Due Date: Dec 8th (23:00 ET), 2023

Instructions

- *For all questions, show your work!*
- *Use LaTeX and the template we provide when writing your answers. You may reuse most of the notation shorthands, equations and/or tables. See the assignment policy on the course website for more details.*
- *The use of AI tools like Chat-GPT to find answers or parts of answers for any question in this assignment is not allowed. However, you can use these tools to improve the quality of your writing, like fixing grammar or making it more understandable. If you do use these tools, you must clearly explain how you used them and which questions or parts of questions you applied them to. Failing to do so or using these tools to find answers or parts of answers may result in your work being completely rejected, which means you'll receive a score of 0 for the entire theory or practical section.*
- *Submit your answers electronically via Gradescope.*
- *TAs for this assignment are* ***Thomas Jiralerspong, Sahar Dastani, and Shuo Zhang.***

Question 1 (5-5-5-5)**.** (**Autoregressive Models**)

One way to enforce autoregressive conditioning is via masking the weight parameters. [1] Consider a two-hidden-layer convolutional neural network without kernel flipping, with kernel size 3×3 and padding size 1 on each border (so that an input feature map of size 5×5 is convolved into a 5×5 output). Define mask of type A and mask of type B as

$$(\boldsymbol{M}^A)_{::ij} := \begin{cases} 1 & \text{if } i < 2 \\ 1 & \text{if } i = 2 \text{ and } j < 2 \\ 0 & \text{elsewhere} \end{cases} \qquad (\boldsymbol{M}^B)_{::ij} := \begin{cases} 1 & \text{if } i < 2 \\ 1 & \text{if } i = 2 \text{ and } j \leq 2 \\ 0 & \text{elsewhere} \end{cases}$$

where the index starts from 1. Masking is achieved by multiplying the kernel with the binary mask (elementwise). Specify the receptive field of the output pixel that corresponds to the third row and the third column (index 33 of Figure 1 (Left)) in each of the following 4 cases:

11	12	13	14	15
21	22	23	24	25
31	32	33	34	35
41	42	43	44	45
51	52	53	54	55

11	12	13	14	15
21	22	23	24	25
31	32	33	34	35
41	42	43	44	45
51	52	53	54	55

FIGURE 1 – (Left) 5×5 convolutional feature map. (Right) Template answer.

1. If we use $\boldsymbol{M}^A$ for the first layer and $\boldsymbol{M}^A$ for the second layer.
2. If we use $\boldsymbol{M}^A$ for the first layer and $\boldsymbol{M}^B$ for the second layer.

1. An example of this is the use of masking in the Transformer architecture.

3. If we use $\boldsymbol{M}^B$ for the first layer and $\boldsymbol{M}^A$ for the second layer.
4. If we use $\boldsymbol{M}^B$ for the first layer and $\boldsymbol{M}^B$ for the second layer.

Your answer should look like Figure 1 (Right).

Answer 1. Based on questions's assumptions we have two Masks, A:

1	1	1
1	0	0
0	0	0

And B:

1	1	1
1	1	0
0	0	0

so based on that and receptive field, we have:

Q1.1)

11	12	13	14	15
21	22	23	24	25
31	32	33	34	35
41	42	43	44	45
51	52	53	54	55

Q1.2)

11	12	13	14	15
21	22	23	24	25
31	32	33	34	35
41	42	43	44	45
51	52	53	54	55

Q1.3)

11	12	13	14	15
21	22	23	24	25
31	32	33	34	35
41	42	43	44	45
51	52	53	54	55

Q1.4)

11	12	13	14	15
21	22	23	24	25
31	32	33	34	35
41	42	43	44	45
51	52	53	54	55

Question 2 (5-5)**.** (**Normalizing Flows**) In this question, we study some properties of normalizing flows. Let $X \sim P_X$ and $U \sim P_U$ be, respectively, the distribution of the data and a base distribution (e.g. an isotropic gaussian). We define a normalizing flow as $F : \mathcal{U} \to \mathcal{X}$ parametrized by $\boldsymbol{\theta}$. Starting with P_U and then applying F will induce a new distribution $P_{F(U)}$ (used to match P_X). Since normalizing flows are invertible, we can also consider the distribution $P_{F^{-1}(X)}$.

However, some flows, like planar flows, are not easily invertible in practice. If we use P_U as the base distribution, we can only sample from the flow but not evaluate the likelihood. Alternatively, if we use P_X as the base distribution, we can evaluate the likelihood, but we will not be able to sample.

2.1 Show that $D_{KL}[P_X||P_{F(U)}] = D_{KL}[P_{F^{-1}(X)}||P_U]$. In other words, the forward KL divergence between the data distribution and its approximation can be expressed as the reverse KL divergence between the base distribution and its approximation.

2.2 Suppose two scenarios: 1) you don't have samples from $p_X(\boldsymbol{x})$, but you can evaluate $p_X(\boldsymbol{x})$, 2) you have samples from $p_X(\boldsymbol{x})$, but you cannot evaluate $p_X(\boldsymbol{x})$. For each scenario, specify if you would use the forward KL divergence $D_{KL}[P_X||P_{F(U)}]$ or the reverse KL divergence $D_{KL}[P_{F(U)}||P_X]$ as the objective to optimize. Justify your answer.

Answer 2. 2.1) As we know, the Kullback-Leibler divergence from distribution P to distribution Q for a continuous random variable X is given by:

$$D_{KL}(P \parallel Q) = \int_{-\infty}^{\infty} p(x) \log \left(\frac{q(x)}{p(x)} \right) dx$$

To demonstrate the necessary equality, the change of variables formula for probability densities can be employed. In the case of:

$$F : U \to X$$

which represents a normalizing flow in this scenario, the density of

$$Y = F(X)$$

is determined as follows:

$$p_Y(y) = p_X(F^{-1}(y)) \left| \frac{\mathrm{d}y}{\mathrm{d}F^{-1}(y)} \right|$$

Here, $p_Y(y)$ denotes the density of the transformed random variable via the flow transformation.

$p_X(x)$ represents the density of the original random variable X. The term $\frac{\mathrm{d}F^{-1}(y)}{\mathrm{d}y}$ signifies the derivative of the inverse function F^{-1} with respect to y, commonly referred to as the Jacobian determinant of the inverse function F.
We aim to demonstrate the following relationship using the change of variables formula in probability:

$$D_{KL}(P_X \parallel P_{F(U)}) = D_{KL}(P_{F^{-1}(x)} \parallel P_U)$$

To establish this, we consider:

$$D_{KL}(P_X \parallel P_{F(U)}) = \int p_X(x) \log \left(\frac{p_X(x)}{p_F(f(x))} \right) dx$$

Additionally, we have:

$$D_{KL}(P_{F^{-1}(x)} \parallel P_U) = \int p_{F^{-1}(x)} \log \left(\frac{p_{F^{-1}(x)}}{p_U(u)} \right) dx$$

By inserting the substitution $p_{F^{-1}(x)} = p_U(F(x)) \left| \frac{\mathrm{d}F(x)}{\mathrm{d}x} \right|$ into the second integral and changing the integration variable from x to $u = F(x)$, we reframe the integral involving $p_{F^{-1}(x)}$ as an integral over $p_U(u)$. This step utilizes the relationship between $p_X(x)$ and $p_U(u)$, connected by the transformation F and its Jacobian. Assuming that F is a bijection, and $P_{F(U)}$ and $P_{F^{-1}(x)}$ represent the respective transformed distributions, the equivalence of these integrals demonstrates the desired equality.

2.2.1): In this scenario, our optimization objective will be the reverse Kullback-Leibler divergence,

$$D_{KL}(P_{F^{-1}(x)} \parallel P_U)$$

.
This choice is appropriate because the reverse KL divergence necessitates the evaluation of the data distribution's probability density, $p_X(x)$, which is feasible in our case. The reverse KL divergence is particularly advantageous in scenarios where avoiding areas of zero probability in the true distribution, $p_X(x)$, is crucial. It imposes substantial penalties on the model $P_{F^{-1}(x)}$ in regions where $p_X(x)$ is zero, ensuring a more robust model fitting.

2.2.2): In this scenario, the optimization objective in this context is the forward Kullback-Leibler divergence:

$$D_{KL}(P_X \parallel P_{F(U)})$$

Given the inability to directly evaluate $p_X(x)$, the reverse KL divergence, which depends on assessing the density $p_X(x)$, is not computationally feasible. Conversely, the forward KL divergence can be estimated using samples from $p_X(x)$ by maximizing the likelihood of these samples under the model distribution $P_{F(U)}$. This approach is more practical when only samples are available, as it does not necessitate the normalization of the true distribution. The forward KL treats the true

distribution as fixed and integrates over the model distribution, making the computation process generally more straightforward.

Note: I used Grammerly and ChatGPT to improve the quality of my writing.

Question 3 (3-8-3-14)**.** (**Variational Autoencoders**)

1. Let $p_x^*(.)$ be the true data distribution and $p_x(.;\theta)$ be the model distribution parametrized over θ, a natural criterion to define if $p_x(.;\theta)$ is accurately portraying $p_x^*(.)$ is the *Maximum Likelihood Estimation* (MLE). Sometimes, knowledge about the data can lead us to adopt a model with hidden intermediate variable z to approximate the data distribution, where only the joint distribution $p_{x,z}(.,.,\theta)$ are explicitly defined. For such models, we need to calculate the marginal likelihood $p_x(.) = \int_z p_{x,z}(.,z,\theta)dz$, however, this proves to be difficult. Why ?
   (a) We do not know about $p_(x|z)$ and thus cannot calculate the integral.
   (b) Integration over the hidden variable z can prove to be intractable due to the complexity of $p_(x|z)$ and the curse of dimensionality.
   (c) We don't know and cannot assume what z looks like (i.e. what kind of distribution) and thus cannot calculate the integral.
   (d) The integral over the hidden variable z is intractable because it does not follow a standard distribution like Gaussian or Bernoulli.
2. To avoid the above problem, we can try to avoid $p_x(.)$ and instead aim to establish a lower bound function of it. This involves rewriting the log of the marginal likelihood $\log p_x(.) = \log \int_z p_{x,z}(.,z,\theta)dz$ as a combination of a KL divergence and an *Evidence Lower Bound* (ELBO). This process is facilitated by the introduction of an approximate posterior $q(z|x)$ which approximates the unknown true posterior $p(z|x)$. The choice of q is arbitrary, but we often choose it from simpler classes of distributions such as the Gaussian for practical reasons. Your task is to derive the ELBO function in two ways:
   (a) By decomposing the marginal likelihood as the combination of a KL-divergence between variational and true posteriors over z ($D_{KL}(q(z|x)||p(z|x))$) and the ELBO.
   (b) By using the Jensen Inequality.
3. What is the significance of the above result ? Select all that apply.
   (a) $p_x(.)$ has a lower bound which is the ELBO.
   (b) Maximizing the ELBO is equivalent to minimizing the distributional difference between the approximation $q(z|x)$ and the true (but intractable) $p(z|x)$.
   (c) The ELBO offers a theoretical bound but is not useful in practice for training models with latent variables.
   (d) The choice of q affects the tightness of the lower bound.
4. This question is about importance weighted autoencoder. When training a variational autoencoder, the standard training objective is to maximize the evidence lower bound (ELBO). Here we consider another lower bound, called the Importance Weighted Lower Bound (IWLB), a tighter bound than ELBO, defined as

$$\mathcal{L}_k = \mathbf{E}_{z_{1:k}\sim q(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{1}{k}\sum_{j=1}^{k}\frac{p(\boldsymbol{x}, z_j)}{q(z_j \mid \boldsymbol{x})}\right]$$

for an observed variable $\boldsymbol{x}$ and a latent variable $\boldsymbol{z}$, k being the number of importance samples. The model we are considering has joint that factorizes as $p(\boldsymbol{z}, \boldsymbol{x}) = p(\boldsymbol{x} \mid \boldsymbol{z})p(\boldsymbol{z})$, $\boldsymbol{x}$ and $\boldsymbol{z}$ being the observed and latent variables, respectively. In the following questions, one needs to make use of the Jensen's inequality:

$$f(\mathbf{E}[X]) \leq \mathbf{E}[f(X)]$$

for a convex function f.

(a) Show that IWLB is a lower bound on the log likelihood $\log p(\boldsymbol{x})$.
(b) Given a special case where $k = 2$, prove that $\mathcal{L}_2$ is a tighter bound than the ELBO (with $k = 1$).

Answer 3. 3.1.a): is incorrect because, the probability $p(x \mid z)$ is generally established and recognized. This term represents the likelihood of observing the data, given the latent variables, a relationship that is delineated by the model's structure.

3.1.b): is correct because, the primary challenge encountered here stems from the complexities involved in integrating over z. This difficulty is largely due to the high-dimensional characteristics of the data and the complex interplay between the observed and latent variables.

3.1.c): is incorrect because, assumptions about zz are indeed made. Typically, zz is presumed to follow a standard distribution, like a Gaussian distribution, which aids in implementing the reparameterization trick.

3.1.d): is incorrect because,the assumption that zz adheres to a standard distribution, often Gaussian. This assumption is crucial for enabling calculations using the reparameterization trick.

3.2.a) As we konw, the Evidence Lower Bound (ELBO) is formulated to provide a feasible objective when the actual posterior $p(z \mid x)$ is not computationally manageable. The task at hand involves deriving the ELBO by decomposing the marginal likelihood into a mix of a KL-divergence term and the ELBO itself. Here's an outline of the ELBO derivation:
We can consider the marginal likelihood:

$$\log p_x(.) = \log \int p_{x,z}(., z, \theta)\, dz$$

We introduce an approximate posterior $q(z \mid x)$ and reformulate the log of the marginal likelihood using the KL divergence:

$$\log p_x(x) = D_{KL}(q(z \mid x) \parallel p(z \mid x)) + L(q)$$

Here, $L(q)$ denotes the ELBO. To demonstrate this, consider the identity:

$$D_{KL}(q(z \mid x) \parallel p(z \mid x)) = \mathbb{E}_{q(z|x)}[\log q(z \mid x) - \log p(z \mid x)]$$

Given that the KL divergence is always non-negative, it follows that the ELBO serves as a lower bound of the log-likelihood:

$$\log p_x(x) \geq L(q)$$

Where:

$$L(q) = \mathbb{E}_{q(z|x)}[\log p_x(x \mid z)] - D_{KL}(q(z \mid x) \parallel p(z))$$

The first term represents the expected log-likelihood under the approximate posterior, while the second term, the KL divergence between the approximate posterior and the latent variables' prior, functions as a regularization component. Maximizing the ELBO relative to the variational parameters (of $q(z \mid x)$) aims to optimize the bound's tightness. Practically, this means the approximate posterior becomes a closer approximation to the actual posterior.

3.2.b) In the realm of variational inference and the Evidence Lower Bound (ELBO), Jensen's Inequality plays a pivotal role in establishing a lower bound for the logarithm of the marginal likelihood, which is essentially the ELBO. To conceptualize how Jensen's Inequality facilitates the derivation of the ELBO, consider the following: Given the concavity of the logarithm function, Jensen's Inequality implies that for a random variable Z and a distribution $q(z)$, the inequality below is valid:

$$\log \mathbb{E}_q[Z] \geq \mathbb{E}_q[\log Z]$$

In the case of the marginal likelihood $p_x(x)$:

$$\log p_x(x) = \log \int p_x(x|z)p(z)\, dz$$

This integral is complex to compute directly. However, by sampling z from a distribution $q(z|x)$ rather than $p(z)$, we can express it as:

$$\log p_x(x) = \log \mathbb{E}_{q(z|x)} \left[\frac{p_x(x|z)p(z)}{q(z|x)} \right]$$

Utilizing Jensen's Inequality and acknowledging the concavity of the logarithm:

$$\log p_x(x) \geq \mathbb{E}_{q(z|x)} \left[\log \frac{p_x(x|z)p(z)}{q(z|x)} \right]$$

This expected value constitutes the ELBO:

$$\mathbb{E}_{q(z|x)}[\log p_x(x|z)] - \mathbb{E}_{q(z|x)}[\log \frac{q(z|x)}{p(z)}]$$

Commonly represented as:

$$L(q) = \mathbb{E}_{q(z|x)}[\log p_x(x|z)] - D_{KL}(q(z|x) \parallel p(z))$$

This derived ELBO can be maximized relative to the parameters of $q(z|x)$ to closely approximate the elusive $p(z|x)$. This concept is central to variational inference and the training of VAEs, where the ELBO serves as a crucial objective function maximized during training.

3.3.a) True - The marginal likelihood $p_x(.)$ indeed has a lower bound, the ELBO, a cornerstone in variational inference. The ELBO serves as a practical surrogate for the actual log likelihood, which is often computationally unfeasible.
3.3.b) True - Optimizing the ELBO effectively reduces the discrepancy between the approximate posterior $q(z|x)$ and the actual posterior $p(z|x)$. This is attributable to the ELBO encompassing a term that is the negative KL divergence between $q(z|x)$ and $p(z|x)$, where minimizing this term aligns with reducing the KL divergence.
3.3.c) False - The ELBO transcends theoretical importance, being vital in practical applications. It is the key objective in training models such as VAEs, especially when direct log likelihood optimization is not viable.
3.3.d) True - The selection of the variational distribution q significantly influences the ELBO's effectiveness. The form of $q(z|x)$ determines how closely the ELBO approximates the true log likelihood. A more expressive q can lead to a tighter bound and a more accurate posterior approximation.

3.4.a) Starting with the definition of the Importance-Weighted Lower Bound (IWLB), we have:

$$L_k = E_{z^{1:k}\sim q(z|x)}\left[\log\left(\frac{1}{k}\sum_{j=1}^{k}\frac{p(x,z_j)}{q(z_j|x)}\right)\right]$$

We aim to demonstrate that this serves as a lower bound for $\log p(x)$. Given that $p(x) = \int p(x,z)dz$, we can express:

$$\log p(x) = \log \int p(x,z)dz = \log \int \frac{p(x,z)}{q(z|x)}q(z|x)dz$$

Now, we apply Jensen's Inequality to the convex function $f = -\log$ to this expectation, considering $-\log$ (which is convex):

$$\log p(x) = -\log E_{q(z|x)}\left[\frac{1}{p(x,z)q(z|x)}\right] \geq -E_{q(z|x)}\left[\log\left(\frac{1}{p(x,z)q(z|x)}\right)\right]$$

Simplifying the right side, we obtain:

$$\log p(x) \geq E_{q(z|x)}\left[\log\left(\frac{p(x,z)}{q(z|x)}\right)\right]$$

This is the ELBO for a single sample. IWLB generalizes this by taking the average over kk samples instead of just one.

3.4.b) To demonstrate this, we need to compare L_1 (which represents the Evidence Lower Bound or ELBO) with L_2. For $k = 1$, L_1 is simply the ELBO, as previously explained. For $k = 2$, we have:

$$L_2 = E_{z^{1:2}\sim q(z|x)}\left[\log\left(\frac{1}{2}\sum_{j=1}^{2}\frac{p(x,z_j)}{q(z_j|x)}\right)\right]$$

When applying Jensen's Inequality for $k = 2$, we consider that the function inside the expectation is the average of two independent and identically distributed random variables. The variance of this average is lower than the variance of a single sample. A lower variance in the function inside the logarithm results in a higher expectation of the logarithm due to the concavity of the logarithm function. Therefore, L_2 is more tightly bounded than L_1, indicating that it provides a superior (closer to the true $\log p(x)$) lower bound.

Note: I used Grammerly and ChatGPT to improve the quality of my writing.

Question 4 (2-2-2-3-3-10). (**Generative Adversarial Networks**)

1. Consider a Generative Adversarial Network (GAN) which successfully produces images of apples. Which of the following propositions is false ?
   (a) The generator aims to learn the distribution of apple images.
   (b) The discriminator can be used to classify images as apple vs. non-apple.
   (c) After training the GAN, the discriminator loss eventually reaches a constant value.
   (d) The generator can produce unseen images of apples.
2. Which of the following cost functions is the non-saturating cost function for the generator in GANs (G is the generator and D is the discriminator) ? Note that the cost function will be minimized w.r.t the generator parameters during training.
   (a) $J^{(G)} = \frac{1}{m}\sum_{i=1}^{m} \log(1 - D(G(z^{(i)})))$
   (b) $J^{(G)} = -\frac{1}{m}\sum_{i=1}^{m} \log(D(G(z^{(i)})))$
   (c) $J^{(G)} = \frac{1}{m}\sum_{i=1}^{m} \log(1 - G(D(z^{(i)})))$
   (d) $J^{(G)} = -\frac{1}{m}\sum_{i=1}^{m} \log(G(D(z^{(i)})))$
3. After training a neural network, you observe a large gap between the training accuracy (100%) and the test accuracy (42%). Which of the following methods is commonly used to reduce this gap ?
   (a) Generative Adversarial Networks
   (b) Dropout
   (c) Sigmoid activation
   (d) RMSprop optim
4. Given the two options of (A) saturating cost and (B) non-saturating cost, which cost function would you choose to train a GAN ? Explain your reasoning. (1-2 sentences)
5. You are training a standard GAN, and at the end of the first epoch you take note of the values of the generator and discriminator losses. At the end of epoch 100, the values of the loss functions are approximately the same as they were at the end of the first epoch. Why are the quality of generated images at epoch 1 and epoch 100 not necessarily similar ? (1-2 sentences)
6. Let p_0 and p_1 be two probability distributions with densities f_0 and f_1 (respectively). We want to explore what we can do with a trained GAN discriminator. A trained discriminator is thought to be one which is "close" to the optimal one:

$$D^* := \arg\max_D \mathbb{E}_{\boldsymbol{x}\sim p_1}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{x}\sim p_0}[\log(1 - D(\boldsymbol{x}))].$$

(a) For the first part of this problem, derive an expression we can use to estimate the Jensen-Shannon divergence (JSD) between p_0 and p_1 using a trained discriminator. We remind that the definition of JSD is $\text{JSD}(p_0, p_1) = \frac{1}{2}\big(KL(p_0\|\mu) + KL(p_1\|\mu)\big)$, where $\mu = \frac{1}{2}(p_0 + p_1)$.

(b) For the second part, we want to demonstrate that a optimal GAN Discriminator (i.e. one which is able to distinguish between examples from p_0 and p_1 with minimal NLL loss) can be used to express the probability density of a datapoint $\boldsymbol{x}$ under f_1, $f_1(\boldsymbol{x})$ in terms of $f_0(\boldsymbol{x})$ [2]. Assume f_0 and f_1 have the same support. Show that $f_1(\boldsymbol{x})$ can be estimated by $f_0(\boldsymbol{x})D(\boldsymbol{x})/(1 - D(\boldsymbol{x}))$ by establishing the identity $f_1(\boldsymbol{x}) = f_0(\boldsymbol{x})D^*(\boldsymbol{x})/(1 - D^*(\boldsymbol{x}))$.

Hint: Find the closed form solution for D^*.

Answer 4. 4.1.a) True. In GANs, the generator's role is to learn to produce data that is indistinguishable from real data, in this case, images of apples.

4.1.b) False. The discriminator in a GAN is trained to distinguish between real and generated data not apple or non-apple.

4.1.c) True, The ideal scenario in training a GAN is for the generator to become so good at generating images that the discriminator can't distinguish real from fake, which means the discriminator's loss would be around 0.5 (unable to distinguish)

4.1.d)True. The goal of the generator is to create new data instances that resemble the training data but are not copies of it. This means it should be able to generate new images of apples that it has never seen before.

4.2) the correct answer is b, This function is known as the non-saturating cost function used in GANs. Rather than focusing on minimizing the probability of the discriminator correctly identifying samples, this function aims to maximize the probability of the discriminator making incorrect identifications. This approach is advantageous, especially at the early stages of training, as it provides more substantial gradients when $D(G(z^{(i)}))$ is low. The larger logarithmic term during these stages ensures a more robust learning signal for the generator.

4.3) Among these options, **Dropout** (b) is the technique primarily utilized for reducing overfitting, thereby narrowing the gap between training and test accuracy. GANs, a class of algorithms in unsupervised machine learning involving two contesting neural networks, are not typically used for mitigating overfitting. On the other side, This activation function, used for modeling binary outcomes in neural networks, does not inherently address overfitting. While RMSprop, an adaptive learning rate method, enhances training efficiency, particularly in noisy or sparse gradient.

4.4) I would choose the non-saturating cost function (B) to train a GAN. This is because non-saturating cost functions provide stronger gradient signals to the generator, especially early in training, which helps to avoid the vanishing gradient problem and generally results in a more stable and efficient training process.

2. You might need to use the "functional derivative" to solve this problem. See "19.4.2 Calculus of Variations" of the Deep Learning book or "Appendix D Calculus of Variations" of Bishop's Pattern Recognition and Machine Learning for more information.

4.5) The similarity in loss values at epoch 1 and epoch 100 in a GAN doesn't necessarily equate to comparable image quality, as these loss functions primarily evaluate the relative performance between the generator and discriminator rather than the outright quality of the images. Additionally, the generator and discriminator may have undergone different learning and improvement trajectories over time, leading to enhanced image quality at epoch 100 despite similar loss values.

4.6.a) To estimate the Jensen-Shannon Divergence (JSD) between two probability distributions p_0 and p_1 utilizing a trained discriminator D, consider the following, JSD is articulated as:

$$\text{JSD}(p_0, p_1) = \frac{1}{2}\text{KL}(p_0||\mu) + \frac{1}{2}\text{KL}(p_1||\mu)$$

where:

$$\mu = \frac{1}{2}(p_0 + p_1)$$

The Kullback-Leibler divergence (KL divergence) quantifies the variance between two probability distributions. The KL divergence from p_0 to μ is:

$$\text{KL}(p_0||\mu) = \mathbb{E}_{x\sim p_0}\left[\log \frac{p_0(x)}{\mu(x)}\right]$$

The discriminator D aims to approximate 1 for samples from p_1 and 0 for samples from p_0. At its optimal state D^*, it fulfills:

$$D^*(x) = \frac{p_1(x)}{p_1(x) + p_0(x)}$$

Therefore, the KL divergences can be reformulated using D^* to estimate the JSD as:

$$\text{JSD}(p_0, p_1) = \frac{1}{2}\left(\mathbb{E}_{x\sim p_0}\left[\log \frac{p_0(x)}{p_0(x) + p_1(x)}\right] + \mathbb{E}_{x\sim p_1}\left[\log \frac{p_1(x)}{p_0(x) + p_1(x)}\right]\right)$$

$$\text{JSD}(p_0, p_1) = \frac{1}{2}\left(\mathbb{E}_{x\sim p_0}\left[\log \frac{1 - D^*(x)}{D^*(x)}\right] + \mathbb{E}_{x\sim p_1}\left[\log \frac{D^*(x)}{1 - D^*(x)}\right]\right)$$

4.6.b) D^* represents the optimal discriminator, tasked with distinguishing between data originating from two distributions characterized by density functions f_0 and f_1. The optimal discriminator, at its peak performance, is defined by the formula:

$$D^*(x) = \frac{f_1(x)}{f_0(x) + f_1(x)}$$

We aim to represent $f_1(x)$, the probability density of a data point x under f_1, using $f_0(x)$ and $D^*(x)$. Beginning with the definition of D^*, we can deduce $f_1(x)$ as:

$$D^*(x) = \frac{f_1(x)}{f_0(x) + f_1(x)}$$

To determine $f_1(x)$, we rearrange terms as follows:

$$D^*(x) \cdot (f_0(x) + f_1(x)) = f_1(x)$$

Expanding this, we get:

$$D^*(x) \cdot f_0(x) + D^*(x) \cdot f_1(x) = f_1(x)$$

Subtracting $D^*(x) \cdot f_1(x)$ from both sides leads to:

$$D^*(x) \cdot f_0(x) = f_1(x) - D^*(x) \cdot f_1(x)$$

Simplifying, we obtain:

$$D^*(x) \cdot f_0(x) = f_1(x) \cdot (1 - D^*(x))$$

Dividing both sides by $1 - D^*(x)$ yields:

$$\frac{D^*(x) \cdot f_0(x)}{1 - D^*(x)} = f_1(x)$$

Thus, $f_1(x)$ is expressed in terms of $f_0(x)$ and $D^*(x)$ as:

$$f_1(x) = \frac{f_0(x) \cdot D^*(x)}{1 - D^*(x)}$$

Note: I used Grammerly and ChatGPT to improve the quality of my writing.

Question 5 (5-5-5-5)**.** (**Self-Supervised Learning: Paper Review**)

In this question, you are going to write a **one page review** of the A Simple Framework for Contrastive Learning of Visual Representations paper.

Your review should have the following four sections: Summary, Strengths, Weaknesses, and Reflections. For each of these sections, below we provide a set of questions you should ask about the paper as you read it. Then, discuss your thoughts about these questions in your review.

(5.1) **Summary:**

(a) What is this paper about ?
(b) What is the main contribution ?
(c) Describe the main approach and results. Just facts, no opinions yet.

(5.2) **Strengths:**

(a) Is there a new theoretical insight ?
(b) Or a significant empirical advance ? Did they solve a standing open problem ?
(c) Or a good formulation for a new problem ?
(d) Any good practical outcome (code, algorithm, etc) ?
(e) Are the experiments well executed ?
(f) Useful for the community in general ?

(5.3) **Weaknesses:**

(a) What can be done better ?
(b) Any missing baselines ? Missing datasets ?
(c) Any odd design choices in the algorithm not explained well ? Quality of writing ?
(d) Is there sufficient novelty in what they propose ? Minor variation of previous work ?
(e) Why should anyone care ? Is the problem interesting and significant ?

(5.4) **Reflections:**

(a) How does this relate to other concepts you have seen in the class ?
(b) What are the next research directions in this line of work ?
(c) What (directly or indirectly related) new ideas did this paper give you ? What would you be curious to try ?

This question is subjective and so we will accept a variety of answers. You are expected to analyze the paper and offer your own perspective and ideas, beyond what the paper itself discusses.

Answer 5. SimCLR, a sophisticated framework for self-supervised learning of visual representations, which proposes an innovative departure from the traditionally complex mechanisms required for this task. By dispensing with the need for specialized architectures or extensive memory banks, SimCLR represents a paradigm shift towards a more streamlined approach in the field of contrastive learning. The core insight of SimCLR is its reliance on a carefully curated composition of simple data augmentations, which, as the paper convincingly demonstrates, can serve as a powerful substitute for more intricate self-supervised learning methods. The principal contribution of SimCLR lies in its empirical success—demonstrating through rigorous testing that a series of learnable, nonlinear transformations, when applied between the representation layers and the contrastive loss function, can yield significant performance enhancements. This is further bolstered by advocating for the use of larger batch sizes and an extended training regimen. The paper's experimental results are compelling, with SimCLR achieving a remarkable 76.5% top-1 accuracy on the ImageNet dataset. This not only marks a 7% relative improvement over the previous state-of-the-art but also serves to highlight the potential of SimCLR in addressing one of machine learning's most pressing challenges: learning visual representations without the crutch of human supervision. The paper's strengths are manifold. It begins with a solid theoretical foundation, presenting a persuasive argument that contrastive learning's complexity can be reduced without negatively impacting its efficacy. The framework's empirical advances are substantiated by robust experimentation, resulting in a significant leap in performance benchmarks. SimCLR proves itself not just in theory but as a practical tool, demonstrating versatility across various network architectures and improving upon existing methods in self-supervised and semi-supervised learning. The methodological rigor of the experiments underpins the framework's effectiveness, and the results have far-reaching implications for the machine learning community, particularly for those invested in the domain of unsupervised learning. Nonetheless, the paper does not exhaust the exploration of data augmentation strategies. There is room for a more expansive investigation into the effects of different augmentation combinations on the performance of the framework. Although the paper's scope is impressive, it might benefit from validation across a broader spectrum of datasets, which would strengthen the argument for SimCLR's generalizability. The paper could also delve deeper into explaining the algorithmic decisions, especially those concerning the selection and application of data augmentations, to provide readers with a more granular understanding of the framework's inner workings. From a novelty

perspective, while SimCLR builds on the established principles of contrastive learning, it manages to distinguish itself through its simplicity and superior performance, warranting recognition in the field. The paper validates the importance of the problem it seeks to solve automated learning of visual representations is crucial for the advancement of AI technologies. The reflections inspired by the paper extend beyond its immediate findings. SimCLR's approach resonates with current trends in unsupervised learning and echoes the educational drive towards minimizing the reliance on labeled data. The paper prompts questions about future research directions, particularly the potential application of the SimCLR framework to other types of data, such as audio or textual content, which could revolutionize the field of multimodal learning. Furthermore, the simplicity of the SimCLR model challenges the prevailing complexity in current models, inspiring a reassessment of whether simpler models could, in fact, be more beneficial in certain contexts. This opens up a fertile ground for experimentation, possibly leading to novel methodologies in domains where data is abundant, but labels are scarce or expensive to obtain.

Note: I used Grammerly and ChatGPT to improve the quality of my writing.