Deep Generative Models

Lecture 6

Roman Isachenko

Moscow Institute of Physics and Technology
Yandex School of Data Analysis

2025, Autumn

Recap of Previous Lecture

Likelihood-Free Learning

- Likelihood isn't a perfect metric for generative models.
- Likelihood may be intractable.

Imagine we have two sets of samples:

- $\{\mathbf{x}_i\}_{i=1}^{n_1} \sim p_{\text{data}}(\mathbf{x})$ – real samples;
- $\{\mathbf{x}_i\}_{i=1}^{n_2} \sim p_{\boldsymbol{\theta}}(\mathbf{x})$ – generated (fake) samples.

$$p(y = 1|\mathbf{x}) = P(\mathbf{x} \sim p_{\text{data}}(\mathbf{x})); \quad p(y = 0|\mathbf{x}) = P(\mathbf{x} \sim p_{\boldsymbol{\theta}}(\mathbf{x}))$$

Assumption

The generative distribution $p_{\boldsymbol{\theta}}(\mathbf{x})$ matches the true distribution $p_{\text{data}}(\mathbf{x})$ if we can't distinguish between them using a discriminative model $p(y|\mathbf{x})$.

- **Generator:** a generative model $\mathbf{x} = \mathbf{G}(\mathbf{z})$ that produces more realistic samples.
- **Discriminator:** a classifier $D(\mathbf{x}) \in [0, 1]$ distinguishing real from generated samples.

Recap of Previous Lecture

GAN Optimality Theorem

The minimax game

$$\min_{G} \max_{D} \Bigl[\underbrace{\mathbb{E}_{p_{\text{data}}(\mathbf{x})} \log D(\mathbf{x}) + \mathbb{E}_{p(\mathbf{z})} \log(1 - D(\mathbf{G}(\mathbf{z})))}_{V(G,D)}\Bigr]$$

has a global optimum at $p_{\text{data}}(\mathbf{x}) = p_{\boldsymbol{\theta}}(\mathbf{x})$, and then $D^*(\mathbf{x}) = 0.5$.

$$\min_{G} V(G, D^*) = \min_{G} \left[2\mathrm{JSD}(\pi \| p) - \log 4\right] = -\log 4, \quad p_{\text{data}}(\mathbf{x}) = p_{\boldsymbol{\theta}}(\mathbf{x}).$$

If the generator can be **any** function and the discriminator is **optimal** at each step, then the generator is **guaranteed to converge** to the data distribution.

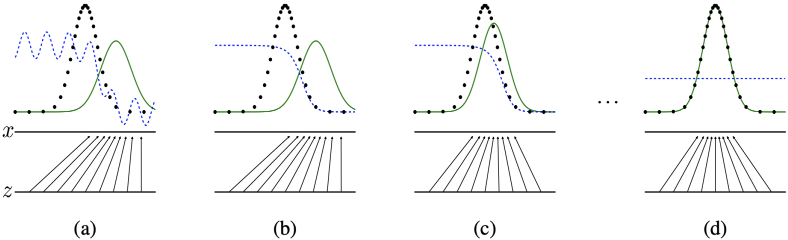
Goodfellow I. J. et al. Generative Adversarial Networks, 2014

Recap of Previous Lecture

- The generator is updated in the parameter space; the discriminator isn't optimal at every iteration.
- Both generator and discriminator loss typically oscillate during GAN training.

Objective

$$\min_{\boldsymbol{\theta}} \max_{\boldsymbol{\phi}} \left[\mathbb{E}_{p_{\text{data}}(\mathbf{x})} \log D_{\boldsymbol{\phi}}(\mathbf{x}) + \mathbb{E}_{p(\mathbf{z})} \log(1 - D_{\boldsymbol{\phi}}(\mathbf{G}_{\boldsymbol{\theta}}(\mathbf{z}))) \right]$$



Goodfellow I. J. et al. Generative Adversarial Networks, 2014

Recap of Previous Lecture

Main Issues With Standard GANs

- Vanishing gradients (solution: non-saturating GAN).
- Mode collapse (arises from Jensen-Shannon divergence).

Standard GAN

$$\min_{\boldsymbol{\theta}} \max_{\boldsymbol{\phi}} \left[\mathbb{E}_{p_{\text{data}}(\mathbf{x})} \log D_{\boldsymbol{\phi}}(\mathbf{x}) + \mathbb{E}_{p(\mathbf{z})} \log(1 - D_{\boldsymbol{\phi}}(\mathbf{G}_{\boldsymbol{\theta}}(\mathbf{z}))) \right]$$

Informal Theoretical Results

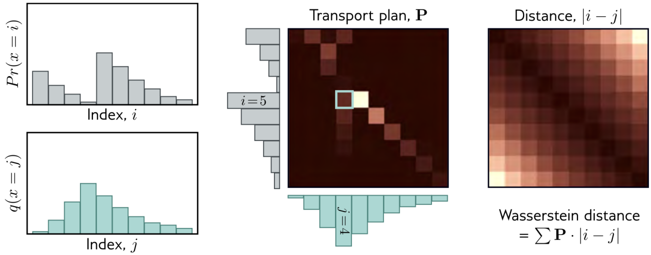
Both the data distribution $p_{\text{data}}(\mathbf{x})$ and the generative distribution $p_{\boldsymbol{\theta}}(\mathbf{x})$ are low-dimensional with disjoint supports. In such cases,

$$\mathrm{KL}(p_{\text{data}} \| p_{\boldsymbol{\theta}}) = \mathrm{KL}(p_{\boldsymbol{\theta}} \| p_{\text{data}}) = \infty, \quad \mathrm{JSD}(p_{\text{data}} \| p_{\boldsymbol{\theta}}) = \log 2.$$

Goodfellow I. J. et al. Generative Adversarial Networks, 2014

Arjovsky M., Bottou L. Towards Principled Methods for Training Generative Adversarial Networks, 2017

Recap of Previous Lecture



Wasserstein Distance

$$W(\pi \| p) = \inf_{\gamma \in \Gamma(\pi, p)} \mathbb{E}_{(\mathbf{x}_1, \mathbf{x}_2) \sim \gamma} \|\mathbf{x}_1 - \mathbf{x}_2\| = \inf_{\gamma \in \Gamma(\pi, p)} \int \|\mathbf{x}_1 - \mathbf{x}_2\| \gamma(\mathbf{x}_1, \mathbf{x}_2) d\mathbf{x}_1 d\mathbf{x}_2$$

- $\gamma(\mathbf{x}_1, \mathbf{x}_2)$ – transportation plan (amount of "dirt" to transport from $\mathbf{x}_1$ to $\mathbf{x}_2$).
- $\Gamma(\pi, p)$ – set of all joint distributions $\gamma(\mathbf{x}_1, \mathbf{x}_2)$ with marginals π and p ($\int \gamma(\mathbf{x}_1, \mathbf{x}_2) d\mathbf{x}_1 = p(\mathbf{x}_2)$, $\int \gamma(\mathbf{x}_1, \mathbf{x}_2) d\mathbf{x}_2 = \pi(\mathbf{x}_1)$).
- $\gamma(\mathbf{x}_1, \mathbf{x}_2)$ – the amount; $\|\mathbf{x}_1 - \mathbf{x}_2\|$ – the distance.

Simon J.D. Prince. Understanding Deep Learning, 2023

Recap of Previous Lecture

Theorem (Kantorovich-Rubinstein Duality)

$$W(p_{\text{data}} \| p_{\boldsymbol{\theta}}) = \frac{1}{K} \max_{\|f\|_L \leq K} \left[\mathbb{E}_{p_{\text{data}}(\mathbf{x})} f(\mathbf{x}) - \mathbb{E}_{p_{\boldsymbol{\theta}}(\mathbf{x})} f(\mathbf{x}) \right],$$

where $\|f\|_L \leq K$ denotes K-Lipschitz continuous functions.

WGAN Objective

$$\min_{\boldsymbol{\theta}} W(p_{\text{data}} \| p_{\boldsymbol{\theta}}) = \min_{\boldsymbol{\theta}} \max_{\phi \in \boldsymbol{\Phi}} \left[\mathbb{E}_{p_{\text{data}}(\mathbf{x})} f_{\phi}(\mathbf{x}) - \mathbb{E}_{p(\mathbf{z})} f_{\phi}(\mathbf{G}_{\boldsymbol{\theta}}(\mathbf{z})) \right].$$

- The function f in WGAN is called the *critic*.
- If parameters ϕ lie in a compact set $\boldsymbol{\Phi} \in [-c, c]^d$, then $f(\mathbf{x}, \phi)$ is K-Lipschitz continuous.

$$K \cdot W(p_{\text{data}} \| p_{\boldsymbol{\theta}}) = \max_{\|f\|_L \leq K} \left[\mathbb{E}_{p_{\text{data}}(\mathbf{x})} f(\mathbf{x}) - \mathbb{E}_{p_{\boldsymbol{\theta}}(\mathbf{x})} f(\mathbf{x}) \right] \geq$$
$$\geq \max_{\phi \in \boldsymbol{\Phi}} \left[\mathbb{E}_{p_{\text{data}}(\mathbf{x})} f_{\phi}(\mathbf{x}) - \mathbb{E}_{p_{\boldsymbol{\theta}}(\mathbf{x})} f_{\phi}(\mathbf{x}) \right]$$

Arjovsky M., Chintala S., Bottou L. Wasserstein GAN, 2017

Outline

1. Evaluation of Likelihood-Free Models
   - Frechet Inception Distance (FID)
   - Precision-Recall
   - CLIP Score
   - Human Evaluation
2. Langevin Dynamics
3. Score Matching
4. Denoising Score Matching

Outline

1. Evaluation of Likelihood-Free Models
   - Frechet Inception Distance (FID)
   - Precision-Recall
   - CLIP Score
   - Human Evaluation
2. Langevin Dynamics
3. Score Matching
4. Denoising Score Matching

Evaluation of Likelihood-Free Models

Likelihood-Based Models

- **Train:** fit the model.
- **Validation:** tune hyperparameters.
- **Test:** assess generalization by reporting likelihood.

image credit: https://deepgenerativemodels.github.io

Evaluation of Likelihood-Free Models

Likelihood-Based Models

- **Train:** fit the model.
- **Validation:** tune hyperparameters.
- **Test:** assess generalization by reporting likelihood.

Not all models have tractable likelihoods
(VAE: compare ELBO values; GAN: **???**).

---

image credit: https://deepgenerativemodels.github.io

Evaluation of Likelihood-Free Models

Likelihood-Based Models

- **Train:** fit the model.
- **Validation:** tune hyperparameters.
- **Test:** assess generalization by reporting likelihood.

Not all models have tractable likelihoods
(VAE: compare ELBO values; GAN: **???**).

Desirable Properties for Samples

- Sharpness

image credit: *https://deepgenerativemodels.github.io*

Evaluation of Likelihood-Free Models

Likelihood-Based Models

- **Train:** fit the model.
- **Validation:** tune hyperparameters.
- **Test:** assess generalization by reporting likelihood.

Not all models have tractable likelihoods
(VAE: compare ELBO values; GAN: **???**).

Desirable Properties for Samples

- Sharpness

Low sharpness

High sharpness

- Diversity

Low diversity

High diversity

image credit: https://deepgenerativemodels.github.io

Outline

Wasserstein Metric

$$W_s(\pi \| p) = \inf_{\gamma \in \Gamma(\pi, p)} \left(\mathbb{E}_{(\mathbf{x}_1, \mathbf{x}_2) \sim \gamma} \| \mathbf{x}_1 - \mathbf{x}_2 \|^s \right)^{1/s}$$

Heusel M. et al. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium, 2017

Wasserstein Metric

$$W_s(\pi\|p) = \inf_{\gamma\in\Gamma(\pi,p)} \left(\mathbb{E}_{(\mathbf{x}_1,\mathbf{x}_2)\sim\gamma}\|\mathbf{x}_1 - \mathbf{x}_2\|^s\right)^{1/s}$$

Wasserstein GAN (Optimal Transport)

$$W(\pi\|p) = \inf_{\gamma\in\Gamma(\pi,p)} \mathbb{E}_{(\mathbf{x}_1,\mathbf{x}_2)\sim\gamma}\|\mathbf{x}_1-\mathbf{x}_2\| = \inf_{\gamma\in\Gamma(\pi,p)} \int \|\mathbf{x}_1 - \mathbf{x}_2\|\gamma(\mathbf{x}_1,\mathbf{x}_2)d\mathbf{x}_1 d\mathbf{x}_2$$

Heusel M. et al. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium, 2017

Wasserstein Metric

$$W_s(\pi\|p) = \inf_{\gamma \in \Gamma(\pi,p)} \left(\mathbb{E}_{(\mathbf{x}_1,\mathbf{x}_2)\sim\gamma}\|\mathbf{x}_1 - \mathbf{x}_2\|^s\right)^{1/s}$$

Wasserstein GAN (Optimal Transport)

$$W(\pi\|p) = \inf_{\gamma \in \Gamma(\pi,p)} \mathbb{E}_{(\mathbf{x}_1,\mathbf{x}_2)\sim\gamma}\|\mathbf{x}_1 - \mathbf{x}_2\| = \inf_{\gamma \in \Gamma(\pi,p)} \int \|\mathbf{x}_1 - \mathbf{x}_2\|\gamma(\mathbf{x}_1,\mathbf{x}_2)d\mathbf{x}_1 d\mathbf{x}_2$$

Theorem

If $\pi(\mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}_\pi, \boldsymbol{\Sigma}_\pi)$, $p(\mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}_p, \boldsymbol{\Sigma}_p)$, then

$$W_2^2(\pi\|p) = \|\boldsymbol{\mu}_\pi - \boldsymbol{\mu}_p\|^2 + \mathrm{tr}\left[\boldsymbol{\Sigma}_\pi + \boldsymbol{\Sigma}_p - 2\left(\boldsymbol{\Sigma}_\pi^{1/2}\boldsymbol{\Sigma}_p\boldsymbol{\Sigma}_\pi^{1/2}\right)^{1/2}\right]$$

Heusel M. et al. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium, 2017

Wasserstein Metric

$$W_s(\pi\|p) = \inf_{\gamma \in \Gamma(\pi, p)} \left(\mathbb{E}_{(\mathbf{x}_1, \mathbf{x}_2) \sim \gamma} \|\mathbf{x}_1 - \mathbf{x}_2\|^s \right)^{1/s}$$

Wasserstein GAN (Optimal Transport)

$$W(\pi\|p) = \inf_{\gamma \in \Gamma(\pi, p)} \mathbb{E}_{(\mathbf{x}_1, \mathbf{x}_2) \sim \gamma} \|\mathbf{x}_1 - \mathbf{x}_2\| = \inf_{\gamma \in \Gamma(\pi, p)} \int \|\mathbf{x}_1 - \mathbf{x}_2\| \gamma(\mathbf{x}_1, \mathbf{x}_2) d\mathbf{x}_1 d\mathbf{x}_2$$

Theorem

If $\pi(\mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}_\pi, \boldsymbol{\Sigma}_\pi)$, $p(\mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}_p, \boldsymbol{\Sigma}_p)$, then

$$W_2^2(\pi\|p) = \|\boldsymbol{\mu}_\pi - \boldsymbol{\mu}_p\|^2 + \text{tr} \left[\boldsymbol{\Sigma}_\pi + \boldsymbol{\Sigma}_p - 2 \left(\boldsymbol{\Sigma}_\pi^{1/2} \boldsymbol{\Sigma}_p \boldsymbol{\Sigma}_\pi^{1/2} \right)^{1/2} \right]$$

Frechet Inception Distance

$$\mathrm{FID}(p_{\text{data}}, p_{\boldsymbol{\theta}}) = W_2^2(p_{\text{data}} \| p_{\boldsymbol{\theta}})$$

Heusel M. et al. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium, 2017

Frechet Inception Distance (FID)

$$\mathrm{FID}(p_{\text{data}}, p_{\boldsymbol{\theta}}) = \|\boldsymbol{\mu}_{\text{data}} - \boldsymbol{\mu}_{\boldsymbol{\theta}}\|^2 + \mathrm{tr}\left[\boldsymbol{\Sigma}_{\text{data}} + \boldsymbol{\Sigma}_{\boldsymbol{\theta}} - 2\left(\boldsymbol{\Sigma}_{\text{data}}^{1/2}\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\boldsymbol{\Sigma}_{\text{data}}^{1/2}\right)^{1/2}\right]$$

- FID is computed in the latent space $\mathbf{z}$.
- We use a pretrained image embedder to get latent representations $\mathbf{z} = \mathbf{f}(\mathbf{x})$.
- $\boldsymbol{\mu}_{\text{data}}$, $\boldsymbol{\Sigma}_{\text{data}}$ and $\boldsymbol{\mu}_{\boldsymbol{\theta}}$, $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}$ are statistics of latent representations for samples from $p_{\text{data}}(\mathbf{x})$ and $p_{\boldsymbol{\theta}}(\mathbf{x})$.

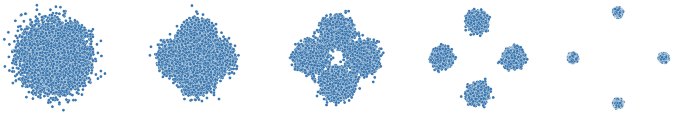
Jayasumana S. et al. Rethinking FID: Towards a Better Evaluation Metric for Image Generation, 2024

Frechet Inception Distance (FID)

$$\mathrm{FID}(p_{\text{data}}, p_{\boldsymbol{\theta}}) = \|\boldsymbol{\mu}_{\text{data}} - \boldsymbol{\mu}_{\boldsymbol{\theta}}\|^2 + \mathrm{tr}\left[\boldsymbol{\Sigma}_{\text{data}} + \boldsymbol{\Sigma}_{\boldsymbol{\theta}} - 2\left(\boldsymbol{\Sigma}_{\text{data}}^{1/2}\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\boldsymbol{\Sigma}_{\text{data}}^{1/2}\right)^{1/2}\right]$$

- FID is computed in the latent space $\mathbf{z}$.
- We use a pretrained image embedder to get latent representations $\mathbf{z} = \mathbf{f}(\mathbf{x})$.
- $\boldsymbol{\mu}_{\text{data}}$, $\boldsymbol{\Sigma}_{\text{data}}$ and $\boldsymbol{\mu}_{\boldsymbol{\theta}}$, $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}$ are statistics of latent representations for samples from $p_{\text{data}}(\mathbf{x})$ and $p_{\boldsymbol{\theta}}(\mathbf{x})$.

$FID(p(\mathbf{x}), \mathcal{N}(0, \mathbf{I}))$



FD	0.0	0.0	0.0	0.0	0.0

Jayasumana S. et al. Rethinking FID: Towards a Better Evaluation Metric for Image Generation, 2024

Frechet Inception Distance (FID)

$$\mathrm{FID}(p_{\mathrm{data}}, p_{\boldsymbol{\theta}}) = \|\boldsymbol{\mu}_{\mathrm{data}} - \boldsymbol{\mu}_{\boldsymbol{\theta}}\|^2 + \mathrm{tr}\left[\boldsymbol{\Sigma}_{\mathrm{data}} + \boldsymbol{\Sigma}_{\boldsymbol{\theta}} - 2\left(\boldsymbol{\Sigma}_{\mathrm{data}}^{1/2}\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\boldsymbol{\Sigma}_{\mathrm{data}}^{1/2}\right)^{1/2}\right]$$

Jayasumana S. et al. Rethinking FID: Towards a Better Evaluation Metric for Image Generation, 2024

Frechet Inception Distance (FID)

$$\mathrm{FID}(p_{\text{data}}, p_{\boldsymbol{\theta}}) = \|\boldsymbol{\mu}_{\text{data}} - \boldsymbol{\mu}_{\boldsymbol{\theta}}\|^2 + \mathrm{tr}\left[\boldsymbol{\Sigma}_{\text{data}} + \boldsymbol{\Sigma}_{\boldsymbol{\theta}} - 2\left(\boldsymbol{\Sigma}_{\text{data}}^{1/2}\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\boldsymbol{\Sigma}_{\text{data}}^{1/2}\right)^{1/2}\right]$$

Drawbacks

- Depends on the pretrained classification network.
- Uses the normality assumption.
- May not correlate with human evaluation.

Model	Model-A	Model-B
FID	21.40	18.42
FID_∞	20.16	17.19
Human rater preference	92.5%	6.9%

Jayasumana S. et al. Rethinking FID: Towards a Better Evaluation Metric for Image Generation, 2024

Outline

1. Evaluation of Likelihood-Free Models
   - Frechet Inception Distance (FID)
   - Precision-Recall
   - CLIP Score
   - Human Evaluation

2. Langevin Dynamics

3. Score Matching

4. Denoising Score Matching

Precision-Recall

Desirable Properties for Samples

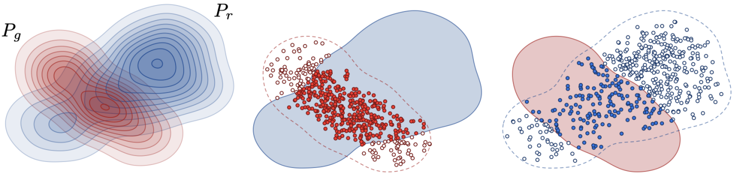
- **Sharpness**: generated samples should possess high visual quality.
- **Diversity**: their variation should match that in the training data.

Kynkäänniemi T. et al. Improved precision and recall metric for assessing generative models, 2019

Precision-Recall

Desirable Properties for Samples

- **Sharpness**: generated samples should possess high visual quality.
- **Diversity**: their variation should match that in the training data.



(a) Example distributions (b) Precision (c) Recall

- **Precision** denotes the fraction of generated images that look realistic.
- **Recall** measures how well the generator covers the training data manifold.

Kynkäänniemi T. et al. Improved precision and recall metric for assessing generative models, 2019

Precision-Recall

- $\mathcal{S}_{\text{data}} = \{\mathbf{x}_i\}_{i=1}^n \sim p_{\text{data}}(\mathbf{x})$ – real samples;
- $\mathcal{S}_{\boldsymbol{\theta}} = \{\mathbf{x}_i\}_{i=1}^n \sim p_{\boldsymbol{\theta}}(\mathbf{x})$ – generated samples.

Kynkäänniemi T. et al. Improved precision and recall metric for assessing generative models, 2019

Precision-Recall

- $\mathcal{S}_{\text{data}} = \{\mathbf{x}_i\}_{i=1}^n \sim p_{\text{data}}(\mathbf{x})$ – real samples;
- $\mathcal{S}_{\boldsymbol{\theta}} = \{\mathbf{x}_i\}_{i=1}^n \sim p_{\boldsymbol{\theta}}(\mathbf{x})$ – generated samples.

Define a binary function:

$$\mathbb{I}(\mathbf{x}, \mathcal{S}) = \begin{cases} 1, & \text{if } \exists\ \mathbf{x}' \in \mathcal{S} : \|\mathbf{x} - \mathbf{x}'\|_2 \leq \|\mathbf{x}' - \text{NN}_k(\mathbf{x}', \mathcal{S})\|_2; \\ 0, & \text{otherwise.} \end{cases}$$

Kynkäänniemi T. et al. Improved precision and recall metric for assessing generative models, 2019

Precision-Recall

- $\mathcal{S}_{\text{data}} = \{\mathbf{x}_i\}_{i=1}^n \sim p_{\text{data}}(\mathbf{x})$ – real samples;
- $\mathcal{S}_{\boldsymbol{\theta}} = \{\mathbf{x}_i\}_{i=1}^n \sim p_{\boldsymbol{\theta}}(\mathbf{x})$ – generated samples.

Define a binary function:

$$\mathbb{I}(\mathbf{x}, \mathcal{S}) = \begin{cases} 1, & \text{if } \exists \; \mathbf{x}' \in \mathcal{S} : \|\mathbf{x} - \mathbf{x}'\|_2 \leq \|\mathbf{x}' - \text{NN}_k(\mathbf{x}', \mathcal{S})\|_2; \\ 0, & \text{otherwise.} \end{cases}$$

$$\text{Pr}(\mathcal{S}_{\text{data}}, \mathcal{S}_{\boldsymbol{\theta}}) = \frac{1}{n} \sum_{\mathbf{x} \in \mathcal{S}_{\boldsymbol{\theta}}} \mathbb{I}(\mathbf{x}, \mathcal{S}_{\text{data}}); \quad \text{Rec}(\mathcal{S}_{\text{data}}, \mathcal{S}_{\boldsymbol{\theta}}) = \frac{1}{n} \sum_{\mathbf{x} \in \mathcal{S}_{\text{data}}} \mathbb{I}(\mathbf{x}, \mathcal{S}_{\boldsymbol{\theta}}).$$

Kynkäänniemi T. et al. Improved precision and recall metric for assessing generative models, 2019

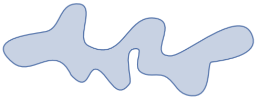
Precision-Recall

- $\mathcal{S}_{\text{data}} = \{\mathbf{x}_i\}_{i=1}^{n} \sim p_{\text{data}}(\mathbf{x})$ – real samples;
- $\mathcal{S}_{\boldsymbol{\theta}} = \{\mathbf{x}_i\}_{i=1}^{n} \sim p_{\boldsymbol{\theta}}(\mathbf{x})$ – generated samples.

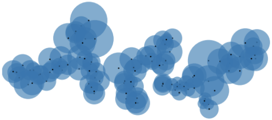
Define a binary function:

$$\mathbb{I}(\mathbf{x}, \mathcal{S}) = \begin{cases} 1, & \text{if } \exists\ \mathbf{x}' \in \mathcal{S} : \|\mathbf{x} - \mathbf{x}'\|_2 \leq \|\mathbf{x}' - \text{NN}_k(\mathbf{x}', \mathcal{S})\|_2; \\ 0, & \text{otherwise.} \end{cases}$$

$$\text{Pr}(\mathcal{S}_{\text{data}}, \mathcal{S}_{\boldsymbol{\theta}}) = \frac{1}{n} \sum_{\mathbf{x} \in \mathcal{S}_{\boldsymbol{\theta}}} \mathbb{I}(\mathbf{x}, \mathcal{S}_{\text{data}}); \quad \text{Rec}(\mathcal{S}_{\text{data}}, \mathcal{S}_{\boldsymbol{\theta}}) = \frac{1}{n} \sum_{\mathbf{x} \in \mathcal{S}_{\text{data}}} \mathbb{I}(\mathbf{x}, \mathcal{S}_{\boldsymbol{\theta}}).$$



(a) True manifold



(b) Approx. manifold

Kynkäänniemi T. et al. Improved precision and recall metric for assessing generative models, 2019

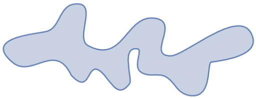
Precision-Recall

- $\mathcal{S}_{\text{data}} = \{\mathbf{x}_i\}_{i=1}^n \sim p_{\text{data}}(\mathbf{x})$ – real samples;
- $\mathcal{S}_{\boldsymbol{\theta}} = \{\mathbf{x}_i\}_{i=1}^n \sim p_{\boldsymbol{\theta}}(\mathbf{x})$ – generated samples.

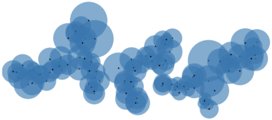
Define a binary function:

$$\mathbb{I}(\mathbf{x}, \mathcal{S}) = \begin{cases} 1, & \text{if } \exists \ \mathbf{x}' \in \mathcal{S} : \|\mathbf{x} - \mathbf{x}'\|_2 \leq \|\mathbf{x}' - \text{NN}_k(\mathbf{x}', \mathcal{S})\|_2; \\ 0, & \text{otherwise.} \end{cases}$$

$$\text{Pr}(\mathcal{S}_{\text{data}}, \mathcal{S}_{\boldsymbol{\theta}}) = \frac{1}{n} \sum_{\mathbf{x} \in \mathcal{S}_{\boldsymbol{\theta}}} \mathbb{I}(\mathbf{x}, \mathcal{S}_{\text{data}}); \quad \text{Rec}(\mathcal{S}_{\text{data}}, \mathcal{S}_{\boldsymbol{\theta}}) = \frac{1}{n} \sum_{\mathbf{x} \in \mathcal{S}_{\text{data}}} \mathbb{I}(\mathbf{x}, \mathcal{S}_{\boldsymbol{\theta}}).$$



(a) True manifold

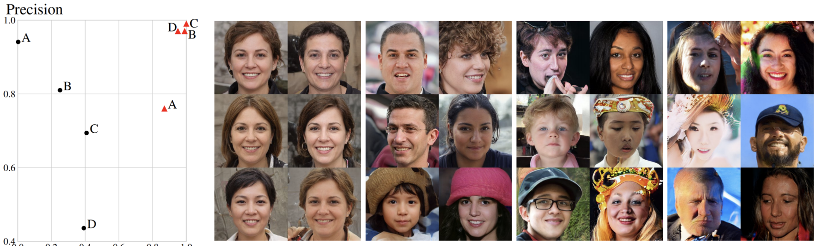


(b) Approx. manifold

Embed the samples using a pretrained network (as in FID).

Kynkäänniemi T. et al. Improved precision and recall metric for assessing generative models, 2019

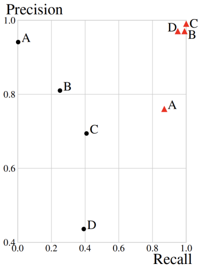
Precision-Recall



A (FID = 91.7) **B** (FID = 16.9) **C** (FID = 4.5) **D** (FID = 16.7)

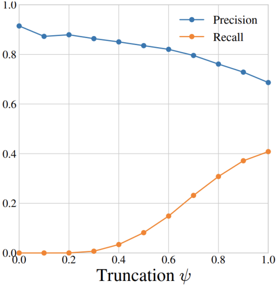
Kynkäänniemi T. et al. Improved precision and recall metric for assessing generative models, 2019

Precision-Recall



A (FID = 91.7) **B** (FID = 16.9) **C** (FID = 4.5) **D** (FID = 16.7)

$\psi = 0.0$ $\psi = 0.3$ $\psi = 0.7$ $\psi = 1.0$



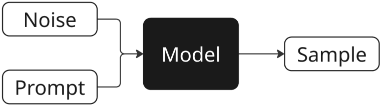
Kynkäänniemi T. et al. Improved precision and recall metric for assessing generative models, 2019

Outline

CLIP Score

Unconditional Model

Conditional Model

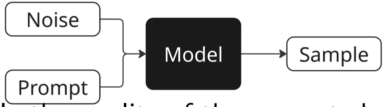


Radford A. et al. Learning transferable visual models from natural language supervision, 2021

CLIP Score

Unconditional Model

Conditional Model



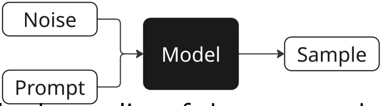
We need a way to measure not only the quality of the generated image, but also how well it's aligned with the prompt.

Radford A. et al. Learning transferable visual models from natural language supervision, 2021

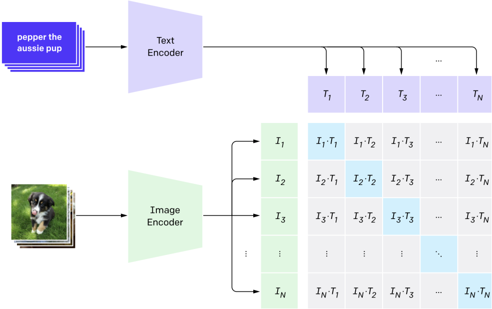
CLIP Score

Unconditional Model

Conditional Model



We need a way to measure not only the quality of the generated image, but also how well it's aligned with the prompt.



Radford A. et al. Learning transferable visual models from natural language supervision, 2021

Outline

Human Evaluation

- No automated metric is perfect.
- The best way to evaluate generative models is by human assessment.
- It's important to assess various properties.

YandexART 2.5, 2025

Human Evaluation

- No automated metric is perfect.
- The best way to evaluate generative models is by human assessment.
- It's important to assess various properties.

Аспект	Yandex ART 2.0	Mj 6.1	Mj 6	Ideogram	Recraft	Google Imagen3	Dall-E 3	FLUX	SBER Kandi3.1
Релевантность	0,59	0,58	0,63	0,45	0,51	0,50	0,50	0,54	0,75
Эстетика	0,49	0,55	0,55	0,51	0,51	0,61	0,61	0,54	0,59
Комплексность	0,44	0,73	0,70	0,68	0,76	0,75	0,75	0,71	0,74
Дефектность	0,69	0,57	0,68	0,55	0,59	0,63	0,63	0,50	0,75
Предпочтение	0,66	0,60	0,69	0,49	0,54	0,63	0,63	0,51	0,84

YandexART 2.5, 2025

Outline

Energy-Based Models

Unnormalized Density

$$p_{\boldsymbol{\theta}}(\mathbf{x}) = \frac{\hat{p}_{\boldsymbol{\theta}}(\mathbf{x})}{Z_{\boldsymbol{\theta}}}, \quad \text{where } Z_{\boldsymbol{\theta}} = \int \hat{p}_{\boldsymbol{\theta}}(\mathbf{x}) d\mathbf{x}$$

- $\hat{p}_{\boldsymbol{\theta}}(\mathbf{x})$ can be any non-negative function.
- If we reparameterize as $\hat{p}_{\boldsymbol{\theta}}(\mathbf{x}) = \exp(-f_{\boldsymbol{\theta}}(\mathbf{x}))$, we eliminate the non-negativity constraint.

Energy-Based Models

Unnormalized Density

$$p_{\boldsymbol{\theta}}(\mathbf{x}) = \frac{\hat{p}_{\boldsymbol{\theta}}(\mathbf{x})}{Z_{\boldsymbol{\theta}}}, \quad \text{where } Z_{\boldsymbol{\theta}} = \int \hat{p}_{\boldsymbol{\theta}}(\mathbf{x}) d\mathbf{x}$$

- $\hat{p}_{\boldsymbol{\theta}}(\mathbf{x})$ can be any non-negative function.
- If we reparameterize as $\hat{p}_{\boldsymbol{\theta}}(\mathbf{x}) = \exp(-f_{\boldsymbol{\theta}}(\mathbf{x}))$, we eliminate the non-negativity constraint.

Unnormalized Density

The gradient of the normalized log-density equals that of the unnormalized log-density:

$$\nabla_{\mathbf{x}} \log p_{\boldsymbol{\theta}}(\mathbf{x}) = \nabla_{\mathbf{x}} \log \hat{p}_{\boldsymbol{\theta}}(\mathbf{x}) - \nabla_{\mathbf{x}} \log Z_{\boldsymbol{\theta}} = \nabla_{\mathbf{x}} \log \hat{p}_{\boldsymbol{\theta}}(\mathbf{x})$$

Energy-Based Models

Unnormalized Density

$$p_{\boldsymbol{\theta}}(\mathbf{x}) = \frac{\hat{p}_{\boldsymbol{\theta}}(\mathbf{x})}{Z_{\boldsymbol{\theta}}}, \quad \text{where } Z_{\boldsymbol{\theta}} = \int \hat{p}_{\boldsymbol{\theta}}(\mathbf{x}) d\mathbf{x}$$

- $\hat{p}_{\boldsymbol{\theta}}(\mathbf{x})$ can be any non-negative function.
- If we reparameterize as $\hat{p}_{\boldsymbol{\theta}}(\mathbf{x}) = \exp(-f_{\boldsymbol{\theta}}(\mathbf{x}))$, we eliminate the non-negativity constraint.

Unnormalized Density

The gradient of the normalized log-density equals that of the unnormalized log-density:

$$\nabla_{\mathbf{x}} \log p_{\boldsymbol{\theta}}(\mathbf{x}) = \nabla_{\mathbf{x}} \log \hat{p}_{\boldsymbol{\theta}}(\mathbf{x}) - \nabla_{\mathbf{x}} \log Z_{\boldsymbol{\theta}} = \nabla_{\mathbf{x}} \log \hat{p}_{\boldsymbol{\theta}}(\mathbf{x})$$

- Suppose we already have this density (normalized or not) $p_{\boldsymbol{\theta}}(\mathbf{x})$.
- How can we sample from the model?

Langevin Dynamics

Theorem

Consider energy-based model $p(\mathbf{x}) = \frac{\hat{p}(\mathbf{x})}{Z}$, $\hat{p}(\mathbf{x}) = \exp(-f(\mathbf{x}))$, with continuously differentiable $f(\mathbf{x}) : \mathbb{R}^m \to \mathbb{R}$ that satisfies

- L-smoothness: $\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|$;
- Strong convexity: $(\nabla f(\mathbf{x}) - \nabla f(\mathbf{y}))^\top(\mathbf{x} - \mathbf{y}) \geq m\|\mathbf{x} - \mathbf{y}\|^2$ for some $m > 0$.

Consider a Markov chain $\mathbf{x}_{l+1} = \mathbf{x}_l + \frac{\eta}{2} \cdot \nabla_{\mathbf{x}_l} \log \hat{p}(\mathbf{x}_l) + \sqrt{\eta} \cdot \boldsymbol{\epsilon}_l$, where $\boldsymbol{\epsilon}_l \sim \mathcal{N}(0, \mathbf{I})$. Then, for any $\eta < \frac{2}{L}$

- The Markov chain has a unique stationary distribution π_η.
- $W_2(\pi_\eta, p) \leq C\eta$, and as $\eta \to 0$ we have $\pi_\eta \xrightarrow{d} p$.

Dalalyan A. Further and stronger analogy between sampling and optimization: Langevin Monte Carlo and gradient descent, 2017.

Langevin Dynamics

Theorem (Informal)

Let $\mathbf{x}_0$ be a random vector. Under mild regularity conditions, samples from the following dynamics will eventually follow $p_{\boldsymbol{\theta}}(\mathbf{x})$ (for sufficiently small η and large l):

$$\mathbf{x}_{l+1} = \mathbf{x}_l + \frac{\eta}{2} \cdot \nabla_{\mathbf{x}_l} \log p_{\boldsymbol{\theta}}(\mathbf{x}_l) + \sqrt{\eta} \cdot \boldsymbol{\epsilon}_l, \quad \boldsymbol{\epsilon}_l \sim \mathcal{N}(0, \mathbf{I}).$$

Song Y. Generative Modeling by Estimating Gradients of the Data Distribution, blog post, 2021

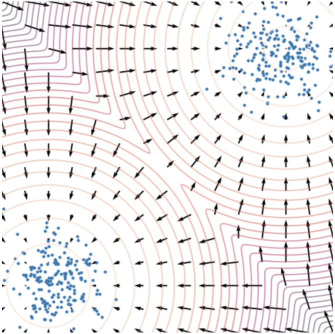
Langevin Dynamics

Theorem (Informal)

Let $\mathbf{x}_0$ be a random vector. Under mild regularity conditions, samples from the following dynamics will eventually follow $p_{\boldsymbol{\theta}}(\mathbf{x})$ (for sufficiently small η and large l):

$$\mathbf{x}_{l+1} = \mathbf{x}_l + \frac{\eta}{2} \cdot \nabla_{\mathbf{x}_l} \log p_{\boldsymbol{\theta}}(\mathbf{x}_l) + \sqrt{\eta} \cdot \boldsymbol{\epsilon}_l, \quad \boldsymbol{\epsilon}_l \sim \mathcal{N}(0, \mathbf{I}).$$

- What if $\boldsymbol{\epsilon}_l = \mathbf{0}$?
- The density $p_{\boldsymbol{\theta}}(\mathbf{x})$ is the **stationary** distribution of the Markov chain.
- The gradient is taken with respect to $\mathbf{x}$, not $\boldsymbol{\theta}$.
- $\nabla_{\mathbf{x}} \log p_{\boldsymbol{\theta}}(\mathbf{x})$ defines a vector field.



Song Y. Generative Modeling by Estimating Gradients of the Data Distribution, blog post, 2021

Outline

Score Matching

Score Function

$$\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}) = \nabla_{\mathbf{x}} \log p_{\boldsymbol{\theta}}(\mathbf{x})$$

Song Y. Generative Modeling by Estimating Gradients of the Data Distribution, blog post, 2021

Score Matching

Score Function

$$\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}) = \nabla_{\mathbf{x}} \log p_{\boldsymbol{\theta}}(\mathbf{x})$$

Langevin Dynamics

If we know the score function $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}) = \nabla_{\mathbf{x}} \log p_{\boldsymbol{\theta}}(\mathbf{x})$, we can generate samples from the model using Langevin dynamics:

$$\mathbf{x}_{l+1} = \mathbf{x}_l + \frac{\eta}{2} \cdot \nabla_{\mathbf{x}_l} \log p_{\boldsymbol{\theta}}(\mathbf{x}_l) + \sqrt{\eta} \cdot \boldsymbol{\epsilon}_l = \mathbf{x}_l + \frac{\eta}{2} \cdot \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_l) + \sqrt{\eta} \cdot \boldsymbol{\epsilon}_l.$$

Song Y. Generative Modeling by Estimating Gradients of the Data Distribution, blog post, 2021

Score Matching

Score Function

$$\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}) = \nabla_{\mathbf{x}} \log p_{\boldsymbol{\theta}}(\mathbf{x})$$

Langevin Dynamics

If we know the score function $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}) = \nabla_{\mathbf{x}} \log p_{\boldsymbol{\theta}}(\mathbf{x})$, we can generate samples from the model using Langevin dynamics:

$$\mathbf{x}_{l+1} = \mathbf{x}_l + \frac{\eta}{2} \cdot \nabla_{\mathbf{x}_l} \log p_{\boldsymbol{\theta}}(\mathbf{x}_l) + \sqrt{\eta} \cdot \boldsymbol{\epsilon}_l = \mathbf{x}_l + \frac{\eta}{2} \cdot \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_l) + \sqrt{\eta} \cdot \boldsymbol{\epsilon}_l.$$

Fisher Divergence

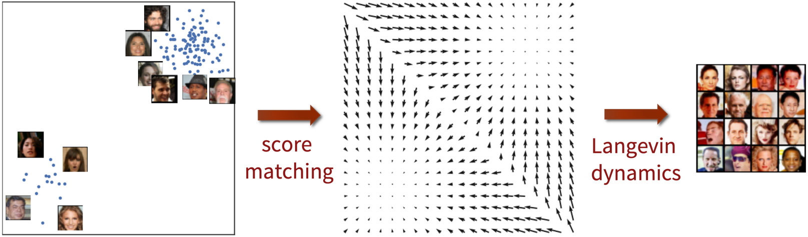
$$\begin{aligned} D_F(p_{\text{data}}, p_{\boldsymbol{\theta}}) = \frac{1}{2}\mathbb{E}_{\pi} \big\| \nabla_{\mathbf{x}} \log p_{\boldsymbol{\theta}}(\mathbf{x}) - \nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x}) \big\|_2^2 = \\ = \frac{1}{2}\mathbb{E}_{\pi} \big\| \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}) - \nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x}) \big\|_2^2 \rightarrow \min_{\boldsymbol{\theta}} \end{aligned}$$

Song Y. Generative Modeling by Estimating Gradients of the Data Distribution, blog post, 2021

Score Matching

Fisher Divergence

$$D_F(p_{\text{data}}, p_{\boldsymbol{\theta}}) = \frac{1}{2}\mathbb{E}_{\pi}\left\|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}) - \nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x})\right\|_2^2 \to \min_{\boldsymbol{\theta}}$$



Data samples

$\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\} \overset{\text{i.i.d.}}{\sim} p(\mathbf{x})$

Scores

$\mathbf{s}_\theta(\mathbf{x}) \approx \nabla_{\mathbf{x}} \log p(\mathbf{x})$

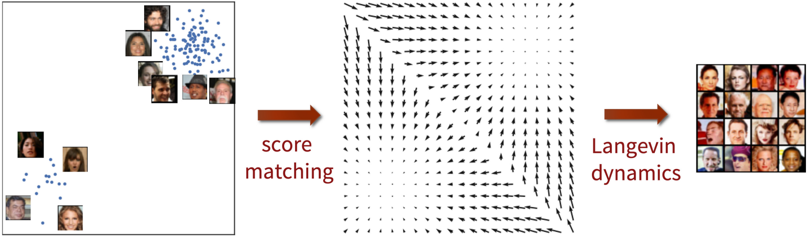
New samples

Song Y. Generative Modeling by Estimating Gradients of the Data Distribution, blog post, 2021

Score Matching

Fisher Divergence

$$D_F(p_{\text{data}}, p_{\boldsymbol{\theta}}) = \frac{1}{2}\mathbb{E}_\pi \left\| \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}) - \nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x}) \right\|_2^2 \to \min_{\boldsymbol{\theta}}$$



Data samples $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\} \overset{\text{i.i.d.}}{\sim} p(\mathbf{x})$

Scores $\mathbf{s}_\theta(\mathbf{x}) \approx \nabla_{\mathbf{x}} \log p(\mathbf{x})$

New samples

Problem: We don't know $\nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x})$.

Song Y. Generative Modeling by Estimating Gradients of the Data Distribution, blog post, 2021

Outline

1. Evaluation of Likelihood-Free Models
   - Frechet Inception Distance (FID)
   - Precision-Recall
   - CLIP Score
   - Human Evaluation
2. Langevin Dynamics
3. Score Matching
4. Denoising Score Matching

Denoising Score Matching

Let us perturb the original data $\mathbf{x} \sim p_{\text{data}}(\mathbf{x})$ with Gaussian noise:

$$\mathbf{x}_\sigma = \mathbf{x} + \sigma \cdot \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}), \quad q(\mathbf{x}_\sigma | \mathbf{x}) = \mathcal{N}(\mathbf{x}, \sigma^2 \cdot \mathbf{I})$$

Vincent P. A Connection Between Score Matching and Denoising Autoencoders, 2010

Denoising Score Matching

Let us perturb the original data $\mathbf{x} \sim p_{\text{data}}(\mathbf{x})$ with Gaussian noise:

$$\mathbf{x}_\sigma = \mathbf{x} + \sigma \cdot \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}), \quad q(\mathbf{x}_\sigma | \mathbf{x}) = \mathcal{N}(\mathbf{x}, \sigma^2 \cdot \mathbf{I})$$

$$q(\mathbf{x}_\sigma) = \int q(\mathbf{x}_\sigma | \mathbf{x}) p_{\text{data}}(\mathbf{x}) d\mathbf{x}.$$

Vincent P. A Connection Between Score Matching and Denoising Autoencoders, 2010

Denoising Score Matching

Let us perturb the original data $\mathbf{x} \sim p_{\text{data}}(\mathbf{x})$ with Gaussian noise:

$$\mathbf{x}_\sigma = \mathbf{x} + \sigma \cdot \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}), \quad q(\mathbf{x}_\sigma | \mathbf{x}) = \mathcal{N}(\mathbf{x}, \sigma^2 \cdot \mathbf{I})$$

$$q(\mathbf{x}_\sigma) = \int q(\mathbf{x}_\sigma | \mathbf{x}) p_{\text{data}}(\mathbf{x}) d\mathbf{x}.$$

Assumption

The solution to

$$\frac{1}{2} \mathbb{E}_{q(\mathbf{x}_\sigma)} \left\| \mathbf{s}_{\boldsymbol{\theta}, \sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma) \right\|_2^2 \to \min_{\boldsymbol{\theta}}$$

satisfies $\mathbf{s}_{\boldsymbol{\theta}, \sigma}(\mathbf{x}_\sigma) \approx \mathbf{s}_{\boldsymbol{\theta}, 0}(\mathbf{x}_0) = \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x})$ if σ is sufficiently small.

Vincent P. A Connection Between Score Matching and Denoising Autoencoders, 2010

Denoising Score Matching

Let us perturb the original data $\mathbf{x} \sim p_{\text{data}}(\mathbf{x})$ with Gaussian noise:

$$\mathbf{x}_\sigma = \mathbf{x} + \sigma \cdot \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}), \quad q(\mathbf{x}_\sigma | \mathbf{x}) = \mathcal{N}(\mathbf{x}, \sigma^2 \cdot \mathbf{I})$$

$$q(\mathbf{x}_\sigma) = \int q(\mathbf{x}_\sigma | \mathbf{x}) p_{\text{data}}(\mathbf{x}) d\mathbf{x}.$$

Assumption

The solution to

$$\frac{1}{2} \mathbb{E}_{q(\mathbf{x}_\sigma)} \left\| \mathbf{s}_{\boldsymbol{\theta}, \sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma) \right\|_2^2 \to \min_{\boldsymbol{\theta}}$$

satisfies $\mathbf{s}_{\boldsymbol{\theta}, \sigma}(\mathbf{x}_\sigma) \approx \mathbf{s}_{\boldsymbol{\theta}, 0}(\mathbf{x}_0) = \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x})$ if σ is sufficiently small.

- The score function of the noised data nearly matches the score function of the original data.
- The score function $\mathbf{s}_{\boldsymbol{\theta}, \sigma}(\mathbf{x}_\sigma)$ is parameterized by σ.
- **Note:** We don't know $q(\mathbf{x}_\sigma)$, just as we don't know $p_{\text{data}}(\mathbf{x})$.

Vincent P. A Connection Between Score Matching and Denoising Autoencoders, 2010

Denoising Score Matching

Theorem

Under mild regularity conditions, the following holds:

$$\mathbb{E}_{q(\mathbf{x}_\sigma)} \Big\| \mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma) \Big\|_2^2 =$$

$$= \mathbb{E}_{p_{\text{data}}(\mathbf{x})} \mathbb{E}_{q(\mathbf{x}_\sigma|\mathbf{x})} \Big\| \mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma|\mathbf{x}) \Big\|_2^2 + \text{const}(\boldsymbol{\theta})$$

Vincent P. A Connection Between Score Matching and Denoising Autoencoders, 2010

Denoising Score Matching

Theorem

Under mild regularity conditions, the following holds:

$$\mathbb{E}_{q(\mathbf{x}_\sigma)} \left\| \mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma) \right\|_2^2 =$$
$$= \mathbb{E}_{p_{\text{data}}(\mathbf{x})} \mathbb{E}_{q(\mathbf{x}_\sigma|\mathbf{x})} \left\| \mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma|\mathbf{x}) \right\|_2^2 + \text{const}(\boldsymbol{\theta})$$

Gradient of the Noise Kernel

$$\mathbf{x}_\sigma = \mathbf{x} + \sigma \cdot \boldsymbol{\epsilon}, \quad q(\mathbf{x}_\sigma|\mathbf{x}) = \mathcal{N}(\mathbf{x}, \sigma^2 \cdot \mathbf{I})$$

Vincent P. A Connection Between Score Matching and Denoising Autoencoders, 2010

Denoising Score Matching

Theorem

Under mild regularity conditions, the following holds:

$$\mathbb{E}_{q(\mathbf{x}_\sigma)} \left\| \mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma) \right\|_2^2 =$$
$$= \mathbb{E}_{p_{\text{data}}(\mathbf{x})} \mathbb{E}_{q(\mathbf{x}_\sigma|\mathbf{x})} \left\| \mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma|\mathbf{x}) \right\|_2^2 + \text{const}(\boldsymbol{\theta})$$

Gradient of the Noise Kernel

$$\mathbf{x}_\sigma = \mathbf{x} + \sigma \cdot \boldsymbol{\epsilon}, \quad q(\mathbf{x}_\sigma|\mathbf{x}) = \mathcal{N}(\mathbf{x}, \sigma^2 \cdot \mathbf{I})$$

$$\nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma|\mathbf{x}) = -\frac{\mathbf{x}_\sigma - \mathbf{x}}{\sigma^2} = -\frac{\boldsymbol{\epsilon}}{\sigma}$$

Vincent P. A Connection Between Score Matching and Denoising Autoencoders, 2010

Denoising Score Matching

Theorem

Under mild regularity conditions, the following holds:

$$\mathbb{E}_{q(\mathbf{x}_\sigma)} \left\| \mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma) \right\|_2^2 =$$

$$= \mathbb{E}_{p_{\text{data}}(\mathbf{x})} \mathbb{E}_{q(\mathbf{x}_\sigma|\mathbf{x})} \left\| \mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma|\mathbf{x}) \right\|_2^2 + \text{const}(\boldsymbol{\theta})$$

Gradient of the Noise Kernel

$$\mathbf{x}_\sigma = \mathbf{x} + \sigma \cdot \boldsymbol{\epsilon}, \quad q(\mathbf{x}_\sigma|\mathbf{x}) = \mathcal{N}(\mathbf{x}, \sigma^2 \cdot \mathbf{I})$$

$$\nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma|\mathbf{x}) = -\frac{\mathbf{x}_\sigma - \mathbf{x}}{\sigma^2} = -\frac{\boldsymbol{\epsilon}}{\sigma}$$

- The right-hand side doesn't require computing $\nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma)$ or even $\nabla_{\mathbf{x}_\sigma} \log p_{\text{data}}(\mathbf{x}_\sigma)$.
- $\mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_\sigma)$ is trained to **denoise** the noised samples $\mathbf{x}_\sigma$.

Vincent P. A Connection Between Score Matching and Denoising Autoencoders, 2010

Denoising Score Matching

Initial objective:

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\left\|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}) - \nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x})\right\|_2^2 \to \min_{\boldsymbol{\theta}}$$

Song Y. Generative Modeling by Estimating Gradients of the Data Distribution, blog post, 2021

Denoising Score Matching

Initial objective:

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\left\|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}) - \nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x})\right\|_2^2 \to \min_{\boldsymbol{\theta}}$$

Noised objective:

$$\mathbb{E}_{q(\mathbf{x}_\sigma)}\left\|\mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}} \log q(\mathbf{x}_\sigma)\right\|_2^2 \to \min_{\boldsymbol{\theta}}$$

Song Y. Generative Modeling by Estimating Gradients of the Data Distribution, blog post, 2021

Denoising Score Matching

Initial objective:

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\left\|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}) - \nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x})\right\|_2^2 \to \min_{\boldsymbol{\theta}}$$

Noised objective:

$$\mathbb{E}_{q(\mathbf{x}_\sigma)}\left\|\mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}} \log q(\mathbf{x}_\sigma)\right\|_2^2 \to \min_{\boldsymbol{\theta}}$$

This is equivalent to a denoising task:

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\mathbb{E}_{q(\mathbf{x}_\sigma|\mathbf{x})}\left\|\mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma|\mathbf{x})\right\|_2^2 \to \min_{\boldsymbol{\theta}}$$

Song Y. Generative Modeling by Estimating Gradients of the Data Distribution, blog post, 2021

Denoising Score Matching

Initial objective:

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x})} \left\| \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}) - \nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x}) \right\|_2^2 \to \min_{\boldsymbol{\theta}}$$

Noised objective:

$$\mathbb{E}_{q(\mathbf{x}_\sigma)} \left\| \mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}} \log q(\mathbf{x}_\sigma) \right\|_2^2 \to \min_{\boldsymbol{\theta}}$$

This is equivalent to a denoising task:

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x})} \mathbb{E}_{q(\mathbf{x}_\sigma|\mathbf{x})} \left\| \mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma|\mathbf{x}) \right\|_2^2 \to \min_{\boldsymbol{\theta}}$$

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x})} \mathbb{E}_{\mathcal{N}(0,\mathbf{I})} \left\| \mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x} + \sigma \cdot \boldsymbol{\epsilon}) + \frac{\boldsymbol{\epsilon}}{\sigma} \right\|_2^2 \to \min_{\boldsymbol{\theta}}$$

Langevin Dynamics

$$\mathbf{x}_{l+1} = \mathbf{x}_l + \frac{\eta}{2} \cdot \mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_l) + \sqrt{\eta} \cdot \boldsymbol{\epsilon}_l, \quad \boldsymbol{\epsilon}_l \sim \mathcal{N}(0, \mathbf{I}).$$

Song Y. Generative Modeling by Estimating Gradients of the Data Distribution, blog post, 2021

Summary

- Frechet Inception Distance is the most popular metric for evaluating implicit generative models.
- Precision-recall allows for choosing a model that balances sample quality and diversity.
- The CLIP score is widely used to measure text-to-image alignment.
- The gold standard for evaluating generated image quality is human assessment.
- Langevin dynamics enable sampling from generative models using gradients of the log-likelihood.
- Score matching proposes minimizing Fisher divergence to estimate the score function.
- Denoising score matching optimizes Fisher divergence on noisy data, making it estimable with samples.