Variational Autoencoders

ENEE 6583 Neural Nets

Dr. Alsamman

Slide Credits:

Supervised vs Unsupervised Learning

- Supervised (SL) : $\{X, Y\}$: $M\{X\} \rightarrow Y$
  - Given inputs X, corresponding labels (outputs) Y
  - Learn mapping M that maps X to Y
- Unsupervised (USL): $M\{X\} \rightarrow Y$
  - Given only inputs X
  - Find a mapping to Y that optimizes some objective function

Why Unsupervised : No label possible

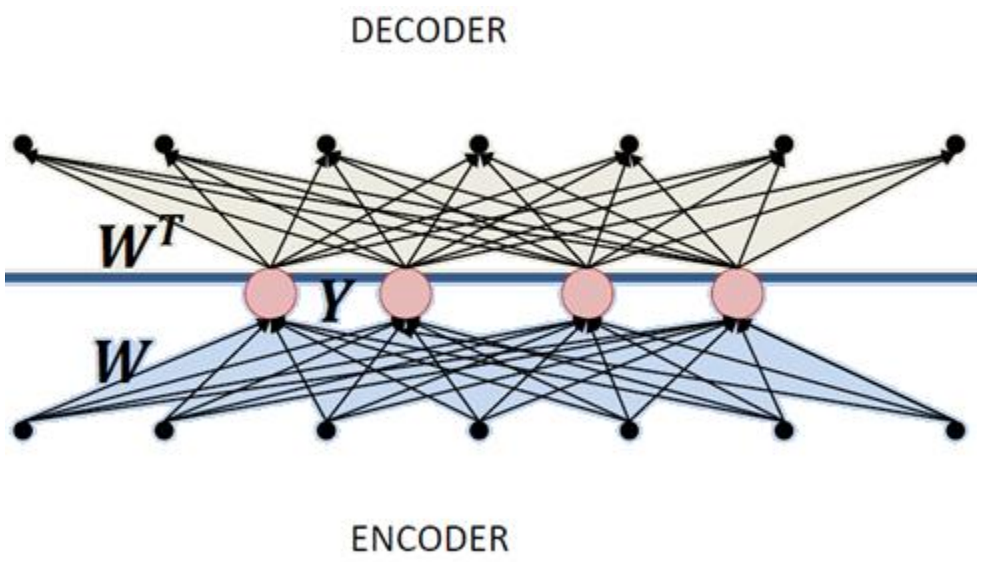
- Hidden data representation
  - Data compression
  - Data organization
  - Explore hidden structures within data
- Applications:
  - Organize computer clusters
  - Group users according to interest
  - Marketing: Recommend products/services
  - Detect fault/intrusion
  - Find similarity
- Driven by an objective function

Why Unsupervised : Price

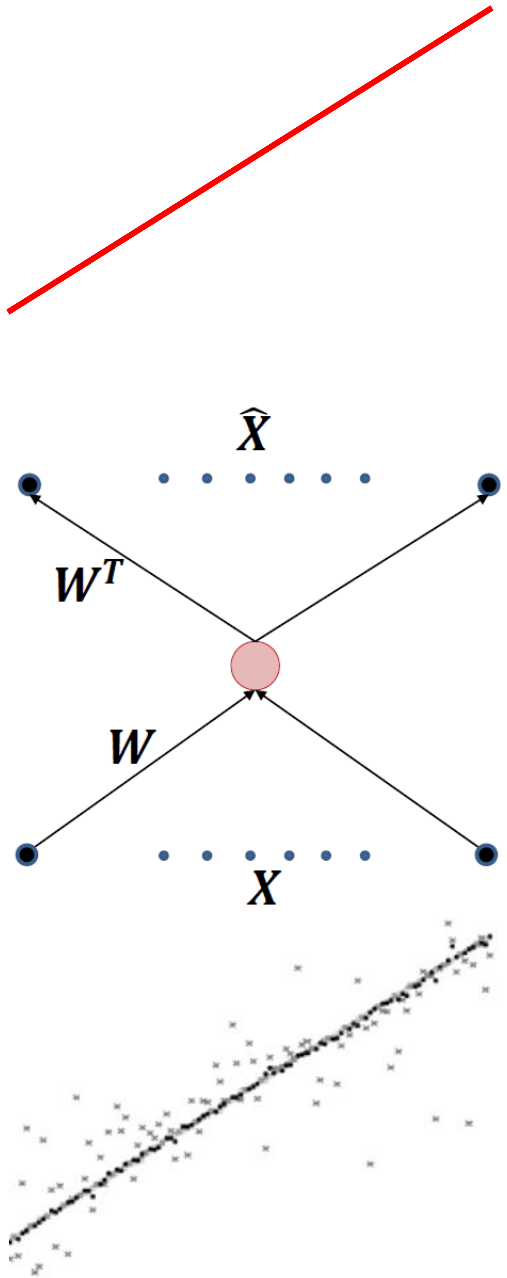
- Data is
  - Decreasing in price
  - Increasing in: volume, speed,
  - Varying in modality
- Advanced tech =>
  - cheaper tech => cheaper data (price)
  - better sensors => more data (volume, speed)
  - more tech services => user data (modes, speed)
- Expert labeling is expensive
  - Mechanical Turk
    Is 'data labeling' the new blue-collar job of the AI era?
    www.techrepublic.com/article/is-data-labeling-the-new-blue-collar-job-of-the-ai-era/

Autoencoder

- Encode the input: $M\{X\} \rightarrow Y$
  - Analysis
- Decoder is the reverse: $M^{-1}\{Y\} \rightarrow \hat{X}$
  - Synthesis
  - Identical to encoder network
- Unsupervised learning
- Objective function: $X \approx \hat{X}$
  - $E = \left|X - \hat{X}\right|^2 = 0$
  - Find W for $E \approx 0$



Linear AE



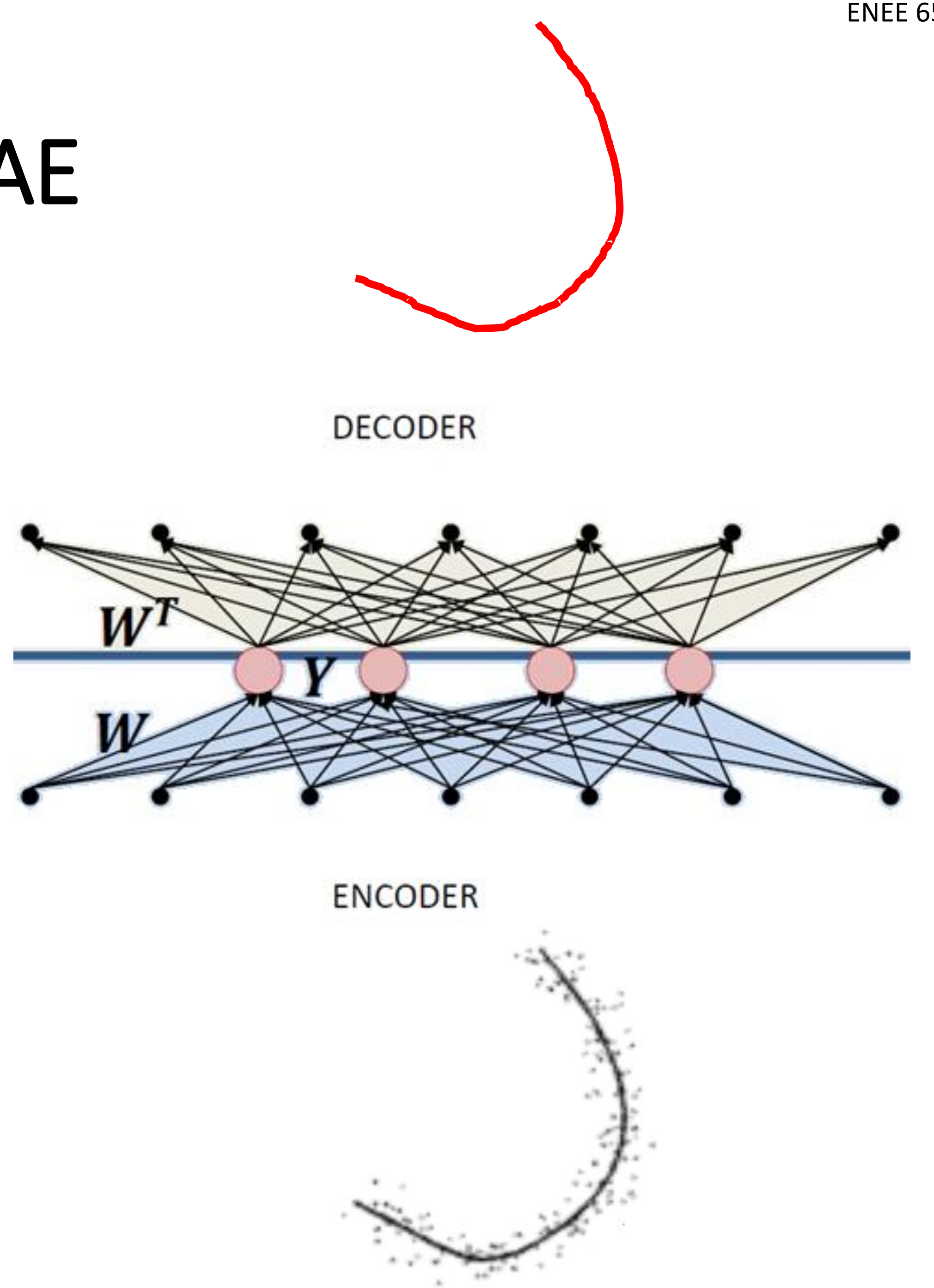
- Linear encoding: Linear activations
- Equations:

$$
\begin{aligned}
Y &= WX \\
\hat{X} &= W^T Y \\
E &= |X - W^T W X|^2
\end{aligned}
$$

  - W is a principal component (PCA)
  - Line along that max energy
  - Matrix theory: max Eigen vector

Nonlinear AE

- Nonlinear activations
- Nonlinear CA
- Learn the nonlinear manifold



Applications

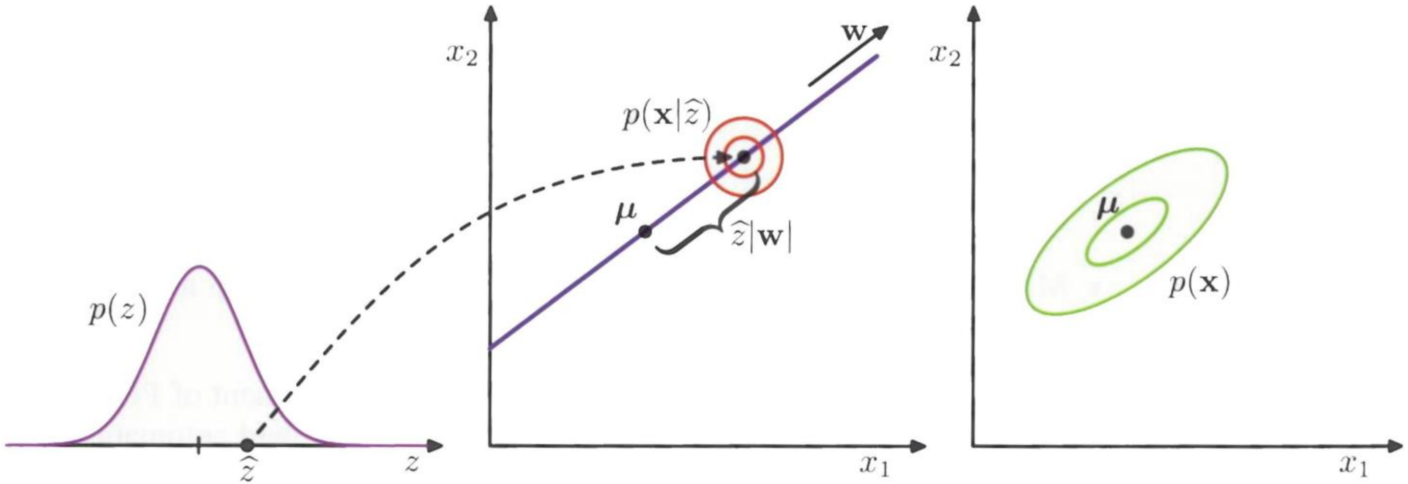
- Denoising data
- Data compression/encryption
- Classification
  - Reduced dimension leads to a unique manifold
- Mix source separation
  - Multiple sources with unique manifolds linearly mixed together
  - Encoder: Source separation
  - Decoder: Generate sources

Generative Models

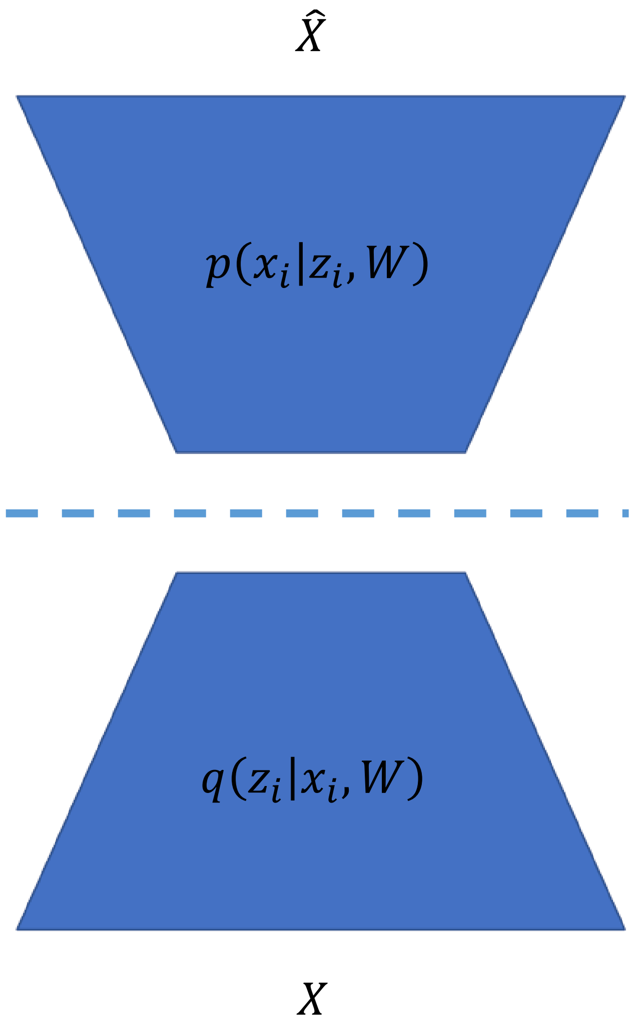
- Discriminative learning: classification
  - Supervised process
  - Find difference
  - E.g.: MNIST classify digit as 0,1,…,9
- Generative learning: creation
  - Unsupervised learning
  - Find similarity
  - E.g.: machine translation

Latent Space Generative Models

- Data are generated from a real-valued latent space
- Latent space: unknown model space of given data
  - Model is probabilistic (PDF/PMF, μ, Σ)
- Data: samples from that space are used to create
- Factor Analysis:
  - Assume a model PDF/PMF
  - Objective: based on given data observation, $\mathbf{x}$, determine statistics (μ, Σ) and $p(\mathbf{x})$



Variational AE



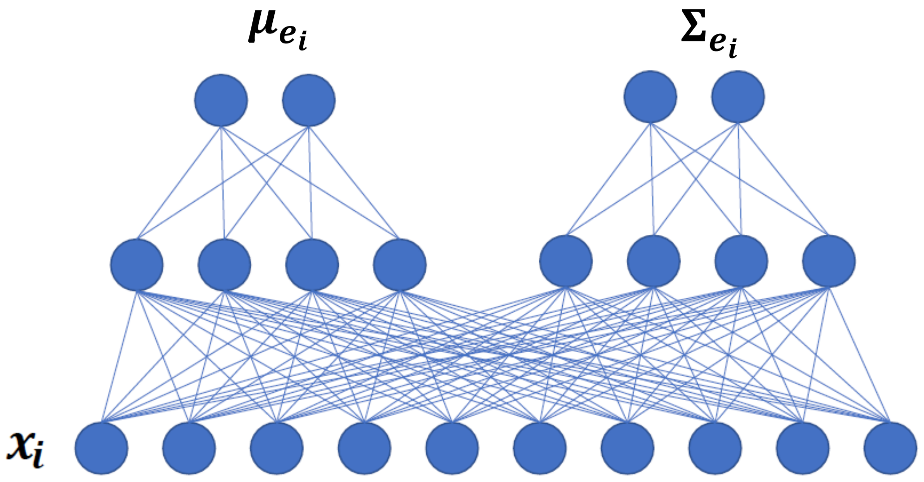
- Decoder: $p(x_i|z_i, W)$
  - Nnet, Generative model
  - Estimates the probability distribution of input X given the laten variable Z

- Normally distributed Latent Space
- characterized by μ, σ

- Encoder: $q(z_i|x_i, W)$
  - Nnet, Inference model
  - Estimates the probability distribution of the latent space given the data X

Encoder

- $q(z_i|x_i,\phi) = \mathcal{N}\left(z_i|\mu_{e_i},\Sigma_{e_i}\right)$
- Encoder output is :$\mu_{e_i} = u_e(x_i,W_1),\quad \Sigma_{e_i} = \text{diag}\big(s_e(x_i,W_2)\big)$
- Two networks: u_e, s_e
- W_1: weights of network u_e
- W_2: weights of network s_e
- ϕ: combination of weights W_1, W_2



Decoder

- $p(x_i|z_i, \theta) = \mathcal{N}\left(x_i|\mu_{e_i}, \Sigma_{e_i}\right)$
- Sample Z space: generate z_i based on μ_{e_i} and Σ_{e_i}
- Decoder output: $\widehat{x_i}$

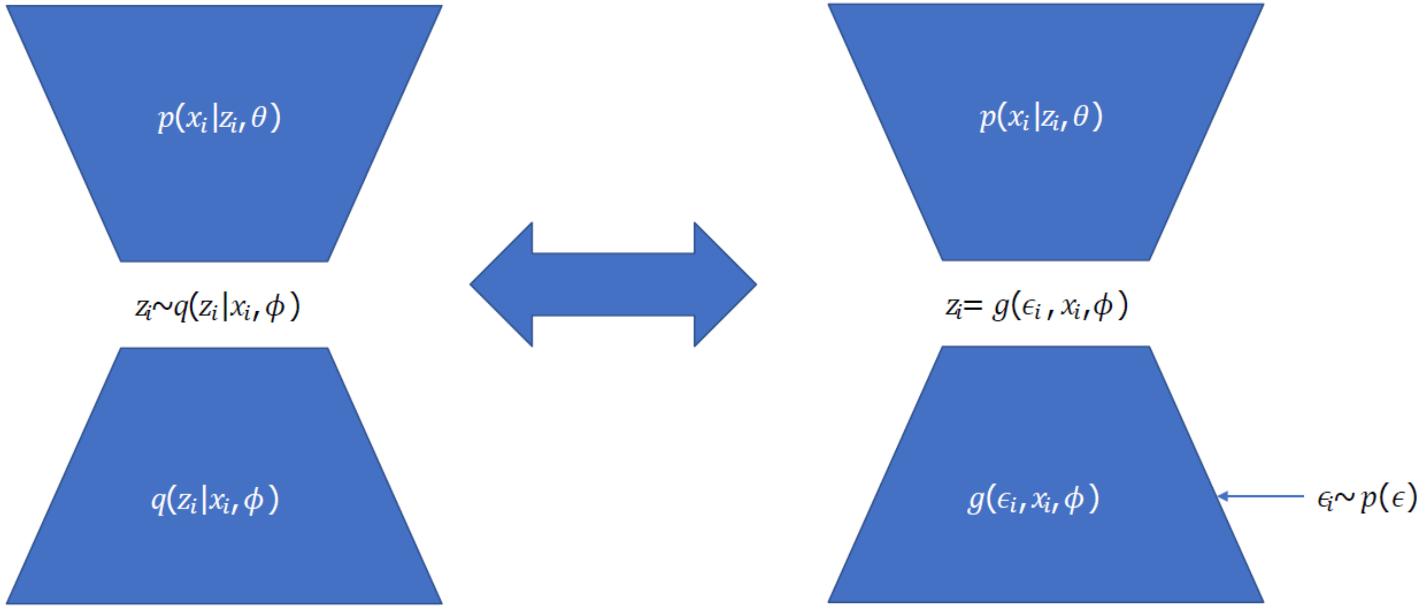
KL Divergence

- Kullback-Leibler divergence
  - Measures the information lost when a probability distribution, q, is used to approximate another probability distribution p.

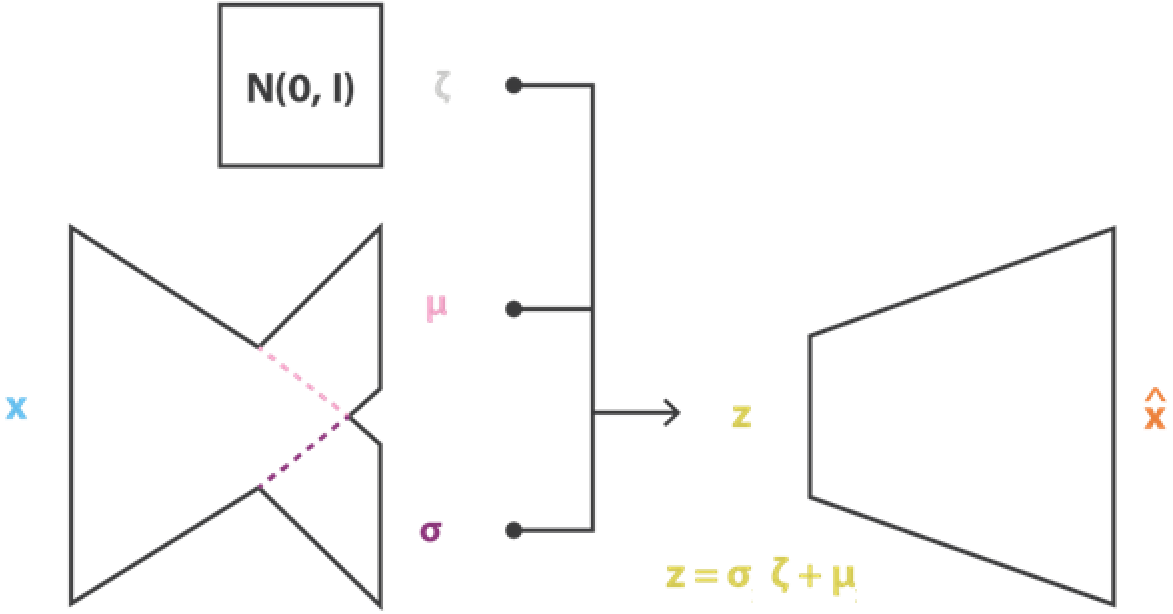
$$D_{KL}(p(x)||q(x)) = \sum_{x} p(x) \ln \frac{p(x)}{q(x)}$$

Reparametrization

- Let $z_i = g(\epsilon_i, x_i, \phi)$
- ϵ_i drawn from Gaussian $p(\epsilon)$
- Z deterministic depends on ϕ
- Now we can backpropagate!
- $z = \mu + \sigma \odot \epsilon$
- $\epsilon \sim \mathcal{N}(0,1)$

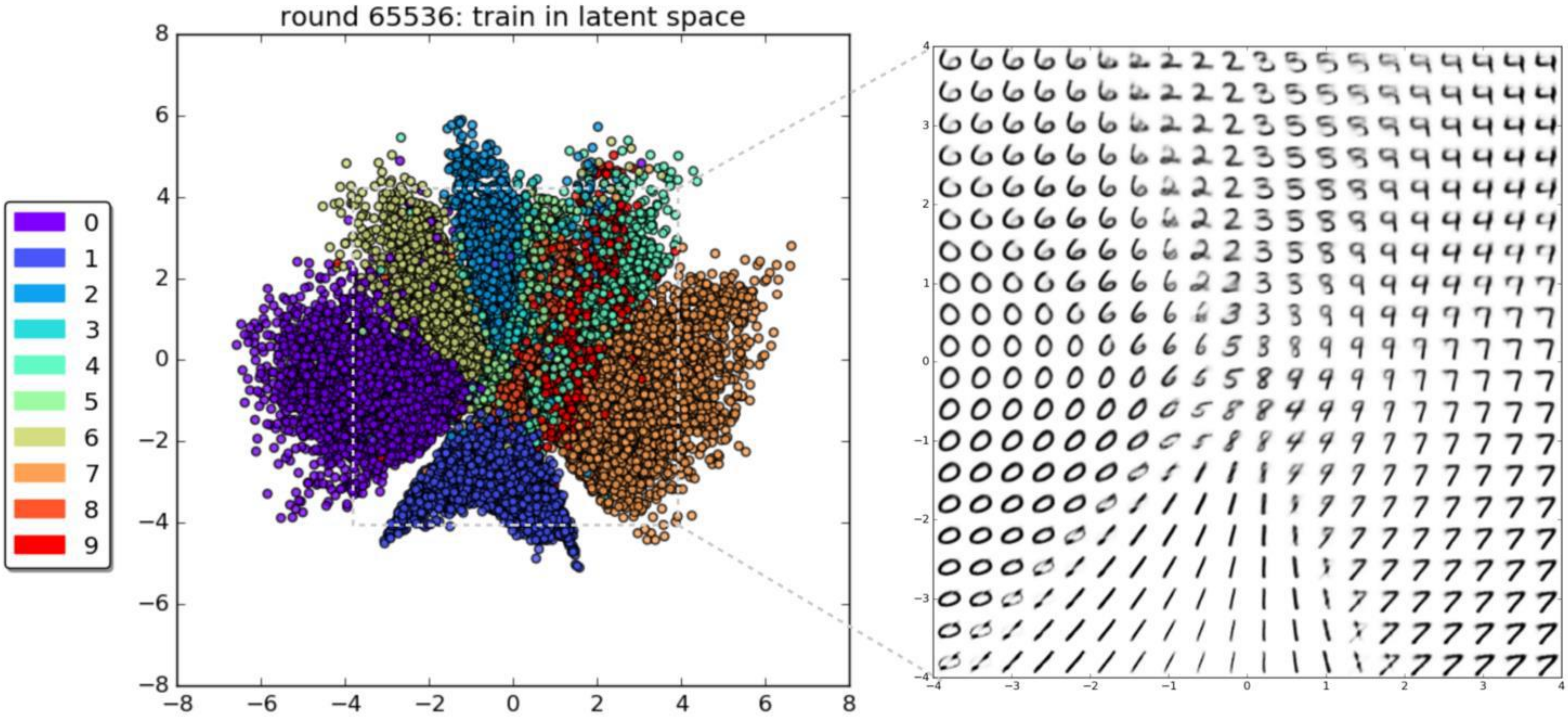


Re-parametrization



$$\text{loss} = C\,||\,x - \hat{x}\,||^2 + KL[\,N(\mu_x, \sigma_x), N(0, I)\,]$$

Edited from https://towardsdatascience.com/understanding-variational-autoencoders-vaes-f70510919f73


round 65536: train in latent space
0
1
2
3
4
5
6
7
8
9
8 6 4 2 0 −2 −4 −6 −8
−8 −6 −4 −2 0 2 4 6 8