

Generative Adversarial Networks

Davi Barreira

FGV - Escola de Matemática Aplicada

Table of contents

1. Introdução
2. Formalização Teórica
3. Variações
4. Problemas Típicos
5. Anexo

Introdução

Generative Adversarial Networks (GAN) foram originalmente introduzidas por Goodfellow et al. (2014). Essas redes são utilizadas com o objetivo de gerar dados sintéticos realísticos a partir de dados reais.



Figure 1: Faces geradas por GANs¹.

¹Faces geradas por Karras et al. (2018)

Introdução

A geração de novas amostras sintéticas tem diferentes utilidades, como aprendizado semi-supervisionado, geração de exemplos adversariais, *style transfer*, entre outros.



Figure 2: Style transfer utilizando CycleGan ².

²<https://towardsdatascience.com/style-transfer-with-gans-on-hd-images-88e8efcf3716>

Introdução

A ideia geral por trás das GANs é utilizar duas redes neurais competindo uma com a outra, sendo uma rede responsável por gerar amostras parecidas com os dados reais (*gerador*) , enquanto a outra busca identificar quando o dado é real ou sintético (*descriminador*).

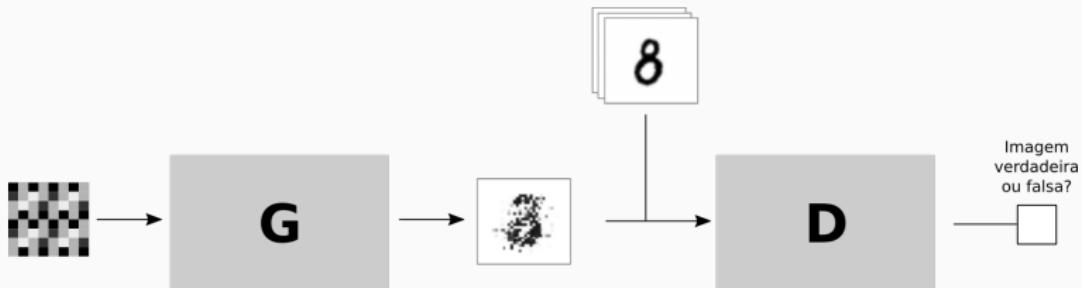


Figure 3: Desenho esquemático de uma GAN "convencional".

Formalização Teórica

Na formalização teórica da modelagem das redes adversariais, consideraremos que o gerador e o descriminador são ambos *multilayer perceptrons*. Os dados reais possuem uma distribuição $p_{data}(\mathbf{x})$, enquanto p_g é a distribuição do gerador e $p_z(z)$ é a priori do ruído de entrada. A função $G(z, \theta_g)$ é a função diferenciável que transforma \mathbf{z} no dado sintético, onde θ_g são os parâmetros da rede. A função $D(\mathbf{x}, \theta_d)$ retorna a probabilidade de \mathbf{x} ter sido amostrada de p_{data} invés de p_g .

- p_g - Distribuição dos dados sintéticos;
- p_z - Distribuição priori dos rúidos de entrada;
- p_{data} - Distribuição real dos dados;
- $G(z, \theta_g)$ - Função geradora;
- $D(\mathbf{x}, \theta_d)$ - Função discriminadora.

Formalização Teórica

Nós treinamos D buscando maximizar a capacidade de discernir dados de p_{data} de p_g . Ao mesmo tempo que treinamos G para minimizar $\log(1 - D(G(z)))$. O treino da rede se resume ao problema de otimização dado pela seguinte função objetivo:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{data}(x)} [\log(D(x))] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))]$$

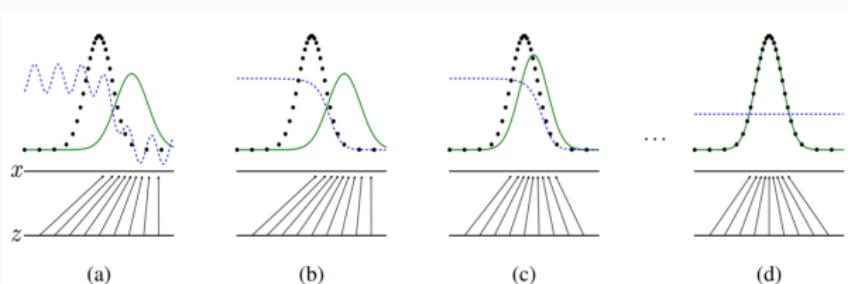


Figure 4: De (a) até (d), o desenho ilustra a evolução do algoritmo ao ser treinado. A linha azul representa a distribuição do discriminador, a linha verde representa a p_g , e os pontos pretos representam p_{data} ³.

³Imagen de Goodfellow et al. (2014)

Formalização Teórica

Algorithm 1: GAN descrita em Goodfellow et al. (2014)

for número de iterações de treino **do**

for k passos **do**

 Amostre m valores $\{z^{(1)}, \dots, z^{(m)}\}$ da priori $p_z(z)$;

 Amostre m exemplos $\{x^{(1)}, \dots, x^{(m)}\}$ da função dos dados $p_{data}(x)$;

 Atualize o *discriminator* utilizando *stochastic gradient descent*:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^m \left[\log D(x^{(i)}) + \log(1 - D(G(z^{(i)}))) \right]$$

end

 Amostre m valores $\{z^{(1)}, \dots, z^{(m)}\}$ da priori $p_z(z)$;

 Atualize o *generator* utilizando *stochastic gradient descent*:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^m \log(1 - D(G(z^{(i)})))$$

end

Formalização Teórica

Vamos estabelecer alguns resultados teóricos do funcionamento do algoritmo.

Proposição 1. Para G fixo, o discriminador D ótimo é $D_G^*(x) = \frac{p_{data}(x)}{p_{data}(x) + p_g(x)}$.

Teorema 1. O mínimo global da função objetivo é atingido se, e somente se, $p_g = p_{data}$. Neste ponto, o mínimo é $-\log 4$.

Proposição 2. Se G e D tiverem capacidade suficiente, e, em cada passo do Algoritmo 1, o discriminador atingir o seu ótimo dado G com p_g sendo atualizado para melhorar o critério

$$\mathbb{E}_{x \sim p_{data}(x)} [\log(D(x))] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))]$$

então p_g converge para p_{data} .

Formalização Teórica

Proposição 1. Para G fixo, o discriminador D ótimo é $D_G^*(x) = \frac{p_{data}(x)}{p_{data}(x) + p_g(x)}$.

Demonstração:

$$V(D, G) = \mathbb{E}_{x \sim p_{data}(x)} [\log(D(x))] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))]$$

Formalização Teórica

Proposição 1. Para G fixo, o discriminador D ótimo é $D_G^*(x) = \frac{p_{data}(x)}{p_{data}(x) + p_g(x)}$.

Demonstração:

$$\begin{aligned} V(D, G) &= \mathbb{E}_{x \sim p_{data}(x)} [\log(D(x))] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))] \\ &= \int_x p_{data}(x) \log(D(x)) dx + \int_z p_z(z) \log(1 - D(G(z))) dz \end{aligned}$$

Formalização Teórica

Proposição 1. Para G fixo, o discriminador D ótimo é $D_G^*(x) = \frac{p_{data}(x)}{p_{data}(x) + p_g(x)}$.

Demonstração:

$$V(D, G) = \mathbb{E}_{x \sim p_{data}(x)} [\log(D(x))] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))]$$

$$= \int_x p_{data}(x) \log(D(x)) dx + \int_z p_z(z) \log(1 - D(G(z))) dz$$

$$x = G(z) \implies z = G^{-1}(x) \implies dz = (G^{-1})'(x)dx$$

$$p_g(x) = p_z(G^{-1}(x))(G^{-1})'(x)dx$$

Formalização Teórica

Proposição 1. Para G fixo, o discriminador D ótimo é $D_G^*(x) = \frac{p_{data}(x)}{p_{data}(x) + p_g(x)}$.

Demonstração:

$$V(D, G) = \mathbb{E}_{x \sim p_{data}(x)} [\log(D(x))] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))]$$

$$= \int_x p_{data}(x) \log(D(x)) dx + \int_z p_z(z) \log(1 - D(G(z))) dz$$

$$x = G(z) \implies z = G^{-1}(x) \implies dz = (G^{-1})'(x) dx$$

$$p_g(x) = p_z(G^{-1}(x))(G^{-1})'(x) dx$$

$$= \int_x p_{data}(x) \log(D(x)) dx + \int_x p_z(G^{-1}(x)) \log(1 - D(x))(G^{-1})'(x) dx$$

Formalização Teórica

Proposição 1. Para G fixo, o discriminador D ótimo é $D_G^*(x) = \frac{p_{data}(x)}{p_{data}(x) + p_g(x)}$.

Demonstração:

$$V(D, G) = \mathbb{E}_{x \sim p_{data}(x)} [\log(D(x))] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))]$$

$$= \int_x p_{data}(x) \log(D(x)) dx + \int_z p_z(z) \log(1 - D(G(z))) dz$$

$$x = G(z) \implies z = G^{-1}(x) \implies dz = (G^{-1})'(x) dx$$

$$p_g(x) = p_z(G^{-1}(x))(G^{-1})'(x) dx$$

$$= \int_x p_{data}(x) \log(D(x)) dx + \int_x p_z(G^{-1}(x)) \log(1 - D(x))(G^{-1})'(x) dx$$

$$= \int_x p_{data}(x) \log(D(x)) dx + \int_x p_g(x) \log(1 - D(x)) dx$$

Formalização Teórica

Proposição 1. Para G fixo, o discriminador D ótimo é $D_G^*(x) = \frac{p_{data}(x)}{p_{data}(x) + p_g(x)}$.

Demonstração:

$$V(D, G) = \mathbb{E}_{x \sim p_{data}(x)} [\log(D(x))] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))]$$

$$= \int_x p_{data}(x) \log(D(x)) dx + \int_z p_z(z) \log(1 - D(G(z))) dz$$

$$\begin{aligned} x = G(z) &\implies z = G^{-1}(x) \implies dz = (G^{-1})'(x)dx \\ p_g(x) &= p_z(G^{-1}(x))(G^{-1})'(x)dx \end{aligned}$$

$$= \int_x p_{data}(x) \log(D(x)) dx + \int_x p_z(G^{-1}(x)) \log(1 - D(x))(G^{-1})'(x) dx$$

$$= \int_x p_{data}(x) \log(D(x)) dx + \int_x p_g(x) \log(1 - D(x)) dx$$

$$= \int_x p_{data}(x) \log(D(x)) + p_g(x) \log(1 - D(x)) dx$$

Formalização Teórica

Proposição 1. Para G fixo, o discriminador D ótimo é $D_G^*(x) = \frac{p_{data}(x)}{p_{data}(x) + p_g(x)}$.

Demonstração:

$$\max_D V(D, G) = \max_D \int_x p_{data}(x) \log(D(x)) + p_g(x) \log(1 - D(x)) dx$$

Formalização Teórica

Proposição 1. Para G fixo, o discriminador D ótimo é $D_G^*(x) = \frac{p_{data}(x)}{p_{data}(x) + p_g(x)}$.

Demonstração:

$$\max_D V(D, G) = \max_D \int_X p_{data}(x) \log(D(x)) + p_g(x) \log(1 - D(x)) dx$$

$$\frac{\partial}{\partial D(x)} (p_{data}(x) \log(D(x)) + p_g(x) \log(1 - D(x))) = 0$$

Formalização Teórica

Proposição 1. Para G fixo, o discriminador D ótimo é $D_G^*(x) = \frac{p_{data}(x)}{p_{data}(x) + p_g(x)}$.

Demonstração:

$$\max_D V(D, G) = \max_D \int_x p_{data}(x) \log(D(x)) + p_g(x) \log(1 - D(x)) dx$$

$$\frac{\partial}{\partial D(x)} (p_{data}(x) \log(D(x)) + p_g(x) \log(1 - D(x))) = 0$$

$$\implies \frac{p_{data}(x)}{D(x)} - \frac{p_g(x)}{1 - D(x)} = 0$$

$$\implies D(x) = \frac{p_{data}(x)}{p_{data}(x) + p_g(x)}$$

□

Formalização Teórica

Teorema 1. O mínimo global da função objetivo é atingido se, e somente se, $p_g = p_{data}$. Neste ponto, o mínimo é $-\log 4$.

Demonstração:

\implies) Seja $p_g = p_{data}$, $D_G^*(x) = \frac{1}{2}$. Assim,

$$V(D, G) = \mathbb{E}_{x \sim p_{data}(x)} [\log(1/2)] + \mathbb{E}_{x \sim p_g(x)} [\log(1/2)] = -\log 4$$

Formalização Teórica

Teorema 1. O mínimo global da função objetivo é atingido se, e somente se, $p_g = p_{data}$. Neste ponto, o mínimo é $-\log 4$.

Demonstração:

\Leftarrow) Seja $C(G) = \max_D V(G, D)$, assim

$$C(G) = \int_x p_{data}(x) \log(D_G^*(x)) + p_g(x) \log(1 - D_g^*(x)) dx$$

Formalização Teórica

Teorema 1. O mínimo global da função objetivo é atingido se, e somente se, $p_g = p_{data}$. Neste ponto, o mínimo é $-\log 4$.

Demonstração:

\Leftarrow) Seja $C(G) = \max_D V(G, D)$, assim

$$\begin{aligned} C(G) &= \int_x p_{data}(x) \log(D_G^*(x)) + p_g(x) \log(1 - D_g^*(x)) dx \\ &= \int_x p_{data}(x) \log\left(\frac{p_{data}(x)}{p_{data}(x) + p_g(x)}\right) + p_g(x) \log\left(\frac{p_g(x)}{p_{data}(x) + p_g(x)}\right) dx \end{aligned}$$

Formalização Teórica

Teorema 1. O mínimo global da função objetivo é atingido se, e somente se, $p_g = p_{data}$. Neste ponto, o mínimo é $-\log 4$.

Demonstração:

\Leftarrow) Seja $C(G) = \max_D V(G, D)$, assim

$$\begin{aligned} C(G) &= \int_x p_{data}(x) \log(D_G^*(x)) + p_g(x) \log(1 - D_g^*(x)) dx \\ &= \int_x p_{data}(x) \log\left(\frac{p_{data}(x)}{p_{data}(x) + p_g(x)}\right) + p_g(x) \log\left(\frac{p_g(x)}{p_{data}(x) + p_g(x)}\right) dx \\ &= \int_x p_{data}(x) \log\left(2^{-1} \cdot \frac{p_{data}(x)}{\frac{p_{data}(x) + p_g(x)}{2}}\right) + p_g(x) \log\left(2^{-1} \cdot \frac{p_g(x)}{\frac{p_{data}(x) + p_g(x)}{2}}\right) dx \end{aligned}$$

Formalização Teórica

Teorema 1. O mínimo global da função objetivo é atingido se, e somente se, $p_g = p_{data}$. Neste ponto, o mínimo é $-\log 4$.

Demonstração:

\Leftarrow) Seja $C(G) = \max_D V(G, D)$, assim

$$\begin{aligned} C(G) &= \int_x p_{data}(x) \log(D_G^*(x)) + p_g(x) \log(1 - D_g^*(x)) dx \\ &= \int_x p_{data}(x) \log\left(\frac{p_{data}(x)}{p_{data}(x) + p_g(x)}\right) + p_g(x) \log\left(\frac{p_g(x)}{p_{data}(x) + p_g(x)}\right) dx \\ &= \int_x p_{data}(x) \log\left(2^{-1} \cdot \frac{p_{data}(x)}{\frac{p_{data}(x) + p_g(x)}{2}}\right) + p_g(x) \log\left(2^{-1} \cdot \frac{p_g(x)}{\frac{p_{data}(x) + p_g(x)}{2}}\right) dx \\ &= \mathbb{E}_{x \sim p_{data}(x)} \left[-\log(2) + \frac{p_{data}(x)}{p_{data}(x) + p_g(x)} \right] + \mathbb{E}_{x \sim p_g(x)} \left[-\log(2) + \frac{p_g(x)}{p_{data}(x) + p_g(x)} \right] \end{aligned}$$

Formalização Teórica

Teorema 1. O mínimo global da função objetivo é atingido se, e somente se, $p_g = p_{data}$. Neste ponto, o mínimo é $-\log 4$.

Demonstração:

$$\begin{aligned} C(G) &= KL \left[p_{data}(x) \parallel \frac{p_{data}(x) + p_g(x)}{2} \right] + KL \left[p_g(x) \parallel \frac{p_{data}(x) + p_g(x)}{2} \right] - \log 4 \\ &= 2 \cdot JSD [p_{data} \parallel p_g] - \log 4 \end{aligned}$$

Onde KL é a distância Kullback-Leibler e JSD é a divergência de Jensen-Shannon. Assim:

$$\min_G C(G) = \min_G (2 \cdot JSD [p_{data} \parallel p_g] - \log 4)$$

O mínimo da divergência JSD é zero e só é atingido se, e somente se, $p_g = p_{data}$.⁴

⁴Estamos assumindo que o modelo gerativo é capaz de reproduzir perfeitamente a distribuição dos dados

□

Formalização Teórica

Proposição 2. Se G e D tiverem capacidade suficiente, e, em cada passo do Algoritmo 1, o discriminador atingir o seu ótimo dado G com p_g sendo atualizado para melhorar o critério

$$\mathbb{E}_{x \sim p_{data}(x)} [\log(D(x))] + \mathbb{E}_{x \sim p_g(x)} [1 - \log(D(x))]$$

então p_g converge para p_{data} .

Demonstração:

Considere $V(G, D) = U(p_g, D)$. Assim, para um D fixo, U é função de p_g . Note que $U(p_g, D)$ é convexo, pois

$$U(p_g, D) = \mathbb{E}_{x \sim p_{data}(x)} [\log(D(x))] + \mathbb{E}_{x \sim p_g(x)} [1 - \log(D(x))] \therefore$$

Formalização Teórica

Proposição 2. Se G e D tiverem capacidade suficiente, e, em cada passo do Algoritmo 1, o discriminador atingir o seu ótimo dado G com p_g sendo atualizado para melhorar o critério

$$\mathbb{E}_{x \sim p_{data}(x)} [\log(D(x))] + \mathbb{E}_{x \sim p_g(x)} [1 - \log(D(x))]$$

então p_g converge para p_{data} .

Demonstração:

Considere $V(G, D) = U(p_g, D)$. Assim, para um D fixo, U é função de p_g . Note que $U(p_g, D)$ é convexo, pois

$$U(p_g, D) = \mathbb{E}_{x \sim p_{data}(x)} [\log(D(x))] + \mathbb{E}_{x \sim p_g(x)} [1 - \log(D(x))] \therefore$$

$$\begin{aligned} U(\alpha p + (1 - \alpha)q, D) &= \int_x \alpha \cdot p(x) \log(D(x)) + (1 - \alpha)q(x) \log(1 - D(x)) dx \\ &= \alpha \mathbb{E}_{x \sim p(x)} [\log(D(x))] + (1 - \alpha) \mathbb{E}_{x \sim q(x)} [1 - \log(D(x))] \\ &= \alpha U(p, D) + (1 - \alpha)U(q, D) \end{aligned}$$

Formalização Teórica

Proposição 2. Se G e D tiverem capacidade suficiente, e, em cada passo do Algoritmo 1, o discriminador atingir o seu ótimo dado G com p_g sendo atualizado para melhorar o critério

$$\mathbb{E}_{x \sim p_{data}(x)} [\log(D(x))] + \mathbb{E}_{x \sim p_g(x)} [1 - \log(D(x))]$$

então p_g converge para p_{data} .

Demonstração:

Como $U(p_g, D)$ é um função convexa, podemos utilizar um algoritmo de descida de gradiente para atingir o seu mínimo no ponto onde esse gradiente é igual a zero, e que como provado no **Teorema 1**, é um mínimo global. \square

Formalização Teórica

Proposição 2. Se G e D tiverem capacidade suficiente, e, em cada passo do Algoritmo 1, o discriminador atingir o seu ótimo dado G com p_g sendo atualizado para melhorar o critério

$$\mathbb{E}_{x \sim p_{data}(x)} [\log(D(x))] + \mathbb{E}_{x \sim p_g(x)} [1 - \log(D(x))]$$

então p_g converge para p_{data} .

Demonstração:

Como $U(p_g, D)$ é um função convexa, podemos utilizar um algoritmo de descida de gradiente para atingir o seu mínimo no ponto onde esse gradiente é igual a zero, e que como provado no **Teorema 1**, é um mínimo global. \square

Na prática, a GAN otimiza os parâmetros θ_g invés de p_g , então a prova não se aplica, já que o *multilayer perceptron* aproxima um subconjunto da família de p_g .

Variações

Além do modelo tradicional apresentado, variações de GANs tem surgido para diferentes aplicações.

- **DCGAN:** Uso de Convolutional Neural Networks em GANs para melhorar o processo de geração de dados sintéticos.
- **Face Inpainting:** Utilização de GANs para "reconstrução" de imagens com partes faltantes.



Figure 5: Inpainting usando GAN (Yeh et al., 2017).

- **Transferência Imagem-Imagem:** Utilização de GANs para transformar um grupo de imagens em outro, como no exemplo de transferência de estilo.

Variações

A arquitetura **CycleGAN** (Zhu et al., 2017) é uma das mais utilizadas para transferências imagem-imagem.

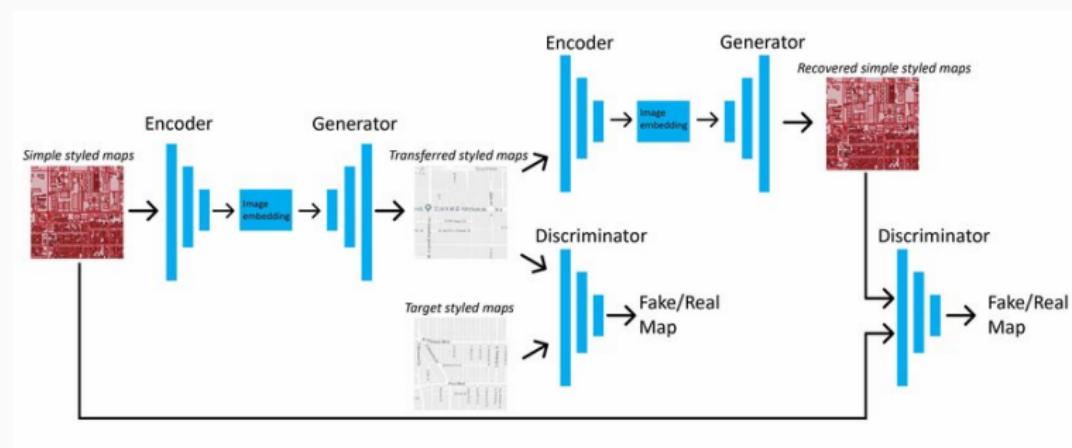


Figure 6: Arquitetura de uma CycleGAN (Kang et al., 2019).

Problemas Típicos

GANs tradicionais possuem falhas comuns que tem sido foco de pesquisas. Arjovsky and Bottou (2017) clarificaram a fonte desses problemas e propuseram algumas soluções.

- **Dissipação de gradientes:** Na prática, ao treinar o descriminador até seu estado ótimo ($\frac{P_{data}}{P_{data} + P_g}$), os gradientes tendem a se dissipar no treinamento do gerador

$$\lim_{||D - D^*|| \rightarrow 0} \nabla_{\theta} \mathbb{E}_{z \sim p(z)} [\log (1 - D(G_{\theta}(z)))] = 0$$

- **“Mode Colapse”:** Normalmente, você deseja que sua GAN produza uma ampla variedade de saídas. No entanto, se um gerador produz uma saída especialmente plausível, ele pode aprender a produzir apenas essa saída.
- **Falha de Convergência:** GANs podem apresentar instabilidade na atualização dos gradientes do gerador, levando a falhas de convergência.

Problemas Típicos

Soluções propostas para esses problemas são:

- **Dissipação de gradientes:**

1. Alterar a função objetivo do gerador para
 $C(G) = \max \log(D^*(G(z)))$ (Goodfellow et al., 2014).
2. WGAN - Utilizar métrica Wasserstein, definida abaixo, como função objetivo (Arjovsky et al., 2017).

- **“Mode Colapse”:**

1. WGAN.

- **Falha de Convergência:**

1. Adicionar ruído nas entradas do discriminador.
2. Penalizar pesos do Discriminador.

References i

- Arjovsky, M. and Bottou, L. (2017). Towards principled methods for training generative adversarial networks.
- Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein gan.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc.
- Kang, Y., Gao, S., and Roth, R. (2019). Transferring multiscale map styles using generative adversarial networks. *International Journal of Cartography*.

References ii

- Karras, T., Laine, S., and Aila, T. (2018). A style-based generator architecture for generative adversarial networks. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4396–4405.
- Yeh, R. A., Chen, C., Yian Lim, T., Schwing, A. G., Hasegawa-Johnson, M., and Do, M. N. (2017). Semantic image inpainting with deep generative models. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5485–5493.
- Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2242–2251.

Neste anexo apresentamos mais informações sobre o problema de dissipação de gradientes e a solução utilizando WGAN.

Solução para Dissipação de Gradientes:

(i) Alterar a função objetivo do gerador para $C(G) = \max - \log(D^*(G(z)))$.

Dessa forma foi provado que

$$\mathbb{E}_{z \sim p(z)} [-\nabla_{\theta} \log D^*(G_{\theta}(z))|_{\theta=\theta_0}] = \nabla_{\theta} [KL(p_{g_{\theta}} || p_{data}) - 2JSD(p_{g_{\theta}} || p_{data})]|_{\theta=\theta_0}$$

Note que:

- JSD's estão com sinais opostos. Significa que estão pressionando para que as distribuições sejam diferentes, o que parece ser uma falha na atualização.
- $KL(p_{g_{\theta}} || p_{data})$ não é equivalente ao da máxima verossimilhança.

Isso explica porque os GANs (quando estabilizados) criam amostras de boa aparência e justifica porque os GANs sofrem com uma quantidade grande de "Mode Dropping".

Anexo

Solução para Dissipação de Gradientes:

(ii) (WGAN) - Utilizar métrica Wasserstein, definida abaixo, como função objetivo.

$$W(p_{data}, p_g) = \inf_{\gamma \in \Gamma(p_{data}, p_g)} \mathbb{E}_{(x,y) \sim \gamma} [|x - y|]$$

Anexo

Solução para Dissipação de Gradientes:

(ii) (WGAN) - Utilizar métrica Wasserstein, definida abaixo, como função objetivo.

$$W(p_{data}, p_g) = \inf_{\gamma \in \Gamma(p_{data}, p_g)} \mathbb{E}_{(x,y) \sim \gamma} [|x - y|]$$

Porém, implementar essa definição é impraticável. Em seu lugar, através da dualidade Kartrovich - Rubinstein, podemos aproxima-la por

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

Solução para Dissipação de Gradientes:

(ii) (WGAN) - Utilizar métrica Wasserstein, definida abaixo, como função objetivo.

$$W(p_{data}, p_g) = \inf_{\gamma \in \Gamma(p_{data}, p_g)} \mathbb{E}_{(x,y) \sim \gamma} [|x - y|]$$

Porém, implementar essa definição é impraticável. Em seu lugar, através da dualidade Kartrovich - Rubinstein, podemos aproxima-la por

$$W(p_{data}, p_g) = \sup_{\|f\|_L \leq 1} \frac{1}{K} (\mathbb{E}_x \ p_{data}[f(x)] - \mathbb{E}_x \ p_g[f(x)])$$

Suponha que f pertença à família de funções K-Lipschitz $\{f_w\}_{w \in W}$. Temos então

$$W(p_{data}, p_g) = \max_{w \in W} \mathbb{E}_x \ p_{data}[f_w(x)] - \mathbb{E}_z \ p_z[f_w(G(z))], \text{ tal que } \|f_w\|_L \leq K$$

Solução para Dissipação de Gradientes:

(ii) (WGAN) - Utilizar métrica Wasserstein, definida abaixo, como função objetivo.

$$W(p_{data}, p_g) = \inf_{\gamma \in \Gamma(p_{data}, p_g)} \mathbb{E}_{(x,y) \sim \gamma} [|x - y|]$$

Porém, implementar essa definição é impraticável. Em seu lugar, através da dualidade Kartrovich - Rubinstein, podemos aproxima-la por

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

Suponha que f pertença à família de funções K-Lipschitz $\{f_w\}_{w \in W}$. Temos então

$$W(p_{data}, p_g) = \max_{w \in W} \mathbb{E}_{x \sim p_{data}} [f_w(x)] - \mathbb{E}_{z \sim p_z} [f_w(G(z))], \text{ tal que } \|f_w\|_L \leq K$$

O Discriminador é usado para aprender w e encontrar f_w . A função W então se torna.

$$W(p_{data}, p_g) = \max_{w \in W} \mathbb{E}_{x \sim p_{data}} [D_w(x)] - \mathbb{E}_{z \sim p_z} [D_w(G(z))], \text{ tal que } \|D_w\|_L \leq K$$

Anexo

Dessa forma, D deixa de ser um Discriminador (com saída [0,1]) que buscava descobrir quais amostras são falsas ou verdadeiras e se transforma em um Crítico (com saída ≥ 0), treinado para aprender funções contínuas K-Lipschitz que ajudarão a computar a métrica Wasserstein.

A função objetivo da nova GAN (WGAN) passa a ser então.

$$V(G, D) = \min_{\theta} \max_{w \in W} \mathbb{E}_{x \sim p_{data}} [D_w(x)] - \mathbb{E}_{z \sim p_z} [D_w(G_{\theta}(z))], \text{ tal que } \|D_w\|_L \leq K$$

Uma vez que a métrica Wasserstein tem um comportamento suave ela elimina o problema da dissipação do gradiente.