Mathematics of Generative Models : Homework 1

Nisha Chandramoorthy (nishac@uchicago.edu)

In this homework, we will explore probability flows in 2D.

1. Consider the ODE

$$\frac{dx}{dt} = v(x, t) \quad t \in [0, 1]. \tag{1}$$

The solution to the ODE above at time t starting with the initial condition $x \in \mathbb{R}^2$ is given by $\varphi^t(x) \in \mathbb{R}^2$. The function $(t, x) \to \varphi^t(x)$ is also called the flow of the vector field v. Now, we will consider an ensemble of initial conditions distributed according to a probability density ρ_0. Derive the continuity equation,

$$\frac{\partial \rho_t}{\partial t} = -\text{div}(\rho_t \, v(\cdot, t)), \tag{2}$$

which gives us the probability density, ρ_t, of $\varphi^t(x)$ when the x is a sample from the density ρ_0. (5 points)

2. Using the previous part or otherwise, derive the ODE for the evolution of the log density along a sample path,

$$\frac{d \log(\rho_t(\varphi^t(x)))}{dt} = -\text{div}(v(\cdot, t))(\varphi^t(x)). \tag{3}$$

(5 points)

3. Let ρ_1 be a bimodal Gaussian distribution, $\rho_1(x) = w_1 (2\pi)|\Sigma_1|^{-1/2} e^{-(x-\mu_1)^\top \Sigma_1^{-1}(x-\mu_1)} + w_2 (2\pi) |\Sigma_2|^{-1/2} e^{-(x-\mu_2)^\top \Sigma_2^{-1}(x-\mu_2)}$, with the following values:

$$w_1 = 0.2, \ w_2 = 0.8, \ \mu_1 = \begin{bmatrix} -2 \\ -2 \end{bmatrix}, \ \mu_2 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}, \ \Sigma_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \ \Sigma_2 = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \tag{4}$$

How will you generate samples from this density? You can write a pseudocode or explain the logic. (5 points)

4. Let the samples generated from the previous part be $x_1, \cdots, x_m$. For any sample $x_i \sim \rho_1$, set $\log \rho_1(x_i) = 0$. Initialize a neural network, v, that takes in x, t and approximates the vector $v(x, t) \in \mathbb{R}^2$ in (1) such that ρ_0 is a standard normal in 2D. We will do this by maximizing the log likelihood of the samples. Although in this case, we have the target density in closed form, we generally do not. So, using the target samples $x_1, \cdots, x_m$, we will follow this procedure to estimate the log likelihood of the samples:

- Time integrate the flow, (1), backward in time, using the final condition, $\varphi^1(x) = x_i, i \in [m]$.

- Solve the log probability equation (3) backward in time. The solution at time $t = 0$ is given by $\log \rho_0(x_0)$, where $\varphi^1(x_0) = x_1$. You can evaluate $\log \rho_0$ anywhere since ρ_0 is a Gaussian distribution.

- Then, recognize from (3) that $\log \rho_1(x_i) = \log \rho_0(x) - \int_0^1 \mathrm{div}(v(\cdot, t))(\varphi^t(x))\, dt$.

- Set up the maximum likelihood (ML) problem to maximize $\sum_{i=1}^m \log \rho_1(x_i)$.

In the above ML problem, we seek the parameters of the NN, v, to maximize $\sum_{i=1}^m \log \rho_1(x_i) = \log \rho_0(x) - \int_0^1 \mathrm{div}(v(\cdot, t))(\varphi^t(x))\, dt$.

Use a standard optimizer such as Stochastic Gradient Descent or ADAM and report the following results:

1. Plot the vector field as contour plots at a few times. What do you observe about the smoothness of your vector field? (5 points)

2. Generate sample points (approximately) from p_{data} using the trained vector field. Briefly state the procedure to generate. Plot a kernel density estimate using the generated points and compare against p_{data} (10 points)

3. How good are your results? In other words, give some quantitative metrics for how well the generated samples reflect p_{data}. What can you change to make your generative model better? (5 points)