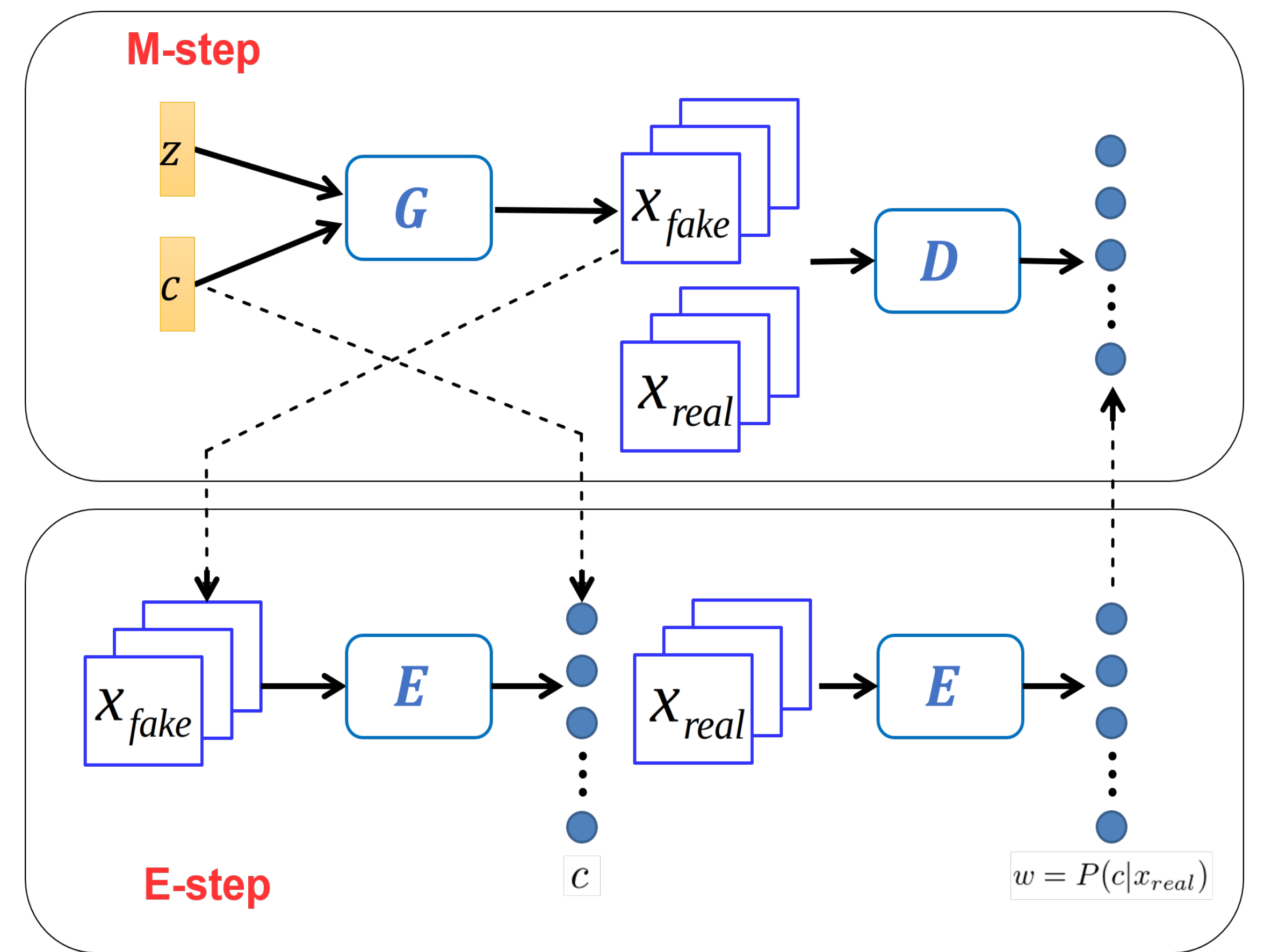


Introduction

- Expectation maximization (EM) algorithm is to find the maximum likelihood solution for models having latent variables. A typical example is the Gaussian mixture model (GMM), which requires a Gaussian assumption. However, natural images are highly non-Gaussian so that GMM is not ideal for performing image clustering task on pixel space.
- Generative adversarial network (GAN) has been proved to be powerful in learning data distribution. Previous work has shown that GAN can be modified to perform maximum likelihood estimation (MLE).
- Instead of applying GMM on the image clustering task, we propose a GAN based EM learning framework that can maximize the likelihood of images and estimate the latent variables. We call this model GAN-EM, which is a framework for image clustering, semi-supervised classification, and dimensionality reduction. In M-step, we design a novel loss function for discriminator of GAN to perform MLE on data with soft class label assignments.
- The GAN-EM model outperforms most other popular clustering methods. In the meantime, the quality of the generated images is also comparable to different GAN variants. Examples of generated images:



Methodology



M-step: We design a novel loss function for GAN discriminator to perform MLE on data with soft class label assignments. Specifically, a conditional generator captures data distribution for K classes, and a discriminator tells whether a sample is real or fake for each class.

$$L_G = -\frac{1}{2} \mathbb{E}_{c \sim U} \mathbb{E}_{z \sim \mathcal{N}} \exp\{\sigma^{-1}[D_c(G(z, c))]\}$$

$$L_D = \mathbb{E}_{c \sim U} \mathbb{E}_{z \sim \mathcal{N}} \sum_{i=1}^K \log(1 - D_i(G(z, c))) + \mathbb{E}_{x \sim P_r} \sum_{i=1}^K w_i \log(D_i(x))$$

E-step: We introduce another neural network, called E-net, to fit the data distribution. The E-net is trained on the data generated by the generator, acting as a reversed generator.

$$L_E = \mathbb{E}_{z \sim \mathcal{N}} \mathbb{E}_{c \sim \phi} \text{CE}\{E(G(z, c)), u\}$$

Experiments

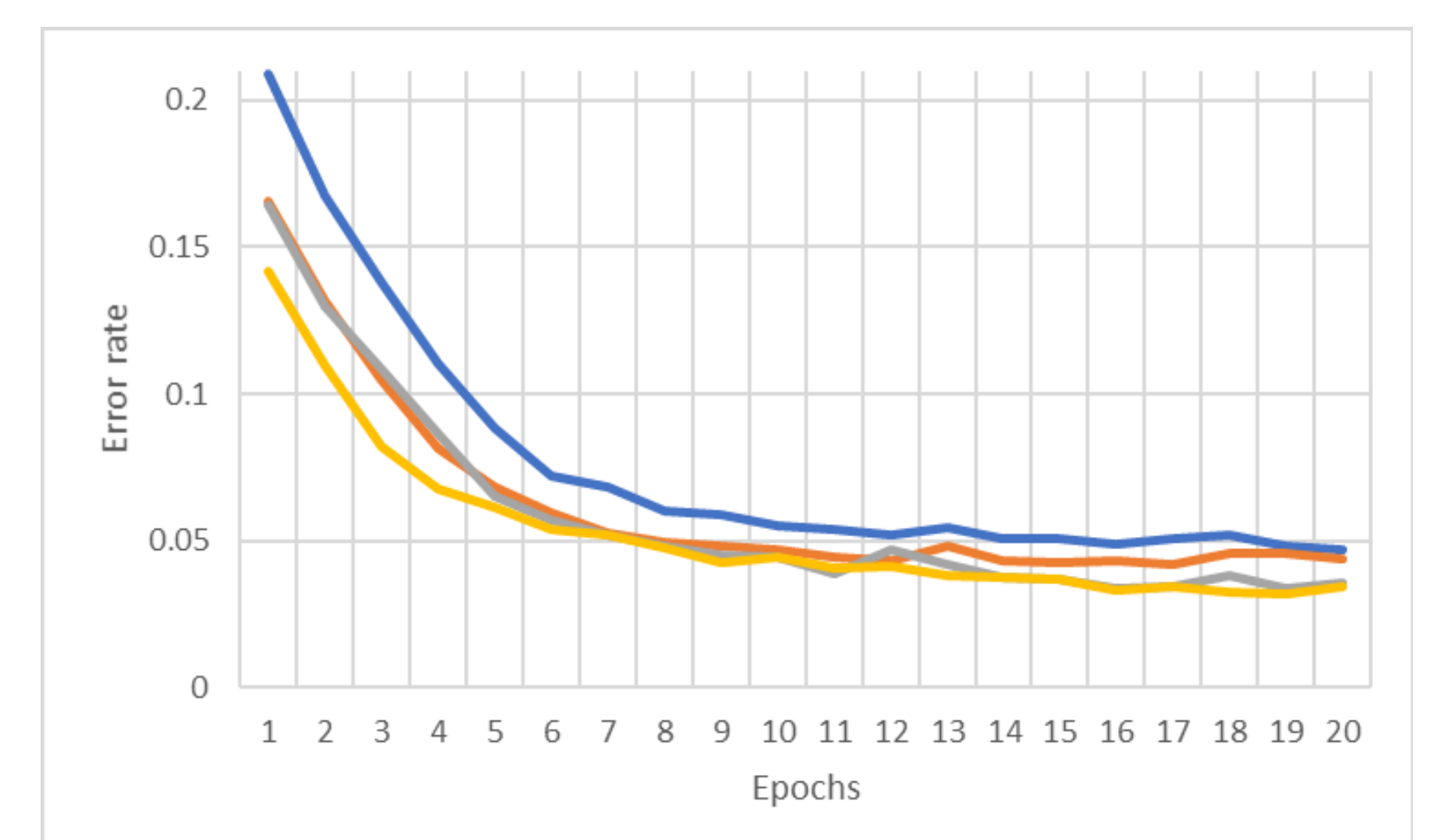
	MNIST (Unsupervised)	MNIST (100 labels)	MNIST (1000 labels)	SVHN (1000 labels)	CelebA (Unsupervised)	CelebA (100 labels)
K-means	46.51 ¹	-	-	-	-	-
GMM	32.61(±0.06) ¹	-	-	-	-	-
DEC	15.7 ¹	-	-	-	-	-
VAE	-	3.33(±0.14)	2.40(±0.02)	36.02(±0.10)	-	45.38
AAE	4.10(± 1.13) ³	1.90(±0.10)	1.60(±0.08)	17.70(±0.30)	42.88	31.03
CatGAN	4.27 ²	1.91(±0.10)	1.73(±0.28)	-	44.57	34.78
InfoGAN	5.00 ⁴	-	-	-	-	-
Improved GAN	-	0.93(±0.06)	-	8.11(±1.30)	-	-
VaDE	5.54 ¹	-	-	-	43.64	-
PixelGAN	5.27(±1.81) ³	1.08(±0.15)	-	6.96(±0.55)	44.27	32.54
GANMM	35.70(±0.45) ¹	-	-	-	49.32	-
GAN-EM	4.20(± 0.51) ¹ 4.04(± 0.42)² 3.97(± 0.37)³	1.09(± 0.18)	1.03(± 0.15)	6.05(±0.26)	42.09	28.82

Clustering: We perform unsupervised clustering on MNIST and CelebA datasets. GAN-EM achieves SOTA results on MNIST with 10, 20, and 30 clusters.

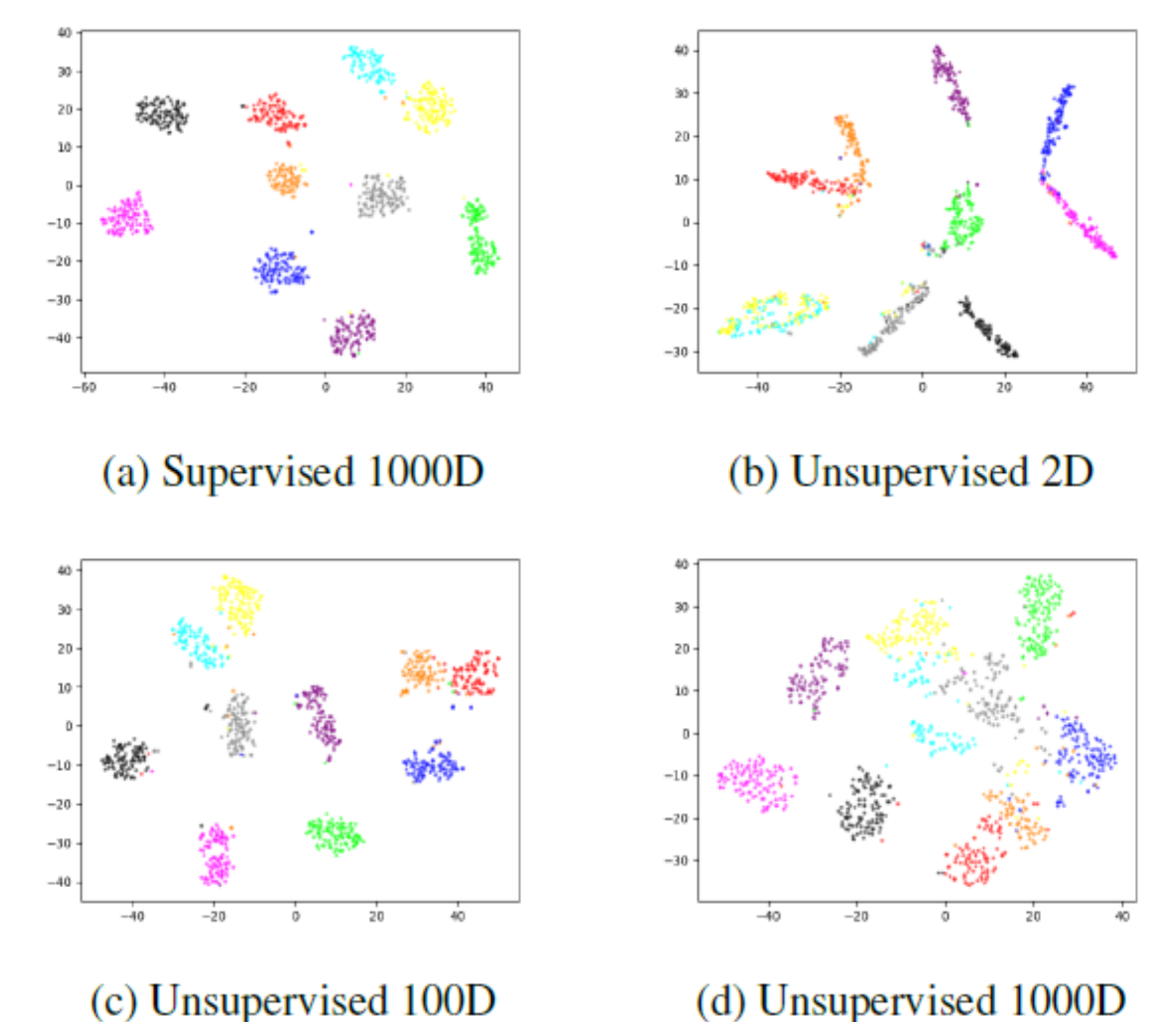
Semi-supervised classification: We extend our model to semi-supervised classification tasks where only a small part of samples' labels are known. We use almost the same training strategies as the clustering task except that we add the supervision to the E-net in every E-step. We evaluate the performance on MNIST, SVHN, and CelebA.

Dimensionality reduction: We add a new layer to the E-net before the output layer. After convergence of training, we take the output of this newly added layer as extracted features after dimensionality reduction.

Acknowledgements. This work was supported in part by NSF IIS 1741472, IIS 1813709, and CHE 1764415. This article solely reflects the opinions and conclusions of its authors and not the funding agents.



Convergence of test error rate of GAN-EM on MNIST.



Representation of unsupervised dimensionality reduction on MNIST. Each color denotes one class of digit.