

Advanced Computer Vision

Практический курс

Савельева Юлия Олеговна, 3 семестр, 17.11.2022

Wasserstein GAN

Архитектура, инициализация и препроцессинг

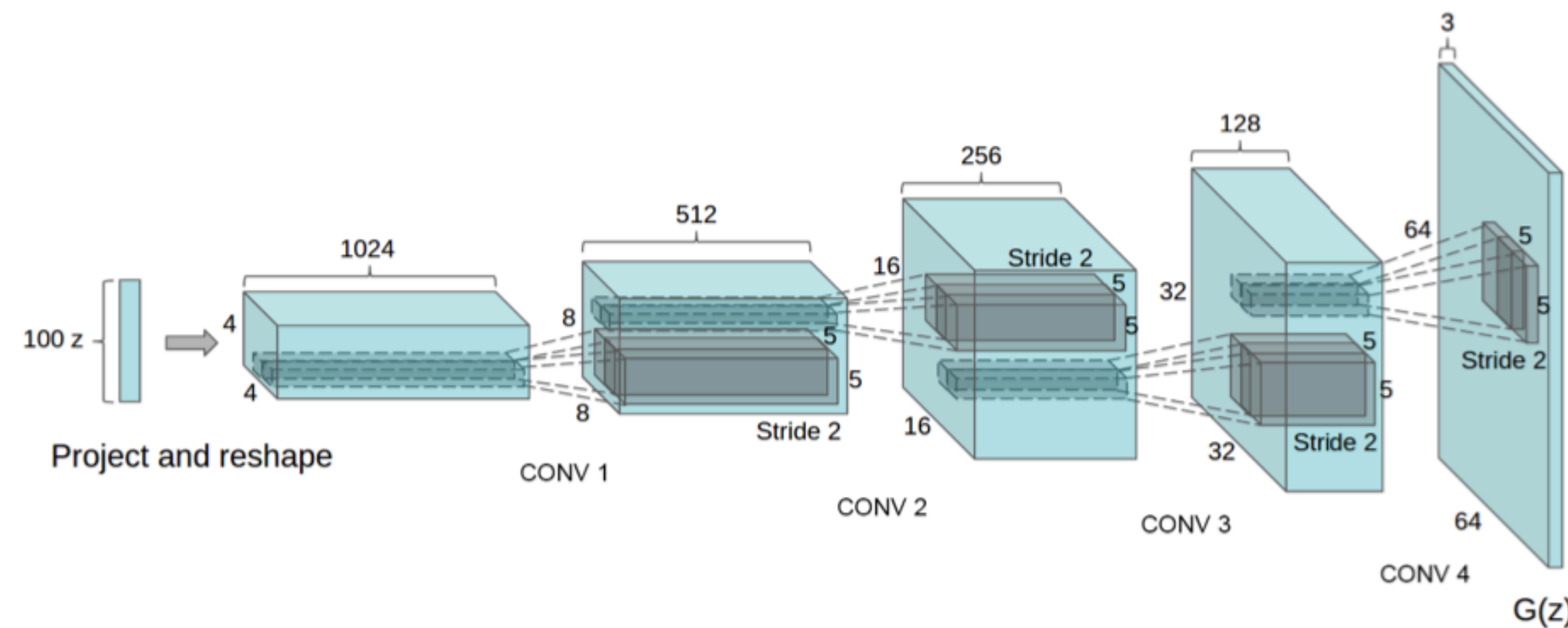
Реализовать нейронную сеть из статьи DCGAN:

<https://arxiv.org/pdf/1511.06434.pdf>

Здесь можно посмотреть реализацию архитектуры, инициализации весов и препроцессинга (но избавиться от Sigmoid в дискриминаторе!):

https://pytorch.org/tutorials/beginner/dcgan_faces_tutorial.html

P.S. В датасете MNIST картинки 28x28, просто применяем к ним такой же препроцессинг, чтобы они растянулись до 64x64.



Wasserstein GAN

Целевая функция

Статья, на которую опираться при реализации:

<https://arxiv.org/pdf/1704.00028.pdf>

Гиперпараметры из псевдокода оставить такими же.

Реализация подсчета gradient penalty можно подсмотреть в функции

TOGradOfDisc:

https://github.com/jalola/compare-tensorflow-pytorch/blob/master/test/gan_pytorch_model.py , где

`interpolates = eps * real + (1 - eps) * fake`

Algorithm 1 WGAN with gradient penalty. We use default values of $\lambda = 10$, $n_{\text{critic}} = 5$, $\alpha = 0.0001$, $\beta_1 = 0$, $\beta_2 = 0.9$.

Require: The gradient penalty coefficient λ , the number of critic iterations per generator iteration n_{critic} , the batch size m , Adam hyperparameters α, β_1, β_2 .

Require: initial critic parameters w_0 , initial generator parameters θ_0 .

```
1: while  $\theta$  has not converged do
2:   for  $t = 1, \dots, n_{\text{critic}}$  do
3:     for  $i = 1, \dots, m$  do
4:       Sample real data  $\mathbf{x} \sim \mathbb{P}_r$ , latent variable  $\mathbf{z} \sim p(\mathbf{z})$ , a random number  $\epsilon \sim U[0, 1]$ .
5:        $\tilde{\mathbf{x}} \leftarrow G_{\theta}(\mathbf{z})$ 
6:        $\hat{\mathbf{x}} \leftarrow \epsilon \mathbf{x} + (1 - \epsilon) \tilde{\mathbf{x}}$ 
7:        $L^{(i)} \leftarrow D_w(\tilde{\mathbf{x}}) - D_w(\mathbf{x}) + \lambda(\|\nabla_{\hat{\mathbf{x}}} D_w(\hat{\mathbf{x}})\|_2 - 1)^2$ 
8:     end for
9:      $w \leftarrow \text{Adam}(\nabla_w \frac{1}{m} \sum_{i=1}^m L^{(i)}, w, \alpha, \beta_1, \beta_2)$ 
10:   end for
11:   Sample a batch of latent variables  $\{\mathbf{z}^{(i)}\}_{i=1}^m \sim p(\mathbf{z})$ .
12:    $\theta \leftarrow \text{Adam}(\nabla_{\theta} \frac{1}{m} \sum_{i=1}^m -D_w(G_{\theta}(\mathbf{z})), \theta, \alpha, \beta_1, \beta_2)$ 
13: end while
```

Wasserstein GAN

Процесс обучения

При обучении советую мониторить значение целевой функции генератора и дискриминатора, а также точность дискриминатора на реальных и сгенерированных картинках.

Обучение лучше начать с генерации только одной буквы из MNIST!

Этот тьюториал полезен для того, чтобы понимать, как процесс обучения будет идти и сколько ждать результатов (архитектуру отсюда не использовать):

[https://
machinelearningmastery.com/
practical-guide-to-gan-failure-
modes/](https://machinelearningmastery.com/practical-guide-to-gan-failure-modes/)

Дедлайн 01.12.2022 23:59