Normalizing Flows (Part 3)

Dr. Alireza Aghamohammadi

Forward and Inverse Mappings

In normalizing flows, the forward and inverse mappings between distributions are implemented using neural networks:

- ❖ The forward mapping $f(z, \phi)$ is typically represented as a neural network composed of several layers, each denoted by $f_k(\bullet, \phi_k)$, where ϕ_k are the parameters of the k-th layer.
- ❖ The inverse mapping is obtained by applying the inverse of each layer, $f_k^{-1}(\bullet, \phi_k)$, in the reverse order. This enables the transformation from a complex distribution back to a simpler, known prior.

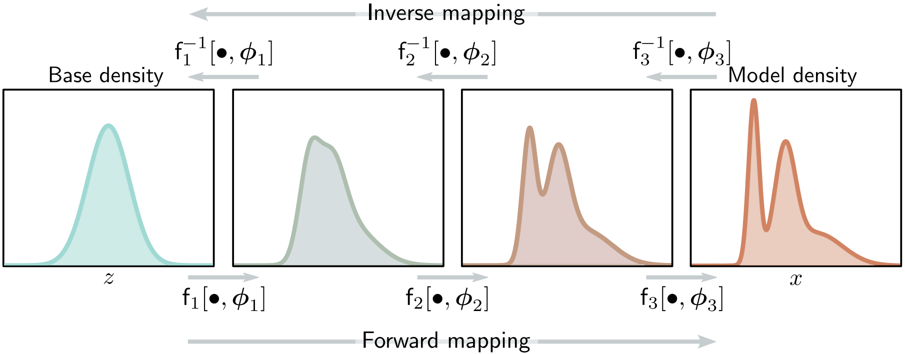

Image credit: Understanding Deep Learning

Training Normalizing Flows

We train normalizing flows using a dataset $\{x_i\}$ of I training examples with the negative log-likelihood criterion:

$$\hat{\phi} = \underset{\phi}{\text{argmin}} \left[\sum_{i=1}^{I} \log |\det(J_f(z_i, \phi))| - \log [P_r(z_i)] \right]$$

where $z_i = f^{-1}(x_i, \phi)$ and

$$|\det(J_f(z_i, \phi))| = |\det(J_{f_k}(f_{k-1}, \phi_k))| \cdot \left|\det(J_{f_{k-1}}(f_{k-2}, \phi_{k-1}))\right| \cdots |\det(J_{f_1}(z, \phi_1))|$$

Requirements for Practical Normalizing Flows

To make normalizing flows practical, the neural network layers f_k must satisfy the following four properties:

1. The network layers must be collectively expressive enough to transform a multivariate standard normal distribution into any arbitrary density.
2. Each network layer must be invertible.
3. The inverse of each layer must be computable efficiently.
4. The determinant of the Jacobian for each layer must be computable efficiently.