Normalizing Flows

Abdul Fatir Ansari[1], Devamanyu Hazarika[1], and Remmy A. M. Zen[1]

National University of Singapore
{abdulfatir,devamanyu,remmy}@u.nus.edu

Abstract. **Keywords:** Normalizing flows · .

1 Introduction

2 Background

2.1 Autoencoder

2.2 Variational Autoencoder

3 Normalizing Flows

Before defining normalizing flows, let's consider a univariate distribution with density function $p(x)$. Define a continuous, differentiable, and increasing function f. Define $y = f(x)$ where $x \sim p(x)$. The density function of the random variable Y can then be derived analytically using the Cumulative Distribution Function (CDF) as follows.

$$
\begin{aligned}
F_Y(y) &= P(Y \leq y) && (1)\\
&= P(f(X) \leq y) && (2)\\
&= P(X \leq f^{-1}(y)) = F_X(f^{-1}(y)) && (3)
\end{aligned}
$$

We end up with the CDF of the random variable X at the point $f^{-1}(y)$. Now, $p(y) = F'_Y(y)$ by definition, where

$$F_Y(y) = F_X(f^{-1}(y)) = \int_{-\infty}^{f^{-1}(y)} p(x)dx \tag{4}$$

Differentiating Eq. (4) with respect to y (using the Fundamental Theorem of Calculus and the chain rule) we get

$$p(y) = p(f^{-1}(y)) \cdot \frac{df^{-1}}{dy} \tag{5}$$

When f is a decreasing function, we get $p(y) = p(f^{-1}(y)) \cdot \frac{df^{-1}}{dy}$. For an invertible function in general, Eq. (5) can be written as

$$p(y) = p(f^{-1}(y)) \cdot \left| \frac{df^{-1}}{dy} \right| \tag{6}$$

Eq. (6) can be extended to the multivariate case where the derivative is replaced by the determinant of the Jacobian matrix J

$$p(\mathbf{y}) = p(f^{-1}(\mathbf{y})) \cdot \left| \det \frac{\partial f^{-1}}{\partial \mathbf{y}} \right| = p(f^{-1}(\mathbf{y})) \cdot \left| \det \frac{\partial f}{\partial f^{-1}(\mathbf{y})} \right|^{-1} \tag{7}$$

In the above equation, the second equality comes from the inverse function theorem. Successive applications of such smooth, invertible transformation on a random variable with known density is called a *normalizing flow*.

Computation of the probability density of the transformed random variable requires the computation of the determinant of the Jacobian matrix which is computationally expensive as it scales with $O(d^3)$ where d is the dimensionality of the random variable. Developing transformations with cheap determinant computation has been the primary focus of many recent works.

4 Applications

Literature on normalizing flows can be broadly classified into two parts: ones using normalizing flows for improved variational inference and ones using normalizing flows for density estimation.

4.1 Variational Inference

Variational methods perform inference by approximating the true posterior $p(z|x)$ using a simpler variational family $q_\phi(z|x)$. Recent works have focused on improving the variational posterior used in the VAE which is generally set to a multivariate normal distribution with diagonal covariance matrix $\mathcal{N}(\boldsymbol{\mu}, \sigma^2\mathbf{I})$. It is clear that such a simplistic, unimodal choice for the posterior can be arbitrarily far away from the true posterior which can be a complex multi-modal distribution.

Recent works seek to convert samples from a simple variational posterior (such as the multivariate normal distribution) into a richer distribution by applying a series of smooth, invertible transformations or a flow. Let $\mathbf{z}_0$ be a sample from a simple distribution $q_0(\mathbf{z}_0)$ and $\mathbf{z}_K$ be a sample obtained by applying a flow of length K on $\mathbf{z}_0$, i.e., $\mathbf{z}_K = f_K \circ f_{K-1} \circ \cdots \circ f_1(\mathbf{z}_0)$. Using Eq. (7), the density function $q_K(\mathbf{z}_K)$ is given by

$$q_K(\mathbf{z}_K) = q_0(\mathbf{z}_0) \prod_{k=1}^{K} \left| \det \frac{\partial f_k}{\partial \mathbf{z}_{k-1}} \right|^{-1} \tag{8}$$

The variational lower bound (or evidence lower bound) in VAEs (Eq. ()) can now be modified by setting $q(\mathbf{z}|\mathbf{x}) = q_K(\mathbf{z}_K|\mathbf{x})$

$$\begin{aligned} \mathcal{L} &= \mathbb{E}_{q(\mathbf{z}_K|\mathbf{x})} [\log p_\theta(\mathbf{x}, \mathbf{z}_K) - \log q(\mathbf{z}_K|\mathbf{x})] & (9) \\ &= \mathbb{E}_{q(\mathbf{z}_0|\mathbf{x})} [\log p_\theta(\mathbf{x}, \mathbf{z}_K) - \log q(\mathbf{z}_K|\mathbf{x})] & (10) \end{aligned}$$

where $q(\mathbf{z}_0|\mathbf{x})$ is the simple initial density. Plugging in Eq. (8) into Eq. (10), we get a modified bound for flow-based VAEs

$$\mathcal{L} = \mathbb{E}_{q_0(\mathbf{z_0}|\mathbf{x})} \left[\log p(\mathbf{x}, \mathbf{z}_K) - \log q_0(\mathbf{z}_0|\mathbf{x}) + \sum_{k=1}^{K} \log \left| \det \frac{\partial f_k}{\partial \mathbf{z}_{k-1}} \right| \right] \tag{11}$$

Planar and Radial Flows Planar and Radial Flows [5] are one of the earliest flows proposed in the context of variational inference.

Planar flows use functions of the form

$$f(\mathbf{z}) = \mathbf{z} + \mathbf{u}h(\mathbf{w}^\top \mathbf{z} + b) \tag{12}$$

where $\mathbf{u}, \mathbf{w} \in \mathbb{R}^d$, $b \in \mathbb{R}$, and h is an element-wise non-linearity such as tanh. The Jacobian is then given by

$$\det \frac{\partial f(\mathbf{z})}{\partial \mathbf{z}} = (1 + h'(\mathbf{w}^\top \mathbf{z} + b)\mathbf{w}^\top \mathbf{u}) \tag{13}$$

which can be computed in $O(d)$ time.

Radial flows use functions of the form

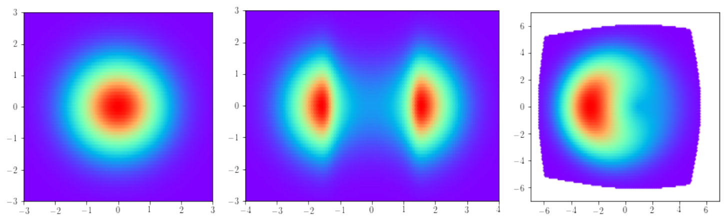
$$f(\mathbf{z}) = \mathbf{z} + \beta h(\alpha, r)(\mathbf{z} - \mathbf{z}_0) \tag{14}$$

where $\alpha \in \mathbb{R}^+$, $\beta \in \mathbb{R}$, $h(\alpha, r) = (\alpha + r)^{-1}$ and $r = ||\mathbf{z} - \mathbf{z}_0||$. The Jacobian is then given by

$$\det \frac{\partial f(\mathbf{z})}{\partial \mathbf{z}} = (1 + \beta h(\alpha, r) + \beta h'(\alpha, r) r) (1 + \beta h(\alpha, r))^{d-1} \tag{15}$$

For a detailed derivation of Jacobians of Planar and Radial flows please refer Appendix. Fig. 1 shows how planar and radial flows change a standard normal distribution.

Inverse Autoregressive Flows [3]



(a) Initial Density (b) Planar Flow (c) Radial Flow

Fig. 1: Change in standard normal density on application of length 1 planar and radial flows.

4.2 Density Estimation

Non-linear Independent Components Estimation

Real-valued Non-Volume Preserving

5 Normalizing Flows in Probabilistic Programming Languages

[1]

6 Recent Advances

6.1 Pixel Recurrent Neural Network

[4]

6.2 Wavenet

[6]

6.3 Glow

[2]

7 Conclusion

8 Contribution

References

1. Dillon, J.V., Langmore, I., Tran, D., Brevdo, E., Vasudevan, S., Moore, D., Patton, B., Alemi, A., Hoffman, M., Saurous, R.A.: Tensorflow distributions. arXiv preprint arXiv:1711.10604 (2017)
2. Kingma, D.P., Dhariwal, P.: Glow: Generative flow with invertible 1x1 convolutions. arXiv preprint arXiv:1807.03039 (2018)
3. Kingma, D.P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I., Welling, M.: Improved variational inference with inverse autoregressive flow. In: Advances in Neural Information Processing Systems. pp. 4743–4751 (2016)
4. Oord, A.v.d., Kalchbrenner, N., Kavukcuoglu, K.: Pixel recurrent neural networks. arXiv preprint arXiv:1601.06759 (2016)
5. Rezende, D., Mohamed, S.: Variational inference with normalizing flows. In: International Conference on Machine Learning. pp. 1530–1538 (2015)
6. Van Den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A.W., Kavukcuoglu, K.: Wavenet: A generative model for raw audio. In: SSW. p. 125 (2016)

A Appendix (Abdul Fatir Ansari)

The Jacobian is then given by

$$\frac{\partial f(\mathbf{z})}{\partial \mathbf{z}} = \mathbf{I} + \mathbf{u}h'(\mathbf{w}^\top \mathbf{z} + b)\mathbf{w}^\top$$

Now, using the matrix determinant lemma

$$\det \frac{\partial f(\mathbf{z})}{\partial \mathbf{z}} = (1 + h'(\mathbf{w}^\top \mathbf{z} + b)\mathbf{w}^\top \mathbf{I}^{-1}\mathbf{u}) \det(\mathbf{I}) \tag{16}$$

$$= (1 + h'(\mathbf{w}^\top \mathbf{z} + b)\mathbf{w}^\top \mathbf{u}) \tag{17}$$

B Appendix (Devamanyu Hazarika)

C Appendix (Remmy Zen)