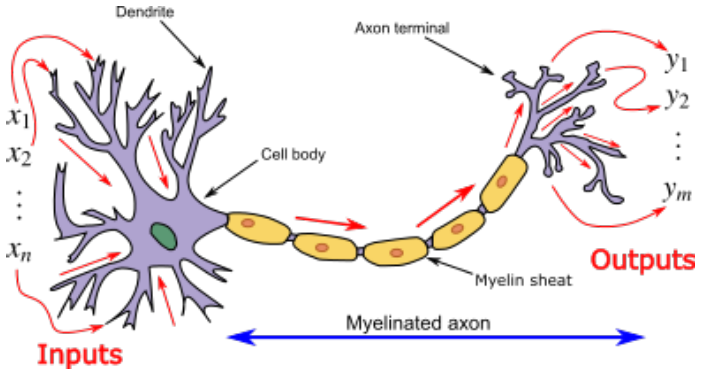
ITENE: Intrinsic Transfer Entropy Neural Estimator

Jingjing Zhang, Osvaldo Simeone, Zoran Cvetkovic, Eugenio Abela, and Mark Richardson

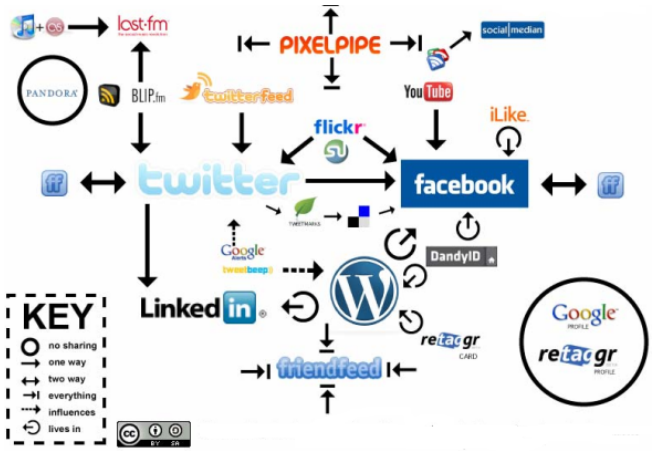
IZS, 27/02/2020

Estimating Information Flow

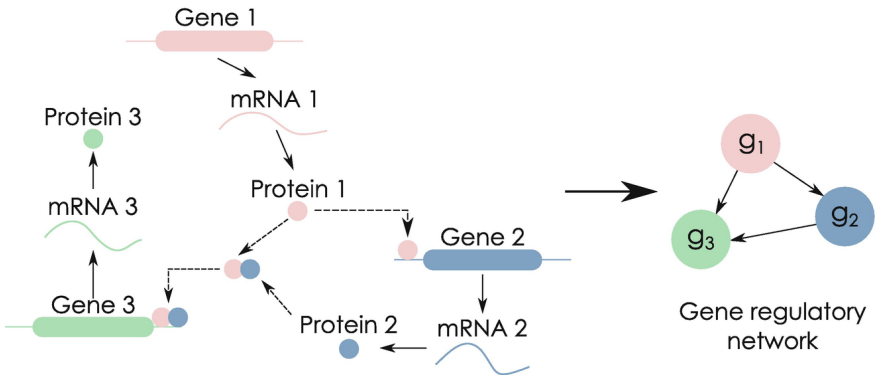
(a) transportation network



(b) biological neural network



(c) social network



(d) gene-regulatory network

Metrics: Transfer Entropy

- Jointly stationary random process $\{X_t, Y_t\}, t = 1, 2, \ldots$
- Transfer Entropy (TE) from $\{X_t\}$ to $\{Y_t\}$ with memory parameters (m, n) [Schreiber et al '00, Wibral et al '13]:

$$\mathrm{TE}_{X \to Y}(m, n) = I(X_{t-m}^{t-1}; Y_t | Y_{t-n}^{t-1})$$

- Interpretation: Improvement (in log-loss) of the prediction of Y_t from X_{t-m}^{t-1} when Y_{t-n}^{t-1} is known

Metrics: Transfer Entropy

- Jointly stationary random process $\{X_t, Y_t\}, t = 1, 2, \ldots$
- Transfer Entropy (TE) from $\{X_t\}$ to $\{Y_t\}$ with memory parameters (m, n) [Schreiber et al '00, Wibral et al '13]:

$$\mathrm{TE}_{X \to Y}(m, n) = I(X^-; Y^0 | Y^-)$$

- Interpretation: Improvement (in log-loss) of the prediction of Y^0 from X^- when Y^- is known

Intrinsic vs Synergistic Information Flow

- Conditioning on Y^- in the TE

$$\mathrm{TE}_{X\to Y}(m,n) = I(X^-; Y^0 | Y^-)$$

captures two forms of information flow from the past of $\{X_t\}$ to Y_t [James et al '18, Williams and Beer '10, Bertschinger et al '14]:

  - intrinsic, or exclusive: information extracted from X^- in addition to that present in Y^-
  - synergistic: information that can only be extracted by observing both X^- and Y^-

Transfer Entropy Decomposition

- This suggests the decomposition [James et al '18]

$$\text{TE} = \text{Intrinsic TE (ITE)} + \text{Synergistic TE (STE)}$$

Transfer Entropy Decomposition

- This suggests the decomposition [James et al '18]

$$\mathrm{TE} = \mathrm{Intrinsic\ TE\ (ITE)} + \mathrm{Synergistic\ TE\ (STE)}$$

- Possible operational definition [James et al '18]:
  - ITE = Rate of secret key that can be generated via public communications when the two legitimate parties have X^- and Y^0, respectively, and the adversary has Y^-

Transfer Entropy Decomposition

- This suggests the decomposition [James et al '18]

$$\text{TE} = \text{Intrinsic TE (ITE)} + \text{Synergistic TE (STE)}$$

- Possible operational definition [James et al '18]:
  - ITE = Rate of secret key that can be generated via public communications when the two legitimate parties have X^- and Y^0, respectively, and the adversary has Y^-
- Using an upper bound on this rate [Maurer and Wolf '99] as proposed in [James et al '18] yields the intrinsic conditional mutual information:

$$\text{ITE}_{X\to Y}(m,n) = \inf_{p(\bar{y}^-|y^-)} I(X^-; Y^0|\bar{Y}^-)$$

Intrinsic vs Synergistic Information Flow: Example

X_{-1}	Y_0	Y_{-1}	Pr
0	0	0	$1/4$
0	1	1	$1/4$
1	0	1	$1/4$
1	1	0	$1/4$

- $Y_0 = X_{-1} \oplus Y_{-1}$
- $\mathrm{TE}_{X \to Y}(1,1) = I(X_{-1}; Y_0 | Y_{-1}) = 1$ bit
- $\mathrm{ITE}_{X \to Y}(1,1) = I(X_{-1}; Y_0) = 0$ bit
- $\mathrm{STE}_{X \to Y}(1,1) = 1$ bit

Estimating Mutual Information and TE

- Discrete random variables:
  - plug-in methods [Kontoyiannis and Skoularidou '16, Jiao et al '17]
  - Bayesian estimators [Wolpert and Wolf '95]
  - universal compression [Jiao et al '13]
- Continuous random variables:
  - (non-parametric) kernel methods [Schreiber '00]
  - (non-parametric) k-nearest-neighbor (k-NN) methods [Kraskov et al '04]
  - (parametric) Maximum Likelihood [Suzuki et al '08]
  - (parametric) Mutual Information Neural Estimator (MINE) [Belghazi et al '18, Mukherjee et al '19]
  - Popular toolboxes: Java Information Dynamics Toolkit (JIDT) [Lizier '14] and TRENTOOL toolbox [Lindner et al '11]
- Hybrid discrete and continuous variables [Moon et al '17]

Main Contributions

- ITE Neural Estimator (ITENE) of the ITE based on
  - MINE
  - reparameterization trick (pathwise estimation of Monte Carlo gradients)
  - Combination is akin to Generative Adversarial Networks (GANs) [Goodfellow et al '14] (or contrastive learning, or likelihood ratio learning)
- Experimental results

Background: Classifier-Based Mutual Information Neural Estimator (MINE)

- Donsker-Varadhan (DV) equality

$$I(U;V) = \sup_{r(u,v)} \mathrm{E}_{p(u,v)}[\log(r(U,V))] - \log(\mathrm{E}_{p(u)p(v)}[r(U,V)])$$

Background: Classifier-Based Mutual Information Neural Estimator (MINE)

- Donsker-Varadhan (DV) equality

$$I(U;V) = \sup_{r(u,v)} \mathrm{E}_{p(u,v)}[\log(r(U,V))] - \log(\mathrm{E}_{p(u)p(v)}[r(U,V)])$$

- Optimal solution

$$r^*(u,v) = \frac{p(u,v)}{p(u)p(v)}$$

which is the optimal statistic for the test

$$\mathcal{H}_0 : U, V \sim p(u)p(v)$$
$$\mathcal{H}_1 : U, V \sim p(u,v)$$

Background: Classifier-Based Mutual Information Neural Estimator (MINE)

- From samples $\left\{ \begin{bmatrix} u_1 \\ v_1 \end{bmatrix}, \begin{bmatrix} u_2 \\ v_2 \end{bmatrix}, ..., \begin{bmatrix} u_T \\ v_T \end{bmatrix} \right\} \underset{\text{i.i.d.}}{\sim} p(u, v)$, construct labelled data set

$$\left\{ \begin{bmatrix} u_1 \\ v_1 \\ a_1 = 1 \end{bmatrix}, ..., \begin{bmatrix} u_T \\ v_T \\ a_T = 1 \end{bmatrix}, \begin{bmatrix} u_1 \\ v_{\pi(1)} \\ a_{T+1} = 0 \end{bmatrix}, ..., \begin{bmatrix} u_T \\ v_{\pi(T)} \\ a_{2T} = 0 \end{bmatrix} \right\}$$

- Train a discriminative classifier $p_\theta(a|u, v)$ modelled as a neural network

Background: Classifier-Based Mutual Information Neural Estimator (MINE)

- From samples $\left\{ \begin{bmatrix} u_1 \\ v_1 \end{bmatrix}, \begin{bmatrix} u_2 \\ v_2 \end{bmatrix}, ..., \begin{bmatrix} u_T \\ v_T \end{bmatrix} \right\} \underset{\text{i.i.d.}}{\sim} p(u, v)$, construct labelled data set

$$\left\{ \begin{bmatrix} u_1 \\ v_1 \\ a_1 = 1 \end{bmatrix}, ..., \begin{bmatrix} u_T \\ v_T \\ a_T = 1 \end{bmatrix}, \begin{bmatrix} u_1 \\ v_{\pi(1)} \\ a_{T+1} = 0 \end{bmatrix}, ..., \begin{bmatrix} u_T \\ v_{\pi(T)} \\ a_{2T} = 0 \end{bmatrix} \right\}$$

- Train a discriminative classifier $p_\theta(a|u, v)$ modelled as a neural network
- Using classifier $p_\theta(a|u, v)$ and assuming equi-probable hypotheses

$$\hat{r}_\theta(u, v) = \frac{p_\theta(a = 1|u, v)}{p_\theta(a = 0|u, v)}$$

Background: Classifier-Based Mutual Information Neural Estimator (MINE)

- From samples $\left\{ \begin{bmatrix} u_1 \\ v_1 \end{bmatrix}, \begin{bmatrix} u_2 \\ v_2 \end{bmatrix}, ..., \begin{bmatrix} u_T \\ v_T \end{bmatrix} \right\} \underset{\text{i.i.d.}}{\sim} p(u, v)$, construct labelled data set

$$\left\{ \begin{bmatrix} u_1 \\ v_1 \\ a_1 = 1 \end{bmatrix}, ..., \begin{bmatrix} u_T \\ v_T \\ a_T = 1 \end{bmatrix}, \begin{bmatrix} u_1 \\ v_{\pi(1)} \\ a_{T+1} = 0 \end{bmatrix}, ..., \begin{bmatrix} u_T \\ v_{\pi(T)} \\ a_{2T} = 0 \end{bmatrix} \right\}$$

- Train a discriminative classifier $p_\theta(a|u, v)$ modelled as a neural network
- Using classifier $p_\theta(a|u, v)$ and assuming equi-probable hypotheses

$$\hat{r}_\theta(u, v) = \frac{p_\theta(a = 1|u, v)}{p_\theta(a = 0|u, v)}$$

- Based on empirical distributions $\hat{p}(u)\hat{p}(v)$ and $\hat{p}(u)$ and $\hat{p}(v)$ on an held-out set, compute

$$\hat{I}(U; V) = \mathrm{E}_{\hat{p}(u,v)}[\log(\hat{r}_\theta(U, V))] - \log(\mathrm{E}_{\hat{p}(u)\hat{p}(v)}[\mathrm{clip}_\tau(\hat{r}_\theta(U, V))])$$

Background: Classifier-Based Mutual Information Neural Estimator (MINE)

- MINE is biased, and, if the classifier has sufficient capacity, consistent (when $\tau \to \infty$) [Belghazi et al '18].

Background: Classifier-Based Mutual Information Neural Estimator (MINE)

- MINE is biased, and, if the classifier has sufficient capacity, consistent (when $\tau \to \infty$) [Belghazi et al '18].
- If the estimate $\hat{r}_\theta(u, v)$ equals the true likelihood ratio, under the randomness of the sampling procedure generating the data set, we have for $\tau \to \infty$ [Song and Ermon '19]

$$\lim_{T \to \infty} T \ \mathrm{Var}[\widehat{\mathrm{I}}(U; V)] \geq e^{I(U;V)} - 1.$$

Transfer Entropy Neural Estimator (TENE)

- We can write

$$\mathrm{TE}_{X\to Y}(m,n) = I(X^-; Y^0|Y^-) = I(X^-; Y^0, Y^-) - I(X^-; Y^-)$$

- TENE [Mukherjee et al '19]: Given samples $\{x_t, y_t\}$, we obtain data set $\{x^-, y^-, y^0\}$, to which we apply MINE twice as

$$\widehat{\mathrm{TE}}_{X\to Y}(m,n) = \hat{I}(X^-; Y^0, Y^-) - \hat{I}(X^-; Y^-)$$

- TENE inherits the properties of MINE reviewed above.

Intrinsic Transfer Entropy Neural Estimator (ITENE)

- ITE:

$$\mathrm{ITE}_{X\to Y}(m,n) = \inf_{p(\bar{y}^-|y^-)} I(X^-; Y^0|\bar{Y}^-)$$

Intrinsic Transfer Entropy Neural Estimator (ITENE)

- ITE:

$$\mathrm{ITE}_{X\to Y}(m,n) = \inf_{p(\bar{y}^-|y^-)} I(X^-; Y^0|\bar{Y}^-)$$

- For a fixed channel, given samples $\{x^-, y^-, y^0\}$, generate samples $\bar{y}^- \sim p(\bar{y}^-|y^-)$
- Obtain $\widehat{\mathrm{TE}}_{X\to Y}(m,n) = \hat{I}(X^-; Y^0|\bar{Y}^-)$
- This estimate is differentiable in samples $\bar{y}^-$ used to average over $p(\bar{y}^-|y^-)$

Intrinsic Transfer Entropy Neural Estimator (ITENE)

- Reparameterization of $p_\phi(\bar{y}^-|y^-)$ [Mohamed et al '19]

$$\bar{y}^-_\phi = \mu_\phi(y^-) + \sigma_\phi(y^-) \odot \epsilon$$

where:

- $\mu_\phi(y^-)$ and $\log \sigma_\phi(y^-)$ are disjoint sets of outputs of a neural network with weights ϕ
- $\odot$ is the element-wise product
- $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ is a Gaussian vector independent of all other variables

Intrinsic Transfer Entropy Neural Estimator (ITENE)

- Reparameterization of $p_\phi(\bar{y}^-|y^-)$ [Mohamed et al '19]

$$\bar{y}^-_\phi = \mu_\phi(y^-) + \sigma_\phi(y^-) \odot \epsilon$$

where:

  - $\mu_\phi(y^-)$ and $\log \sigma_\phi(y^-)$ are disjoint sets of outputs of a neural network with weights ϕ
  - $\odot$ is the element-wise product
  - $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ is a Gaussian vector independent of all other variables

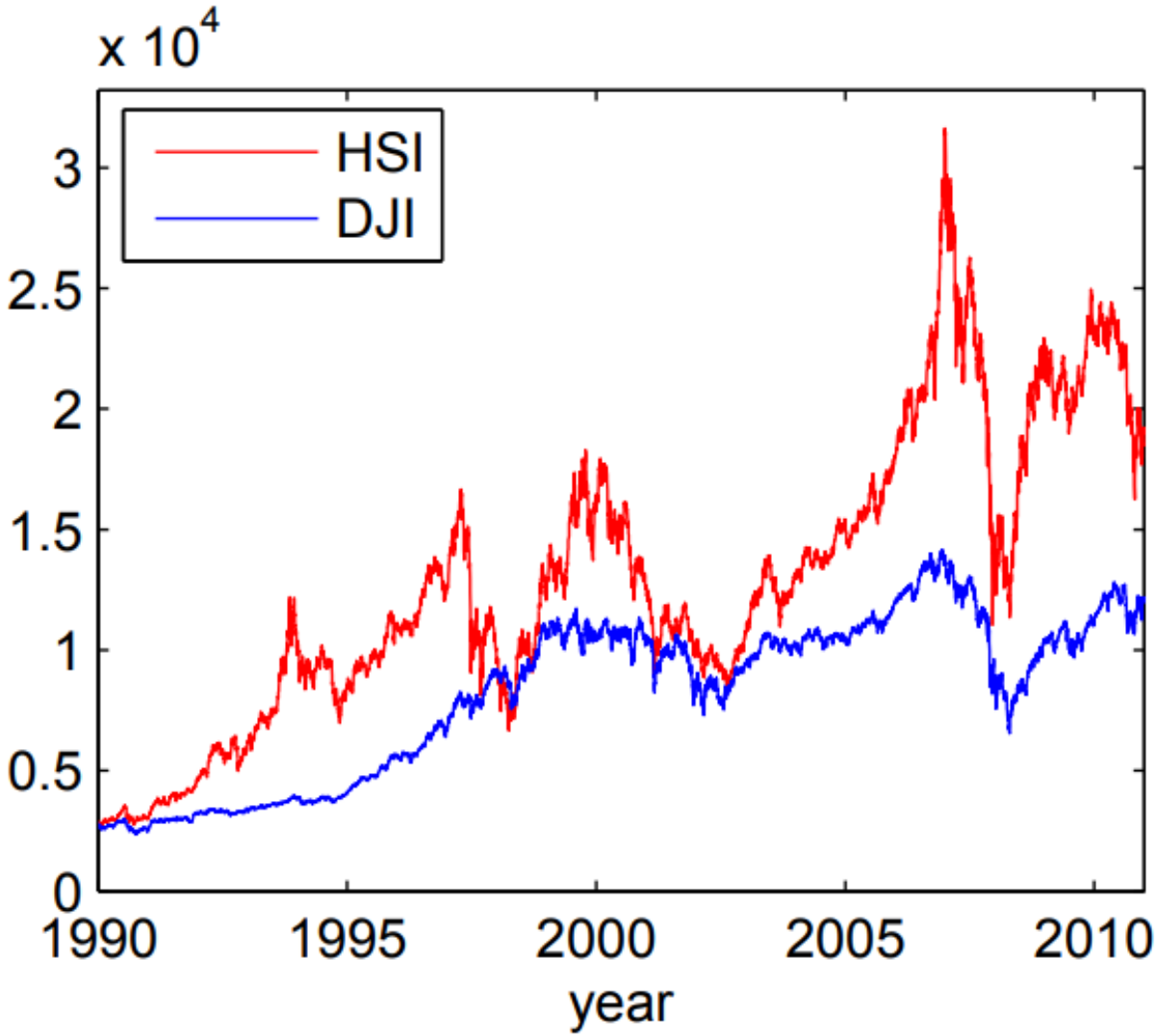
- ITENE tackles via gradient descent the problem

$$\widehat{\mathrm{ITE}}_{X \to Y}(m, n) = \inf_\phi \left(\hat{I}_\phi(X^-; Y^0 | \bar{Y}^-) \right)$$

using a nested loop.

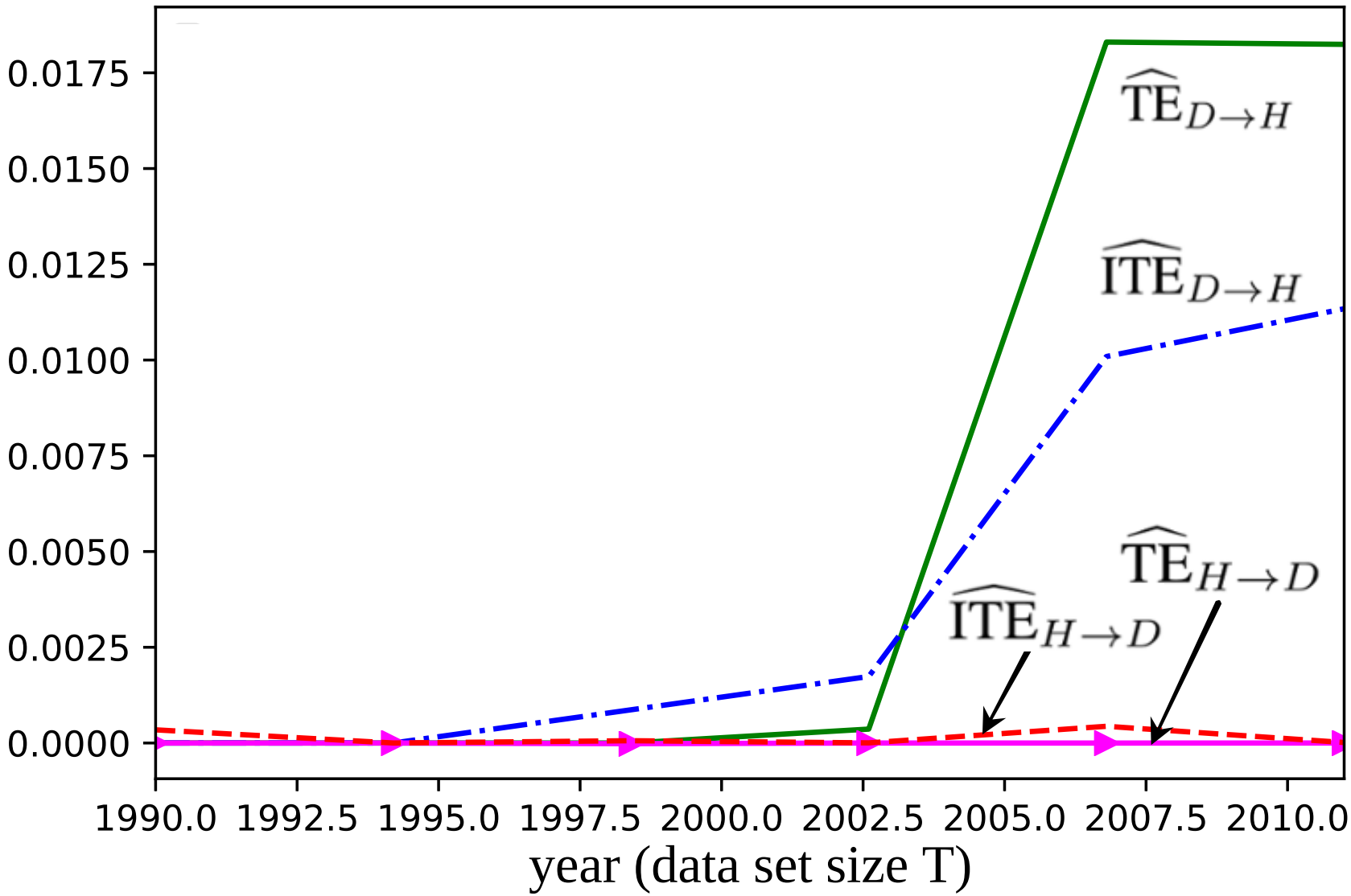
Experiments

- Dow Jones Index (DJIA) $\leftrightarrow$ Hang Seng Index (HSI)



Experiments

- Dow Jones Index (DJIA) $\leftrightarrow$ Hang Seng Index (HSI)



Concluding Remarks

- Proposed an estimator for Intrinsic Transfer Entropy (ITE) between two time series based on two-sample neural network classifiers and the reparameterization trick
- Future work:
  - applications to larger-scale data sets
  - theoretical properties

Acknowledgement

- European Research Council (ERC) under the European Union's Horizon 2020 Research and Innovation Programme (Grant Agreement No. 725731) and a King's Together award

References

[Schreiber '00] T. Schreiber, "Measuring information transfer," Phys. Rev. Lett., vol. 85, pp. 461–464, Jul. 2000.

[Wibral et al '13] R. Vicente, M. Wibral, and G. Lindner, Michaeland Pipa, "Transfer entropy—a model-free measure of effective connectivity for the neurosciences," Journal of Computational Neuroscience, vol. 30, no. 1, pp. 45–67, Feb. 2011.

[Massey et al '90] J. L. Massey, "Causality, feedback and directed information," in Proc. Int. Symp. Information Theory Applications (ISITA), Waikiki, Hawaii, Nov. 1990.

[Permuter et al '11] H. H. Permuter, Y. Kim, and T. Weissman, "Interpretations of directed information in portfolio theory, data compression, and hypothesis testing," IEEE Trans. Inf. Theory, vol. 57, no. 6, pp. 3248–3259, Jun. 2011.

[James et al '18] R. G. James, B. D. M. Ayala, B. Zakirov, and J. P. Crutchfield, "Modes of information flow." [Online]. Available: https://arxiv.org/abs/1808.06723.

References

[Maurer et al '99] U. M. Maurer and S. Wolf, "Unconditionally secure key agreement and the intrinsic conditional information," IEEE Trans. Inf. Theory, vol. 45, no. 2, pp. 499–514, Mar. 1999.

[Freedman et al '81] D. Freedman and P. Diaconis, "On the histogram as a density estimator: L2 theory," Probability Theory and Related Fields, vol. 57, no. 4, pp. 453–476, Dec. 1981.

[Kraskov et al '04] A. Kraskov, H. Stögbauer, and P. Grassberger, "Estimating mutual information," Phys. Rev. E, vol. 69, Jun. 2004.

[Frenzel et al'07] S. Frenzel and B. Pompe, "Partial mutual information for coupling analysis of multivariate time series," Phys. Rev. Lett., vol. 99, p. 204101(4), Nov. 2007.

[Suzuki et al '08] T. Suzuki, M. Sugiyama, J. Sese, and T. Kanamori, "Approximating mutual information by maximum likelihood density ratio estimation," in Proc. of the Int. Conf. on New Challenges for Feature Selection in Data Min. and Knowledge Discovery, 2008, pp. 5–20.

References

[Wolpert et al '95] D. H. Wolpert and D. R. Wolf, "Estimating functions of probability distributions from a finite set of samples," Phys. Rev. E, vol. 52, pp. 6841–6854, Dec. 1995.

[Lizier '14] J. T. Lizier, "JIDT: An information-theoretic toolkit for studying the dynamics of complex systems," Frontiers in Robotics and AI, vol. 1, p. 11, Dec. 2014.

[Lindner et al '11] 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, 119, Nov. 2011.

[Belghazi et al '18] M. I. Belghazi, A. Baratin, S. Rajeshwar, S. Ozair, Y. Bengio, A. Courville, and R. D. Hjelm, "Mutual information neural estimation," in Proc. Int. Conf. on Machine Learning, Stockholm, Sweden, Jul. 2018.

[Mohamed et al '19] S. Mohamed, M. Rosca, M. Figurnov, and A. Mnih, "Monte carlo gradient estimation in machine learning." [Online]. Available: https://arxiv.org/abs/1906.10652

References

[Miller '03] Miller EG. A new class of entropy estimators for multi-dimensional densities. In2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 Apr 6 (Vol. 3, pp. III-297). IEEE.

Metrics: Transfer Entropy

- Directed Information (DI) from $\{X_t\}$ to $\{Y_t\}$ [Massey et al '90] [Permuter et al '11]

$$\mathrm{DI}_{X \to Y} = \frac{1}{T} \sum_{t=1}^{T} I(X_1^{t-1}; Y_t | Y_1^{t-1})$$

- For jointly Markov processes[1] $\{X_t, Y_t\}$ with memory parameters m and n, the TE is an upper bound on the DI

[1] This implies the Markov chain $Y_t - (X_{t-m}^{t-1}, Y_{t-n}^{t-1}) - (X_1^{t-m-1}, Y_1^{t-n-1})$.

Intrinsic Transfer Entropy Neural Estimator (ITENE)

- We have

$$\hat{I}_\phi(X^-; Y^0, \bar{Y}^-) = \mathrm{E}_{\hat{p}(x^-,y^0,y^-)}\left[\mathrm{E}_{p(\epsilon)}[\log(\hat{r}_\theta(X^-, Y^0, \bar{Y}^-_\phi))]\right] - \log(\mathrm{E}_{\hat{p}(x^-)\hat{p}(y^0,y^-)}\left[\mathrm{E}_{p(\epsilon)}[\mathrm{clip}_\tau(\hat{r}_\theta(X^-, Y^0, \bar{Y}^-_\phi))]\right]$$

where θ obtained from MINE and $\bar{Y}^-_{\phi,t} = \mu_\phi(\bar{Y}_t) + \sigma_\phi(\bar{Y}_t) \odot \epsilon_t$ for i.i.d. samples $\epsilon_t \sim \mathcal{N}(0, \mathbf{I})$.

Intrinsic Transfer Entropy Neural Estimator (ITENE)

- We have

$$\hat{I}_\phi(X^-; Y^0, \bar{Y}^-) = \mathrm{E}_{\hat{p}(x^-,y^0,y^-)}\left[\mathrm{E}_{p(\epsilon)}[\log(\hat{r}_\theta(X^-, Y^0, \bar{Y}^-_\phi))]\right] - \log(\mathrm{E}_{\hat{p}(x^-)\hat{p}(y^0,y^-)}\left[\mathrm{E}_{p(\epsilon)}[\mathrm{clip}_\tau(\hat{r}_\theta(X^-, Y^0, \bar{Y}^-_\phi))]\right]$$

  where θ obtained from MINE and $\bar{Y}^-_{\phi,t} = \mu_\phi(\bar{Y}_t) + \sigma_\phi(\bar{Y}_t) \odot \epsilon_t$ for i.i.d. samples $\epsilon_t \sim \mathcal{N}(0, \mathbf{I})$.

- And

$$\hat{I}_\phi(X^-; \bar{Y}^-) = \mathrm{E}_{\hat{p}(x^-,y^-)}\left[\mathrm{E}_{p(\epsilon)}[\log(\hat{r}_{\theta'}(X^-, \bar{Y}^-_\phi))]\right] - \log(\mathrm{E}_{\hat{p}(x^-)\hat{p}(y^-)}\left[\mathrm{E}_{p(\epsilon)}[\mathrm{clip}_\tau(\hat{r}_{\theta'}(X^-, \bar{Y}^-_\phi))]\right]$$

  where θ' is obtained from MINE

Intrinsic Transfer Entropy Neural Estimator (ITENE)

- Alternate optimization:
  - (θ, θ') fom MINE
  - ϕ using stochastic gradient descent
- Gradient via reparameterization trick [Mohamed et al '19]

$$\nabla_\phi \hat{I}_\phi(X^-; Y^0, \bar{Y}^-) = \mathrm{E}_{\hat{p}(x^-,y^0,y^-)}\left[\mathrm{E}_{p(\epsilon)}\left[\frac{\nabla_{\bar{y}_\phi^-}\hat{r}_\theta}{\hat{r}_\theta} \times \mathrm{J}_\phi \bar{y}_\phi^-\right]\right] - \frac{\mathrm{E}_{\hat{p}(x^-)\hat{p}(y^0,y^-)}\left[\mathrm{E}_{p(\epsilon)}[\nabla_{\bar{y}_\phi^-}\hat{r}_\theta \times \mathrm{J}_\phi \bar{y}_\phi^-]\right]}{\mathrm{E}_{\hat{p}(x^-)\hat{p}(y^0,y^-)}\left[\mathrm{E}_{p(\epsilon)}[\hat{r}_\theta]\right]}$$

where, we have the gradient

$$\nabla_{\bar{y}_\phi^-}\hat{r}_\theta = \frac{\nabla_{\bar{y}_\phi^-} p_\theta(a=1|x^0, y^-, \bar{y}_\phi^-)}{(1 - p_\theta(a=1|x^0, y^-, \bar{y}_\phi^-))^2}$$

and similarly for $\nabla_\phi \hat{I}_\phi(X^-; \bar{Y}^-)$

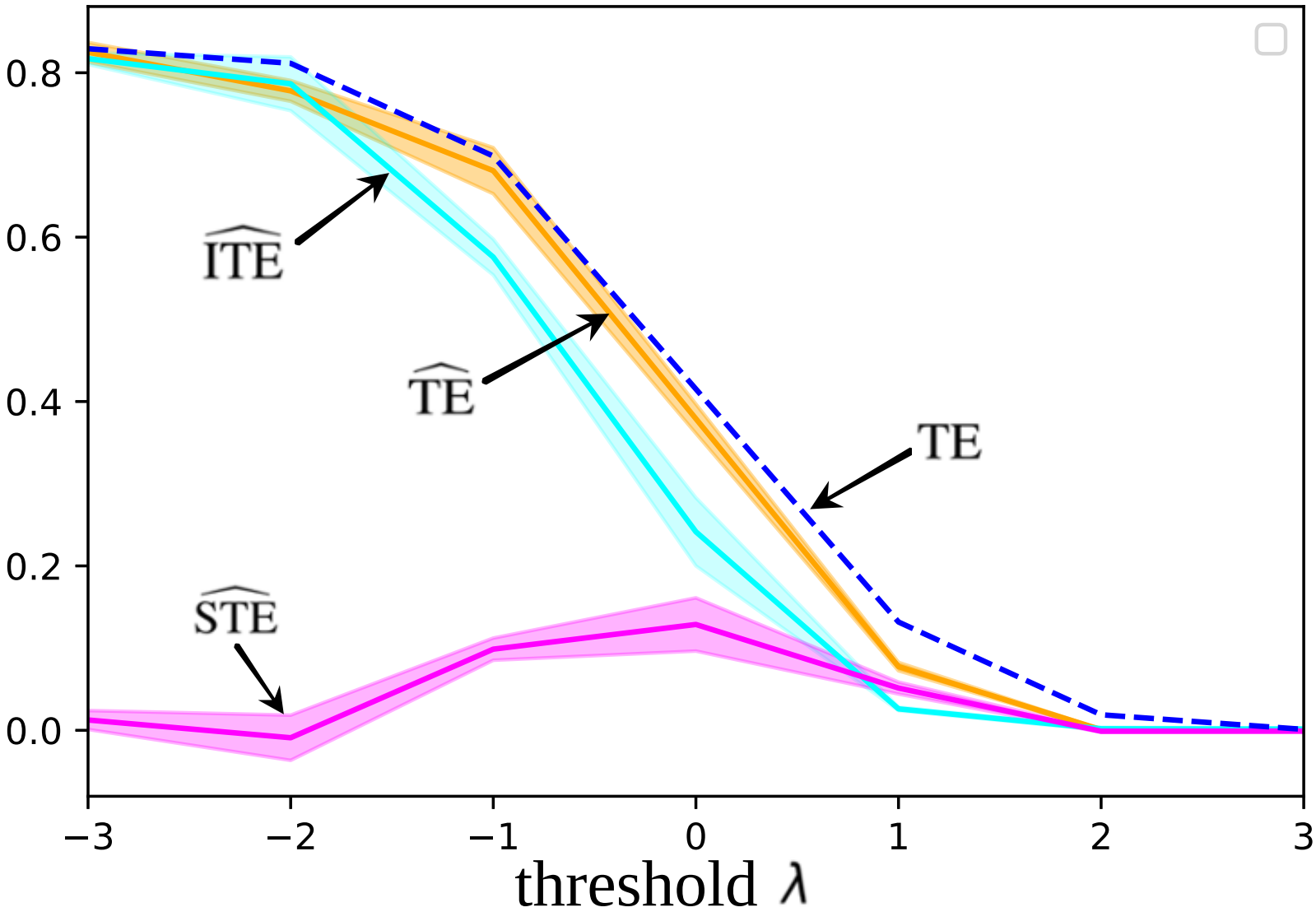
Experiments

- Toy example:

$$Y_t = \begin{cases} Z_t, & \text{if } Y_{t-1} < \lambda \\ \rho X_{t-1} + \sqrt{1-\rho^2} Z_t, & \text{if } Y_{t-1} \geq \lambda, \end{cases}$$

- Intuitions:
  - for large values of the threshold λ, no information flow between $\{X_t\}$ and $\{Y_t\}$
  - for small values, purely intrinsic flow of information
  - for intermediate values, partly synergistic information flow: knowing both Y_{t-1} and X_{t-1} is instrumental in obtaining information about Y_t

Experiments



Experiments