

1. どんなもの？

実践的な視点からGANsの現状を分析した論文.

GAN Landscapeを深く調査するだけでなく，現状のSOTAを再現した.

2. 先行研究と比べてどこがすごいの？

先行研究の調査論文.

Loss, Regularization and Normalization, architecturesの3つ観点からGAN Landscapeを調査.

落とし穴，再現性の問題，実践的な考慮事項を議論.

3. 技術や手法の“キモ”はどこにある？

3.1 Loss Functions

- Q をモデル分布， P をターゲット分布とする
- Non-saturating loss[8]
 - P と Q 間のJensen-Shannon divergenceを最小化することに等しい.

$$\mathcal{L}_D = -\mathbb{E}_{x \sim P}[\log(D(x))] - \mathbb{E}_{\hat{x} \sim Q}[\log(1 - D(\hat{x}))]$$

$$\mathcal{L}_G = -\mathbb{E}_{\hat{x} \sim Q}[\log(D(\hat{x}))]$$

- Wasserstein GAN(WGAN)[2]
 - P と Q 間のWasserstein distanceを最小化する.
 - 最適なDiscriminatorのもとで証明されている最適輸送への接続を利用することで，Generatorに関する評価関数の最小化は， P と Q 間のWasserstein距離の最小化である.
 - 欠点は，Kantorovich-Rubinstein 双対性を利用することによる1-Lipschitz Discriminatorを補償しなければならないこと.

$$\mathcal{L}_D = -\mathbb{E}_{x \sim P}[D(x)] + \mathbb{E}_{\hat{x} \sim Q}[D(\hat{x})]$$

$$\mathcal{L}_G = -\mathbb{E}_{\hat{x} \sim Q}[D(\hat{x})]$$

- Least squared loss[19]
 - P と Q 間のPearson χ^2 divergenceを最小化することに対応.
 - 直観的に, この損失は滑らかで, JS定式化のsigmoid cross entropy よりゆっくりと飽和する.

$$\mathcal{L}_D = -\mathbb{E}_{x \sim P}[(D(x) - 1)^2] + \mathbb{E}_{\hat{x} \sim Q}[D(\hat{x})^2]$$

$$\mathcal{L}_G = -\mathbb{E}_{\hat{x} \sim Q}[(D(\hat{x}) - 1)^2]$$

3.2 Regularization and Normalization of the Discriminator

- Gradient norm penalty
 - 1-Lipschitznessに反して, 簡単な制約として解釈できる(WGAN-GP). 代わりの最適結合として, 勾配は訓練データと生成データ間の線形補間を評価する.
 - 欠点は, 訓練中の変化するモデル分布 Q に依存することである.
- Discriminator normalization
 - 最適化からの視点と表現力からの視点の両方から役にたつ.
 - 最適化の視点から
 - Batch normalization
 - レイヤーの活性化前のノードを平均 β と標準偏差 γ で正規化する. ここで β と γ は, 各レイヤーの各ノードに関して学習されるパラメータ. 正規化はバッチレベルで行われ, 各ノードに関しては別々に行われる.
 - Layer normalization
 - レイヤー内の全ての隠れノードは, 同じ正規化項 β と γ と共有する. しかし, 異なるデータは, 別々に正規化される.
 - 表現力からの視点から
 - NNを合成写像として考え, spectralの特性分析する. 特に, Discriminatorに関して, 線形演算子を有界にすることは, 最大

特異値を制御すればよい(Spectral normalization[20]).

- 注意：この手法は、線形レイヤーとReLU活性化の1-Lipschitznessが保証する。しかし、畳み込みマップのカーネルのスペクトルノルムを1に制限することは、畳み込み写像のスペクトルノルムを1に制限することではない。実際、使用される stride と padding に依存する。

3.3 Generator and Discriminator Architecture

- DCGANとResNetの構造を考える。これらの構造は、どこにでもいるため、 128×128 の画像で考える。
- **SNDCGAN**

Table 2: SNDCGAN architecture.

(a) SNDCGAN discriminator			(b) SNDCGAN generator		
LAYER	KERNEL	OUTPUT SHAPE	LAYER	KERNEL	OUTPUT SHAPE
Conv, lReLU	[3, 3, 1]	$h \times w \times 64$	z	-	128
Conv, lReLU	[4, 4, 2]	$h/2 \times w/2 \times 128$	Linear, BN, ReLU	-	$h/8 \times w/8 \times 512$
Conv, lReLU	[3, 3, 1]	$h/2 \times w/2 \times 128$	Deconv, BN, ReLU	[4, 4, 2]	$h/4 \times w/4 \times 256$
Conv, lReLU	[4, 4, 2]	$h/4 \times w/4 \times 256$	Deconv, BN, ReLU	[4, 4, 2]	$h/2 \times w/2 \times 128$
Conv, lReLU	[3, 3, 1]	$h/4 \times w/4 \times 256$	Deconv, BN, ReLU	[4, 4, 2]	$h \times w \times 64$
Conv, lReLU	[4, 4, 2]	$h/8 \times w/8 \times 512$	Deconv, Tanh	[3, 3, 1]	$h \times w \times 3$
Conv, lReLU	[3, 3, 1]	$h/8 \times w/8 \times 512$			
Linear	-	1			

Figure 1

- **ResNet**

Table 3: ResNet 19 architecture corresponding to "resnet_small" in https://github.com/pfnet-research/sngan_projection

(a) ResNet19 discriminator				(b) ResNet19 generator			
LAYER	KERNEL	RS	OUTPUT SHAPE	LAYER	KERNEL	RS	OUTPUT SHAPE
ResBlock	[3, 3, 1]	D	$64 \times 64 \times 64$	z	-	-	128
ResBlock	[3, 3, 1]	D	$32 \times 32 \times 128$	Linear	-	-	$4 \times 4 \times 512$
ResBlock	[3, 3, 1]	D	$16 \times 16 \times 256$	ResBlock	[3, 3, 1]	U	$8 \times 8 \times 512$
ResBlock	[3, 3, 1]	D	$8 \times 8 \times 256$	ResBlock	[3, 3, 1]	U	$16 \times 16 \times 256$
ResBlock	[3, 3, 1]	D	$4 \times 4 \times 512$	ResBlock	[3, 3, 1]	U	$32 \times 32 \times 256$
ResBlock	[3, 3, 1]	D	$2 \times 2 \times 512$	ResBlock	[3, 3, 1]	U	$64 \times 64 \times 128$
ReLU, MP	-	-	512	ResBlock	[3, 3, 1]	U	$128 \times 128 \times 64$
Linear	-	-	1	BN, ReLU	-	-	$128 \times 128 \times 64$
				Conv	[3, 3, 1]	-	$128 \times 128 \times 3$
				Sigmoid	-	-	$128 \times 128 \times 3$

Table 4: ResNet block definition.

(a) ResBlock discriminator				(b) ResBlock generator			
LAYER	KERNEL	RS	OUTPUT SHAPE	LAYER	KERNEL	RS	OUTPUT SHAPE
Shortcut	[3, 3, 1]	D	$h/2 \times w/2 \times c_{out}$	Shortcut	[3, 3, 1]	U	$2h \times 2w \times c_{out}$
BN, ReLU	-	-	$h \times w \times c_{in}$	BN, ReLU	-	-	$h \times w \times c_{in}$
Conv	[3, 3, 1]	-	$h \times w \times c_{out}$	Conv	[3, 3, 1]	U	$2h \times 2w \times c_{out}$
BN, ReLU	-	-	$h \times w \times c_{out}$	BN, ReLU	-	-	$2h \times 2w \times c_{out}$
Conv	[3, 3, 1]	D	$h/2 \times w/2 \times c_{out}$	Conv	[3, 3, 1]	-	$2h \times 2w \times c_{out}$
Addition	-	-	$h/2 \times w/2 \times c_{out}$	Addition	-	-	$2h \times 2w \times c_{out}$

Figure2

3.4 Evaluation Metrics

- Inception Score (IS)

- 直観的に、意味のある物体を含むサンプルの条件付きラベル分布は低いエントロピーを持つべきである。また、サンプルの多様性はエントロピーが高くなければいけない。これらは、以下のようにあらわされる。

$$IS = \exp(E_{x \sim Q}[d_{KL}(p(y|x), p(y))])$$

- 欠点は、直観的にラベル全体の事前分布を含んでおり、適切な距離でないことである。

- Frechet Inception Distance (FID)

- P と Q からのサンプルは最初に特徴空間(InceptionNetの特定の層)に埋め込む。埋め込まれたデータは、多変量ガウス分布に従うと仮定する。平均と共分散は推定され、2つガウス分布間のFrechet distance が計算される。

$$FID = ||\mu_x - \mu_y||_2^2 + \text{Tr}(\Sigma_x + \Sigma_y - 2(\Sigma_x \Sigma_y)^{\frac{1}{2}})$$

- ここで、 (μ_x, Σ_x) と (μ_y, Σ_y) は P と Q からの埋め込まれたサンプルの平均と共分散である。
- FIDは、ISよりも、1クラスにつき1つの画像だけを生成するモデル（mode-collapse状態のモデル）は、完全なISを得ることができるが、悪いFIDとなる。
- Multi-scale Structural Similarity for Image Quality (MS-SSIM) and Diversity
 - GANsにおける最大の問題点は、mode-collapseとmode-droppingである。生成サンプルの多様性に関して、代わりとして、与えられたクラス内のペアごとのMS-SSIMスコアを使うことが提案されている。
 - 欠点として、クラスに対応する生成されたサンプルを知らないという点がある。だから、1つのクラスデータセットに一般的に適用される

4. どうやって有効だと検証した？

- CIFAR10, CELEBA-HQ-128, LSUN-BEDROOMのデータセットを使用
- DiscriminatorとGeneratorアーキテクチャによる違い

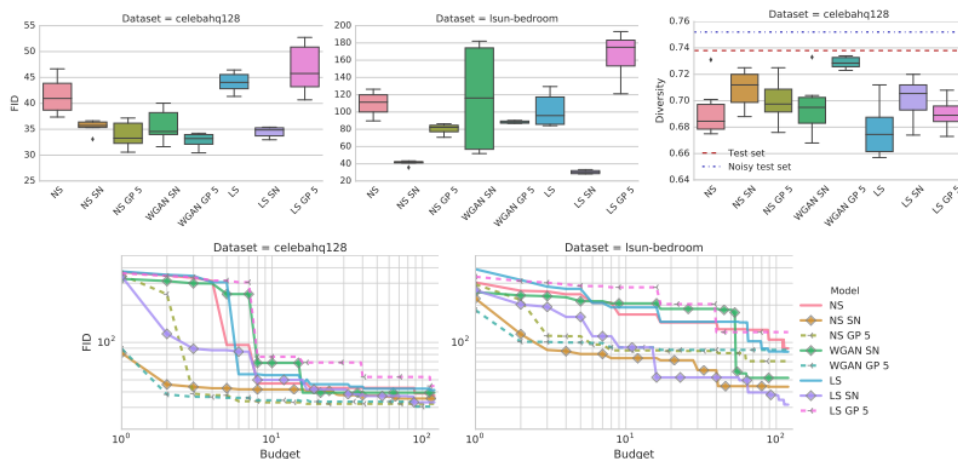


Figure 1: The non-saturating (NS) loss is stable over both data sets. Gradient penalty and spectral normalization improve the model quality. From the computational budget perspective (i.e. how many models one needs to train to reach a certain FID), both spectral normalization and gradient penalty perform better than the baseline, but the former is more efficient.

Figure3

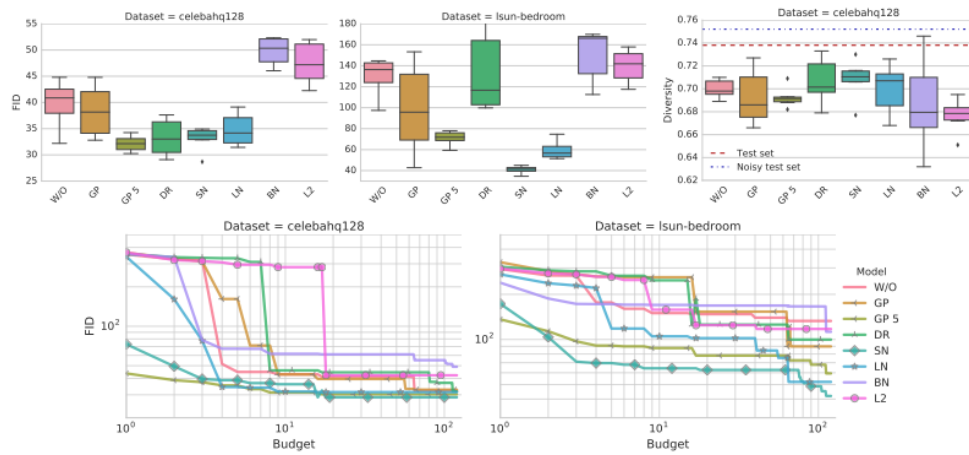


Figure 2: Both the gradient penalty (GP) and spectral normalization (SN) perform well and should be considered as viable approaches, and the latter is computationally cheaper. Unfortunately none fully address the stability issues.

Figure4

- Spectral normalizationとGradient penaltyの組み合わせ 結果

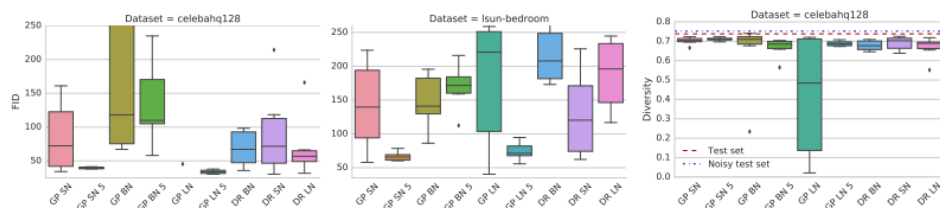


Figure 3: Gradient penalty coupled with spectral normalization (SN) or layer normalization (LN) strongly improves the performance over the baseline.

Figure5

- DiscriminatorとGeneratorアーキテクチャによる違い

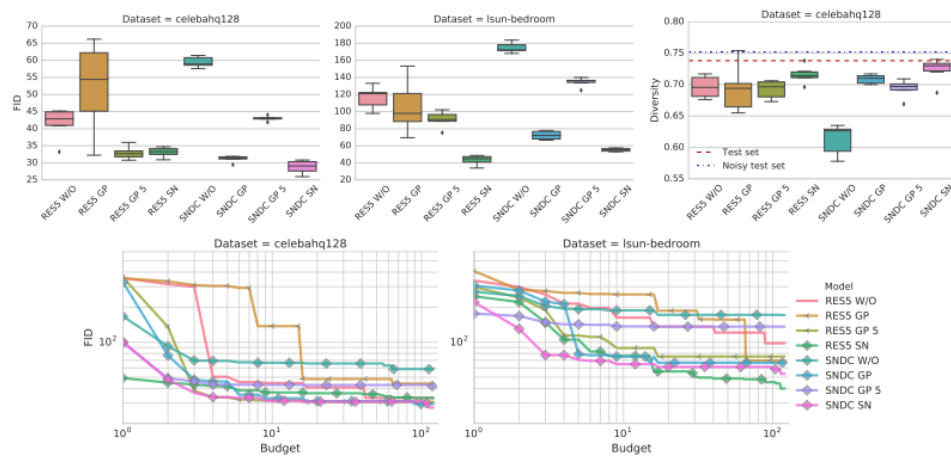


Figure 4: Impact of the discriminator and generator architecture for the non-saturating GAN loss. Both Spectral normalization and Gradient penalty can help improve upon the non-regularized baseline.

Figure6

- デフォルトの選択として, non-saturating lossとspectral normalizationを使うべき.
- 追加の計算予算が与えられれば, gradient penaltyを追加し, 収束するまでモデルを訓練することを提案
- わずかな追加の利益は, 正規化と正則化を組み合わせることによって得られる. 正規化と正則化は, Discriminatorのリプシッツ定数の重要性を経験的に確認しているものを使用する.
- ResNet styleアーキテクチャに適用された多くのトリックは, 品質においてわずかな変化をもたらす. 高い計算コストによって, 避けるべきである.

5. 議論はあるか？

- Generatorにも, spectral normalizationのような表現能力に関する視点を持たせたほうがいいのか
- 一般的な問題として, メトリックの問題, データセットの前処理, 非決定的, 実装の詳細の欠如がある.

6. 次に読むべき論文はあるか？

- WGAN-GP
 - Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of Wasserstein GANs. Advances in Neural Information Processing Systems (NIPS), 2017.
- MS-SSIM
 - Zhou Wang, Eero P Simoncelli, and Alan C Bovik. Multiscale structural similarity for image quality assessment. In Asilomar Conference on Signals, Systems and Computers, 2003.
 - Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier GANs. In International Conference on Machine Learning (ICML), 2017.
 - William Fedus, Mihaela Rosca, Balaji Lakshminarayanan, Andrew M Dai, Shakir Mohamed, and Ian Goodfellow. Many paths to equilibrium: Gans do not need to decrease adivergence at every

step. In International Conference on Learning Representations (ICLR), 2018.

- Inception Score
 - Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In Advances in Neural Information Processing Systems (NIPS), 2016.
- FID
 - Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, Günter Klambauer, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a Nash equilibrium. Advances in Neural Information Processing Systems (NIPS), 2017.

論文情報・リンク

- Kurach, K., Lucic, M., Zhai, X., Michalski, M., & Gelly, S. (n.d.). The GAN Landscape: Losses, Architectures, Regularization, and Normalization.