

“Homework: Exploring the DDPM Forward Process and Noise Schedules”

Mengyan Jing

2026-02-18

Homework: Exploring the DDPM Forward Process and Noise Schedules

Course: STAT 9100

Topic: Denoising Diffusion Probabilistic Models

Total Points: 40

Instructions

- Complete all 4 questions below.
- Use the provided starter code file `hw_starter.py` as your starting point.
- For coding questions: fill in the marked `### TODO ###` sections, run the code, and save your plots.
- For written questions: write 2–4 sentences each. You can put written answers as comments in the code file, or in a separate document.
- Submit: your completed Python file and any generated plots.

Background

Recall from the lecture that the DDPM forward process adds noise to data using:

$$x_t = \sqrt{\bar{\alpha}_t} \cdot x_0 + \sqrt{1 - \bar{\alpha}_t} \cdot \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, I)$$

where $\bar{\alpha}_t = \prod_{s=1}^t (1 - \beta_s)$ is the cumulative product of the noise schedule.

The **Signal-to-Noise Ratio** at timestep t is:

$$\text{SNR}(t) = \frac{\bar{\alpha}_t}{1 - \bar{\alpha}_t}$$

Question 1: Implement a Custom Noise Schedule (10 points)

In the starter code, the `linear` and `cosine` schedules are already implemented. Your task is to implement a `sqrt` (square root) schedule where:

$$\bar{\alpha}_t = 1 - \sqrt{t/T}$$

(a) Complete the `sqrt_schedule()` function in the starter code. It should return an array of $\bar{\alpha}_t$ values for $t = 1, \dots, T$. (5 pts)

(b) Run the plotting code to generate an SNR comparison of all three schedules (linear, cosine, sqrt) on one figure. Include this plot in your submission. In 2–3 sentences: how does the sqrt schedule compare to linear and cosine? Does it drop too fast, too slow, or somewhere in between? (5 pts)

Question 2: Visualize the Forward Process with Different Schedules (10 points)

The starter code provides a function `q_sample(x0, t, alpha_bar)` that noises an image.

(a) Complete the code section that generates a **2-row figure** comparing the forward process using the `linear` schedule (top row) and your `sqrt` schedule (bottom row), at timesteps $t = 0, 200, 400, 600, 800, 999$. Use the same random seed for both rows so the noise realization is identical. Include this plot. (5 pts)

(b) At which timestep does each schedule make the image “unrecognizable” (in your judgment)? Which schedule wastes more timesteps on pure noise? (2–3 sentences) (5 pts)

Question 3: The Effect of T (Total Timesteps) (10 points)

In the original DDPM, $T = 1000$. But what if we used fewer total steps?

(a) Using the `cosine schedule`, compute and plot the SNR curves for $T = 50$, $T = 200$, and $T = 1000$ on the same figure. The x-axis should be the **fraction** t/T (from 0 to 1) so the curves are comparable. Include this plot. (5 pts)

(b) Based on your plot, answer: does the cosine schedule maintain a similar shape regardless of T ? Why might this property be important for sampling with fewer steps (as in the Improved DDPM paper)? (2–3 sentences) (5 pts)

Question 4: Connecting to the Reverse Process (10 points — Written Only)

No coding required for this question. Answer based on your understanding from the lecture.

(a) The DDPM training loss is:

$$\mathcal{L}_{\text{simple}} = \mathbb{E}_{t, x_0, \varepsilon} [\|\varepsilon - \varepsilon_\theta(x_t, t)\|^2]$$

Explain in your own words: why does the network predict the noise ε rather than directly predicting the clean image x_0 ? What advantage does this reparameterization give? (4 pts)

- (b) Consistency models can generate images in 1–2 steps instead of 1000. In 2–3 sentences, explain the key idea that makes this possible. What property must the consistency model satisfy? **(3 pts)**
- (c) Name one advantage and one disadvantage of consistency models compared to standard DDPMs. **(3 pts)**
-

Reminder: Start from `hw_starter.py`. The `### TODO ###` markers show exactly where you need to write code. Good luck!