AGM-TE: Approximate Generative Model Estimator of Transfer Entropy for Causal Discovery

Daniel Kornai
Department of Physics and Astronomy, University College London
Ricardo Silva
Department of Statistical Science, University College London
Nikolaos Nikolaou
Department of Physics and Astronomy, University College London

Editors: Biwei Huang and Mathias Drton

Abstract

The discovery of causal interactions from time series data is an increasingly common approach in science and engineering. Many of the approaches for solving it rely on an information-theoretic measure called *transfer entropy* [TE] to infer directed causal interactions. However, TE is difficult to estimate from empirical data, as non-parametric methods are hindered by the curse of dimensionality, while existing ML methods suffer from slow convergence or overfitting.

In this work, we introduce AGM-TE, a novel ML method that estimates TE using the difference in the predictive capabilities of two alternative probabilistic forecasting models. In a comprehensive suite of TE estimation benchmarks [with 100+ tasks], AGM-TE achieves SoTA results in terms of accuracy and data efficiency when compared to existing non-parametric and ML estimators. AGM-TE further differentiates itself with the ability to estimate *conditional transfer entropy*, which helps mitigate the effect of confounding variables in systems with many interacting components. We demonstrate the strengths of our approach empirically by recovering patterns of brain connectivity from 250+ dimensional spike data that are consistent with known neuroanatomical results.

Overall, we believe AGM-TE represents a significant step forward in the application of transfer entropy to problems of causal discovery from observational time series data.

Keywords: Causal Discovery, Causal Models, Machine Learning, Transfer Entropy

1. Introduction: Causality, Causal Discovery, and Transfer Entropy

A fundamental goal of science and engineering is to understand *causal interactions* between variables in the systems that we study and design. Following Pearl (2010), we say that X causes Y [denoted as $X \to Y$] if and only if an intervention on X [denoted $\text{do}(X = x)$] has an effect on $p(y|\text{do}(X = x))$, the post-interventional distribution of Y. As correlation alone does not imply causation, the interventional distribution $p(y|\text{do}(x))$ may differ substantially from the conditional $p(y|x)$. For example, since sunlight (Y) is required for plant growth, it correlates with plant height (X). However, intervening on plant height [say by providing fertiliser] will clearly not change how much the sun shines. That is, $p(y|\text{do}(x)) \neq p(y|x)$ for sunlight Y and plant growth X.

1.1. Data Driven Causal Discovery

The traditional approach by which causal interactions are discovered is to conduct experiments, which allow us to directly observe the consequences of a given intervention. However, in many fields such as neuroscience, earth system science, or economics, manipulations of the complex

© 2025 D. Kornai, R. Silva & N. Nikolaou.

system being studied are either expensive, unethical, or impossible. Currently, the main alternative is computer simulations. However, these are often time-consuming, computationally expensive, and also require substantial amounts of expert knowledge, which may in turn impose potentially unwarranted assumptions on the systems being studied (Runge et al., 2019).

This constraint on the availability of *interventional data* stands in increasingly stark contrast to the vast amounts of *observational data* that can now be acquired due to advancements in recording technology in many domains (Sozzi et al., 2021; Steinmetz et al., 2021; Urai et al., 2022). This has motivated the development of approaches that can infer properties of the data generating process directly from observational datasets. Such *data driven causal discovery* methods have found applications in diverse fields such as epidemiology (Liang and Mikler, 2014), neuroscience (Matiasz et al., 2017), industrial control systems (Chatterjee and Dethlefs, 2020), materials science (Sun et al., 2021), and climate science (Böhnisch et al., 2023).

1.2. Causal Discovery in Timeseries

Causal discovery methods rely on assumptions about the data generating and observation processes (Spirtes and Zhang, 2016). For causal discovery methods operating on timeseries data, the core assumption about the data generating process is that causal structures have to be consistent with time order. That is, we assume *cause precedes effect*. Given this assumption, causation of X by Y implies that the past of Y influences the present [and future] of X.

To express this idea more precisely, let us introduce some notation. Consider the temporally discrete time series Y, with observations at each timestep $t \in \{0, 1, ..., T\}$. For a given timestep t, the value at the next timestep is denoted as y_{t+1}. The notation $\mathbf{y_{t-}}$ is a shorthand for $y_{1:t}$, the past values of Y up to and including time t. $y_{t+1}|\mathbf{y_{t-}}$ denotes the value of y_{t+1} given the past of Y. The notation for the values of the second time series X is similar (e.g. $\mathbf{x_{t-}}$). We can then say that X *causes* Y *if* Y *is not independent of the past of* X *after conditioning on the past of* Y (Peters et al., 2017). The converse statement concerning the lack of causation if Y is independent of the past of X after conditioning on the past of Y also true.

$$y_{t+1} \not\perp\!\!\!\perp \mathbf{x_{t-}} | \mathbf{y_{t-}} \iff y_{t+1}|\mathbf{y_{t-}} \neq y_{t+1}|\mathbf{x_{t-}}, \mathbf{y_{t-}} \implies X \rightarrow Y \tag{1}$$

$$y_{t+1} \perp\!\!\!\perp \mathbf{x_{t-}} | \mathbf{y_{t-}} \iff y_{t+1}|\mathbf{y_{t-}} = y_{t+1}|\mathbf{x_{t-}}, \mathbf{y_{t-}} \implies X \not\rightarrow Y \tag{2}$$

This captures the notion of direct causation in the interventional framework under the assumption of faithfulness if no unmeasured confounders exist (Zhang and Spirtes, 2015).

The framework of *predictive causality* (Wiener, 1956) is one way to translate this link between conditional independence and causality into a method for causal discovery. For dependent variables X and Y, we say that X *causes* Y *if the prediction of a future value of* Y *is significantly enhanced by the history of* X. To measure this improvement in predictions, Granger (1969) introduced the idea of using two autoregressive forecasting models for the future of Y: one considering only the past of Y, the other considering the past of both Y and X. A model comparison procedure can reveal [or rule out] a predictive improvement due to additional information from X, and thereby infer the presence of a causal interaction. If the models are similarly good, then X and Y are conditionally independent given the past of Y, and therefore $X \not\rightarrow Y$. If the model using both Y and X is better at predicting Y, this suggests increasing evidence in favour of $X \rightarrow Y$ [See Appendix A for details].

1.3. Defining Transfer Entropy

Granger causality is a special case [for linearly interacting Gaussian variables] of a more general information-theoretic concept called *transfer entropy* (Schreiber, 2000; Barnett et al., 2009; Barrett et al., 2010). Understanding transfer entropy requires first introducing the *generalised Markov property*. Recall our notation for the temporally discrete time series Y. $\mathbf{y_t}$ is a shorthand for the past k values of Y, that is $y_{t-k+1:t}$. $P(y_{t+1}|\mathbf{y_t})$ then denotes the probability distribution of a specific y_{t+1} given the past k past values of Y. If we also consider a time series X, we can say that the combined system $S(Y, X)$ has the generalized Markov property, if:

$$P(y_{t+1}|\mathbf{y_t}) = P(y_{t+1}|\mathbf{x_t}, \mathbf{y_t}), \text{ for } t \in 0 : T - 1$$

that is, conditioning on the past values of X does not influence the future probabilities of Y if we have already conditioned on the past of Y. Note how this is a probabilistic version of Eq. 1.

Transfer entropy (denoted $T_{X \to Y}$) simply measures the deviation from the generalized Markov property. It can be equivalently formulated as conditional mutual information [CMI], a KL divergence, or a difference of conditional entropies (Mondal et al., 2020)

$$\begin{aligned} \mathcal{T}_{X \to Y} &:= \mathcal{I}(\mathbf{x_t}; y_{t+1}|\mathbf{y_t}) && (3) \\ &:= D_{KL}(P(\mathbf{x_t}, y_{t+1}\mathbf{y_t})||P(\mathbf{x_t}, \mathbf{y_t})P(y_{t+1}|\mathbf{y_t})) && (4) \\ &:= \mathcal{H}(y_{t+1}|\mathbf{y_t}) - \mathcal{H}(y_{t+1}|\mathbf{x_t}, \mathbf{y_t}) && (5) \end{aligned}$$

Transfer entropy is therefore measuring the additional information [reduction in uncertainty] on y_{t+1} available in the past of X, that is not already captured in the past of Y. In all cases, $\mathcal{T}_{X \to Y} \geq 0$, as considering the past of an independent X will not increase our uncertainty in Y. If $\mathcal{T}_{X \to Y} = 0$, there is no information flow from X to Y, therefore $X \not\to Y$. Conversely, $T_{X \to Y} \gg 0$ implies a directed causal interaction $X \to Y$, with the magnitude quantifying its strength.

1.4. Applications of Transfer Entropy to Causal Discovery

Transfer entropy estimation methods can complement existing causal discovery approaches by driving feature selection or filtering decisions [see Wollstadt et al. (2019), Assaad et al. (2022), Castri et al. (2023), Bonetti et al. (2024)]. It is also possible to infer causal relationships directly using estimated TE values. For example, Bauer et al. (2007) use TE to identify the direction of disturbance propagation in a chemical plant, Shovon et al. (2016) use TE to infer brain connectivity from EEG data, while Kim et al. (2020) use TE to infer gene regulatory networks.

These applications in causal discovery all depend on methods that infer transfer entropy from data. Unfortunately, information-theoretic quantities are notoriously difficult to estimate from finite data (Sricharan et al., 2013; McAllester and Stratos, 2020). As such, existing implementations of the three major classes of TE estimation methods each have their own significant issues relating to accuracy and sample size requirements [see Section 2]. Indeed, difficulties with "practical computability" have been one of the longest standing critiques of TE as a measure of causality (Runge et al., 2012; Runge, 2018; Castri et al., 2023).

We therefore set out to improve the computability of TE from empirical data, as we believe this could help TE estimation become more widely adopted in causal discovery.

2. Background: Estimating Transfer Entropy from Data

There are three major classes of approaches for estimating transfer entropy from data. These each use one of the three TE definitions presented in the previous section, with *kNN-KSG*, *Variational ML*, and *Cross Entropy ML* methods using the formulations in Eqs. 3, 4 and 5 respectively.

In this section, we present a historical overview of the development of these estimation paradigms. We detail the theoretical and implementational challenges associated with computability in each approach in order to demonstrate the longstanding difficulty of the TE estimation task and to better contextualise our novel method that will be introduced in Section 3.

2.1. kNN-KSG

The oldest class of methods for TE estimation use a k nearest neighbours [*kNN*] approach to estimate the probability densities that are used to define entropy. These methods [such as Lindner et al. (2011)], are based on the approach of Frenzel and Pompe (2007), which itself generalises the *KSG* method of Kraskov et al. (2004) for estimating MI. The fundamental idea of such kNN TE estimation methods is to take the decomposition of the CMI corresponding to TE

$$\mathcal{T}_{X\to Y} := \mathcal{I}(\mathbf{x_t}; y_{t+1}|\mathbf{y_t}) = \mathcal{H}(y_{t+1}, \mathbf{y_t}) + \mathcal{H}(\mathbf{x_t}, \mathbf{y_t}) - \mathcal{H}(\mathbf{y_t}) - \mathcal{H}(y_{t+1}, \mathbf{x_t}, \mathbf{y_t})$$

and use the estimator of Kozachenko and Leonenko (1987)[Appendix B.1.1] for each entropy term.

Due to their long history and good empirical performance in simple systems, kNN-KSG methods remain a widespread approach. However, both theory and empirical results show that these KSG approaches suffer from the curse of dimensionality, with performance degrading exponentially for vector-valued variables (Sricharan et al., 2013; Gao et al., 2018; Zhao and Lai, 2020).

2.2. Variational ML

Working separately from practitioners interested in estimating CMI for causality, machine learning researchers were interested in measuring the mutual information between high-dimensional input data [such as images] and learned representations. To develop a MI estimator that can cope with variables of such high dimensionality, Belghazi et al. (2018) developed an innovative new class of methods which have since proven useful in estimating transfer entropy. In these *variational machine learning* methods, $\mathcal{I}(\mathbf{x_t}; y_{t+1}|\mathbf{y_t})$ is estimated by using neural networks to parametrise the Donsker-Varadhan variational lower bound on the KL divergence between $p(\mathbf{x_t}, y_{t+1}, \mathbf{y_t})$, and $p(\mathbf{x_t}, \mathbf{y_t})p(y_{t+1}|\mathbf{y_t})$ [See Appendix B.2 for details].

$$\begin{aligned}\mathcal{T}_{X\to Y} &:= D_{KL}(P(\mathbf{x_t}, y_{t+1}\mathbf{y_t})||P(\mathbf{x_t}, \mathbf{y_t})P(y_{t+1}|\mathbf{y_t}))\\ &\geq DV(P(\mathbf{x_t}, y_{t+1}\mathbf{y_t})||P(\mathbf{x_t}, \mathbf{y_t})P(y_{t+1}|\mathbf{y_t}))\end{aligned}$$

Variational approaches have significantly improved performance in vector-valued data compared to kNN-KSG methods, scaling up to 100-dimensional variables [see Mukherjee et al. (2019) and Mondal et al. (2020)]. However, McAllester and Stratos (2020) proved that any distribution-free high confidence lower bound on mutual information cannot be larger than $\mathcal{O}(\ln N)$ [Where N is sample size]. Since this subsumes the Donsker-Varadhan bound as a special case, variational CMI estimators [Mukherjee et al. (2019), Mondal et al. (2020)] and TE estimators [Zhang et al. (2019), Luxembourg et al. (2024)] require exponentially large datasets.

2.3. Cross Entropy ML

The theoretical arguments of McAllester and Stratos (2020) also suggest that methods using upper bounds on entropies will not suffer from the convergence issues of variational approaches.

One such upper bound, the cross entropy [CE] quantifies the expected level of *surprisal* when using $q(x)$ to model observations drawn from some true $x \sim p(x)$. The CE is an upper bound on the entropy [expected surprisal] of the true distribution $p(x)$, as it is the sum of this true entropy and the non-negative KL divergence [relative entropy, or excess surprisal] of $p(x)$ with respect to $q(x)$.

$$\begin{aligned} H(p(x), q(x)) &:= \mathbb{E}_{x \sim p(x)}[-\log q(x)] \\ &:= \mathcal{H}(p(x)) + D_{KL}(p(x)||q(x)) \end{aligned}$$

If the true $p(x)$ is unknown, but we have access to n samples of $x \sim p(x)$, we can empirically estimate the cross entropy as

$$H(p(x), q(x)) \approx \hat{H}(p(x), q(x)) := \frac{1}{N} \sum\nolimits_{i=1}^{N} [-\log q(x)]$$

As $\mathcal{H}(p(x))$ is fixed, fitting a [sufficiently expressive] model $q(x)$ to a dataset by minimising the empirical cross entropy will make our estimated $\hat{H}(p(x), q(x))$ converge to $\mathcal{H}(p(x))$.

Shalev et al. (2020) were the first to apply this approach to estimate the conditional entropies in the definition of TE (Eq. 5). They use a neural network to parametrise a discrete categorical distribution over y_{t+1} given inputs from $\mathbf{y_t}$, and estimate $\mathcal{H}(y_{t+1}|\mathbf{y_t})$ using the empirical CE:

$$\mathcal{H}(y_{t+1}|\mathbf{y_t}) \approx \hat{H}(y_{t+1}|\mathbf{y_t}) := \frac{1}{T} \sum\nolimits_{i=1}^{T} [-\log q(y_{t+1}|\mathbf{y_t})]$$

After training a second model to parametrise a discrete distribution over y_{t+1} given both $\mathbf{y_t}$ and $\mathbf{x_t}$, we estimate $\mathcal{T}_{X \to Y}$ by subtracting the resulting empirical cross entropies.

$$\begin{aligned} \mathcal{T}_{X \to Y} &:= \mathcal{H}(y_{t+1}|\mathbf{y_t}) - \mathcal{H}(y_{t+1}|\mathbf{x_t}, \mathbf{y_t}) \\ &\approx \frac{1}{T} \sum\nolimits_{t=1}^{T} [-\log q(y_{t+1}|\mathbf{y_t})] - \frac{1}{T} \sum\nolimits_{t=1}^{T} [-\log q(y_{t+1}|\mathbf{y_t}, \mathbf{x_t})] \end{aligned}$$

The NJEE method of Shalev et al. (2020) and the later DETE approach of Garg et al. (2022) implementing this idea have favourable theoretical convergence and consistency properties, and were shown to outperform kNN-KSG and variational ML estimators in synthetic data benchmarks.

Outlook for Cross Entropy TE Estimators

All causal discovery methods rely on assumptions about the data generating and observation processes. This means that the extent to which a given method can be successful depends on the validity of the assumptions for the systems being investigated (Runge, 2018).

As such, while DETE and NJEE have solved the problems of exponentially degrading performance in kNN-KSG methods and the unrealistic sample size requirements of variational ML approaches, we nonetheless believe there is a significant opportunity for improving the computability of TE with cross entropy methods if we are able to identify and mitigate two key limiting assumptions inherent to current implementations. In the upcoming section, we will introduce our proposal for mitigating *truncation error* and *categorical overfitting*.

3. Our Novel Method: AGM-TE

The main motivation behind our proposed method is to improve the practical computability of TE by understanding and mitigating some of the unwarranted assumptions imposed by current cross entropy approaches. In this section, we start by introducing these two assumptions, and detail how methods from the probabilistic modelling of dynamical systems [see Salinas et al. (2017); Rangapuram et al. (2018); Lin et al. (2022)] are incorporated into TE estimation in the new approach we named *Approximate Generative Model* estimator of *Transfer Entropy* or AGM-TE[1].

We then detail the model architecture and training procedure incorporating these results, and present comprehensive comparative benchmarks with existing estimators from the three major classes to establish the validity and efficacy of our approach. Finally, we discuss how *conditional transfer entropy* can help mitigate the effect of confounding variables in systems with many interacting components, and how our method estimates this quantity.

3.1. Motivating AGM-TE

The first unwarranted assumption [source of error] relates to the estimation of causal effects in the presence of large temporal delays. Runge et al. (2012) and Castri et al. (2023) argue that existing TE estimation methods with a fixed finite Markov order introduce significant *truncation error*, as the original definition of TE considers all possible past timesteps. Such truncation error would lead to an underestimation of TE when the sampling frequency is high, relative to the timescales of the processes being investigated. We believe that the use of recurrent neural networks [RNNs] could help mitigate this first issue, as RNNs can approximate dynamical systems [including their long-range dependencies] to arbitrary accuracy (Schäfer and Zimmermann, 2006; Chen et al., 2023).

The second [and perhaps more serious] issue relates to the use of categorical distributions by DETE and NJEE for modelling observations. First, as a discrete distribution with finite support, modelling observations from any other type of data generating distributions induces some combination of truncation and discretisation error. Second, from the perspective of the categorical cross entropy loss, all unobserved values around a given y_{t+1} are equally improbable. This assumption is trivially violated for many distributions [such as the Gaussian], and also incentivises *overfitting* to observations. This overfitting leads to an underestimation of uncertainty that cascades into an overestimation of TE [see Appendix C for a detailed demonstration and explanation]. A well-calibrated probabilistic model for our observations could potentially remedy this issue.

We therefore set out to produce a data efficient cross entropy TE estimator that supports a wide variety of data types [discrete, continuous, event-based] with appropriate likelihood models, and utilises a recurrent neural network [RNN] for modelling dynamics.

3.2. Architecture of the AGM

The namesake of our method, the AGM, is a probabilistic model of the data generating dynamical system. It combines a neural network *latent dynamics model* and a parametric *observation model* to yield a predicted distribution over y_{t+1} given past observations of y_t [or alternatively both x_t and y_t]. The role of the neural network is to facilitate the learning of complex temporal dynamics, while the parametric observation model ensures that the uncertainty of the model is correctly calibrated to the inherent stochasticity of the generative process.

[1] The Python implementation of AGM-TE can be found at https://github.com/dkornai/AGM-TE

Here, we briefly detail our approach for the model of $y_{t+1}|\mathbf{y_t}$. For each t, we first map our observed y_t to a latent state s_t using an affine transformation $a()$. The *latent dynamics model* $f_{\theta_f}()$ then uses a recurrent neural network to *predict* the upcoming latent state $\hat{s}_{t+1}$. We then map the latent state to a predicted distribution over y_{t+1} using an *observation model*. This consists of a function $g_{\theta_g}()$ mapping from the predicted latent to a parameter vector ϕ_{t+1}, and q, the user-defined distribution parametrised by ϕ_{t+1} [e.g. multivariate Gaussian, Poisson, Laplace, ...]. By choosing the distribution appropriately for the data type [e.g. a Poisson for event data], the predicted probabilities over y_{t+1} become well-calibrated.

To summarise, the AGM for $y_{t+1}|\mathbf{y_t}$ can be expressed as the following composition

$$y_{t+1} \dot{\sim} q(y_{t+1}; \phi_{t+1} = g_{\theta_g}(f_{\theta_f}(a(y_t)))$$

which is shortened to $q_\Theta(y_{t+1}|\mathbf{y_t})$ for convenience. A more detailed description of these calculations, the NNs used in the latent dynamics model, and an introduction to the various choices for the parametric distribution in the observation model are provided in Appendix D.

3.3. Training Procedure and TE Estimation

The AGM for $y_{t+1}|\mathbf{y_t}, \mathbf{x_t}$ takes as input the concatenation $\{y_t, x_t\}$, and similarly aims to predict the distribution of y_{t+1}. To infer predictive causality, we can compare the ability to forecast y_{t+1} under the base model $q_\Theta(y_{t+1}|\mathbf{y_t})$ and the alternative model $q_\Theta(y_{t+1}|\mathbf{y_t}, \mathbf{x_t})$ to see if considering X has made our predictions better. To do so, we must first fit the models to data.

The AGM for $y_{t+1}|\mathbf{y_t}$ is trained on a $T \times d$ dataset $\mathbf{Y}$, containing observations of the d-dimensional y_t over time $t \in 1 : T$. To measure the fit of the AGM for a given y_{t+1}, we calculate the negative log likelihood [NLL] of drawing the true y_{t+1} from the predicted $q(y_{t+1}|\mathbf{y_t})$. Our loss function to minimise is then the average NLL over the dataset, which is also the empirical estimate of the cross entropy between the data distribution p and our model q (Goodfellow et al., 2016).

$$\text{Loss}(\Theta_1, [\mathbf{Y}]) := \frac{1}{T} \sum\nolimits_{t=1}^{T} [-\log q_{\Theta_1}(y_{t+1}|\mathbf{y_t})] = \hat{H}(p(y_{t+1}|\mathbf{y_t}), q_{\Theta_1}(y_{t+1}|\mathbf{y_t}))$$

The alternative model $q(y_{t+1}|\mathbf{y_t}, \mathbf{x_t})$ is trained with a concatenation of $\mathbf{Y}$ and the $T \times e$ dataset $\mathbf{X}$ [the timeseries of X] to similarly minimise the empirical NLL [cross entropy].

$$\text{Loss}(\Theta_2, [\mathbf{Y}, \mathbf{X}]) := \frac{1}{T} \sum\nolimits_{t=1}^{T} [-\log q_{\Theta_2}(y_{t+1}|\mathbf{y_t}, \mathbf{x_t})] = \hat{H}(p(y_{t+1}|\mathbf{y_t}, \mathbf{x_t}), q_{\Theta_2}(y_{t+1}|\mathbf{y_t}, \mathbf{x_t}))$$

Optimisation of AGM parameters Θ with respect to the loss entails the maximisation of the likelihood of observing each y_{t+1}, which facilitates learning of the underlying temporal dynamics and stochasticity of the data generating process by the AGM. A detailed example of learning $q(y_{t+1}|\mathbf{y_t})$ and $q(y_{t+1}|\mathbf{y_t}, \mathbf{x_t})$ for an input driven dynamical system can be found in Appendix E.

As we previously established, minimisation of the CE results in its gradual convergence to the entropy of the data generating distribution. Therefore, as training progresses, $\text{Loss}(\Theta_1, [\mathbf{Y}])$ and $\text{Loss}(\Theta_2, [\mathbf{YX}])$ become increasingly accurate estimates of $\mathcal{H}(y_{t+1}|\mathbf{y_t})$ and $\mathcal{H}(y_{t+1}|\mathbf{y_t}, \mathbf{x_t})$ respectively. The transfer entropy $\mathcal{T}_{X \to Y}$ can thus be estimated from the data using the models as:

$$\mathcal{T}_{X \to Y} \approx \hat{\mathcal{T}}_{X \to Y} := \text{Loss}(\Theta_1, [\mathbf{Y}]) - \text{Loss}(\Theta_2, [\mathbf{YX}]) \tag{6}$$

Simply put, AGM-TE estimates $\mathcal{T}_{X \to Y}$ as the decrease in uncertainty of a predictive model of y_t that comes from also considering the past of a [potentially] causal variable X. Detailed theoretical arguments for the convergence of this estimator to the true $\mathcal{T}_{X \to Y}$ can be found in Appendix F.

3.4. Validation and Comparative Benchmarks in Synthetic Data

To establish the efficacy of our proposed approach, we compare the performance of AGM-TE to existing kNN-KSG [NPEET (Ver Steeg and Galstyan, 2012)], variational ML [C-MI-GAN (Mondal et al., 2020)], and cross entropy ML approaches [NJEE (Shalev et al., 2020)] in synthetic data. To our knowledge, results from the set of 100+ different TE estimation tasks presented here constitute the most comprehensive quantitative comparison of TE estimation methods to date.

To generate time series datasets, we use two bivariate systems, a linear-Gaussian [LG] model, and a Joint Process [JP] model. These systems have analytically tractable $\mathcal{T}_{X\to Y}$ and $\mathcal{T}_{Y\to X}$ [see Appendix G], which act as the ground truth for our comparison of alternative methods. Full details of the benchmarking methods and a comprehensive discussion of the results can be found in Appendix H. Here, we present a subset that highlights the general trends observed.

To test sample efficiency, we fixed the λ parameters of the 1D LG and JP systems, and generated datasets of increasing length T. For each T, we add the absolute errors across the four estimated TEs of the two systems. Plotting the total error across sample sizes (Fig. 1A), we see that AGM-TE achieves favourable small-sample size performance comparable to non-parametric methods, in contrast to other ML approaches that overestimate TE in small samples (Fig. 22).

We also investigated the ability of the methods to correctly estimate TE as data dimensionality increases. In the *redundant stacking* tests, we concatenate additional channels of i.i.d. noise to data from a 1D system, which should leave the true TE unchanged. These tests highlight how kNN-KSG approaches fail as dimensionality increases, while the performance of ML methods is mostly unaffected (Fig. 1B). In our *linear stacking* tests we concatenate data from d independently simulated 1D systems, causing the true transfer entropy to increase linearly with d. This highlights the comparative advantages of cross entropy ML methods relative to variational ML approaches, which fail to correctly estimate higher TE values due to the lower bound used (Fig. 1C).

The performance of the four methods in the battery of 116 different TE inference tasks [40 1D and 76 multi-D] can be summarised with two metrics. For the 57 cases where the true TE was positive, we measured the percentage error relative to the true value. NPEET had an average percentage error of 46.85%, which fell to 8.18% for C-MI-GAN, and 6.85% for NJEE. In the remaining 59 cases where the true TE is 0, we measured the average estimated TE, which acts as a proxy for false causal positives. NPEET inferred an average TE of 0.0009, which was better than the ML-based C-MI-GAN [0.0034] and NJEE [0.0113]. The best results for both metrics [4.96% and 0.0005] were achieved by AGM-TE, demonstrating that our approach achieves SoTA TE estimation performance.

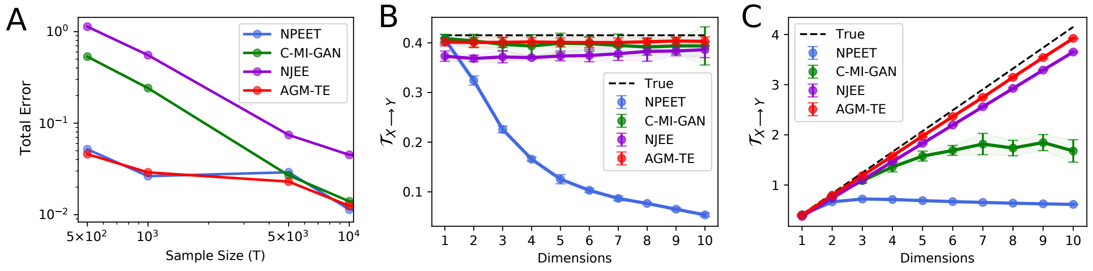

Figure 1: **A:** Total error across four TE estimates from the Joint Process and Linear Gaussian systems at various samples sizes. **B:** Estimates of $\mathcal{T}_{X\to Y}$ for multidimensional variables generated by redundant stacking in the Joint Process model. **C:** Estimates of $\mathcal{T}_{X\to Y}$ for multidimensional variables generated by linear stacking in the Joint Process model.

3.5. Novel Feature: Conditional Transfer Entropy Estimation

For the transfer entropy estimation approaches discussed so far, we assumed that the causal relationship between X and Y is fully described considering only those variables. That is, we assumed *causal sufficiency* (Peters et al., 2017). But what if we have a larger system, and access to measurements from a third variable, Z? How can ignoring the effect of Z lead to erroneous or incomplete conclusions about causal relationships between X and Y?

Consider a case in which Z affects both X and Y [denoted $X \leftarrow Z \rightarrow Y$]. If the information from Z reaches X before it reaches Y, then knowing $\mathbf{x_t}$ reduces uncertainty in y_{t+1}, so we will have $\mathcal{T}_{X\rightarrow Y} > 0$, despite the fact that there is no causal interaction between X and Y. Similarly, we can consider a case where the causal diagram is $X \rightarrow Z \rightarrow Y$, which is referred to as a *chain*. Here, we will again have $\mathcal{T}_{X\rightarrow Y} > 0$, despite the fact that X does not directly cause Y.

To avoid such issues caused by confounding variables, *conditional transfer entropy* [CTE], denoted $\mathcal{T}_{X\rightarrow Y|Z}$ was proposed (James et al., 2016). As with basic TE, one of the ways to formulate conditional transfer entropy is using a difference in conditional entropies:

$$\mathcal{T}_{X\rightarrow Y|Z} := \mathcal{H}(y_{t+1}|\mathbf{y_t}, \mathbf{z_t}) - \mathcal{H}(y_{t+1}|\mathbf{x_t}, \mathbf{y_t}, \mathbf{z_t}) \tag{7}$$

In other words, CTE is measuring the additional information on Y available in the past of X, that is not already available in the past of Y or the conditioning variable Z.

While kNN CTE estimators have been developed [see Shahsavari Baboukani et al. (2020)], incorporating additional variables increases the dimensionality of the problem, exacerbating known limitations of kNN-based methods. To our knowledge, AGM-TE is the first ML method to tackle CTE estimation. This is done using a cross entropy approximation of Eq. 7:

$$\hat{\mathcal{T}}_{X\rightarrow Y|Z} := \frac{1}{T}\sum\nolimits_{t=1}^{T}[-\log q_{\Theta_1}(y_{t+1}|\mathbf{y_t}, \mathbf{z_t})] - \frac{1}{T}\sum\nolimits_{t=1}^{T}[-\log q_{\Theta_2}(y_{t+1}|\mathbf{y_t}, \mathbf{x_t}, \mathbf{z_t})] \tag{8}$$

If we denote by $\mathcal{Z}$ the concatenation of all system variables that are not X or Y, $T_{X\rightarrow Y|\mathcal{Z}}$ isolates the causal relationship between our two variables of interest by accounting for potential confounding effects from all other measured factors. A good estimator of $T_{X\rightarrow Y|\mathcal{Z}}$ therefore reduces false causal positives in complex systems with many potentially interacting elements (Novelli and Lizier, 2021). This makes CTE useful for causal feature selection (Bonetti et al., 2024).

3.6. Summary and Outlook

By incorporating methods from the probabilistic modelling of dynamical systems, we believe our approach offers four fundamental advantages that differentiate it from existing TE estimators:

- The ability to model long-range temporal dependencies
- The flexibility to specify an appropriately calibrated observation model for specific data types
- Favourable scaling with high dimensional variables
- The ability to estimate conditional transfer entropy

In the following section, we apply our CTE estimator to multi-hundred-dimensional neural recordings to infer effective connectivity between six interacting brain regions in the mammalian visual cortex. This empirical demonstration is intended to highlight all of these advantages.

4. Scientific Application: Inferring Effective Brain Connectivity

4.1. Introduction: Data and Problem Statement

The mammalian brain is an exceedingly complex information processing system. Much of the information being processed by this system is represented in the frequency, timing, and spatial distribution of *spikes* [millisecond fluctuations in the membrane potential of neural cells], making them fundamental to how the brain computes (Bear, 2020).

Recent advancements in electrophysiological recording technology have led to the development of high-density electrode arrays called *Neuropixels* (Jun et al., 2017). These devices enable researchers to record spike trains (Fig. 2) from hundreds of neurons simultaneously.

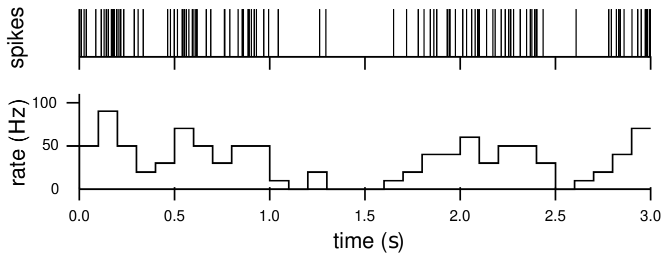

Figure 2: Schematic of a spike train for a single neuron, and the corresponding firing rate estimate generated by counting spikes in every 100 ms interval. Source: Dayan and Abbott (2004)

The spatio-temporal patterns of spiking are fundamentally constrained by the anatomical network connectivity of the brain (Knox et al., 2019). Would it therefore be possible to tackle the inverse problem, and infer the mesoscale network connectivity of the brain directly from the patterns of information flow in the neural activity recorded by Neuropixels? Doing so would suggest that the method used for inferring information flow generates results that match known constraints.

EFFECTIVE CONNECTIVITY INFERENCE AS AN APPLICATION OF CAUSAL DISCOVERY

Effective connectivity [EC] refers to the extent to which activity in one brain region causes activity in another area (Friston, 1994). As the integration within and between functionally specialized brain regions is mediated by patterns of effective connectivity, characterisation of EC is an important objective in many areas of neuroscience (Zeidman et al., 2019). The vast quantities of high fidelity observational spike data, combined with the relative lack of interventional experiments (Steinmetz et al., 2021) makes neuroscience a prime candidate for the application of causal discovery methods.

Due to its ability to detect the complex, non-linear relationships common in the brain, TE has emerged as a promising candidate for quantifying the causal interactions that underlie effective connectivity (Vicente et al., 2010; Chicharro and Panzeri, 2014). However, TE estimation methods for spike data are unable to analyse modern Neuropixel datasets, due to their reliance on kNN methods (Lizier et al., 2010; Shorten et al., 2021), which suffer from the curse of dimensionality.

In this section, we show that AGM-TE infers conditional transfer entropy values from multi-hundred dimensional spike datasets recorded in live animals that imply effective connectivity patterns consistent with known neuroanatomical results. This analysis demonstrates the scalability of our method to large datasets, and highlights the potential for conditional transfer entropy as a way to robustly discover causal interactions in systems with many interacting components.

4.2. Methods

Rationale and Goals

Effective connectivity is inherently constrained by structural connectivity (Friston, 2011), the availability of physical information-carrying connections between neural populations (Škoch et al., 2022). If TE is indeed a good measure of EC, we should see a high degree of correspondence between TE estimates derived from recordings of live brains, and the structural connectivity inferred by the dissection of dead specimens. In our experiments, we sought to infer the effective connectivity of six regions of the mouse visual cortex using CTE, and compare this to an anatomical ground truth.

Empirical Data Sources

Our reference structural connectivity matrix [to which we will compare our inferred EC matrix] is taken from the retrograde tracer experiments of Gămănuţ et al. (2018), which used fluorescent markers to trace physical connections between a target population of neurons and their inputs.

The spike dataset was taken from (Siegle et al., 2021), who used Neuropixels to record the activity of hundreds of neurons in six regions of the visual cortex [V1, RL, LM, AL, PM, AM] (Fig. 26) in eight awake mice viewing diverse visual stimuli [See Appendix I.1 for details on dataset sizes]. These stimuli served to excite the visual system, allowing us to observe the propagation of sensory information between regions of the visual cortex.

Modelling Neural Data with AGM-TE

In our spike dataset, each observation y_t is a vector of discrete values which correspond to the number of spikes occurring in a $\Delta = 100$ ms time step for each of the d neurons in a target region. As this is a type of event data, to model these observations, we assume that they are generated by a set of independent Poisson processes [one for each neuron] (Dayan and Abbott, 2004; Schimel et al., 2021). The AGM therefore takes as inputs observations of past spike counts, and uses these to infer a d dimensional rate vector μ_{t+1}, which parametrise the Poisson distributions for each neuron from which we expect the number of spikes to be drawn at $t+1$.

For our *latent dynamics model* f_{θ_f}, we use the gated recurrent unit [GRU] of Cho et al. (2014), a type of RNN that can effectively account for long range temporal dependencies that we expect in the underlying neural processes. In our observation model g_{θ_g} maps of the predicted latent state vector $\hat{s}_{t+1}$ to the positive real-valued rate parameter vector μ_{t+1}, ensuring that such parameters remain positive. For each t, and each neuron, we can calculate the probability of observing a given amount of spike events, given the predicted rate of spiking. We minimise the empirical expected value of the corresponding NLL to train the AGMs [See Appendix D.3.2 for details]. If the optimisation is successful, μ_{t+1} should correspond to our estimate of the *latent* firing rate of the neuron, which may be > 0, even when no actual spikes are *observed* during the interval.

Effective Connectivity Analysis

In a neural system with n regions, we estimate $n^2 - n$ distinct EC values to fill all off-diagonal elements of an $n \times n$ effective connectivity matrix. We defined our measure of effective connectivity between regions X and Y as a normalised conditional transfer entropy:

$$\hat{\text{EC}}_{X \to Y} = \frac{1}{d} \mathcal{T}_{X \to Y | \mathcal{Z}}$$

where $\mathcal{Z}$ is a concatenation of all remaining regions that are not X or Y, and d is the dimensionality of Y. Our effective connectivity measure therefore quantifies the average per-neuron rate of information flow. To estimate this rate, we consider the gain in predictive capacity of the AGM that *also* considers the past of the X region when determining the future spike counts of the Y region, compared to a base model relying on only Y and the remaining $\mathcal{Z}$ regions.

4.3. Results and Discussion

Validation on Simulated Data

To assess the efficacy of our approach in inferring effective connectivity, we first validated our proposed method on synthetic data generated by a spiking readout of a rate-based neural model of three interacting brain regions. We found that the CTE estimates of AGM-TE correspond to the known ground truth network structure. We also found that when compared to using classic TE, spuriously inferred connectivity is significantly reduced by conditioning [See Appendix I.3 for details].

Results in the Empirical Dataset

In the empirical dataset, the effective connectivity matrix inferred using AGM-TE (Fig. 3) matches the known pattern of connectivity dominated by the connections from V1 to its primary targets: RL, LM, AL and PM (Froudarakis et al., 2019; Siegle et al., 2021).

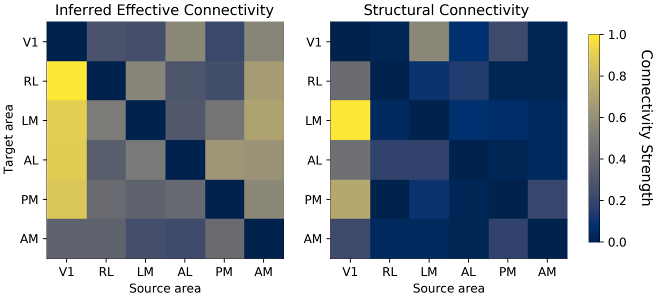

Figure 3: Comparison of the effective connectivity matrix inferred by AGM-TE with a structural connectivity matrix derived from cortical tracer results

To provide a quantification of this qualitative correspondence, we used a bootstrapping procedure [Appendix I.4] to estimate the significance of the correlation coefficient calculated between the inferred effective connectivity matrix and the anterograde tracer results of Gămănuţ et al. (2018). The procedure suggested that the probability of achieving the degree of structural correspondence that we observed in our results by chance alone is $< 0.3\%$, showing that our inferred patterns of effective connectivity are highly consistent with the known structural connectivity of these areas.

To our knowledge, these experiments represent not only the first time TE estimation methods have been applied to Neuropixel data [which by itself represents a significant step in the application of TE to neuroscience], but also the first time empirical datasets with hundreds of dimensions have been successfully processed by any TE estimation method. These results also demonstrate how conditional TE can effectively isolate the interactions of a given variable pair in systems with multiple interacting elements, paving the way for the application of CTE to the discovery of causal graphs.

5. Conclusions

Data-driven approaches for the discovery of causal interactions have found applications in an increasingly wide range of scientific and engineering domains (Vicente et al., 2010; Wibral et al., 2014; Runge, 2018; Massaro et al., 2023). However, despite its theoretical benefits, issues with the "practical computability" of transfer entropy have thus far impeded the widespread integration of TE into such causal discovery pipelines (Castri et al., 2023).

Summary of Results

In this paper, we took inspiration from probabilistic models of dynamical systems to formulate AGM-TE, a versatile and robust cross entropy ML approach for estimating TE in complex systems.

In our benchmarks, which represent the most comprehensive comparison of TE estimation methods to date, AGM-TE demonstrated superior performance compared to existing approaches in terms of both accuracy and false positives. We have also made our synthetic data generators used in the comparison available to the public[2].

We believe that the combination of a recurrent neural network for the dynamics model, and a user-defined parametric observation model leads to four key advantages compared to existing methods: the ability to handle temporally extended interactions, the use of appropriately calibrated observation models for specific data types, favourable scaling with high dimensional data, and the ability to estimate conditional transfer entropy. We demonstrated these benefits, and validated the conceptual promise of CTE to isolate causal interactions between variable pairs in systems with multiple confounders by inferring a pattern of brain connectivity from spike data consistent with known neuroanatomical results. This also represents the first instance in which a TE estimation method is able to scale to the large datasets being generated by recent advancements in neural recording technologies.

Future Work

An interesting potential application of transfer entropy estimation not explored in this text is in the area of *control*, which concerns manipulating system inputs to achieve desired behaviours. As the problem of control can be posed as a task of reducing the entropy of the state vector x using the control inputs u (Touchette and Lloyd, 2004; Lozano-Durán and Arranz, 2022), the transfer entropy between a given control variable $u^{[i]}$ and state variable $x^{[i]}$ is an important parameter that determines the extent to which $u^{[i]}$ can be used to control [reduce the entropy of] $x^{[i]}$ in a multi-input multiple-output [MIMO] system (Westphal et al., 2024). While classic TE has already been applied to chemical process control tasks (Bauer et al., 2007; Lee et al., 2020), we believe that the use of conditional transfer entropy would be especially beneficial in such MIMO systems.

We look forward [and hope to contribute] to the continued development and application of causal discovery methods to the many areas of science and engineering permeated by vast quantities of available observational data.

Acknowledgements

RS were partially funded by the EPSRC Open Fellowship EP/W024330/1.

[2] Data generating systems used in the benchmarks can be found at https://github.com/dkornai/TE-datasim

References

Charles K. Assaad, Emilie Devijver, and Eric Gaussier. Survey and evaluation of causal discovery methods for time series. *Journal of Artificial Intelligence Research*, 73:767–819, February 2022. ISSN 1076-9757. doi: 10.1613/jair.1.13428.

Lionel Barnett, Adam B. Barrett, and Anil K. Seth. Granger causality and transfer entropy are equivalent for gaussian variables. *Physical Review Letters*, 103(23):238701, December 2009. ISSN 1079-7114. doi: 10.1103/physrevlett.103.238701.

Adam B. Barrett, Lionel Barnett, and Anil K. Seth. Multivariate granger causality and generalized variance. *Physical Review E*, 81(4):041907, April 2010. ISSN 1550-2376. doi: 10.1103/physreve.81.041907.

Margret Bauer, John W. Cox, Michelle H. Caveness, James J. Downs, and Nina F. Thornhill. Finding the direction of disturbance propagation in a chemical process using transfer entropy. *IEEE Transactions on Control Systems Technology*, 15(1):12–21, January 2007. ISSN 1063-6536. doi: 10.1109/tcst.2006.883234.

Mark Bear. *Neuroscience*. Jones and Bartlett Learning, LLC, Burlington, 4th ed edition, 2020. ISBN 9781284618747.

Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeswar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and R Devon Hjelm. Mine: Mutual information neural estimation. 2018. doi: 10.48550/ARXIV.1801.04062.

Paolo Bonetti, Alberto Maria Metelli, and Marcello Restelli. Causal feature selection via transfer entropy. In *2024 International Joint Conference on Neural Networks (IJCNN)*, pages 1–10. IEEE, June 2024. doi: 10.1109/ijcnn60899.2024.10651028.

A Böhnisch, E Felsche, and R Ludwig. European heatwave tracks: using causal discovery to detect recurring pathways in a single-regional climate model large ensemble. *Environmental Research Letters*, 18(1):014038, January 2023. ISSN 1748-9326. doi: 10.1088/1748-9326/aca9e3.

Luca Castri, Sariah Mghames, Marc Hanheide, and Nicola Bellotto. Enhancing causal discovery from robot sensor data in dynamic scenarios. In Mihaela van der Schaar, Cheng Zhang, and Dominik Janzing, editors, *Proceedings of the Second Conference on Causal Learning and Reasoning*, volume 213 of *Proceedings of Machine Learning Research*, pages 243–258. PMLR, 11–14 Apr 2023.

Joyjit Chatterjee and Nina Dethlefs. Temporal causal inference in wind turbine scada data using deep learning for explainable ai. *Journal of Physics: Conference Series*, 1618:022022, September 2020. ISSN 1742-6596. doi: 10.1088/1742-6596/1618/2/022022.

Xiuqiong Chen, Yangtianze Tao, Wenjie Xu, and Stephen Shing-Toung Yau. Recurrent neural networks are universal approximators with stochastic inputs. *IEEE Transactions on Neural Networks and Learning Systems*, 34(10):7992–8006, October 2023. ISSN 2162-2388. doi: 10.1109/tnnls.2022.3148542.

Yonghong Chen, Steven L. Bressler, and Mingzhou Ding. Frequency decomposition of conditional granger causality and application to multivariate neural field potential data. *Journal of Neuroscience Methods*, 150(2):228–237, January 2006. ISSN 0165-0270. doi: 10.1016/j.jneumeth.2005.06.011.

Daniel Chicharro and Stefano Panzeri. Algorithms of causal inference for the analysis of effective connectivity among brain regions. *Frontiers in Neuroinformatics*, 8, July 2014. ISSN 1662-5196. doi: 10.3389/fninf.2014.00064.

Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*. Association for Computational Linguistics, 2014. doi: 10.3115/v1/w14-4012.

Peter Dayan and Laurence F. Abbott. *Theoretical neuroscience*. Computational neuroscience. MIT Press, Cambridge, Mass. [u.a.], 2004. ISBN 9780262541855.

M. D. Donsker and S. R. S. Varadhan. Asymptotic evaluation of certain markov process expectations for large time. iv. *Communications on Pure and Applied Mathematics*, 36(2):183–212, March 1983. ISSN 1097-0312. doi: 10.1002/cpa.3160360204.

Tom Edinburgh, Stephen J. Eglen, and Ari Ercole. Causality indices for bivariate time series data: A comparative review of performance. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 31(8), August 2021. ISSN 1089-7682. doi: 10.1063/5.0053519.

Stefan Frenzel and Bernd Pompe. Partial mutual information for coupling analysis of multivariate time series. *Physical Review Letters*, 99(20):204101, November 2007. ISSN 1079-7114. doi: 10.1103/physrevlett.99.204101.

Karl J. Friston. Functional and effective connectivity in neuroimaging: A synthesis. *Human Brain Mapping*, 2(1–2):56–78, January 1994. ISSN 1097-0193. doi: 10.1002/hbm.460020107.

Karl J. Friston. Functional and effective connectivity: A review. *Brain Connectivity*, 1(1):13–36, January 2011. ISSN 2158-0022. doi: 10.1089/brain.2011.0008.

Emmanouil Froudarakis, Paul G. Fahey, Jacob Reimer, Stelios M. Smirnakis, Edward J. Tehovnik, and Andreas S. Tolias. The visual cortex in context. *Annual Review of Vision Science*, 5(1): 317–339, September 2019. ISSN 2374-4650. doi: 10.1146/annurev-vision-091517-034407.

Weihao Gao, Sewoong Oh, and Pramod Viswanath. Demystifying fixed k-nearest neighbor information estimators. *IEEE Transactions on Information Theory*, 64(8):5629–5661, August 2018. ISSN 1557-9654. doi: 10.1109/tit.2018.2807481.

Sahil Garg, Umang Gupta, Yu Chen, Syamantak Datta Gupta, Yeshaya Adler, Anderson Schneider, and Yuriy Nevmyvaka. Estimating transfer entropy under long ranged dependencies. *The 38th Conference on Uncertainty in Artificial Intelligence*, 2022.

John Geweke. Measurement of linear dependence and feedback between multiple time series. *Journal of the American Statistical Association*, 77(378):304–313, June 1982. ISSN 1537-274X. doi: 10.1080/01621459.1982.10477803.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

C. W. J. Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 37(3):424, August 1969. ISSN 0012-9682. doi: 10.2307/1912791.

Răzvan Gămănuţ, Henry Kennedy, Zoltán Toroczkai, Mária Ercsey-Ravasz, David C. Van Essen, Kenneth Knoblauch, and Andreas Burkhalter. The mouse cortical connectome, characterized by an ultra-dense cortical graph, maintains specificity by distinct connectivity profiles. *Neuron*, 97 (3):698–715.e10, February 2018. ISSN 0896-6273. doi: 10.1016/j.neuron.2017.12.037.

Ryan G. James, Nix Barnett, and James P. Crutchfield. Information flows? a critique of transfer entropies. *Physical Review Letters*, 116(23):238701, June 2016. ISSN 1079-7114. doi: 10.1103/physrevlett.116.238701.

Xiaoxuan Jia, Joshua H. Siegle, Séverine Durand, Greggory Heller, Tamina K. Ramirez, Christof Koch, and Shawn R. Olsen. Multi-regional module-based signal transmission in mouse visual cortex. *Neuron*, 110(9):1585–1598.e9, May 2022. ISSN 0896-6273. doi: 10.1016/j.neuron.2022.01.027.

James J. Jun, Nicholas A. Steinmetz, Joshua H. Siegle, Daniel J. Denman, Marius Bauza, Brian Barbarits, Albert K. Lee, Costas A. Anastassiou, Alexandru Andrei, Çağatay Aydın, Mladen Barbic, Timothy J. Blanche, Vincent Bonin, João Couto, Barundeb Dutta, Sergey L. Gratiy, Diego A. Gutnisky, Michael Häusser, Bill Karsh, Peter Ledochowitsch, Carolina Mora Lopez, Catalin Mitelut, Silke Musa, Michael Okun, Marius Pachitariu, Jan Putzeys, P. Dylan Rich, Cyrille Rossant, Weilung Sun, Karel Svoboda, Matteo Carandini, Kenneth D. Harris, Christof Koch, John O'Keefe, and Timothy D. Harris. Fully integrated silicon probes for high-density recording of neural activity. *Nature*, 551(7679):232–236, November 2017. ISSN 1476-4687. doi: 10.1038/nature24636.

Junil Kim, Simon T. Jakobsen, Kedar N Natarajan, and Kyoung-Jae Won. Tenet: gene network reconstruction using transfer entropy reveals key regulatory factors from single cell transcriptomic data. *Nucleic Acids Research*, 49(1):e1–e1, November 2020. ISSN 1362-4962. doi: 10.1093/nar/gkaa1014.

Sanggyun Kim, David Putrino, Soumya Ghosh, and Emery N. Brown. A granger causality measure for point process models of ensemble neural spiking activity. *PLoS Computational Biology*, 7(3): e1001110, March 2011. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1001110.

Joseph E. Knox, Kameron Decker Harris, Nile Graddis, Jennifer D. Whitesell, Hongkui Zeng, Julie A. Harris, Eric Shea-Brown, and Stefan Mihalas. High-resolution data-driven model of the mouse connectome. *Network Neuroscience*, 3(1):217–236, January 2019. ISSN 2472-1751. doi: 10.1162/netn_a_00066.

L. F. Kozachenko and N. N. Leonenko. Sample estimate of the entropy of a random vector. *Problemy Peredachi Informatsii*, 23:9–16, 1987.

Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. Estimating mutual information. *Physical Review E*, 69(6):066138, June 2004. ISSN 1550-2376. doi: 10.1103/physreve.69.066138.

Hodong Lee, Changsoo Kim, Sanha Lim, and Jong Min Lee. Data-driven fault diagnosis for chemical processes using transfer entropy and graphical lasso. *Computers and Chemical Engineering*, 142:107064, November 2020. ISSN 0098-1354. doi: 10.1016/j.compchemeng.2020.107064.

Yiheng Liang and Armin R. Mikler. Big data problems on discovering and analyzing causal relationships in epidemiological data. In *2014 IEEE International Conference on Big Data (Big Data)*. IEEE, October 2014. doi: 10.1109/bigdata.2014.7004421.

Fan Lin, Yao Zhang, Ke Wang, Jianxue Wang, and Morun Zhu. Parametric probabilistic forecasting of solar power with fat-tailed distributions and deep neural networks. *IEEE Transactions on Sustainable Energy*, 13(4):2133–2147, October 2022. ISSN 1949-3037. doi: 10.1109/tste.2022.3186517.

Michael Lindner, Raul Vicente, Viola Priesemann, and Michael Wibral. Trentool: A matlab open source toolbox to analyse information flow in time series data with transfer entropy. *BMC Neuroscience*, 12(1), November 2011. ISSN 1471-2202. doi: 10.1186/1471-2202-12-119.

Joseph T. Lizier, Jakob Heinzle, Annette Horstmann, John-Dylan Haynes, and Mikhail Prokopenko. Multivariate information-theoretic measures reveal directed information structure and task relevant changes in fmri connectivity. *Journal of Computational Neuroscience*, 30(1):85–107, August 2010. ISSN 1573-6873. doi: 10.1007/s10827-010-0271-2.

Damiano Lombardi and Sanjay Pant. Nonparametrick-nearest-neighbor entropy estimator. *Physical Review E*, 93(1):013310, January 2016. ISSN 2470-0053. doi: 10.1103/physreve.93.013310.

Adrián Lozano-Durán and Gonzalo Arranz. Information-theoretic formulation of dynamical systems: Causality, modeling, and control. *Physical Review Research*, 4(2):023195, June 2022. ISSN 2643-1564. doi: 10.1103/physrevresearch.4.023195.

Omer Luxembourg, Dor Tsur, and Haim Permuter. Treet: Transfer entropy estimation via transformer, 2024.

Daniele Massaro, Saleh Rezaeiravesh, and Philipp Schlatter. On the potential of transfer entropy in turbulent dynamical systems. *Scientific Reports*, 13(1), December 2023. ISSN 2045-2322. doi: 10.1038/s41598-023-49747-1.

Nicholas J. Matiasz, Justin Wood, Wei Wang, Alcino J. Silva, and William Hsu. Computer-aided experiment planning toward causal discovery in neuroscience. *Frontiers in Neuroinformatics*, 11, February 2017. ISSN 1662-5196. doi: 10.3389/fninf.2017.00012.

David McAllester and Karl Stratos. Formal limitations on the measurement of mutual information, 2020.

Arnab Kumar Mondal, Arnab Bhattacharya, Sudipto Mukherjee, Prathosh AP, Sreeram Kannan, and Himanshu Asnani. C-mi-gan : Estimation of conditional mutual information using minmax formulation, 2020.

Sudipto Mukherjee, Himanshu Asnani, and Sreeram Kannan. Ccmi : Classifier based conditional mutual information estimation, 2019.

Leonardo Novelli and Joseph T. Lizier. Inferring network properties from time series using transfer entropy and mutual information: Validation of multivariate versus bivariate approaches. *Network Neuroscience*, pages 1–32, March 2021. ISSN 2472-1751. doi: 10.1162/netn_a_00178.

Judea Pearl. An introduction to causal inference. *The International Journal of Biostatistics*, 6(2), January 2010. ISSN 1557-4679. doi: 10.2202/1557-4679.1203.

Jonas Peters, Dominik Janzing, and Scholkopf Bernhard. *Elements of causal inference*. The MIT Press, Cambridge, Massachusetts, 2017. ISBN 9780262037310.

Ben Poole, Sherjil Ozair, Aaron van den Oord, Alexander A. Alemi, and George Tucker. On variational bounds of mutual information, 2019.

Syama Sundar Rangapuram, Matthias W Seeger, Jan Gasthaus, Lorenzo Stella, Yuyang Wang, and Tim Januschowski. Deep state space models for time series forecasting. *Advances in Neural Information Processing Systems*, 31, 2018.

J. Runge. Causal network reconstruction from time series: From theoretical assumptions to practical estimation. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 28(7), July 2018. ISSN 1089-7682. doi: 10.1063/1.5025050.

Jakob Runge, Jobst Heitzig, Norbert Marwan, and Jürgen Kurths. Quantifying causal coupling strength: A lag-specific measure for multivariate time series related to transfer entropy. *Physical Review E*, 86(6):061121, December 2012. ISSN 1550-2376. doi: 10.1103/physreve.86.061121.

Jakob Runge, Sebastian Bathiany, Erik Bollt, Gustau Camps-Valls, Dim Coumou, Ethan Deyle, Clark Glymour, Marlene Kretschmer, Miguel D. Mahecha, Jordi Muñoz-Marí, Egbert H. van Nes, Jonas Peters, Rick Quax, Markus Reichstein, Marten Scheffer, Bernhard Schölkopf, Peter Spirtes, George Sugihara, Jie Sun, Kun Zhang, and Jakob Zscheischler. Inferring causation from time series in earth system sciences. *Nature Communications*, 10(1), June 2019. ISSN 2041-1723. doi: 10.1038/s41467-019-10105-3.

David Salinas, Valentin Flunkert, and Jan Gasthaus. Deepar: Probabilistic forecasting with autoregressive recurrent networks, 2017.

Marine Schimel, Ta-Chu Kao, Kristopher T. Jensen, and Guillaume Hennequin. ilqr-vae: control-based learning of input-driven dynamics with applications to neural data. October 2021. doi: 10.1101/2021.10.07.463540.

Thomas Schreiber. Measuring information transfer. *Physical Review Letters*, 85(2), July 2000.

Anton Maximilian Schäfer and Hans Georg Zimmermann. *Recurrent Neural Networks Are Universal Approximators*, pages 632–640. Springer Berlin Heidelberg, 2006. ISBN 9783540386278. doi: 10.1007/11840817_66.

Payam Shahsavari Baboukani, Carina Graversen, Emina Alickovic, and Jan Østergaard. Estimating conditional transfer entropy in time series using mutual information and nonlinear prediction. *Entropy*, 22(10):1124, October 2020. ISSN 1099-4300. doi: 10.3390/e22101124.

Yuval Shalev, Amichai Painsky, and Irad Ben-Gal. Neural joint entropy estimation, 2020.

David P. Shorten, Richard E. Spinney, and Joseph T. Lizier. Estimating transfer entropy in continuous time between neural spike trains or other event-based data. *PLOS Computational Biology*, 17(4):e1008054, April 2021. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1008054.

Md. Hedayetul Islam Shovon, Nanda Nandagopal, Ramasamy Vijayalakshmi, Jia Tina Du, and Bernadine Cocks. Directed connectivity analysis of functional brain networks during cognitive activity using transfer entropy. *Neural Processing Letters*, 45(3):807–824, February 2016. ISSN 1573-773X. doi: 10.1007/s11063-016-9506-1.

Joshua H. Siegle, Xiaoxuan Jia, Séverine Durand, Sam Gale, Corbett Bennett, Nile Graddis, Greggory Heller, Tamina K. Ramirez, Hannah Choi, Jennifer A. Luviano, Peter A. Groblewski, Ruweida Ahmed, Anton Arkhipov, Amy Bernard, Yazan N. Billeh, Dillan Brown, Michael A. Buice, Nicolas Cain, Shiella Caldejon, Linzy Casal, Andrew Cho, Maggie Chvilicek, Timothy C. Cox, Kael Dai, Daniel J. Denman, Saskia E. J. de Vries, Roald Dietzman, Luke Esposito, Colin Farrell, David Feng, John Galbraith, Marina Garrett, Emily C. Gelfand, Nicole Hancock, Julie A. Harris, Robert Howard, Brian Hu, Ross Hytnen, Ramakrishnan Iyer, Erika Jessett, Katelyn Johnson, India Kato, Justin Kiggins, Sophie Lambert, Jerome Lecoq, Peter Ledochowitsch, Jung Hoon Lee, Arielle Leon, Yang Li, Elizabeth Liang, Fuhui Long, Kyla Mace, Jose Melchior, Daniel Millman, Tyler Mollenkopf, Chelsea Nayan, Lydia Ng, Kiet Ngo, Thuyahn Nguyen, Philip R. Nicovich, Kat North, Gabriel Koch Ocker, Doug Ollerenshaw, Michael Oliver, Marius Pachitariu, Jed Perkins, Melissa Reding, David Reid, Miranda Robertson, Kara Ronellenfitch, Sam Seid, Cliff Slaughterbeck, Michelle Stoecklin, David Sullivan, Ben Sutton, Jackie Swapp, Carol Thompson, Kristen Turner, Wayne Wakeman, Jennifer D. Whitesell, Derric Williams, Ali Williford, Rob Young, Hongkui Zeng, Sarah Naylor, John W. Phillips, R. Clay Reid, Stefan Mihalas, Shawn R. Olsen, and Christof Koch. Survey of spiking in the mouse visual system reveals functional hierarchy. *Nature*, 592(7852):86–92, January 2021. ISSN 1476-4687. doi: 10.1038/s41586-020-03171-x.

Marco Sozzi, Ahmed Kayad, Stefano Gobbo, Alessia Cogato, Luigi Sartori, and Francesco Marinello. Economic comparison of satellite, plane and uav-acquired ndvi images for site-specific nitrogen application: Observations from italy. *Agronomy*, 11(11):2098, October 2021. ISSN 2073-4395. doi: 10.3390/agronomy11112098.

Peter Spirtes and Kun Zhang. Causal discovery and inference: concepts and recent methodological advances. *Applied Informatics*, 3(1), February 2016. ISSN 2196-0089. doi: 10.1186/s40535-016-0018-x.

Kumar Sricharan, Dennis Wei, and Alfred O. Hero. Ensemble estimators for multivariate entropy estimation. *IEEE Transactions on Information Theory*, 59(7):4374–4388, July 2013. ISSN 1557-9654. doi: 10.1109/tit.2013.2251456.

Nicholas A. Steinmetz, Cagatay Aydin, Anna Lebedeva, Michael Okun, Marius Pachitariu, Marius Bauza, Maxime Beau, Jai Bhagat, Claudia Böhm, Martijn Broux, Susu Chen, Jennifer Colonell, Richard J. Gardner, Bill Karsh, Fabian Kloosterman, Dimitar Kostadinov, Carolina Mora-Lopez, John O'Callaghan, Junchol Park, Jan Putzeys, Britton Sauerbrei, Rik J. J. van Daal, Abraham Z. Vollan, Shiwei Wang, Marleen Welkenhuysen, Zhiwen Ye, Joshua T. Dudman, Barundeb Dutta, Adam W. Hantman, Kenneth D. Harris, Albert K. Lee, Edvard I. Moser, John O'Keefe, Alfonso

Renart, Karel Svoboda, Michael Häusser, Sebastian Haesler, Matteo Carandini, and Timothy D. Harris. Neuropixels 2.0: A miniaturized high-density probe for stable, long-term brain recordings. *Science*, 372(6539), April 2021. ISSN 1095-9203. doi: 10.1126/science.abf4588.

Klaas Enno Stephan, Nikolaus Weiskopf, Peter M. Drysdale, Peter A. Robinson, and Karl J. Friston. Comparing hemodynamic models with dcm. *NeuroImage*, 38(3):387–401, November 2007. ISSN 1053-8119. doi: 10.1016/j.neuroimage.2007.07.040.

Patrick A. Stokes and Patrick L. Purdon. A study of problems encountered in granger causality analysis from a neuroscience perspective. *Proceedings of the National Academy of Sciences*, 114 (34), August 2017. ISSN 1091-6490. doi: 10.1073/pnas.1704663114.

Xiao Sun, Bahador Bahmani, Nikolaos N. Vlassis, WaiChing Sun, and Yanxun Xu. Data-driven discovery of interpretable causal relations for deep learning material laws with uncertainty propagation. *Granular Matter*, 24(1), November 2021. ISSN 1434-7636. doi: 10.1007/s10035-021-01137-y.

Hugo Touchette and Seth Lloyd. Information-theoretic approach to the study of control systems. *Physica A: Statistical Mechanics and its Applications*, 331(1–2):140–172, January 2004. ISSN 0378-4371. doi: 10.1016/j.physa.2003.09.007.

Anne E. Urai, Brent Doiron, Andrew M. Leifer, and Anne K. Churchland. Large-scale neural recordings call for new insights to link brain and behavior. *Nature Neuroscience*, 25(1):11–19, January 2022. ISSN 1546-1726. doi: 10.1038/s41593-021-00980-9.

Greg Ver Steeg and Aram Galstyan. Information transfer in social media. In *Proceedings of the 21st international conference on World Wide Web*, WWW 2012. ACM, April 2012. doi: 10.1145/2187836.2187906.

Raul Vicente, Michael Wibral, Michael Lindner, and Gordon Pipa. Transfer entropy—a model-free measure of effective connectivity for the neurosciences. *Journal of Computational Neuroscience*, 30(1):45–67, August 2010. ISSN 1573-6873. doi: 10.1007/s10827-010-0262-3.

Charles Westphal, Stephen Hailes, and Mirco Musolesi. Information-theoretic state variable selection for reinforcement learning, 2024.

Michael Wibral, Raul Vicente, and Michael Lindner. *Transfer Entropy in Neuroscience*, pages 3–36. Springer Berlin Heidelberg, 2014. ISBN 9783642544743. doi: 10.1007/978-3-642-54474-3_1.

Norbert Wiener. *The theory of Prediction*. Modern Mathematics for the Engineer. McGraw-Hill, 1956.

Patricia Wollstadt, Joseph Lizier, Raul Vicente, Conor Finn, Mario Martinez-Zarzuela, Pedro Mediano, Leonardo Novelli, and Michael Wibral. Idtxl: The information dynamics toolkit xl: a python package for the efficient analysis of multivariate information dynamics in networks. *Journal of Open Source Software*, 4(34):1081, February 2019. ISSN 2475-9066. doi: 10.21105/joss.01081.

Peter Zeidman, Amirhossein Jafarian, Nadège Corbin, Mohamed L. Seghier, Adeel Razi, Cathy J. Price, and Karl J. Friston. A guide to group effective connectivity analysis, part 1: First level analysis with dcm for fmri. *NeuroImage*, 200:174–190, October 2019. ISSN 1053-8119. doi: 10.1016/j.neuroimage.2019.06.031.

Jiji Zhang and Peter Spirtes. The three faces of faithfulness. *Synthese*, 193(4):1011–1027, February 2015. ISSN 1573-0964. doi: 10.1007/s11229-015-0673-9.

Jingjing Zhang, Osvaldo Simeone, Zoran Cvetkovic, Eugenio Abela, and Mark Richardson. Itene: Intrinsic transfer entropy neural estimator, 2019.

Puning Zhao and Lifeng Lai. Analysis of knn information estimators for smooth distributions. *IEEE Transactions on Information Theory*, 66(6):3798–3826, June 2020. ISSN 1557-9654. doi: 10.1109/tit.2019.2945041.

Antonín Škoch, Barbora Rehák Bučková, Jan Mareš, Jaroslav Tintěra, Pavel Sanda, Lucia Jajcay, Jiří Horáček, Filip Španiel, and Jaroslav Hlinka. Human brain structural connectivity matrices–ready for modelling. *Scientific Data*, 9(1), August 2022. ISSN 2052-4463. doi: 10.1038/s41597-022-01596-9.

Appendix A. Granger Causality

Informally, we say that "X *Granger causes* Y if the future value of Y can be better predicted by using both the values of X and Y up to time t than by using only the past of Y itself".

DETAILED DESCRIPTION

To measure this effect, we consider two models, $\hat{y}_{t+1} = g_1(\mathbf{y_t})$ with error $\hat{y}_{t+1} - y_{t+1} = \epsilon_1$, and $\hat{y}_{t+1} = g_2(\mathbf{x_t}, \mathbf{y_t})$ with error ϵ_2. We can then formally express the above statement as:

$$y_{t+1}|\mathbf{y_t} \neq y_{t+1}|\mathbf{x_t}, \mathbf{y_t} \implies \sum_{i=0}^{T}(\epsilon_1)^2 \gg \sum_{i=0}^{T}(\epsilon_2)^2 \implies X \text{ Granger causes } Y$$

This means that X is inferred to Granger cause Y when the residual sum of squares (RSS) of the model for $y_{t+1}|\mathbf{x_t}, \mathbf{y_t}$ is significantly lower than in the model of $y_{t+1}|\mathbf{y_t}$.

The classic GC test was developed in a linear vector autoregressive (VAR) setting. Given the timeseries X and Y, to see if $X \not\rightarrow Y$ or $X \rightarrow Y$ is preferred, we formulate two alternative regression models for Y, one including only the past of Y, and one using both X and Y:

$$y_{t+1} = \hat{y}_{t+1} + \epsilon_1 = \sum_{i=0}^{k} a_i y_{t-i} + \epsilon_1$$
$$y_{t+1} = \hat{y}_{t+1} + \epsilon_2 = \sum_{i=0}^{k} a_i y_{t-i} + \sum_{i=0}^{k} b_i x_{t-i} + \epsilon_2$$

The RSS of the two models ($\sum(\epsilon_1)^2$, $\sum(\epsilon_2)^2$) is used to calculate an F-statistic. We reject the null hypothesis $X \not\rightarrow Y$ if the observed F exceeds the $(1-\alpha)\%$ quantile of an F-distribution with appropriate degrees of freedom.

DETERMINING THE STRENGTH OF THE CAUSAL INTERACTION

However, this approach only yields a *binary* decision on whether $X \not\rightarrow Y$ or $X \rightarrow Y$ is preferred. How can we infer the strength of the causal relationship? Geweke (1982) proposed that the magnitude of the linear causal interaction can be estimated using the logarithm of a ratio computed from the variance of the ϵ_1 and ϵ_2 error terms:

$$\text{GC}_{X \rightarrow Y} = \log \frac{\text{Var}(\epsilon_1)}{\text{Var}(\epsilon_2)}$$

While this remains a widespread method, there are several fundamental disadvantages to VAR-based Granger causality strength (GCS). First and most fundamentally, the use of a VAR model for the time series restricts GC to detecting *causality of the mean* (Runge, 2018). Second, VAR models restrict GC to continuous valued Gaussian time series, with only linear causal interactions. However, even when evaluating on synthetic data generated from a VAR model, Chen et al. (2006) find that negative GCS estimates are common. When these assumptions are violated, such as in neurological data, Stokes and Purdon (2017) find that GCS estimates can be either severely biased or of high variance, both leading to spurious results.

Appendix B. Some Details of Existing TE Estimation Methods

TE can be defined as conditional mutual information. As most TE estimators adapt an existing CMI estimator, key advancements in TE inference can be traced to methodological developments in CMI inference. For this reason, we use the phrases CMI and TE interchangeably throughout this section.

B.1. kNN-KSG Methods

Modern kNN-based CMI estimators are all descended from the entropy estimator of Kozachenko and Leonenko (1987). We provide a basic introduction based on Lombardi and Pant (2016).

B.1.1. The Kozachenko and Leonenko Estimator for Entropy

For a random variable X with probability density $p(x)$, a Monte Carlo estimate of entropy is

$$\hat{\mathcal{H}}(X) := -\frac{1}{N}\sum_{i=1}^{N}\log\left(p(x_i)\right) \tag{9}$$

where N is the finite number of samples. Since $p(x_i)$ is unknown, our goal is to estimate $\hat{p(x_i)}$, which can then be substituted into the equation above to yield $\hat{\mathcal{H}}(X)$.

In kNN methods, the fundamental idea is to utilise the distance between a sample point and its k-th nearest neighbour as a way to estimate the local density. If the k-th nearest neighbour of a point x_i is at a distance of ϵ, the volume V of the ϵ-ball centred at the sample point can be approximated as $c_d\epsilon^d$, where c_d is the volume of the unit sphere in d dimensions (Fig. 4). If we assume a uniform density within this volume, the probability mass P_i in the ϵ-ball around x_i can be approximated as $P_i \approx c_d\epsilon^d p(x_i)$, yielding an estimate

$$\hat{p(x_i)} = P_i/c_d\epsilon^d$$

Taking the logarithm of this yields $\log(\hat{p(x_i)}) = \log(P_i) - \log(c_d) - d\log(\epsilon)$. The expected value of $\log(P_i)$ can be calculated using the digamma function ψ as $\mathbb{E}[\log(P_i)] = \psi(k) - \psi(N)$. Approximating $\log(P_i)$ using its expected value then yields

$$\log(\hat{p(x_i)}) = \psi(k) - \psi(N) - \log(c_d) - d\log(\epsilon(i))$$

where $\epsilon(i)$ is the distance of the i-th sample to its k-th nearest neighbour. Substitution into Eq. 9, and simplifying then gives the Kozachenko and Leonenko estimator

$$\hat{\mathcal{H}}(X) := \psi(N) - \psi(k) + \log(c_d) + \frac{d}{N}\sum_{i=1}^{N}\log(\epsilon(i)) \tag{10}$$

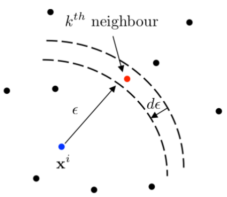

Figure 4: The k-nearest neighbour and ϵ-ball. Source: Lombardi and Pant (2016)

B.2. Variational Methods

In variational CMI estimation approaches, $\mathcal{I}(X;Y|Z)$ is estimated by maximizing a lower bound on the KL divergence between $P = p(x,y,z)$, and $Q = p(x,z)p(y|z)$. These variational estimators are descendants of the innovative approach to MI estimation proposed by Belghazi et al. (2018).

USING NEURAL NETWORKS TO ESTIMATE MI USING THE KL DIVERGENCE

The key idea behind variational methods in general is to cast the problem of approximating some function f^* as an optimisation problem by introducing a family of parametrised distributions $f \in \mathcal{F}$ [called variational distributions] and then find the member of this family that is closest to the true function f^* using optimisation methods.

In the case of Belghazi et al. (2018), f^* is the lower bound of Donsker and Varadhan (1983) on the KL divergence, which for probability distributions p and q over $\mathcal{X}$ with a finite KL divergence, and for the class of functions $f(x)$:$\mathcal{X} \to \mathbb{R} \in \mathcal{F}$ bounded in expectation is:

$$D_{KL}(p||q) \geq DV(p||q) := \sup_{f \in \mathcal{F}} \mathbb{E}_{x\sim p(x)}[f(x)] - \log(\mathbb{E}_{x\sim q(x)}[\exp(f(x))])$$

As the function $f(x)$ simply takes a sample x from $x \sim p(x)$ or $x \sim q(x)$, and outputs a scalar, Belghazi et al. (2018) define $\mathcal{F}$ to be the set of functions that can be approximated by a NN $g_\theta(x)$ with a fixed architecture. The variational problem over $\mathcal{F}$ is then further reduced to an optimization problem in terms of the network parameters θ. If the loss function is defined as the empirical (sample based) estimate of $DV_{g_\theta}(p||q)$ for a specific g_θ as:

$$\hat{DV}_{g_\theta}(p||q) = \frac{1}{n}\sum_{i=1}^{n}[g_\theta(x_i)] - \log(\frac{1}{n}\sum_{i=1}^{n}\exp(g_\theta(x_i)))$$

then optimizing θ via gradient ascent will lead to the gradual increase in the lower bound. This technique results in an estimator which convergences with increasing sample size to the true KL divergence with probability $= 1$.

VARIATIONAL FORMULATION OF THE CMI ESTIMATION PROBLEM

If we want to find $\mathcal{I}(X;Y|Z)$ via the KL divergence between $P = p(x,y,z)$, and $Q = p(x,z)p(y|z)$, we are faced with two fundamental problems:

- The first problem is rather practical. The specific implementation of the KL estimating NN g_θ (henceforth notated as R for *recognition model*), needs to be selected carefully, as Poole et al. (2019) showed that results from Belghazi et al. (2018) are very sensitive to the choice of architecture and hyperparameters.

- The second problem is unique to CMI estimation, and relates to the available data. We are only given samples from $s \sim p(x,y,z)$, but the divergence estimator would also require samples from $s \sim p(x,z)p(y|z)$. How can we learn some generative model $Q_{Y|Z} = q(y|z)$ to approximate the conditional $p(y|z)$ distribution?

Example Implementation of a Variational CMI Estimator

Estimation of Conditional Mutual Information Using MinMax Formulation [C-MI-GAN] Mondal et al. (2020) theorised that treating the problem of learning to approximate $p(y|z)$ as $Q_{Y|Z}$, and training the recognition model $R(s)$ to estimate the KL divergence separately may have been responsible for some of the convergence issues with the earlier variational CMI estimator of Mukherjee et al. (2019). They therefore devised a combined architecture (Fig. 5), and an appropriate joint training procedure that is similar to generative adversarial networks.

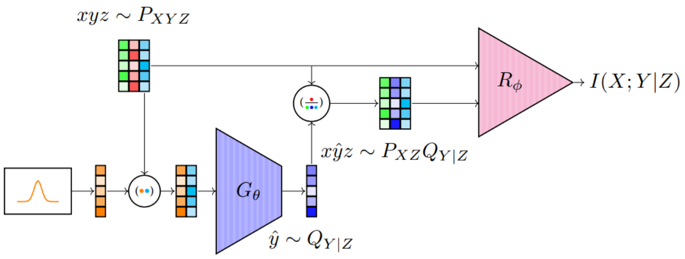

Figure 5: Block Diagram for C-MI-GAN. Samples drawn from any simplistic noise distribution are concatenated with the samples from the marginal P_Z and fed to the generator as input. The generated samples from the $Q_{Y|Z}$ distribution are then concatenated with samples from P_{XZ} and given as input to the regression network R along with samples from the original P_{XYZ} distribution. $\mathcal{I}(X;Y|Z)$ is obtained by negating the loss of the trained regression network R. Source: Mondal et al. (2020) Figure 1

We examine the loss function of Mondal et al. (2020) in more detail to help us understand how it jointly expresses the two objectives inherent to CMI estimation:

$$L(Q_{Y|Z}, R) =$$

$$\inf_{Q_{Y|Z}} \sup_{R \in \mathcal{R}} \left[\left(\int_{s \sim P_{XYZ}} P_{XYZ} R(s) \, \mathrm{d}s \right) - \log \left(\int_{s \sim P_{XZ}Q_{Y|Z}} P_{XZ}Q_{Y|Z} \exp(R(s)) \, \mathrm{d}s \right) \right]$$

The authors show (Mondal et al. (2020) *Theorem 1*) that given samples s from either $s \sim P_{XYZ}$ or $s \sim P_{XZ}Q_{Y|Z}$, the most optimal regression network R^* approximates the function:

$$R^*(s) = \log \frac{P_{XYZ}}{P_{XZ}Q_{Y|Z}} + c$$

which is an estimate of the [negative] KL divergence $-D_{KL}(P_{XYZ}||P_{XZ}Q_{Y|Z})$, which is exactly the property we seek to estimate. In turn, given the optimal regression network R^*, Mondal et al. (2020) *Theorem 2* shows that the loss function reduces to

$$L(Q_{Y|Z}, R) = \mathcal{I}(X;Y|Z) + D_{KL}(P_{Y|Z}||Q_{Y|Z})$$

which means that the loss is optimised when the model $Q_{Y|Z}$ approximates $P_{Y|Z}$ closely. These two theorems thus show that optimizing the joint loss by alternating the training of $Q_{Y|Z}$ and $R(s)$ with respect to the loss function $L(Q_{Y|Z}, R)$ will eventually lead to an accurate estimate of CMI.

Appendix C. Why Does the Categorical Distribution Cause TE Overestimation?

C.1. Introduction

One of the definitions of $\mathcal{T}_{X \to Y}$ is via the difference between the conditional entropies $\mathcal{H}(y_{t+1}|\mathbf{y_t})$ and $\mathcal{H}(y_{t+1}|\mathbf{y_t}, \mathbf{x_t})$. If either entropy estimate is biased, this will lead to erroneous estimates of TE. In the Background section, we posited that the categorical distribution used in currently available cross entropy methods [NJEE and DETE] incentivises the models to overfit the training data, leading to an underestimation of uncertainty, and therefore entropy.

In this section, we demonstrate this argument empirically using various TE estimators to analyse synthetic data from a system with analytically tractable transfer entropy. We show how at small sample sizes existing non-parametric kNN-KSG estimators, and our new method [AGM-TE] have much lower rates of false positives than cross entropy methods. We also provide a detailed look into how the biases of current categorical cross entropy methods cause overestimation.

C.2. Comparative Analysis of Overestimation in TE Estimation Methods

We compared the extent to which four methods across the three different inference paradigms overestimate TE across various sample sizes. We test kNN-KSG [NPEET (Ver Steeg and Galstyan, 2012)], variational ML [C-MI-GAN (Mondal et al., 2020)], existing categorical cross entropy ML approaches [NJEE (Shalev et al., 2020)], and our method AGM-TE.

To generate data, we used a synthetic system, the 1D linear Gaussian generative model [details in Appendix G]. This generative model yields timeseries for two scalar variables, X and Y, with the true causal relationship being $Y \to X$. By Eq. 15 of the generative model, y_{t+1} only depends on y_t, and is therefore independent of X. This further implies $p(y_{t+1}|\mathbf{y_t}) = p(y_{t+1}|\mathbf{y_t}, \mathbf{x_t})$, and thereby $\mathcal{H}(y_{t+1}|\mathbf{y_t}) = \mathcal{H}(y_{t+1}|\mathbf{y_t}, \mathbf{x_t})$, and thus the true $\mathcal{T}_{X \to Y} = 0$. We will show that NJEE will violate all of these equalities, and therefore overestimates TE.

We generated training datasets with a size of 500, 1000, 5000, and 10000, and plotted the mean TE estimation error across five replicate runs of the different methods in Fig. 6. These results clearly demonstrate that categorical cross entropy methods [represented by NJEE] have the highest rates of overestimation at small sample sizes. Notably, at a sample size of 500, NJEE estimates $\hat{\mathcal{T}}_{X \to Y} \approx 0.3$, despite the true $\mathcal{T}_{X \to Y} = 0$. Variational methods are better [$\hat{\mathcal{T}}_{X \to Y} \approx 0.1$], while non-parametric kNN-KSG estimators, and AGM-TE are nearly error free [$\hat{\mathcal{T}}_{X \to Y} \approx 0.01$].

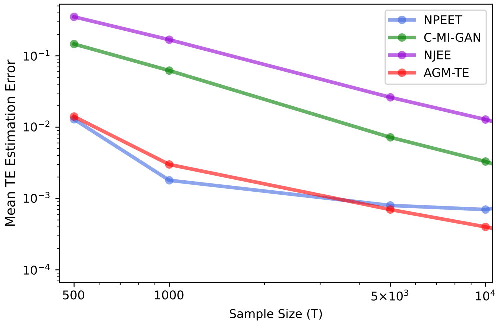

Figure 6: Mean TE estimation errors across five runs of four TE estimation methods depend heavily on sample size in the 1D LG system. The true $\mathcal{T}_{X \to Y} = 0$

C.3. Mechanisms of Overestimation in Categorical Cross Entropy Methods

Why do categorical cross entropy methods such as NJEE and DETE overestimate TE at small sample sizes? We believe the answer lies in the overfitting of the categorical distribution to small datasets. To support this argument, we trained the two NJEE models for $y_{t+1}|\mathbf{y_t}$ and $y_{t+1}|\mathbf{y_t}, \mathbf{x_t}$ using default hyperparameters of the publicly available code on a small dataset of $T = 200$ for 20000 epochs. We demonstrate how, in the model, $q(y_{t+1}|\mathbf{y_t}) \neq q(y_{t+1}|\mathbf{y_t}, \mathbf{x_t})$, despite the fact that $p(y_{t+1}|\mathbf{y_t}) = p(y_{t+1}|\mathbf{y_t}, \mathbf{x_t})$ in the true data.

Since we know how the data was generated, given the actual values of y_t, we can calculate the true distribution of y_{t+1}, as the observed value is always drawn as $y_{t+1} \sim \mathcal{N}(\mu = b_y y_t, \sigma_y^2)$. This means we can plot the true expected value and uncertainty of y_{t+1}, and compare it to the model approximation $q(y_{t+1}|\mathbf{y_t})$ by NJEE in Fig. 7. As the issue here is not immediately apparent, we obtain a more interpretable visualisation of how the true and model distributions compare. We plot the model residual PMF [obtained by sliding each predicted categorical distribution over y_{t+1} such that the maximum probability is the middle bin, and averaging], and a discrete approximation to the true residual [inherent noise in the generative model, distributed as $\mathcal{N}(0, \sigma_y^2)$] in Fig. 8. From this, we can see that the categorical distribution fit by NJEE has underestimated the uncertainty of y_{t+1}, and inferred a much sparser distribution than what the variable actually follows.

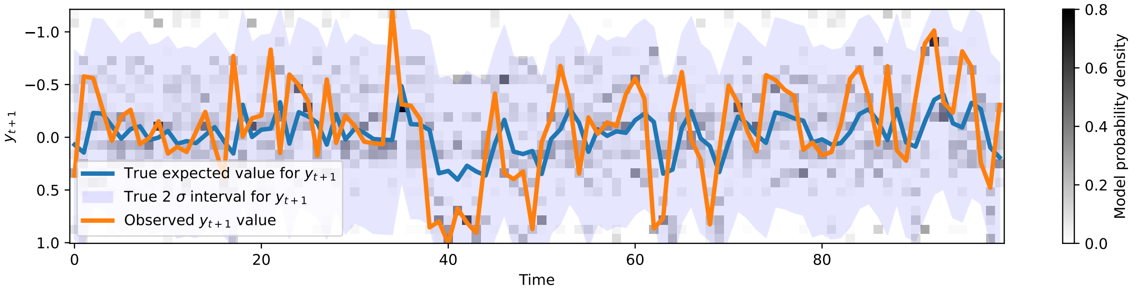

Figure 7: Comparison of the true $p(y_{t+1}|\mathbf{y_t})$ and the model probability density.

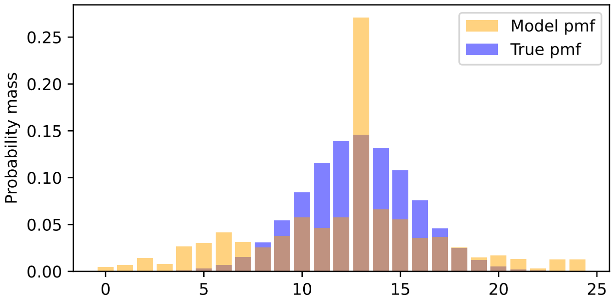

Figure 8: Comparison of true uncertainty [due to noise in the generative model] and residual model uncertainty around $p(y_{t+1}|\mathbf{y_t})$ reveals a slight underestimation of uncertainty by NJEE $[\hat{\mathcal{H}}(y_{t+1}|\mathbf{y_t}) < \mathcal{H}(y_{t+1}|\mathbf{y_t})]$.

However, the issue is much worse for the second model, which approximates $p(y_{t+1}|\mathbf{y_t}, \mathbf{x_t})$. If we plot the predictions of NJEE in the training data (Fig. 9), and compare it to the observed value for y_{t+1}, we see that the model concentrates all probability in the bin corresponding to the observed value at each timestep in the dataset, rather than learning the true data generating distribution.

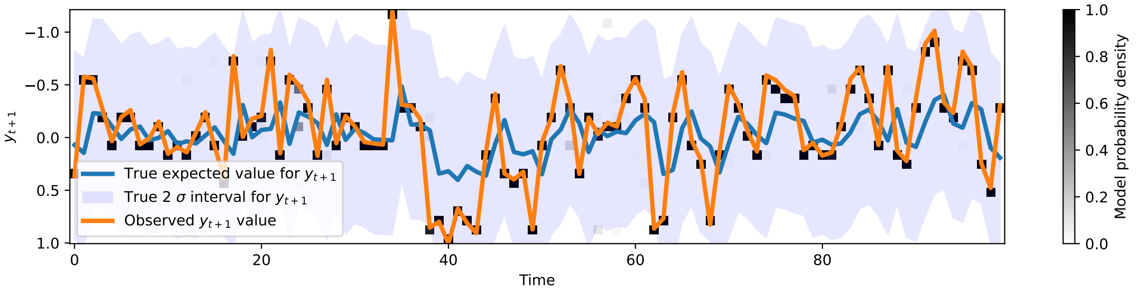

Figure 9: NJEE overfits training data when given access to information from the past of X.

Comparing the model residual PMF and a discrete approximation to the true residual (Fig. 10) provides further evidence that the second model has overfit severely, and vastly underestimated the uncertainty of y_{t+1}. This is despite the fact that X does not carry information about the future of Y, so the uncertainty should be identical between the competing models.

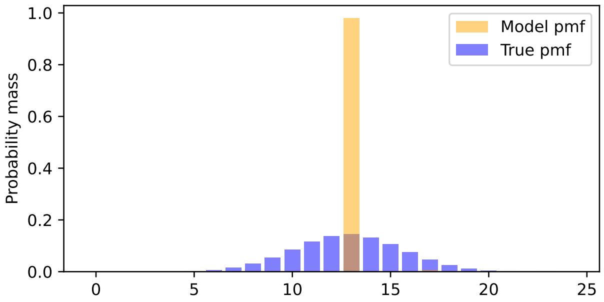

Figure 10: Comparison of true and residual uncertainty around $p(y_{t+1}|\mathbf{y_t}, \mathbf{x_t})$ shows a significant underestimation of uncertainty [$\hat{\mathcal{H}}(y_{t+1}|\mathbf{y_t}, \mathbf{x_t}) \ll \mathcal{H}(y_{t+1}|\mathbf{y_t}, \mathbf{x_t})$].

SUMMARY OF ARGUMENTS AND OUTLOOK

When using categorical cross entropy methods at small sample sizes, both models underestimate the true uncertainty [and therefore the entropy] of y_{t+1}. However, the issue is much worse for the second model that also considers the past of X. The additional data allows the model to overfit, causing $\hat{\mathcal{H}}(y_{t+1}|\mathbf{y_t}, \mathbf{x_t})$ to be significantly lower than $\hat{\mathcal{H}}(y_{t+1}|\mathbf{y_t})$, leading to the false inference of a positive transfer entropy value [$\hat{\mathcal{T}}_{X\to Y} \approx 1.1$], when the ground truth $\mathcal{T}_{X\to Y} = 0$.

When sample size is constant, later benchmarks on high dimensional systems show that NJEE also infers a false positive TE that increases in magnitude with the dimensionality of the variable [see appendix H Fig. 24 and Fig. 25]. Combined with the above results, this suggests that categorical cross entropy methods such as NJEE have a tendency to overestimate TE when the amount of data is small, relative to the complexity of the data generating system. This means that they can achieve admirable performance at the high sample sizes of synthetic benchmarks, but are expected to falter in empirical datasets where the amount of data is smaller, and systems are more complex.

Appendix D. A Detailed Description of the Approximate Generative Model

D.1. Further Architectural Details

We provide more detail for the simpler case of the AGM of $y_{t+1}|\mathbf{y_t}$. The overarching goal of the AGM predicting the d-dimensional Y from itself is to yield a predicted distribution over each observed y_{t+1} given $\mathbf{y}_t$ at each t. To accomplish this, the following computations are conducted:

1. Mapping of the observation $y_t \in \mathbb{R}^d$ to the latent state $s_t \in \mathbb{R}^m$ is accomplished using an affine transform $a()$ of the form $s_t = \mathbf{W}_{ol} y_t + \mathbf{b}_{ol}$, where $\mathbf{W}_{ol}$ is an observation-to-latent weight matrix, and $\mathbf{b}_{ol}$ is an observation-to-latent bias vector.
2. The *latent dynamics model* $f_{\theta_f}(s_t)$ yields the predicted latent state $\hat{s}_{t+1} \in \mathbb{R}^m$ from the current latent state. It can be implemented using a feedforward or recurrent NN [see Appendix D.2]. The parameters of this NN are denoted as θ_f.
3. The *observation model* maps the predicted latent $\hat{s}_{t+1}$ to a distribution over y_{t+1}. Mapping of $\hat{s}_{t+1}$ to a set of parameters $\phi_{t+1} \in \mathbb{R}^n$ is done by $g_{\theta_g}()$, which may be a simple affine transform, or involve a more complicated nonlinear function [see D.3.2 for an example]. The parameters for this transform are denoted by θ_g. The chosen *parametric distribution* q [e.g. multivariate Gaussian (see D.3.1), Poisson (see D.3.2), ...] is then parametrised by ϕ_{t+1}.

We optimise the model to make the true y_{t+1} be approximately distributed as $q(y_{t+1}; \phi_{t+1})$. The parameters of the AGM [which are optimised w.r.t the NLL] to achieve this are $\Theta = \{\mathbf{W}_{ol}, b_{ol}, \theta_f, \theta_g\}$.

In the case of the model for $y_{t+1}|\mathbf{y_t}, \mathbf{x_t}$, which also takes into account the effect of the e-dimensional variable X, the only difference is that the starting affine transformation $a()$ will take as input the concatenated $\{y_t, x_t\} \in \mathbb{R}^{d+e}$ to yield the latent $s_t \in \mathbb{R}^m$. The latent dimensionality m is shared between the models for $y_{t+1}|\mathbf{y_t}$ and $y_{t+1}|\mathbf{y_t}, \mathbf{x_t}$. This way, the parameter counts will be nearly identical. However, model parameter values are not shared, as they are optimised separately.

D.2. The Two Types of Latent Dynamics Model

The NN implementing $s_t \rightarrow \hat{s}_{t+1}$ can be either feedforward or recurrent. This architectural choice has both theoretical consequences in terms of the Markov order of the model, and practical consequences in terms of compute speed. Recall that $\mathbf{y_t}$ is a shorthand for $y_{[t-k+1:t]}$, the past k values of Y up to and including time t. If $k = 1$, we say that our model $q(y_{t+1}|\mathbf{y_t})$ has Markov order one.

D.2.1. Using a feedforward NN for the Dynamics Model

When using a feedforward neural network [which is just a composition of affine and nonlinear transformations] for $s_t \rightarrow \hat{s}_{t+1}$, we have the computational graph of Fig. 11, which clearly illustrates how our resulting AGM will have Markov order one. This means that any direct influence of more distant past states (e.g. y_{t-1}, y_{t-2}) is ignored when predicting the distribution over y_{t+1}.

While this may seem like a severe limitation [and indeed Runge et al. (2012) and Castri et al. (2023) argue that it is], all previous cross entropy ML TE methods rely on this assumption. Additionally, since each distribution over y_{t+1} is calculated from a single input, all calculations across $t \in 1 : T$ can be implemented in parallel, greatly speeding up training and inference.

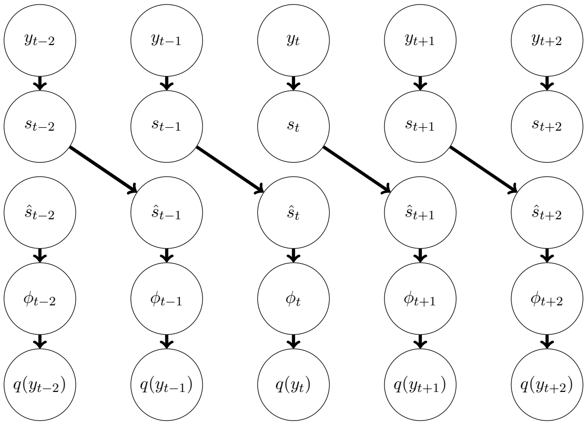

Figure 11: Computational graph for an AGM of $y_{t+1}|\mathbf{y_t}$ with a feedforward NN dynamics model

D.2.2. USING A RECURRENT NEURAL NETWORK FOR THE DYNAMICS MODEL

To address the critique of Runge et al. (2012) and Castri et al. (2023) relating to errors in TE estimation due to the truncation induced by a small k, we also implement a recurrent neural network [RNN] for calculating $s_t \rightarrow \hat{s}_{t+1}$. The key idea is that RNNs introduce a hidden state $h_t \in \mathbb{R}^m$, which is updated according to $h_{t+1} = \tanh(s_t W_{ih} + b_{ih} + h_t W_{hh} + b_{hh})$, where h is the hidden state, s_t is the affine transformed value of Y at time t, W_{ih} is the input-to-hidden weight matrix, b_{ih} is the input-to-hidden bias vector, h_{t-1} is the previous hidden state, W_{hh} is the hidden-to-hidden weight matrix, and b_{hh} is the hidden-to-hidden bias. h_0 is set to a vector of zeros. We then simply set $\hat{s}_{t+1} = h_{t+1}$. Fig. 12 outlines how longer term influences from y_t, y_{t-1}, y_{t-2}, ..., $y_{t=2}$, $y_{t=1}$ are transmitted through h_t to our estimated distribution over y_{t+1} as a consequence of this architecture.

However, training and inference are significantly slower, due to the operations being sequential, rather than parallel. We nonetheless believe that the RNN model should be chosen if the causal interactions occur with a larger or unknown delay [which is most empirical systems].

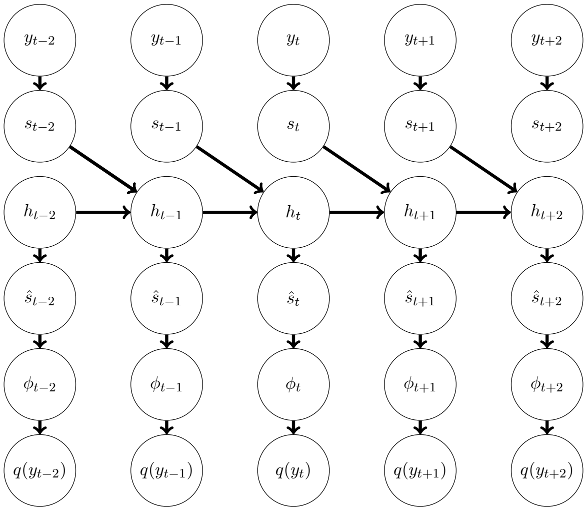

Figure 12: Computational graph for an AGM of $y_{t+1}|\mathbf{y_t}$ with an RNN dynamics model.

D.3. Choices for the Observation Model

In our opinion, one of the fundamental advantages of AGM-TE over existing cross entropy methods is the ability to use datatype-specific probability distributions for y_{t+1}.

D.3.1. Multivariate Diagonal Gaussian Observation Model

The multivariate diagonal Gaussian observation model is selected by choosing the `'gaussian'` value for the `obs_model_type` parameter when initialising the model.

Under this observation model, each observation along each dimension of Y is Gaussian distributed, with no covariance. This makes it appropriate for data with continuous variables where we assume symmetric, non-sparse observation noise, such as a temperature sensor.

This distribution is parametrised by $\phi_{t+1} \in \mathbb{R}^{2\times d}$, the concatenation of the mean vector $\mu_{t+1} \in \mathbb{R}^d$ and the variance vector $\sigma_{t+1} \in \mathbb{R}^d$ [$\phi_t = \{\mu_t, \sigma_t\}$]. The mapping g_{θ_g} from the predicted latent state output of the RNN $\hat{s}_{t+1} \in \mathbb{R}^m$ to this concatenated parameter vector ϕ_{t+1} is simply an affine transformation, with a corresponding learnable readout matrix $\mathbf{R}$ and bias vector $\mathbf{b}_{lo}$.

$$\phi_{t+1} := \{\mu_{t+1}, \sigma_{t+1}\} = g_\theta(\hat{s}_{t+1}) := \mathbf{R}\hat{s}_{t+1} + \mathbf{b}_{lo}$$

Under this chosen observation model, the negative log likelihood of observing a specific y_t is

$$-\log \mathcal{L}(y_t; \phi_t) = \frac{1}{2}\left[d\log(2\pi) + \sum_{j=1}^{d} \log(\sigma_t^{[j]})^2 + \sum_{j=1}^{d} \frac{(y_t^{[j]} - \mu_t^{[j]})^2}{(\sigma_t^{[j]})^2}\right] \tag{11}$$

where $^{[j]}$ denotes the j-th element in a given vector. This loss function is minimised when the means that are maximally similar to future observations, and variances are equal to the expected squared prediction error. This ensures that model uncertainty is well calibrated.

D.3.2. Collection of Poissons Observation Model

The collection of Poissons observation model is selected by choosing the `'poisson'` value for the `obs_model_type` parameter when initialising the model.

Under this observation model, each observation along each dimension of Y is Poisson distributed. This makes it an appropriate choice for *event-based* data, such as spike trains in neuroscience, or timings of certain occurrences in an industrial control system.

This collection of d independent Poisson distributions is defined by the rate parameter vector μ_t [$\phi_t = \{\mu_t\}$]. As rates are strictly non-negative, g_{θ_g}, which maps of the predicted latent state vector $\hat{s}_{t+1}$ to the positive real-valued rate parameter vector μ_{t+1} is defined as

$$\phi_{t+1} := \mu_{t+1} = g_\theta(\hat{s}_{t+1}) := \beta \exp(\mathbf{R}\hat{s}_{t+1} + \mathbf{b}_{lo})\Delta$$

Δ is the sampling rate of data. The latent-to-neuron readout matrix $\mathbf{R}$, the rate bias vector $\mathbf{b}_{lo}$, and the rate multiplication vector β constitute the learnable parameters of the observation model.

the negative log likelihood of observing the event count vector y_t is

$$-\log \mathcal{L}(y_t; \phi_t) = -\sum_{j=1}^{d} (y_t^{[j]} \log \mu_t^{[j]} - \mu_t^{[j]} + \log y_t^{[j]}!) \tag{12}$$

This incentivises μ_{t+1} to be an estimate of the *latent* rate of events, which may be > 0, even when no actual events are *observed* during the interval.

Appendix E. What is the Learned Approximate Generative Model?

In this section, we show an example of the probabilistic forecasting approach implemented by AGM-TE as it models an input driven dynamical system, and introduce how model performance differences capture the essential idea of predictive causality.

E.1. The Synthetic Brain Activity Data Generating Model

Our chosen generative model is a neural state equation, which models how activity of brain regions changes in response to external inputs (Stephan et al., 2007). The stochastic external inputs come from a simple Markov model with two states, 0 or 1. The transitions between states follow a Markov process, while durations spent in each state (1 or 0) follow Exponential distributions, with mean up and mean down times determined by rate parameters. We generate data for M independent input channels, which yields the dataset $\mathbf{X}$ containing the input vector x_t for each time step.

These inputs then drive a model of neural dynamics in a system of N interacting brain regions. The neural activity of each region is represented as a vector valued state variable y_t. The change of y_t over time is determined by a discrete time linear differential equation:

$$y_{t+1} = y_t + [\mathbf{A}y_{t-1} + \mathbf{C}x_{t-1} + \mathcal{N}(0, \sigma_r)] \tag{13}$$

where $\mathbf{A}$ is an $N \times N$ connectivity matrix between regions, which determines how the activity of one region excites or inhibits other regions, and $\mathbf{C}$ is an $N \times M$ matrix which determines how each brain region will react to each input channel. At each timestep t, stochastic noise from a normal distribution with standard deviation σ_r is also added to the current vector of neural activity, providing additional randomness. The causal relationship is denoted $X \to Y$. Fig. 13 shows an example of the inputs and the neural activity that is induced as a response.

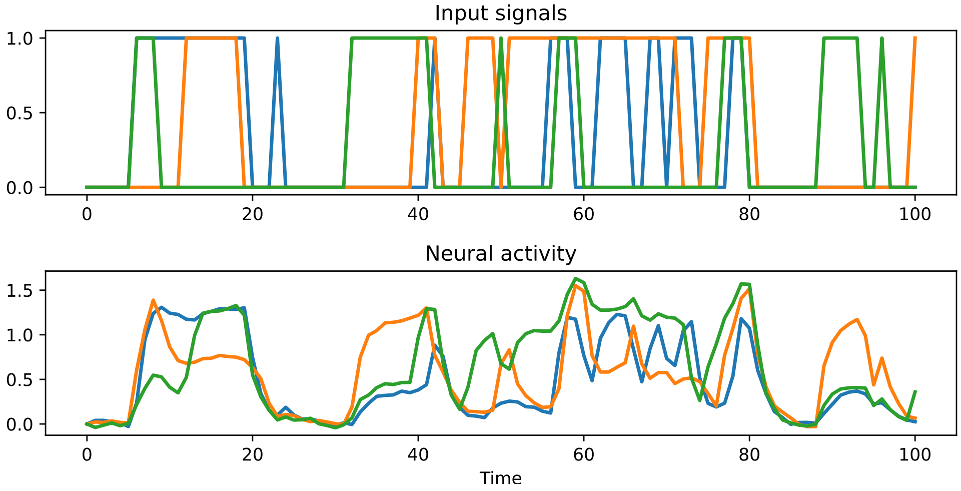

Figure 13: Example trajectories from the neural activity simulator. The 3-dimensional neural activity Y(bottom) is determined by 3-dimensional stochastic noise inputs X, and the connectivity of the neurons within the network.

E.2. Problem Setup and Visualisation of the Trained Models

To decide if $X \to Y$ or $X \not\to Y$ in the framework of predictive causality, we require two alternative models, $q_{\Theta_1}(y_{t+1}|\mathbf{y_t})$ and $q_{\Theta_2}(y_{t+1}|\mathbf{x_t}, \mathbf{y_t})$. Our goal for both models is to generate distributions that approximate y_{t+1} [in this case, the vector of neural activity]. Since q_{Θ_1} is given only the past of Y, while q_{Θ_2} is given the past of both Y and X, a better forecast of y_{t+1} by q_{Θ_2} implies $X \to Y$.

We choose to model observations of neural activity using a Gaussian [see Appendix D.3.1]. For our latent dynamics model $f_\theta()$, we use an RNN with a single hidden layer and 8 neurons. Example visualisations of the two alternative models generating predicted distributions in unseen test data are shown Figures 14 and 15. At first glance, both models provide a reasonable probabilistic prediction. However, a more detailed examination of errors reveals a significant gap in performance.

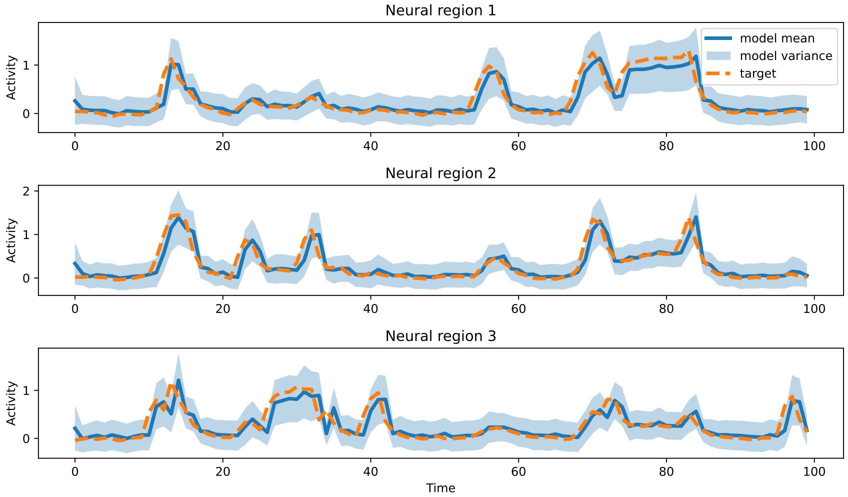

Figure 14: AGM-TE approximates $y_{t+1}|\mathbf{y_t}$ in the test data in a system with a 3-dimensional Y. Predictions are blue (with the model's variance shaded) and the target is dashed orange.

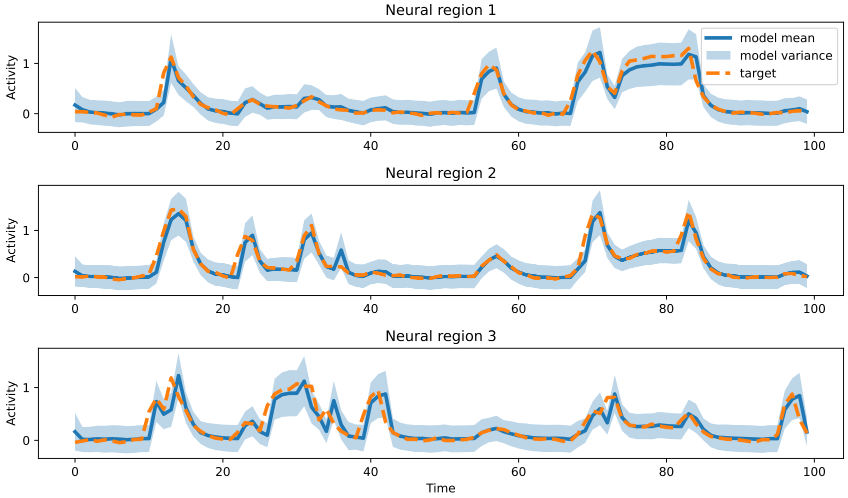

Figure 15: AGM-TE approximates $y_{t+1}|\mathbf{x_t}, \mathbf{y_t}$ in the test data in a system with a 3-dimensional Y.

E.3. Interpreting Model Behaviour from the Perspective of Predictive Causality

If both models have uncertainties that reflect their tendency to make errors, it becomes meaningful to compare these quantities. If we examine the predicted distributions over y_{t+1}, we reassuringly find that 94-96% of observations from the previously unseen dataset fall within the 95% confidence interval of the model. This means that since the total variance [across the three neural dimensions] of the distributions forecasted by the AGMs is markedly lower for the q_{Θ_2} model [Fig. 16], information from X consistently reduces uncertainty in the future of Y.

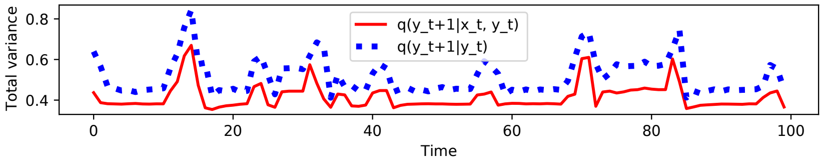

Figure 16: Variance over time in the competing models of Y.

Plotting the cumulative residual sum of squared differences between the predicted mean and the true y_{t+1} confirms this interpretation, as the second model indeed accumulates less error over time in the unseen data [Fig. 17]. Taken together, these results suggest that X improves our predictions of Y, which means that we correctly recover the $X \rightarrow Y$ relationship from the generative model.

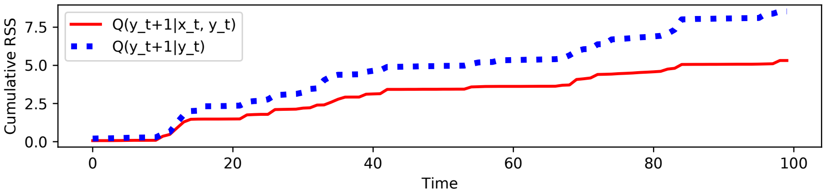

Figure 17: Cumulative residual sum of squares [RSS] over time in the competing models of Y

Model Performance in the Anti-Causal Direction

The *specificity* of this approach can be checked by investigating model performance in the anti-casual direction. We train two AGMs to model the noisy inputs X, with one model using only the past of X, and the alternative using both X and the neural activity Y. When comparing cumulative squared prediction errors in an unseen dataset, the competing models achieve identical poor performance [Fig. 18]. This implies $Y \not\rightarrow X$, as defined in the generative model.

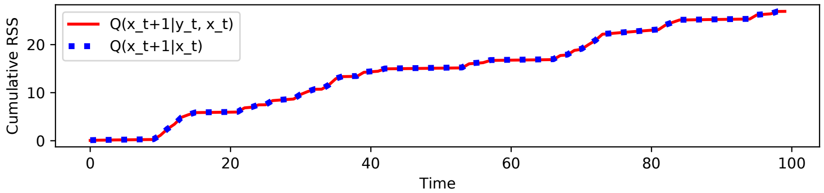

Figure 18: Cumulative residual sum of squares [RSS] over time in the competing models of X

Appendix F. Theoretical Argument for the Convergence of $\hat{\mathcal{T}}_{X\rightarrow Y}$ to the True $\mathcal{T}_{X\rightarrow Y}$

Recall that the losses for the two models are defined such that they estimate the cross entropy

$$\begin{aligned}\text{Loss}(\Theta_1,[\mathbf{Y}]) &:= \frac{1}{T}\sum\nolimits_{t=1}^{T}\left[-\log q(y_{t+1}|\mathbf{y_t})\right] = \hat{H}(p(y_{t+1}|\mathbf{y_t}), q(y_{t+1}|\mathbf{y_t}))\\ \text{Loss}(\Theta_2,[\mathbf{Y},\mathbf{X}]) &:= \frac{1}{T}\sum\nolimits_{t=1}^{T}\left[-\log q(y_{t+1}|\mathbf{y_t},\mathbf{x_t})\right] = \hat{H}(p(y_{t+1}|\mathbf{y_t},\mathbf{x_t}), q(y_{t+1}|\mathbf{y_t},\mathbf{x_t}))\end{aligned}$$

As a cross entropy is the sum of the true entropy of the data generating distribution and the KL divergence between the true and model distributions, our estimated $\hat{\mathcal{T}}_{X\rightarrow Y}$ can be decomposed as

$$\begin{aligned}\hat{\mathcal{T}}_{X\rightarrow Y} &:= \text{Loss}(\Theta_1,[\mathbf{Y}]) - \text{Loss}(\Theta_2,[\mathbf{YX}])\\ &= \hat{H}(p(y_{t+1}|\mathbf{y_t}), q(y_{t+1}|\mathbf{y_t})) - \hat{H}(p(y_{t+1}|\mathbf{y_t},\mathbf{x_t}), q(y_{t+1}|\mathbf{y_t},\mathbf{x_t}))\\ &= [\mathcal{H}(y_{t+1}|\mathbf{y_t}) + D_{KL}(p(y_{t+1}|\mathbf{y_t})||q(y_{t+1}|\mathbf{y_t}))]\\ &\quad - [\mathcal{H}(y_{t+1}|\mathbf{y_t},\mathbf{x_t}) + D_{KL}(p(y_{t+1}|\mathbf{y_t},\mathbf{x_t})||q(y_{t+1}|\mathbf{y_t},\mathbf{x_t}))]\end{aligned}$$

Rearranging our four terms, we find that our estimated $\hat{\mathcal{T}}_{X\rightarrow Y}$ is equal to the true $\mathcal{T}_{X\rightarrow Y}$ plus an error term formed from the subtraction of two KL divergences.

$$\begin{aligned}\hat{\mathcal{T}}_{X\rightarrow Y} &= [\mathcal{H}(y_{t+1}|\mathbf{y_t}) - \mathcal{H}(y_{t+1}|\mathbf{y_t},\mathbf{x_t})]\\ &\quad + [D_{KL}(p(y_{t+1}|\mathbf{y_t})||q(y_{t+1}|\mathbf{y_t})) - D_{KL}(p(y_{t+1}|\mathbf{y_t},\mathbf{x_t})||q(y_{t+1}|\mathbf{y_t},\mathbf{x_t}))]\\ &= \mathcal{T}_{X\rightarrow Y}\\ &\quad + [D_{KL}(p(y_{t+1}|\mathbf{y_t})||q(y_{t+1}|\mathbf{y_t})) - D_{KL}(p(y_{t+1}|\mathbf{y_t},\mathbf{x_t})||q(y_{t+1}|\mathbf{y_t},\mathbf{x_t}))]\end{aligned}$$

This has two important consequences. First, as KL divergences are non-negative, they counteract each other, which leads to a smaller magnitude of the overall error bias (Garg et al., 2022). Second, since the objective of training is to minimise the cross entropies of each model [which entails the minimisation of KL divergences, since the true entropy is fixed], the process of training will gradually reduce the error term, causing our estimate to converge to the true $\mathcal{T}_{X\rightarrow Y}$.

We demonstrate this empirically by conducting 50 replicate re-analyses of a dataset with $T = 20000$ from the Linear Gaussian generative model [see Appendix G] which has an analytically tractable transfer entropy. Fig. 19 clearly shows how the estimated TEs converge over training.

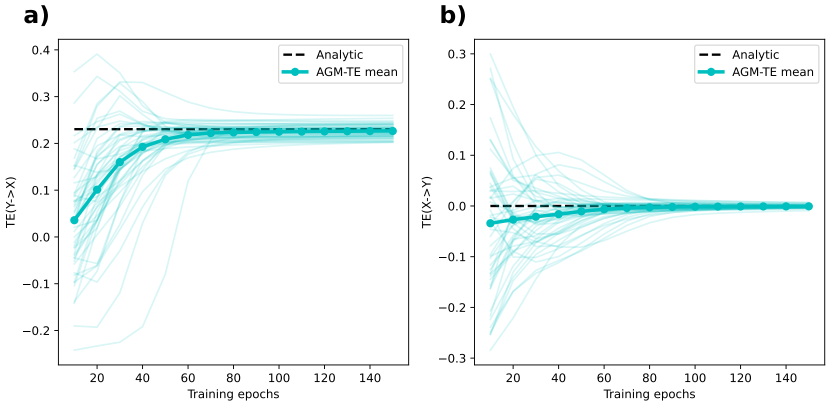

Figure 19: 50 replicate AGM-TE analyses converge during training to the true **a)** $\mathcal{T}_{Y\rightarrow X}$ and **b)** $\mathcal{T}_{X\rightarrow Y}$ values of a Linear Gaussian data generating system.

Appendix G. Generating Synthetic Benchmark Data

G.1. Source Code Availability

We provide the data generating systems in the Python package `te_datasim` [available on github], which may be of interest to other researchers developing or comparing TE estimators, or other causal discovery methods. The `Simulator` class in the package allows users to set up data generators with known parameters. The `.simulate()` method allows sampling of datasets with arbitrary lengths, and the `.analytic_transfer_entropy()` method calculates ground truth transfer entropy values. This package was used to generate the benchmark data used in this article.

G.2. The Two Base System Classes

G.2.1. THE *Linear Gaussian* [LG] MODEL

The linear Gaussian model generates scalar values x_t and y_t using the following equations:

$$x_{t+1} = b_x x_t + \mathcal{E} \sim \mathcal{N}(0, \sigma_x^2) + \lambda y_t \tag{14}$$

$$y_{t+1} = b_y y_t + \mathcal{E} \sim \mathcal{N}(0, \sigma_y^2) \tag{15}$$

From the equations, it is clear that $\mathcal{T}_{X \to Y}$ is always 0. Edinburgh et al. (2021) derived an analytic formula for how $\mathcal{T}_{Y \to X}$ depends on the parameters of the equations. Most importantly, how $\mathcal{T}_{Y \to X}$ increases with the *coupling parameter* λ.

$$\mathcal{T}_{Y \to X} = \frac{1}{2} \log \left[\frac{((1-b_y^2)((1-b_x b_y)^2)\sigma_x^4) + (2\lambda^2(1-(b_x b_y))\sigma_x^2\sigma_y^2) + (\lambda^4\sigma_y^4)}{((1-b_y^2)(1-b_x b_y)^2\sigma_x^4) + (\lambda^2(1-b_x^2 b_y^2)(\sigma_x^2\sigma_y^2))} \right] \tag{16}$$

This process can also be described graphically using the full time causal graph of Fig. 20.

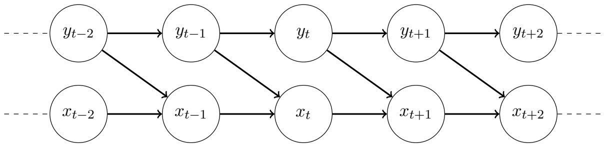

Figure 20: Full time causal graph for the Linear Gaussian system

This bivariate system is accessed through the `BVLinearGaussianSimulator` class.

G.2.2. THE *Joint Process* [JP] [LG] MODEL

The joint process model generates scalar values x_t and y_t using the following equations

$$x_t = \mathcal{E} \sim \mathcal{N}(0, 1) \tag{17}$$

$$y_t = \begin{cases} \mathcal{E} \sim \mathcal{N}(0, 1) & \text{if } y_{t-1} < \lambda \\ \rho x_{t-1} + \sqrt{1-\rho^2} \times \mathcal{E} \sim \mathcal{N}(0, 1) & \text{if } y_{t-1} \geq \lambda \end{cases} \tag{18}$$

λ is a *threshold parameter*. Intuitively, for large λ, there is no information flow from the past of X to Y, while for smaller λ, y_t is increasingly determined by x_{t-1}. Specifically, Zhang et al. (2019) find that the transfer entropy is

$$\mathcal{T}_{X\to Y} = -0.5Q(\lambda)\log(1-\rho^2) \tag{19}$$

where Q is the complementary cumulative distribution function of a standard Gaussian variable. For this system, $\mathcal{T}_{Y\to X}$ is always 0.
This can also be described graphically using the full time causal graph of Fig. 21.

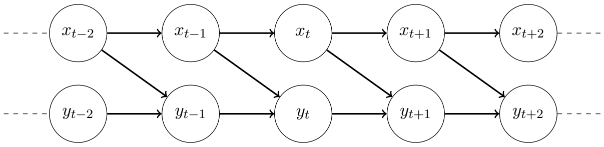

Figure 21: Full time causal graph for the Joint Process system

This bivariate system is accessed through the `BVJointProcesSimulator` class.

G.3. Scaling Data Generation to Multiple Dimensions

In many empirical systems of interest, variables may be vector-valued. To facilitate the generation of such data [from the `MVLinearGaussianSimulator` and `MVJointProcessSimulator` classes], while retaining the analytical tractability of the resulting transfer entropies, we compose scalar datasets into vector-valued ones using the following procedures:

G.3.1. Linear Stacking

If we concatenate data generated by d independent copies of a system with scalar variables $X^{[i]}$ and $Y^{[i]}$ for $i \in 1, 2, 3, ..., d$, the transfer entropy between the resulting d-dimensional vector valued variables $x_t = \{x_t^{[1]}, x_t^{[2]}, ..., x_t^{[d]}\}$ and $y_t = \{y_t^{[1]}, y_t^{[2]}, ..., y_t^{[d]}\}$ is the sum of the transfer entropies of the individual systems. This is because $X^{[i]}$ may only carry information about $Y^{[i]}$, but never $Y^{[j\neq i]}$.

$$\mathcal{T}_{X\to Y} = \sum_{i=1}^{d} \mathcal{T}_{X^{[i]}\to Y^{[i]}}$$

G.3.2. Redundant Stacking

The transfer entropy between any pair of variables that are both drawn i.i.d. is 0. If we have a d-dimensional system, which we concatenate with n variables that are drawn i.i.d., the transfer entropy of the resulting $(d+n)$-dimensional system remains unchanged.

$$\mathcal{T}_{X\to Y} = \sum_{i=1}^{d} \mathcal{T}_{X^{[i]}\to Y^{[i]}} + \sum_{i=d+1}^{d+n} \mathcal{T}_{X^{[i]}\to Y^{[i]}} = \sum_{i=1}^{d} \mathcal{T}_{X^{[i]}\to Y^{[i]}} + 0$$

Naturally, linear stacking may be combined with redundant stacking as needed.

Appendix H. TE Estimation Benchmarks

H.1. Estimator Methods

For the existing ML methods C-MI-GAN and NJEE, we used the implementations and default hyperparameters provided in public repositories by the original authors.

For AGM-TE, we used a Gaussian observation model, and a feedforward NN dynamics model with $16 \times d$ neurons across two hidden layers. We optimised the model using stochastic gradient descent with a learning rate of 0.01 for 1000 epochs.

H.2. Benchmarking Scenarios and Detailed Results

H.2.1. Sample Efficiency Testing

To test the sample efficiency of the four methods, we used datasets from the 1D LG and JP systems with a given λ [0.5 for LG and 0 for JP]. We varied the number of samples from 500 to 10000 to establish the results of Fig. 22, which show that both the error and variance of AGM-TE are lower at a given sample size than competing ML methods.

We can also observe that previous ML methods [especially NJEE] tend to overestimate TE at lower sample sizes, a problem that does not affect AGM-TE. We provide a detailed exploration and explanation of the overestimation by NJEE in Appendix C.

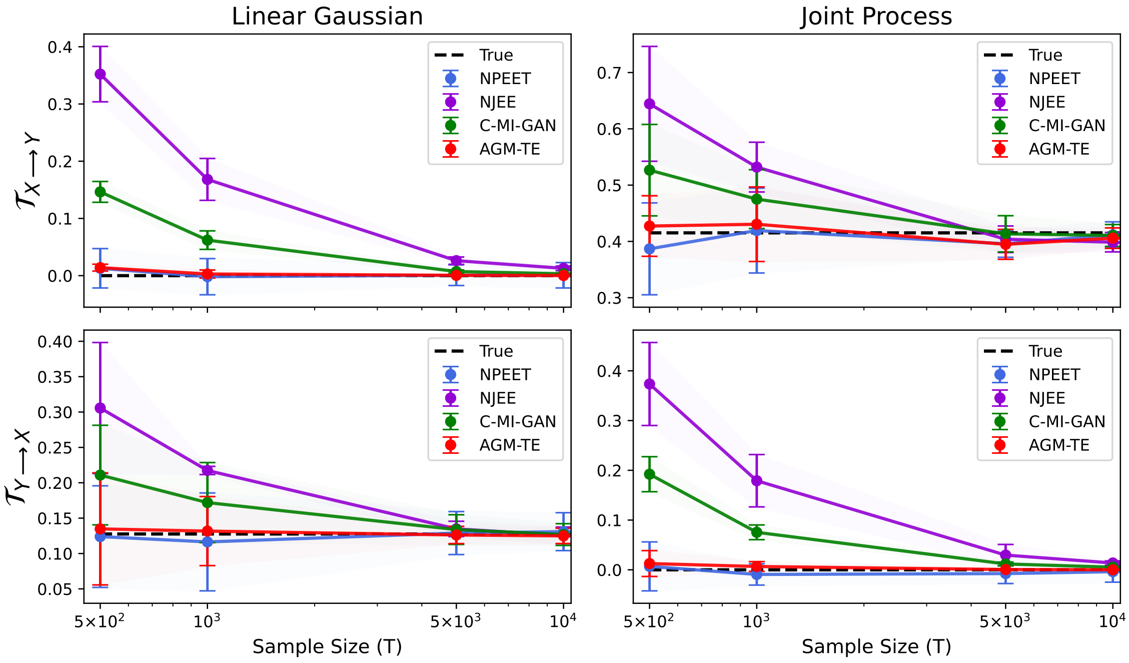

Figure 22: Convergence of the estimated to the true TE with sample size across the four different methods, and four different data generating systems.

H.2.2. 1D SYSTEM

In the first set of benchmarks [not discussed in the main text], we gradually change the true TE by sweeping the λ parameter of both models in 9 steps to vary the TE between the target and causal variable in the LG and JP models. This leads to 36 different transfer entropy estimation tasks, of which 19 have a true TE of 0 [all anti-causal cases, and $\lambda = 0$ in the LG model].

We use a sample size of 10000, and perform five replicate simulations to establish the results of Fig. 23, which show that all four methods are able to accurately estimate the true TE. Note the accurate and low-variance inference of 0 TE values by AGM-TE. This would suggest a low false positive rate when detecting causal interactions. The mean absolute errors for the four transfer entropy estimation tasks classes [LG $\mathcal{T}_{X \to Y}$, LG $\mathcal{T}_{Y \to X}$, JP $\mathcal{T}_{X \to Y}$, JP $\mathcal{T}_{Y \to X}$] are added along the sweep of λ to generate the cumulative absolute error [CAE] scores of Table 1, which show that AGM-TE performs best in three out of the four transfer entropy estimation tasks.

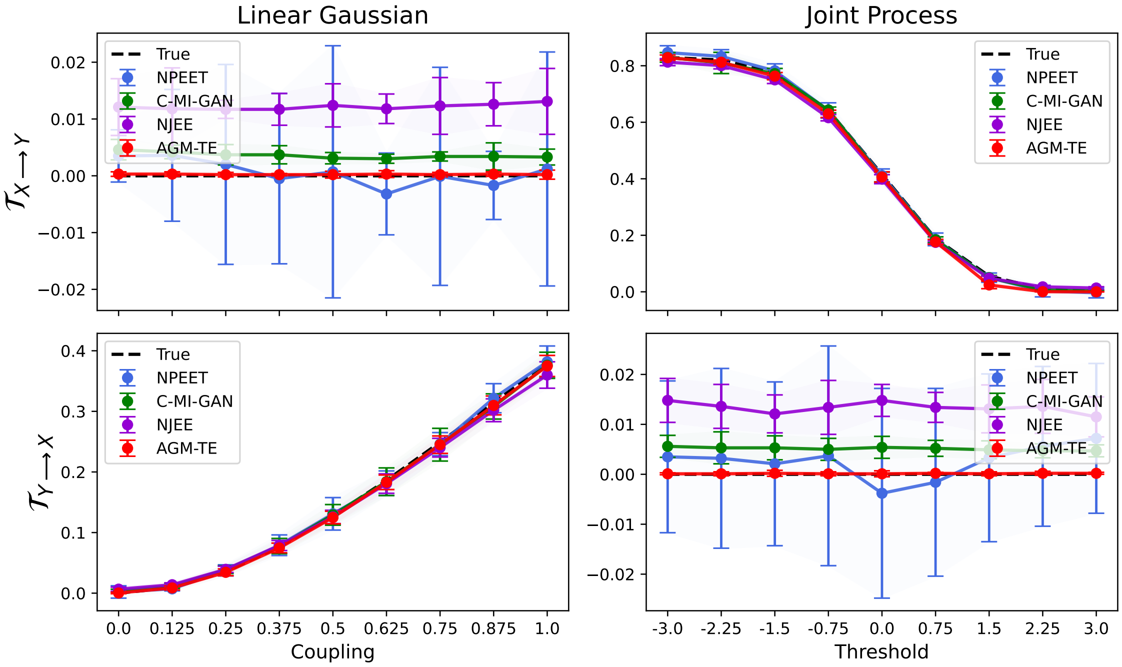

Figure 23: True and estimated transfer entropies in the 1D systems

method	LG $\mathcal{T}_{X \to Y}$	LG $\mathcal{T}_{Y \to X}$	JP $\mathcal{T}_{X \to Y}$	JP $\mathcal{T}_{Y \to X}$
NPEET	0.0165	0.0294	0.0617	0.0340
C-MI-GAN	0.0324	0.0291	**0.0403**	0.0461
NJEE	0.1095	0.0654	0.1439	0.1203
AGM-TE	**0.0022**	**0.0228**	0.0949	**0.0013**

Table 1: Cumulative absolute error of each method in the 1D benchmarks

H.2.3. REDUNDANT STACKING TO INCREASE DIMENSIONALITY

In the redundant stacking benchmarks, we concatenate $n \in 0 : 9$ channels of i.i.d. Gaussian noise onto data generated from a 1D JP or LG system while leaving λ fixed [0.5 for LG and 0 for JP], leading to a set of 40 TE estimation tasks that gradually increase in dimensionality, but have a constant true TE value for a given system.

We use a sample size of 100000, and conduct 5 replicate simulations to establish the results of Fig. 24, which empirically demonstrate the poor dimensionality scaling of kNN-KSG methods. While all ML methods are able to complete this task reasonably well, NJEE suffers from a gradually increasing [albeit small] false positive TE in the anti-causal direction. This echoes the small sample size results in Appendix C, together suggesting that previous cross entropy methods relying on the categorical distribution will overestimate TE when the sample size is small relative to the complexity of the data generating system. The cumulative absolute errors in the 40 tasks (Table 2) indicate that AGM-TE performed the best.

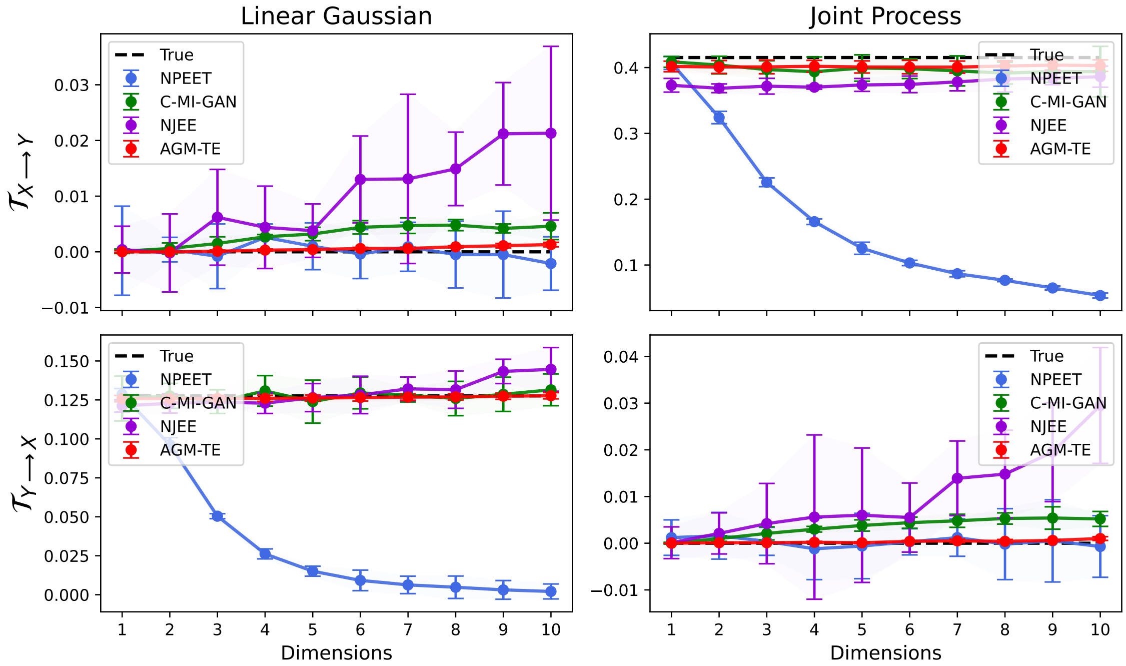

Figure 24: True and estimated transfer entropies in the redundant stacking tests

method	LG $\mathcal{T}_{X\to Y}$	LG $\mathcal{T}_{Y\to X}$	JP $\mathcal{T}_{X\to Y}$	JP $\mathcal{T}_{Y\to X}$
NPEET	0.0094	0.9359	2.5188	0.0079
C-MI-GAN	0.0308	0.0213	0.1770	0.0350
NJEE	0.0985	0.0628	0.3896	0.1012
AGM-TE	**0.0053**	**0.0112**	**0.1352**	**0.0034**

Table 2: Cumulative absolute error of each method in the redundant stacking benchmarks

H.2.4. Linear Stacking to Increase Dimensionality

To test the ability of the estimators to converge to high TE values, we concatenate $d \in 1 : 10$ channels of data from independently simulated runs of the 1D LG or JP systems with a given λ [0.5 for LG and 0 for JP]. This leads to a set of 40 TE estimation tasks where the true TE increases linearly with the number of dimensions.

We use a sample size of 100000, and conduct 5 replicate simulations to establish the results of Fig. 25. These tests empirically demonstrate the slow convergence of variational methods theorised by McAllester and Stratos (2020), as C-MI-GAN is able to successfully scale in the LG system with a smaller starting TE, but reaches a plateau in the JP system, unlike the cross entropy methods [AGM-TE and NJEE]. As in the redundant stacking tests, the NPEET is unable to correctly estimate TE for high dimensional data. Table 3 again shows AGM-TE producing the best results in terms of cumulative absolute error, outperforming alternatives by at least a factor of two.

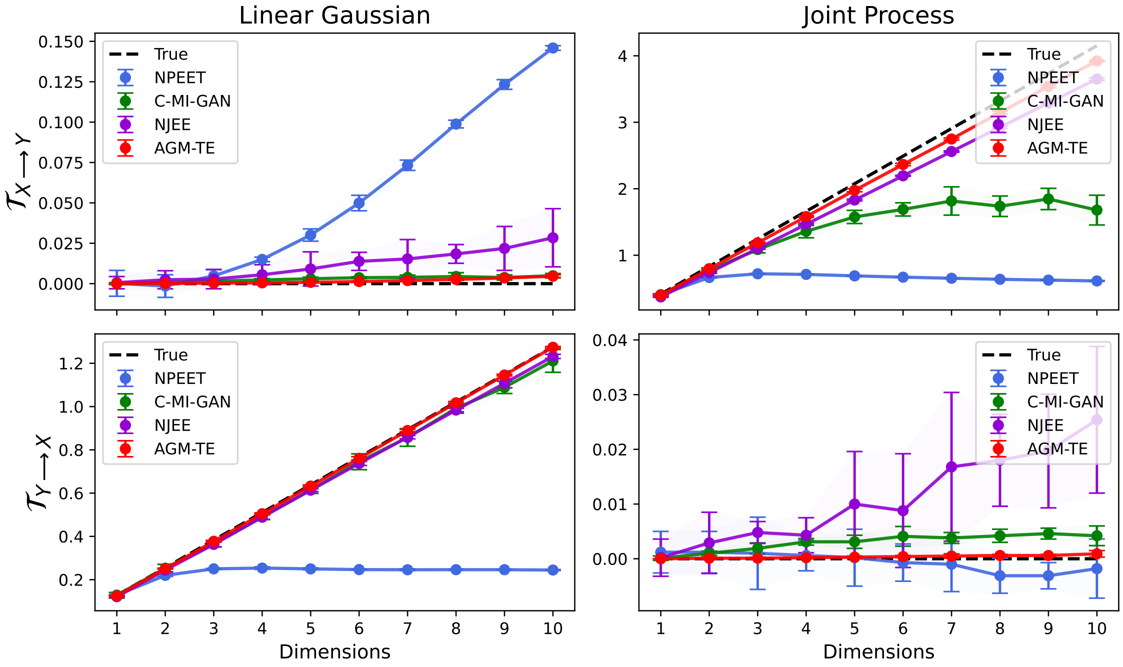

Figure 25: True and estimated transfer entropies in the linear stacking tests

method	LG $\mathcal{T}_{X\rightarrow Y}$	LG $\mathcal{T}_{Y\rightarrow X}$	JP $\mathcal{T}_{X\rightarrow Y}$	JP $\mathcal{T}_{Y\rightarrow X}$
NPEET	0.5428	4.6885	16.4389	0.0139
C-MI-GAN	0.0283	0.2725	8.8677	0.0301
NJEE	0.1181	0.2672	2.704	0.1109
AGM-TE	**0.0156**	**0.0397**	**1.166**	**0.0037**

Table 3: Cumulative absolute error of each method in the linear stacking benchmarks

Appendix I. Detailed Methods for Neural Data Analysis

I.1. Details of the Spike Dataset

The dataset of Siegle et al. (2021) consists of recordings from six areas of the mouse visual cortex (Fig. 26) in awake animals viewing diverse visual stimuli.

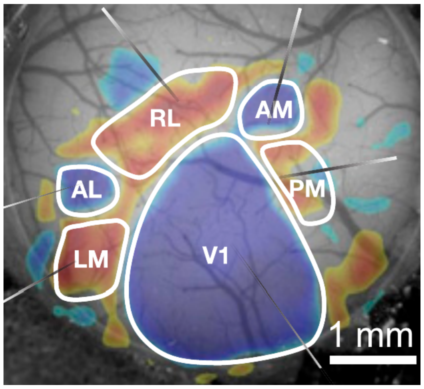

Figure 26: Six areas of the mouse visual cortex recorded with Neuropixels. Source: Jia et al. (2022)

There were a total of eight $\sim$ 3 hour recordings, each from a different specimen [Allen Brain Observatory Session IDs 719161530, 750332458, 750749662, 754312389, 755434585, 756029989, 791319847, 797828357]. In the eight datasets, the number of total recorded neurons across the 6 regions ranged from 155 to 345, with a mean of 272. The number of neurons per region ranged from 9 to 91, with a mean of 45, making each variable highly multidimensional.

During a recording, 15 to 18 [mean 16] trials involving a single type of active visual stimulation [e.g. flashes, movies] were interspersed with periods of rest. We processed the raw data by isolating periods of 100 seconds around the start of each stimulation trial [10 s before, 90 s after]. For each neuron, we counted the number of spikes within 100 ms bins, yielding a total of 1000 time points per channel per trial. The total dataset from a single specimen is therefore an array of size approximately $[b \approx 16 \times d \approx 272 \times T = 1000]$, where b is the batch size [numer of trials] d is the number of neurons, and T is the number of timesteps in a 100 second trial.

I.2. AGM Configuration

For the empirical dataset, our latent dynamics model f was a GRU RNN with 50 units in a single hidden layer. In the synthetic dataset, a smaller dynamics model with 5 units was used. We used the Poisson observation model detailed in Appendix D.3.2.

We used the Adam optimiser during training, which lasted for 10000 epochs with a starting learning rate of 0.001, which decayed by a factor of 0.9 every 100 epochs.

I.3. Validation in Synthetic Data

Before analysing empirical data, we validate the effective connectivity inference capabilities of AGM-TE in a small-scale synthetic system that mimics the spiking activity of the brain, in a process similar to the methods of Kim et al. (2011).

Data Generation

To this end, we sample spiking data from 15 “neurons” by using the input driven neural state equation model [Appendix E.1] to control the rates of a set of independent Poisson processes. For our experiments, **A** defines a system composed of three regions with five neurons in each. Our input matrix **C** channels inputs to the five neurons in the first region. This “driver” region then directly excites “primary driven” and “secondary driven” regions. The “primary driven” region also excites the “secondary driven” region, introducing a confounding effect (Fig. 27).

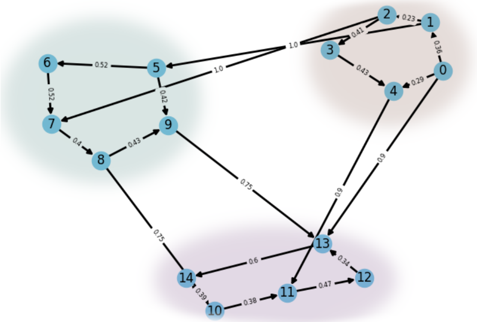

Figure 27: Network connectivity of synthetic system of three regions and 15 total neurons. Connectivity is defined by the matrix **A**, a parameter of Eq. 13

The 15 dimensional neural state variable y_t is then passed through $\mu_t = \beta \exp(y_t + b)\Delta$ to yield the non-negative vector μ_t, which specifies the rate parameters of an inhomogeneous Poisson process for each of the 15 “neurons”. The β and b parameters for transformation were chosen to reasonably match the mean (≈ 0.65) and standard deviation (≈ 0.85) of spike counts per time bin seen in neurons from the empirical dataset of Siegle et al. (2021).

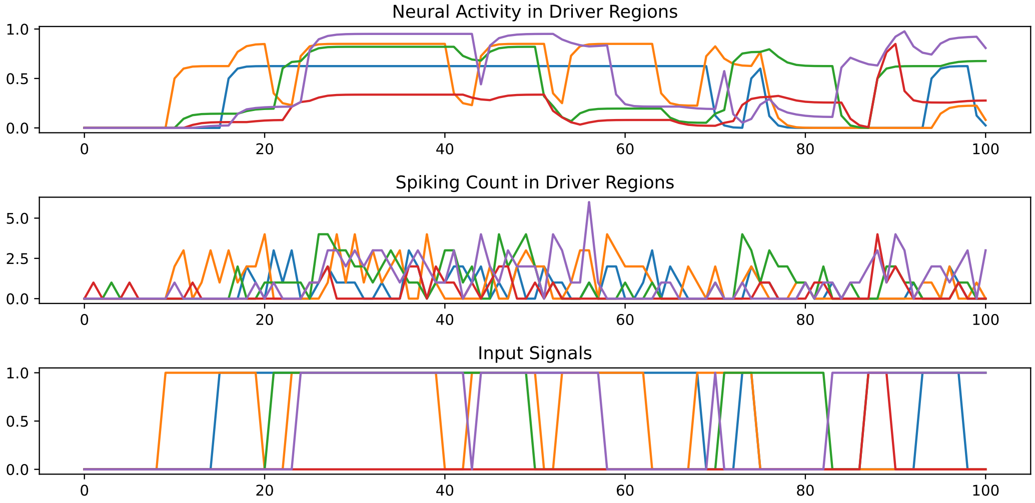

Figure 28: Neural activity, observed number of spikes, and driving noise in the 5 neurons of the driver region in the synthetic dataset.

Analysis and Interpretation

We estimate six CTEs, of which two [from the "driver" region to the "primary driven" and "secondary driven" regions] are known to be non-zero. We conducted 8 replicate runs, inferring CTE estimates from a dataset of 16000 timesteps. Averaging the estimates over replicates and normalising the highest value to 1 yields the effective connectivity matrix of Fig. 29. This matches the ground truth connectivity structure of the synthetic system.

When effective connectivity was estimated using classical transfer entropy, the total inferred information flow for the four non-causal interactions increased by a factor of three, demonstrating that conditioning helps reduce false causal positives.

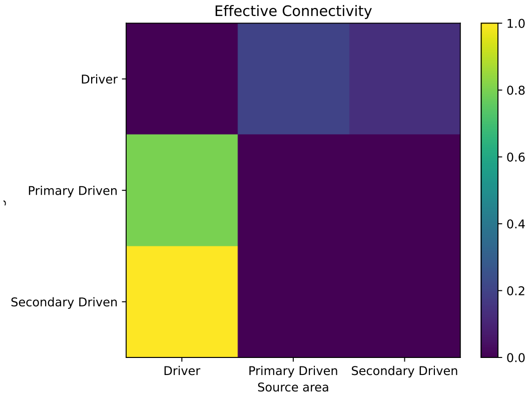

Figure 29: Effective connectivity of synthetic system as inferred by AGM-TE matches the ground truth network structure

I.4. Details of the Empirical EC Analysis Procedure

For each of the eight datasets, we conducted three replicate EC matrix inference passes. The 24 results were then averaged to form our final empirical EC estimate, and the scores in the matrix are then normalized such that the largest value is 1.

To quantify the correspondence between the inferred effective connectivity matrix $\mathbf{C_I}$ and the reference structural connectivity matrix $\mathbf{C_R}$ of Gămănuţ et al. (2018) [which was also normalised to have a maximum value of 1], we started by measuring the correlation coefficient of matrix elements, which yielded a result of $\rho_{\mathbf{C_I},\mathbf{C_R}} = 0.5638$ with a p-value of 0.0003.

Rather than reporting the p-value directly, we used a bootstrapping procedure to estimate the probability of achieving correlations of this magnitude or higher. We take random samples [with replacement] from off-diagonal values of $\mathbf{C_I}$ to fill the off-diagonal values of a sampled connectivity matrix $\mathbf{C_S}$, and measure the correlation coefficient between values of $\mathbf{C_R}$ and $\mathbf{C_S}$. Repeating the procedure 25000 times yields a distribution over correlation coefficients of matrices with elements [effective connectivity values] similar to $\mathbf{C_I}$, but placed randomly in off-diagonal locations, which enables us to estimate the degree to which the structure of the matrices [the order of values] match.

The original $\rho_{\mathbf{C_I},\mathbf{C_R}}$ value was found to be in the 99.7th percentile of the bootstrap ρ distribution, indicating that our inferred effective connectivity matrix exhibits a structure that is more similar to the results of Gămănuţ et al. (2018) than would be expected by chance.