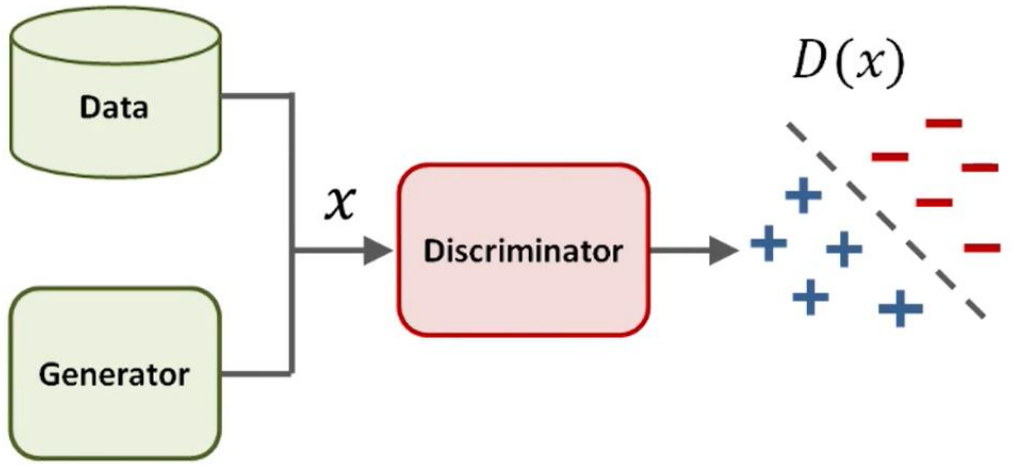
# VARIATIONAL DISCRIMINATOR BOTTLENECK:
## IMPROVING IMITATION LEARNING, INVERSE RL, AND GANS BY CONSTRAINING INFORMATION FLOW

2020.09.14

https://xbpeng.github.io/projects/VDB/
https://youtu.be/yMBDhE1wmaQ

# Introduction




Adversarial learning은 복잡한 내부 상관 구조를 가진 고차원 데이터에 대한 분포를 모델링하는
유망한 접근방식을 제공한다. 이러한 방법은 일반적으로 generator와 discriminator로 구성된다.

그러나 이들은 주요 최적화 챌린지에 시달리며, 그 중 하나는 generator와 discriminator의 성능 균형을 맞추는 것

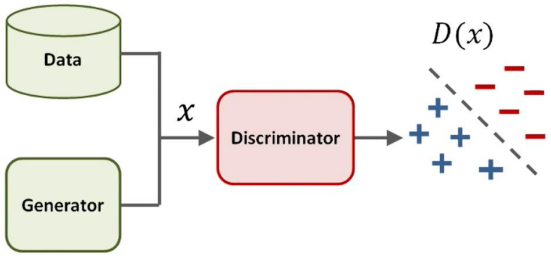
# Introduction

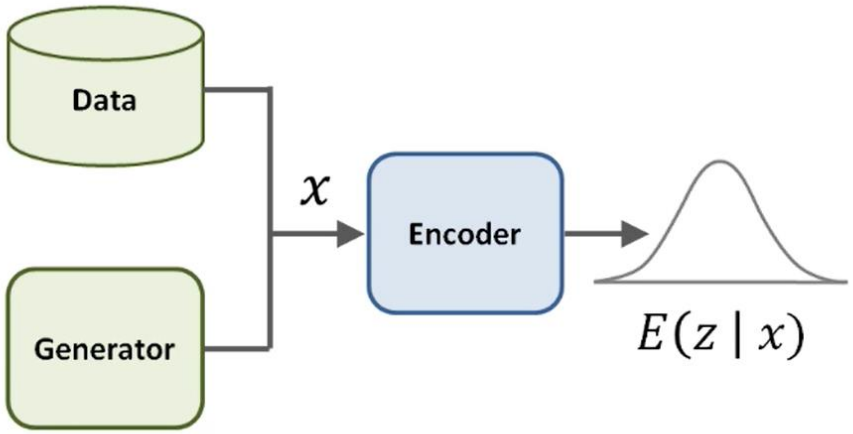
본 연구 – information bottleneck 역할을 하는 variation approximation을 사용해 discriminator의 성능을 적절하게 유지하는 방안에 대해 다룬다
→ 즉, information bottleneck을 통해 discriminator의 information flow를 제한하는 기술!

Adversarial learning을 위한 adaptive stochastic regularization method를 Variational Discriminator Bottleneck (VDB)라 하고, 이것이 논문의 주요 기여

이를 통해 imitation tasks, learning dynamic continuous control from video demonstrations, inverse reinforcement learning과 같이 여러가지 영역에서 폭넓게 사용되어 좋은 성능을 낼 수 있음을 보인다.

# Overview

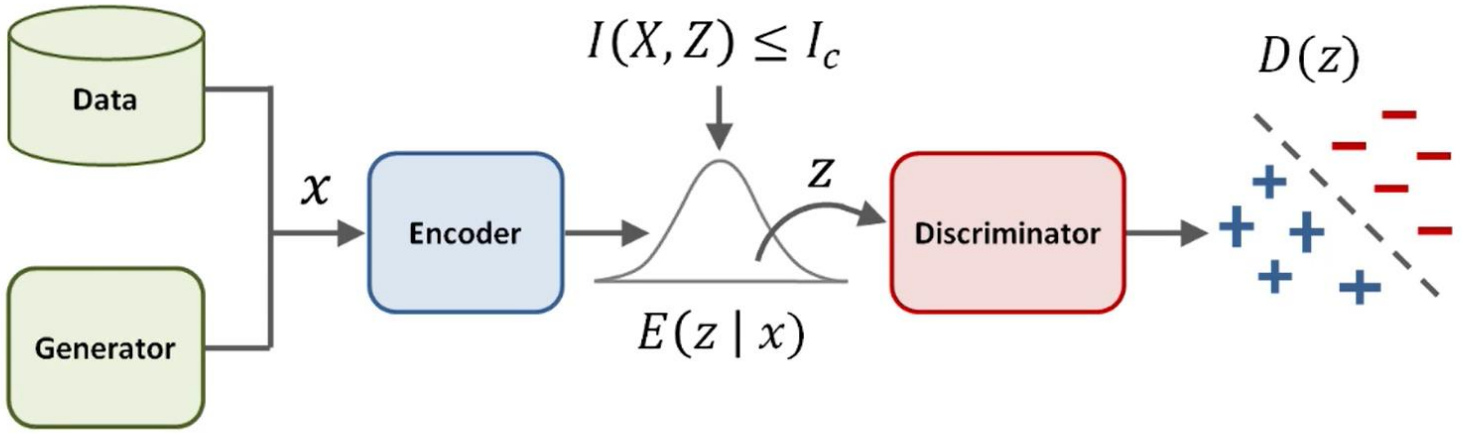



We propose a method for regularizing the discriminator using a variational information bottleneck
An encoder first maps each sample to a latent distribution

# Overview

The discriminator is then trained to classify samples from the latent distribution




An information bottleneck is placed on the discriminator, which limits the mutual information between the input and the embedding
The regularizer encourages the discriminator to focus on the most discerning features between real and fake samples

# Preliminaries

Supervised learning 관점에서의 variational information bottleneck에 대해 리뷰함
(VDB는 같은 원리에 기초함)

Feature $x_i$, label $y_i$로 이뤄진 데이터 셋 $\{x_i, y_i\}$가 주어졌을 때, standard maximum likelihood estimate $\mathrm{q}(y_i|x_i)$는 다음 식에 따라 결정된다

$$\min_{q} \quad \mathbb{E}_{\mathbf{x},\mathbf{y}\sim p(\mathbf{x},\mathbf{y})}\left[-\log q(\mathbf{y}|\mathbf{x})\right]$$

→ 안타깝게도 이 estimate는 overfitting하기 쉽고, model이 데이터의 특이점(idiosyncrasy)를 결과물에 반영하는 경우가 종종 생김

# Preliminaries

이에 Alemi et al. (2016)은 모델이 가장 차별적인 특징(discriminative feature)에만 초점을 맞추도록 유도하기 위해 information bottleneck을 사용해 모델을 정규화 할 것을 제안함

$$J(q, E) = \min_{q,E} \quad \mathbb{E}_{\mathbf{x},\mathbf{y}\sim p(\mathbf{x},\mathbf{y})} \left[\mathbb{E}_{\mathbf{z}\sim E(\mathbf{z}|\mathbf{x})} \left[-\log q(\mathbf{y}|\mathbf{z})\right]\right]$$
$$\text{s.t.} \quad I(X, Z) \leq I_c.$$

- $E(z|x)$ – feature $x$를 latent distribution에 mapping하는 encoder
- $I(X, Z)$ – encoding된 feature와 original feature 간의 mutual information
- $I_c$ – mutual information에 대한 upper bound

# Preliminaries

Mutual information은 하나의 확률변수를 관측했을 때 또 다른 확률변수에서 얻을 수 있는 정보량을 뜻함

$I(X, Z)$는 다음과 같이 정의되고, 이때 p(x)는 데이터 집합에서 주어진 분포다

$$I(X, Z) = \int p(\mathrm{x}, \mathrm{z}) \log \frac{p(\mathrm{x}, \mathrm{z})}{p(\mathrm{x})p(\mathrm{z})} \, d\mathrm{x} \, d\mathrm{z} = \int p(\mathrm{x}) E(\mathrm{z}|\mathrm{x}) \log \frac{E(\mathrm{z}|\mathrm{x})}{p(\mathrm{z})} \, d\mathrm{x} \, d\mathrm{z}$$

Mutual information을 계산하기 위해서는 marginal distribution $p(z) = \int E(z|x)p(x)dx$를 계산해야 하는데, 이게 약간 challenging해서, variational lower bound를 사용한다.

# Preliminaries

marginal에 대한 근사함수 $r(z)$를 도입하자

$KL[p(z)||r(z)] \geq 0$으로부터 $p(z)$와 $r(z)$의 관계식을 유도해보면 다음과 같다.

$$KL[p(z)||r(z)] = \int p(z) \log \frac{p(z)}{r(z)} dz = \int p(z) \log p(z)\, dz - \int p(z) \log r(z)\, dz \geq 0$$

$$\int p(z) \log p(z)\, dz \geq \int p(z) \log r(z)\, dz$$

위 관계를 이용해 $I(X, Z)$의 upper bound를 구할 수 있다

# Preliminaries

앞선 식으로부터,

$$
\begin{aligned}
I(X,Z) &= \int p(x)E(z|x)\log\frac{E(z|x)}{p(z)}dxdz \\
&\leq \int p(x)E(z|x)\log\frac{E(z|x)}{r(z)}dxdz \\
&= \mathbb{E}_{x\sim p(x)}\left[KL[E(z|x)\|r(z)]\right].
\end{aligned}
$$

$$
\therefore I(X,Z) = \mathbb{E}_{x\sim p(x)}[KL[E(z|x)||r(z)]
$$

KL divergence를 이용하여 $I(X,Z)$의 upper bound를 구할 수 있다!

# Preliminaries

$I(X,Z)$의 upper bound를 구했으니, 이를 활용하여 앞서 정의한 regularized objective function $J(q,E)$에 대한 upper bound를 구해보자

$$\tilde{J}(q,E) \geq J(q,E)$$

$$\tilde{J}(q,E) = \min_{q,E} \mathbb{E}_{x,y\sim p(x,y)}\left[\mathbb{E}_{z\sim E(z|x)}\left[-\log q(y|z)\right]\right]$$
$$\text{s.t. } \mathbb{E}_{x\sim p(x)}\left[KL[E(z|x)\|r(z)]\right] \leq I_c.$$

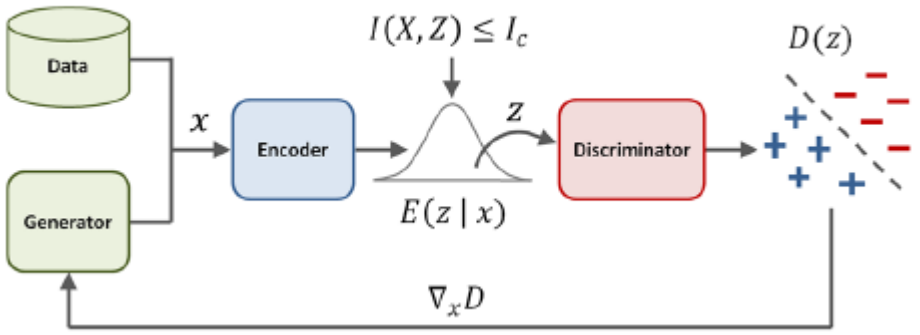
# Preliminaries

Unconstrained optimization으로 위 문제에 접근한다면 문제를 Lagrangian 형태로 변형하여 unconstrained problem으로 변환할 수 있다 (with coefficient $\beta$)

$$\min_{q,E} \quad \mathbb{E}_{\mathbf{x},\mathbf{y}\sim p(\mathbf{x},\mathbf{y})}\left[\mathbb{E}_{\mathbf{z}\sim E(\mathbf{z}|\mathbf{x})}\left[-\log q(\mathbf{y}|\mathbf{z})\right]\right] + \beta\left(\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[\mathrm{KL}\left[E(\mathbf{z}|\mathbf{x})||r(\mathbf{z})\right]\right] - I_c\right)$$

Variational Information Bottleneck (VIB)은 overfitting을 줄이는 효과가 있으며, adversarial examples에 대해 robust한 특성을 보인다.

# Variational Discriminator Bottelneck




표준 GAN의 framework를 우선 고려하면,

$$\max_G \min_D \quad \mathbb{E}_{\mathbf{x}\sim p^*(\mathbf{x})}\left[-\log\left(D(\mathbf{x})\right)\right] + \mathbb{E}_{\mathbf{x}\sim G(\mathbf{x})}\left[-\log\left(1-D(\mathbf{x})\right)\right]$$

여기서 discriminator에 encoder $E(z|x)$를 도입하여 GAN's discriminator + VIB의 문제를 정의해보자

# Variational Discriminator Bottelneck

$$J(D,E) = \min_{D,E} \quad \mathbb{E}_{x\sim p^*(\mathbf{x})}\left[\mathbb{E}_{\mathbf{z}\sim E(\mathbf{z}|\mathbf{x})}\left[-\log\left(D(\mathbf{z})\right)\right]\right] + \mathbb{E}_{\mathbf{x}\sim G(\mathbf{x})}\left[\mathbb{E}_{\mathbf{z}\sim E(\mathbf{z}|\mathbf{x})}\left[-\log\left(1-D(\mathbf{z})\right)\right]\right]$$
$$\text{s.t.} \quad \mathbb{E}_{\mathbf{x}\sim\tilde{p}(\mathbf{x})}\left[\mathrm{KL}\left[E(\mathbf{z}|\mathbf{x})||r(\mathbf{z})\right]\right] \leq I_c,$$

with $\tilde{p} = \frac{1}{2}p^* + \frac{1}{2}G$ being a mixture of the target distribution and the generator

Mixture distribution은 (특히 초반의) 학습이 잘 되지 않은 G에 의해 일어날 수 있는 high variance를 방지한다

위의 regularizer를 Variation Discriminator Bottleneck (VDB)라고 한다.

# Variational Discriminator Bottelneck

Lagrange multiplier $\beta$를 사용해 다음과 같이 나타낼 수도 있다.

$$J(D,E) = \min_{D,E} \max_{\beta \geq 0} \quad \mathbb{E}_{\mathbf{x}\sim p^*(\mathbf{x})}\left[\mathbb{E}_{\mathbf{z}\sim E(\mathbf{z}|\mathbf{x})}\left[-\log\left(D(\mathbf{z})\right)\right]\right] + \mathbb{E}_{\mathbf{x}\sim G(\mathbf{x})}\left[\mathbb{E}_{\mathbf{z}\sim E(\mathbf{z}|\mathbf{x})}\left[-\log\left(1-D(\mathbf{z})\right)\right]\right]$$
$$+ \beta\left(\mathbb{E}_{\mathbf{x}\sim\tilde{p}(\mathbf{x})}\left[\mathrm{KL}\left[E(\mathbf{z}|\mathbf{x})||r(\mathbf{z})\right]\right] - I_c\right).$$

# Variational Discriminator Bottelneck

Dual Gradient Descent*를 통해 위 문제를 푼다면, $D, E$ $\beta$에 대한 업데이트는 아래 과정을 반복한다.

$$D, E \leftarrow \arg\min_{D,E} \mathcal{L}(D, E, \beta)$$
$$\beta \leftarrow \max\left(0,\ \beta + \alpha_\beta \left(\mathbb{E}_{\mathbf{x}\sim\tilde{p}(\mathbf{x})}\left[\mathrm{KL}\left[E(\mathbf{z}|\mathbf{x})||r(\mathbf{z})\right]\right] - I_c\right)\right)$$

where $\mathcal{L}(D, E, \beta)$ is the Lagrangian

$$\mathcal{L}(D, E, \beta) = \mathbb{E}_{\mathbf{x}\sim p^*(\mathbf{x})}\left[\mathbb{E}_{\mathbf{z}\sim E(\mathbf{z}|\mathbf{x})}\left[-\log\left(D(\mathbf{z})\right)\right]\right] + \mathbb{E}_{\mathbf{x}\sim G(\mathbf{x})}\left[\mathbb{E}_{\mathbf{z}\sim E(\mathbf{z}|\mathbf{x})}\left[-\log\left(1 - D(\mathbf{z})\right)\right]\right] + \beta\left(\mathbb{E}_{\mathbf{x}\sim\tilde{p}(\mathbf{x})}\left[\mathrm{KL}\left[E(\mathbf{z}|\mathbf{x})||r(\mathbf{z})\right]\right] - I_c\right),$$

where $\alpha_\beta$ is a stepsize for the dual variable for dual gradient descent

* https://wikidocs.net/22603

# Variational Discriminator Bottelneck

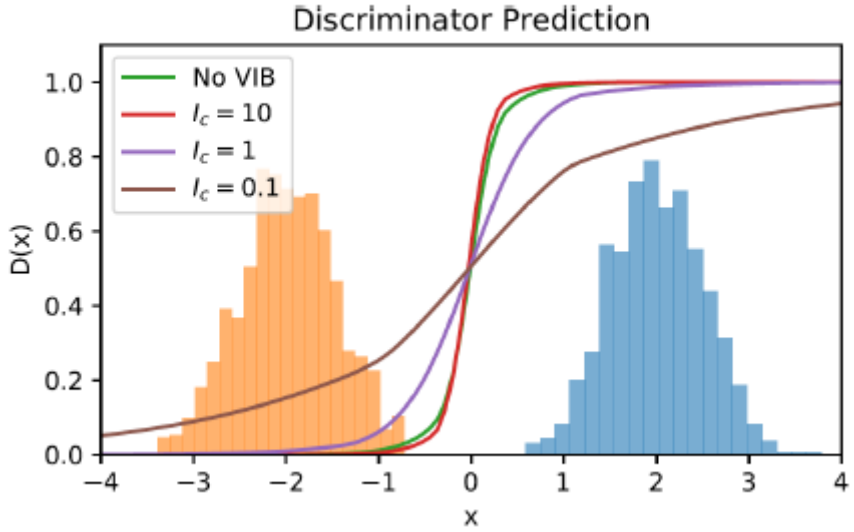
- 실험에서 $r(z) = \mathcal{N}(0, I)$, r을 정규분포로 설정한다
- Encoder는 mean $\mu_E$, diagonal covariance matrix $\sum_E(x)$ 의 정규분포로 정의한다. $\mathrm{E}(z|x) = \mathcal{N}(\mu_E(x), \sum_E(x))$
- Generator의 목적함수에는 Z에 대한 expectation이 아니라 $\mu_E(x)$를 이용한 근사식을 사용, 실험에서 충분한 성능을 냈다

$$\max_G \quad \mathbb{E}_{\mathbf{x}\sim G(\mathbf{x})}\left[-\log\left(1 - D(\mu_E(\mathbf{x}))\right)\right]$$

- Discriminator는 sigmoid를 activation으로 사용하는 single linear unit으로 모델링 되었다.

$$D(\mathbf{z}) = \sigma(\mathbf{w}_D^T\mathbf{z} + \mathbf{b}_D), \text{ with weights } \mathbf{w}_D \text{ and bias } \mathbf{b}_D$$

# Variational Discriminator Bottelneck




2개의 가우시안 분포에 대한 discriminator의 decision boundary를 나타낸것

$I_c$가 낮아질수록 decision boundary가 완만해지는 것을 볼 수 있다
→ generator 학습을 위한 informative gradient가 더욱 제공될 것이다!

# VAIL: Variational Adversarial Imitation Learning

GAIL을 통해 target policy $\pi^*(s)$와 policy $\pi(s)$로부터 state distributions를 구분하는 discriminator를 제안함

$$\max_{\pi} \min_{D} \mathbb{E}_{s\sim\pi^*(s)}[-\log(D(s))] + \mathbb{E}_{s\sim\pi(s)}[-\log(1-D(s))]$$

VDB를 discriminator에 도입하면 최적화 문제는 아래와 같이 변형되고, 이를 VAIL(Variational Adversarial Imitation Learning)이라고 칭한다

$$J(D,E) = \min_{D,E} \max_{\beta\geq 0} \; \mathbb{E}_{\mathbf{s}\sim\pi^*(\mathbf{s})}\left[\mathbb{E}_{\mathbf{z}\sim E(\mathbf{z}|\mathbf{s})}\left[-\log\left(D(\mathbf{z})\right)\right]\right] + \mathbb{E}_{\mathbf{s}\sim\pi(\mathbf{s})}\left[\mathbb{E}_{\mathbf{z}\sim E(\mathbf{z}|\mathbf{s})}\left[-\log\left(1-D(\mathbf{z})\right)\right]\right]$$
$$+ \beta\left(\mathbb{E}_{\mathbf{s}\sim\tilde{\pi}(\mathbf{s})}\left[\mathrm{KL}\left[E(\mathbf{z}|\mathbf{s})||r(\mathbf{z})\right]\right] - I_c\right).$$

where $\tilde{\pi} = \frac{1}{2}\pi^* + \frac{1}{2}\pi$ represents a mixture of the target policy and the agent's policy

# VAIRL: Variational Adversarial Inverse Reinforcement Learning

VDB는 AIRL에도 적용될 수 있다 → 이걸 VAIRL이라고 부를거다!

Disentangled reward function을 학습하는 discriminator를 다음과 같이 정의함

$$D(\mathrm{s}, \mathrm{a}, \mathrm{s}') = \frac{\exp\left(f(\mathrm{s}, \mathrm{a}, \mathrm{s}')\right)}{\exp\left(f(\mathrm{s}, \mathrm{a}, \mathrm{s}')\right) + \pi(\mathrm{a}|\mathrm{s})},$$

where $f(\mathrm{s}, \mathrm{a}, \mathrm{s}') = g(\mathrm{s}, \mathrm{a}) + \gamma h(\mathrm{s}') - h(\mathrm{s})$, with $g$ and $h$ being learned functions.

# VAIRL: Variational Adversarial Inverse Reinforcement Learning

VAIRL에서는 stochastic encoders $E_g(z_g|s), E_h(z_h|s), g(z_g), h(z_h)$와 latent variable에 대한 함수 $g(z_g), h(z_h)$를 도입되며 discriminator는 아래와 같이 변형된다.

$$D(\mathbf{s}, \mathbf{a}, \mathbf{z}) = \frac{\exp\left(f(\mathbf{z}_g, \mathbf{z}_h, \mathbf{z}'_h)\right)}{\exp\left(f(\mathbf{z}_g, \mathbf{z}_h, \mathbf{z}'_h)\right) + \pi(\mathbf{a}|\mathbf{s})},$$

$$\text{for } \mathbf{z} = (\mathbf{z}_g, \mathbf{z}_h, \mathbf{z}'_h) \text{ and } f(\mathbf{z}_g, \mathbf{z}_h, \mathbf{z}'_h) = D_g(\mathbf{z}_g) + \gamma D_h(\mathbf{z}'_h) - D_h(\mathbf{z}_h).$$

# VAIRL: Variational Adversarial Inverse Reinforcement Learning

또한 VAIRL의 최적화 문제는 아래와 같다.

$$
\begin{aligned}
J(D,E) = \min_{D,E} \max_{\beta \geq 0} \quad & \mathbb{E}_{\mathbf{s},\mathbf{s}' \sim \pi^*(\mathbf{s},\mathbf{s}')} \left[ \mathbb{E}_{\mathbf{z} \sim E(\mathbf{z}|\mathbf{s},\mathbf{s}')} \left[ -\log\left(D(\mathrm{s},\mathrm{a},\mathrm{z})\right) \right] \right] \\
& + \mathbb{E}_{\mathbf{s},\mathbf{s}' \sim \pi(\mathbf{s},\mathbf{s}')} \left[ \mathbb{E}_{\mathbf{z} \sim E(\mathbf{z}|\mathbf{s},\mathbf{s}')} \left[ -\log\left(1 - D(\mathrm{s},\mathrm{a},\mathrm{z})\right) \right] \right] \\
& + \beta \left( \mathbb{E}_{\mathbf{s},\mathbf{s}' \sim \tilde{\pi}(\mathbf{s},\mathbf{s}')} \left[ \mathrm{KL}\left[ E(\mathrm{z}|\mathrm{s},\mathrm{s}') || r(\mathrm{z}) \right] \right] - I_c \right),
\end{aligned}
$$

where $\pi(s, s')$ denotes the joint distribution of successive states from a policy, and $E(\mathrm{z}|\mathrm{s},\mathrm{s}') = E_g(\mathrm{z}_g|\mathrm{s}) \cdot E_h(\mathrm{z}_h|\mathrm{s}) \cdot E_h(\mathrm{z}'_h|\mathrm{s}')$.

# Experiments

#VAIL: Variational Adversarial Imitating Learning


Mocap clip의 demonstration을 얼마나 잘 따라하는지 측정하는 실험이다.

128차원의 encoding $Z$, information constraint $I_c$= 0.5, dual stepsize $\alpha_\beta = 10^{-5}$ 를 사용.
policy의 학습에는 PPO를 이용

# Experiments

Humanoid: Dance

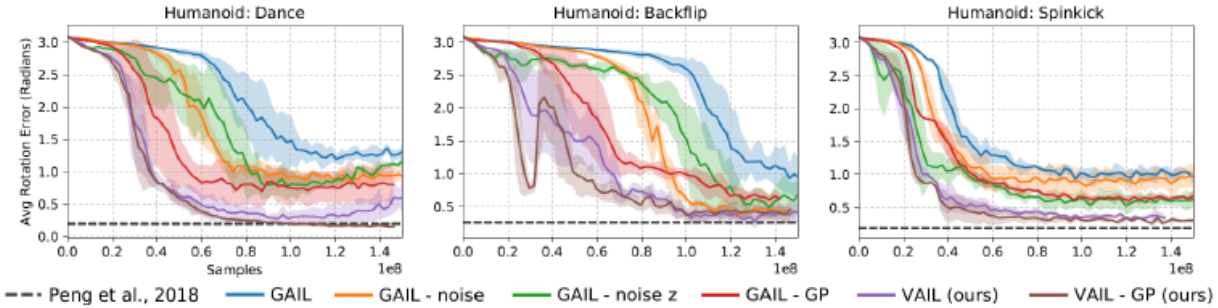




Figure 4: Learning curves comparing VAIL to other methods for motion imitation. Performance is measured using the average joint rotation error between the simulated character and the reference motion. Each method is evaluated with 3 random seeds.

| Method | Backflip | Cartwheel | Dance | Run | Spinkick |
|---|---|---|---|---|---|
| BC | 3.01 | 2.88 | 2.93 | 2.63 | 2.88 |
| Merel et al., 2017 | $1.33 \pm 0.03$ | $1.47 \pm 0.12$ | $2.61 \pm 0.30$ | $0.52 \pm 0.04$ | $1.82 \pm 0.35$ |
| GAIL | $0.74 \pm 0.15$ | $0.84 \pm 0.05$ | $1.31 \pm 0.16$ | $0.17 \pm 0.03$ | $1.07 \pm 0.03$ |
| GAIL - noise | $0.42 \pm 0.02$ | $0.92 \pm 0.07$ | $0.96 \pm 0.08$ | $0.21 \pm 0.05$ | $0.95 \pm 0.14$ |
| GAIL - noise z | $0.67 \pm 0.12$ | $0.72 \pm 0.04$ | $1.14 \pm 0.08$ | $0.14 \pm 0.03$ | $0.64 \pm 0.09$ |
| GAIL - GP | $0.62 \pm 0.09$ | $0.69 \pm 0.05$ | $0.80 \pm 0.32$ | $0.12 \pm 0.02$ | $0.64 \pm 0.04$ |
| VAIL (ours) | $\mathbf{0.36 \pm 0.13}$ | $0.40 \pm 0.08$ | $0.40 \pm 0.21$ | $0.13 \pm 0.01$ | $0.34 \pm 0.05$ |
| VAIL - GP (ours) | $0.46 \pm 0.17$ | $\mathbf{0.31 \pm 0.02}$ | $\mathbf{0.15 \pm 0.01}$ | $\mathbf{0.10 \pm 0.01}$ | $\mathbf{0.31 \pm 0.02}$ |
| Peng et al., 2018 | 0.26 | 0.21 | 0.20 | 0.14 | 0.19 |

Table 1: Average joint rotation error (radians) on humanoid motion imitation tasks. VAIL outperforms the other methods for all skills evaluated, except for policies trained using the manually-designed reward function from (Peng et al., 2018).

VAIL과 VAIL-GP (Gradient Penalty to the discriminator)가 가장 좋은 성능을 보였으며,

handcrafted reward(Peng et al.)를 사용한 경우와 전반적으로 상당히 근접한 결과를 얻어냈다.

# Experiments

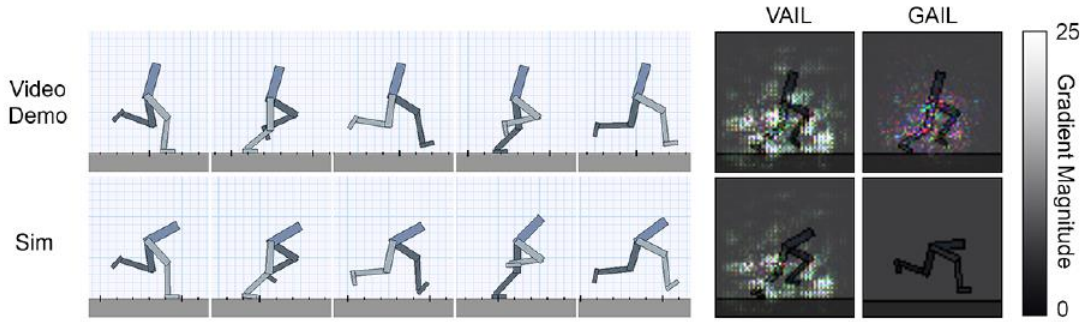



Figure 5: **Left:** Snapshots of the video demonstration and the simulated character trained with VAIL. The policy learns to run by directly imitating the video. **Right:** Saliency maps that visualize the magnitude of the discriminator's gradient with respect to all channels of the RGB input images from both the demonstration and the simulation. Pixel values are normalized between $[0, 1]$.

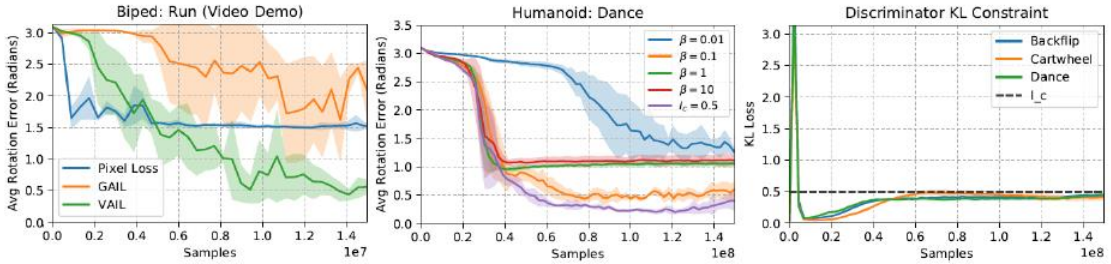


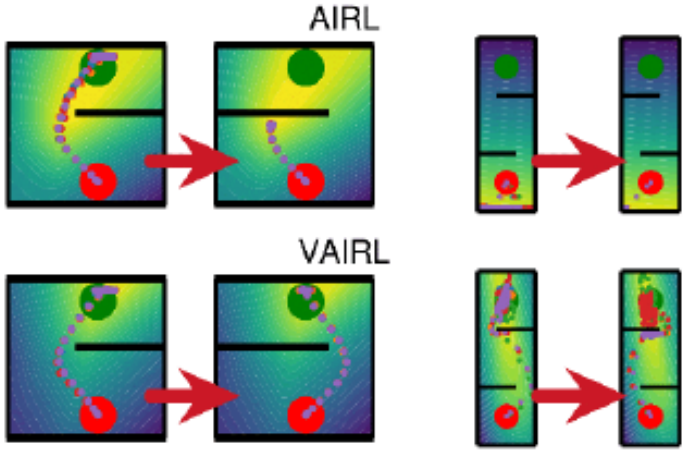

Figure 6: **Left:** Learning curves comparing policies for the video imitation task trained using a pixel-wise loss as the reward, GAIL, and VAIL. Only VAIL successfully learns to run from a video demonstration. **Middle:** Effect of training with fixed values of $\beta$ and adaptive $\beta$ ($I_c = 0.5$). **Right:**. KL loss over the course of training with adaptive $\beta$. The dual gradient descent update for $\beta$ effectively enforces the VDB constraint $I_c$.

# Experiments

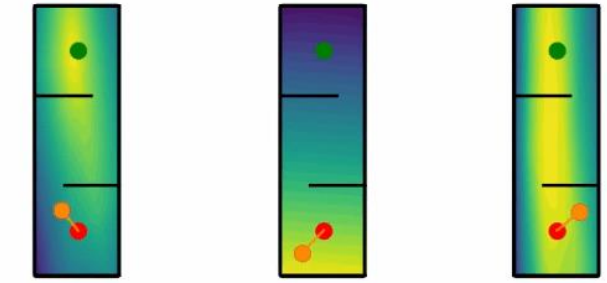
#VAIRL: Variational Adversarial Inverse Reinforcement Learning

- VDB는 시연에서 reward functions을 학습하기 위해 inverse RL을 정규화하는데 적용할 수 있음
- 학습된 reward functions은 다양한 환경으로 transferred 할 수 있으며, 새로운 policy를 훈련하는데 사용될 수 있음
- C-Maze와 S-Maze 환경에서 변동이 발생하더라도 유의미한 behaviour를 얼마나 잘 유지하는지 측정해봄
- C-Maze에서 AIRL이 gradient penalty 없이는 overfitting으로 인해 transferring에 실패하는 모습을 종종 보인 반면에, VAIRL은 gradient penalty 없이도 transferring task에 좀 더 안정적인 모습을 보였다. 또한 KL constraint를 사용하지 않았을 때 두 개의 task에서 VAIRL의 성능이 떨어지는 것을 관찰할 수 있었다.

# Experiments




| Method | Transfer environments | |
|---|---|---|
| | C-maze | S-maze |
| GAIL | -24.6±7.2 | 1.0±1.3 |
| VAIL | -65.6±18.9 | 20.8±39.7 |
| AIRL | -15.3±7.8 | -0.2±0.1 |
| AIRL - GP | **-9.14±0.4** | -0.14±0.3 |
| VAIRL ($\beta = 0$) | -25.5±7.2 | 62.3±33.2 |
| VAIRL (ours) | -10.0±2.2 | 74.0±38.7 |
| VAIRL - GP (ours) | -9.18±0.4 | **156.5±5.6** |
| TRPO expert | -5.1 | 153.2 |




VAIRL은 AIRL과 다르게 Smoother reward functions을 학습함
→ 그래서 좌우반전을 할 경우 VAIRL은 이동 가능 (in C-Maze)

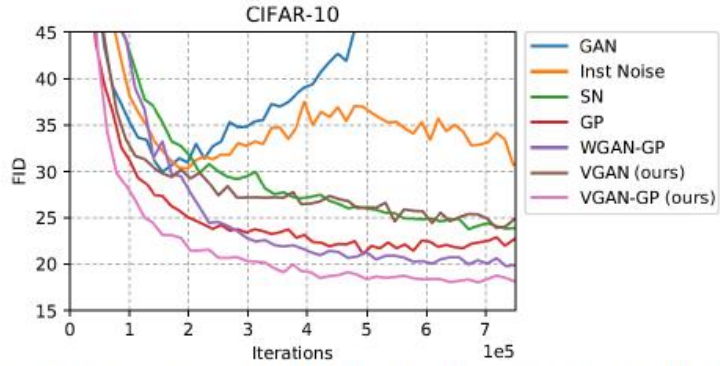
S-Maze에서는 AIRL이 불안정하여 의미있는 reward function을 획득할 수 없었음
→ 반면에 VAIRL은 합리적인 reward를 배울 수 있었고, gradient penalty가 추가되면서 성능 더욱 향상되어짐

# Experiments

#VGAN: Variational Adversarial Generative Adversarial Networks

- VDB를 이미지 생성모델에 적용해 봄
- CIFAR-10, CelebA, CelebAHQ 데이터셋을 이용해 실험진행
- stabilization technicques인 WGAN-GP, Spectral Normalization (SN), Gradient Penalty (GP) 및 original GAN과 성능 비교
- 성능 측정에는 Fréchet Inception Distance (FID)를 이용
- 모든 methods는 같은 base model 사용 - Mescheder et al. (2018)의 resnet architecture
- VGAN의 경우 KL constraint $I_c$외의 모든 파라미터는 Mescheder et al. (2018)의 것을 그대로 사용했다.

# Experiments




| Method | FID |
|---|---|
| GAN | 63.6 |
| Inst Noise | 30.7 |
| SN | 23.9 |
| GP | 22.6 |
| WGAN-GP | 19.9 |
| VGAN (ours) | 24.8 |
| VGAN-SN (ours) | 71.8 |
| VGAN-GP (ours) | **18.1** |

Figure 8: Comparison of VGAN and other methods on CIFAR-10, with performance evaluated using the Fréchet Inception Distance (FID).

Figure 9: VGAN samples on CIFAR-10, CelebA 128×128, and CelebAHQ 1024×1024.

CIFAR-10에 대한 다양한 method들의 성능

FID는 낮을수록 좋은 성능을 낸다고 할 수 있다

논문의 접근방식으로 생성해낸 이미지들

VDB에 Gradient Penalty가 적용된 것이 가장 좋은 성능을 보임
VDB에 SN이 적용된 경우에는 쉽게 diverging하는 모습이 관찰되었다.

# Conclusion

- 본 논문에서는 adversarial learning을 위한 general regularization technique인 variational discriminator bottleneck을 제시함
- VDB는 다양한 도메인에 광범위하게 적용가능, 이전 기존 기법보다 크게 개선
- 실험은 비디오 모방을 위한 유망한 결과를 만들어냈지만, 그 결과는 주로 합성 장면의 비디오로 이루어짐 우리는 이 기술을 실제 영상을 모방하는 것으로 확장하는 것이 흥미로운 방향일 것