CS 236, Fall 2019
Midterm Exam

This exam is worth 110 points. You have 3 hours to complete it. You are allowed to consult notes, books, and use a laptop but no communication or network access is allowed. Good luck!

Stanford University Honor Code

The Honor Code is the University's statement on academic integrity written by students in 1921. It articulates University expectations of students and faculty in establishing and maintaining the highest standards in academic work:

- The Honor Code is an undertaking of the students, individually and collectively:
  - that they will not give or receive aid in examinations; that they will not give or receive unpermitted aid in class work, in the preparation of reports, or in any other work that is to be used by the instructor as the basis of grading;
  - that they will do their share and take an active part in seeing to it that others as well as themselves uphold the spirit and letter of the Honor Code.
- The faculty on its part manifests its confidence in the honor of its students by refraining from proctoring examinations and from taking unusual and unreasonable precautions to prevent the forms of dishonesty mentioned above. The faculty will also avoid, as far as practicable, academic procedures that create temptations to violate the Honor Code.
- While the faculty alone has the right and obligation to set academic requirements, the students and faculty will work together to establish optimal conditions for honorable academic work.

Signature

I attest that I have not given or received aid in this examination, and that I have done my share and taken an active part in seeing to it that others as well as myself uphold the spirit and letter of the Stanford University Honor Code.

Name / SUnetID:

Signature:

Question	Score	Question	Score
1	/ 14	5	/ 22
2	/ 6	6	/ 10
3	/ 8	7	/ 20
4	/ 12	8	/ 18

Total score: **/ 110**

Note: Partial credit will be given for partially correct answers. Zero points will be given to answers left blank.

---

1. **[14 points total] General Conceptual Questions**
   For each of the statements below, state True or False. Explain your answer for full points.

   (a) **[2 points]** Let p be a discrete probability distribution. Consider two autoregressive factorizations of p based on two different orderings of the variables, where the conditionals are represented in tabular format. Then these two factorizations always require the same number of parameters.

   (b) **[2 points]** Given any latent variable model $p_\theta(\boldsymbol{x}, \boldsymbol{z})$, there always exists an autoregressive model q_ψ such that $p_\theta(\boldsymbol{x}) = \prod_i q_\psi(\boldsymbol{x}_i \mid \boldsymbol{x}_{<i})$.

   (c) **[2 points]** Let p_{data} and q_{data} be two data distributions over the same domain $\mathcal{X}$, and p_θ a generative model. Suppose $E_{\boldsymbol{x}\sim p_{\text{data}}}[\log p_\theta(\boldsymbol{x})] > E_{\boldsymbol{x}\sim q_{\text{data}}}[\log p_\theta(\boldsymbol{x})]$. Then $D_{KL}(p_{\text{data}}||p_\theta) < D_{KL}(q_{\text{data}}||p_\theta)$.

   (d) **[2 points]** You train two variational autoencoders on the same dataset. The first model achieves a better (higher) ELBO on the training set. This implies the first model also achieves higher log-likelihood on the training set.

   (e) **[2 points]** Flow models are constructed by composing multiple transformations $\boldsymbol{z} = f_k \circ f_{k-1} \circ \ldots f_2 \circ f_1(\boldsymbol{x})$, and each intermediate $f_i(\cdot)$ yields a lower-dimensional representation of the input $\boldsymbol{x}$.

   (f) **[2 points]** Unlike autoregressive models, a GAN style likelihood-free training objective can be used to optimize for the reverse KL-divergence between the data distribution and the model distribution. Assume that we have access to infinitely powerful discriminators but only finite capacity generators.

   (g) **[2 points]** Assume the discriminator of a GAN is performing binary classification and trained via the cross-entropy objective. In the training dataset for the discriminator, we further assume that real datapoints from the data distribution are assigned label $Y = 1$ and fake datapoints from the generator are assigned label $Y = 0$. Then for any choice of the generator, the optimal discriminator will assign probability 1 to the real datapoints and 0 to the fake datapoints.

2. **[6 points total] Autoregressive models**
Given some inputs $\{\boldsymbol{x}_i\}_{i=0}^3$, Figure 1 shows three neural networks used for language modeling. The values $\{\hat{\boldsymbol{x}}_i\}_{i=1}^4$ denote the neural networks' corresponding outputs.

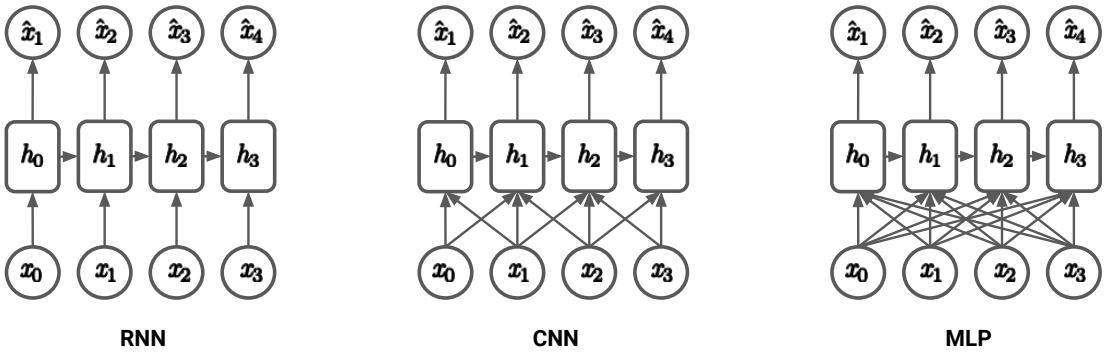

Figure 1

(a) **[2 points]** Which of these models satisfy the auto-regressive assumption?

(b) **[4 points]** For the models that don't satisfy the auto-regressive assumption, what are the minimum number of connections that we would mask to make these models auto-regressive? Please redraw each model with only the subset of connections such that they satisfy autoregressive property.

3. **[8 points total] MADE: Masked Autoencoders**

   In a Masked Autoencoder Distribution Estimator (MADE), each layer of model is masked such that the overall model preserves the autoregressive property. The typical MADE training procedure involves sampling a different set of masks for each mini-batch (while preserving the autoregressive property).

   (a) **[5 points]** In this question, we shall focus on the **input layer mask** (the mask applied to the input layer weight matrix in the MADE model). Consider a MADE model where the input dimensionality is n (i.e. $\boldsymbol{x} \in \mathbb{R}^n$) and the first hidden layer has h units. Recall the MADE mask construction procedure: for each unit in the hidden layer, pick a random integer $i \in \{1, \ldots, n-1\}$. That hidden unit is allowed to depend only on the first i input units (according to the chosen input ordering). Based on this sampling procedure, how many distinct input layer masks can be constructed? Assume a fixed ordering for the input units has already been chosen.

   (b) **[3 points]** Is training with all combinations of masks more likely to cause the models to **underfit** or **overfit** to the training data relative to using one mask? Why?

4. **[12 points total] Latent Variable Models**

   Let $\mathcal{D}_{train} = \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N\}$, $\mathcal{D}_{val} = \{\boldsymbol{x}_{N+1}, \cdots, \boldsymbol{x}_{N+K}\}$, $\mathcal{D}_{test} = \{\boldsymbol{x}_{N+K+1}, \cdots, \boldsymbol{x}_{N+K+M}\}$ be the training, validation, and test splits of the MNIST dataset, which are all disjoint. Consider the following generative model:

$$p_\sigma(\boldsymbol{x}) = \frac{1}{N}\sum_{i=1}^{N} \mathcal{N}(\boldsymbol{x}; \boldsymbol{x}_i, \sigma^2 I) \tag{1}$$

   where $\sigma \in \mathbb{R}$ and $\mathcal{N}(\boldsymbol{x}; \mu, \Sigma)$ denotes a Gaussian density with mean μ and covariance Σ evaluated at $\boldsymbol{x}$.

   (a) **[2 points]** Provide an efficient algorithm for evaluating $p_\sigma(\boldsymbol{x})$.

   (b) **[2 points]** Provide an efficient algorithm for sampling from $p_\sigma(\boldsymbol{x})$.

   (c) **[2 points]** Let σ^* be the maximum likelihood estimate of σ on the training set. Suppose that we have our train loss ($-\frac{1}{N}\sum_{i=1}^{N} \log p_{\sigma^*}(\boldsymbol{x}_i)$) and test loss ($-\frac{1}{M}\sum_{i=N+1+K}^{N+M+K} \log p_{\sigma^*}(\boldsymbol{x}_i)$). In the space below, rank the {train, test} loss terms in ascending order.

   (d) **[2 points]** You now train $p_\sigma(\boldsymbol{x})$ via maximum likelihood on $\mathcal{D}_{val}$, i.e., you maximize

$$\sigma' = \arg\max_\sigma \frac{1}{K}\sum_{i=N+1}^{N+K} \log p_\sigma(\boldsymbol{x}_i)$$

   Will $(\sigma')^2$ be smaller or larger than $(\sigma^*)^2$? Note that $p_\sigma(\boldsymbol{x})$ is still defined as in Eq. (1).

(e) **[2 points]** Will $p_{\sigma'}(\boldsymbol{x})$ or $p_{\sigma*}(\boldsymbol{x})$ produce samples more similar to handwritten digits? Why?

(f) **[2 points]** Will $p_{\sigma'}(\boldsymbol{x})$ or $p_{\sigma*}(\boldsymbol{x})$ work better for anomaly detection (like in Homework 1)?

5. **[22 points total] Adaptive Importance Weighted Autoencoder**

In this question, we consider a joint distribution $p(\boldsymbol{x}, \boldsymbol{z})$ and are interested in evaluating $\log p(\boldsymbol{x})$ at a fixed choice of $\boldsymbol{x}$. In class, we have shown how a variational distribution $q(\boldsymbol{z})$ can be used a construct a lower bound to the log-likelihood, $\log \int p(\boldsymbol{x}, \boldsymbol{z}) \mathrm{d}\boldsymbol{z} = \log p(\boldsymbol{x})$, via

$$\log p(\boldsymbol{x}) \geq \mathbb{E}_{q(\boldsymbol{z})} \left[\log \frac{p(\boldsymbol{x}, \boldsymbol{z})}{q(\boldsymbol{z})} \right]. \tag{2}$$

Given an estimate of the lower bound, one can try to adaptively improve $q(\boldsymbol{z})$ to achieve a tighter lower bound. In this question, we start with an initial variational distribution $q_1(\boldsymbol{z})$ and any function f that takes the current distribution q_i and sample $\boldsymbol{z}^{(i)}$ as input and returns a new distribution q_{i+1}. We now consider the following stochastic procedure:

$$\textbf{For } i \in \{1, 2, \ldots, T\}\textbf{:} \tag{3}$$

$$\boldsymbol{z}^{(i)} \sim q_i(\boldsymbol{z}) \tag{4}$$

$$w^{(i)} = \frac{p(\boldsymbol{x}, \boldsymbol{z}^{(i)})}{q_i(\boldsymbol{z}^{(i)})} \tag{5}$$

$$q_{i+1} = f(q_i, \boldsymbol{z}^{(i)}). \tag{6}$$

Along the way, we also collect $\{w^{(1)}, \ldots, w^{(T)}\}$ and design the estimator

$$\log \frac{1}{T} \sum_{i=1}^{T} w^{(i)}. \tag{7}$$

This estimator is a random variable since it depends on the sampling of $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(T)}\}$.

(a) **[10 points]** Prove that, for any choice of f and initial distribution q_1, the expected value of the estimator in Eq. 7 is always a lower bound for $\log p(\boldsymbol{x})$. (Remember that the expectation is with respect to the sampling of $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(T)}\}$ according to the original stochastic procedure.)

(b) **[12 points]** Suppose we compare the following two estimators,

$$A = \frac{1}{T} \sum_{i=1}^{T} w^{(i)} \tag{8}$$

$$B = w^{(1)} \tag{9}$$

These estimators are random variables since they depend on the sampling of $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(T)}\}$ according to the original stochastic procedure. These two estimators also depend on the choice of (q_1, f). Furthermore, note that both A and B are always unbiased estimators of $p(\boldsymbol{x})$. Propose a choice of (q_1, f) such that

$$\mathbb{E}[\log A] < \mathbb{E}[\log B], \tag{10}$$

where the expectation is with respect to the sampling of $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(T)}\}$. To get full credit, you have to use your choice of (q_1, f) and prove that $\mathbb{E}[\log A] < \mathbb{E}[\log B]$.

6. **[10 points total] Non-Saturating Generative Adversarial Networks**

   Recall that the original generative adversarial network training objective is

$$\min_{\theta} \max_{D} \mathbb{E}_{p_{\text{data}}(\boldsymbol{x})} \left[\log D(\boldsymbol{x})\right] + \mathbb{E}_{p_{\theta}(\boldsymbol{x})} \left[\log(1 - D(\boldsymbol{x}))\right]. \tag{11}$$

   Let $D^*_{\theta}(\boldsymbol{x})$ denote the optimal discriminator to the inner objective for any fixed choice of θ,

$$\max_{D} \mathbb{E}_{p_{\text{data}}(\boldsymbol{x})} \left[\log D(\boldsymbol{x})\right] + \mathbb{E}_{p_{\theta}(\boldsymbol{x})} \left[\log(1 - D(\boldsymbol{x}))\right]. \tag{12}$$

   In this problem, we shall consider the following "non-saturating" objective function

$$\min_{\theta} -\mathbb{E}_{p_{\theta}(\boldsymbol{x})} \left[\log D^*_{\theta}(\boldsymbol{x})\right]. \tag{13}$$

   Our goal is to characterize this objective function. In particular, prove that

$$\arg\min_{\theta} -\mathbb{E}_{p_{\theta}(\boldsymbol{x})} \left[\log D^*_{\theta}(\boldsymbol{x})\right] = \arg\min_{\theta} D_{\text{KL}} \left(p_{\theta}(\boldsymbol{x}) \| p_{\text{data}}(\boldsymbol{x})\right) - D_{\text{KL}} \left(p_{\theta}(\boldsymbol{x}) \| m_{\theta}(\boldsymbol{x})\right), \tag{14}$$

   where $m_{\theta}(\boldsymbol{x}) = \dfrac{p_{\text{data}}(\boldsymbol{x}) + p_{\theta}(\boldsymbol{x})}{2}$, and D_{KL} denotes the Kullback-Leibler Divergence.

7. **[20 points total] Noise Contrastive Estimation (NCE)**

   Let $p_{\text{data}}(\boldsymbol{x})$ be the data distribution, and $p_{\text{noise}}(\boldsymbol{x})$ be a tractable noise distribution, e.g., multivariate Gaussian. Our goal is to learn a normalized probabilistic model $p_\theta(\boldsymbol{x})$ where θ denotes all trainable parameters. For example, $p_\theta(\boldsymbol{x})$ can be an autoregressive model or flow model. Assume that $\forall \boldsymbol{x}$: $p_{\text{data}}(\boldsymbol{x}) > 0, p_{\text{noise}}(\boldsymbol{x}) > 0$, and $p_\theta(\boldsymbol{x}) > 0$. Assume that the model family $p_\theta(\boldsymbol{x})$ is expressive enough to capture $p_{\text{data}}(\boldsymbol{x})$. The method of Noise Contrastive Estimation (NCE) fits $p_\theta(\boldsymbol{x})$ to $p_{\text{data}}(\boldsymbol{x})$ by solving the following optimization problem:

$$\max_\theta \quad \mathbb{E}_{p_{\text{data}}(\boldsymbol{x})}[\log \sigma(\log p_\theta(\boldsymbol{x}) - \log p_{\text{noise}}(\boldsymbol{x}))] + \mathbb{E}_{p_{\text{noise}}(\boldsymbol{x})}[\log(1 - \sigma(\log p_\theta(\boldsymbol{x}) - \log p_{\text{noise}}(\boldsymbol{x})))], \quad (15)$$

   where $\sigma(a) = 1/(1+e^{-a})$ is the sigmoid non-linearity. Let θ^* be the optimal parameters that maximize the NCE objective. Show that:

$$p_{\theta^*}(\boldsymbol{x}) = p_{\text{data}}(\boldsymbol{x}). \quad (16)$$

8. **[18 points total] Normalizing Flow Models**

   Consider two datasets $\mathcal{D}_A$ and $\mathcal{D}_B$ where the datapoints are sampled i.i.d. from two different data distributions p_{A} and p_{B}. The data distributions are defined over the same space, i.e., the random variables A and B have the same domain.

   We consider two independent flow models for learning generative models over A and B, with latent variables denoted as U and V respectively. Let G_A and G_B denote the parametric mappings from $U \to A$ and $V \to B$ respectively. For both models, we assume the prior over the latent variables to be standard Gaussian. We specify G_A to be an inverse autoregressive flow (IAF) and G_B to be a masked autoregressive flow (MAF). Finally, let q_A and q_B denote the corresponding model densities.

   (a) **[5 points]** Assume G_A and G_B have same number of layers and neural network architectures in every layer. Given a datapoint $\boldsymbol{a} \sim q_A$, let t_1 denote the minimum number of arithmetic operations required for sampling $\boldsymbol{a}$ and evaluating $q_A(\boldsymbol{a})$. Let t_2 denote the minimum number of arithmetic operations required for evaluating $q_B(\boldsymbol{b})$ for some $\boldsymbol{b} \in \mathcal{D}_B$. Is $t_1 > t_2$? Why or why not?

   (b) **[5 points]** Given a point $\boldsymbol{a} \sim q_A$, derive a deterministic transformation of $\boldsymbol{a}$ to generate a sample $\boldsymbol{b} \sim q_B$.

   (c) **[8 points]** Derive an expression for q_B that only involves G_B, G_A, and q_A.