Effect of Incorrect Labels on Conditional GANs

Kumar Tanmay
kr.tanmay147@iitkgp.ac.in

Abstract

Deep generative models are becoming a cornerstone of modern machine learning. Recent work on conditional generative adversarial networks has shown that learning complex, high-dimensional distributions over natural images is within reach. However, training of GANs in challenging as they suffer from issues like non-convergence, mode-collapse, sensitivity to hyper-parameters etc. In this work, we propose a data augmentation technique to stabilise training of GANs in class-conditional setting and show that it increases robustness to destabilizing factors. We also propose a new metric using the classification ability of Discriminator in hopes of achieving a better criterion for evaluating CGANs. Once we train a CGAN on a dataset, we cast the discriminator as a classifier and use its accuracy on a test set as metric. We show that the our metric seems to exhibit a positive correlation with the quality of the generated samples.

1 Introduction

Generating realistic images is regarded as a focal task for measuring the progress of generative models. Automated metrics are either heuristic approximations [1, 2, 3, 4, 5] or intractable density estimations, examined to be inaccurate on high dimensional problems [6, 7, 8]. Human evaluations, such as those given on Amazon Mechanical Turk [2], remain ad-hoc because "results change drastically" [1] based on details of the task design [9, 10, 11]. With both noisy automated and noisy human benchmarks, measuring progress over time has become akin to hill-climbing on noise. Even widely used metrics, such as Inception Score [1] and Fréchet Inception Distance [12], have been discredited for their application to non-ImageNet datasets [13, 14]. Thus, to monitor progress, generative models need a good automated evaluation metric.

Generative Adversarial Networks were introduced as an alternative framework for training generative models in order to sidestep the difficulty of approximating many intractable probabilistic computations [15]. The intuitive idea behind GANs has been to sidestep likelihood-based learning and optimize a two-player min-max objective forcing the Generator and Discriminator to compete amongst each other to reach convergence. Thus, the training objective ensures that the generator uses the discriminative ability of discriminator to get constant feedback and keep improving. This leads us to wonder if a better discriminator implies a better generative model?

Conditional GANs are a simple extension of GANs which allow the model to learn to generate images conditioned on a given class label as well. Can the discriminative/classification ability of the discriminator of a conditional GAN be used as an evaluation criterion for CGANs? In this work, we explore whether the classification ability of the discriminator of a conditional GAN can be used as an evaluation criterion for CGANs. The intuition is to use the discriminator as a predictor of how good the generator is, and come up with a new metric which unlike metrics like Inception score and FID does not throw away the discriminator while evaluating the quality of images from the generator. To do so, after training a conditional GAN, we cast the discriminator as a classifier and measure its classification performance on a test dataset. We train different variants of conditional GANs and evaluate the performance using our metric and see how well our metric correlates with the goodness of sample quality from the generator.

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

Moreover, GAN training is highly unstable and often suffers from major problems like (a) Non-convergence (the model parameters oscillate, destabilize and never converge), (b) Mode collapse (the generator collapses which produces limited varieties of samples), (c) Diminished gradient (the discriminator gets too successful because of which the generator gradient vanishes and learns nothing), (d) Unbalance between the generator and discriminator causing overfitting, (e) Highly sensitive to the hyper-parameter selections. In this work, we also propose a data augmentation technique to stabilize training of GANs in a class-conditional setting and show that it increases robustness to destabilizing factors.

2 Related Work

2.1 Conditional Generative Adversarial Networks

Generative Adversarial Networks can be extended to a conditional model if both the generator and discriminator are conditioned on some extra information y. y could be any kind of auxiliary information, such as class labels or data from other modalities. We can perform the conditioning by feeding y into the both the discriminator and generator as additional input layer (See Figure 1a).

In the generator the prior input noise $z \sim p_z$, and y are combined in joint hidden representation, and the adversarial training framework allows for considerable flexibility in how this hidden representation is composed. Let $G_\theta(z, y)$ be the generated image.

In the discriminator x and y are presented as inputs and to a discriminative function. Let the discriminator's output be $D_\phi(x, y)$.

The objective function of the two-player minimax game would be

$$\min_\theta \max_\phi \mathcal{L}(\theta, \phi) = \mathbf{E}_{(x,y)\sim p_{data}}[\log D_\phi(x, y)] + \mathbf{E}_{z\sim p_z}[\log(1 - D_\phi(G_\theta(z, y), y))] \quad (1)$$

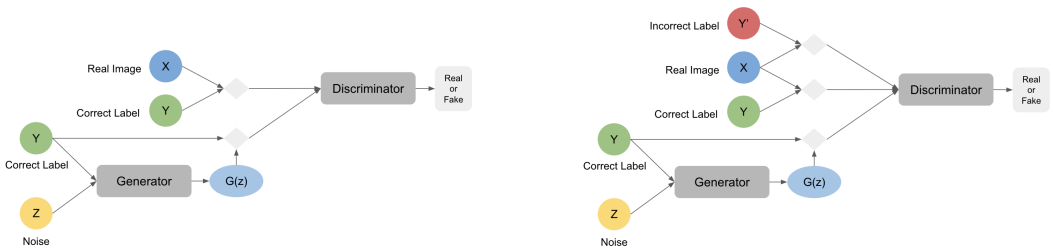
Even though conditional adversarial networks have shown great promise for interesting and useful applications, training them is known to be quite unstable [16, 15] and suffers from problems like non-convergence, mode collapse, diminished gradient, etc. In order to mitigate this, we introduce a data augmentation technique that leads to stable training of GANs in a class-conditional setting. Due to the proposed data augmentation, the training objective is modified which we describe in Section 4.1.

2.2 Evaluation of Generative Models

A lot of different metrics like Inception score [1], Frechet Inception Distance [12], and Kernel Inception Distance [17] have been proposed to evaluate generative models. Prior work has asserted that there exists coarse correlation of human judgment to FID and IS, leading to their widespread adoption. Both metrics depend on the Inception-v3 Network [54], a pretrained ImageNet model, to calculate statistics on the generated output (for IS) and on the real and generated distributions (for FID). But the validity of these metrics when applied to other datasets has been repeatedly called into question [13, 14]. Perturbations imperceptible to humans alter their values, similar to the behavior of adversarial examples [18]. FID depends on a set of real examples and a set of generated examples to compute high-level differences between the distributions, and there is inherent variance to the metric depending on the number of images and which images were chosen—in fact.

Ravuri et al. [19] proposed a metric, called Classification Accuracy Score (CAS), for conditional generative models of images, and found the metric practically useful in uncovering model deficiencies. They train a classifier over the generated samples and then evaluate the performance of this classifier over the real data. Eghbal et al. [20] proposed measures for the fitness of the generated images based on a Gaussian likelihood function from the distribution of the embeddings of the real images from the discriminator by using the activations learned in the pre-final layer.

Recently, Zhou et al. [21] created HYPE, a turnkey solution for human evaluation of generative models. Researchers can upload their model, receive a score, and compare progress via online deployment. Therefore, HYPE is a human benchmark and thus can be considered as the gold standard benchmark. [21] also computes correlations of Inception Score and FID with HYPE and show that



(a) Traditional Conditional Generative Adversarial Networks.

(b) Conditional Generative Adversarial Networks with Data Augmentation.

Figure 1: Conditional GAN training without and with the Proposed Data Augmentation technique.

there exists very weak correlation between human judgement and IS/FID. However, HYPE requires monetary investment to get the human evaluation scores. Therefore, there is a need for a robust automated metric for evaluation of generative models.

In this work, we attempt to come up with a new automated metric for evaluating conditional GANs by leveraging the classification ability of the discriminator. The training objective of CGANs ensures that the generator uses the discriminative ability of discriminator to get constant feedback and keep improving. Therefore, we posit that a better discriminator implies a better generative model and propose a new metric using the classification ability of Discriminator in hopes of achieving a better evaluation criterion.

2.3 Stabilizing Training of Generative Adversarial Networks

The recently proposed Wasserstein GAN (WGAN) [22] makes progress toward stable training of GANs, but sometimes can still generate only poor samples or fail to converge. These problems are often due to the use of weight clipping in WGAN to enforce a Lipschitz constraint on the critic, which can lead to undesired behavior. [23] propose an alternative to clipping weights: penalize the norm of gradient of the critic with respect to its input. Their method performs better than standard WGAN and enables stable training of a wide variety of GAN architectures. [24] propose a novel weight normalization technique called spectral normalization to stabilize the training of the discriminator. We observe that the above methods are still sensitive to hyper-parameter settings. Our proposed data-augmentation technique stabilizes training even under unstable hyper-parameter configuration. Our technique acts as a data-dependent regularizer and serves to assist the the existing techniques towards stable GAN training.

3 Problem Statement

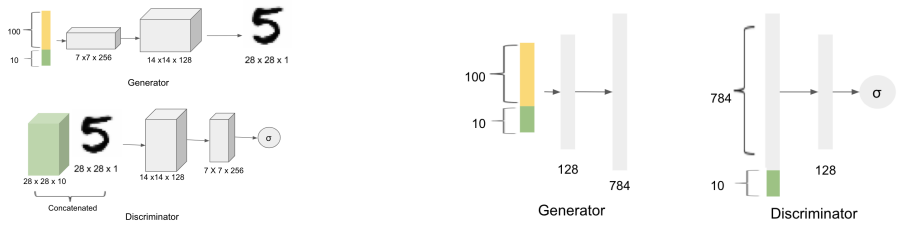
Given multiple Conditional GANs and a dataset, we aim to evaluate and compare their generative ability using the classification ability of the respective discriminators. We split the dataset into a trainset and a testset. We train the CGANs on the trainset. After training, we cast the discriminator as a classifier and measure its classification performance on the testset.

We also propose a new data augmentation technique to stabilise training of CGANs and show that it increases robustness to destabilizing factors.

3.1 Datasets

We perform our experiments using MNIST, FashionMNIST [25], and CIFAR-10 [26] datasets. Both MNIST and FashionMNIST datasets have 60K images in the trainset and 10K images in the testset. Each example in MNIST or FashionMNIST is a 28×28 grayscale image associated with a label from 10 classes.

CIFAR-10 has a train/test split of 50K/10K images. Each example in CIFAR-10 is a 32×32 RGB image associated with a label from 10 classes.



(a) Conditional cDCGAN Architecture.

(b) Conditional cFCGAN Architecture.

Figure 2: Architectures of cDCGAN and cFCGAN.

3.2 CGAN Models

For MNIST and FashionMNIST, we perform our experiments using two variants of CGANs. One is a Deep Convolutional CGAN (cDCGAN) which has convolutional layers in both the generator and discriminator and other is a CGAN with fully connected layers (cFCGAN) in the generator and discriminator. See Figure 2a and 2b for architectural details. Since cDCGAN uses convolutional layers, it is known to generate better samples [5] than cFCGAN. Therefore, we expect cDCGAN to generate better samples and get a better score on our metric than cFCGAN.

For CIFAR-10, we perform our experiments using CGANs (having three Residual Blocks in both the generator and the discriminator) with Spectral Normalization [24] and Projection Discriminator [27]. [27] proposed a novel, projection based way to incorporate the conditional information into the discriminator of GANs that respects the role of the conditional information in the underlining probabilistic model. [24] proposed a novel weight normalization technique called spectral normalization to stabilize the training of the discriminator.

3.3 Evaluation

For the purpose of evaluating our evaluation metric, we see how well our metric correlates with the goodness of sample quality from the generator of the considered CGANs. We also compare our metric with Inception Score and FID.

We also show the effect of our proposed data augmentation on CGAN training, both qualitatively and quantitatively.

4 Approach

Given multiple Conditional GANs, say $\mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3$ and so on, and a dataset $(\mathbf{X}, \mathbf{y})$, say MNIST, we aim to evaluate and compare their generative ability using the classification ability of their respective discriminators. We split the dataset $(\mathbf{X}, \mathbf{y})$ into $(\mathbf{X}_{train}, \mathbf{y}_{train})$ and $(\mathbf{X}_{test}, \mathbf{y}_{test})$. We train the CGANs on $(\mathbf{X}_{train}, \mathbf{y}_{train})$ using our proposed *data augmentation technique*. After training, we cast the discriminator as a classifier and measures its classification performance on $(\mathbf{X}_{test}, \mathbf{y}_{test})$.

Let the train data samples $\mathbf{X}_{train}, \mathbf{y}_{train}$ come from p_{data}.

4.1 Training with our Proposed Data Augmentation

In this section, we describe how we train a CGAN with the proposed data augmentation technique. As we introduce a new data augmentation technique for training CGANs, this necessitates a mathematical formulation.

In the traditional class-conditional setting, the generator takes as input a prior noise $z \sim p_z$ and a class label y to generate image $G_\theta(z, y)$. The discriminator takes as input an image x and its corresponding label y and outputs $D_\phi(x, y)$. The objective function of the two-player minimax game would be

$$\min_\theta \max_\phi \mathcal{L}(\theta, \phi) = \mathbf{E}_{(x,y)\sim p_{data}}[\log D_\phi(x, y)] + \mathbf{E}_{z\sim p_z}[\log(1 - D_\phi(G_\theta(z, y), y))] \quad (2)$$

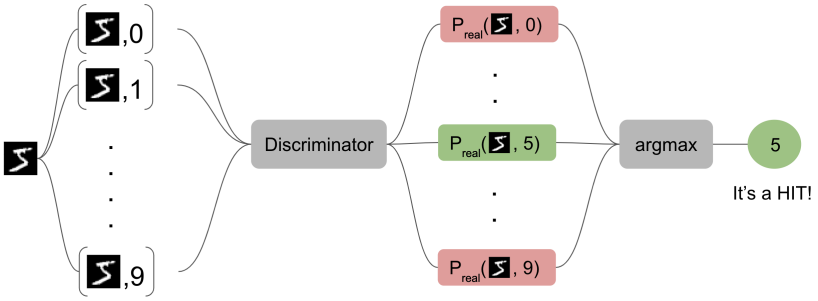


Figure 3: Everytime the highest likelihood is assigned when I is passed with the correct class, it is considered a hit, otherwise a miss. Classification performance of the discriminator = $\frac{\#hits}{\#hits+\#misses}$

Table 1: Evaluation of CGAN variants using various metrics on MNIST dataset

MNIST	cFCGAN			cDCGAN		
	IS	FID	Our Metric	IS	FID	Our Metric
w/o data aug.	**5.89**	**2.13**	0.22	2.85	14.55	0.18
w/ data aug.	5.64	2.19	**0.91**	**6.86**	**5.55**	**0.94**

Figure 1a illustrates the structure and training of a traditional CGAN. However, in our work we introduce a data augmentation technique. In addition to the discriminator receiving (real image, correct label) pairs from p_{data} and (fake image, correct label) pairs from the generator, we also train the discriminator with (real image, fake label) pairs. We do this by passing real images with incorrect labels and task the discriminator to predict it as fake during training. Let us denote these synthetically created (real image, fake label) pairs as coming from $p_{fakelabel}$. Then our mini-max training objective becomes:

$$\min_{\theta}\max_{\phi}\mathcal{L}(\theta,\phi) = \mathbf{E}_{(x,y)\sim p_{data}}[\log D_\phi(x,y)] + \mathbf{E}_{z\sim p_z}[\log(1-D_\phi(G_\theta(z,y),y))] + \mathbf{E}_{(x,y)\sim p_{fakelabel}}[\log(1-D_\phi(x,y))]$$

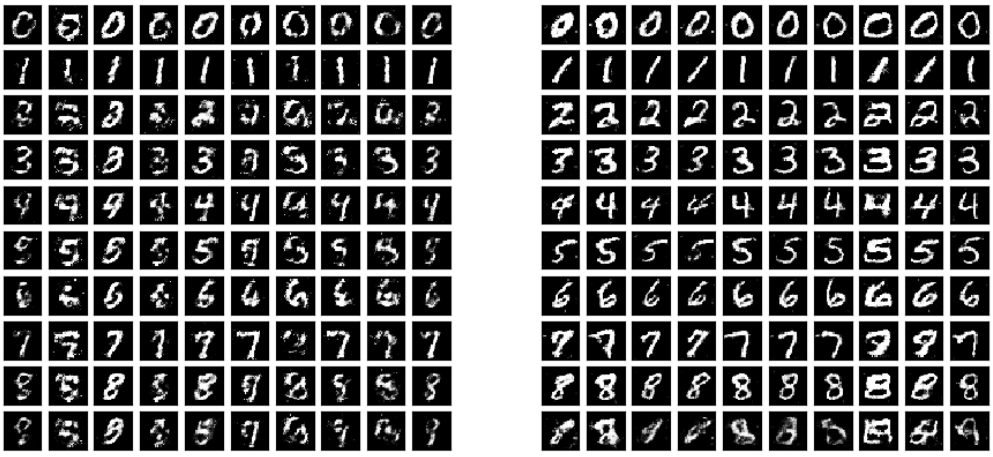
Using this augmented training objective, we train conditional GANs. Figure 1b illustrates the structure and training of a CGAN with our proposed data augmentation technique. Thus, during training, θ and ϕ are updated based on sampled mini-batches of (noise z, label) pairs, (real image, correct label) pairs, (real image, incorrect label) pairs, and the above training objective.

4.2 Proposed Metric

In this section, we define our proposed metric. Let the classes in the dataset be $\{c_1, c_2, ..., c_n\}$. After training the CGAN, an image $I \in \mathbf{X}_{test}$ is passed through the discriminator with each class c_i i.e. $\{(I,c_1),(I,c_2),...,(I,c_n)\}$, resulting in likelihoods $\{p_{real}(I,c_1), p_{real}(I,c_2), .., p_{real}(I,c_n)\}$ being produced by the discriminator. Everytime the highest likelihood is assigned when I is passed with the correct class, it is considered a hit, otherwise a miss. Let the correct class of I be c_I. If $\max(p_{real}(I,c_1), p_{real}(I,c_2), .., p_{real}(I,c_n)) == p_{real}(I,c_I)$, then it is a hit, otherwise a miss. Classification performance of the discriminator is computed this way (i.e. #hits/(#hits + #misses)) and we use this test classification accuracy as an indicator of the generative ability of CGANs. See Figure 3 for reference. A higher score on our metric implies a better generative model.

5 Experiments and Results

As mentioned earlier, we use two CGAN variants i.e. cDCGAN and cFCGAN for performing experiments on MNIST and FashionMNIST. cFCGAN consists of 1 hidden layer of size 128 in both generator and discriminator networks. The generator of cDCGAN consists of 3 transposed convolution



(a) Samples from cFCGAN

(b) Samples from cDCGAN

Figure 4: Generated image samples from cFCGAN and cDCGAN trained with data augmentation under stable hyper-parameter configuration.

Table 2: Evaluation of CGAN variants using various metrics on FashionMNIST dataset

FashionMNIST	cFCGAN			cDCGAN		
	IS	FID	Our Metric	IS	FID	Our Metric
w/o data aug.	**6.52**	**5.8**	0.19	**5.01**	32.35	0.24
w/ data aug.	6.48	6.9	**0.76**	4.89	**27.81**	**0.69**

layers followed by batch normalization layers. The discriminator consists of 3 convolution layers with leaky ReLu activation function. See Figure 2a and 2b for architectural details. For CIFAR-10, we use two CGAN variants (having three Residual blocks in both the generator and the discriminator), (a) CGAN with Projection Discriminator [27], and (b) CGAN with Spectral Normalization and Projection Discriminator [24].

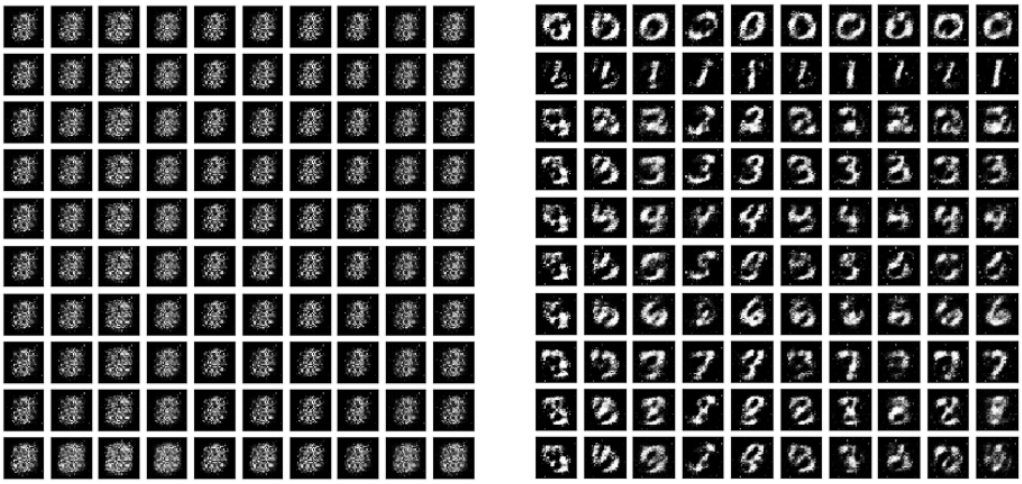
We train the models using the train set of a dataset, with and without our data augmentation. After training, we use various evaluation metrics including ours to compare the models. Tables 1, 2, 3 show the performance of various CGANs evaluated using Inception Score, FID and our metric. Unlike Inception score and our metric, a lower score on FID means a better model.

5.1 Analysis

We find that training with the proposed data augmentation imparts stability to the training of the network. We validate this by inducing instability and evaluating performance both qualitatively and

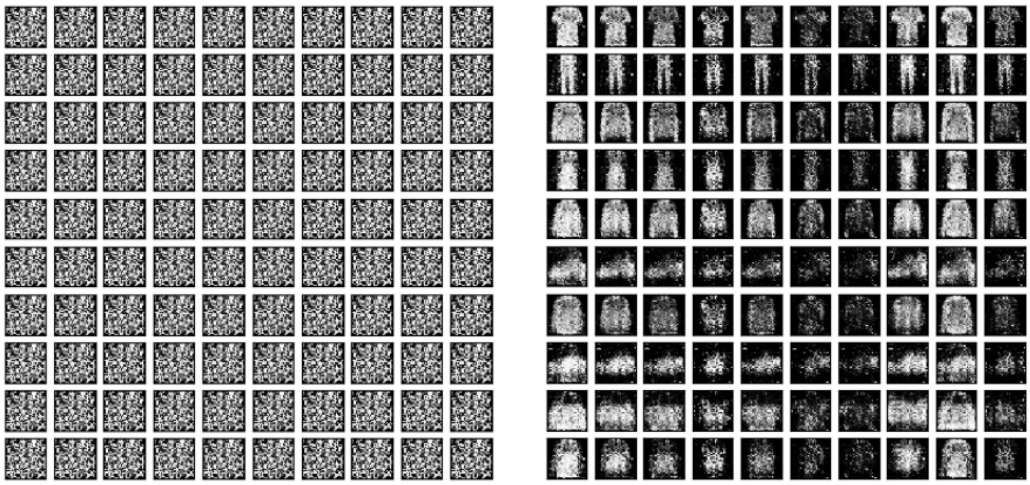
Table 3: Evaluation of CGAN variants using various metrics on CIFAR-10 dataset

CIFAR-10	CGAN w/ Spectral Norm. & Proj. Disc.			CGAN w/ Proj. Disc.		
	IS	FID	Our Metric	IS	FID	Our Metric
w/o data aug.	7.75	16.83	0.27	7.69	16.22	0.23
w/ data aug.	**8.33**	**14.78**	**0.66**	**8.25**	**14.81**	**0.65**



(a) Training without data augmentation (b) Training with the proposed data augmentation

Figure 5: Generated image samples from cFCGAN. Here, the cFCGAN is trained on MNIST with hyper-parameter configuration that tends to have a destabilizing effect.



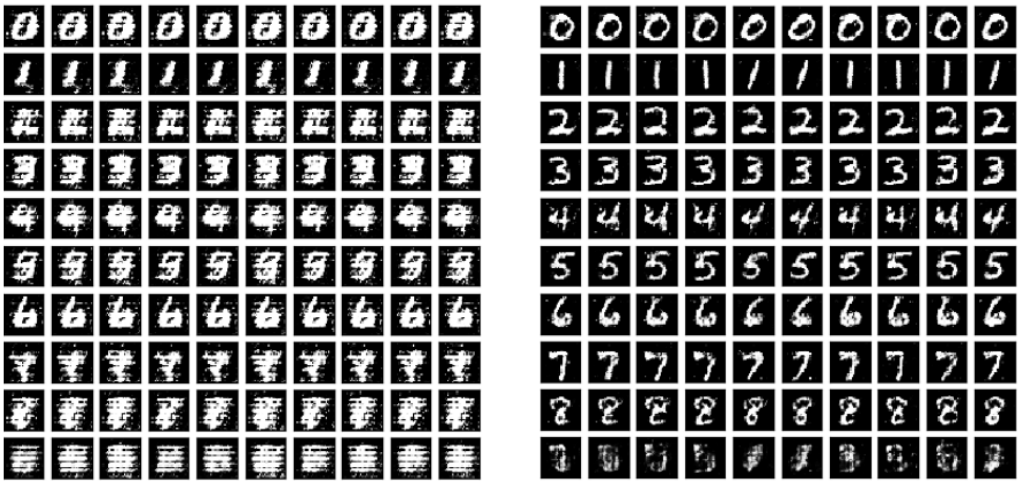
(a) Training without data augmentation (b) Training with the proposed data augmentation

Figure 6: Generated image samples from cFCGAN. Here, the cFCGAN is trained on FashionMNIST with hyper-parameter configuration that tends to have a destabilizing effect.

quantitatively. In order to induce instability, we deliberately change the hyper-parameters. We use the following methods to induce instability:

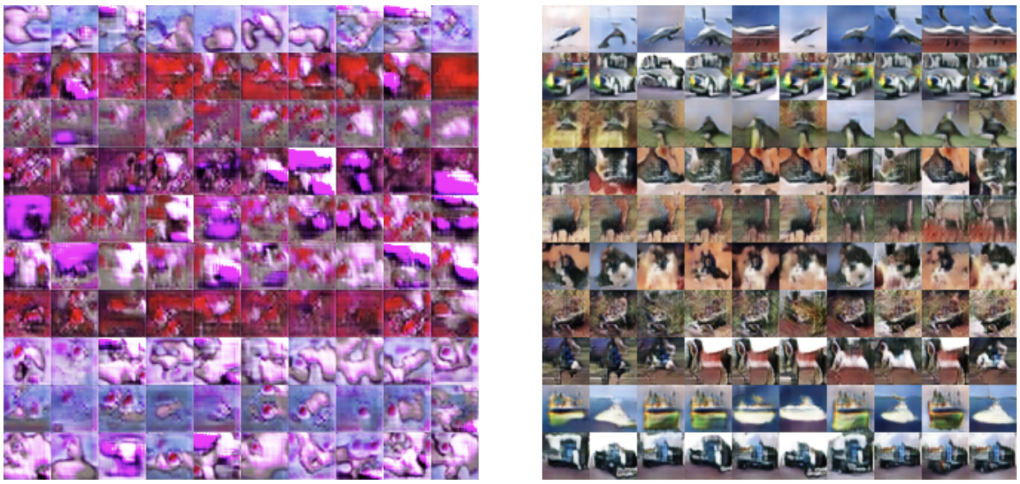
- Changed the activation function in the discriminator from leaky ReLU to ReLU.
- Changed the weight initializer in the generator from xavier to random normal.
- Changed the $beta_1$ (exponential decay rate for the 1st moment estimates) in Adam optimizer from 0.5 to 0.9.
- Changed non-saturating loss to minimax loss.

Training CGANs with the traditional approach shows nearly zero convergence due to this. But with our data augmentation, images are clearer, much less noisy. Figures 5a, 6a, 7a, 8a show the generated samples from CGAN variants when trained without data augmentation under unstable hyper-parameter configuration. It can be seen that the outputs seem to have collapsed to a noisy sample (mode collapse). However, we find that when we train the network with our proposed data



(a) Training without data augmentation (b) Training with the proposed data augmentation

Figure 7: Generated image samples from cDCGAN. Here, the cDCGAN is trained on MNIST with hyper-parameter configuration that tends to have a destabilizing effect.



(a) Training without data augmentation (b) Training with the proposed data augmentation

Figure 8: Generated image samples from CGAN w/ Spectral Normalization and Projection Discriminator. Here, the CGAN is trained with hyper-parameter configuration that tends to have a destabilizing effect.

augmentation under the same hyper-parameter configuration as above, the corresponding CGAN variants tend to converge better and produce cleaner samples. Figures 5b, 6b, 7b, 8b show the corresponding generated samples. It can be seen that the samples look reasonably close to real images even after training under hyper-parameter configuration that have a destabilizing effect.

Can we quantify this? Yes, with FID and Inception Score. We show clear improvement in FID and IS scores when cDCGAN is trained with our data augmentation (under stable hyper-parameter configuration) on MNIST and FashionMNIST (Tables 1 and 2).

However, the same is not true when cFCGAN is trained with our data augmentation on MNIST and FashionMNIST (Tables 1 and 2) under stable hyper-parameter configuration. We get better FID and Inception scores when the network is trained without our data augmentation. But the quality of samples is similar to when the network is trained with our data augmentation.

(a) Training without data augmentation

(b) Training with the proposed data augmentation

Figure 9: Generated image samples from CGAN w/ Spectral Normalization and Projection Discriminator when trained under stable hyper-parameter configuration.

Table 4: Evaluation of cFCGAN (trained on MNIST) using our proposed metric under stable and unstable hyper-parameter configurations

	cFCGAN
w/o data aug. (stable hyper-parameter config.)	0.22
w/o data aug. (unstable hyper-parameter config.)	0.10
w/ data aug. (stable hyper-parameter config.)	0.91
w/ data aug. (unstable hyper-parameter config.)	0.79

On CIFAR-10, our augmentation technique improves IS and FID scores (Table 3) to a value comparable to BigGAN [3] (FID = 14.73) which is state-of-the-art in conditional image generation. This is particularly interesting since a simple data augmentation technique on CGAN with Spectral Normalization and Projection Discriminator results in FID score comparable to that of BigGAN for conditional image generation.

What about our metric? Does a clearer set of generated images mean a higher value on our metric? Yes! We show clear increase in our metric value for better quality samples. Tables 1, 2, 3 show better scores on our metric when the network is trained with our data augmentation. This also correlates positively with the quality of generated samples. These results further verify the efficacy of our proposed evaluation metric.

On CIFAR-10, when we plug in Spectral Normalization into the CGAN (Table 3), we see consistent improvement in Inception Score, FID and our metric.

From Table 4, we further find that the score given by our metric to the cFCGAN model trained on MNIST with unstable hyper-parameter configuration is 0.79, which is lower than the score for the cFCGAN model trained with the original stable hyper-parameter configuration (0.91). This lower score is consistent with the visual quality difference between the samples shown in Figures 4a and 5b.

From Table 1, it can also be seen than cDCGAN performs better than cFCGAN on our metric (when trained on MNIST using our data augmentation). This is expected and is consistent with the fact cDCGAN is a better generative model [5] as it has convolutional layers unlike cFCGAN which has fully connected layers. See Figure 4 for visual results which corroborate the fact that cDCGAN is a better generative model than cFCGAN.

6 Challenges and Future Work

We extended our data augmentation technique to Bidirectional GANs [28]. We do this by pairing an image with encoding of an image from a different class and task the discriminator to predict this pair as fake. However, we didn't observe any improvement in the quality of the samples. We plan to look into it further. Due to computational and resource constraints, we were unable to extend our experiments to BigGAN [3] in a class-conditional setting. Motivated by the results, we have included it in our future pipeline and plan to submit this work to a conference. We also plan to compute the correlation between our metric and HYPE [21], which is a human benchmark and thus can be considered as the gold standard benchmark. This will give a quantitative grounding and validation to our metric. However, it requires monetary investment of upto 150$ for setting up the Amazon Mechanical Turk Study upon which HYPE is built. In addition to this, we also plan to try extending our data-augmentation approach to unconditional generation by replacing the human-labeled classes for each image with a soft label which could perhaps be pre-learnt beforehand. This would allow us to broaden the domain in which our technique can be useful for stable training.

7 Conclusion

In this work, we try to tackle the issue of unstable training of GANs in a class-conditional setting by proposing a new data augmentation technique. In traditional CGANs, the Discriminator receives two types of (Image, Label) pairs. It receives (Real Image, Correct Label) from p_{data} which we label as "Real Pair" and (Fake Image, Correct Label) from the generator which we label as "Fake Pair". Here, we also train the discriminator with (Real Image, Fake Label) pairs by passing real images with wrong labels and label it as "Fake Pair" during training. We show, both qualitatively and quantitatively, that the proposed data augmentation technique stabilizes training and increases robustness to destabilizing factors. We posit that our approach serves as a data-dependent regularizer towards stable GAN training in a class-conditional setting. We also attempt to devise a new automated metric for evaluation of CGANs. Existing automated metrics do not have strong correlation with human evaluation. We propose a new metric using the classification ability of Discriminator in hopes of achieving a better evaluation criterion. Given multiple Conditional GANs, we train them on a dataset. Then, we cast the discriminator as a classifier and use its accuracy on a test set as metric. We show that our metric correlates positively with visual inspection whereas Inception score and FID fail to do so in some cases.

References

[1] 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*, pages 2234–2242, 2016.

[2] Emily L Denton, Soumith Chintala, Rob Fergus, et al. Deep generative image models using a laplacian pyramid of adversarial networks. In *Advances in neural information processing systems*, pages 1486–1494, 2015.

[3] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

[4] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019.

[5] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[6] Lucas Theis, Aäron van den Oord, and Matthias Bethge. A note on the evaluation of generative models. *arXiv preprint arXiv:1511.01844*, 2015.

[7] Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

[8] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[9] John Le, Andy Edmonds, Vaughn Hester, and Lukas Biewald. Ensuring quality in crowdsourced search relevance evaluation: The effects of training question distribution. In *SIGIR 2010 workshop on crowdsourcing for search evaluation*, volume 2126, pages 22–32, 2010.

[10] Angli Liu, Stephen Soderland, Jonathan Bragg, Christopher H Lin, Xiao Ling, and Daniel S Weld. Effective crowd annotation for relation extraction. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 897–906, 2016.

[11] Aniket Kittur, Ed H Chi, and Bongwon Suh. Crowdsourcing user studies with mechanical turk. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 453–456. ACM, 2008.

[12] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, pages 6626–6637, 2017.

[13] Shane Barratt and Rishi Sharma. A note on the inception score. *arXiv preprint arXiv:1801.01973*, 2018.

[14] Ali Borji. Pros and cons of gan evaluation measures. *Computer Vision and Image Understanding*, 179:41–65, 2019.

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[16] Naveen Kodali, Jacob Abernethy, James Hays, and Zsolt Kira. On convergence and stability of gans. *arXiv preprint arXiv:1705.07215*, 2017.

[17] Mikołaj Bińkowski, Dougal J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. *arXiv preprint arXiv:1801.01401*, 2018.

[18] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016.

[19] Suman Ravuri and Oriol Vinyals. Classification accuracy score for conditional generative models. *arXiv preprint arXiv:1905.10887*, 2019.

[20] Hamid Eghbal-zadeh and Gerhard Widmer. Likelihood estimation for generative adversarial networks. *arXiv preprint arXiv:1707.07530*, 2017.

[21] Sharon Zhou, Mitchell Gordon, Ranjay Krishna, Austin Narcomey, Durim Morina, and Michael S Bernstein. Hype: Human eye perceptual evaluation of generative models. *arXiv preprint arXiv:1904.01121*, 2019.

[22] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

[23] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in neural information processing systems*, pages 5767–5777, 2017.

[24] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.

[25] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, 2017.

[26] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

[27] Takeru Miyato and Masanori Koyama. cgans with projection discriminator. *arXiv preprint arXiv:1802.05637*, 2018.

[28] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. *arXiv preprint arXiv:1605.09782*, 2016.