FIT5215 Deep Learning

Generative Adversarial Networks And Diffusion Models

Teaching team

Department of Data Science and AI
Faculty of Information Technology, Monash University
Email: **trunglm@monash.edu**

Question 1

- What is the main task of a generative model?

A. Learn good latent codes by minimizing the reconstruction error.
B. Learn a model to help predicting labels for a data example.
C. Learn to generate meaningful and good examples that imitate realistic examples in a given dataset from noises
D. Learn to divide data examples into groups for which those in a group have high similarity

Question 1

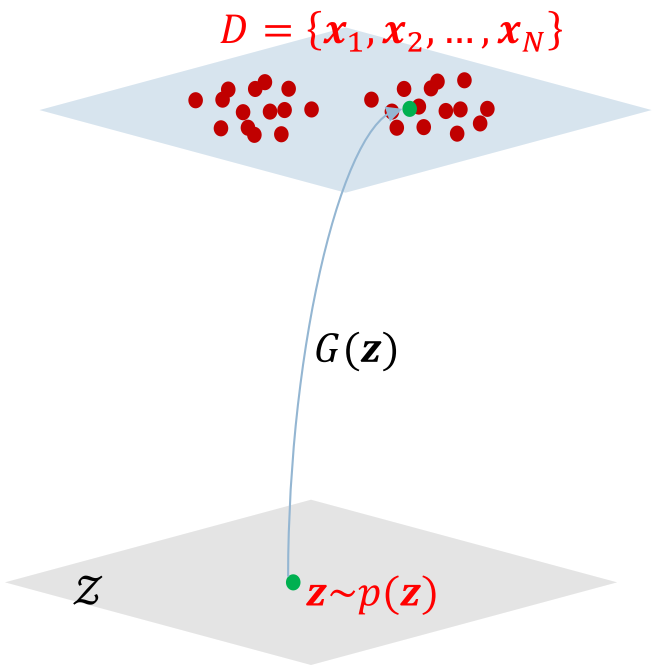
□ What is the main task of a generative model?

A. Learn good latent codes by minimizing the reconstruction error.

B. Learn a model to help predicting labels for a data example.

C. Learn to generate meaningful and good examples that imitate realistic examples in a given dataset from noises ✓

D. Learn to divide data examples into groups for which those in a group have high similarity



Question 2

- ☐ What are correct about GANs?

A. The noise is fed to the discriminator to generate fake examples

B. GANs use a discriminator to be aware of the difference or divergence between the distribution of generated examples and the distribution of data

C. The noise is fed to the generator to generate fake examples

D. The discriminator tries to distinguish the real and fake data examples

Question 2

- What are correct about GANs?

A. The noise is fed to the discriminator to generate fake examples
B. GANs use a discriminator to be aware of the difference or divergence between the distribution of generated examples and the distribution of data ✓
C. The noise is fed to the generator to generate fake examples ✓
D. The discriminator tries to distinguish the real and fake data examples ✓

Question 3

- What are the tasks of generator and discriminator in GANs?

A. Discriminator tries to discriminate the real and fake data

B. Discriminator tries to set high values for real data and low values for fake data

C. Generator tries to fool discriminator by generating examples that are dissimilar to real data

D. Generator tries to fool discriminator by generating examples that mimic real data

Question 3

- What are the tasks of generator and discriminator in GANs?

A. Discriminator tries to discriminate the real and fake data ✓
B. Discriminator tries to set high values for real data and low values for fake data ✓
C. Generator tries to fool discriminator by generating examples that are dissimilar to real data
D. Generator tries to fool discriminator by generating examples that mimic real data ✓

Question 4

- What are correct about the analogy of GANs?

A. Discriminator is the machine that produces fake money
B. Generator is the police who attempts to detect fake money
C. Generator is the machine that produces fake money
D. Discriminator is the police who attempts to detect fake money

Question 4

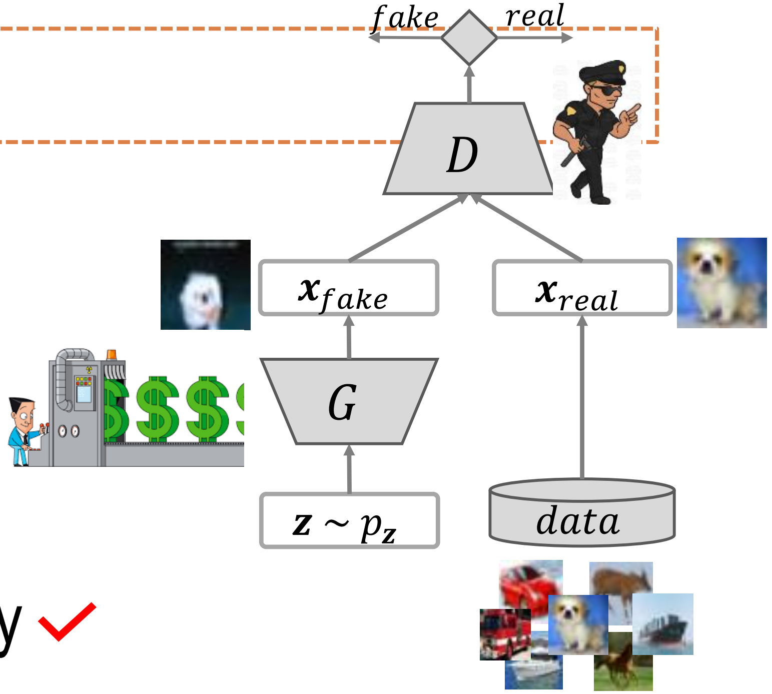
□ What are correct about the analogy of GANs?

A. Discriminator is the machine that produces fake money

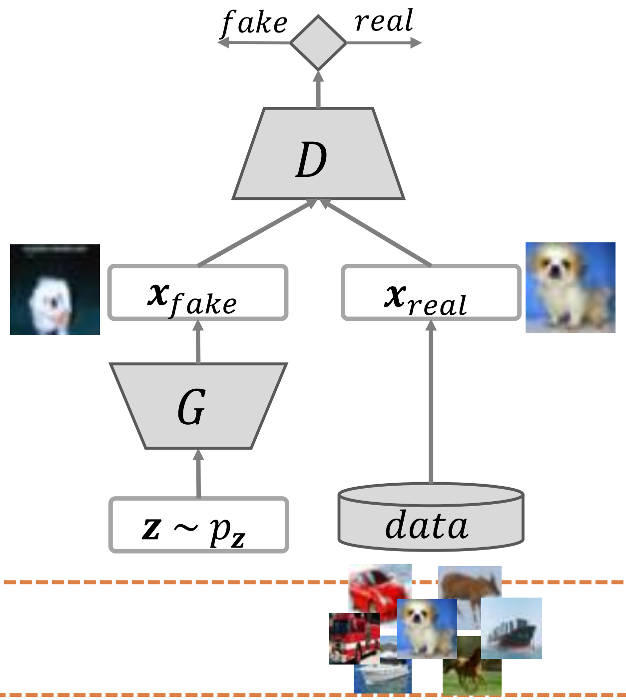
B. Generator is the police who attempts to detect fake money

C. Generator is the machine that produces fake money ✓

D. Discriminator is the police who attempts to detect fake money ✓



Question 5



- How to train GANs?

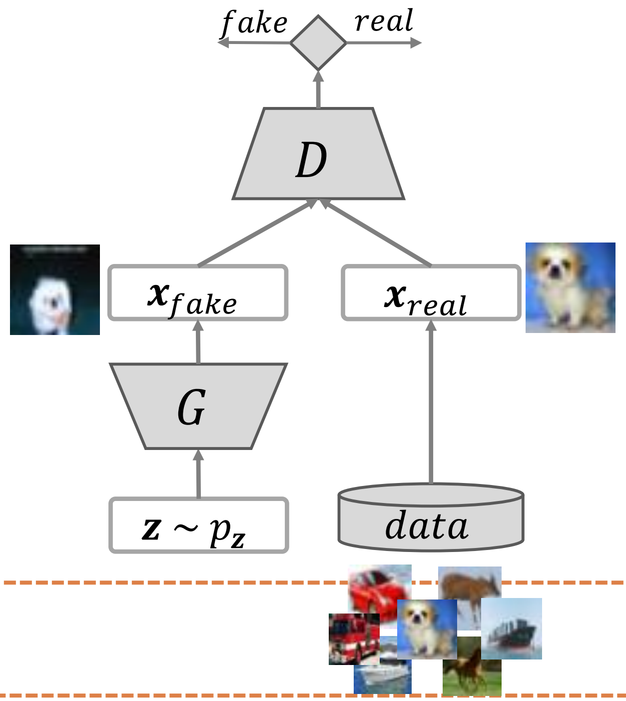
A. $$\min_G \max_D J(G, D) = \mathbb{E}_{x \sim p_d(x)}[\log D(x)] + \mathbb{E}_{z \sim p(z)}[\log(1 - D(G(z))]$$

B. $$\min_{\theta, \Phi} \mathbb{E}_{x \sim \mathbb{P}}[d(\tilde{x}, g_\Phi(f_\theta(x)))]$$

C. $$\max_G \min_D J(G, D) = \mathbb{E}_{x \sim p_d(x)}[\log D(x)] + \mathbb{E}_{z \sim p(z)}[\log(1 - D(G(z))]$$

D. $$\min_G \max_D J(G, D) = \mathbb{E}_{x \sim p_d(x)}[\log(1 - D(x))] + \mathbb{E}_{z \sim p(z)}[\log D(G(z)]$$

Question 5



- How to train GANs?

A. $$\min_G \max_D J(G, D) = \mathbb{E}_{x \sim p_d(x)}[\log D(x)] + \mathbb{E}_{z \sim p(z)}[\log(1 - D(G(z)))]$$

B. $$\min_{\theta, \Phi} \mathbb{E}_{x \sim \mathbb{P}}[d(\tilde{x}, g_\Phi(f_\theta(x)))]$$

C. $$\max_G \min_D J(G, D) = \mathbb{E}_{x \sim p_d(x)}[\log D(x)] + \mathbb{E}_{z \sim p(z)}[\log(1 - D(G(z)))]$$

D. $$\min_G \max_D J(G, D) = \mathbb{E}_{x \sim p_d(x)}[\log(1 - D(x))] + \mathbb{E}_{z \sim p(z)}[\log D(G(z))]$$

Just label real examples 0 and fake examples 1. Discriminator is still trained distinguish real and fake data.

Question 6

- How to update the generator and discriminator alternatively?

A. $$\max_{D} J(G,D) = \mathbb{E}_{\boldsymbol{x} \sim p_d(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z} \sim p(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z}))]$$
$$\max_{G} \mathbb{E}_{\boldsymbol{z}}[\log(D(G(\boldsymbol{z}))]$$

B. $$\min_{D} J(G,D) = \mathbb{E}_{\boldsymbol{x} \sim p_d(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z} \sim p(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z}))]$$
$$\max_{G} \mathbb{E}_{\boldsymbol{z}}[\log(D(G(\boldsymbol{z}))]$$

C. $$\max_{D} J(G,D) = \mathbb{E}_{\boldsymbol{x} \sim p_d(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z} \sim p(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z}))]$$
$$\min_{G} \mathbb{E}_{\boldsymbol{z}}[\log(1 - D(G(\boldsymbol{z}))]$$

D. $$\max_{D} J(G,D) = \mathbb{E}_{\boldsymbol{x} \sim p_d(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z} \sim p(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z}))]$$
$$\max_{G} \mathbb{E}_{\boldsymbol{z}}[\log(1 - D(G(\boldsymbol{z}))]$$

Question 6

- How to update the generator and discriminator alternatively?

A. $$\max_{\mathrm{D}} J(G,D) = \mathbb{E}_{\boldsymbol{x}\sim p_d(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z}))]$$
$$\max_G \mathbb{E}_{\boldsymbol{z}}[\log(D(G(\boldsymbol{z}))]$$
✓

B. $$\min_{\mathrm{D}} J(G,D) = \mathbb{E}_{\boldsymbol{x}\sim p_d(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z}))]$$
$$\max_G \mathbb{E}_{\boldsymbol{z}}[\log(D(G(\boldsymbol{z}))]$$

C. $$\max_{\mathrm{D}} J(G,D) = \mathbb{E}_{\boldsymbol{x}\sim p_d(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z}))]$$
$$\min_G \mathbb{E}_{\boldsymbol{z}}[\log(1 - D(G(\boldsymbol{z}))]$$
✓

D. $$\max_{\mathrm{D}} J(G,D) = \mathbb{E}_{\boldsymbol{x}\sim p_d(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z}))]$$
$$\max_G \mathbb{E}_{\boldsymbol{z}}[\log(1 - D(G(\boldsymbol{z}))]$$

Question 7

- When the distribution of generated/fake examples p_g and the data distribution p_d becomes more overlapping, what are correct?

A. The task of discriminator becomes easier.

B. The task of discriminator becomes harder.

C. The accuracy of discriminator approaches 50% (random guest).

D. The discriminator loss (binary cross-entropy loss) becomes smaller.

Question 7

- When the distribution of generated/fake examples p_g and the data distribution p_d becomes more overlapping, what are correct?

A. The task of discriminator becomes easier.

B. The task of discriminator becomes harder. ✓

C. The accuracy of discriminator approaches 50% (random guest). ✓

D. The discriminator loss (binary cross-entropy loss) becomes smaller.

Question 8

- What are correct about the discriminator?

A. We need to apply sigmoid at the output layer of discriminator

B. $D(\boldsymbol{x})$ is the probability $\boldsymbol{x}$ to be a fake/generated example

C. $D(\boldsymbol{x})$ is the probability $\boldsymbol{x}$ to be a real example

D. $1 - D(\boldsymbol{x})$ is the probability $\boldsymbol{x}$ to be a fake/generated example

Question 8

- What are correct about the discriminator?

A. We need to apply sigmoid at the output layer of discriminator ✓
B. $D(\boldsymbol{x})$ is the probability $\boldsymbol{x}$ to be a fake/generated example
C. $D(\boldsymbol{x})$ is the probability $\boldsymbol{x}$ to be a real example ✓
D. $1 - D(\boldsymbol{x})$ is the probability $\boldsymbol{x}$ to be a fake/generated example ✓

Question 9

- Given a distribution of generated examples with pdf p_g and data distribution with pdf p_d, the discriminator D tries to discriminate generated and real examples by minimizing the binary cross-entropy. What are correct for the optimal discriminator D^*?

A. $D^*(\boldsymbol{x}) = \frac{1}{p_g(\boldsymbol{x})+1}$

B. $D^*(\boldsymbol{x}) = \frac{p_g(\boldsymbol{x})}{p_g(\boldsymbol{x})+p_d(\boldsymbol{x})}$

C. $D^*(\boldsymbol{x}) = \frac{p_d(\boldsymbol{x})}{p_g(\boldsymbol{x})+p_d(\boldsymbol{x})}$

D. $D^*(\boldsymbol{x}) = \frac{1}{1+\frac{p_g(\boldsymbol{x})}{p_d(\boldsymbol{x})}}$

Question 9

- Given a distribution of generated examples with pdf p_g and data distribution with pdf p_d, the discriminator D tries to discriminate generated and real examples by minimizing the binary cross-entropy. What are correct for the optimal discriminator D^*?

A. $D^*(\boldsymbol{x}) = \frac{1}{p_g(\boldsymbol{x})+1}$

B. $D^*(\boldsymbol{x}) = \frac{p_g(\boldsymbol{x})}{p_g(\boldsymbol{x})+p_d(\boldsymbol{x})}$

C. $D^*(\boldsymbol{x}) = \frac{p_d(\boldsymbol{x})}{p_g(\boldsymbol{x})+p_d(\boldsymbol{x})}$ ✓

D. $D^*(\boldsymbol{x}) = \frac{1}{1+\frac{p_g(\boldsymbol{x})}{p_d(\boldsymbol{x})}}$ ✓

Question 10

□ At the Nash equilibrium point of the min-max game of GAN, what are correct?

A. $D^*(\boldsymbol{x}) = 1, \forall \boldsymbol{x}$

B. $D^*(\boldsymbol{x}) = 0.5, \forall \boldsymbol{x}$

C. $p_d(\boldsymbol{x}) = p_{g^*}(\boldsymbol{x}), \forall \boldsymbol{x}$

D. $p_{g^*}(\boldsymbol{x}) = p_d^2(\boldsymbol{x}), \forall \boldsymbol{x}$

Question 10

- At the Nash equilibrium point of the min-max game of GAN, what are correct?

A. $D^*(\boldsymbol{x}) = 1, \forall \boldsymbol{x}$

B. $D^*(\boldsymbol{x}) = 0.5, \forall \boldsymbol{x}$ ✓

C. $p_d(\boldsymbol{x}) = p_{g^*}(\boldsymbol{x}), \forall \boldsymbol{x}$ ✓

D. $p_{g^*}(\boldsymbol{x}) = p_d^2(\boldsymbol{x}), \forall \boldsymbol{x}$

Question 11

- Mode collapse of GANs happens when

A. The generator is too strong and can generate data to cover all data modes
B. The generator is too weak and can generate data to cover all data modes
C. The discriminator cannot classify well real and fake data
D. The generated data can cover only a few modes in real data and miss many other modes

Question 11

- Mode collapse of GANs happens when

A. The generator is too strong and can generate data to cover all data modes
B. The generator is too weak and can generate data to cover all data modes
C. The discriminator cannot classify well real and fake data
D. The generated data can cover only a few modes in real data and miss many other modes ✓

Question 12

- What is the main source of mode collapsing problem of GANs?

A. The generator is too weak and cannot generate data to cover all data modes
B. The discriminator cannot classify well real and fake data
C. We cannot solve the min-max problem of GANs perfectly
D. When updating the generator, there is not any constraints for it to generate data corresponding to all modes

Question 12

- What is the main source of mode collapsing problem of GANs?

A. The generator is too weak and cannot generate data to cover all data modes
B. The discriminator cannot classify well real and fake data
C. We cannot solve the min-max problem of GANs perfectly ✓
D. When updating the generator, there is not any constraints for it to generate data corresponding to all modes ✓

$$\max_{D} J(G, D) = \mathbb{E}_{\boldsymbol{x} \sim p_d(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z} \sim p(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))]$$

$$\max_{G} \mathbb{E}_{\boldsymbol{z}}[\log(D(G(\boldsymbol{z})))] \Rightarrow$$

Just guide $G(z)$ to move toward some modes to gain high D values
Not require to move generated examples to all data modes.

Question 13

□ What are the issues of training GANs?

A. Mode collapse, unrealistic generated images for complex datasets, unstable training
B. Mode collapse, unrealistic generated images, too many parameters
C. Mode collapse, hard to train discriminator, too many parameters
D. Mode collapse, hard to train discriminator, hard to train generator

Question 13

- What are the issues of training GANs?

A. Mode collapse, unrealistic generated images for complex datasets, unstable training ✓
B. Mode collapse, unrealistic generated images, too many parameters
C. Mode collapse, hard to train discriminator, too many parameters
D. Mode collapse, hard to train discriminator, hard to train generator

Question 14

```python
import torch
import torch.nn as nn

class Model(nn.Module):
    def __init__(self):
        super(Model, self).__init__()

        self.model = nn.Sequential(
            nn.Linear(100, 7 * 7 * 256, bias=False),  # Dense layer in Keras is equivalent to Linear in PyTorch
            nn.BatchNorm1d(7 * 7 * 256),
            nn.LeakyReLU(),
            nn.Unflatten(1, (256, 7, 7)),  # Reshape in Keras is equivalent to Unflatten in PyTorch

            nn.ConvTranspose2d(256, 128, kernel_size=(5, 5), stride=(3, 3), padding=(2, 2), output_padding=(0, 0), bias=False)
        )

    def forward(self, x):
        return self.model(x)

# Example usage
model = Model()
x = torch.randn(16, 100)  # Example input with batch size 16 and input shape (100,)
output = model(x)
print(output.shape)  # Should match the output shape after the transpose convolution
```

- What are the shape of the output tensor of the model (not count the batch size)?

A. (16,128, 19, 19)

B. (16,128, 20, 20)

C. (16,128, 19, 18)

D. (16,128, 21, 21)

Question 14

```python
import torch
import torch.nn as nn

class Model(nn.Module):
    def __init__(self):
        super(Model, self).__init__()

        self.model = nn.Sequential(
            nn.Linear(100, 7 * 7 * 256, bias=False),  # Dense layer in Keras is equivalent to Linear in PyTorch
            nn.BatchNorm1d(7 * 7 * 256),
            nn.LeakyReLU(),
            nn.Unflatten(1, (256, 7, 7)),  # Reshape in Keras is equivalent to Unflatten in PyTorch

            nn.ConvTranspose2d(256, 128, kernel_size=(5, 5), stride=(3, 3), padding=(2, 2), output_padding=(0, 0), bias=False)
        )

    def forward(self, x):
        return self.model(x)

# Example usage
model = Model()
x = torch.randn(16, 100)  # Example input with batch size 16 and input shape (100,)
output = model(x)
print(output.shape)  # Should match the output shape after the transpose convolution
```

- What are the shape of the output tensor of the model (not count the batch size)?

A. (16,128, 19, 19) ✓

B. (16,128, 20, 20)

C. (16,128, 19, 18)

D. (16,128, 21, 21)

Output Size=(I−1)×S−2P+K+output padding

(7-1)*3 – 4 + 5 = 19

Question 15

- Matching the forward and backward processes of diffusion models (DMs) to?

A. Forward process of DMs

B. Backward process of DMs

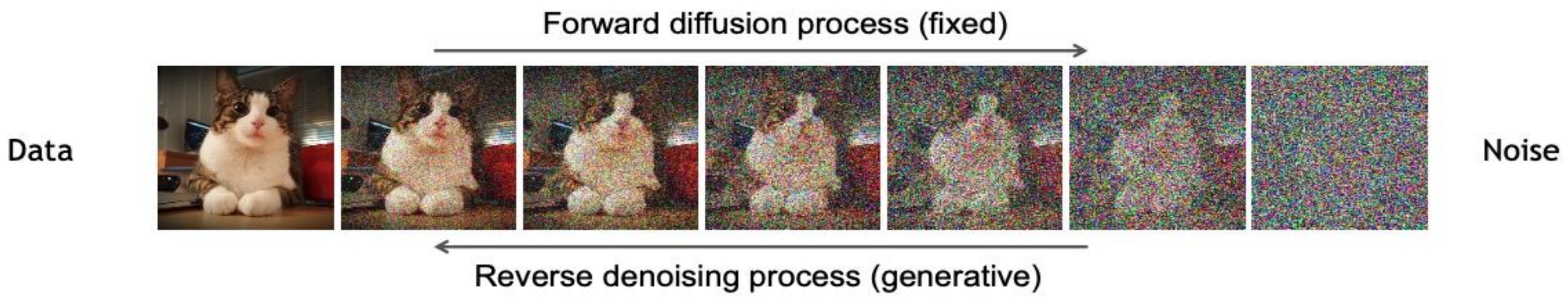
1. Go backward from noises and do denoising at each step
2. Do shrinking and adding noises to clean images to obtain their noisy versions

Question 15

- Matching the forward and backward processes of diffusion models (DMs) to?

A. Forward process of DMs
B. Backward process of DMs

1. Go backward from noises and do denoising at each step
2. Do shrinking and adding noises to clean images to obtain their noisy versions



Question 16

- What are correct about the forwarding process of DMs?

A. $x_t = \sqrt{1-\beta_t}x_{t-1} + \sqrt{\beta_t}\epsilon$ where $\epsilon \sim N(0, \mathbb{I})$

B. $x_{t-1} = \sqrt{1-\beta_t}x_t + \sqrt{\beta_t}\epsilon$ where $\epsilon \sim N(0, \mathbb{I})$

C. $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$ where $\epsilon \sim N(0, \mathbb{I})$ and $\bar{\alpha}_t = \prod_{i=1}^{t}(1-\beta_i)$

D. $x_t = \sqrt{\bar{\alpha}_t}\epsilon + \sqrt{1-\bar{\alpha}_t}x_0$ where $\epsilon \sim N(0, \mathbb{I})$ and $\bar{\alpha}_t = \prod_{i=1}^{t}(1-\beta_i)$

Question 16

- What are correct about the forwarding process of DMs?

A. $x_t = \sqrt{1-\beta_t}x_{t-1} + \sqrt{\beta_t}\epsilon$ where $\epsilon \sim N(0, \mathbb{I})$ ✓

B. $x_{t-1} = \sqrt{1-\beta_t}x_t + \sqrt{\beta_t}\epsilon$ where $\epsilon \sim N(0, \mathbb{I})$

C. $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$ where $\epsilon \sim N(0, \mathbb{I})$ and $\bar{\alpha}_t = \prod_{i=1}^{t}(1-\beta_i)$ ✓

D. $x_t = \sqrt{\bar{\alpha}_t}\epsilon + \sqrt{1-\bar{\alpha}_t}x_0$ where $\epsilon \sim N(0, \mathbb{I})$ and $\bar{\alpha}_t = \prod_{i=1}^{t}(1-\beta_i)$

Question 16

- What are correct about the U-Net $\epsilon_\theta(x_t, t)$ network of DMs?

A. We train this network by minimizing $||\epsilon_\theta(x_t, t) - \epsilon||_2^2$ where $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon$ with $\epsilon \sim N(0, \mathbb{I})$ and $\bar{\alpha}_t = \prod_{i=1}^{t}(1 - \beta_i)$

B. We train this network by maximizing $||\epsilon_\theta(x_t, t) - x_t||_2^2$ where $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon$ with $\epsilon \sim N(0, \mathbb{I})$ and $\bar{\alpha}_t = \prod_{i=1}^{t}(1 - \beta_i)$

C. This network aims to predict the Gaussian noise ϵ we added to a clean image x_0 from its corresponding noisy version $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon$

D. This network is crucial to denoise noises to gain clear/clean images

E. This network helps us forward a clean image to its noisy version.

Question 16

- What are correct about the U-Net $\epsilon_\theta(x_t, t)$ network of DMs?

A. We train this network by minimizing $||\epsilon_\theta(x_t, t) - \epsilon||_2^2$ where $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon$ with $\epsilon \sim N(0, \mathbb{I})$ and $\bar{\alpha}_t = \prod_{i=1}^{t}(1 - \beta_i)$ ✓

B. We train this network by maximizing $||\epsilon_\theta(x_t, t) - x_t||_2^2$ where $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon$ with $\epsilon \sim N(0, \mathbb{I})$ and $\bar{\alpha}_t = \prod_{i=1}^{t}(1 - \beta_i)$

C. This network aims to predict the Gaussian noise ϵ we added to a clean image x_0 from its corresponding noisy version $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon$ ✓

D. This network is crucial to denoise noises to gain clear/clean images ✓

E. This network helps us forward a clean image to its noisy version.

Thanks for your attention!