Advanced Deep Learning for Physics (IN2298)
Exercise 9
Physics Guided Generative Modeling of Circular Data

In this exercise, we will explore how to generate a circle using diffusion models and improve performance by incorporating underlying physical laws. The following notations will be used for the diffusion process:

- β_t: noise schedule, defined as a function of time t.
- $\alpha_t := 1 - \beta_t$.
- $\bar{\alpha}_t := \prod_{s=1}^{t} \alpha_s$.
- Forward diffusion process:

$$q\left(\mathbf{x}_t \mid \mathbf{x}_0\right) = \mathcal{N}\left(\sqrt{\bar{\alpha}_t}\mathbf{x}_0, \left(1-\bar{\alpha}_t\right)\mathbf{I}\right) \tag{1}$$

- Reverse diffusion process using a neural network ϵ_θ:

$$\mu_\theta\left(\mathbf{x}_t, t\right) = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta\left(\mathbf{x}_t, t\right)\right) \tag{2}$$

$$q_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t) = \mathcal{N}\left(\mu_\theta\left(\mathbf{x}_t, t\right), \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t\mathbf{I}\right) \tag{3}$$

(1) Diffuser Design

(a) Design a `Diffuser` class that implements both the forward and reverse diffusion processes described above. Plot α_t, $\bar{\alpha}_t$ and β_t as functions of t. Use **100** diffusion steps ($T = 100$) and the cosine noise schedule from Improved Denoising Diffusion Probabilistic Models:

"*We construct a different noise schedule in terms of* $\bar{\alpha}_t$:

$$\bar{\alpha}_t = \frac{f(t)}{f(0)}, \quad f(t) = \cos\left(\frac{t/T + s}{1+s} \cdot \frac{\pi}{2}\right)^2 \tag{4}$$

To go from this definition to variances β_t*, we note that* $\beta_t = 1 - \frac{\bar{\alpha}_t}{\bar{\alpha}_{t-1}}$. *In practice, we clip* β_t *to be no larger than 0.999 to prevent singularities at the end of the diffusion process near* $t = T$. "

(b) Based on the plot and Eq.2, briefly discuss what problems may arise during the reverse diffusion process when t is close to T?

(2) Circle Dataset

Generate a dataset with 100 samples $\mathbf{x} = [x, y]$ uniformly distributed on a circle defined by:

$$x^2 + y^2 = 1.5^2 \tag{5}$$

Plot the data distribution under the forward diffusion process at time steps $t = 0$, $t = 10$, $t = 20$, $t = 50$, and $t = 100$.

(3) Diffusion Training

Train a diffusion model to generate the circle dataset using your implemented `Diffuser`. Recommended training configuration:

- A simple MLP with 5 linear layers and LeakyReLU activation functions.
- Use a diffusion time embedding dimension of 32 and a hidden dimension of 512 in MLP.
- Adam optimizer with a learning rate of 10^{-3}.
- Train for 1000 epochs with a batch size of 25.

Plot the data distribution generated by the trained diffusion model. Also, compute and report the average *physical loss*, defined as:

$$\mathcal{L}_{\text{Phy}} = |\sqrt{x^2 + y^2} - 1.5| \tag{6}$$

(4) Physics-based Diffusion Sampling

To incorporate physical constraints during sampling, use Tweedie's formula to estimate $\mathbf{x}_0$ from a noisy state $\mathbf{x}_t$:

$$\mathbb{E}\left[\hat{\mathbf{x}}_0 \mid \mathbf{x}_t; \theta\right] = \left(\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta\left(\mathbf{x}_t, t\right)\right) / \sqrt{\bar{\alpha}_t} \tag{7}$$

Implement a physics-guided sampling procedure, where $\mathbf{x}_t$ is corrected after each reverse diffusion step using the gradient of the physical loss:

$$\mathbf{x}_t = \mathbf{x}_t - \nabla_{\mathbf{x}_t} \mathcal{L}_{\text{Phy}}\left(\mathbb{E}\left[\hat{\mathbf{x}}_0 \mid \mathbf{x}_t; \theta\right]\right) \tag{8}$$

Plot the data distribution generated by this physics-based sampling method and report the average physical loss of the sampled data.

Submission instruction

Please upload a single PDF file containing your results along with your code for implementation tasks or your derivation for non-implementation tasks (LaTex typesetting). The uploaded PDF should only include the final code, so please trim empty spaces and your intermediate work before submitting.

The easiest way to generate such a PDF is by using Jupyter notebooks and LaTex (we recommend MiKTeX for Windows users). With Jupyter and LaTeX installed, you can create a PDF from your notebook by running *jupyter nbconvert –to pdf your-notebook.ipynb*

Additional information

This is an individual assignment. Plagiarism will result in the loss of eligibility for the bonus this semester.

If you have any questions about the exercises, please contact us via the forum on Moodle. If you need further face-to-face discussion, please join our weekly online Q&A session (every Monday at 15:00 and 16:00 via BBB).