Deep Generative Models
Lecture 10

Roman Isachenko

Moscow Institute of Physics and Technology

2020

Evaluation of likelihood-free models

How to evaluate generative models?

Likelihood-based models

- Split data to train/val/test.
- Fit model on the train part.
- Tune hyperparameters on the validation part.
- Evaluate generalization by reporting likelihoods on the test set.

Not all models have tractable likelihoods

- VAE: compare ELBO values.
- GAN: ???

Evaluation of likelihood-free models

Let take some pretrained image classification model to get the conditional label distribution $p(y|\mathbf{x})$ (e.g. ImageNet classifier).

What do we want from samples?

- **Sharpness**

Low sharpness

High sharpness

The conditional distribution $p(y|\mathbf{x})$ should have low entropy (each image $\mathbf{x}$ should have distinctly recognizable object).

- **Diversity**

Low diversity

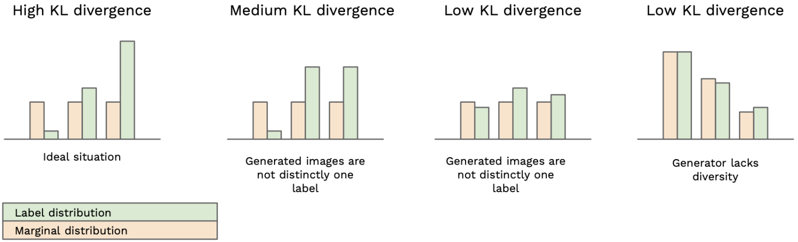
High diversity

The marginal distribution $p(y) = \int p(y|\mathbf{x})p(\mathbf{x})d\mathbf{x}$ should have high entropy (there should be as many classes generated as possible).

Evaluation of likelihood-free models

What do we want from samples?

- **Sharpness.** The conditional distribution $p(y|\mathbf{x})$ should have low entropy (each image $\mathbf{x}$ should have distinctly recognizable object).
- **Diversity.** The marginal distribution $p(y) = \int p(y|\mathbf{x})p(\mathbf{x})d\mathbf{x}$ should have high entropy (there should be as many classes generated as possible).



https://medium.com/octavian-ai/a-simple-explanation-of-the-inception-score-372dff6a8c7a

Evaluation of likelihood-free models

What do we want from samples?

- Sharpness $\Rightarrow$ low $H(y|\mathbf{x}) = -\sum_y \int_{\mathbf{x}} p(y, \mathbf{x}) \log p(y|\mathbf{x}) d\mathbf{x}$.
- Diversity $\Rightarrow$ high $H(y) = -\sum_y p(y) \log p(y)$.

Inception Score

$$
\begin{aligned}
IS &= \exp(H(y) - H(y|\mathbf{x})) \\
&= \exp\left(-\sum_y p(y) \log p(y) + \sum_y \int_{\mathbf{x}} p(y, \mathbf{x}) \log p(y|\mathbf{x}) d\mathbf{x}\right) \\
&= \exp\left(\sum_y \int_{\mathbf{x}} p(y, \mathbf{x}) \log \frac{p(y|\mathbf{x})}{p(y)} d\mathbf{x}\right) \\
&= \exp\left(\mathbb{E}_{\mathbf{x}} \sum_y p(y|\mathbf{x}) \log \frac{p(y|\mathbf{x})}{p(y)}\right) = \exp\left(\mathbb{E}_{\mathbf{x}} KL(p(y|\mathbf{x}) || p(y))\right)
\end{aligned}
$$

---

https://arxiv.org/abs/1606.03498

Evaluation of likelihood-free models

Inception Score

$$IS = \exp\left(\mathbb{E}_{\mathbf{x}} KL(p(y|\mathbf{x})||p(y))\right)$$

IS limitations

- Inception score depends on the quality of the pretrained classifier $p(y|\mathbf{x})$.
- If generator produces images with a different set of labels from the classifier training set, IS will be low.
- If the generator produces one image per class, the IS will be perfect (there is no measure of intra-class diversity).
- IS only require samples from the generator and do not take into account the desired data distribution $\pi(\mathbf{x})$ directly (only implicitly via a classifier).

---

https://arxiv.org/abs/1801.01973

Evaluation of likelihood-free models

Theorem

If $\pi(\mathbf{x})$ and $p(\mathbf{x}|\boldsymbol{\theta})$ has moment generation functions then

$$\pi(\mathbf{x}) = p(\mathbf{x}|\boldsymbol{\theta}) \Leftrightarrow \mathbb{E}_\pi \mathbf{x}^k = \mathbb{E}_p \mathbf{x}^k, \quad \forall k \geq 1.$$

This is intractable to calculate all moments.

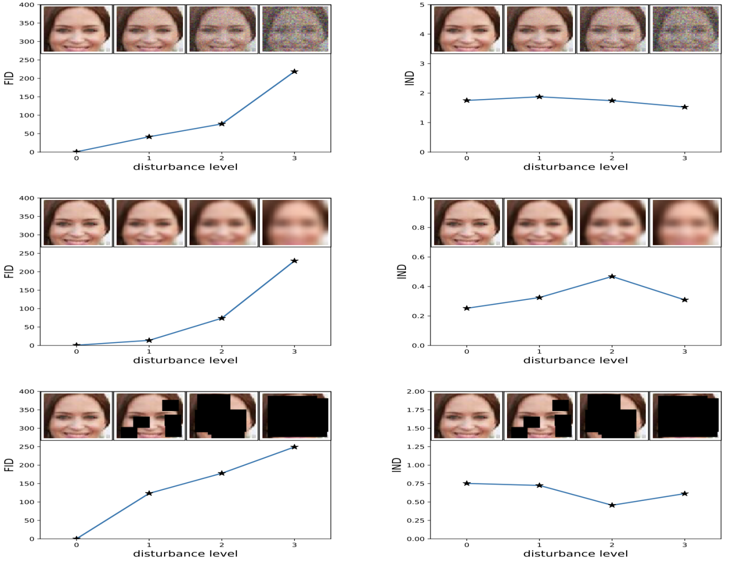
Frechet Inception Distance

$$D^2(\pi, p) = \|\mathbf{m}_\pi - \mathbf{m}_p\|_2^2 + \mathrm{Tr}\left(\mathbf{C}_\pi + \mathbf{C}_p - 2\sqrt{\mathbf{C}_\pi \mathbf{C}_p}\right)$$

- $\mathbf{m}_\pi$, $\mathbf{C}_\pi$ are mean vector and covariance matrix of feature representations for real samples from $\pi(\mathbf{x})$
- $\mathbf{m}_p$, $\mathbf{C}_p$ are mean vector and covariance matrix of feature representations for generated samples from $p(\mathbf{x}|\boldsymbol{\theta})$.
- Representations are output of intermediate layer from pretrained classification model.

https://arxiv.org/abs/1706.08500

Evaluation of likelihood-free models



https://arxiv.org/abs/1706.08500

Evaluation of likelihood-free models

Frechet Inception Distance

$$D^2(\pi, p) = \|\mathbf{m}_\pi - \mathbf{m}_p\|_2^2 + \mathrm{Tr}\left(\mathbf{C}_\pi + \mathbf{C}_p - 2\sqrt{\mathbf{C}_\pi \mathbf{C}_p}\right)$$

FID limitations

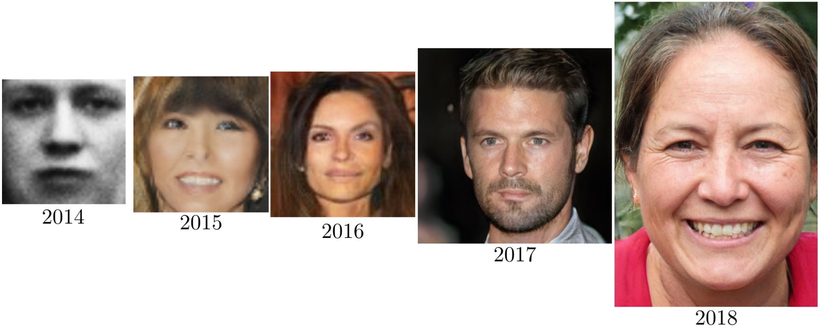
- FID depends on the pretrained classification model.
- FID needs a large samples size for evaluation.
- Calculation of FID is slow.
- FID extimates only two sample moments.

https://arxiv.org/abs/1706.08500

Summary

- Wasserstein GAN uses Kantorovich-Rubinstein duality to estimate Wasserstein distance.
- Gradient Penalty proposes the regularizer to enforce Lipschitzness.
- Spectral normalization is a weight normalization technique to enforce Lipshitzness.
- f-divergence family is a unified framework for divergence minimization.
- Inception Score and Frechet Inception Distance are the common metrics for GAN evaluation.

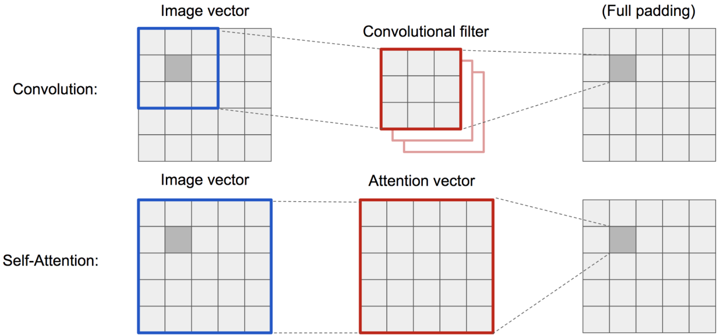
Evolution of GANs



- **Vanilla GAN** https://arxiv.org/abs/1406.2661
- **DCGAN** https://arxiv.org/abs/1511.06434
- **CoGAN** https://arxiv.org/abs/1606.07536
- **ProGAN** https://arxiv.org/abs/1710.10196
- **StyleGAN** https://arxiv.org/abs/1812.04948

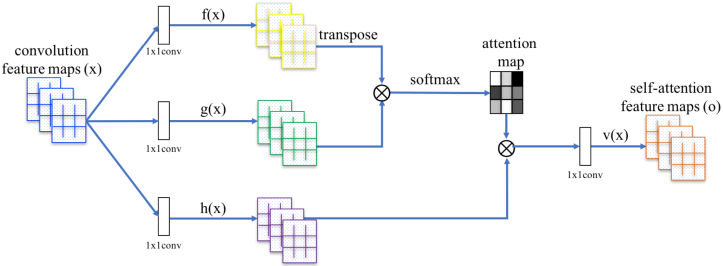
Self-Attention GAN

- Convolutional layers process the information in a local neighborhood.
- Using convolutional layers alone is computationally inefficient for modeling long-range dependencies in images.



https://lilianweng.github.io/lil-log/2018/06/24/attention-attention.html

Self-Attention GAN



- $\mathbf{x}$ – feature vector for one feature location.
- N – number of feature locations.

$$\mathbf{f}(\mathbf{x}) = \mathbf{W}_f(\mathbf{x}), \quad \mathbf{g}(\mathbf{x}) = \mathbf{W}_g(\mathbf{x}), \quad \mathbf{h}(\mathbf{x}) = \mathbf{W}_h(\mathbf{x}), \quad \mathbf{v}(\mathbf{x}) = \mathbf{W}_v(\mathbf{x})$$

$$s_{ij} = \mathbf{f}(\mathbf{x}_i)^T \mathbf{g}(\mathbf{x}_j), \quad a_{ij} = \frac{\exp s_{ij}}{\sum_{i=1}^N \exp s_{ij}}, \quad \mathbf{o}_j = \mathbf{v}\left(\sum_{i=1}^N a_{ij}\mathbf{h}(\mathbf{x}_i)\right)$$

https://arxiv.org/abs/1805.08318

Self-Attention GAN

Technical Details

- Hinge loss for training.
- Spectral Normalization in both the generator and the discriminator.
- Separate learning rates for the generator and the discriminator.

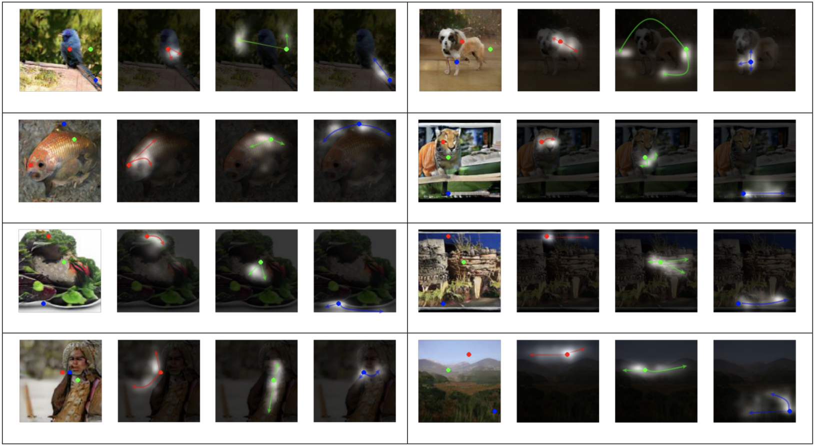
Model	no attention	SAGAN				Residual			
		$feat_8$	$feat_{16}$	$feat_{32}$	$feat_{64}$	$feat_8$	$feat_{16}$	$feat_{32}$	$feat_{64}$
FID	22.96	22.98	22.14	**18.28**	18.65	42.13	22.40	27.33	28.82
IS	42.87	43.15	45.94	51.43	**52.52**	23.17	44.49	38.50	38.96

Model	Inception Score	Intra FID	FID
AC-GAN (Odena et al., 2017)	28.5	260.0	/
SNGAN-projection (Miyato & Koyama, 2018)	36.8	92.4	27.62*
SAGAN	**52.52**	**83.7**	**18.65**

https://arxiv.org/abs/1805.08318

Self-Attention GAN

Visualization of attention maps



https://arxiv.org/abs/1805.08318

BigGAN

Model description

- Self-Attention GAN baseline.
- Class-conditional generator.
- Increasing batch size gives tremendous benefit (allows to cover more modes).
- Increasing model size is helpful (wider helps as much as deeper).
- Hinge loss for training.
- Orthogonal regularization for smoothness the generator output.
- Truncation trick for balancing between diversity and fidelity.

---

https://arxiv.org/abs/1809.11096

BigGAN

- Orthogonal regularization

$$\|\mathbf{W}^T\mathbf{W} - \mathbf{I}\|^2 \quad \Rightarrow \|\mathbf{W}^T\mathbf{W} - \mathrm{diag}(\mathbf{W}^T\mathbf{W})\|^2$$

- Truncation trick. Coordinates of samples $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$ which fall outside a predefined range are resampled to fall inside that range.

Batch	Ch.	Param (M)	Shared	Skip-z	Ortho.	Itr $\times 10^3$	FID	IS
256	64	81.5	SA-GAN Baseline			1000	18.65	52.52
512	64	81.5	✗	✗	✗	1000	15.30	58.77($\pm$1.18)
1024	64	81.5	✗	✗	✗	1000	14.88	63.03($\pm$1.42)
2048	64	81.5	✗	✗	✗	732	12.39	76.85($\pm$3.83)
2048	96	173.5	✗	✗	✗	295($\pm$18)	9.54($\pm$0.62)	92.98($\pm$4.27)
2048	96	160.6	✓	✗	✗	185($\pm$11)	9.18($\pm$0.13)	94.94($\pm$1.32)
2048	96	158.3	✓	✓	✗	152($\pm$7)	8.73($\pm$0.45)	98.76($\pm$2.84)
2048	96	158.3	✓	✓	✓	165($\pm$13)	8.51($\pm$0.32)	99.31($\pm$2.10)
2048	64	71.3	✓	✓	✓	371($\pm$7)	10.48($\pm$0.10)	86.90($\pm$0.61)

https://arxiv.org/abs/1809.11096

BigGAN

Samples (512x512)

https://arxiv.org/abs/1809.11096

BigGAN

Interpolations

https://arxiv.org/abs/1809.11096

Progressive Growing GAN

Problems with HR image generation

- Disjoint manifolds $\Rightarrow$ gradient problem.
- Small minibatch $\Rightarrow$ training instability.

Solution

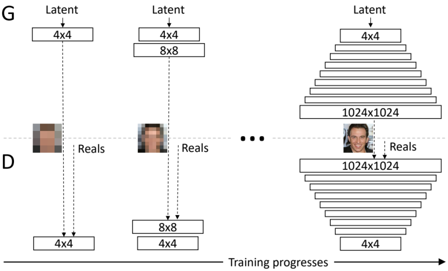
Grow both the generator and discriminator progressively, starting from LR images, and add new layers that introduce higher-resolution details as the training progresses.

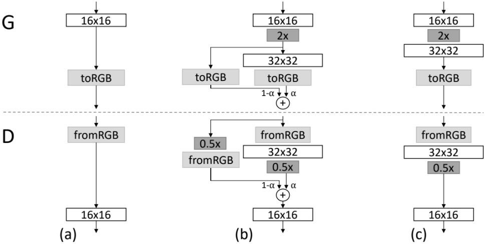
- Train GAN which generate 4x4 images (just 2 convolutions for G and D).
- Add upsampling layers to G, downsampling layers to D.
- Train GAN which generate 8x8 images.
- etc.

---

https://arxiv.org/abs/1710.10196

Progressive Growing GAN





https://arxiv.org/abs/1710.10196

Progressive Growing GAN

Samples (1024x1024)

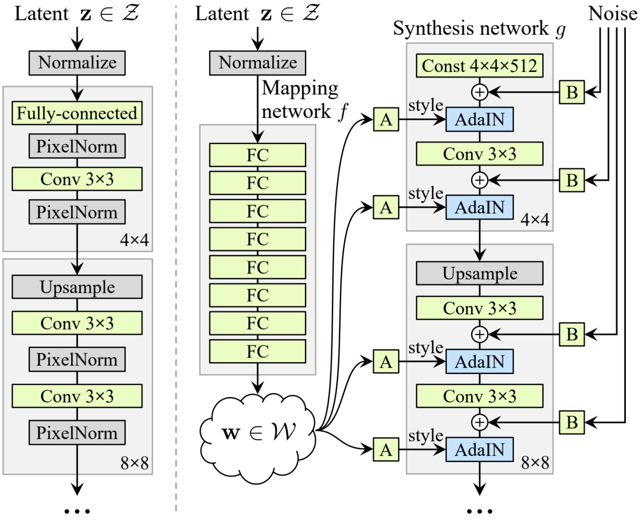
https://arxiv.org/abs/1710.10196

StyleGAN

	Method	CelebA-HQ	FFHQ
A	Baseline Progressive GAN [30]	7.79	8.04
B	+ Tuning (incl. bilinear up/down)	6.11	5.25
C	+ Add mapping and styles	5.34	4.85
D	+ Remove traditional input	5.07	4.88
E	+ Add noise inputs	**5.06**	4.42
F	+ Mixing regularization	5.17	**4.40**

https://arxiv.org/abs/1812.04948

StyleGAN



(a) Traditional (b) Style-based generator

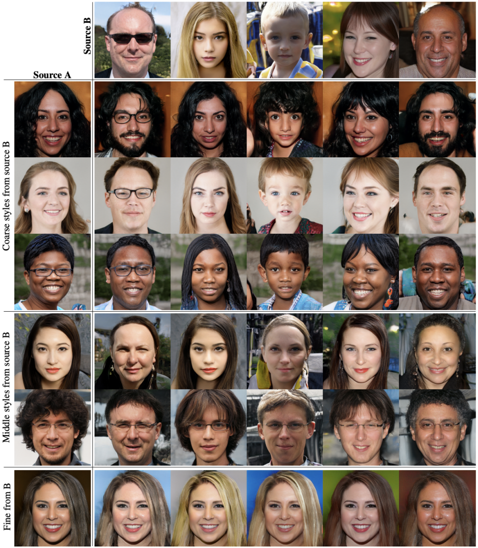
https://arxiv.org/abs/1812.04948

StyleGAN

Samples (1024x1024)

https://arxiv.org/abs/1812.04948

StyleGAN



https://arxiv.org/abs/1812.04948

References

- A Note on the Inception Score
  https://arxiv.org/abs/1801.01973
  Summary: Inception Score is not an ideal metric.
- GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium
  https://arxiv.org/abs/1706.08500
  Summary: Frechet inception distance was proposed for GAN evaluation.
- **SAGAN**: Self-Attention Generative Adversarial Networks
  https://arxiv.org/abs/1805.08318
  Summary: Self-attention was proposed for G and D. Hinge loss was used for training. Spectral Normalization was injected not only for D, but also for G.
- **BigGAN**: Large Scale GAN Training for High Fidelity Natural Image Synthesis
  https://arxiv.org/abs/1809.11096
  Summary: SAGAN as a baseline. High-quality image generation. Increasing batch is really helpful (covering more modes). Propose orthogonalization regularization. Use truncation trick for trade-off between sample fidelity and variety.
- **ProGAN**: Progressive Growing of GANs for Improved Quality, Stability, and Variation
  https://arxiv.org/abs/1710.10196
  Summary: The key idea is to grow both the generator and discriminator progressively: starting from a low resolution, we add new layers that model increasingly fine details as training progresses.