Mode Collapse in GANs

IT3030 Deep Learning - Bonus Challenge in Assignment 3

Lars Christian Ek Folkestad

April 2020

Abstract

GANs apply a game-theoretic approach to learning difficult manifolds such as natural images. GANs can be notoriously difficult to train in part because they tend to model only some modes of the true distribution, referred to as *mode collapse*. This document will give an overview of some the attempts to remedy this problem.

1 General Adversarial Networks (GANs)

Deep generative models learns the probability distributions of complex, high-dimensional training distributions such as natural images in an unsupervised manner. In contrast to most deep generative models like variational autoencoders (VAEs), general adversarial networks (GANs) do not define any explicit density function that can be tractably computed. Instead, GANs take a game-theoretic approach in which they sample from a simple distribution (e.g. random noise) and learn a transformation to generate the training distribution through a two-player minimax game [5].

The two players are two networks referred to as the generator network G and the discriminator network D. The generator tries to fool the discriminator by generating real-looking images, whereas the discriminator tries to distinguish between real and fake images.

2 Problems in GANs

The original GAN paper [2] states that the disadvantages of GANs primarily are that there is no explicit representation of $p_g(\boldsymbol{x})$, and that D must be synchronized well with G during training. In particular, G must not be trained too much without updating D, in order to avoid *mode collapse*. GANs suffer from mode collapse when the generator network learns how to generate samples from a few modes of the the data distribution but misses many other modes, even though samples from the missing modes occur throughout the training data. In practice, this happens when G identifies a solution seen as most likely to fool D, and begins to generate lots of samples that look just like that solution. GANs are consequently very difficult to train because this fixation stalls the learning by mapping all input noise to that same solution.

3 Mitigating Mode Collapse

As mode collapse is both one of the major problems and one of the hardest problems to solve in GANs, the topic is widely researched. Common attempts to remedy mode collapse are *Wasserstein GANs*, *Unrolled GANs* and *AdaGAN*. Although the approaches of these techniques are quite different, they all set out to force the generator to broaden its scope by preventing it from optimizing for a single fixed discriminator [3]. Other approaches such as *VEEGAN* uses implicit variational learning principles to encourage the generator to map to the entirety of the true data distribution.

3.1 Wasserstein GANs

Training GANs implemented with Wasserstein loss (WGAN) does not require maintaining a careful balance in training of the discriminator and the generator [1]. With a more informative loss function, the discriminator can in fact be trained to optimality without worrying about vanishing gradients. Mode collapse is avoided because the discriminator learns to reject the outputs that the generator stabilizes on. This forces the generator to try something new. In the original WGAN paper, the researchers reports never seeing any evidence of mode collapse for the WGANs . The WGAN algorithm does however still suffer from unstable traning, slow convergence after weight clipping and vanishing gradients. Replacing weight clipping with a gradient penalty has shown some improvement [4].

3.2 Unrolled GANs

Unrolled GANs [6] reduce mode collapse by defining the generator objective with respect to unrolled optimization of the discriminator. That is, the generator loss function incorporates not only the current discriminator's classifications, but also takes counterplay of the discriminator into account by including the outputs of future discriminator versions. Thus, the generator is prevented from over-optimizing for a single discriminator. Although effective for mitigating mode collapse, the method is resource intensive with the computational cost increasing linearly with the number of unrolling steps. This is due to a more complicated gradient calculation as well as the need for each generator update to simulate multiple discriminator updates.

3.3 AdaGAN: Boosting Generative Models

AdaGAN [8] draws inspiration from boosting techniques by training a collection of incrementally defined generators. In each iteration a new generator is added into a mixture model, and a GAN algorithm is run on a reweighed sample of the training data based on the results of the discriminator from the previous iteration. Changing the weights over time counteract mode collapse because the mixture model is forced to focus on the modes it has not been able to properly generate so far. The algorithm is able to progressively cover all modes of the true data distribution, converging either exponentially or in a finite number of steps. The drawbacks of the algorithm is, according the the authors of the AdaGAN paper, that

the corresponding latent representation no longer has a smooth structure because the generative model consists of a mixture of networks. By the very same reason, the algorithm will naturally also be more computationally expensive.

3.4 `VEEGAN`: Implicit Variational Learning

`VEEGAN`, Variational Encoder Enhancement to GANs, introduces a reconstructor network to reverse the action of the generator by mapping from the true data distribution to Gaussian random noise [7]. The generator and the discriminator is trained jointly to introduce an implicit variational principle to get an upper bound on the cross entropy between the reconstructed and original noise distribution. This further encourages the reconstructor network to approximately reverse the action of the generator. Mode collapse is mitigated because the effects of the reconstructor network in turn encourages the generator network to map from the noise distribution to the entirety of the true data distribution. The paper states that `VEEGAN` is much more effective than several state-of-the art GAN methods such as the Unrolled GAN at avoiding mode collapse while still generating good quality samples. Since an additional network is added it is likely that the algorithm is more computationally expensive than vanilla GANs.

References

[1] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein GAN, 2017.

[2] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.

[3] Google Developers. Real World GANs: Common Problems.

[4] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville. Improved Training of Wasserstein GANs, 2017.

[5] F.-F. Li, J. Johnson, and S. Yeung. CS231n Lecture 13: Generative Models, May 2017.

[6] L. Metz, B. Poole, D. Pfau, and J. Sohl-Dickstein. Unrolled Generative Adversarial Networks. *CoRR*, abs/1611.02163, 2016.

[7] A. Srivastava, L. Valkov, C. Russell, M. U. Gutmann, and C. Sutton. VEEGAN: Reducing Mode Collapse in GANs using Implicit Variational Learning, 2017.

[8] I. Tolstikhin, S. Gelly, O. Bousquet, C.-J. Simon-Gabriel, and B. Schölkopf. AdaGAN: Boosting Generative Models, 2017.