Spectral Normalization Demystified

Maxim Kochurov, Rasul Karimov, Sergei Kozlukov

Skoltech

Dec 2018

Background

Myato et al. did great a job.

Spectral Normalization for Generative Adversarial Networks
https://arxiv.org › cs ▾
by T Miyato - 2018 - Cited by 157 - Related articles
Feb 16, 2018 - **Spectral Normalization for Generative Adversarial Networks**. One of the challenges in the study of **generative adversarial networks** is the instability of its training. In this paper, we propose a novel weight **normalization** technique called **spectral normalization** to stabilize the training of the discriminator.

157 references so far. But what did they did so useful?

Motivation

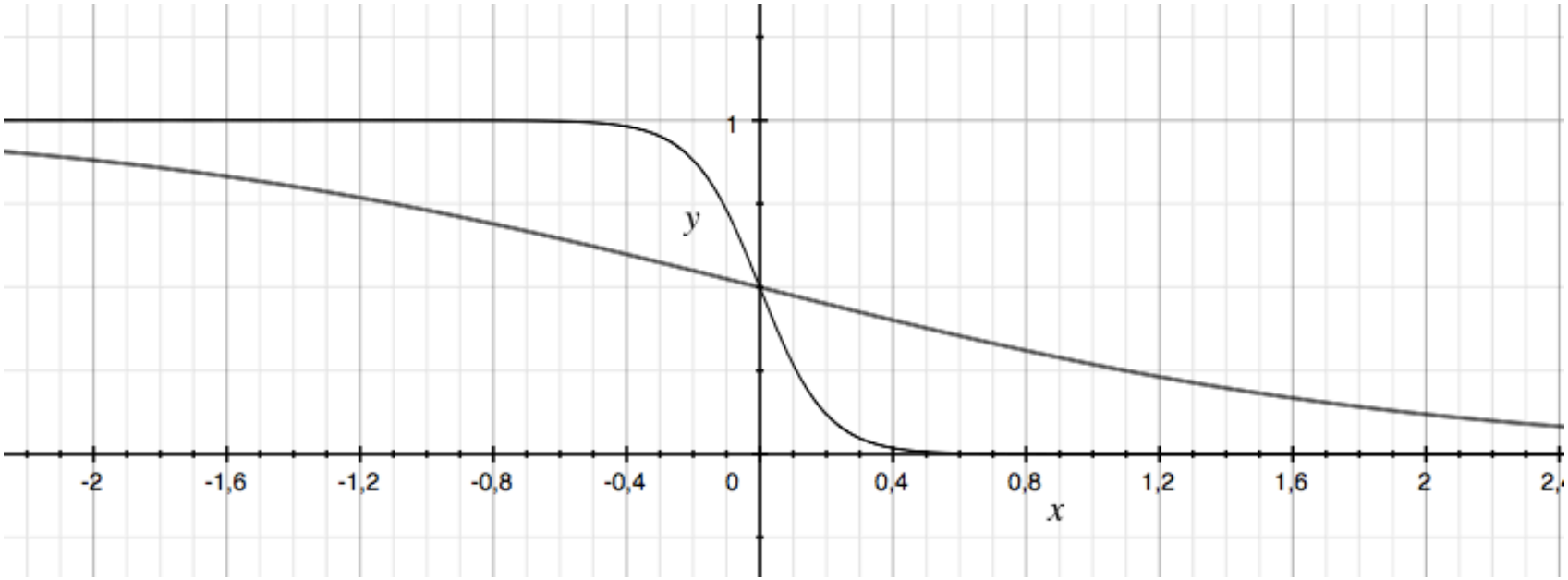
It is known [AB17] that training GANs is in many ways a peculiar problem:

- GANs cannot fit continuous distributions.
- **Perfect discriminator**: training D_W with g_θ fixed results in $\nabla_x D_W(x) \to 0$ which prevents further training of generator.
- **Mode collapse**: Training g_θ with D_W fixed makes generator learn just one "perfect" fake.

[Miy+18] proposed to use "Spectral Normalization" to regularize the discriminator and fix problems of vanishing gradients.

Why it should work?

Main claim in that paper was to bound Lipshitz constant of discriminator. That is a reasonable point. If you see on the sigmoid function, Lipshitz constraint prevents from vanishing gradients.



Spectral normalization - how does it work?

Dense Case

$$v \leftarrow \frac{W^\top u}{\|W^\top u\|_2},$$

$$u \leftarrow \frac{Wv}{\|Wv\|_2},$$

$$\sigma(W) \approx u^\top W v$$

for $W \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$.

Convolutional Case

$$\overline{W} = \text{reshape}(W),$$

$$\tilde{v} \leftarrow \frac{\overline{W}^\top \tilde{u}}{\|\overline{W}^\top \tilde{u}\|_2},$$

$$\tilde{u} \leftarrow \frac{\overline{W}\tilde{v}}{\|\overline{W}\tilde{v}\|_2},$$

$$\sigma(\overline{W}) \approx \tilde{u}^\top \overline{W} \tilde{v}$$

for $W \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}} \times h \times w}$ and $\overline{W} \in \mathbb{R}^{d_{\text{out}} \times (d_{\text{in}} h w)}$.

Then set $W \leftarrow \bar{W}/\sigma(\bar{W})$

.

Fair spectral norm

Relationship of singular values of $\overline{W}$ and singular values of convolution with W is a mystery.

$$v \leftarrow \frac{W^\top * u}{\|W^\top * u\|_2},$$

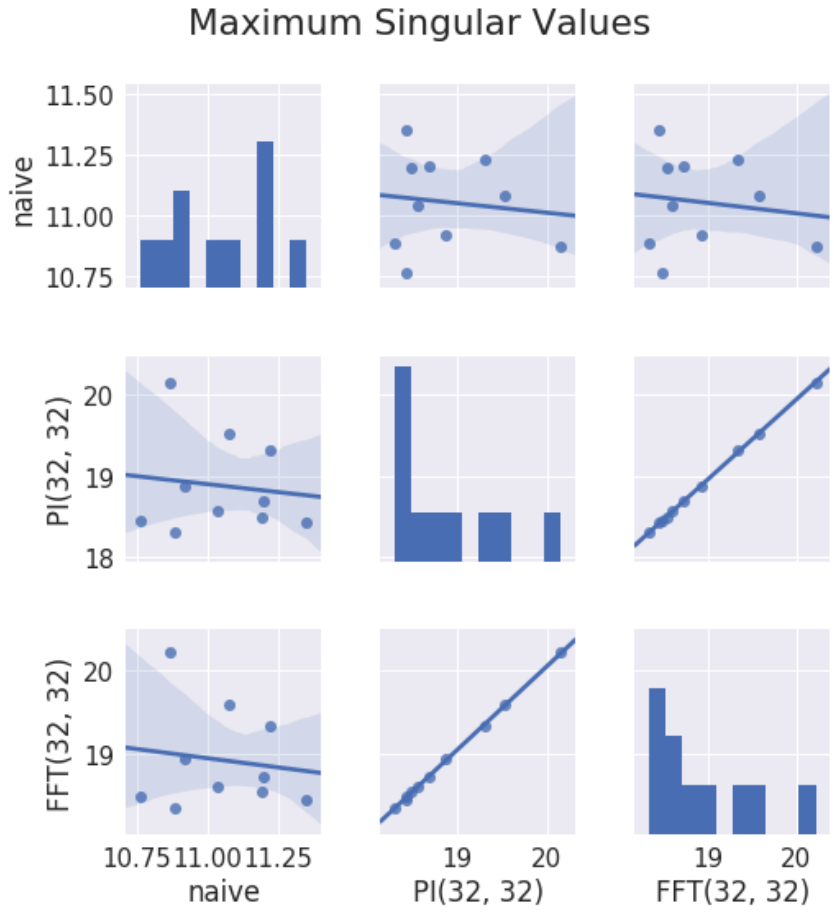
$$u \leftarrow \frac{W * v}{\|W * v\|_2},$$

$$\sigma(W) \approx \operatorname{vec}(u)^\top \operatorname{vec}(W * v)$$

for $W \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}} \times h \times w}$ and $*$ denoting convolution (possibly transposed convolution with $W^\top$).
Miyatu's "wrong" spectral norm performs way better than correct one.

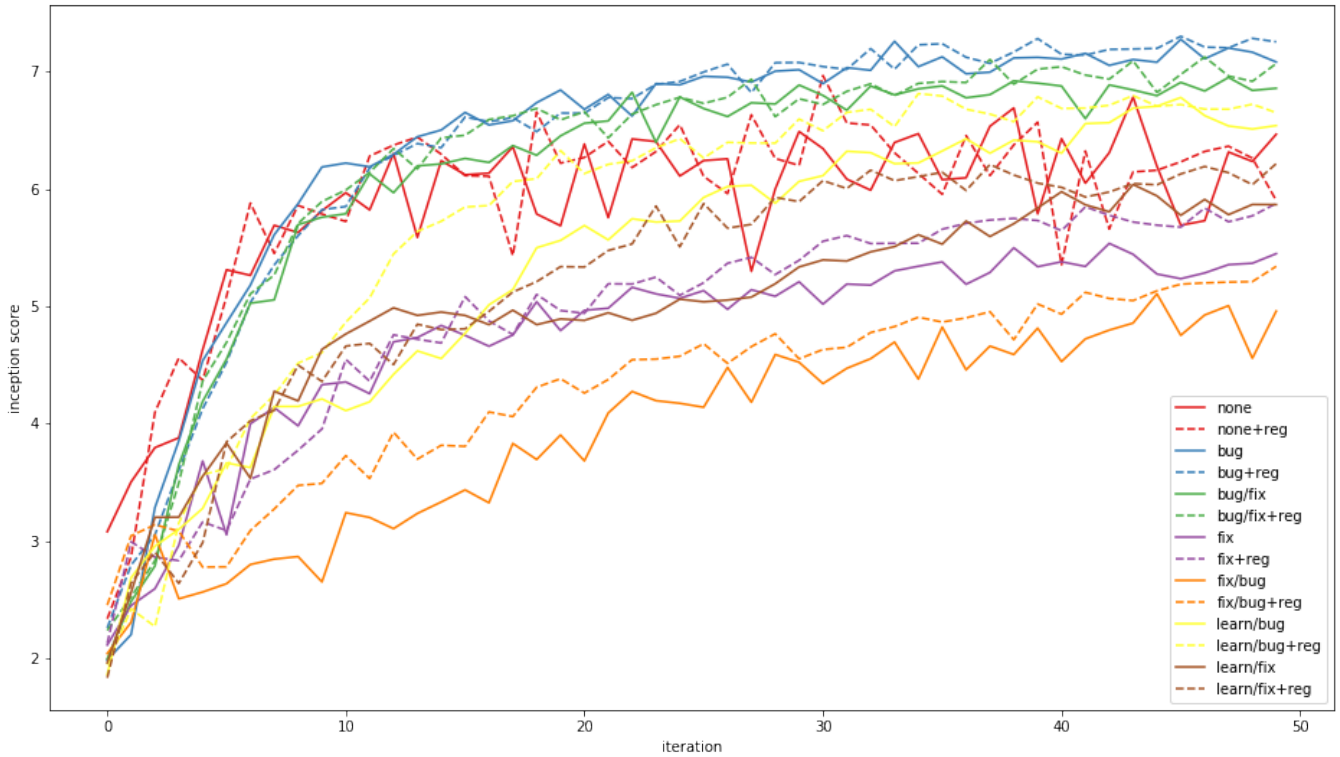
Things differ a lot

- Singular values of reshaped kernel sometimes over- and sometimes under-estimate actual singular values.
- Resulting Lipshitz constant for discriminator is higher than expected



Things do not work

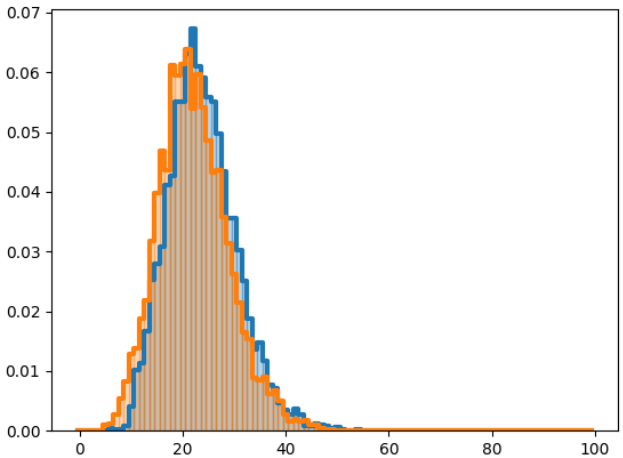
Inception scores training curve



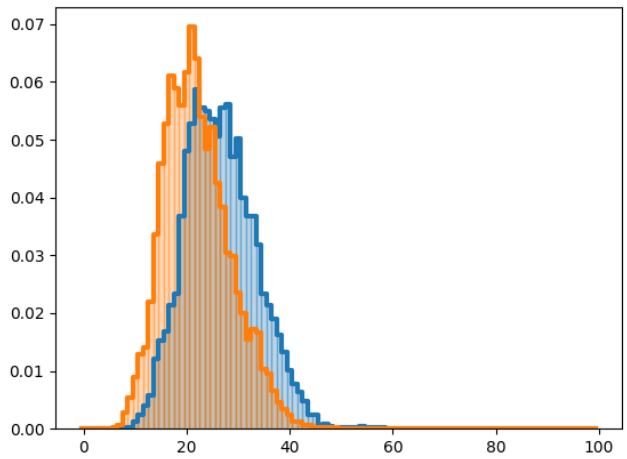
We tried make things work with fair convolutional power iteration, but the most stable run was using power iteration with reshaping (blue curve).

Things do not work

Geometry score for fair and bugged spectral norm



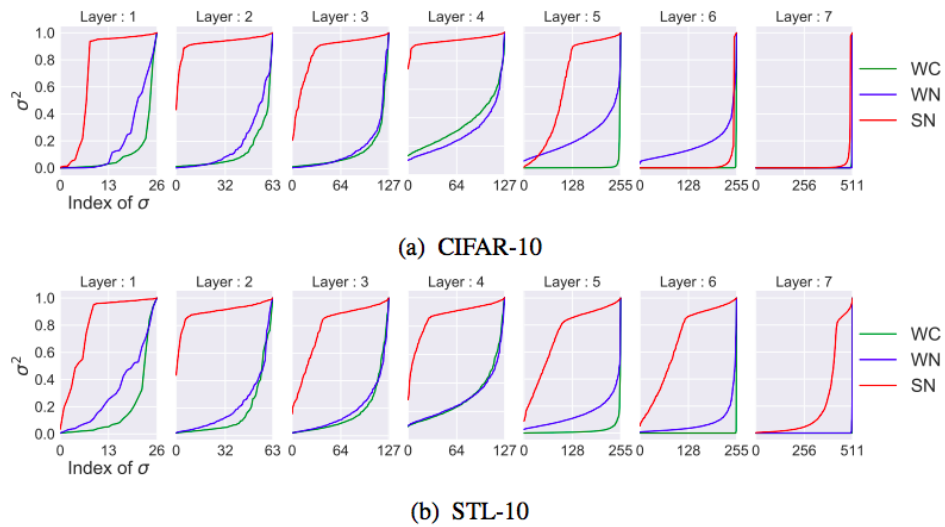
Bugged Geom. score: 0.0019



Fair Geom. score: 0.0113

Closer Look

- With fair spectral norm the performance is worse.
- Looking at spectrum curves one might notice that for red (SN) curve is concave, while others (not SN) are convex.
- Is it good or bad? Can we check?



(a) CIFAR-10

(b) STL-10

Figure 3: Squared singular values of weight matrices trained with different methods: Weight clipping (WC), Weight Normalization (WN) and Spectral Normalization (SN). We scaled the singular values so that the largest singular values is equal to 1. For WN and SN, we calculated singular values of the normalized weight matrices.

Any Ideas How to Check?

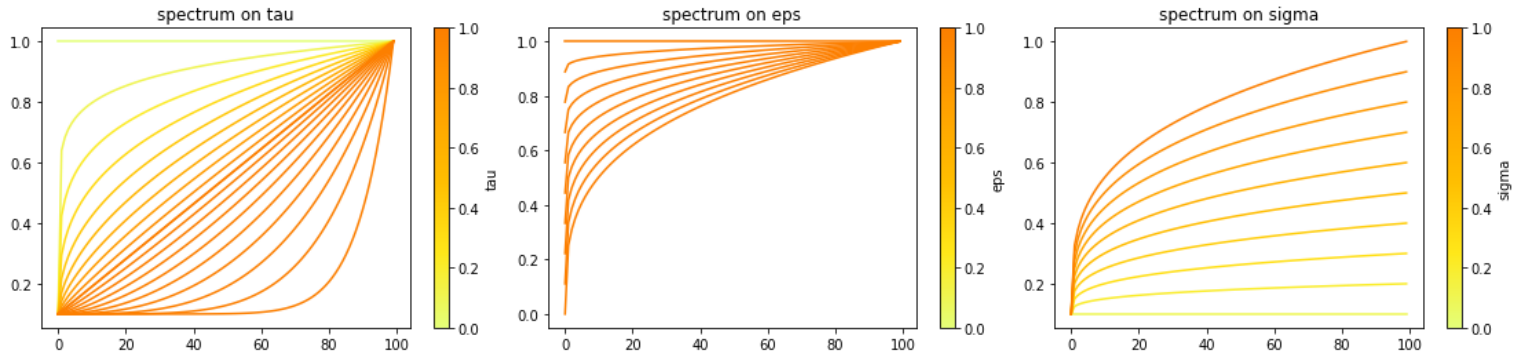
That is how we come up with further experiments:

- We need somehow control spectrum form.
- It is challenging to perform SVD every time.
- We can try to use Riemannian optimization and fix spectrum.
- General SVD parameterization [Mis+12] is still too hard.
- We decide to try out $W = \mathrm{diag}(\lambda)X$, with orthogonal X: $X^\top X = I$. This is the case of optimization on a Stiefel manifold.
- Does orthogonality matter? (seems to be ok)

We still struggle with checking these hypotheses. Any other ideas are welcome!

Results for Riemannian Approach

In short: results are bad. Trained models have much worse visual and quantitative performance. We parameterized spectrum with a (non-)learnable curve $\lambda(\tau, \varepsilon, \sigma)$ or just learnable grid λ_{grid}



We trained weights $W = \text{diag}(\lambda)X$ using Riemannian Adam optimizer from here: `https://github.com/ferrine/geoopt`.

Possible Explanations of Failure

- Optimization in interior is not possible (main hypothesis)
- Sorted Spectrum: **both** sorted/not sorted **worked bad** in grid parameterization
- Implementation of Riemannian Adam (tested on an optimization task)
- Side effects of adaptive scheme (not checked, can try RSGD)

Gradient Norm Analysis

Parametrization matters: learning maximum singular vars leads to vanishing gradients

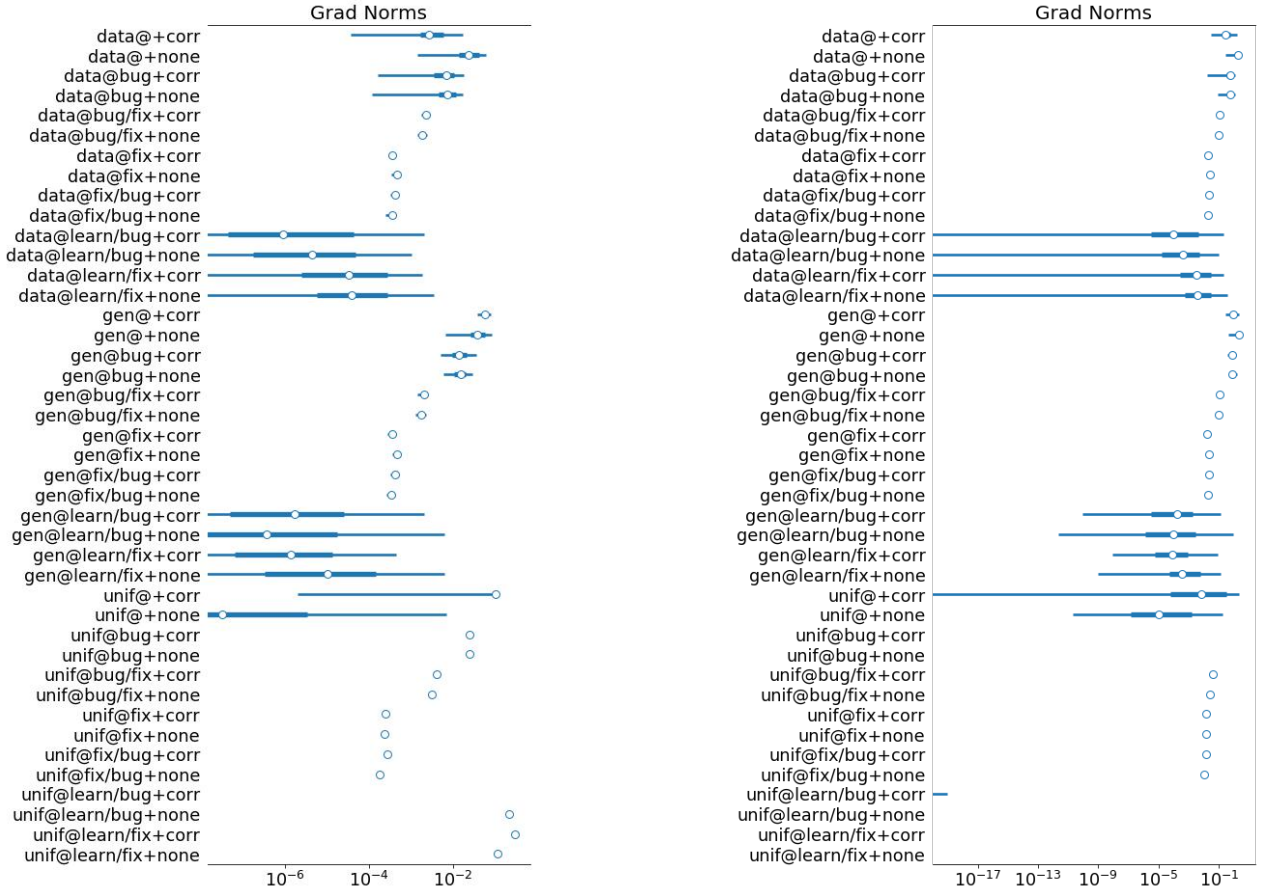


Figure: Gradients with respect to Loss and Sigmoid

References I

Martin Arjovsky and Léon Bottou. "Towards principled methods for training generative adversarial networks". In: *arXiv preprint arXiv:1701.04862* (2017).

Bamdev Mishra et al. "Fixed-rank matrix factorizations and Riemannian low-rank optimization". In: *CoRR* abs/1209.0430 (2012). arXiv: 1209.0430. URL: http://arxiv.org/abs/1209.0430.

Takeru Miyato et al. "Spectral normalization for generative adversarial networks". In: *arXiv preprint arXiv:1802.05957* (2018).