entropy

MDPI

Article

A Bayesian Model for Bivariate Causal Inference

Maximilian Kurthen * and Torsten Enßlin

Max-Planck-Institut für Astrophysik, Karl-Schwarzschildstr. 1, 85748 Garching, Germany; ensslin@mpa-garching.mpg.de

* Correspondence: MKurthen@live.de

Received: 30 October 2019; Accepted: 24 December 2019; Published: 29 December 2019

Abstract: We address the problem of two-variable causal inference without intervention. This task is to infer an existing causal relation between two random variables, i.e., $X \to Y$ or $Y \to X$, from purely observational data. As the option to modify a potential cause is not given in many situations, only structural properties of the data can be used to solve this ill-posed problem. We briefly review a number of state-of-the-art methods for this, including very recent ones. A novel inference method is introduced, *Bayesian Causal Inference* (*BCI*) which assumes a generative Bayesian hierarchical model to pursue the strategy of Bayesian model selection. In the adopted model, the distribution of the cause variable is given by a Poisson lognormal distribution, which allows to explicitly regard the discrete nature of datasets, correlations in the parameter spaces, as well as the variance of probability densities on logarithmic scales. We assume Fourier diagonal Field covariance operators. The model itself is restricted to use cases where a direct causal relation $X \to Y$ has to be decided against a relation $Y \to X$, therefore we compare it other methods for this exact problem setting. The generative model assumed provides synthetic causal data for benchmarking our model in comparison to existing state-of-the-art models, namely *LiNGAM*, *ANM-HSIC*, *ANM-MML*, *IGCI*, and *CGNN*. We explore how well the above methods perform in case of high noise settings, strongly discretized data, and very sparse data. *BCI* performs generally reliably with synthetic data as well as with the real world *TCEP* benchmark set, with an accuracy comparable to state-of-the-art algorithms. We discuss directions for the future development of *BCI*.

Keywords: causal inference; bayesian model selection; information field theory; cause–effect pairs; additive noise

1. Introduction

1.1. Motivation and Significance of the Topic

Causal Inference regards the problem of drawing conclusions about how some entity we can observe does—or does not—influence or is being influenced by another entity. Having knowledge about such law-like causal relations enables us to predict what will happen ($\widehat{=}$ the effect) if we know how the circumstances ($\widehat{=}$ the cause) do change. For example, one can draw the conclusion that a street will be wet (the effect) whenever it rains (the cause). Knowing that it will rain, or indeed observing the rainfall itself, enables one to predict that the street will be wet. Less trivial examples can be found in the fields of epidemiology (identifying some bacteria as the cause of a disease) or economics (knowing how taxes will influence the GDP of a country). Under ideal conditions the system under investigation can be manipulated. Such interventions might allow to set causal variables to specific values which allows to study their effects statistically. In many applications, like in astronomy, geology, global economics, and others, this is hardly possible. For example, in the area of astrophysics, observed properties of galaxies or galaxy clusters include redshift, size, spectral features, fluxes at different wavelengths for lines of sight through the Milky Way. There, an identification of causal directions has

to rest on the hints the causal relation imprints onto the available data only. Restricting on the decision between a direct causal relation $X \to Y$ vs. $Y \to X$ ignores important possibilities (such as hidden confounders), however is still important as a decision within of a subset of possible causal relations and has been giving rise to own benchmark datasets [2].

Especially within the fields of data science and machine learning, specific tasks from causal inference have been attracting much interest recently. The authors of [3] propose that causal inference stands as a third main task of data science besides description and prediction. Reference [4] claims that the task of causal inference will be the next "big problem" for Machine Learning. Such a specific problem is the two variable causal inference, also addressed as the *cause–effect problem* by [5]. Given purely observational data from two random variables, X and Y, which are directly causally related, the challenge is to infer the correct causal direction. Interestingly, this is an incorporation of a fundamental asymmetry between cause and effect which does always hold and can be exploited to tackle such an inference problem. Given two random variables, X and Y, which are related causally, $X \to Y$ ("X causes Y"), there exists a fundamental independence between the distribution of the cause $\mathcal{P}(X)$ and the mechanism which relates the cause X to the effect Y. This independence, however, does not hold in the reverse direction. Most of the proposed methods for the inference of such a causal direction make use of this asymmetry in some way, either by considering the independence directly [2,6], or by taking into account the algorithmic complexity for the description of the factorization $\mathcal{P}(X)\mathcal{P}(Y|X)$ and comparing it to the complexity of the reverse factorization $\mathcal{P}(Y)\mathcal{P}(X|Y)$.

The point we want to make here is, that from the perspective of Bayesian probability theory, causal inference looks like a classical hypothesis testing problem. For this, the probability $\mathcal{P}(\boldsymbol{d}|X \to Y, \mathcal{M})$ is to be compared to $\mathcal{P}(\boldsymbol{d}|Y \to X, \mathcal{M})$. Here, $\boldsymbol{d}$ is the data, $X \to Y$ is the causal direction, and $\mathcal{M}$ is a (meta-) model of how causal relations between X and Y are typically realized. In the following we will adopt a specif choice for the model $\mathcal{M}$. This choice and those of our numerical approximations could and should be criticized and eventually improved. The hypothesis we are following here, however, is that, given $\mathcal{M}$, Bayesian inference is everything that is needed to judge causal directions. Different algorithms for causal inference might just result from different assumptions and different numerical approximations made.

1.2. Structure of the Work

The rest of the paper will be structured as follows. In Section 2 we will briefly outline and specify the problem setting. We also will review existing methods here, namely Additive Noise Models, Information Geometric Causal Inference, and Learning Methods. Section 3 will describe our inference model which is based on a hierarchical Bayesian model. In Section 4 we will accompany the theoretical framework with experimental results. To that end we outline the "forward model" which allows to sample causally related data in Section 4.1. We describe a specific algorithm for the inference model in Section 4.2, which is then tested on various benchmark data (Section 4.3). The performance is evaluated and compared to state-of-the-art methods mentioned in Section 2. We conclude in Section 5 by assessing that our model generally can show competitive classification accuracy and propose possibilities to further advance the model.

2. Problem Setting and Related Work

Here and in the following we assume two random variables, X and Y, which map onto measurable spaces $\mathcal{X}$ and $\mathcal{Y}$. Our problem, the two-variable causal inference, is to determine if X causes Y or Y causes X, given only observations from these random variables. The possibilities that they are unrelated ($X \perp\!\!\!\perp Y$), connected by a confounder Z ($X \leftarrow Z \to Y$), or interrelated ($X \leftrightarrow Y$) are ignored here for the sake of clarity.

2.1. Problem Setting

Regarding the definition of causality we refer to the *do-calculus* introduced by [7]. Informally, the intervention $\mathrm{do}(X = x)$ can be described as setting the random variable X to attain the value x. This allows for a very compact definition of a causal relation $X \to Y$ ("X causes Y") via

$$X \to Y \Leftrightarrow \mathcal{P}(y|\mathrm{do}(x)) \neq \mathcal{P}(y|\mathrm{do}(x')) \tag{1}$$

for some x, x' being realizations of X and y being a realization of Y [2] (in general, the probabilities $P(y|X = x)$ and $P(y|\mathrm{do}(X = x))$ are different).

While the conditional probability $P(y|X = x)$ corresponds to just observing the value x for the variable X, the do-probability $P(y|\mathrm{do}(X = x))$ corresponds to a direct manipulation of the system, only modifying x, without changing any other variable directly.

We want to focus on the case of two observed variables, where either $X \to Y$ or $Y \to X$ holds. Our focus is on the specific problem to decide, in a case where two variables X and Y are observed, whether $X \to Y$ holds or $Y \to X$ We suppose to have access to a finite number of samples from the two variables, i.e., samples $\boldsymbol{x} = (x_1, ..., x_N)$ from X and $\boldsymbol{y} = (y_1, ..., y_N)$ from Y. Our task is to decide the true causal direction using only these samples:

Problem 1. *Prediction of causal direction for two variables*
Input: *A finite number of sample data* $\boldsymbol{d} \equiv (\boldsymbol{x}, \boldsymbol{y})$, *where* $\boldsymbol{x} = (x_1, ..., x_N), \boldsymbol{y} = (y_1, ..., y_N)$
Output: *A predicted causal direction* $\mathcal{D} \in \{X \to Y, Y \to X\}$

2.2. Related Work

Approaches to causal inference from purely observational data are often divided intothree groups [8,9], namely constraint-based, score-based, and asymmetry-based methods. Sometimes this categorization is extended by considering learning methods as a fourth, separate group. Constraint-based and score-based methods are using conditioning on external variables. In a two-variable case there are no external variables, so they are of little interest here.

Asymmetry-based methods exploit an inherent asymmetry between cause and effect. This asymmetry can be framed in different terms. We will follow a similar overview here as [5], where we also refer to for a more detailed discussion. One way is to use the concept of algorithmic complexity—given a true direction $X \to Y$, the factorization of the joint probability into $\mathcal{P}(X, Y) = \mathcal{P}(X)\mathcal{P}(Y|X)$ will be less complex than the reverse factorization $\mathcal{P}(Y)\mathcal{P}(X|Y)$. Such an approach is often used by *Additive Noise Models* (ANMs). This family of inference models assume additive noise, i.e., in the case $X \to Y$, Y is determined by some function f, mapping X to Y, and some collective noise variable E_Y, i.e., $Y = f(X) + E_Y$, where X is independent of E_Y.

An early model called *LiNGAM* [10] uses Independent Component Analysis on the data belonging to the variables. This model however makes the assumptions of linear relations and non-Gaussian noise.

A more common approach is to use some kind of regression (e.g., Gaussian process regression) to get an estimate on the function f and measure how well the model such obtained fits the data. The latter is done by measuring independence between the cause variable and the regression residuum (*ANM-HSIC*, [2,11]), or by employing a Bayesian model selection (introduced besides other methods as (*ANM-HSIC*, [12]).

Another way of framing the asymmetry mentioned above is to state that the mechanism relating cause and effect should be independent of the cause. This formulation is employed by the concept of *IGCI* (*Information Geometric Causal Inference*, [6]).

The recent advances in the field of deep learning are represented in an approach called *CGNN* (*Causal Generative Neural Networks*, [13]). The authors use Generative Neural Networks to model the distribution of one variable given samples from the other variable. As Neural Networks are able to approximate nearly arbitrary functions, the direction where such a artificial modelling is closer to the real distributions (inferred from the samples) is preferred.

Finally, *KCDC* (*Kernel Conditional Deviance for Causal Inference*, [9]) uses the thought of asymmetry in the algorithmic complexity directly on the conditional distributions $\mathcal{P}(X|Y = y), \mathcal{P}(Y|X = x)$. The model measures the variance of the conditional mean embeddings of the above distributions and prefers the direction with the less varying embedding.

A somewhat related approach employed by [14] uses a Bayesian model in combination with MCMC sampling in order to reconstruct Bayesian networks. The authors of [15] use a Bayesian approach specifically in the domain of causal model discovery and compare it to constraint-based approaches. However, an approach which considers multi-variable causal models is fundamentally different from a 2-variable scenario (as conditioning on a third variable is not possible in the latter case).

3. A Bayesian Inference Model

Our contribution incorporates the concept of Bayesian model selection and builds on the formalism of *Information Field Theory* (*IFT, the information theory for fields* [16]). Parts of this paper are taken from the master's thesis of one of the authors [1]. Bayesian model selection compares two competing models, in our case $X \to Y$ and $Y \to X$, and asks for the ratio of the marginalized likelihoods,

$$\mathcal{O}_{X\to Y} = \frac{\mathcal{P}(\boldsymbol{d}|X \to Y, M)}{\mathcal{P}(\boldsymbol{d}|Y \to X, M)},$$

the so called *Bayes Factor*. Here, M denotes the hyperparameters which are assumed to be the same for both models and are yet to be specified.

In the setting of the present causal inference problem, a similar approach has already been used by [12]. This approach however does use a Gaussian mixture model for the distribution of the cause variable while we model the logarithmic cause distribution as a more flexible Gaussian random field (or Gaussian process) [17] and explicitly consider an additional discretization via introducing counts in bins (using a Poissonian statistic on top). Gaussian random fields have, in principle, an infinite number of degrees of freedom, making them an interesting choice to model our distribution of the cause variable and the function relating cause and effect. The formalism of *IFT* borrows computational methods from quantum field theory for computations with such random fields.

Throughout the following we will consider $X \to Y$ as the true underlying direction which we derive our formalism for. The derivation for $Y \to X$ will follow analogously by switching the variables. We will begin with deriving in Section 3.1 the distribution of the cause variable, $\mathcal{P}(X|X \to Y, M)$. In Section 3.2 we continue by considering the conditional distribution $\mathcal{P}(Y|X, X \to Y, M)$. Combining those results, we compute then the full Bayes factor in Section 3.3.

3.1. Distribution of the Cause Variable

Without imposing any constraints, we reduce our problem to the interval $[0, 1]$ by assuming that $\mathcal{X} = \mathcal{Y} = [0, 1]$. This can always be ensured by rescaling the data. We make the assumption that in principle the cause variable X follows a lognormal distribution.

$$\mathcal{P}(x|\beta) \propto e^{\beta(x)}$$

with $\beta \in \mathbb{R}^{[0,1]}$ (throughout the paper we will use the set theory notation for functions, i.e., for a function $f : X \to Y$ we write $f \in Y^X$, which allows a concise statement of domain and codomain without imposing any further restrictions), being some signal field which follows a zero-centered normal distribution, $\beta \sim \mathcal{N}(\beta|0, B)$.

Here we write B for the covariance operator $\mathbb{E}_{\beta\sim\mathcal{P}(\beta)}[\beta(x_0)\beta(x_1)] = B(x_0, x_1)$.

We postulate statistical homogeneity for the covariance, that is

$$\mathbb{E}_{\beta\sim\mathcal{P}(\beta)}[\beta(x)] = \mathbb{E}[\beta(x + t)]$$
$$\mathbb{E}_{\beta\sim\mathcal{P}(\beta)}[\beta(x)\beta(y)] = \mathbb{E}[\beta(x + t)\beta(y + t)]$$

i.e., first and second moments should be independent on the absolute location. The *Wiener–Khintchine Theorem* now states that the covariance has a spectral decomposition, i.e., it is diagonal in Fourier space, under this condition (see, e.g., [18]). Denoting the Fourier transform by $\mathcal{F}$, i.e., in the one dimensional case, $\mathcal{F}[f](q) = \int \mathrm{d}x\, e^{-iqx} f(x)$. Therefore, the covariance can be completely specified by a one dimensional function:

$$(\mathcal{F}B\mathcal{F}^{-1})(k,q) = 2\pi\delta(k-q)P_\beta(k)$$

Here, $P_\beta(k)$ is called the *power spectrum*.

Building on these considerations we now regard the problem of discretization. Measurement data itself is usually not purely continuous but can only be given in a somewhat discretized way (e.g., by the measurement device itself or by precision restrictions imposed from storing the data). Another problem is that many numerical approaches to inference tasks, such as Gaussian Process regression, use finite bases as approximations in order to efficiently obtain results [2,12]. Here, we aim to directly confront these problems by imposing a formalism where the discretization is inherent.

So instead of taking a direct approach with the above formulation, we use a Poissonian approach and consider an equidistant grid $\{z_1, ..., z_{n_{\mathrm{bins}}}\}$ in the $[0,1]$ interval. This is equivalent to defining bins, where the z_j are the midpoints of the bins. We now take the measurement counts, k_i, which gives the number of x-measurements within the i-th bin. For these measurement counts we now take a Poisson lognormal distribution as an Ansatz, that is, we assume that the measurement counts for the bins are Poisson distributed, where the means follow a lognormal distribution. We argue that this is indeed a justified approach here, as in a discretized scenario we have to deal with count-like integer data (where a Poisson distribution is natural). The lognormal distribution of the Poisson parameters is in our eyes well-justified here. On the one hand, it takes into account the non-negativity of the Poisson parameter. On the other hand, only proposing a normal distribution of the log allows for a uncertainty in the order of magnitude while permitting for spatial correlation in this log density.

We can model this discretization by applying a response operator $R : \mathbb{R}^{[0,1]} \to \mathbb{R}^{n_{\mathrm{bins}}}$ to the lognormal field. This is done in the most direct way via employing a Dirac delta distribution

$$R_{jx} \equiv \delta(x - z_j)$$

In order to allow for a more compact notation we will use an index notation from now on, e.g., $f_x = f(x)$ for some function f or $O_{xy} = O(x,y)$ for some operator O. Whenever the indices are suppressed, an integration (in the continuous case) or dot product (in the discrete case) is understood, e.g., $(Of)_x \equiv O_{xy} f_y = \int \mathrm{d}y O_{xy} f_y = \int \mathrm{d}y O(x,y) f(y)$ In the following we will use bold characters for finite dimensional vectors, e.g., $\boldsymbol{\lambda} \equiv (\lambda_1, ..., \lambda_{n_{\mathrm{bins}}})^T$. By inserting such a finite dimensional vector in the argument of a function, e.g., $\beta(\boldsymbol{x})$ we refer to a vector consisting of the function evaluated at each entry of $\boldsymbol{x}$, that is $\beta(\boldsymbol{z}) \equiv (\beta(z_1), ..., \beta(z_{n_{\mathrm{bins}}})$. Later on we will use the notation $\widehat{\cdot}$ which raises some vector to a diagonal matrix ($\widehat{\boldsymbol{x}}_{ij} \equiv \delta_{ij} x_i$ (no summation implicated)). We will use this notation analogously for fields, e.g., ($\widehat{\beta}_{uv} \equiv \delta(u-v)\beta(u)$). Writing $\mathbf{1}^\dagger \boldsymbol{R}$ denotes the dot product of the vector $\boldsymbol{R}$ with a vector of ones and hence corresponds to the summation of the entries of $\boldsymbol{R}$ ($\mathbf{1}^\dagger \boldsymbol{R} = \sum_j R_j$). Now we can state the probability distribution for k_j, the measurement count in bin j:

$$\mathcal{P}(k_j|\lambda_j) = \frac{\lambda_j^{k_j} e^{-\lambda_j}}{k_j!}$$

$$\lambda_j = \mathbb{E}_{(k|\beta)}[k_j] = \rho e^{\beta_{z_j}} = \int dx R_{jx} e^{\beta_x} = \rho (R e^\beta)_j$$

Therefore, considering the whole vector $\boldsymbol{\lambda}$ of bin means and the vector $\boldsymbol{k}$ of bin counts at once:

$$\begin{aligned}
\boldsymbol{\lambda} &= \rho \boldsymbol{R} e^{\beta} = \rho e^{\beta(\boldsymbol{z})} \\
\mathcal{P}(\boldsymbol{k}|\boldsymbol{\lambda}) &= \prod_j \frac{\lambda_j^{k_j} e^{-\lambda_j}}{k_j!} = \prod_j \frac{(R_j e^{\beta})^{k_j} e^{-R_j e^{\beta}}}{k_j!} = \frac{(\prod_j (R_j e^{\beta})^{k_j}) e^{-\mathbf{1}^\dagger \boldsymbol{R} e^{\beta}}}{\prod_j k_j!} \\
\mathcal{P}(\boldsymbol{x}|\boldsymbol{k}) &= \frac{1}{N!}
\end{aligned}$$

The last equation follows from the consideration that given the counts $(k_1, ..., k_{n_{\text{bins}}})$ for the bins, only the positions of the observations $(x_1, ..., x_N)$ is fixed, but the ordering is not. The N observations can be ordered in $N!$ ways.

Now considering the whole vector of bin counts $\boldsymbol{k}$ at once, we get

$$\mathcal{P}(\boldsymbol{k}|\beta) = \frac{e^{\sum_j k_j \beta(z_j)} e^{-\boldsymbol{\rho}^\dagger e^{\beta(\boldsymbol{z})}}}{\prod_j k_j!} = \frac{e^{\boldsymbol{k}^\dagger \beta(\boldsymbol{z}) - \boldsymbol{\rho}^\dagger e^{\beta(\boldsymbol{z})}}}{\prod_j k_j!} \tag{2}$$

A marginalization in β involving a Laplace approximation around the most probable $\beta = \beta_0$ leads to (see Appendix A for a detailed derivation):

$$\mathcal{P}(\boldsymbol{x}|P_\beta, X \to Y) \approx \frac{1}{N!} \frac{e^{+\boldsymbol{k}^\dagger \boldsymbol{\beta_0} - \boldsymbol{\rho}^\dagger e^{\boldsymbol{\beta_0}} - \frac{1}{2}\beta_0^\dagger B^{-1} \beta_0}}{\left| \rho B \widehat{e^{\boldsymbol{\beta_0}}} + \mathbb{1} \right|^{\frac{1}{2}} \prod_j k_j!} \tag{3}$$

$$\mathcal{H}(\boldsymbol{x}|P_\beta, X \to Y) \approx \mathcal{H}_0 + \frac{1}{2} \log |\rho B \widehat{e^{\boldsymbol{\beta_0}}} + \widehat{\mathbb{1}}| + \log(\prod_j k_j!) - \boldsymbol{k}^\dagger \boldsymbol{\beta_0} + \boldsymbol{\rho}^\dagger e^{\boldsymbol{\beta_0}} + \frac{1}{2} \beta_0^\dagger B^{-1} \beta_0 \tag{4}$$

where $\mathcal{H}(\cdot) \equiv -\log(\mathcal{P}(\cdot))$ is called the *information Hamiltonian* in *IFT* and $\mathcal{H}_0$ collects all terms which do not depend on the data $\boldsymbol{d}$.

3.2. Functional Relation of Cause and Effect

Similarly to β, we suppose a Gaussian distribution for the function f, relating Y to X:

$$\mathbb{R}^{[0,1]} \ni f \sim \mathcal{N}(0|f, F)$$

Proposing a Fourier diagonal covariance F once more, determined by a power spectrum P_f,

$$(\mathcal{F} F \mathcal{F}^{-1})(k, q) = 2\pi \delta(k - q) P_f(k),$$

we assume additive Gaussian noise, using the notation $f(\boldsymbol{x}) \equiv (f(x_1), ..., f(x_N))^T$ and $\boldsymbol{\epsilon} \equiv (\epsilon_1, ..., \epsilon_N)^T$.

So, in essence we are proposing a Gaussian Process Regression with a Fourier diagonal covariance. We have

$$\begin{aligned}
\boldsymbol{y} &= f(\boldsymbol{x}) + \boldsymbol{\epsilon} \\
\boldsymbol{\epsilon} &\sim \mathcal{N}(\epsilon|0, \mathcal{E}) \\
\mathcal{E} &\equiv \text{diag}(\varsigma^2, \varsigma^2, ...) = \varsigma^2 \mathbb{1} \in \mathbb{R}^{N \times N},
\end{aligned} \tag{5}$$

that is, each independent noise sample is drawn from a zero-mean Gaussian distribution with given variance ς^2.

Knowing the noise $\boldsymbol{\epsilon}$, the cause $\boldsymbol{x}$ and the causal mechanism f completely determines $\boldsymbol{y}$ via Equation (5). Therefore, $\mathcal{P}(\boldsymbol{y}|\boldsymbol{x}, f, \boldsymbol{\epsilon}, X \to Y) = \delta(\boldsymbol{y} - f(\boldsymbol{x}) - \boldsymbol{\epsilon})$. We can now state the conditional

distribution for the effect variable measurements $\boldsymbol{y}$, given the cause variable measurements $\boldsymbol{x}$. Marginalizing out the dependence on the relating function f and the noise $\boldsymbol{\epsilon}$ we get:

$$\begin{aligned}\mathcal{P}(\boldsymbol{y}|\boldsymbol{x}, P_f, \varsigma, X \to Y) &= \int \frac{\mathrm{d}^N \boldsymbol{q}}{(2\pi)^N} e^{i\boldsymbol{q}^\dagger \boldsymbol{y} - \frac{1}{2}\boldsymbol{q}(\tilde{F}+\mathcal{E})\boldsymbol{q}} \\ &= (2\pi)^{-\frac{N}{2}} \left|\tilde{F}+\mathcal{E}\right|^{-\frac{1}{2}} e^{-\frac{1}{2}\boldsymbol{y}^\dagger(\tilde{F}+\mathcal{E})^{-1}\boldsymbol{y}} \end{aligned} \tag{6}$$

In the equation above, $\tilde{F}$ denotes a the $N \times N$-matrix with entries $\tilde{F}_{ij} = F(x_i, x_j)$ (this type of matrix, i.e., the evaluation of covariance or kernel at certain positions, is sometimes called a Gram matrix). Again, we give a detailed computation in the Appendix A.

3.3. Computing the Bayes Factor

Now we are able to calculate the full likelihood of the data $\boldsymbol{d} = (\boldsymbol{x}, \boldsymbol{y})$ given our assumptions P_β, P_f, ς for the direction $X \to Y$ and vice versa $Y \to X$ (via the full model as given in Figure 1 As we are only interested in the ratio of the probabilities and not in the absolute probabilities itself, it suffices to calculate the Bayes factor:

$$\begin{aligned} O_{X\to Y} &= \frac{\mathcal{P}(\boldsymbol{d}|P_\beta, P_f, \varsigma, X \to Y)}{\mathcal{P}(\boldsymbol{d}|P_\beta, P_f, \varsigma, Y \to X)} \\ &= \exp[\mathcal{H}(\boldsymbol{d}|P_\beta, P_f, \varsigma, Y \to X) - \mathcal{H}(\boldsymbol{d}|P_\beta, P_f, \varsigma, X \to Y)] \end{aligned}$$

Here we used again the information Hamiltonian $\mathcal{H}(\cdot) \equiv -\log \mathcal{P}(\cdot)$

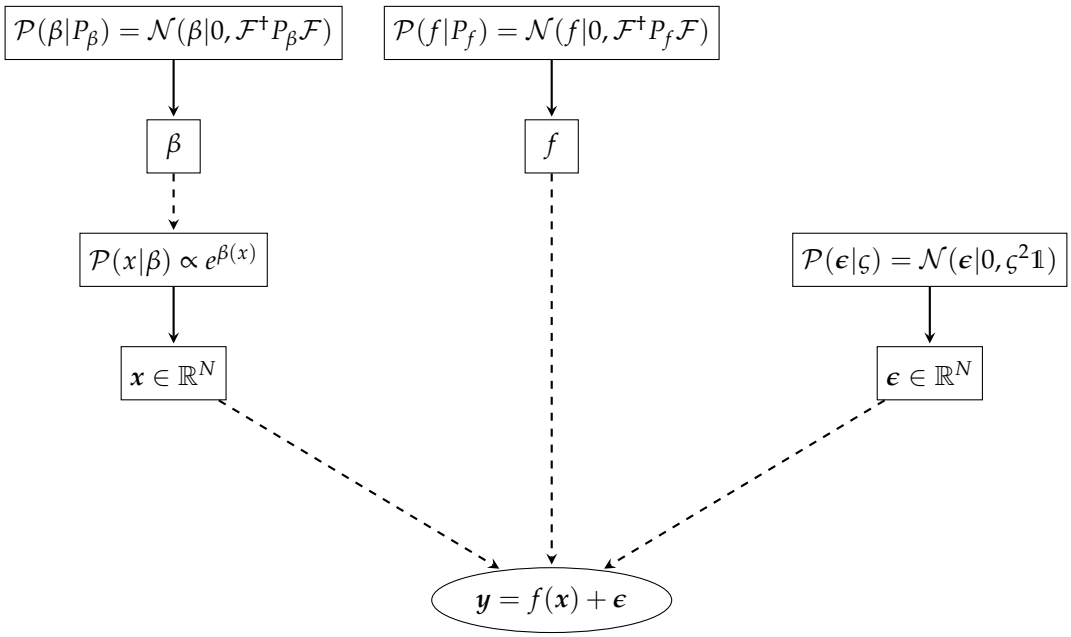

Figure 1. Overview over the used Bayesian hierarchical model, for the case $X \to Y$.

Making use of Equations (3) and (6) we get, using the calculus for conditional distributions on the Hamiltonians, $\mathcal{H}(A, B) = \mathcal{H}(A|B) + \mathcal{H}(B)$,

$$\begin{aligned} \mathcal{H}(\boldsymbol{d}|P_\beta, P_f, \varsigma, X \to Y) &= \mathcal{H}(\boldsymbol{x}|P_\beta, X \to Y) + \mathcal{H}(\boldsymbol{y}|\boldsymbol{x}, P_f, \varsigma, X \to Y) \\ &= \mathcal{H}_0 + \log(\prod_j k_j!) + \frac{1}{2}\log|\rho B e^{\widehat{\boldsymbol{\beta_0}}} + \mathbb{1}| - \boldsymbol{k}^\dagger \boldsymbol{\beta_0} + \\ &+ \boldsymbol{\rho}^\dagger e^{\boldsymbol{\beta_0}} + \frac{1}{2}\beta_0^\dagger B^{-1}\beta_0 + \frac{1}{2}\boldsymbol{y}^\dagger(\tilde{F}+\mathcal{E})^{-1}\boldsymbol{y} + \frac{1}{2}\left|\tilde{F}+\mathcal{E}\right| \end{aligned} \tag{7}$$

In this formula we suppressed the dependence of $\tilde{F}, \beta_0$ on $\boldsymbol{x}$ (for the latter, the dependence is not explicit, but rather implicit as β_0 is determined by the minimum of the $\boldsymbol{x}$-dependent functional γ). We omit stating $\mathcal{H}(\boldsymbol{d}|P_\beta, P_f, \varsigma, Y \to X)$ explicitly as the expression is just given by taking Equation (7) and switching $\boldsymbol{x}$ and $\boldsymbol{y}$ or X and Y, respectively.

4. Implementation and Benchmarks

We can use our model in a forward direction to generate synthetic data with a certain underlying causal direction. We describe this process in Section 4.1. In Section 4.2 we give an outline on the numerical implementation of the inference algorithm. This algorithm is tested on and compared on benchmark data. To that end we use synthetic data and real world data. We describe the specific datasets and give the results in Section 4.3.

4.1. Sampling Causal Data via a Forward Model

To estimate the performance of our algorithm and compare it with other existing approaches, a benchmark dataset is of interest to us. Such benchmark data is usually either real world data or synthetically produced. While we will use the *TCEP* benchmark set of [2] in Section 4.3.2, we also want to use our outlined formalism to generate artificial data representing causal structures. Based on our derivation for cause and effect we implement a forward model (1) to generate data $\boldsymbol{d}$ as following Algorithm 1.

Algorithm 1 Sampling of causal data via forward model

Input: Power spectra P_β, P_f, noise variance ς^2, number of bins n_{bins}, desired number (As we draw the number of samples from Poisson distribution in each bin, we do not deterministically control the total number of samples) of samples $\tilde{N}$
Output: N samples $(d_i) = (x_i, y_i)$ generated from a causal relation of either $X \to Y$ or $Y \to X$

1. Draw a sample field $\beta \in \mathbb{R}^{[0,1]}$ from the distribution $\mathcal{N}(\beta|0, B)$
2. Set an equally spaced grid with n_{bins} points in the interval $[0,1]$: $\boldsymbol{z} = (z_1, ..., z_{n_{\text{bins}}}), z_i = \frac{i-0.5}{n_{\text{bins}}}$
3. Calculate the vector of Poisson means $\boldsymbol{\lambda} = (\lambda_1, ...\lambda_{n_{\text{bins}}})$ with $\lambda_i \propto e^{\beta(z_i)}$
4. At each grid point $i \in \{1, ..., n_{\text{bins}}\}$, draw a sample k_i from a Poisson distribution with mean λ_i: $k_i \sim \mathcal{P}_{\lambda_i}(k_i)$
5. Set $N = \sum_{i=1}^{n_{\text{bins}}} k_i$
6. For each $i \in \{1, ..., n_{\text{bins}}\}$ add k_i times the element z_i to the set of measured x_j. Construct the vector $\boldsymbol{x} = (..., \underbrace{z_i, z_i, z_i}_{k_i \text{ times}}, ...)$
7. Draw a sample field $f \in \mathbb{R}^{[0,1]}$ from the distribution $\mathcal{N}(f|0, F)$. Rescale f s.th. $f \in [0,1]^{[0,1]}$
8. Draw a multivariate noise sample $\boldsymbol{\epsilon} \in \mathbb{R}^N$ from a normal distribution with zero mean and variance ς^2, $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{\epsilon}|0, \varsigma^2)$
9. Generate the effect data $\boldsymbol{y}$ by applying f to $\boldsymbol{x}$ and adding $\boldsymbol{\epsilon}$: $\boldsymbol{y} = f(\boldsymbol{x}) + \boldsymbol{\epsilon}$
10. With probability $\frac{1}{2}$ return $\boldsymbol{d} = (\boldsymbol{x}^T, \boldsymbol{y}^T)$, otherwise return $\boldsymbol{d} = (\boldsymbol{y}^T, \boldsymbol{x}^T)$

Comparing the samples for different power spectra (see Figure 2), we decide to sample data with power spectra $P(q) = \frac{1000}{q^4+1}$ and $P(q) = \frac{1000}{q^6+1}$, as these seem to resemble "natural" mechanisms, see Figure 2.

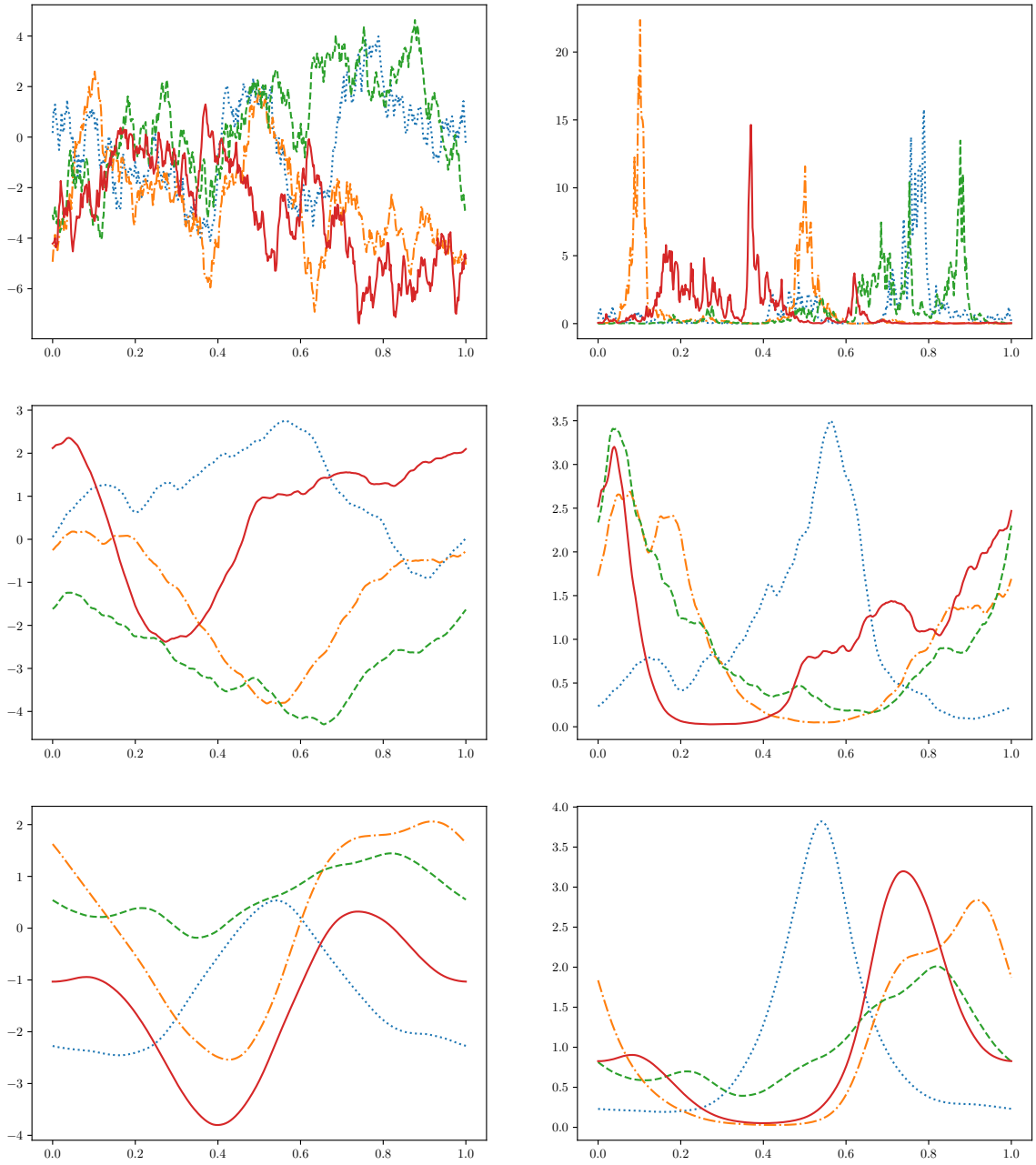

Figure 2. Different field samples from the distribution $\mathcal{N}(\cdot|0, \mathcal{F}^\dagger \widehat{P} \mathcal{F})$ (on the left) with the power spectrum $P(q) \propto \frac{1}{q^2+1}$ (top), $P(q) \propto \frac{1}{q^4+1}$ (middle), $P(q) \propto \frac{1}{q^6+1}$ (bottom). On the left, the field values themselves are plotted, on the right an exponential function is applied to those and the fields are normalized, i.e., as in our formulation $\lambda_j \propto \frac{e^{\beta(z_j)}}{\int \mathrm{d}z e^{\beta(z)}}$ (Same colors/line styles on the right and the left indicate the same underlying functions (colors itself chosen just for distinguishability).

4.2. Implementation of the Bayesian Causal Inference Model

Based on our derivation in Section 3 we propose a specific algorithm to decide the causal direction of a given dataset and therefore give detailed answer for Problem 1. Basically, the task comes down to find the minimum β_0 for the saddle point approximation and calculate the terms given in Equation (7):

We provide an implementation of Algorithm 2 in *Python* (https://gitlab.mpcdf.mpg.de/ift/bayesian_causal_inference). We approximate the operators B, F as matrices $\in \mathbb{R}^{n_{\text{bins}} \times n_{\text{bins}}}$, which allows us to explicitly numerically compute the determinants and the inverse. As the most critical part we

consider the minimization of β, i.e., step 4 in Algorithm 2. As we are however able to analytically give the curvature Γ_β and the gradient $\partial_\beta \gamma$ of the energy γ to minimize, we can use a Newton-scheme here. We derive satisfying results (see Figure 3) using the *Newton-CG* algorithm [19], provided by the *SciPy*-Library [20]. After testing our algorithm on different benchmark data, we choose the default hyperparameters as

$$P_\beta = P_f \propto \frac{1}{q^4 + 1}, \tag{8}$$

$$\varsigma^2 = 0.01, \tag{9}$$

$$r = 512, \tag{10}$$

$$\rho = 1. \tag{11}$$

Algorithm 2 2-variable causal inference

Input: Finite sample data $\boldsymbol{d} \equiv (\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^{N\times 2}$, Hyperparameters $P_\beta, P_f, \varsigma^2, r$

Output: Predicted causal direction $\mathcal{D}_{X\to Y} \in \{X \to Y, Y \to X\}$

1. Rescale the data to the $[0,1]$ interval. That is, $\min\{x_1, ..., x_N\} = \min\{y_1, ..., y_N\} = 0$ and $\max\{x_1, ..., x_N\} = \max\{y_1, ..., y_N\} = 1$
2. Define an equally spaced grid of $(z_1, ..., z_{n_{\text{bins}}})$ in the interval $[0,1]$
3. Calculate matrices $\boldsymbol{B}, \boldsymbol{F}$ representing the covariance operators B and F evaluated at the positions of the grid, i.e., $\boldsymbol{B}_{ij} = B(z_i, z_j)$
4. Find the $\beta_0 \in \mathbb{R}^{[0,1]}$ for which γ, as defined in Appendix A.1 (Equation (A2)), becomes minimal
5. Calculate the $\boldsymbol{d}$-dependent terms of the information Hamiltonian in Equation (7) (i.e., all terms except $\mathcal{H}_0$)
6. Repeat steps 4 and 5 with $\boldsymbol{y}$ and $\boldsymbol{x}$ switched
7. Calculate the Bayes factor $\mathcal{O}_{X\to Y}$
8. If $\mathcal{O}_{X\to Y} > 1$, return $X \to Y$, else return $Y \to X$

While fixing the power spectra might seem somewhat arbitrary, we remark that this corresponds to fixing a kernel, e.g., as a squared exponential kernel, which is done in many publications (e.g., [9,13]).

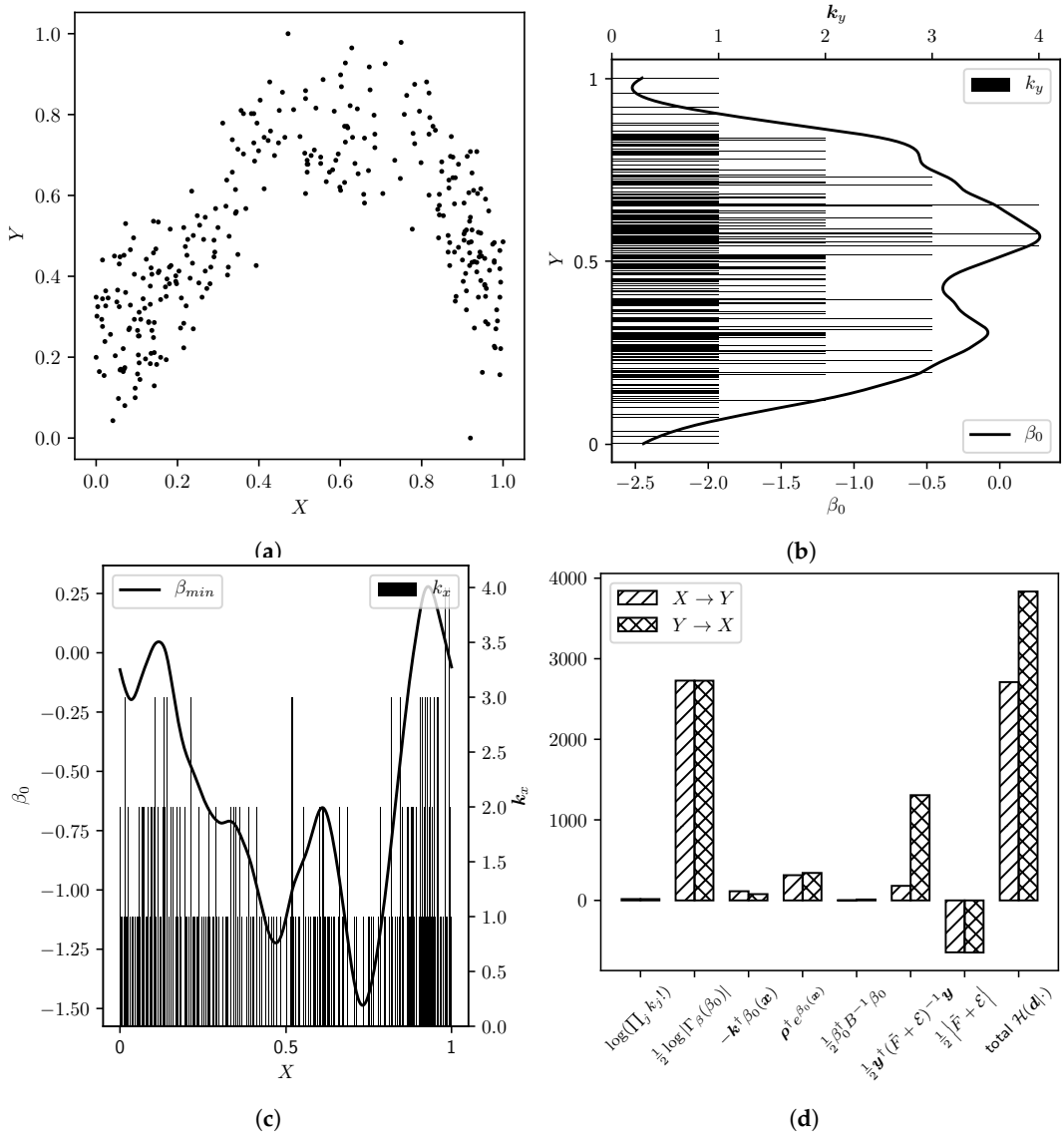

Figure 3. Illustration of a *Bayesian Causal Inference* run on synthetic data generated for causality $X \to Y$. Here, the method clearly favours this causality with an odds ratio of $\mathcal{O}_{X \to Y} \approx 10^{500} : 1$. (**a**) Synthetic data, with causality $X \to Y$; (**b**) Count histogram ($\boldsymbol{k}$) and inferred β_0 for the model in the direction $Y \to X$; (**c**) Count histogram ($\boldsymbol{k}$) and inferred β_0 for the model in the direction $X \to Y$; (**d**) Values of terms in $\mathcal{H}(\boldsymbol{d}|P_\beta, P_f, \varsigma, X \to Y)$ and $\mathcal{H}(\boldsymbol{d}|P_\beta, P_f, \varsigma, Y \to X)$. Smaller values increase the probability of the respective direction.

Future extensions of our method might learn P_β and P_f if the data is rich enough.

4.3. Benchmark Results

We compare our outlined model, in the following called *BCI* (*Bayesian Causal Inference*), to a number of state-of-the-art approaches. The selection of the considered methods is influenced by the ones in recent publications, e.g., [9,13]. Namely, we include the *LiNGAM* algorithm, acknowledging it as one of the oldest models in this field and a standard reference in many publications. We also use the *ANM* Algorithm [2] with HSIC and Gaussian Process Regression (*ANM-HSIC*) as well as the *ANM-MML* approach [12]. The latter uses a Bayesian Model Selection, arguably the closest to the algorithm proposed in this publication, at least to our best knowledge. We further include the *IGCI* algorithm, as it differs fundamentally in its formulation from the ANM algorithms and has shown

strong results in recent publications [2,9,13]. We employ the IGCI algorithm with entropy estimation for scoring and a Gaussian distribution as reference distribution.

Finally, *CGNN* [13] represents the rather novel influence of deep learning methods. We use the implementation provided by the authors, with itself uses *Python* with the *Tensorflow* [21] library. The most critical hyper-parameter here is, as the authors themselves mention, the number of hidden neurons which we set to a value of $n_h = 30$, as this is the default in the given implementation and delivers generally good results. We use 32 runs each, as recommended by the authors of the algorithm.

A comparison with the *KCDC* algorithm would be interesting, unfortunately the authors did not provide any computational implementation so far (October 2019). We compare the mentioned algorithms to *BCI* on basis of synthetic and real world data. For the synthetic data we use our forward model, as outlined in Section 4.1 with varying parameters. For the real world data we use the well-known *TCEP* dataset (*Tuebingen Cause Effect Pairs*, [2]).

4.3.1. Results for Synthetic Benchmark Data

We generate our synthetic data adopting the power spectra $P(q) = \frac{1}{q^4+1}$ for both, P_β, P_f. We further set $n_{\text{bins}} = 512$, $\tilde{N} = 300$, and $\varsigma^2 = 0.05$ as default settings. We provide the results of the benchmarks in Table 1. While *BCI* achieves almost perfect results (98%), the assessed *ANM* algorithms provide perfect performance here. As a first variation, we explore the influence of high and very high noise on the performance of the inference models. Therefore, we set the parameter $\varsigma^2 = 0.2$ for high noise and $\varsigma^2 = 1$ for very high noise in Algorithm 1, while keeping the other parameters set to the default values. While our *BCI* algorithm is affected but still performs reliably with an accuracy of $\geq$ 90%, the *ANM* algorithms are remarkably robust in the presence of the noise. This is likely due to the fact that the distribution of the true cause $\mathcal{P}(X)$ is not influenced by high noise and this distribution is assessed on its own by those.

As our model uses a Poissonian approach, which explicitly considers discretization effects of data measurement, it is of interest how the performance behaves when using a strong discretization. We emulate such a situation by employing our forward model with a very low number of bins. Again, we keep all parameters to default values and set $n_{\text{bins}} = 16$ and $n_{\text{bins}} = 8$ for synthetic data with high and very high discretization. The *ANM* models again turn out to be robust again discretization. *CGNN* and *IGCI* perform significantly worse here. In the case of *IGCI* this can be explained by the entropy estimation, which simply removes non-unique samples. Our *BCI* algorithm is able to achieve over 90% accuracy here.

We explore another challenge for inference algorithms by strongly reducing the number of samples. While we sampled about 300 observations with our other forward models so far, here we reduce the number of observed samples to 30 and 10 samples. In this case *BCI* performs very well compared to the other models, in fact it is able to outperform them in the case of just 10 samples being given. We note that of course *BCI* does have the advantage that it "knows" the hyperparameters of the underlying forward model. Yet we consider the results as encouraging, and this advantage will be removed in the confrontation with real world data.

Table 1. Accuracy for the synthetic data benchmark. All parameters for the forward model besides the mentioned one are kept to default values, namely $n_{\text{bins}} = 512$, $\tilde{N} = 300$, $\varsigma^2 = 0.05$.

Model	Default	$\varsigma^2 = 0.2$	$\varsigma^2 = 1$	$n_{\text{bins}} = 16$	$n_{\text{bins}} = 8$	30 Samples	10 Samples
BCI	0.98	0.94	0.90	0.93	0.97	0.92	0.75
LiNGAM	0.30	0.31	0.40	0.23	0.21	0.44	0.45
ANM-HSIC	1.00	0.98	0.94	0.99	1.00	0.91	0.71
ANM-MML	1.00	0.99	0.99	1.00	1.00	0.98	0.69
IGCI	0.65	0.60	0.58	0.24	0.09	0.48	0.40
CGNN	0.72	0.75	0.77	0.57	0.22	0.46	0.39

4.3.2. Results for Real World Benchmark Data

The most widely used benchmark set with real world data is the *TCEP* [2]. We use the 102 2-variable datasets from the collection with weights as proposed by the maintainers. As some of the contained datasets include a high number of samples (up to 11,000), we randomly subsample large datasets to 500 samples each in order to keep computation time maintainable. We did not include the *LiNGAM* algorithm here, as we experienced computational problems with obtaining results here for certain datasets (namely `pair0098`). The authors of [13] report the accuracy of *LiNGAM* on the *TCEP* dataset to be around 40%. *BCI* performs generally comparable to established approaches as *ANM* and *IGCI*. *CGNN* performs best with an accuracy about 70% here, a bit lower than the one reported by [13] of around 80%. The reason for this is arguably to be found in the fact that we set all hyperparameters to fixed values, while [13] used a leave-one-out-approach to find the best setting for the hyperparameter n_h.

Motivated by the generally strong performance of our approach in the case of sparse synthetic data, we also explore a situation where real world data is only sparsely available. To that end, we subsample all *TCEP* datasets down to 75 randomly chosen samples kept for each one. To circumvent the chance of an influence of the subsampling procedure we average the results over 20 different subsamplings. The results are as well given in Table 2. The loss in accuracy of our model is rather small.

Table 2. Accuracy for TCEP Benchmark.

Model	TCEP	TCEP with 75 Samples
BCI	0.64	0.60
ANM-HSIC	0.63	0.54
ANM-MML	0.58	0.56
IGCI	0.66	0.62
CGNN	0.70	0.69

5. Discussion

The problem of purely bivariate causal discovery is a rather restricted one as it ignores the possibility of causal independence or hidden confounders. However, it is a fundamental one and still remains as a part of the decision within subsets of possible causal structures. It also might be a valuable contribution to real world problems such as in astrophysics, where one might be interested in discovering the main causal direction within the multitude of discovered variables. The Bayesian Causal Inference method introduced builds on the formalism of information field theory. In this regard, we employed the concept of Bayesian model selection and made the assumption of additive noise, i.e., $\boldsymbol{x} = f(\boldsymbol{y}) + \boldsymbol{\epsilon}$. In contrast to other methods which do so, such as *ANM-MML*, we do not model the cause distribution by a Gaussian mixture model but by a Poisson Lognormal statistic.

We could show that our model is able to provide classification accuracy in the present causal inference task that is comparable to the one of other methods in the field. Our method is of course restricted to a bivariate problem, i.e., determining the direction of a causal relation between two one-dimensional variables. One difference from our model to existing ones is arguably to be found in the choice of the covariance operators. While our method uses, at its heart, Gaussian Process Regression, most other publications which do so use squared exponential kernels. We, however, choose a covariance which is governed by a $\frac{1}{q^4+1}$ power spectrum. This permits detecting more structures at small scales than methods using a squared exponential kernel.

As a certain weak point of *BCI*, we consider the approximation of the uncomputable path integrals via the Laplace approximation. A thorough investigation of error bounds (e.g., [22]) is yet to be carried out. As an alternative, one can think about sampling-based approaches to approximate the integrals. A recent publication [23] introduced a harmonic mean-based sampling approach to approximate moderate dimensional integrals. Adopting such a technique to our very high dimensional case might be promising to improve *BCI*. Furthermore, the novel *Metric Gaussian Variational Inference method*

(MGVI [24]) might allow to go beyond the saddle point approximation method used here. At this point, we also want to mention that our numerical implementation will have difficulty to scale well to very large (lots of data points) problems. This is, however, also an issue for competing models and a subsampling procedure can always be used to decrease the scale of the problem.

Another interesting perspective is provided by deeper hierarchical models. While the outlined method took the power spectra and the noise variance as fixed hyperparameters, it would also be possible to infer these as well in an extension of the method. *MGVI* has already permitted the inference of rather complex hierarchical Bayeisan models [25]. Yet, the implementation of our model with fixed noise variance and power spectra was able to deliver competitive results with regard to state-of-the-art methods in the benchmarks. In particular, our method seems to be slightly superior in the low sample regime, probably due to the more appropriate Poisson statistic used. We consider this as an encouraging result for a first work in the context of information field theory-based causal inference.

Author Contributions: Conceptualization, M.K. and T.E.; methodology, M.K. and T.E.; software, M.K.; validation, M.K. and T.E.; formal analysis, M.K.; investigation, M.K. and T.E.; resources, T.E.; data curation, M.K.; writing–original draft preparation, M.K.; writing–review and editing, T.E.; visualization, M.K.; supervision, T.E.; project administration, T.E. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Explicit Derivations

We give explicit derivations for the obtained results

Appendix A.1. Saddle Point Approximation for the Derivation of the Cause Likelihood

Marginalizing β we get

$$\begin{aligned}\mathcal{P}(\boldsymbol{x}|P_\beta, X \to Y) &= \frac{1}{N!}\int \mathcal{D}[\beta]\mathcal{P}(x|\beta, X \to Y)\mathcal{P}(\beta|P_\beta) \\ &= \frac{1}{N!}|2\pi B|^{-\frac{1}{2}}\int \mathcal{D}[\beta]\frac{e^{\boldsymbol{k}^\dagger\beta(\boldsymbol{z})-\boldsymbol{\rho}^\dagger e^{\beta(\boldsymbol{z})}}}{\prod_j k_j!}e^{-\frac{1}{2}\beta^\dagger B^{-1}\beta} = \\ &= \frac{|2\pi B|^{-\frac{1}{2}}}{N!\prod_j k_j!}\int \mathcal{D}[\beta]e^{-\gamma[\beta]} \end{aligned} \tag{A1}$$

where

$$\gamma[\beta] \equiv -\boldsymbol{k}^\dagger\beta(\boldsymbol{z}) + \boldsymbol{\rho}^\dagger e^{\beta(\boldsymbol{z})} + \frac{1}{2}\beta^\dagger B^{-1}\beta. \tag{A2}$$

We approach this integration by a saddle point approximation. In the following we will denote the functional derivative by ∂, i.e., $\partial_{f_z} \equiv \frac{\delta}{\delta f(z)}$.

Taking the first and second order functional derivative of γ w.r.t. β, we get

$$\begin{aligned}\partial_\beta\gamma[\beta] &= -\boldsymbol{k}^\dagger + \rho(e^{\beta(\boldsymbol{z})})^\dagger + \beta^\dagger B^{-1} \\ \partial_\beta\partial_\beta\gamma[\beta] &= \widehat{\rho e^{\beta(\boldsymbol{z})}} + B^{-1}.\end{aligned}$$

The above derivatives are still defined in the space of functions $\mathbb{R}^{[0,1]}$, that is

$$\begin{aligned}\boldsymbol{k}^\dagger_u &\equiv \sum_{j=1}^{n_{\text{bins}}} k_j(\tilde{R}_j)_u \\ (\widehat{\rho e^{\beta(\boldsymbol{z})}})_{uv} &= \rho\sum_{j=1}^{n_{\text{bins}}}(\tilde{R}_j)_u(\tilde{R}_j)_v e^{\beta(u)}.\end{aligned}$$

The latter expression therefore represents a diagonal operator with $e^{\beta(z)}$ as diagonal entries. Let β_0 denote the function that minimizes the functional γ, i.e.,

$$\left.\frac{\delta\gamma[\beta]}{\delta\beta}\right|_{\beta=\beta_0} = 0.$$

We expand the functional γ up to second order around β_0,

$$\begin{aligned}\int \mathcal{D}[\beta]e^{-\gamma[\beta]} &= \int \mathcal{D}[\beta]e^{-\gamma[\beta_0]-(\frac{\delta\gamma[\beta]}{\delta\beta}|_{\beta=\beta_0})^\dagger\beta-\frac{1}{2}\beta^\dagger(\frac{\delta^2\gamma[\beta]}{\delta\beta^\dagger\beta}|_{\beta=\beta_0})\beta+\mathcal{O}(\beta^3)}\\ &\approx e^{-\gamma[\beta_0]}\left|2\pi\left(\frac{\delta^2\gamma[\beta]}{\delta\beta^2}|_{\beta=\beta_0}\right)^{-1}\right|^{\frac{1}{2}}\\ &= e^{+\boldsymbol{k}^\dagger\boldsymbol{\beta_0}-\boldsymbol{\rho}^\dagger e^{\boldsymbol{\beta_0}}-\frac{1}{2}\beta_0^\dagger B^{-1}\beta_0}\left|\frac{1}{2\pi}(\widehat{\rho e^{\boldsymbol{\beta_0}}}+B^{-1})\right|^{-\frac{1}{2}},\end{aligned} \tag{A3}$$

where we dropped higher order terms of β, used that the gradient at $\beta=\beta_0$ vanishes and evaluated the remaining Gaussian integral.

Plugging the result of Equation (A3) into Equation (A1) and using

$$|2\pi B|^{-\frac{1}{2}}\left|\frac{1}{2\pi}(\widehat{\rho e^{\boldsymbol{\beta_0}}}+B^{-1})\right|^{-\frac{1}{2}} = \left|B(\widehat{\rho e^{\boldsymbol{\beta_0}}}+B^{-1})\right|^{-\frac{1}{2}} = \left|\rho B\widehat{e^{\boldsymbol{\beta_0}}}+\mathbb{1}\right|^{-\frac{1}{2}}$$

we get

$$\begin{aligned}\mathcal{P}(\boldsymbol{x}|P_\beta, X\to Y) &\approx \frac{1}{N!}\frac{e^{+\boldsymbol{k}^\dagger\boldsymbol{\beta_0}-\boldsymbol{\rho}^\dagger e^{\boldsymbol{\beta_0}}-\frac{1}{2}\beta_0^\dagger B^{-1}\beta_0}}{\left|\rho B\widehat{e^{\boldsymbol{\beta_0}}}+\mathbb{1}\right|^{\frac{1}{2}}\prod_j k_j!}, \textit{and}\\ \mathcal{H}(\boldsymbol{x}|P_\beta, X\to Y) &\approx \mathcal{H}_0+\frac{1}{2}\log|\rho B\widehat{e^{\boldsymbol{\beta_0}}}+\widehat{\mathbb{1}}|+\log(\prod_j k_j!)-\boldsymbol{k}^\dagger\boldsymbol{\beta_0}+\boldsymbol{\rho}^\dagger e^{\boldsymbol{\beta_0}}+\frac{1}{2}\beta_0^\dagger B^{-1}\beta_0,\end{aligned}$$

Appendix A.2. Explicit Derivation of the Effect Likelihood

Here we will derive the explicit expression of the likelihood of the effect, which can be carried out analytically. We start with the expression for the likelihood of the effect data $\boldsymbol{y}$, given the cause data $\boldsymbol{x}$, the power spectrum P_f, the noise variance ς^2 and the causal direction $X\to Y$. That is, we simply marginalize over the possible functions f and noise $\boldsymbol{\epsilon}$.

$$\begin{aligned}\mathcal{P}(\boldsymbol{y}|\boldsymbol{x}, P_f, \varsigma, X\to Y) &= \int \mathcal{D}[f]\,\mathrm{d}^N\boldsymbol{\epsilon}\,\mathcal{P}(\boldsymbol{y}|\boldsymbol{x}, f, \boldsymbol{\epsilon}, X\to Y)\mathcal{P}(\boldsymbol{\epsilon}|\varsigma)\mathcal{P}(f|P_f)\\ &= \int \mathcal{D}[f]\,\mathrm{d}^N\boldsymbol{\epsilon}\,\delta(\boldsymbol{y}-f(\boldsymbol{x})-\boldsymbol{\epsilon})\mathcal{N}(\boldsymbol{\epsilon}|0,\mathcal{E})\mathcal{N}(f|0,F)\end{aligned}$$

Above we just used the equation $\boldsymbol{y}=f(\boldsymbol{x})+\boldsymbol{\epsilon}$ and used the distributions for f and $\boldsymbol{\epsilon}$. We will now use the Fourier representation of the delta distribution, specifically $\delta(x)=\int\frac{\mathrm{d}q}{2\pi}e^{iqx}$.

$$\delta(\boldsymbol{y}-f(\boldsymbol{x})-\boldsymbol{\epsilon}) = \int\frac{\mathrm{d}^N\boldsymbol{q}}{(2\pi)^N}e^{i\boldsymbol{q}^\dagger(\boldsymbol{y}-\boldsymbol{\epsilon}-f(\boldsymbol{x}))} = \int\frac{\mathrm{d}^N\boldsymbol{q}}{(2\pi)^N}e^{i\boldsymbol{q}^\dagger(\boldsymbol{y}-\boldsymbol{\epsilon}-f(\boldsymbol{x}))}$$

Once more we employ a vector of response operators, mapping $\mathbb{R}^{\mathbb{R}}$ to $\mathbb{R}^N$,

$$\boldsymbol{R}_x \equiv (R_{1x},...,R_{Nx})^T = (\delta(x-x_1),...,\delta(x-x_N))^T.$$

This allows to represent the evaluation $f(\boldsymbol{x}) = \boldsymbol{R}^\dagger f$, i.e., as a linear dot-product. Using the well known result for Gaussian integrals with linear terms (see, e.g., [26]),

$$\int \mathcal{D}[u] e^{-\frac{1}{2}u^\dagger A u + b^\dagger u} = \left|\frac{A}{2\pi}\right|^{-\frac{1}{2}} e^{\frac{1}{2}b^\dagger A b} \tag{A4}$$

We are able to analytically do the path integral over f,

$$\begin{aligned}\mathcal{P}(\boldsymbol{y}|\boldsymbol{x}, P_f, \varsigma, X \to Y) &= |2\pi F|^{-\frac{1}{2}} \int \mathcal{D}[f]\, \mathrm{d}^N\boldsymbol{\epsilon} \frac{\mathrm{d}^N\boldsymbol{q}}{(2\pi)^N} e^{i\boldsymbol{q}^\dagger(\boldsymbol{y}-\boldsymbol{\epsilon}-\boldsymbol{R}^\dagger f) - \frac{1}{2}f^\dagger F^{-1} f} \mathcal{N}(\boldsymbol{\epsilon}|0, \mathcal{E}) \\ &= \int \mathrm{d}^N\boldsymbol{\epsilon} \frac{\mathrm{d}^N\boldsymbol{q}}{(2\pi)^N} e^{i\boldsymbol{q}^\dagger(\boldsymbol{y}-\boldsymbol{\epsilon}) + (-i)^2\frac{1}{2}\boldsymbol{q}^\dagger\boldsymbol{R}^\dagger F\boldsymbol{R}\boldsymbol{q}} \mathcal{N}(\boldsymbol{\epsilon}|0, \mathcal{E})\end{aligned}$$

Now, we do the integration over the noise variable, $\boldsymbol{\epsilon}$, by using the equivalent of Equation (A4) for the vector-valued case:

$$\begin{aligned}\mathcal{P}(\boldsymbol{y}|\boldsymbol{x}, P_f, \varsigma, X \to Y) &= |2\pi\mathcal{E}|^{-\frac{1}{2}} \int \mathrm{d}^N\boldsymbol{\epsilon} \frac{\mathrm{d}^N\boldsymbol{q}}{(2\pi)^N} e^{i\boldsymbol{q}^\dagger(\boldsymbol{y}-\boldsymbol{\epsilon}) - \frac{1}{2}\boldsymbol{q}^\dagger\boldsymbol{R}^\dagger F\boldsymbol{R}\boldsymbol{q} - \frac{1}{2}\boldsymbol{\epsilon}^\dagger\mathcal{E}^{-1}\boldsymbol{\epsilon}} \\ &= \int \frac{\mathrm{d}^N\boldsymbol{q}}{(2\pi)^N} e^{i\boldsymbol{q}^\dagger\boldsymbol{y} - \frac{1}{2}\boldsymbol{q}(\boldsymbol{R}^\dagger F\boldsymbol{R} + \mathcal{E})\boldsymbol{q}}\end{aligned}$$

In the following we will write $\mathbb{R}^{N\times N} \ni \tilde{F} = \boldsymbol{R}^\dagger F \boldsymbol{R}$, with entries $\tilde{F}_{ij} = F(x_i, x_j)$. The integration over the Fourier modes $\boldsymbol{q}$, again via the multivariate equivalent of Equation (A4), will give the preliminary result:

$$\begin{aligned}\mathcal{P}(\boldsymbol{y}|\boldsymbol{x}, P_f, \varsigma, X \to Y) &= \int \frac{\mathrm{d}^N\boldsymbol{q}}{(2\pi)^N} e^{i\boldsymbol{q}^\dagger\boldsymbol{y} - \frac{1}{2}\boldsymbol{q}(\tilde{F}+\mathcal{E})\boldsymbol{q}} \\ &= (2\pi)^{-\frac{N}{2}} \left|\tilde{F} + \mathcal{E}\right|^{-\frac{1}{2}} e^{-\frac{1}{2}\boldsymbol{y}^\dagger(\tilde{F}+\mathcal{E})^{-1}\boldsymbol{y}}\end{aligned}$$

References

1. Kurthen, M. Bayesian Causal Inference. Master's Thesis, Ludwig Maximilian University, Munich, Germany, 2018.
2. Mooij, J.M.; Peters, J.; Janzing, D.; Zscheischler, J.; Schölkopf, B. Distinguishing Cause from Effect Using Observational Data: Methods and Benchmarks. *J. Mach. Learn. Res.* **2016**, *17*, 1–102.
3. Hernán, M.A.; Hsu, J.; Healy, B. Data science is science's second chance to get causal inference right: A classification of data science tasks. *arXiv* **2018**, arXiv:1804.10846.
4. Pearl, J. Theoretical Impediments to Machine Learning With Seven Sparks from the Causal Revolution. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining WSDM '18*; ACM: New York, NY, USA, 2018; p. 3.
5. Peters, J.; Janzing, D.; Schölkopf, B. *Elements of Causal Inference: Foundations and Learning Algorithms*; MIT Press: Cambridge, MA, USA, 2017.
6. Daniusis, P.; Janzing, D.; Mooij, J.; Zscheischler, J.; Steudel, B.; Zhang, K.; Schölkopf, B. Inferring deterministic causal relations. In Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence, Catalina Island, CA, USA, 8–11 July 2010; AUAI Press: Corvallis, OR, USA, 2010; pp. 143–150.
7. Pearl, J. *Causality: Models, Reasoning, and Inference*; Cambridge University Press: New York, NY, USA, 2000; pp. 44 and 70.
8. Spirtes, P.; Zhang, K. Causal discovery and inference: Concepts and recent methodological advances. *Appl. Inform.* **2016**, *3*, 3. [CrossRef] [PubMed]
9. Mitrovic, J.; Sejdinovic, D.; Teh, Y.W. Causal Inference via Kernel Deviance Measures. In *Advances in Neural Information Processing Systems 31*; Curran Associates, Inc.: Red Hook, NY, USA, 2018; pp. 6986–6994.

10. Shimizu, S.; Hoyer, P.O.; Hyvärinen, A.; Kerminen, A. A linear non-Gaussian acyclic model for causal discovery. *J. Mach. Learn. Res.* **2006**, *7*, 2003–2030.
11. Hoyer, P.O.; Janzing, D.; Mooij, J.M.; Peters, J.; Schölkopf, B. Nonlinear causal discovery with additive noise models. In *Advances in Neural Information Processing Systems 21*; Koller, D., Schuurmans, D., Bengio, Y., Bottou, L., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2009; pp. 689–696.
12. Stegle, O.; Janzing, D.; Zhang, K.; Mooij, J.M.; Schölkopf, B. Probabilistic latent variable models for distinguishing between cause and effect. In *Advances in Neural Information Processing Systems 23*; Lafferty, J.D., Williams, C.K.I., Shawe-Taylor, J., Zemel, R.S., Culotta, A., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2010; pp. 1687–1695.
13. Goudet, O.; Kalainathan, D.; Caillou, P.; Lopez-Paz, D.; Guyon, I.; Sebag, M.; Tritas, A.; Tubaro, P. Learning Functional Causal Models with Generative Neural Networks. *arXiv* **2017**, arXiv:1709.05321.
14. Friedman, N.; Koller, D. Being Bayesian about network structure. A Bayesian approach to structure discovery in Bayesian networks. *Mach. Learn.* **2003**, *50*, 95–125. [CrossRef]
15. Heckerman, D.; Meek, C.; Cooper, G. A Bayesian approach to causal discovery. In *Innovations in Machine Learning*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 1–28.
16. Enßlin, T.A.; Frommert, M.; Kitaura, F.S. Information field theory for cosmological perturbation reconstruction and nonlinear signal analysis. *Phys. Rev. D* **2009**, *80*, 105005. [CrossRef]
17. Rasmussen, C.E.; Williams, C.K. *Gaussian Processes for Machine Learning*; Adaptive Computation and Machine Learning; MIT Press: Cambridge, MA, USA, 2006; pp. 13, 248.
18. Chatfield, C. *The Analysis of Time Series: An Introduction*, 6th ed.; Chapman & Hall/CRC Texts in Statistical Science, CRC Press: Boca Raton, FL, USA, 2016; pp. 109–114.
19. Nocedal, J.; Wright, S.J. *Numerical Optimization*, 2nd ed.; Springer: New York, NY, USA, 2006.
20. Virtanen, P.; Gommers, R.; Oliphant, T.; Haberland, M.; Reddy, T.; Cournapeau, D.; Burovski, E.; Peterson, P.; Weckesser, W.; Bright, J.; et al. SciPy 1.0–Fundamental Algorithms for Scientific Computing in Python. *arXiv* **2019**, arXiv:1907.10121.
21. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. *arXiv* **2016**, arXiv:1603.04467.
22. Majerski, P. Simple error bounds for the multivariate Laplace approximation under weak local assumptions. *arXiv* **2015**, arXiv:1511.00302.
23. Caldwell, A.; Schick, R.C.; Schulz, O.; Szalay, M. Integration with an Adaptive Harmonic Mean Algorithm. *arXiv* **2018**, arXiv:1808.08051.
24. Knollmüller, J.; Enßlin, T.A. Metric Gaussian Variational Inference. *arXiv* **2019**, arXiv:1901.11033.
25. Hutschenreuter, S.; Enßlin, T.A. The Galactic Faraday depth sky revisited. *arXiv* **2019**, arXiv:1903.06735.
26. Greiner, W.; Bromley, D.; Reinhardt, J. *Field Quantization*; Springer: Berlin/Heidelberg, Germany, 2013; p. 353.