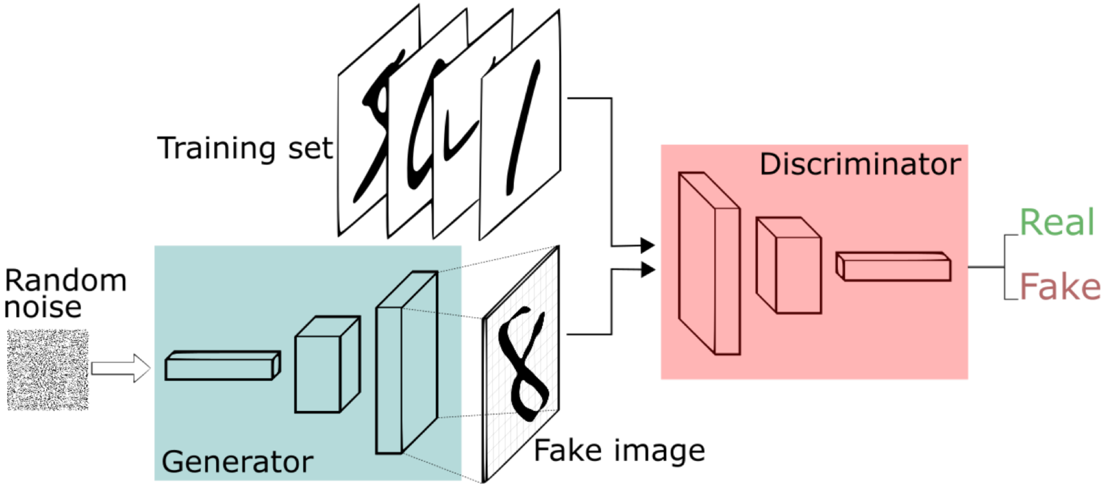
UVA CS 6316: Machine Learning : 2019 Fall
Course Project: Deep2Reproduce @
https://github.com/qiyanjun/deep2reproduce/tree/master/2019Fall

Which Training Methods for GANs do actually Converge?

Reproduced by: Zijie Pan, Kaiming Cheng

Dec 06. 2019

Motivation



Generative Adversarial Networks (GANs), proposed by Goodfellow in 2014, is a powerful latent variable model, showing dominant abilities to generate realistic image samples after training on sample data.

Problem: GANs are hard to train and gradient descent optimization results no convergence.

Main Question in this paper:

How will GAN training become locally asymptotically stable in the general case?

Claim / Target Task

Tasks in this paper

1. Proposed Dirac-GAN configurations: Prove the necessity of absolute continuity.
2. Analyze unregularized and common regularized GAN training algorithm stability on Dirac-GAN
3. Proposed simplified gradient penalties leads to convergence

Background

GANs are defined by a min-max two-player game between a discriminative network Dψ(x) and generative network Gθ(z)

Objective function:

$$\min_G \max_D \quad \left(\mathbb{E}_{x\sim p_{\text{data}}}[\log D(x)] + \mathbb{E}_{z\sim p_{\text{latent}}}[\log(1 - D(G(z)))]\right).$$

Our goal when training GANs is to find a Nash-equilibrium.

(Mescheder et al. (2017)) shown that local convergence of GAN training near an equilibrium point (θ^*, ψ^*) can be analyzed by looking at the spectrum of the Jacobian F_0 h (θ^*, ψ^*) at the equilibrium:

Eigenvalues with absolute value **bigger** than 1: <u>Not Converge</u>
Eigenvalues with absolute value **smaller** than 1: <u>Converge with linear rate $O(|\lambda_{max}|^k)$</u>
Eigenvalues are all on the **unit circle**: <u>Converge (sublinera rate), Diverge or Neither</u>

Related Work

Mescheder et al. (2017)

When λ^2 is very close to zero, it is very likely to get imagery number for λ. Thus, we can require intractably small learning rates to achieve convergence.

Sønderby et al., 2016; Arjovsky & Bottou, 2017:

Show that for common use cases of GANs, we don’t have the property of absolute continuity for the data distributions like natural images.

<u>Techniques that lead to local convergence:</u>

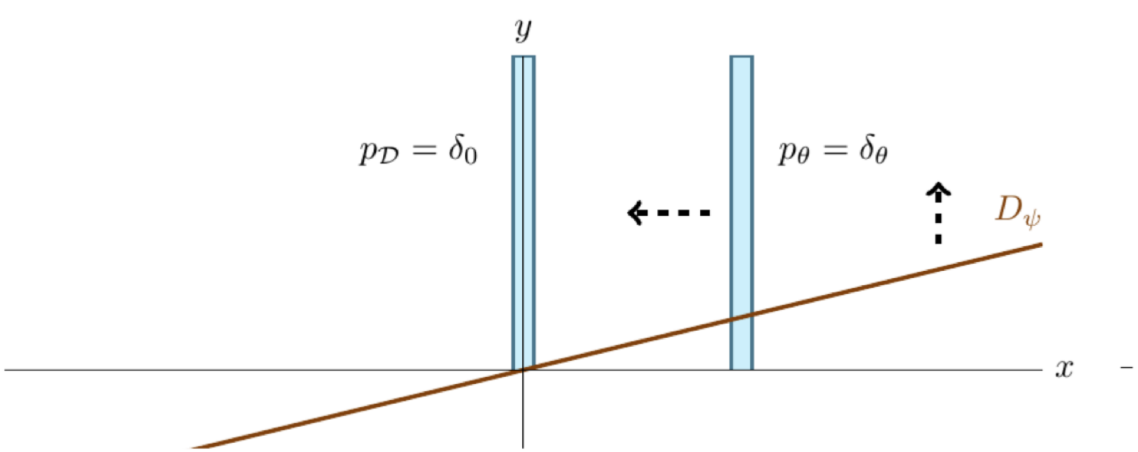
- **Arjovsky et al. (2017)**: **Propose** WGAN training
- **Sønderby et al., 2016; Arjovsky & Bottou, 2017: Propose** instance noise
- **Roth et al., 2017: Propose** zero-centered gradient penalties

...

Proposed Idea (1)

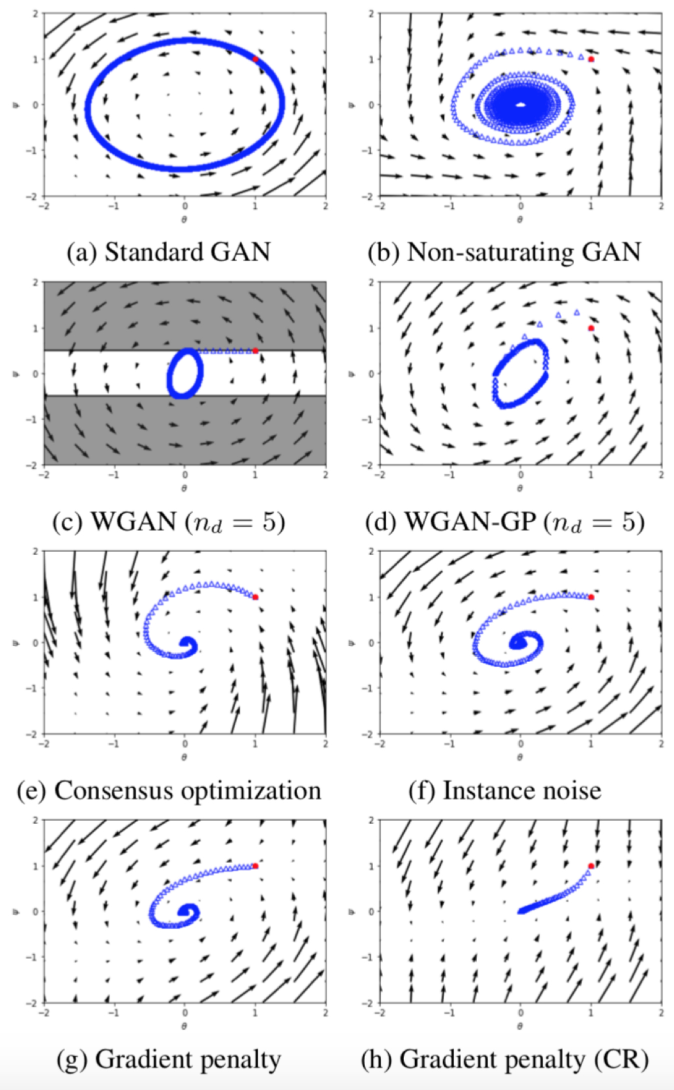
Proposed Counter-example:

- Dirac-GAN
  - Not absolute continuity → Nonconvergence
  - No optimal discriminator parameter (except 0)
  - No incentive for the discriminator to move to the equilibrium when generator is the target distribution.



Vector Field of Dirac-GAN for Different Training Algorithm

Method	Local convergence (a.c. case)	Local convergence (general case)
unregularized (Goodfellow et al., 2014)	✓	✗
WGAN (Arjovsky et al., 2017)	✗	✗
WGAN-GP (Gulrajani et al., 2017)	✗	✗
DRAGAN (Kodali et al., 2017)	✓	✗
Instance noise (Sønderby et al., 2016)	✓	✓
ConOpt (Mescheder et al., 2017)	✓	✓
Gradient penalties (Roth et al., 2017)	✓	✓
Gradient penalty on real data only	✓	✓
Gradient penalty on fake data only	✓	✓

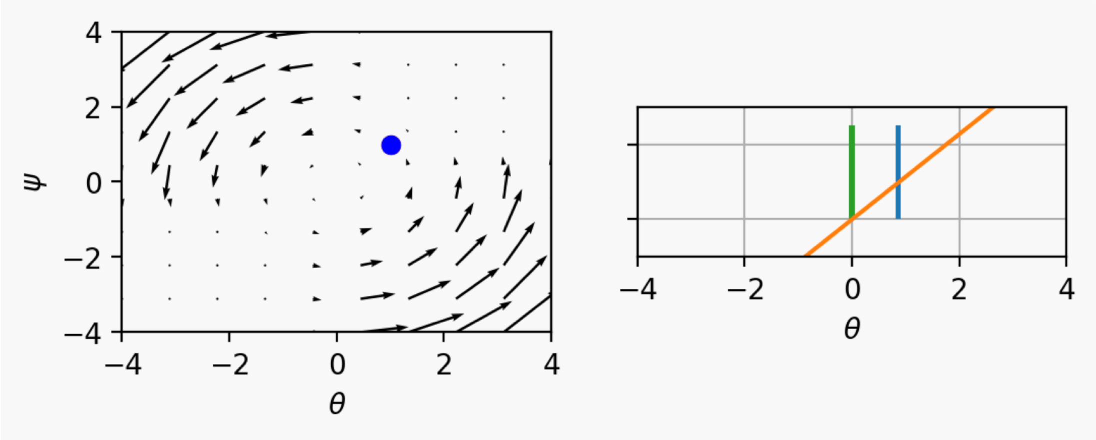


(a) Standard GAN
(b) Non-saturating GAN
(c) WGAN ($n_d = 5$)
(d) WGAN-GP ($n_d = 5$)
(e) Consensus optimization
(f) Instance noise
(g) Gradient penalty
(h) Gradient penalty (CR)

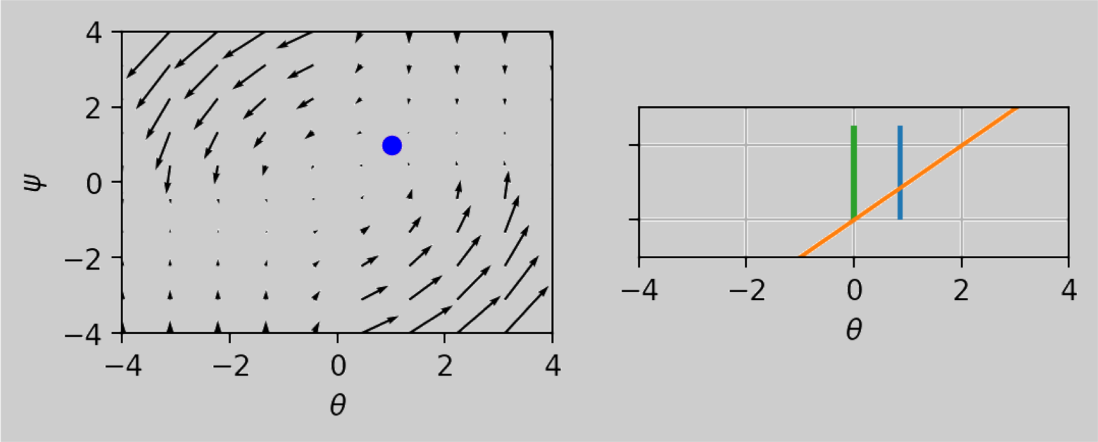
Animated Convergence Results for Unregularized GAN vs Gradient Penalty

Unregularized → Not always stable

WGAN and WGAN GP→ Not always stable



Instance noise & zero-centered & gradient penalties -> stable



Proposed Solution (1)

Inspired from zero-center gradient penalties (**Roth et al., 2017)**

Simplified regularization term:

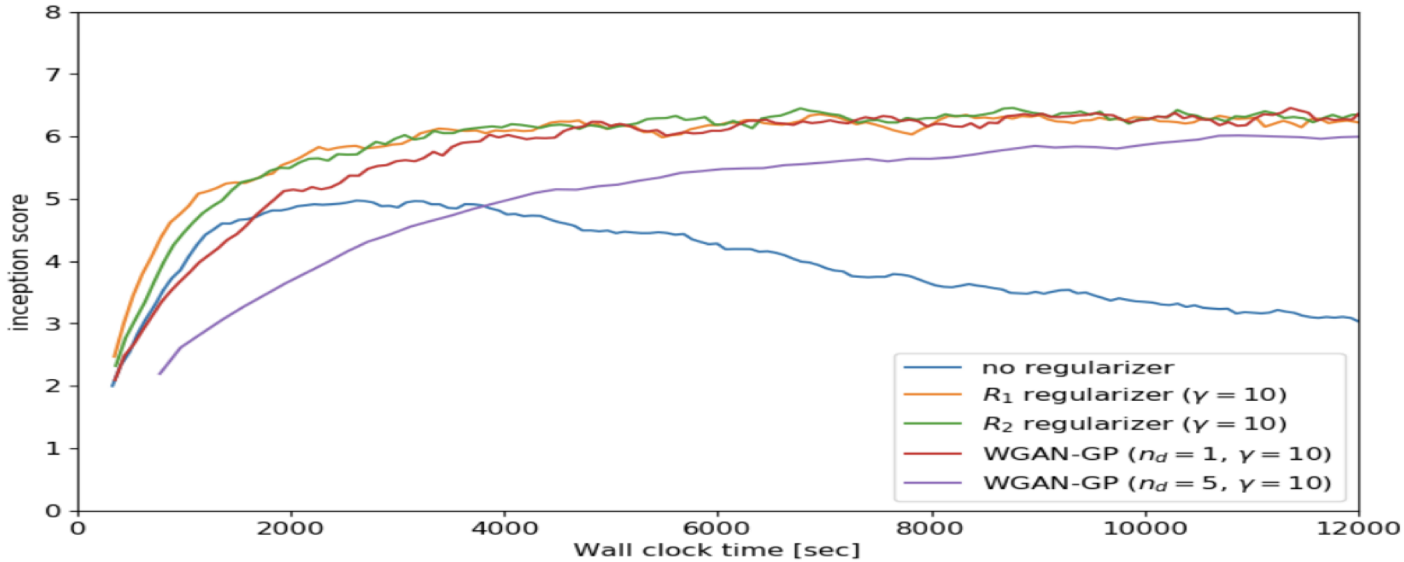
- $$R_1(\psi) := \frac{\gamma}{2} \mathrm{E}_{p_{\mathcal{D}}(x)} \left[\|\nabla D_\psi(x)\|^2 \right]$$
- $$R_2(\theta, \psi) := \frac{\gamma}{2} \mathrm{E}_{p_\theta(x)} \left[\|\nabla D_\psi(x)\|^2 \right]$$

Data Summary

There are in total three different datasets:

- 2-D Example (2D Gaussian,Line,Circle,Four Lines)
    - (not implemented)
- CIFAR-10 dataset
- CelebA-HQ dataset at resolution 1024×1024. （Karras, T., Aila, T., Laine, S., and Lehtinen, J. Progressive growing of gans for improved quality, stability, and variation. CoRR, abs/1710.10196, 2017.）

Experimental Results




```
  )
  (resnet_5_0): ResnetBlock(
    (conv_0): Conv2d(1024, 1024, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (conv_1): Conv2d(1024, 2048, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (conv_s): Conv2d(1024, 2048, kernel_size=(1, 1), stride=(1, 1), bias=False)
  )
  (fc): Linear(in_features=32768, out_features=1, bias=True)
)
https://s3.eu-central-1.amazonaws.com/avg-projects/gan_stability/models/lsun_bedroom-df4e7dd2.pt
=> Loading checkpoint from url...
Computing inception score...
/home/ec2-user/anaconda3/envs/pytorch_p36/lib/python3.6/site-packages/torch/nn/functional.py:2539: UserWarnin
g: Default upsampling behavior when mode=bilinear is changed to align_corners=False since 0.4.0. Please speci
fy align_corners=True if the old behavior is desired. See the documentation of nn.Upsample for details.
  "See the documentation of nn.Upsample for details.".format(mode))
Inception score: 2.4715 +- 0.0223
Creating samples...
100%|████████████████████████████████████████████████████████████████| 1/1 [00:02<00:00,  2.12s/it]
```

Experimental Results

Good behavior of WGAN-GP is surprising

Explanation in the next slide

```yaml
training:
  out_dir: output/default
  gan_type: standard
  reg_type: wgangp
  reg_param: 10.
  batch_size: 64
  nworkers: 16
  take_model_average: true
  model_average_beta: 0.999
  model_average_reinit: false
  monitoring: tensorboard
  sample_every: 1000
  sample_nlabels: 20
  inception_every: -1
  save_every: 900
  backup_every: 100000
  restart_every: -1
  optimizer: rmsprop
  lr_g: 0.0001
  lr_d: 0.0001
  lr_anneal: 1.
  lr_anneal_every: 150000
  d_steps: 1
  equalize_lr: false
  model_file: model.pt
test:
  batch_size: 32
  sample_size: 64
  sample_nrow: 8
  use_model_average: true
  compute_inception: true
  conditional_samples: true
  model_file: model.pt
interpolations:
  nzs: 10
  nsubsteps: 75
```

Experimental Analysis

We see that the R1- and R2-regularizers and WGAN-GP perform similarly and they achieve good results

While we find that unregularized GAN training quickly leads to mode-collapse for these problems, our simple R1-regularizer enables stable training.

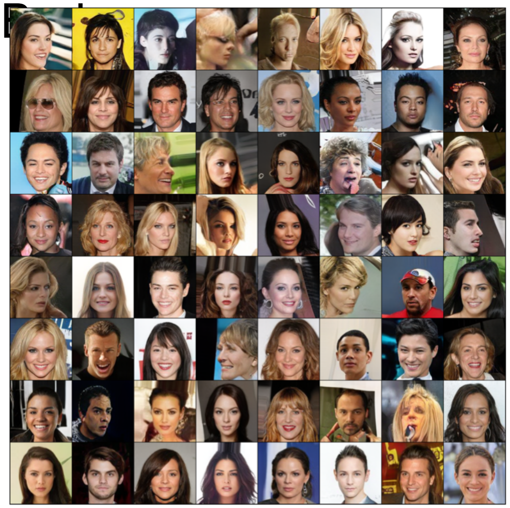
Reason:

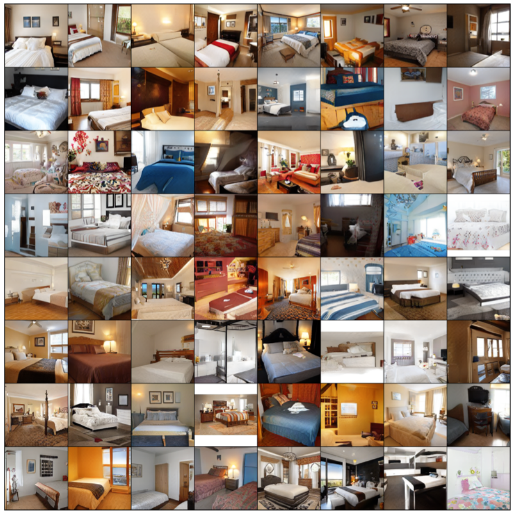
WGAN-GP oscillates in narrow circles around the equilibrium which might be enough to produce images of sufficiently high quality.

Reproduced Results

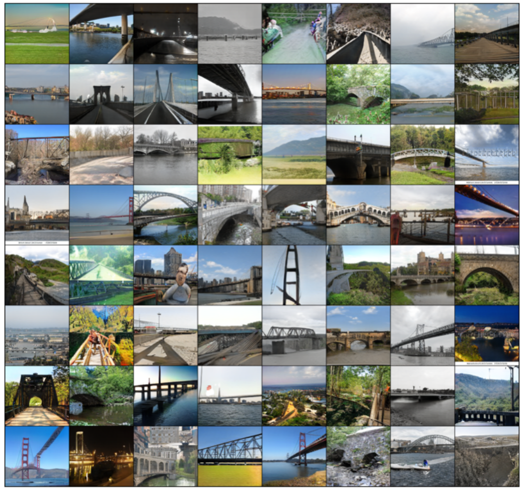
- Sample Output (Regularized) on different datasets, showing with regularizer we can have stable training:
- - Celebrate

-





- -



Reproduced Results

```
def discriminator_trainstep(self, x_real, y, z):
    toggle_grad(self.generator, False)
    toggle_grad(self.discriminator, True)
    self.generator.train()
    self.discriminator.train()
    self.d_optimizer.zero_grad()
    x_real.requires_grad_()
    d_real = self.discriminator(x_real, y)
    dloss_real = self.compute_loss(d_real, 1)

    if self.reg_type == 'real' or self.reg_type == 'real_fake':
        dloss_real.backward(retain_graph=True)
        reg = self.reg_param * compute_grad2(d_real, x_real).mean()
        reg.backward()
    else:
        dloss_real.backward()
```

4.1. Simplified gradient penalties

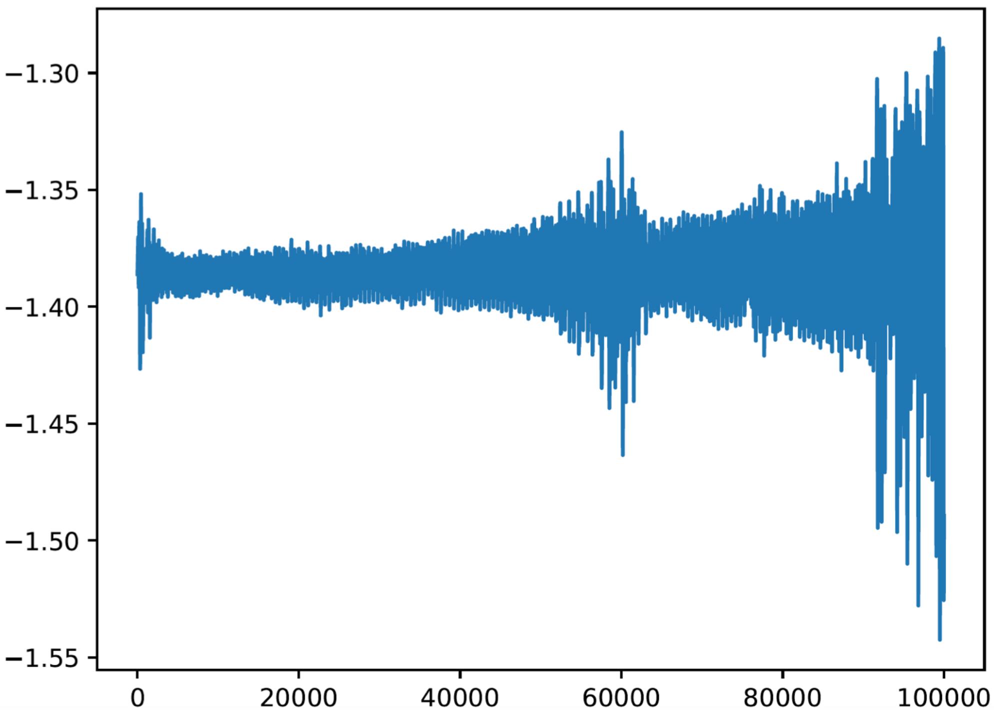
Our analysis suggests that the main effect of the zero-centered gradient penalties proposed by Roth et al. (2017) on local stability is to penalize the discriminator for deviating from the Nash-equilibrium. The simplest way to achieve this is to penalize the gradient on real data alone: when the generator distribution produces the true data distribution and the discriminator is equal to 0 on the data manifold, the gradient penalty ensures that the discriminator cannot create a non-zero gradient orthogonal to the data manifold without suffering a loss in the GAN game.

This leads to the following regularization term:

$$R_1(\psi) := \frac{\gamma}{2} \mathrm{E}_{p_{\mathcal{D}}(x)} \left[\|\nabla D_\psi(x)\|^2 \right]. \tag{9}$$

Reproduced Results

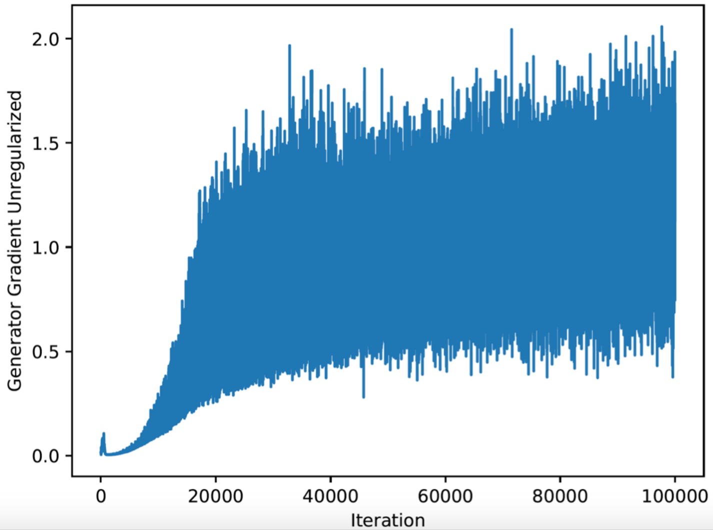
Loss with unregularized → Not Converge

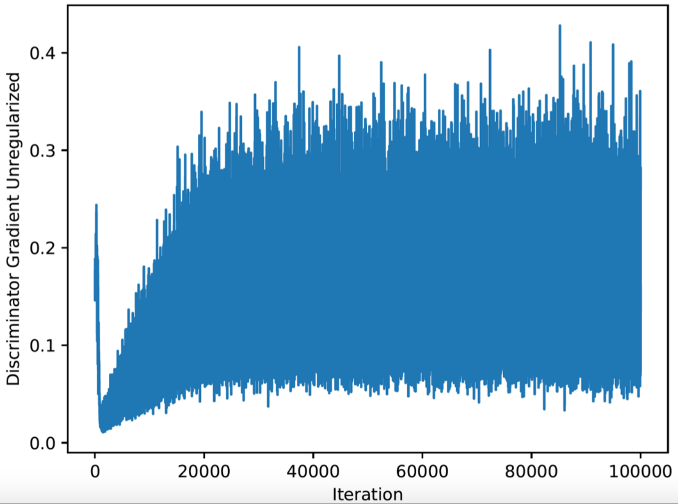



```python
params = dict(
    batch_size=512,
    disc_learning_rate=1e-4,
    gen_learning_rate=1e-4,
    beta1=0.5,
    epsilon=1e-8,
    max_iter=100001,
    viz_every=1000,
    z_dim=256,
    x_dim=2,
    unrolling_steps=0,
    regularizer_weight=0,
)
```

Reproduced Results

Unregularized → not converging!





Conclusion and Future Work

Results we have so far:

$$\lambda_{1/2} = -\frac{\gamma}{2} \pm \sqrt{\frac{\gamma^2}{4} - f'(0)^2}$$

- Negative hyperparameter: No convergence
- For eqn above: the second term has magic properties:
  - Near Nash Equilibrium, No rotation
  - Away from the Nash Equilibrium, transition from rotational convergence to non-convergence
- Convex combination of R1 and R2 have same convergence results.

Conclusion and Future Work (Cont.)

Conclusion Cont.

- Unregularized gradient based GAN optimization is not always locally convergent.
- WGANs and WGAN-GP do not always lead to local convergence whereas instance noise and zero-centered gradient penalties do.
- Local convergence achieved for simplified zero-centered gradient penalties under suitable assumptions.

Future Work

- Extend the theory to the non-realizable case (Not well understood or well-behaved to be modelled accurately)

References

- Nagarajan, V. and Kolter, J. Z. Gradient descent GAN optimization is locally stable. In Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA, pp. 5591–5600, 2017.
- Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. C., and Bengio, Y. Generative adversarial nets. In Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada, pp. 2672–2680, 2014.
- Mescheder, L. M., Nowozin, S., and Geiger, A. The numer- ics of gans. In Advances in Neural Information Processing Systems 30: Annual Conference on Neural Informa tion Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA, pp. 1823–1833, 2017
- Sønderby, C. K., Caballero, J., Theis, L., Shi, W., and Huszár, F. Amortised MAP inference for image super- resolution. CoRR, abs/1610.04490, 2016.
- Yazici, Y., Foo, C. S., Winkler, S., Yap, K., Piliouras, G., and Chandrasekhar, V. The unusual effectiveness of averaging in GAN training. CoRR, abs/1806.04498, 2018.
- Salimans, T., Goodfellow, I. J., Zaremba, W., Cheung, V., Radford, A., and Chen, X. Improved techniques for training gans. In Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain, pp. 2226–2234, 2016
- Roth, K., Lucchi, A., Nowozin, S., and Hofmann, T. Stabilizing training of generative adversarial networks through regularization. In Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA, pp. 2015–2025, 2017
- Odena, A., Olah, C., and Shlens, J. Conditional image synthesis with auxiliary classifier gans. In Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017, pp. 2642–2651, 2017.
- .... (http://proceedings.mlr.press/v80/mescheder18a/mescheder18a.pdf)

Working Split

- Kaiming Cheng: Coding, Model Training, Presentation

- Zijie Pan: Concept Research, Coding, Presentation

Thank you!