Improved Generative Models through Topological Data Analysis

Adrian Lopez,John Peterson, Anurag Kashyap, and Nirav Patel

May 3, 2020

Abstract

Generative Adversarial Networks have made significant progress in the field of image generation, but they have serious downsides including training difficulties due to mode collapse. We propose two novel solutions based on persistence homology, a method from the field of topological data analysis, in an attempt to generate better quality images and reduce training difficulty. The first being incorporating topological information into the loss function of the generator network. The second solution is implementing a new topological layer into the discriminator network. We present and compare the results of these two techniques against a baseline GAN trained with the standard architecture all trained on the MNIST handwritten digit data set. Our results show that adding a topological loss function both improves performance and training speed, while adding a topological layer hurt both performance and training speed.

1 Introduction

A generative adversarial network (GAN) consists of two deep neural networks used to estimate generative models via an adversarial process [3]. Two models are trained, namely G known as the generator and D known as the discriminator, where G captures the distribution of the data, and D estimates the probability that the model came from the training data rather than from G [3]. GANs have had tremendous success in generative settings, however they still face challenges in producing high quality images [8]. In this paper, we aim to enhance the image generation quality of GANs by incorporating topological information as part of the model during training.

Topological data analysis (TDA) is a field which allows one to study the shape and structure of data by identifying topological features such as connected components and loops [1]. TDA makes use of the concept of topological concept of homology, which allows us to characterize and count these topological features. It does so through persistent homology, which is an incredibly useful tool for understanding the topology of point clouds. By computing filtrations of simplicial complices, we can identify birth and death times of n-dimensional features, which can be plotted on persistence diagrams.

We aimed to enhance our results by incorporating topological information to our GAN through two ways: one by incorporating topological information in the loss function, and by adding a differentiable topological layer that incorporates topological information by means of persistent homology [2]. This can have a profound effect during training as it can be used to construct and incorporate topological priors as well as to regularize the weights during learning respectively [2]. To test the effects, we trained three different GANs using the MNIST dataset, one acting as a control containing no topological information, one containing a topological loss function, and another one containing a topological layer. Then we compared the results of these two experiments against the control.

2 Literature Review

Much previous work has looked into boosting GAN performance in image synthesis, such as two GANs stacked together, using an autoencoder to learn the high-level structure of images, while using a denoiser network to cature photo realistic effects, and using semantic label maps on conditianal GANs [8] [4] [7].

Recently, Gabrielsson, et. al. constructed a differential topological layer that computes persistent homology based on level set filtrations and edge based filtrations in an aim to capture topological information during training. This layer has a myriad of uses, one being for regularization given that topology explores the "general structure" of data, as well as for other uses such as to incorporate a topological prior as part of the loss function[2].

3 Methodology

3.1 Dataset

To train the GANs, we used the MNIST dataset [6].

3.2 Generative Adverserial Network

Discriminator

We composed the discriminator as follows:

- Convolutional layer with dimension 784,1024
- Leaky ReLu with $\alpha = .2$
- Dropout Layer
- Convolutional layer with dimension 1024,512

- Leaky ReLu with $\alpha = .2$
- Dropout Layer
- Convolutional layer with dimension 512,256
- Leaky ReLu with $\alpha = .2$
- Dropout Layer
- Convolutional with dimension 256,1
- Sigmoid

Generator

We composed the generator as follows:

- Convolutional layer with dimension 100,128
- Leaky ReLu with $\alpha = .2$
- Convolutional layer with dimension 128,256
- Batch Normalization
- Leaky ReLu with $\alpha = .2$
- Convolutional layer with dimension 256,512
- Leaky ReLu with $\alpha = .2$
- Convolutional layer with dimension 512,1024
- Leaky ReLu with $\alpha = .2$
- Convolutional layer with dimension 1024
- Tanh

3.3 Topological Loss Function

The persistence diagram is constructed from a topological level set filtration. The loss function defined from the persistence diagrams is given by,

$$\mathcal{E}(p, q, i_0; PD) = \sum_{i=i_0}^{\infty} |d_i - b_i|^p (\frac{d_i + b_i}{2})^q$$

Summing over the lifetimes i, beginning at i_0, the most persistent point in the persistence diagram, PD. The values of b_i and d_i refer to the birth and death data respectively. The loss function allows the improved topological generator model to produce images with the right number of local maxima and reduce background noise. We trained the generator model, pre-trained in a standard GAN setup for only 50 epochs, for 25 more iterations using the loss function $\mathcal{E}(1, 0, 2; PD)$ and stochastic gradient descent optimization resulting in an improved topological generator model.

3.4 Topological Layer

To incorporate topological information into the GAN, we encoded persistence diagrams of the images as fixed length vectors and concatenated them to the input of the discriminator. The discriminator then learned to exploit this information, which improved the performance of the generator. This network architecture and encoding was proposed by Gabrielsson [2].

4 Results

Our results strongly suggest that incorporating topological information in the loss function has many benefits, such as maximizing image clarity by ensuring the images have the correct number of local maxima. This process also had the added benefit of speeding up training when compared to the control suggesting that this loss function is highly favorable during training. We also observed that incorporating a topological layer, the image boosting performance seemed to be diminished suggesting that this was not beneficial in training.The topological layer posed several implementation challenges, which made its training slower compared to the topological loss function architecture. Also, the persistence diagrams were encoded by adding up the information of the individual events, which destroyed valuable information about the topology of the digits. We believe this was the primary reason why this architecture was inferior to the topological loss function.

Recently, a new topological layer based on persistent landscapes has been shown to boost efficiency for classification problems using deep neural networks. This layer has been shown to produce more positive results and the authors proved several key theorems, such as showing that this layer is resilient to both noise and outliers,suggesting that using this layer can outperform the one previously used [5]). As a future experiment, we would use this layer instead of the one previously used and see how it affects performance.

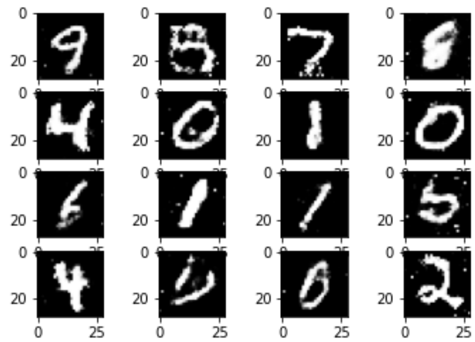

Figure 1: Standard GAN

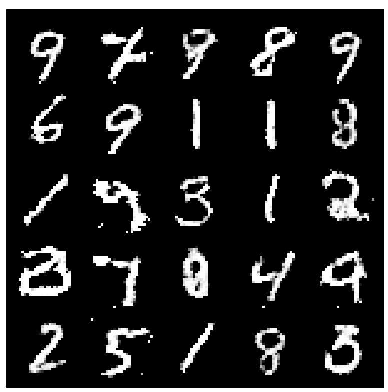

Figure 2: GAN with Topological Loss Function. Note this had the best generative performance.

Figure 3: GAN with Topological Layer

5 Conclusion

In conclusion, we demonstrated that image performance can be boosted by incorporating topological information in the form of persistence in the loss function which also had the added benefit of speeding up training. We also determined that a topological layer hurt image generating performance, but recent work suggests that this performance can be improved with the incorporation of a more modern topological layer.

Acknowledgments

We thank Dr. Sayan Mukherjee for helpful discourse and instruction throughout the course of the semester.

References

[1] Herbert Edelsbrunner and John Harer. Persistent homology-a survey. *Contemporary mathematics*, 453:257–282, 2008.

[2] Rickard Brüel Gabrielsson, Bradley J. Nelson, Anjan Dwaraknath, Primoz Skraba, Leonidas J. Guibas, and Gunnar E. Carlsson. A topology layer for machine learning. *CoRR*, abs/1905.12200, 2019.

[3] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.

[4] Yong Guo, Qi Chen, Jian Chen, Qingyao Wu, Qinfeng Shi, and Mingkui Tan. Auto-embedding generative adversarial networks for high resolution image synthesis. *CoRR*, abs/1903.11250, 2019.

[5] Kwangho Kim, Jisu Kim, Joon Sik Kim, Frederic Chazal, and Larry Wasserman. Efficient topological layer based on persistent landscapes, 2020.

[6] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010.

[7] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[8] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas. Stackgan++: Realistic image synthesis with stacked generative adversarial networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(8):1947–1962, 2019.