Combating the Instability of Mutual Information-based Losses via Regularization (Supplementary material)

Kwanghee Choi[*1] **Siyeong Lee**[*2]

[1]Sogang University
[2]NAVER LABS

A PROOFS

In this section, we provide proof of all theoretical results mentioned in the manuscript.

A.1 PROOF OF THE $D_{\textbf{ReDV}}$ REPRESENTATION

In this subsection, we consider two probability distributions $\mathbb{P}$ and $\mathbb{Q}$, with $\mathbb{P}$ absolutely continuous with respect to $\mathbb{Q}$. In addition, assume that both distributions are absolutely continuous with respect to Lebesgue measure μ on some compact domain Ω.

We first show that there exists the family of optimal function for the DV representation (Donsker and Varadhan, 1975).

Lemma 1. *All functions of the form $T = \log \frac{d\mathbb{P}}{d\mathbb{Q}} + C^*$ is optimal for the DV representation D_{DV}.*

Proof. To show this theorem, we borrow the proof of the dual representation for the KL divergence (Belghazi et al., 2018).

For a function T, let Δ_T be the gap

$$\Delta_T := D_{KL}(P||Q) - \left(\mathbb{E}_{\mathbb{P}}(T) - \log \mathbb{E}_{\mathbb{Q}}(e^T)\right). \tag{1}$$

By Theorem 1 of MINE (Donsker and Varadhan, 1975), we already knew that there exists an optimal function $T^* = \log \frac{d\mathbb{P}}{d\mathbb{Q}} + C$ for some $C \in \mathbb{R}$ such that $\Delta_{T^*} = 0$.

Consider a function $T = \log \frac{d\mathbb{P}}{d\mathbb{Q}} + C^*$ for $C^* \in \mathbb{R}$. The function T can be rewritten as $(T^* - C) + C^*$.

Since

$$\mathbb{E}_{\mathbb{P}}(T) = \mathbb{E}_{\mathbb{P}}(T^* - C + C^*) \tag{2}$$
$$= \mathbb{E}_{\mathbb{P}}(T^*) - C + C^*, \tag{3}$$

and

$$\log(\mathbb{E}_{\mathbb{Q}}(e^T)) = \log(\mathbb{E}_{\mathbb{Q}}(e^{T^*-C+C^*})) \tag{4}$$
$$= \log(e^{C^*-C}\mathbb{E}_{\mathbb{Q}}(e^{T^*})) \tag{5}$$
$$= (C^* - C) + \log(\mathbb{E}_{\mathbb{Q}}(e^{T^*})), \tag{6}$$

$$\mathbb{E}_{\mathbb{P}}(T) - \log(\mathbb{E}_{\mathbb{Q}}(e^T)) = \mathbb{E}_{\mathbb{P}}(T^*) - \log(\mathbb{E}_{\mathbb{Q}}(e^{T^*})). \tag{7}$$

[*] These authors contributed equally to this work.

Therefore, for the function T,

$$\Delta_T = D_{KL}(P||Q) - \big(\mathbb{E}_{\mathbb{P}}(T) - \log \mathbb{E}_{\mathbb{Q}}(e^T)\big) = D_{KL}(P||Q) - \Big(\mathbb{E}_{\mathbb{P}}(T^*) - \log \mathbb{E}_{\mathbb{Q}}(e^{T^*})\Big) = \Delta_{T^*} = 0. \tag{8}$$

As a result, optimal functions takes the form $T = \log \frac{d\mathbb{P}}{d\mathbb{Q}} + C^*$ for some constant $C^* \in \mathbb{R}$. □

Theorem. *(Theorem 1 restated) Let* d *be a distance function on* $\mathbb{R}$*. For any constant* $C^* \in \mathbb{R}$ *and any class of functions* $\mathcal{T}$ *mapping from* Ω *to* $\mathbb{R}$*, we have a novel dual representation of* KL *divergence*

$$D_{ReDV} := \sup_{T \in \mathcal{T}} \mathbb{E}_{\mathbb{P}}(T) - \log(\mathbb{E}_{\mathbb{Q}}(e^T)) - d(\log(\mathbb{E}_{\mathbb{Q}}(e^T)), C^*) = D_{KL}(\mathbb{P}||\mathbb{Q}). \tag{9}$$

Proof. i) For any T,

$$\mathbb{E}_{\mathbb{P}}(T) - \log(\mathbb{E}_{\mathbb{Q}}(e^T)) - d(\log(\mathbb{E}_{\mathbb{Q}}(e^T)), C^*) \leq \mathbb{E}_{\mathbb{P}}(T) - \log(\mathbb{E}_{\mathbb{Q}}(e^T)). \tag{10}$$

Therefore, $\sup_{T:\Omega\to\mathbb{R}} \mathbb{E}_{\mathbb{P}}(T) - \log(\mathbb{E}_{\mathbb{Q}}(e^T)) - d(\log(\mathbb{E}_{\mathbb{Q}}(e^T)), C^*) \leq D_{KL}(\mathbb{P}||\mathbb{Q})$.

ii) By the lemma above, there exists $T^* = \log \frac{d\mathbb{P}}{d\mathbb{Q}} + C^*$ such that

$$D_{KL}(\mathbb{P}||\mathbb{Q}) = \mathbb{E}_{\mathbb{P}}(T^*) - \log(\mathbb{E}_{\mathbb{Q}}(e^{T^*})) \tag{11}$$

and

$$\log(\mathbb{E}_{\mathbb{Q}}(e^{T^*})) = \log(\mathbb{E}_{\mathbb{Q}}(e^{C^*}\frac{d\mathbb{P}}{d\mathbb{Q}})) = \log(\int e^{C^*}\frac{d\mathbb{P}}{d\mathbb{Q}}d\mathbb{Q}) = C^*. \tag{12}$$

Therefore,

$$\sup_{T:\Omega\to\mathbb{R}} \mathbb{E}_{\mathbb{P}}(T) - \log(\mathbb{E}_{\mathbb{Q}}(e^T)) - d(\log(\mathbb{E}_{\mathbb{Q}}(e^T)), C^*) \geq \mathbb{E}_{\mathbb{P}}(T^*) - \log(\mathbb{E}_{\mathbb{Q}}(e^{T^*})) - d(\log(\mathbb{E}_{\mathbb{Q}}(e^{T^*})), C^*) \tag{13}$$

$$= D_{KL}(\mathbb{P}||\mathbb{Q}). \tag{14}$$

Combining i) and ii) finishes the proof. □

A.2 EXTENSION TO NWJ REPRESENTATION

In this subsection, we show that our regularizer can also be applied to the NWJ representation (Nguyen et al., 2010).

Theorem. *Let* d *be a distance function on* $\mathbb{R}$*. We have another dual representation such that*

$$D_{ReNWJ} := (\mathbb{P}||\mathbb{Q}) = \sup_{T:\Omega\to\mathbb{R}} \mathbb{E}_{\mathbb{P}}(T) - \mathbb{E}_{\mathbb{Q}}(e^{T-1}) - d(\mathbb{E}_{\mathbb{Q}}(e^{T-1}), 1) = D_{KL}(\mathbb{P}||\mathbb{Q}). \tag{15}$$

Proof. As d is a distance function, $d(\mathbb{E}_{\mathbb{Q}}(e^{T-1}), 1) \geq 0$.

i) For any T,

$$\mathbb{E}_{\mathbb{P}}(T) - \mathbb{E}_{\mathbb{Q}}(e^{T-1}) - d(\mathbb{E}_{\mathbb{Q}}(e^{T-1}), 1) \leq \mathbb{E}_{\mathbb{P}}(T) - \mathbb{E}_{\mathbb{Q}}(e^{T-1}). \tag{16}$$

Therefore, $\sup_{T:\Omega\to\mathbb{R}} \mathbb{E}_{\mathbb{P}}(T) - \mathbb{E}_{\mathbb{Q}}(e^{T-1}) - d(\mathbb{E}_{\mathbb{Q}}(e^{T-1}), 1) \leq D_{KL}(\mathbb{P}||\mathbb{Q})$.

ii) By Poole et al. (2019), there exists $T^* = \log \frac{d\mathbb{P}}{d\mathbb{Q}} + 1$ such that

$$D_{KL}(\mathbb{P}||\mathbb{Q}) = \mathbb{E}_{\mathbb{P}}(T^*) - \mathbb{E}_{\mathbb{Q}}(e^{T^*-1}). \tag{17}$$

$$\mathbb{E}_{\mathbb{P}}(T^*) = \mathbb{E}_{\mathbb{P}}(1 + \log(\frac{d\mathbb{P}}{d\mathbb{Q}})) = 1 + D_{KL}(\mathbb{P}||\mathbb{Q}). \tag{18}$$

and

$$\mathbb{E}_{\mathbb{Q}}(e^{T^*-1}) = \mathbb{E}_{\mathbb{Q}}(\frac{d\mathbb{P}}{d\mathbb{Q}}) = 1. \tag{19}$$

Therefore,

$$\sup_{T:\Omega\to\mathbb{R}} \mathbb{E}_{\mathbb{P}}(T) - \mathbb{E}_{\mathbb{Q}}(e^{T-1}) - d(\mathbb{E}_{\mathbb{Q}}(e^{T-1}), 1) \geq \mathbb{E}_{\mathbb{P}}(T^*) - \mathbb{E}_{\mathbb{Q}}(e^{T^*-1}) - d(\mathbb{E}_{\mathbb{Q}}(e^{T^*-1}), 1) \tag{20}$$

$$= D_{KL}(\mathbb{P}||\mathbb{Q}). \tag{21}$$

Combining i) and ii) finishes the proof. □

A.3 MATHEMATICAL PROPERTIES OF I_{ReMINE}

This subsection presents the proof of the consistency and the sample complexity of I_{ReMINE}. To show these properties, we assume that the input space of the functions below is a compact domain, and all measures are absolutely continuous with respect to the Lebesgue measure. We will restrict to families of feedforward functions with continuous activations, with a single output neuron. To avoid unnecessary heavy notation, we denote $\mathbb{P} = \mathbb{P}_{XY}$ and $\mathbb{Q} = \mathbb{P}_X \otimes \mathbb{P}_Y$ as the joint distribution and the product of marginals unless specified.

First, we define the sample complexity of the MI estimator. As mentioned by Belghazi et al. (2018), this property is related to the *approximation* problem, which addresses the size of the family of function T_θ, and the *estimation* problem, which addresses whether it is a reliable estimator.

Definition 1. *The MI estimator $\hat{I}(X,Y)_n$ is strongly consistent if for all $\epsilon > 0$, there exists a positive integer N and a choice of statistics networks such that $\forall n \geq N, |I(X,Y) - \hat{I}(X,Y)_n| \leq \epsilon$, where the probability is over a set of samples.*

Consistency proof

Lemma 2. *(Approximation) Let η > 0. There exists a neural network function T_θ with parameters $\theta \in \Theta$ such that*

$$|\hat{I}_{ReMINE}(X,Y) - I_{ReMINE}(X,Y)| \leq \eta, \tag{22}$$

where

$$\hat{I}_{ReMINE}(X,Y) = \sup_{\theta\in\Theta} \mathbb{E}_{\mathbb{P}}(T_\theta) - \log(\mathbb{E}_{\mathbb{Q}}(e^{T_\theta}) - d(\log(\mathbb{E}_{\mathbb{Q}}(e^{T_\theta}), C^*)). \tag{23}$$

Proof. Without loss of generality, we set $T^* = \log\frac{d\mathbb{P}}{d\mathbb{Q}}$. By construction, T^* satisfies:

$$\mathbb{E}_{\mathbb{P}}(T^*) = I(X,Y), \quad \mathbb{E}_{\mathbb{Q}}(e^{T^*}) = 1, \quad \log(\mathbb{E}_{\mathbb{Q}}(e^{T^*})) = 0 \tag{24}$$

For a function T,

$$I_{\text{ReMINE}}(X,Y) - \hat{I}_{\text{ReMINE}}(X,Y) \tag{25}$$

$$\leq \mathbb{E}_{\mathbb{P}}(T^* - T) + \log(\mathbb{E}_{\mathbb{Q}}(e^T)) + d(\log(\mathbb{E}_{\mathbb{Q}}(e^T), C^*) - d(\log(\mathbb{E}_{\mathbb{Q}}(e^{T^*}), C^*) \tag{26}$$

$$\leq \mathbb{E}_{\mathbb{P}}(T^* - T) + \log(\mathbb{E}_{\mathbb{Q}}(e^T)) + d(\log(\mathbb{E}_{\mathbb{Q}}(e^T), \log(\mathbb{E}_{\mathbb{Q}}(e^{T^*})) \tag{27}$$

$$\leq \mathbb{E}_{\mathbb{P}}(T^* - T) + \mathbb{E}_{\mathbb{Q}}(e^T - e^{T^*}) + d(\mathbb{E}_{\mathbb{Q}}(e^T) - 1, 0) \tag{28}$$

where we used the inequality $\log x \leq x - 1$ and $d(\cdot)$ is the distance function induced by norm on $\mathbb{R}$ (e.g., absolute or square error). Fix $\eta > 0$. By the universal approximation theorem, we may choose a feedforward network function $T_\theta \leq M$ such that

$$\mathbb{E}_{\mathbb{P}}|T^* - T_\theta| \leq \frac{\eta}{3}, \quad \mathbb{E}_{\mathbb{Q}}|T^* - T_\theta| \leq \frac{\eta}{3}e^{-M}, \quad \text{and } d(\mathbb{E}_{\mathbb{Q}}|T_\theta - T^*|, 0) \leq \frac{\eta}{3\cdot d(e^M, 0)} \tag{29}$$

Since exp is Lipschitz continuous with constant e^M on $(-\infty, M]$, we have

$$\mathbb{E}_{\mathbb{Q}}|e^{T^*} - e^{T_\theta}| \leq e^M \mathbb{E}_{\mathbb{Q}}|T^* - T_\theta| \leq \frac{\eta}{3}, \tag{30}$$

and

$$\begin{aligned} d(\mathbb{E}_{\mathbb{Q}}(e^{T}) - 1, 0) &= d(\mathbb{E}_{\mathbb{Q}}(e^{T_\theta}) - \mathbb{E}_{\mathbb{Q}}(e^{T^*}), 0) = d(\mathbb{E}_{\mathbb{Q}}|e^{T_\theta} - e^{T^*}|, 0) \qquad (31) \\ &\leq d(e^{M}\mathbb{E}_{\mathbb{Q}}|T_\theta - T^*|, 0) \leq d(e^{M}, 0) \cdot d(\mathbb{E}_{\mathbb{Q}}|T_\theta - T^*|, 0) \leq \frac{\eta}{3}. \qquad (32) \end{aligned}$$

From **??**, **??**, **??**, **??** and the triangular inequality, we then obtain:

$$|\hat{I}_{\text{ReMINE}}(X,Y) - I_{\text{ReMINE}}(X,Y)| < \eta. \tag{33}$$

□

Lemma 3. *(Esitmation) Let $\eta > 0$. Given a neural network function T_θ with parameters $\theta \in \Theta$, there exists $N \in \mathbb{N}$ such that*

$$\forall n \geq N, \mathcal{P}(|\hat{I}_{\textit{ReMINE}}(X,Y)_n - \hat{I}_{\textit{ReMINE}}(X,Y)| \leq \eta) = 1, \tag{34}$$

where $\hat{I}_{\textit{ReMINE}}(X,Y)_n$ is the ReMINE representation which is empirically obtained by n samples.

Proof. We start by using the triangular inequality to write,

$$\begin{aligned} |\hat{I}_{\text{ReMINE}}(X,Y)_n - \sup_{\theta\in\Theta} \hat{I}_{\text{ReMINE}}(T_\theta)| \leq \sup_{\theta\in\Theta} |\mathbb{E}_{\mathbb{P}}(T_\theta) - \mathbb{E}_{\mathbb{P}_n}(T_\theta)| &+ \sup_{\theta\in\Theta} |\log \mathbb{E}_{\mathbb{Q}}(e^{T_\theta}) - \log \mathbb{E}_{\mathbb{Q}_n}(e^{T_\theta})| \\ &+ \sup_{\theta\in\Theta} d(|\log \mathbb{E}_{\mathbb{Q}}(e^{T_\theta}) - \log \mathbb{E}_{\mathbb{Q}_n}(e^{T_\theta})|, 0). \qquad (35) \end{aligned}$$

Since the function T_θ is uniformly bounded by a constant M and $\log$ is Lipschitz continuous with constant e^M, we have

$$|\log \mathbb{E}_{\mathbb{Q}}(e^{T_\theta}) - \log \mathbb{E}_{\mathbb{Q}_n}(e^{T_\theta})| \leq e^{M} |\mathbb{E}_{\mathbb{Q}}(e^{T_\theta}) - \mathbb{E}_{\mathbb{Q}_n}(e^{T_\theta})| \tag{36}$$

and

$$d(|\log \mathbb{E}_{\mathbb{Q}}(e^{T_\theta}) - \log \mathbb{E}_{\mathbb{Q}_n}(e^{T_\theta})|, 0) \leq d(e^{M}, 0) \cdot d(|\mathbb{E}_{\mathbb{Q}}(e^{T_\theta}) - \mathbb{E}_{\mathbb{Q}_n}(e^{T_\theta})|, 0). \tag{37}$$

Since Θ is compact and the feedforward network function is continuous, T_θ and e^{T_θ} satisfy the uniform law of large numbers (Belghazi et al., 2018). Given $\epsilon > 0$, we can thus choose $N \in \mathbb{N}$ such that $\forall n \geq N$ and with probability 1,

$$\sup_{\theta\in\Theta} |\mathbb{E}_{\mathbb{P}}(T_\theta) - \mathbb{E}_{\mathbb{P}_n}(T_\theta)| \leq \frac{\eta}{3}, \tag{38}$$

$$\sup_{\theta\in\Theta} |\mathbb{E}_{\mathbb{Q}}(e^{T_\theta}) - \mathbb{E}_{\mathbb{Q}_n}(e^{T_\theta})| \leq e^{-M}\frac{\eta}{3}, \tag{39}$$

$$\sup_{\theta\in\Theta} d(|\mathbb{E}_{\mathbb{Q}}(e^{T_\theta}) - \mathbb{E}_{\mathbb{Q}_n}(e^{T_\theta})|, 0) \leq \frac{1}{d(e^{M}, 0)}\frac{\eta}{3}. \tag{40}$$

Hence, this leads to

$$|\hat{I}_{\text{ReMINE}}(X,Y)_n - \hat{I}_{\text{ReMINE}}(X,Y)| \leq \frac{\eta}{3} + \frac{\eta}{3} + \frac{\eta}{3} = \eta. \tag{41}$$

□

Theorem. *ReMINE is strongly consistent.*

Proof. Let $\epsilon > 0$. We apply **??** and **??** to find a neural network function T_θ and $N \in \mathbb{N}$ such that **??** and **??** hold with $\eta = \epsilon/2$. By the triangular inequality, for all $n \geq N$ and with probability one, we have:

$$\begin{aligned} |I(X,Y) - \hat{I}_{\text{ReMINE}}(X,Y)_n| &= |I_{\text{ReMINE}}(X,Y) - \hat{I}_{\text{ReMINE}}(X,Y)_n| \quad (\because \text{Theorem 1}) \\ &\leq |I_{\text{ReMINE}}(X,Y) - \hat{I}_{\text{ReMINE}}(X,Y)| + |\hat{I}_{\text{ReMINE}}(X,Y)_n - \hat{I}_{\text{ReMINE}}(X,Y)| \leq \epsilon \qquad (42) \end{aligned}$$

which proves the consistency.

□

Sample complexity proof

Theorem. *Assume that the function T_θ are M-bounded and $\mathcal{L}$-lipschitz with respect to the parameter θ. The domain θ is bounded, so that $||\theta|| \leq K$ for some constant K. When using k mini-batches to estimate MI, we have*

$$\mathcal{P}(|\hat{I}_{\textit{ReMINE}}(X, Y) - I(X, Y)| \leq \epsilon) \geq 1 - \delta \tag{43}$$

whenever the number of samples n for each batch satisfies

$$n \geq \frac{2M^2(d\log(24KL\sqrt{d}/\epsilon) + 2dM + log(2/\delta))}{\epsilon^2 k}. \tag{44}$$

Proof. As the optimal T^* of I_{ReMINE} is also the solution of I_{MINE}, we can use the same proof process of the Theorem 6 in (Belghazi et al., 2018). Contrast to MINE (Belghazi et al., 2018), we start from $\mathcal{P}(|\mathbb{E}_{\mathbb{Q}}[f] - \mathbb{E}_{\hat{\mathbb{Q}}}[f]| > \epsilon/6) \leq 2\exp(\frac{-\epsilon^2 nk}{2M^2})$ by the Hoeffding inequality, because we use $n \cdot k$ samples and our loss function consists of three terms including the regularization term. □

A.4 MATHEMATICAL PROPERTIES OF $I_{\textbf{ReNWJ}}$

Consistency Proof We show the proof of the consistency for the ReNWJ based estimator. Same to the proof of ReMINE consistency, we assume that the input space of the functions below is a compact domain, and all measures are absolutely continuous with respect to the Lebesgue measure. We will also restrict to families of feedforward functions with continuous activations, with a single output neuron. We provide a proof for the case where $d(\cdot, \cdot)$ is the log-Euclidean distance in this subsection.

Lemma 4. *(Approximation) Let η > 0. There exists a neural network function T_θ with parameters $\theta \in \Theta$ such that*

$$|\hat{I}_{\textit{ReNWJ}}(X, Y) - I_{\textit{ReNWJ}}(X, Y)| \leq \eta \tag{45}$$

where

$$\hat{I}_{\textit{ReNWJ}}(X, Y) = \sup_{\theta \in \Theta} \mathbb{E}_{\mathbb{P}}(T_\theta) - \mathbb{E}_{\mathbb{Q}}(e^{T_\theta - 1}) - d(\mathbb{E}_{\mathbb{Q}}(e^{T_\theta - 1}), 1). \tag{46}$$

Proof. Without loss of generality, we set $T^* = \log \frac{d\mathbb{P}}{d\mathbb{Q}} + 1$. By construction, T^* satisfies

$$\mathbb{E}_{\mathbb{P}}(T^*) = 1 + I(X, Y), \quad \mathbb{E}_{\mathbb{Q}}(e^{T^* - 1}) = 1. \tag{47}$$

For a function T,

$$\begin{aligned}
&I_{\text{ReNWJ}}(X, Y) - \hat{I}_{\text{ReNWJ}}(X, Y) &(48)\\
&\leq \mathbb{E}_{\mathbb{P}}(T^* - T) + \mathbb{E}_{\mathbb{Q}}(e^{T-1}) - \mathbb{E}_{\mathbb{Q}}(e^{T^*-1}) + d(\mathbb{E}_{\mathbb{Q}}(e^{T-1}), 1) - d(\mathbb{E}_{\mathbb{Q}}(e^{T^*-1}), 1) &(49)\\
&\leq \mathbb{E}_{\mathbb{P}}(T^* - T) + \mathbb{E}_{\mathbb{Q}}(e^{T-1} - e^{T^*-1}) + d(\mathbb{E}_{\mathbb{Q}}(e^{T-1}), \mathbb{E}_{\mathbb{Q}}(e^{T^*-1})) &(50)\\
&\leq \mathbb{E}_{\mathbb{P}}(T^* - T) + e^{-1}\mathbb{E}_{\mathbb{Q}}(e^{T} - e^{T^*}) + d(\mathbb{E}_{\mathbb{Q}}(e^{T-1}), 1) &(51)
\end{aligned}$$

where $d(\cdot, \cdot)$ is the log-Euclidean distance on $\mathbb{R}$. Fix $\eta > 0$. By the universal approximation theorem, we may choose a feedforward network function $T_\theta \leq M$ with $M > 1$ such that

$$\mathbb{E}_{\mathbb{P}}|T^* - T_\theta| \leq \frac{\eta}{3}, \quad \mathbb{E}_{\mathbb{Q}}|T^* - T_\theta| \leq \frac{\eta}{3}e^{1-M}, \quad \text{and } d(\mathbb{E}_{\mathbb{Q}}(e^{T_\theta}), e) \leq \frac{\eta}{3}. \tag{52}$$

Since $\exp$ is Lipschitz continuous with constant e^M on $(-\infty, M]$, we have

$$\mathbb{E}_{\mathbb{Q}}|e^{T^*} - e^{T_\theta}| \leq e^M \mathbb{E}_{\mathbb{Q}}|T^* - T_\theta| \leq \frac{\eta}{3}e. \tag{53}$$

And

$$d(\mathbb{E}_{\mathbb{Q}}(e^{T-1}), 1) = d(\mathbb{E}_{\mathbb{Q}}(e^{T_\theta}), \mathbb{E}_{\mathbb{Q}}(e^{T^*})) \leq d(\mathbb{E}_{\mathbb{Q}}(e^{T_\theta}), e) \leq \frac{\eta}{3}. \tag{54}$$

From **??**, **??**, **??** and the triangular inequality, we then obtain

$$|\hat{I}_{\text{ReNWJ}}(X,Y) - I_{\text{ReNWJ}}(X,Y)| < \eta. \tag{55}$$

□

Lemma 5. *(Estimation) Let $\eta > 0$. Given a neural network function T_θ with parameters $\theta \in \Theta$, there exists $N \in \mathbb{N}$ such that*

$$\forall n \geq N, \mathcal{P}(|\hat{I}_{\mathit{ReNWJ}}(X,Y)_n - \hat{I}_{\mathit{ReNWJ}}(X,Y)| \leq \eta) = 1, \tag{56}$$

where $\hat{I}_{\mathit{ReNWJ}}(X,Y)_n$ is the ReNWJ representation which is empirically obtained by n samples.

Proof. We start by using the triangular inequality to write,

$$|\hat{I}_{\text{ReNWJ}}(X,Y)_n - \sup_{\theta\in\Theta} \hat{I}_{\text{ReNWJ}}(T_\theta)| \leq \sup_{\theta\in\Theta} |\mathbb{E}_{\mathbb{P}}(T_\theta) - \mathbb{E}_{\mathbb{P}_n}(T_\theta)| + \sup_{\theta\in\Theta} |\mathbb{E}_{\mathbb{Q}}(e^{T_\theta - 1}) - \mathbb{E}_{\mathbb{Q}_n}(e^{T_\theta - 1})| + \sup_{\theta\in\Theta} d(\mathbb{E}_{\mathbb{Q}}(e^{T_\theta - 1}), \mathbb{E}_{\mathbb{Q}_n}(e^{T_\theta - 1})). \tag{57}$$

Since Θ is compact and the feedforward network T_θ is continuous and uniformly bounded by a constant M, T_θ and e^{T_θ} satisfy the uniform law of large numbers (Belghazi et al., 2018). Given $\epsilon > 0$, we can thus choose $N \in \mathbb{N}$ such that $\forall n \geq N$ and with probability 1,

$$\sup_{\theta\in\Theta} |\mathbb{E}_{\mathbb{P}}(T_\theta) - \mathbb{E}_{\mathbb{P}_n}(T_\theta)| \leq \frac{\eta}{3}, \tag{58}$$

$$\sup_{\theta\in\Theta} e^{-1} |\mathbb{E}_{\mathbb{Q}}(e^{T_\theta}) - \mathbb{E}_{\mathbb{Q}_n}(e^{T_\theta})| \leq \frac{\eta}{3} e^{-M}, \tag{59}$$

$$\sup_{\theta\in\Theta} d(\frac{\mathbb{E}_{\mathbb{Q}}(e^{T_\theta})}{\mathbb{E}_{\mathbb{Q}_n}(e^{T_\theta})}, 1) \leq \frac{\eta}{3}. \tag{60}$$

Hence, this leads to

$$|\hat{I}_{\text{ReNWJ}}(X,Y)_n - \hat{I}_{\text{ReNWJ}}(X,Y)| \leq \frac{\eta}{3} + \frac{\eta}{3} + \frac{\eta}{3} = \eta. \tag{61}$$

□

Theorem. *The ReNWJ estimator is strongly consistent.*

Proof. Let $\epsilon > 0$. We apply **??** and **??** to find a neural network function T_θ and $N \in \mathbb{N}$ such that **??** and **??** hold with $\eta = \epsilon/2$. By the triangular inequality, for all $n \geq N$ and with probability one, we have

$$\begin{aligned} |I(X,Y) - \hat{I}_{\text{ReNWJ}}(X,Y)_n| &= |I_{\text{ReNWJ}}(X,Y) - \hat{I}_{\text{ReNWJ}}(X,Y)_n| \quad (\because \text{ReNWJ representation}) \\ &\leq |I_{\text{ReNWJ}}(X,Y) - \hat{I}_{\text{ReNWJ}}(X,Y)| + |\hat{I}_{\text{ReNWJ}}(X,Y)_n - \hat{I}_{\text{ReNWJ}}(X,Y)| \leq \epsilon \end{aligned} \tag{62}$$

which proves the consistency. □

Sample complexity proof

Theorem. *Assume that the function $1 \leq |T_\theta| < M$ is $\mathcal{L}$-lipschitz with respect to the parameter θ. The domain θ is bounded, so that $||\theta|| \leq K$ for some constant K. When using k mini-batches to estimate MI and $d(x,1) \leq |x-1|$, we have*

$$\mathcal{P}(|\hat{I}_{\mathit{ReNWJ}}(X,Y) - I(X,Y)| \leq \epsilon) \geq 1 - \delta \tag{63}$$

whenever the number of samples n for each batch satisfies

$$n \geq \frac{2M^2(d\log(24KL\sqrt{d}/\epsilon) + 2dM + log(2/\delta))}{\epsilon^2 k}. \tag{64}$$

Proof. By taking the assumptions of **??**, we begin with **??**, **??** and **??**. By the Hoeffding inequality, for all function f,

$$\mathcal{P}(|\mathbb{E}_{\mathbb{Q}}[f] - \mathbb{E}_{\hat{\mathbb{Q}}}[f]| > \epsilon/6) \leq 2\exp(\frac{-\epsilon^2(n \cdot k)}{2M^2}). \tag{65}$$

To extend this inequality to a uniform inequality over all functions T_θ and e^{T_θ}, we choose a minimal cover of the domain $\Theta \subset \mathbb{R}^d$ by a finite set of small balls of radius η, $\Theta \subset \cup_j B_\eta(\theta_j)$, and the union bound. The minimal cardinality of such covering is bounded by the covering number $N_\eta(\Theta)$ of Θ,

$$N_\eta(\Theta) \leq \left(\frac{2K\sqrt{d}}{\eta}\right)^d. \tag{66}$$

Successively applying a union bound in **??** with the set of functions $\{T_{\theta_j}\}_j$, and $\{e^{T_{\theta_j}}\}_j$, We have

$$\mathcal{P}\left(max_j|\mathbb{E}_{\mathbb{Q}}(T_{\theta_j}) - \mathbb{E}_{\hat{\mathbb{Q}}}(T_{\theta_j})| \geq \frac{\epsilon}{6}\right) \leq 2N_\eta(\Theta)\exp(-\frac{\epsilon^2(n \cdot k)}{2M^2}), \tag{67}$$

$$\mathcal{P}\left(max_j|\mathbb{E}_{\mathbb{Q}}(e^{T_{\theta_j}}) - \mathbb{E}_{\hat{\mathbb{Q}}}(e^{T_{\theta_j}})| \geq \frac{\epsilon}{6}\right) \leq 2N_\eta(\Theta)\exp(-\frac{\epsilon^2(n \cdot k)}{2M^2}). \tag{68}$$

We now choose that ball radius to be $\eta = \frac{\epsilon}{12L}e^{-2M}$. Solving for n the inequation,

$$2N_\eta(\Theta)\exp(-\frac{\epsilon^2 n}{2M^2}) \leq \delta, \tag{69}$$

we deduce from **??** that, whenever **??** holds, with probability at least $1 - \delta$, for all $\theta \in \Theta$,

$$\begin{aligned}|\mathbb{E}_{\mathbb{Q}}(T_\theta) - \mathbb{E}_{\hat{\mathbb{Q}}}(T_\theta)| &\leq |\mathbb{E}_{\mathbb{Q}}(T_\theta) - \mathbb{E}_{\mathbb{Q}}(T_{\theta_j})| + |\mathbb{E}_{\mathbb{Q}}(T_{\theta_j}) - \mathbb{E}_{\hat{\mathbb{Q}}}(T_{\theta_j})| + |\mathbb{E}_{\hat{\mathbb{Q}}}(T_{\theta_j}) - \mathbb{E}_{\hat{\mathbb{Q}}}(T_\theta)| \\ &\leq \frac{\epsilon}{12}e^{-2M} + \frac{\epsilon}{6} + \frac{\epsilon}{12}e^{-2M} < \frac{\epsilon}{3}.\end{aligned} \tag{70}$$

Similarly, using **??**, we get that with probabilty at least $1 - \delta$,

$$|\mathbb{E}_{\mathbb{Q}}(e^{T_\theta - 1}) - \mathbb{E}_{\hat{\mathbb{Q}}}(e^{T_\theta - 1})| \leq \frac{\epsilon}{3} < e \cdot \frac{\epsilon}{3}. \tag{71}$$

Hence,

$$\begin{aligned}|\hat{I}_{\text{ReNWJ}}(X,Y) - I(X,Y)| &\leq |\mathbb{E}_{\mathbb{Q}}(T_{\theta_j}) - \mathbb{E}_{\hat{\mathbb{Q}}}(T_{\theta_j})| + |\mathbb{E}_{\mathbb{Q}}(e^{T_{\theta_j}-1}) - \mathbb{E}_{\hat{\mathbb{Q}}}(e^{T_{\theta_j}-1})| + d(\mathbb{E}_{\mathbb{Q}}(e^{T_{\theta_j}}), \mathbb{E}_{\hat{\mathbb{Q}}}(e^{T_{\theta_j}})) \\ &\leq |\mathbb{E}_{\mathbb{Q}}(T_{\theta_j}) - \mathbb{E}_{\hat{\mathbb{Q}}}(T_{\theta_j})| + e^{-1}|\mathbb{E}_{\mathbb{Q}}(e^{T_{\theta_j}}) - \mathbb{E}_{\hat{\mathbb{Q}}}(e^{T_{\theta_j}})| + |\mathbb{E}_{\mathbb{Q}}(e^{T_{\theta_j}}) - \mathbb{E}_{\hat{\mathbb{Q}}}(e^{T_{\theta_j}})| \leq \epsilon.\end{aligned} \tag{72}$$

□

A.5 THE PROPERTY OF MI ESTIMATORS

The variance of the exponential value of the statistic network's output according to the bias of optimal functions on the distribution $\mathbb{Q}$.

Theorem. *Let $Q^{(n)}$ be the empirical distributions of n i.i.d. samples from $\mathbb{Q}$. For the optimal $T_1 = \log\frac{dp}{dq} + C_1$ and $T_2 = \log\frac{dp}{dq} + C_2$ where $C_1 \geq C_2$,*

$$Var_{\mathbb{Q}}(\mathbb{E}_{\mathbb{Q}^{(n)}}(e^{T_1})) \geq Var_{\mathbb{Q}}(\mathbb{E}_{\mathbb{Q}^{(n)}}(e^{T_2})). \tag{73}$$

Proof. Consider that

$$\mathrm{Var}_{\mathbb{Q}}(e^{T_1}) = e^{2C_1}\left(\mathbb{E}_{\mathbb{Q}}((\frac{d\mathbb{P}}{d\mathbb{Q}})^2) - (\mathbb{E}_{\mathbb{Q}}(\frac{d\mathbb{P}}{d\mathbb{Q}}))^2\right), \tag{74}$$

and

$$\mathrm{Var}_{\mathbb{Q}}(e^{T_2}) = e^{2C_2}\left(\mathbb{E}_{\mathbb{Q}}((\frac{d\mathbb{P}}{d\mathbb{Q}})^2) - (\mathbb{E}_{\mathbb{Q}}(\frac{d\mathbb{P}}{d\mathbb{Q}}))^2\right). \tag{75}$$

By Song and Ermon (2020), the variance of the mean of n i.i.d. random variable then gives us

$$\mathrm{Var}_{\mathbb{Q}}(\mathbb{E}_{\mathbb{Q}^{(n)}}(e^{T_1})) = \frac{\mathrm{Var}_{\mathbb{Q}}(e^{T_1})}{n}, \mathrm{Var}_{\mathbb{Q}}(\mathbb{E}_{\mathbb{Q}^{(n)}}(e^{T_2})) = \frac{\mathrm{Var}_{\mathbb{Q}}(e^{T_2})}{n}. \tag{76}$$

Since $e^x \geq 1$ for all $x \geq 0$,

$$\frac{\mathrm{Var}_{\mathbb{Q}}(\mathbb{E}_{\mathbb{Q}^{(n)}}(e^{T_1}))}{\mathrm{Var}_{\mathbb{Q}}(\mathbb{E}_{\mathbb{Q}^{(n)}}(e^{T_2}))} = \frac{\frac{\mathrm{Var}_{\mathbb{Q}}(e^{T_1})}{n}}{\frac{\mathrm{Var}_{\mathbb{Q}}(e^{T_2})}{n}} = e^{2(C_1-C_2)} \geq 1. \tag{77}$$

Therefore, the variance of T_1 is equal to or less than the that of T_2 on $\mathbb{Q}$. □

Proof of estimation bias caused by drifting

Theorem. *When used on DV representation, the two averaging strategies below produce a biased MI estimate if the drifting problem occurs.*

1. *Macro-averaging (similar to that of Poole et al. (2019)): Establish a single estimate through the average of estimated MI from each batch.*
2. *Micro-averaging: Calculate the DV representation using the average of the each individual network outputs.*

Proof. We start from the definition of I_{DV}, where

$$I_{\mathrm{DV}}(X, Y) = \mathbb{E}_{\mathbb{P}}(T(x, y)) - \log(\mathbb{E}_{\mathbb{Q}}(e^{T(x,y)})) \tag{78}$$

becomes the objective function to estimate MI, i.e. MINE.

Let $T_{ij}^{(J)}$ and $T_{ij}^{(M)}$ denote the ij-th element of outputs for $\mathbb{P}_m$ and $\mathbb{Q}_n$ respectively, where i is the index of batch and j is the index of sample inside the batch, and the non-drifting output as T_{ij}^*, and the drifting constant for each batch C_i. Then, $T_{ij} = T_{ij}^* + C_i$.

When the number of batch is B and each batch size is N,

1. Macro averaging:

$$\frac{1}{B}\Sigma_i[\frac{1}{N}\Sigma_j T_{ij}^{(J)} - \log(\frac{1}{N}\Sigma_j e^{T_{ij}^{(M)}})] \tag{79}$$

$$=\frac{1}{B}\Sigma_i[\frac{1}{N}\Sigma_j (T_{ij}^{(J*)} + C_i) - \log(\frac{1}{N}\Sigma_j e^{T_{ij}^{(M*)}+C_i})] \tag{80}$$

$$=\frac{1}{B}\Sigma_i[\frac{1}{N}\Sigma_j (T_{ij}^{(J*)} + C_i) - \log(\frac{1}{N}e^{C_i}\Sigma_j e^{T_{ij}^{(M*)}})] \tag{81}$$

$$=\frac{1}{B}\Sigma_i[\frac{1}{N}\Sigma_j T_{ij}^{(J*)} - \log(e^{-C_i}\frac{1}{N}e^{C_i}\Sigma_j e^{T_{ij}^{(M*)}})] \tag{82}$$

$$=\frac{1}{B}\Sigma_i[\frac{1}{N}\Sigma_j T_{ij}^{(J*)} - \log(\frac{1}{N}\Sigma_j e^{T_{ij}^{(M*)}})] \tag{83}$$

$$=\frac{1}{NB}\Sigma_{ij} T_{ij}^{(J*)} - \frac{1}{B}\Sigma_i[\log(\frac{1}{N}\Sigma_j e^{T_{ij}^{(M*)}})] \tag{84}$$

$$\neq\frac{1}{NB}\Sigma_{ij} T_{ij}^{(J*)} - \log(\frac{1}{NB}\Sigma_{ij} e^{T_{ij}^{(M*)}}) \tag{85}$$

2. Micro averaging:

$$\begin{aligned} & \frac{1}{NB}\Sigma_{ij}T_{ij}^{(J)} - \log(\frac{1}{NB}\Sigma_{ij}e^{T_{ij}^{(M)}}) && (86)\\ =& \frac{1}{NB}\Sigma_{ij}(T_{ij}^{(J*)} + C_i) - \log(\frac{1}{NB}\Sigma_{ij}e^{(T_{ij}^{(M*)}+C_i)}) && (87)\\ =& \frac{1}{NB}\Sigma_{ij}T_{ij}^{(J*)} - \log[(\frac{1}{NB}\Sigma_{ij}e^{(T_{ij}^{(M*)}+C_i)})^{\frac{1}{B}\Sigma_i C_i}] && (88)\\ \neq& \frac{1}{NB}\Sigma_{ij}T_{ij}^{(J*)} - \log(\frac{1}{NB}\Sigma_{ij}e^{T_{ij}^{(M*)}}) && (89) \end{aligned}$$

□

We emphasize that we have to stop the drifting via the regularization term of ReDV.

Wrong estimation derived from biased values According to the theorem above, the MI estimate derived from the average of the values estimated from the mini-batch in DV representation-based estimators will lead to erroneous results. However, the micro-averaging strategy is often used to measure the performance of MI estimators (MINE or InfoNCE), as shown in Fig. 6 of Cheng et al. (2020).

A.6 THE PROOF FOR THE VALIDITY OF OUR BENCHMARK

We assume that the dataset used for our benchmark satisfies the single label assumption where there exists exactly one label for every sample inside the dataset. Note that the assumption implies that $p(y|x) = 1$. In other words, we assume statistical dependence between X and Y (Tishby and Zaslavsky, 2015).

Theorem. *(Supervised Learning Benchmark) Consider a dataset $D = (X, Y)$ where Y is the label for sample X, and $H(Y)$ is the entropy of Y.*

$$I(X, Y) = H(Y) \tag{90}$$

Proof.

$$\begin{aligned} I(X;Y) &= \int_{X,Y} P(X,Y)\log\frac{P(X,Y)}{P(X)P(Y)} && (91)\\ &= \int_x\int_y P(x,y)\log\frac{P(y|x)}{P(y)}dydx && (92)\\ &= \int_x\int_y P(x)P(y|x)\log\frac{P(y|x)}{P(y)}dydx && (93)\\ &= \int_x P(x)\left(\int_y P(y|x)\log\frac{P(y|x)}{P(y)}dy\right)dx && (94)\\ &= \int_R P(x^*)\log\frac{1}{P(y^*)} \quad \text{(where } R \text{ is the region where } y^* \text{ is a correct label for the given } x^*\text{)} && (95)\\ &= \sum_c \int_{R_c} P(x^*, c)\log\frac{1}{P(c)} \quad \text{(where } R \text{ is partitioned by the label } c \text{ to yield } R_c\text{)} && (96)\\ &= \sum_c \log\frac{1}{P(c)}\int_{R_c} P(x^*, c) \quad (\because P(c) \text{ is constant inside the } R_c) && (97)\\ &= \sum_c \log\frac{1}{P(c)}P(c) \quad (\because \int_{R_c} P(x^*, c) = P(c)\text{, i.e., marginalization)} && (98)\\ &= H(Y) && (99) \end{aligned}$$

□

Theorem. *(Contrastive Learning Benchmark) Consider a dataset $D = (X, Y)$. Let X_1 be a sample drawn from the dataset with the label Y and X_2 be another sample drawn from the subset of D where all the samples inside the subset are with the same label Y. Assume that D also satisfies the single label assumption.*

$$I(X_1, X_2) = I(X_1, Y) = I(X_2, Y) = H(Y) \tag{100}$$

Proof.

$$\begin{aligned}
P(X_1, X_2) &= \sum_{y_i} P(X_1, X_2, Y) \quad (\because \text{marginalization}) \\
&= \sum_{y_i} P(X_1)P(Y|X_1)P(X_2|Y, X_1) \quad (\because \text{factorization}) \\
&= \sum_{y_i} P(Y)P(X_1|Y)P(X_2|Y) \quad (\because X_1 \text{ and } X_2 \text{ are independent for given } Y) \\
&= \sum_{i} P(y_i)P(X_1|y_i)P(X_2|y_i)
\end{aligned}$$

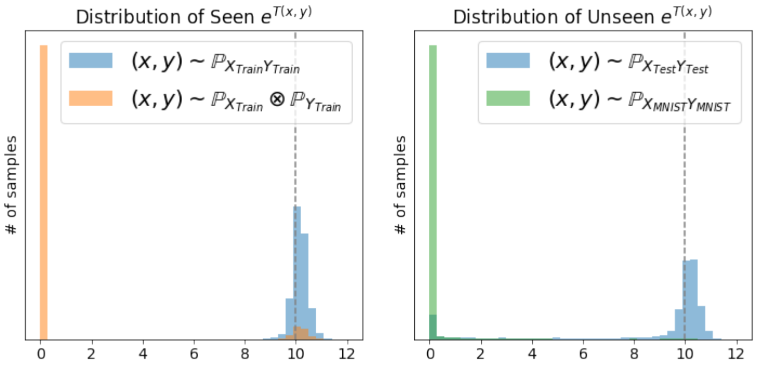
$$\begin{aligned}
P(X_1) &= \sum_{y_i} P(X_1, Y) = \sum_{y_i} P(Y)P(X_1|Y) = \sum_{i} P(y_i)P(X_1|y_i) \\
P(X_2) &= \sum_{y_i} P(X_2, Y) = \sum_{y_i} P(Y)P(X_2|Y) = \sum_{i} P(y_i)P(X_2|y_i)
\end{aligned}$$

$$\begin{aligned}
\frac{P(X_1, X_2)}{P(X_1)P(X_2)} &= \frac{\sum_i P(y_i)P(X_1|y_i)P(X_2|y_i)}{\sum_i P(y_i)P(X_1|y_i) \sum_{y_i} P(y_i)P(X_2|y_i)} \\
&= \frac{\sum_i P(y_i)P(X_1|y_i)P(X_2|y_i)}{\sum_i P(y_i)^2 P(X_1|y_i)P(X_2|y_i)} \quad (\because X_1 \text{ and } X_2 \text{ has the same label})
\end{aligned}$$

Let R_i be the region where (X, y_i) such as i-th class label y_i is a correct label for the given X_1.

$$\begin{aligned}
I(X_1, X_2) &= \int_{X_1, X_2} P(X_1, X_2) \log \frac{P(X_1, X_2)}{P(X_1)P(X_2)} \\
&= \int_{X_1, X_2} \left(\sum_i P(y_i)P(X_1|y_i)P(X_2|y_i) \right) \log \frac{\sum_i P(y_i)P(X_1|y_i)P(X_2|y_i)}{\sum_i P(y_i)^2 P(X_1|y_i)P(X_2|y_i)} \\
&= \sum_i P(y_i) \int_{X_2} P(X_2|y_i) \left(\int_{R_i} P(X_1|y_i) \log \frac{P(y_i)P(X_1|y_i)P(X_2|y_i)}{P(y_i)^2 P(X_1|y_i)P(X_2|y_i)} dx_1 \right) dx_2 \\
&= \sum_i P(y_i) \int_{X_2} P(X_2|y_i) \left(\int_{R_i} P(X_1|y_i) \log \frac{1}{P(y_i)} dx_1 \right) dx_2 \\
&= \sum_i P(y_i) \log \frac{1}{P(y_i)} \int_{X_2} P(X_2|y_i) \left(\int_{R_i} P(X_1|y_i) dx_1 \right) dx_2 \\
&= \sum_i P(y_i) \log \frac{1}{P(y_i)} \\
&= H(Y)
\end{aligned}$$

□



(a) (b)

Figure 1: Histogram of the exponential of the network outputs $e^{T(x,y)}$ which is trained with CLB CIFAR10. Training samples and unseen samples are fed to (a) and (b), respectively.

B DIRECTLY UTILIZING THE STATISTICS NETWORK OUTPUTS FOR OUT-OF-DISTRIBUTION TASK

We observe the SLB CIFAR10-trained network outputs when seen or unseen samples are fed to the statistics network T in **??**. Note that we can take $e^{T(x,y)} = \frac{d\mathbb{P}_{XY}}{d\mathbb{P}_X \otimes \mathbb{P}_Y}$ for granted, thanks to regularization. **??** (a) shows the distribution of $e^{T(x,y)}$ for the training set samples $(x, y) \sim \mathbb{P}_{X_{\text{Train}}Y_{\text{Train}}}$. As 90% of $(x, y) \sim \mathbb{P}_X \otimes \mathbb{P}_Y$ is wrongly labeled, the majority yields $e^{T(x,y)} = 0$. The likelihood ratio for the $(x, y) \sim \mathbb{P}_{XY}$ is 10, and all the samples are centered around the ideal value as expected. CIFAR10 test set samples $(x, y) \sim \mathbb{P}_{X_{\text{Test}}Y_{\text{Test}}}$ also yield similar results, where some of the samples are wrongly positioned, being the test error of T. Surprisingly, when we feed MNIST (LeCun et al., 1998) training samples $(x, y) \sim \mathbb{P}_{X_{\text{MNIST}}Y_{\text{MNIST}}}$, model successfully classifies nearly all the samples to be less likely to occur in $\mathbb{P}_{X_{\text{Train}}Y_{\text{Train}}}$. This implies that exploiting the network outputs with the viewpoints of MI may show usefulness in out-of-distribution detection.

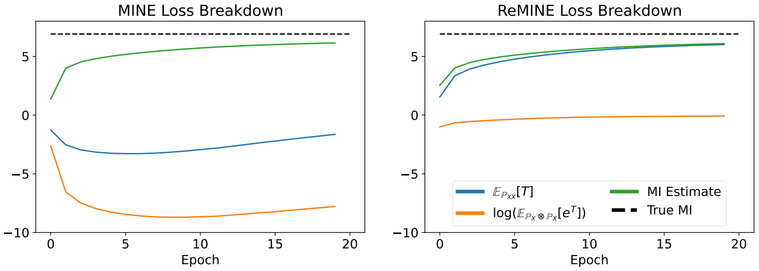


Figure 2: Training T_θ using I_{MINE} and I_{ReMINE} with batch size 100 for 20 epochs. We breakdown the MI loss into two components. We split both losses into first term $\mathbb{E}_{\mathbb{P}_{XX}}(T)$ and second term $\log \mathbb{E}_{\mathbb{P}_X \otimes \mathbb{P}_X}(e^T)$.

C EXPERIMENTS ON IMAGENET

We test on the ImageNet dataset with 1000 classes, where we use the batch size of 100. We set the batch size to be relatively small to observe how different losses behave, whereas multiple contrastive learning literature such as Chen et al. (2020); He et al. (2020) uses large batch sizes to avoid instability. We train for 20 epochs to observe the early stages of training.

First, we can observe in **??** that the regularizer successfully solves the drifting problem of I_{MINE}. Also, **??** shows that I_{NWJ} fails in the contrastive learning benchmark. I_{NWJ} explodes within a few steps of training, where the regularizer successfully avoids the problem to yield a feasible output. Note that we did not observe the losses till convergence; we have to train much longer to obtain a more accurate performance of MI estimation and test accuracy. However, we can see that in the supervised

Task	Loss	MI Estimation		Test Accuracy	
		Original	Regularized	Original	Regularized
Supervised Learning Benchmark	CE	-	-	0.0795	-
	MINE	6.147	6.110	0.1056	0.1081
	NWJ	6.072	6.075	0.1020	0.1005
Contrastive Learning Benchmark	MINE	1.095	1.140	0.0103	0.0098
	NWJ	0.000	1.008	0.0010	0.0072

Table 1: Our supervised and contrastive learning benchmark results on ImageNet dataset. We provide the MI estimation and test accuracy, where we clip the negative MI estimations to 0. We compare the performance of original and regularized loss. We also add the accuracy of standard cross-entropy loss (CE) for comparison. Similar to Section 5.2, we choose the regularization weight $\lambda \in \{0.1, 0.01, 0.001\}$ that shows the best MI estimation results.

learning benchmark, which is the relatively easier benchmark, all the losses are already close to the optimal MI even in the earlier epochs. We can also observe a similar trade-off between the MI estimation and test accuracy in **??**. Future works on large-scale datasets are needed to observe the behaviors further.

D EXPERIMENTAL DETAILS

In this section, we provide the experiment details in the manuscript with the accompanying code `https://github.com/Siyeong-Lee/Deconstructing-MINE`.

D.1 HARDWARE SPECIFICATION

We use a single NVIDIA DGX A100 machine with 8 GPUs for all the experiments. All the experiments except for our benchmark experiments take less than 10 minutes and a single GPU to compute. It takes less than 2 days to compute all the benchmark experiments: 4 settings, 12 losses, and 5 seeds running on 8 GPUs and 4 processes per GPU.

D.2 DETAILED SETTINGS FOR ONE-HOT DATASET EXPERIMENTS

We describe the detailed settings for Fig. 1, Fig. 2, Fig. 3, Fig. 4, and Fig. 5. We choose $N = 16$ for the one-hot discrete dataset $X \sim U(1, N)$. We use a simple statistics network T with a concatenated vector of dimension $N \times 2 = 32$ as input. We pass the input through two fully connected layers with ReLU activation by widths: $32 - 256 - 1$. The last layer outputs a single scalar with no bias and activation. We use stochastic gradient descent (SGD) with learning rate 0.1 to optimize the statistics network unless specified.

D.3 DETAILED SETTINGS FOR OUR BENCHMARK

We describe the detailed settings for Table 2 and Fig. 6. We use ResNet-18 (He et al., 2016) as the backbone network and use Adam optimizer with the default learning rate 0.001, $\beta_1 = 0.9$ and $\beta_2 = 0.999$. We use batch size 100 for CIFAR100 and 10 for CIFAR10. We train for different epochs per each benchmark: 40 epochs (SLB CIFAR10), 100 epochs (SLB CIFAR100), 100 epochs (CLB CIFAR10), and 150 epochs (CLB CIFAR100). We choose enough number of epochs for all the losses to be fully converged for each of the benchmarks. We rerun the same experiment 5 times with different seeds.

D.4 DETAILED SETTINGS FOR THE 20D CORRELATED GAUSSIAN TASK

We describe the detailed settings for Fig. 7. We sampled (x, y) from d-dimensional correlated Gaussian dataset where $X \sim N(\mathbf{0}, \mathbf{I}_d)$ and $Y \sim N(\rho X, (1 - \rho^2)\mathbf{I}_d)$ given the correlation parameter $0 \leq \rho < 1$, which is taken from Belghazi et al. (2018). The true MI for the dataset is $I(X, Y) = -\frac{d}{2} \log(1 - \rho^2)$. For the statistics network architecture, we consider the architecture similar to **??** where we concatenate the inputs (x, y) to pass through three fully connected layers with ReLU activation (excluding the output layer) by widths $40 - 256 - 256 - 1$, same as the network used in Poole et al. (2019). We used the same optimizer with **??**.