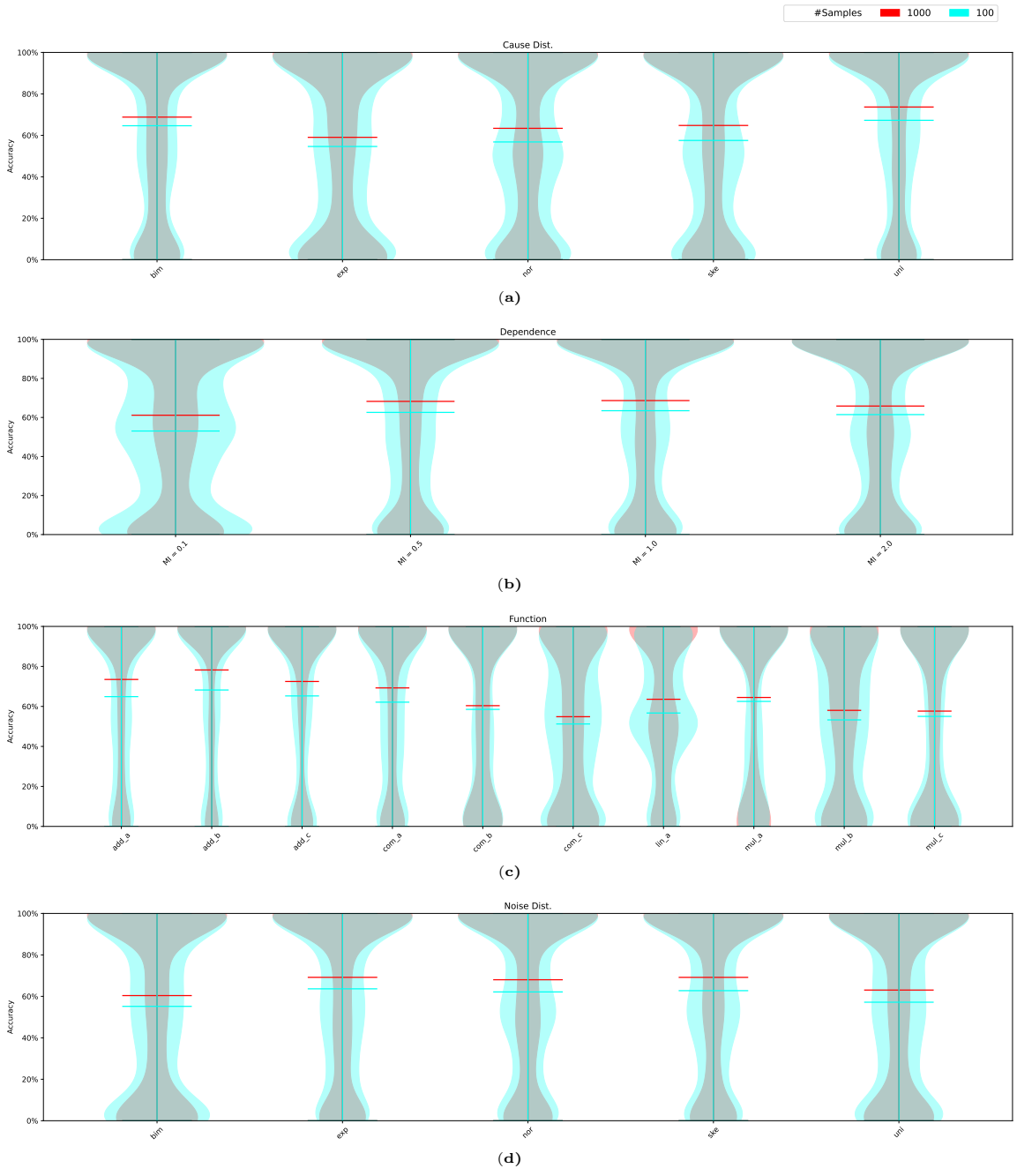



Figure 9: Distribution of accuracies *w.r.t.* individual configurations on our proposed data collection per building block as violins. Please note, the width of the violins is scaled to unit width individually.

### 7.3.2 Result Overview for Single Building Blocks

In addition to the violin plots *w.r.t.* the individual methods as presented in Section 7.3.1, we are also interested in providing insight into the impact of the remaining building blocks onto the distribution of results. Thus, we employ different splits of the results and plot the individual configurations of each remaining building block as a separate violin in Fig. 9. In more detail, we have 828 single accuracies per method and 14 methods, which results in 11 592 single accuracies in total (for 100 and 1000 samples respectively, see Section 4). Instead of separating these results into individual violins for each method, we split the results according to the remaining building blocks, *e.g.,* into the results obtained on each of the five cause distributions while we build the violin over all the remaining building blocks (*i.e.,* averaged over methods, dependence strength, functional dependence and noise distribution).

We can generally observe that the majority of mass of each violin is concentrated at the top, *i.e.,* correct decision of causal direction. Please consider, since these results are averaged over all methods, we can conclude that in the pool of applied methods there are at least some methods that can identify the causal direction mostly correct under given circumstances. Further we can see that the violins that represent 100 samples appear to be thicker overall than the violins representing results obtained with 1000 samples. This effect can be explained by the individual scaling to unit with of the violins that results in stretching the 100 samples violins to the same width as the corresponding 1000 samples counterpart (mostly to match the width near 100% accuracy). However, we can see that the relative mass of the 100 samples violins is more spread out while the mean indication is always lower than the mean indication for the 1000 samples counterpart. We can thus once again conclude that more samples help in general to achieve better results (compare individual evaluations in Appendix D or Section 7.3.6).

In case of the cause distributions (see Fig. 9.a), we can observe that bimodal normal and uniform causes appear to comprise the most reliably identifiable scenarios, *i.e.,* the violins are relatively thin near the 0% accuracy mark. The scenarios involving exponential causes appear to be the most challenging ones, *i.e.,* the violins are relatively thick near the 0% accuracy mark. Interestingly, the 100 samples violins for normal and skewed normal causes appear to have a "belly" near the 50% mark, which is due to alternating true and false predictions.

As for dependence strength (see Fig. 9.b), we can observe that very low dependence, *i.e.,* MI $= 0.1$, appears to be the most challenging scenario. Further, we can also see an obvious "belly" near the 50% mark for 100 samples, which is again a sign for random guessing, *i.e.,* alternating true and false predictions. Also high dependence strength such as MI $= 0.1$ results in more relative mass of the violin to be concentrated near the 0% accuracy mark, *i.e.,* more frequent false predictions. The best results could be obtained for dependence strength values in between those to corner cases (compare individual results in Appendix D).

In case of functional dependence (see Fig. 9.c), we can see that additive functions appear to comprise the easiest scenarios while the violins for all other functional dependencies suggest that the methods do not deliver reliable results. The most challenging scenarios appear to include the com_b and mul_b function. Interestingly, the violin for the

lin_a function shows a clear "belly" near the 50% mark for both sample sizes that might be caused by unidentifiable cases (see, *e.g.,* Shimizu et al., 2006).

Finally, the violins for the noise distributions (see Fig. 9.d) show a opposite picture than those for the cause distributions, *i.e.,* bimodal normal and uniform noise appear to be the most challenging scenarios, while the violins for exponential, normal and skewed normal noise are thinnest near the 0% accuracy mark.

### 7.3.3 Interaction Effects of Characteristics

Here we are interested in the interaction of the single building blocks among each other. To this end, we show the obtained accuracies for different combinations of configurations in Fig. 10.

Generally, we observe in Fig. 10.a,c,e,g that more samples lead to higher average accuracies while a low dependency strength results in lower accuracies (see Fig. 10.b,d,f,g). Nevertheless, high dependency does not necessarily result in high accuracy nor does low dependency. We can observe that the majority of methods are more suited for values around MI = 0.5 and MI = 1.0. It can also be observed that the methods, which perform well on the data in general (compare Fig. 6.e), also achieve a high accuracy overall considering the interactions shown here (*e.g.,* SLOPPY or EMD). However, these methods also fail (*e.g.,* see lin_a function in Fig. 10.f) while other, methods excel (*e.g.,* LiNGAM or NLME, note the assumptions given in Appendices D.7 and D.8).

Further, we can spot some configurations that lead to poor accuracies for all methods (*e.g.,* uniform noise + MI = 0.1 in Fig. 10.d or mul_b function in Fig. 10.e,f). Accuracies around 50% are as good as guessing in this setting. Hence, piled results around 50% are a sign that those algorithms might guess the causal direction (*i.e.,* alternating true/false predictions). In contrast, results near 0% are either a sign for systematic errors, *i.e.,* predicting always the wrong direction (note, guessing always wrong in a binary setting is as hard as guessing always right), or for many invalid decisions (see Section 5).

### 7.3.4 Individual Configurations with Lowest Maximal Accuracy

One goal of our investigation is to unveil combinations of data set characteristics that none of the considered methods is able to handle properly. To do so, we investigate the obtained accuracies directly. Since we are interested in the particular configurations no method is able to predict entirely correctly, we first designate the best performing method for each individual setting (*i.e.,* we obtain the best method and the maximal accuracy for each particular configuration). We plot the 20 individual configurations with the lowest obtained maximal accuracy in Fig. 11. In more detail, if we consider the last row of Fig. 11, we can see that PNL achieved 53.0 % accuracy on data sampled with the configuration specified by the entries of the first five columns. This means that the remaining 13 methods considered in this manuscript achieved accuracies below 53.0 % in the same settings.

From an overall perspective, we can see a lot of normal cause distributions, small samples sizes with 100 samples, lin_a functional dependencies and normal noise distributions. We can also see that almost every dependence strength among these worst configurations belongs to the corner cases, *i.e.,* MI = 0.1 and MI = 2.0. This can be interpreted as a expected results. Linear dependence with normal distributed cause / noise

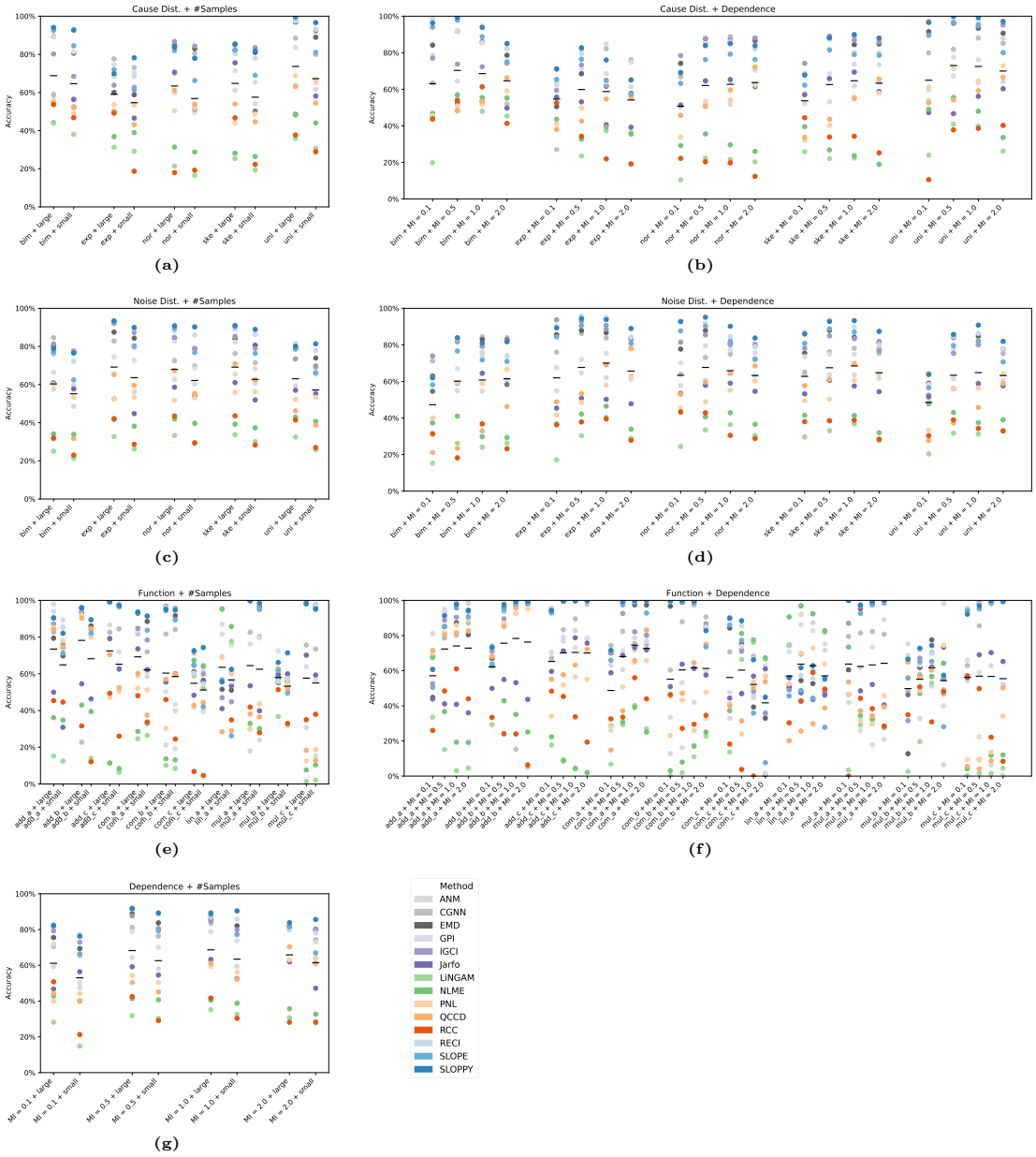



Figure 10: Accuracies of building blocks for individual methods *w.r.t.* interaction of different building blocks of our proposed data collection. Average accuracies for each configuration are marked with a black dash.

belongs to the unidentifiable cases described by (see, *e.g.,* Shimizu et al., 2006). Further, small sample sizes might not be enough to represent the underlying SCM well and thus lead to false predictions. Finally, the corner cases of the used dependence strengths are either too "noisy" or too "deterministic" for the considered methods to work well. However, we also see some configurations that do not fit into this scheme, which might be worth investigating

| Cause Dist. | #Samples | Dependence | Function | Noise Dist. | Max. Acc. | Best Methods |
|---|---|---|---|---|---|---|
| normal | 100 | MI = 2.0 | lin_a | uniform | 67.0 % | GPI, NLME |
| skewed normal | 100 | MI = 0.1 | mul_c | uniform | 67.0 % | Jarfo |
| normal | 100 | MI = 2.0 | lin_a | skewed normal | 67.0 % | NLME |
| normal | 1000 | MI = 1.0 | lin_a | normal | 67.0 % | RCC |
| exponential | 100 | MI = 0.5 | mul_b | skewed normal | 67.0 % | IGCI |
| skewed normal | 100 | MI = 0.1 | com_c | skewed normal | 66.0 % | GPI |
| normal | 100 | MI = 1.0 | com_c | bimodal normal | 65.0 % | QCCD |
| skewed normal | 100 | MI = 2.0 | lin_a | skewed normal | 65.0 % | LiNGAM |
| normal | 100 | MI = 0.1 | add_a | uniform | 63.0 % | QCCD |
| normal | 100 | MI = 0.1 | com_a | uniform | 63.0 % | Jarfo |
| normal | 100 | MI = 0.1 | add_a | bimodal normal | 62.0 % | QCCD |
| normal | 100 | MI = 2.0 | lin_a | normal | 61.0 % | EMD |
| skewed normal | 100 | MI = 2.0 | lin_a | normal | 61.0 % | LiNGAM |
| normal | 100 | MI = 1.0 | lin_a | normal | 60.0 % | RCC |
| normal | 1000 | MI = 0.1 | lin_a | normal | 60.0 % | GPI |
| normal | 100 | MI = 2.0 | com_c | bimodal normal | 60.0 % | Jarfo |
| normal | 100 | MI = 0.5 | lin_a | normal | 59.0 % | EMD |
| normal | 1000 | MI = 2.0 | lin_a | normal | 56.0 % | LiNGAM, SLOPPY |
| normal | 100 | MI = 0.1 | lin_a | normal | 55.0 % | RECI |
| normal | 100 | MI = 0.1 | add_b | uniform | 53.0 % | PNL |

Figure 11: The 20 individual data set characteristics with lowest maximal accuracy among the 14 considered methods.

further. Generally, we see accuracies near the 50 % mark, which is a sign for alternating true / false predictions and thus random guessing (consider the individual random baselines, see Section 5.2 and Appendix D). However, we can see methods that do not perform well overall among the best performing methods in this investigation. For example, LiNGAM and NLME deliver best results on data involving lin_a functions, which is the setting those methods are meant for (see Appendices D.7 and D.8). We also see that Jarfo is able to outperform all other methods in some of these particular settings. Given the overall result comparison shown in Section 7.2 where Jarfo is among the lowest performing methods, this shows again that there is no one method that is suitable for all configurations and that, in turn, each might have its niche.

### 7.3.5 Importance of Building Blocks using Shapley Values

We are interested in investigating the importance of each individual building block towards the resulting performance. To do so, we compute mean Shapley values as described in Section 5.5. Here, we are interested in giving a more comprehensive overview (see Fig. 12), while we will present method specific results in Appendix D as well as more in depth analysis in Appendix B.

As we can see from Fig. 12, the most crucial part for the resulting accuracy is the method (*i.e.,* the method-feature does have the largest influence on the auxiliary regressor, compare, *e.g.,* Section 7.3.1). This implies that the most important aspect to get reliable predictions for causal directions is to choose the right method, *i.e.,* something we can actually do given a data sample. Following-up, but with a large margin to the method, there are the

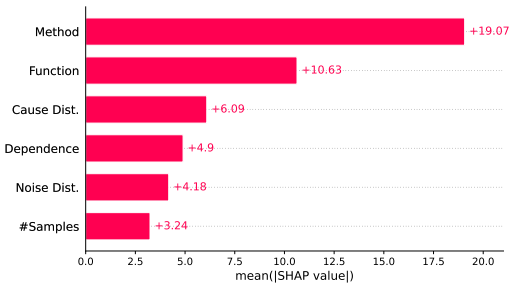



Figure 12: Importance of individual data set characteristics as mean Shapley values.

| Setting | p-Value |
|---|---|
| Method ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function, Noise Dist.] | 0.0e+00 |
| Cause Dist. ⫫ Accuracy \| [Method, #Samples, Dependence, Function, Noise Dist.] | 1.6e-91 |
| #Samples ⫫ Accuracy \| [Method, Cause Dist., Dependence, Function, Noise Dist.] | 1.7e-38 |
| Dependence ⫫ Accuracy \| [Method, Cause Dist., #Samples, Function, Noise Dist.] | 1.1e-47 |
| Function ⫫ Accuracy \| [Method, Cause Dist., #Samples, Dependence, Noise Dist.] | 6.2e-175 |
| Noise Dist. ⫫ Accuracy \| [Method, Cause Dist., #Samples, Dependence, Function] | 2.2e-49 |

Figure 13: Significances at $\alpha = 0.05$-level for conditional independencies of individual data set characteristics *w.r.t.* the obtained accuracy. Green color coding indicates dependence while red color coding indicates independence.

data inherent aspects such as distributions or dependence strength, while the functional dependence is the second most important factor. These building blocks can usually not be adjusted in real test scenarios, *i.e.,* their ground-truth is unknown and they are fixed for a given data sample. However, these aspects do have a crucial impact on the applicability of different methods (see Appendix D). Interestingly, the number of available samples is the least influential factor even though more samples appear to lead to better results in general (compare, *e.g.,* Section 7.3.2).

### 7.3.6 Conditional Independencies among Building Blocks

In previous sections we were mostly interested in a comparative analysis of the impact of the single building blocks from an accuracy-perspective. Here, we want to investigate whether the individual building blocks have an influence onto the methods performance in a more direct manner, *i.e.,* we measure the conditional independence of individual building blocks *w.r.t.* the obtained accuracies as described in Section 5.7.

As we can observe from the results given in Fig. 13, the individual building blocks, conditioned on the corresponding remaining building blocks, are not independent *i.e.,* each building block is needed to explain the resulting accuracies, even if we have information from all other building blocks available. Thus, our proposed data collection does not comprise any redundant configuration from a general perspective. However, this might be different for individual methods (see Appendix D). We extend this investigation in Appendix C and examine the independencies for individual methods from a broader perspective, while we dive into the details for each method in Appendix D.

## 8. Discussion

We now conclude our investigations with a discussion. We will consider the method perspective in Section 8.1 and the data collections in Section 8.2. A final conclusion is given in Section 8.3. As mentioned, we postponed a part of our evaluation to the appendix to maintain a certain level of compactness (see Section 1). Nevertheless, these parts are important for the bigger picture of this study. We will therefore also refer to parts of the appendix in the following.

### 8.1 Method results

There is a vast range of methods available in the literature today and we were only able to cover a sample of these in this manuscript due to various reasons. A list of missing methods together with some explanations is given in Appendix E. As mentioned, we plan to extend the plethora of considered methods on `causeme.net`.

*Applicability:* In our evaluation we have seen diverse results on our bSCMC data (see Section 7 and Appendix D). We observed methods that work on a range of data conditions but fail on others. But, we also saw methods that fail on the majority of cases and ace in some special scenarios they are made for. Generally, there is no method that is able to predict the true causal direction regardless of the data characteristics. Each method can roughly be assigned into one of the three types discussed in Section 2.2 and, thus, comes with its own underlying assumptions. These individual assumptions are partially causing the diverse outcomes we observed in our evaluation. Even though the results appear inconclusive at first glance, we see that the data characteristics and the method inherent strengths and flaws mostly match.

However, we also saw that even if a method is designed to deal with a certain data characteristic (*e.g.,* a certain functional dependence), compound effects and interactions of the building blocks contained in our data reveal that the range of applicability of individual methods is not that obvious in advance.

Further, the majority of cases we analyzed can be treated with at least some methods that were considered in this manuscript. However, we also discovered some blind spots where no method could deliver acceptable results. Even if we aim to deliver a wide range of individual configurations within bSCMC, the search space is far from being exhausted.

As a consequence, from a user perspective we still need to be very careful which method we apply and trust without evaluating it. We saw in the Shapley analysis that the method is the most influential factor when it comes to prediction accuracy, *i.e.,* getting the right method is most important among all building blocks. The method is also the only ingredient a user can choose given a set of data in a real-world test scenario. However, the right choice of method is heavily dependent on the usually unknown data set characteristics, which makes this a rather hard problem to solve without further knowledge. If only parts of the underlying data characteristics are known, the range of methods that may be suitable for a certain problem is still large.

*Usability:* We provide insight into the out-of-the-box performance of the considered methods. Thus, we apply the code without further tuning or adjustment of parameters. However, we believe that adapting the methods to the actual data would, in general, improve the results. Although hyper-parameter optimization is implemented in the code in some cases, the actual adaptation is strongly connected to the underlying data set characteristics, which are usually unknown. Hence, a fine-tuning of methods is far away from being a trivial task.

Further factors to consider are the methods runtime (see also Section 7) and for some methods the requirement of sufficient amounts of labeled training data that ideally has to match the test data.

Finally, some methods are more potent than others in that they also predict dependence or independence and not only causal directions. These more potent methods are in a way more "honest" and, if not forced to decide for a direction as in our analysis, might just output a dependence rather than the causal direction. This may be interpreted as "don't know". Thus, users have to be aware of potential output of the applied methods and have to interpret them properly.

### 8.2 Data Collections and Benchmarks

Current available data collections have some deficiencies that make it hard to compare methods (see Section 3.2). In short, these are (1) limited and biased synthetic data in method papers, (2) limited data collections size, (3) unknown biases in real data collections, and (4) outdatedness. Especially in real data there is also uncertainty about the underlying (5) causal ground truth, and the (6) data generating process characteristics. We made a step towards tackling these aspects and proposed the first iteration of the bSCMC data collection. Already this first attempt pointed out some systematic problems of methods (see Section 7), but, nevertheless, our choice of available configurations for each building block may not represent domain-specific setups. Our goal was to evaluate how methods perform on particular data characteristics based on an underlying SCM. However, as already pointed out, *e.g.,* by Mooij et al. (2016), generating realistic looking data is a non-trivial task and our choices might not reflect the data characteristics of domain-specific real data.

### 8.3 Conclusion and Future Improvements

*Summary:* Our goal was to provide a "user-friendly" comparison study of the current state-of-the-art for bivariate causal discovery methods. We studied recent benchmarks and introduced a novel data collection to systematically investigate the effect of the individual characteristics of underlying SCMs. We used the methods "as is" since this is how they would be applied by users. All data and results will be published on `causeme.net`.

Our main findings are that no method exists that outperforms all others for any data characteristic, but some methods have clear advantages over others for specific settings. We also identified data configurations where no method delivered satisfactory accuracies and found settings that result in systematic biases of methods. Further, we found that the choice of method is the most influential factor when it comes to prediction accuracy in general according to a Shapley analysis. We hope that our study will help users to find which method works best for their data, and method developers to overcome current deficiencies.

*Outlook:* Our proposed data collection only covers a selected portion of the search space of configurations of an SCM. We aim to provide more configurations including domain-specific data on `causeme.net`. Ideally, such domain-specific collections should be based on theoretical or empirical knowledge about the building blocks of underlying SCMs in specific domains. However, given that often problems are of a multivariate nature, such "induced" SCMs might be hard to obtain. We further plan to advance on the method side. Given that no single method is able to deliver satisfactory results for all settings as of now, one can consider meta and ensemble methods that adapt themselves to the given data properties. First attempts into this direction were already made, *e.g.,* by Fonollosa (2016) or Ton et al. (2020).

## Acknowledgments

This work used resources of the Deutsches Klimarechenzentrum (DKRZ) granted by its Scientific Steering Committee (WLA) under project ID 1083.

Jakob Runge has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (Grant CausalEarth with agreement No. 948112).

## References

Hirotugu Akaike. Information measures and model selection. *Int Stat Inst*, 1983.

Bryon Aragam and Qing Zhou. Concave penalized estimation of sparse gaussian Bayesian networks. *Journal of Machine Learning Research*, 2015.

Loïc Barrault, Ondřej Bojar, Marta R Costa-Jussa, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, et al. Findings of the 2019 conference on machine translation (wmt19). In *Conference on Machine Translation*, 2019.

Alex Berg, Jia Deng, and Li Fei-Fei. Large scale visual recognition challenge 2010, 2010. URL `image-net.org/challenges`.

Helen M Berman, John Westbrook, Zukang Feng, Gary Gilliland, Talapady N Bhat, Helge Weissig, Ilya N Shindyalov, and Philip E Bourne. The protein data bank. *Nucleic acids research*, 2000.

Patrick Blöbaum, Dominik Janzing, Takashi Washio, Shohei Shimizu, and Bernhard Schölkopf. Cause-effect inference by comparing regression errors. In *International Conference on Artificial Intelligence and Statistics*, 2018.

Gianluca Bontempi and Maxime Flauder. From dependency to causality: a machine learning approach. *The Journal of Machine Learning Research*, 2015.

Kailash Budhathoki and Jilles Vreeken. Causal inference by stochastic complexity. *arXiv preprint arXiv:1702.06776*, 2017.

Peter Bühlmann, Jonas Peters, Jan Ernest, et al. Cam: Causal additive models, high-dimensional order search and penalized regression. *The Annals of Statistics*, 2014.

Ruichu Cai, Jie Qiao, Kun Zhang, Zhenjie Zhang, and Zhifeng Hao. Causal discovery with cascade nonlinear additive noise models. *arXiv preprint arXiv:1905.09442*, 2019.

Gregory J Chaitin. Algorithmic information theory. *IBM journal of research and development*, 1977.

Zhitang Chen, Kun Zhang, Laiwan Chan, and Bernhard Schölkopf. Causal discovery via reproducing kernel hilbert space embeddings. *Neural computation*, 2014.

Zhitang Chen, Shengyu Zhu, Yue Liu, and Tim Tse. Causal discovery by kernel intrinsic invariance measure. *arXiv preprint arXiv:1909.00513*, 2019.

David Maxwell Chickering. Learning equivalence classes of Bayesian-network structures. *Journal of Machine Learning Research*, 2002a.

David Maxwell Chickering. Optimal structure identification with greedy search. *Journal of machine learning research*, 2002b.

Pierre Comon. Independent component analysis, a new concept? *Signal processing*, 1994.

Povilas Daniusis, Dominik Janzing, Joris Mooij, Jakob Zscheischler, Bastian Steudel, Kun Zhang, and Bernhard Schölkopf. Inferring deterministic causal relations. *arXiv preprint arXiv:1203.3475*, 2012.

A Philip Dawid. Conditional independence in statistical theory. *Journal of the Royal Statistical Society: Series B (Methodological)*, 1979.

A Philip Dawid. Statistical causality from a decision-theoretic perspective. *Annual Review of Statistics and Its Application*, 2015.

Dua Dheeru and Efi Karra Taniskidou. UCI machine learning repository, 2017. URL `archive.ics.uci.edu/ml`.

Michael P Fay and Michael A Proschan. Wilcoxon-mann-whitney or t-test? on assumptions for hypothesis tests and multiple interpretations of decision rules. *Statistics surveys*, 2010.

José AR Fonollosa. Conditional distribution variability measures for causality detection. *arXiv preprint arXiv:1601.06680*, 2016.

Nir Friedman and Iftach Nachman. Gaussian process networks. *arXiv preprint arXiv:1301.3857*, 2013.

Tomer Galanti, Ofir Nabati, and Lior Wolf. A critical view of the structural causal model. *arXiv preprint arXiv:2002.10007*, 2020.

Tian Gao, Debarun Bhattacharjya, Elliot Nelson, Miao Liu, and Yue Yu. IDYNO: Learning nonparametric DAGs from interventional dynamic data. In *International Conference on Machine Learning*, 2022.

Zoubin Ghahramani. Probabilistic machine learning and artificial intelligence. *Nature*, 2015.

Nicola Gnecco, Nicolai Meinshausen, Jonas Peters, and Sebastian Engelke. Causal discovery in heavy-tailed models. *arXiv preprint arXiv:1908.05097*, 2019.

Daniel Goldfarb and Scott Evans. Causal inference via conditional kolmogorov complexity using mdl binning. *arXiv preprint arXiv:1911.00332*, 2019.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

Olivier Goudet, Diviyan Kalainathan, Philippe Caillou, Isabelle Guyon, David Lopez-Paz, and Michele Sebag. Learning functional causal models with generative neural networks. In *Explainable and Interpretable Models in Computer Vision and Machine Learning*, 2018.

Olivier Goudet, Diviyan Kalainathan, Michèle Sebag, and Isabelle Guyon. Learning bivariate functional causal models. In *Cause Effect Pairs in Machine Learning*, 2019.

Arthur Gretton, Kenji Fukumizu, Choon H Teo, Le Song, Bernhard Schölkopf, and Alex J Smola. A kernel statistical test of independence. In *Advances in Neural Information Processing Systems*, 2008.

Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on tabular data? *arXiv preprint arXiv:2207.08815*, 2022.

Isabelle Guyon. Chalearn cause effect pairs challenge. *http://www.causality.inf.ethz.ch/cause-effect.php*, 2013.

Isabelle Guyon, Alexander Statnikov, and Berna Bakir Batu. *Cause Effect Pairs in Machine Learning*. Springer, 2019.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning; Data Mining, Inference and Prediction*. Springer, 2008.

Daniel Hernandez-Lobato, Pablo Morales-Mombiela, David Lopez-Paz, and Alberto Suarez. Non-linear causal inference using gaussianity measures. *The Journal of Machine Learning Research*, 2016.

Patrik O Hoyer, Dominik Janzing, Joris M Mooij, Jonas Peters, and Bernhard Schölkopf. Nonlinear causal discovery with additive noise models. In *Advances in Neural Information Processing Systems*, 2009.

Aapo Hyvärinen. Pairwise measures of causal direction in linear non-gaussian acyclic models. In *Asian Conference on Machine Learning*, 2010.

Aapo Hyvärinen and Stephen M Smith. Pairwise likelihood ratios for estimation of non-gaussian structural equation models. *Journal of Machine Learning Research*, 2013.

Guido W Imbens and Donald B Rubin. *Causal inference in statistics, social, and biomedical sciences*. Cambridge University Press, 2015.

Dominik Janzing. The cause-effect problem: Motivation, ideas, and popular misconceptions. In *Cause Effect Pairs in Machine Learning*, 2019.

Dominik Janzing and Bernhard Schölkopf. Causal inference using the algorithmic markov condition. *Transactions on Information Theory*, 2010.

Dominik Janzing, Patrik O Hoyer, and Bernhard Schölkopf. Telling cause from effect based on high-dimensional observations. *arXiv preprint arXiv:0909.4386*, 2009a.

Dominik Janzing, Xiaohai Sun, and Bernhard Schölkopf. Distinguishing cause and effect via second order exponential models. *arXiv preprint arXiv:0910.5561*, 2009b.

Dominik Janzing, Joris Mooij, Kun Zhang, Jan Lemeire, Jakob Zscheischler, Povilas Daniušis, Bastian Steudel, and Bernhard Schölkopf. Information-geometric approach to inferring causal directions. *Artificial Intelligence*, 2012a.

Dominik Janzing, Jonas Peters, Joris Mooij, and Bernhard Schölkopf. Identifying confounders using additive noise models. *arXiv preprint arXiv:1205.2640*, 2012b.

Dominik Janzing, Eleni Sgouritsa, Oliver Stegle, Jonas Peters, and Bernhard Schölkopf. Detecting low-complexity unobserved causes. *arXiv preprint arXiv:1202.3737*, 2012c.

Martin Jørgensen and Søren Hauberg. Reparametrization invariance in non-parametric causal discovery. *arXiv preprint arXiv:2008.05552*, 2020.

Diviyan Kalainathan and Olivier Goudet. Causal discovery toolbox: Uncover causal relationships in python. *arXiv preprint arXiv:1903.02278*, 2019.

Diviyan Kalainathan, Olivier Goudet, Isabelle Guyon, David Lopez-Paz, and Michèle Sebag. Structural agnostic modeling: Adversarial learning of causal graphs. *arXiv preprint arXiv:1803.04929*, 2018.

Yutaka Kano, Shohei Shimizu, et al. Causal inference using nonnormality. In *International symposium on science of modeling, the 30th anniversary of the information criterion*, 2003.

Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. Estimating mutual information. *Physical review E*, 2004.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2011.

Jan Lemeire and Dominik Janzing. Replacing causal faithfulness with algorithmic independence of conditionals. *Minds and Machines*, 2013.

Hebi Li, Qi Xiao, and Jin Tian. Supervised whole dag causal discovery. *arXiv preprint arXiv:2006.04697*, 2020.

David Lopez-Paz and Maxime Oquab. Revisiting classifier two-sample tests. *arXiv preprint arXiv:1610.06545*, 2016.

David Lopez-Paz, Krikamol Muandet, Bernhard Schölkopf, and Iliya Tolstikhin. Towards a learning theory of cause-effect inference. In *International Conference on Machine Learning*, 2015.

David Lopez-Paz, Robert Nishihara, Soumith Chintala, Bernhard Schölkopf, and Léon Bottou. Discovering causal signals in images. In *Conference on Computer Vision and Pattern Recognition*, 2017.

Alexander Marx and Jilles Vreeken. Telling cause from effect using mdl-based local and global regression. In *International conference on data mining*, 2017.

Alexander Marx and Jilles Vreeken. Identifiability of cause and effect using regularized regression. In *International Conference on Knowledge Discovery & Data Mining*, 2019.

Jovana Mitrovic, Dino Sejdinovic, and Yee Whye Teh. Causal inference via kernel deviance measures. In *Advances in Neural Information Processing Systems*, 2018.

Ricardo Pio Monti, Kun Zhang, and Aapo Hyvarinen. Causal discovery with general non-linear relationships using non-linear ica. *arXiv preprint arXiv:1904.09096*, 2019.

Ricardo Pio Monti, Ilyes Khemakhem, and Aapo Hyvarinen. Autoregressive flow-based causal discovery and inference. *arXiv preprint arXiv:2007.09390*, 2020.

Joris Mooij and Dominik Janzing. Distinguishing between cause and effect. In *Causality: Objectives and Assessment*, pages 147–156. PMLR, 2010.

Joris Mooij, Dominik Janzing, Jonas Peters, and Bernhard Schölkopf. Regression by dependence minimization and its application to causal inference in additive noise models. In *International conference on machine learning*, 2009.

Joris M Mooij, Dominik Janzing, Tom Heskes, and Bernhard Schölkopf. On causal discovery with cyclic additive noise model. In *International symposium on science of modeling, the 30th anniversary of the information criterion*, 2003.

Joris M Mooij, Oliver Stegle, Dominik Janzing, Kun Zhang, and Bernhard Schölkopf. Probabilistic latent variable models for distinguishing between cause and effect. In *Advances in Neural Information Processing Systems*, 2010.

Joris M Mooij, Jonas Peters, Dominik Janzing, Jakob Zscheischler, and Bernhard Schölkopf. Distinguishing cause from effect using observational data: methods and benchmarks. *The Journal of Machine Learning Research*, 2016.

Christopher Nowzohour and Peter Bühlmann. Score-based causal learning in additive noise models. *Statistics*, 2016.

Judea Pearl. Causal diagrams for empirical research. *Biometrika*, 1995.

Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2000a.

Judea Pearl. Models, reasoning and inference. *Cambridge, UK: CambridgeUniversityPress*, 2000b.

Jonas Peters and Peter Bühlmann. Identifiability of gaussian structural equation models with equal error variances. *Biometrika*, 2014.

Jonas Peters, Joris M Mooij, Dominik Janzing, and Bernhard Schölkopf. Causal discovery with continuous additive noise models. *The Journal of Machine Learning Research*, 2014.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.

J Quinn, Joris M Mooij, Tom Heskes, and Michael Biehl. Learning of causal relations. In *European Symposium on Artificial Neural Networks*, 2011.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, 2008.

Ali Rahimi and Benjamin Recht. Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. In *Advances in Neural Information Processing Systems*, 2009.

Carl Edward Rasmussen and Christopher KI Williams. *Gaussian process for machine learning*. MIT press, 2006.

Hans Reichenbach. *The direction of time*. Univ of California Press, 1991.

Jorma Rissanen. Modeling by shortest data description. *Automatica*, 1978.

Jakob Runge, Sebastian Bathiany, Erik Bollt, Gustau Camps-Valls, Dim Coumou, Ethan Deyle, Clark Glymour, Marlene Kretschmer, Miguel D Mahecha, Jordi Muñoz-Marí, et al. Inferring causation from time series in earth system sciences. *Nature communications*, 2019.

Jakob Runge, Andreas Gerhardus, Gherardo Varando, Veronika Eyring, and Gustau Camps-Valls. Causal inference for time series. *Nature Reviews Earth & Environment*, 2023.

Bernhard Schölkopf. Causality for machine learning. *arXiv preprint arXiv:1911.10500*, 2019.

Bernhard Schölkopf and Alex Smola. *Learning with Kernels*. MIT Press, Cambridge, MA, USA, 2008.

Bernhard Schölkopf and Alexander J Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

Gideon Schwarz et al. Estimating the dimension of a model. *The annals of statistics*, 1978.

Eleni Sgouritsa, Dominik Janzing, Philipp Hennig, and Bernhard Schölkopf. Inference of cause and effect with unsupervised inverse regression. In *Artificial intelligence and statistics*, 2015.

Shreya Shankar, Yoni Halpern, Eric Breck, James Atwood, Jimbo Wilson, and D Sculley. No classification without representation: Assessing geodiversity issues in open data sets for the developing world. *arXiv preprint arXiv:1711.08536*, 2017.

Lloyd S Shapley. Notes on the n-person game—ii: The value of an n-person game. *Lloyd S Shapley*, 1951.

Shohei Shimizu and Kenneth Bollen. Bayesian estimation of causal direction in acyclic structural equation models with individual-specific confounder variables and non-gaussian distributions. *J. Mach. Learn. Res.*, 2014.

Shohei Shimizu, Patrik O Hoyer, Aapo Hyvärinen, and Antti Kerminen. A linear non-gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 2006.

Shohei Shimizu, Takanori Inazumi, Yasuhiro Sogawa, Aapo Hyvärinen, Yoshinobu Kawahara, Takashi Washio, Patrik O Hoyer, and Kenneth Bollen. Directlingam: A direct method for learning a linear non-gaussian structural equation model. *Journal of Machine Learning Research*, 2011.

Adrian FM Smith and David J Spiegelhalter. Bayes factors and choice criteria for linear models. *Journal of the Royal Statistical Society: Series B (Methodological)*, 1980.

Stephen M Smith, Karla L Miller, Gholamreza Salimi-Khorshidi, Matthew Webster, Christian F Beckmann, Thomas E Nichols, Joseph D Ramsey, and Mark W Woolrich. Network modelling methods for fmri. *Neuroimage*, 2011.

Pater Spirtes, Clark Glymour, Richard Scheines, Stuart Kauffman, Valerio Aimale, and Frank Wimberly. Constructing Bayesian network models of gene expression networks from microarray data. *Atlantic Symposium on Computational Biology (North Carolina)*, 2000a.

Peter Spirtes, Clark N Glymour, Richard Scheines, and David Heckerman. *Causation, prediction, and search*. MIT press, 2000b.

Alexander Statnikov, Mikael Henaff, Nikita I Lytkin, and Constantin F Aliferis. New methods for separating causes from effects in genomics data. *BMC genomics*, 2012.

Xiaohai Sun, Dominik Janzing, and Bernhard Schölkopf. Causal inference by choosing graphs with most plausible markov kernels. In *International Symposium on Artificial Intelligence and Mathematics*, 2006.

Natasa Tagasovska, Thibault Vatter, and Valérie Chavez-Demoulin. Nonparametric quantile-based causal discovery. *arXiv preprint arXiv:1801.10579*, 2018.

Natasa Tagasovska, Valérie Chavez-Demoulin, and Thibault Vatter. Distinguishing cause from effect using quantiles: Bivariate quantile causal discovery. In *International Conference on Machine Learning*, 2020.

Jean-Francois Ton, Dino Sejdinovic, and Kenji Fukumizu. Meta learning for causal direction. *arXiv preprint arXiv:2007.02809*, 2020.

Matthew J Vowels, Necati Cihan Camgoz, and Richard Bowden. D'ya like dags? a survey on structure learning and causal discovery. *arXiv preprint arXiv:2103.02582*, 2021.

Kun Zhang and Aapo Hyvärinen. Distinguishing causes from effects using nonlinear acyclic causal models. In *International Conference on Causality*, 2008.

Kun Zhang and Aapo Hyvärinen. On the identifiability of the post-nonlinear causal model. In *Uncertainty in Artificial Intelligence*, 2009.

Xun Zheng, Bryon Aragam, Pradeep K Ravikumar, and Eric P Xing. Dags with no tears: Continuous optimization for structure learning. In *Advances in Neural Information Processing Systems*, 2018.

Shengyu Zhu and Zhitang Chen. Causal discovery with reinforcement learning. *arXiv preprint arXiv:1906.04477*, 2019.

## Appendix A. Analysis of Difference in Results for Individual Configurations Across Methods

We want to analyze the overall difference in results of individual configurations of all building blocks in addition to the investigation *w.r.t.* individual configurations for each method in Appendix D. Thus, we evaluate the statistical significance of pairwise difference in results as described in Section 5.6 for all obtained results at once (*i.e.,* we aggregate results across methods). Obtained results are presented in Fig. 14. For a better intuition, refer to the distributions shown as violins in Figs. 8 and 9.

Generally, we can see that most of the pairwise difference are significant, *i.e.,* different configurations lead to different results. We can therefore conclude that the majority of individual configurations (including methods) does have a significant impact on the results and are thus crucial to the obtained causal direction. However, we can also see that few configurations are not significantly different. For example, we see that the normal and the skewed normal cause distribution do not yield significantly different results (see Fig. 14.a, compare Fig. 9.a). This does mean that across all methods there is no difference in results, but does not imply that each individual method is not prone to the differences of normal and skewed normal cause distributions. More interestingly, we also see that the results of individual methods are not significantly different (see Fig. 14.g). Some of these connections can be explained by very similar results *w.r.t.* individual configurations (*e.g.,* RECI and SLOPPY, see Appendices D.12 and D.14). However, other connections become apparent through the aggregation across all building blocks when evaluating the difference in results for individual methods (see Section 5.6). Due to the design of this analysis, we evaluate if the results of two different methods might belong to the same underlying distribution and neglect the connection of individual configuration and obtained result. For example if method A achieves good results in scenario I and bad results for scenario II, a method B with exact opposite behavior might deliver results that are not significantly different from method A *w.r.t.* this aggregation since the distribution of results does not change at all. However, the non-significance for CGNN and GPI casts a different light on the average results presented in Section 7.2 where GPI and CGNN are competing for the top-performing method in three of four categories. Please note, however, that the significances obtained here are computed on our proposed data collection alone, while the evaluation in Section 7.2 refers to all used data collections (including the established ones).

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 5.6e-35 | | | |
| (3) normal | 5.3e-19 | 6.1e-05 | | |
| (4) skewednormal | 1.7e-12 | 1.2e-07 | 1.2e-01 | |
| (5) uniform | 3.4e-06 | 1.8e-62 | 3.6e-42 | 6.5e-31 |

**(a)** cause distribution

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 2.9e-21 | | | |
| (3) normal | 4.5e-17 | 2.4e-01 | | |
| (4) skewednormal | 1.5e-20 | 8.3e-01 | 3.3e-01 | |
| (5) uniform | 5.8e-03 | 4.6e-12 | 7.0e-09 | 1.8e-11 |

**(b)** noise distribution

| | (1) | (2) | (3) |
|---|---|---|---|
| (1) MI = 0.1 | | | |
| (2) MI = 0.5 | 1.6e-35 | | |
| (3) MI = 1.0 | 1.5e-49 | 1.2e-03 | |
| (4) MI = 2.0 | 5.3e-30 | 9.7e-01 | 3.5e-03 |

**(c)** mutual information

| | (1) |
|---|---|
| (1) 1000 | |
| (2) 100 | 7.9e-75 |

**(d)** sample size

| | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) |
|---|---|---|---|---|---|---|---|---|---|
| (1) add_a | | | | | | | | | |
| (2) add_b | 1.9e-07 | | | | | | | | |
| (3) add_c | 2.8e-01 | 2.2e-04 | | | | | | | |
| (4) com_a | 3.5e-03 | 1.1e-14 | 1.8e-04 | | | | | | |
| (5) com_b | 8.7e-12 | 4.2e-29 | 1.4e-12 | 1.3e-04 | | | | | |
| (6) com_c | 1.1e-66 | 1.8e-106 | 1.1e-64 | 1.9e-40 | 2.5e-16 | | | | |
| (7) lin_a | 1.3e-34 | 5.8e-69 | 1.2e-37 | 1.4e-16 | 1.0e-02 | 2.6e-11 | | | |
| (8) mul_a | 9.8e-02 | 1.9e-08 | 2.6e-02 | 5.1e-01 | 4.1e-04 | 5.6e-30 | 8.9e-12 | | |
| (9) mul_b | 1.4e-30 | 9.0e-53 | 1.0e-30 | 1.0e-17 | 4.8e-06 | 2.4e-02 | 6.8e-03 | 1.9e-14 | |
| (10) mul_c | 7.8e-18 | 1.7e-35 | 7.9e-18 | 4.4e-09 | 6.3e-03 | 2.0e-06 | 8.9e-01 | 1.4e-08 | 3.7e-02 |

**(f)** functional dependence

| | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) | (10) | (11) | (12) | (13) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (1) ANM | | | | | | | | | | | | | |
| (2) CGNN | 1.0e-18 | | | | | | | | | | | | |
| (3) EMD | 5.8e-74 | 9.5e-39 | | | | | | | | | | | |
| (4) GPI | 2.7e-16 | 5.0e-01 | 2.7e-35 | | | | | | | | | | |
| (5) IGCI | 3.9e-85 | 1.1e-47 | 1.3e-02 | 1.8e-44 | | | | | | | | | |
| (6) Jarfo | 3.7e-11 | 9.9e-121 | 1.5e-178 | 2.3e-81 | 5.5e-182 | | | | | | | | |
| (7) LiNGAM | 8.3e-90 | 3.3e-193 | 1.5e-240 | 4.1e-179 | 3.0e-250 | 5.1e-116 | | | | | | | |
| (8) NLME | 1.8e-42 | 2.6e-137 | 5.0e-187 | 6.7e-118 | 1.5e-196 | 2.2e-45 | 1.0e-17 | | | | | | |
| (9) PNL | 3.1e-11 | 5.4e-94 | 1.1e-164 | 3.7e-70 | 5.0e-171 | 9.3e-01 | 1.5e-88 | 3.0e-32 | | | | | |
| (10) QCCD | 6.2e-04 | 2.1e-38 | 1.1e-86 | 3.2e-31 | 5.1e-98 | 9.1e-01 | 6.1e-60 | 1.0e-21 | 3.2e-01 | | | | |
| (11) RCC | 1.9e-67 | 1.3e-177 | 8.3e-223 | 1.6e-155 | 1.2e-231 | 6.4e-80 | 1.2e-05 | 2.0e-05 | 5.2e-60 | 3.8e-40 | | | |
| (12) RECI | 4.3e-120 | 1.8e-90 | 6.2e-15 | 1.7e-80 | 4.5e-07 | 6.9e-231 | 7.8e-263 | 2.0e-220 | 6.3e-217 | 1.5e-127 | 1.2e-251 | | |
| (13) SLOPE | 1.9e-79 | 4.8e-49 | 6.4e-03 | 2.6e-43 | 9.1e-01 | 6.4e-164 | 7.5e-223 | 1.8e-174 | 1.6e-156 | 1.7e-88 | 2.5e-206 | 2.3e-06 | |
| (14) SLOPPY | 5.5e-131 | 4.5e-96 | 1.2e-15 | 2.4e-87 | 1.3e-07 | 6.2e-264 | 1.1e-289 | 2.1e-246 | 4.4e-243 | 9.5e-142 | 7.0e-280 | 9.3e-01 | 1.2e-06 |

**(g)** method

Figure 14: Significances at $\alpha = 0.05$-level *w.r.t.* individual configurations for all methods. The numbering of the row and column heads correspond to each other. Green color coding indicates significant differences while red color coding indicates no significant difference.

## Appendix B. Importance of Data Set Characteristics using Shapley Values Across Methods

In addition to the overall analysis *w.r.t.* the importance of individual data set characteristics in Section 7.3.5, we are also interested in a more granular investigation. Thus, we split the obtained results for individual characteristics and perform the same pipeline as described in Section 5.5 to obtain Shapley explanations. For example, to obtain the Shapley values in Fig. 15, we split all the available 23 184 results (*i.e.,* 828 valid configurations $\times$ 14 methods $\times$ 2 sample sizes) into five groups where each group corresponds to results obtained on data set configurations involving one of five cause distributions (■ bimodal normal, ■ normal, *etc.*). We then compute the Shapley explanations for each group individually which allows us to compare if the importance of characteristics is different among configurations.

The obtained results can be found in Fig. 15 for the cause distributions, in Fig. 16 for the sample sizes, in Fig. 17 for the dependence strength, in Fig. 18 for the functional dependence, and in Fig. 19 for the noise distributions. In general, it can be seen that the method is the most influential factor across all splits (compare Section 7.3.5). The second most influential factor is very consistently the functional dependence, while the sample size almost always is the least influential factor. Interestingly, the cause distribution is only for ■ mul_b function (see Fig. 18) more influential than the method. We attribute this to the challenging shapes the data can have due to interaction of the functional dependence and cause distributions.

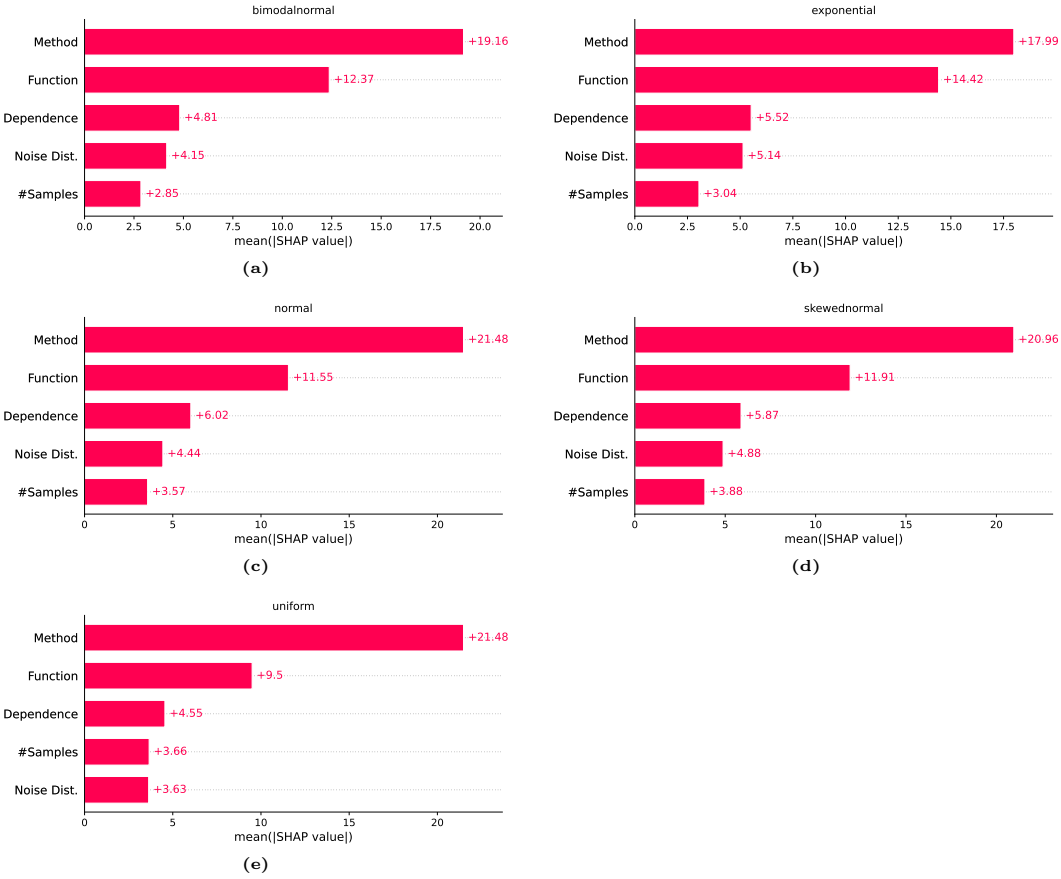



Figure 15: Importance of individual data set characteristics as mean Shapley values for single cause distributions.

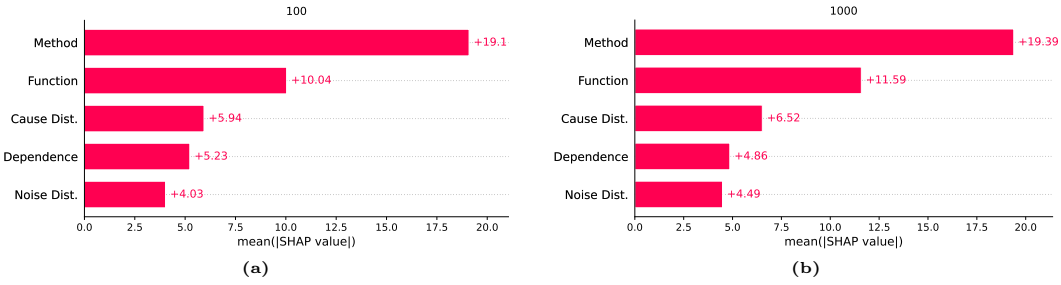



Figure 16: Importance of individual data set characteristics as mean Shapley values for single sample sizes.

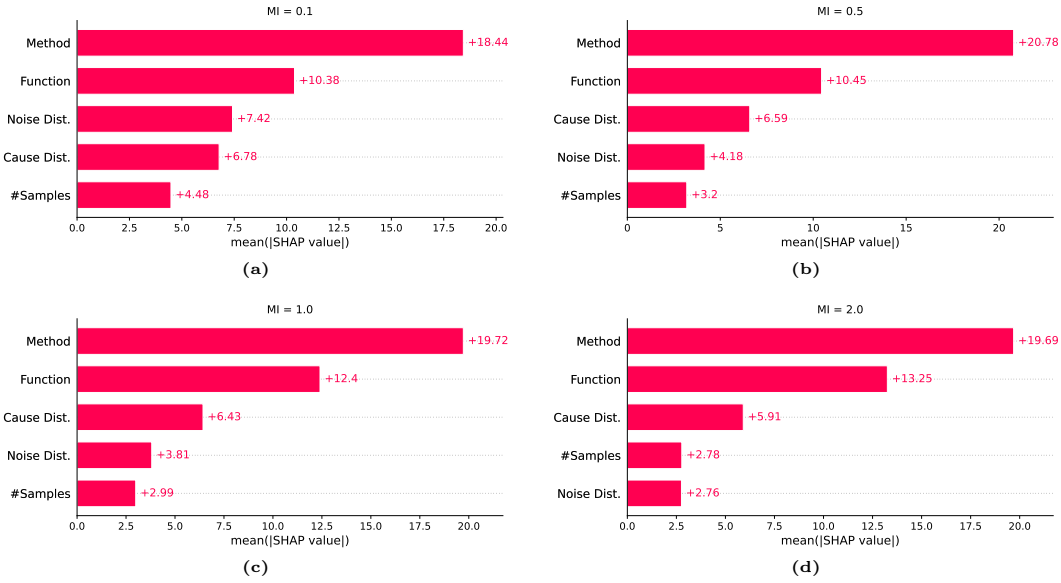



Figure 17: Importance of individual data set characteristics as mean Shapley values for single dependency strengths.

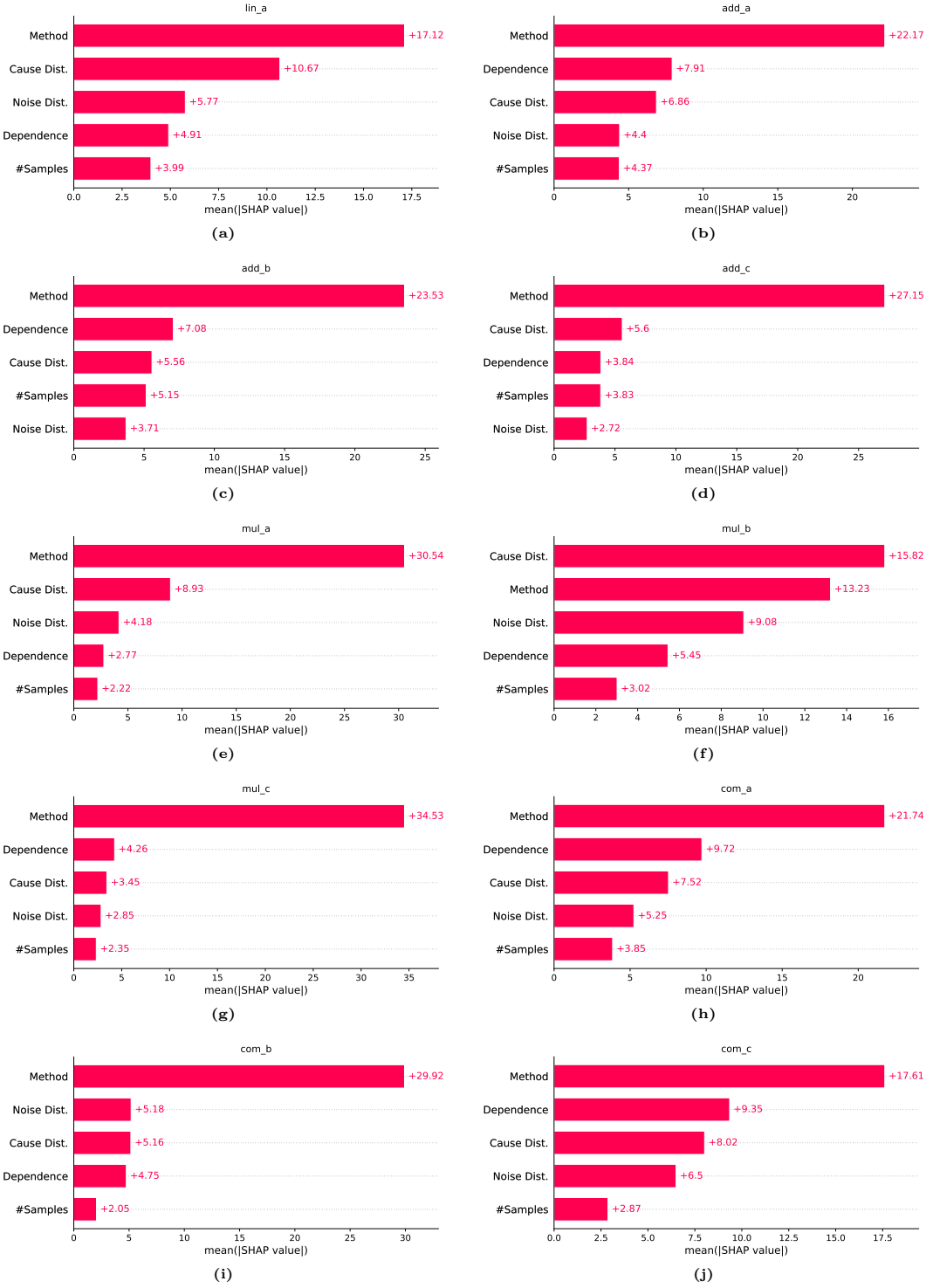



Figure 18: Importance of individual data set characteristics as mean Shapley values for single functional dependencies.

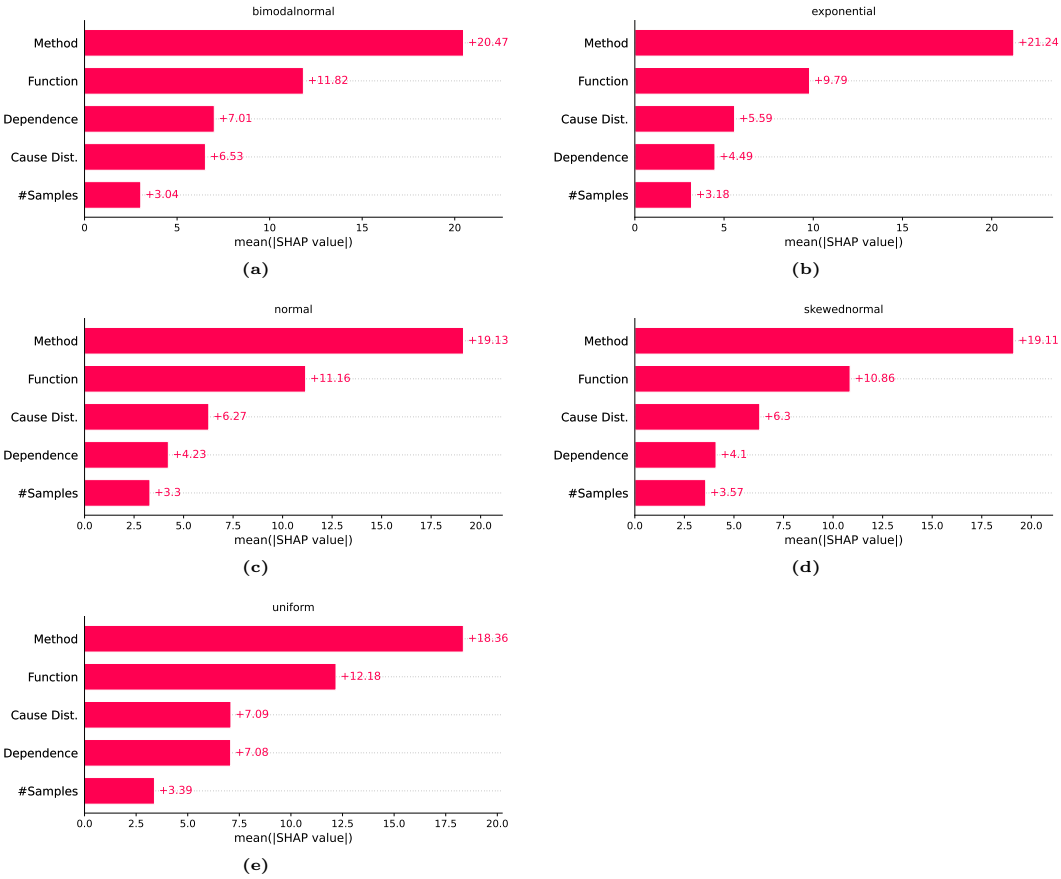



Figure 19: Importance of individual data set characteristics as mean Shapley values for single noise distributions.

| Method | Setting | p-Value |
|---|---|---|
| ANM | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 2.9e-01 |
| CGNN | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 2.0e-74 |
| EMD | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 7.9e-50 |
| GPI | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 1.6e-29 |
| IGCI | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 3.5e-58 |
| Jarfo | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 3.9e-14 |
| LiNGAM | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 2.2e-15 |
| NLME | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 3.0e-47 |
| PNL | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 3.4e-23 |
| QCCD | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 5.3e-05 |
| RCC | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 4.1e-28 |
| RECI | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 5.2e-40 |
| SLOPE | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 1.2e-28 |
| SLOPPY | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 3.0e-41 |

Figure 20: Conditional independence with significance level of $\alpha = 0.05$ of individual data set characteristics *w.r.t.* the obtained accuracy for single cause distributions.

## Appendix C. Analysis of Conditional Independencies among Building Blocks Across Methods

In addition to the statistical analysis in Section 7.3.6, we are also interested in a more detailed look *w.r.t.* the independence regarding the results for individual methods. As in the other more detailed investigations, we split the results accordingly (compare, *e.g.,* Appendix B) and perform conditional independence tests as described in Section 5.7. The obtained results can be found in Fig. 20 for the cause distributions, in Fig. 21 for the sample sizes, in Fig. 22 for the dependence strength, in Fig. 23 for the functional dependence, and in Fig. 24 for the noise distributions. In general, it can be seen that conditional independence can not be assumed for the cause distributions (except for ■ ANM), for the dependence strength, the functional dependence, and for the noise distributions (except for ■ NLME) given a significance level of $\alpha = 5\%$. Thus, we can mostly confirm that each of these building blocks does have a significant impact on the methods performance since we can assume for p-values below the set significance level that the considered building block and the obtained accuracy are dependent despite the information from the remaining building blocks being available. However, we can also see in case of the sample size that several methods results are conditionally independent, *i.e.,* the sample size does not have a significant impact on the results given the other building blocks. This is, for example, consistent with the observation we made in Appendix B, where the sample size is almost always the least influential building block. Finally, please note that an in-depth discussion of individual independencies is given in the corresponding sub-sections of Appendix D.

| Method | Setting | p-Value |
|---|---|---|
| ANM | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 6.1e-06 |
| CGNN | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 3.1e-07 |
| EMD | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 4.6e-04 |
| GPI | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 5.6e-15 |
| IGCI | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 3.8e-03 |
| Jarfo | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 1.1e-06 |
| LiNGAM | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 8.6e-05 |
| NLME | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 1.0e+00 |
| PNL | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 4.5e-01 |
| QCCD | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 2.5e-05 |
| RCC | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 1.1e-13 |
| RECI | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 5.3e-02 |
| SLOPE | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 1.1e-23 |
| SLOPPY | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 3.6e-01 |

Figure 21: Conditional independence with significance level of $\alpha = 0.05$ of individual data set characteristics *w.r.t.* the obtained accuracy for single sample sizes.

| Method | Setting | p-Value |
|---|---|---|
| ANM | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 2.4e-08 |
| CGNN | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 2.9e-14 |
| EMD | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 3.5e-13 |
| GPI | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 1.3e-16 |
| IGCI | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 1.4e-04 |
| Jarfo | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 5.7e-04 |
| LiNGAM | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 3.1e-07 |
| NLME | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 1.2e-04 |
| PNL | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 2.7e-35 |
| QCCD | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 3.9e-31 |
| RCC | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 5.4e-03 |
| RECI | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 1.2e-14 |
| SLOPE | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 5.3e-14 |
| SLOPPY | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 6.3e-13 |

Figure 22: Conditional independence with significance level of $\alpha = 0.05$ of individual data set characteristics *w.r.t.* the obtained accuracy for single dependency strengths.

| Method | Setting | p-Value |
|---|---|---|
| ANM | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 2.2e-235 |
| CGNN | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 9.5e-51 |
| EMD | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 8.2e-147 |
| GPI | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 8.3e-68 |
| IGCI | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 1.6e-156 |
| Jarfo | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 8.5e-53 |
| LiNGAM | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 3.1e-201 |
| NLME | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 9.8e-223 |
| PNL | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 1.6e-209 |
| QCCD | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 7.7e-141 |
| RCC | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 2.9e-36 |
| RECI | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 2.0e-135 |
| SLOPE | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 1.7e-154 |
| SLOPPY | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 1.7e-102 |

Figure 23: Conditional independence with significance level of $\alpha = 0.05$ of individual data set characteristics *w.r.t.* the obtained accuracy for single functional dependencies.

| Method | Setting | p-Value |
|---|---|---|
| ANM | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 1.1e-06 |
| CGNN | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 1.5e-09 |
| EMD | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 9.0e-06 |
| GPI | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 3.9e-08 |
| IGCI | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 1.0e-20 |
| Jarfo | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 3.2e-10 |
| LiNGAM | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 2.2e-02 |
| NLME | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 2.2e-01 |
| PNL | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 4.1e-02 |
| QCCD | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 6.8e-63 |
| RCC | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 3.3e-02 |
| RECI | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 2.0e-20 |
| SLOPE | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 1.3e-19 |
| SLOPPY | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 3.6e-21 |

Figure 24: Conditional independence with significance level of $\alpha = 0.05$ of individual data set characteristics *w.r.t.* the obtained accuracy for single noise distributions.

## Appendix D. In-depth Evaluation of Individual Methods

The focus of the main manuscript is on a compact comparative overview of the obtained results. In the following, an additional introduction as well as in-depth evaluation of each method is presented to give more insight into the strengths and weaknesses of each method.

Each section from Appendix D.1 to Appendix D.14 begins with a compact introduction of the individual methods. This is followed by a presentation and brief discussion of the results for each method, where we try to link the performance of the methods to the respective underlying assumptions for each method where possible. The presentation of the results is structured as follows: The first page shows the results on each considered data collection as a footprint (see Section 5.3). We also present a more fine grained view on the results on our proposed data collection *w.r.t.* the functional dependence (most important building block after the method itself, compare Section 7.3.5) as a footprint. Further, we show the most influential building blocks according to their Shapley values computed as introduced in Section 5.5 as well as a statistical analysis for conditional independencies of building blocks and accuracy as discussed in Section 5.7. Thus, the first page of each result display delivers a good overview about the obtained accuracy and the most influential factors. The following pages show a more in-depth view on the results as footprints and the distribution of results for each building block as violins (see Section 5.4). Further, we also show the number of invalid decisions per data collection (see Section 5.2) as well as the statistical tests to investigate whether there is a significant difference between configurations (see Section 5.6).

### D.1 Additive Noise Model

#### D.1.1 Description

The additive noise model ($ANM$[8]) is presented in Hoyer et al. (2009). It assumes the functional dependence $Y = f_X(X)+\epsilon_Y$. Here, $f_X(\cdot)$ is a non-linear function and $\epsilon_Y$ is an additive non-specified noise term independent from $X$. The non-linearity helps to identify the causal direction since it breaks symmetries between $X$ and $Y$. The causal direction is determined by fitting a non-linear regression $\hat{f}_X(X)$ and then calculate the residual $\hat{\epsilon}_Y = Y - \hat{f}_X(X)$. If $\hat{\epsilon}_Y$ is independent from $X$, the causal direction is accepted to be the true one, or is rejected otherwise ($X$ and $Y$ being swapped for the test of the opposite direction). For implementing the individual components, the authors rely on Gaussian processes (Rasmussen and Williams, 2006) and the kernel-based Hilbert-Schmidt independence criterion (*HSIC*) as introduced in (Gretton et al., 2008). The authors present proofs for the identifiability (all probability densities are strictly positive, involved functions and densities are three times differentiable). They further give examples for non-identifiable models, *i.e.,* cases where a model can be found for the anti-causal direction. A deeper review including further ablations and derivations can be found, *e.g.,* in (Mooij et al., 2016).

#### D.1.2 Results

Obtained results are shown in Figs. 25 to 31. The results on data sets available in the literature are in an expected range (see Fig. 25.a). We can observe accuracies between $\sim 35\%$ on CE-Multi and $\sim 80\%$ on CE-Gauss. Examining the different functional dependencies of our bSCMC data (see Fig. 25.b) shows that ANM delivers high performance on data generated with a linear (*i.e.,* lin_a) or additive functional dependence (*i.e.,* add_a, add_b, add_c), *i.e.,* functions that meet the underlying assumptions (compare Fig. 29). However, the method tends to fail on data with low MI even though the functional dependence includes additive noise (*i.e.,* on add_a, add_b, add_c, see Fig. 28.a,b,c). Interestingly, we obtain more diverse results *w.r.t.* MI on lin_a (see Fig. 28.d) for different values of MI, which we attribute to compound effects of the building blocks. But we see also that a normal cause in connection with a normal noise and the linear function lin_a results in accuracies at $\sim 50\%$ as expected (Shimizu et al., 2006) since this constitutes a non-identifiable case. Further, our results regarding add_c (see Fig. 28.c) reveal that the distribution of the cause can have a huge impact (see bimodal normal cause distribution) and can eventually lead to systematic errors depending on the actual mechanism. From Fig. 28, we can generally observe that more data helps to identify the causal direction. All the results obtained are actual decisions taken by the algorithm as Fig. 30 shows, *i.e.,* no invalid decisions occurred. Examining the statistical dependencies in Fig. 31 shows that most of the pairwise tests for cause distributions and noise distributions are not significant. Note the protocol for statistical testing, which does not take into account compound effects. However, Fig. 27 tells that the cause distribution is not statistically significant *w.r.t.* the impact on the resulting accuracy. Especially the functional dependence (see Fig. 31.f

8. We use the publicly available code contained in the causal discovery toolbox (Kalainathan and Goudet, 2019). The data is centered at zero mean and scaled to unit variance before the method is applied.

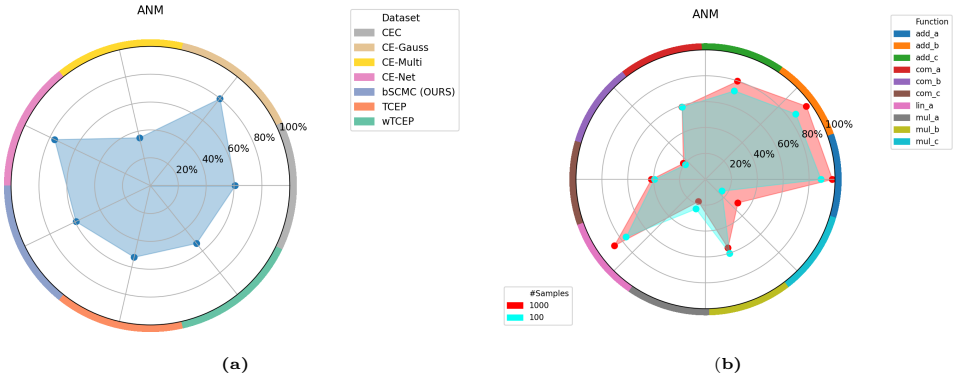



Figure 25: Result footprints for ANM.

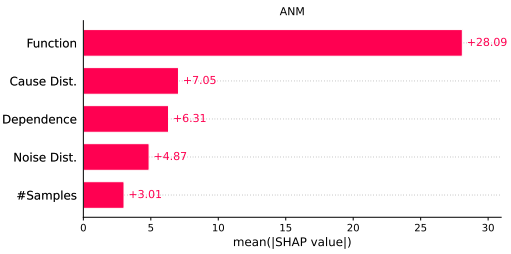



Figure 26: Importance of individual data set characteristics as mean Shapley values for ANM.

| Method | Setting | p-Value |
|---|---|---|
| ANM | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 2.9e-01 |
| ANM | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 6.1e-06 |
| ANM | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 2.4e-08 |
| ANM | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 2.2e-235 |
| ANM | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 1.1e-06 |

Figure 27: Significances at $\alpha = 0.05$-level for conditional independencies of individual data set characteristics *w.r.t.* the obtained accuracy for ANM. Green color coding indicates dependence while red color coding indicates independence.

and Fig. 26) is playing a major role, which is an expected outcome given the underlying assumptions and the implementation.

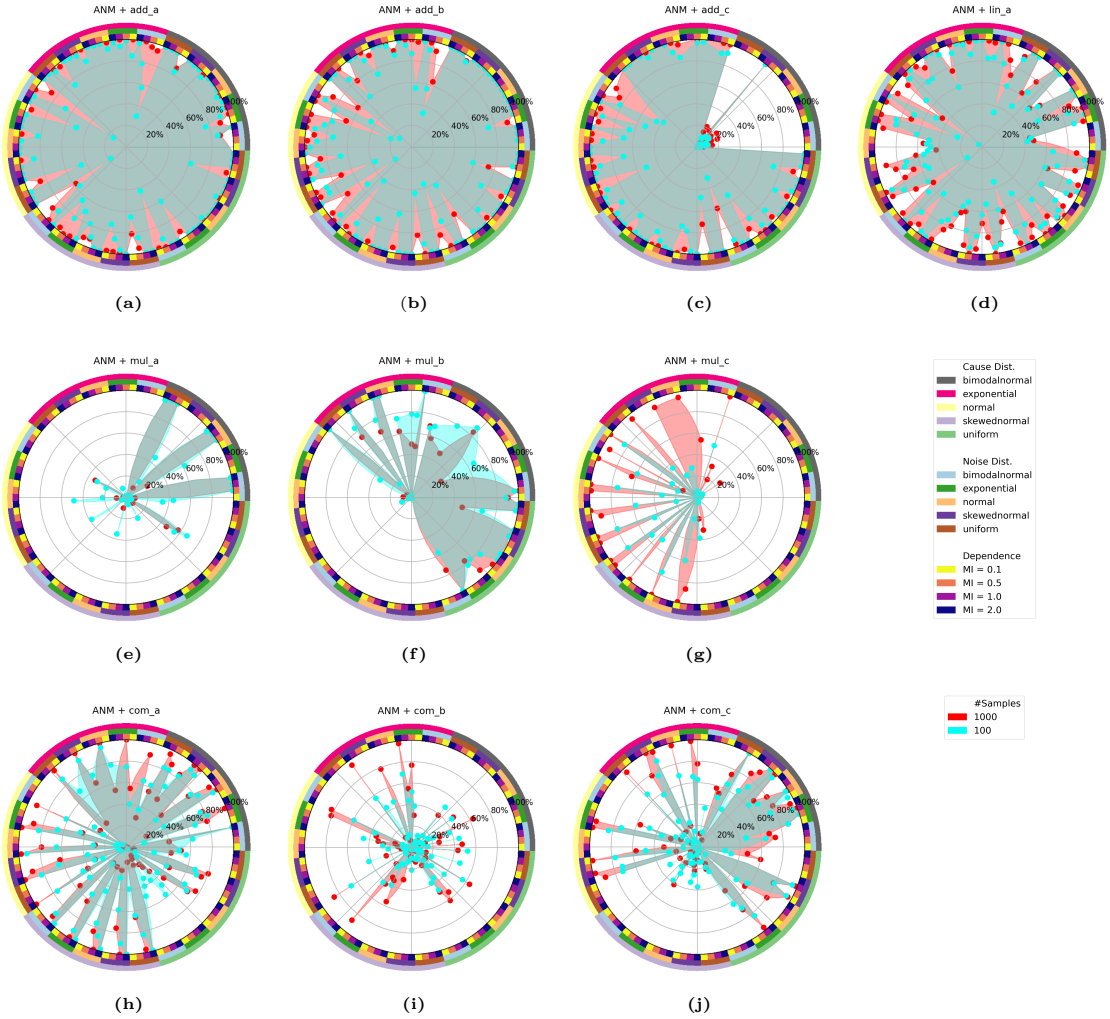



Figure 28: Detailed result overview on our bSCMC data as footprints for ANM.

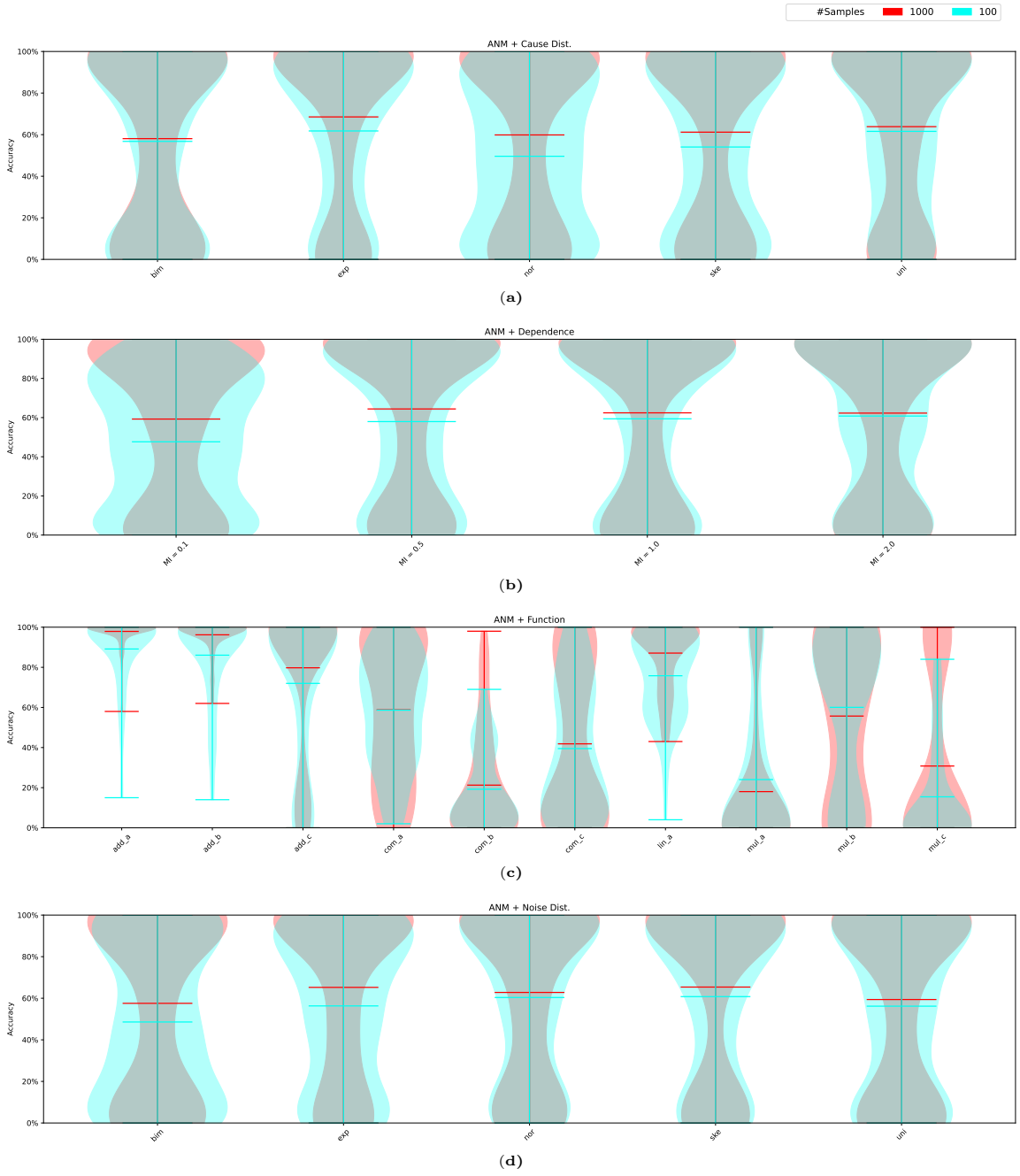



Figure 29: Distribution of accuracies *w.r.t.* individual configurations on our proposed data collection as violins for ANM. Please note, the width of the violins is scaled to unit width individually.

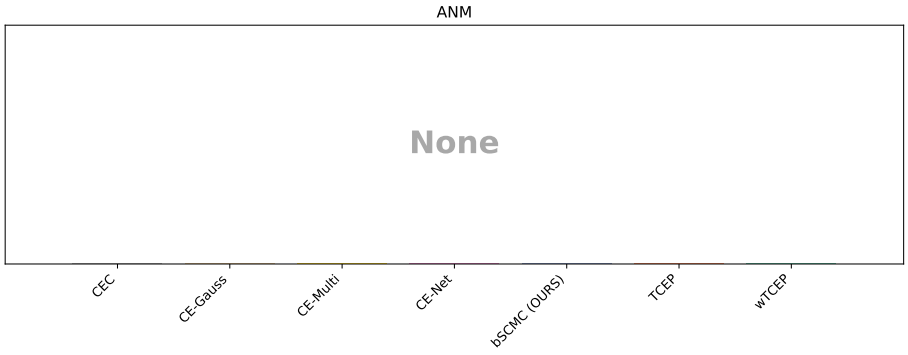



Figure 30: Percentage of invalid decisions for ANM. Note that the y-axis is logarithmically scaled.

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 1.1e-01 | | | |
| (3) normal | 1.6e-01 | 8.2e-04 | | |
| (4) skewednormal | 6.4e-01 | 3.2e-02 | 3.2e-01 | |
| (5) uniform | 2.2e-01 | 5.6e-01 | 4.5e-03 | 9.2e-02 |

**(a)** cause distribution

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 4.7e-02 | | | |
| (3) normal | 4.8e-03 | 3.5e-01 | | |
| (4) skewednormal | 2.6e-03 | 2.0e-01 | 8.3e-01 | |
| (5) uniform | 1.1e-01 | 8.1e-01 | 2.3e-01 | 1.7e-01 |

**(b)** noise distribution

| | (1) | (2) | (3) |
|---|---|---|---|
| (1) MI = 0.1 | | | |
| (2) MI = 0.5 | 1.1e-05 | | |
| (3) MI = 1.0 | 1.6e-05 | 8.7e-01 | |
| (4) MI = 2.0 | 2.0e-06 | 3.3e-01 | 4.4e-01 |

**(c)** mutual information

| | (1) |
|---|---|
| (1) 1000 | |
| (2) 100 | 3.1e-05 |

**(d)** sample size

| | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) |
|---|---|---|---|---|---|---|---|---|---|
| (1) add_a | | | | | | | | | |
| (2) add_b | 5.9e-01 | | | | | | | | |
| (3) add_c | 8.9e-04 | 3.5e-03 | | | | | | | |
| (4) com_a | 3.6e-33 | 3.3e-29 | 6.5e-11 | | | | | | |
| (5) com_b | 4.9e-60 | 1.4e-58 | 2.8e-33 | 6.1e-26 | | | | | |
| (6) com_c | 4.4e-45 | 5.2e-42 | 2.1e-20 | 9.0e-07 | 8.8e-08 | | | | |
| (7) lin_a | 9.2e-15 | 1.5e-11 | 9.3e-03 | 2.5e-12 | 1.3e-52 | 1.7e-25 | | | |
| (8) mul_a | 2.5e-36 | 3.2e-36 | 4.5e-25 | 8.4e-19 | 2.2e-03 | 8.0e-10 | 4.2e-29 | | |
| (9) mul_b | 3.2e-14 | 8.3e-13 | 6.4e-06 | 6.6e-01 | 7.7e-10 | 3.2e-03 | 5.2e-04 | 1.2e-08 | |
| (10) mul_c | 3.7e-42 | 3.3e-41 | 5.0e-28 | 1.2e-18 | 1.4e-02 | 3.5e-09 | 1.2e-31 | 8.6e-01 | 7.4e-09 |

**(f)** functional dependence

Figure 31: Significances at $\alpha = 0.05$-level *w.r.t.* individual configurations for ANM. The numbering of the row and column heads correspond to each other. Green color coding indicates significant differences while red color coding indicates no significant difference.

## D.2 Causal Generative Neural Networks

### D.2.1 Description

Utilizing neural networks to learn a deep generative model of the joint distribution of observed variables is presented in (Goudet et al., 2018). To do so, the authors minimize the maximum mean discrepancy between generated and observed data. In a bit more detail, the optimization process consists of two parts: (1) An optimization of the structure (*i.e.,* the links) of a DAG fitting the data, and (2) an optimization of the functional dependence between variables given a fixed structure. Since the model class that can be learned by a sufficiently flexible neural network is not restricted in general, the functional dependence between the observed variables is not restricted as well. The approach is called Causal Generative Neural Networks ($CGNN$[9]).

### D.2.2 Results

The obtained results can be found in Figs. 32 to 38. We can generally observe that the method achieves results roughly comparable to those found in previous studies. Thus, we can see results ranging from $\sim 57\%$ on CEC13 to $\sim 88\%$ on CE-Multi (see Fig. 32.a). The method does not make (or explicitly spell out) any assumptions *w.r.t.* functional dependence. We do not see any obvious pattern either in the general overview (see Fig. 32.a) nor in the function overview of our synthetic bSCMC data (see Fig. 32.b). However, we can observe overall that more samples per realization lead to improved accuracies, which is not surprising considering the nature of the approach. It is generally surprising that the method is able to achieve quite competitive results even in the small sample regime where only 100 samples are available to reproduce the generation process. Further, while there appears to be no apparent preference of the method *w.r.t.* the functional dependency (except for mul_b, compare Fig. 36.c), there are quite obvious star-shaped patterns observable in all the detailed results plots (see Fig. 35). They are the result of good performances within the higher range of dependence (*i.e.,* MI $= 1.0$ and MI $= 2.0$) and poorer results within the lower range of dependence (*i.e.,* MI $= 0.5$ and MI $= 0.1$). This is also visible in Fig. 36.b. We can also see some hints towards systematic errors (*i.e.,* an average accuracy of 0% that means the algorithm always decides the wrong direction) for bimodal normal cause distributions (compare Fig. 35 and Fig. 36.a). Since the number of invalid decisions (Fig. 37) is very low, these systematic errors result from actual wrong decisions. Besides the overall good results for the method, we feel urged to mention the slow runtime of its implementation (which is not a surprise since it tries to recreate the whole generation process). By evaluating the significances presented in Fig. 38 and Fig. 34, we can conclude that there is no individual building block that does not impact the performance of the algorithm at all. Considering the importance of individual characteristics, we can conclude from Fig. 33, that the functional dependence as well as the cause distributions appear to matter the most. We see a consistent pattern here comparing this to the mentioned

9. The authors provide two different implementations of their code (PyTorch and tensorflow). We use the one contained within the causal discovery toolbox (Kalainathan and Goudet, 2019) that is also co-authored by the first author of this method. As for parameters, we left everything by default as suggested by the causal discovery toolbox.

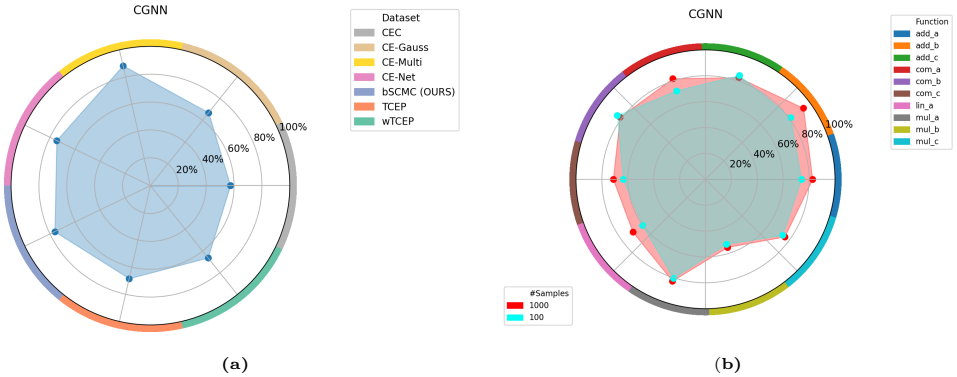



Figure 32: Result footprints for CGNN.

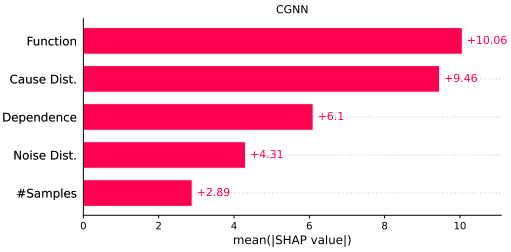



Figure 33: Importance of individual data set characteristics as mean Shapley values for CGNN.

| Method | Setting | p-Value |
|---|---|---|
| CGNN | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 2.0e-74 |
| CGNN | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 3.1e-07 |
| CGNN | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 2.9e-14 |
| CGNN | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 9.5e-51 |
| CGNN | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 1.5e-09 |

Figure 34: Significances at $\alpha = 0.05$-level for conditional independencies of individual data set characteristics *w.r.t.* the obtained accuracy for CGNN. Green color coding indicates dependence while red color coding indicates independence.

failure cases visible, *e.g.,* in Fig. 36. However, these may become more visible considering compound effects.

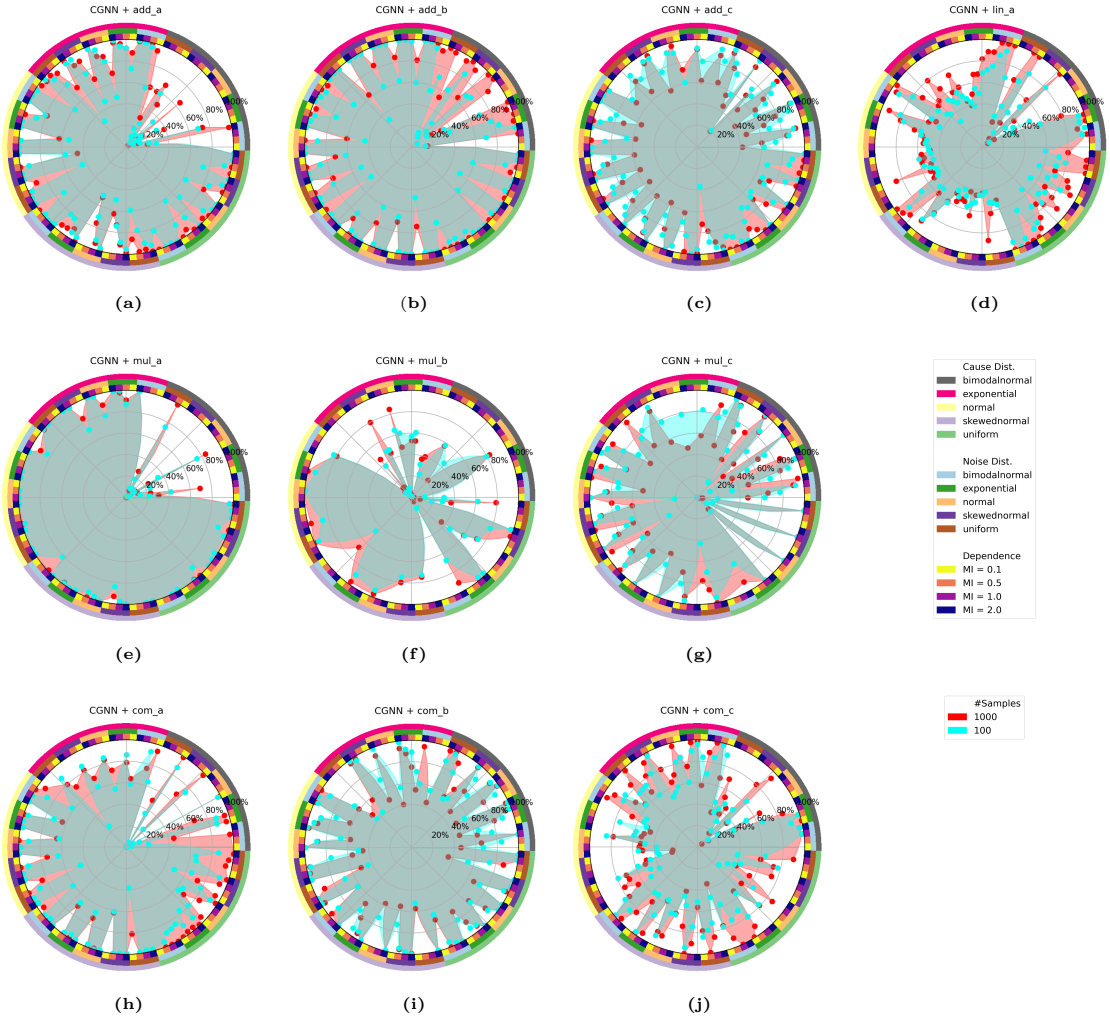



Figure 35: Detailed result overview on our bSCMC data as footprints for CGNN.

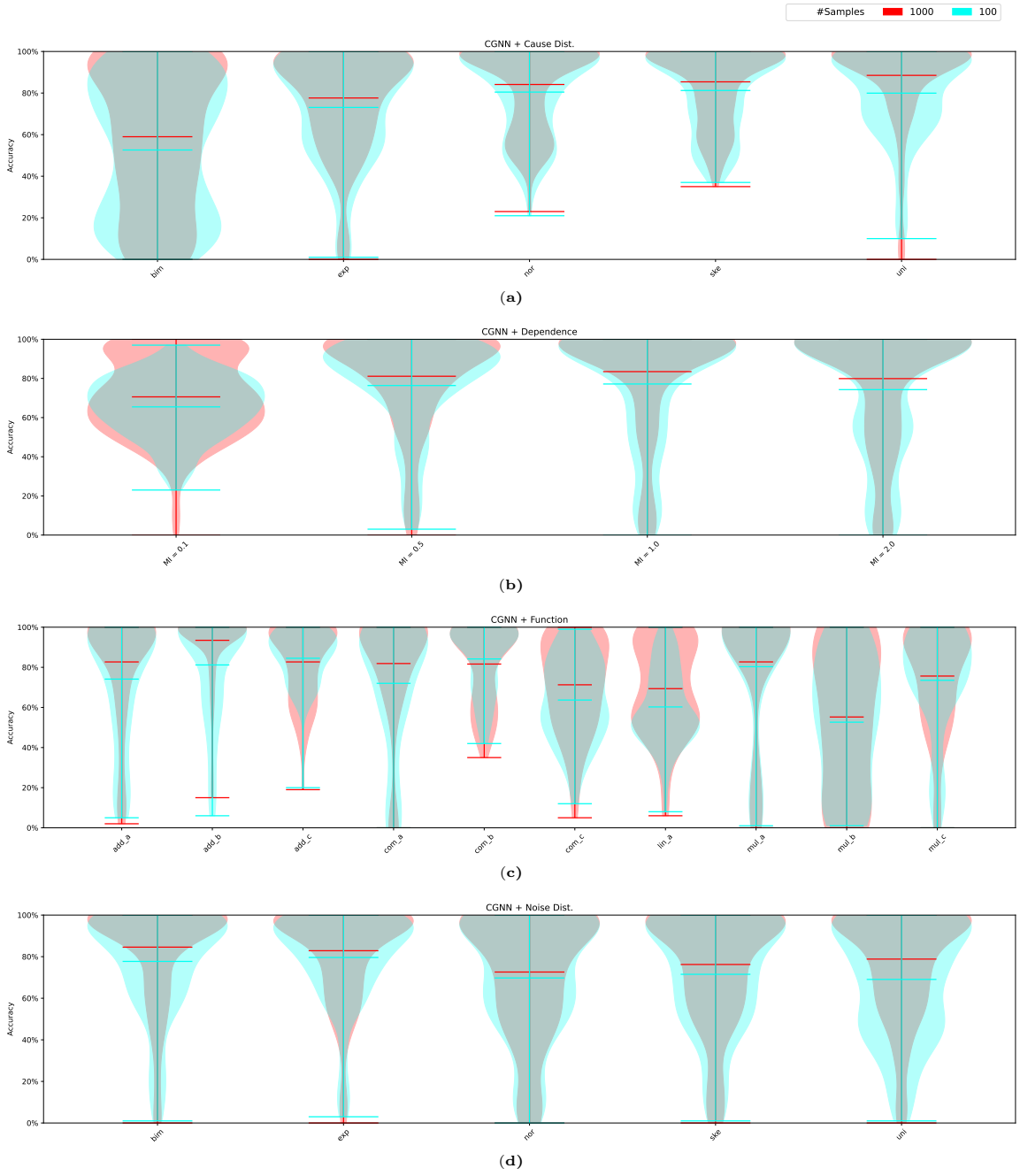



Figure 36: Distribution of accuracies *w.r.t.* individual configurations on our proposed data collection as violins for CGNN. Please note, the width of the violins is scaled to unit width individually.

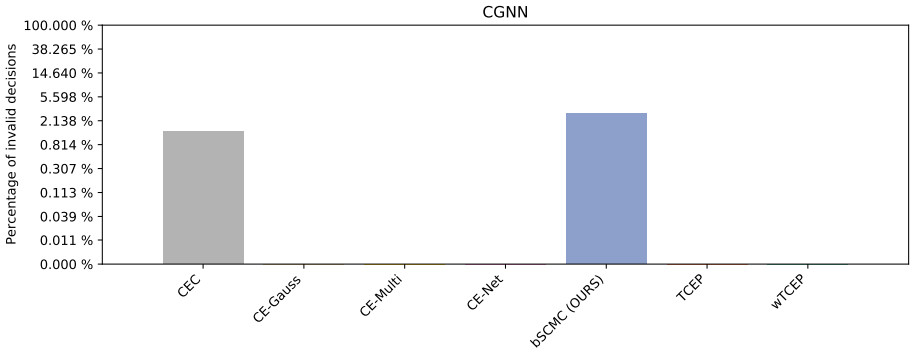



Figure 37: Percentage of invalid decisions for CGNN. Note that the y-axis is logarithmically scaled.

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 1.2e-13 | | | |
| (3) normal | 1.4e-30 | 1.2e-05 | | |
| (4) skewednormal | 7.1e-30 | 2.4e-05 | 6.6e-01 | |
| (5) uniform | 1.9e-34 | 3.0e-09 | 1.5e-01 | 4.5e-02 |

**(a)** cause distribution

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 1.6e-01 | | | |
| (3) normal | 9.3e-06 | 2.1e-04 | | |
| (4) skewednormal | 9.2e-05 | 2.8e-03 | 4.8e-01 | |
| (5) uniform | 2.9e-04 | 5.1e-03 | 3.7e-01 | 8.3e-01 |

**(b)** noise distribution

| | (1) | (2) | (3) |
|---|---|---|---|
| (1) MI = 0.1 | | | |
| (2) MI = 0.5 | 5.8e-22 | | |
| (3) MI = 1.0 | 6.3e-29 | 1.1e-05 | |
| (4) MI = 2.0 | 1.7e-18 | 2.9e-04 | 9.5e-01 |

**(c)** mutual information

| | (1) |
|---|---|
| (1) 1000 | |
| (2) 100 | 9.8e-10 |

**(d)** sample size

| | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) |
|---|---|---|---|---|---|---|---|---|---|
| (1) add_a | | | | | | | | | |
| (2) add_b | 1.4e-05 | | | | | | | | |
| (3) add_c | 4.3e-01 | 2.0e-07 | | | | | | | |
| (4) com_a | 3.7e-01 | 4.7e-07 | 9.5e-01 | | | | | | |
| (5) com_b | 5.6e-01 | 1.8e-07 | 9.5e-01 | 8.8e-01 | | | | | |
| (6) com_c | 4.0e-10 | 3.7e-24 | 1.6e-13 | 1.0e-08 | 7.9e-12 | | | | |
| (7) lin_a | 1.2e-12 | 7.0e-28 | 3.6e-18 | 2.9e-11 | 1.2e-15 | 1.8e-01 | | | |
| (8) mul_a | 1.9e-03 | 6.0e-01 | 3.7e-05 | 5.5e-05 | 6.4e-05 | 2.0e-14 | 1.9e-15 | | |
| (9) mul_b | 7.8e-10 | 1.7e-18 | 1.1e-11 | 4.7e-08 | 4.7e-11 | 6.8e-03 | 4.0e-02 | 5.1e-12 | |
| (10) mul_c | 1.0e-02 | 3.1e-10 | 1.4e-02 | 7.0e-02 | 2.2e-02 | 2.4e-04 | 1.7e-06 | 9.1e-07 | 6.0e-06 |

**(f)** functional dependence

Figure 38: Significances at $\alpha = 0.05$-level *w.r.t.* individual configurations for CGNN. The numbering of the row and column heads correspond to each other. Green color coding indicates significant differences while red color coding indicates no significant difference.

### D.3 Causal Discovery via Reproducing Kernel Hilbert Space Embeddings

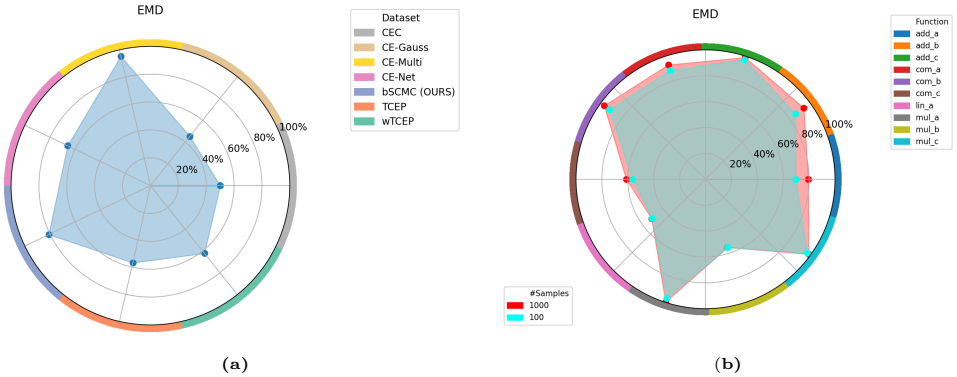
#### D.3.1 Description

Chen et al. (2014) propose a method that follows the idea of the independence between cause and mechanism. In more detail, they derive a suitable independence or uncorrelatedness criterion and justify, based on this, an asymmetry between cause and effect in the true causal direction. This asymmetry is framed as a complexity metric that measures the distance of the marginal and the conditional *w.r.t.* a uniform reference distribution. Since this general criterion is difficult to compute in practice, the authors propose a method based on reproducing kernel Hilbert space embeddings that exploits those asymmetries. Due to this step, the authors call their method *EMD*[10] (coming from EMbeDding). They show that the embedding preserves the relative magnitude of the complexity metrics while the computation is feasible. The complexity measure is lower in the true causal direction which allows for deciding upon the causal direction through given observational data. The method does not assume any kind of functional dependence between cause and effect as well as it is not restricted to additive noise. Further, it can also be used with multi-dimensional data (due to the application of kernels) or multiple variables.

#### D.3.2 Results

Results obtained by this method can be found in Figs. 39 to 45. The overall results *w.r.t.* all data collections range from $\sim 45\%$ accuracy on CE-Gauss to $\sim 95\%$ on CE-Multi (see Fig. 39.a). By examining the functional dependencies (see Fig. 39.b), we can see that more data appears to improve the performance only marginally. Thus, the method appears to work well already with only 100 samples. However, the significances of the results reveal that even if the improvements appear to be small, they are significant (see Fig. 45.d). We can also see that the performance *w.r.t.* functions mul_b, com_c, and also surprisingly lin_a are worse than on the remaining functions (see also Fig. 42.d,f,j and Fig. 43.c). A possible reason might be that the proposed complexity metrics used by the method are not capable to grasp more complex data in case of mul_b and com_c while the difference in complexity in the linear case can only be caused by the data distributions. Also the general results for the exponential cause distribution supporting this (see Fig. 43.a). An observation which might support this is that in case of, *e.g.,* add_a and add_b, the method fails in the low dependence regime (*i.e.,* MI $= 0.1$, see Fig. 42.a,b and Fig. 43.b). However, this does not explain the performance of bimodal normal and uniform cause distributions in comparison with the remaining cause distributions for lin_a (Fig. 42.d and Fig. 43.a). This result appears inconclusive. We assume the reason lies within compound effects of the individual configurations that enable the complexity metric to grasp the correct direction or not. Also that the functional dependence does have by far the largest importance of all the characteristics might be a sign for that (see Fig. 40). Examining the significances of the results (see Fig. 45 and Fig. 41) does support

10. We obtained our results using the code provided at `bitbucket.org/dhernand/gr_causal_inference` while leaving method-inherent parameters to their default values. The data is not transformed in any way before applying the authors code. However, the authors suggest to use bootstrapping to cope with small sample sizes. We refrain from doing so in order to achieve better comparability between methods since other methods might benefit from this as well.




(a) (b)

Figure 39: Result footprints for EMD.

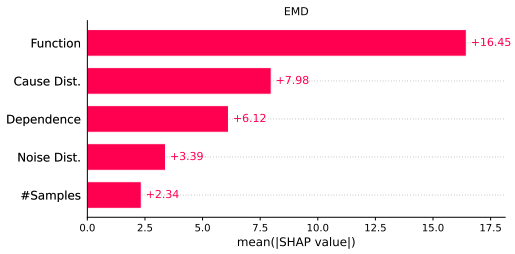



Figure 40: Importance of individual data set characteristics as mean Shapley values for EMD.

| Method | Setting | p-Value |
|---|---|---|
| EMD | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 7.9e-50 |
| EMD | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 4.6e-04 |
| EMD | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 3.5e-13 |
| EMD | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 8.2e-147 |
| EMD | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 9.0e-06 |

Figure 41: Significances at $\alpha = 0.05$-level for conditional independencies of individual data set characteristics *w.r.t.* the obtained accuracy for EMD. Green color coding indicates dependence while red color coding indicates independence.

this impression since all the building blocks are conditional dependent in general while some configurations from every building block do not deliver significantly different results. Compound effects are thus probable. Surprisingly, there are no pairwise significant different results *w.r.t.* different noise distributions (see Fig. 45.b). Finally, by considering Fig. 44 we can observe that all the low accuracies are indeed caused by deciding for the wrong direction (*i.e.*, there is no invalid decision).

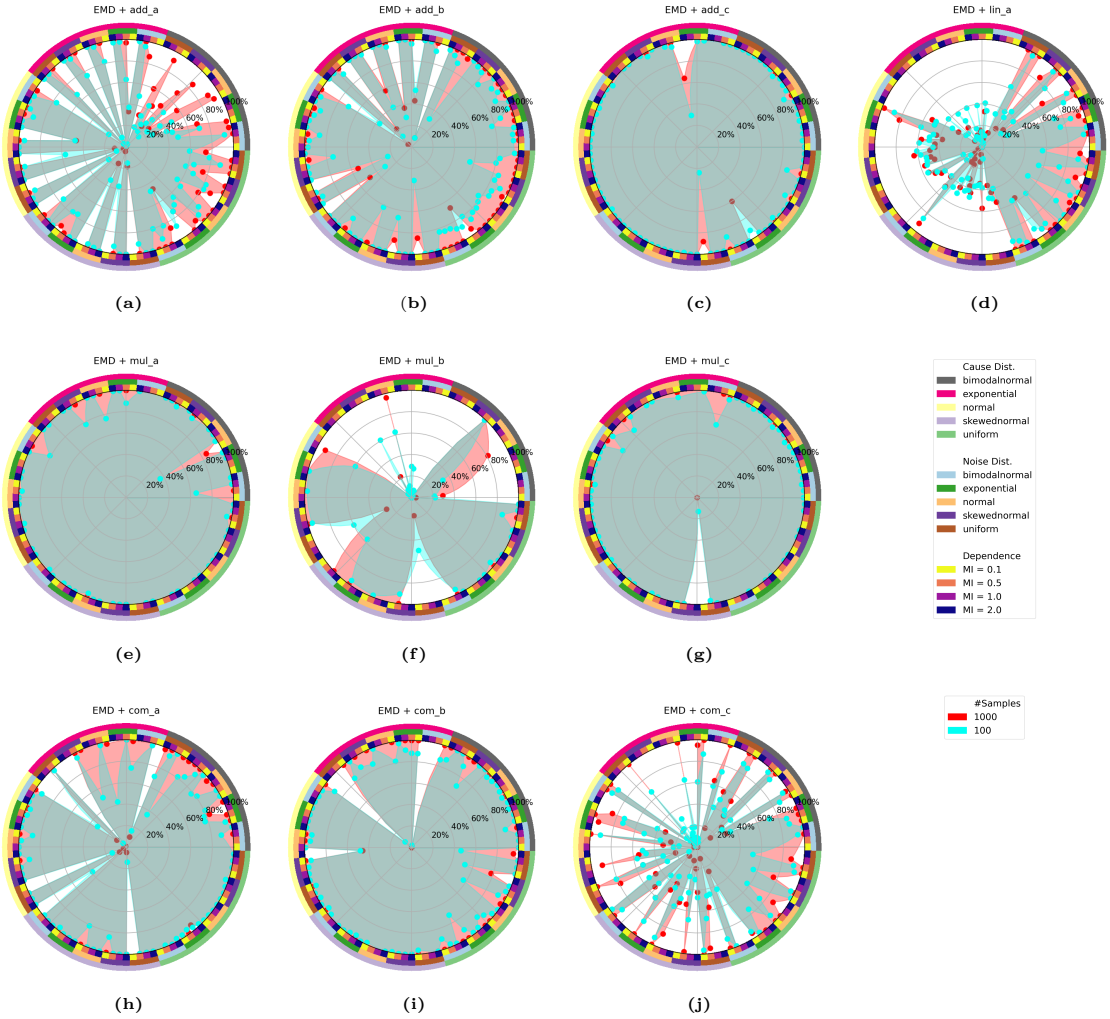



Figure 42: Detailed result overview on our bSCMC data as footprints for EMD.

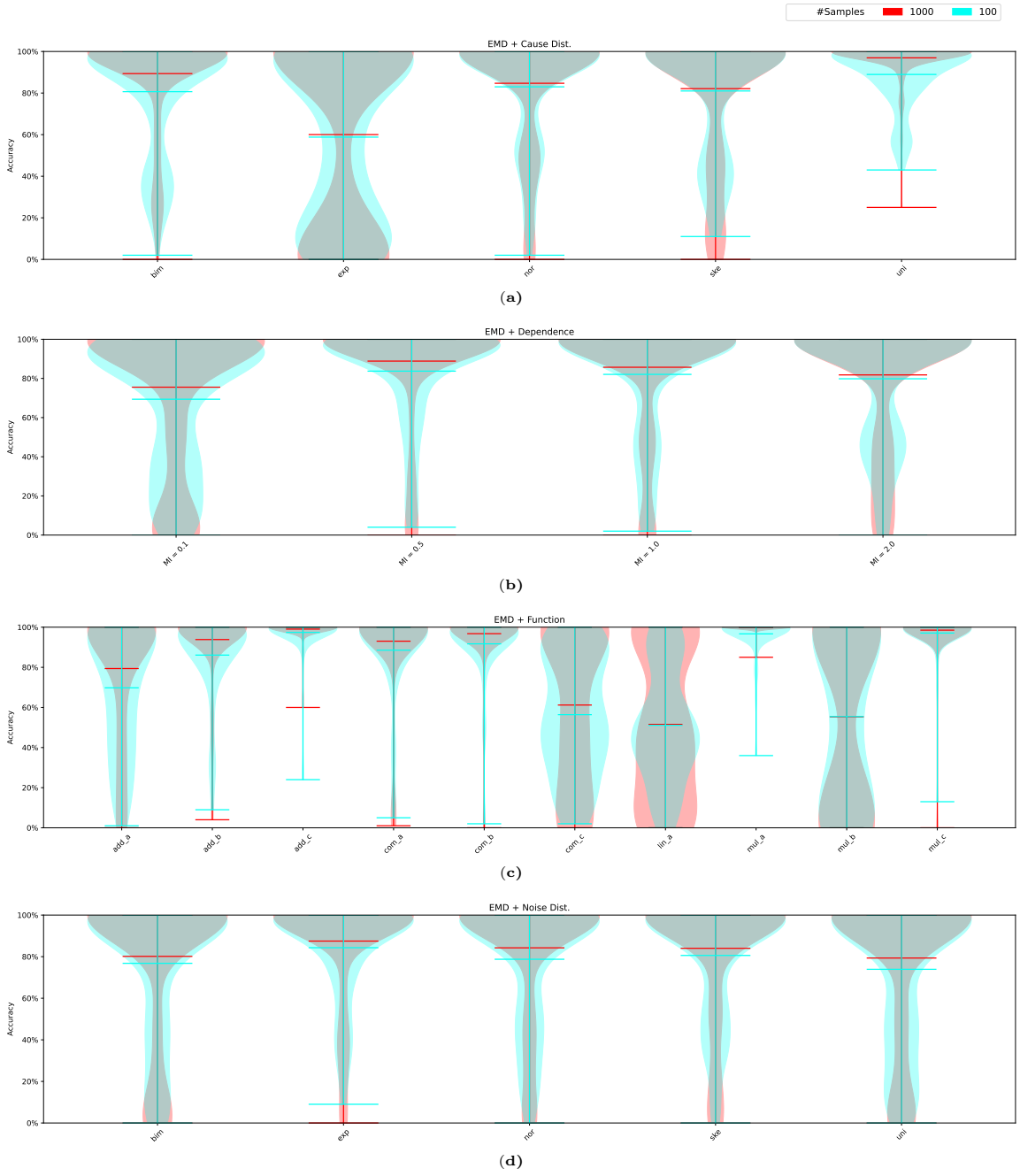



Figure 43: Distribution of accuracies *w.r.t.* individual configurations on our proposed data collection as violins for EMD. Please note, the width of the violins is scaled to unit width individually.

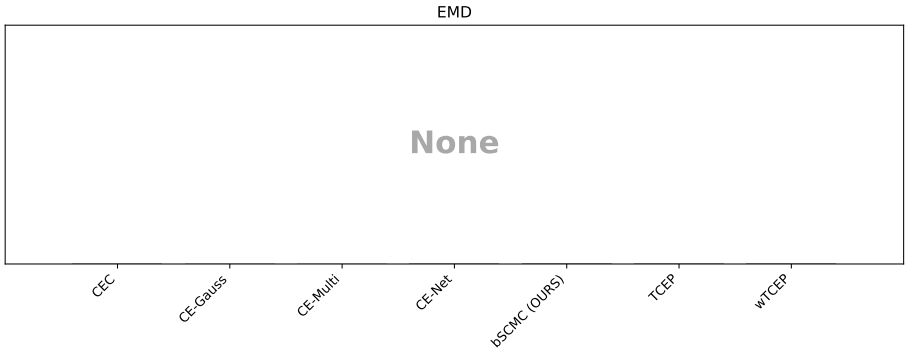



Figure 44: Percentage of invalid decisions for EMD. Note that the y-axis is logarithmically scaled.

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 6.0e-11 | | | |
| (3) normal | 1.5e-01 | 1.2e-13 | | |
| (4) skewednormal | 1.3e-01 | 1.1e-13 | 9.1e-01 | |
| (5) uniform | 1.1e-03 | 9.8e-19 | 4.8e-02 | 8.0e-02 |

**(a)** cause distribution

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 5.0e-01 | | | |
| (3) normal | 7.5e-01 | 6.3e-01 | | |
| (4) skewednormal | 4.2e-01 | 9.5e-01 | 6.0e-01 | |
| (5) uniform | 9.2e-01 | 3.3e-01 | 6.0e-01 | 3.2e-01 |

**(b)** noise distribution

| | (1) | (2) | (3) |
|---|---|---|---|
| (1) MI = 0.1 | | | |
| (2) MI = 0.5 | 1.0e-09 | | |
| (3) MI = 1.0 | 1.0e-12 | 7.1e-02 | |
| (4) MI = 2.0 | 1.8e-10 | 2.2e-01 | 6.4e-01 |

**(c)** mutual information

| | (1) |
|---|---|
| (1) 1000 | |
| (2) 100 | 3.8e-29 |

**(d)** sample size

| | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) |
|---|---|---|---|---|---|---|---|---|---|
| (1) add_a | | | | | | | | | |
| (2) add_b | 2.9e-07 | | | | | | | | |
| (3) add_c | 1.2e-24 | 1.1e-10 | | | | | | | |
| (4) com_a | 1.8e-09 | 1.3e-01 | 9.3e-07 | | | | | | |
| (5) com_b | 1.6e-11 | 3.6e-02 | 1.9e-06 | 7.4e-01 | | | | | |
| (6) com_c | 1.3e-06 | 2.9e-22 | 4.6e-42 | 9.7e-24 | 5.7e-28 | | | | |
| (7) lin_a | 2.2e-11 | 1.4e-29 | 7.3e-48 | 1.9e-29 | 1.3e-34 | 4.2e-02 | | | |
| (8) mul_a | 3.9e-19 | 2.5e-08 | 9.5e-01 | 1.6e-05 | 3.1e-05 | 1.0e-32 | 2.8e-37 | | |
| (9) mul_b | 5.0e-03 | 8.2e-09 | 4.1e-21 | 3.6e-10 | 5.5e-11 | 9.9e-01 | 3.7e-01 | 2.5e-17 | |
| (10) mul_c | 6.0e-18 | 8.4e-07 | 1.7e-01 | 5.9e-04 | 9.3e-04 | 6.8e-33 | 1.1e-37 | 2.0e-01 | 1.2e-15 |

**(f)** functional dependence

Figure 45: Significances at $\alpha = 0.05$-level *w.r.t.* individual configurations for EMD. The numbering of the row and column heads correspond to each other. Green color coding indicates significant differences while red color coding indicates no significant difference.

## D.4 Probabilistic Latent Variable Models

### D.4.1 Description

Mooij et al. (2010) approach the problem under the following assumptions. The effect $Y$ has to be a function $f_X$ of the cause $X$ while some independent noise term $\epsilon_Y \perp\!\!\!\perp X$ is included (compare Section 2.1), *i.e.,* the causal mechanism has the general form $Y = f_X(X, \epsilon_Y)$. They do not restrict the function (it only has to be invertible) or the noise (has to be standard normal, but can be transformed due to the non-restricted $f_X$). Instead, they define priors on the marginal distributions of the cause and the conditional distribution of the effect with the noise being a latent variable. Those priors with low complexity are favored, and hence, they do not only consider the complexity of the mechanism, but also the complexity of the cause distribution. The prior of the cause is realized by a Gaussian mixture model and the mechanism is implemented as a Gaussian process (kernels and parameters have to be set). Moreover, the involved variables are standardized and further regularization is added to account for numerical stability. As typical for the MDL principle, the assumption that the true causal direction yields lower complexities than the anti-causal one is the actual key. To account for this, the authors employ a Bayesian model selection approach involving no explicit independence test. We refer to this approach as *GPI*[11]. The authors evaluate several variants, while the approach described above and that we also use in our experiments is called GPI-MML.

### D.4.2 Results

The results of the GPI approach are shown in Figs. 46 to 52. Overall, we can observe results ranging from $\sim 55\%$ accuracy for the CEC13 data to $\sim 82\%$ on the CE-Gauss and the CE-Net data (see Fig. 46.a), which is roughly in the expected range defined by previous studies. By examining the finer details presented by our synthetic bSCMC data, we can observe that more data helps to achieve notably better results (see, *e.g.,* Fig. 46.b). We can also see that there appears to be no cause or noise distribution particularly standing out (see Fig. 49 or Fig. 50). A reason might be that the Gaussian mixture models are able to represent those distributions and thus are a suitable choice. Further, we can see that simpler functions (*i.e.,* lin_a, add_a, add_b, and add_c, see Fig. 49.a,b,c,d and Fig. 50.c) yield better results than the remaining more complex functions. Thus, it is not suprising that the functional dependence comprises the most influential factor as can be seen in Fig. 47. This might be explained by modeling the functional dependencies with Gaussian processes, which might not be able to represent the functions (using the default parameters). Another supporting observation is the good performance on CE-Gauss that contains data samples from Gaussian processes (see Fig. 46.a). Further, we can observe that the method tends to struggle in the low dependence scheme (*i.e.,* MI $= 0.1$, see Fig. 49 and Fig. 50.b), which might again be a sign of fitting issues. The very diverse results obtained, *e.g.,* for com_a, com_b, or com_c (see Fig. 49.h,i,j and Fig. 50.c) suggest that the method is impacted from every single building block. This claim is further

11. We use the Matlab implementation available at `github.com/ssamot/causality`. Since we use Python to run our experiments, we had to invoke the code from python accordingly. As for parameters, we left everything by default while the data was standardized before applying the method.

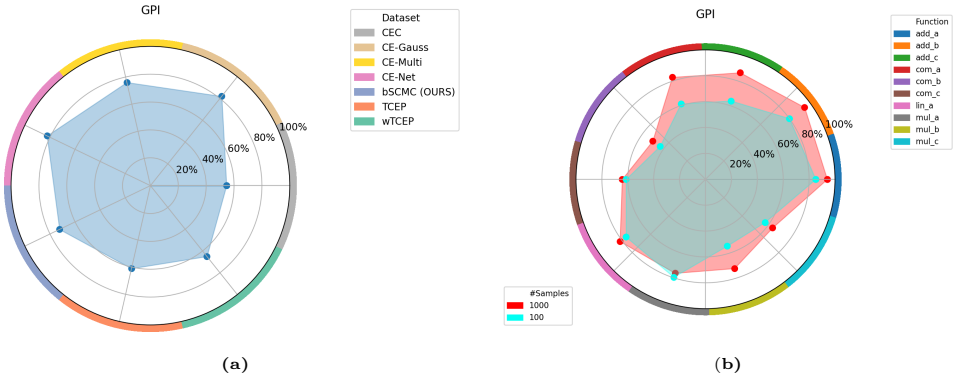



Figure 46: Result footprints for GPI.

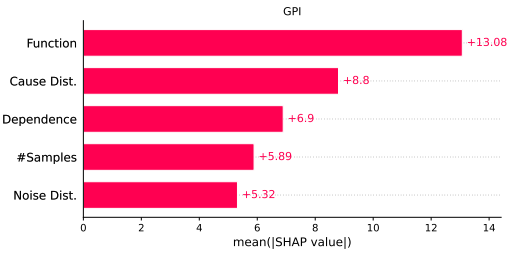



Figure 47: Importance of individual data set characteristics as mean Shapley values for GPI.

| Method | Setting | p-Value |
|---|---|---|
| GPI | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 1.6e-29 |
| GPI | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 5.6e-15 |
| GPI | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 1.3e-16 |
| GPI | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 8.3e-68 |
| GPI | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 3.9e-08 |

Figure 48: Significances at $\alpha = 0.05$-level for conditional independencies of individual data set characteristics *w.r.t.* the obtained accuracy for GPI. Green color coding indicates dependence while red color coding indicates independence.

supported by the significances that are presented in Fig. 52 and Fig. 48. Finally, the results are only very little impacted by invalid decisions in case of our ■ bSCMC data.

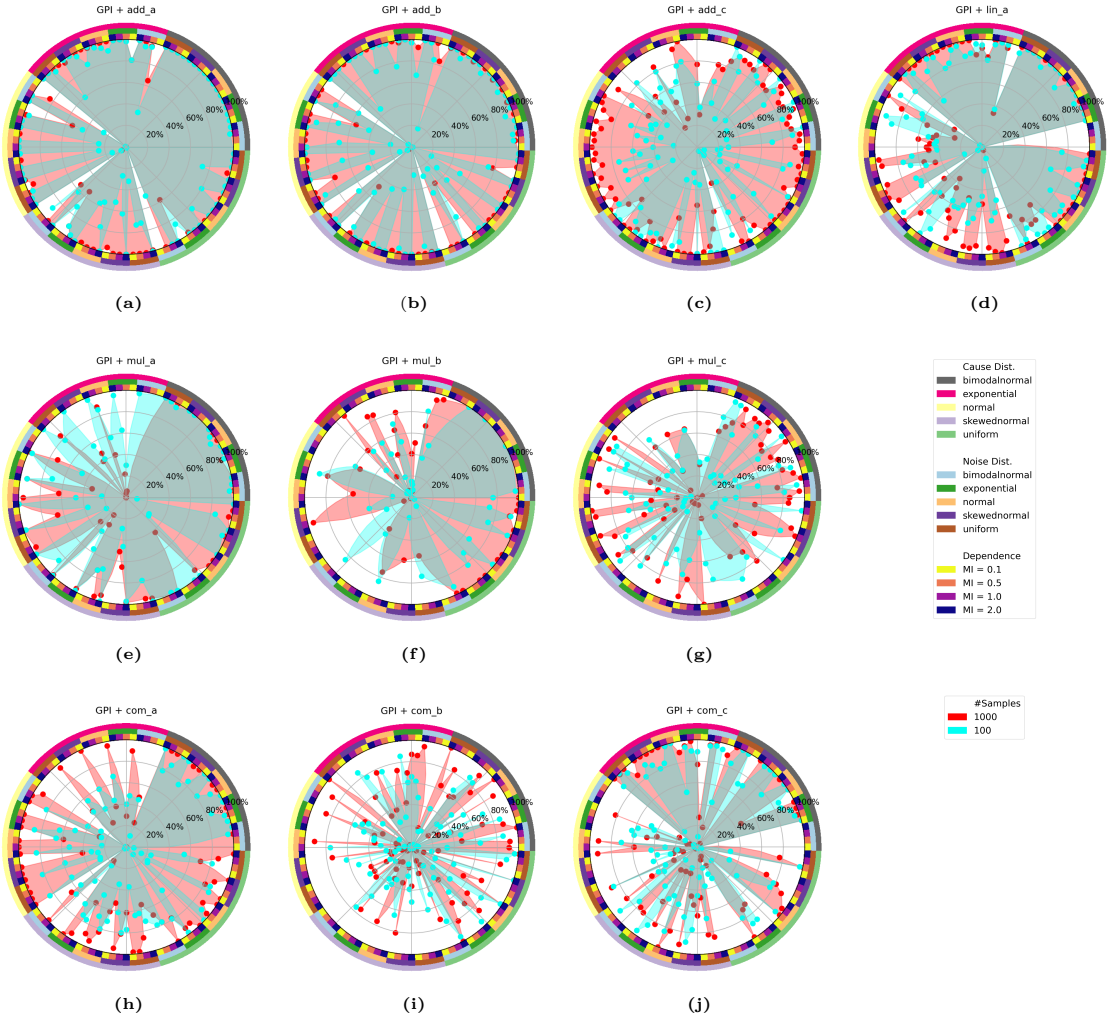



Figure 49: Detailed result overview on our bSCMC data as footprints for GPI.

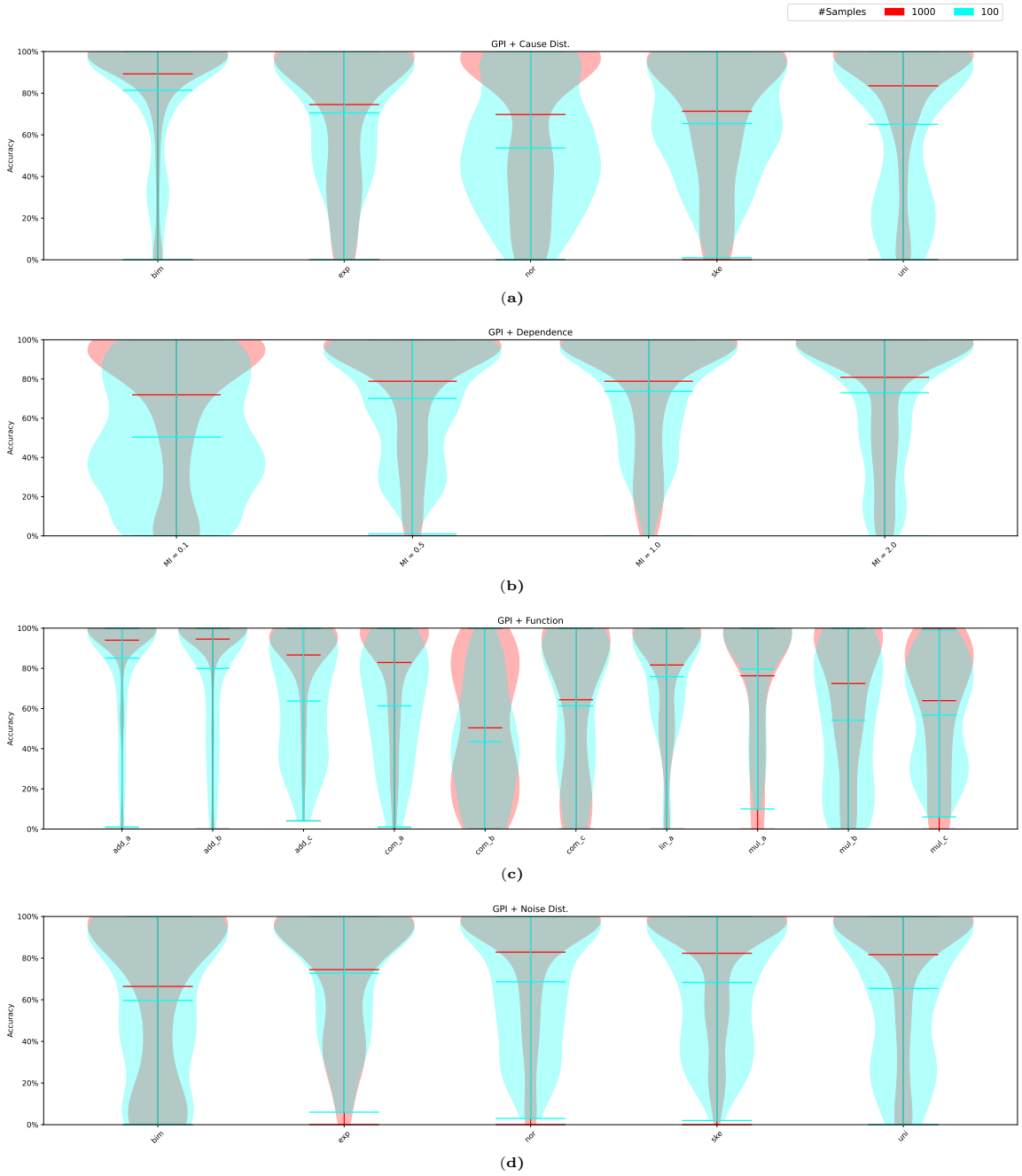



Figure 50: Distribution of accuracies *w.r.t.* individual configurations on our proposed data collection as violins for GPI. Please note, the width of the violins is scaled to unit width individually.

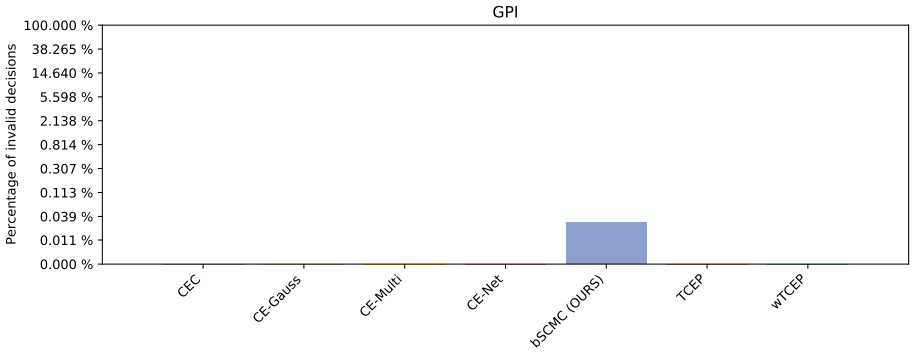



Figure 51: Percentage of invalid decisions for GPI. Note that the y-axis is logarithmically scaled.

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 1.0e-10 | | | |
| (3) normal | 6.9e-23 | 9.4e-05 | | |
| (4) skewednormal | 6.6e-20 | 2.1e-02 | 2.9e-02 | |
| (5) uniform | 4.8e-06 | 9.9e-02 | 2.1e-07 | 7.0e-05 |

**(a)** cause distribution

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 5.7e-03 | | | |
| (3) normal | 3.8e-06 | 2.6e-02 | | |
| (4) skewednormal | 1.9e-05 | 8.8e-02 | 6.2e-01 | |
| (5) uniform | 6.0e-05 | 8.5e-02 | 6.6e-01 | 1.0e+00 |

**(b)** noise distribution

| | (1) | (2) | (3) |
|---|---|---|---|
| (1) MI = 0.1 | | | |
| (2) MI = 0.5 | 1.5e-07 | | |
| (3) MI = 1.0 | 1.8e-11 | 4.4e-02 | |
| (4) MI = 2.0 | 3.2e-15 | 1.9e-04 | 7.2e-02 |

**(c)** mutual information

| | (1) |
|---|---|
| (1) 1000 | |
| (2) 100 | 9.3e-20 |

**(d)** sample size

| | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) |
|---|---|---|---|---|---|---|---|---|---|
| (1) add_a | | | | | | | | | |
| (2) add_b | 5.1e-01 | | | | | | | | |
| (3) add_c | 5.8e-20 | 6.3e-17 | | | | | | | |
| (4) com_a | 5.8e-16 | 2.9e-13 | 8.4e-01 | | | | | | |
| (5) com_b | 2.5e-38 | 1.2e-34 | 4.9e-17 | 2.2e-13 | | | | | |
| (6) com_c | 2.4e-24 | 3.7e-21 | 3.0e-03 | 1.1e-02 | 3.9e-06 | | | | |
| (7) lin_a | 2.0e-11 | 4.2e-09 | 1.5e-02 | 2.3e-02 | 1.1e-21 | 1.5e-06 | | | |
| (8) mul_a | 3.7e-07 | 8.9e-06 | 2.8e-03 | 1.5e-02 | 2.8e-16 | 2.6e-06 | 4.2e-01 | | |
| (9) mul_b | 5.6e-15 | 6.9e-13 | 3.6e-02 | 6.2e-02 | 1.2e-04 | 8.5e-01 | 4.4e-04 | 5.0e-04 | |
| (10) mul_c | 2.3e-31 | 9.0e-28 | 2.0e-07 | 1.8e-05 | 2.3e-04 | 8.8e-02 | 1.2e-11 | 1.6e-10 | 1.5e-01 |

**(f)** functional dependence

Figure 52: Significances at $\alpha = 0.05$-level *w.r.t.* individual configurations for GPI. The numbering of the row and column heads correspond to each other. Green color coding indicates significant differences while red color coding indicates no significant difference.

## D.5 Information Geometric Causal Inference

### D.5.1 Description

The approach from (Daniusis et al., 2012), information geometric causal inference ($IGCI$[12]), relies on the idea that the function $f_X(\cdot)$ that maps from cause to effect (*i.e.,* $Y = f_X(X)$, deterministic or low noise) and the probability density of the cause $p(X)$ are independent. In other words, the marginal distribution $P(X)$ and the conditional distribution $P(Y|X)$ do not contain information about each other. The actual decision for the causal direction can be translated to $C_{X \to Y} < 0$ with $C_{X \to Y} = H(Y) - H(X)$ after some preprocessing. Here, $H(\cdot)$ denotes a short hand for more complex differential entropies that need to be estimated properly. This may be interpreted in such a way that the function $f$ introduces new irregularities, which leads to a lower entropy for the effect $Y$. However, the authors point out that this method requires sufficiently small amounts of noise or even no noise. Since the case of a non-invertible $f$ would be trivial under these conditions, the authors assume $f$ to be a bijective function. A further and more comprehensive review of this approach, including alternative implementations, can, *e.g.,* be found in Mooij et al. (2016).

### D.5.2 Results

The results for IGCI can be found in Figs. 53 to 59. Results obtained on all data collections range from $\sim$ 45% accuracy for the CE-Gauss data to $\sim$ 92% on CE-Multi (see Fig. 53.a) and are roughly within the expected spectrum of accuracy seen in previous studies. Generally, our bSCMC data shows that already 100 samples appear to be sufficient since more data yields only marginal benefits (see, *e.g.,* Fig. 53.b). However, even this small improvement is significant (see Fig. 59.d). Further, we can observe an expected pattern where the method fails in the low dependence scheme with more "noise" (*e.g.,* MI $= 0.1$ on add_a or add_b, see Fig. 56.a,b and also Fig. 57.b). However, we can also observe that this pattern only appears in connection with certain functional dependencies (*i.e.,* not apparent in, *e.g.,* mul_a, see Fig. 56.e). The impact of the function can further be seen in the accuracies achieved in the linear case (*i.e.,* lin_a, see Fig. 56.d and Fig. 57.c), or more complex cases (*e.g.,* com_c, see Fig. 56.j). While this appears to be counter-intuitive at first glance, the reason might be found in the estimation of entropies. Generally, we can observe that a uniform cause distribution leads to better results (*e.g.,* for lin_a, see Fig. 56.d and Fig. 57.a). Thus, the constellation of the building blocks are likely to have a compound effect that is, in some combinations, responsible for the method failing. This is, *e.g.,* apparent in the results obtained for lin_a versus those obtained for mul_a (see Fig. 56.d,e and Fig. 57.c). A supporting observation, especially for the cause distributions, is given by the significances that are shown in Fig. 59. It can be seen that almost all cause distributions lead to significantly different results (see Fig. 59.a). Surprisingly, we can also see that the strength of dependency appears to be among the least impacting factors (see Fig. 59.c and Fig. 54). However, we attribute this to the mentioned compound effects in connection with very diverse results for the other building blocks. Further, we observe that

12. We use the code publicly provided by the authors which is, *e.g.,* contained in `github.com/ssamot/causality`. In detail, we use a uniform reference measure with an entropy estimator for estimating the marginal entropies. The data is normalized between $[0, 1]$ before the method is applied.

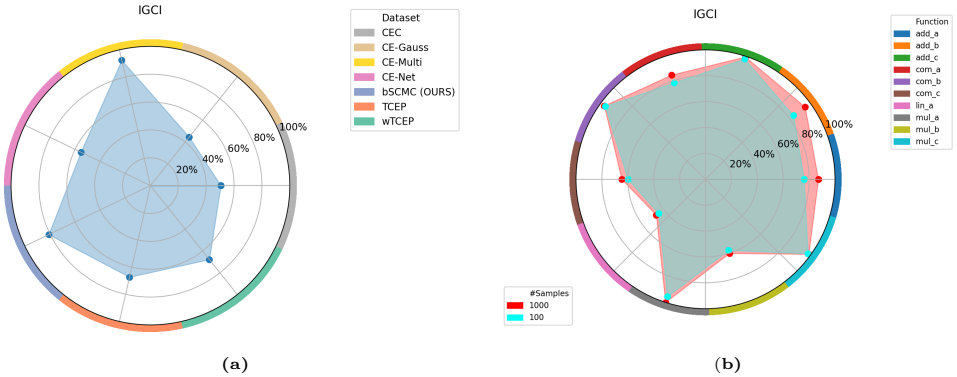



Figure 53: Result footprints for IGCI.

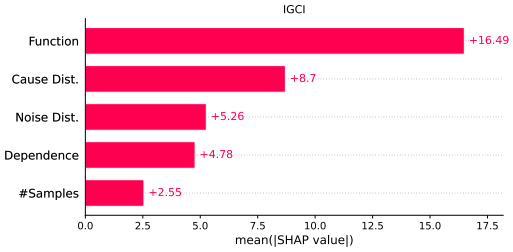



Figure 54: Importance of individual data set characteristics as mean Shapley values for IGCI.

| Method | Setting | p-Value |
|---|---|---|
| IGCI | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 3.5e-58 |
| IGCI | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 3.8e-03 |
| IGCI | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 1.4e-04 |
| IGCI | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 1.6e-156 |
| IGCI | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 1.0e-20 |

Figure 55: Significances at $\alpha = 0.05$-level for conditional independencies of individual data set characteristics *w.r.t.* the obtained accuracy for IGCI. Green color coding indicates dependence while red color coding indicates independence.

all the building blocks have a general impact on the performance (see Fig. 55). Finally, invalid decisions only slightly influence the results *w.r.t.* ■ CEC13 (see Fig. 58).

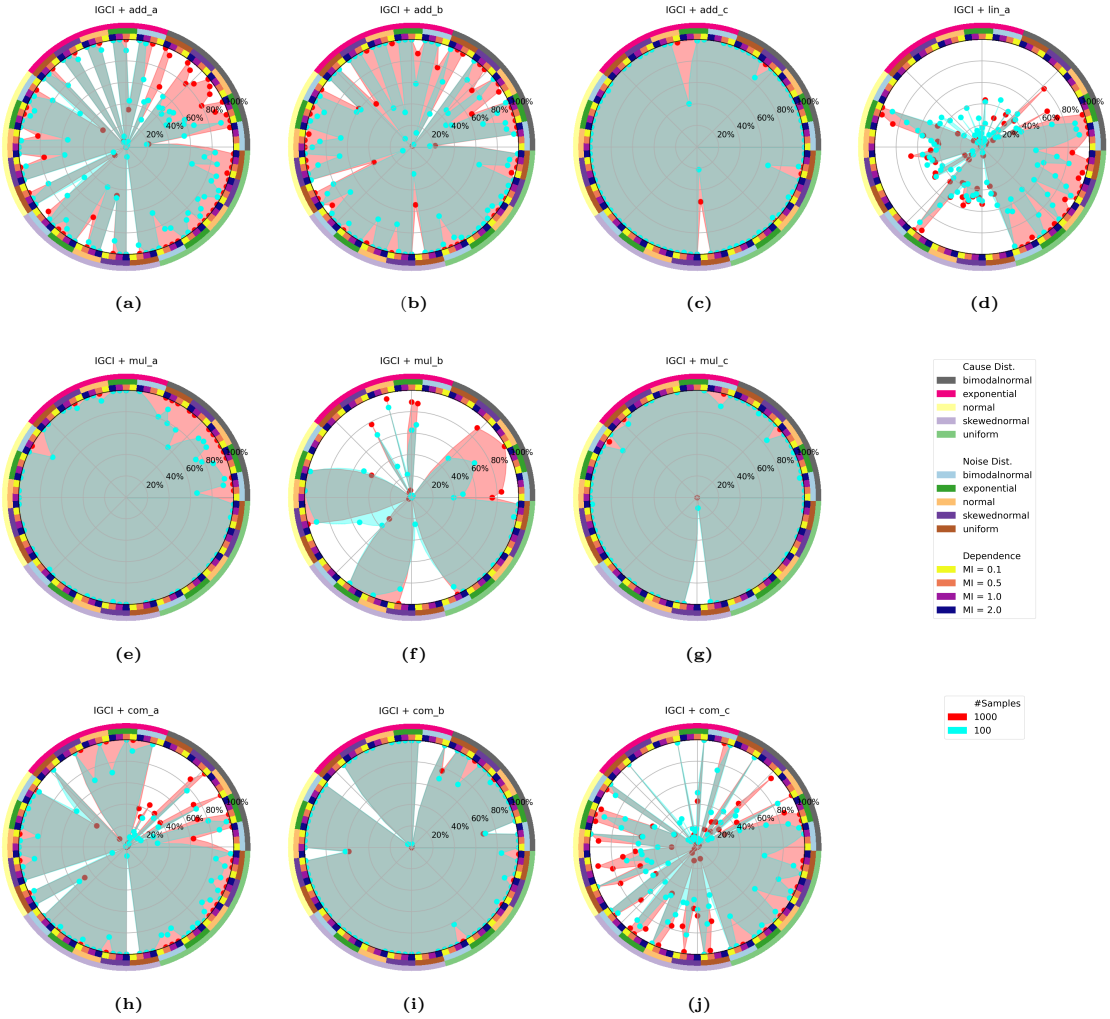



Figure 56: Detailed result overview on our bSCMC data as footprints for IGCI.

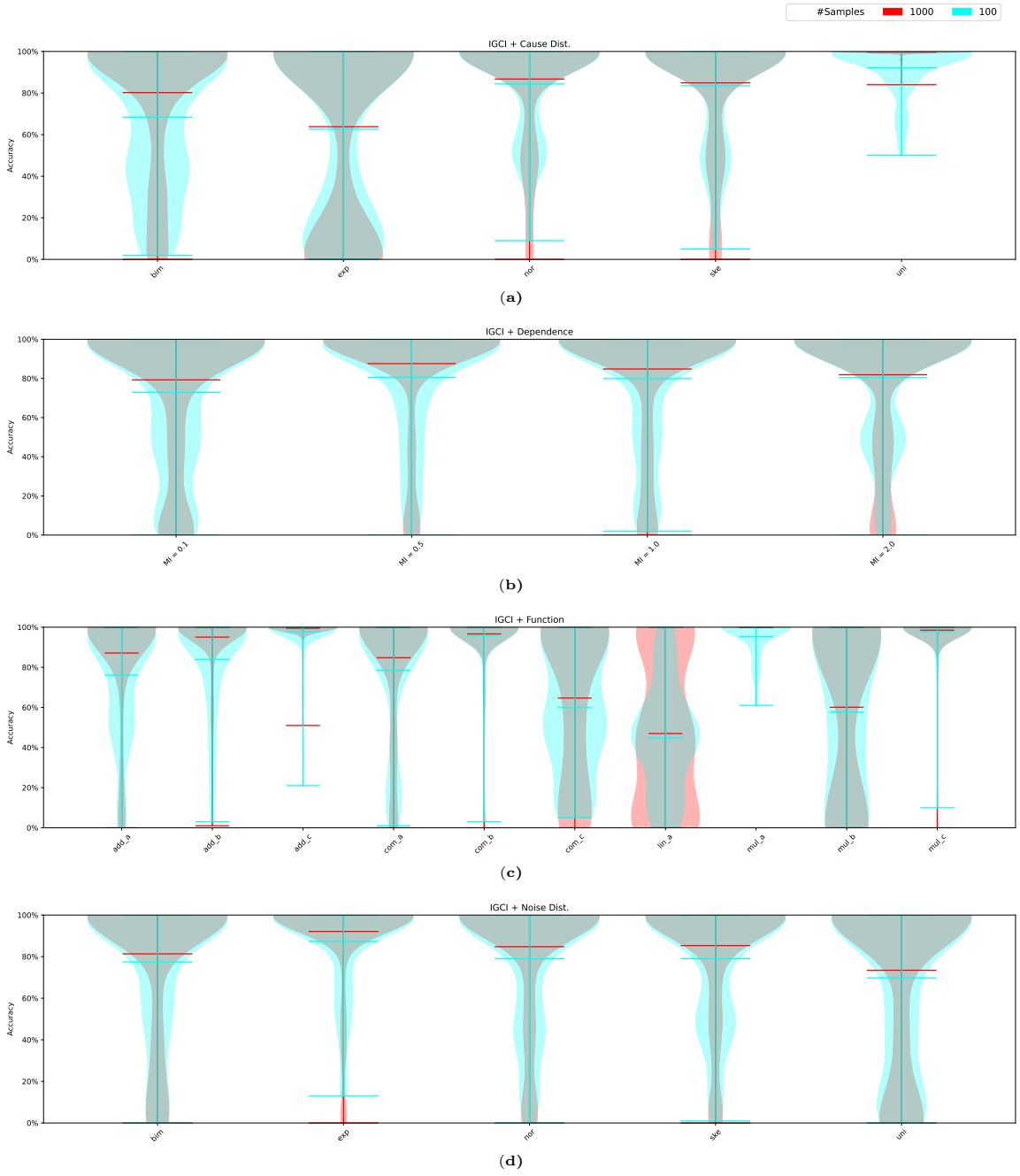



Figure 57: Distribution of accuracies *w.r.t.* individual configurations on our proposed data collection as violins for IGCI. Please note, the width of the violins is scaled to unit width individually.

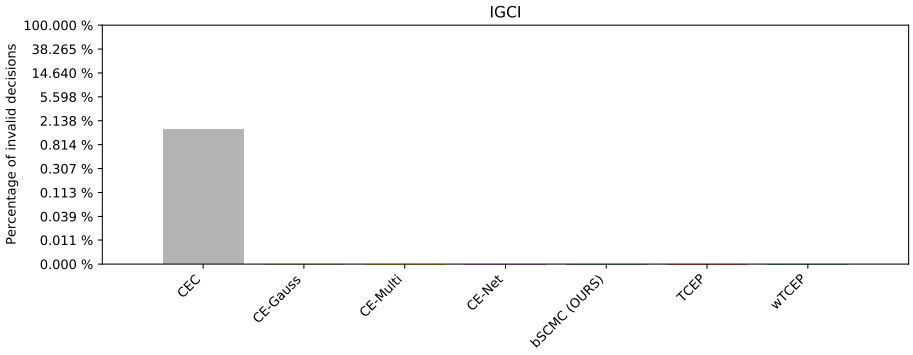



Figure 58: Percentage of invalid decisions for IGCI. Note that the y-axis is logarithmically scaled.

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| **(1)** bimodalnormal | | | | |
| **(2)** exponential | 6.7e-02 | | | |
| **(3)** normal | 1.7e-07 | 6.2e-09 | | |
| **(4)** skewednormal | 1.7e-06 | 4.9e-08 | 7.6e-01 | |
| **(5)** uniform | 7.9e-17 | 8.0e-14 | 2.9e-03 | 2.7e-03 |

**(a)** cause distribution

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| **(1)** bimodalnormal | | | | |
| **(2)** exponential | 5.5e-04 | | | |
| **(3)** normal | 5.2e-01 | 2.9e-03 | | |
| **(4)** skewednormal | 4.3e-01 | 3.9e-03 | 8.8e-01 | |
| **(5)** uniform | 6.2e-02 | 1.8e-07 | 8.3e-03 | 5.6e-03 |

**(b)** noise distribution

| | (1) | (2) | (3) |
|---|---|---|---|
| **(1)** MI = 0.1 | | | |
| **(2)** MI = 0.5 | 4.4e-02 | | |
| **(3)** MI = 1.0 | 1.6e-03 | 1.8e-01 | |
| **(4)** MI = 2.0 | 6.7e-03 | 3.4e-01 | 6.9e-01 |

**(c)** mutual information

| | (1) |
|---|---|
| **(1)** 1000 | |
| **(2)** 100 | 8.9e-18 |

**(d)** sample size

| | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) |
|---|---|---|---|---|---|---|---|---|---|
| **(1)** add_a | | | | | | | | | |
| **(2)** add_b | 2.1e-03 | | | | | | | | |
| **(3)** add_c | 2.5e-22 | 6.9e-14 | | | | | | | |
| **(4)** com_a | 1.1e-01 | 2.6e-01 | 2.0e-14 | | | | | | |
| **(5)** com_b | 2.0e-19 | 4.9e-12 | 9.7e-01 | 1.1e-12 | | | | | |
| **(6)** com_c | 6.5e-08 | 3.2e-16 | 4.2e-38 | 1.7e-09 | 6.1e-34 | | | | |
| **(7)** lin_a | 8.4e-24 | 1.6e-35 | 8.9e-55 | 3.0e-22 | 1.5e-49 | 2.1e-06 | | | |
| **(8)** mul_a | 3.3e-12 | 1.7e-06 | 4.7e-02 | 2.2e-07 | 7.2e-02 | 2.5e-25 | 5.9e-40 | | |
| **(9)** mul_b | 1.7e-04 | 9.5e-09 | 2.4e-25 | 3.8e-06 | 2.7e-23 | 5.8e-01 | 1.6e-02 | 3.6e-16 | |
| **(10)** mul_c | 5.3e-18 | 2.8e-11 | 9.7e-01 | 9.2e-12 | 9.5e-01 | 9.5e-31 | 2.8e-44 | 5.9e-02 | 5.0e-21 |

**(f)** functional dependence

Figure 59: Significances at $\alpha = 0.05$-level *w.r.t.* individual configurations for IGCI. The numbering of the row and column heads correspond to each other. Green color coding indicates significant differences while red color coding indicates no significant difference.

## D.6 Conditional Distribution Variability

### D.6.1 Description

Deciding for the causal direction by utilizing several features constructed from the data is proposed in (Fonollosa, 2016). The list of derived features is diverse while the data is preprocessed accordingly (*e.g.,* discretization of real-valued variables or normalizing to zero mean and unit variance). It ranges from several information-theoretic measures (*e.g.,* entropy and mutual information) over more basic features of the distributions of the variables (*e.g.,* skewness or kurtosis) to the incorporation of external approaches (*e.g.,* HSIC or IGCI). Gradient boosting classifiers (Hastie et al., 2008) are trained utilizing the features to decide among the three cases: $X \to Y$, $X \leftarrow Y$, and $X \perp\!\!\!\perp Y$. In detail, they combine the following three approaches using equal weights to obtain the final decision: (1) A single model for deciding among the three cases, (2) two separate models, one for the direction (*i.e.,* $X \to Y$ versus $X \leftarrow Y$) and one for dependence (*i.e.,* $X \perp\!\!\!\perp Y$ versus $X \not\perp\!\!\!\perp Y$), and (3) two separate models, one for each direction (*i.e.,* $X \to Y$ versus rest and $X \leftarrow Y$ versus rest). This whole pipeline is referred to as *Jarfo*[13].

### D.6.2 Results

Results obtained can be found in Figs. 60 to 66. Generally, the method achieves results between $\sim 31\%$ (■ TCEP) and $\sim 63\%$ (■ CE-Net and ■ CE-Multi). It should be mentioned that we achieve only $\sim 45\%$ accuracy on the ■ CEC13 data, which is lower than the previous reported results (see Fig. 60.a). Please note that we use only the part of the CEC13 data suitable for our purpose (*i.e.,* leaving out discrete or independent data). Thus, the obtained results might differ already. However, from the results for our synthetic ■ bSCMC data, we can see that there is no clear preference of the method *w.r.t.* single characteristics (see Fig. 63 and Fig. 64). A reason might be that the classifiers rely on a wide range of features, which come with their own diverse underlying assumptions. This might also explain that the importance of the individual building blocks is relatively close to each other (see Fig. 62). We can further see that larger sample sizes help to improve the results (*i.e.,* ■ 1 000 samples versus ■ 100 samples). Although we can find cases where fewer data works better. Those cases mostly appear when results obtained on ■ 1 000 samples lead to systematic errors (*i.e.,* mostly predicting the wrong direction, see, *e.g.,* Fig. 63.b,c,d or Fig. 64.c). A reason might again be the wide range of incorporated features. These observations are also apparent in the obtained significances presented in Fig. 66 while all the individual building blocks are relevant (see Fig. 62). Finally, the training data might be of a different domain than the data we sample for our ■ bSCMC collection (or, more likely, does only match a part of it). However, it should be noted that the algorithm does produce a high number of invalid decisions (see Fig. 65) that, of course, lead to very poor results. In summary it can be said that we were not able to reproduce the expected results for Jarfo. A reason might be that the implementation struggles with some technical issues. We show those results anyway

13. We use the causal discovery toolbox (Kalainathan and Goudet, 2019) to realize this method which internally uses the author's original code. We left all parameters as defined by the toolbox, while the data was standardized before applying the method. To train the Jarfo model, we used the training part of the CEC13 data (*i.e.,* the final train data as well as SUP1 and SUP2). During our experiments, we observed the code throwing several math warnings which might distort the results of the method.

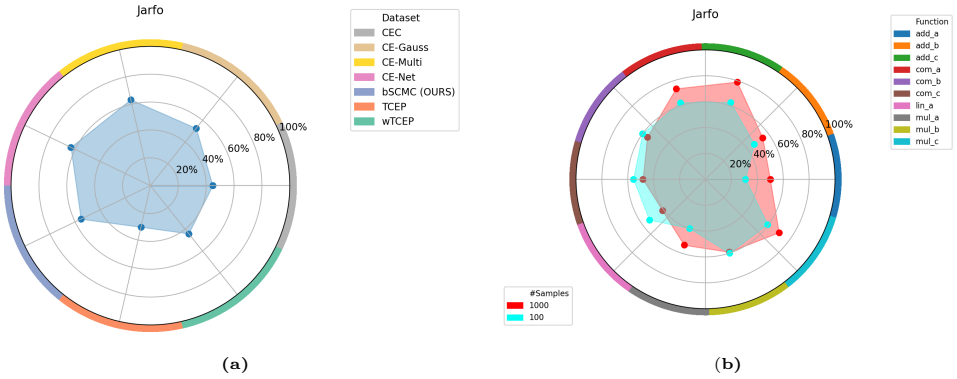



Figure 60: Result footprints for Jarfo.

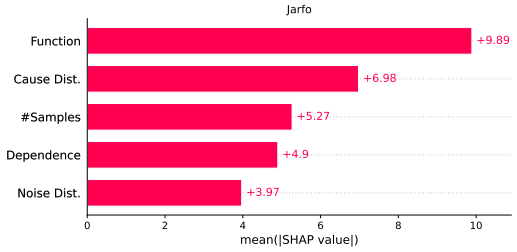



Figure 61: Importance of individual data set characteristics as mean Shapley values for Jarfo.

| Method | Setting | p-Value |
|---|---|---|
| Jarfo | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 3.9e-14 |
| Jarfo | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 1.1e-06 |
| Jarfo | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 5.7e-04 |
| Jarfo | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 8.5e-53 |
| Jarfo | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 3.2e-10 |

Figure 62: Significances at $\alpha = 0.05$-level for conditional independencies of individual data set characteristics *w.r.t.* the obtained accuracy for Jarfo. Green color coding indicates dependence while red color coding indicates independence.

since a major insight we want to provide with our manuscript is the performance of the methods when the code is applied "as is".

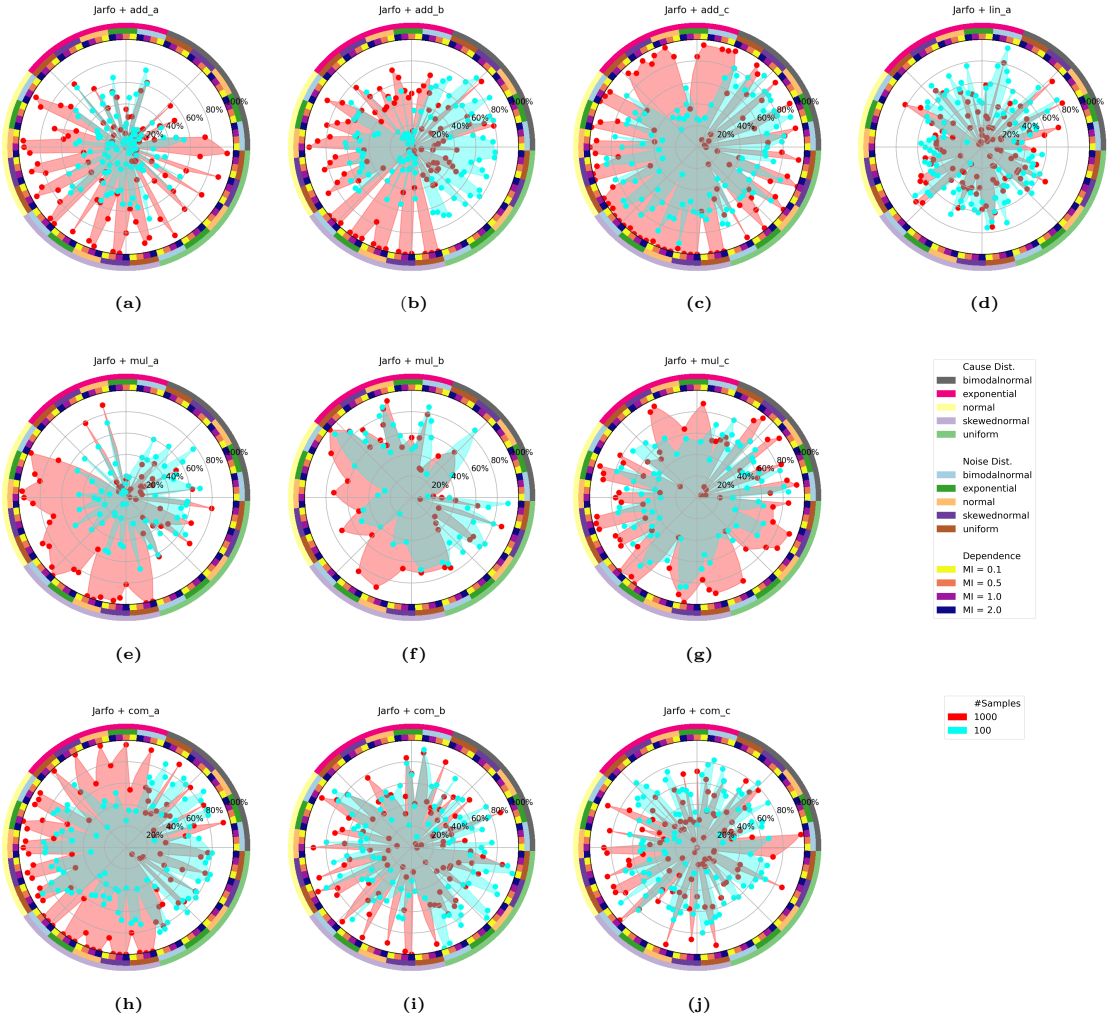



Figure 63: Detailed result overview on our bSCMC data as footprints for Jarfo.

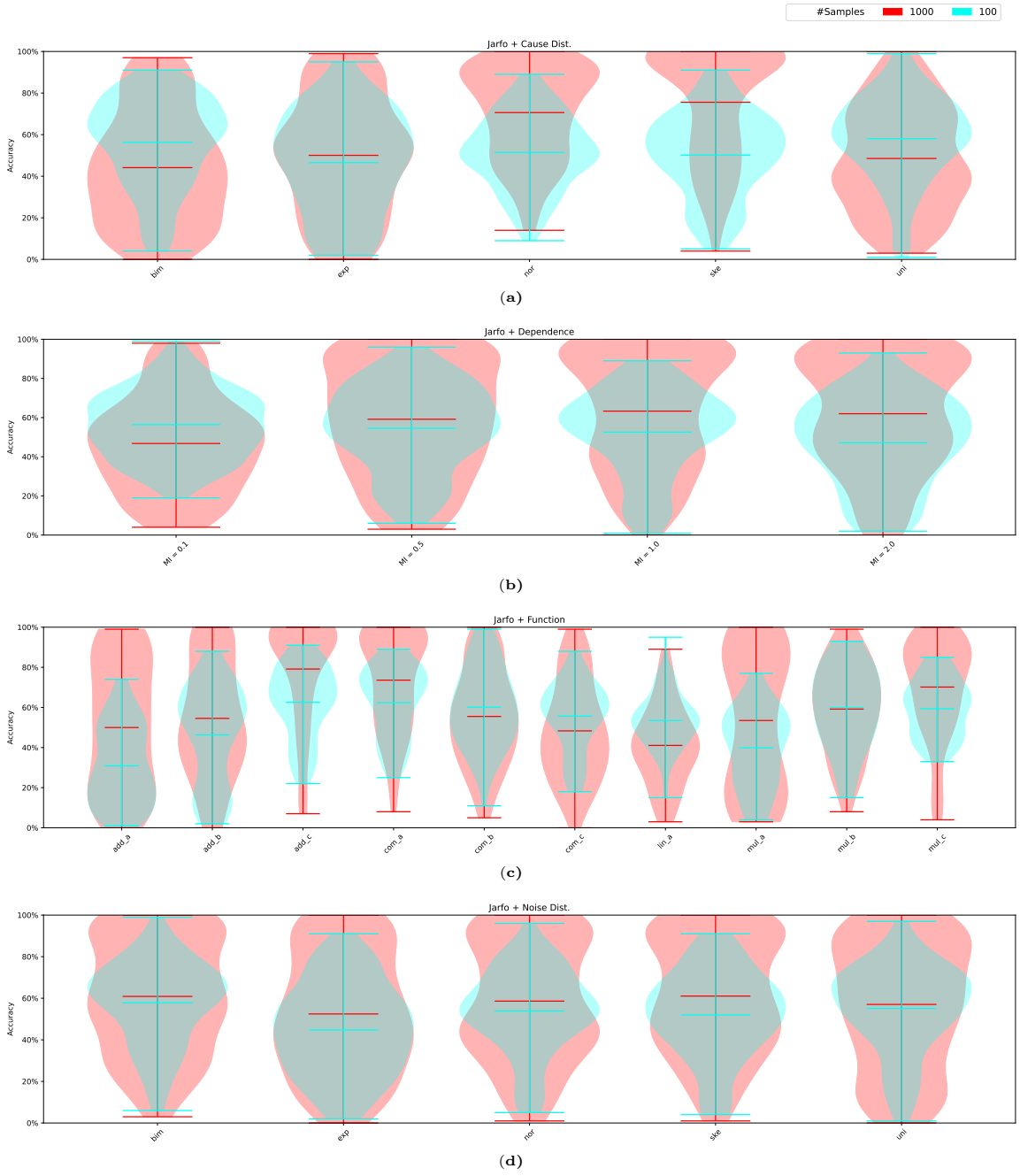



Figure 64: Distribution of accuracies *w.r.t.* individual configurations on our proposed data collection as violins for Jarfo. Please note, the width of the violins is scaled to unit width individually.

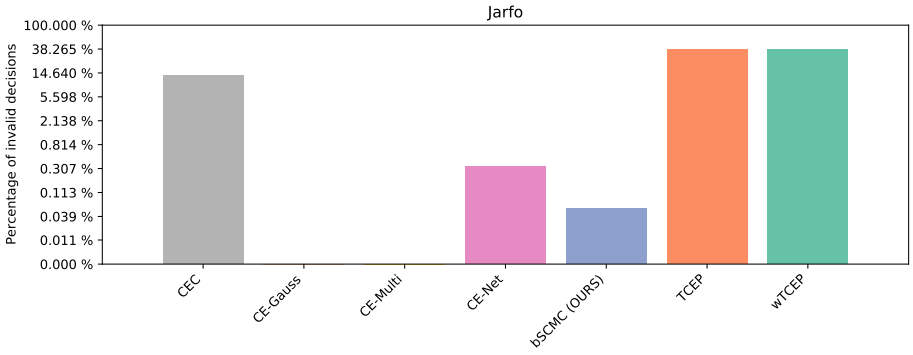



Figure 65: Percentage of invalid decisions for Jarfo. Note that the y-axis is logarithmically scaled.

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 3.0e-01 | | | |
| (3) normal | 1.3e-07 | 5.2e-09 | | |
| (4) skewednormal | 7.7e-09 | 2.2e-10 | 1.8e-01 | |
| (5) uniform | 1.4e-01 | 1.2e-02 | 1.2e-04 | 1.2e-05 |

**(a)** cause distribution

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 8.0e-08 | | | |
| (3) normal | 8.3e-02 | 7.0e-05 | | |
| (4) skewednormal | 1.2e-01 | 5.2e-05 | 9.2e-01 | |
| (5) uniform | 1.9e-01 | 7.5e-05 | 7.0e-01 | 7.9e-01 |

**(b)** noise distribution

| | (1) | (2) | (3) |
|---|---|---|---|
| (1) MI = 0.1 | | | |
| (2) MI = 0.5 | 8.9e-04 | | |
| (3) MI = 1.0 | 1.0e-04 | 5.4e-01 | |
| (4) MI = 2.0 | 6.2e-02 | 2.2e-01 | 6.6e-02 |

**(c)** mutual information

| | (1) |
|---|---|
| (1) 1000 | |
| (2) 100 | 4.9e-05 |

**(d)** sample size

| | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) |
|---|---|---|---|---|---|---|---|---|---|
| (1) add_a | | | | | | | | | |
| (2) add_b | 1.7e-04 | | | | | | | | |
| (3) add_c | 7.7e-23 | 4.4e-13 | | | | | | | |
| (4) com_a | 8.5e-20 | 5.7e-10 | 1.3e-01 | | | | | | |
| (5) com_b | 1.8e-10 | 1.4e-02 | 3.0e-08 | 2.3e-05 | | | | | |
| (6) com_c | 1.6e-06 | 5.7e-01 | 2.6e-14 | 8.4e-11 | 2.4e-02 | | | | |
| (7) lin_a | 4.9e-04 | 1.4e-01 | 7.4e-22 | 7.8e-18 | 5.5e-06 | 1.9e-02 | | | |
| (8) mul_a | 5.3e-02 | 1.7e-01 | 1.4e-12 | 2.2e-10 | 2.8e-04 | 3.6e-02 | 4.7e-01 | | |
| (9) mul_b | 2.5e-08 | 6.7e-03 | 6.5e-05 | 4.3e-03 | 4.5e-01 | 6.6e-03 | 3.8e-06 | 3.9e-04 | |
| (10) mul_c | 4.1e-15 | 1.2e-06 | 3.6e-03 | 1.8e-01 | 3.4e-03 | 2.9e-07 | 4.1e-14 | 8.0e-08 | 8.5e-02 |

**(f)** functional dependence

Figure 66: Significances at $\alpha = 0.05$-level *w.r.t.* individual configurations for Jarfo. The numbering of the row and column heads correspond to each other. Green color coding indicates significant differences while red color coding indicates no significant difference.

## D.7 Linear non-Gaussian Acyclic Model

### D.7.1 Description

The authors of (Shimizu et al., 2006) propose an approach that relies on a linear data generation process with non-Gaussian noise. The authors refer to their method as the linear non-Gaussian acyclic model (*LiNGAM*[14]). More precisely, the model is $\mathbf{x} = \mathbf{B}\mathbf{x} + \mathbf{e}$ with $\mathbf{x}$ denoting the continuous random variables with zero mean and no confounding, $\mathbf{B}$ depicting their connections among each other, and $\mathbf{e}$ being a noise vector. Hence, a single variable is composed as $x_i = \sum_{k(j)<k(i)} b_{ij} x_j + \epsilon_i$ with $\epsilon_i$ being the non-Gaussian noise term. Here, $k(\cdot)$ denotes the causal order between variables, which ensures that no later variable causes an earlier variable, *i.e.,* there is no cycle in the corresponding graph. Thus, $\mathbf{B}$ has to be a lower triangular matrix with zeros on its diagonal. Solving for $\mathbf{x}$ leads to $\mathbf{x} = (\mathbf{I} - \mathbf{B})^{-1}\,\mathbf{e}$, which defines, given the non-Gaussianity of $\mathbf{e}$, the standard linear independent component analysis (Comon, 1994). The original work of Shimizu et al. (2006) approaches this problem using the ICA algorithm. However, as the authors detail in their follow-up work (Shimizu et al., 2011), the ICA version yields some potential problems such as convergence issues or the algorithm being not scale-invariant. Thus, they propose a direct approach where the effect of exogenous variables are removed from other variables utilizing independence tests and regressions. The authors show that the LiNGAM model also holds for the residuals. We use this direct approach in our experiments. Further, while the derivation is outlined for multiple variables, we operate in a bivariate setting, which implies some simplifications.

### D.7.2 Results

Results obtained for LiNGAM can be found in Figs. 67 to 73. Overall, the method achieved a range from $\sim$ 29% on CE-Net data to $\sim$ 79% on CE-Gauss data (see Fig. 67.a). A reason for this wide spectrum of results can be found through the more detailed results delivered by our bSCMC data. Generally, we can observe that LiNGAM works best by far on data generated with lin_a, which is not surprising given the theoretical foundation of the approach (see Fig. 67.b and Fig. 71.c). Thus, it is not surprising to see that the functional dependence is the most influential building block (see Fig. 68). By zooming further in, we can also see that the method achieves $\sim$ 50% on data generated with normal cause and normal noise distributions in connection with the lin_a function (see Fig. 70.d). This can also be explained by the theoretical foundation. We can also see that the method benefits from more data, which might be a natural result from more robust regressions. This improvement is also significant (see Fig. 73.d). However, it can also be seen that the method fails for bimodal normal noise distributions in the linear case (see Fig. 70.d). Also, we can observe good performance for particular cases that are actually not covered by the model (*e.g.,* bimodal normal cause and skewed normal noise distribution on data generated using the mul_a function, see Fig. 70.e). However, we can see a wide range of cases where the model fails and achieves very poor results (compare Fig. 71). This can be explained by considering the number of invalid decisions shown in Fig. 72. Here it can be seen that LiNGAM produces a high number of those on each data set. This is due to

14. We use the original code provided by the authors that is available at `github.com/cdt15/lingam` to conduct our experiments. In more detail, we apply the direct version of LiNGAM including default parameters with preceding centering of the data (*i.e.,* setting the data mean to zero).

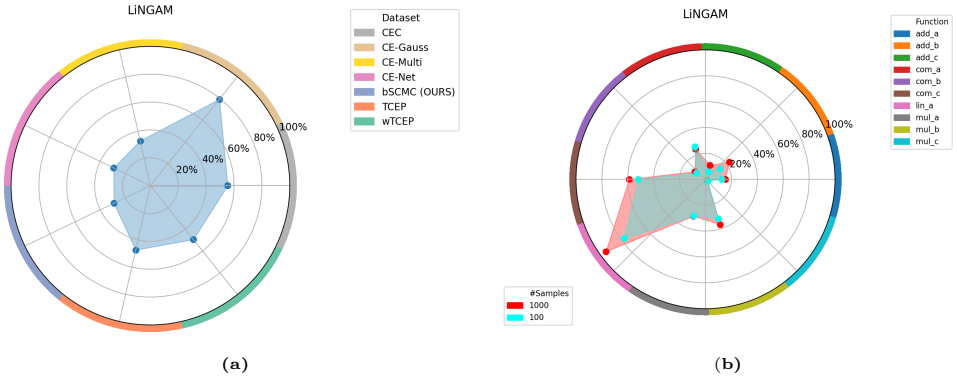



Figure 67: Result footprints for LiNGAM.

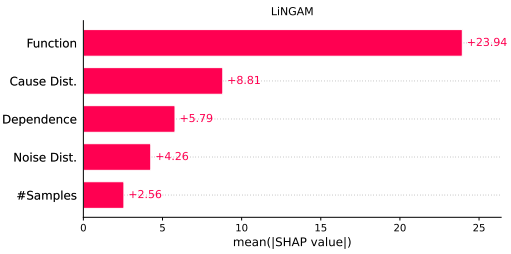



Figure 68: Importance of individual data set characteristics as mean Shapley values for LiNGAM.

| Method | Setting | p-Value |
|---|---|---|
| LiNGAM | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 2.2e-15 |
| LiNGAM | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 8.6e-05 |
| LiNGAM | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 3.1e-07 |
| LiNGAM | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 3.1e-201 |
| LiNGAM | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 2.2e-02 |

Figure 69: Significances at $\alpha = 0.05$-level for conditional independencies of individual data set characteristics *w.r.t.* the obtained accuracy for LiNGAM. Green color coding indicates dependence while red color coding indicates independence.

the fact that the method tends to output independence if the underlying assumptions are violated. Finally, we ca see that each of the building blocks does have a significant impact on the performance (see Fig. 73 and Fig. 69).

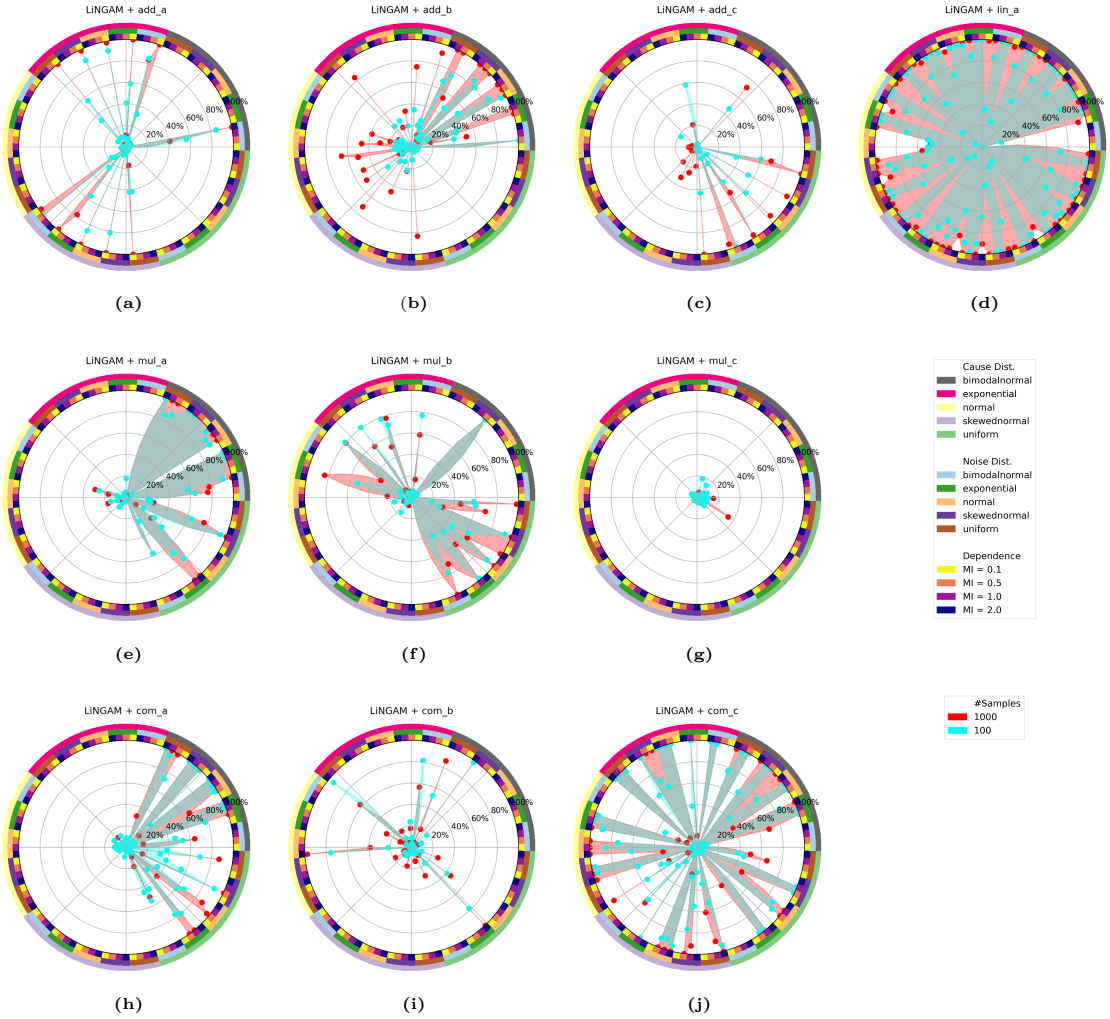



Figure 70: Detailed result overview on our bSCMC data as footprints for LiNGAM.

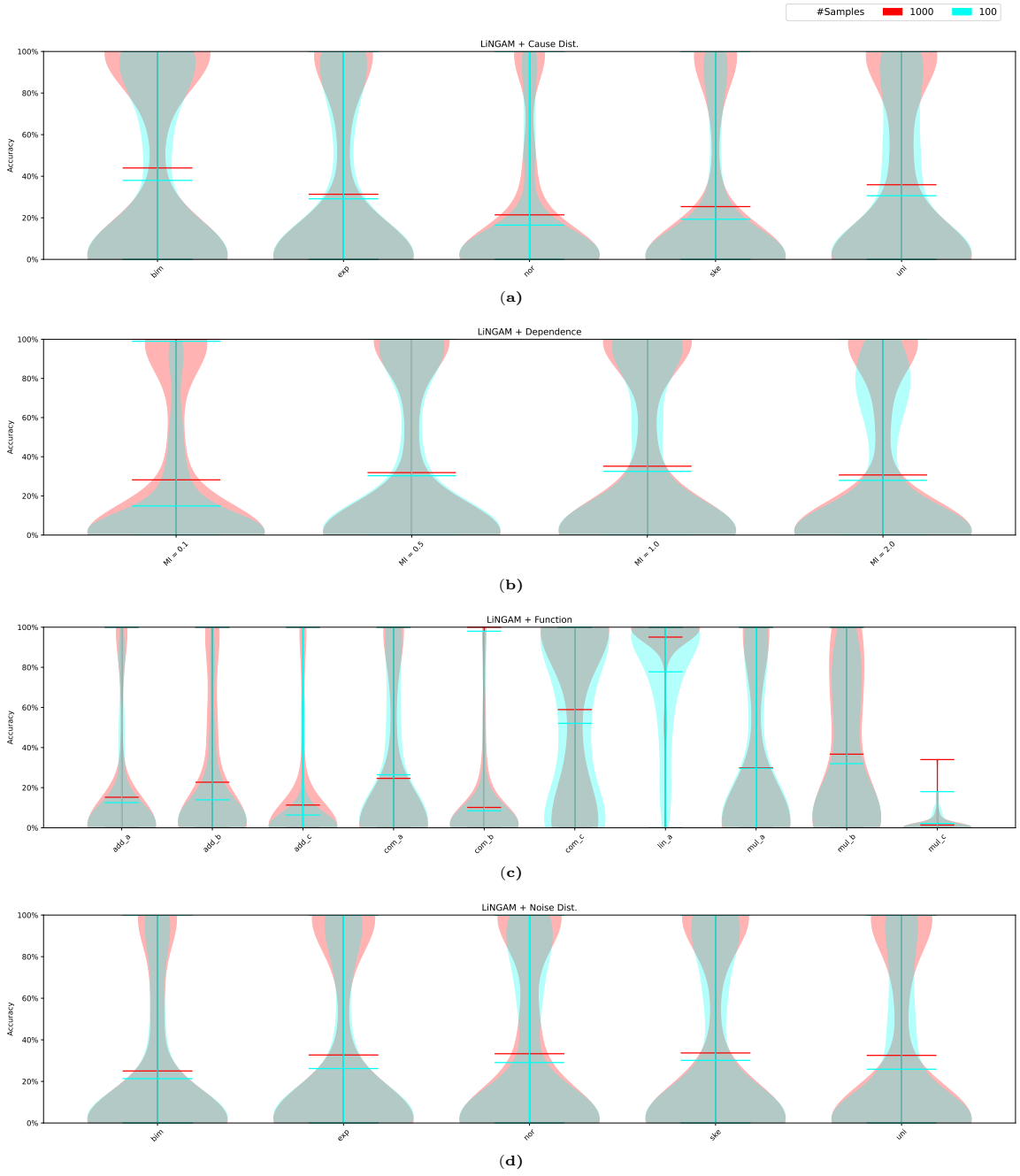



Figure 71: Distribution of accuracies *w.r.t.* individual configurations on our proposed data collection as violins for LiNGAM. Please note, the width of the violins is scaled to unit width individually.

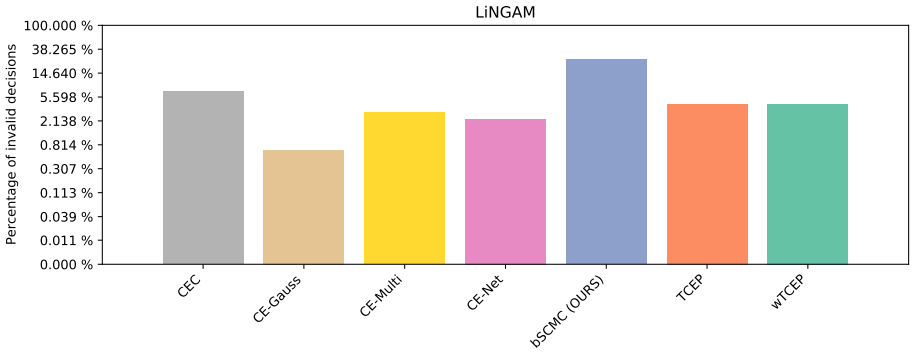



Figure 72: Percentage of invalid decisions for LiNGAM. Note that the y-axis is logarithmically scaled.

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| **(1)** bimodalnormal | | | | |
| **(2)** exponential | 3.5e-04 | | | |
| **(3)** normal | 8.5e-08 | 1.6e-01 | | |
| **(4)** skewednormal | 7.2e-09 | 3.9e-02 | 3.1e-01 | |
| **(5)** uniform | 1.2e-01 | 1.6e-02 | 7.1e-06 | 7.8e-07 |

**(a)** cause distribution

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| **(1)** bimodalnormal | | | | |
| **(2)** exponential | 2.5e-02 | | | |
| **(3)** normal | 2.4e-02 | 9.4e-01 | | |
| **(4)** skewednormal | 5.6e-03 | 6.3e-01 | 5.5e-01 | |
| **(5)** uniform | 1.3e-01 | 4.3e-01 | 4.6e-01 | 2.0e-01 |

**(b)** noise distribution

| | (1) | (2) | (3) |
|---|---|---|---|
| **(1)** MI = 0.1 | | | |
| **(2)** MI = 0.5 | 6.8e-04 | | |
| **(3)** MI = 1.0 | 6.0e-06 | 2.7e-01 | |
| **(4)** MI = 2.0 | 2.8e-03 | 5.5e-01 | 8.3e-02 |

**(c)** mutual information

| | (1) |
|---|---|
| **(1)** 1000 | |
| **(2)** 100 | 6.2e-01 |

**(d)** sample size

| | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) |
|---|---|---|---|---|---|---|---|---|---|
| **(1)** add_a | | | | | | | | | |
| **(2)** add_b | 9.6e-04 | | | | | | | | |
| **(3)** add_c | 1.8e-06 | 4.2e-12 | | | | | | | |
| **(4)** com_a | 1.7e-02 | 9.7e-01 | 1.7e-09 | | | | | | |
| **(5)** com_b | 5.4e-01 | 1.0e-04 | 1.6e-04 | 1.7e-03 | | | | | |
| **(6)** com_c | 2.8e-18 | 2.4e-13 | 2.5e-28 | 3.8e-10 | 3.3e-21 | | | | |
| **(7)** lin_a | 8.3e-50 | 7.4e-48 | 9.0e-57 | 1.4e-35 | 1.5e-54 | 3.5e-10 | | | |
| **(8)** mul_a | 1.4e-03 | 2.7e-01 | 1.4e-10 | 3.4e-01 | 1.0e-04 | 1.2e-06 | 4.1e-26 | | |
| **(9)** mul_b | 1.8e-11 | 2.3e-05 | 6.1e-21 | 3.1e-04 | 6.3e-13 | 1.8e-03 | 1.0e-23 | 1.7e-02 | |
| **(10)** mul_c | 9.3e-06 | 3.9e-13 | 8.5e-01 | 1.5e-09 | 1.3e-04 | 2.5e-27 | 6.2e-54 | 1.0e-10 | 1.8e-22 |

**(f)** functional dependence

Figure 73: Significances at $\alpha = 0.05$-level *w.r.t.* individual configurations for LiNGAM. The numbering of the row and column heads correspond to each other. Green color coding indicates significant differences while red color coding indicates no significant difference.

## D.8 Non-linear Maximum Entropy Approximation

### D.8.1 Description

Hyvärinen and Smith (2013) present a method to orient the direction of the causal link between two or more non-Gaussian random variables. They first approach this under the framework of LiNGAM, *i.e.,* they assume the data generation process to follow a linear non-Gaussian acyclic model. Hyvärinen (2010) show how to compute the likelihood under LiNGAM given observational data whose ratio can be used to decide for causal directions. Since the modeling of the involved densities as well as the influence of outliers might be troublesome in practical applications, the authors propose a maximum entropy approximation of the likelihood ratio. They further extend this approach to consider the non-linear case using maximum entropy approximation (*NLME*[15]).

### D.8.2 Results

Results obtained for this method can be found in Figs. 74 to 80. Please note, the results mostly match those obtained for LiNGAM in Appendix D.7.2, which is not surprising given the close connection between both methods. Thus, we consider the same explanations given in Appendix D.7.2 can also be applied here while we are able to see minor improvements gained from the slightly different approach of NLME in comparison to LiNGAM. In more detail, we see the results range from $\sim$ 31% on CE-Net data to $\sim$ 79% on CE-Gauss data (see Fig. 74.a). As comparison, LiNGAM achieved $\sim$ 29% and $\sim$ 79% on the same data (see Fig. 67.a). Further, we can observe some improvements, *e.g.,* in the small sample regime (*e.g.,* for lin_a, see Fig. 77.d versus Fig. 70.d), or some minor general improvements *w.r.t.* applicability to further functional dependencies (*e.g.,* for mul_b, see Fig. 77.f versus Fig. 70.f, or for com_c, see Fig. 77.j versus Fig. 70.j, also compare Fig. 78 and Fig. 71). We attribute this to be the direct result of the conceptual improvements that NLME contains in comparison to LiNGAM. This is also visible considering the number of invalid decisions (compare Fig. 79 versus Fig. 72). Here we can observe a much lower count of invalid decisions. The method does not predict independence that often (which might also be caused partly by a different implementation). Considering the significances presented in Fig. 80 and Fig. 76, we can again see that not every building block influences the results of NLME which the noise distribution and the sample size being not significant. Finally, as for LiNGAM, the functional dependence is the most influential building block (see Fig. 75), while the noise distribution and the sample size are the least important characteristics.

15. We rely on the original code published by the authors at `www.cs.helsinki.fi/u/ahyvarin/code/pwcausal` using the general entropy-based method to obtain our results. The data is standardized before the method is applied.

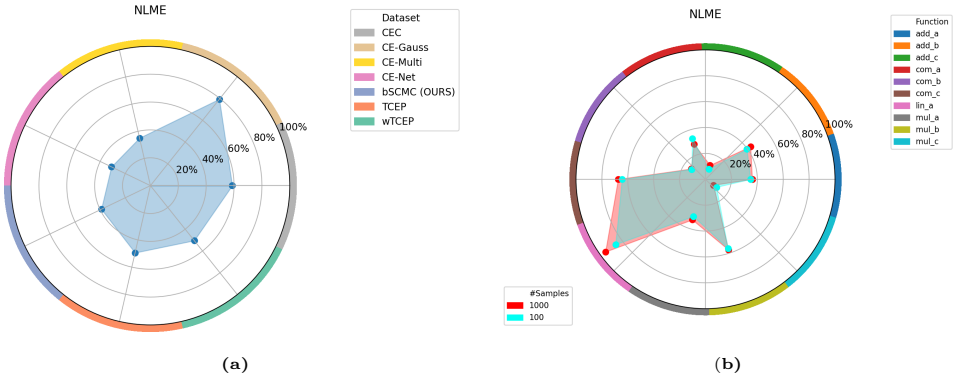



Figure 74: Result footprints for NLME.

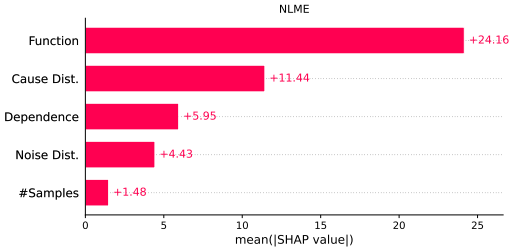



Figure 75: Importance of individual data set characteristics as mean Shapley values for NLME.

| Method | Setting | p-Value |
|---|---|---|
| NLME | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 3.0e-47 |
| NLME | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 1.0e+00 |
| NLME | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 1.2e-04 |
| NLME | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 9.8e-223 |
| NLME | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 2.2e-01 |

Figure 76: Significances at $\alpha = 0.05$-level for conditional independencies of individual data set characteristics *w.r.t.* the obtained accuracy for NLME. Green color coding indicates dependence while red color coding indicates independence.

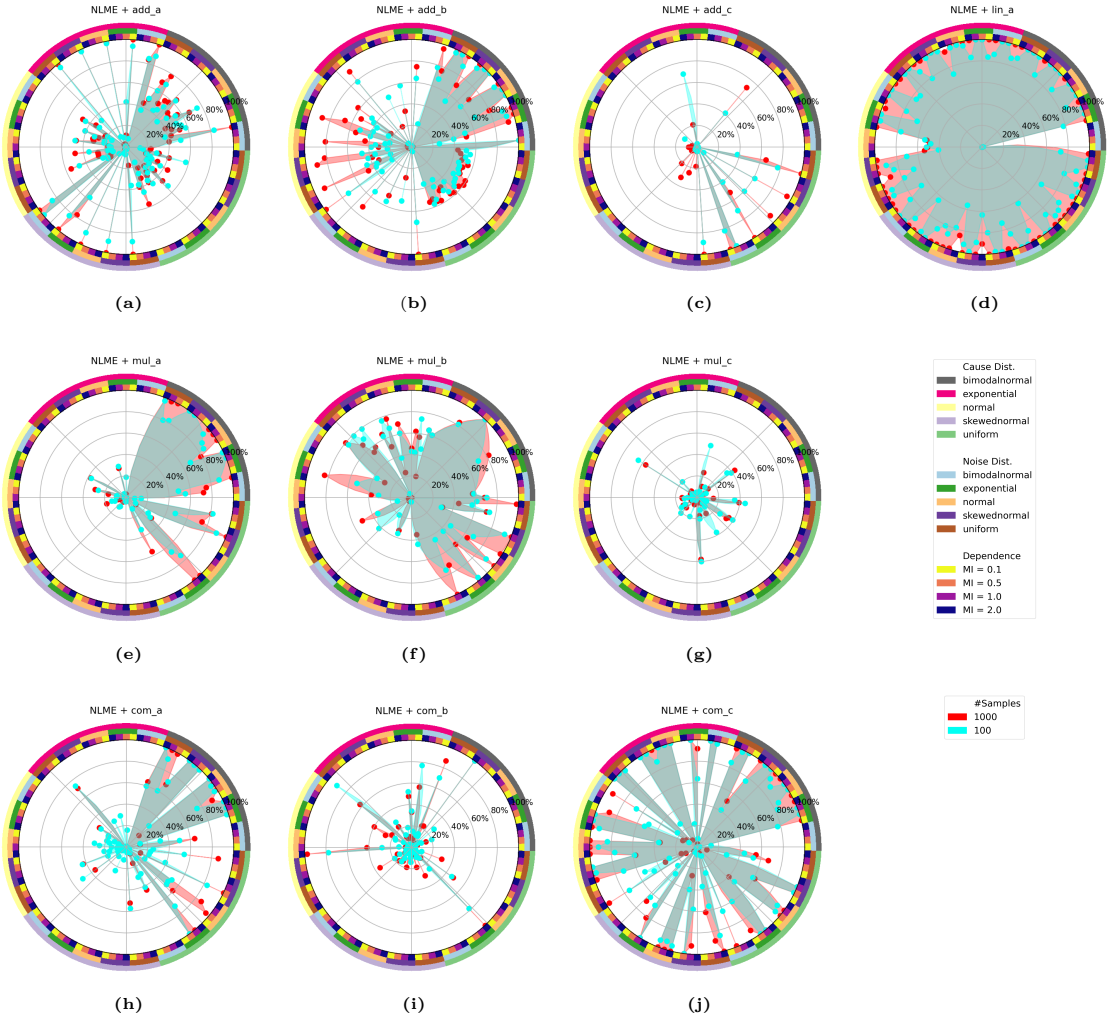



Figure 77: Detailed result overview on our bSCMC data as footprints for NLME.

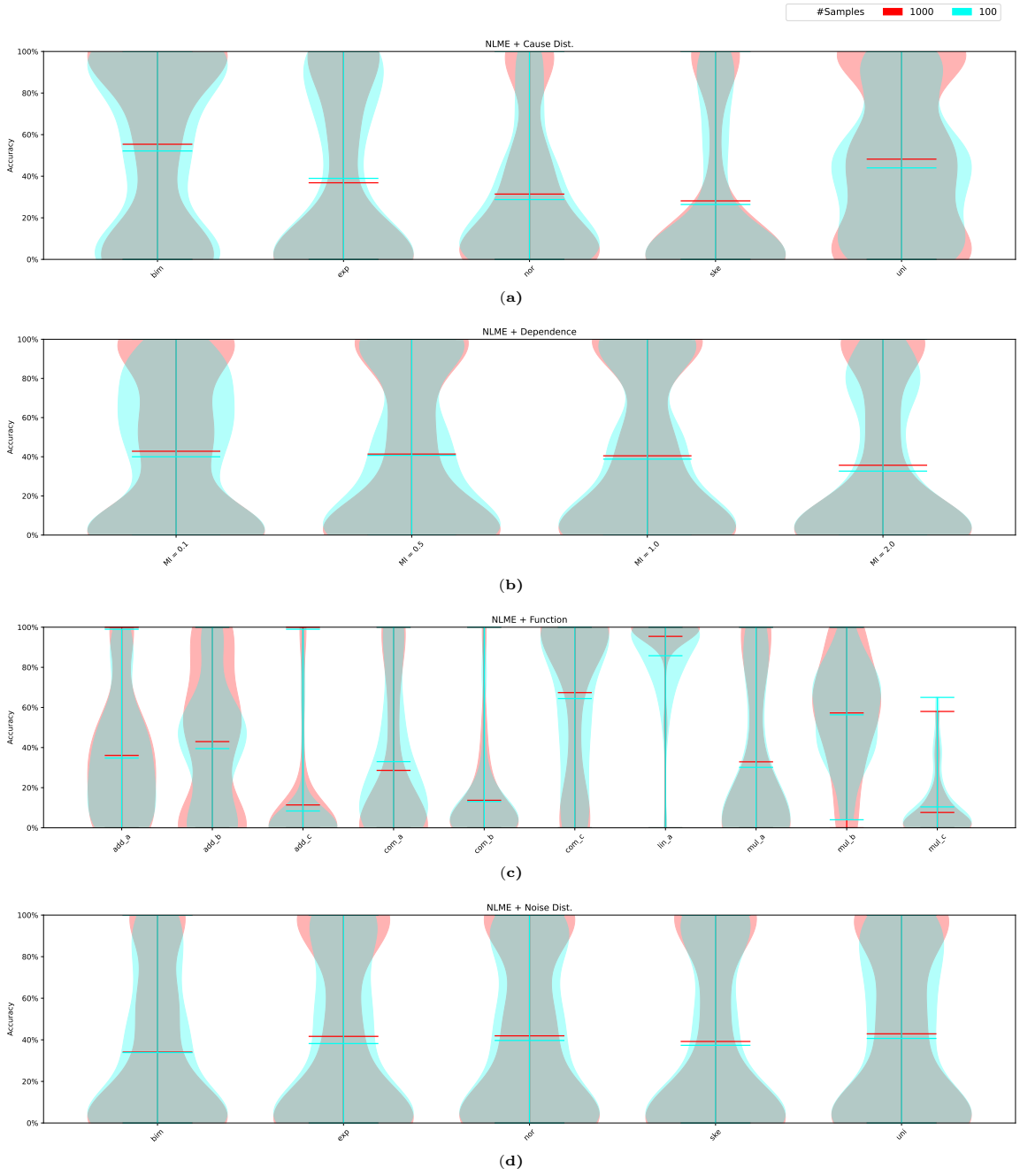



Figure 78: Distribution of accuracies *w.r.t.* individual configurations on our proposed data collection as violins for NLME. Please note, the width of the violins is scaled to unit width individually.

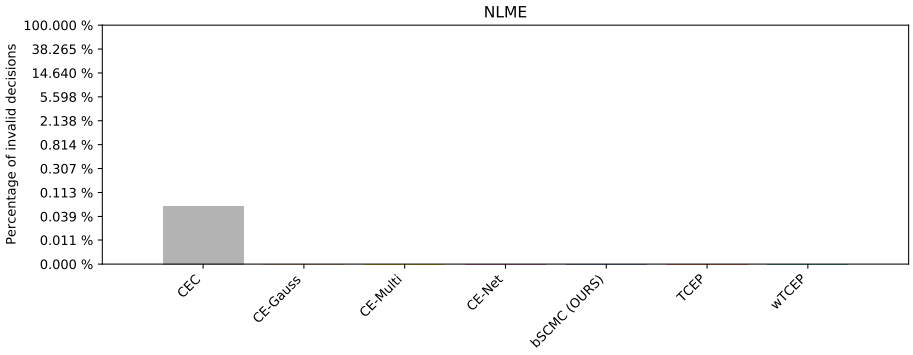



Figure 79: Percentage of invalid decisions for NLME. Note that the y-axis is logarithmically scaled.

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| **(1)** bimodalnormal | | | | |
| **(2)** exponential | 3.9e-07 | | | |
| **(3)** normal | 6.5e-10 | 7.4e-01 | | |
| **(4)** skewednormal | 6.7e-15 | 1.9e-02 | 3.6e-05 | |
| **(5)** uniform | 2.7e-02 | 1.2e-04 | 2.7e-09 | 1.2e-14 |

**(a)** cause distribution

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| **(1)** bimodalnormal | | | | |
| **(2)** exponential | 1.9e-01 | | | |
| **(3)** normal | 3.0e-02 | 5.2e-01 | | |
| **(4)** skewednormal | 3.7e-01 | 6.9e-01 | 2.6e-01 | |
| **(5)** uniform | 3.9e-02 | 6.0e-01 | 9.6e-01 | 3.7e-01 |

**(b)** noise distribution

| | (1) | (2) | (3) |
|---|---|---|---|
| **(1)** MI = 0.1 | | | |
| **(2)** MI = 0.5 | 7.9e-01 | | |
| **(3)** MI = 1.0 | 7.2e-01 | 6.0e-01 | |
| **(4)** MI = 2.0 | 1.9e-02 | 7.2e-03 | 3.1e-02 |

**(c)** mutual information

| | (1) |
|---|---|
| **(1)** 1000 | |
| **(2)** 100 | 8.6e-01 |

**(d)** sample size

| | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) |
|---|---|---|---|---|---|---|---|---|---|
| **(1)** add_a | | | | | | | | | |
| **(2)** add_b | 1.3e-01 | | | | | | | | |
| **(3)** add_c | 3.8e-27 | 1.2e-20 | | | | | | | |
| **(4)** com_a | 3.9e-03 | 8.1e-03 | 8.1e-14 | | | | | | |
| **(5)** com_b | 5.5e-14 | 1.8e-09 | 1.9e-15 | 2.5e-02 | | | | | |
| **(6)** com_c | 5.2e-14 | 1.1e-11 | 3.2e-39 | 7.5e-15 | 1.8e-27 | | | | |
| **(7)** lin_a | 3.0e-47 | 1.1e-40 | 1.5e-57 | 1.6e-36 | 8.6e-55 | 1.2e-08 | | | |
| **(8)** mul_a | 7.0e-03 | 3.0e-02 | 2.1e-11 | 8.6e-01 | 1.5e-01 | 3.4e-12 | 4.2e-29 | | |
| **(9)** mul_b | 9.0e-09 | 1.0e-04 | 5.8e-29 | 4.2e-09 | 9.5e-24 | 1.8e-03 | 2.0e-24 | 5.3e-07 | |
| **(10)** mul_c | 6.1e-17 | 7.4e-12 | 6.9e-10 | 6.5e-04 | 7.9e-02 | 8.8e-29 | 1.8e-52 | 1.1e-02 | 2.3e-26 |

**(f)** functional dependence

Figure 80: Significances at $\alpha = 0.05$-level *w.r.t.* individual configurations for NLME. The numbering of the row and column heads correspond to each other. Green color coding indicates significant differences while red color coding indicates no significant difference.

## D.9 Post Non-linear Noise Model

### D.9.1 Description

The authors of (Zhang and Hyvärinen, 2008) present a general model that relies on the independence of cause $X$ and an independent noise term $\epsilon_Y$. This model is known as the post linear noise model ($PNL$[16]) and is defined as $Y = g_X(f_X(X) + \epsilon_Y)$. The functions $g(\cdot)$ and $f(\cdot)$ can be non-linear in general while only $g(\cdot)$ has to be invertible to identify the causal direction. Using this model allows for higher flexibility *w.r.t.* the covered data generation processes (*e.g.,* multiplicative noise through utilizing $\exp(\cdot)$). However, the authors could identify five cases where the usage of this model leads to unidentifiability (Zhang and Hyvärinen, 2009). To decide for the direction of the causal relationship, the functional is rephrased as $\epsilon_Y = g_X^{-1}(Y) - f_X(X)$. Hence, the approach is to fit functions for $g_X^{-1}$ and $f_X$ using multi-layer perceptrons in such a way that $X \perp\!\!\!\perp \hat{\epsilon_Y}$, which leads to a constrained ICA problem. This can be approached by minimizing the mutual information between $X$ and $\hat{\epsilon_Y}$. An independence test is then conducted using HSIC. Further, it should be noted that the ANM approach is a special case of PNL where the outer function $g(\cdot)$ is the identity function.

### D.9.2 Results

Results obtained for PNL can be found in Figs. 81 to 87. We can observe accuracies ranging from $\sim$ 45% accuracy on our bSCMC data to $\sim$ 80% on CE-Gauss (see Fig. 81.a). Overall we see results in the expected range of accuracies. By examining the performance *w.r.t.* the individual building blocks of our synthetic bSCMC data, we can observe that the method achieves better results on more simple functional dependencies such as lin_a, add_a, add_b (see Fig. 81.b and Fig. 85.c). Thus, it is not surprising that the functional dependence is the most influential building block according to Fig. 82. Further, some compound effects are apparent. For example, in case of add_a (see Fig. 84.a), we can see that, depending on the combination of distribution types, the method fails for low dependency strengths (*e.g.,* for MI = 0.1, bimodal normal cause, and uniform noise distribution). In other configurations, this low dependency strength does only have marginal impact (*e.g.,* for MI = 0.1, uniform cause, and exponential noise distribution). A reason might be that the method relies on fitting functions using perceptrons. Depending on the implementation and parameterization, those perceptrons might not be able to suit the necessary complexity well. An observation which supports this is that we generally see that more data does help the model to deduct a direction (see Fig. 85, compare Fig. 8 for further aggregation). There are apparent "bellies" around the $\sim$ 50% accuracy mark in the small sample regime, *i.e.,* 100 samples, which is a sign for random decisions. In contrast, 1 000 samples lead to either correct decisions or systematic errors. However, we do not see any significance in Fig. 87.d or Fig. 83 *w.r.t.* the sample size. Even if this seems counter-intuitive at first glance, these results can be explained by the symmetry of the results (see Fig. 8 for a more compact visualization) in connection with the linear

16. We rely on the Matlab implementation available at `github.com/ssamot/causality`, which is an adapted version of the authors original code, to obtain our results. Since we use Python to run our experiments, we had to invoke the code from python accordingly. We use the default parameters of the provided implementation and do not perform any data preprocessing.

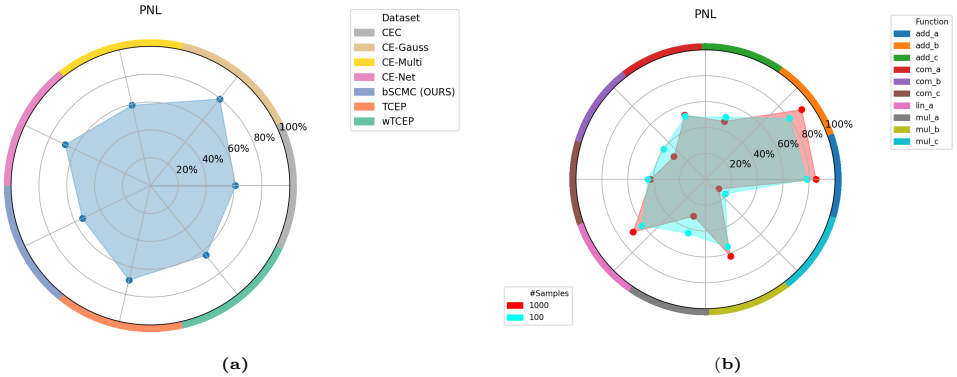



Figure 81: Result footprints for PNL.

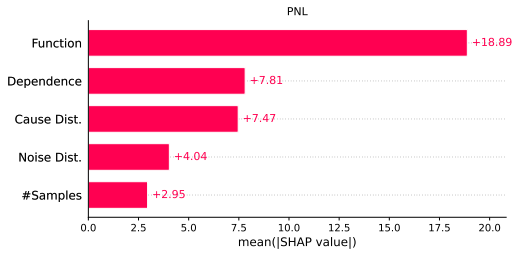



Figure 82: Importance of individual data set characteristics as mean Shapley values for PNL.

| Method | Setting | p-Value |
|---|---|---|
| PNL | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 3.4e-23 |
| PNL | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 4.5e-01 |
| PNL | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 2.7e-35 |
| PNL | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 1.6e-209 |
| PNL | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 4.1e-02 |

Figure 83: Significances at $\alpha = 0.05$-level for conditional independencies of individual data set characteristics *w.r.t.* the obtained accuracy for PNL. Green color coding indicates dependence while red color coding indicates independence.

nature of the significance tests applied (see Section 5.7). As Figs. 83 and 87 suggest, all the remaining building blocks individually influence the results of PNL. Finally, the method does not produce any invalid decisions (see Fig. 86).

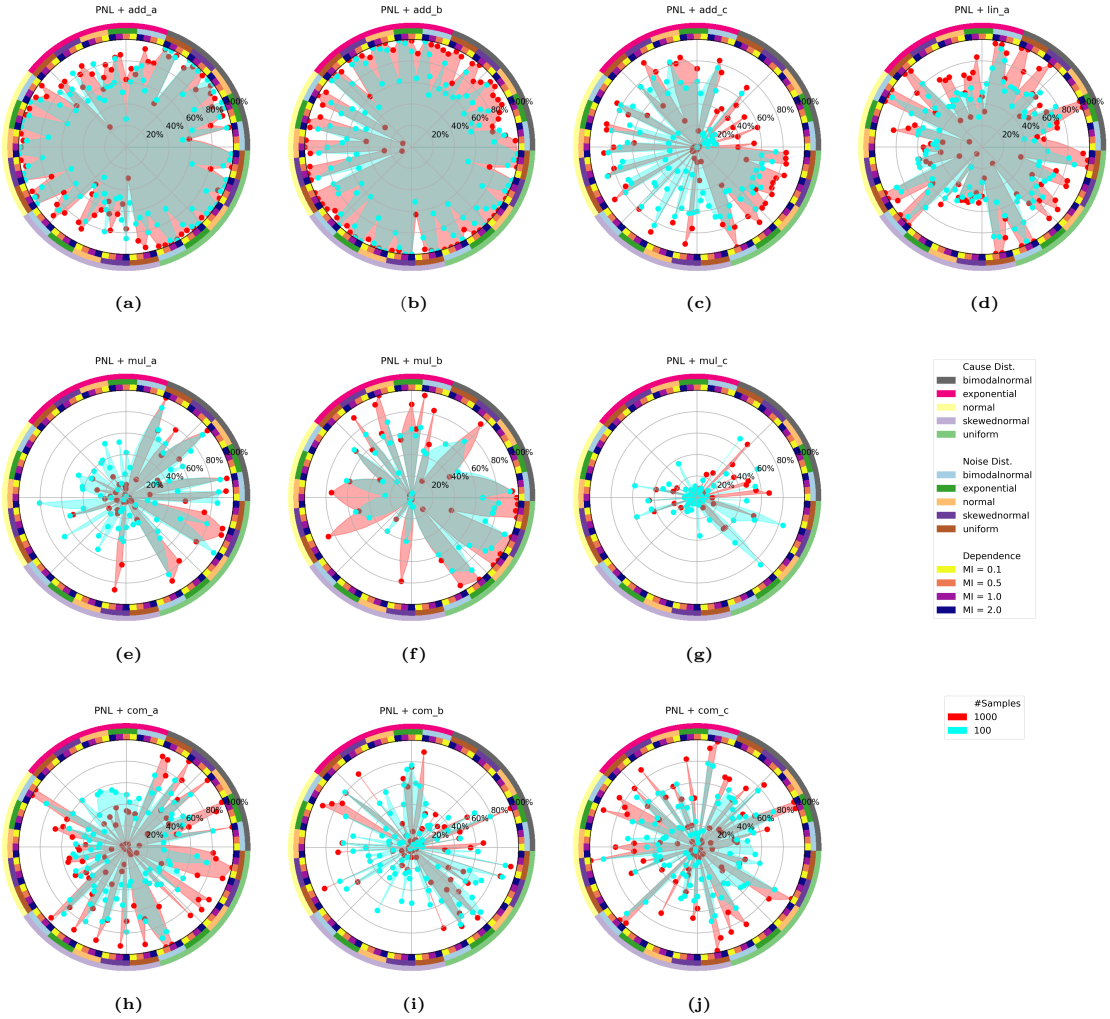



Figure 84: Detailed result overview on our bSCMC data as footprints for PNL.

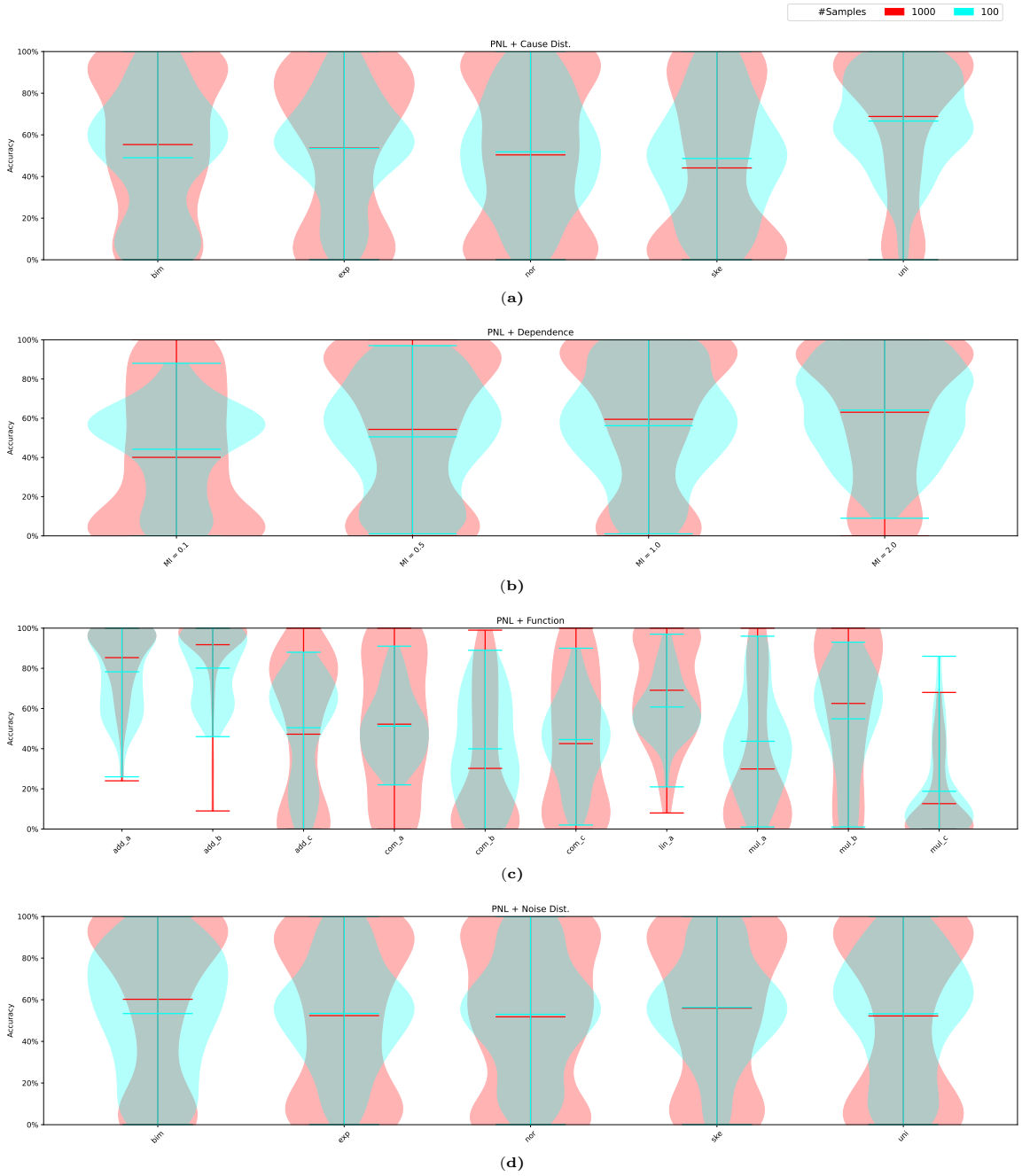



Figure 85: Distribution of accuracies *w.r.t.* individual configurations on our proposed data collection as violins for PNL. Please note, the width of the violins is scaled to unit width individually.

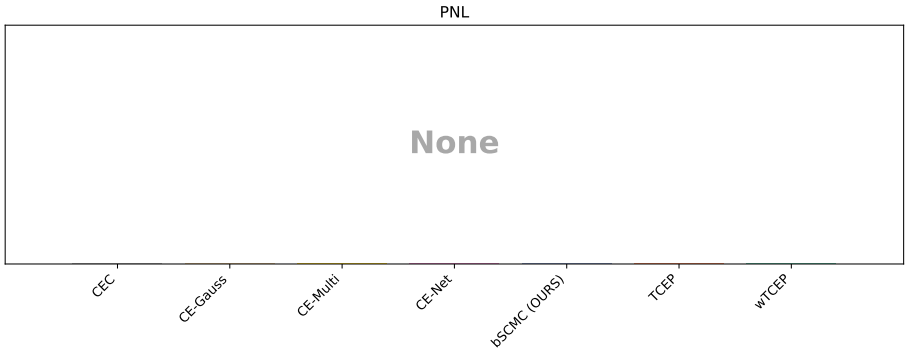



Figure 86: Percentage of invalid decisions for PNL. Note that the y-axis is logarithmically scaled.

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 5.6e-01 | | | |
| (3) normal | 6.6e-01 | 2.9e-01 | | |
| (4) skewednormal | 3.5e-02 | 6.6e-03 | 8.0e-02 | |
| (5) uniform | 1.4e-10 | 2.8e-09 | 2.2e-11 | 5.3e-16 |

**(a)** cause distribution

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 8.0e-02 | | | |
| (3) normal | 4.7e-02 | 8.3e-01 | | |
| (4) skewednormal | 4.8e-01 | 2.0e-01 | 1.3e-01 | |
| (5) uniform | 1.2e-01 | 8.9e-01 | 6.8e-01 | 3.5e-01 |

**(b)** noise distribution

| | (1) | (2) | (3) |
|---|---|---|---|
| (1) MI = 0.1 | | | |
| (2) MI = 0.5 | 1.6e-06 | | |
| (3) MI = 1.0 | 1.7e-12 | 1.2e-02 | |
| (4) MI = 2.0 | 2.7e-22 | 1.1e-06 | 2.7e-02 |

**(c)** mutual information

| | (1) |
|---|---|
| (1) 1000 | |
| (2) 100 | 1.1e-01 |

**(d)** sample size

| | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) |
|---|---|---|---|---|---|---|---|---|---|
| (1) add_a | | | | | | | | | |
| (2) add_b | 2.0e-03 | | | | | | | | |
| (3) add_c | 4.2e-27 | 1.8e-35 | | | | | | | |
| (4) com_a | 1.9e-26 | 1.0e-32 | 8.7e-01 | | | | | | |
| (5) com_b | 1.4e-39 | 4.9e-45 | 4.5e-05 | 1.2e-07 | | | | | |
| (6) com_c | 8.3e-35 | 1.3e-40 | 4.1e-02 | 8.5e-03 | 2.7e-03 | | | | |
| (7) lin_a | 2.6e-14 | 1.2e-21 | 8.5e-05 | 2.4e-07 | 6.9e-20 | 4.3e-14 | | | |
| (8) mul_a | 5.2e-31 | 5.7e-35 | 2.1e-03 | 5.0e-06 | 4.7e-01 | 2.3e-02 | 1.5e-16 | | |
| (9) mul_b | 4.2e-09 | 1.1e-13 | 1.4e-02 | 1.9e-02 | 3.1e-08 | 8.0e-05 | 4.3e-01 | 1.3e-06 | |
| (10) mul_c | 5.7e-50 | 3.4e-51 | 2.9e-18 | 3.4e-26 | 2.4e-08 | 2.2e-19 | 4.1e-42 | 7.7e-10 | 3.1e-19 |

**(f)** functional dependence

Figure 87: Significances at $\alpha = 0.05$-level *w.r.t.* individual configurations for PNL. The numbering of the row and column heads correspond to each other. Green color coding indicates significant differences while red color coding indicates no significant difference.

### D.10 Nonparametric Quantile-Based Causal Discovery

#### D.10.1 Description

The authors of (Tagasovska et al., 2018) build upon the idea of MDL for deciding upon the causal direction given observational data. Since the underlying concept of Kolmogorov complexity is not computable directly, they rely on quantile scoring as a proxy thereof while the conditional quantiles are computed fully non-parametrically by copulas (*i.e.,* multivariate distributions with uniform margins). Hence, the method is named quantile copula causal discovery (*QCCD*[17]). By using copulas, the dependence between variables can be modeled without any assumptions about the scales or functional dependencies. The authors derive their method from a decision-theoretic viewpoint and show the connection between description length and quantile scoring. Thus, a score is computed for each direction while the causal direction can be decided from the lower quantile score.

#### D.10.2 Results

The results we obtain for QCCD can be found in Figs. 88 to 94. Generally we can see results in the expected range *w.r.t.* previous studies, *i.e.,* accuracies ranging from $\sim 51\%$ on CE-Multi to $\sim 80\%$ on CE-Net (see Fig. 88.a). The finer configurations made available by our bSCMC data reveal that the performance of the method is generally very dependent on the configuration of the data generation process. For example, the method is able to decide the causal direction correctly on average when additive functions are involved (*i.e.,* add_a, add_b, and add_c, see Fig. 88.b). However, having a closer look into the fine-grained results, we can see that the other building blocks (*i.e.,* dependency strengths and data distributions) can have a severe impact. For example, there is a systematic error for data generated using the add_a function in connection with exponential causes and a bimodal normal noise distribution (see Fig. 91.a). A further example are the results for data with com_b functions, normal causes, and exponential noise distributions (see Fig. 91.i or Fig. 92.c) where the method performs worse in comparison to most other configurations using the mul_c function (see also Fig. 92.c). We can further see in Fig. 92.d the systematic error occurring when bimodal normal cause distribution is used within the data generation process. Thus, it is not surprising that the noise distribution is the second most impactful building block after the functional dependence (see Fig. 89). Hence, we conclude that the method is very dependent on the data composition (*i.e.,* compound effects among the building blocks are probable). However, we can observe overall that more data helps the model to achieve better results (*i.e.,* 1 000 samples result in higher accuracies than 100 samples per realization). This improvement is also significant (see Fig. 94.d). Further, we can conclude from the remaining parts of Fig. 94 as well as from Fig. 90 that all the building blocks do have an impact on the results already on their own (*i.e.,* even without compound effects). Finally, the method does not produce invalid decisions (see Fig. 93).

17. In order to obtain our results, we rely on the code published by the authors (available at `github.com/tagas/qccd`) while we apply standardization as a data preprocessing step.

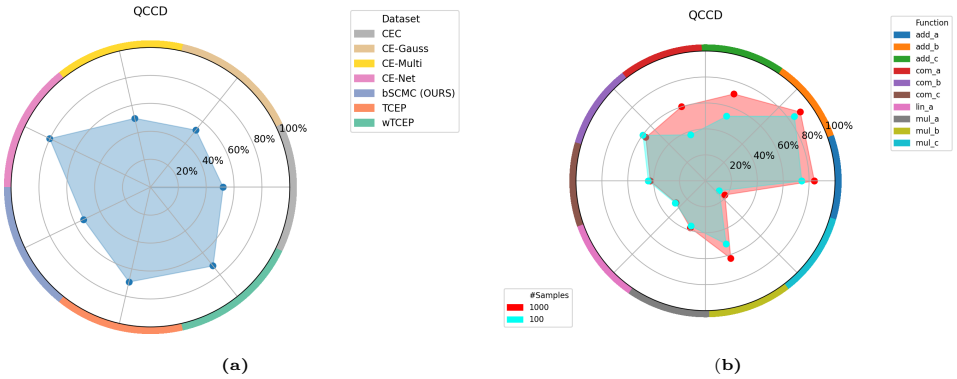



Figure 88: Result footprints for QCCD.

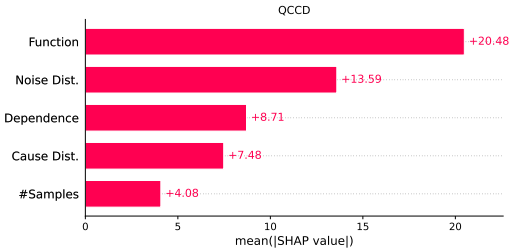



Figure 89: Importance of individual data set characteristics as mean Shapley values for QCCD.

| Method | Setting | p-Value |
|---|---|---|
| QCCD | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 5.3e-05 |
| QCCD | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 2.5e-05 |
| QCCD | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 3.9e-31 |
| QCCD | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 7.7e-141 |
| QCCD | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 6.8e-63 |

Figure 90: Significances at $\alpha = 0.05$-level for conditional independencies of individual data set characteristics *w.r.t.* the obtained accuracy for QCCD. Green color coding indicates dependence while red color coding indicates independence.

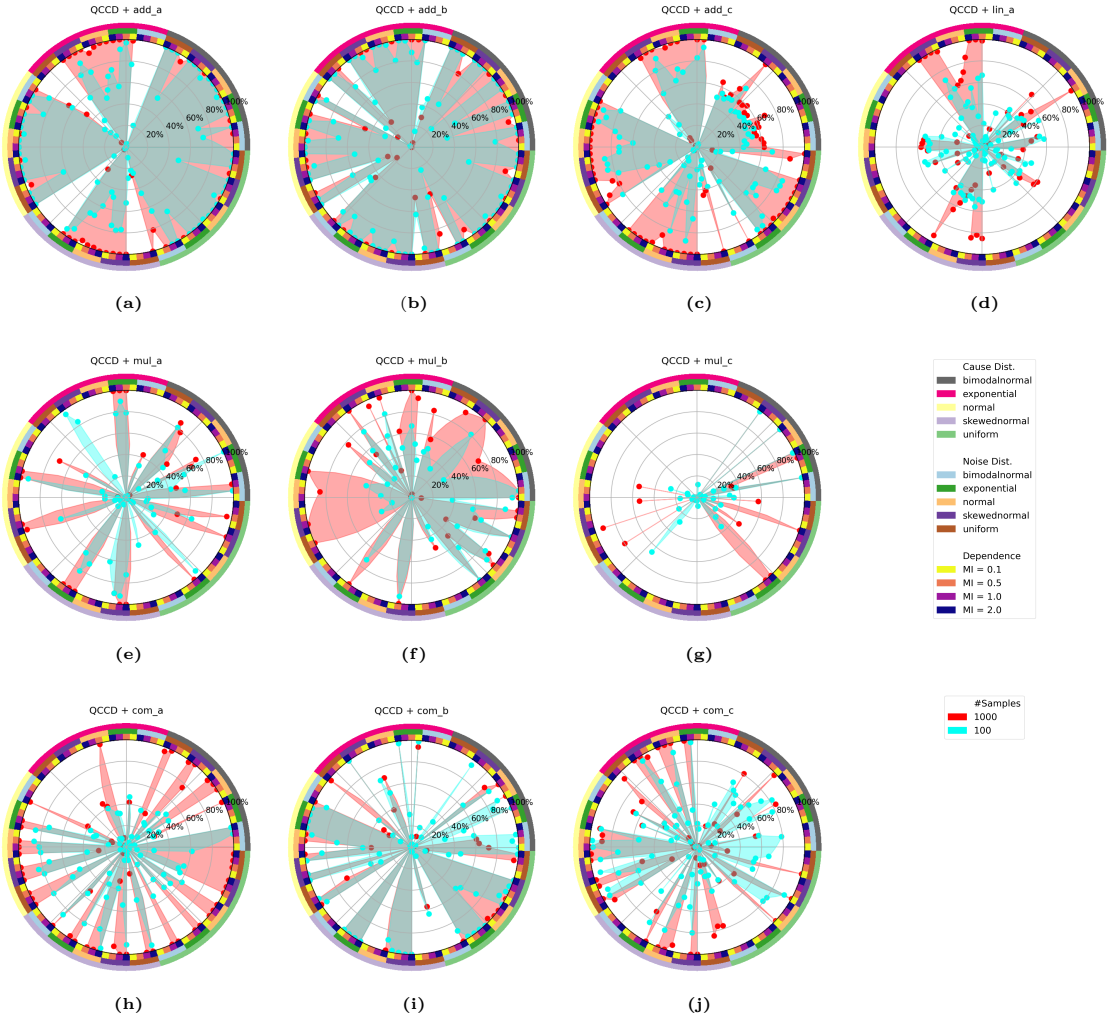



Figure 91: Detailed result overview on our bSCMC data as footprints for QCCD.

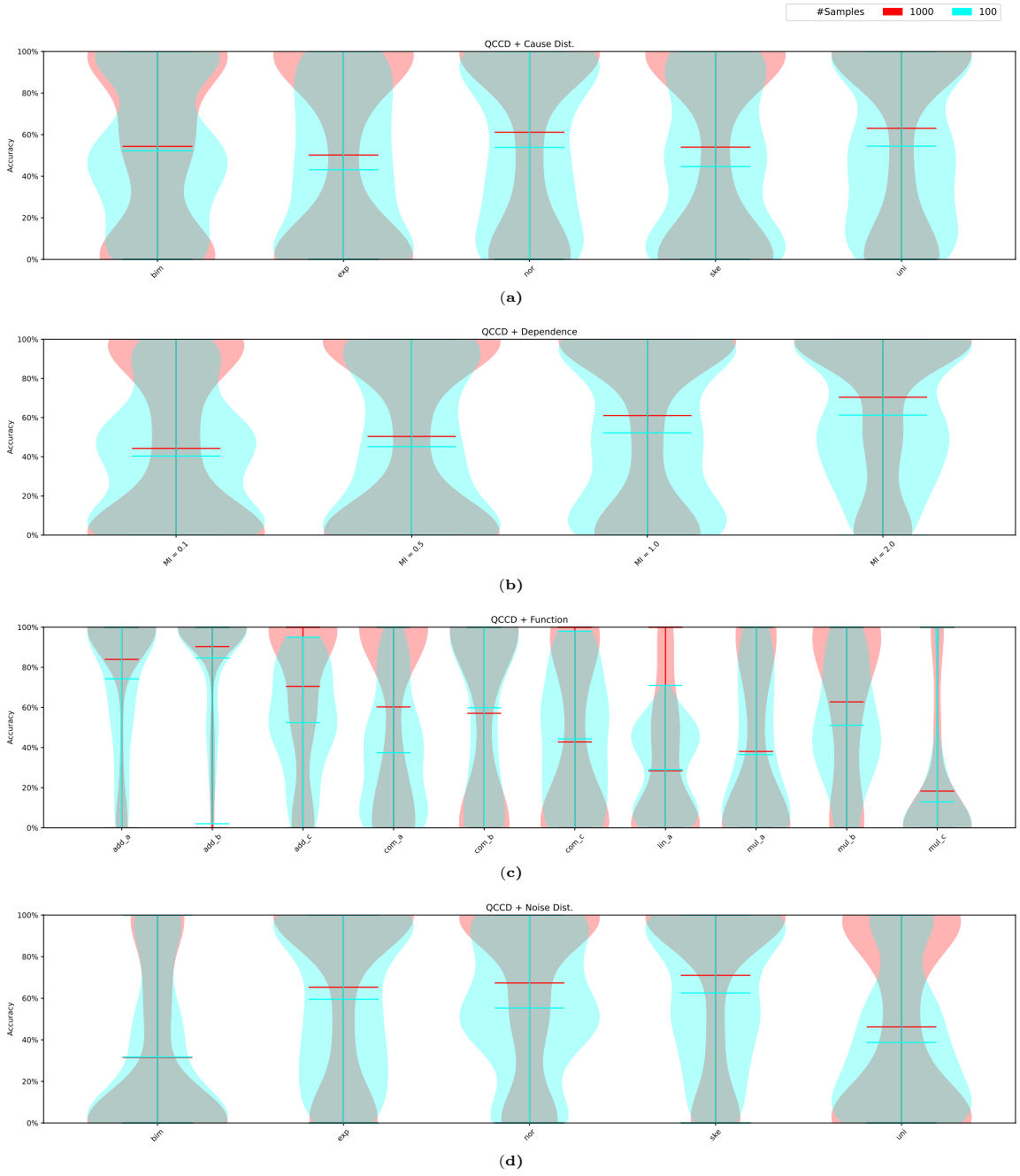



Figure 92: Distribution of accuracies *w.r.t.* individual configurations on our proposed data collection as violins for QCCD. Please note, the width of the violins is scaled to unit width individually.

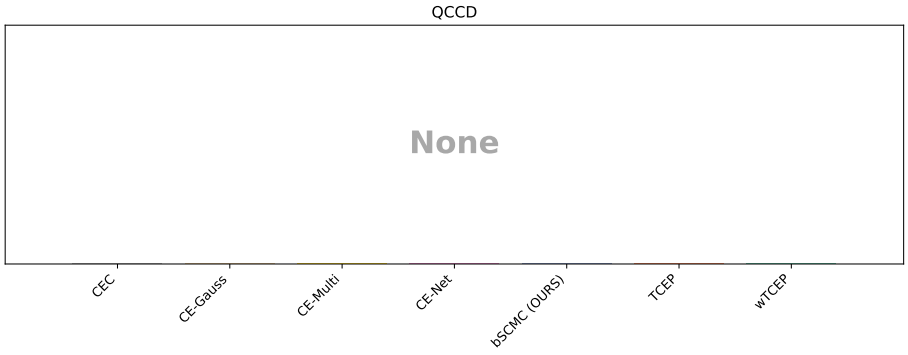



Figure 93: Percentage of invalid decisions for QCCD. Note that the y-axis is logarithmically scaled.

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 1.2e-02 | | | |
| (3) normal | 1.6e-01 | 1.7e-04 | | |
| (4) skewednormal | 1.3e-01 | 3.2e-01 | 5.9e-03 | |
| (5) uniform | 4.9e-02 | 1.9e-05 | 5.0e-01 | 9.4e-04 |

**(a)** cause distribution

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 2.7e-21 | | | |
| (3) normal | 2.6e-19 | 3.1e-01 | | |
| (4) skewednormal | 2.7e-26 | 2.6e-01 | 2.3e-02 | |
| (5) uniform | 3.9e-05 | 4.6e-10 | 6.5e-09 | 3.8e-14 |

**(b)** noise distribution

| | (1) | (2) | (3) |
|---|---|---|---|
| (1) MI = 0.1 | | | |
| (2) MI = 0.5 | 5.5e-02 | | |
| (3) MI = 1.0 | 1.3e-07 | 2.1e-03 | |
| (4) MI = 2.0 | 6.3e-20 | 4.6e-11 | 1.4e-03 |

**(c)** mutual information

| | (1) |
|---|---|
| (1) 1000 | |
| (2) 100 | 1.8e-04 |

**(d)** sample size

| | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) |
|---|---|---|---|---|---|---|---|---|---|
| (1) add_a | | | | | | | | | |
| (2) add_b | 1.7e-02 | | | | | | | | |
| (3) add_c | 7.9e-10 | 3.2e-18 | | | | | | | |
| (4) com_a | 2.0e-13 | 8.7e-23 | 3.3e-03 | | | | | | |
| (5) com_b | 1.2e-07 | 2.2e-14 | 9.8e-01 | 1.1e-01 | | | | | |
| (6) com_c | 2.6e-22 | 4.4e-34 | 1.8e-06 | 5.8e-02 | 8.1e-05 | | | | |
| (7) lin_a | 5.6e-35 | 9.4e-49 | 1.8e-19 | 4.1e-06 | 6.7e-10 | 1.0e-03 | | | |
| (8) mul_a | 3.5e-20 | 1.2e-29 | 6.8e-08 | 1.1e-03 | 9.2e-06 | 1.3e-01 | 4.8e-01 | | |
| (9) mul_b | 7.9e-09 | 3.0e-15 | 2.7e-01 | 2.2e-01 | 3.7e-01 | 5.2e-03 | 7.8e-09 | 3.8e-04 | |
| (10) mul_c | 3.7e-35 | 5.8e-45 | 4.7e-26 | 1.1e-17 | 2.3e-17 | 2.1e-12 | 6.5e-10 | 2.0e-06 | 2.6e-15 |

**(f)** functional dependence

Figure 94: Significances at $\alpha = 0.05$-level *w.r.t.* individual configurations for QCCD. The numbering of the row and column heads correspond to each other. Green color coding indicates significant differences while red color coding indicates no significant difference.

### D.11 Randomized Causation Coefficient

#### D.11.1 Description

Lopez-Paz et al. (2015) propose a two step approach utilizing features of the data distributions. In the first stage, the variables are transformed into a feature representation using kernel mean embeddings (Schölkopf and Smola, 2002). They rely on the fact that potentially infinite dimensional embeddings can be approximated by randomly chosen elements of the Hilbert space (Rahimi and Recht, 2008, 2009). Due to this random embedding, the method is denoted as randomized causation coefficient ($RCC$[18]). The actual embedding used is a combination of embeddings of $X$ and $Y$ in isolation as well as the joint embeddings.

This is followed in stage two by a binary classifier, which is trained on those embeddings to decide whether $X$ causes $Y$, or $Y$ causes $X$. Obviously, this method requires labeled held-out training data to obtain the classifier. They showed empirically that a random forest provides best results. It is assumed that the training and the test data is drawn from the same hidden underlying mother distribution. The authors state that this mother distribution is also the key for identifiability. If empirical distributions drawn from this mother distribution can be assigned with a label for the causal dependency in a deterministic manner, the models are identifiable. In contrast, if this assignment is non-deterministic, the problem becomes unidentifiable.

#### D.11.2 Results

The obtained results are shown in Figs. 95 to 101. Generally, we see RCC to achieve accuracies between $\sim$ 35% on our bSCMC data to $\sim$ 73% on the CE-Net data (see Fig. 95.a). By inspecting the single aspects contained in our synthetic bSCMC data, we can mainly see two things. First, the method is very sample size dependent (*i.e.,* results on data with 100 per realization are much worse than those with 1 000 samples). This difference is also significant (see Fig. 101.d). Second, there is no particular scenario apparent where the method is always working (*i.e.,* it is not independent from single building blocks). For example, even though RCC is able to achieve good results on the exponential cause distribution in case of add_c, mul_c, or com_b functions (see Fig. 98.c,g,i), it fails in a vast majority of cases on data generated with com_c functions (see Fig. 98.j). Further, it mostly fails with a systematic error (compare Fig. 100), *i.e.,* the classifier is confident about its decision. This behavior can also be found in Fig. 99. Thus, we assume that the data generator included in the author's code does not deliver training data suitable for the given test data, *i.e.,* the underlying assumption of a common mother distribution is violated. Training the model on the provided training part of CEC13 or even on further realizations of our bSCMC data might vastly improve the results. However, we want to evaluate the out-of-the-box performance of the algorithms and thus do not optimize the method or its training data. Further, we can conclude from Fig. 101 that the performance of RCC is not influenced by most noise distributions as well as mutual information values up to 1.0. However, all the individual building blocks have a general impact on the performance

18. We use the original code provided by the authors that is available at `github.com/lopezpaz/causation_learning_theory`. In order to run our experiments, we had to adapt it accordingly. Further, the code comes with an own training data generator, which we used to initially train our prediction model. We do not apply any sort of data preprocessing by ourselves.

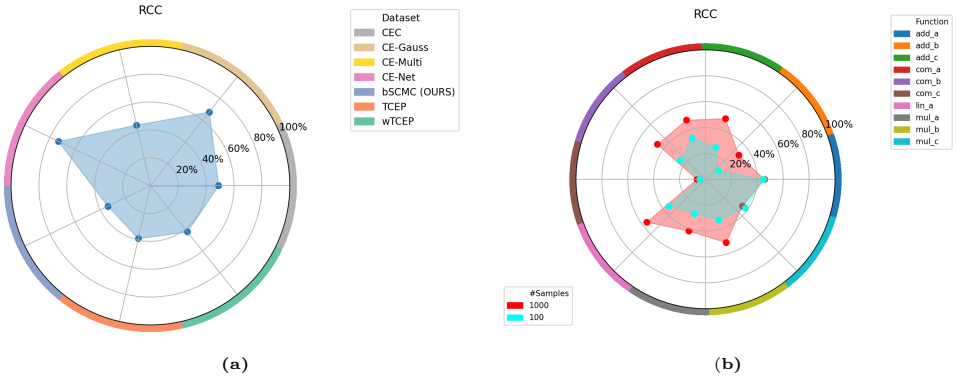



Figure 95: Result footprints for RCC.

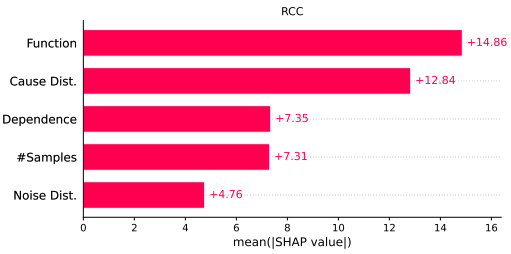



Figure 96: Importance of individual data set characteristics as mean Shapley values for RCC.

| Method | Setting | p-Value |
|---|---|---|
| RCC | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 4.1e-28 |
| RCC | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 1.1e-13 |
| RCC | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 5.4e-03 |
| RCC | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 2.9e-36 |
| RCC | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 3.3e-02 |

Figure 97: Significances at $\alpha = 0.05$-level for conditional independencies of individual data set characteristics *w.r.t.* the obtained accuracy for RCC. Green color coding indicates dependence while red color coding indicates independence.

(see Fig. 97). This independence of the algorithm comes more from failing rather than from performing good regardless of the configuration. Finally, by examining the results in Fig. 96, we can see that the functional dependence and the cause distribution are the most influential building blocks.

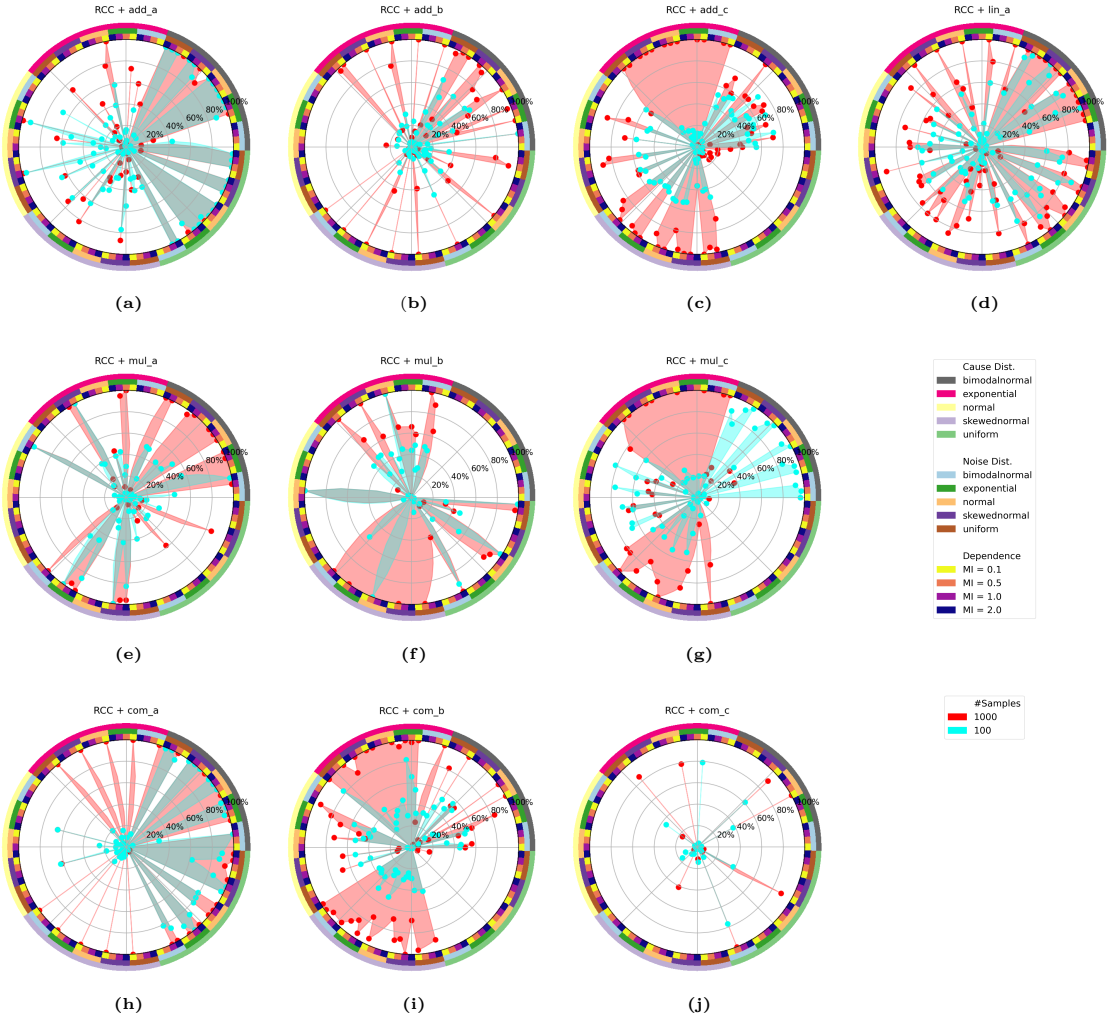



Figure 98: Detailed result overview on our bSCMC data as footprints for RCC.

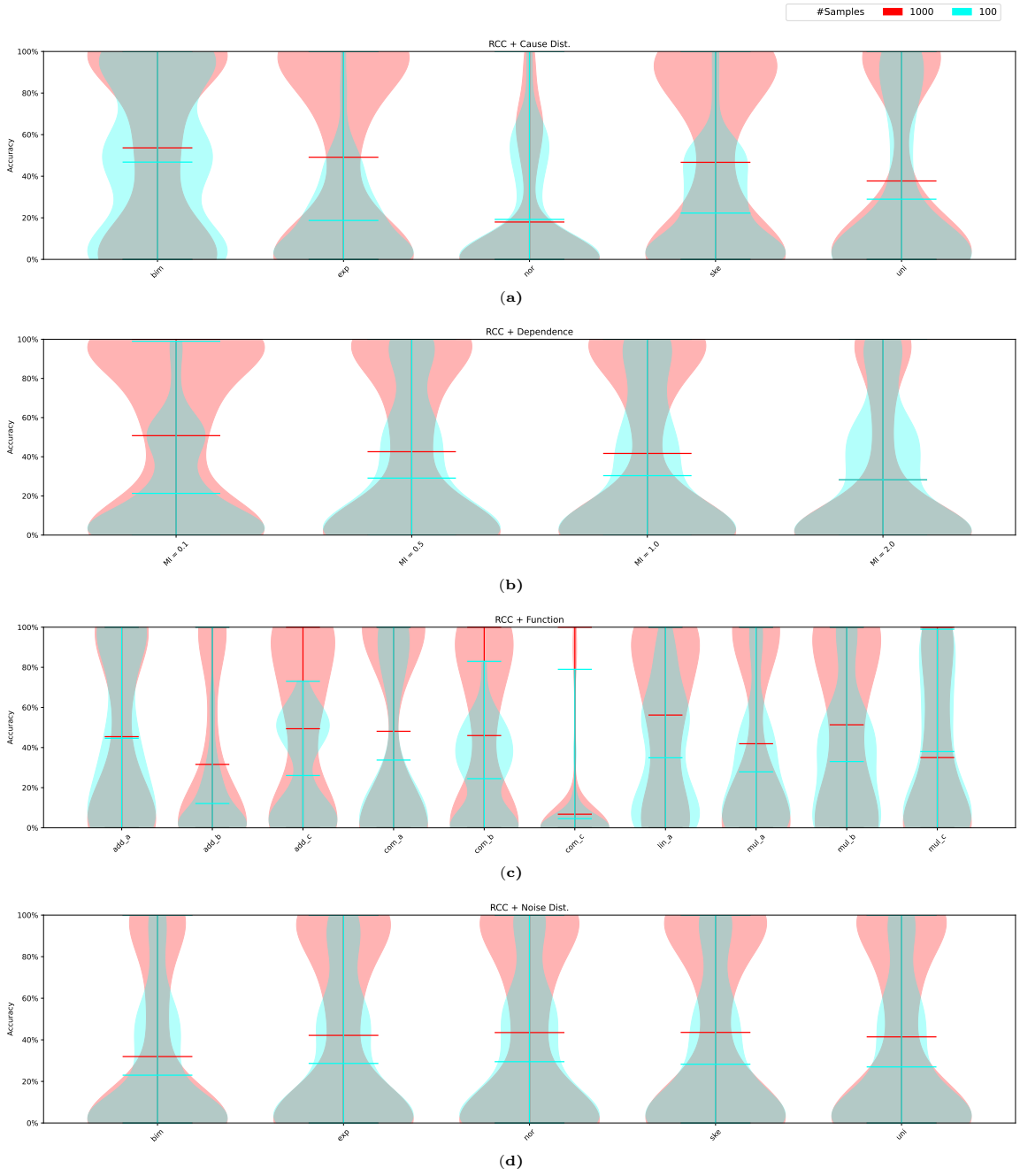



Figure 99: Distribution of accuracies *w.r.t.* individual configurations on our proposed data collection as violins for RCC. Please note, the width of the violins is scaled to unit width individually.

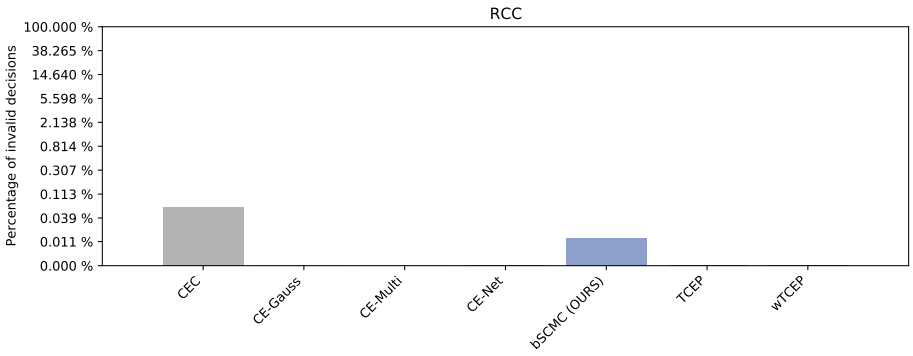



Figure 100: Percentage of invalid decisions for RCC. Note that the y-axis is logarithmically scaled.

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 1.1e-05 | | | |
| (3) normal | 9.6e-26 | 1.6e-09 | | |
| (4) skewednormal | 1.3e-06 | 8.2e-01 | 5.9e-11 | |
| (5) uniform | 2.4e-06 | 5.2e-01 | 4.3e-10 | 5.0e-01 |

**(a)** cause distribution

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 2.4e-03 | | | |
| (3) normal | 9.7e-04 | 7.3e-01 | | |
| (4) skewednormal | 1.1e-03 | 7.5e-01 | 9.3e-01 | |
| (5) uniform | 4.9e-03 | 8.5e-01 | 5.5e-01 | 6.3e-01 |

**(b)** noise distribution

| | (1) | (2) | (3) |
|---|---|---|---|
| (1) MI = 0.1 | | | |
| (2) MI = 0.5 | 3.4e-01 | | |
| (3) MI = 1.0 | 2.4e-01 | 8.3e-01 | |
| (4) MI = 2.0 | 2.4e-05 | 1.6e-03 | 4.0e-03 |

**(c)** mutual information

| | (1) |
|---|---|
| (1) 1000 | |
| (2) 100 | 2.6e-03 |

**(d)** sample size

| | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) |
|---|---|---|---|---|---|---|---|---|---|
| (1) add_a | | | | | | | | | |
| (2) add_b | 6.6e-09 | | | | | | | | |
| (3) add_c | 1.6e-01 | 3.7e-07 | | | | | | | |
| (4) com_a | 5.8e-02 | 5.0e-03 | 5.9e-01 | | | | | | |
| (5) com_b | 5.6e-03 | 8.5e-04 | 9.2e-02 | 3.2e-01 | | | | | |
| (6) com_c | 1.3e-33 | 4.9e-16 | 4.1e-33 | 8.2e-21 | 2.0e-22 | | | | |
| (7) lin_a | 9.0e-01 | 6.0e-12 | 3.0e-02 | 9.4e-02 | 8.1e-04 | 6.1e-39 | | | |
| (8) mul_a | 5.1e-02 | 4.4e-04 | 3.0e-01 | 7.8e-01 | 6.7e-01 | 7.5e-25 | 9.8e-03 | | |
| (9) mul_b | 5.5e-01 | 3.8e-05 | 7.7e-01 | 4.4e-01 | 1.0e-01 | 5.5e-22 | 3.3e-01 | 2.6e-01 | |
| (10) mul_c | 2.3e-02 | 4.4e-04 | 3.8e-01 | 6.9e-01 | 5.7e-01 | 2.9e-24 | 1.1e-02 | 8.7e-01 | 2.5e-01 |

**(f)** functional dependence

Figure 101: Significances at $\alpha = 0.05$-level *w.r.t.* individual configurations for RCC. The numbering of the row and column heads correspond to each other. Green color coding indicates significant differences while red color coding indicates no significant difference.

## D.12 Regression Error based Causal Inference

### D.12.1 Description

Inferring the causal direction using the least-squares errors on regressors fitted for both directions is presented in (Blöbaum et al., 2018). The authors refer to this approach as regression error based causal inference (*RECI*[19]), which has the advantage that it does not involve an independence test of any sort. This approach follows the intuition that the effect is easier to predict from the cause. In turn, the error should be higher when one predicts the cause from the effect. Besides this intuitive approach, the authors prove this asymmetry under certain conditions. In detail, the authors assume the underlying model to be $Y = f_X(X) + \epsilon_Y$ while they allow dependence between $\epsilon_Y$ and $X$. The noise $\epsilon_Y$ needs to be sufficiently small while $X$ and $Y$ need to be equally scaled real-valued variables. Further, $f_X$ is allowed to be invertible (non-invertible functions lead to even larger errors in anti-causal direction). The key for this approach lies in the independence of the function $f_X$, the marginal $P(X)$, and the conditional distribution of the noise $P(\epsilon_Y|X)$.

From a technical perspective, the authors found that simple models, which tend to underfit the data, perform better in general. This is supported by the intuition that learning anti-causal relationships may require more complex models. A simple model might be capable of approximating the causal direction quite well while it increases the error in the anti-causal direction even further. The authors evaluated different kinds of models such as several functional dependencies (*e.g.,* logarithmic or polynomial), support vector regression, and neural networks, and conclude that the choice of models and parameters thereof are heavily task-dependent.

### D.12.2 Results

Results obtained for RECI are presented in Figs. 102 to 108. Generally, we can see results in a range from $\sim$ 52% using the CEC13 data to $\sim$ 85% on the CE-Multi data (see Fig. 102.a). By subdividing the results on our bSCMC data, we can see that the method achieves good results in the majority of individual settings. However, it performs worst on average on data generated with mul_b, lin_a, and com_c functions (see Fig. 102.b and Fig. 106.c). Further examining the results obtained on data generated by those functions reveal further dependencies, *e.g., w.r.t.* the cause distributions (*e.g.,* very good results for uniform, see Fig. 105.d,j and Fig. 102.a). From Fig. 103 we can see that the functional dependence is by far the most influential building block. However, we can also see that the method tends to fail in the low and high dependence regime (*i.e.,* MI = 0.1 and MI = 2.0, see Fig. 105.b,h and Fig. 106.b) in some cases. While this behavior might appear inconclusive on first glance, we attribute it to the fact that certain combinations of the building blocks might lead to data sets where the regression error, using the models proposed by the authors, is lower in the anti-causal direction in general (see also Appendices D.13.2 and D.14.2). This is supported by the fact that the number of samples per realization appears to have only marginal impact in cases where the method fails (it fails with an accuracy close to 0%, *i.e.,* a systematic error). Generally, the influence of the

19. We use the implementation given by the causal discovery toolbox (Kalainathan and Goudet, 2019) for our experiments. We use the default parameter setting as provided by the toolbox while we normalize the data to $[0, 1]$ before applying the method.

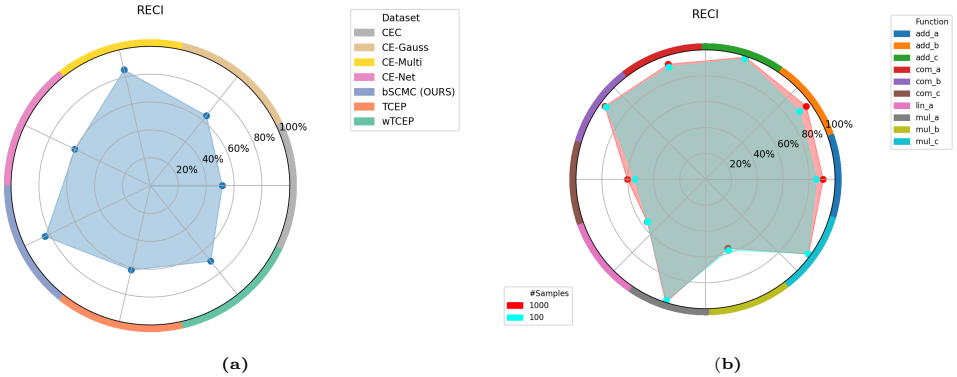



Figure 102: Result footprints for RECI.

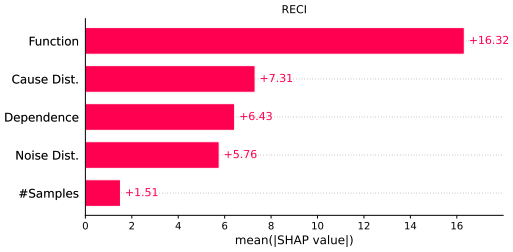



Figure 103: Importance of individual data set characteristics as mean Shapley values for RECI.

| Method | Setting | p-Value |
|---|---|---|
| RECI | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 5.2e-40 |
| RECI | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 5.3e-02 |
| RECI | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 1.2e-14 |
| RECI | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 2.0e-135 |
| RECI | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 2.0e-20 |

Figure 104: Significances at $\alpha = 0.05$-level for conditional independencies of individual data set characteristics *w.r.t.* the obtained accuracy for RECI. Green color coding indicates dependence while red color coding indicates independence.

sample size on the resulting accuracy is not significant in general (see Fig. 104) which is caused by the model performing with only ■ 100 well already, even if the improvement of having more samples available is significant. However, all the remaining building blocks do have a significant impact on the performance of the method (see Fig. 108 and Fig. 104) while there were no invalid decisions taken by the method (see Fig. 107).

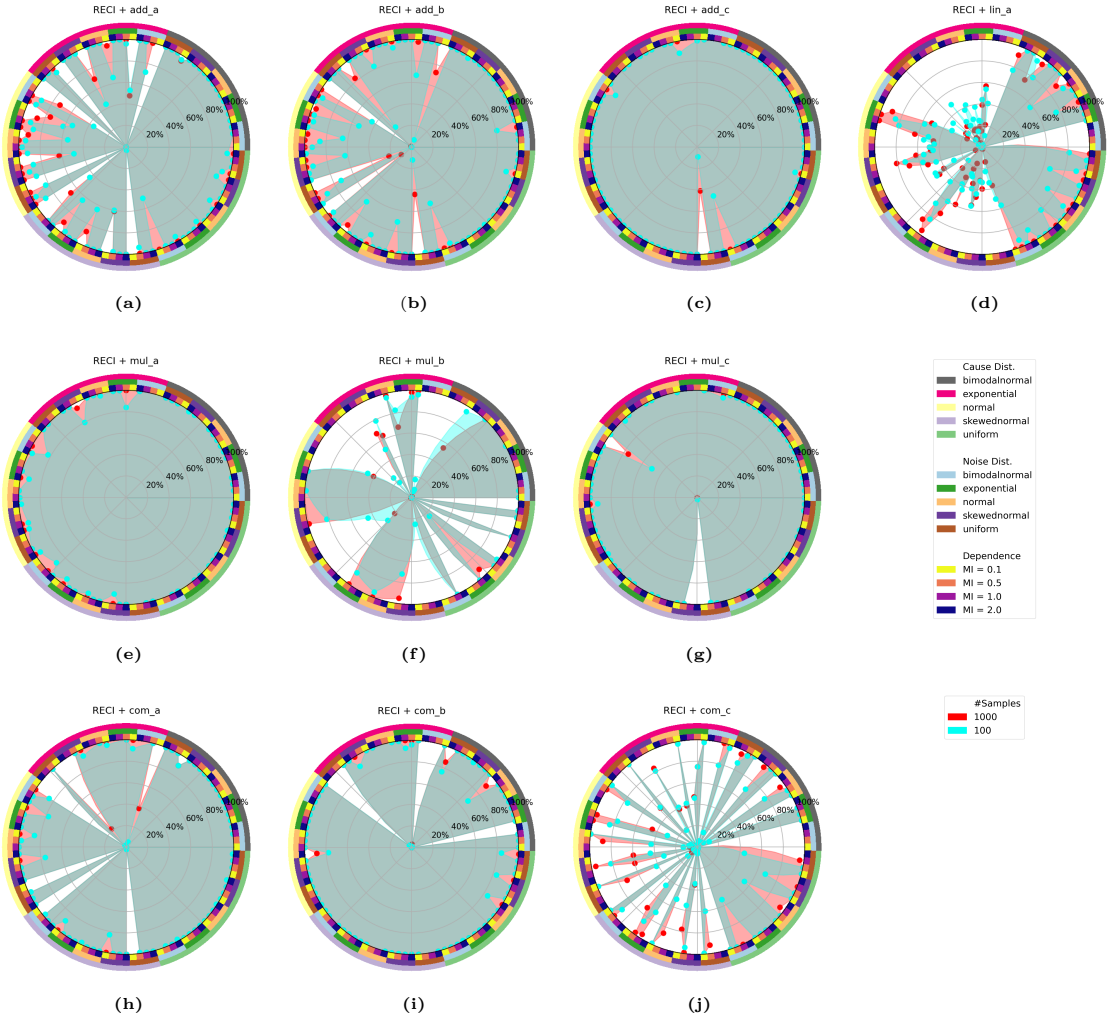



Figure 105: Detailed result overview on our bSCMC data as footprints for RECI.

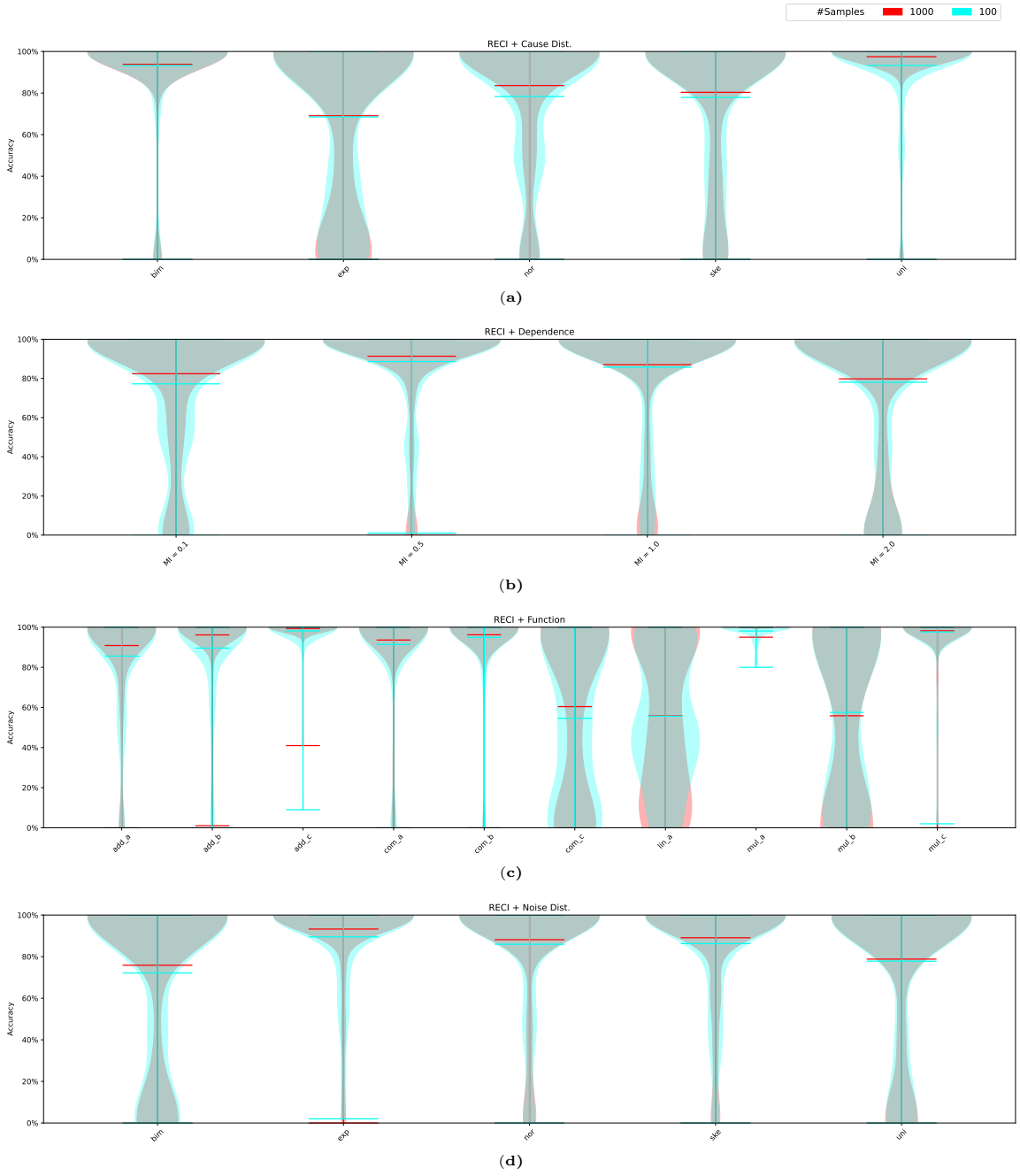



Figure 106: Distribution of accuracies *w.r.t.* individual configurations on our proposed data collection as violins for RECI. Please note, the width of the violins is scaled to unit width individually.

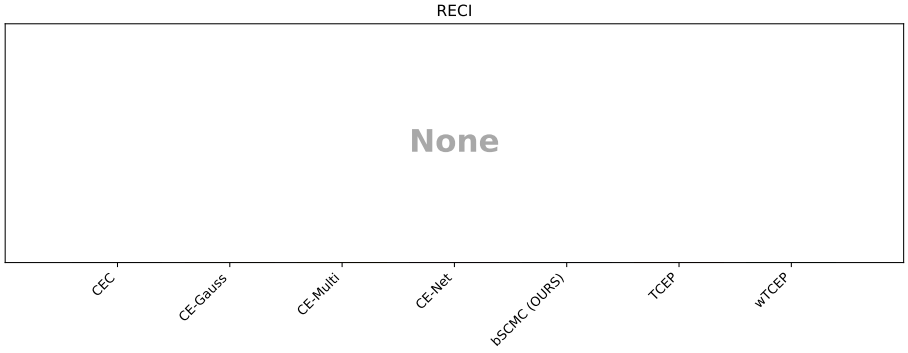



Figure 107: Percentage of invalid decisions for RECI. Note that the y-axis is logarithmically scaled.

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 7.5e-21 | | | |
| (3) normal | 2.3e-16 | 8.9e-03 | | |
| (4) skewednormal | 2.9e-11 | 1.0e-03 | 3.5e-01 | |
| (5) uniform | 8.6e-01 | 1.6e-21 | 1.1e-16 | 1.9e-11 |

**(a)** cause distribution

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 7.8e-06 | | | |
| (3) normal | 1.1e-03 | 2.1e-01 | | |
| (4) skewednormal | 8.1e-04 | 1.9e-01 | 1.0e+00 | |
| (5) uniform | 5.1e-01 | 7.3e-05 | 6.5e-03 | 5.7e-03 |

**(b)** noise distribution

| | (1) | (2) | (3) |
|---|---|---|---|
| (1) MI = 0.1 | | | |
| (2) MI = 0.5 | 1.6e-04 | | |
| (3) MI = 1.0 | 6.3e-05 | 6.1e-01 | |
| (4) MI = 2.0 | 5.3e-01 | 3.9e-03 | 8.1e-04 |

**(c)** mutual information

| | (1) |
|---|---|
| (1) 1000 | |
| (2) 100 | 1.1e-09 |

**(d)** sample size

| | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) |
|---|---|---|---|---|---|---|---|---|---|
| (1) add_a | | | | | | | | | |
| (2) add_b | 1.3e-02 | | | | | | | | |
| (3) add_c | 3.7e-14 | 1.1e-08 | | | | | | | |
| (4) com_a | 1.2e-03 | 3.5e-01 | 1.4e-06 | | | | | | |
| (5) com_b | 1.6e-07 | 1.6e-03 | 6.1e-03 | 2.9e-02 | | | | | |
| (6) com_c | 1.7e-14 | 2.7e-22 | 8.7e-38 | 7.9e-22 | 1.1e-28 | | | | |
| (7) lin_a | 5.8e-21 | 3.0e-29 | 1.2e-44 | 1.9e-27 | 3.6e-35 | 6.4e-01 | | | |
| (8) mul_a | 7.1e-06 | 1.5e-02 | 1.0e-03 | 1.4e-01 | 5.9e-01 | 7.3e-24 | 2.9e-30 | | |
| (9) mul_b | 6.5e-10 | 1.3e-15 | 5.0e-31 | 4.7e-16 | 4.2e-22 | 7.2e-01 | 9.4e-01 | 2.1e-18 | |
| (10) mul_c | 1.0e-09 | 1.6e-05 | 2.7e-01 | 5.4e-04 | 1.3e-01 | 1.7e-28 | 3.4e-34 | 4.0e-02 | 3.4e-23 |

**(f)** functional dependence

Figure 108: Significances at $\alpha = 0.05$-level *w.r.t.* individual configurations for RECI. The numbering of the row and column heads correspond to each other. Green color coding indicates significant differences while red color coding indicates no significant difference.

### D.13 MDL-based Local and Global Regression

#### D.13.1 Description

In (Marx and Vreeken, 2017) the *SLOPE*[20] criterion is introduced, a criterion which decides according to the complexity of functional dependence in the causal and the anti-causal direction. Thus, the authors propose to fit models in both directions and measure the complexity and the fit. The considered functions are linear, quadratic, cubic, reciprocal, and exponential functions while also compounds of deterministic and non-deterministic functions are considered. However, the general functional dependence is assumed to be $X = f_X(X) + \epsilon_Y$ and the regression error is treated as a Gaussian with zero mean (justified by fitting the models by minimizing the sum of errors). The measures of both, fit and error, are based on bit length of the encoding of those, *i.e.,* based on compression. Hence, the penalty of compression costs for complex functions is balanced with the compression cost for regression errors.

#### D.13.2 Results

Results obtained for SLOPE can be found in Figs. 109 to 115. The results appear to be quite similar to those of RECI shown in Fig. 102 and range from $\sim 41\%$ on the CEC13 data to $\sim 88\%$ on the CE-Multi data (Fig. 109.a). Interestingly, roughly the same pattern *w.r.t.* single building blocks of our synthetic bSCMC data appears as in case of RECI (and thus also roughly the same significances, see Fig. 115, and results for the Shapley analysis, see Fig. 110). Please note, when comparing Fig. 113 to Fig. 106, keep the independent scaling of each violin to unit width in mind. However, we can observe some differences here as well while the results for the lin_a function in Fig. 106.c appear to be the most apparent one. While we concluded in case of RECI that certain configurations of the data generation process could lead to data with a regression error that is lower in anti-causal direction, the same argument might hold in terms of complexity. Intuitively, if the anti-causal direction is "easier" to infer, the model leading to this data might also be less complex as in the causal direction. Thus, deciding according to fit and complexity, as in the case of SLOPE, might give most similar predictions in those cases. In contrast to RECI, we can observe a high performance gain for SLOPE in settings containing more samples (*i.e.,* 1 000 samples result in higher accuracies than 100 samples per realization). It is therefore not surprising, in contrast to RECI, that every building block does have a significant influence on the resulting accuracy (see Fig. 111). However, please note that the results also include a number of invalid decisions (see Fig. 114).

20. We rely on the original code that was published by the authors at `eda.mmci.uni-saarland.de/prj/slope/` to conduct our experiments. The data is normalized to the same domain before the method is applied. We leave the involved parameters as set by the provided code.

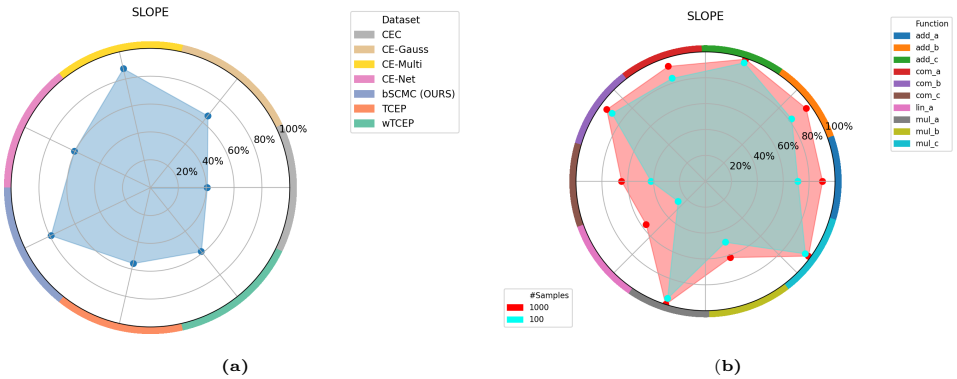



Figure 109: Result footprints for SLOPE.

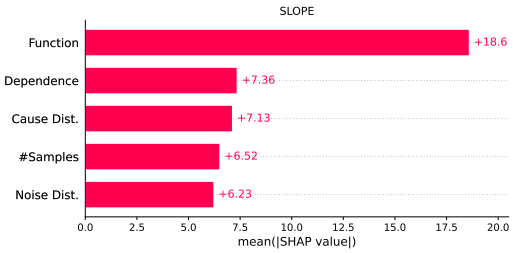



Figure 110: Importance of individual data set characteristics as mean Shapley values for SLOPE.

| Method | Setting | p-Value |
|---|---|---|
| SLOPE | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 1.2e-28 |
| SLOPE | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 1.1e-23 |
| SLOPE | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 5.3e-14 |
| SLOPE | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 1.7e-154 |
| SLOPE | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 1.3e-19 |

Figure 111: Significances at $\alpha = 0.05$-level for conditional independencies of individual data set characteristics *w.r.t.* the obtained accuracy for SLOPE. Green color coding indicates dependence while red color coding indicates independence.

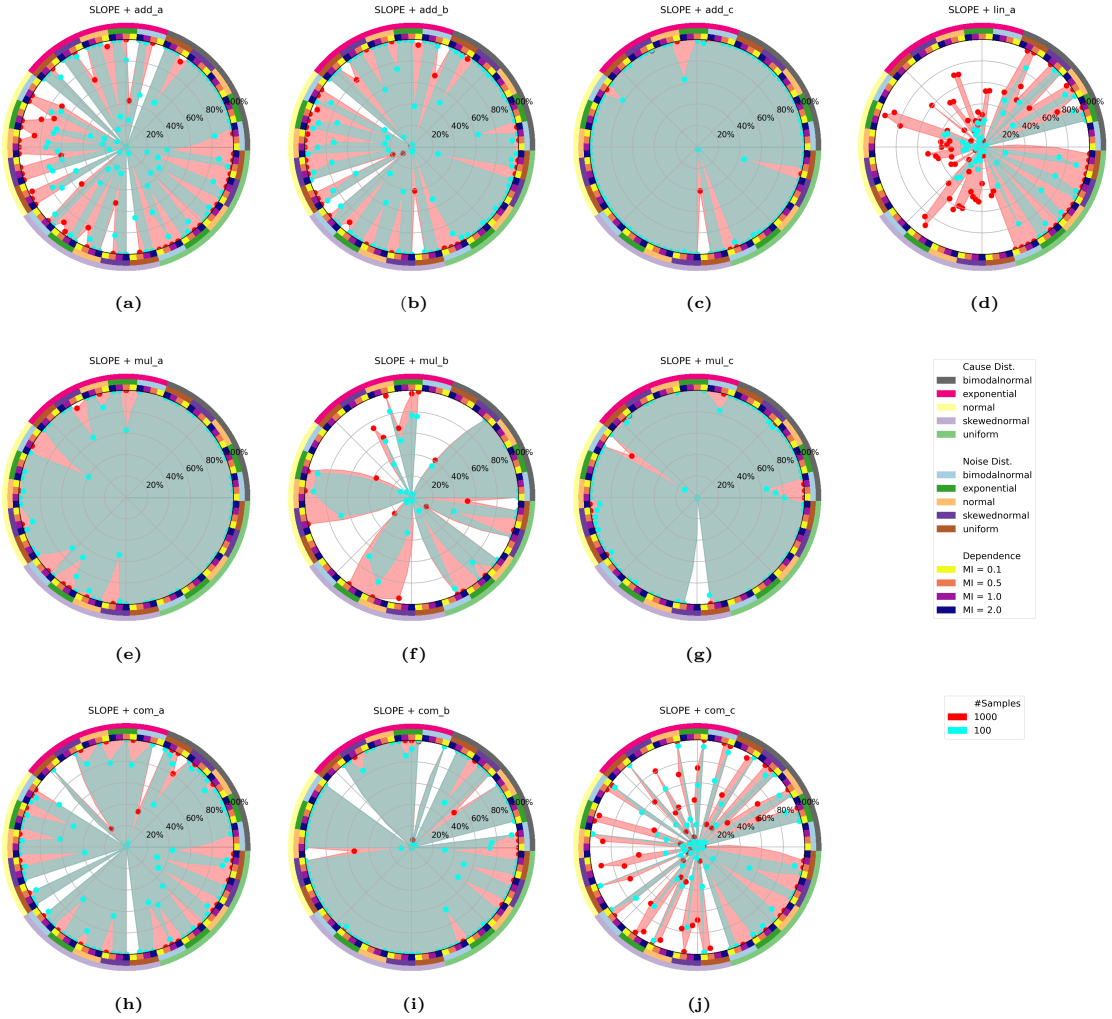



Figure 112: Detailed result overview on our bSCMC data as footprints for SLOPE.

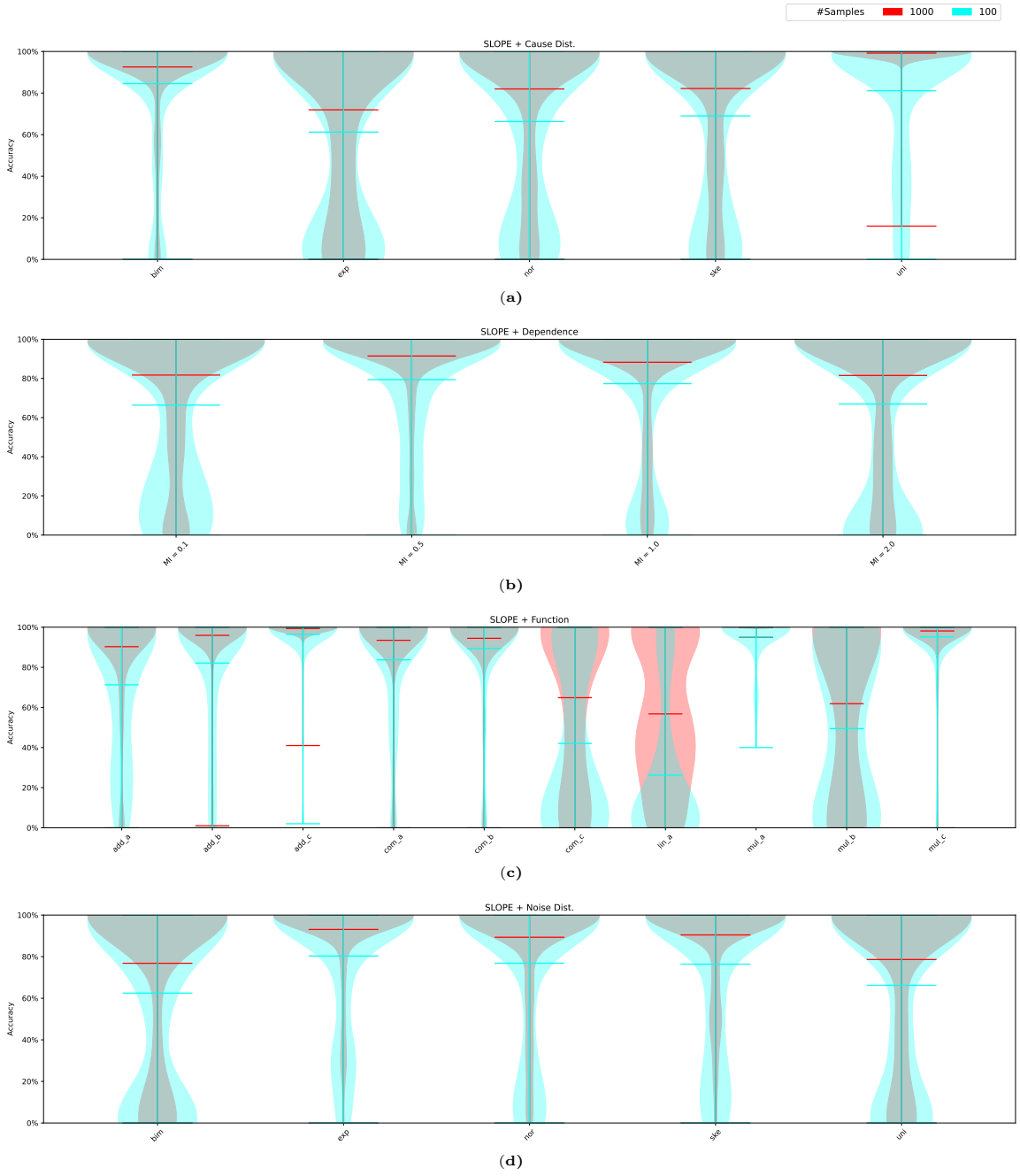



Figure 113: Distribution of accuracies *w.r.t.* individual configurations on our proposed data collection as violins for SLOPE. Please note, the width of the violins is scaled to unit width individually.

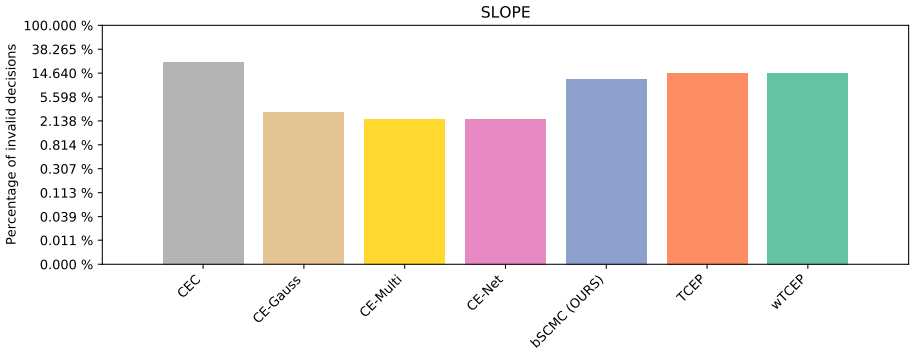



Figure 114: Percentage of invalid decisions for SLOPE. Note that the y-axis is logarithmically scaled.

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| **(1)** bimodalnormal | | | | |
| **(2)** exponential | 1.2e-13 | | | |
| **(3)** normal | 2.3e-09 | 3.6e-02 | | |
| **(4)** skewednormal | 7.0e-06 | 2.1e-03 | 2.4e-01 | |
| **(5)** uniform | 8.5e-01 | 4.6e-15 | 2.2e-10 | 1.4e-06 |

**(a)** cause distribution

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| **(1)** bimodalnormal | | | | |
| **(2)** exponential | 3.6e-05 | | | |
| **(3)** normal | 1.1e-04 | 8.2e-01 | | |
| **(4)** skewednormal | 2.5e-04 | 6.3e-01 | 8.2e-01 | |
| **(5)** uniform | 5.6e-01 | 3.9e-04 | 9.3e-04 | 1.7e-03 |

**(b)** noise distribution

| | (1) | (2) | (3) |
|---|---|---|---|
| **(1)** MI = 0.1 | | | |
| **(2)** MI = 0.5 | 7.8e-04 | | |
| **(3)** MI = 1.0 | 5.1e-05 | 3.9e-01 | |
| **(4)** MI = 2.0 | 5.2e-02 | 3.1e-01 | 5.5e-02 |

**(c)** mutual information

| | (1) |
|---|---|
| **(1)** 1000 | |
| **(2)** 100 | 9.8e-33 |

**(d)** sample size

| | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) |
|---|---|---|---|---|---|---|---|---|---|
| **(1)** add_a | | | | | | | | | |
| **(2)** add_b | 3.3e-02 | | | | | | | | |
| **(3)** add_c | 7.3e-14 | 5.8e-10 | | | | | | | |
| **(4)** com_a | 2.3e-02 | 7.4e-01 | 1.6e-08 | | | | | | |
| **(5)** com_b | 2.3e-08 | 2.4e-05 | 1.3e-01 | 1.5e-04 | | | | | |
| **(6)** com_c | 7.1e-13 | 7.8e-21 | 2.8e-39 | 4.3e-19 | 6.1e-29 | | | | |
| **(7)** lin_a | 1.1e-21 | 1.1e-30 | 5.3e-47 | 8.3e-28 | 1.6e-36 | 8.7e-03 | | | |
| **(8)** mul_a | 3.2e-08 | 3.5e-05 | 1.4e-01 | 2.0e-04 | 9.2e-01 | 1.6e-27 | 1.0e-34 | | |
| **(9)** mul_b | 2.1e-06 | 1.9e-10 | 3.5e-24 | 6.1e-10 | 2.2e-17 | 7.0e-01 | 2.2e-02 | 7.3e-17 | |
| **(10)** mul_c | 1.2e-08 | 2.2e-05 | 1.4e-01 | 1.3e-04 | 9.2e-01 | 1.6e-28 | 1.1e-35 | 9.7e-01 | 2.4e-17 |

**(f)** functional dependence

Figure 115: Significances at $\alpha = 0.05$-level *w.r.t.* individual configurations for SLOPE. The numbering of the row and column heads correspond to each other. Green color coding indicates significant differences while red color coding indicates no significant difference.

## D.14 Identifiability of Cause and Effect using Regularized Regression

### D.14.1 Description

The authors of (Marx and Vreeken, 2019) extend on the idea to fit a regression model in both possible causal directions and decide according to the regression residuals. In their work, the authors argue that the true complexity of the functional dependence is not known and hence, the decision might be obtained using under- or overfitted regression models. To overcome this, the authors propose to rely on regularized regression which allows for comparing models with different complexities. This approach is derived under the Kolmogorov complexity framework, which states that the best anti-causal model needs at least as many parameters as the true causal one. The method is based on an Identifiable Regression-based Scoring Function (IRSF), a function that assigns scores based on the model fit and the used number of parameters. The authors use two different implementations of their approach, one based on a set of different basis functions and one on cubic splines, while the latter is used for their evaluation. As a scoring function the Akaike information criterion (AIC, Akaike (1983)) and the Bayesian information criterion (BIC, Schwarz et al. (1978)) are used. This method is called *SLOPPY*[21] due to its connection to the SLOPE criterion (see Appendix D.13), *i.e.,* scoring fit and complexity, that was proposed by the same authors.

### D.14.2 Results

Results obtained for SLOPPY can be found in Figs. 116 to 122. As stated in the introduction of this method, it is closely connected to SLOPE and thus, it is not that surprising that the obtained results appear to be quite similarly (compare Fig. 109). In general, we see accuracies between $\sim 41\%$ using the ■ CEC13 data to $\sim 90\%$ on the ■ CE-Multi data (see Fig. 116.a), which is marginally higher than those of SLOPE. We attribute this improvement *w.r.t.* SLOPE mainly to the increased accuracies in the small sample regime (*i.e.,* ■ 100 samples per realization). Since both methods, *i.e.,* SLOPE and SLOPPY, perform very similar, and roughly rely on the same idea, we draw the same conclusion to interpret the shown behavior (see Appendix D.13.2). Certain configurations of functional dependency, mutual information, cause and noise distribution lead to data where a simpler regression models with better fit in anti-causal direction than in causal direction can be found (while the family of models is restricted by the methods). Please note, a better fit in one direction does not necessarily mean a good fit (the fit in the opposite direction could just be even worse). This is also in line with our observations regarding RECI in Fig. 102. Due to the similarity of results, we see roughly the same significances again in Fig. 122 and Fig. 118 as well as matching patterns among the violins in Fig. 120. Finally, depending on the data set, the method produces a number of invalid decisions (see Fig. 121).

21. In order to perform our experiments, we rely on the code provided by the authors (see `eda.mmci.uni-saarland.de/prj/sloppy/`) including the default choice of involved parameters. Before applying the method, we normalize the data.

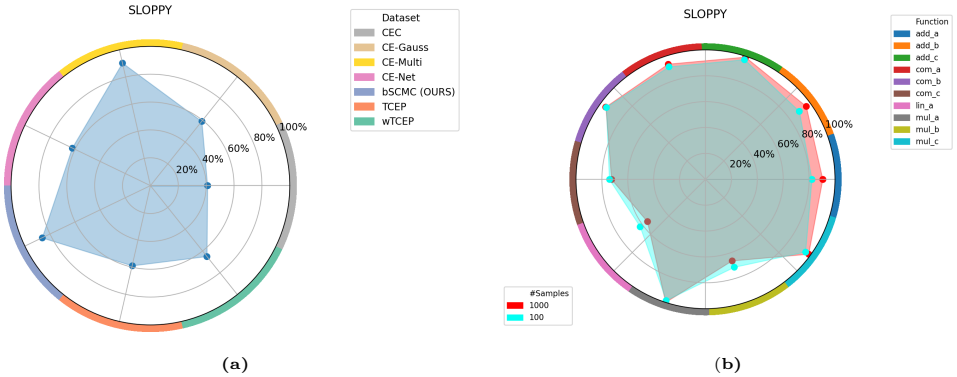



Figure 116: Result footprints for SLOPPY.

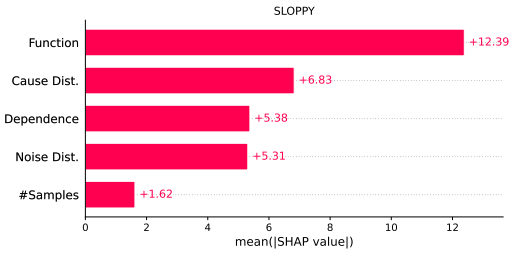



Figure 117: Importance of individual data set characteristics as mean Shapley values for SLOPPY.

| Method | Setting | p-Value |
|---|---|---|
| SLOPPY | Cause Dist. ⫫ Accuracy \| [#Samples, Dependence, Function, Noise Dist.] | 3.0e-41 |
| SLOPPY | #Samples ⫫ Accuracy \| [Cause Dist., Dependence, Function, Noise Dist.] | 3.6e-01 |
| SLOPPY | Dependence ⫫ Accuracy \| [Cause Dist., #Samples, Function, Noise Dist.] | 6.3e-13 |
| SLOPPY | Function ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Noise Dist.] | 1.7e-102 |
| SLOPPY | Noise Dist. ⫫ Accuracy \| [Cause Dist., #Samples, Dependence, Function] | 3.6e-21 |

Figure 118: Significances at $\alpha = 0.05$-level for conditional independencies of individual data set characteristics *w.r.t.* the obtained accuracy for SLOPPY. Green color coding indicates dependence while red color coding indicates independence.

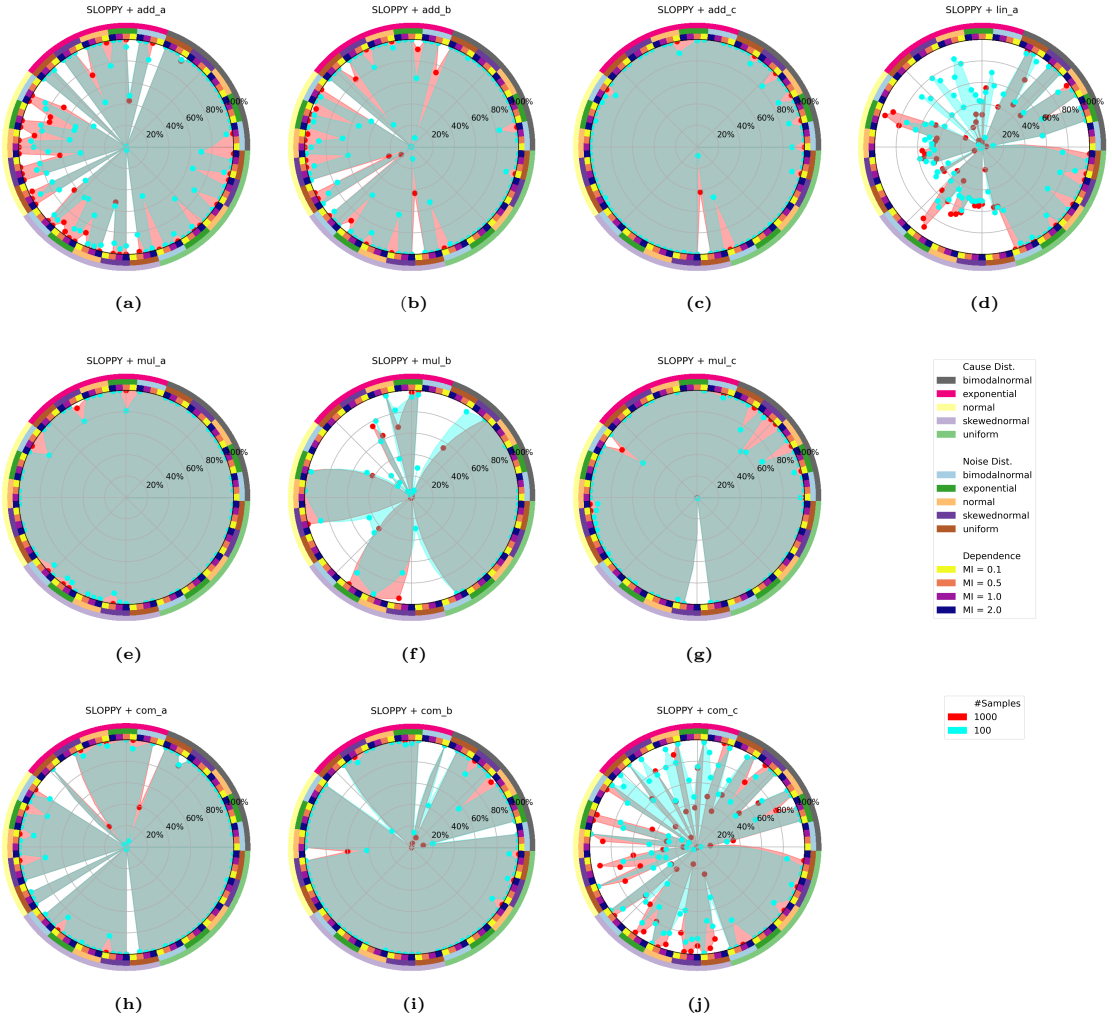



Figure 119: Detailed result overview on our bSCMC data as footprints for SLOPPY.

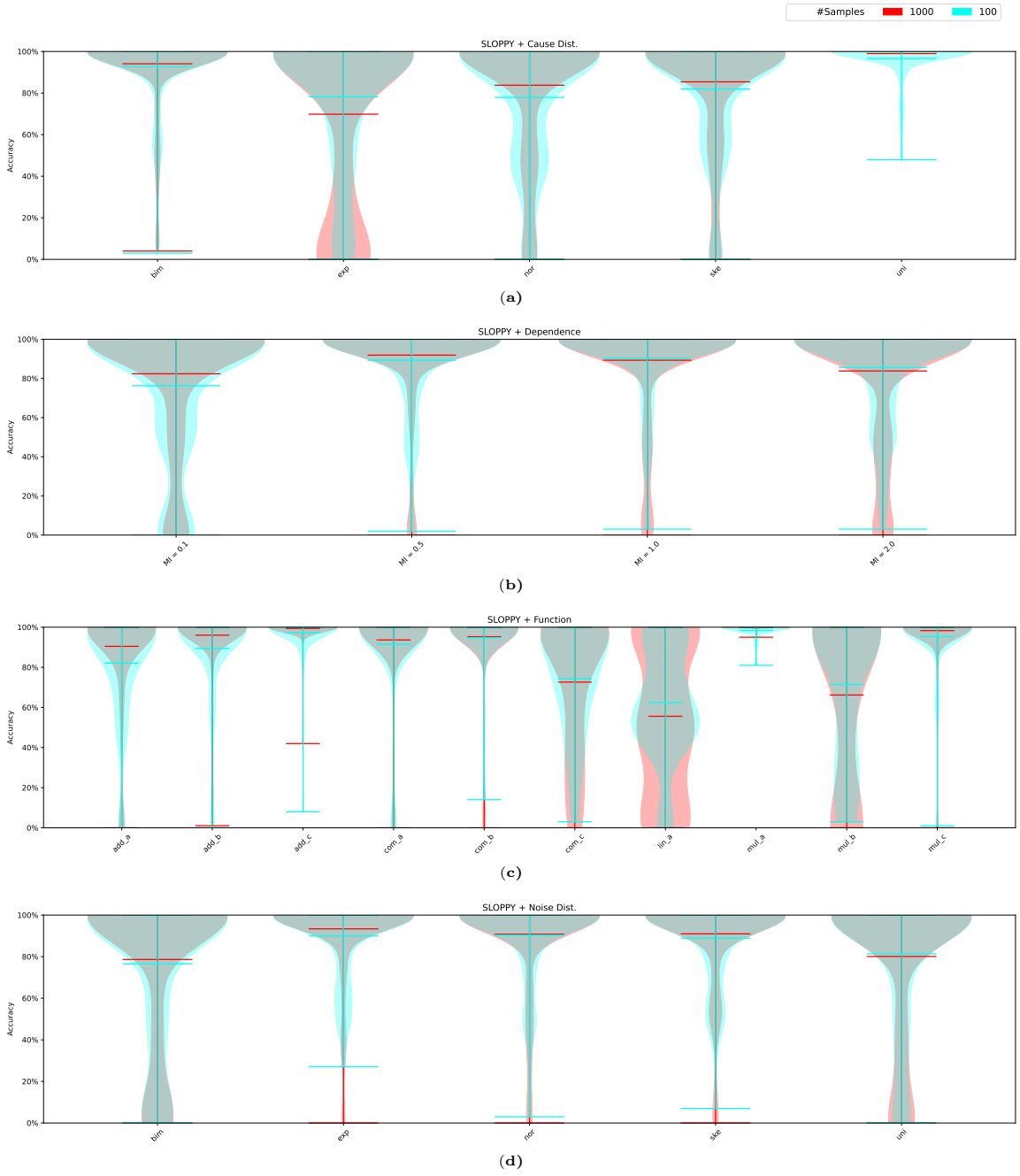



Figure 120: Distribution of accuracies *w.r.t.* individual configurations on our proposed data collection as violins for SLOPPY. Please note, the width of the violins is scaled to unit width individually.

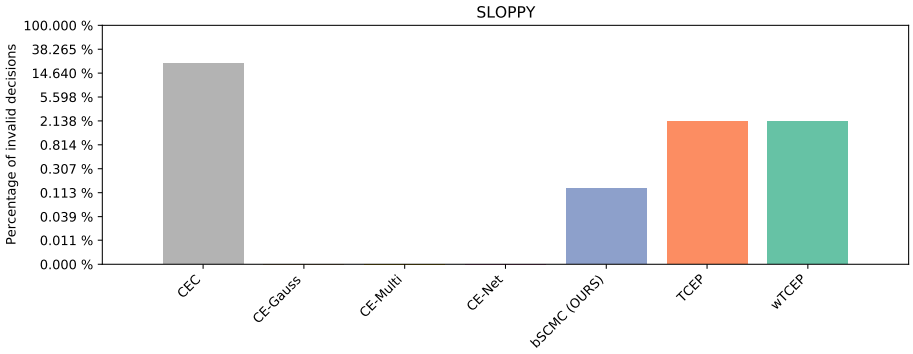



Figure 121: Percentage of invalid decisions for SLOPPY. Note that the y-axis is logarithmically scaled.

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 8.0e-15 | | | |
| (3) normal | 2.0e-11 | 8.4e-02 | | |
| (4) skewednormal | 4.8e-10 | 2.7e-02 | 5.1e-01 | |
| (5) uniform | 1.1e-04 | 2.6e-27 | 5.0e-24 | 1.3e-22 |

**(a)** cause distribution

| | (1) | (2) | (3) | (4) |
|---|---|---|---|---|
| (1) bimodalnormal | | | | |
| (2) exponential | 1.2e-04 | | | |
| (3) normal | 3.4e-04 | 7.8e-01 | | |
| (4) skewednormal | 1.6e-03 | 4.1e-01 | 6.0e-01 | |
| (5) uniform | 7.8e-01 | 2.5e-04 | 6.7e-04 | 3.4e-03 |

**(b)** noise distribution

| | (1) | (2) | (3) |
|---|---|---|---|
| (1) MI = 0.1 | | | |
| (2) MI = 0.5 | 1.3e-05 | | |
| (3) MI = 1.0 | 1.5e-06 | 5.7e-01 | |
| (4) MI = 2.0 | 3.9e-03 | 1.7e-01 | 6.8e-02 |

**(c)** mutual information

| | (1) |
|---|---|
| (1) 1000 | |
| (2) 100 | 3.4e-13 |

**(d)** sample size

| | (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) |
|---|---|---|---|---|---|---|---|---|---|
| (1) add_a | | | | | | | | | |
| (2) add_b | 9.5e-05 | | | | | | | | |
| (3) add_c | 1.6e-13 | 9.8e-05 | | | | | | | |
| (4) com_a | 5.1e-07 | 1.3e-01 | 2.5e-02 | | | | | | |
| (5) com_b | 4.9e-10 | 4.4e-03 | 3.6e-01 | 1.9e-01 | | | | | |
| (6) com_c | 1.1e-07 | 2.3e-18 | 2.9e-31 | 3.5e-20 | 7.3e-25 | | | | |
| (7) lin_a | 8.2e-18 | 1.7e-29 | 2.3e-41 | 6.8e-29 | 3.2e-34 | 4.6e-05 | | | |
| (8) mul_a | 2.6e-10 | 1.3e-03 | 8.1e-01 | 7.7e-02 | 5.4e-01 | 3.2e-24 | 2.2e-32 | | |
| (9) mul_b | 3.2e-03 | 9.9e-09 | 1.2e-17 | 2.0e-10 | 5.0e-14 | 7.1e-01 | 8.5e-03 | 8.1e-14 | |
| (10) mul_c | 1.1e-07 | 5.7e-02 | 6.8e-02 | 7.2e-01 | 3.8e-01 | 3.2e-21 | 2.3e-30 | 1.6e-01 | 2.7e-11 |

**(f)** functional dependence

Figure 122: Significances at $\alpha = 0.05$-level *w.r.t.* individual configurations for SLOPPY. The numbering of the row and column heads correspond to each other. Green color coding indicates significant differences while red color coding indicates no significant difference.

## Appendix E. Missing Methods

Although we aimed to give a broad overview of state-of-the-art methods, the list of methods available is even larger. Thus, we want to point the interested reader toward `causeme.net`, an interactive and growing benchmark platform where an extended version of the in this manuscript proposed data collection including results for more methods will be hosted. Besides bivariate causal discovery, this platform also provides extensive comparisons for causal discovery within time series.

During our investigation, we experimented with a larger number of methods than those discussed in this study. Those methods are not part of this manuscript due to various reasons (*e.g.,* technical or applicability issues, lack of publicly available code and non-responding authors, own implementations that do not match reported results, resource and time requirements for evaluating all these methods, *etc.*). A non-exhaustive list of further work introducing, refining or evaluating concepts is (Goldfarb and Evans, 2019; Bühlmann et al., 2014; Cai et al., 2019; Aragam and Zhou, 2015; Budhathoki and Vreeken, 2017; Sgouritsa et al., 2015; Bontempi and Flauder, 2015; Gnecco et al., 2019; Monti et al., 2020; Hernandez-Lobato et al., 2016; Mitrovic et al., 2018; Chen et al., 2019; Lopez-Paz et al., 2017; Hyvärinen and Smith, 2013; Monti et al., 2019; Zheng et al., 2018; Kalainathan et al., 2018; Zhu and Chen, 2019; Jørgensen and Hauberg, 2020; Ton et al., 2020; Peters et al., 2014; Shimizu and Bollen, 2014; Lopez-Paz and Oquab, 2016; Janzing et al., 2009a; Friedman and Nachman, 2013; Kano et al., 2003; Sun et al., 2006; Mooij et al., 2009, 2003; Janzing et al., 2012a; Peters and Bühlmann, 2014; Nowzohour and Bühlmann, 2016; Vowels et al., 2021; Janzing et al., 2009b; Galanti et al., 2020; Tagasovska et al., 2020; Janzing and Schölkopf, 2010).

## Appendix F. Densities

In Figs. 123 to 142 we depict the densities $P(X, Y)$ that our realizations are sampled from for each individual configuration.

We obtained these densities as follows. Each valid configuration was represented by 10 000 samples (compare Section 4) while the underlying model is always $X \dashrightarrow Y$ for visualization purposes. The data may contain outliers due to the very diverse settings and especially due to the exponential distributions. This is in general something a method for bivariate causal discovery should be able to handle. However, it does make the plotting of the densities infeasible. Thus, we removed outliers using *sklearn.neighbors.LocalOutlierFactor*. The density of the remaining data points was then computed by a kernel density estimator using *scipy.stats.gaussian_kde*. An offset of 0.25 times the actual range of values was added for better visualization. The actual plots are generated with *matplotlib*'s *contourf*.

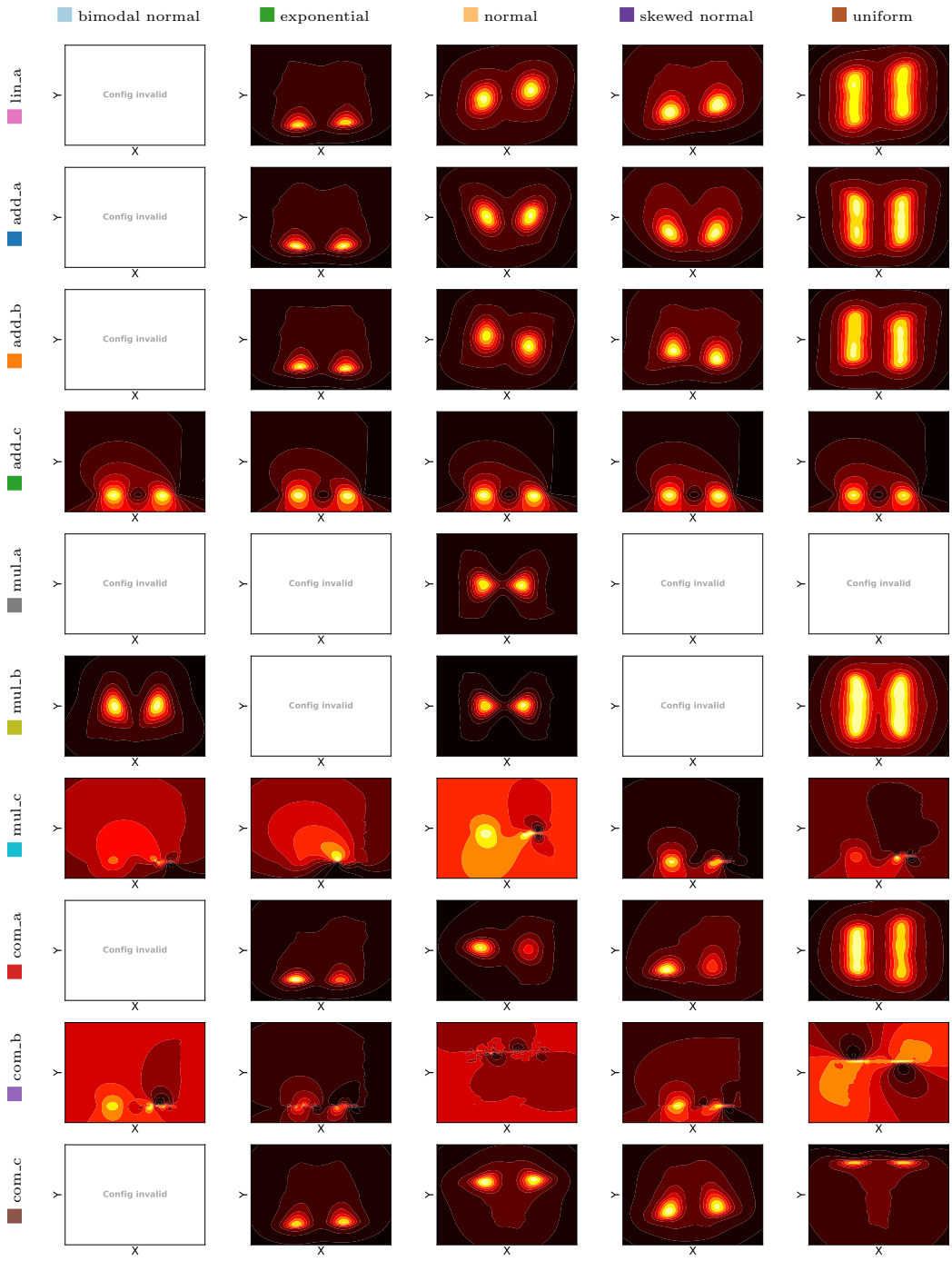



Figure 123: Density contour plot considering a ■ bimodal normal cause distribution and a dependency strength of ■ MI = 0.1. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

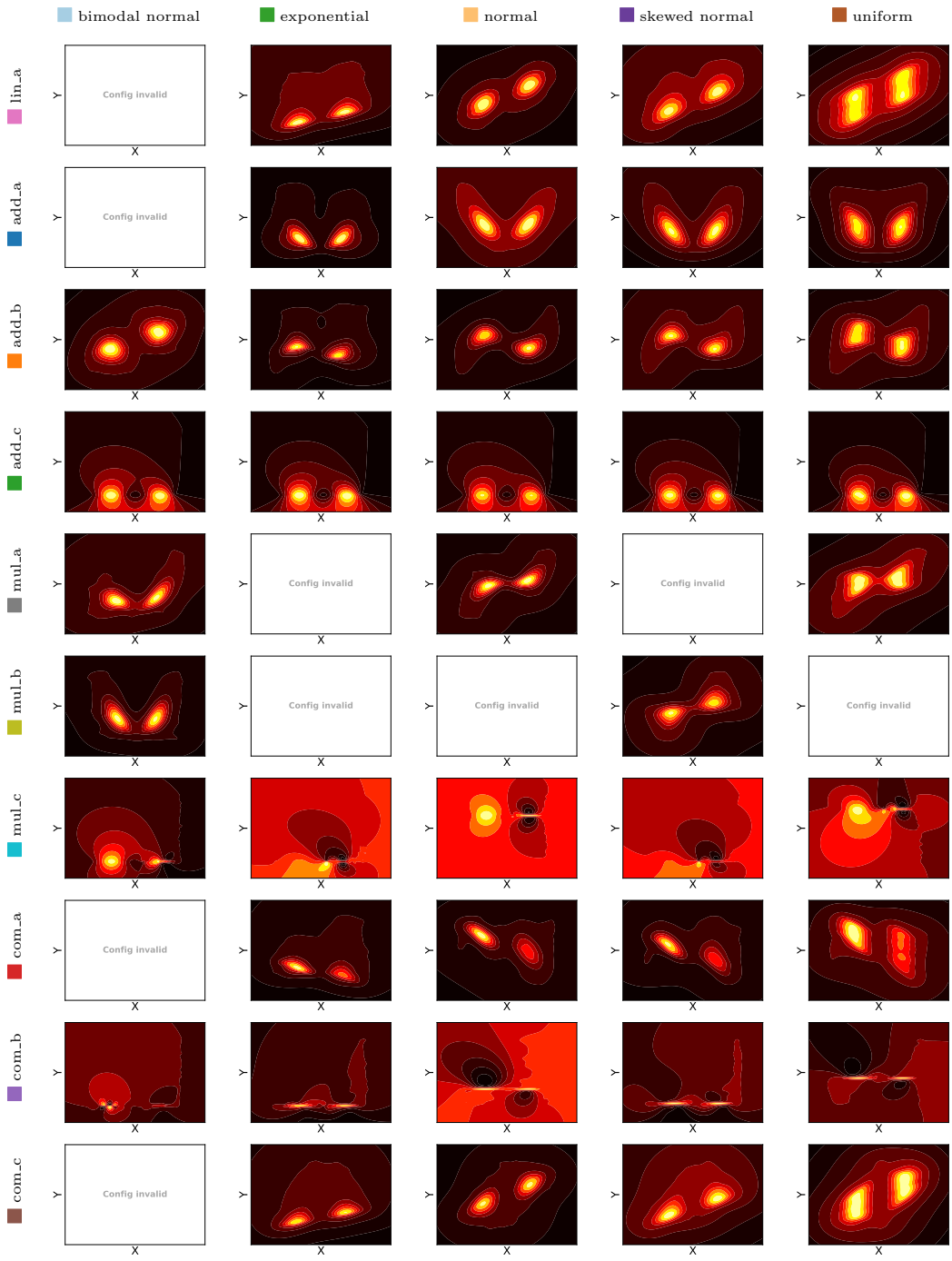



Figure 124: Density contour plot considering a ■ bimodal normal cause distribution and a dependency strength of ■ MI = 0.5. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

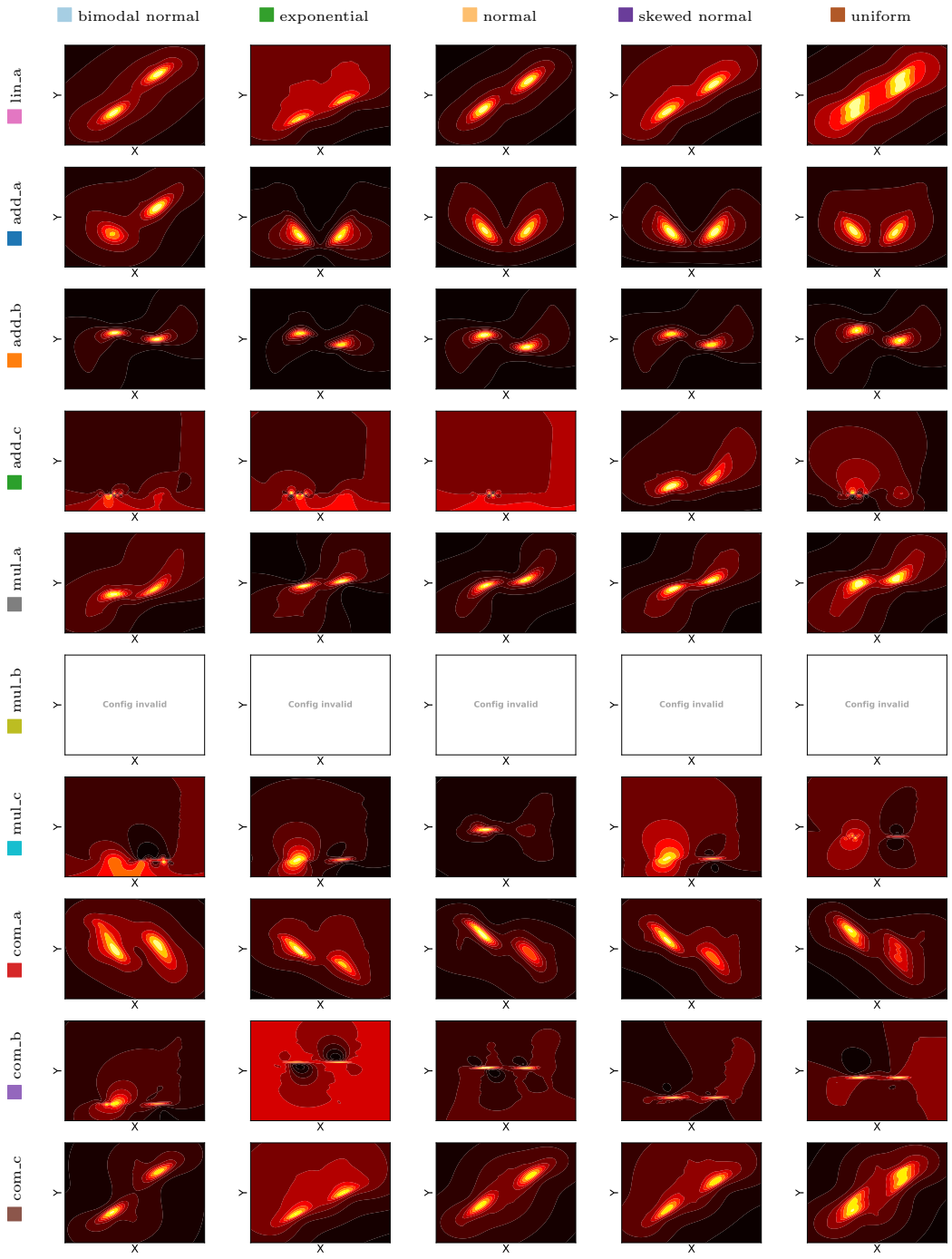



Figure 125: Density contour plot considering a ■ bimodal normal cause distribution and a dependency strength of ■ MI = 1.0. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

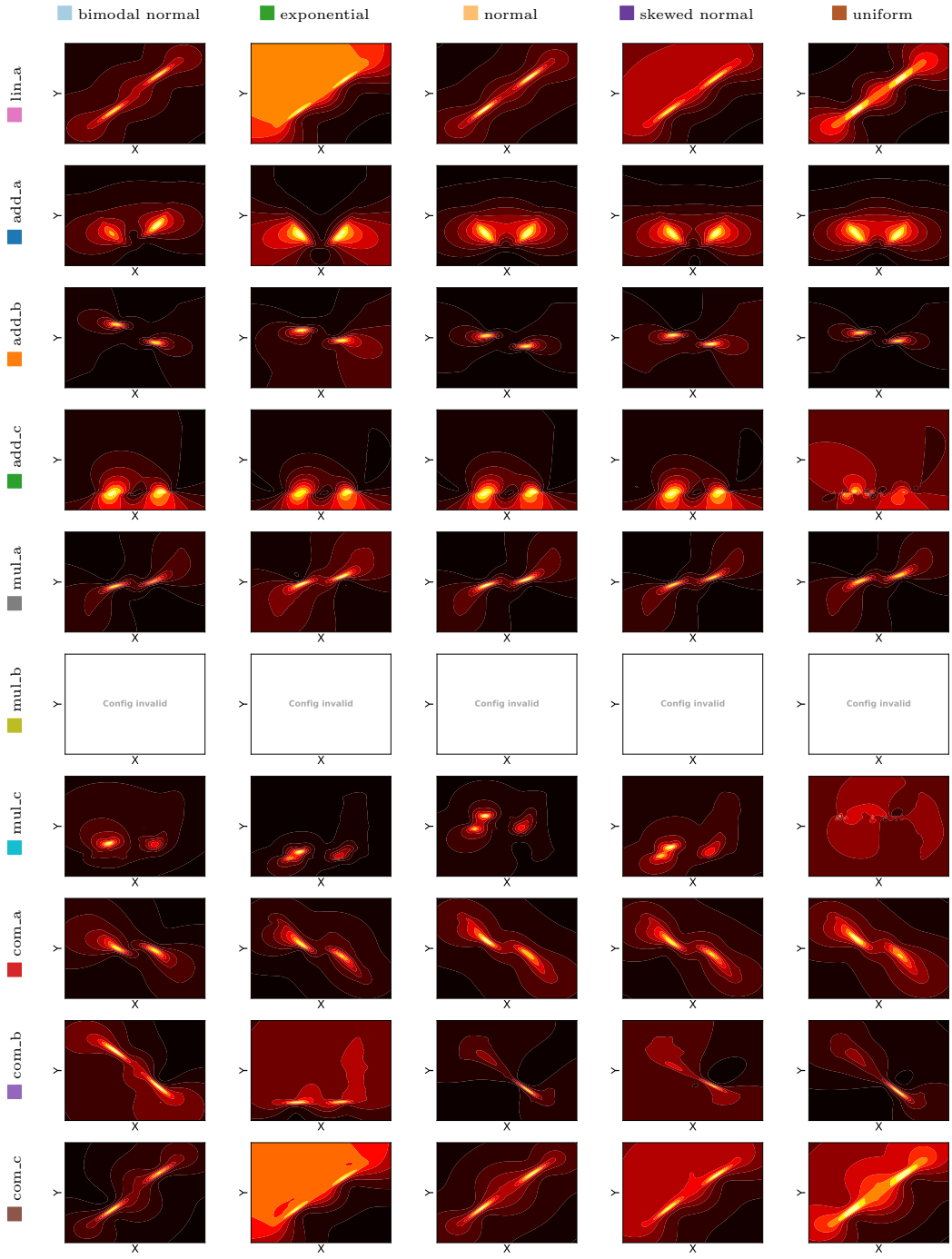



Figure 126: Density contour plot considering a ■ bimodal normal cause distribution and a dependency strength of ■ MI = 2.0. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

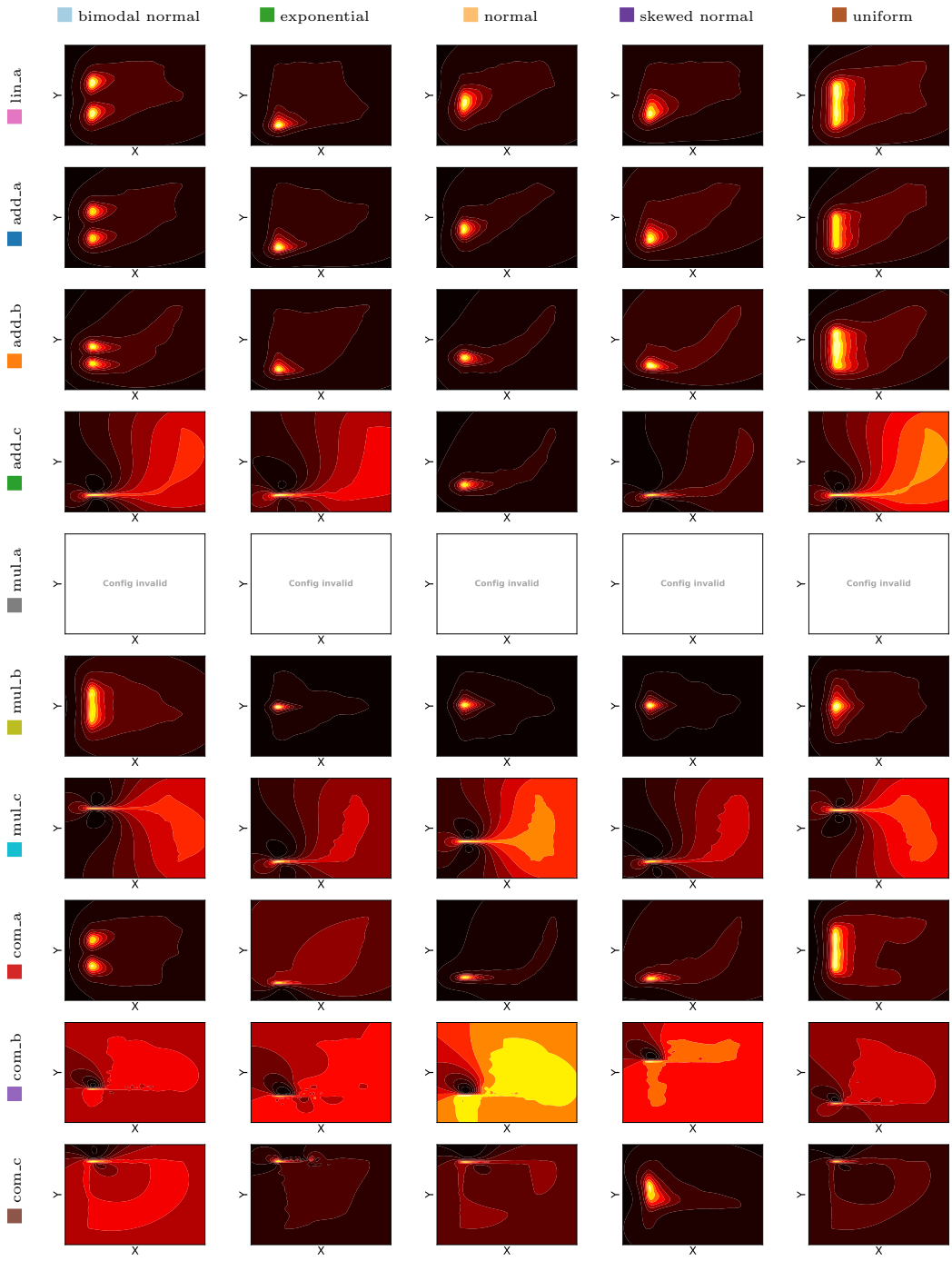



Figure 127: Density contour plot considering a exponential cause distribution and a dependency strength of MI = 0.1. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

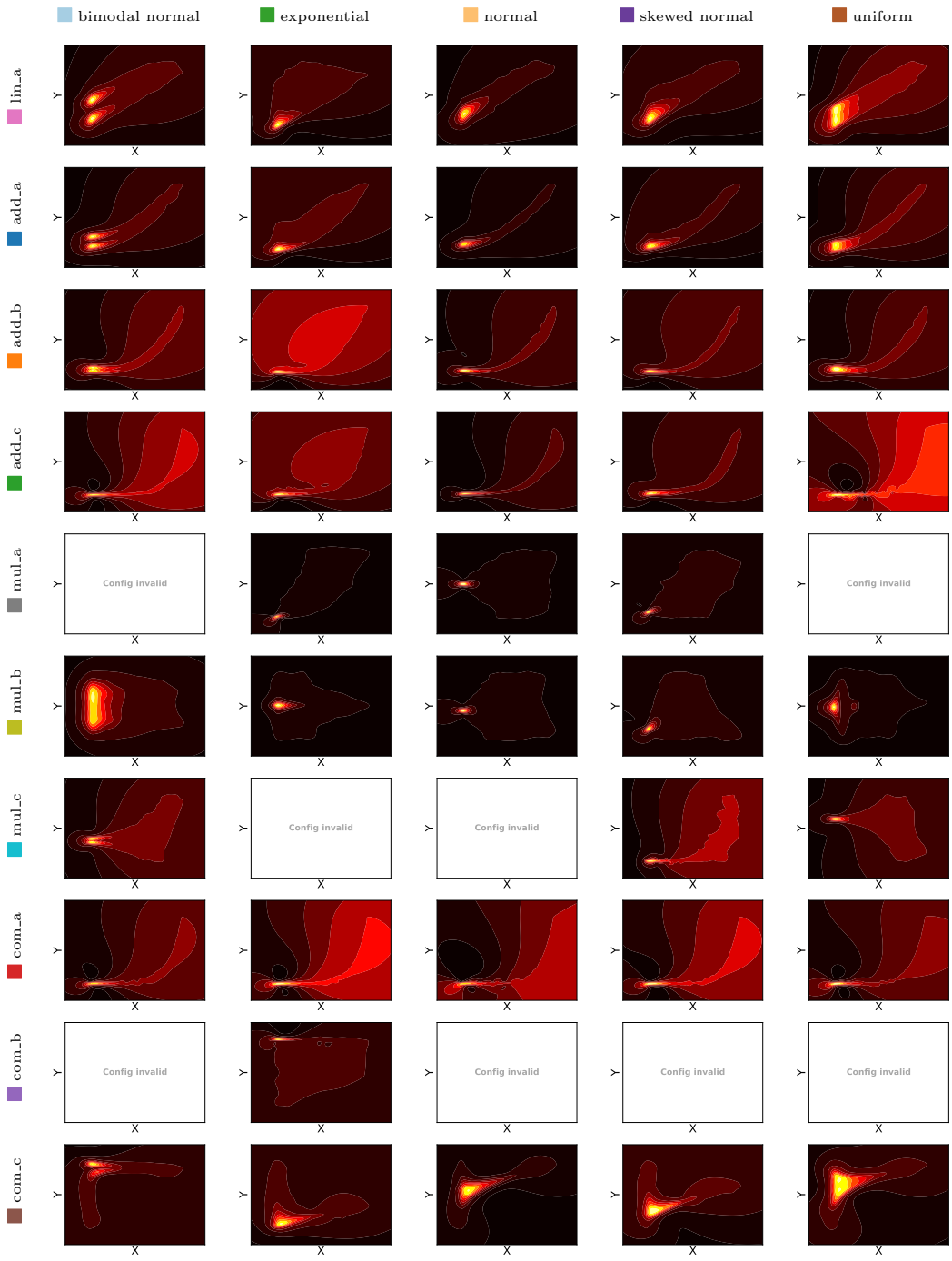



Figure 128: Density contour plot considering a exponential cause distribution and a dependency strength of MI = 0.5. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

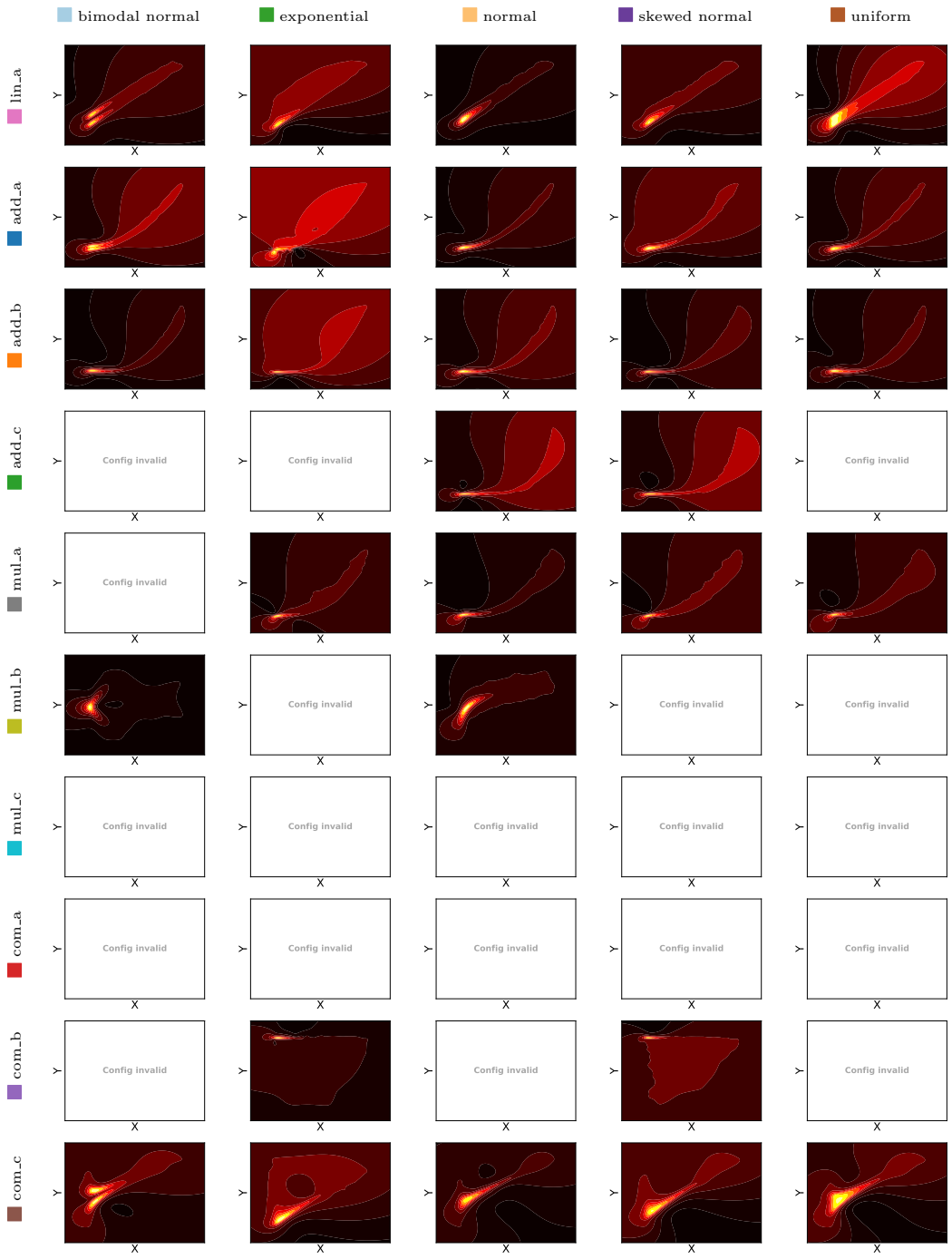



Figure 129: Density contour plot considering a exponential cause distribution and a dependency strength of MI = 1.0. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

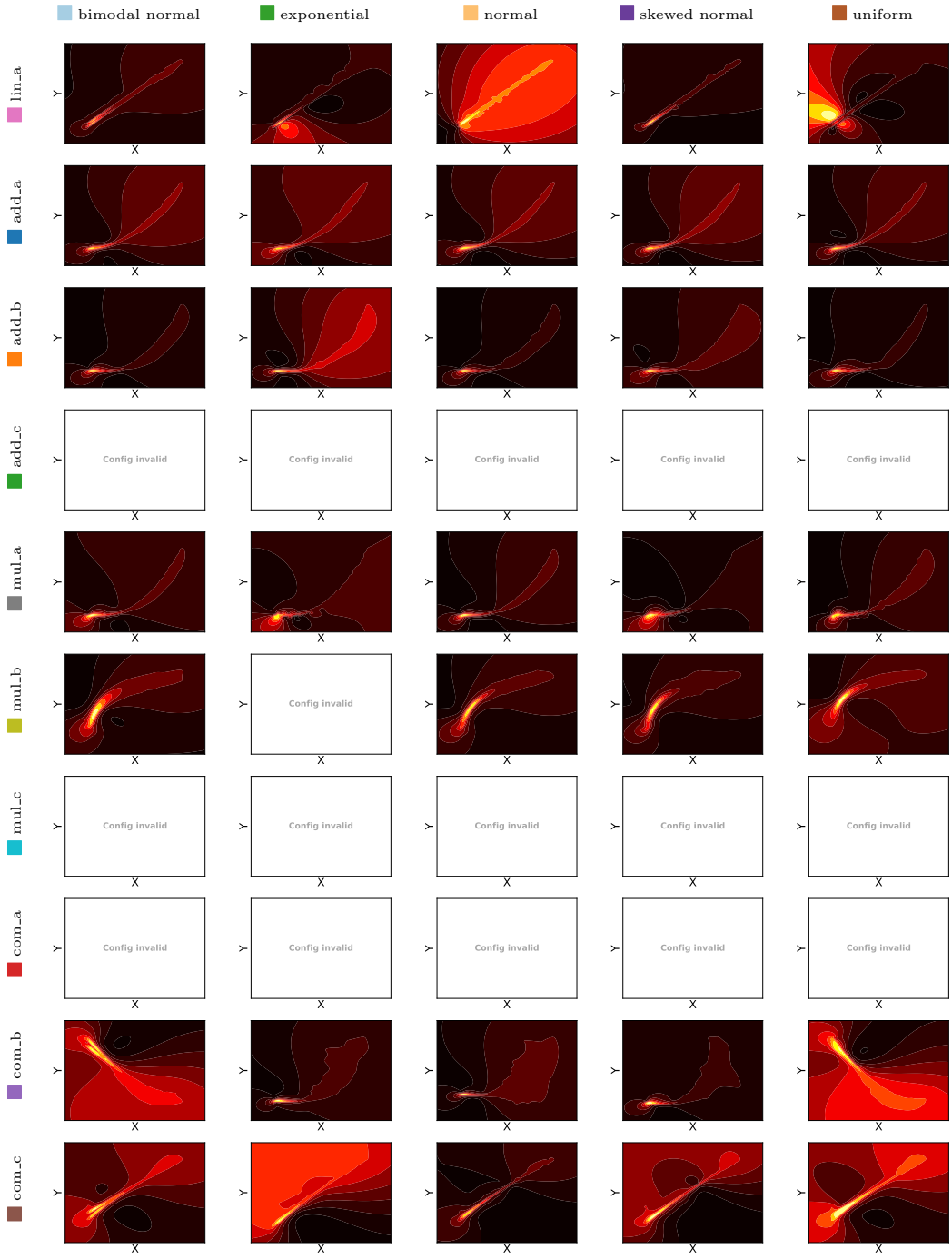



Figure 130: Density contour plot considering a exponential cause distribution and a dependency strength of MI = 2.0. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

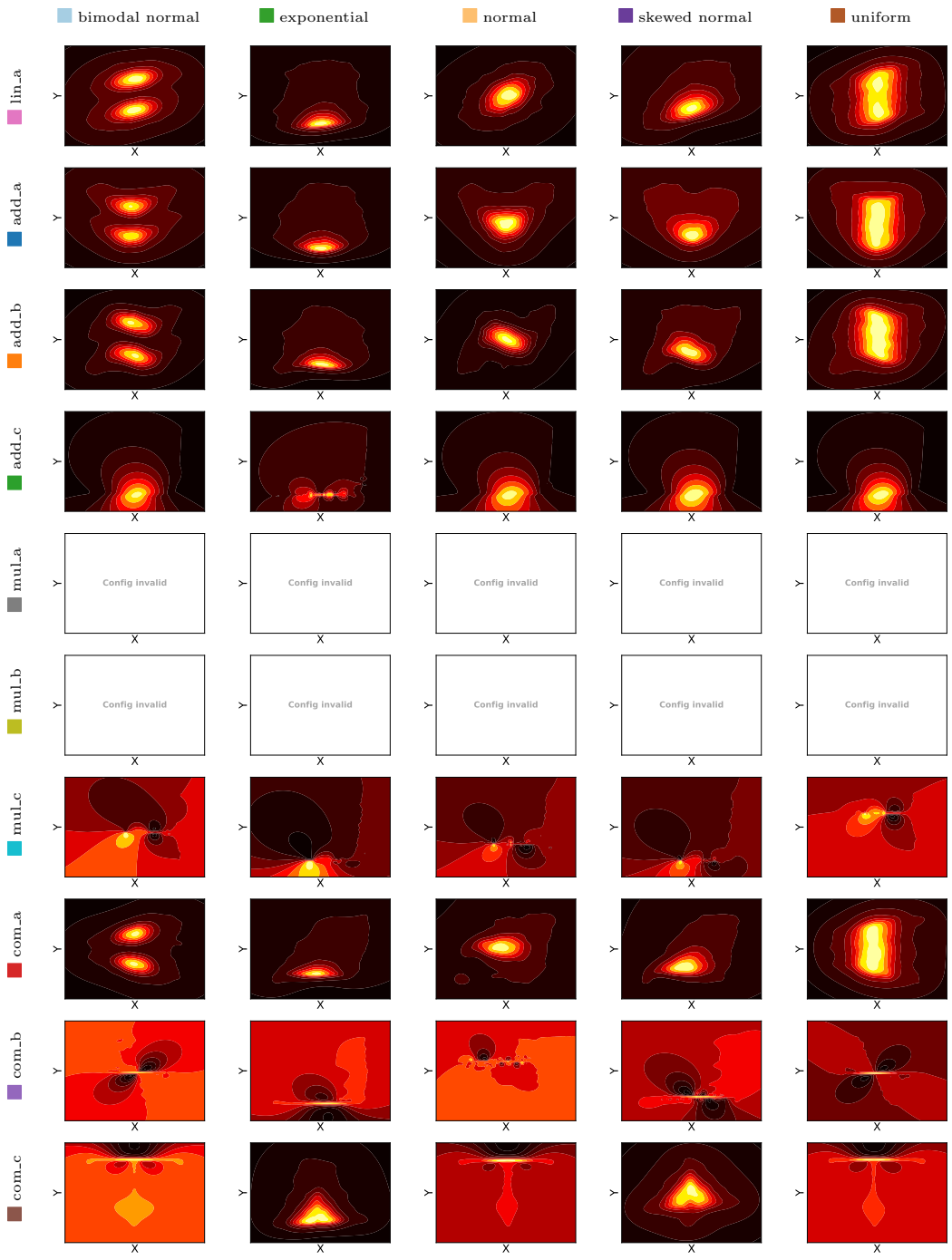



Figure 131: Density contour plot considering a normal cause distribution and a dependency strength of MI = 0.1. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

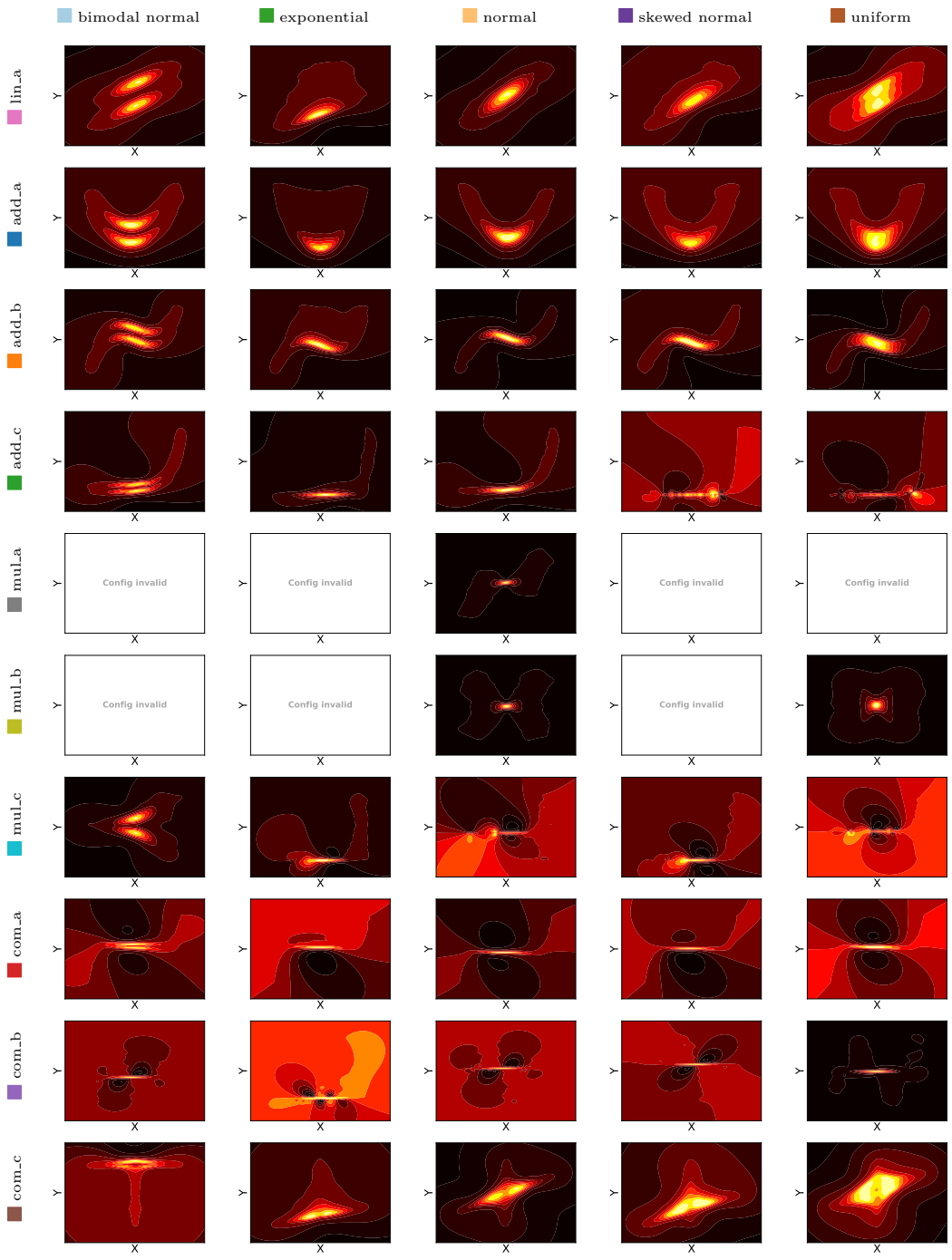



Figure 132: Density contour plot considering a normal cause distribution and a dependency strength of MI = 0.5. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

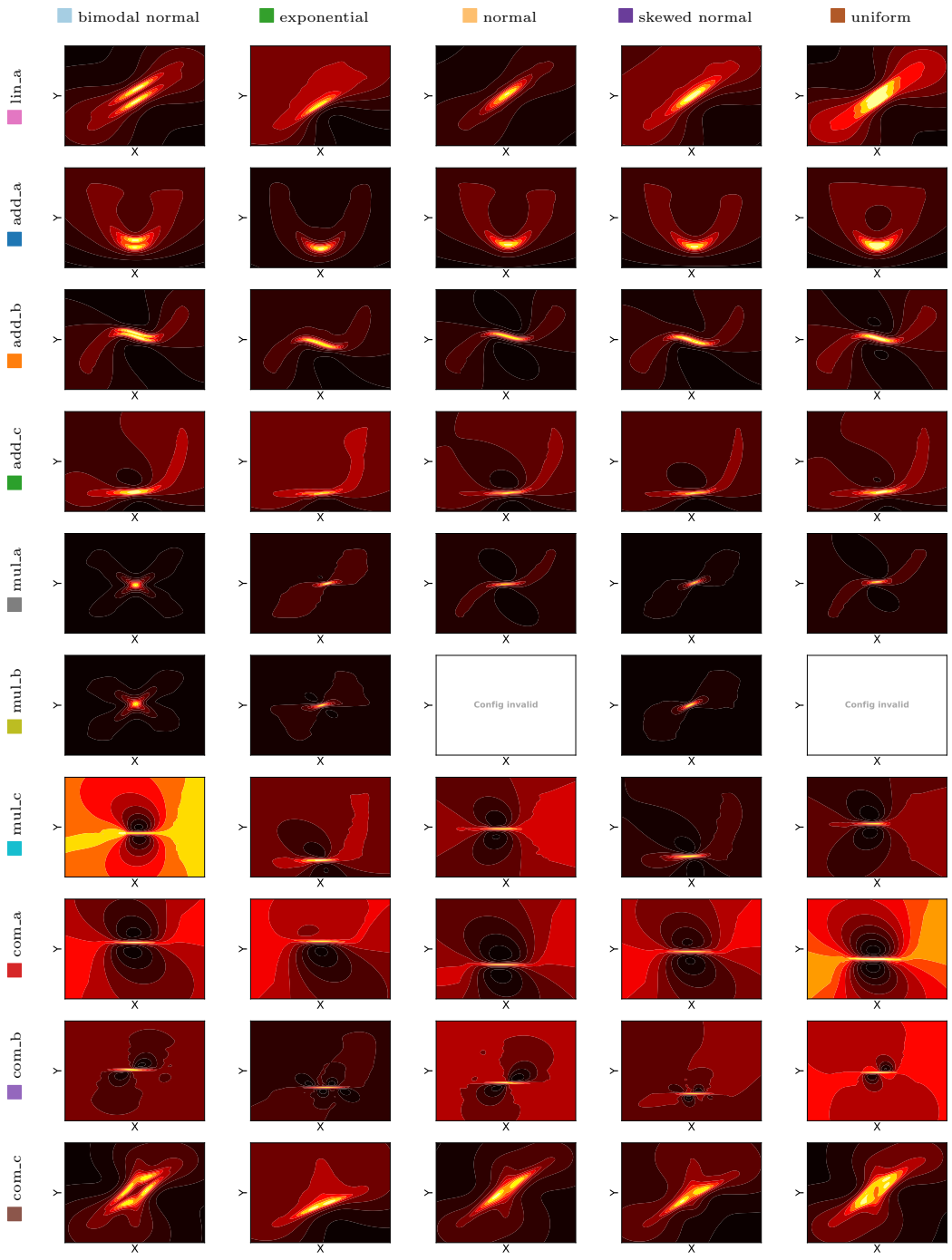



Figure 133: Density contour plot considering a normal cause distribution and a dependency strength of MI = 1.0. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

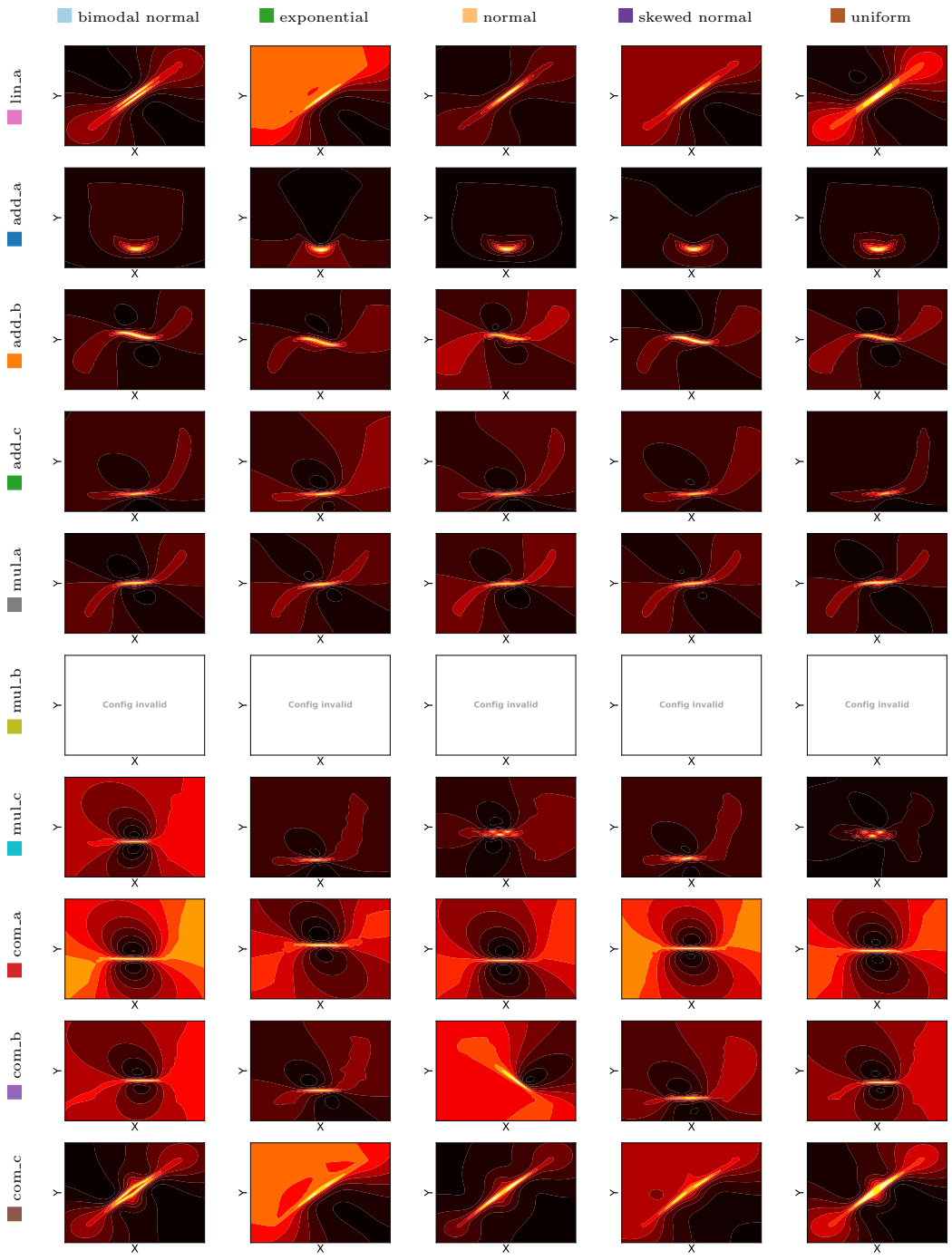



Figure 134: Density contour plot considering a normal cause distribution and a dependency strength of MI = 2.0. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

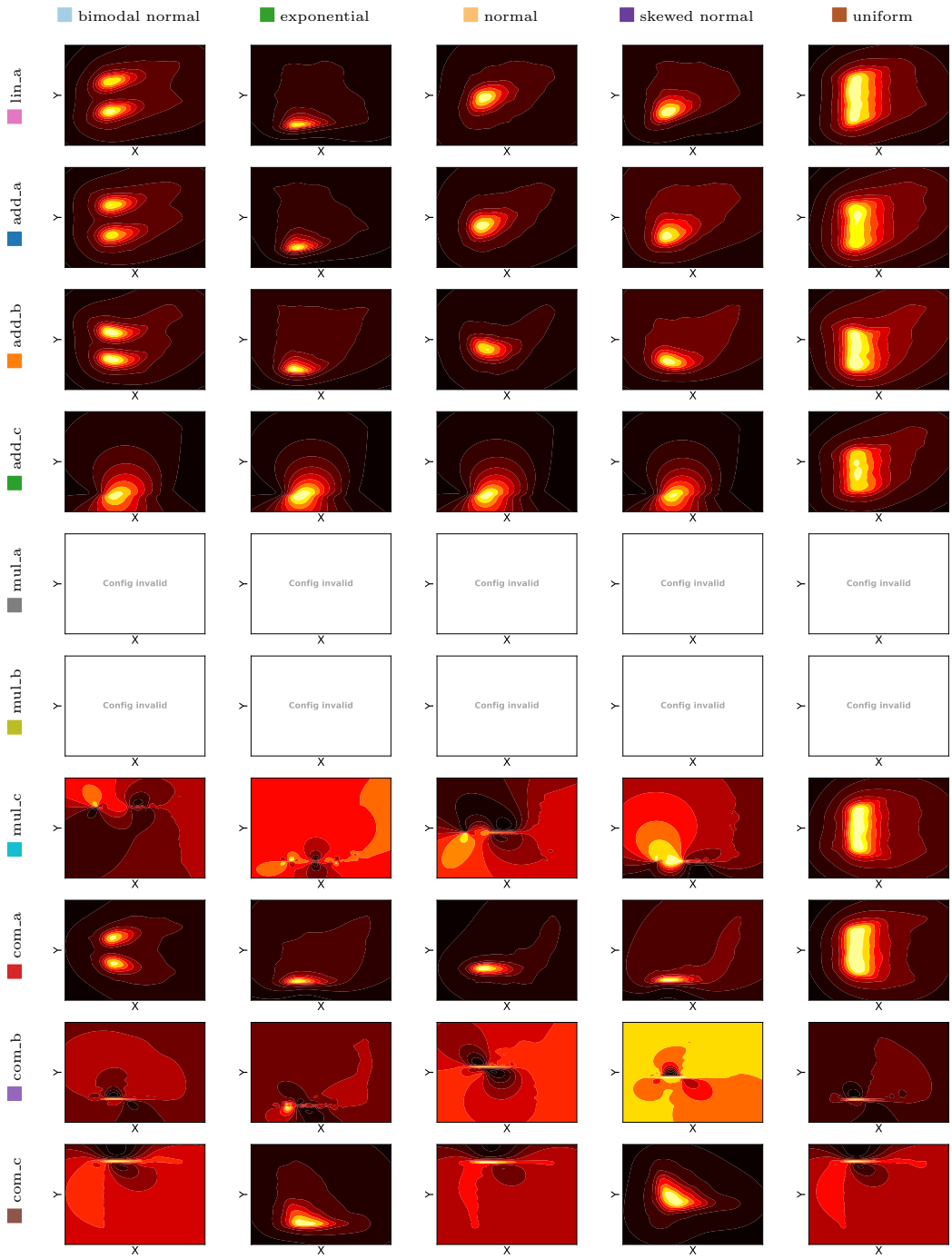



Figure 135: Density contour plot considering a skewed normal cause distribution and a dependency strength of MI = 0.1. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

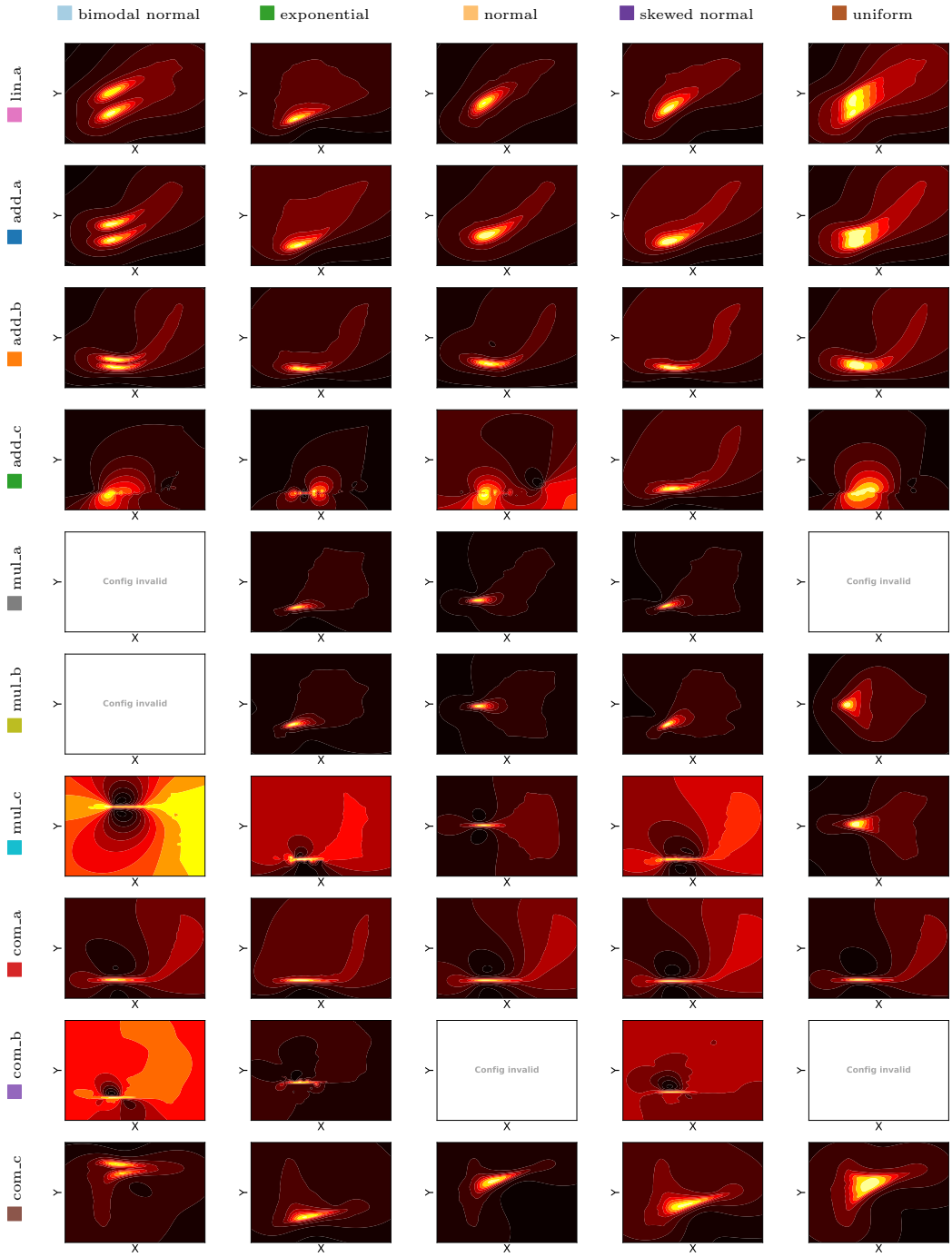



Figure 136: Density contour plot considering a skewed normal cause distribution and a dependency strength of MI = 0.5. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

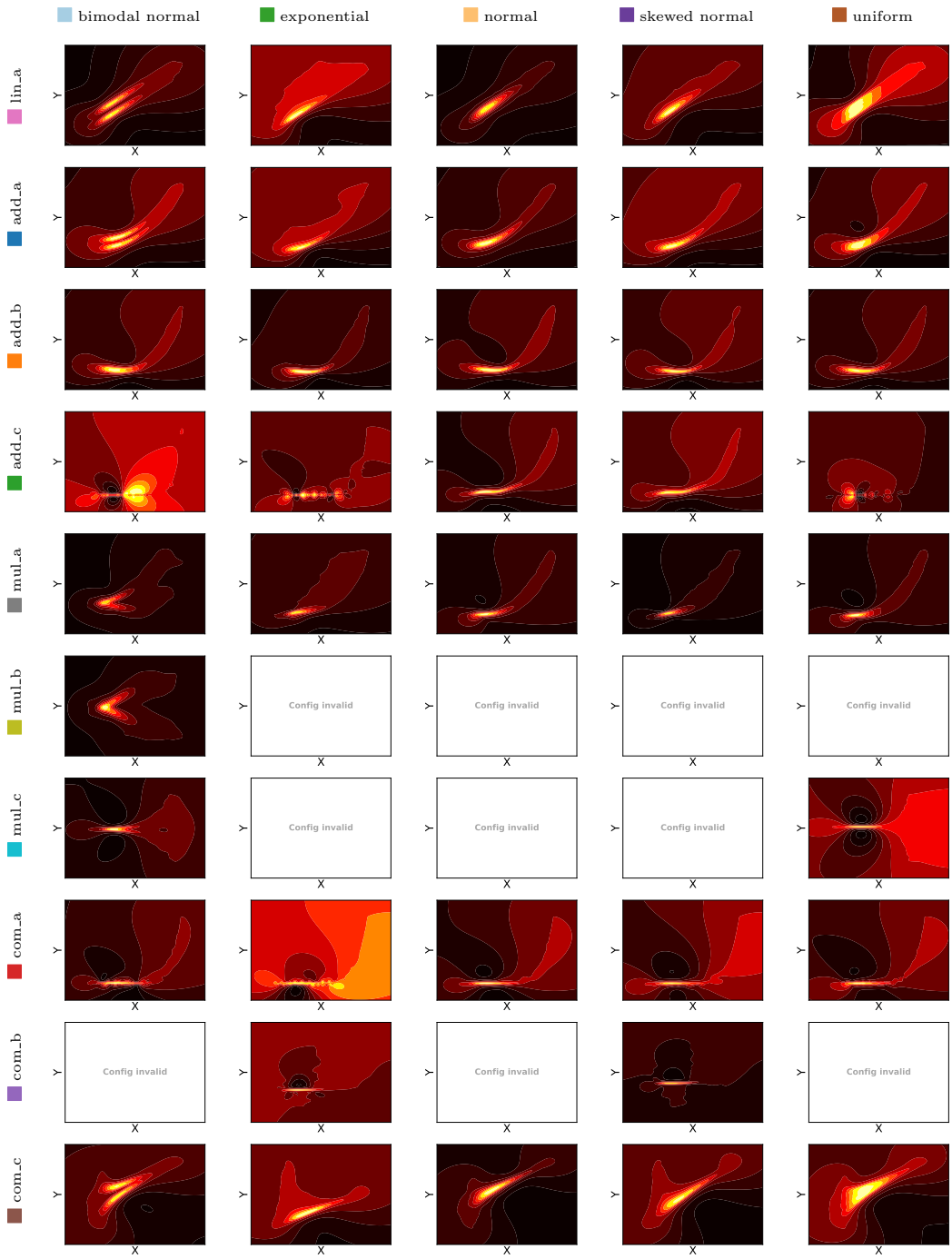



Figure 137: Density contour plot considering a skewed normal cause distribution and a dependency strength of MI = 1.0. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

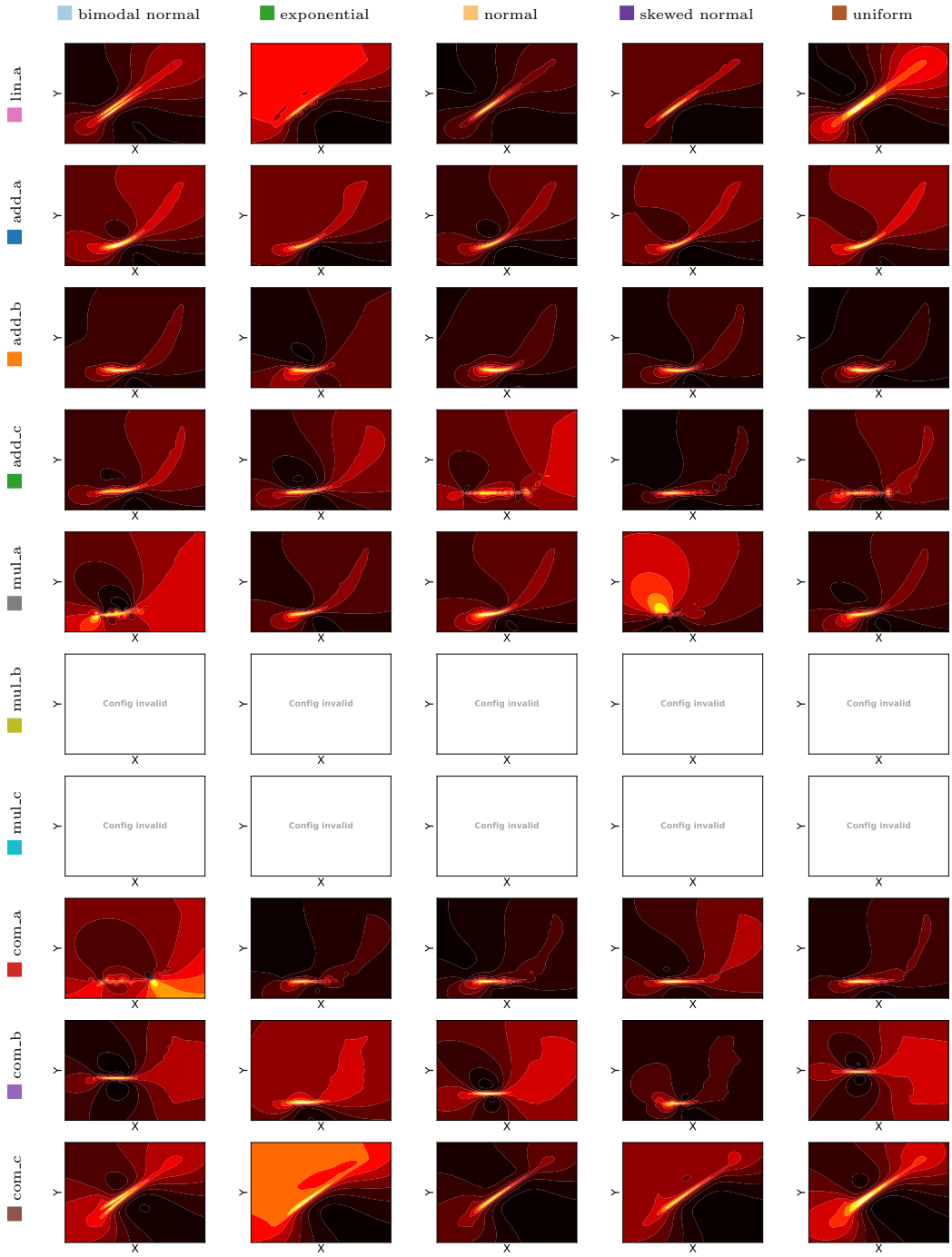



Figure 138: Density contour plot considering a skewed normal cause distribution and a dependency strength of MI = 2.0. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

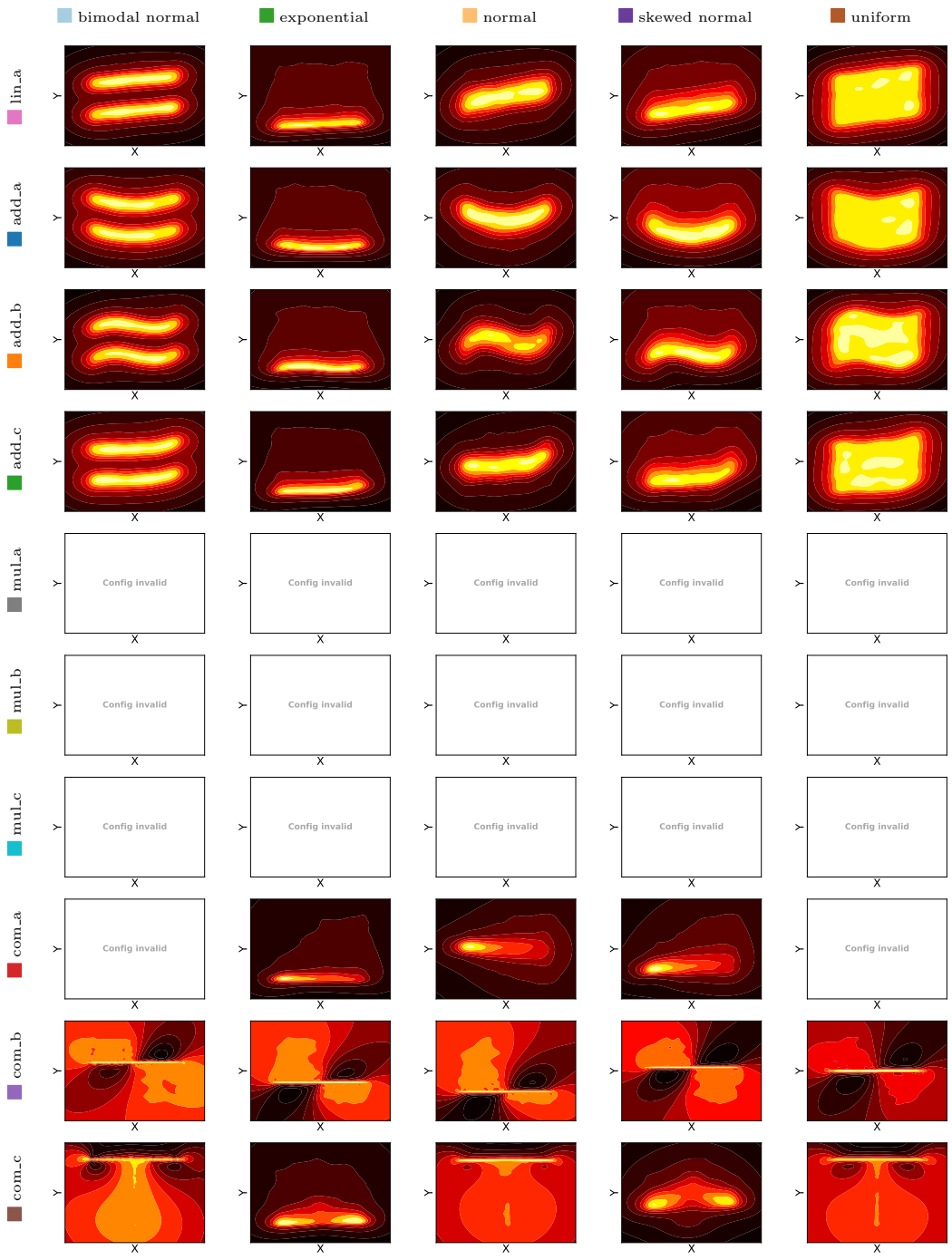



Figure 139: Density contour plot considering a uniform cause distribution and a dependency strength of MI = 0.1. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

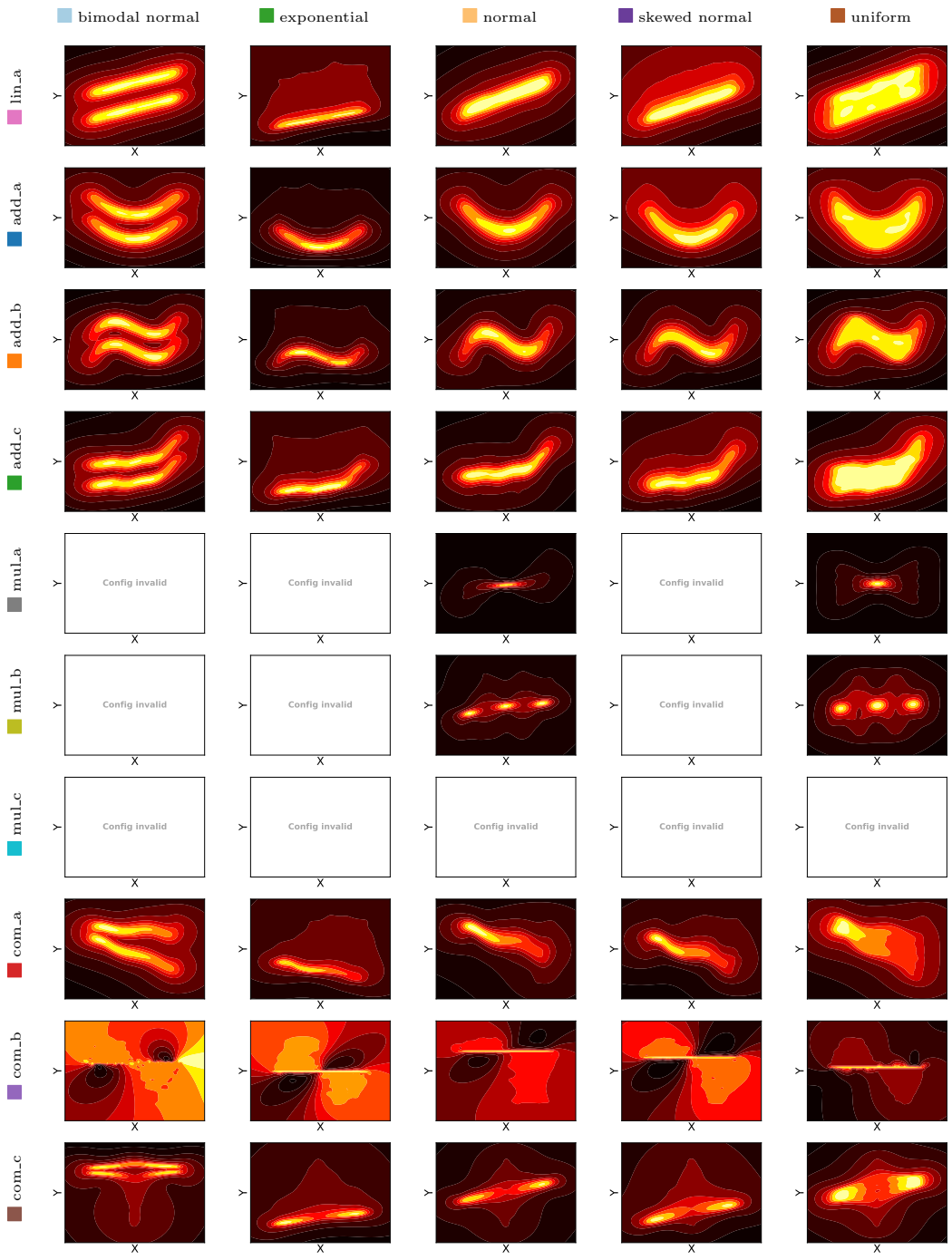



Figure 140: Density contour plot considering a uniform cause distribution and a dependency strength of MI = 0.5. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

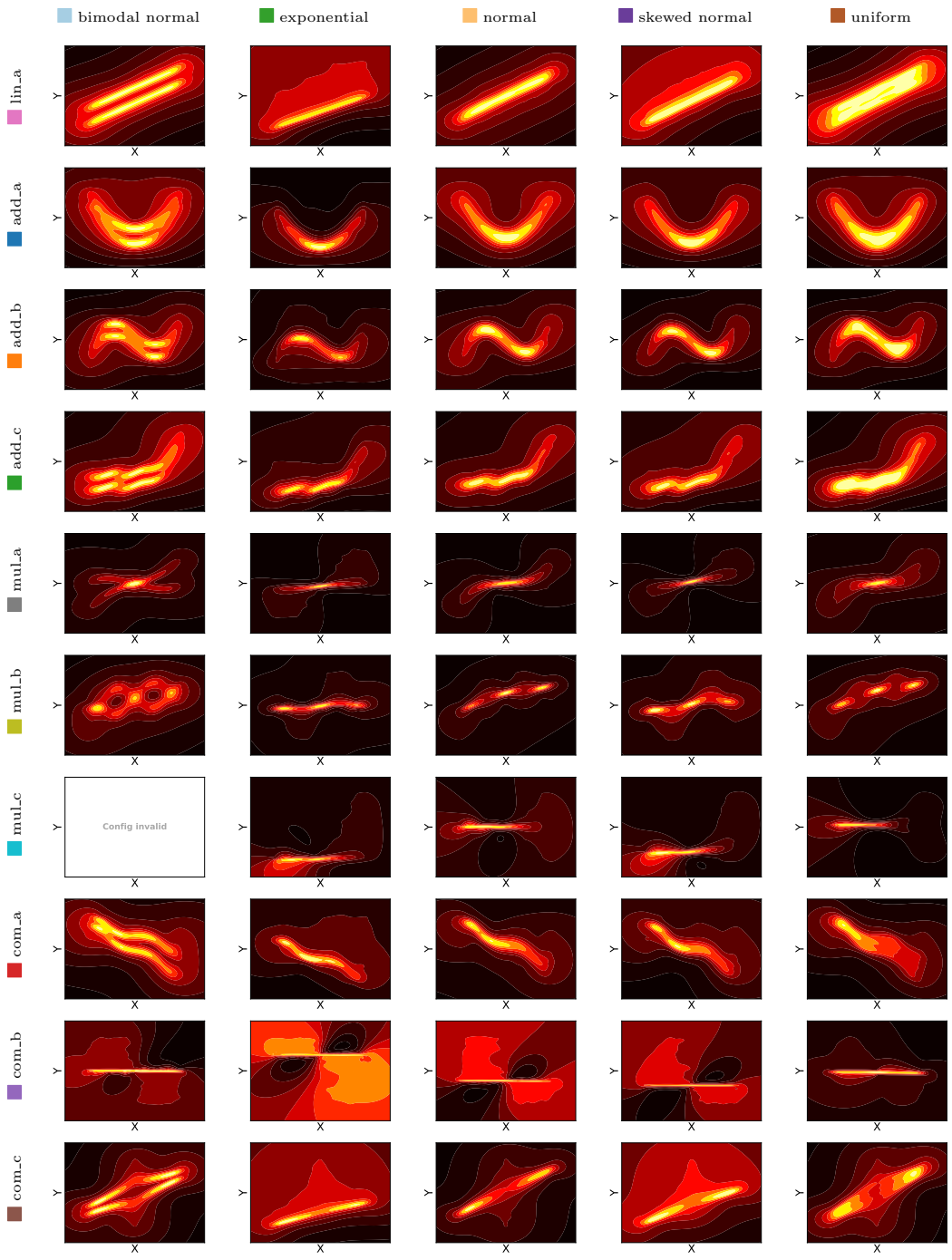



Figure 141: Density contour plot considering a uniform cause distribution and a dependency strength of MI = 1.0. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.

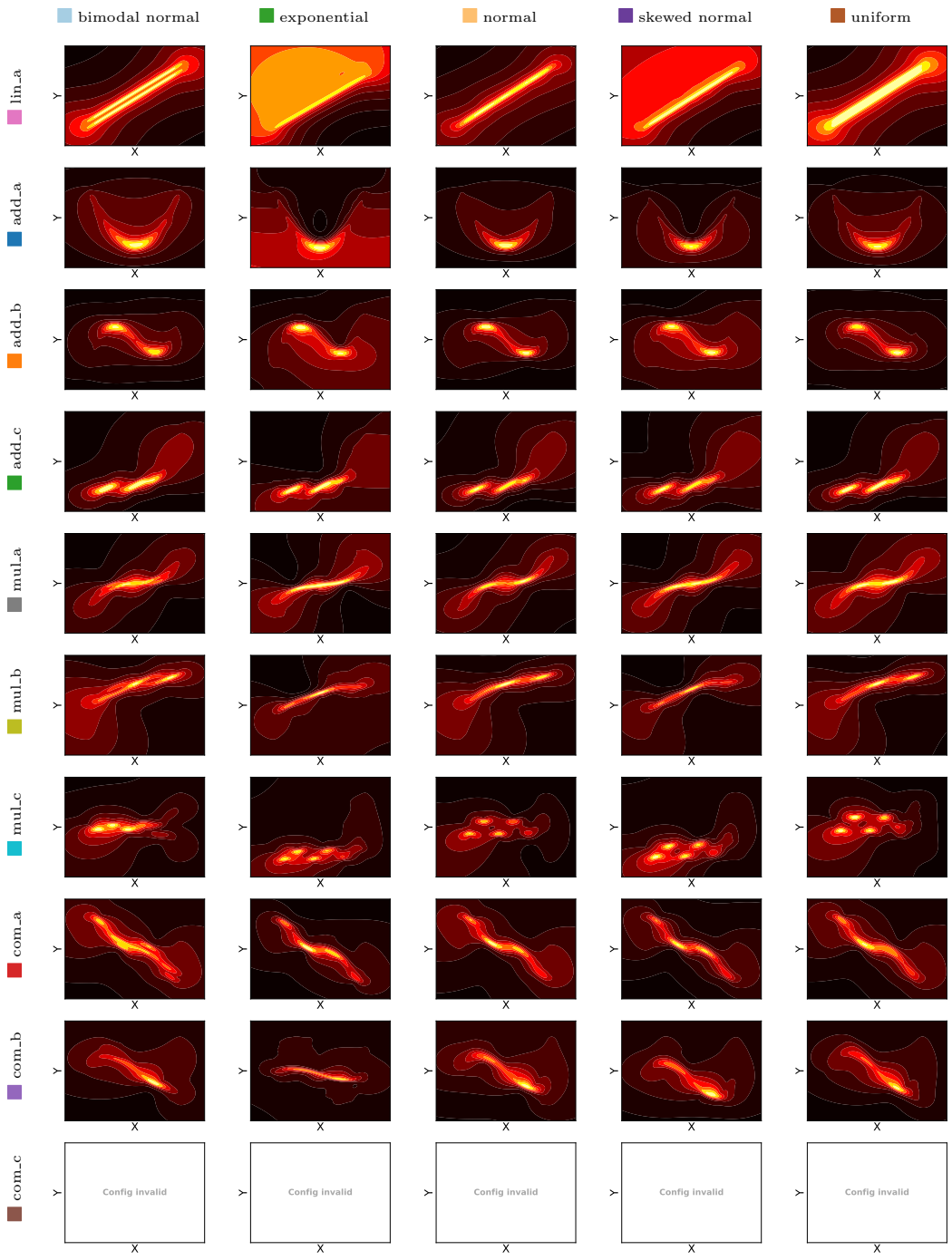



Figure 142: Density contour plot considering a uniform cause distribution and a dependency strength of MI = 2.0. Shown are the functional dependencies versus the noise distributions. In these plots the underlying causal model is always $X \to Y$.