TENDE: Transfer Entropy Neural Diffusion Estimation

Simon Pedro Galeano Muñoz
KAUST, Saudi Arabia

Mustapha Bounoua
EURECOM, France

Giulio Franzese
EURECOM, France

Pietro Michiardi
EURECOM, France

Maurizio Filippone
KAUST, Saudi Arabia

Abstract

Transfer entropy is a fundamental measure for quantifying directed information flow in time series, with applications spanning neuroscience, finance, and complex systems analysis. However, existing estimation methods suffer from the curse of dimensionality, require restrictive distributional assumptions, or need exponentially large datasets for reliable convergence. We address these limitations in the literature by proposing TENDE (Transfer Entropy Neural Diffusion Estimation), a novel approach that leverages score-based diffusion models to estimate transfer entropy through conditional mutual information. By learning score functions of the relevant conditional distributions, TENDE provides flexible, scalable estimation while making minimal assumptions about the underlying data-generating process. We demonstrate superior accuracy and robustness compared to existing neural estimators and other state-of-the-art approaches across synthetic benchmarks and real data.

1 INTRODUCTION

Estimating dependencies between variables is a fundamental problem in Statistics and Machine Learning. For time series, this becomes particularly important in applications in neuroscience (Parente and Colosimo, 2021; El-Yaagoubi et al., 2025; Wang et al., 2025), where researchers analyze information flow between brain regions, and in finance (Patton, 2012; Gong and Huser, 2022; Caserini and Pagnottoni, 2022), where understanding relationships between assets is crucial for risk assessment. The challenge is to quantify these dependencies without assuming specific functional relationships between time series, while making minimal distributional assumptions. Transfer entropy (TE), introduced by Schreiber (2000), addresses this by measuring directed information flow between time series through conditional mutual information (CMI). However, the high-dimensional nature of the problem, considering both current values and historical lags, makes reliable estimation difficult.

Existing methods face significant limitations. Traditional approaches based on k-nearest neighbors (Lindner et al., 2011) suffer from the curse of dimensionality. Recent neural estimators using variational bounds (Zhang et al., 2019) can require exponentially large datasets for convergence (McAllester and Stratos, 2020), while copula-based methods (Redondo et al., 2023) or the use of entropy arguments (Kornai et al., 2025) impose restrictive assumptions. Recent advances in score-based diffusion models (Song et al., 2020) offer a promising solution. These models excel at learning complex probability distributions by estimating score functions, and accurate density estimation is sufficient for computing information-theoretic measures. Building on connections between diffusion models and KL divergence estimation (Franzese et al., 2023), we can leverage these advances for transfer entropy estimation.

In this work, we propose TENDE (Transfer Entropy Neural Diffusion Estimation), which uses score-based diffusion models to estimate transfer entropy. Our approach is flexible and scalable, and makes minimal distributional assumptions while providing accurate estimates even in high-dimensional settings.

The paper is organized as follows: § 2 introduces the fundamental concepts of transfer entropy and its formal definition. § 3 reviews related work on estimation methods, and § 4 presents our diffusion-based estimator. § 5 provides a comparative analysis against KNN,

Proceedings of the 29th International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300.

copula, cross-entropy, and Donsker-Varadhan based approaches. § 6 demonstrates the method on the Santa Fe B time series dataset to illustrate its practical applicability. § 7 concludes discussing future directions.

2 BACKGROUND

2.1 Mutual Information and Conditional Mutual Information

Capturing the dependence between random variables is a recurrent problem in several applications of Statistics and Machine Learning. The Mutual Information (MI) is an attractive measure of dependence when the relation between the variables is unknown and possibly nonlinear. The MI is defined as follows: let $X \in \mathbb{R}^{N_x}$ and $Y \in \mathbb{R}^{N_y}$ be random variables with joint probability density $p_{X,Y}$ and marginal densities p_X and p_Y respectively[1], the mutual information between X and Y is given by

$$I(X;Y) = D_{\mathrm{KL}}\left[p_{X,Y} \,||\, p_X \; p_Y\right], \tag{1}$$

where $D_{\mathrm{KL}}\left[p \,||\, q\right]$ denotes the Kullback–Leibler (KL) Divergence between the distributions p and q and is defined as

$$D_{\mathrm{KL}}\left[p \,||\, q\right] = \mathbb{E}_{x\sim p}\left[\log\left(\frac{p(x)}{q(x)}\right)\right]. \tag{2}$$

It is worth recalling that the MI between X and Y equals zero if and only if $p_{X,Y} = p_X \; p_Y$, that is, if and only if X and Y are independent random variables. While MI has been widely employed in diverse domains, it only captures unconditional pairwise dependencies. In many applications, however, the relationship between two variables may be driven by a set of other variables. To address this, MI naturally extends to its conditional form, the conditional mutual information, which quantifies the dependence between two random variables X and Y given a third variable Z. Formally, CMI is defined as follows:

$$I(X;Y|Z) = \mathbb{E}_{z\sim p_Z}\left[I(X_z;Y_z)\right]. \tag{3}$$

Here X_z and Y_z denote the random variables $X|Z=z$ and $Y|Z=z$, thus Eq. (3) represents the average MI between X and Y where Z is known, that is, the mean KL divergence between p_{X_z,Y_z} and $p_{X_z}\; p_{Y_z}$ where p_{X_z,Y_z}, p_{X_z}, and p_{Y_z} represent the joint density of X and Y and the marginal densities of X and Y conditioned on $Z=z$ respectively. Analogously p_Z is the marginal density of the random variable Z.

[1] MI can be defined also for variables without densities and in more generic spaces, but for the purpose of this work the restriction considered here is sufficient.

This perspective is crucial in scenarios where the apparent association between X and Y may be entirely driven by their joint dependence on Z, rather than reflecting a direct relationship. By conditioning on Z, CMI provides a principled way to disentangle direct from indirect dependencies, offering a more refined characterization of the underlying dependence structure. Such considerations are particularly important in complex systems where interactions among variables are often mediated through latent or observed confounders.

Although MI and CMI are measures of general dependence between random, neither can capture the directionality of the dependence since we have $I(X;Y) = I(Y;X)$ and $I(X;Y|Z) = I(Y;X|Z)$; the equalities can be easily seen from the symmetric form in which the joint and marginal distributions appear in the KL divergence. In many applications such as the ones described in Baccalá and Sameshima (2001); Kayser et al. (2009); Wang et al. (2022); Cîrstian et al. (2023), it is highly desirable to identify not only whether two variables are dependent but also the direction of dependence, as this could provide insight into the underlying mechanisms that govern the system at hand. Without accounting for directionality, analyses may overlook critical asymmetries in the flow of information that determine how complex systems evolve.

2.2 Transfer Entropy

To solve this issue Schreiber (2000) developed the concept of transfer entropy. Let $\{X_t\}$ and $\{Y_t\}$ denote N_x-dimensional and N_y-dimensional time series, respectively. Define

$$\begin{cases} \mathbf{Y}_{t-\ell} = [Y_{t-1},\; \ldots,\; Y_{t-\ell}] \\ \mathbf{X}_{t-k} = [X_{t-1},\; \ldots,\; X_{t-k}], \end{cases}$$

for some natural numbers ℓ, k. Thus, the TE from $\{X_t\}$ to $\{Y_t\}$ is given by

$$\mathrm{TE}_{X\to Y}(k,\ell) = I\left(Y_t; \mathbf{X}_{t-k} | \mathbf{Y}_{t-\ell}\right). \tag{4}$$

TE quantifies how much Y_t depends on the past of $\{X_t\}$ once its past is already known. If Y_t is independent of $\mathbf{X}_{t-k}$ once $\mathbf{Y}_{t-\ell}$ is observed, then $TE(X \to Y; k, \ell) = 0$. Hence, a positive transfer entropy indicates that the past of $\{X_t\}$ contains unique predictive information about Y_t that is not already present in its own history. It can be observed from the definition TE that it is not symmetric, this is because in general

$$I\left(X_t; \mathbf{Y}_{t-k} | \mathbf{X}_{t-\ell}\right) \neq I\left(Y_t; \mathbf{X}_{t-k} | \mathbf{Y}_{t-\ell}\right).$$

Transfer entropy has thus become a widely used tool for analyzing directed dependencies in time. However,

its practical application is often limited by challenges related to reliable estimation, particularly in finite-sample and high-dimensional settings (Zhao and Lai, 2020; Gao et al., 2018).

3 RELATED WORK

There are several proposals in the literature on how to estimate the TE between two time series. The first class of proposed methods for this matter, such as the work by Lindner et al. (2011) is based on the use of k-nearest neighbors, leveraging the entropy representation of TE. These estimators are inspired by the methodology described by Frenzel and Pompe (2007), which uses the approach by Kozachenko (1987) to estimate the entropy terms. Although these classical methods remained popular for their ease of use, theoretical and experimental results suggest that they suffer from the curse of dimensionality, as discussed in Zhao and Lai (2020); Gao et al. (2018).

More recently, copulas were used to estimate TE using the fact that MI can be represented as the copula entropy (Ma and Sun, 2011). Redondo et al. (2023) exploit the ability of copulas to decouple marginal effects from the dependence structure, thereby improving the robustness and interpretability in TE estimation. Nevertheless, the simplifying assumption commonly employed in vine copula decompositions (Bedford and Cooke, 2002) to mitigate the curse of dimensionality does not always hold in practice, as demonstrated by Derumigny and Fermanian (2020) and Gijbels et al. (2021). A more comprehensive discussion of this issue is provided by Nagler (2025).

In parallel, neural estimators have been proposed to overcome the limitations of both k-nearest neighbors and copula-based methods. These approaches leverage the expressive power of neural networks to model complex, nonlinear dependencies between time series without requiring explicit assumptions about the underlying distributions. Among recent proposals, there are two main concepts that are used as the building blocks for the estimation of TE. On the one hand, approaches such as (Zhang et al., 2019; Luxembourg et al., 2025) take advantage of the Donsker-Varadhan variational lower bound on the KL divergence; however, the arguments provided by McAllester and Stratos (2020) imply that methods using this lower bound as means to compute TE require exponentially large datasets. On the other hand, the proposals of Garg et al. (2022); Shalev et al. (2022), and Kornai et al. (2025) use cross-entropy arguments to compute the TE, following the suggestion that methods using upper bounds on entropies will not suffer convergence issues of variational approaches. Despite overcoming the limitations of variational methods, Garg et al. (2022) and Shalev et al. (2022) use categorical distributions as means to compute the TE. Even though Kornai et al. (2025) overcame this limitation by avoiding categorical distributions in favor of a parametric estimation of the conditionals, the need to choose a parametric form represents a limitation. We also note the related line of work on neural estimation of directed information for sequential settings (Tsur et al., 2023a), and approaches based on sliced mutual information (Goldfeld and Greenewald, 2021; Tsur et al., 2023b) that address the curse of dimensionality through lower-dimensional projections.

4 METHODS

4.1 General overview of score-based KL divergence estimation

Recent developments in generative modeling (Song et al., 2020) and information-theoretic learning have opened new avenues for TE estimation. In particular, score-based diffusion models provide a principled mechanism to approximate data distributions through the estimation of their score functions, thereby enabling flexible modeling of high-dimensional systems. Parallel to this, advances in mutual information estimation (Franzese et al., 2023; Kong et al., 2023) have improved the accuracy and scalability of this task in less restrictive scenarios. A natural extension is to integrate these two approaches, leveraging the expressive power of diffusion models for distributional representation, while employing modern mutual information and entropy estimators to compute CMI as the building block to quantify directional dependencies.

Recall that X denotes a N_x-dimensional random variable with probability distribution p_X. Under certain regularity conditions, Hyvärinen and Dayan (2005) showed that it is possible to associate the density p_X with the score function S^{p_X}, where for a generic distribution p_X we denote $S^{p_X}(x) := \nabla \log(p_X(x))$, with derivatives taken with respect to x. In addition, it is possible to construct a diffusion process $\{X_t\}_{t\in[0,T]}$ such that $X_0 \sim p_X$ and $X_T \sim p_{X_T}$ where p_{X_T} is a distribution such that there is a tractable way to sample efficiently from it. This diffusion process is modeled as the solution of the following stochastic differential equation:

$$\begin{cases} dX_t = f_t X_t dt + g_t dW_t \\ X_0 \sim p_X, \end{cases} \tag{5}$$

with given continuous functions $f_t \leq 0$, $g_t \geq 0$ for each $t \in [0,T]$, and dW_t is a Brownian motion. The random variable X_t is associated with its density p_{X_t} and therefore with the time-varying score $S^{p_{X_t}}(x)$.

One of the results by Bounoua et al. (2024a) (see also Franzese et al. (2023)) states that if there is another probability density q_X, serving as a reference distribution, for which q_{X_t} is generated by the same diffusion process described in Eq. (5), then the KL divergence between p_X and q_X can be expressed as

$$\int_0^T \frac{g_t^2}{2}\mathbb{E}_{x\sim p_{X_t}}\left[\left\|S^{p_{X_t}}(x) - S^{q_{X_t}}(x)\right\|^2\right] dt + D_{KL}\left[p_{X_T}||q_{X_T}\right], \tag{6}$$

where $\|\cdot\|$ denotes the standard Euclidean norm in $\mathbb{R}^{N_x}$.

This result is a remarkable way to link KL divergence with diffusion processes, given the knowledge on the score functions of p_{X_t} and q_{X_t}. Nonetheless, the availability of such objects is out of reach in practical applications, and that is why this work instead considers parametric approximations of scores. Thus, for a generic distribution p, its score $S^{p_X}(x)$ is approximated by a neural network $S^{p_X}(x;\theta^\star)$ where $\theta^\star$ is obtained by minimizing the loss of denoising score matching (Vincent, 2011). Thus, as stated in Song et al. (2020) for the case of the time-varying score, $\theta^\star$ is obtained by minimizing

$$\int_0^T \mathbb{E}_{x\sim p}\mathbb{E}_{\tilde{x}|x\sim p_{0t}}\left[\left\|S^{p_{X_t}}(\tilde{x};\theta) - S^{p_{0t}}(\tilde{x}|x)\right\|^2\right] dt, \tag{7}$$

where $p_{0t}(\cdot|x)$ denotes the transition density of X_t conditioned on $X_0 = x$, and $S^{p_{0t}}(\tilde{x}|x) = \nabla_{\tilde{x}} \log p_{0t}(\tilde{x}|x)$ is the corresponding score function evaluated at $\tilde{x}$. The marginal score $S^{p_{X_t}}$ at diffusion time t is the quantity being approximated by the neural network. The term inside the integral in Eq. (7) is equivalent to

$$\int p(x)p_{0t}(\tilde{x}|x)\left\|S^{p_{X_t}}(\tilde{x};\theta) - S^{p_{0t}}(\tilde{x}|x)\right\|^2 d\tilde{x}dx. \tag{8}$$

Following the work of Franzese et al. (2023), we adopt the quantity $e(p,q)$ as an estimator of the KL divergence between p and q, with

$$e(p,q) = \int_0^T \frac{g_t^2}{2}\mathbb{E}_{x\sim p_t}\left[\left\|S^{p_{X_t}}(x;\theta_1^\star) - S^{q_{X_t}}(x;\theta_2^\star)\right\|^2\right] dt. \tag{9}$$

This is simply the first term of Eq. (6), where parametric scores are used instead of the true score functions. Under the assumption that the learned scores are sufficiently accurate, the terminal KL divergence $D_{KL}[p_{X_T}||q_{X_T}]$ becomes negligible for large T, and thus (Franzese et al., 2023)

$$e(p,q) \simeq D_{KL}[p||q].$$

A detailed discussion of the approximation error, decomposed into the score estimation error and the terminal divergence, is provided in § A.3.

4.2 Score-based entropy estimation

We now turn our attention to the estimation of entropy using score functions. For this, consider X as previously defined in § 2.1, the entropy is defined as $H(X) = \mathbb{E}_{x\sim p_X}[-\log\ p_X(x)]$, thus it is possible to relate the entropy of a random variable with the KL divergence in the following manner. Let $\varphi_\sigma(\cdot)$ denote the density of a N_x-dimensional centered Gaussian random variable with covariance $\sigma^2\mathbf{I}_{N_x}$, then the entropy of X can be written as

$$H(X) = \frac{N_x}{2}\log(2\pi\sigma^2) + \mathbb{E}_{x\sim p_X}\left[\frac{\|x\|^2}{2\sigma^2}\right] - D_{KL}[p_X||\varphi_\sigma]. \tag{10}$$

Thus, it can be shown that the entropy of X can be estimated as

$$H(X;\sigma) \simeq \frac{N_x}{2}\log\left(2\pi\sigma^2\right) + \mathbb{E}_{x\sim p_X}\left[\frac{\|x\|^2}{2\sigma^2}\right] - e(p_X,\varphi_\sigma) - \frac{N_x}{2}\left(\log(\chi_T) - 1 + \frac{1}{\chi_T}\right). \tag{11}$$

Where for $t\in[0,T]$, $\chi_t = \left(k_t^2\sigma^2 + k_t^2\int_0^t \frac{g_s^2}{k_s^2}ds\right)$ with $k_t = \exp\left\{\int_0^t f_s ds\right\}$. The derivations of Eq. (10) and Eq. (11) can be found in § A.1.

4.3 Score-based conditional mutual information and transfer entropy estimation

In this work, we are interested in the estimation of TE, which is formulated in terms of CMI. For ease of exposition, we provide estimators of the CMI and then state how to use such estimators to compute TE between two time series. Consider random variables $X\in\mathbb{R}^{N_x}$, $Y\in\mathbb{R}^{N_y}$, and $Z\in\mathbb{R}^{N_z}$. The main result in Franzese et al. (2023) provides an accurate way to estimate the KL divergence between two densities p and q utilizing diffusion models, so quantities such as MI or entropies can be estimated since they can be represented in terms of KL divergences. The notation for random variables, conditional random variables, and their respective densities remains analogous to the notation used in § 2. With this in mind, we take advantage of the following expressions that are equivalent to CMI

$$I(X;Y|Z) = \mathbb{E}_{[y,z]\sim p_{Y,Z}}\left[D_{\mathrm{KL}}\left[p_{X_{y,z}}\ ||\ p_{X_z}\right]\right], \tag{12}$$
$$= \mathcal{H}(X|Z) - \mathcal{H}(X|Y,Z), \tag{13}$$
$$= I(X;[Y,Z]) - I(X;Z), \tag{14}$$

where $\mathcal{H}(X|Z) = \mathbb{E}_{z\sim p_z}[H(X_z)]$, the definition of $\mathcal{H}(X|Z,Y)$ is analogous.

Using the estimator $e(\cdot,\cdot)$ from Eq. (9) to approximate each KL divergence term, we obtain the following CMI estimators

$$\mathbb{E}_{[y,z]\sim p_{Y,Z}}\left[D_{\mathrm{KL}}\left[p_{X_{y,z}} \,||\, p_{X_z}\right]\right] \simeq \mathbb{E}_{[y,z]\sim p_{Y,Z}}\left[e(p_{X_{y,z}}, p_{X_z})\right], \tag{15}$$

$$\mathcal{H}(X|Z) - \mathcal{H}(X|Y,Z) \simeq \mathbb{E}_{[y,z]\sim p_{Y,Z}}\left[e(p_{X_{y,z}}, \varphi_\sigma)\right] - \mathbb{E}_{z\sim p_Z}\left[e(p_{X_z}, \varphi_\sigma)\right], \tag{16}$$

$$I(X;[Y,Z]) - I(X;Z) \simeq \mathbb{E}_{[y,z]\sim p_{Y,Z}}\left[e(p_{X_{y,z}}, p_X)\right] - \mathbb{E}_{z\sim p_Z}\left[e(p_{X_z}, p_X)\right]. \tag{17}$$

It is worth mentioning that it is possible to perturb the conditional entropy terms in Eq. (16) by adding and subtracting $e(p_X, \varphi_\sigma)$ appropriately, leading to individual estimators for $I(X;[Y,Z])$ and $I(X;Z)$. As a result, we also propose the following estimator for CMI

$$CMI(X;Y|Z) \simeq \hat{I}(X;[Y,Z]) - \hat{I}(X;Z), \tag{18}$$

with

$$\hat{I}(X;[Y,Z]) = \mathbb{E}_{[y,z]\sim p_{Y,Z}}\left[e(p_{X_{y,z}}, p_X)\right] - e(p_X, \varphi_\sigma),$$

and

$$\hat{I}(X;Z) = \mathbb{E}_{z\sim p_Z}\left[e(p_{X_z}, p_X)\right] - e(p_X, \varphi_\sigma).$$

Among the proposed estimators, Eq. (15) is generally preferable as it is guaranteed to be non-negative, since it directly estimates a KL divergence. The estimators in Eq. (16)–Eq. (18) are valuable when the individual components (e.g., conditional entropies or mutual informations) are of independent interest; however, as difference-based estimators, they may be more susceptible to error propagation. Derivations of the estimators are available in § A.2.

4.3.1 TE estimation

Let $\{X_t\}_{t=1}^T$ be the source series with dimensionality N_{x}, and let $\{Y_t\}_{t=1}^T$ be the target series with dimensionality N_{y}. Choose source and target lags $k, \ell \in \mathbb{N}$. For each time index t with $t > \max(k,\ell)$, a sample is constructed as follows. The future target is given by $Y := Y_t \in \mathbb{R}^{N_{\mathrm{y}}}$. The past of the source is represented as $X := [X_{t-1}, X_{t-2}, \ldots, X_{t-k}] \in \mathbb{R}^{kN_{\mathrm{x}}}$. The past of the target, lags as the conditioning set, is represented as $Z := [Y_{t-1}, Y_{t-2}, \ldots, Y_{t-\ell}] \in \mathbb{R}^{\ell N_{\mathrm{y}}}$.

Stacking these triplets for $t = \max(k,\ell)+1, \ldots, T$ produces a dataset

$$\{(X^{(i)}, Y^{(i)}, Z^{(i)})\}_{i=1}^{T-\max(k,\ell)},$$

which can be directly employed for conditional mutual information estimation. When the underlying processes $\{X_t\}$ and $\{Y_t\}$ are jointly stationary, each window follows the same distribution, so sample averages over temporal windows serve as ergodic approximations of the required expectations. By definition, the transfer entropy from X to Y with lags (k,ℓ) is then expressed as

$$\mathrm{TE}_{X\to Y}(k,\ell) = I(Y; X|Z).$$

Once this dataset is constructed, it can be used to train our proposed score-based conditional mutual information estimator and compute the TE. The way in which $\mathrm{TE}_{Y\to X}(\ell,k)$ can be computed is analogous to what is described above by simply exchanging the roles between $\{X_t\}$ and $\{Y_t\}$.

4.3.2 Algorithm overview

In this work, we employ the variance preserving stochastic differential equation as described in Song et al. (2020) to construct the diffusion process. A key practical advantage of the VP formulation is that the transition density $p_{0t}(\cdot|x)$ is available in closed form as a Gaussian, so obtaining diffused data at any time t requires only sampling from this known distribution rather than numerically solving the SDE. Leveraging the implementation of Bounoua et al. (2024b), we make use of a single score network that approximates all the score functions required to estimate transfer entropy, amortizing the learning of two or three score functions into a single model. In Algorithm 1, the `conditional` approach groups the estimators in Eq. (15) and Eq. (16), which rely only on conditional scores, while the `joint` approach groups the estimators in Eq. (17) and Eq. (18), which additionally require the marginal score. The implementation to estimate the TE in the direction $Y \to X$ is obtained by swapping the roles of X_t and Y_t. Regarding the encoding in the third argument of the network, 1 indicates the variable for which the score is learned, -1 denotes that the corresponding input is marginalized out (set to zero), and 0 indicates that the input is treated as a conditioning signal. Additional details on the network architecture and the amortization procedure are provided in § D.

5 SYNTHETIC BENCHMARK

We now evaluate the estimators proposed in § 4.3 using the benchmark by Kornai et al. (2025) testing our estimators against the methods by Kornai et al. (2025) (Agm), Steeg and Galstyan (2013) (Npeet), an adaptation of (Belghazi et al., 2018) (MINE) to compute conditional mutual information as a means of computing TE, the Transformer-based estimator TREET (Luxembourg et al., 2025), and the conditional independence testing framework implemented in Tigramite (Runge et al., 2019).

Algorithm 1: TENDE

Data: $[X_t, Y_t]$
parameter: `approach` $\in \{$`conditional`, `joint`$\}$, σ, `estimator` $\in \{1, 2\}$
Obtain Y, X, Z as described in § 4.3.1
$t^\star \sim \mathcal{U}[0, T]$
`// diffuse signals to timestep` $t^\star$
$[Y_{t^\star}, X_{t^\star}, Z_{t^\star}] \leftarrow$
$k_{t^\star}[Y, X, Z] + \left(k_{t^\star}^2 \int_0^{t^\star} k_s^{-2} g_s^2 \, \mathrm{d}s\right)^{\frac{1}{2}} [\epsilon_1, \epsilon_2, \epsilon_3]$,
with $\epsilon_{1,2,3} \sim \gamma_1$
`// Use the score network to compute the required scores`
if *approach* $=$ *conditional* **then**
  $S_{x,z}^{p_{Y_{t^\star}}} \leftarrow net_\theta([Y_{t^\star}, X, Z], t^\star, [1, 0, 0])$
  $S_z^{p_{Y_{t^\star}}} \leftarrow net_\theta([Y_{t^\star}, X, Z], t^\star, [1, -1, 0])$
  if *estimator* $= 1$ **then**
    $\hat{I} \leftarrow T \frac{g_{t^\star}^2}{2} \left\| S_{x,z}^{p_{Y_{t^\star}}} - S_z^{p_{Y_{t^\star}}} \right\|^2$ Eq. (15)
  else
    $\chi_{t^\star} \leftarrow \left(k_{t^\star}^2 \sigma^2 + k_{t^\star}^2 \int_0^{t^\star} k_s^{-2} g_s^2 \, \mathrm{d}s\right)$
    $I_1 \leftarrow \left\| S_{x,z}^{p_{Y_{t^\star}}} + \frac{Y_{t^\star}}{\chi_{t^\star}} \right\|^2$
    $I_2 \leftarrow \left\| S_z^{p_{Y_{t^\star}}} + \frac{Y_{t^\star}}{\chi_{t^\star}} \right\|^2$
    $\hat{I} \leftarrow T \frac{g_{t^\star}^2}{2} [I_1 - I_2]$ Eq. (16)
else
  $S_{y,z}^{p_{X_{t^\star}}} \leftarrow net_\theta([Y_{t^\star}, X, Z], t^\star, [1, 0, 0])$
  $S_z^{p_{X_{t^\star}}} \leftarrow net_\theta([Y_{t^\star}, X, Z], t^\star, [1, -1, 0])$
  $S^{p_{X_{t^\star}}} \leftarrow net_\theta([[Y_{t^\star}, X, Z], t^\star, [1, -1, -1])$
  if *estimator* $= 1$ **then**
    $I_1 \leftarrow \left\| S_{x,z}^{p_{Y_{t^\star}}} - S^{p_{Y_{t^\star}}} \right\|^2$
    $I_2 \leftarrow \left\| S_z^{p_{Y_{t^\star}}} - S^{p_{Y_{t^\star}}} \right\|^2$
    $\hat{I} \leftarrow T \frac{g_{t^\star}^2}{2} (I_1 - I_2)$ Eq. (17)
  else
    $\chi_{t^\star} \leftarrow \left(k_{t^\star}^2 \sigma^2 + k_{t^\star}^2 \int_0^{t^\star} k_s^{-2} g_s^2 \, \mathrm{d}s\right)$
    $I_1 \leftarrow \left\| S_{x,z}^{p_{Y_{t^\star}}} + \frac{Y_{t^\star}}{\chi_{t^\star}} \right\|^2 - \left\| S^{p_{Y_{t^\star}}} + \frac{Y_{t^\star}}{\chi_{t^\star}} \right\|^2$
    $I_2 \leftarrow \left\| S_z^{p_{Y_{t^\star}}} + \frac{Y_{t^\star}}{\chi_{t^\star}} \right\|^2 - \left\| S^{p_{Y_{t^\star}}} + \frac{Y_{t^\star}}{\chi_{t^\star}} \right\|^2$
    $\hat{I} \leftarrow T \frac{g_{t^\star}^2}{2} (I_1 - I_2)$ Eq. (18)
return $\hat{I}$

The empirical validation uses two different types of time series for which the TE is known. The first of these is given by a two-dimensional vector autoregressive process of order 1 which can be described as follows:

$$\begin{cases} x_t = b_x x_{t-1} + \lambda y_{t-1} + \varepsilon_t^x \\ y_t = b_y y_{t-1} + \varepsilon_t^y, \end{cases} \tag{19}$$

where both ε_t^x and ε_t^y are independent zero-mean Gaussian innovations with variances σ_x^2 and σ_y^2 respectively. As it can be seen in Eq. (19), y_t is independent of the past of x_t so the TE from X to Y is zero. Furthermore, note that x_t depends on the past of y_t so the TE is positive. A closed form for this expression can be found in Edinburgh et al. (2021). We refer to this process as **linear Gaussian system** in the figures.

The second kind of time series is a bivariate process whose realizations are generated according to the following scheme. Let $x_t \sim N(0, 1)$ and $z_t \sim N(0, 1)$ be independent, let $\rho \in (-1, 1)$, and construct y_t as follows:

$$y_t = \begin{cases} z_{t-1}, \ y_{t-1} < \lambda, \\ \rho x_{t-1} + \sqrt{1 - \rho^2} z_{t-1}, \ y_{t-1} \geq \lambda. \end{cases} \tag{20}$$

Thus, the bivariate system is given by $[x_t, y_t]$: we refer to this process as **joint system** in the figures. In this case, the TE from Y to X is null, but as shown by Zhang et al. (2019), the TE in the other direction is given by $-\frac{1}{2}(1-\Phi(\lambda)) \log(1-\rho^2)$, where $\Phi(\cdot)$ is the cumulative distribution function of a standard Gaussian random variable. It can be seen from the processes described above that in both cases the parameter λ controls the strength of the dependency measured by the TE between the components of the system.

5.1 TE estimation benchmark

Benchmarking. We consider four different tasks to evaluate the performance of the estimators. For all tasks, each reported result corresponds to the average of estimations over 5 seeds, where for every seed a new dataset is generated and the model is reinitialized and retrained from the ground up. Following the setup by Kornai et al. (2025), we use 10000 samples to estimate the transfer entropy in all tasks except for the sample size benchmark. Moreover, λ is fixed to 0 in the Gaussian system and to 0.5 in the joint system for the tasks in which λ is not varied, while the remaining parameters are kept consistent with those (Kornai et al., 2025). More experiments can be found in § C.

Sample size effect. We focus on computing the transfer entropy for varying sample sizes to analyze how the accuracy of the estimates improves as the number of observations increases. In this case, the different sample sizes considered for both systems are $T = 500, 1000, 5000, 10000$.

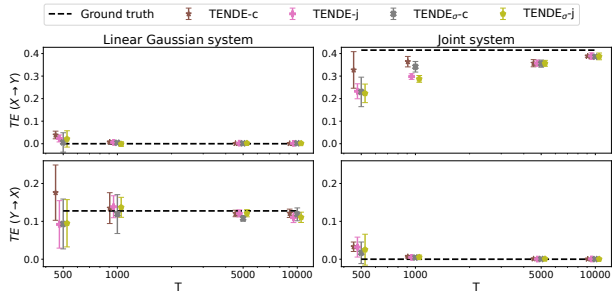

Figure 1: Transfer entropy estimation across sample sizes for linear Gaussian and joint systems.

Consistency. We examine a two-dimensional system where the parameter λ is varied, allowing us to study how changes in coupling strength affect the measured transfer entropy. For this matter, we simulate both systems using nine evenly distributed values of λ between 0 and 1.

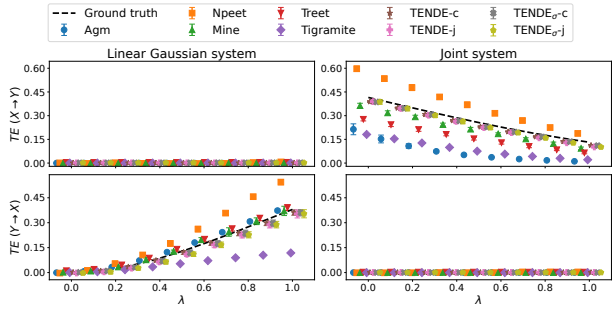

Figure 2: Transfer entropy estimation for varying coupling strength (λ).

Redundant stacking. We stack d redundant dimensions onto both x_t and y_t, but which do not contribute to the transfer entropy. More precisely, we consider a $2d$-dimensional time series $[\tilde{x}_t, \tilde{y}_t]$ with

$$\begin{cases} \tilde{x}_t = \left[x_t, \varepsilon^x_{t,1}, \dots \varepsilon^x_{t,d}\right] \\ \tilde{y}_t = \left[y_t, \varepsilon^y_{t,1}, \dots \varepsilon^y_{t,d}\right], \end{cases} \tag{21}$$

where the redundant dimensions $\left(\varepsilon^x_{t,i}, \varepsilon^x_{t,j}\right)$ are independent Gaussian white noise processes for $1 \leq i, j \leq d$, hence $\mathrm{TE}_{\tilde{X}\to\tilde{Y}}(k,\ell) = \mathrm{TE}_{X\to Y}(k,\ell)$. A proof of this fact is available in § B.1

Linear stacking. We consider a scenario in which d replicates of the processes x_t and y_t are stacked in such a way that dependence exists only between corresponding components, making the transfer entropy additive across dimensions. That is, the $2d$-dimensional

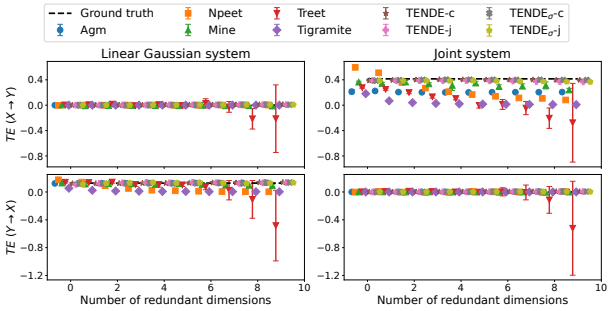

Figure 3: Transfer entropy estimation with added redundant (noise) dimensions.

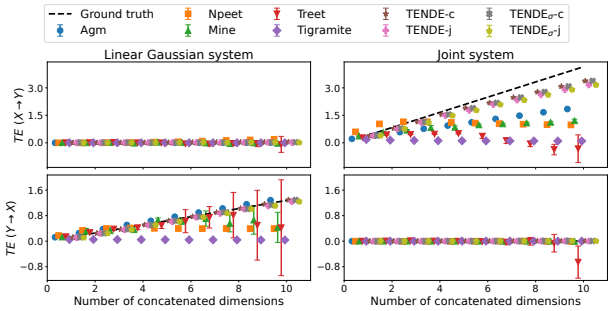

Figure 4: Transfer entropy estimation for linearly stacked systems where multiple independent process copies create additive transfer entropy.

time series $[\tilde{x}_t, \tilde{y}_t]$ is given by

$$\begin{cases} \tilde{x}_t = [x_{t,1}, \dots x_{t,d}] \\ \tilde{y}_t = [y_{t,1}, \dots y_{t,d}], \end{cases} \tag{22}$$

where both collections of processes $\{x_{t,i}\}$ and $\{y_{t,i}\}$ for $1 \leq i \leq d$ are independent replicates of x_t and y_t respectively, that is, $x_{t,i} \perp y_{t,j}$ for $i \neq j$ and $x_{t,i} \not\perp y_{t,j}$ if $i = j$, thus the transfer entropy between is given by $\mathrm{TE}_{\tilde{X}\to\tilde{Y}}(k,\ell) = \sum_{i=1}^{d} \mathrm{TE}_{X_i\to Y_i}(k,\ell)$. The details for this fact are provided in § B.2.

Discussion. The synthetic benchmark results demonstrate TENDE's superior performance across all evaluation scenarios, particularly in high-dimensional settings where traditional methods fail. In the sample size experiments (Figure 1), our estimators converge reliably to the ground truth as data increases. When varying the coupling strength (Figure 2), TENDE accurately captures the expected trends, unlike competing estimators that show instability. Under redundant stacking (Figure 3), our approach remains robust to irrelevant noise dimensions, maintaining stable estimates while others degrade sharply; notably, TREET exhibits large variance and produces negative estimates at higher dimensions, highlighting the instability of variational approaches in this

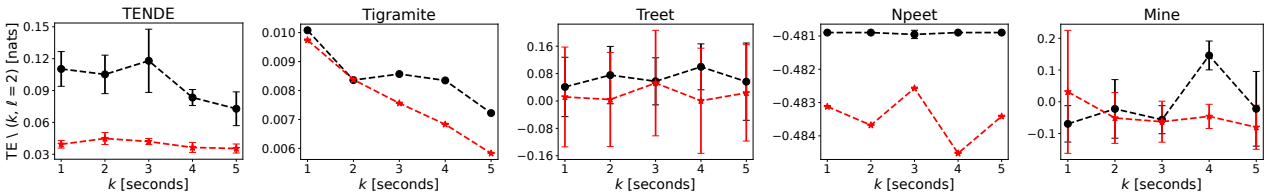

Figure 5: Transfer entropy $TE(k, \ell = 2)$ between breathing and heart signals as a function of lag k for each estimator. Black denotes the TE from the respiration force to the heart rate, whereas red denotes TE in the other direction. The reported error bars correspond to the standard deviations over 5 seeds.

regime. Tigramite consistently underestimates the transfer entropy, yielding near-zero values. Finally, in the linearly stacked setting (Figure 4), TENDE scales additively with the number of independent process copies, matching theoretical expectations, whereas both TREET and Tigramite fail to track the growing ground truth. These results highlight that the score-based framework naturally handles complex conditional distributions without restrictive assumptions, contrasting with k-nearest neighbor methods that suffer from the curse of dimensionality and variational approaches requiring exponentially large datasets. While AGM performs well under correct parametric assumptions, TENDE achieves comparable or superior performance without such prior knowledge, making it a more robust and practical estimator for real-world applications. Additional results at higher dimensions and with larger sample sizes are reported in § C.

6 REAL DATA ANALYSIS

The Santa Fe Time Series Competition Data Set B is a multivariate physiological dataset recorded from a patient in a sleep laboratory in Boston, Massachusetts (Rigney et al., 1993; Ichimaru and Moody, 1999). It comprises synchronized measurements of heart rate, chest (respiration) volume, and blood oxygen concentration, sampled at 2 Hz (every 0.5 seconds); see Figure 6.

To be consistent with previous works that analyze this dataset (e.g., Caţaron and Andonie (2018)), we only consider the chunk of the time series from index 2350 to index 3550.

The TE analysis on the Santa Fe dataset, shown in Figure 5, reveals consistently higher values from respiration force to heart rate than in the reverse direction, with magnitudes roughly two to three times larger across most of the examined lags. A decay in the transfer of information is observed when conditioning on more three seconds of past respiratory activity, while the reverse direction remains comparatively stable across lags. This asymmetry suggests that the identified directional coupling is robust to the specific lag choice rather than an artifact of delay structure, aligning with prior findings in physiological data (Schreiber, 2000; Kaiser and Schreiber, 2002; Luxembourg et al., 2025; Caţaron and Andonie, 2018). When compared against alternative estimators, also included in Figure 5, TENDE produces more stable and physiologically interpretable estimates. MINE and Npeet exhibit greater variability and deviations from expected trends. TREET recovers the correct directional asymmetry but with substantially larger error bars, while Tigramite yields estimates that are orders of magnitude smaller than those of all other methods. The declining TE values from respiration to heart rate at longer lags further indicate that extended cardiac history reduces the incremental predictive contribution of the breathing signal, although interpretation must remain cautious given the complexities of coupled physiological systems. Finally, a comparison with AGM was not performed, since its available implementation only supports transfer entropy estimation with a single lag, preventing inclusion under the longer conditioning on the past of the signals setting considered here.

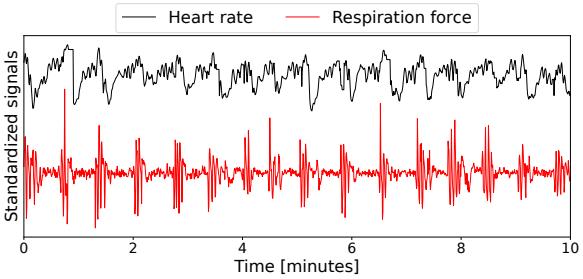

Figure 6: Sampled heartbeat and respiration force time series from the Santa Fe dataset shown over 10 minutes.

7 CONCLUSIONS

Quantifying directed information flow in time series remains a central problem in many applications, e.g.,

in neuroscience, finance, and complex systems analysis. In this paper, we introduced TENDE (Transfer Entropy Neural Diffusion Estimation), a novel approach that leverages score-based diffusion models for flexible and scalable estimation of transfer entropy via conditional mutual information with minimal assumptions. Experiments on synthetic benchmarks and real-world datasets show that TENDE achieves high accuracy and robustness, outperforming existing neural estimators and other competitors from the state-of-the-art. Looking ahead, we aim to extend TENDE to handle nonstationary dynamics and explore amortization across lags to improve efficiency in long time series. While TENDE inherits the computational cost of training diffusion models, it offers a principled and effective framework for transfer entropy estimation, paving the way for more reliable analysis of dependencies in complex dynamical systems.

References

L. A. Baccalá and K. Sameshima. Partial directed coherence: a new concept in neural structure determination. *Biological cybernetics*, 84(6):463–474, 2001.

T. Bedford and R. M. Cooke. Vines–a new graphical model for dependent random variables. *The Annals of statistics*, 30(4):1031–1068, 2002.

M. I. Belghazi, S. Rajeswar, A. Baratin, D. Hjelm, and A. Courville. MINE: Mutual information neural estimation, 2018.

M. Bounoua, G. Franzese, and P. Michiardi. S*omega* i: Score-based o-information estimation. *arXiv preprint arXiv:2402.05667*, 2024a.

M. Bounoua, G. Franzese, and P. Michiardi. Multimodal latent diffusion. *Entropy*, 26(4), 2024b.

N. A. Caserini and P. Pagnottoni. Effective transfer entropy to measure information flows in credit markets. *Statistical Methods & Applications*, 31(4):729–757, 2022.

A. Caţaron and R. Andonie. Transfer information energy: A quantitative indicator of information transfer between time series. *Entropy*, 20(5):323, 2018.

R. Cîrstian, J. Pilmeyer, A. Bernas, J. F. Jansen, M. Breeuwer, A. P. Aldenkamp, and S. Zinger. Objective biomarkers of depression: A study of granger causality and wavelet coherence in resting-state fmri. *Journal of Neuroimaging*, 33(3):404–414, 2023.

J.-F. Collet and F. Malrieu. Logarithmic Sobolev inequalities for inhomogeneous Markov semi-groups. *ESAIM: Probability and Statistics*, 12:492–504, 2008.

A. Derumigny and J.-D. Fermanian. On kendall's regression. *Journal of Multivariate Analysis*, 178: 104610, 2020.

T. Edinburgh, S. J. Eglen, and A. Ercole. Causality indices for bivariate time series data: A comparative review of performance. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 31(8), 2021.

A. B. El-Yaagoubi, S. Aslan, F. Gomawi, P. V. Redondo, S. Roy, M. S. Sultan, M. S. Talento, F. T. Tarrazona, H. Wu, K. W. Cooper, et al. Methods for brain connectivity analysis with applications to rat local field potential recordings. *Entropy*, 27(4): 328, 2025.

G. Franzese, M. Bounoua, and P. Michiardi. Minde: Mutual information neural diffusion estimation. *arXiv preprint arXiv:2310.09031*, 2023.

S. Frenzel and B. Pompe. Partial mutual information for coupling analysis of multivariate time series. *Physical review letters*, 99(20):204101, 2007.

W. Gao, S. Oh, and P. Viswanath. Demystifying fixed kk -nearest neighbor information estimators. *IEEE Transactions on Information Theory*, 64(8):5629–5661, 2018.

S. Garg, U. Gupta, Y. Chen, S. D. Gupta, Y. Adler, A. Schneider, and Y. Nevmyvaka. Estimating transfer entropy under long ranged dependencies. In *Uncertainty in Artificial Intelligence*, pages 685–695. PMLR, 2022.

I. Gijbels, M. Omelka, and N. Veraverbeke. Omnibus test for covariate effects in conditional copula models. *Journal of Multivariate Analysis*, 186:104804, 2021.

Z. Goldfeld and K. Greenewald. Sliced mutual information: A scalable measure of statistical dependence. In *Advances in Neural Information Processing Systems*, volume 34, pages 17567–17578, 2021.

Y. Gong and R. Huser. Asymmetric tail dependence modeling, with application to cryptocurrency market data. *The Annals of Applied Statistics*, 16(3): 1822–1847, 2022.

A. Hyvärinen and P. Dayan. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4), 2005.

Y. Ichimaru and G. Moody. Development of the polysomnographic database on cd-rom. *Psychiatry and clinical neurosciences*, 53(2):175–177, 1999.

A. Kaiser and T. Schreiber. Information transfer in continuous processes. *Physica D: Nonlinear Phenomena*, 166(1-2):43–62, 2002.

A. S. Kayser, F. T. Sun, and M. D'Esposito. A comparison of granger causality and coherency in fmri-

based analysis of the motor system. *Human brain mapping*, 30(11):3475–3494, 2009.

X. Kong, R. Brekelmans, and G. V. Steeg. Information-theoretic diffusion. In *ICLR*, 2023.

D. Kornai, R. Silva, and N. Nikolaou. Agm-te: Approximate generative model estimator of transfer entropy for causal discovery. *Proceedings of Machine Learning Research TBD*, 1:44, 2025.

L. Kozachenko. Sample estimate of the entropy of a random vector. *Probl. Pered. Inform.*, 23:9, 1987.

M. Lindner, R. Vicente, V. Priesemann, and M. Wibral. Trentool: A matlab open source toolbox to analyse information flow in time series data with transfer entropy. *BMC neuroscience*, 12(1):119, 2011.

O. Luxembourg, D. Tsur, and H. Permuter. Treet: Transfer entropy estimation via transformers. *IEEE Access*, 2025.

J. Ma and Z. Sun. Mutual information is copula entropy. *Tsinghua Science and Technology*, 16(1):51–54, 2011.

D. McAllester and K. Stratos. Formal limitations on the measurement of mutual information. In *International Conference on Artificial Intelligence and Statistics*, pages 875–884. PMLR, 2020.

T. Nagler. Simplified vine copula models: state of science and affairs. *Risk Sciences*, page 100022, 2025.

F. Parente and A. Colosimo. Modelling a multiplex brain network by local transfer entropy. *Scientific reports*, 11(1):15525, 2021.

A. J. Patton. A review of copula models for economic time series. *Journal of Multivariate Analysis*, 110: 4–18, 2012.

P. V. Redondo, R. Huser, and H. Ombao. Measuring information transfer between nodes in a brain network through spectral transfer entropy. *arXiv preprint arXiv:2303.06384*, 2023.

D. R. Rigney, A. L. Goldberger, W. C. Ocasio, Y. Ichimaru, G. B. Moody, and R. G. Mark. Multichannel physiological data: Description and analysis. In A. S. Weigend and N. A. Gershenfeld, editors, *Time Series Prediction: Forecasting the Future and Understanding the Past*, pages 105–129. Addison-Wesley, Reading, MA, USA, 1993.

J. Runge, P. Nowack, M. Kretschmer, S. Flaxman, and D. Sejdinovic. Detecting and quantifying causal associations in large nonlinear time series datasets. *Science Advances*, 5(11):eaau4996, 2019.

T. Schreiber. Measuring information transfer. *Physical review letters*, 85(2):461, 2000.

Y. Shalev, A. Painsky, and I. Ben-Gal. Neural joint entropy estimation. *IEEE Transactions on Neural Networks and Learning Systems*, 35(4):5488–5500, 2022.

Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

G. V. Steeg and A. Galstyan. Information-theoretic measures of influence based on content dynamics, 2013.

D. Tsur, Z. Aharoni, Z. Goldfeld, and H. Permuter. Neural estimation and optimization of directed information over continuous spaces. *IEEE Transactions on Information Theory*, 69(8):4777–4798, 2023a.

D. Tsur, Z. Goldfeld, and K. Greenewald. Max-sliced mutual information. In *Advances in Neural Information Processing Systems*, volume 36, pages 80338–80351, 2023b.

P. Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7): 1661–1674, 2011.

L. Wang, L. Wei, L. Jin, Y. Li, Y. Wei, W. He, L. Shi, Q. Sun, W. Li, Q. Li, et al. Different features of a metabolic connectivity map and the granger causality method in revealing directed dopamine pathways: A study based on integrated pet/mr imaging. *American Journal of Neuroradiology*, 43(12): 1770–1776, 2022.

Z. Wang, J. Liang, S. Shi, P. Zhai, and L. Zhang. Time-variant granger causality analysis for intuitive perception collision risk in driving scenario: an eeg study. *Frontiers in Neuroscience*, 19:1604751, 2025.

J. Zhang, O. Simeone, Z. Cvetkovic, E. Abela, and M. Richardson. Itene: Intrinsic transfer entropy neural estimator. *arXiv preprint arXiv:1912.07277*, 2019.

P. Zhao and L. Lai. Analysis of knn information estimators for smooth distributions. *IEEE Transactions on Information Theory*, 66(6):3798–3826, 2020.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. Yes.
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. No. In our case, the estimation relies on diffusion models whose implementation and training are tied to the choice of neural architecture, parameterization of the score function and other related training factors such as optimization schemes.
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. Yes, the code is available in the following GitHub repository: https://github.com/SimonPeterG/TENDE.

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. Yes.
   (b) Complete proofs of all theoretical results. Yes.
   (c) Clear explanations of any assumptions. Yes.

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). Yes, the code is available in the following GitHub repository: https://github.com/SimonPeterG/TENDE.
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). Yes.
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). Yes.
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). Yes.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. Yes.
   (b) The license information of the assets, if applicable. Not applicable.
   (c) New assets either in the supplemental material or as a URL, if applicable. Not applicable.
   (d) Information about consent from data providers/curators. Not applicable.
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. Not applicable.

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. Not applicable.
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. Not applicable.
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. Not applicable.

Appendix

A Detailed derivations

A.1 Entropy by using an auxiliary Gaussian random variable and its estimation

We will first focus on the derivation of Eq. (10) and how to use it as the means to estimate the entropy of a random variable.

Recall that X denotes a N_{x}-dimensional random variable with density p_X, and that φ_σ denotes the density of a N_{x}-dimensional centered Gaussian random variable with covariance $\sigma^2\mathbf{I}_{N_{\mathrm{x}}}$. Thus, the KL Divergence between p_X and φ_σ is given by:

$$\begin{aligned}
D_{\mathrm{KL}}\left[p_X \mid\mid \varphi_\sigma\right] &= \mathbb{E}_{x\sim p_X}\left[\log\left(\frac{p_X(x)}{\varphi_\sigma(x)}\right)\right] \\
&= \mathbb{E}_{x\sim p_X}\left[\log\left(p_X(x)\right)\right] - \mathbb{E}_{x\sim p_X}\left[\log\left(\varphi_\sigma(x)\right)\right] \\
&= \mathbb{E}_{x\sim p_X}\left[\log\left(p_X(x)\right)\right] - \mathbb{E}_{x\sim p_X}\left[-\frac{\|x\|^2}{2\sigma^2} - \frac{N_{\mathrm{x}}}{2}\log\left(2\pi\sigma^2\right)\right] \\
&= \mathbb{E}_{x\sim p_X}\left[\log\left(p_X(x)\right)\right] - \left(\mathbb{E}_{x\sim p_X}\left[-\frac{\|x\|^2}{2\sigma^2}\right] - \frac{N_{\mathrm{x}}}{2}\log\left(2\pi\sigma^2\right)\right).
\end{aligned}$$

Thus, rearranging the terms and noticing that $H(X) = -\mathbb{E}_{x\sim p_X}\left[\log\left(p_X(x)\right)\right]$ we obtain the desired equality, that is:

$$H(X) = \frac{N_{\mathrm{x}}}{2}\log\left(2\pi\sigma^2\right) + \mathbb{E}_{x\sim p_X}\left[\frac{\|x\|^2}{2\sigma^2}\right] - D_{\mathrm{KL}}\left[p_X \mid\mid \varphi_\sigma\right].$$

With this in mind, we can now use the estimator of the KL Divergence stated in Eq. (9) to estimate the entropy. Notice that there are two unknown densities involved in Eq. (9), therefore two parametric scores are required. However, that is not the case here since p_X is the only unknown, hence, only a single score network is required to estimate the KL Divergence between p_X and φ_σ. It is important to keep in mind that if we construct the following diffusion process,

$$\begin{cases} dX_t = f_t X_t dt + g_t dW_t \\ X_0 \sim \varphi_\sigma, \end{cases}$$

the score function associated with X_t is known and is given by $S^{\varphi_{\sigma X_t}}(x) = -\frac{x}{\chi_t}$, where $\chi_t = \left(k_t^2\sigma^2 + k_t^2\int_0^t \frac{g_s^2}{k_s^2}ds\right)$ with $k_t = \exp\left\{\int_0^t f_s ds\right\}$. Replacing q by φ_σ yields:

$$\begin{aligned}
D_{\mathrm{KL}}\left[p_X \mid\mid \varphi_\sigma\right] &= \int_0^T \frac{g_t^2}{2}\mathbb{E}_{x\sim p_{X_t}}\left[\left\|S^{p_{X_t}}(x) - S^{\varphi_{\sigma X_t}}(x)\right\|^2\right]dt + D_{KL}\left[p_{X_T}||\varphi_{\sigma_{X_T}}\right] \\
&= \int_0^T \frac{g_t^2}{2}\mathbb{E}_{x\sim p_{X_t}}\left[\left\|S^{p_{X_t}}(x) - S^{\varphi_{\sigma X_t}}(x)\right\|^2\right]dt + D_{\mathrm{KL}}\left[\varphi_1 \mid\mid \varphi_{\sqrt{\chi_T}}\right] \\
&= \int_0^T \frac{g_t^2}{2}\mathbb{E}_{x\sim p_{X_t}}\left[\left\|S^{p_{X_t}}(x) - S^{\varphi_{\sigma X_t}}(x)\right\|^2\right]dt + \frac{N_{\mathrm{x}}}{2}\left(\log(\chi_T) - 1 + \frac{1}{\chi_T}\right) \\
&\simeq e(p_X, \varphi_\sigma) + \frac{N_{\mathrm{x}}}{2}\left(\log(\chi_T) - 1 + \frac{1}{\chi_T}\right).
\end{aligned}$$

The first equality is simply Eq. (6); the second equality follows from the fact that using the variance preserving stochastic differential equation, $p_{X_T} \simeq \varphi_1$ for T large enough. Similarly, we have that when X_0 is sampled from

φ_σ the random variable $X_T \sim \varphi_{\sqrt{\chi_T}}$, thus $D_{KL}\left[p_{X_T}||\varphi_{\sigma_{X_T}}\right] = D_{\mathrm{KL}}\left[\varphi_1 \,||\, \varphi_{\sqrt{\chi_T}}\right]$. The third equality arises due to the fact that $D_{\mathrm{KL}}\left[\varphi_1 \,||\, \varphi_{\sqrt{\chi_T}}\right]$ is available in closed form, and the last equality is simply obtained by replacing the first term with its respective approximation. Finally, we have:

$$\begin{aligned}
H(X) &= \frac{N_{\mathrm{x}}}{2}\log\left(2\pi\sigma^2\right) + \mathbb{E}_{x\sim p_X}\left[\frac{\|x\|^2}{2\sigma^2}\right] - D_{\mathrm{KL}}\left[p_X \,||\, \varphi_\sigma\right] \\
&\simeq \frac{N_{\mathrm{x}}}{2}\log\left(2\pi\sigma^2\right) + \mathbb{E}_{x\sim p_X}\left[\frac{\|x\|^2}{2\sigma^2}\right] - \left[e(p_X,\varphi_\sigma) + \frac{N_{\mathrm{x}}}{2}\left(\log(\chi_T) - 1 + \frac{1}{\chi_T}\right)\right] \\
&= \frac{N_{\mathrm{x}}}{2}\log\left(2\pi\sigma^2\right) + \mathbb{E}_{x\sim p_X}\left[\frac{\|x\|^2}{2\sigma^2}\right] - e(p_X,\varphi_\sigma) - \frac{N_{\mathrm{x}}}{2}\left(\log(\chi_T) - 1 + \frac{1}{\chi_T}\right). \\
&= H(X;\sigma)
\end{aligned}$$

A.2 Derivation of TE estimators

A.2.1 TE as expected KL Divergence

Deriving the estimator proposed in Eq. (15) is a straightforward application of Eq. (12) and the fact that $e(\cdot,\cdot)$ is our estimator for KL Divergence (see § 4.1), thus we have:

$$\begin{aligned}
I(X,Y|Z) &= \mathbb{E}_{[y,z]\sim p_{Y,Z}}\left[D_{\mathrm{KL}}\left[p_{X_{y,z}} \,||\, p_{X_z}\right]\right] \\
&\simeq \mathbb{E}_{[y,z]\sim p_{Y,Z}}\left[e(p_{X_{y,z}}, p_{X_z})\right].
\end{aligned}$$

A.2.2 TE as difference of conditional entropies

Recall that $I(X;Y|Z) = \mathcal{H}(X|Z) - \mathcal{H}(X|Y,Z)$. Using Eq. (11) we have:

$$\begin{aligned}
\mathcal{H}(X|Y,Z) &\simeq \mathbb{E}_{[y,z]\sim p_{Y,Z}}\left[\frac{N_{\mathrm{x}}}{2}\log(2\pi\sigma^2) + \mathbb{E}_{x\sim p_{X_{y,z}}}\left[\frac{\|x\|^2}{2\sigma^2}\right] - e(p_{X_{y,z}},\varphi_\sigma) - \frac{N_{\mathrm{x}}}{2}\left(\log(\chi_T) - 1 + \frac{1}{\chi_T}\right)\right] \\
&= \frac{N_{\mathrm{x}}}{2}\log(2\pi\sigma^2) + \mathbb{E}_{x\sim p_X}\left[\frac{\|x\|^2}{2\sigma^2}\right] - \mathbb{E}_{[y,z]\sim p_{Y,Z}}\left[e(p_{X_{y,z}},\varphi_\sigma)\right] - \frac{N_{\mathrm{x}}}{2}\left(\log(\chi_T) - 1 + \frac{1}{\chi_T}\right).
\end{aligned}$$

In a similar fashion, it is possible to obtain the following:

$$\mathcal{H}(X|Z) \simeq \frac{N_{\mathrm{x}}}{2}\log(2\pi\sigma^2) + \mathbb{E}_{x\sim p_X}\left[\frac{\|x\|^2}{2\sigma^2}\right] - \mathbb{E}_{z\sim p_Z}\left[e(p_{X_z},\varphi_\sigma)\right] - \frac{N_{\mathrm{x}}}{2}\left(\log(\chi_T) - 1 + \frac{1}{\chi_T}\right).$$

Thus, it immediately follows that:

$$\begin{aligned}
I(X;Y|Z) &= \mathcal{H}(X|Z) - \mathcal{H}(X|Y,Z) \\
&\simeq \mathbb{E}_{[y,z]\sim p_{Y,Z}}\left[e(p_{X_{y,z}},\varphi_\sigma)\right] - \mathbb{E}_{z\sim p_Z}\left[e(p_{X_z},\varphi_\sigma)\right].
\end{aligned}$$

A.2.3 TE as difference of mutual informations

We leverage the representation of conditional mutual information as the difference of mutual informations in the case of the estimator proposed in Eq. (14), that is $I(X;Y|Z) = I(X;[Y,Z]) - I(X;Z)$. Furthermore we represent the mutual informations as the expectation over KL Divergencies as follows:

$$\begin{aligned}
I(X;Y|Z) &= I(X;[Y,Z]) - I(X;Z) \\
&= \mathbb{E}_{[y,z]\sim p_{Y,Z}}\left[D_{\mathrm{KL}}\left[p_{X_{y,z}} \,||\, p_X\right]\right] - \mathbb{E}_{z\sim p_Z}\left[D_{\mathrm{KL}}\left[p_{X_z} \,||\, p_X\right]\right] \\
&\simeq \mathbb{E}_{[y,z]\sim p_{Y,Z}}\left[e(p_{X_{y,z}}, p_X)\right] - \mathbb{E}_{z\sim p_Z}\left[e(p_{X_z}, p_X)\right].
\end{aligned}$$

A.3 Approximation error

We now discuss the quality of the approximation $e(p,q) \approx D_{KL}[p\|q]$ introduced in Eq. (9). Recall from Eq. (6) that the exact KL divergence decomposes as

$$D_{KL}[p\|q] = \underbrace{\int_0^T \frac{g_t^2}{2}\mathbb{E}_{x\sim p_{X_t}}\left[\left\|S^{p_{X_t}}(x) - S^{q_{X_t}}(x)\right\|^2\right] dt}_{\text{score difference term}} + \underbrace{D_{KL}[p_{X_T}\|q_{X_T}]}_{\text{terminal divergence}}.$$

Since $e(p, q)$ replaces the true scores with their parametric approximations in the first term, the estimation error is given by

$$e(p, q) - D_{KL}[p\|q] = d - D_{KL}[p_{X_T}\|q_{X_T}],$$

where, defining the score errors $\epsilon_t^p(x) := S^{p_{X_t}}(x; \theta^\star) - S^{p_{X_t}}(x)$ and $\epsilon_t^q(x) := S^{q_{X_t}}(x; \theta^\star) - S^{q_{X_t}}(x)$, the term d has the form (Franzese et al., 2023)

$$d = \int_0^T \frac{g_t^2}{2} \mathbb{E}_{x\sim p_{X_t}} \left[\|\epsilon_t^p(x) - \epsilon_t^q(x)\|^2 + 2 \left\langle S^{p_{X_t}}(x) - S^{q_{X_t}}(x),\ \epsilon_t^p(x) - \epsilon_t^q(x) \right\rangle \right] dt.$$

Two observations are worth noting. First, d is neither necessarily positive nor negative, so the estimator $e(p, q)$ is neither an upper nor a lower bound of the true KL divergence. This frees our approach from the pessimistic results of McAllester and Stratos (2020) that affect variational estimators. Second, common-mode score errors cancel: if $\epsilon_t^p(x) = \epsilon_t^q(x)$, then $d = 0$ regardless of the individual error magnitudes.

Regarding the terminal divergence $D_{KL}[p_{X_T}\|q_{X_T}]$, for the Variance Preserving schedule used in this work, the contraction properties of the diffusion semigroup (Collet and Malrieu, 2008) ensure that both p_{X_T} and q_{X_T} converge to the same stationary distribution as T grows, rendering this term numerically negligible for the values of T used in practice.

B Proofs

B.1 Invariance of the TE when stacking redundant dimensions

Recall that in § 5.1 we defined the redundant setting as the stacking of d redundant dimensions onto both x_t and y_t. More generally, we could consider two time series $\tilde{x}_t$ and $\tilde{y}_t$ defined as follows:

$$\begin{cases} \tilde{x}_t = \left[x_t, \varepsilon^x_{t,1}, \dots \varepsilon^x_{t,d_x}\right] \\ \tilde{y}_t = \left[y_t, \varepsilon^y_{t,1}, \dots \varepsilon^y_{t,d_y}\right]. \end{cases}$$

The redundant dimensions $\varepsilon^x_{t,i}$ and $\varepsilon^y_{t,j}$ are taken to be mutually independent collections. In particular, for all t, t', i, i', j, j' we have $\varepsilon^x_{t,i} \perp\!\!\!\perp \varepsilon^x_{t',i'}$, $\varepsilon^y_{t,j} \perp\!\!\!\perp \varepsilon^y_{t',j'}$, and $\varepsilon^x_{t,i} \perp\!\!\!\perp \varepsilon^y_{t',j'}$. Moreover, each of these redundant components is independent of the original processes, i.e., $\{\varepsilon^x_{t,i}\}_{t,i} \perp\!\!\!\perp (x_t, y_t)$ and $\{\varepsilon^y_{t,j}\}_{t,j} \perp\!\!\!\perp (x_t, y_t)$. To avoid clutter, we drop the subscripts on the distribution functions as well as the distribution functions in the expectations. That being said, let $k, \ell \in \mathbb{N}$ be the lags and construct $\tilde{\mathbf{x}}_{t-k}$ and $\tilde{\mathbf{y}}_{t-\ell}$ as defined in § 2.2. Also, define $\varepsilon^x_t = \left[\varepsilon^x_{t,1}, \dots \varepsilon^x_{t,d_x}\right]$, and define ε^y_t similarly. First, consider the distribution of $\tilde{y}_t$ and $\tilde{\mathbf{x}}_{t-k}$ conditioned on $\tilde{\mathbf{y}}_{t-\ell}$. We can see that:

$$\begin{aligned} p\left(\tilde{y}_t, \tilde{\mathbf{x}}_{t-k} | \tilde{\mathbf{y}}_{t-\ell}\right) &= \frac{p\left(\tilde{y}_t, \tilde{\mathbf{x}}_{t-k}, \tilde{\mathbf{y}}_{t-\ell}\right)}{p\left(\tilde{\mathbf{y}}_{t-\ell}\right)} \\ &= \frac{p\left(y_t, \mathbf{x}_{t-k}, \mathbf{y}_{t-\ell}, \varepsilon^y_t, \varepsilon^x_{t-k}, \varepsilon^y_{t-\ell}\right)}{p\left(\mathbf{y}_{t-\ell}, \varepsilon^y_{t-\ell}\right)} \\ &= \frac{p\left(y_t, \mathbf{x}_{t-k}, \mathbf{y}_{t-\ell}\right) p\left(\varepsilon^y_t\right) p\left(\varepsilon^x_{t-k}\right) p\left(\varepsilon^y_{t-\ell}\right)}{p\left(\mathbf{y}_{t-\ell}\right) p\left(\varepsilon^y_{t-\ell}\right)} \\ &= p\left(y_t, \mathbf{x}_{t-k} | \mathbf{y}_{t-\ell}\right) p\left(\varepsilon^y_t\right) p\left(\varepsilon^x_{t-k}\right), \end{aligned}$$

where the third equality comes from the construction of the system $[\tilde{x}_t, \tilde{y}_t]$ and the other equalities are immediate to deduce. Using similar arguments, it is possible to show that $p\left(\tilde{\mathbf{x}}_{t-k} | \tilde{\mathbf{y}}_{t-\ell}\right) = p\left(\mathbf{x}_{t-k} | \mathbf{y}_{t-\ell}\right) p\left(\varepsilon^x_{t-k}\right)$ and $p\left(\tilde{y}_t | \tilde{\mathbf{y}}_{t-\ell}\right) = p\left(y_t | \mathbf{y}_{t-\ell}\right) p\left(\varepsilon^y_t\right)$, thus

$$\frac{p\left(\tilde{y}_t, \tilde{\mathbf{x}}_{t-k} | \tilde{\mathbf{y}}_{t-\ell}\right)}{p\left(\tilde{\mathbf{x}}_{t-k} | \tilde{\mathbf{y}}_{t-\ell}\right) p\left(\tilde{y}_t | \tilde{\mathbf{y}}_{t-\ell}\right)} = \frac{p\left(y_t, \mathbf{x}_{t-k} | \mathbf{y}_{t-\ell}\right)}{p\left(\mathbf{x}_{t-k} | \mathbf{y}_{t-\ell}\right) p\left(y_t | \mathbf{y}_{t-\ell}\right)}. \tag{23}$$

Finally, consider the transfer entropy from $\tilde{x}$ to $\tilde{y}$

$$\begin{aligned} \mathrm{TE}_{\tilde{X} \to \tilde{Y}}(k, \ell) &= \mathbb{E}_{\tilde{\mathbf{y}}_{t-k}}\left[D_{\mathrm{KL}}\left[p\left(\tilde{y}_t, \tilde{\mathbf{x}}_{t-k} | \tilde{\mathbf{y}}_{t-\ell}\right) \;||\; p\left(\tilde{\mathbf{x}}_{t-k} | \tilde{\mathbf{y}}_{t-\ell}\right) p\left(\tilde{y}_t | \tilde{\mathbf{y}}_{t-\ell}\right)\right]\right] \\ &= \mathbb{E}_{\tilde{y}_t, \tilde{\mathbf{x}}_{t-k}, \tilde{\mathbf{y}}_{t-\ell}}\left[\log\left(\frac{p\left(\tilde{y}_t, \tilde{\mathbf{x}}_{t-k} | \tilde{\mathbf{y}}_{t-\ell}\right)}{p\left(\tilde{\mathbf{x}}_{t-k} | \tilde{\mathbf{y}}_{t-\ell}\right) p\left(\tilde{y}_t | \tilde{\mathbf{y}}_{t-\ell}\right)}\right)\right] \\ &= \mathbb{E}_{\tilde{y}_t, \tilde{\mathbf{x}}_{t-k}, \tilde{\mathbf{y}}_{t-\ell}}\left[\log\left(\frac{p\left(y_t, \mathbf{x}_{t-k} | \mathbf{y}_{t-\ell}\right)}{p\left(\mathbf{x}_{t-k} | \mathbf{y}_{t-\ell}\right) p\left(y_t | \mathbf{y}_{t-\ell}\right)}\right)\right] \\ &= \mathbb{E}_{y_t, \mathbf{x}_{t-k}, \mathbf{y}_{t-\ell}}\left[\log\left(\frac{p\left(y_t, \mathbf{x}_{t-k} | \mathbf{y}_{t-\ell}\right)}{p\left(\mathbf{x}_{t-k} | \mathbf{y}_{t-\ell}\right) p\left(y_t | \mathbf{y}_{t-\ell}\right)}\right)\right] \\ &= \mathrm{TE}_{X \to Y}(k, \ell). \end{aligned}$$

Where the first two equalities follow from the definition of TE, the third equality is consequence of Eq. (23), furthermore, the forth equality follows from the fact that the expression at hand does not depend on the redundant dimensions anymore. The last equality follows from the definition of TE.

The proof in the other direction is identical.

B.2 Additivity of the TE when independent components are stacked

in § 5.1 we defined the stacking setting as stacking of d independent replicates of the processes x_t and y_t in such a way that dependence exists only between corresponding components. More generally consider $\{x_{t,i}\}$ and $\{y_{t,i}\}$. The components $x_{t,i}$ and $x_{t',j}$ are assumed to be independent for all t, t', i, j, and analogously $y_{t,i}$ and $y_{t',j}$ are independent for all indices. The only dependence between the two processes arises when the second sub-index coincides, that is, $x_{t,i}$ and $y_{t',i}$ may be dependent, while $x_{t,i}$ and $y_{t',j}$ are independent for $i \neq j$. With these assumptions, we construct the series $\tilde{x}_t$ and $\tilde{y}_t$ as:

$$\begin{cases} \tilde{x}_t = [x_{t,1}, \dots x_{t,d}] \\ \tilde{y}_t = [y_{t,1}, \dots y_{t,d}] \,, \end{cases}$$

As in § B.1, we avoid cluttering the notation by dropping the subscripts on the distribution functions and the distribution functions in the expectations. Similarly, let $k, \ell \in \mathbb{N}$ be the lags and construct $\tilde{\mathbf{x}}_{t-k}$ and $\tilde{\mathbf{y}}_{t-\ell}$ as defined in § 2.2. First, consider the distribution of $\tilde{y}_t$ and $\tilde{\mathbf{x}}_{t-k}$ conditioned on $\tilde{\mathbf{y}}_{t-\ell}$, thus we can see that

$$\begin{aligned} p\left(\tilde{y}_t, \tilde{\mathbf{x}}_{t-k} | \tilde{\mathbf{y}}_{t-\ell}\right) &= \frac{p\left(\tilde{y}_t, \tilde{\mathbf{x}}_{t-k}, \tilde{\mathbf{y}}_{t-\ell}\right)}{p\left(\tilde{\mathbf{y}}_{t-\ell}\right)} \\ &= \frac{p\left(y_{t,1}, \mathbf{x}_{t-k,1}, \mathbf{y}_{t-\ell,1}, \dots, y_{t,d}, \mathbf{x}_{t-k,d}, \mathbf{y}_{t-\ell,d}\right)}{p\left(\mathbf{y}_{t-\ell,1}, \dots, \mathbf{y}_{t-\ell,1}\right)} \\ &= \frac{\prod_{j=1}^{d} p\left(y_{t,j}, \mathbf{x}_{t-k,j}, \mathbf{y}_{t-\ell,j}\right)}{\prod_{j=1}^{d} p\left(\mathbf{y}_{t-\ell,j}\right)} \\ &= \prod_{j=1}^{d} p\left(y_{t,j}, \mathbf{x}_{t-k,j} | \mathbf{y}_{t-\ell,j}\right). \end{aligned}$$

The first and second equalities are immediate and the third one arises from the design of the system; the forth equality is immediate as well. Using the same arguments, it is possible to obtain similar decompositions for the other quantities of interest, namely, $p\left(\tilde{\mathbf{x}}_{t-k} | \tilde{\mathbf{y}}_{t-\ell}\right)$ and $p\left(\tilde{y}_t | \tilde{\mathbf{y}}_{t-\ell}\right)$. That is, $p\left(\tilde{\mathbf{x}}_{t-k} | \tilde{\mathbf{y}}_{t-\ell}\right) = \prod_{j=1}^{d} p\left(\mathbf{x}_{t-k,j} | \mathbf{y}_{t-\ell,j}\right)$ and $p\left(\tilde{y}_t | \tilde{\mathbf{y}}_{t-\ell}\right) = \prod_{j=1}^{d} p\left(y_{t,j} | \mathbf{y}_{t-\ell,j}\right)$, hence

$$\frac{p\left(\tilde{y}_t, \tilde{\mathbf{x}}_{t-k} | \tilde{\mathbf{y}}_{t-\ell}\right)}{p\left(\tilde{\mathbf{x}}_{t-k} | \tilde{\mathbf{y}}_{t-\ell}\right) p\left(\tilde{y}_t | \tilde{\mathbf{y}}_{t-\ell}\right)} = \prod_{j=1}^{d} \frac{p\left(y_{t,j}, \mathbf{x}_{t-k,j} | \mathbf{y}_{t-\ell,j}\right)}{p\left(\mathbf{x}_{t-k,j} | \mathbf{y}_{t-\ell,j}\right) p\left(y_{t,j} | \mathbf{y}_{t-\ell,j}\right)}. \tag{24}$$

Finally, consider the transfer entropy from $\tilde{x}$ to $\tilde{y}$

$$\begin{aligned} \mathrm{TE}_{\tilde{X} \to \tilde{Y}}(k, \ell) &= \mathbb{E}_{\tilde{\mathbf{y}}_{t-k}}\left[D_{\mathrm{KL}}\left[p\left(\tilde{y}_t, \tilde{\mathbf{x}}_{t-k} | \tilde{\mathbf{y}}_{t-\ell}\right) \;||\; p\left(\tilde{\mathbf{x}}_{t-k} | \tilde{\mathbf{y}}_{t-\ell}\right) p\left(\tilde{y}_t | \tilde{\mathbf{y}}_{t-\ell}\right)\right]\right] \\ &= \mathbb{E}_{\tilde{y}_t, \tilde{\mathbf{x}}_{t-k}, \tilde{\mathbf{y}}_{t-\ell}}\left[\log\left(\frac{p\left(\tilde{y}_t, \tilde{\mathbf{x}}_{t-k} | \tilde{\mathbf{y}}_{t-\ell}\right)}{p\left(\tilde{\mathbf{x}}_{t-k} | \tilde{\mathbf{y}}_{t-\ell}\right) p\left(\tilde{y}_t | \tilde{\mathbf{y}}_{t-\ell}\right)}\right)\right] \\ &= \mathbb{E}_{\tilde{y}_t, \tilde{\mathbf{x}}_{t-k}, \tilde{\mathbf{y}}_{t-\ell}}\left[\log\left(\prod_{j=1}^{d} \frac{p\left(y_{t,j}, \mathbf{x}_{t-k,j} | \mathbf{y}_{t-\ell,j}\right)}{p\left(\mathbf{x}_{t-k,j} | \mathbf{y}_{t-\ell,j}\right) p\left(y_{t,j} | \mathbf{y}_{t-\ell,j}\right)}\right)\right] \\ &= \sum_{j=1}^{d} \mathbb{E}_{\tilde{y}_t, \tilde{\mathbf{x}}_{t-k}, \tilde{\mathbf{y}}_{t-\ell}}\left[\log\left(\frac{p\left(y_{t,j}, \mathbf{x}_{t-k,j} | \mathbf{y}_{t-\ell,j}\right)}{p\left(\mathbf{x}_{t-k,j} | \mathbf{y}_{t-\ell,j}\right) p\left(y_{t,j} | \mathbf{y}_{t-\ell,j}\right)}\right)\right] \\ &= \sum_{j=1}^{d} \mathbb{E}_{y_{t,j}, \mathbf{x}_{t-k,j}, \mathbf{y}_{t-\ell,j}}\left[\log\left(\frac{p\left(y_{t,j}, \mathbf{x}_{t-k,j} | \mathbf{y}_{t-\ell,j}\right)}{p\left(\mathbf{x}_{t-k,j} | \mathbf{y}_{t-\ell,j}\right) p\left(y_{t,j} | \mathbf{y}_{t-\ell,j}\right)}\right)\right] \\ &= \sum_{j=1}^{d} \mathrm{TE}_{X_j \to Y_j}(k, \ell). \end{aligned}$$

Here the first two equalities follow from the definition of TE, and the third equality is consequence of Eq. (24). The forth equality is immediate, and the fifth equality follows from the fact that the expression inside the sum only depends on the $j-$th process. Finally, the last equality follows from the definition of TE.

The proof in the other direction is identical.

C Further details on the synthetic benchmark and additional experiments

C.1 Details on the experimental benchmark

All the stochastic systems analyzed in this study were simulated using the publicly available code at the following link[2] provided by Kornai et al. (2025), ensuring consistency with the original experimental setup. The implementations of the NPEET[3] (Steeg and Galstyan, 2013), AGM[4] (Kornai et al., 2025), TREET[5] (Luxembourg et al., 2025), and Tigramite (Runge et al., 2019)[6] were used with their default settings. Furthermore, the MINE-based transfer entropy estimator was implemented by leveraging the formulation of transfer entropy as the difference between two mutual information terms (see Eq. (14)), which allows for the application of neural estimation techniques originally developed for mutual information. In this case, the implementation was obtained using the Benchmarking Mutual Information package[7]. The implementation of TENDE was based on the publicly available code for MINDE[8], adapting it to the transfer entropy estimation framework. For the TENDE variants that include σ as a hyperparameter, we set $\sigma = 1$, following the configuration adopted in Franzese et al. (2023), where this value was shown to yield stable and reliable performance across a variety of stochastic systems. Furthermore, as in Franzese et al. (2023), importance sampling was employed during the estimation of transfer entropy. Finally, for all models, the default hyperparameters provided in their original implementations were used during training to ensure fair and reproducible comparisons. Tigramite was excluded from the stacking benchmarks beyond 10 dimensions due to scalability constraints. For the 70-dimensional benchmarks with $T = 50000$, the number of training epochs for TREET was reduced due to numerical instabilities (NaN losses) encountered under the default configuration.

C.2 Beyond Gaussian benchmarks

In this section, we evaluate TENDE and the competitors we considered in § 5 across more challenging distributions. MI-invariant transformations are applied to the data to construct such settings. Since TE can be written in terms of MI, the invariance of MI implies invariance of TE, that is, applying MI-invariant transformations to the data leaves the ground truth value of the TE unchanged.

[2] TE_datasim
[3] NPEET
[4] AGM_TE
[5] TREET
[6] Tigramite
[7] Benchmarking Mutual Information
[8] MINDE

C.2.1 Half cube

Inspired by the work of Franzese et al. (2023) and Bounoua et al. (2024a), we consider the MI-invariant transformation defined as $x \mapsto x\sqrt{|x|}$.

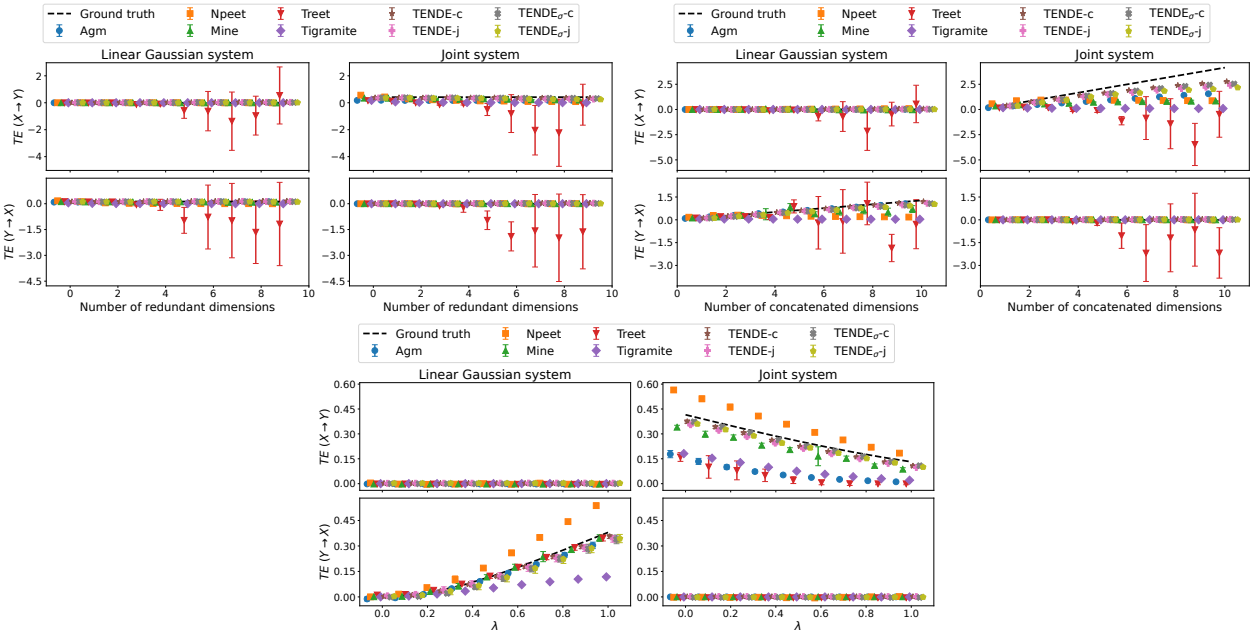

Figure 7: Estimated transfer entropy for the linear Gaussian and joint systems under redundant and linear stacking (top) and varying coupling strength λ (bottom). Both systems are modified using the half-cube mapping.

Across all configurations, the TENDE estimators continue to align closely with the analytical ground truth and exhibit consistent behavior across different regimes. In the redundant stacking setting (top-left), where independent noise dimensions are added, TENDE maintains stable estimates across varying numbers of redundant dimensions. In contrast, Treet produces negative estimates with large variance at higher dimensions, and Tigramite consistently underestimates the transfer entropy. In the linear stacking scenario (top-right), TENDE accurately captures the expected linear trend, whereas alternative estimators tend to underestimate the magnitude of transfer entropy and show noticeable bias as dimensionality grows; Treet again exhibits high instability. For the simple coupling system (bottom), TENDE maintains close agreement with the ground truth, while Npeet and Tigramite deviate at higher coupling values, and Treet shows substantial variability across the range of λ.

C.2.2 CDF

Following again Franzese et al. (2023) and Bounoua et al. (2024a), the second MI-invariant transformation we consider is $x \mapsto \Phi^{-1}(x)$, where $\Phi(\cdot)$ denotes the cumulative distribution function (CDF) of a standard Gaussian random variable, mapping all the data to the interval $[0, 1]$.

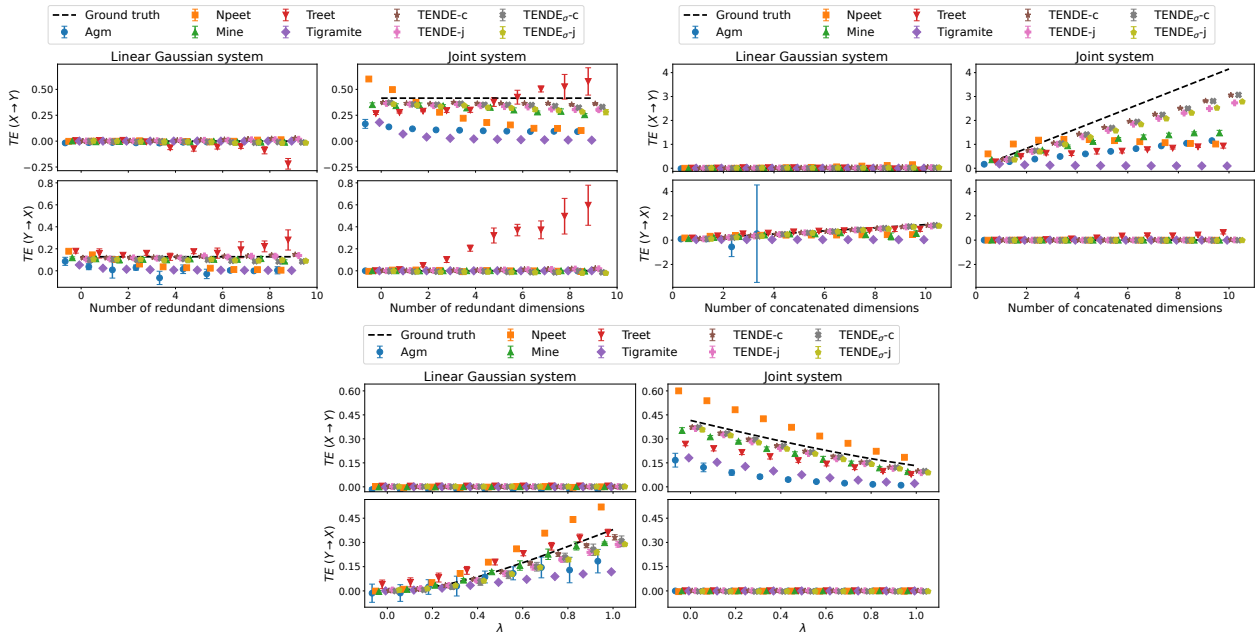

Figure 8: Estimated transfer entropy for the linear Gaussian and joint systems under redundant and linear stacking (top) and varying coupling strength λ (bottom). Both systems are modified using the CDF mapping.

Across all configurations, the TENDE estimators continue to align closely with the analytical ground truth, confirming their robustness under the CDF transformation. In the redundant stacking scenario (top-left), TENDE correctly maintains stable estimates across varying numbers of redundant dimensions, while Treet exhibits large negative deviations with increasing variance and Tigramite yields near-zero estimates. In the linear stacking setting (top-right), where transfer entropy should increase linearly with the number of informative dimensions, TENDE maintains accurate scaling, whereas Treet collapses at higher dimensions and the remaining alternative methods consistently underestimate. For the simple coupling system (bottom), TENDE follows the expected monotonic trend with λ, closely matching the ground truth, while Npeet and Tigramite show noticeable deviations at stronger coupling values.

C.3 Results at higher dimensions

We evaluate TENDE and the baseline estimators on higher-dimensional versions of the benchmarks described in § 5.1. In these experiments, the time series length is $T = 10000$ and both systems use 35 redundant or concatenated dimensions, resulting in 70-dimensional processes.

C.3.1 Redundant stacking

Table 1: Estimated transfer entropy (mean ± standard deviation) versus ground truth for the 70-dimensional redundant stacking benchmark (linear Gaussian system, $T = 10000$). Results are sorted by proximity to the ground truth within each direction.

Direction	Method	Estimated TE ± Std	Ground Truth
$X \to Y$	Mine	0.00 ± 0.02	0.0000
	TENDE$_\sigma$-j	0.05 ± 0.00	0.0000
	TENDE-j	0.05 ± 0.00	0.0000
	TENDE$_\sigma$-c	0.06 ± 0.01	0.0000
	TENDE-c	0.07 ± 0.00	0.0000
	Agm	0.12 ± 0.00	0.0000
	Npeet	-0.00 ± 0.02	0.0000
	Treet	-3.02 ± 2.59	0.0000
$Y \to X$	Mine	0.06 ± 0.04	0.1276
	TENDE$_\sigma$-j	0.20 ± 0.02	0.1276
	TENDE-j	0.21 ± 0.02	0.1276
	TENDE$_\sigma$-c	0.23 ± 0.02	0.1276
	Agm	0.24 ± 0.01	0.1276
	TENDE-c	0.25 ± 0.01	0.1276
	Npeet	0.00 ± 0.01	0.1276
	Treet	-1.53 ± 0.86	0.1276

Table 2: Estimated transfer entropy (mean ± standard deviation) versus ground truth for the 70-dimensional redundant stacking benchmark (joint system, $T = 10000$). Results are sorted by proximity to the ground truth within each direction.

Direction	Method	Estimated TE ± Std	Ground Truth
$X \to Y$	TENDE-c	0.36 ± 0.04	0.4152
	TENDE$_\sigma$-c	0.36 ± 0.05	0.4152
	Agm	0.31 ± 0.00	0.4152
	TENDE-j	0.20 ± 0.07	0.4152
	TENDE$_\sigma$-j	0.20 ± 0.08	0.4152
	Mine	0.15 ± 0.03	0.4152
	Npeet	0.02 ± 0.01	0.4152
	Treet	-3.52 ± 1.14	0.4152
$Y \to X$	TENDE$_\sigma$-j	0.04 ± 0.00	0.0000
	TENDE-j	0.05 ± 0.00	0.0000
	TENDE$_\sigma$-c	0.05 ± 0.00	0.0000
	TENDE-c	0.05 ± 0.00	0.0000
	Agm	0.11 ± 0.01	0.0000
	Npeet	-0.00 ± 0.01	0.0000
	Mine	-0.02 ± 0.02	0.0000
	Treet	-2.60 ± 1.57	0.0000

In the redundant stacking setting at 70 dimensions, the TENDE variants consistently rank among the top estimators in both systems. For the linear Gaussian system (Table 1), all methods correctly identify the null transfer entropy in the $X \to Y$ direction, while in the $Y \to X$ direction, TENDE variants provide estimates closest to the ground truth alongside Agm. For the joint system (Table 2), TENDE-c and TENDE$_\sigma$-c achieve the best approximation of the non-zero transfer entropy in the $X \to Y$ direction, and all TENDE variants remain close to zero in the null $Y \to X$ direction. Across both systems, Npeet fails to detect the non-zero transfer entropy, Treet produces negative estimates with high variance, and Mine underestimates substantially. These

results confirm that the score-based framework remains robust to irrelevant noise dimensions even when the score network must process over 100 input variables.

C.3.2 Linear stacking

Table 3: Estimated transfer entropy (mean ± standard deviation) versus ground truth for the 70-dimensional linear stacking benchmark (linear Gaussian system, $T = 10000$). Results are sorted by proximity to the ground truth within each direction.

Direction	Method	Estimated TE ± Std	Ground Truth
$X \to Y$	Agm	0.13 ± 0.00	0.0000
	TENDE$_\sigma$-j	0.20 ± 0.01	0.0000
	TENDE-j	0.21 ± 0.01	0.0000
	TENDE$_\sigma$-c	0.28 ± 0.01	0.0000
	TENDE-c	0.39 ± 0.03	0.0000
	Npeet	0.52 ± 0.01	0.0000
	Mine	-0.05 ± 0.05	0.0000
	Treet	-0.69 ± 2.32	0.0000
$Y \to X$	Agm	4.45 ± 0.01	4.4660
	TENDE$_\sigma$-c	4.52 ± 0.06	4.4660
	TENDE-j	4.35 ± 0.06	4.4660
	TENDE$_\sigma$-j	4.34 ± 0.06	4.4660
	TENDE-c	4.86 ± 0.02	4.4660
	Mine	0.65 ± 0.38	4.4660
	Npeet	0.14 ± 0.00	4.4660
	Treet	-1.80 ± 1.79	4.4660

Table 4: Estimated transfer entropy (mean ± standard deviation) versus ground truth for the 70-dimensional linear stacking benchmark (joint system, $T = 10000$). Results are sorted by proximity to the ground truth within each direction.

Direction	Method	Estimated TE ± Std	Ground Truth
$X \to Y$	TENDE-c	6.94 ± 0.80	14.5314
	TENDE$_\sigma$-c	6.86 ± 0.79	14.5314
	Agm	6.79 ± 0.02	14.5314
	TENDE$_\sigma$-j	4.67 ± 0.23	14.5314
	TENDE-j	4.65 ± 0.22	14.5314
	Mine	0.92 ± 0.07	14.5314
	Npeet	0.81 ± 0.01	14.5314
	Treet	-0.63 ± 1.42	14.5314
$Y \to X$	Npeet	0.01 ± 0.01	0.0000
	TENDE$_\sigma$-j	0.04 ± 0.00	0.0000
	TENDE-j	0.04 ± 0.00	0.0000
	TENDE$_\sigma$-c	0.05 ± 0.01	0.0000
	TENDE-c	0.05 ± 0.01	0.0000
	Agm	0.11 ± 0.00	0.0000
	Mine	-0.01 ± 0.01	0.0000
	Treet	-1.10 ± 1.51	0.0000

In the linear stacking setting at 70 dimensions, the transfer entropy grows additively with the number of independent process copies, resulting in large ground truth values that are particularly challenging to estimate. For the linear Gaussian system (Table 3), Agm achieves the closest estimate in the $Y \to X$ direction, followed closely by the TENDE variants, all of which recover the ground truth of 4.47 nats within a margin of 0.15 nats. In the joint system (Table 4), the ground truth of 14.53 nats proves challenging for all methods; nevertheless, TENDE-c and TENDE$_\sigma$-c achieve the best approximations at approximately 6.9 nats, outperforming Agm (6.8) and substantially outperforming Mine, Npeet, and Treet, which all remain below 1 nat. In both systems, all TENDE variants correctly identify the null direction, and Treet consistently produces negative estimates with high variance. These results suggest that while the score-based framework scales better than competing approaches, very high-dimensional stacking settings with large ground truth values remain challenging and benefit from increased

sample sizes, as explored in Table 5. Increasing the sample size from $T = 10000$ to $T = 50000$ reveals a striking

Table 5: Effect of increasing sample size on transfer entropy estimation for the 70-dimensional linear stacking benchmark (joint system). Results compare $T = 10000$ and $T = 50000$ observations, sorted by proximity to the ground truth at $T = 50000$.

Direction	Method	Ground Truth	$T = 10000$	$T = 50000$
$X \to Y$	TENDE-c	14.5314	6.94 ± 0.80	10.86 ± 0.10
	TENDE$_\sigma$-c	14.5314	6.86 ± 0.79	10.83 ± 0.09
	TENDE-j	14.5314	4.65 ± 0.22	10.35 ± 0.15
	TENDE$_\sigma$-j	14.5314	4.67 ± 0.23	10.33 ± 0.15
	Agm	14.5314	6.79 ± 0.02	6.50 ± 0.01
	Mine	14.5314	0.92 ± 0.07	1.24 ± 0.08
	Npeet	14.5314	0.81 ± 0.01	1.00 ± 0.00
	Treet	14.5314	-0.63 ± 1.42	-1.08 ± 3.16
$Y \to X$	TENDE$_\sigma$-c	0.0000	0.05 ± 0.01	0.01 ± 0.00
	TENDE-c	0.0000	0.05 ± 0.01	0.01 ± 0.00
	TENDE-j	0.0000	0.04 ± 0.00	0.01 ± 0.00
	TENDE$_\sigma$-j	0.0000	0.04 ± 0.00	0.01 ± 0.00
	Agm	0.0000	0.11 ± 0.00	0.02 ± 0.00
	Mine	0.0000	-0.01 ± 0.01	-0.00 ± 0.00
	Npeet	0.0000	0.01 ± 0.01	-0.00 ± 0.00
	Treet	0.0000	-1.10 ± 1.51	-1.49 ± 0.86

difference in how the estimators leverage additional data. In the $X \to Y$ direction, where the ground truth is 14.53 nats, all four TENDE variants improve dramatically: TENDE-c rises from 6.94 to 10.86 nats, and even the joint variants more than double their estimates, while simultaneously reducing their standard deviations by an order of magnitude. In contrast, Agm shows no improvement (in fact slightly decreasing from 6.79 to 6.50), and Mine and Npeet remain below 1.3 nats at both sample sizes. Treet worsens, producing more negative estimates with higher variance. In the null $Y \to X$ direction, all TENDE variants move closer to zero with more data, confirming that the improvements in the non-null direction are not artifacts of general overestimation. These results demonstrate that TENDE is uniquely capable of leveraging larger datasets in high-dimensional regimes, and suggest that with sufficient data the remaining gap to the ground truth can be further reduced.

D Implementation details

Unique denoising network. For the implementation of TENDE, we adopt the Variance Preserving Stochastic Differential Equation framework (Song et al., 2020). The latter perturbs the data using an SDE parameterized by a drift f_t and a diffusion coefficient g_t. Following Bounoua et al. (2024b), we amortize the learning of all required parametric scores by using a single denoising score network.

Training. Training is carried out through a randomized procedure. At each step, one of the possible encodings, which represents one of the score denoising score functions required for the computation of TE (joint, conditional, or marginal), is chosen. These denoising score functions are learned by the unique score network following the procedure described above. In total, estimating TE requires estimating either two or three score functions, which is something we achieve with a single denoising score network.

SDE parameters. We adopt the Variance Preserving framework proposed by Song et al. (2020), where the drift and diffusion coefficients in Eq. (5) are given by $f_t = -\frac{1}{2}\beta(t)$ and $g_t = \sqrt{\beta(t)}$, with a linear noise schedule $\beta(t) = \beta_{\min} + (\beta_{\max} - \beta_{\min})t$. In our implementation, we set $\beta_{\min} = 0.1$ and $\beta_{\max} = 20$. Under this parameterization, the transition density $p_{0t}(\cdot|x)$ is Gaussian with mean $k_t x$ and variance $\left(k_t^2 \int_0^t k_s^{-2} g_s^2 \, ds\right) \mathbf{I}_{N_x}$, where $k_t = \exp\left\{\int_0^t f_s \, ds\right\}$, allowing exact sampling of $X_t|X_0 = x$ without numerical integration of the SDE.

Neural architecture, runtime, and preprocessing. Our implementation adopts the architecture from the MINDE framework (Franzese et al., 2023). The score network is a U-Net-style MLP with residual blocks, skip connections, and GroupNorm normalization. The network accepts as input the concatenation of the three variables Y, X, Z as described in § 4.3.1, along with an encoding vector $[e_1, e_2, e_3] \in \{-1, 0, 1\}^3$ that specifies the role of each input: 1 indicates the variable for which the score is learned, 0 denotes a conditioning signal (kept at its original value), and -1 indicates marginalization (the input is set to zero). The diffusion time $t^\star$ is embedded through a learned two-layer MLP with GELU activation and injected into each residual block via a scale-shift mechanism. The output layer is initialized to zero to ensure stable early training. We use the Adam optimizer with exponential moving average (momentum $m = 0.999$) and importance sampling at both training and inference time. For preprocessing, standard z-score normalization is applied to all time series prior to training. Regarding computational cost, the average runtime for estimating the transfer entropy between two one-dimensional time series with $T = 10000$ observations is approximately 20 minutes per pair of estimates (both directions) on a single NVIDIA A100 GPU.

Algorithm 2: TENDE (Single Training Step)

Data: $[X_t, Y_t, Z_t]$
parameter: `approach` $\in \{$`conditional`, `joint`$\}$, $net_\theta()$, with θ current parameters
Obtain Y, X, Z as described in § 4.3.1
$t^\star \sim \mathcal{U}[0, T]$

`// diffuse signals to timestep` $t^\star$

$[Y_{t^\star}, X_{t^\star}, Z_{t^\star}] \leftarrow k_{t^\star}[Y, X, Z] + \left(k_{t^\star}^2 \int_0^{t^\star} k_s^{-2} g_s^2 \, \mathrm{d}s\right)^{\frac{1}{2}} [\epsilon_1, \epsilon_2, \epsilon_3]$, with $\epsilon_{1,2,3} \sim \gamma_1$

if *approach* = *conditional* **then**
  $c \sim \mathcal{U}\{0, 1\}$ `// Sample` c `from a discrete uniform in` $\{0, 1\}$
else
  $c \sim \mathcal{U}\{0, 1, 2\}$ `// Sample` c `from a discrete uniform in` $\{0, 1, 2\}$
if $c = 0$ **then**
  `// Estimated conditional score on source and target`
  $\frac{\hat{\epsilon}}{\left(k_{t^\star}^2 \int_0^{t^\star} k_s^{-2} g_s^2 \, \mathrm{d}s\right)^{\frac{1}{2}}} \leftarrow net_\theta([Y_{t^\star}, X, Z], t^\star, [1, 0, 0])$
else if $c = 1$ **then**
  `// Estimated conditional score only on the target`
  $\frac{\hat{\epsilon}}{\left(k_{t^\star}^2 \int_0^{t^\star} k_s^{-2} g_s^2 \, \mathrm{d}s\right)^{\frac{1}{2}}} \leftarrow net_\theta([Y_{t^\star}, X, Z], t^\star, [1, -1, 0])$
else
  `// Estimated unconditional score on the target`
  $\frac{\hat{\epsilon}}{\left(k_{t^\star}^2 \int_0^{t^\star} k_s^{-2} g_s^2 \, \mathrm{d}s\right)^{\frac{1}{2}}} \leftarrow net_\theta([Y_{t^\star}, X, Z], t^\star, [1, -1, -1])$
$L = \frac{g_{t^\star}^2}{\left(k_{t^\star}^2 \int_0^{t^\star} k_s^{-2} g_s^2 \, \mathrm{d}s\right)^{\frac{1}{2}}} \|\epsilon - \hat{\epsilon}\|^2$ `// Compute Montecarlo sample associated to Equation (7)`
return Update θ according to gradient of L