Conditional Mutual Information Neural Estimators

Sina Molavipour

KTH Royal Institute of Technology – School of EECS

45th International Conference on Acoustics, Speech, and Signal Processing
May 4–8, 2020

Joint work with Prof. Mikael Skoglund and Dr. Germán Bassi

Motivation

- Conditional mutual information (CMI) appears in many applications, for example:
  - It characterizes the capacity of some communication channels
  - It is the basis for defining notions of causal influence
- Although there are conventional methods to estimate the CMI, they suffer from the curse of dimensionality
- Recent studies suggest neural networks to be used to estimate information-theoretic quantities such as mutual information (MI)
- The extensions to estimate the CMI is **not** trivial and is addressed in this work

Motivation

- Conditional mutual information (CMI) appears in many applications, for example:
  - It characterizes the capacity of some communication channels
  - It is the basis for defining notions of causal influence
- Although there are conventional methods to estimate the CMI, they suffer from the curse of dimensionality
- Recent studies suggest neural networks to be used to estimate information-theoretic quantities such as mutual information (MI)
- The extensions to estimate the CMI is **not** trivial and is addressed in this work

Motivation

- Conditional mutual information (CMI) appears in many applications, for example:
  - It characterizes the capacity of some communication channels
  - It is the basis for defining notions of causal influence
- Although there are conventional methods to estimate the CMI, they suffer from the curse of dimensionality
- Recent studies suggest neural networks to be used to estimate information-theoretic quantities such as mutual information (MI)
- The extensions to estimate the CMI is **not** trivial and is addressed in this work

Motivation

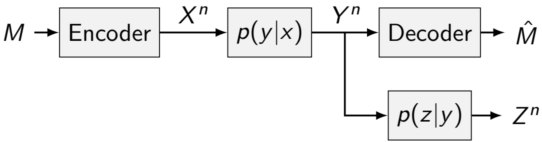
- Conditional mutual information (CMI) appears in many applications, for example:
  - It characterizes the capacity of some communication channels
  - It is the basis for defining notions of causal influence
- Although there are conventional methods to estimate the CMI, they suffer from the curse of dimensionality
- Recent studies suggest neural networks to be used to estimate information-theoretic quantities such as mutual information (MI)
- The extensions to estimate the CMI is **not** trivial and is addressed in this work

CMI as channel capacity

- CMI characterizes the capacity of communication channels such as:
  - Relay channel
  - Random state channel
  - Degraded wiretap channel (DWTC)
- The secrecy capacity of DWTC is $I(X; Y|Z)$

CMI as channel capacity

- CMI characterizes the capacity of communication channels such as:
  - Relay channel
  - Random state channel
  - Degraded wiretap channel (DWTC)
- The secrecy capacity of DWTC is $I(X; Y|Z)$



Definition

- For continuous random variables in $\mathcal{X}$ such that $(X, Y, Z) \sim p(x, y, z)$, the CMI is defined as below

Definition

$$I(X; Y|Z) := E_{p(z)}\left[D\left(\, p(x, y|Z)\, ||\, p(x|Z)\, p(y|Z)\,\right)\right]$$
$$= \int\int\int p(x, y, z) \log \frac{p(x, y, z)}{p(x|z)p(y, z)} dxdydz$$

Estimation of CMI

Several estimators have been proposed to estimate the CMI including:

- **Parametric estimators**: A model is assumed for the data, the parameters of the model are estimated, and CMI is computed
- **Kernel methods**: The densities are computed as sums of kernel functions and the estimated densities are plugged into the expression of CMI
- **Partitioning methods**: The space is partitioned into cells and the number of samples in each cell are counted to derive the estimator for CMI

Estimation of CMI

Several estimators have been proposed to estimate the CMI including:

- **Parametric estimators**: A model is assumed for the data, the parameters of the model are estimated, and CMI is computed
- **Kernel methods**: The densities are computed as sums of kernel functions and the estimated densities are plugged into the expression of CMI
- **Partitioning methods**: The space is partitioned into cells and the number of samples in each cell are counted to derive the estimator for CMI

Estimation of CMI

Several estimators have been proposed to estimate the CMI including:

- **Parametric estimators**: A model is assumed for the data, the parameters of the model are estimated, and CMI is computed
- **Kernel methods**: The densities are computed as sums of kernel functions and the estimated densities are plugged into the expression of CMI
- **Partitioning methods**: The space is partitioned into cells and the number of samples in each cell are counted to derive the estimator for CMI

Estimation of CMI (cont'd)

- ***k*-nearest neighbor (*k*-NN) estimator**: In this method the parameter k determines the radius of the ball around a given point in the space that captures all the k nearest samples to that point.
  - There is a well-known estimator for MI proposed in (Kraskov et al., 2004)[1] also know as KSG.
  - To estimate CMI, extensions of KSG have been proposed such as (Runge et al., 2017)[2]

[1] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. "Estimating mutual information". In: *Physical Review E* (2004).

[2] Jakob Runge. "Conditional independence testing based on a nearest-neighbor estimator of conditional mutual information". In: *arXiv:1709.01447* (2017).

Neural Estimators for MI and CMI

- **Neural estimators**: The methods are based on variational bounds for relative entropy
  - An estimator for MI was proposed by (Belghazi et al., 2018)[3]
  - This line of work was extended in (Mukherjee et al., 2019)[4] to estimate CMI

[3] Mohamed Ishmael Belghazi et al. "MINE: Mutual Information Neural Estimation". In: *35th Int. Conf. Mach. Learn. (ICML)*. 2018.

[4] Sudipto Mukherjee, Himanshu Asnani, and Sreeram Kannan. "CCMI: Classifier based Conditional Mutual Information Estimation". In: *arXiv:1906.01824* (2019).

Variational bounds

- The following lower bound holds for the relative entropy, and it is known as Donsker-Varadhan (DV) bound[5]:

Definition (DV bound)

$$D(p||q) \geq E_{p(x)}[f(X)] - \log E_{q(x)}[e^{f(X)}]$$

- A weaker lower bound can be derived which is also conventional to use (denoted here as NWJ bound)

Definition (NWJ bound)

$$D(p||q) \geq E_{p(x)}[f(X)] - e^{-1} E_{q(x)}[e^{f(X)}]$$

[5] M. D Donsker and S. S. Varadhan. "Asymptotic evaluation of certain Markov process expectations for large time. IV". In: (1983).

Variational bounds for CMI

Definition (DV bound for CMI)

$$I(X;Y|Z) \geq E_{p(x,y,z)}[f(X,Y,Z)] - \log E_{p(x|z)p(y,z)}[e^{f(X,Y,Z)}]$$

- **DV**: The bound is tight for $f^*_{DV}(\cdot) = \log \frac{p(x,y,z)}{p(x|z)p(y,z)}$

Definition (NWJ bound for CMI)

$$I(X;Y|Z) \geq E_{p(x,y,z)}[f(X,Y,Z)] - e^{-1} E_{p(x|z)p(y,z)}[e^{f(X,Y,Z)}]$$

- **NWJ**: The bound is tight for $f^*_{NWJ}(\cdot) = 1 + \log \frac{p(x,y,z)}{p(x|z)p(y,z)}$

Variational bounds for CMI

Definition (DV bound for CMI)

$$I(X;Y|Z) \geq E_{p(x,y,z)}[f(X,Y,Z)] - \log E_{p(x|z)p(y,z)}[e^{f(X,Y,Z)}]$$

- **DV**: The bound is tight for $f^*_{DV}(\cdot) = \log \frac{p(x,y,z)}{p(x|z)p(y,z)}$

Definition (NWJ bound for CMI)

$$I(X;Y|Z) \geq E_{p(x,y,z)}[f(X,Y,Z)] - e^{-1} E_{p(x|z)p(y,z)}[e^{f(X,Y,Z)}]$$

- **NWJ**: The bound is tight for $f^*_{NWJ}(\cdot) = 1 + \log \frac{p(x,y,z)}{p(x|z)p(y,z)}$

Challenges of Estimating CMI

Consider n triples (x, y, z) are available s.t. $(X, Y, Z) \sim p(x, y, z)$. Estimation of CMI using the introduced variational bounds encounters the following challenges:

- Since the density functions $p(x, y, z)$ and $p(x|z)p(y, z)$ are not available, to estimate the CMI, we compute the expectations using sample averages

> Let $\mathcal{B}^b_{joint}$ and $\mathcal{B}^b_{prod}$ be respectively batches of b triples (x, y, z) such that $(X, Y, Z) \sim p(x, y, z)$ and $(X, Y, Z) \sim p(x|z)p(y, z)$

- To compute a tight lower bound, it is required to properly approximate the density ratio $\Gamma^*(x, y, z) = \frac{p(x,y,z)}{p(x|z)p(y,z)}$

Challenges of Estimating CMI

Consider n triples (x, y, z) are available s.t. $(X, Y, Z) \sim p(x, y, z)$. Estimation of CMI using the introduced variational bounds encounters the following challenges:

- Since the density functions $p(x, y, z)$ and $p(x|z)p(y, z)$ are not available, to estimate the CMI, we compute the expectations using sample averages

Let $\mathcal{B}^b_{joint}$ and $\mathcal{B}^b_{prod}$ be respectively batches of b triples (x, y, z) such that $(X, Y, Z) \sim p(x, y, z)$ and $(X, Y, Z) \sim p(x|z)p(y, z)$

- To compute a tight lower bound, it is required to properly approximate the density ratio $\Gamma^*(x, y, z) = \frac{p(x,y,z)}{p(x|z)p(y,z)}$

Challenges of Estimating CMI

Consider n triples (x, y, z) are available s.t. $(X, Y, Z) \sim p(x, y, z)$. Estimation of CMI using the introduced variational bounds encounters the following challenges:

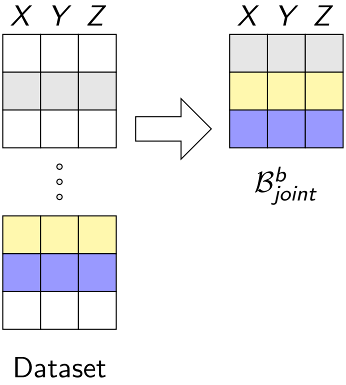
- Since the density functions $p(x, y, z)$ and $p(x|z)p(y, z)$ are not available, to estimate the CMI, we compute the expectations using sample averages

> Let $\mathcal{B}^b_{joint}$ and $\mathcal{B}^b_{prod}$ be respectively batches of b triples (x, y, z) such that $(X, Y, Z) \sim p(x, y, z)$ and $(X, Y, Z) \sim p(x|z)p(y, z)$

- To compute a tight lower bound, it is required to properly approximate the density ratio $\Gamma^*(x, y, z) = \frac{p(x,y,z)}{p(x|z)p(y,z)}$

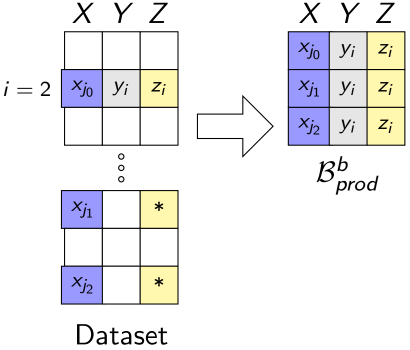
Construct sample batch

- **The joint batch** $\mathcal{B}^b_{joint}$: Let $\mathcal{I}_b$ be a set of b random distinct integers in $[1:n]$. For each $i \in \mathcal{I}_b$, we put (x_i, y_i, z_i) in the batch.



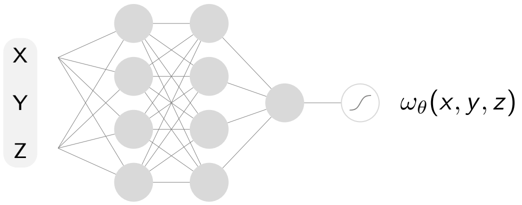
Construct sample batch (cont'd)

- **The product batch $\mathcal{B}^b_{prod}$:** We use the notion of k-NN to re-sample the dataset such that the samples are distributed according to $p(x|z)p(y,z)$.
  Let $\mathcal{I}_m$ be a set of m random distinct integers in $[1:n]$. For each $i \in \mathcal{I}_m$, let $\mathcal{A}_{z_i}$ be the set of indices of k nearest neighbors of z_i in z^n. We put all triples (x_j, y_i, z_i) for $i \in \mathcal{I}_m$ and $j \in \mathcal{A}_{z_i}$.



Neural network classifier

- To approximate the density ratio $\Gamma^*(x, y, z)$, (Mukherjee et al. 2019)[6] proposed using a neural classifier ω_θ parameterized with θ such that:
  - The input of the network is a triple (x, y, z) that either is generated according to $p(x, y, z)$ or $p(x|z)p(y, z)$
  - The neural network classifies the input based on its density
  - The last layer of the neural network is a **sigmoid** function



[6] Sudipto Mukherjee, Himanshu Asnani, and Sreeram Kannan. "CCMI: Classifier based Conditional Mutual Information Estimation". In: *arXiv:1906.01824* (2019).

Loss function

- The loss function to optimize θ is the expected **binary cross entropy** loss

$$L(\omega_\theta) := -E_{p(q)p(x,y,z|q)}\big[Q \log \omega_\theta(X, Y, Z) + (1 - Q)\log(1 - \omega_\theta(X, Y, Z))\big],$$

where $Q \in \{0, 1\}$ is the corresponding label of an input

Lemma

Let $\omega^*(x, y, z)$ be the minimizer of $L(\omega)$. Then:

$$\Gamma^*(x, y, z) = \frac{\omega^*(x, y, z)}{1 - \omega^*(x, y, z)}.$$

- So by minimizing $L(\omega_\theta)$, with sufficient samples and proper network, we can approximate the density ratio $\Gamma^*(x, y, z)$ and accordingly f^*_{DV} and f^*_{NWJ}

Loss function

- The loss function to optimize θ is the expected **binary cross entropy** loss

$$L(\omega_\theta) := -E_{p(q)p(x,y,z|q)}\big[Q \log \omega_\theta(X, Y, Z) + (1 - Q) \log(1 - \omega_\theta(X, Y, Z))\big],$$

where $Q \in \{0, 1\}$ is the corresponding label of an input

Lemma

Let $\omega^*(x, y, z)$ be the minimizer of $L(\omega)$. Then:

$$\Gamma^*(x, y, z) = \frac{\omega^*(x, y, z)}{1 - \omega^*(x, y, z)}.$$

- So by minimizing $L(\omega_\theta)$, with sufficient samples and proper network, we can approximate the density ratio $\Gamma^*(x, y, z)$ and accordingly f^*_{DV} and f^*_{NWJ}

CMI neural estimators

- In practice, we don't have $L(\omega_\theta)$ and we compute the empirical loss $L_{2b}(\omega_\theta)$ using the training data batches $\mathcal{B}^b_{joint}$ and $\mathcal{B}^b_{prod}$
- $\hat{\theta} = \arg\min_\theta L_{2b}(\omega_\theta)$ and we obtain $\hat{\Gamma}(x,y,z) = \frac{\omega_{\hat{\theta}}(x,y,z)}{1-\omega_{\hat{\theta}}(x,y,z)}$

Definition

$$\hat{I}^{b,\hat{\theta}}_{DV} := \frac{1}{b} \sum_{(x,y,z)\in\mathcal{B}^b_{\text{joint}}} \log \hat{\Gamma}(x,y,z) - \log \frac{1}{b} \sum_{(x,y,z)\in\mathcal{B}^b_{\text{prod}}} \hat{\Gamma}(x,y,z),$$

$$\hat{I}^{b,\hat{\theta}}_{NWJ} := 1 + \frac{1}{b} \sum_{(x,y,z)\in\mathcal{B}^b_{\text{joint}}} \log \hat{\Gamma}(x,y,z) - \frac{1}{b} \sum_{(x,y,z)\in\mathcal{B}^b_{\text{prod}}} \hat{\Gamma}(x,y,z).$$

- While $\hat{I}^{b,\hat{\theta}}_{NWJ}$ is an **unbiased** estimator, $\hat{I}^{b,\hat{\theta}}_{DV}$ is **biased**

The bias problem

- In practice, the estimators are computed for several trials and the results are averaged

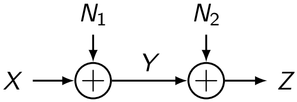
$$\overline{\hat{I}^{b,\hat{\theta}}_{DV}} = \frac{1}{T}\sum_{t=1}^{T} \hat{I}^{b,\hat{\theta}}_{DV}(t) \qquad \& \qquad \overline{\hat{I}^{b,\hat{\theta}}_{NWJ}} = \frac{1}{T}\sum_{t=1}^{T} \hat{I}^{b,\hat{\theta}}_{NWJ}(t)$$

- So while $\overline{\hat{I}^{b,\hat{\theta}}_{NWJ}}$ estimates a tight lower bound for CMI, $\overline{\hat{I}^{b,\hat{\theta}}_{DV}}$ is **neither** estimating a lower bound nor an upper bound

Experimental results

DWTC

- Gaussian model: $\begin{cases} X \sim \mathcal{N}(0, P) \\ Y \sim \mathcal{N}(X, \sigma_1^2) \\ Z \sim \mathcal{N}(Y, \sigma_2^2) \end{cases}$



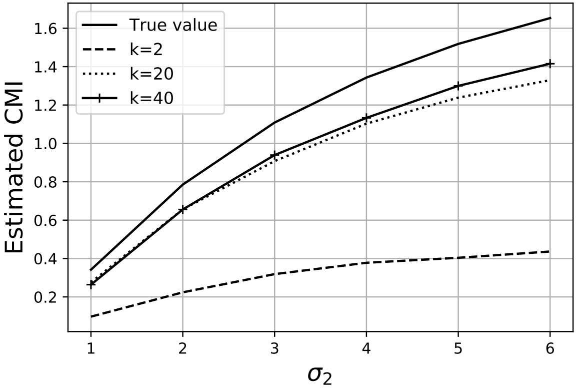
- The secrecy capacity is

$$I(X; Y|Z) = \frac{1}{2}\log\left(1 + \frac{P}{\sigma_1^2}\right) - \frac{1}{2}\log\left(1 + \frac{P}{\sigma_1^2 + \sigma_2^2}\right)$$

Experimental results

Estimation performance

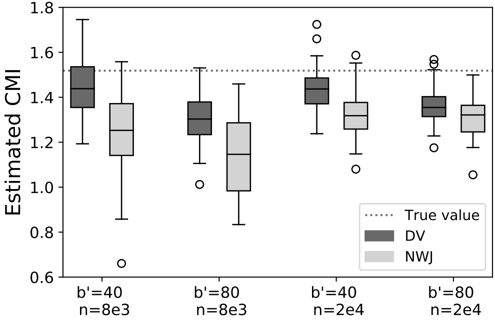
- $P = 100$, $\sigma_1 = 1$, $n = 2e4$ and $b = n/2$
- The results are for the DV bound, averaged for $T = 20$ trials



Experimental results

Bias problem

- $P = 100$, $\sigma_1 = 1$, $\sigma_2 = 5$ and $b = n/2$
- To verify the bias problem, $\hat{I}_{DV}^{b',\hat{\theta}}$ and $\hat{I}_{NWJ}^{b',\hat{\theta}}$ are computed with batches of size b' instead
- The results are averaged for $T = 20$ trials, and repeated 50 times for the box plots



Summary

- The variational bounds enabled proposing neural estimators, and recent works have shown significant improvements that can be achieved using these estimators
- The k-NN method for batching shows desirable performance, and increasing k with respect to n improves the result
- If the intention of the estimation is the CMI, both DV and NWJ estimators can be used
- If we need a lower bound for CMI, the NWJ estimator is a more **justified** method regarding the bias problem
- As a future direction, we are improving the k-NN batch construction and we achieved a better performance comparing to other methods