extravaGAN

Sabina Askerova, Yannis Fontaine, Newman Chen

WGAN (GP)

Pseudocode:

while θ has not converged **do**
  for $t = 0, ..., n_{\text{critic}}$ **do**
    Sample $\{x^{(i)}\}_{i=1}^{m} \sim \mathbb{P}_r$ a batch from the real data.
    Sample $\{z^{(i)}\}_{i=1}^{m} \sim p(z)$ a batch of prior samples.
    $g_w \leftarrow \nabla_w \left[\frac{1}{m}\sum_{i=1}^{m} f_w(x^{(i)}) - \frac{1}{m}\sum_{i=1}^{m} f_w(g_\theta(z^{(i)}))\right] \longrightarrow L^{(i)} \leftarrow D_w(\tilde{\boldsymbol{x}}) - D_w(\boldsymbol{x}) + \lambda(\|\nabla_{\hat{\boldsymbol{x}}} D_w(\hat{\boldsymbol{x}})\|_2 - 1)^2$
    $w \leftarrow w + \alpha \cdot \text{RMSProp}(w, g_w)$
    $w \leftarrow \text{clip}(w, -c, c)$
  end for
  Sample $\{z^{(i)}\}_{i=1}^{m} \sim p(z)$ a batch of prior samples.
  $g_\theta \leftarrow -\nabla_\theta \frac{1}{m}\sum_{i=1}^{m} f_w(g_\theta(z^{(i)}))$
  $\theta \leftarrow \theta - \alpha \cdot \text{RMSProp}(\theta, g_\theta)$

Wasserstein / Earth-moving distance:

$$W(\mathbb{P}_r, \mathbb{P}_g) = \inf_{\gamma \in \Pi(\mathbb{P}_r, \mathbb{P}_g)} \mathbb{E}_{(x,y)\sim\gamma}\left[\, \|x - y\| \,\right]$$

alpha (Learning rate) = 0.0004

lambda (gradient penalty coefficient) = 10

n_critic (ratio of training loops of discriminator to generator = 5

FID: 42.21, Precision: 0.54, Recall: 0.25.

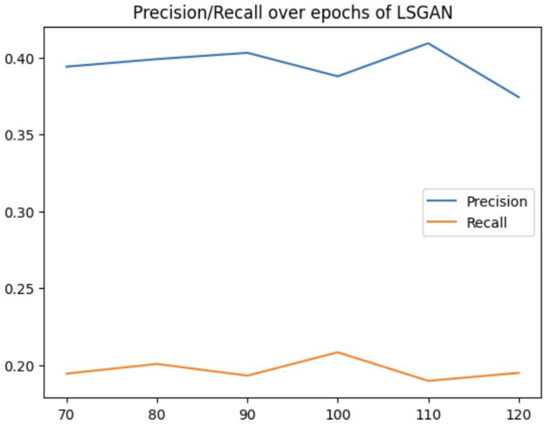
LSGAN

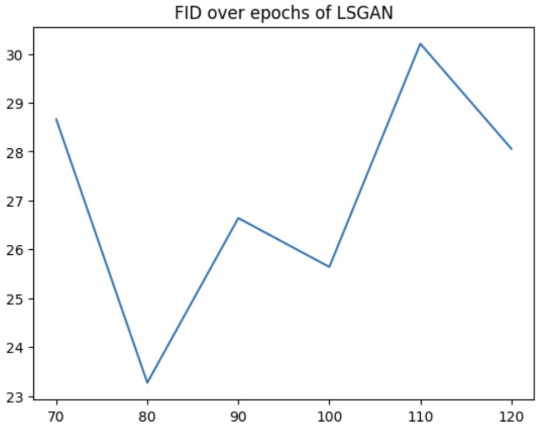
$$\min_D V_{\text{LSGAN}}(D) = \frac{1}{2}\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}\left[(D(\boldsymbol{x})-1)^2\right] + \frac{1}{2}\mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}\left[(D(G(\boldsymbol{z})))^2\right]$$

$$\min_G V_{\text{LSGAN}}(G) = \frac{1}{2}\mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}\left[(D(G(\boldsymbol{z}))-1)^2\right].$$

Equivalent to minimizing Pearson x^2 divergence, quantifying how different one probability distribution is from another

Reduces but does not eliminate vanishing gradient nor mode collapse problems





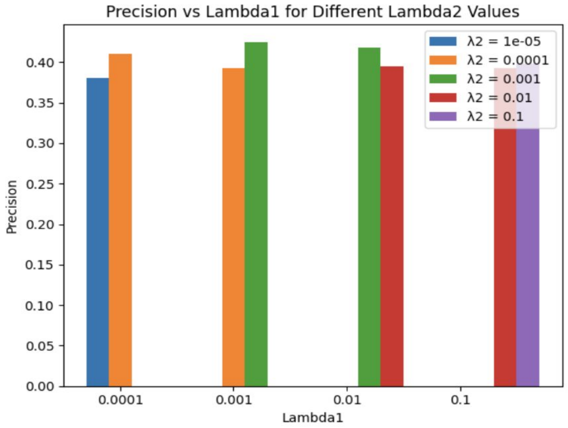
Precision: 0.391 (run locally)
Recall: 0.1737
FID: 26.58

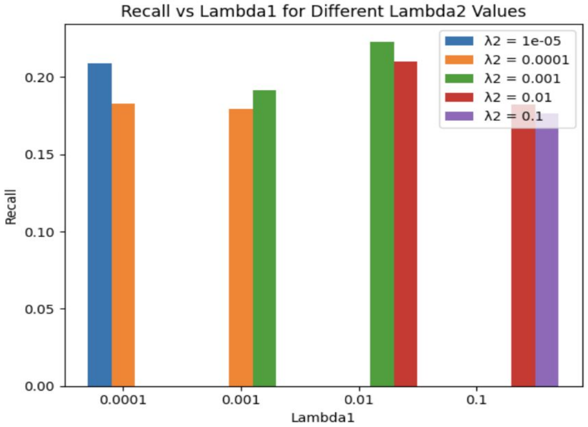
CI-LSGAN

$$\begin{cases} \min\limits_{D} V(D,G) = E_{x\sim P_{\text{data}}(x)}\left[(D(x)-1)^2\right] + E_{z\sim P_z(z)}\left[(D(G(z)))^2\right] + \lambda_1 E_{\hat{x}\sim P_{\hat{x}}}\left[\left(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1\right)^2\right], \\ \min\limits_{G} V(D,G) = E_{z\sim P_z(z)}\left[(D(G(z))-1)^2\right] + \lambda_2 E_{z\sim P_z(z), x\sim P_{\text{data}}(x)}\left[(G(z)-x)^2\right]. \end{cases}$$

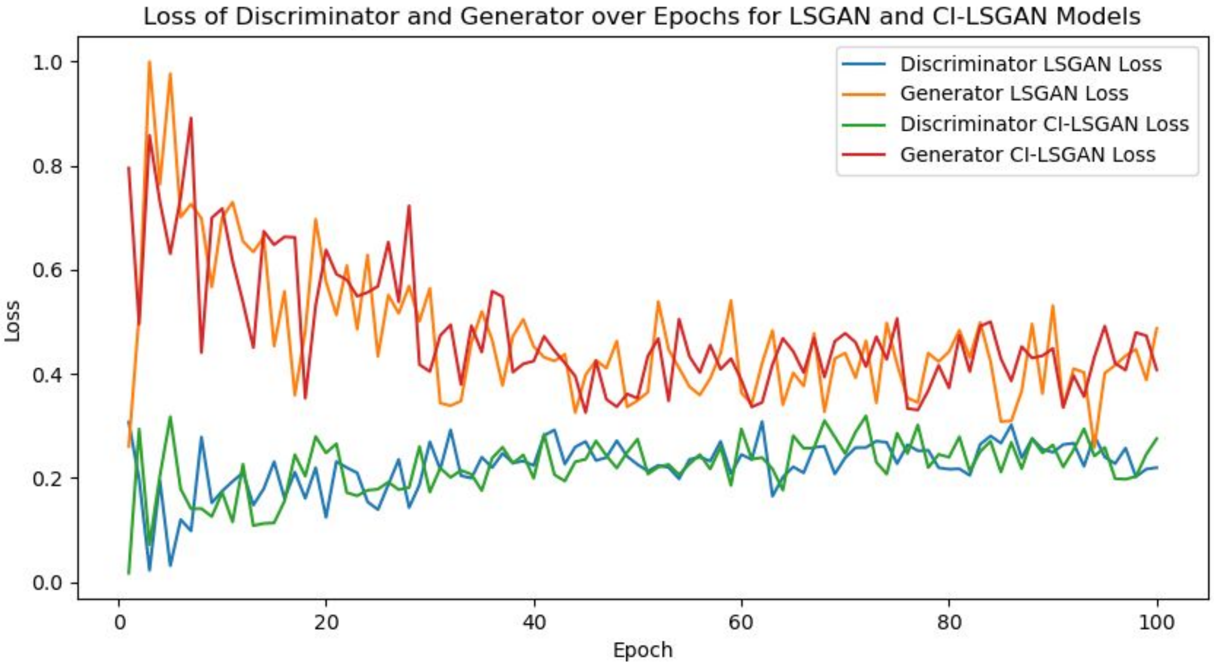
Adding constraint (gradient penalty / reconstruction constraint) terms to the discriminator and generator loss functions

Aims to further reduce vanishing gradient / mode collapse



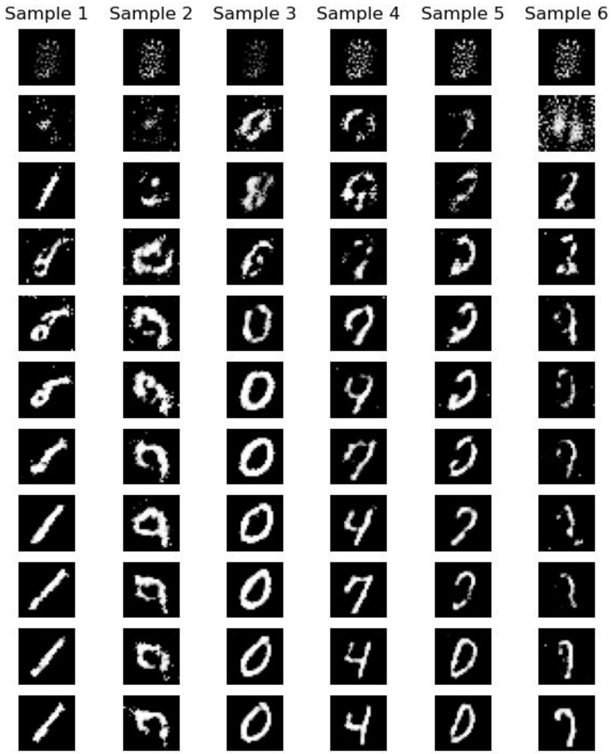


CI-LSGAN

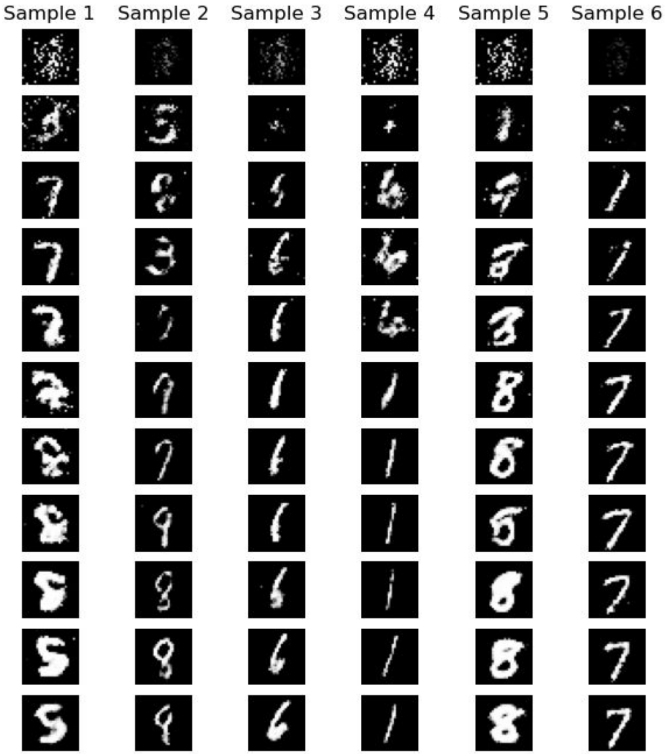


Loss of Discriminator and Generator over epochs for LSGAN and CI-LSGAN

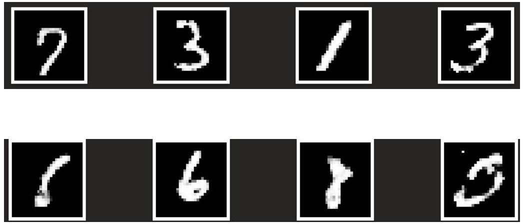
CI-LSGAN



Generation for LSGAN every 10 epochs



Generation for CI-LSGAN every 10 epochs



Precision: 0.373 (run locally)
Recall: 0.2016
FID: 26.34

Improved Consistency Regularization (ICR)

Goal: Improve GAN training stability and output quality

Combines two consistency regularization techniques:

- Balanced Consistency Regularization (bCR)
- Latent Consistency Regularization (zCR)

ICR for GANs

Algorithm 1 Balanced Consistency Regularization (bCR)

Input: parameters of generator θ_G and discriminator θ_D, consistency regularization coefficient for real images λ_{real} and fake images λ_{fake}, augmentation transform T (for images, e.g. shift, flip, cutout, etc).

for number of training iterations **do**

- Sample batch $z \sim p(z)$, $x \sim p_{\text{real}}(x)$
- Augment both real $T(x)$ and fake $T(G(z))$ images
- $L_D \leftarrow D(G(z)) - D(x)$
- $L_{\text{real}} \leftarrow \|D(x) - D(T(x))\|^2$
- $L_{\text{fake}} \leftarrow \|D(G(z)) - D(T(G(z)))\|^2$
- $\theta_D \leftarrow \text{AdamOptimizer}(L_D + \lambda_{\text{real}} L_{\text{real}} + \lambda_{\text{fake}} L_{\text{fake}})$
- $L_G \leftarrow -D(G(z))$
- $\theta_G \leftarrow \text{AdamOptimizer}(L_G)$

end for

T(x): Augmented image
x: Original image
G(z): Generated image
D(): Discriminator output

Algorithm 2 Latent Consistency Regularization (zCR)

Input: parameters of generator θ_G and discriminator θ_D, consistency regularization coefficient for generator λ_{gen} and discriminator λ_{dis}, augmentation transform T (for latent vectors, e.g. adding small perturbation noise$\sim \mathcal{N}(0, \sigma_{\text{noise}})$).

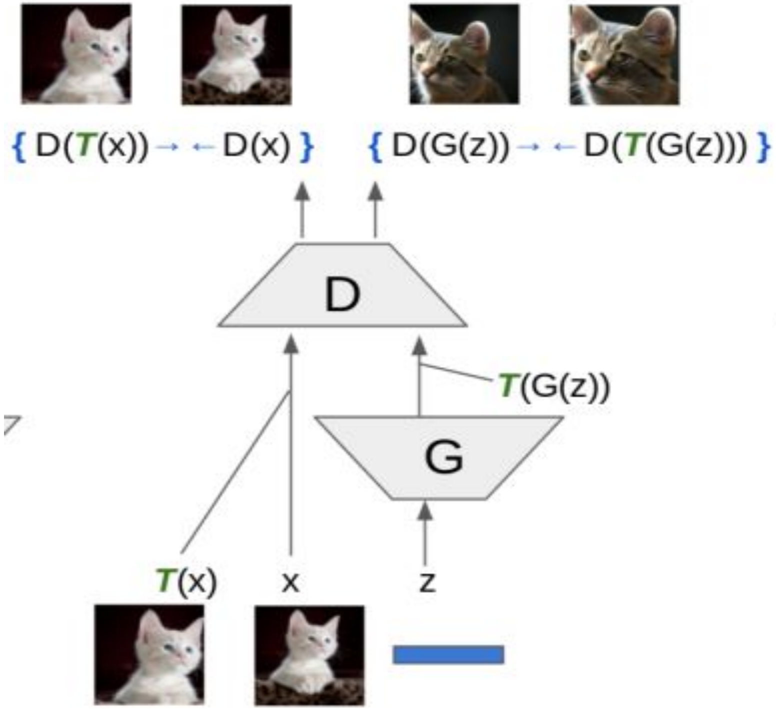
for number of training iterations **do**

- Sample batch $z \sim p(z)$, $x \sim p_{\text{real}}(x)$
- Sample perturbation noise $\Delta z \sim \mathcal{N}(0, \sigma_{\text{noise}})$
- Augment latent vectors $T(z) \leftarrow z + \Delta z$
- $L_D \leftarrow D(G(z)) - D(x)$
- $L_{\text{dis}} \leftarrow \|D(G(z)) - D(G(T(z)))\|^2$
- $\theta_D \leftarrow \text{AdamOptimizer}(L_D + \lambda_{\text{dis}} L_{\text{dis}})$
- $L_G \leftarrow -D(G(z))$
- $L_{\text{gen}} \leftarrow -\|G(z) - G(T(z))\|^2$
- $\theta_G \leftarrow \text{AdamOptimizer}(L_G + \lambda_{\text{gen}} L_{\text{gen}})$

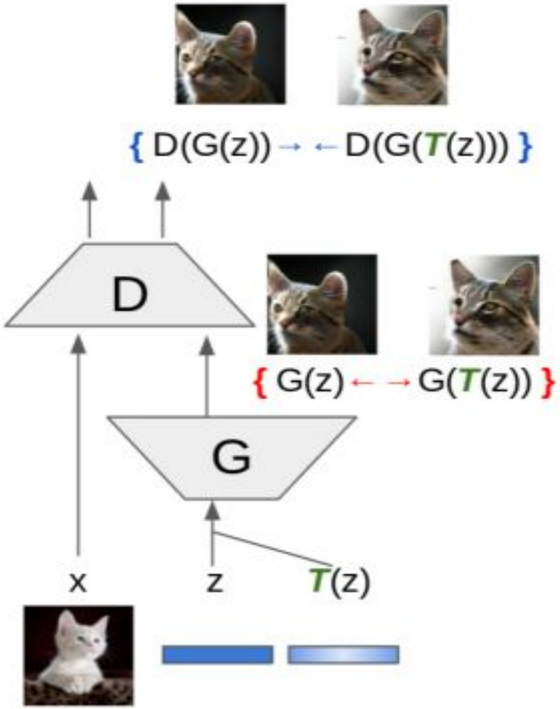
end for

z: Original latent vector
δ: Small random perturbation
G(): Generator output

Improved Consistency Regularization (ICR) for GANs



Balanced Consistency Regularization (bCR)



Latent Consistency Regularization (zCR)

ICR LSGAN training visualization

Epoch 10

Epoch 20

Epoch 30

Epoch 40

Epoch 50

Epoch 60

Epoch 70

Epoch 80

Epoch 90

Epoch 100

* Noise factor of 0.03 for latent space in the generator
Horizontal flip as augmentation in the discriminator

Results

We tried noise factor parameter values of 0.03, 0.05 and 0.1

sigma = 0.03

FID 61.626
precision 0.282
recall 0.079

Example artifacts

sigma = 0.05

FID 60.286,
precision 0.243,
recall 0.101

Example artifacts

sigma = 0.1

FID 54.920,
precision 0.299,
recall 0.088

Example artifacts

Thank you for your attention