Ivan Rubachev
HSE, 2019

Semi-supervised Learning with Deep Generative Models

Diederik P. Kingma*, **Danilo J. Rezende**†, **Shakir Mohamed**†, **Max Welling***
*Machine Learning Group, Univ. of Amsterdam, {D.P.Kingma, M.Welling}@uva.nl
†Google Deepmind, {danilor, shakir}@google.com

https://arxiv.org/abs/1406.5298

Semi-supervised learning

- Self training
- Transductive SVMs
- Graph based methods
- …
- *Generative models*

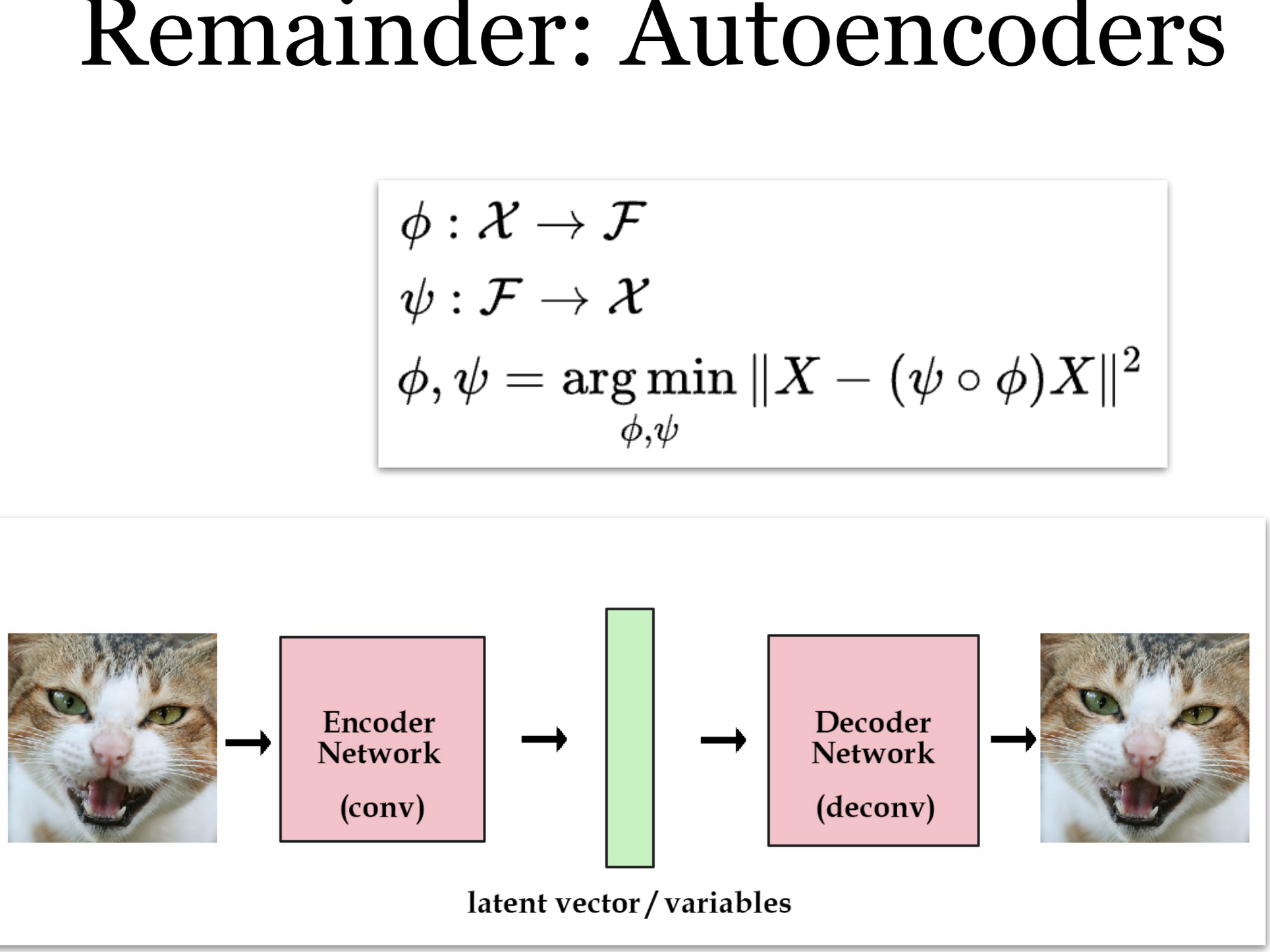
cake

Remainder: Autoencoders

$$\phi : \mathcal{X} \to \mathcal{F}$$

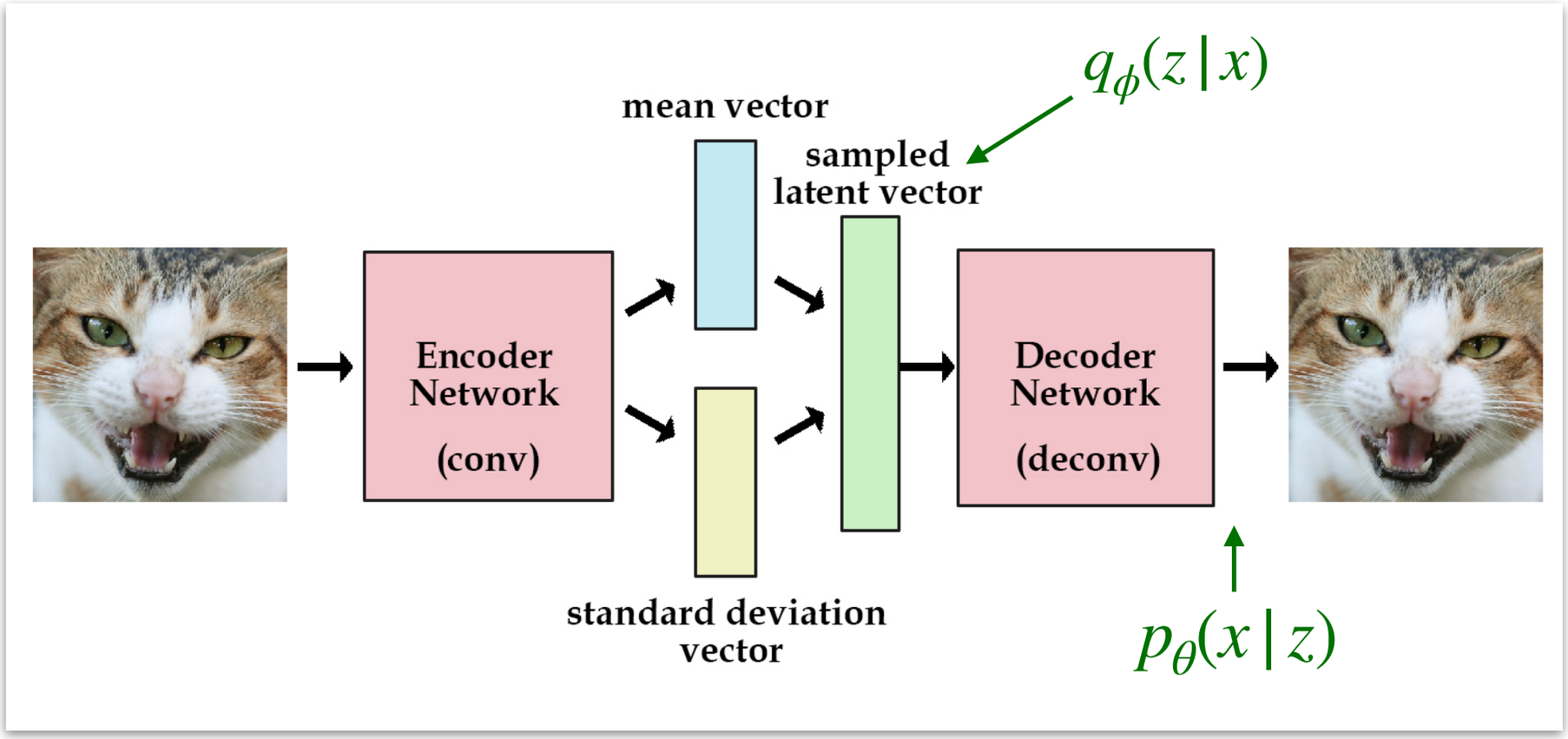
$$\psi : \mathcal{F} \to \mathcal{X}$$

$$\phi, \psi = \underset{\phi,\psi}{\arg\min} \|X - (\psi \circ \phi)X\|^2$$



http://kvfrans.com/variational-autoencoders-explained/
https://towardsdatascience.com/generating-images-with-autoencoders-77fd3a8dd368

Remainder: VAE



Maximize ELBO: $\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] - KL[q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z})]$

http://kvfrans.com/variational-autoencoders-explained/
https://arxiv.org/abs/1406.5298

Remainder: VAE

More details

$X = \{x^{(i)}\}_{i=1}^{N}$ — data

z — latent variables, with prior distribution $p(z)$

x is generated from latent variables $p_\theta(x \mid z)$

Want to find:

- Posterior $p_\theta(z \mid x)$
- Parameters θ

$$p_\theta(z \mid x) = \frac{p_\theta(x \mid z) p_\theta(z)}{\int p_\theta(x \mid z) p_\theta(z) dz}$$

May be intractable

Remainder: VAE

The variational bound

$q_\phi(z|x)$ — approximation of a posterior $p_\theta(z|x)$

$$\log p_\theta\left(x^{(i)}\right) = KL\left(q_\phi\left(z|x^{(i)}\right)\|p_\theta\left(z|x^{(i)}\right)\right) + \mathscr{L}\left(\theta,\phi;x^{(i)}\right)$$

$$\geqslant \mathscr{L}\left(\theta,\phi;x^{(i)}\right) = \mathbb{E}_{q_\phi(z|x)}\left[-\log q_\phi(z|x) + \log p_\theta(x,z)\right]$$

$$= -KL\left(q_\phi\left(z|x^{(i)}\right)\|p_\theta(z)\right) + \mathbb{E}_{q_\phi\left(z|x^{(i)}\right)}\left[\log p_\theta\left(x^{(i)}|z\right)\right]$$

Optimizing the lower bound

(equivalent to minimizing $KL\left(q_\phi(z|x)\|p(z|x)\right)$):

$$\max_{\phi,\theta} \mathscr{L}\left(\theta;\phi;x\right)$$

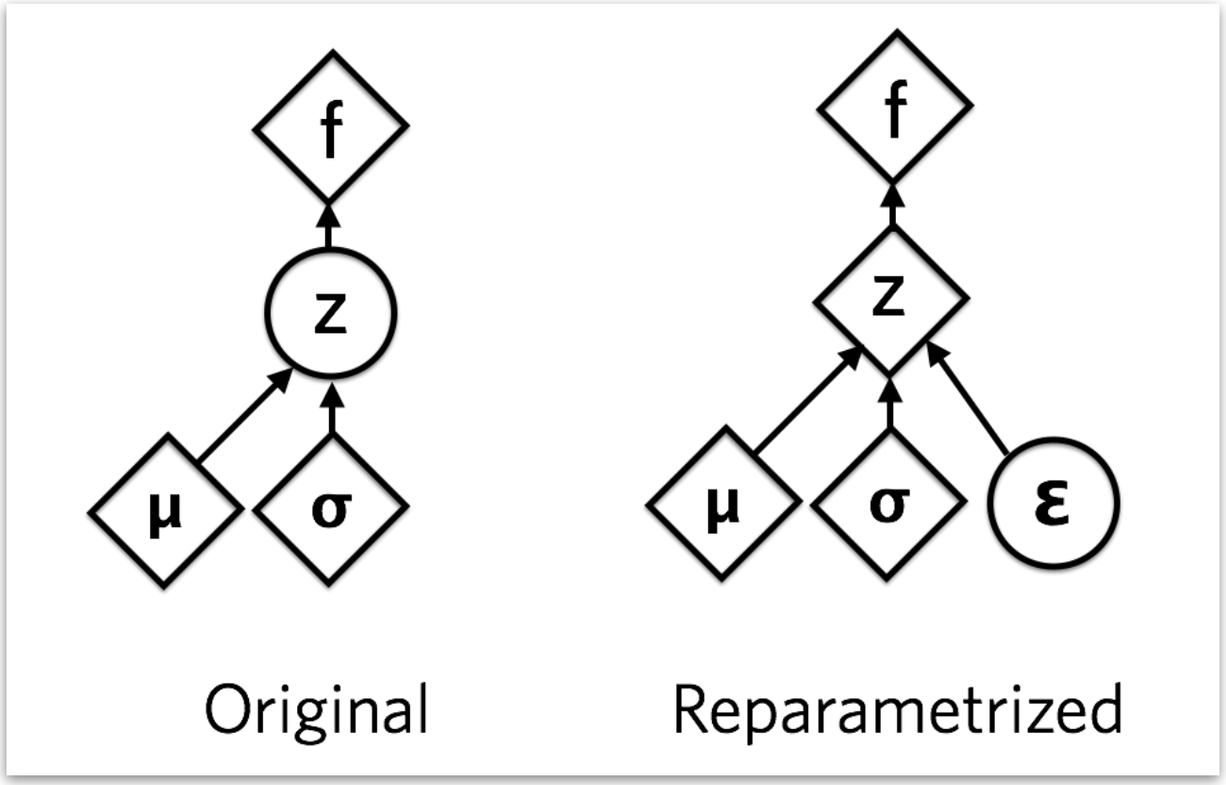
$$\frac{\partial}{\partial\{\phi,\theta\}}\mathscr{L} \quad \text{— how?}$$

$$\nabla_\phi \mathbb{E}_{q_\phi(\mathbf{z})}[f(\mathbf{z})] = \mathbb{E}_{q_\phi(\mathbf{z})}\left[f(\mathbf{z})\nabla_{q_\phi(\mathbf{z})}\log q_\phi(\mathbf{z})\right] \simeq \frac{1}{L}\sum_{l=1}^{L} f(\mathbf{z})\nabla_{q_\phi\left(\mathbf{z}^{(l)}\right)}\log q_\phi\left(\mathbf{z}^{(l)}\right)$$

$$\mathbf{z}^{(l)} \sim q_\phi\left(\mathbf{z}|\mathbf{x}^{(i)}\right)$$

Remainder: VAE

Reparametrization trick



$$p(z) = \mathcal{N}(z \mid 0,\ I)$$

$$p_\theta(x \mid z) = f(x; z; \theta)$$

$$q_\phi(z \mid x) = \mathcal{N}\left(\mu_\phi(x), \mathrm{diag}(\sigma^2_\phi(x))\right)$$

$$z = \mu_\phi(x) + \sigma_\phi(x) \odot \epsilon,$$

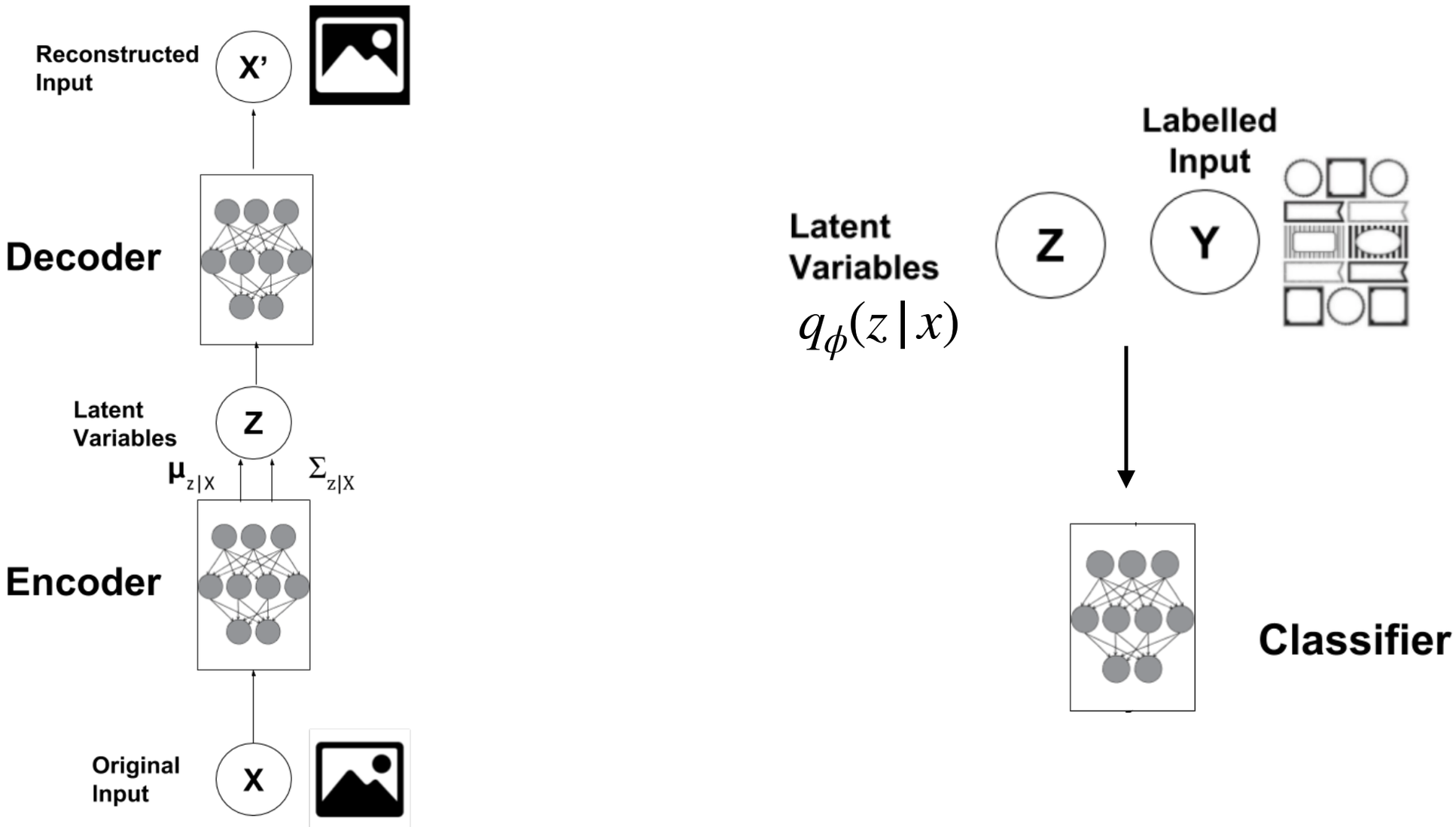
$$\epsilon \sim \mathcal{N}(0,1)$$

$$\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}^{(i)})}\left[f(\mathbf{z})\right] = \mathbb{E}_{p(\boldsymbol{\epsilon})}\left[f(g_\phi(\boldsymbol{\epsilon}, \mathbf{x}^{(i)}))\right] \simeq \frac{1}{L}\sum_{l=1}^{L} f(g_\phi(\boldsymbol{\epsilon}^{(l)}, \mathbf{x}^{(i)}))$$

Can compute gradients!

https://jaan.io/what-is-variational-autoencoder-vae-tutorial/

Latent-feature discriminative model (m1)



http://bjlkeng.github.io/posts/semi-supervised-learning-with-variational-autoencoders/

Generative semi-supervised model (m2)

$$p(y) = \mathrm{Cat}(y \mid \pi)$$

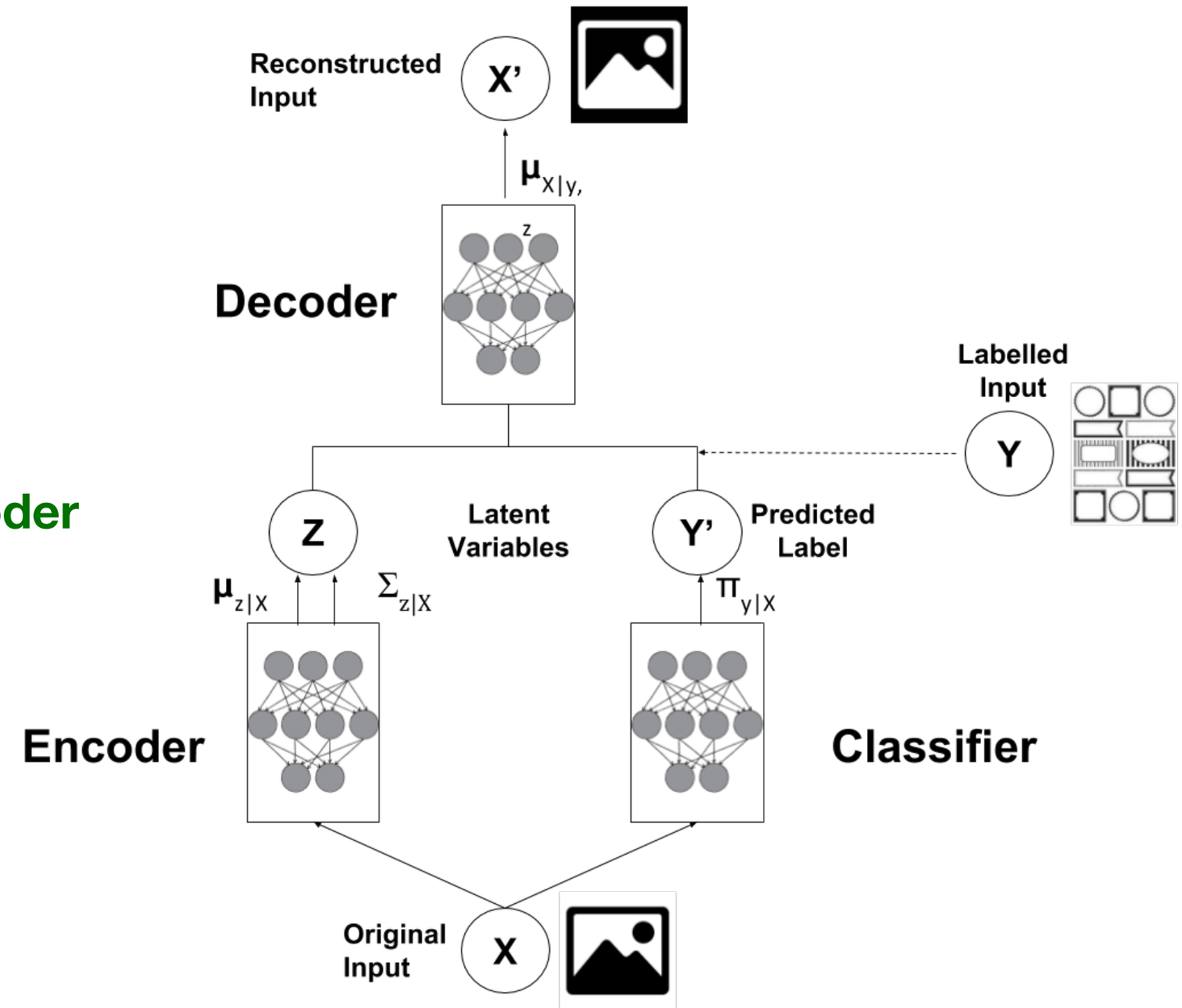
$$p(z) = \mathcal{N}(z \mid 0, I)$$

$$p_\theta(x \mid y, z) = f(x; y; z; \theta)$$ **Decoder**

$$q_\phi(z, y \mid x) = q_\phi(z \mid x) q_\phi(y \mid x)$$

$$q_\phi(z \mid x) = \mathcal{N}(\mu_\phi(x), \mathrm{diag}(\sigma^2_\phi(x)))$$ **Encoder**

$$q_\phi(y \mid x) = \mathrm{Cat}(y \mid \pi_\phi(x))$$ **Classifier**



http://bjlkeng.github.io/posts/semi-supervised-learning-with-variational-autoencoders/

Generative semi-supervised model (m2)

Unlabeled data:

$$\log p(x) \geqslant \mathbb{E}_{q_\phi(y,z|x)}[\log p_\theta(x\,|\,y,z) + \log p_\theta(y) + \log p(z) - \log q_\phi(y,z\,|\,x)]$$
$$= \sum_y q_\phi(y\,|\,x)\mathscr{L}(x,y) + \mathscr{H}(q_\phi(y\,|\,x)) = \mathscr{U}(x)$$

where $\mathscr{L}(x,y) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x\,|\,y,z) + \log p_\theta(y)] - KL(q_\phi(z\,|\,x)\|p_\theta(z))$

Labeled data:

$$\log p(x,y) \geqslant \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x\,|\,y,z) + \log p_\theta(y) + \log p_\theta(z) - \log q_\phi(z\,|\,x)] = \mathscr{L}(x,y)$$

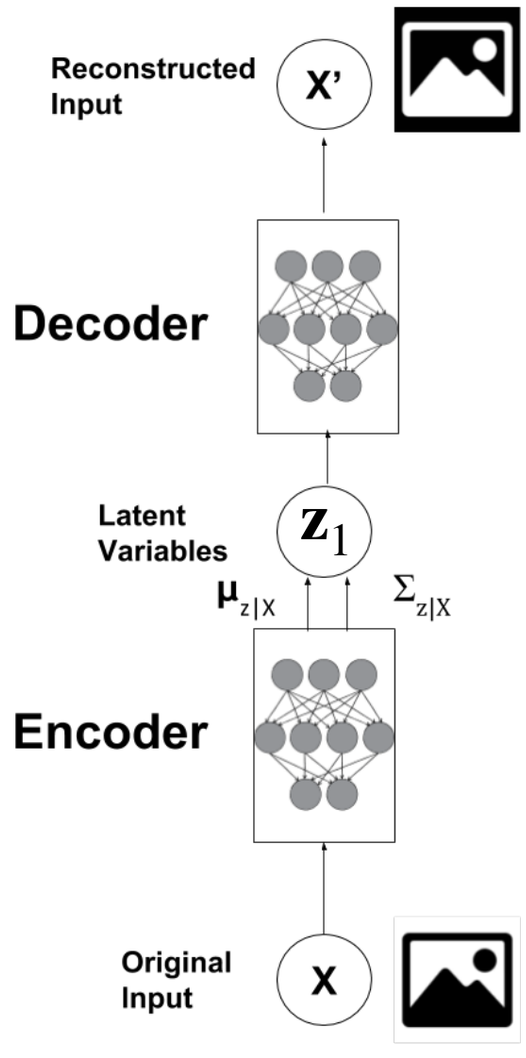
Loss:

$$\mathscr{J} = -\sum_{x,y\,\in\,\text{labeled}} \mathscr{L}(x,y) - \sum_{x\,\in\,\text{unlabeled}} \mathscr{U}(x)$$

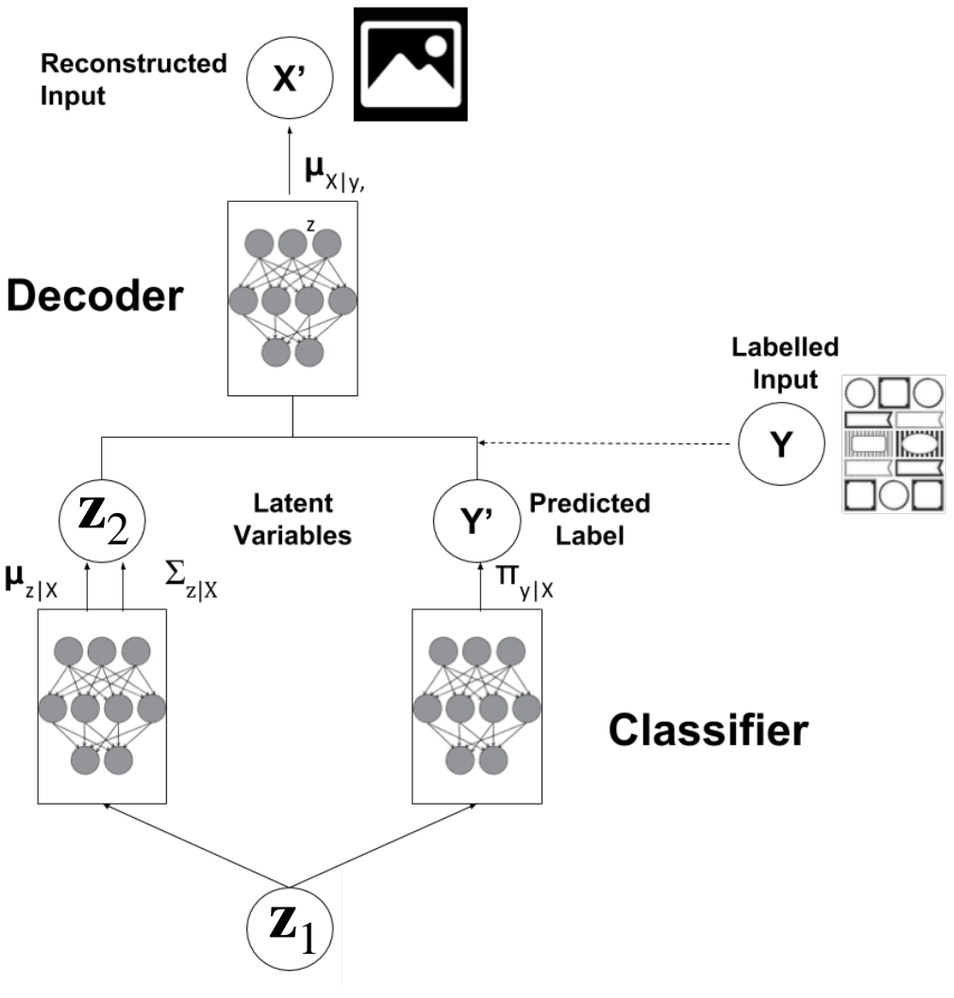
$$\mathscr{J}^\alpha = \mathscr{J} + \alpha \cdot \mathbb{E}_{p_l(x,y)}\left[-\log q_\phi(y\,|\,x)\right]$$

Stacked m1+m2

$$p_\theta(x, y, z_1, z_2) = p(y)p(z_2)p_\theta(z_1 \mid y, z_2)p_\theta(x \mid z_1)$$



+



Error rate on MNIST

- 50,000 training examples
- N labeled examples

Table 1: Benchmark results of semi-supervised classification on MNIST with few labels.

N	NN	CNN	TSVM	CAE	MTC	AtlasRBF	M1+TSVM	M2	M1+M2
100	25.81	22.98	16.81	13.47	12.03	8.10 ($\pm$ 0.95)	11.82 ($\pm$ 0.25)	11.97 ($\pm$ 1.71)	**3.33** ($\pm$ 0.14)
600	11.44	7.68	6.16	6.3	5.13	–	5.72 ($\pm$ 0.049)	4.94 ($\pm$ 0.13)	**2.59** ($\pm$ 0.05)
1000	10.7	6.45	5.38	4.77	3.64	3.68 ($\pm$ 0.12)	4.24 ($\pm$ 0.07)	3.60 ($\pm$ 0.56)	**2.40** ($\pm$ 0.02)
3000	6.04	3.35	3.45	3.22	2.57	–	3.49 ($\pm$ 0.04)	3.92 ($\pm$ 0.63)	**2.18** ($\pm$ 0.04)

Error rate on SVHN

- 70,000 training examples
- 10 classes

Table 2: Semi-supervised classification on the SVHN dataset with 1000 labels.

KNN	TSVM	M1+KNN	M1+TSVM	M1+M2
77.93 (± 0.08)	66.55 (± 0.10)	65.63 (± 0.15)	54.33 (± 0.11)	**36.02** (± 0.10)

Error rate on NORB

- 194,400 training examples
- 5 classes

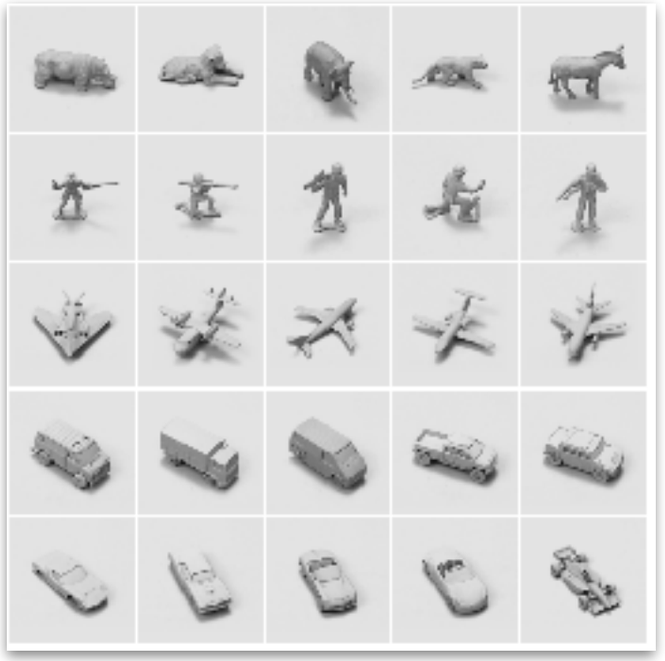


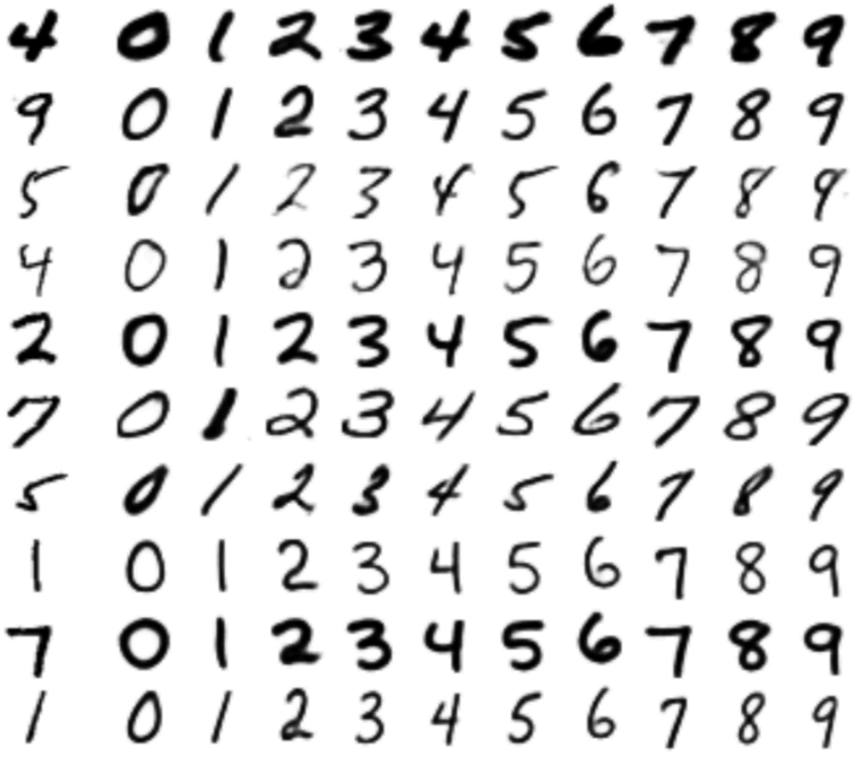
Table 3: Semi-supervised classification on the NORB dataset with 1000 labels.

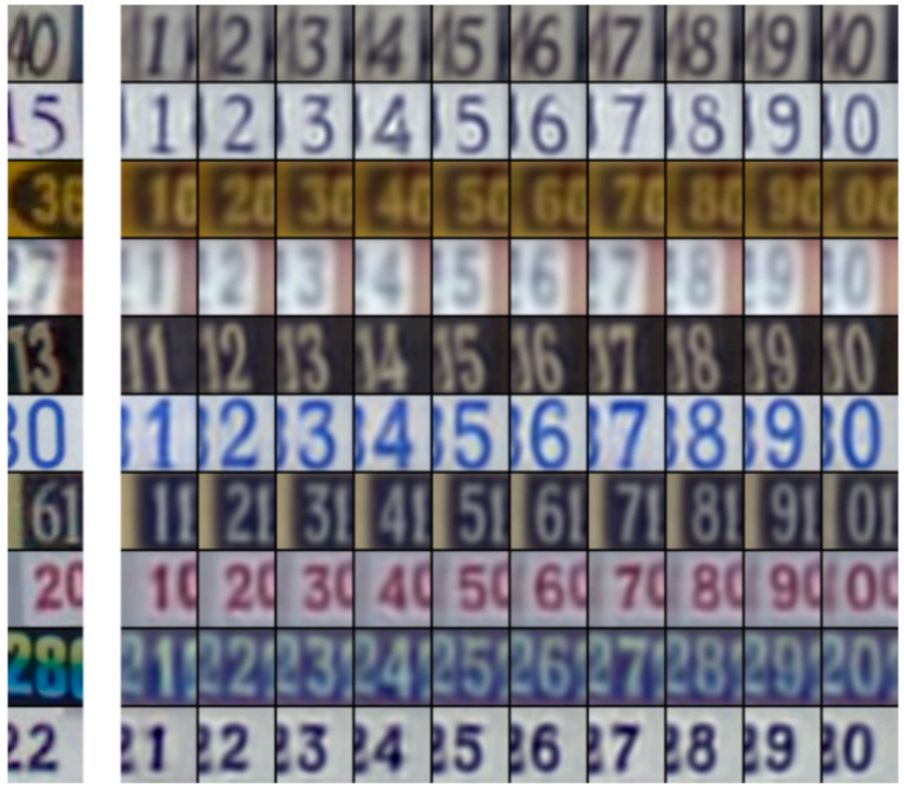
KNN	TSVM	M1+KNN	M1+TSVM
78.71 (± 0.02)	26.00 (± 0.06)	65.39 (± 0.09)	**18.79** (± 0.05)

Conditional generation

Different styles generated by varying latent variables

Conditional generation

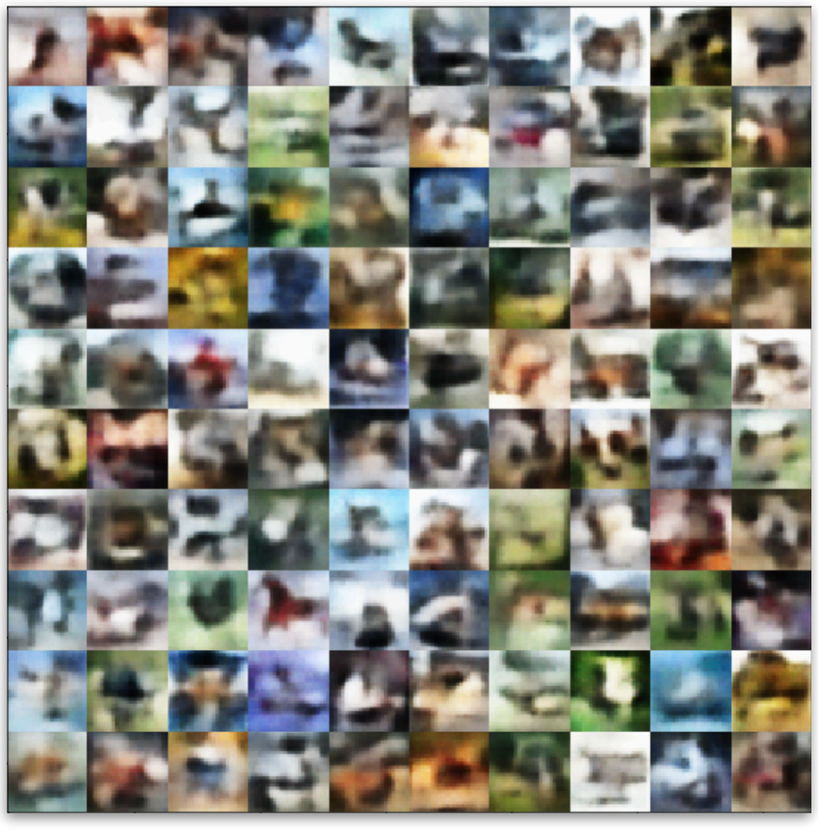




Analogies generated by varying classes

Conclusion

- Almost as efficient as other Neural Network approaches
- Linear scale with # of classes
- Not SOTA today (2019)



https://github.com/bjlkeng/sandbox/blob/master/notebooks/vae-semi_supervised_learning/vae-m2-fit.ipynb