Supplementary Material of “Learning Deep Latent Variable Models by Short-Run MCMC Inference with Optimal Transport Correction”

Dongsheng An, Jianwen Xie, Ping Li
Cognitive Computing Lab, Baidu Research
10900 NE 8th St. Bellevue, WA 98004, USA
{dongshengan15, jianwen.kenny, pingli98}@gmail.com

In the supplementary material, we will provide more experimental details to support the main text of the paper.

1. Experimental Details

Datasets In the experiments, we mainly use the MNIST datatset [3] ($28 \times 28 \times 1$), SVHN dataset [4] ($32 \times 32 \times 3$) and CelebA dataset [5] ($64 \times 64 \times 3$). For the first two datasets, we use all of the samples in the training set, namely 60,000 for the MNIST dataset and 73,257 for the SVHN dataset. For the CelebA dataset, we randomly select 60,000 images for the purpose of quick convergence. All of the training images are resized and scaled to the range of $[-1, 1]$.

Model architectures The architectures of the models are presented in Tab. 1, where the numbers of latent dimensions are set to be $30, 64, 64$ for the MNIST dataset, SVHN dataset and CelebA dataset, respectively.

Optimization The parameters for the generators are initialized with Xavier normal [1] and then optimized with the Adam optimizer [2] with $\beta_1 = 0.5$ and $\beta_2 = 0.99$. For all of the experiments, we set the batch size to be 2,000. In Alg. 1 of the paper, both L and K are set to be 50. The hyperparameter α is set to be 0.5 for the MNIST dataset, and 0.3 for the SVHN and CelebA datasets. The step sizes s for MNIST, SVHN and CelebA datasets are set to be 0.3, 3.0, 3.0, respectively. We also set $\sigma = 0.3$ for all of the models.

Computational cost Due to the involvement of the short-run MCMC and the optimal transport, it is necessary to consider the running time of the whole pipeline. Here we take the SVHN dataset which includes 73,257 images with the size $32 \times 32 \times 3$ as an example. We train our model on two NVIDIA TitanX GPUs. For each iteration, the inference step with $K = 30$ takes about 124 minutes, the correction step by optimal transport takes about 10 minutes and the learning step with $L_2 = 2$ takes 5 minutes. Generally, we need to run $10 \sim 15$ iterations for the model, which will consume about one day.

References

[1] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 249–256, 2010. 1

[2] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2014. 1

[3] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, pages 2278–2324, 1998. 1

[4] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011. 1

[5] Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. From facial expression recognition to interpersonal relation prediction. *International Journal of Computer Vision (IJCV)*, 126(5):550–569, 2018. 1

Table 1. The architectures of the generators for different datasets.

Model	layer	number of outputs	kernel size	stride	padding	output_padding	BN	activation
MNIST	Input z	30	-	-	-	-	-	-
	Linear	1024	-	-	-	-	yes	ReLU
	Linear	7*7*128	-	-	-	-	Yes	ReLU
	convT	14*14*64	2*2	2	-	-	Yes	ReLU
	convT	28*28*3	2*2	2	-	-	-	Tanh
SVHN	Input z	64	-	-	-	-	-	-
	convT	2*2*64*8	-	-	-	-	-	
	convT	4*4*64*4	5*5	2	2	1	Yes	ReLU
	convT	8*8*64*2	5*5	2	2	1	Yes	ReLU
	convT	16*16*64	5*5	2	2	1	Yes	ReLU
	convT	32*32*3	5*5	2	2	1	-	Tanh
CelebA	Input z	64	-	-	-	-	-	-
	convT	4*4*128*8	-	-	-	-	-	-
	convT	8*8*128*4	5*5	2	2	1	Yes	ReLU
	convT	16*16*128*2	5*5	2	2	1	Yes	ReLU
	convT	32*32*128	5*5	2	2	1	Yes	ReLU
	convT	64*64*3	5*5	2	2	1	-	Tanh