# ALICE: Towards Understanding Adversarial Learning for Joint Distribution Matching

Chunyuan Li[1], Hao Liu[2], Changyou Chen[3], Yunchen Pu[1], Liqun Chen[1], Ricardo Henao[1] Lawrence Carin[1]

http://chunyuan.li [1]Duke University [2]Nanjing University [3]University at Buffalo

## Highlights

### Main Contributions

1. Raise the non-identifiability issues in bidirectional adversarial learning
2. Propose ALICE algorithms: a conditional entropy framework to remedy the issues
3. Unify ALI/BiGAN, CycleGAN/DiscoGAN/DualGAN and Conditional GAN as joint distribution matching

## Non-identifiability issues

### Generative Adversarial Networks (GAN)

Marginal distribution matching: $p(\boldsymbol{x}) = q(\boldsymbol{x})$

### Adversarially Learned Inference (ALI)

Joint distribution matching: $p(\boldsymbol{x},\boldsymbol{z}) = q(\boldsymbol{x},\boldsymbol{z})$

**Importan details**: Universal distribution approximators for the sampling procedure of conditionals $\tilde{\boldsymbol{x}} \sim p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$ and $\tilde{\boldsymbol{z}} \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ are carried out as:

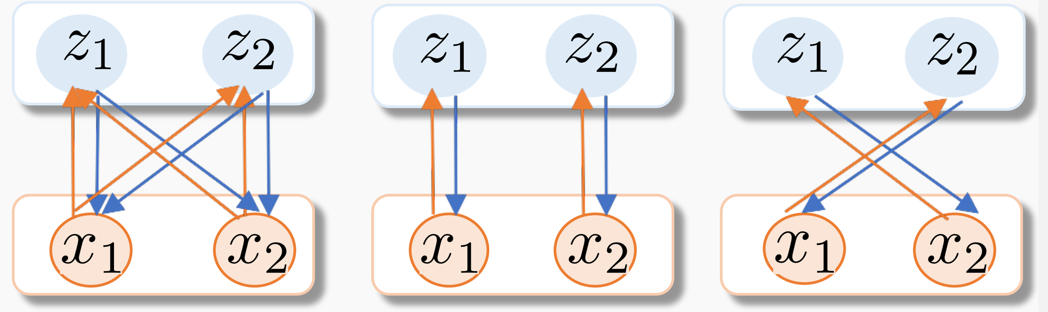
$$\tilde{\boldsymbol{x}} = g_{\boldsymbol{\theta}}(\boldsymbol{z},\boldsymbol{\epsilon}),\ \boldsymbol{z} \sim p(\boldsymbol{z}),\ \boldsymbol{\epsilon} \sim \mathcal{N}(0,\mathbf{I}),\ \text{and}$$
$$\tilde{\boldsymbol{z}} = g_{\boldsymbol{\phi}}(\boldsymbol{x},\boldsymbol{\zeta}),\ \boldsymbol{x} \sim q(\boldsymbol{x}),\ \boldsymbol{\zeta} \sim \mathcal{N}(0,\mathbf{I}),$$

Issues: The correlation between $\boldsymbol{x}$ and $\boldsymbol{z}$ is not specified.

### Problem Illustration

- In (a), for $0<\delta<1$, we can generate "realistic" $\boldsymbol{x}$ from any sample of $p(\boldsymbol{z})$, but with poor reconstruction.
- In (b) $\delta=1$ or (c) $\delta=0$, only one of the solutions will be meaningful in supervised learning.

**Any $\delta \in [0,1]$ is a valid solution of ALI ?!**




| | $z_1$ | $z_2$ |
|---|---|---|
| $x_1$ | $\delta/2$ | $(1-\delta)/2$ |
| $x_2$ | $(1-\delta)/2$ | $\delta/2$ |

(a)

| | $z_1$ | $z_2$ |
|---|---|---|
| $x_1$ | 1/2 | 0 |
| $x_2$ | 0 | 1/2 |

(b)

| | $z_1$ | $z_2$ |
|---|---|---|
| $x_1$ | 0 | 1/2 |
| $x_2$ | 1/2 | 0 |

(c)

Many applications require meaningful mappings.

1. In unsupervised learning, the inferred latent code can reconstruct its $\boldsymbol{x}$ itself with high probability. $\delta \to 1$ or $\delta \to 0$
2. In supervised learning, the task-specified correspondence between samples imposes restrictions on the mappings.

## ALICE Algorithms

### Adversarially Learned Inference with Conditional Entropy (ALICE)

$$\min_{\boldsymbol{\theta},\boldsymbol{\phi}}\max_{\boldsymbol{\omega}} \underbrace{\mathcal{L}_{\text{ALICE}}(\boldsymbol{\theta},\boldsymbol{\phi},\boldsymbol{\omega})}_{\text{Our ALICE Objective}} = \underbrace{\mathcal{L}_{\text{ALI}}(\boldsymbol{\theta},\boldsymbol{\phi},\boldsymbol{\omega})}_{\text{ALI Objective}} + \underbrace{\mathcal{L}_{\text{CE}}(\boldsymbol{\theta},\boldsymbol{\phi})}_{\text{CE Regularizer}}. \quad (1)$$

In unsupervised learning, cycle-consistency is considered to upperbound CE:

1. **Explicit cycle-consistency** Prescribed the distribution forms, *e.g.* $\ell_k$-norm
2. **Implicit cycle-consistency** Adversarially learned "perfect" reconstruction

$$\min_{\boldsymbol{\theta},\boldsymbol{\phi}}\max_{\boldsymbol{\eta}} \mathcal{L}^{\text{A}}_{\text{Cycle}}(\boldsymbol{\theta},\boldsymbol{\phi},\boldsymbol{\eta}) = \mathbb{E}_{\boldsymbol{x}\sim q(\boldsymbol{x})}[\log \sigma(f_{\boldsymbol{\eta}}(\boldsymbol{x},\boldsymbol{x}))] + \mathbb{E}_{\hat{\boldsymbol{x}}\sim p_{\boldsymbol{\theta}}(\hat{\boldsymbol{x}}|\boldsymbol{z}),\boldsymbol{z}\sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})} \log(1-\sigma(f_{\boldsymbol{\eta}}(\boldsymbol{x},\hat{\boldsymbol{x}})))]. \quad (2)$$

In semi-supervised learning, the pairwise information is leveraged to approximate CE:

1. **Explicit mapping** Prescribed the forms, $\ell_k$-norm or standard supervised losses
2. **Implicit mapping** Implicit mapping via conditional GAN

$$\min_{\boldsymbol{\theta}}\max_{\boldsymbol{\chi}} \mathcal{L}^{\text{A}}_{\text{Map}}(\boldsymbol{\theta},\boldsymbol{\chi}) = \mathbb{E}_{\boldsymbol{x},\boldsymbol{z}\sim\tilde{\pi}(\boldsymbol{x},\boldsymbol{z})}[\log \sigma(f_{\boldsymbol{\chi}}(\boldsymbol{x},\boldsymbol{z})) + \mathbb{E}_{\hat{\boldsymbol{x}}\sim p_{\boldsymbol{\theta}}(\hat{\boldsymbol{x}}|\boldsymbol{z})} \log(1-\sigma(f_{\boldsymbol{\chi}}(\hat{\boldsymbol{x}},\boldsymbol{z})))]. \quad (3)$$

## A Unified Perspective for Bivariate GANs

### ALI is equivalent to CycleGAN

CycleGAN is easier to train, as it decomposes the joint distribution matching objective (as in ALI) into four subproblems.

$$\underbrace{H^{q_\phi}(\boldsymbol{x}|\boldsymbol{z})}_{\text{Conditional entropy}} + \underbrace{\mathbb{E}_{q_\phi(\boldsymbol{z})}[\text{KL}(q_\phi(\boldsymbol{x}|\boldsymbol{z})\|p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}))]}_{\text{Conditional distribution matching}} = \underbrace{-\mathbb{E}_{q_\phi(\boldsymbol{x},\boldsymbol{z})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})]}_{\text{Cycle consistency}} \quad (4)$$

### Stochastic mapping *vs.* Deterministic mapping

Deterministic mappings imply cycle-consistency in theory (as in BiGAN), but have practical difficulties. When cycle-consistency is satisfied *e.g.* optimum of (2), (*i*) a deterministic mapping enforces the conditionals are matched. (*ii*) The matched conditionals enforce $H^{q_\phi}(\boldsymbol{x}|\boldsymbol{z})=0$, indicating the mapping becomes deterministic.

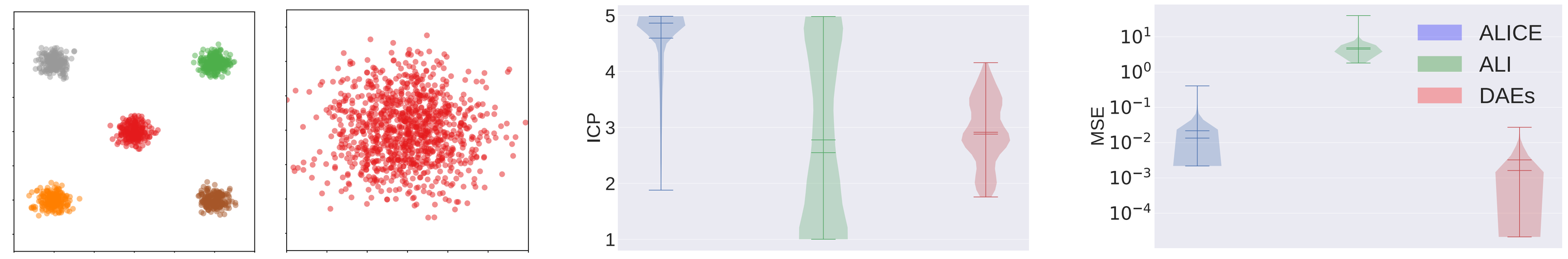
### Conditional GAN is doing joint distribution matching

When the optimum in (3) is achieved, $\tilde{\pi}(\boldsymbol{x},\boldsymbol{z}) = p_{\theta^*}(\boldsymbol{x},\boldsymbol{z}) = q_{\phi^*}(\boldsymbol{x},\boldsymbol{z})$.
One can leverage the empirically-defined distributions $\tilde{\pi}(\boldsymbol{x},\boldsymbol{z})$ implied by paired data, to resolve the ambiguity issues in unsupervised bivariate GANs.

## Experiments

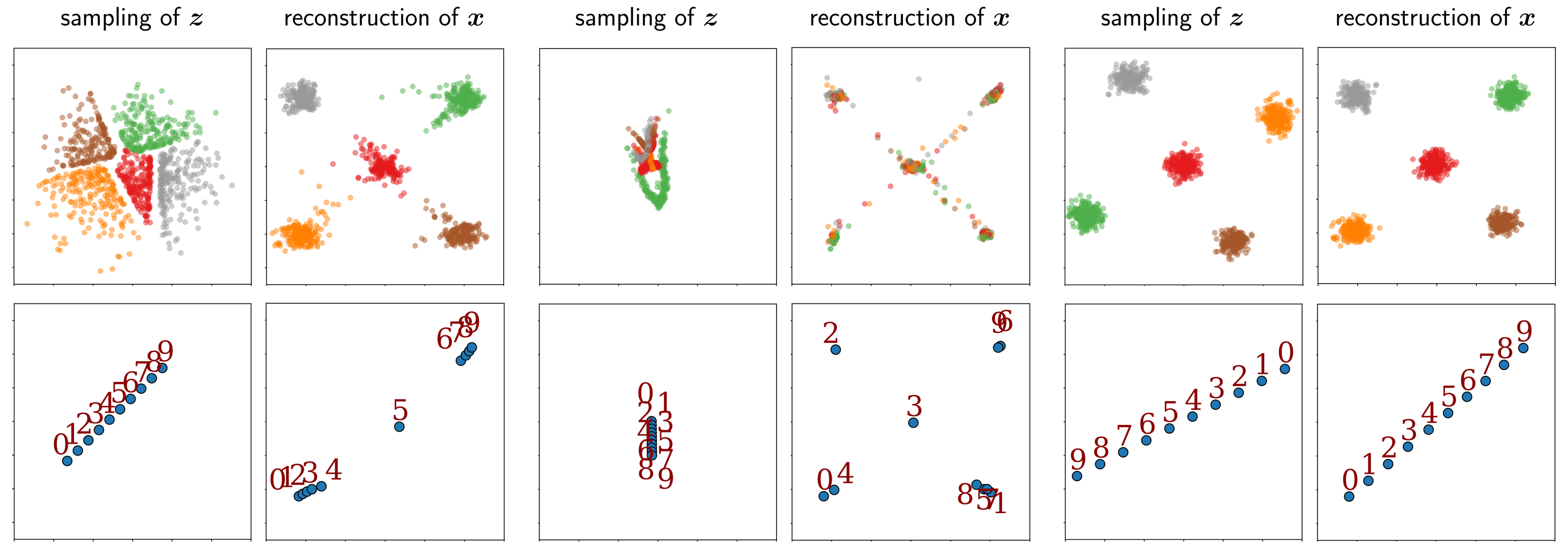
### I. Toy dataset

Grid search over a set of hyper-parameters for 576 experiments




(a) True $\boldsymbol{x}$ (b) True $\boldsymbol{z}$ (c) Inception Score (d) MSE

Figure: Generation (c) and reconstruction (d) results on toy data (a,b).




(a) ALICE (b) ALI (c) Denoising autoencoders

Figure: Sampling of $\boldsymbol{z}$, reconstruction of $\boldsymbol{x}$ and linear interpolation in $\boldsymbol{z}$.

### II. Alice4Alice

ALICE for painting the cartoon "Alice's Wonderland", based on edges

Training set: two domains (edges and cartoon) and two modes (Alice and Rabbit)

ALICE: one pair in each mode is leveraged to resolve ambiguity

CycleGAN: mixing colors due to the non-identifiable issue

Code: `https://github.com/ChunyuanLI/ALICE`