Mode Regularized Generative Adversarial Networks

Tong Che¹  Yanran Li²  Athul Paul Jacob³  Yoshua Bengio¹  Wenjie Li²

¹Montreal Institute for Learning Algorithms, Universite de Montreal, Montreal, Canada
²Department of Computing, The Hong Kong Polytechnic University, Hong Kong
³David R. Cheriton School of Computer Science, University Of Waterloo, Waterloo, Canada

ICLR, 2017
Presenter: Arshdeep Sekhon
1. Unstable gradients of Discriminator because of function shape
2. Missing Modes Problem
Motivation: Discriminator function between manifolds

1. Say data lie on a manifold
2. it is likely that there exist two distinct manifolds: one formed by real data samples and another formed by generated samples
3. training GAN is equivalent to training a characteristic function to be very close to 1 on the data manifold, and 0 on the generation manifold.
4. Desired: the trained discriminator produces stable and smooth gradients
5. since the discriminator objective does not directly depend on the behavior of the discriminator in other parts of the space, training can easily fail if the shape of the discriminator function is not as expected.
1. The generated data belongs to a few modes.
Motivation: Supervised learning targets are more stable. Why?

- the optimization target for the GAN generator is a learned discriminator.
- While in supervised models, the optimization targets are distance functions with nice geometric properties.

- The latter usually provides much easier training gradients than the former, especially at the early stages of training.
- incorporate a supervised training signal as a regularizer on top of the discriminator target
1. Generator: $G(z) : Z \rightarrow X$
2. Also jointly train an encoder $E(x) : X \rightarrow Z$
3. If $d$ is some similarity metric and $p_d$ is data generating distribution, add regularize:

$$\mathbb{E}_{x \in p_d}[d(x \odot E(x))] \quad (1)$$
1. move the generated manifold to the real data manifold using gradient descent

2. Use two signals for this:
   - gradient provided by discriminator
   - using geometric distances
Why do missing modes exist?

- For a typical GAN model, since all modes have similar D values, there is no reason why the generator cannot collapse to just a few major modes. [No penalty for missing modes]
- Fewer data samples are generated by generator using missing modes
Missing Modes Problem

1. Major Mode $M_1$
2. Minor Mode $M_2$
3. Only when $G(z)$ is very close to the mode $M_2$ can the generator get gradients to push itself towards the minor mode $M_2$
4. However, it is possible that such $z$ is of low or zero probability in the prior distribution $p_0$.

---

```
Figure 2: Illustration of missing modes problem.
```

Tong Che, Yanran Li, Athul Paul Jacob, Yoshua Bengio, Wenjie Li (Harvard University, Harvard Medical School)
Consider $M_0$ is a minor mode

1. if $G \circ E$ is a good autoencoder, then $G(E(x))$ will be very close to $M_0$

2. Add the mode regularizer:

$$
\mathbb{E}_{x \in p_d}[\log(D(G \circ E(x)))]
$$

(2)

3. to encourage $G(E(x))$ to move towards a nearby mode of the data generating distribution. New objectives:

$$
T_G = -\mathbb{E}_z[\log(D(G(z)))] + \mathbb{E}_{x \in p_d}[\lambda_1 \log(d(x, G \circ E(x))) + \lambda_2 \log(D(G \circ E(x)))]
$$

(3)

$$
T_E = \mathbb{E}_{x \in p_d}[\lambda_1 \log(d(x, G \circ E(x))) + \lambda_2 \log(D(G \circ E(x)))]
$$

(4)
But, training is very sensitive to hyperparameter tuning

Divide the training into two steps:

- **Manifold Step**: try to match the generation manifold and the real data manifold with the help of an encoder and the geometric metric loss
- **Diffusion Step**: try to distribute the probability mass on the generation manifold fairly according to the real data distribution.
Manifod Diffusion GAN example

1. **MANIFOLD:** we train a discriminator $D_1$ which separates between the samples $x$ and $G \circ E(x)$, for $x$ from the data.

2. optimize $G$ with respect to the regularized GAN loss $E[\log D_1(G \circ E(x)) + \lambda d(x, G \circ E(x))]$ in order to match the two manifolds.

3. **DIFFUSION:** train a discriminator $D_2$ between distributions $G(z)$ and $G \circ E(x)$.

4. train $G$ to maximize $\log D_2(G(z))$. 
Evaluation Metrics for Mode Missing

1. Inception score: a good assessment for sample quality from a labelled dataset

   \[ \exp(E_x KL(p(y|x)\|p(y))) \]  

2. MODE Score

   \[ \exp(E_x KL(p(y|x)\|p(y)) - KL(p(y)\|p(y))) \]  

3. According to human evaluation experiences, the MODE score is better. (variety and visual quality)
Evaluation Metrics for Mode Missing

1. In datasets with no labels
2. train a third party discriminator between the real data and the generated data from the model.
3. not used to train the generator
4. \[ D^*(s) \approx \frac{p_g(s)}{p_g(s) + p_d(s)} \] (7)
5. \( p_g \) is the probability density of the generator and \( p_d \) is the density of the data distribution.
6. To prevent \( D^*(s) \) from learning a perfect 0-1 separation: inject a zero mean gaussian noise to inputs while training \( D^* \)
7. After training, we test \( D \) on the test set \( T \) of the real dataset.
8. If for any test sample \( t \in T \), the discrimination value \( D(t) \) is close to 1, we can conclude that the mode corresponding to \( t \) is missing
Experiments: MNIST

Figure 3: The distributions of MODE scores for GAN and regularized GAN.
Table 3: Number of images on the missing modes on CelebA estimated by a third-party discriminator. The numbers in the brackets indicate the dimension of prior $z$. $\sigma$ denotes the standard deviation of the added Gaussian noise applied at the input of the discriminator to regularize it. MDGAN achieves a very high reduction in the number of missing modes, in comparison to other methods.

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>DCGAN (100)</th>
<th>DCGAN (200)</th>
<th>Reg-GAN (100)</th>
<th>Reg-GAN (200)</th>
<th>MDGAN (200)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.5</td>
<td>5463</td>
<td>17089</td>
<td>754</td>
<td>3644</td>
<td>74</td>
</tr>
<tr>
<td>4.0</td>
<td>590</td>
<td>15832</td>
<td>42</td>
<td>391</td>
<td>13</td>
</tr>
</tbody>
</table>
Qualitatively evaluation of generated samples

<table>
<thead>
<tr>
<th>Method</th>
<th>Images</th>
</tr>
</thead>
<tbody>
<tr>
<td>MDGAN</td>
<td><img src="MDGAN.png" alt="Images" /></td>
</tr>
<tr>
<td>Regularized GAN</td>
<td><img src="RegularizedGAN.png" alt="Images" /></td>
</tr>
<tr>
<td>ALI</td>
<td><img src="ALI.png" alt="Images" /></td>
</tr>
<tr>
<td>VAEGAN</td>
<td><img src="VAEGAN.png" alt="Images" /></td>
</tr>
<tr>
<td>DCGAN</td>
<td><img src="DCGAN.png" alt="Images" /></td>
</tr>
</tbody>
</table>