CS 598-DGM: Spring'25 Deep Generative Models Homework 2

(Due Monday, March 31, 11:59 pm)

(Due Monday, April 21, 11:59 pm)

- The homework is due at 11 : 59pm on the due date. We will be using Gradescope for the homework assignments. Please do NOT email a copy of your solution. Contact the TAs (Zhijie, Rohan) if you are having technical difficulties in submitting the assignment. We will NOT accept late submissions.
- Please make sure that each question is clearly marked. You may use as many pages as needed but do not change the order of the questions and answers.
- You are expected to typeset the solutions, i.e., handwritten solutions will not be graded. We encourage you to use $\LaTeX$. When submitting on Gradescope, you are required to assign the correct pages for each sub-problem to the provided outline. If pages are incorrectly assigned or left unassigned on Gradescope, it will result in no credit, and regrade requests regarding this will be declined. Double-check your submission to ensure accuracy.
- Please use Slack first if you have questions about the homework. You can also come to our (zoom) office hours and/or send us e-mails. If you are sending us emails with questions on the homework, please start subject with "CS 598-DGM: " and send the email to all course staff: Arindam, Zhijie, and Rohan.
- The homework consists of written assignments. Please submit your report as a PDF file.

1. (30 points) In the context of Denoising AutoEncoders (DAEs), one uses the smoothed (or noisy) distribution $q_\sigma(\tilde{\mathbf{x}}) = \int_x q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x}) q_0(\mathbf{x}) d\mathbf{x}$ for the modeling for a suitable choice of $q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x})$. Let $s_\theta(\tilde{\mathbf{x}})$ denote the score function to be estimated.

   (a) (10 points) Let

$$J_1(\theta) = \mathbb{E}_{q_\sigma(\tilde{\mathbf{x}})} \left[\frac{1}{2} \left\| s_\theta(\tilde{\mathbf{x}}) - \nabla_{\tilde{\mathbf{x}}} \log q_\sigma(\tilde{\mathbf{x}}) \right\|^2 \right] \tag{1}$$

$$J_2(\theta) = \mathbb{E}_{q_\sigma(\mathbf{x}, \tilde{\mathbf{x}})} \left[\frac{1}{2} \left\| s_\theta(\tilde{\mathbf{x}}) - \nabla_{\tilde{\mathbf{x}}} \log q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x}) \right\|^2 \right] \tag{2}$$

   Show that $J_1(\theta) = J_2(\theta) + c$ under suitable regularity conditions where c is a constant independent of θ. Please specify the conditions you have used to establish the identity.

   Solution. We aim to show that $J_1(\theta) = J_2(\theta) + c$, where c is a constant independent of θ.
   We use the fact that the marginal score can be written as an expectation over the conditional score:

$$\nabla_{\tilde{\mathbf{x}}} \log q_\sigma(\tilde{\mathbf{x}}) = \mathbb{E}_{\mathbf{x}|\tilde{\mathbf{x}}} \left[\nabla_{\tilde{\mathbf{x}}} \log q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x}) \right].$$

   This holds under standard regularity conditions. Now expand $J_1(\theta)$ using this identity:

$$\begin{aligned} J_1(\theta) &= \frac{1}{2} \mathbb{E}_{\tilde{\mathbf{x}}} \left\| s_\theta(\tilde{\mathbf{x}}) - \mathbb{E}_{\mathbf{x}|\tilde{\mathbf{x}}} \left[\nabla_{\tilde{\mathbf{x}}} \log q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x}) \right] \right\|^2 \\ &= \frac{1}{2} \mathbb{E}_{\tilde{\mathbf{x}}} \left[\mathbb{E}_{\mathbf{x}|\tilde{\mathbf{x}}} \left\| s_\theta(\tilde{\mathbf{x}}) - \nabla_{\tilde{\mathbf{x}}} \log q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x}) \right\|^2 \right] \\ &\quad - \frac{1}{2} \mathbb{E}_{\tilde{\mathbf{x}}} \left[\mathrm{Var}_{\mathbf{x}|\tilde{\mathbf{x}}} \left(\nabla_{\tilde{\mathbf{x}}} \log q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x}) \right) \right] \\ &= J_2(\theta) + c, \end{aligned}$$

   where

$$c = -\frac{1}{2} \mathbb{E}_{\tilde{\mathbf{x}}} \left[\mathrm{Var}_{\mathbf{x}|\tilde{\mathbf{x}}} \left(\nabla_{\tilde{\mathbf{x}}} \log q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x}) \right) \right].$$

   Since c is independent of the learnable parameter θ, minimizing $J_1(\theta)$ is equivalent to minimizing $J_2(\theta)$. □

(b) (20 points) For this problem, we assume that the conditional distribution $q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x}) = \mathcal{N}\left(\mathbf{x}, \sigma^2 \mathbb{I}\right)$. Consider three univariate smoothed distributions $q_{\sigma_0}(\tilde{\mathbf{x}}, \mathbf{x}), q_{\sigma_1}(\tilde{\mathbf{x}}, \mathbf{x})$, and $q_{\sigma_2}(\tilde{\mathbf{x}}, \mathbf{x})$ with $0 < \sigma_0 < \sigma_1 < \sigma_2$. Note that the $\sigma_i, i = 0, 1, 2$ correspond to the standard deviations of the corresponding Gaussian conditional distributions $q_{\sigma_i}(\tilde{\mathbf{x}} \mid \mathbf{x})$ and $q_{\sigma_i}(\tilde{\mathbf{x}}, \mathbf{x}) = q_{\sigma_i}(\tilde{\mathbf{x}} \mid \mathbf{x}) q_0(\mathbf{x})$. Further, we assume $\sigma_0^2 \leq \frac{\sigma_1^2}{d}$ where d is the dimensionality, i.e., $\tilde{\mathbf{x}} \in \mathbb{R}^{d}$[1]

In this setting, with $q_{\sigma_i}, i = 0, 1, 2$, denoting joint distributions, Professor Super Smooth claims that we always have

$$KL\left(q_{\sigma_1} \| q_{\sigma_0}\right) \leq KL\left(q_{\sigma_2} \| q_{\sigma_0}\right) \tag{3}$$

i.e., smoothing with a higher variance ($\sigma_2 > \sigma_1$) Gaussian moves the smoothed joint distribution further away from the joint distribution q_{σ_0}.

Do you agree/disagree with the Professor? If you agree, you have to prove the claim. If you disagree, you will have to give a counterexample to the claim.

Solution. We agree with Professor Super Smooth's claim. To prove it, we analyze the structure of the joint distributions.

Each joint distribution has the form:

$$q_{\sigma_i}(\tilde{\mathbf{x}}, \mathbf{x}) = q_0(\mathbf{x}) \cdot \mathcal{N}(\tilde{\mathbf{x}} \mid \mathbf{x}, \sigma_i^2 \mathbb{I}).$$

Hence, the KL divergence between q_{σ_j} and q_{σ_0} becomes:

$$\mathrm{KL}(q_{\sigma_j} \| q_{\sigma_0}) = \mathbb{E}_{\mathbf{x} \sim q_0}\left[\mathrm{KL}\left(\mathcal{N}(\mathbf{x}, \sigma_j^2 \mathbb{I}) \parallel \mathcal{N}(\mathbf{x}, \sigma_0^2 \mathbb{I})\right)\right].$$

For two multivariate Gaussians with the same mean and isotropic covariances, the KL divergence is:

$$\mathrm{KL}\left(\mathcal{N}(\mu, \sigma_j^2 \mathbb{I}) \| \mathcal{N}(\mu, \sigma_0^2 \mathbb{I})\right) = \frac{d}{2}\left(\frac{\sigma_j^2}{\sigma_0^2} - 1 - \log \frac{\sigma_j^2}{\sigma_0^2}\right).$$

Define:

$$f(\alpha) := \frac{\alpha}{\sigma_0^2} - 1 - \log\left(\frac{\alpha}{\sigma_0^2}\right), \quad \text{for } \alpha > 0.$$

The function $f(\alpha)$ is monotonically increasing for $\alpha > \sigma_0^2$. Since $\sigma_2 > \sigma_1 > \sigma_0$, we have:

$$f(\sigma_2^2) > f(\sigma_1^2), \quad \Rightarrow \quad \mathrm{KL}(q_{\sigma_2} \| q_{\sigma_0}) > \mathrm{KL}(q_{\sigma_1} \| q_{\sigma_0}).$$

Therefore, we conclude that increasing the smoothing variance moves the smoothed joint distribution further away from the reference distribution in the KL sense.

Professor Super Smooth's claim is correct. □

[1] Intuitively, with $\sigma_0 \approx 0$, we have $q_{\sigma_0}(\tilde{\mathbf{x}}) \approx q_0(\mathbf{x})$, so q_{σ_0} can be viewed as an accurate approximation of the true distribution q_0.

2. (18 points) Assume you have trained a generative $p_{\hat{\theta}}(x)$ which accurately models some target distribution $p_*(x)$, i.e., $p_{\hat{\theta}}(x) \approx p_*(x)$. We consider the problem of likelihood computation:
(Likelihood Computation) Given x_{test} , what is the value of $p_{\hat{\theta}}(x_{\text{test}})$?
Consider suitable versions of the following four family of models for $p_{\hat{\theta}}(x)$

   (a) Variational Auto-Encoders (VAEs),

   (b) Generative Adversarial Networks (GANs),

   (c) Diffusion model, specifically Score-SDE [1], which uses isotropic Gaussian as the source distribution,

   (d) Flow matching model, specifically Rectified Flow [2], which uses isotropic Gaussian as the source distribution,

   (e) Flow matching model, specifically Rectified Flow [2], which uses non-Gaussian $p_0(x)$ as the source distribution, where we cannot compute $p_0(x)$ given any x, but have n samples $\left\{x_i^{(0)}, i \in [n]\right\}$ from $p_0(x)$, i.e., $x_i^{(0)} \sim p_0(x), i \in [n]$.

[1] (Score-SDE) Y. Song, J. Sohl-Dickstein, D. Kingma, A. Kumar, S. Ermon, B. Poole. Score-based generative modeling through stochastic differential equations. ICLR, 2021.
[2] (Rectified Flow) X. Liu, C. Gong, and Q. Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. ICLR, 2023.

Please answer the following questions:

   (a) ($3 \times 5 = 15$ points) For the each of the above five models above, can the model compute $p_{\hat{\theta}}(x_{\text{test}})$ for any given x_{test} exactly?[1] Briefly justify each answer. You can assume ability for 'simulation', e.g., solving ODE/SDE precisely.

Solution. i. VAE: No. VAE has decoder that learns the conditional distribution $p(x \mid z)$. As the latent variable is often in high dimensional vector space, VAE cannot compute $p(x)$ exactly.

ii. GAN: No. GAN learns to generate pictures directly in an implicit way, and thus has no modeling of the density function.

iii. Score-SDE: Yes. As we can use the learned score function $s_{\hat{\theta}}(x,t)$ that models $\nabla \log p_t(x)$. Consider the integral

$$\log p_{\hat{\theta}}(x) = \log p_T(x_T) - \int_0^T \frac{1}{2}\beta(t)\nabla_x \cdot (s_{\hat{\theta}}(x,t))dt,$$

where p_T is gaussian, and x_T is the noised x.

iv. Rectified Flow with isotropic Gaussian: Yes. As we can use the learned vector field $v_{\hat{\theta}}(x,t)$ to model $\nabla \log p_t(x)$. Thus

$$\log p_{\hat{\theta}}(x) = \log p_0(x_0) + \int_0^1 \nabla_x \cdot v_{\hat{\theta}}(x,t)dt,$$

where p_0 is the isotropic Gaussian.

v. Rectified Flow with non-analyitic p_0: No. As the last case indicates, we need the expression of p_0, however, when only given samples, p_0 is not tractable.

□

[1] If it is an approximation, upper bound, or lower bound, then that is not acceptable in the context of this question.

(b) (3 points)[2] For the models which can compute $p_{\hat{\theta}}(x_{\text{test}})$, briefly outline the computation needed, starting from the model for $p_{\hat{\theta}}(x)$ and x_{test}.

Solution. The computation needed will be:

For diffusion model,

(a) integrate the integrand $\frac{1}{2}\beta(t)\nabla_x \cdot (s_{\hat{\theta}}(x,t))$ from 0 to T

(b) compute the other endpoint x_T of the trajectory of x via the diffusion, and plug in the target pdf p_T

(c) compute via the formula in last subquestion.

For rectified flow,

(a) integrate the integrand $\nabla_x \cdot v_{\hat{\theta}}(x,t)$ from 0 to 1

(b) compute the other endpoint x_0 of the trajectory of x via the vector field

(c) compute via the formula in last subquestion.

□

[2]We are not giving the breakdown over the five models as that will reveal which ones can actually do the computation correctly.

3. (10 points) The question considers the generative model presented in this paper-we will refer to the model as Score-SDE:
Y. Song, J. Sohl-Dickstein, D. Kingma, A. Kumar, S. Ermon, B. Poole. Score-based generative modeling through stochastic differential equations ICLR, 2021.

You have been given three things

(a) A trained Score-SDE model $p_{\theta_1}(x)$ over Dogs, where the model uses isotropic Gaussians as the source distribution $p_0(x)$,

(b) A trained Score-SDE model $p_{\theta_2}(x)$ over Huskies, where the model uses isotropic Gaussians as the source distribution $p_0(x)$,

(c) A set of n samples $\left\{x_i^{(1)}, i \in [n]\right\}$ of Dogs,

where θ_1, θ_2 are respectively the Score-SDE model parameters for Dogs and Huskies. Please see the Score-SDE model for details on what these parameters are.

(a) (5 points) Present an inference algorithm which samples a Dog $x_i^{(1)}$ uniformly and then uses suitable reverse SDE to generate a Husky sample.

Solution. Consider the following steps:

i. uniformly sample from the dog pictures $x^(1)_i$'s.
ii. solve the forward SDE with a drift term, setting the drift term $f(x,t) = p_{\theta_1}(x)$
iii. solve the reverse SDE, use the same drift term; using the score function multiplied with $g^2(t)$.

□

(b) (5 points) Argue why such inference may be faster than starting from an isotropic Gaussian sample.

Solution. The distribution of Dogs, in some metric, should be closer to that of Huskies. Thus, if the original SDE takes T steps going from Huskies to final distirbution, it would mean fewer SDE solves need to be doen if one could start from the distributions of Dogs. □

4. . (42 points) Let $X_0 \sim p_0(x)$ (not necessarily Gaussian) be the source distribution and $X_1 \sim p_1(x)$ be the target distribution for generative modeling. This question considers conditional flow matching (CFM) and Schrödinger bridge (SB) for generative modeling.

(a) (5 points) What is main difference between training flow models based on maximum likelihood and conditional flow matching? Please explain the difference using suitable mathematical notation.

Solution. For likelyhood, we can compute two loss functions, the first one base log-likely hood

$$L(\theta) = \mathbb{E}_{x\sim q_1}[\log p_1(x)]$$

as it is the negative MLE. For CFM,

$$L_{CFM}(\theta) = \mathbb{E}_{t\sim U[0,1], x_1\sim q, x_t\sim p_t(x|x_1)}[\|u_\theta(t,x) - u_t(x \mid x_1)\|^2].$$

□

(b) (5 points) For CFM, given any arbitrary choice of conditional probability path $p_t(x \mid z)$, can the conditional velocity field (VF) $u_t(x \mid z)$ be obtained in closed form? Clearly justify your answer.

Solution. It is generally not possible to obtain $u_t(x \mid z)$ in closed form. As this is normally done by doing:

$$\frac{\partial p_t(x \mid z)}{\partial t} = -\nabla \cdot (u_t(x \mid z) p_t(x \mid z)).$$

If p_t is a general function, there is no closed form. We could solve it before as it is linear or Gaussian. □

(c) (12 points) We consider affine conditional flows of the following form for flow matching:

$$\psi_t(x \mid x_1) = \alpha_t x_1 + \sigma_t x, \quad \alpha_0 = 0 = \sigma_1, \alpha_1 = 1, \text{ and } \dot{\alpha}_t, -\dot{\sigma}_t > 0, t \in (0,1) \tag{4}$$

i. (4 points) Show that $\alpha_t = t, \sigma_t = (1-t)$ is a valid choice for the parameters.

Solution. It is straightforwad to verify that $\alpha_0 = 0, \alpha_1 = 1, \sigma_1 = 0$ and derivative of α_t and $-\sigma_t$ are both positive, as they are both 1. □

ii. (4 points) What is the marginal VF for the above choices of the parameters?

Solution. For the conditional VF:

$$u_t(x \mid x_1) = \frac{d}{dt}(\alpha_t x_1 + \sigma_t x) = x_1 - x$$

For the marginal VF:

$$u_t(x) = \mathbb{E}_{x_1}[u_t(x \mid x_1)] = \mathbb{E}_{x_1}[x_1] - x$$

□

iii. (4 points) Will the corresponding transport path be the same as the one obtained by running optimal transport[3] between p_0 and p_1 ? Clearly justify your answer.

Solution. No. The linear interpolation fund the shortest path for a particular pair, but globally it restrict the path to be straight lines. However, OT found the shortest path considering the total distance globally. □

(d) (10 points) Consider a series $\beta(s) > 0, s \in [0,1]$, and construct the parameters

$$\alpha_t = e^{-\frac{1}{2}T(1-t)}, \quad \sigma_t = \left(1 - \alpha_t^2\right), \quad T(t) = \int_0^t \beta(s)ds \tag{5}$$

i. (6 points) Draw plots of α_t, σ_t separately over $t \in [0,1]$ for $\beta(s) = 598, s \in [0,1]$.
Solution. See figure 1 □

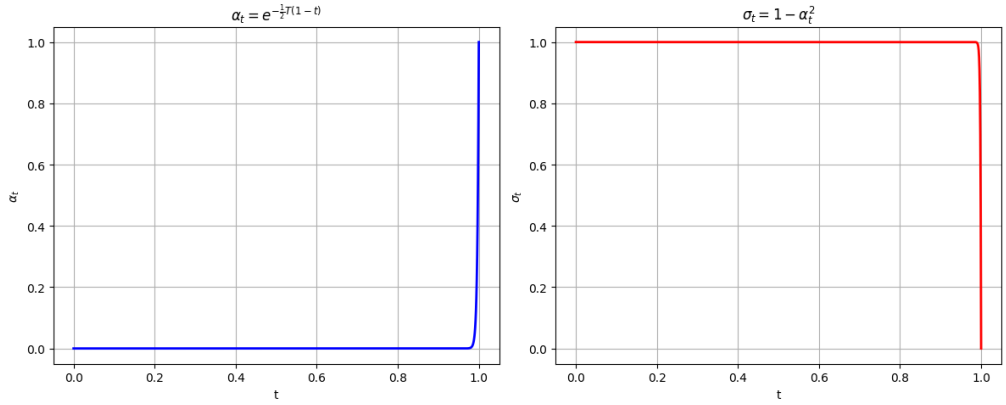

Figure 1: Plots of α_t and σ_t

ii. (4 points) Does the resulting α_t, σ_t satisfy the conditions in (4)? Clearly justify your answer.
Solution. No. It doesn't follow $\alpha_0 = 0$. □

(e) (10 points) This question considers Brownian bridge and Schrödinger bridge in the context of generative modeling.

i. (5 points) What is the main difference between a Brownian bridge and a Schrödinger bridge in the context of generative modeling? Please explain the difference using suitable mathematical notation.

Solution. In generative modeling, a Brownian bridge interpolates between a source distribution p_0 and a target distribution p_1 using a *prior stochastic process*, typically standard Brownian motion W_t, conditioned on endpoints:

$$X_t = (1-t)X_0 + tX_1 + \sqrt{2\varepsilon}\, B_t^{\text{bridge}}, \quad X_0 \sim p_0,\ X_1 \sim p_1,$$

where B_t^{bridge} is a Brownian bridge and ε is the diffusion scale.
In contrast, a Schrödinger bridge (SB) constructs an entropic interpolation between p_0 and p_1 by finding the most likely path under a reference stochastic process $\mathbb{Q}$ (e.g., Brownian motion), minimizing the Kullback-Leibler divergence:

$$\text{SB:} \quad \min_{\mathbb{P}} \mathrm{KL}(\mathbb{P} \,\|\, \mathbb{Q}) \quad \text{subject to } \mathbb{P}_{t=0} = p_0,\ \mathbb{P}_{t=1} = p_1,$$

where $\mathbb{P}$ denotes the law of the learned stochastic process.
Thus, the Brownian bridge uses a fixed, conditioned prior path, while the Schrödinger bridge learns a control that optimally adjusts the prior to match the marginals. □

ii. (5 points) Which model would you choose for faster inference time? Clearly justify your answer.
Solution. The Brownian bridge allows *faster inference*, since it involves sampling from a known conditioned process with analytical or direct simulation methods. No learned dynamics or iterative computation are needed at test time.
In contrast, Schrödinger bridge models typically require solving a stochastic control problem or sampling from time-dependent drift fields, often learned via score matching or iterative Sinkhorn updates, which are computationally expensive at inference.
Hence, the Brownian bridge is preferred for faster inference. □

[3] We are considering quadratic cost optimal transport, as discussed in class.