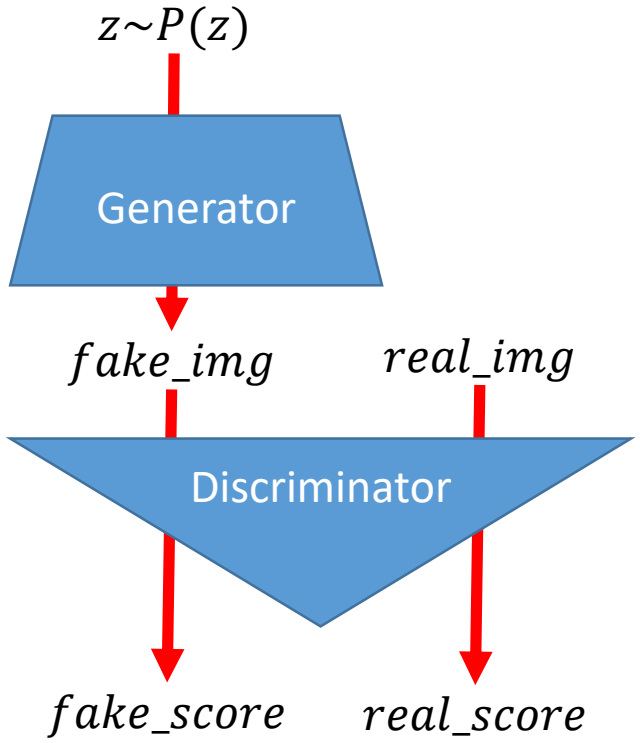
InfoGAN

助教：來俊聖

Standard GAN

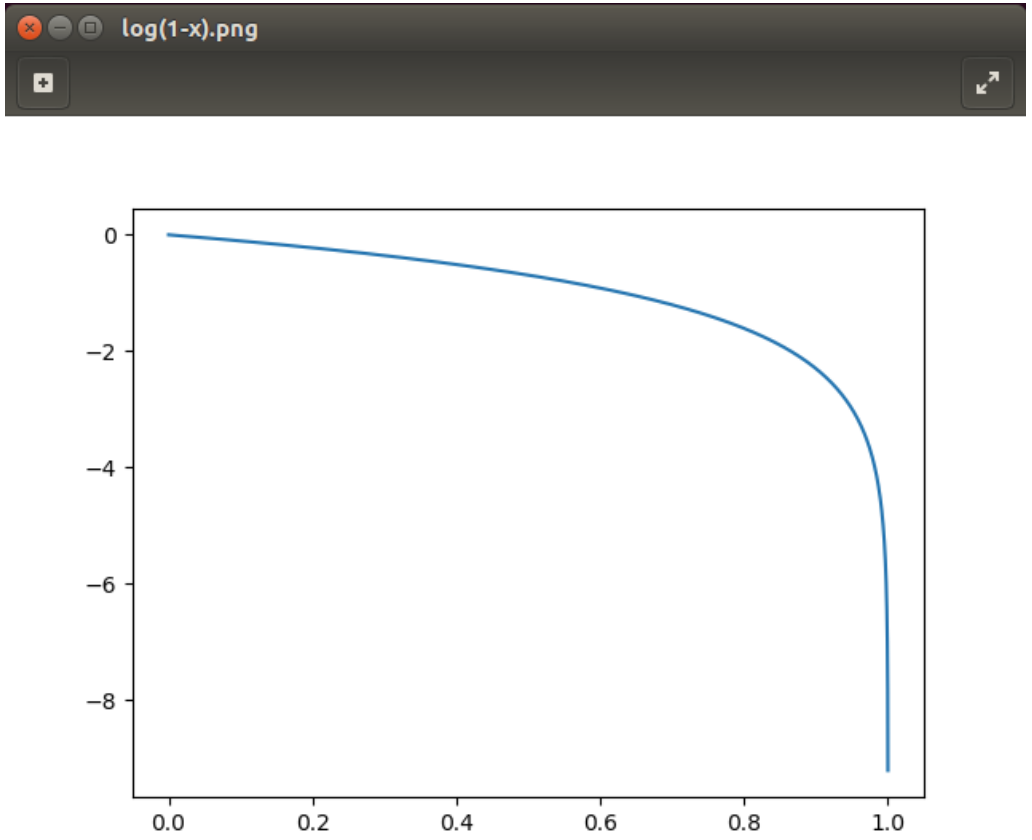
$$L_D = -\log\left(\mathrm{D}(I_{real})\right) - \log\left(1 - D\left(G(z)\right)\right)$$

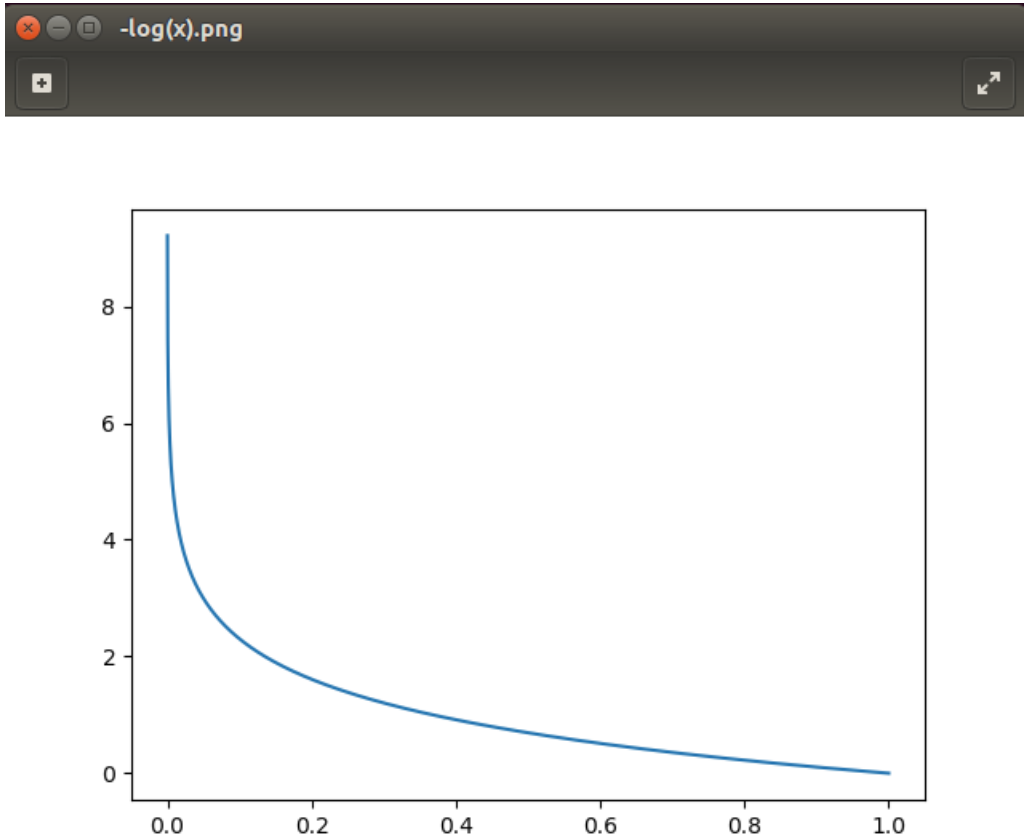
$$\begin{cases} L_G = \log\left(1 - D\left(G(z)\right)\right) \\ L_G = -\log\left(D\left(G(z)\right)\right) \end{cases}$$



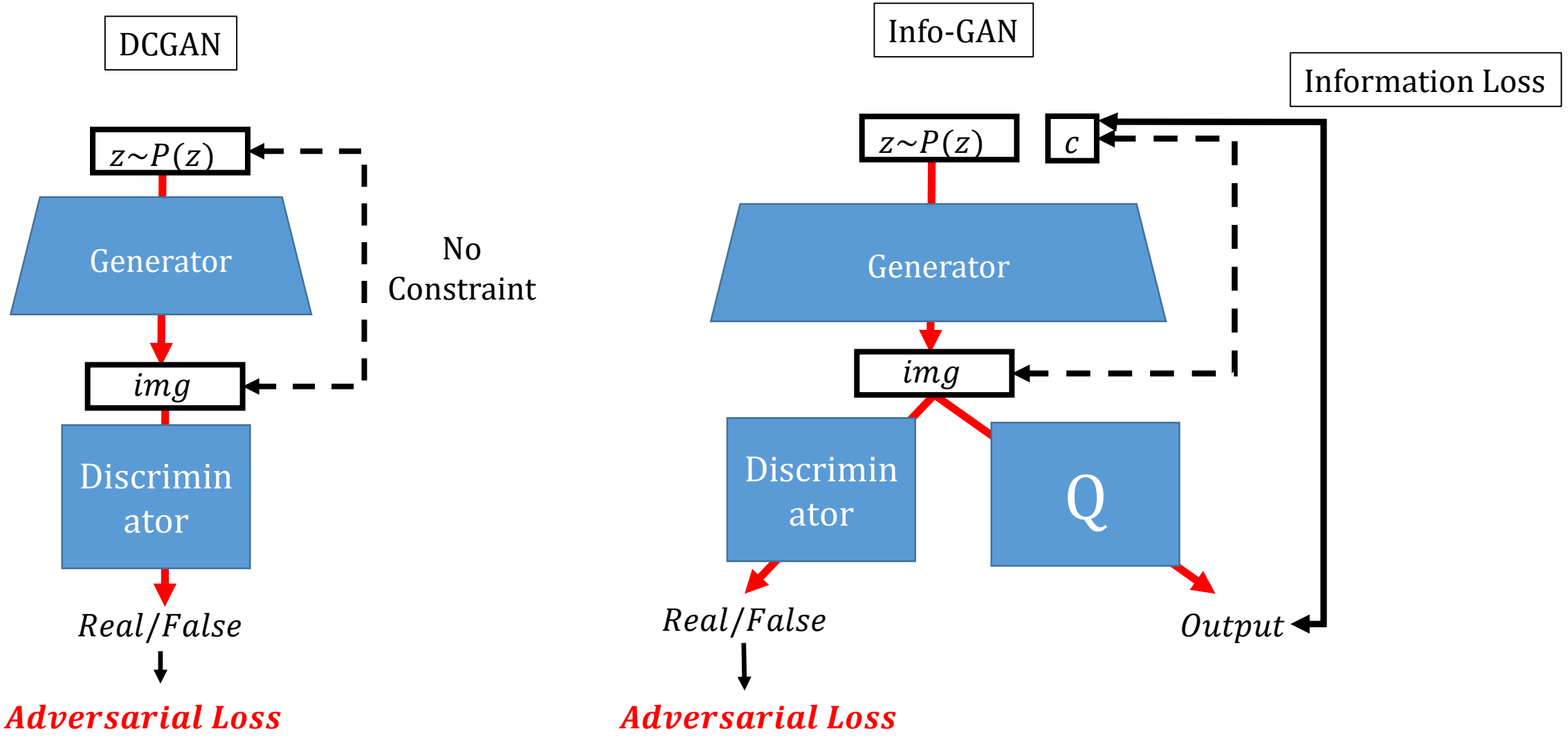
In this lab, you can use either of them, but you should tell me which one you use in your report.

Standard GAN Loss





Info-GAN

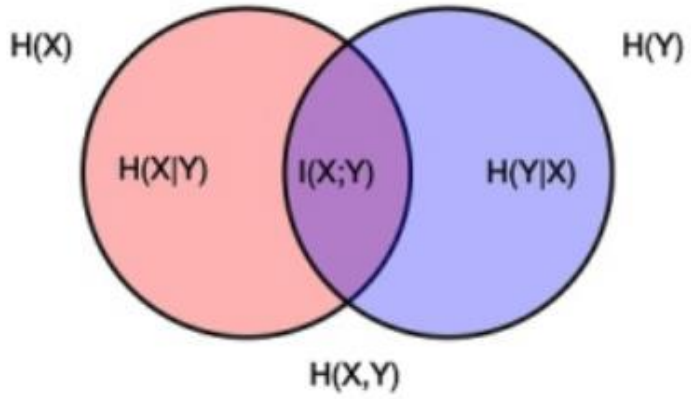

DCGAN
$z \sim P(z)$
Generator
img
Discrimin
ator
$Real/False$
No
Constraint
Adversarial Loss
Info-GAN
Information Loss
$z \sim P(z)$
c
Generator
img
Discrimin
ator
Q
$Real/False$
$Output$
Adversarial Loss

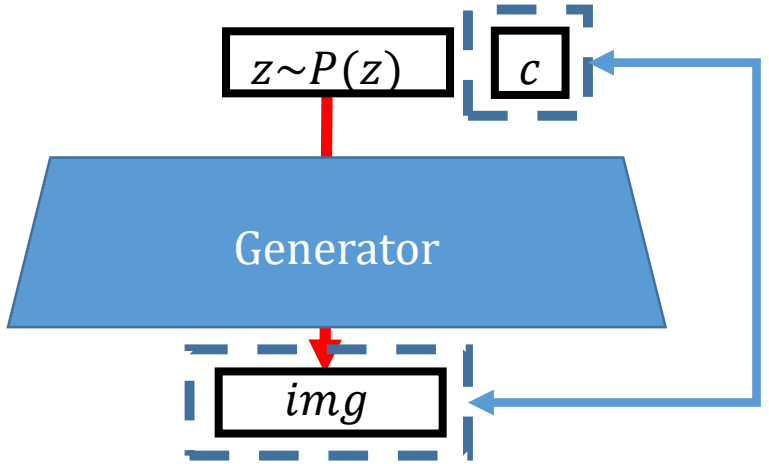
Info-GAN

In Information Theory:
Mutual Information: I(X;Y) = H(X)-H(X|Y) = H(Y)-H(Y|X)

$$H(Y) = -\sum_{y} \log p(y)p(y)$$

$$I(X;Y) = H(X) - H(X \mid Y) = H(Y) - H(Y \mid X)$$





$I(c; G(z, c))$

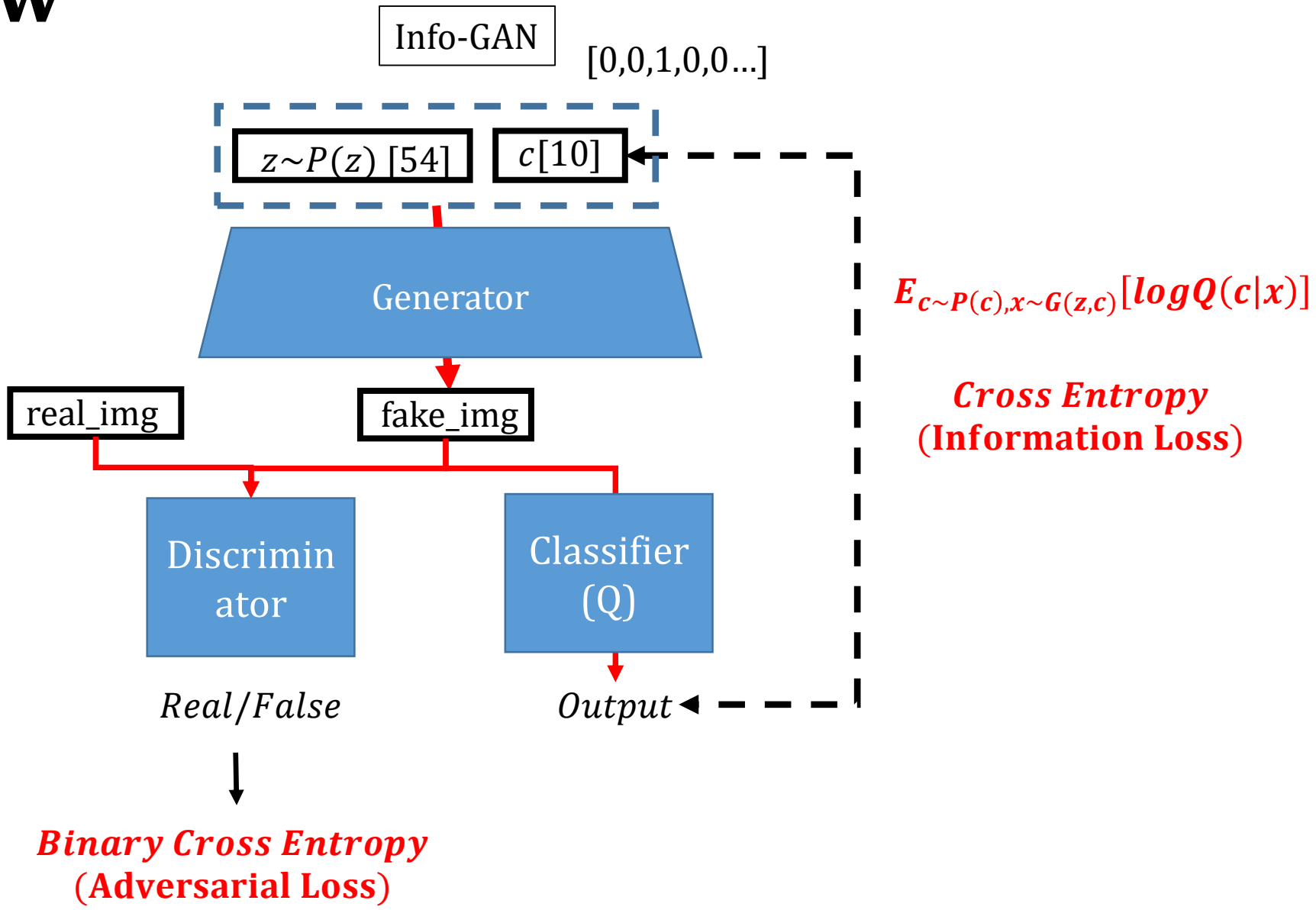
Info-GAN pf.

$$
\begin{aligned}
&\mathrm{I}(\mathrm{c};\mathrm{G}(\mathrm{z},\mathrm{c})) = \mathrm{H}(\mathrm{c})\text{-}\mathrm{H}(\mathrm{c}|\mathrm{G}(\mathrm{z},\mathrm{c})) \\
&= E_{x\sim G(z,c)}\left[E_{c'\sim P(C|x)}[\log P(c'|x)]\right] + \mathrm{H}(\mathrm{c}) \\
&= E_{x\sim G(z,c)}\left[D_{KL}(P(\cdot\,|x)||Q(\cdot\,|x)) + E_{c'\sim P(C|x)}[\log Q(c'|x)]\right] + \mathrm{H}(\mathrm{c}) \\
&\geq E_{x\sim G(z,c)}\left[E_{c'\sim P(C|x)}[\log Q(c'|x)]\right] + \mathrm{H}(\mathrm{c}) \\
&= \boldsymbol{E_{c\sim P(c),x\sim G(z,c)}[\log Q(c|x)] + \mathrm{H}(\mathrm{c})}
\end{aligned}
$$

We can increase the **lower bound of mutual Information** by other distribution!!

Info-GAN Flow

Take MNIST for example



Info-GAN Loss

Adversarial Loss

$$L_D = -\log\big(\mathrm{D}(I_{real})\big) - \log(1 - D(G(z)))$$

$$\begin{cases} L_G = \log\Big(1 - D\big(G(z)\big)\Big) \\ L_G = -\log\Big(D\big(G(z)\big)\Big) \end{cases}$$

Update D ⇌ Update G/Q

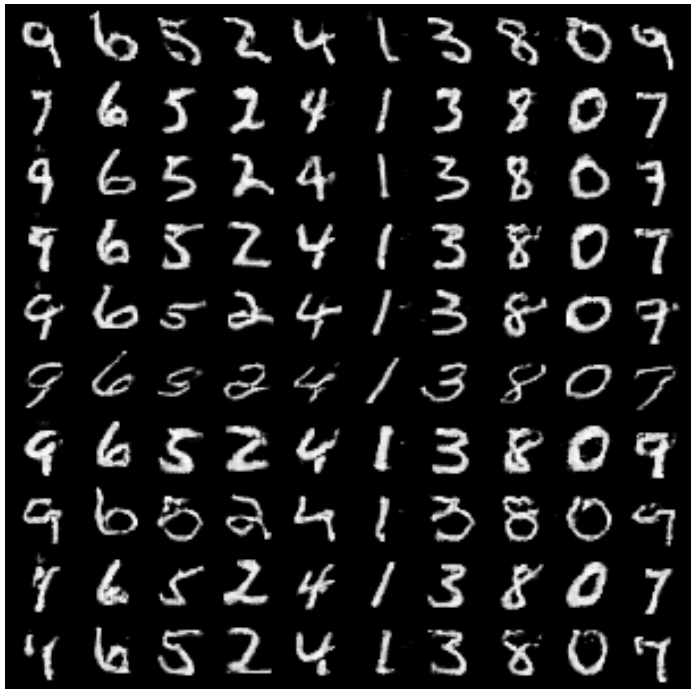
Information Loss

$$L_I(Q, G) = -\log\big(Q(c|G(z, c))\big)$$

Procedure

1. Update D network
   a. feed real_image into D network, compute bce loss for D network
   b. Create noise and concatenate with one-hot latent named z
   c. Feed z into the G model and generate fake_image
   d. Feed fake_image into D network, compute bce loss for D network
2. Update G and Q network
   a. Generate fake_image like previously, feed into discriminator and compute bce loss for G network
   b. Feed fake_image into Q network, then compute classification loss (crossentropy) for G network

Expected Outputs



To do list

- Complete all model
- Complete training procedure
  - The details are at the pseudo code part.
  - You need to tune learning rate and the weight of each loss at this part.
- Bonus
  - At this part, you need to can directly run the training script with a new model for CelebA, but you need to load data yourself.
  - Torchvision.datasets.ImageFolder will help, please check it out.

Hyper-parameters

- mnist
  1. c_size = 10
  2. z_size = 62
  3. Total epochs = 50
  4. Optimizer: Adam
- CelebA
  1. c_size = 100
  2. z_size = 128
  3. Total epochs = 10
  4. Optimizer: Adam
- All other parameters need to be tuned
- You can even change the hyper parameters above

Important Date

- Deadline: 10/30 11:59 a.m.
- Demo date: 10/30

- Zip report and source code into a .zip file and name it as DLP_LAB5_yourID_name.zip
- Email to alanlai199.cs07g@nctu.edu.tw with email title DLP_LAB5_yourID_name

Info-GAN Report Spec

MNIST

- 1. Introduction (10%)
- 2. Experiment setups: (20%)
  - A. How you implement InfoGAN
    - i. Adversarial loss
    - ii. Maximizing mutual information
  - B. Which loss function of generator you used? What's different?
- 3. Results (30%)
  - A. results of your samples
  - B. Training loss curves
- 4. Discussion (20%)
- 5. Demo (20%)
  - Show your result and explain your code
- (*optional*) Bonus: CelebA (15%)
  - Results of your samples (10%)
  - Demo (5%)