Introduction to GAN

Generative Adversarial Networks

Junheng (Jeff) Hao
“Adversarial Training is the coolest thing since sliced bread.”

-- Yann LeCun
Roadmap

1. Generative Modeling
2. GAN 101: What is GAN? How does it work?
3. Improvement: From GAN to W-GAN
4. GAN Applications
5. Summary
Roadmap

1. Generative Modeling
2. GAN 101: What is GAN? How does it work?
3. Improvement: W-GAN
4. GAN Applications
5. Summary
Generative vs Discriminative

**Discriminative Models**
- Learn $P(y|x)$
- Directly characterizes the decision boundary between classes only
- Examples: Logistic Regression, SVM, etc

**Generative Models**
- Learn $P(x,y)$
- Characterize how data is generated (distribution of individual class)
- Examples: Naive Bayes, HMM, etc.
What can generative model do?

- Use high-dimensional, complicated probability distributions
- Missing data, semi-supervised learning
- Multimodal outputs
- Generation tasks
- ...
Review: Maximum Likelihood

ML Method:
\[
\theta^* = \arg \max_{\theta} \mathbb{E}_{x \sim p_{\text{data}}} \log p_{\text{model}}(x \mid \theta)
\]
which is equivalent to:
\[
\theta^* = \arg \min_{\theta} D_{KL}(p_{\text{data}}(x) \mid \mid p_{\text{model}}(x; \theta))
\]
that minimizes Kullback–Leibler divergence (KL-divergence).

\[
KL(P_1 \mid \mid P_2) = \mathbb{E}_{x \sim P_1} \log \frac{P_1}{P_2}
\]
Roadmap

1. Generative Modeling

2. GAN 101: What is GAN? How does it work?

3. Improvement: W-GAN

4. GAN Applications

5. Research Frontiers

6. Summary
Story: Counterfeiter and Police

- Counterfeiter: Try to make fake money
- Police: Allow legitimate money and catch fake money

*Question:* What is the equilibrium?
Adversarial Nets Framework

Generator is trained to map a noise sample to a synthetic data sample that can "fool" the discriminator.

Discriminator is trained to distinguish real data samples from synthesized samples.
Generator Network

Characteristics:

- Must be differentiable (for training)
- Trainable for any size of $z$
- Normally $z$ has higher dimension than $x$

$$x = G(z; \theta^{(G)})$$
Minimax Game

\[
J^{(D)} = -\frac{1}{2} \mathbb{E}_{x \sim p_{data}} \log D(x) - \frac{1}{2} \mathbb{E}_z \log (1 - D(G(z)))
\]

\[
J^{(G)} = -J^{(D)} \quad \text{Resembles of Jensen-Shannon Divergence (See later)}
\]

Discriminator: Cross-entropy

Generator: minimize log-probability of \( D \) being correct

- Question: What is the solution to \( D(x) \)?
Solution: Minimax Game

Assume both densities are non-zero everywhere (keep in mind)

Solve for where the functional derivatives are zero:

\[
\frac{\delta}{\delta D(x)} J^{(D)} = 0
\]

\[
D^*(x) = \frac{P_r(x)}{P_r(x) + P_g(x)}
\]
GAN: Loss function Change

Discriminator: Cross-entropy

Generator: maximize log-probability of discriminator being mistaken

Advantage: Generator can still learn when discriminator rejects generator samples

Disadvantage: See later

\[ J^{(D)} = -\frac{1}{2} \mathbb{E}_{x \sim p_{\text{data}}} \log D(x) - \frac{1}{2} \mathbb{E}_{z} \log (1 - D(G(z))) \]

\[ J^{(G)} = -\frac{1}{2} \mathbb{E}_{z} \log D(G(z)) \]
Example: 1-D GAN
DCGAN: A better design of GAN
DCGAN: Results
Tips and Tricks of GAN

- One-side label smoothing
- Batch Normalization
- ...
1. Generative Modeling
2. GAN 101: What is GAN? How does it work?
3. Improvement: W-GAN
4. GAN Applications
5. Summary
Maximum Likelihood = Good samples?

$q^* = \arg\min_q D_{KL}(p\|q)$

$\rightarrow$

$q^* = \arg\min_q D_{KL}(q\|p)$
GAN: Problems

- Difficulty on training
- Loss of discriminator and generator can not indicate the performance of generated samples.
- Low diversity of generated samples and mode collapse.
Analysis of Loss function

Previous loss function for generator in Minimax’s Game:

\[ \mathbb{E}_{x \sim P_r} [\log D(x)] + \mathbb{E}_{x \sim P_g} [\log(1 - D(x))] \]

Then we have

\[ \mathbb{E}_{x \sim P_r} \log \frac{P_r(x)}{\frac{1}{2} [P_r(x) + P_g(x)]} + \mathbb{E}_{x \sim P_g} \log \frac{P_g(x)}{\frac{1}{2} [P_r(x) + P_g(x)]} - 2 \log 2 \]

Which is the form of Jensen-Shannon Divergence.

\[ JS(P_1||P_2) = \frac{1}{2} KL(P_1||\frac{P_1 + P_2}{2}) + \frac{1}{2} KL(P_2||\frac{P_1 + P_2}{2}) \]
Wasserstein distance (Earth-Mover Distance)

\[ W(P_r, P_g) = \inf_{\gamma \sim \Pi(P_r, P_g)} \mathbb{E}_{(x,y) \sim \gamma} [||x - y||] \]

Question: Which distance measurement can provide meaningful gradient?
Exercise 1

Given two uniform distribution $AB(P_1)$ and $CD(P_2)$, $\theta$ controls the distance of $P_1$ and $P_2$.

What is the $\text{KL}(P_1||P_2)$, $\text{JS}(P_1||P_2)$, $\text{W}(P_1||P_2)$?
Problem with \{JS, KL\} divergence

Solution:

\[
JS(P_1 || P_2) = \begin{cases} 
\log 2 & \text{if } \theta \neq 0 \\
0 & \text{if } \theta = 0
\end{cases}
\]

\[
KL(P_1 || P_2) = KL(P_1 || P_2) = \begin{cases} 
+\infty & \text{if } \theta \neq 0 \\
0 & \text{if } \theta = 0
\end{cases}
\]

\[
W(P_0, P_1) = |\theta|
\]

Note:

Only Wasserstein Distance can provide effective gradient!

JS and KL divergence cannot reflect the similarity between two probability distributions!
From Wasserstein to W-GAN

Transform the objective:

\[ W(P_r, P_g) = \inf_{\gamma \sim \Pi(P_r, P_g)} \mathbb{E}_{(x,y) \sim \gamma}[||x - y||] \]

\[ W(P_r, P_g) = \frac{1}{K} \sup_{||f||_L \leq K} \mathbb{E}_{x \sim P_r}[f(x)] - \mathbb{E}_{x \sim P_g}[f(x)] \]

\[ K \cdot W(P_r, P_g) \approx \max_{w: ||f_w||_L \leq K} [\mathbb{E}_{x \sim P_r}[f_w(x)] - \mathbb{E}_{x \sim P_g}[f_w(x)]] \]

Note: Lipschitz constant \( K \) (range restriction factor)
From Wasserstein to W-GAN

\[ L = \mathbb{E}_{x \sim P_r} [f_w(x)] - \mathbb{E}_{x \sim P_g} [f_w(x)] \]

Using MLP or CNN for \( f_\omega \) with parameter \( \omega \) inside

Generator Loss:

\[ -\mathbb{E}_{x \sim P_g} [f_w(x)] \]

Discriminator Loss:

\[ \mathbb{E}_{x \sim P_g} [f_w(x)] - \mathbb{E}_{x \sim P_r} [f_w(x)] \]
Algorithm 1 WGAN, our proposed algorithm. All experiments in the paper used the default values $\alpha = 0.00005$, $c = 0.01$, $m = 64$, $n_{\text{critic}} = 5$.

Require: : $\alpha$, the learning rate. $c$, the clipping parameter. $m$, the batch size. $n_{\text{critic}}$, the number of iterations of the critic per generator iteration.

Require: : $w_0$, initial critic parameters. $\theta_0$, initial generator’s parameters.

1: while $\theta$ has not converged do
2:     for $t = 0, \ldots, n_{\text{critic}}$ do
3:         Sample $\{x^{(i)}\}_{i=1}^m \sim \mathbb{P}_r$ a batch from the real data.
4:         Sample $\{z^{(i)}\}_{i=1}^m \sim p(z)$ a batch of prior samples.
5:         $g_w \leftarrow \nabla_w \left[ \frac{1}{m} \sum_{i=1}^m f_w(x^{(i)}) - \frac{1}{m} \sum_{i=1}^m f_w(g_\theta(z^{(i)})) \right]$
6:         $w \leftarrow w + \alpha \cdot \text{RMSProp}(w, g_w)$
7:         $w \leftarrow \text{clip}(w, -c, c)$
8:     end for
9:     Sample $\{z^{(i)}\}_{i=1}^m \sim p(z)$ a batch of prior samples.
10:    $g_\theta \leftarrow -\nabla_\theta \frac{1}{m} \sum_{i=1}^m f_w(g_\theta(z^{(i)}))$
11:    $\theta \leftarrow \theta - \alpha \cdot \text{RMSProp}(\theta, g_\theta)$
12: end while
Brief Results: WGAN
Recap: What did WGAN do?

- Define new loss function
- Remove top sigmoid-function layer
- Weight clipping in \([-c, c]\)
- Choice for gradient descent algorithms: RMSProp not Adam
Roadmap

1. Generative Modeling
2. GAN 101: What is GAN? How does it work?
3. Improvement: W-GAN
4. GAN Applications
5. Summary
Tomer Weiss: Building Information Design Synthesis (BIDS)
Tomer Weiss: BIDS: Let GAN help with your design!

- From Sketch directly to BIM
- Basic Input modules: Walls, colors and shapes
- Small dataset (200 pictures samples)
- Still need techniques of computer graphics (Filters, PointNet...)
Others

Image Editing & Video Prediction
Text Generation / Neural Dialogue Generation
Text to Image Synthesis
Drug discovery and biomarker development
Roadmap

1. Generative Modeling
2. GAN 101: What is GAN? How does it work?
3. Improvement: W-GAN
4. GAN Applications
5. **Summary**
Summary

1. GANs are generative models that use supervised learning to approximate a cost function.
2. GANs are relatively new and still require some research to reach their potential. Better theoretical understanding and training algorithms are strongly needed.
3. GANs are crucial to many different state of art image generation and enable many other applications.
Before the end: “GUN”

STOPPING GAN VIOLENCE: GENERATIVE UNADVERSARIAL NETWORKS (GUN)

Algorithm 1 Training algorithm for Generative Unadversarial Networks

1: procedure TRAIN
2:   for #iterations do
3:      Sample $n$ noise samples from prior $p_z(\tilde{z})$ and compute $G(\tilde{z}^{(1)}; \theta_g)$, ..., $G(\tilde{z}^{(n)}; \theta_g)$.  
4:      Sample $n$ data samples $\tilde{x}^{(1)}$, ..., $\tilde{x}^{(n)}$, from the data distribution.  
5:      Let $G$ show pairs $(\tilde{x}^{(i)}, G(\tilde{z}^{(i)}; \theta_g))$ to $M$ as slides of a powerpoint presentation.  
6:      Sample constructive criticism and motivational comments from $M$.  
7:      Update the powerpoint slides and incorporate suggestions into $\theta_G$.  

If you are interested, please see https://arxiv.org/pdf/1703.02528.pdf
Thanks!

Contact Me:
ScAi Research Lab
3551 Boelter Hall
University of California, Los Angeles

Email: jhao@cs.ucla.edu
Supplementary Slides

Proof: Maximum Likelihood = Minimum KL-Divergence
Supplementary Slides

Generative models: Taxonomy

- Maximum Likelihood
  - Explicit density
    - Tractable density: Nonlinear ICA, etc
    - Approximate density: Variational Autoencoder, Boltzmann Machine
  - Implicit density
    - Markov Chain
    - Direct
      - Generative Stochastic Networks
      - GAN