Evaluating and tuning precision and recall for Generative Models

Data Science Lab

Alexandre Vérine

PhD Student,
LAMSADE, Université Dauphine - PSL

October 24, 2023

Context and Motivation

Context

- Evaluating generative models is a significant challenge.
- Among these models, which one emerges as the top choice ?

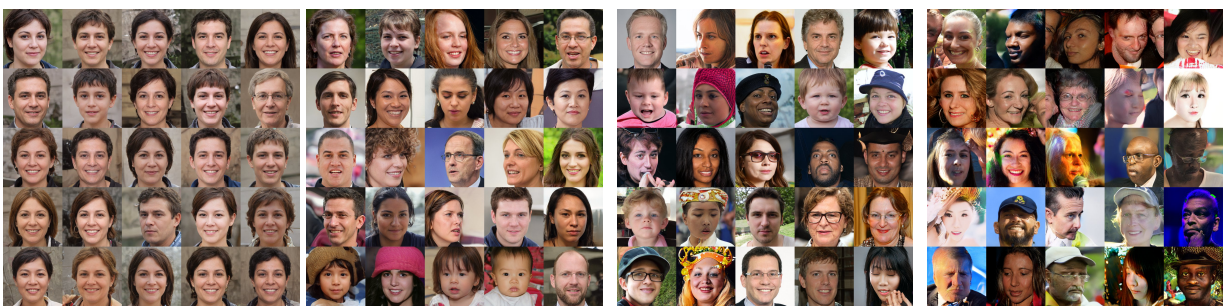
(a) Midjourney (b) DALL-E 2

Figure. Comparison of two generative models: DALL-E and Midjourney given the same prompt: *A dog playing with a child.*

Context and Motivation

Motivation

Traditional metrics such as the Fréchet Inception Distance (FID) encapsulate both quality and diversity in an unclear way:



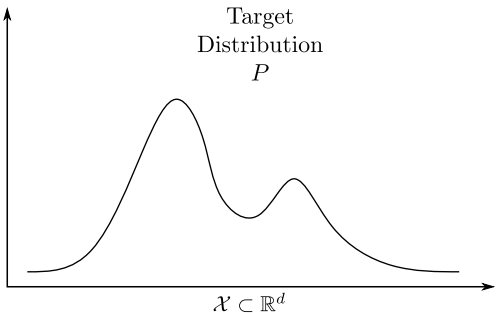
(a) Set A - FID= 91.7 (b) Set B - FID= 16.9 (c) Set C - FID= 4.5 (d) Set D - FID= 16.7

Figure. Source: Kynkäänniemi et al. 2019

Objective: How can we assess quality and diversity independently ?

Context and Motivation

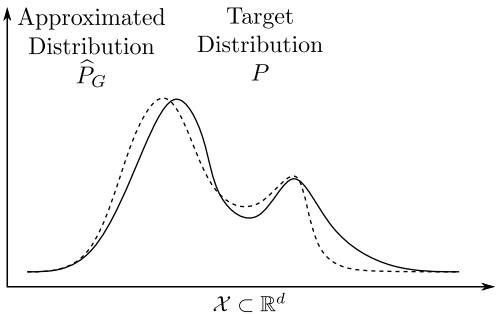
Set Up



- Assumption: There is an unknown target distribution P in $\mathcal{X} \subset \mathbb{R}^d$ that we can sample from.

Context and Motivation

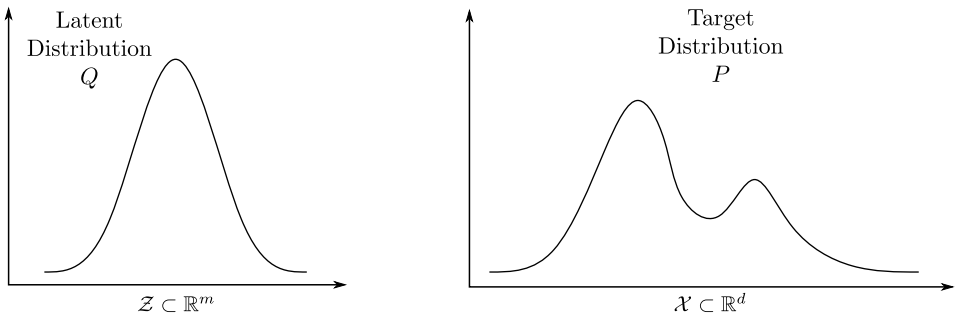
Set Up



- Assumption: There is an unknown target distribution P in $\mathcal{X} \subset \mathbb{R}^d$ that we can sample from.
- Goal: Learn a parameterized distribution $\widehat{P}$ that approximate P:

Context and Motivation

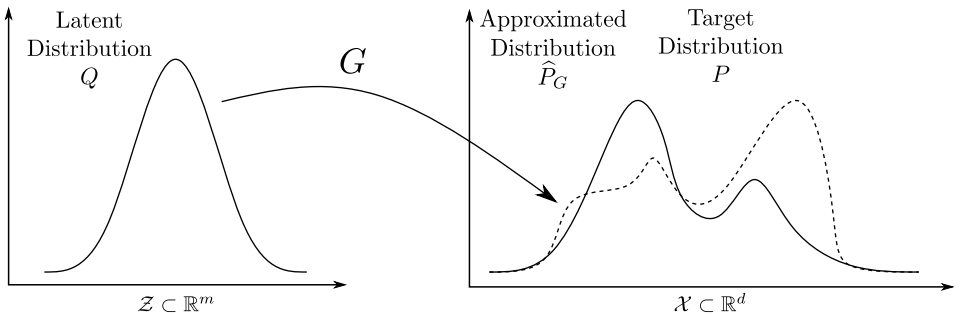
Set Up



- Assumption: There is an unknown target distribution P in $\mathcal{X} \subset \mathbb{R}^d$ that we can sample from.
- Goal: Learn a parameterized distribution $\widehat{P}$ that approximate P:
  1. Consider a distribution Q in a latent space $\mathcal{X} \subset \mathbb{R}^m$, usually $\mathcal{N}(0, I_m)$.

Context and Motivation

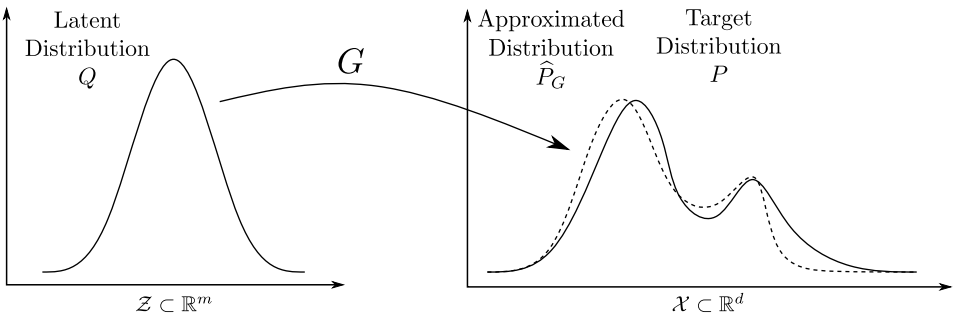
Set Up



- Assumption: There is an unknown target distribution P in $\mathcal{X} \subset \mathbb{R}^d$ that we can sample from.
- Goal: Learn a parameterized distribution $\widehat{P}$ that approximate P:
  1. Consider a distribution Q in a latent space $\mathcal{X} \subset \mathbb{R}^m$, usually $\mathcal{N}(0, I_m)$.
  2. Take a generator model G represented by a neural network. Take $\widehat{P}_G = G\#Q$.

Context and Motivation

Set Up

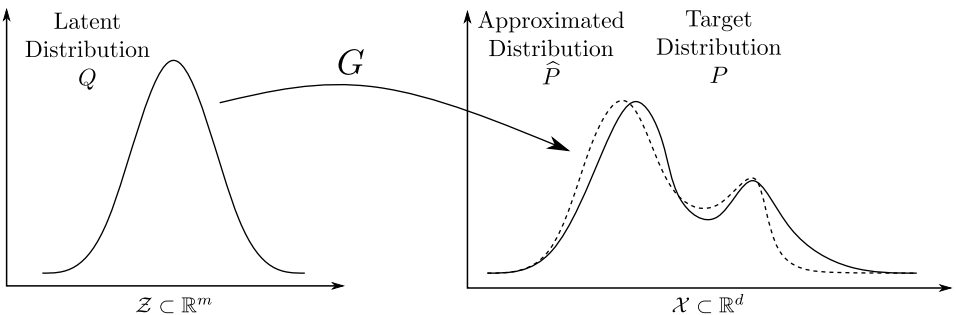


- Assumption: There is an unknown target distribution P in $\mathcal{X} \subset \mathbb{R}^d$ that we can sample from.
- Goal: Learn a parameterized distribution $\widehat{P}$ that approximate P:
  1. Consider a distribution Q in a latent space $\mathcal{X} \subset \mathbb{R}^m$, usually $\mathcal{N}(0, I_m)$.
  2. Take a generator model G represented by a neural network. Take $\widehat{P}_G = G\#Q$.
  3. Compute G^* that minimize a dissimilarity measure D between P and $\widehat{P}_G$:

$$G^* = \underset{G}{\operatorname{argmin}}\, D(P, \widehat{P}_G)$$

Context and Motivation

Set Up



- Assumption: There is an unknown target distribution P in $\mathcal{X} \subset \mathbb{R}^d$ that we can sample from.
- Goal: Learn a parameterized distribution $\widehat{P}$ that approximate P:
  1. Consider a distribution Q in a latent space $\mathcal{X} \subset \mathbb{R}^m$, usually $\mathcal{N}(0, I_m)$.
  2. Take a generator model G represented by a neural network. Take $\widehat{P} = G\#Q$.
  3. Define G that minimize a dissimilarity measure D between P and $\widehat{P}$:

$$G = \operatorname{argmin} D(P, \widehat{P})$$

Precision and Recall for Generative Models

Intuitive Idea

To assess models, we use the notion of Precision and Recall, inspired from the classification literature:

	Precision	Recall
Classification	How much of the positively classified samples are positive ?	How much of the positive samples are positively classified ?
Generation	How much of the generated samples are coherent ?	How much coherent samples can be generated ?

- Sajjadi et al. 2018: The key intuition is that precision should measure how much of $\widehat{P}$ can be generated by a "part" of P while recall should measure how much of P can be generated by a "part" of $\widehat{P}$.

Precision and Recall for Generative Models

Two notable definitions of precision and recall have emerged in recent years:

- **Support-based** (Kynkäänniemi et al. (2019)):
  - based on shared distribution support,
  - concise and interpretable,
  - widely used by the community,
  - intractable and computationally expensive.
- **PR Curves** (Sajjadi et al. (2018), Simon, Webster, and Rabin (2019)):
  - more complex definition, thus less intuitive,
  - mathematically grounded.

Precision and Recall for Generative Models

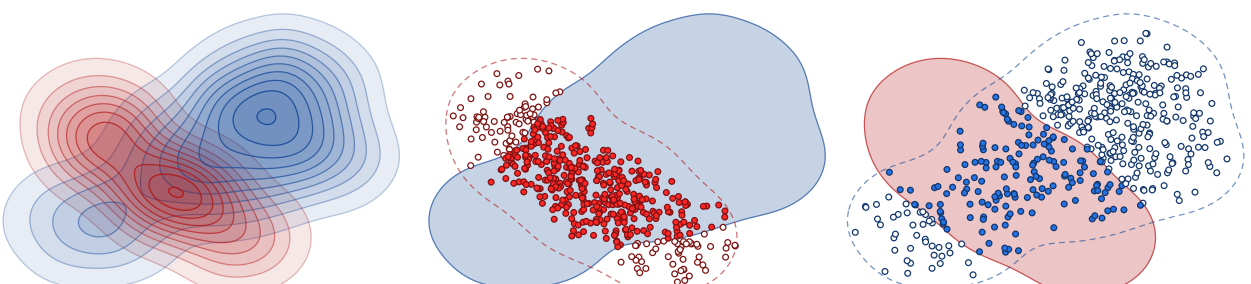
(α, β): the support-based method

- **Precision** is the proportion of generated data that lies on the support of the real data.

$$\alpha = \widehat{P}(\text{Supp}(P))$$

- **Recall** is the proportion of the support of the real data that is covered by the generated data.

$$\beta = P(\text{Supp}(\widehat{P}))$$



(a) Distributions P (blue) and $\widehat{P}$ (red).

(b) Discretisation of $\widehat{P}$. Precision is the rate of the filled red dots with the total number of red dots.

(c) Discretisation of P. Recall is the rate of the filled blue dots with the total number of blue dots

Figure. Example of Precision and Recall. Source Kynkäänniemi et al. (2019).

Precision and Recall for Generative Models

Drawbacks of the Support based method

The drawbacks are that the metric:

- does not reflect the difference in density,
- relies k-nn based method to estimate the support,
- requires finite distribution support.

Precision and Recall for Generative Models

Set of Precision-Recall

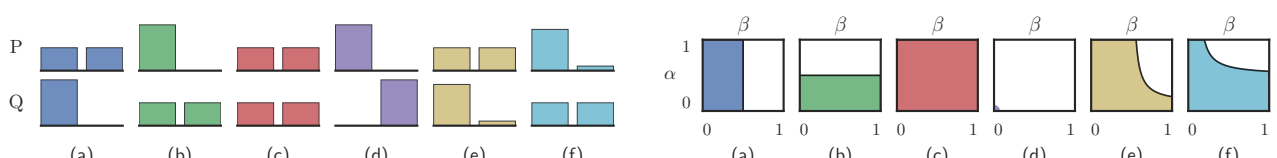
More complex definition for discrete distibutions:

Definition 2.1 (Precision and Recall - Sajjadi et al. 2018)

For $\alpha, \beta \in [0, 1]$, the probability distribution $\widehat{P}$ has a precision α at recall β w.r.t. P if there exist distributions μ, ν_P and $\nu_{\widehat{P}}$ such that

$$P = \beta\mu + (1 - \beta)\nu_P \quad \textit{and} \quad \widehat{P} = \alpha\mu + (1 - \alpha)\nu_{\widehat{P}}$$

The component ν_P denotes the part of P that is "missed" by $\widehat{P}$. Similarly, $\nu_{\widehat{P}}$ denotes the noise part of $\widehat{P}$.



(a) Examples of distributions P and $\widehat{P}$

(b) Corresponding PR sets.

Figure. Examples of PR sets for different distributions P and $\widehat{P}$. Source: Sajjadi et al. 2018.

Precision and Recall for Generative Models

PR-Curves

The frontier of the set of $\mathrm{PR}(P, \widehat{P})$, is the PR-Curve denoted PRD, parameterized by $\lambda \in [0, \infty]$ and can be computed with the functions:

$$\alpha(\lambda) = \sum_{\boldsymbol{x}_i \in \mathcal{X}} \min\left(\lambda p(\boldsymbol{x}_i), \widehat{p}(\boldsymbol{x}_i)\right)$$

and

$$\beta(\lambda) = \sum_{\boldsymbol{x}_i \in \mathcal{X}} \min\left(p(\boldsymbol{x}_i), \widehat{p}(\boldsymbol{x}_i)/\lambda\right)$$

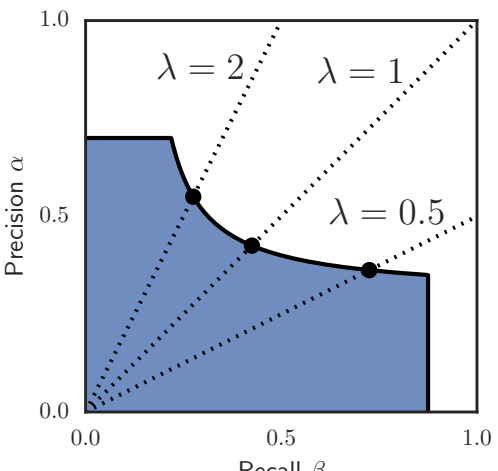


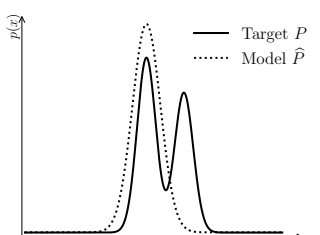
Figure. Example of the PRD set. Source: Sajjadi et al. 2018

Precision and Recall for Generative Models

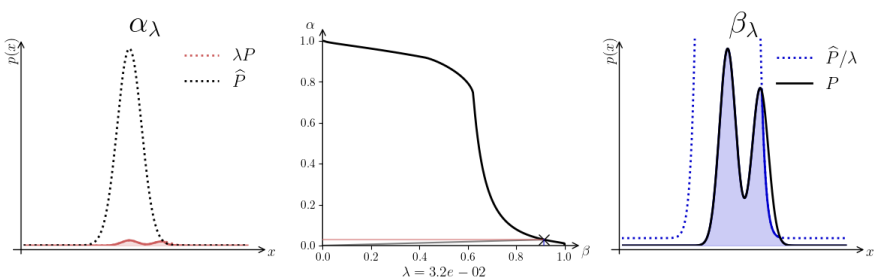
PR-Curves for continuous distributions

Definition extended for continuous distributions:

$$\alpha(\lambda) = \int_{\mathcal{X}} \min\left(\lambda p(\boldsymbol{x}), \widehat{p}(\boldsymbol{x})\right) \mathrm{d}\boldsymbol{x} \quad \text{and} \quad \beta(\lambda) = \int_{\mathcal{X}} \min\left(p(\boldsymbol{x}), \widehat{p}(\boldsymbol{x})/\lambda\right) \mathrm{d}\boldsymbol{x}.$$



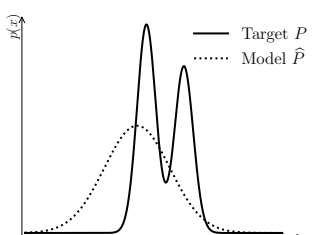
(a) Example with good precision



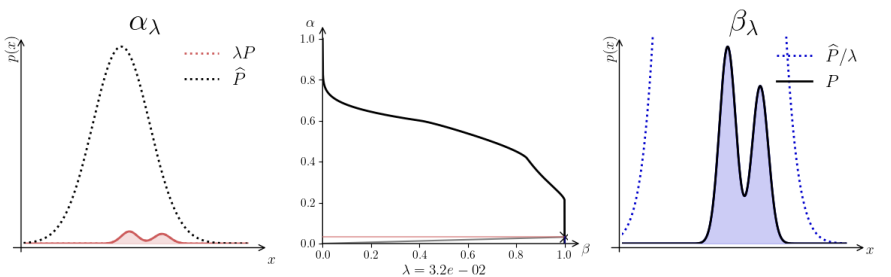
(b) PR-Curves

Definition extended for continuous distributions:

$$\alpha(\lambda) = \int_{\mathcal{X}} \min\left(\lambda p(\boldsymbol{x}), \widehat{p}(\boldsymbol{x})\right) \mathrm{d}\boldsymbol{x} \quad \text{and} \quad \beta(\lambda) = \int_{\mathcal{X}} \min\left(p(\boldsymbol{x}), \widehat{p}(\boldsymbol{x})/\lambda\right) \mathrm{d}\boldsymbol{x}.$$



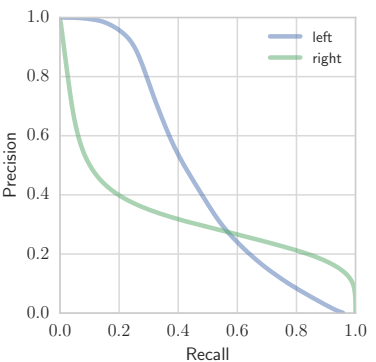
(c) Example with good recall



(d) PR-Curves

In practice:

(e) High Precision, Low Recall



(f) PR-Curves

(g) High Recall, Low Precision

Figure. Examples for two different GANS. Source: Sajjadi et al. 2018.

Precision and Recall for Generative Models

Drawbacks of the PR-Curves

The drawbacks are that the metric:

- rely on density estimation or density ratio estimation,
- is variable in high dimension,
- does not synthesize well.

Precision and Recall for Generative Models

Comparison between the two methods

(a) Set A (b) Set B

(c) Set C (d) Set D

Figure. Samples. Source: Kynkäänniemi et al. 2019

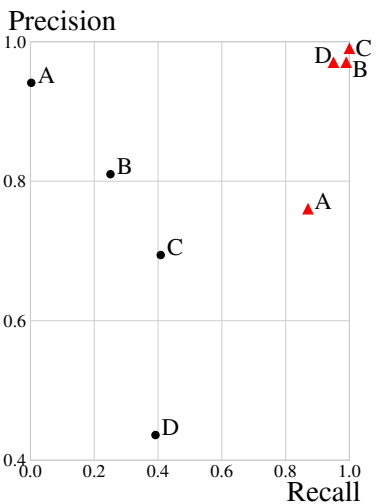
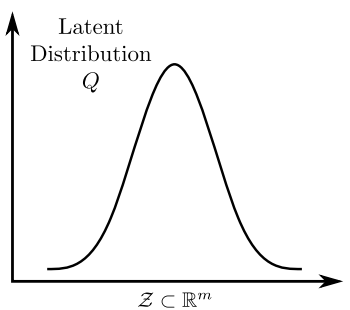


Figure. Support based (black) and PR-Curves (Red)

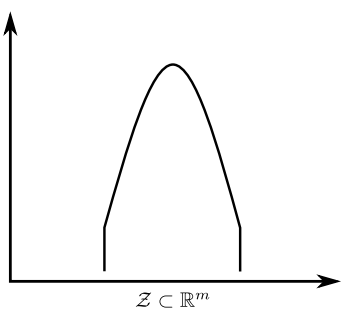
Tricks to Tune Precision and Recall

Hard Truncation

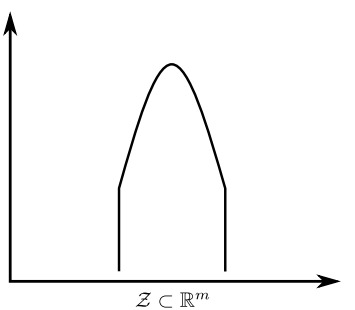
Truncating the latent distribution:



$\psi = +\infty$



$\psi = 1$



$\psi = 0.5$

Tricks to Tune Precision and Recall

Hard Truncation

Figure. From left to right: $\psi = 0.0$, $\psi = 0.3$ $\psi = 0.7$ $\psi = 1.0$.

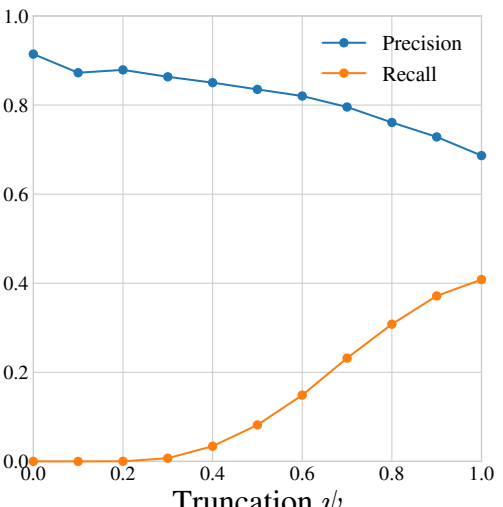
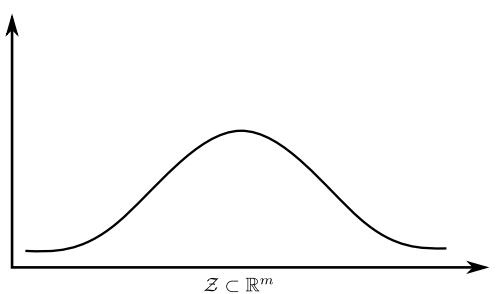


Figure. Source: Kynkäänniemi et al. 2019

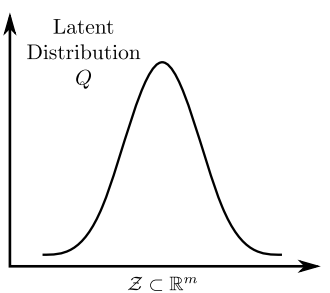
Tricks to Tune Precision and Recall

Soft Truncation

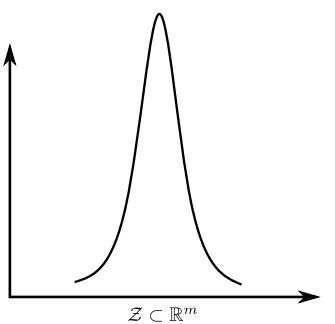
Re-scaling the latent distribution:



$\psi = 2$



$\psi = 1$



$\psi = 0.5$

Tricks to Tune Precision and Recall

Soft Truncation

(a) $\psi = 0.04$ (b) $\psi = 0.5$ (c) $\psi = 1.0$ (d) $\psi = 2.0$

Figure. Soft-Truncation on BigGAN. Source:Brock, Donahue, and Simonyan 2019.

Tricks to Tune Precision and Recall

Rejection Sampling

In Rejection Sampling methods, we use the discriminator, trained as classifier to accept or reject samples:

(a) MNIST

(b) CelebA

- Azadi et al. 2019
- Issenhuth et al. 2022
- Turner et al. 2019

Boosting Generative models:

Figure. Left: Samples from the dataset given high weights by the discriminator. Right: Samples from the dataset given low weights by the discriminator. The next model will focus on the sample on the right. Source: Tolstikhin et al. 2017

- Tolstikhin et al. 2017
- Grover and Ermon 2017

Tricks to Tune Precision and Recall

Gradient Descent

Using the discriminator as a classifier and perform a gradient descent:

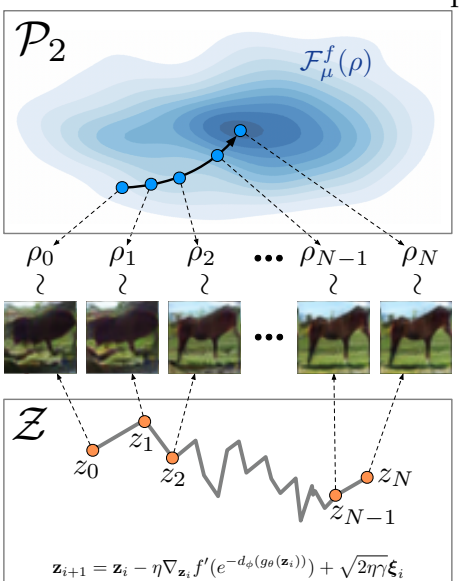


Figure. Source: Ansari, Ang, and Soh 2021

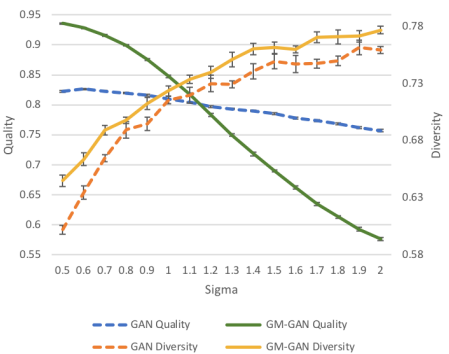
- Ansari, Ang, and Soh 2021
- Tanaka 2019
- Che et al. 2021

Tricks to Tune Precision and Recall

Gaussian Mixtures

Training a Gaussian Mixture $\mathcal{N}(\mu_k, \sigma I)$ in the latent space:

(a) $\sigma = 0.1$ (b) $\sigma = 1$ (c) $\sigma = 2$



(d) Precision and Recall

Figure. Source: Ben-Yosef and Weinshall 2018

- Ben-Yosef and Weinshall 2018
- Pandeva and Schubert 2019

A GAN consists of two components:

- a Generator, $G : \mathcal{Z} \to \mathcal{X}$
- a Discriminator, $T : \mathcal{X} \to \text{dom}(f^*)$

Nowozin, Cseke, and Tomioka (2016) propose a general framework where the GAN minimizes any f-divergence $\mathcal{D}_f$ estimated by solving the dual estimation via the following min-max optimization problem:

$$\min_G \max_T \mathcal{D}^{\text{dual}}_{f,T}(P \| \widehat{P}_G) = \min_G \max_T \mathbb{E}_{\boldsymbol{x} \sim P}\left[T(\boldsymbol{x})\right] - \mathbb{E}_{\boldsymbol{x} \sim \widehat{P}_G}\left[f^*(T(\boldsymbol{x}))\right],$$

- Nowozin, Cseke, and Tomioka 2016
- Arjovsky, Chintala, and Bottou 2017
- Verine et al. 2023

Tricks to Tune Precision and Recall

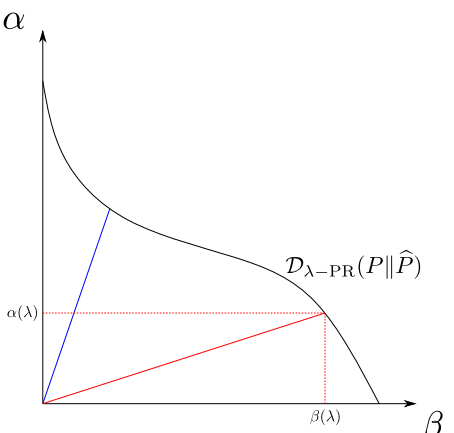
Relation between f-divergence and PR Curve

Theorem 3.1 (P&R as a function of $\mathcal{D}_{\lambda\text{-PR}}$)

For any $P, \widehat{P} \in \mathcal{P}(\mathcal{X})$ and $\lambda \in \mathbb{R}^+ \cup \{+\infty\}$, the PR curve ∂PRD is related to the PR-divergence $\mathcal{D}_{\lambda\text{-PR}}(P\|\widehat{P})$ as:

$$\alpha_\lambda(P\|\,\widehat{P}) = \min(1, \lambda) - \mathcal{D}_{\lambda\text{-PR}}(P\|\widehat{P}).$$

A direct consequence of Theorem 3.1 is that minimizing $\mathcal{D}_{\lambda\text{-PR}}$ is equivalent to maximizing α_λ.

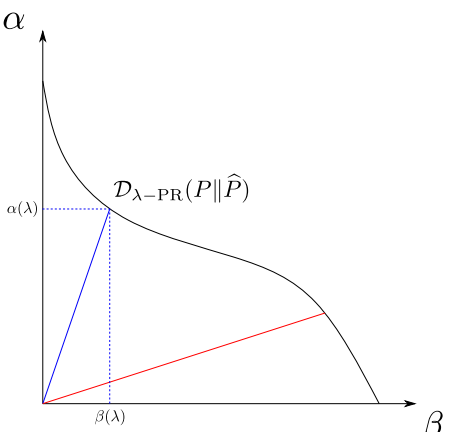


Theorem 3.1 (P&R as a function of $\mathcal{D}_{\lambda\text{-PR}}$)

For any $P, \widehat{P} \in \mathcal{P}(\mathcal{X})$ and $\lambda \in \mathbb{R}^+ \cup \{+\infty\}$, the PR curve ∂PRD is related to the PR-divergence $\mathcal{D}_{\lambda\text{-PR}}(P\|\widehat{P})$ as:

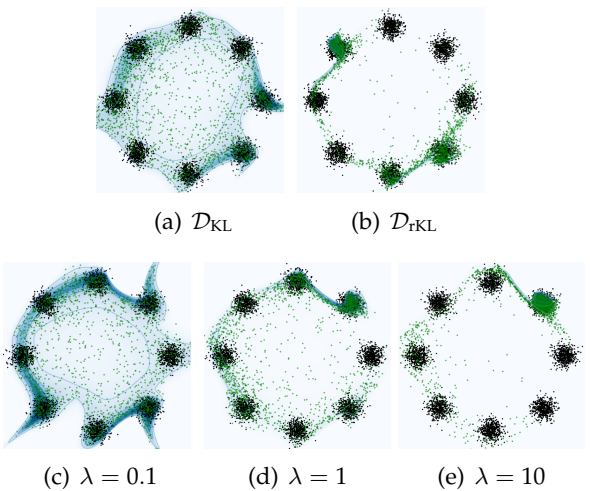
$$\alpha_\lambda(P\|\,\widehat{P}) = \min(1, \lambda) - \mathcal{D}_{\lambda\text{-PR}}(P\|\widehat{P}).$$

A direct consequence of Theorem 3.1 is that minimizing $\mathcal{D}_{\lambda\text{-PR}}$ is equivalent to maximizing α_λ.



Tricks to Tune Precision and Recall

f-GANs



(a) $\mathcal{D}_{\mathrm{KL}}$ (b) $\mathcal{D}_{\mathrm{rKL}}$

(c) $\lambda = 0.1$ (d) $\lambda = 1$ (e) $\lambda = 10$

Figure. Models trained on 2D Gaussians. Samples drawn from the true distribution P are shown in black, samples drawn from the estimated distribution $\widehat{P}$ are shown in green and the log-likelihood of $\widehat{P}$ is shown in blue (darker means higher density).

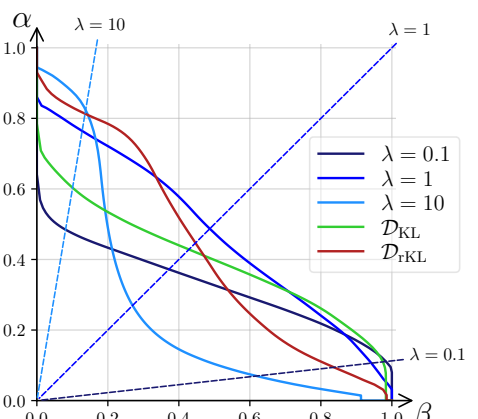
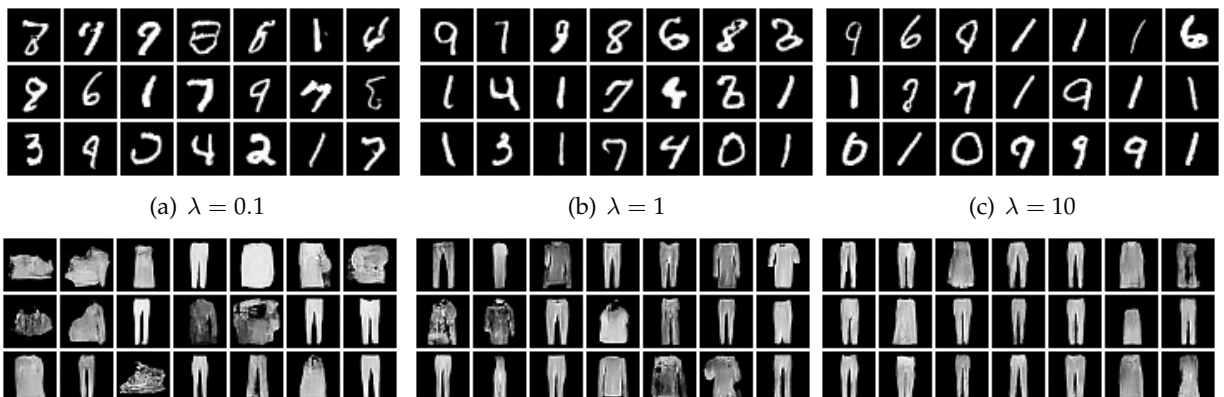


Figure. Corresponding PR Curves.

Tricks to Tune Precision and Recall

f-GAN



(a) $\lambda = 0.1$ (b) $\lambda = 1$ (c) $\lambda = 10$

(d) $\lambda = 0.1$ (e) $\lambda = 1$ (f) $\lambda = 10$

Figure. Samples from a models trained on MNIST and FashionMNIST. Recall decreases as the precision increases for λ between 0.1 and 10.

Thanks !

References I

- Ansari, Abdul Fatir, Ming Liang Ang, and Harold Soh (June 2021). ***Refining Deep Generative Models via Discriminator Gradient Flow.*** arXiv:2012.00780 [cs, stat]. URL: http://arxiv.org/abs/2012.00780 (visited on 07/14/2023).
- Arjovsky, Martin, Soumith Chintala, and Léon Bottou (Dec. 2017). **"Wasserstein GAN".** In: *Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia,* arXiv: 1701.07875. URL: http://arxiv.org/abs/1701.07875 (visited on 10/14/2021).
- Azadi, Samaneh et al. (Feb. 2019). ***Discriminator Rejection Sampling.*** arXiv:1810.06758 [cs, stat]. URL: http://arxiv.org/abs/1810.06758 (visited on 09/13/2022).
- Ben-Yosef, Matan and Daphna Weinshall (Aug. 2018). **"Gaussian Mixture Generative Adversarial Networks for Diverse Datasets, and the Unsupervised Clustering of Images".** In: *arXiv:1808.10356 [cs, stat]*. arXiv: 1808.10356. URL: http://arxiv.org/abs/1808.10356 (visited on 04/06/2022).
- Brock, Andrew, Jeff Donahue, and Karen Simonyan (Feb. 2019). ***Large Scale GAN Training for High Fidelity Natural Image Synthesis.*** arXiv:1809.11096 [cs, stat]. URL: http://arxiv.org/abs/1809.11096 (visited on 05/10/2023).
- Che, Tong et al. (July 2021). ***Your GAN is Secretly an Energy-based Model and You Should use Discriminator Driven Latent Sampling.*** arXiv:2003.06060 [cs, stat]. URL: http://arxiv.org/abs/2003.06060 (visited on 06/01/2023).
- Grover, Aditya and Stefano Ermon (Dec. 2017). ***Boosted Generative Models.*** arXiv:1702.08484 [cs, stat]. URL: http://arxiv.org/abs/1702.08484 (visited on 07/19/2022).

References II

- Issenhuth, Thibaut et al. (Jan. 2022). **"Latent reweighting, an almost free improvement for GANs".** en. In: *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. Waikoloa, HI, USA: IEEE, pp. 3574–3583. ISBN: 978-1-66540-915-5. DOI: 10.1109/WACV51458.2022.00363. URL: https://ieeexplore.ieee.org/document/9706934/ (visited on 09/13/2022).
- Kynkäänniemi, Tuomas et al. (Oct. 2019). **"Improved Precision and Recall Metric for Assessing Generative Models".** In: *33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada*. arXiv: 1904.06991. (Visited on 05/26/2021).
- Nowozin, Sebastian, Botond Cseke, and Ryota Tomioka (June 2016). ***f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization*.** arXiv:1606.00709 [cs, stat]. URL: http://arxiv.org/abs/1606.00709 (visited on 09/13/2022).
- Pandeva, Teodora and Matthias Schubert (Nov. 2019). **"MMGAN: Generative Adversarial Networks for Multi-Modal Distributions".** In: *arXiv:1911.06663 [cs, stat]*. arXiv: 1911.06663. URL: http://arxiv.org/abs/1911.06663 (visited on 04/06/2022).
- Sajjadi, Mehdi S. M. et al. (Oct. 2018). **"Assessing Generative Models via Precision and Recall".** In: *32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada*. arXiv: 1806.00035. URL: http://arxiv.org/abs/1806.00035 (visited on 05/24/2021).

References III

- Simon, Loic, Ryan Webster, and Julien Rabin (May 2019). **"Revisiting precision recall definition for generative modeling".** en. In: *Proceedings of the 36th International Conference on Machine Learning*. ISSN: 2640-3498. PMLR, pp. 5799–5808. URL: https://proceedings.mlr.press/v97/simon19a.html (visited on 12/19/2022).
- Tanaka, Akinori (2019). **"Discriminator optimal transport".** In: *Advances in Neural Information Processing Systems*. Vol. 32. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper_files/paper/2019/hash/8abfe8ac9ec214d68541fcb888c0b4c3-Abstract.html (visited on 07/14/2023).
- Tolstikhin, Ilya et al. (May 2017). ***AdaGAN: Boosting Generative Models.*** arXiv:1701.02386 [cs, stat]. URL: http://arxiv.org/abs/1701.02386 (visited on 10/22/2023).
- Turner, Ryan et al. (May 2019). **"Metropolis-Hastings Generative Adversarial Networks".** en. In: *Proceedings of the 36th International Conference on Machine Learning*. ISSN: 2640-3498. PMLR, pp. 6345–6353. URL: https://proceedings.mlr.press/v97/turner19a.html (visited on 09/13/2022).
- Verine, Alexandre et al. (May 2023). ***Precision-Recall Divergence Optimization for Generative Modeling with GANs and Normalizing Flows.*** arXiv:2305.18910 [cs]. URL: http://arxiv.org/abs/2305.18910 (visited on 07/12/2023).