

Chapter 8. Guidance and Controllable Generation

Choi Yoonseo

Seoul National Univ.

August 18, 2026

Outline

- 1 8.1 Prologue
- 2 8.2 Classifier Guidance (CG)
- 3 8.3 Classifier-Free Guidance (CFG)
- 4 8.4 Training-Free Guidance
- 5 8.5 RLHF → Direct Preference Optimization

Outline

- 1 8.1 Prologue
- 2 8.2 Classifier Guidance (CG)
- 3 8.3 Classifier-Free Guidance (CFG)
- 4 8.4 Training-Free Guidance
- 5 8.5 RLHF → Direct Preference Optimization

8.1 Prologue

지금까지는 무조건분포에서 데이터 생성 ($p_{\text{data}}(x)$) — **사용자가 지시를 줄 수 없었음**

목표 1: Control

- 사용자의 지시에 맞는 데이터를 생성하는 모델을 만들자.
- 조건부 생성 $p_0(x | c)$, $c = \text{label} / \text{text} / \text{sketch} \dots$

목표 2: Alignment

- 사용자의 지시에 맞추면서 동시에 사람이 선호하는 데이터를 생성하는 모델을 만들자.

8.1 Prologue - Control

Recall

- PF-ODE, reverse time SDE: 스코어함수만 근사할 수 있다면 무조건부 분포에 대한 샘플링이 가능
- ex:

$$\frac{d\tilde{x}}{dt} = f(\tilde{x}, t) - \frac{1}{2}g^2(t)\nabla_x \log p_t(\tilde{x}|c)$$

(해석: 스코어항이 p_t 의 확률 밀도가 높은 쪽으로 입자 x 를 끌어당김)
스코어함수를 근사한 후 위의 식에 대입하고 수치적분하면 됨.

- **조건부 분포에 대한 스코어함수**를 근사한다면 마찬가지로 조건부 분포에서 샘플링이 가능할 것.

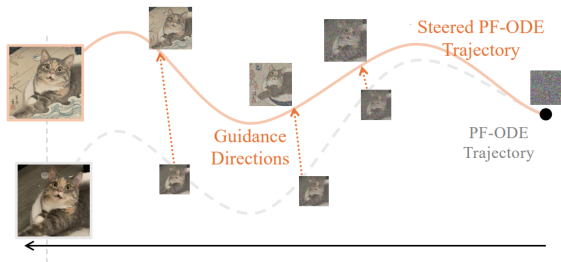
8.1 Prologue - Control

조건부 분포의 스코어를 분해해보자

- Bayes Rule을 이용하면

$$\nabla_{x_t} \log p_t(x_t | c) = \underbrace{\nabla_{x_t} \log p_t(x_t)}_{\text{unconditional direction}} + \underbrace{\nabla_{x_t} \log p_t(c | x_t)}_{\text{guidance direction}}.$$

- 마치 무조건부 분포를 guidance 방향으로 밀어주는 형태
- Diffusion model은 매 스텝별로 노이즈가 조금씩 걷히면서 원하는 분포에 가까워지는 모델.
⇒ 매 스텝마다 guidance 방향으로 밀어주는 것이라고 해석 가능.



8.1 Prologue - Control

조건부 스코어 모델 종류

- $\nabla_{x_t} \log p_t(c | x_t)$ 를 근사하는 방식에 따라 3가지 모델로 나뉨.
- $p_t(c | x_t) = \int p(c | x_0) p(x_0 | x_t) dx_0$ 는 x_0 에 대한 marginalization 때문에 일반적으로 **intractable**

방법	무엇을 학습/사용하나	개입 시점
CG (§ 8.2)	사전학습 무조건 s_ϕ + 별도 분류기 $p_\psi(c x_t, t)$	샘플링 시
CFG (§ 8.3)	단일 조건부 모델 $s_\phi(x_t, t, c)$ (null token으로 무조건 검용)	학습 필요
Training-Free (§ 8.4)	손실 $\ell(x_t, c)$ 로 만든 대리 likelihood $\tilde{p}_t(c x_t)$	샘플링 시

8.1 Prologue - alignment

아이디어

- 생성한 데이터가 조건에는 맞아도 사람이 선호하지 않는 데이터일 수도 있음
⇒ 조건부 분포를 **사람이 선호하는 방향으로** 밀기.

Preference tilt

$$\tilde{p}_0^{\text{tilt}}(x_0 \mid c) \propto p_0(x_0 \mid c) \exp(\beta r(x_0, c)).$$

- r 의 예: 외부 reward/classifier 의 logit, CLIP/perceptual 유사도, 학습된 선호 모델.

Outline

- 1 8.1 Prologue
- 2 8.2 Classifier Guidance (CG)
- 3 8.3 Classifier-Free Guidance (CFG)
- 4 8.4 Training-Free Guidance
- 5 8.5 RLHF → Direct Preference Optimization

8.2 Classifier Guidance (CG)

아이디어

- 오염된 데이터 x_t 가 어떤 label에 속하는 것인지 예측하는 classifier를 만들자.
- (참고) CG는 조건 c 가 이산형 label일 때 사용

Bayes 규칙에 의한 분해 (8.2.1)

$$\nabla_{x_t} \log p_t(x_t | c) = \nabla_{x_t} \log \left(\frac{p_t(x_t)p_t(c|x_t)}{p(c)} \right) = \underbrace{\nabla_{x_t} \log p_t(x_t)}_{\text{uncond. score}} + \underbrace{\nabla_{x_t} \log p_t(c | x_t)}_{\text{classifier gradient}},$$

- 무조건부 스코어는 7장까지 봤던 방법들로 추정 가능.
- classifier gradient만 추정하면 됨.
- ($\nabla_{x_t} \log p(c) = 0 : x_t$ 에 무관하므로 미분에서 소멸.)

8.2 Classifier Guidance (CG) - Training

cross-entropy loss로 $p_t(c \mid x_t)$ 학습

CG의 training objective (eq 8.2.4)

$$\mathbb{E}_{t \sim U[0, T], (x, c) \sim p_{\text{data}}, \epsilon \sim \mathcal{N}(0, I)} \left[-\log p_{\psi}(c \mid x_t, t) \right], \quad x_t = \alpha_t x + \sigma_t \epsilon.$$

- 분류 문제에서 많이 사용하는 loss.
- 직관적 해석: 정답 c 를 맞출 확률이 커질수록 loss 감소.
- **diffusion** 과 독립적으로 학습
- 학습 후 log와 gradient를 취한 것을 classifier guidance로 사용

8.2 Classifier Guidance (CG) - Sampling

s^{CG} 로 **reverse-time SDE / PF-ODE** 의 스코어를 대체해 풀기

$$s^{\text{CG}}(x_t, t, c; \omega) := \underbrace{s_{\phi^*}(x_t, t)}_{\text{uncond.}} + \omega \underbrace{\nabla_{x_t} \log p_{\psi}(c \mid x_t, t)}_{\text{guidance}} \approx \nabla_{x_t} \log p_t(x_t \mid c, \omega).$$

- 학습된 **diffusion** 은 전혀 건드리지 **않음** — 샘플링 시점에만 개입

참고) guidance의 세기를 조절하기 위해 **tilted conditional** 도입

$$\begin{aligned} p_t(x_t \mid c, \omega) &\propto p_t(x_t) p_t(c \mid x_t)^\omega \\ \Rightarrow \nabla_{x_t} \log p_t(x_t \mid c, \omega) &= \nabla_{x_t} \log p_t(x_t) + \omega \nabla_{x_t} \log p_t(c \mid x_t). \end{aligned}$$

- $\omega = 1$: 참 조건부 스코어 $\nabla_{x_t} \log p_t(x_t \mid c)$ 복원.
- $\omega > 1$: 분류기 신호 증폭 $\Rightarrow$ 조건 충실도 $\uparrow$ (다양성 $\downarrow$).
- $0 \leq \omega < 1$: 분류기 신호 약화 $\Rightarrow$ 다양성 $\uparrow$ (조건 약화).

8.2 Classifier Guidance (CG) - 장점과 한계

장점

- ω 로 조건 강도를 명시적으로 조절.
- 사전학습된 모델에 classifier 하나만 추가하면 조건부 분포에서 샘플링.

한계

- Training cost: 모든 noise level 에서 classifier 학습 $\Rightarrow$ 비용 큼.
- Robustness: 큰 t 의 심하게 오염된 x_t 에 대해서는 학습이 어려움.
- Separate training: classifier를 diffusion model를 다른 데이터셋으로 학습하므로 두 모델의 데이터 manifold가 어긋날 수 있음.

Outline

- 1 8.1 Prologue
- 2 8.2 Classifier Guidance (CG)
- 3 8.3 Classifier-Free Guidance (CFG)**
- 4 8.4 Training-Free Guidance
- 5 8.5 RLHF → Direct Preference Optimization

8.3 Classifier-Free Guidance (CFG)

아이디어: 별도 classifier를 없애고 한번에 조건부 분포를 근사하자. How?

guidance 항을 두 스코어의 차로 표현 (8.3.1)

$$\nabla_{x_t} \log p_t(x_t | c, \omega) = \omega \underbrace{\nabla_{x_t} \log p_t(x_t | c)}_{\text{conditional}} + (1 - \omega) \underbrace{\nabla_{x_t} \log p_t(x_t)}_{\text{unconditional}}.$$

유도과정

guidance 항을 변형하면 $\nabla_{x_t} \log p_t(c | x_t) = \nabla_{x_t} \log p_t(x_t | c) - \nabla_{x_t} \log p_t(x_t)$.
또한, (8.2.3)에 따르면 $\nabla_{x_t} \log p_t(x_t | c, \omega) = \nabla_{x_t} \log p_t(x_t) + \omega \nabla_{x_t} \log p_t(c | x_t)$.
이를 대입하여 정리하면

$$\begin{aligned} \nabla_{x_t} \log p_t(x_t | c, \omega) &= \nabla_{x_t} \log p_t(x_t) + \omega (\nabla_{x_t} \log p_t(x_t | c) - \nabla_{x_t} \log p_t(x_t)) \\ &= \omega \underbrace{\nabla_{x_t} \log p_t(x_t | c)}_{\text{conditional}} + (1 - \omega) \underbrace{\nabla_{x_t} \log p_t(x_t)}_{\text{unconditional}}. \end{aligned}$$

8.3 Classifier-Free Guidance (CFG)

ω 해석 방법

$$\nabla_{x_t} \log p_t(x_t \mid c, \omega) = \omega \nabla_{x_t} \log p_t(x_t \mid c) + (1 - \omega) \nabla_{x_t} \log p_t(x_t).$$

- $\omega = 0$: 완전 무조건 (조건 무시).
- $\omega = 1$: 추가 guidance 없는 조건부 스코어.
- $\omega > 1$: 조건부 강조 / 무조건 약화 $\Rightarrow$ 조건 정렬 $\uparrow$ (다양성 $\downarrow$).

8.3 Classifier-Free Guidance (CFG) - Training and Sampling

조건부, 무조건부 파트를 하나의 모델로 학습

- 단일 모델 $s_\phi(x_t, t, c)$ 사용
- 조건부 모델 학습할 때는 $c = \text{조건}$, 무조건부 모델을 학습할 때는 조건 $c = \text{null token}$ 이라 둬.
- 확률 $p_{\text{uncond}} (\approx 0.1)$ 로 c 를 고름.

CFG Training 알고리즘

- 1 $(x, c) \sim p_{\text{data}}(x, c)$
- 2 $c \leftarrow \emptyset$ with prob. p_{uncond}
- 3 $t \sim U[0, T], \epsilon \sim \mathcal{N}(0, I)$
- 4 $x_t = \alpha_t x + \sigma_t \epsilon$
- 5 step on $\nabla_\phi \|s_\phi(x_t, t, c) - s\|^2$

(참고) $s = \nabla_{x_t} \log p_t(x_t | x_0) = \epsilon / \sigma_t$.

Sampling

샘플링 시 스코어를 다음으로 대체 (8.3.3):

$$\nabla_{x_t} \log p_t(x_t | c, \omega) \approx \underbrace{\omega s_{\phi^*}(x_t, t, c)}_{\text{conditional}} + (1 - \omega) \underbrace{s_{\phi^*}(x_t, t, \emptyset)}_{\text{unconditional}} =: s_{\phi^*}^{\text{CFG}}(x_t, t, c; \omega).$$

Outline

- 1 8.1 Prologue
- 2 8.2 Classifier Guidance (CG)
- 3 8.3 Classifier-Free Guidance (CFG)
- 4 8.4 Training-Free Guidance**
- 5 8.5 RLHF → Direct Preference Optimization

8.4 Training-Free Guidance

아이디어: 데이터가 조건에 부합하는 정도를 구해서 추가학습 없이 guidance를 주자.

DDIM

$$x_{t \rightarrow t-1} = \alpha_{t-1} \underbrace{\hat{x}_0(x_t)}_{\text{data space}} - \sigma_{t-1}\sigma_t \underbrace{\hat{s}(x_t)}_{\text{noise space}}, \quad \hat{x}_0(x_t) = x_{\phi^*}(x_t, t), \quad \hat{s}(x_t) = s_{\phi^*}(x_t, t).$$

- DDPM의 결정론적 버전
- (1) x_t 에서 clean 예측 $\hat{x}_0(x_t)$
(2) $t - 1$ 수준으로 되돌림
- 아이디어: data space 혹은 noise space term을 조건에 부합하는 방향으로 밀자

$$x_{t \rightarrow t-1} = \alpha_{t-1} \underbrace{\left(\hat{x}_0(x_t) + \eta_t^{\text{data}} G_0 \right)}_{\text{A. data space}} - \sigma_{t-1}\sigma_t \underbrace{\left(\hat{s}(x_t) + \eta_t^{\text{latent}} G_t \right)}_{\text{B. noise space}}.$$

- 이미 학습된 diffusion $s_{\phi^*}(x_t, t)$ 만 사용(재학습 X, 샘플링 시 개입).
- 참고) 위처럼 구한 샘플이 조건부 분포를 따르지는 않음.

8.4 Training-Free Guidance - A. data space

A. Data space: 예측된 clean 이미지를 ℓ 이 감소하는 방향으로 수정

$$G_0 := -\nabla_{x_0} \ell(x_0, c)$$

- $\ell(\cdot, c) : \mathbb{R}^D \rightarrow \mathbb{R}_{\geq 0}$: 데이터 x 가 조건 c 에 얼마나 부합하는지 나타내는 점수 (x 가 c 에 부합할수록 작음)
- ex) x 가 이미지, c 가 참고 이미지라면 similarity score

Q.

- Q. P.239에서 Guidance in Data Space 는 어떻게 해석하나요?

8.4 Training-Free Guidance - B. Noise space

B. Noise space: 스코어 자리에 조건부 스코어 대입

- 단 $\nabla_{x_t} \log p_t(c | x_t)$ 는 intractable
⇒ surrogate likelihood 로 우회:

$$\tilde{p}_t(c | x_t) \propto \exp(-\eta \ell(\hat{x}_0(x_t), c)) \Rightarrow G_t := \nabla_{x_t} \log \tilde{p}_t(c | x_t) = -\eta \nabla_{x_t} \ell(\hat{x}_0(x_t), c).$$

surrogate likelihood에 대한 직관

- 에너지가 작은 곳에 데이터가 많다.
- ℓ 이 작다는건 c 에 부합한다는 뜻 → 그곳에서 샘플이 많이 뽑혀야 함.
⇒ ℓ 을 에너지로 보는 것이 자연스러움.
- 정리: $e^{-\eta \ell}$ 는 비음수이며 ℓ 이 작을수록 surrogate이 커지므로 energy function은 자연스러운 선택.
- 참고) A 는 ℓ 을 x_0 로, B 는 x_t 로 backprop ($\nabla_{x_t} \hat{x}_0(x_t)^\top \cdot \nabla_{x_0} \log \ell_c|_{x_0=\hat{x}_0}$ 통과) ⇒ B의 계산비용이 더 큼.

8.4 Training-Free Guidance - Inverse Problem (1)

Inverse problem

- 목표: 관측 y 로부터 clean 샘플 구하기. 즉 posterior $p_0(x_0 | y)$ 에서 샘플링.
- 문제 세팅: $y = A(x_0) + \sigma_y z$, $z \sim \mathcal{N}(0, I)$, A : 오염 operator

이를 TFG로 풀면...

- Bayes 분해를 하면

$$\nabla_{x_t} \log p_t(x_t | y) = \underbrace{\nabla_{x_t} \log p_t(x_t)}_{\text{data score (사전학습 diffusion)}} + \underbrace{\nabla_{x_t} \log p_t(y | x_t)}_{\text{measurement alignment (근사 대상)}}.$$

- $\nabla_{x_t} \log p_t(y | x_t)$ 은 일반적으로 다음과 같은 form으로 근사한다.

$$\nabla_{x_t} \log p_t(y | x_t) \approx -\frac{P_t M_t}{\gamma_t}, \quad \begin{cases} M_t : & \text{관측 } y \text{ 와 추정치의 불일치(관측공간)} \\ P_t : & \text{오차를 } x_t \text{ 공간으로 투영} \\ \gamma_t : & \text{guidance 강도} \end{cases}$$

8.4 Training-Free Guidance - Inverse Problem (1)

Score SDE

- **문제 세팅:** A 가 linear이고 $\sigma_y = 0$ 이라 가정
- **아이디어**
 - ▶ measurement alignment항은 결국 $Ax_0 = y$ 라는 정보를 추가적으로 주는 항.
 - ▶ x_t 은 noise level t 만큼 noise가 추가된 데이터이므로, y 도 같은 noise level로 맞추자. ($=y_t$)
그리고 y_t 와 Ax_t 가 가까워지는 방향으로 사전학습된 diffusion 분포를 조정하자.
- **근사식:** $\nabla_{x_t} \log p_t(y | x_t) \approx -A^T(y_t - Ax_t)$
- **직관:** $\|y_t - Ax_t\|^2$ 를 loss라고 두고 이를 미분하면 $-A^T(y_t - Ax_t)$ 이 나옴.

ILVR

- pseudoinverse $A^\dagger$ 로 정교화:

$$\nabla_{x_t} \log p_t(y | x_t) \approx -A^\dagger(y_t - Ax_t) = -(A^T A)^{-1} A^T (y_t - Ax_t).$$

8.4 Training-Free Guidance - Inverse Problem (1)

DPS

- **문제 세팅:** A가 nonlinear이고 $\sigma_y \geq 0$ 이라 가정

- **아이디어**

- ▶ data space에서 y 와 $A\hat{x}_0(x_t)$ 를 비교
- ▶ $p(x_0|x_t)$ 가 평균 $\hat{x}_0(x_t)$ 에 집중되어 있다고 가정하고 $\nabla_{x_t} \log p_t(y|x_t)$ 를 근사

- **수식 전개**

$$\nabla_{x_t} \log p_t(y|x_t) = \nabla_{x_t} \log \int p(y|x_0) p(x_0|x_t) dx_0 \approx \nabla_{x_t} \log p(y|X_0 = \hat{x}_0(x_t))$$

Gaussian noise 가정 ($y = Ax_0 + \sigma_y z$) 하에:

$$p(y|X_0 = \hat{x}_0(x_t)) = \mathcal{N}(y; A\hat{x}_0(x_t), \sigma_y^2 I)$$

gradient를 계산하면:

$$\nabla_{x_t} \log p_t(y|x_t) \approx \frac{1}{\sigma_y^2} [J_A(\hat{x}_0(x_t)) \cdot \nabla_{x_t} \hat{x}_0(x_t)]^\top \underbrace{(y - A\hat{x}_0(x_t))}_{\text{residual}}, J_A(\hat{x}_0(x_t)) := \nabla_{x_0} A(x)|_{x=\hat{x}_0(x_t)}$$

이때 $\hat{x}_0(x_t) = \mathbb{E}[x_0|x_t]$ 는 Tweedie's formula로 추정: $\hat{x}_0(x_t) = \frac{x_t + \sigma_t^2 s_{\phi^*}(x_t, t)}{\alpha_t}$

8.4 Training-Free Guidance - Inverse Problem (2)

문제 세팅: 손상 과정 수식을 모르는 경우. 아는 것은 오염된 데이터와 원본 쌍일 때 $((y, x_0))$

아이디어: 선형보간을 이용해서 x_0 에서 y 로 가는 간단한 경로를 잡자.

- 경로: $x_t = (1 - t)x_0 + ty, t \in [0, 1]$
- 임의의 $0 \leq s \leq t$ 에 대해

$$x_s = \left(1 - \frac{s}{t}\right)x_0 + \frac{s}{t}x_t \xrightarrow{\mathbb{E}[\cdot|x_t]} \mathbb{E}[x_s | x_t] = \left(1 - \frac{s}{t}\right)\mathbb{E}[x_0 | x_t] + \frac{s}{t}x_t.$$

즉, x_t 가 주어졌을 때 x_0 만 잘 예측하면 중간 어느 시점이든 계산 가능.

학습

- $\mathbb{E}[x_0 | x_t]$ 를 신경망으로 학습.
- 다음을 training objective로 설정:

$$\min_{\phi} \mathbb{E}_{(x_0, y), t} \|F_{\phi}((1 - t)x_0 + ty, t) - x_0\|^2 \Rightarrow F^*(x_t, t) = \mathbb{E}[x_0 | x_t].$$

- 이 식의 최적해는 $\mathbb{E}[x_0 | x_t]$

8.4 Training-Free Guidance - Inverse Problem (2)

Sampling

- $\hat{x}_1 = y$ 에서 시작해 점진적으로 refine하기

$$\hat{x}_{t-\Delta t} = \frac{\Delta t}{t} F_\phi(\hat{x}_t, t) + \left(1 - \frac{\Delta t}{t}\right) \hat{x}_t \xrightarrow{\Delta t \rightarrow 0} \frac{dx_t}{dt} = \frac{x_t - F_\phi(x_t, t)}{t}.$$

noise가 가우시안일 때의 Sampling

- $y = x_0 + \sigma\epsilon$ 이면 $x_t = x_0 + t\sigma\epsilon$, 즉 $\alpha_t = 1$, $\sigma_t = t\sigma$.
Tweedie's formula $\mathbb{E}[x_0 | x_t] = x_t + \sigma_t^2 \nabla_{x_t} \log p_t(x_t)$ 를 쓰면

$$\frac{dx_t}{dt} = \frac{x_t - \mathbb{E}[x_0 | x_t]}{t} = \frac{-\sigma_t^2 \nabla_{x_t} \log p_t(x_t)}{t} = -\dot{\sigma}_t \sigma_t \nabla_{x_t} \log p_t(x_t),$$

$$(\sigma_t = t\sigma \Rightarrow \dot{\sigma}_t = \sigma, \dot{\sigma}_t \sigma_t = t\sigma^2 = \sigma_t^2/t.)$$

- 이는 정확히 이 경로의 **PF-ODE**. $\Rightarrow$ diffusion 을 “가우시안 오염에 대한 지도 denoiser 학습” 으로 볼 수 있음.

Outline

- 1 8.1 Prologue
- 2 8.2 Classifier Guidance (CG)
- 3 8.3 Classifier-Free Guidance (CFG)
- 4 8.4 Training-Free Guidance
- 5 8.5 RLHF → Direct Preference Optimization

8.5 RLHF → DPO

Alignment: 사전학습 모델을 사람이 선호하는 출력으로 유도. → 기존에는 RLHF 주로 사용함. **RLHF**

- **RLHF의 3단계**

- ① **Supervised Fine Tuning:** prompt-response 로 base 모델 학습.
- ② **Reward Model:** (인간 선호 O 데이터, 인간 선호 X 데이터) 쌍에 대해 reward $r(c, x)$ 학습 ($r(c, x^w) > r(c, x^l)$ 이도록)
- ③ **RL fine-tuning:** base 모델을 Reward Model로 변형.

- 문제점

- ① RL 이 on-policy 라 불안정. 고비용
- ② 여러 큰 모델(SFT/reward/value) 유지해야
- ③ reward 는 대리 지표 $\Rightarrow$ reward hacking.
- ④ 인간이 선호하는 데이터=승자($\longleftrightarrow$ 패자)

$\Rightarrow$ DPO의 도입

8.5 RLHF $\rightarrow$ DPO

Stage 2: Reward Model

- **notation**

- ▶ $\tilde{x}, x$: 평가하고자 하는 데이터
- ▶ c : 조건
- ▶ y : $\tilde{x}$ 가 x 보다 좋은 정도

- **training objective:** logistic loss

$$\mathcal{L}_{\text{RM}}(\psi) = -\mathbb{E}_{(c, \tilde{x}, x, y) \sim \mathcal{D}} \left[y \log \sigma(\Delta r) + (1 - y) \log (1 - \sigma(\Delta r)) \right],$$

$$\Delta r = r_{\psi}(c, \tilde{x}) - r_{\psi}(c, x), \quad \sigma(u) = 1/(1 + e^{-u})$$

- 승자의 reward가 패자에 비해 클수록 loss가 작아지므로 자연스러운 정의
- BT model 관점으로도 해석 가능

BT model

$$p_{r_{\psi}}(\tilde{x} \succ x \mid c) = \sigma(r_{\psi}(c, \tilde{x}) - r_{\psi}(c, x)) - \text{점수 차이만 중요.}$$

training objective의 해석

- 각 항은 관측 label 과 예측 승률 사이의 cross-entropy (8.5.3):

$$\mathcal{L}_{\text{RM}}(\psi) = \mathbb{E}_{\mathcal{D}} \left[D_{\text{KL}}(\text{Bern}(y) \parallel \text{Bern}(p_{r_\psi})) \right] + \underbrace{\mathbb{E}_{\mathcal{D}} [\mathbb{H}(\text{Bern}(y))]}_{\psi \text{ 무관}}.$$

⇒ 로지스틱 loss 최소화 $\equiv$ 사람의 승률과 예측 승률 사이의 KL 최소화.

Stage 3: Fine tuning

- Reinforcement learning이 도입되는 stage
- **training objective:** 고정된 reward $r := r_{\psi^*}$ 로 π_{θ} 를 fine-tune (8.5.4):

$$\max_{\theta} \mathbb{E}_{c \sim p(c)} \left[\mathbb{E}_{x \sim \pi_{\theta}(\cdot | c)} [r(c, x)] - \beta D_{\text{KL}}(\pi_{\theta}(\cdot | c) \parallel \pi_{\text{ref}}(\cdot | c)) \right],$$

이때 stage 1에서 학습한 모델이 $\pi_{\text{ref}}(x | c) := p_{\phi^*}(x | c)$

- Reward model이 선호하는 샘플을 찾되, reference에서 너무 멀어지지 않도록 함

8.5 RLHF → DPO

각 prompt c 별로 (8.5.4) 의 closed form이 존재 (8.5.5):

식 8.5.4의 closed form

$$\pi^*(x | c) = \frac{1}{Z(c)} \pi_{\text{ref}}(x | c) \exp(r(c, x)/\beta).$$

- β 작을수록 $\exp$ 가 뽀족 $\Rightarrow$ high-reward 집중(reference 에서 멀어짐), 불안정·reward hacking 위험.

이를 뒤집어 임의의 policy π 에 대한 **implicit reward** 정의 (8.5.6):

implicit reward

$$r_{\pi}(c, x) = \beta \log \frac{\pi(x | c)}{\pi_{\text{ref}}(x | c)} + \beta \log Z(c).$$

reward model을 학습하지 않고, 위 식을 L_{RM} 에 대입하여 새로운 loss 정의
 $\Rightarrow$ **reward 모델 · RL 없이** 단일 분류형 fine-tune 으로 RLHF 목표 달성.

8.5 RLHF → DPO

새로운 training objective 유도

1

$$r_{\pi}(c, x^w) - r_{\pi}(c, x^l) = \beta \left(\log \frac{\pi(x^w|c)}{\pi_{\text{ref}}(x^w|c)} - \log \frac{\pi(x^l|c)}{\pi_{\text{ref}}(x^l|c)} \right).$$

2 implicit reward 를 BT loss (8.5.2) 에 대입

$\mathcal{L}_{DPO}$

$$\mathcal{L}_{DPO}(\theta; \pi_{\text{ref}}) = -\mathbb{E}_{(c, x^w, x^l) \sim \mathcal{D}} \left[\log \sigma \left(\beta \left(\log \frac{\pi_{\theta}(x^w|c)}{\pi_{\text{ref}}(x^w|c)} - \log \frac{\pi_{\theta}(x^l|c)}{\pi_{\text{ref}}(x^l|c)} \right) \right) \right].$$

- 의미: reference 대비 (승자의 로그우도 개선) – (패자의 개선) 을 키움.

8.5 RLHF → DPO

하지만...위의 방법을 그대로 Diffusion model에 적용할 수는 없음

- diffusion model에서 $\pi_\theta(x_0 | c)$ 를 구하려면 전체 trajectory $x_{1:T}$ 에 대해 marginalization해야함 (또는 ODE change-of-variables) 필요 $\Rightarrow$ intractable.

trajectory까지 모두 포함한 분포로 바꾸면 어떨까?

- 최종 출력 x_0 대신 **경로** $x_{0:T} = (x_T, \dots, x_0)$ 로 확장.

$\mathcal{L}_{\text{Diff-DPO}}$

$$\mathcal{L}_{\text{Diff-DPO}}(\theta; \pi_{\text{ref}}) := -\mathbb{E}_{(\mathbf{c}, \mathbf{x}_0^w, \mathbf{x}_0^l) \sim \mathcal{D}} \left[\log \sigma(\Delta R(\mathbf{c}; \theta)) \right]$$

, where

$$\Delta R(\mathbf{c}; \theta) := \underbrace{\mathbb{E}_{\mathbf{x}_{1:T}^w \sim \pi_\theta(\cdot | \mathbf{x}_0^w, \mathbf{c})} \left[R_{\pi_\theta}(\mathbf{c}, (\mathbf{x}_0^w, \mathbf{x}_{1:T}^w)) \right]}_{\text{winner path expectation}} - \underbrace{\mathbb{E}_{\mathbf{x}_{1:T}^l \sim \pi_\theta(\cdot | \mathbf{x}_0^l, \mathbf{c})} \left[R_{\pi_\theta}(\mathbf{c}, (\mathbf{x}_0^l, \mathbf{x}_{1:T}^l)) \right]}_{\text{loser path expectation}}$$

$$R_\pi(\mathbf{c}, \mathbf{x}_{0:T}) := \beta \log \frac{\pi(\mathbf{x}_{0:T} | \mathbf{c})}{\pi_{\text{ref}}(\mathbf{x}_{0:T} | \mathbf{c})} + \beta \log Z(\mathbf{c})$$

- 문제점

- 1 Diffusion bridge: x_0 에 고정된 inverse trajectory posterior — 닫힌 해 없음.
- 2 θ 가 두 번 등장: ∇_θ 가 샘플링 분포까지 미분 $\Rightarrow$ 분산이 큰 REINFORCE항이 생김.

8.5 RLHF → DPO

- ① $-\log \sigma(\cdot)$ 는 convex $\Rightarrow$ Jensen inequality 사용해서 내부의 기댓값을 밖으로 뺀 (8.5.9):

$$\mathcal{L}_{\text{Diff-DPO}} \leq -\mathbb{E} \left[\log \sigma \left(\beta \left(\log \frac{\pi_{\theta}(x_{0:T}^w | c)}{\pi_{\text{ref}}(x_{0:T}^w | c)} - \log \frac{\pi_{\theta}(x_{0:T}^l | c)}{\pi_{\text{ref}}(x_{0:T}^l | c)} \right) \right) \right].$$

- ② Markov 성질에 의해 $\pi(x_{0:T} | c) = \pi(x_T | c) \prod_{t=1}^T \pi(x_{t-1} | x_t, c)$

- ③ $\pi_{\theta}(x_T | c) = \pi_{\text{ref}}(x_T | c)$ 라 두면

$$\log \frac{\pi_{\theta}(x_{0:T} | c)}{\pi_{\text{ref}}(x_{0:T} | c)} = \sum_{t=1}^T \log \frac{\pi_{\theta}(x_{t-1} | x_t, c)}{\pi_{\text{ref}}(x_{t-1} | x_t, c)}$$

- ④ $t \sim U(1, \dots, T)$ 를 샘플링하고, 다시 Jensen's ineq을 적용:

$$-\log \sigma \left(\beta \sum_{t=1}^T \Delta_t \right) \leq \mathbb{E}_t \left[-\log \sigma(\beta T \Delta_t) \right], \quad \Delta_t = \log \frac{\pi_{\theta}(x_{t-1}^w | x_t^w, c)}{\pi_{\text{ref}}(x_{t-1}^w | x_t^w, c)} - \log \frac{\pi_{\theta}(x_{t-1}^l | x_t^l, c)}{\pi_{\text{ref}}(x_{t-1}^l | x_t^l, c)}.$$

8.5 RLHF → DPO

가우시안 reverse conditional (ϵ -prediction) 에서

$$\log \frac{\pi_{\theta}(x_{t-1} \mid x_t, c)}{\pi_{\text{ref}}(x_{t-1} \mid x_t, c)} = \text{Const} - \lambda_t \left(\|\hat{\epsilon}_{\theta}(x_t, t, c) - \epsilon\|^2 - \|\hat{\epsilon}_{\text{ref}}(x_t, t, c) - \epsilon\|^2 \right).$$

$\Delta\text{MSE}(x_t; \epsilon) := \|\hat{\epsilon}_{\theta}(x_t, t, c) - \epsilon\|^2 - \|\hat{\epsilon}_{\text{ref}}(x_t, t, c) - \epsilon\|^2$ 로 두면

Diffusion-DPO 최종 목적함수

$$\tilde{\mathcal{L}}_{\text{Diff-DPO}} = -\mathbb{E}_{\substack{(c, x_0^w, x_0^l) \sim \mathcal{D} \\ t \sim U\{1, \dots, T\}, \epsilon^w, \epsilon^l \sim \mathcal{N}(0, I)}} \left[\log \sigma \left(w(t) (\Delta\text{MSE}(x_t^l; \epsilon^l) - \Delta\text{MSE}(x_t^w; \epsilon^w)) \right) \right],$$

$$x_t^w = \alpha_t x_0^w + \sigma_t \epsilon^w, \quad x_t^l = \alpha_t x_0^l + \sigma_t \epsilon^l, \quad w(t) \propto \beta T \lambda_t.$$

- **해석:** frozen reference 대비 **승자의 denoising 오차를 패자보다 더 많이 줄이도록** 학습. reference 는 직접 gradient 는 주지 않지만 margin 안에서 가중치를 형성.

- **통합 원리:** 조건부 스코어 = 무조건 방향 + steering 항 (Bayes 분해).
- **Control:** CG(외부 분류기), CFG(단일 모델), Training-Free(대리 likelihood / inverse problem).
- **Alignment:** RLHF → DPO → Diffusion-DPO — reward 모델·RL 을 제거한 안정적 정렬.
- **주의:** guidance 는 원칙상 정확한 tilted 분포 샘플링이 아니라 휴리스틱.

Remark: Guided PF-ODE과 tilted distribution I

tilted distribution의 스코어를 PF-ODE에 대입한다고 해서 tilted distribution에서 샘플링되지 않는다.

Recall: PF-ODE



$$dx/dt = v_t^{orig}(x), \quad v_t^{orig}(x) = f(t)x - \frac{1}{2}g^2(t)\nabla \log p_t(x)$$

- 이때 $p_t(x)$ 는 연속방정식을 만족하므로, 위의 식을 수치적분하면 $p_t(x)$ 을 구할 수 있음이 알려져있음.
- 연속방정식: $\partial_t p_t + \nabla \cdot (p_t v_t^{orig}) = 0$

하지만 tilted 분포는 연속방정식을 만족하지 않는다.

$$\partial_t \tilde{p}_t^{tilt} + \nabla \cdot (\tilde{p}_t^{tilt} v_t^{tilt}) = \tilde{p}_t^{tilt} \left[\partial_t \log h_t + v_t^{orig} \cdot \nabla \log h_t - \frac{1}{2}g^2(t) (\Delta \log h_t + \|\nabla \log h_t\|^2 + \nabla \log p_t \cdot \nabla \log h_t) - \frac{Z'_t}{Z_t} \right]$$

- $w_t = 0$ 일 때, 즉 무조건부 분포이면 위 식이 0이 됨
- 하지만 그렇지 않을 때 0이 될 이유가 없음
- 따라서 tilted distribution에 대해 PF-ODE 적용했을 때 해당 분포에서 샘플링되지는 않음.

Remark: Guided PF-ODE과 tilted distribution I

$\partial_t \tilde{p}_t^{tilt} + \nabla \cdot (\tilde{p}_t^{tilt} v_t^{tilt})$ 를 계산해보자.

① Part A: $\partial_t \tilde{p}_t^{tilt}$ 계산 $\tilde{p}_t^{tilt} = \frac{p_t h_t}{Z_t}$ 에 product rule 적용:

$$\partial_t \tilde{p}_t^{tilt} = \frac{(\partial_t p_t) h_t + p_t (\partial_t h_t)}{Z_t} - \frac{p_t h_t \cdot Z'_t}{Z_t^2}$$

$\frac{p_t h_t}{Z_t} = \tilde{p}_t^{tilt}$ 로 묶으면:

$$\partial_t \tilde{p}_t^{tilt} = \tilde{p}_t^{tilt} \left[\frac{\partial_t p_t}{p_t} + \partial_t \log h_t - \frac{Z'_t}{Z_t} \right]$$

② Part B: $\nabla \cdot (\tilde{p}_t^{tilt} v_t^{tilt})$ 계산
 $v_t^{tilt} = v_t^{orig} - \frac{1}{2} g^2 \nabla \log h_t$ 이므로 선형 분리:

$$\nabla \cdot (\tilde{p}_t^{tilt} v_t^{tilt}) = \underbrace{\nabla \cdot (\tilde{p}_t^{tilt} v_t^{orig})}_{B1} - \frac{1}{2} g^2(t) \underbrace{\nabla \cdot (\tilde{p}_t^{tilt} \nabla \log h_t)}_{B2}$$

Remark: Guided PF-ODE과 tilted distribution II

3 B1: $\nabla \cdot (\tilde{p}_t^{tilt} v_t^{orig})$
 $\tilde{p}_t^{tilt} = \frac{h_t}{Z_t} \cdot p_t$ 로 쓰면:

$$\nabla \cdot \left(\frac{h_t}{Z_t} p_t v_t^{orig} \right) = \frac{h_t}{Z_t} \nabla \cdot (p_t v_t^{orig}) + p_t v_t^{orig} \cdot \nabla \left(\frac{h_t}{Z_t} \right)$$

두 번째 항: $\nabla \left(\frac{h_t}{Z_t} \right) = \frac{\nabla h_t}{Z_t} = \frac{h_t}{Z_t} \nabla \log h_t$ 이므로:

$$\nabla \cdot (\tilde{p}_t^{tilt} v_t^{orig}) = \frac{h_t}{Z_t} \nabla \cdot (p_t v_t^{orig}) + \tilde{p}_t^{tilt} v_t^{orig} \cdot \nabla \log h_t$$

Remark: Guided PF-ODE과 tilted distribution III

❶ B2: $\nabla \cdot (\tilde{p}_t^{tilt} \nabla \log h_t)$

Divergence의 product rule: $\nabla \cdot (\rho \mathbf{u}) = \mathbf{u} \cdot \nabla \rho + \rho \nabla \cdot \mathbf{u}$ 를 적용:

$$\nabla \cdot (\tilde{p}_t^{tilt} \nabla \log h_t) = \underbrace{\nabla \log h_t \cdot \nabla \tilde{p}_t^{tilt}}_{\text{이 항이 핵심}} + \tilde{p}_t^{tilt} \underbrace{\nabla \cdot (\nabla \log h_t)}_{=\Delta \log h_t}$$

여기서 $\nabla \tilde{p}_t^{tilt}$ 를 전개한다. Z_t 는 x 와 무관하므로:

$$\nabla \tilde{p}_t^{tilt} = \tilde{p}_t^{tilt} \nabla \log \tilde{p}_t^{tilt} = \tilde{p}_t^{tilt} (\nabla \log p_t + \nabla \log h_t)$$

대입하면:

$$\nabla \log h_t \cdot \nabla \tilde{p}_t^{tilt} = \tilde{p}_t^{tilt} \nabla \log h_t \cdot \underbrace{(\nabla \log p_t + \nabla \log h_t)}_{\nabla \log \tilde{p}_t^{tilt}}$$

따라서:

$$\nabla \cdot (\tilde{p}_t^{tilt} \nabla \log h_t) = \tilde{p}_t^{tilt} [\nabla \log p_t \cdot \nabla \log h_t + \|\nabla \log h_t\|^2 + \Delta \log h_t]$$

Remark: Guided PF-ODE과 tilted distribution IV

5 최종 조합: (A) + (B1) - $\frac{1}{2}g^2(B2)$

$$\begin{aligned}\partial_t \tilde{p}_t^{tilt} + \nabla \cdot (\tilde{p}_t^{tilt} v_t^{tilt}) &= \tilde{p}_t^{tilt} \left[\frac{\partial_t p_t}{p_t} + \partial_t \log h_t - \frac{Z'_t}{Z_t} \right] + \frac{h_t}{Z_t} \nabla \cdot (p_t v_t^{orig}) + \tilde{p}_t^{tilt} v_t^{orig} \cdot \nabla \log h_t \\ &\quad - \frac{1}{2} g^2(t) \tilde{p}_t^{tilt} [\nabla \log p_t \cdot \nabla \log h_t + \|\nabla \log h_t\|^2 + \Delta \log h_t]\end{aligned}$$

첫 항과 셋째 항을 묶으면:

$$\tilde{p}_t^{tilt} \frac{\partial_t p_t}{p_t} + \frac{h_t}{Z_t} \nabla \cdot (p_t v_t^{orig}) = \frac{h_t}{Z_t} \underbrace{\left[\partial_t p_t + \nabla \cdot (p_t v_t^{orig}) \right]}_{=0 \text{ by } (*)} = 0$$

이 두 항이 상쇄되므로 :

$$= \tilde{p}_t^{tilt} \left[\partial_t \log h_t + v_t^{orig} \cdot \nabla \log h_t - \frac{1}{2} g^2(t) (\Delta \log h_t + \|\nabla \log h_t\|^2 + \nabla \log p_t \cdot \nabla \log h_t) - \frac{Z'_t}{Z_t} \right]$$

● **Q. guided dynamics와 desired tilted distribution 사이의 미스매치를 보완하는 방법(p.231)에 대해서 추가적인 설명 부탁드립니다.**

● A. Feynman-Kac Correctors(2025)

- ▶ CFG에서 사용하는 분포는 조건부 분포랑 무조건부 분포의 log probability를 보간한 분포인 셈. ($p_{t,\beta}^{geo}$)
- ▶ $\partial p_{t,\beta}^{geo} / \partial t =$ Fokker plank 식에 $p_{t,\beta}^{geo}$ 를 대입한 것 + 잉여항
- ▶ 잉여항은 다음과 같은 closed form으로 나타낼 수 있음:

$$\bar{g}_t(x) = \frac{\sigma_t^2}{2} \beta(\beta - 1) \|\nabla \log q_t^1(x) - \nabla \log q_t^2(x)\|^2, \quad q_t^1 : \text{조건부 분포}, q_t^2 : \text{무조건부 분포}$$

- ▶ 잉여항을 importance weight으로 흡수:

$$dx_t = v_t^{tilt}(x_t) dt + \sigma_t dW_t \quad (\text{SDE는 그대로})$$

$$dw_t = \bar{g}_t(x_t) dt \quad (\text{weight를 누적})$$

- **Q. Guidance를 줄 때 conditioning variable c 는 보통 discrete한가요? 책에 언급된 일본풍 그림 같은 경우에는 그림 스타일이라는 discrete한 조건에 컨디셔닝을 하는 것 같은데, continuous한 집합에 컨디셔닝하는 경우도 있는지 궁금합니다.**
- **A.** CG에서의 c 는 discrete하지만 그 외 방법에서는 프롬프트 등 continuous한 조건이 사용됩니다. 조건을 embedding vector로 변환해서 입력하는 것이 일반적입니다.
- **A.** Unet과 같은 structured network에 embedding vector로 들어가기 때문에 discrete/continuous 관계 없이 가능합니다. 예시로 MNIST 데이터에서 label 0 9 를 given 할수도 있고 (discrete), inpainting에서는 masked image pixel value 를 given 할수도 있습니다 (continuous).

- Q. P.231에서 언급되는 **correction-based approaches**들이 **Dynamics**를 **tilted distribution**을 따르는 방향으로 보정함에도 불구하고 현재 잘 쓰이지 않는 이유는 무엇인가요?
- A. 고차원에서는 계산비용이 크고 정확도가 향상되지 않아서 이미지 diffusion model에서는 잘 쓰이지 않습니다. 반면 분자 설계, 물리 시뮬레이션 등 상대적으로 저차원 문제에서는 사용합니다.

- Q. p.242의 DPS approximation은 "tightly peaked around its mean" 가정에 의존하는데, 이 가정이 성립하지 않을 때 발생하는 approximation error를 이론적으로 분석하거나 이를 개선한 연구가 있는지 궁금합니다.
- A. Guidance with Spherical Gaussian Constraint for Conditional Diffusion(2024, ICML)
 - ① 오차 분석 Jensen 부등식에서 기댓값을 함수 안으로 넣으면서 오차 발생

$$\mathbb{E}_{p(x_0|x_t)}[L(x_0)] \approx L(\mathbb{E}_{p(x_0|x_t)}[x_0])$$

오차의 하한을 구함: Loss L 이 β -strongly convex이고, $x_0|x_t \sim \mathcal{N}(\mu, \Sigma)$ 이면

$$\mathcal{J} = \mathbb{E}[L(x_0)] - L(\mathbb{E}[x_0]) \geq \frac{\beta}{2} \sum_{i=1}^n \lambda_i \quad (\text{이때 } \lambda \text{는 } \Sigma \text{의 고유값})$$

② 새로운 모델 제안

$$\text{DSG: } x_{t \rightarrow t-1} = \underbrace{\mu_\theta(x_t)}_{\text{DDIM mean}} \underbrace{- \sqrt{n} \sigma_t \cdot \frac{\nabla_{x_t} \ell(\hat{x}_0(x_t), y)}{\|\nabla_{x_t} \ell(\hat{x}_0(x_t), y)\|}}_{\text{normalized, geometry-based step size}}$$

$$(\text{비교}) \text{ DPS: } x_{t \rightarrow t-1} = \underbrace{\mu_\theta(x_t)}_{\text{DDIM mean}} \underbrace{- \gamma \nabla_{x_t} \ell(\hat{x}_0(x_t), y)}_{\eta_t^{\text{latent}} G_t}$$

- **Q. Training Free Guidance에서 스타일, 구도 등 여러 종류의 loss를 동시에 사용할 수도 있나요?**
- A. 네. 여러 조건에 대한 loss를 weighted sum으로 정의하면, 전체 guidance도 각 guidance signal의 weighted sum 형태로 구성할 수 있습니다.
다만 이를 각 조건에 대한 surrogate likelihood의 곱으로 해석할 경우, 조건들 사이의 dependency 때문에 단순한 factorization이 실제 joint conditional distribution을 정확히 나타내지 못할 수 있습니다.
또한 optimization 관점에서는 각 loss의 gradient가 서로 반대 방향을 가리키거나 scale이 크게 다를 경우 gradient conflict나 scale imbalance가 발생할 수 있습니다. 이를 완화하기 위해 loss별 weight 조정, timestep dependent guidance strength, gradient normalization, adaptive weighting 등의 방법이 연구되고 있습니다.