Cornell University
CS 5788: Introduction to Generative Models
Spring 2026 Instructor: Andrew Owens

Problem Set 3: Diffusion models

Posted: Tuesday, March 10, 2026 **Due:** Tuesday, April 7, 2026

Please submit your written solution to Gradescope as a `.pdf` file. Please convert your Colab notebooks to PDF. For your convenience, we have included the PDF conversion script at the end of the notebook.

We recommend editing and running your code in Google Colab, although you are welcome to use your local machine instead. Please note that problems marked optional will not be graded.

Problem 3.1 *Normalizing Flows*

We will implement the *Real NVP* normalizing flow model (see **Density Estimation Using Real NVP** by Dinh et al.) to generate face images. The notebook `NormalizingFlow.ipynb` will walk you through the implementation.

Generation examples:

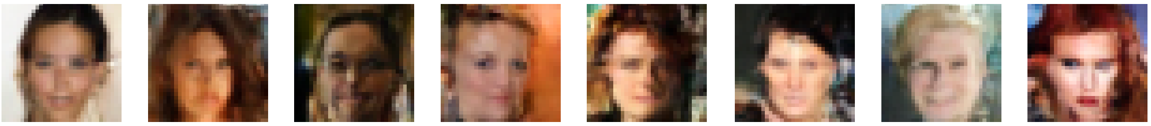

Figure 1: Face images sampled from the normalizing flow model.

`Warning`:

- This takes a long time to train! In our implementation, it took 2-3 hours on a T4 GPU.
- Since we changed the model to be smaller to speed up training (versus the original paper), the results won't be perfect. We show some examples in Fig. 1 for reference.

Problem 3.2 *Diffusion Models*

In the notebook DiffusionTraining.ipynb, you will train your own diffusion model to generate images of handwritten digits! You will implement the architecture

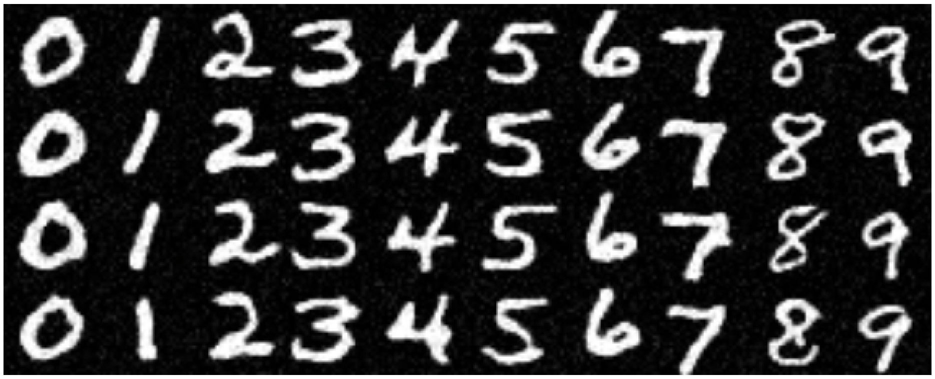

Figure 2: Generated samples of handwritten digits.

Problem 3.3 *Written problems*

Please turn in your answers (either written or typeset) as a separate PDF.

(a) In this problem, we'll explore basic properties of the Gaussian, and use them to help understand the forward diffusion process, which adds noise to the data.

(i) The forward process involves adding noise to examples that are already noisy. Here, we will analyze how the distribution of the data changes when this happens. Recall that the probability density function (PDF) of the sum of two independent random variables $Z = X + Y$ is given by the convolution of their individual PDFs:

$$f_Z(z) = (f_X * f_Y)(z) = \int_{-\infty}^{\infty} f_X(x) f_Y(z - x) dx \tag{1}$$

Suppose $X \sim \mathcal{N}(0, \sigma_1^2)$ and $Y \sim \mathcal{N}(0, \sigma_2^2)$ are independent Gaussian random variables. Using the convolution formula, prove that $Z = X + Y$ is also a Gaussian random variable with variance $\sigma_z^2 = \sigma_1^2 + \sigma_2^2$.

Hint: You will need to complete the square for x inside the exponential of the integral. Recall that $\int_{-\infty}^{\infty} e^{-ax^2} dx = \sqrt{\pi/a}$.

(ii) In the standard diffusion model formulation, we do not only add noise; we also scale the data down slightly before adding noise to keep the total variance constant. Let $X \sim \mathcal{N}(0, \sigma^2)$ and let a be a constant scalar. What is the distribution of the scaled random variable aX?

(iii) In the DDPM forward process, a single step is defined as $x_t = \sqrt{1 - \beta_t} x_{t-1} + \sqrt{\beta_t}\epsilon$, where $\epsilon \sim \mathcal{N}(0, I)$. Show that if x_{t-1} has unit variance ($\sigma^2 = 1$), then x_t also has unit variance.

(iv) Let's look at two consecutive steps in the diffusion process. Let $\alpha_t = 1 - \beta_t$. We have:

- $x_1 = \sqrt{\alpha_1} x_0 + \sqrt{1 - \alpha_1}\epsilon_0$
- $x_2 = \sqrt{\alpha_2} x_1 + \sqrt{1 - \alpha_2}\epsilon_1$

where $\epsilon_0, \epsilon_1 \sim \mathcal{N}(0, \mathbf{I})$ and are independent. Substitute the equation for x_1 into the equation for x_2 to express x_2 in terms of x_0, ϵ_0, and ϵ_1.

(v) Show that x_2 can be written as a single Gaussian transition: $x_2 = \sqrt{\alpha_1 \alpha_2} x_0 + \sqrt{1 - \alpha_1 \alpha_2}\epsilon_{\text{combined}}$. What is the distribution of $\epsilon_{\text{combined}}$?

(vi) In a standard Markov Chain, to calculate the state at x_{500}, you would normally have to compute all 499 previous states. However, in Diffusion Models, we define $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$, which allows us to sample x_t directly:

$$q(x_t | x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t} x_0, (1 - \bar{\alpha}_t)\mathbf{I})$$

During training, we optimize the model by picking a random image x_0 and a random timestep $t \in \{1, \ldots, T\}$ from a batch. Explain why the ability to "jump" to noise level t in $O(1)$ time is important for efficiently training a diffusion model using stochastic gradient descent. What would be the computational cost of training if we had to iterate through $t - 1$ steps for every training sample?

(b) In a Denoising Diffusion Probabilistic Model (DDPM), we define the forward process $q(\mathbf{x}_{1:T}|\mathbf{x}_0)$ and the reverse process $p_\theta(\mathbf{x}_{0:T})$. We have already used Jensen's Inequality to define the Variational Lower Bound (VLB) as:

$$\mathcal{L}_{VLB} = \mathbb{E}_q \left[\log \frac{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}{p_\theta(\mathbf{x}_{0:T})} \right] \tag{2}$$

In the following questions, we are going to show that this loss can be decomposed into a sum of KL divergences, making it possible to train the model step-by-step.

(i) Using the Markov property, write out the full product form for both the forward process $q(\mathbf{x}_{1:T}|\mathbf{x}_0)$ and the reverse process $p_\theta(\mathbf{x}_{0:T})$.
Hint: Remember that $p(\mathbf{x}_T)$ is a standard Gaussian and has no θ parameters.

(ii) In the forward process of a diffusion model, we gradually add noise to an image. For this, the transition $q(\mathbf{x}_t|\mathbf{x}_{t-1})$ is easy to define and sample from. However, our goal is to *reverse* the process and to learn to denoise $\mathbf{x}_t$ back into $\mathbf{x}_{t-1}$. The true reverse transition $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ is intractable because it depends on the entire data distribution. But during training, we have access to the original, clean example $\mathbf{x}_0$. By conditioning on $\mathbf{x}_0$, the reverse step $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$ becomes a simple Gaussian distribution that we can calculate in closed form. This serves as our ground truth target. We train our neural network to approximate this $\mathbf{x}_0$-conditioned distribution.
Use the Markov property and Bayes' rule to show that:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \frac{q(\mathbf{x}_t|\mathbf{x}_0)}{q(\mathbf{x}_{t-1}|\mathbf{x}_0)} \tag{3}$$

(iii) Substitute the expression from (i) into the $\mathcal{L}_{VLB}$ formula. Then, substitute the identity from (ii) into the q terms.

(iv) Show that by rearranging terms, the VLB can be rewritten as:

$$\mathcal{L}_{VLB} = L_T + \sum_{t=2}^{T} L_{t-1} + L_0, \tag{4}$$

where:

$$\begin{aligned} L_T &= -\mathbb{E}_q \left[\log \frac{p(\mathbf{x}_T)}{q(\mathbf{x}_T|\mathbf{x}_0)} \right] = D_{KL}(q(\mathbf{x}_T|\mathbf{x}_0) \parallel p(\mathbf{x}_T)) \text{ and} && (5) \\ L_0 &= -\mathbb{E}_{q(\mathbf{x}_1|\mathbf{x}_0)} \left[\log p_\theta(\mathbf{x}_0|\mathbf{x}_1) \right] && (6) \end{aligned}$$

Please give a short description (one sentence is fine) of what each of these terms intuitively means, and how they affect the learning process.

(v) Show that

$$L_{t-1} = -\mathbb{E}_q \left[\log \frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)} \right] = D_{KL}(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \parallel p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)). \tag{7}$$

Also explain why, due to this term, $\mathcal{L}_{VLB}$ can naturally be interpreted as a *denoising* loss (one or two sentences is fine).

Credits. The diffusion problem was created by Daniel Geng, Ryan Tabrizi, Hang Gao, Liyue Shen, Andrew Owens, and Alexei Efros. The rest of the problem set was written by Yen-Yu Chang, and Andrew Owens.