MO433 — Unsupervised Machine Learning
INSTITUTO DE COMPUTAÇÃO — UNICAMP

Flow Models

Lesson No. 03

Leonardo Alves de Melo *

1 Introduction

A Generative Model aims to create samples of high-dimensional data based on the distribution of the real data, for example generating new images related to a set of images. This kind of model is considered an Unsupervised Learning Model, because there is no need to label each image in the dataset. One kind of Generative Model is the Autoregressive Model, which achieved the state-of-the-art with high quality of sampling, but on the other hand this process can be very slow.

In this summary it will be presented the Flow Models, which is a kind of Generative Models that has a faster sampling process, and has a state-of-the art model presented by the paper of Ho et. al. [1].

2 Flows

The main idea of a Flow Model is transforming the distribution of data in an invertible and stable function that maps the distribution $x \sim p_x$ and latent distribution $z \sim p_z$, where the latent distribution is typically a Gaussian [2]. An example of the effect of a Flow Function in a dataset of toys can be found in Figure 1.

2.1 Formal Definition of Flow

A function f can be considered a Flow if it maps observed data x to a latent space in the form $z = f(x)$ and it is composed of L functions of the form $f(x) = f_1 \circ f_2 \circ ... \circ f_L(x)$, where each f_i is invertible and has a Jacobian.

The maximum likelihood formula for flow models, which can be seen in Equation 2, can be calculated using the Equation 1 and assuming that $z_i \sim \mathcal{N}(0, \boldsymbol{I})$.

$$p_i(\boldsymbol{z}_i) = p_{i-1}(f_i^{-1}(\boldsymbol{z}_i)) \left| \det \frac{\delta f_i^{-1}}{\delta \boldsymbol{z}_i} \right| \tag{1}$$

*RA: 156188 - leonardo.alves.melo.1995@gmail.com

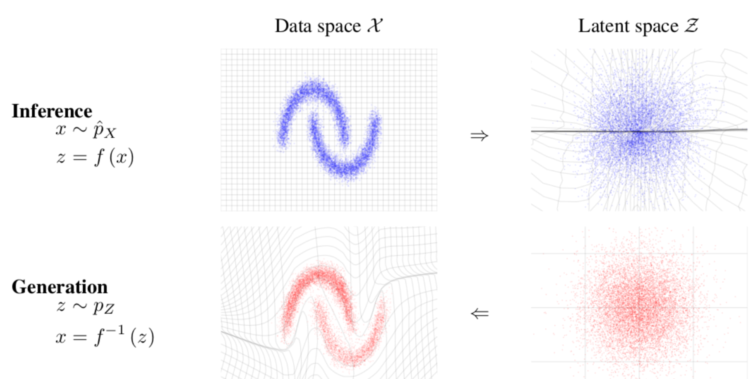

Figure 1. Exemplo of flow function in a toy dataset.

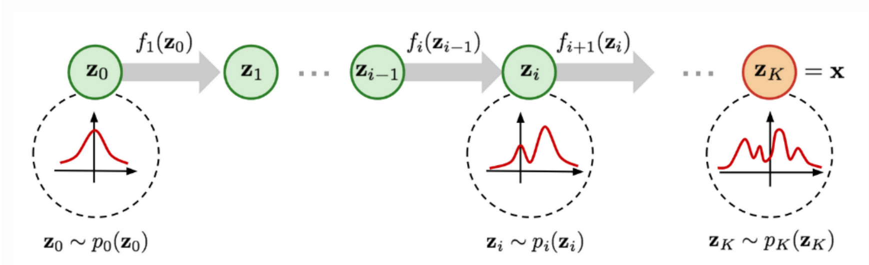

Figure 2. Coupling of function defining a complex flow model **weng2018flow**.

$$\log p(x) = \log \mathcal{N}(f(x); 0, I) + \sum_{i=1}^{L} \log \left| \det \frac{\delta f_i}{\delta f_{i-1}} \right| \tag{2}$$

A representation of multiple features passing through the flow model can be seen in Figure 2, where the group of all of them creates z_K, which is a sample from the distribution.

2.2 Flow++

Flow++ is the state-of-the-art of flows, reaching good sample quality by improving and adding three components to the Flow algorithm. The first one is Variational Dequantization instead of Uniform Dequantization; Dequantization tries to transform discrete data into a continuous one, and the variational approach is better for neural networks because it relaxes the restriction of the noise, establishing a variational lower bound in the fashion of VAEs. The second one is the use of a mixture os logistic functions as the flow function for CDF, as can be seen in Equation 3, where

$$\text{MixLogCDF}(x; \pi, \mu, \mathbf{s}) := \sum_{i=1}^{K} \pi_i \sigma((x - \mu_i).\exp(-s_i))$$

; this is better than the affine coupling, which is bad for non-linearity.

$$\begin{cases} \mathbf{y}_1 & = \mathbf{x}_1 \\ \mathbf{y}_2 & = \sigma^{-1}(\text{MixLogCDF}(\mathbf{x}_2; \pi_\theta(\mathbf{x}_1), \mu_\theta(\mathbf{x}_1), \mathbf{s}_\theta(\mathbf{x}_1))) \end{cases} \tag{3}$$

The third improvement is the use of self-attention with Residual Networks (ResNets) with 1x1 convolutional layer, producing more powerful estimation, as can be seen below:

$$\begin{aligned} \text{Conv} &= \text{Input} \to \text{Nonlinearity} \to \text{Conv}_{3\times 3} \to \text{Nonlinearity} \to \text{Gate} \\ \text{Attn} &= \text{Input} \to \text{Conv}_{1\times 1} \to \text{MultiHeadSelfAttention} \to \text{Gate} \end{aligned}$$

3 Conclusion

Flow Models can reach the autoregressive sampling quality without the need of large amount of time in the sampling process. The Flow++ is a great improvement of previous flows, being the current state-of-the-art.

References

[1] J. Ho, X. Chen, A. Srinivas, Y. Duan, and P. Abbeel, "Flow++: Improving flow-based generative models with variational dequantization and architecture design," *arXiv preprint arXiv:1902.00275*, 2019.

[2] L. Dinh, J. Sohl-Dickstein, and S. Bengio, *Density estimation using real nvp*, 2016. arXiv: 1605.08803 [cs.LG].