CS 236 Winter 2023
Deep Generative Models
Problem Set 3

November 27, 2023

SUNet ID: jchan7
Name: Jason Chan
Collaborators: None

By turning in this assignment, I agree by the Stanford honor code and declare that all of this is my own work.

1 Flow models

1.1

Answer. See flow_network.py for the forward function

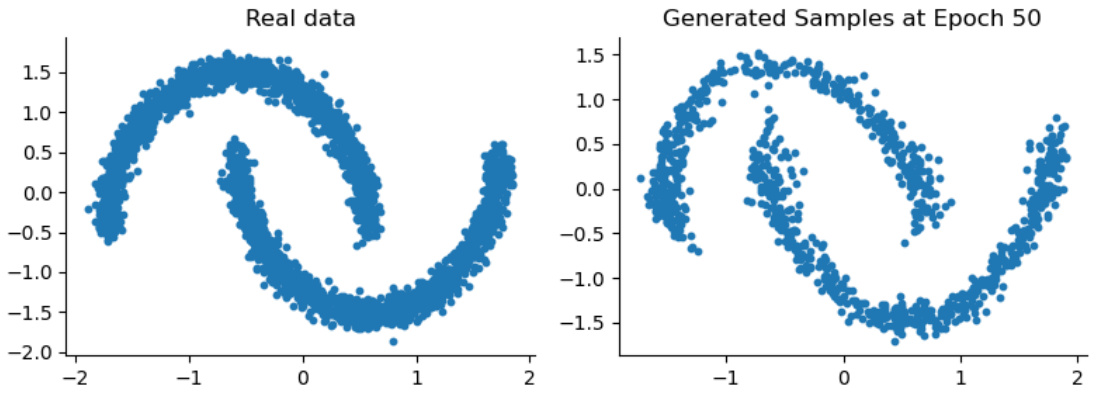
1.2

Answer. See flow_network.py for the inverse function

1.3

Answer. See flow_network.py for the log_probs function

1.4



2 Generative Adversarial Networks

2.1

Answer.

$$\begin{aligned}
\frac{\partial L^G_{\text{minimax}}}{\partial \theta} &= \mathbb{E}_{z\sim N(0,1)}\left[\frac{\partial}{\partial \theta}\log(1-\sigma(h_\phi(G_\theta(z))))\right] \\
&= \mathbb{E}_{z\sim N(0,1)}\left[\frac{\sigma'(h_\phi(G_\theta(z)))\frac{\partial}{\partial \theta}h_\phi(G_\theta(z))}{1-\sigma(h_\phi(G_\theta(z)))}\right] \\
&= \mathbb{E}_{z\sim N(0,1)}\left[\frac{\sigma(h_\phi(G_\theta(z)))(1-\sigma(h_\phi(G_\theta(z))))\frac{\partial}{\partial \theta}h_\phi(G_\theta(z))}{1-\sigma(h_\phi(G_\theta(z)))}\right] \\
&= \mathbb{E}_{z\sim N(0,1)}\left[\sigma(h_\phi(G_\theta(z)))\frac{\partial}{\partial \theta}h_\phi(G_\theta(z))\right] \\
&= \mathbb{E}_{z\sim N(0,1)}\left[D_\phi(G_\theta(z))\frac{\partial}{\partial \theta}h_\phi(G_\theta(z))\right].
\end{aligned}$$

$D_\phi(G_\theta(z)) \approx 0$ is problematic for training the generator because this will make the $\frac{\partial L^G_{\text{minimax}}}{\partial \theta}$ also approximate zero as shown above. When the discriminator classifies an image as fake, it will be slower to train since $\frac{\partial L^G_{\text{minimax}}}{\partial \theta}$ corresponds to smaller steps.

2.2

Answer. See gan.py for loss_nonsaturating_g and loss_nonsaturating_d functions. Figure below is the result after training for one epoch on Fashion MNIST with non-saturating losses.

3 Divergence minimization

3.1

Answer. The GAN loss function is

$$\begin{aligned} L_D(\phi;\theta) &= -\mathbb{E}_{x\sim p_{\text{data}}(x)}[\log D_\phi(x)] - \mathbb{E}_{x\sim p_\theta(x)}[\log(1-D_\phi(x))] \\ &= -\int p_{\text{data}}(x)\log D_\phi(x)\,dx - \int p_\theta(x)\log(1-D_\phi(x))\,dx \\ &= \int -p_{\text{data}}(x)\log D_\phi(x) \;\; - p_\theta(x)\log(1-D_\phi(x))\,dx \end{aligned}$$

Examining the provided hint, the expression for t that minimizes $f(t) = -p_{data}(x)\log t - p_\theta(x)\log(1-t)$ is found by solving for t when the derivative of $\frac{\partial f(t)}{\partial t} = 0$

$$\begin{aligned} \frac{\partial f(t)}{\partial t} &= -\frac{p_{\text{data}}(x)}{t} + \frac{p_\theta(x)}{1-t} = 0 \\ (1-t)p_{\text{data}}(x) &= t p_\theta(x) \\ t &= \frac{p_{\text{data}}(x)}{p_\theta(x) + p_{\text{data}}(x)} \end{aligned}$$

We can see similarities between the GAN loss function and the expression $f(t) = -p_{data}(x)\log t - p_\theta(x)\log(1-t)$. If we substitute $D_\phi(x)$ for t then we get the expression inside the integral for the GAN loss function. We can also say that the loss function can be minimised when the term inside the integral is minimised for every value of dx. This is analogous to finding the optimal value for t. Thus the GAN loss function is minimized when

$$D_\phi(x) = \frac{p_{\text{data}}(x)}{p_\theta(x) + p_{\text{data}}(x)}$$

3.2

Answer.

$$\sigma(h_\phi(x)) = \frac{1}{1+e^{-h_\phi(x)}} = D_\phi(x)$$

From the previous question, $D* = \frac{p_{\text{data}}(x)}{p_\theta(x)+p_{\text{data}}(x)}$ so we can equate this to the sigmoid function above.

$$\begin{aligned}
\frac{p_\theta(x) + p_{\text{data}}(x)}{p_{\text{data}}(x)} &= \frac{1}{1 + e^{-h_\phi(x)}} \\
e^{-h_\phi(x)} &= \frac{p_\theta(x) + p_{\text{data}}(x)}{p_{\text{data}}(x)} - 1 \\
&= \frac{p_\theta(x)}{p_{\text{data}}(x)} \\
h_\phi(x) &= \log \frac{p_{data}(x)}{p_\theta(x)}
\end{aligned}$$

3.3

Answer.

$$\begin{aligned}
L_G(\theta; \phi) &= \mathbb{E}_{x\sim p_\theta(x)}[\log(1 - D_\phi(x))] - \mathbb{E}_{x\sim p_\theta(x)}[\log D_\phi(x)] \\
&= \mathbb{E}_{x\sim p_\theta(x)}\left[\log \frac{(1 - D_\phi(x))}{D_\phi(x)}\right] \\
&= \mathbb{E}_{x\sim p_\theta(x)}\left[\log \frac{p_\theta(x)}{p_{\text{data}}(x)}\right] \\
&= \text{KL}(p_\theta(x)||p_{\text{data}}(x))
\end{aligned}$$

3.4

Answer.

$$\begin{aligned}
-\mathbb{E}_{x\sim p_{\text{data}}(x)}[\log p_\theta(x)] + \mathbb{E}_{x\sim p_{\text{data}}(x)}[\log p_{\text{data}}(x)] &= \mathbb{E}_{p_\theta(x)}\left[\frac{p_{\text{data}}(x)}{p_\theta(x)}\right] \\
&= \text{KL}(p_{\text{data}}(x)||p_\theta(x)).
\end{aligned}$$

But this is not the same as the KL divergence shown above because KL divergence is not symmetric. Therefore the objectives aren't the same.

4 Conditional GAN with Projection Discriminator

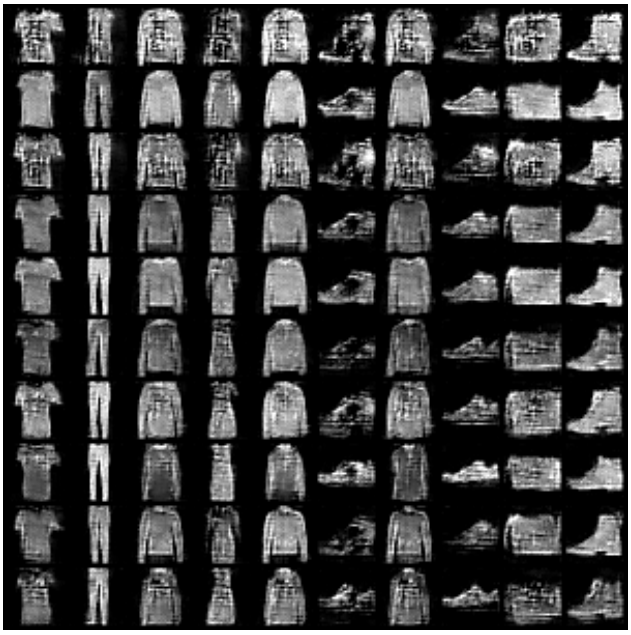
4.1

Answer.

$$
\begin{aligned}
h_\phi(x,y) &= \log \frac{p_{\text{data}}(x,y)}{p_\theta(x,y)} \\
&= \log \frac{p_{\text{data}}(x|y)p_{\text{data}}(y)}{p_\theta(x|y)p_\theta(y)} \\
&= \log \frac{p_{\text{data}}(x|y)}{p_\theta(x|y)} + \log \frac{p_{\text{data}}(y)}{p_\theta(y)} \\
&= -\frac{1}{2}\|\varphi(x) - \mu_y\|^2 + \frac{1}{2}\|\varphi(x) - \hat{\mu}_y\|^2 \\
&= -\frac{1}{2}(\varphi(x)^T\varphi(x) - 2\mu_y^T\varphi(x) + \mu_y^T\mu_y) \\
&\quad + \frac{1}{2}(\varphi(x)^T\varphi(x) - 2\hat{\mu}_y^T\varphi(x) + \hat{\mu}_y^T\hat{\mu}_y) \\
&= \mu_y^T\varphi(x) - \hat{\mu}_y^T\varphi(x) + \frac{1}{2}(\hat{\mu}_y^T\hat{\mu}_y - \mu_y^T\mu_y) \\
&= (\mu_y - \hat{\mu}_y)^T\varphi(x) + \frac{1}{2}(\|\hat{\mu}_y\|^2 - \|\mu_y\|^2) \\
&= \begin{pmatrix} (\mu_1 - \hat{\mu}_1)^T \\ \vdots \\ (\mu_m - \hat{\mu}_m)^T \end{pmatrix} \varphi(x) + \frac{1}{2}\begin{pmatrix} \|\mu_1\|^2 - \|\hat{\mu}_1\|^2 \\ \vdots \\ \|\mu_m\|^2 - \|\hat{\mu}_m\|^2 \end{pmatrix} \\
&= y^T(\mu - \hat{\mu})^T\varphi(x) + \frac{1}{2}y^T(\|\mu\|^2 - \|\hat{\mu}\|^2)
\end{aligned}
$$

4.2

Answer. See gan.py for conditional loss nonsaturating g and conditional loss nonsaturating d functions. Figure below is the result after training for one epoch on Fashion MNIST with conditional non-saturating losses. Columns correspond to categories.



5 Wasserstein GAN

5.1

Answer.

$$\begin{aligned}
\mathrm{KL}(p_\theta(x) \parallel p_{\text{data}}(x)) &= \mathbb{E}_{x\sim\mathcal{N}(\theta,\epsilon^2)}\left[\log \frac{p_\theta(x)}{p_{data}(x)}\right] \\
&= \mathbb{E}_{x\sim\mathcal{N}(\theta,\epsilon^2)}\left[\log p_\theta(x) - \log p_{data}(x)\right] \\
&= \mathbb{E}_{x\sim\mathcal{N}(\theta,\epsilon^2)}\left[\log \exp\left(-\frac{1}{2\epsilon^2}(x-\theta)^2\right) - \log \exp\left(-\frac{1}{2\epsilon^2}(x-\theta_0)^2\right)\right] \\
&= \mathbb{E}_{x\sim\mathcal{N}(\theta,\epsilon^2)}\left[\frac{1}{2\epsilon^2}\left(-(x-\theta)^2 + (x-\theta_0)^2\right)\right] \\
&= \mathbb{E}_{x\sim\mathcal{N}(\theta,\epsilon^2)}\left[\frac{1}{2\epsilon^2}\left(2x\theta - 2x\theta_0 - \theta^2 + \theta_0^2\right)\right] \\
&= \frac{1}{2\epsilon^2}\left(2\theta^2 - 2\theta\theta_0 - \theta^2 + \theta_0^2\right) \\
&= \frac{1}{2\epsilon^2}\left(\theta^2 - 2\theta\theta_0 + \theta_0^2\right) \\
&= \frac{1}{2\epsilon^2}(\theta - \theta_0)^2
\end{aligned}$$

5.2

Answer. When $p_\theta(x)$ and $p_{\text{data}}(x)$ both represent distributions that are narrowly concentrated, as we consider the limit where $\epsilon \to 0$, the behavior of the Kullback-Leibler divergence $\mathrm{KL}(p_\theta(x)\|p_{\text{data}}(x))$, and its corresponding derivative in terms of θ, is that they tend towards infinity, given θ is distinct from θ_0. This presents a challenge in the training process of Generative Adversarial Networks (GANs) with the loss function L_G as defined in the earlier problem 2.3. The generator is then at risk of encountering gradients of enormous magnitude if the discriminator achieves optimal training, which can lead to erratic training behavior and potentially to the divergence of the training process.

5.3

Answer. In situations where θ does not equal θ_0, the loss function L_D diverges negatively as $D_\phi(\theta)$ approaches negative infinity or $D_\phi(\theta_0)$ moves towards positive infinity. Hence, a discriminator D_ϕ that minimizes L_D does not exist in this scenario.

5.4

Answer. Assuming D_ϕ has a derivative constrained between -1 and 1, the optimal discriminator function $D_\phi(x)$, which minimizes L_D, would be a linear function. This function would have a gradient of ± 1, forming a connection between the points (θ_0, c) and $(\theta, c - |\theta - \theta_0|)$. Here, c denotes the value of D_ϕ at θ_0, ensuring the smallest possible value of $D_\phi(\theta)$ within the specified derivative bounds.

5.5

Answer. See gan.py for loss_wasserstein_gp_g and conditional loss_wasserstein_gp_d functions. Figure below is the result after training for one epoch on Fashion MNIST with wasserstein losses.