Flow Contrastive Estimation of Energy-Based Models
Supplementary materials

Ruiqi Gao[1], Erik Nijkamp[1] , Diederik P. Kingma[2], Zhen Xu[2], Andrew M. Dai[2], Ying Nian Wu[1]
[1] UCLA, [2] Google
{ruiqigao, enijkamp}@ucla.edu, {durk, zhenxu, adai}@google.com, ywu@stat.ucla.edu

1. Model architectures

Table 1 summarizes the EBM architectures used in unsupervised learning (subsections 4.1-4.3). The slope of all leaky ReLU (lReLU) [2] functions are set to 0.2. For semi-supervised learning from a 2D example (subsection 4.4), we use the same EBM structure as the one used in unsupervised learning from 2D examples, except that for the top fully connect layer, we change the number of output channels to 2, to model EBMs of two categories respectively. Table 2 summarizes the EBM architectures used in semi-supervised learning from SVHN (subsection 4.4). After each convolutional layer, a weight normalization [3] layer and a leaky ReLU layer is added. The slope of leaky ReLU functions is set to 0.2. A weight normalization layer is added after the top fully connected layer.

Table 1: EBM architectures used in unsupervised learning

2D data	SVHN / CIFAR-10
fc. 128 lReLU	4×4 conv. 64 lReLU, stride 2
fc. 128 lReLU	4×4 conv. 128 lReLU, stride 2
fc. 128 lReLU	4×4 conv. 256 lReLU, stride 2
fc. 1	4×4 conv. 1, stride 1

For Glow model, we follow the setting of [1]. The architecture has multi-scales with levels L. Within each level, there are K flow blocks. Each block has three convolutional layers (or fully-connected layers) with a width of W channels. After the first two layers, a ReLU activation is added. Table 3 summarizes the hyperparameters for different datasets.

2. Synthesis comparison

In figures 1, 2 and 3, we display the synthesized examples from Glow trained by MLE and our FCE.

Table 2: EBM architectures used in semi-supervised learning from SVHN

Conv-small	Conv-large
dropout, $p = 0.2$	
3×3 conv. 64, stride 1	3×3 conv. 128, stride 1
3×3 conv. 64, stride 1	3×3 conv. 128, stride 1
3×3 conv. 64, stride 2	3×3 conv. 128, stride 2
dropout, $p = 0.5$	
3×3 conv. 128, stride 1	3×3 conv. 256, stride 1
3×3 conv. 128, stride 1	3×3 conv. 256, stride 1
3×3 conv. 128, stride 2	3×3 conv. 256, stride 2
dropout, $p = 0.5$	
3×3 conv. 128, stride 1	3×3 conv. 512, stride 1
1×1 conv. 128, stride 1	1×1 conv. 256, stride 1
1×1 conv. 128, stride 1	1×1 conv. 128, stride 1
global max pool, $6 \times 6 \rightarrow 1 \times 1$	
fc. $128 \rightarrow 10$	

Figure 1: Synthesized examples from Glow models learned from SVHN. Left panel is by MLE. Right panel is by our FCE.

References

[1] Diederik P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In *Advances in Neural*

Table 3: Hyperparameters for Glow model architectures

Dataset	Levels L	Blocks per level K	Width W	Layer type	Coupling
2D data	1	10	128	fc	affine
SVHN	3	8	512	conv	additive
CelebA	3	16	512	conv	additive
CIFAR-10	3	32	512	conv	additive

Figure 2: Synthesized examples from Glow models learned from CIFAR-10. Left panel is by MLE. Right panel is by our FCE.

Figure 3: Synthesized examples from Glow models learned from CelebA. Left panel is by MLE. Right panel is by our FCE.

Information Processing Systems, pages 10215–10224, 2018. 1

[2] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. Rectifier nonlinearities improve neural network acoustic models. In *Proc. icml*, volume 30, page 3, 2013. 1

[3] Tim Salimans and Durk P Kingma. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. In *Advances in Neural Information Processing Systems*, pages 901–909, 2016. 1