Autoencoder

Industrial AI Lab.

Prof. Seungchul Lee

Unsupervised Learning

- Definition
  - Unsupervised learning refers to most attempts to extract information from a distribution that do not require human labor to annotate example
  - Main task is to find the ‘best’ representation of the data

- Dimension Reduction
  - Attempt to compress as much information as possible in a smaller representation
  - Preserve as much information as possible while obeying some constraint aimed at keeping the representation simpler
  - This modeling consists of finding “meaningful degrees of freedom” that describe the signal, and are of lesser dimension.

Autoencoders

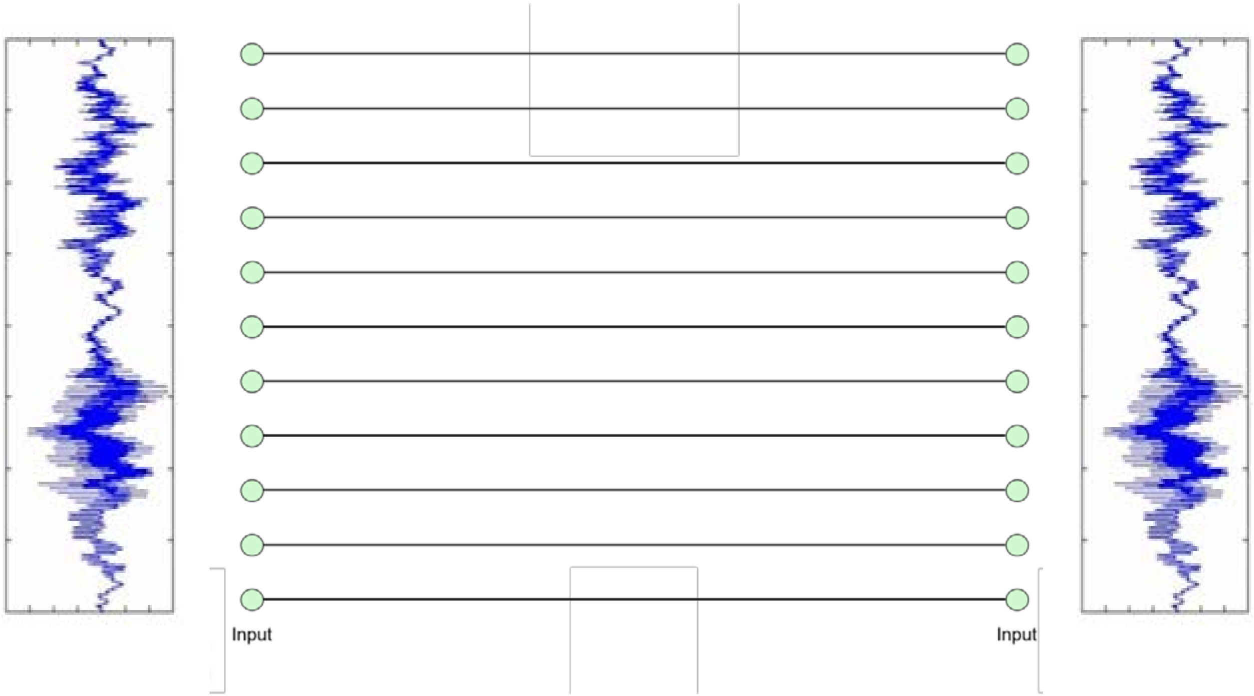
- It is like ‘deep learning version’ of unsupervised learning

- Definition
  - An autoencoder is a neural network that is trained to attempt to copy its input to its output
  - The network consists of two parts: an encoder and a decoder that produce a reconstruction

- Encoder and Decoder
  - Encoder function : $z = f(x)$
  - Decoder function : $x = g(z)$
  - We learn to set $g\big(f(x)\big) = x$

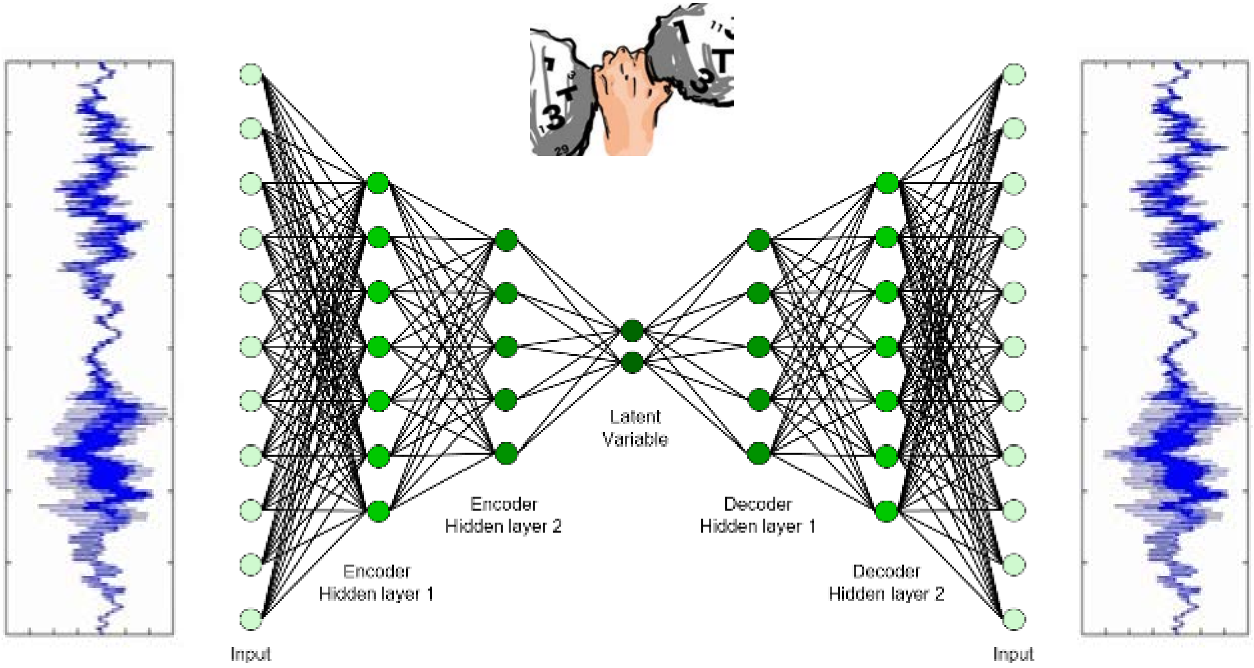
Autoencoder

- Dimension reduction
- Recover the input data



Autoencoder

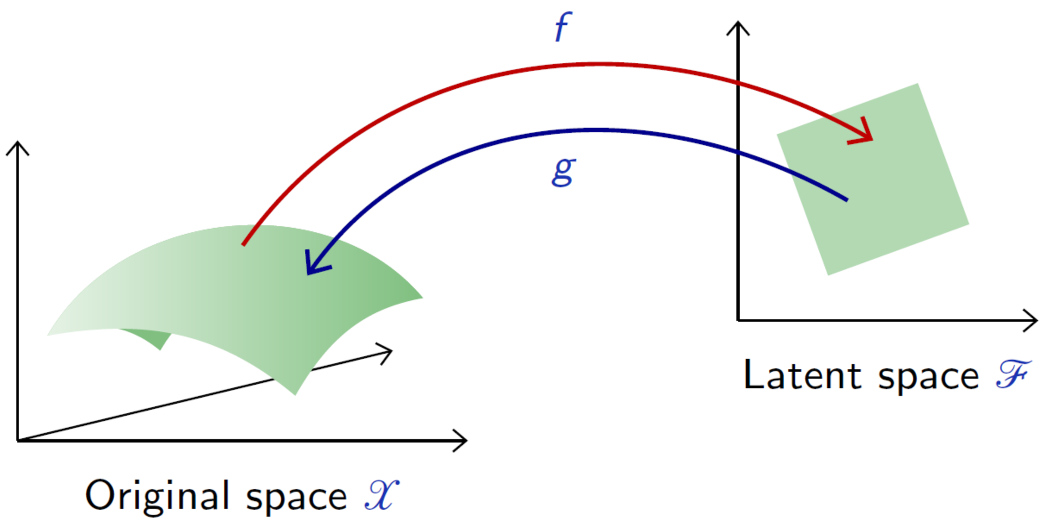
- Dimension reduction
- Recover the input data
  - Learns an encoding of the inputs so as to recover the original input from the encodings as well as possible



Original space Latent space

Autoencoder

- Autoencoder combines an encoder f from the original space $\mathcal{X}$ to a latent space $\mathcal{F}$, and a decoder g to map back to $\mathcal{X}$, such that $g \circ f$ is [close to] the identity on the data



$$\mathbb{E}\left[\|X - g \circ f(X)\|^2\right] \approx 0$$

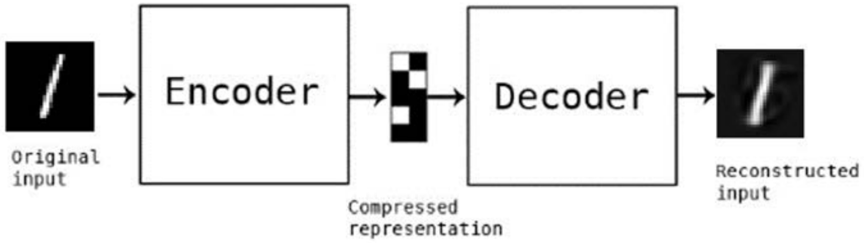
- A proper autoencoder has to capture a "good" parametrization of the signal, and in particular the statistical dependencies between the signal components.

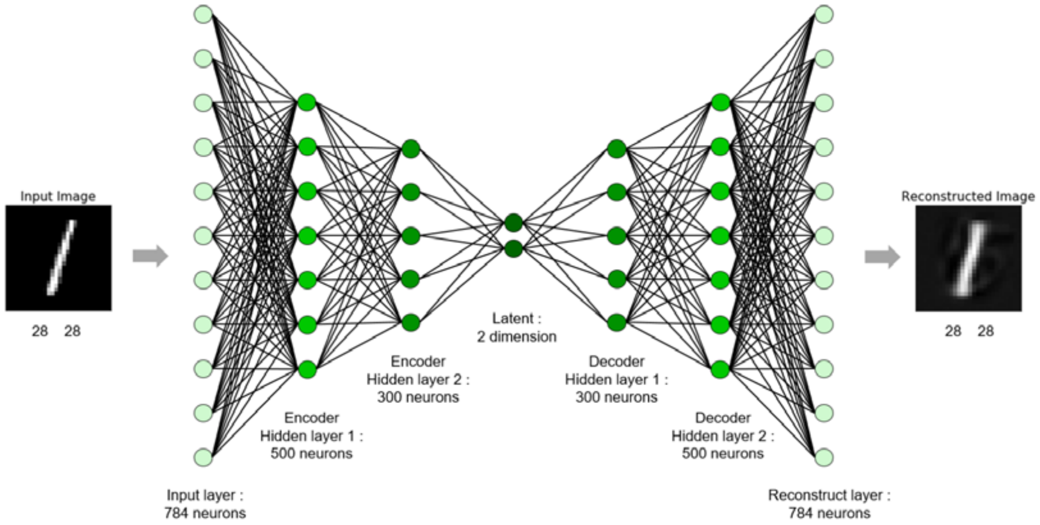
Source: Dr. Francois Fleuret at EPEL

Autoencoder with MNIST

Autoencoder with TensorFlow

- MNIST example
- Use only (1, 5, 6) digits to visualize in 2-D



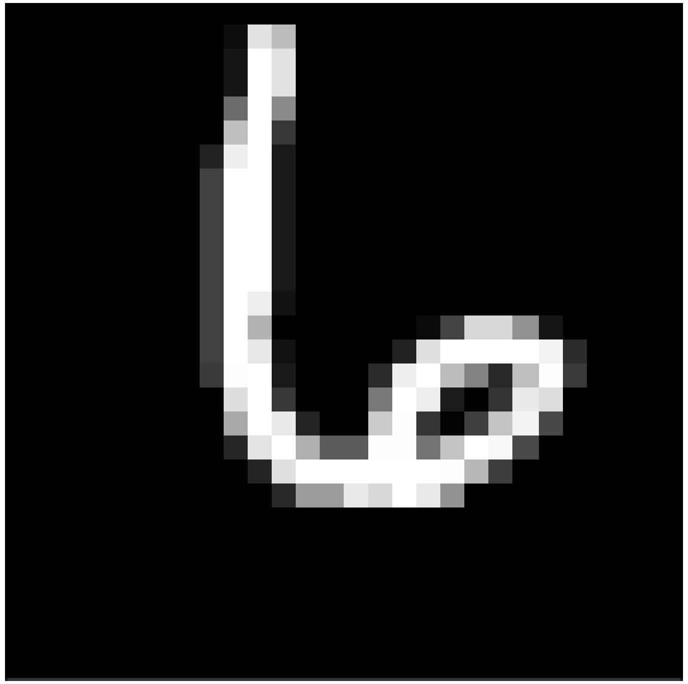


$$\frac{1}{m}\sum_{i=1}^{m}(t_i - y_i)^2$$

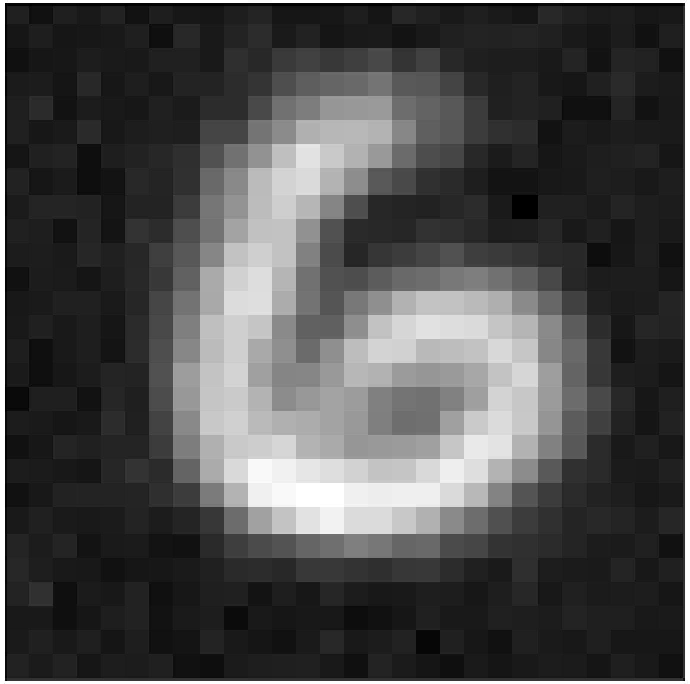
Test or Evaluation

```
test_x, _ = test_batch_maker(1)
x_reconst = sess.run(reconst, feed_dict = {x: test_x})
```

Imput Image



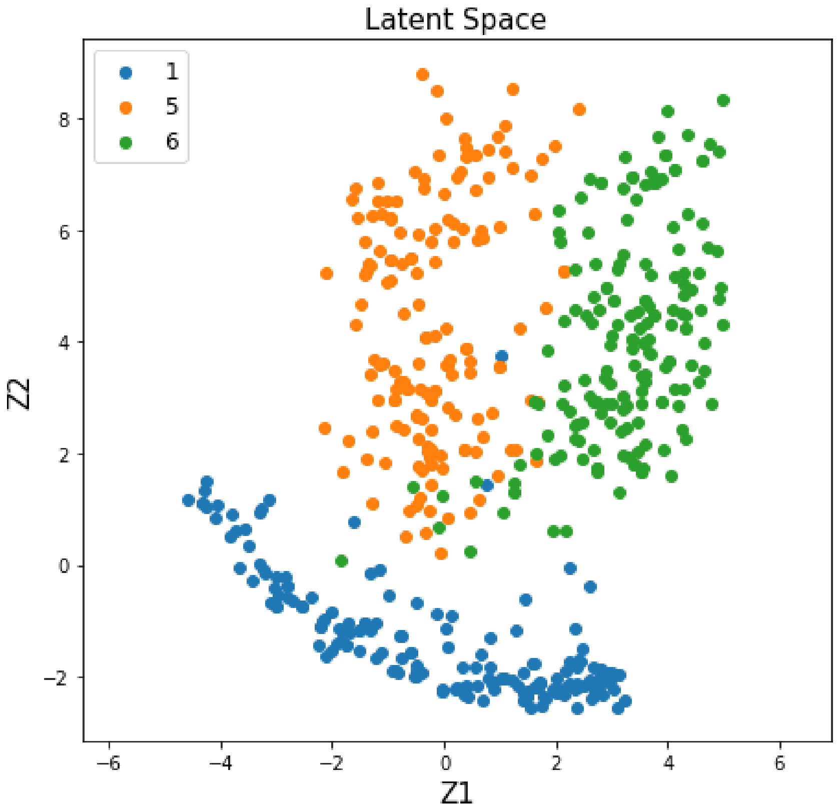
Reconstructed Image



Distribution in Latent Space

- Make a projection of 784-dim image onto 2-dim latent space

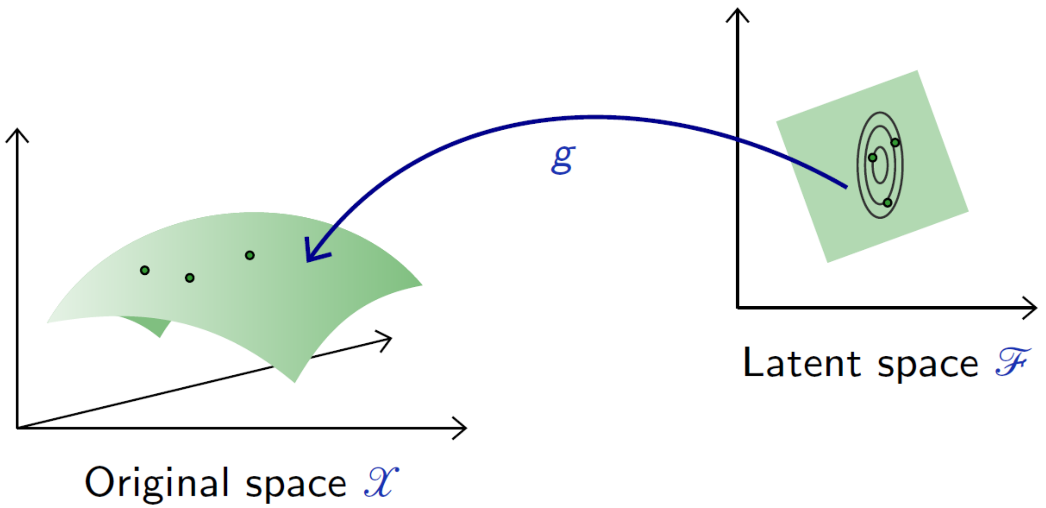
```
test_x, test_y = test_batch_maker(500)
test_y = np.argmax(test_y, axis = 1)
test_latent = sess.run(latent, feed_dict = {x: test_x})
```



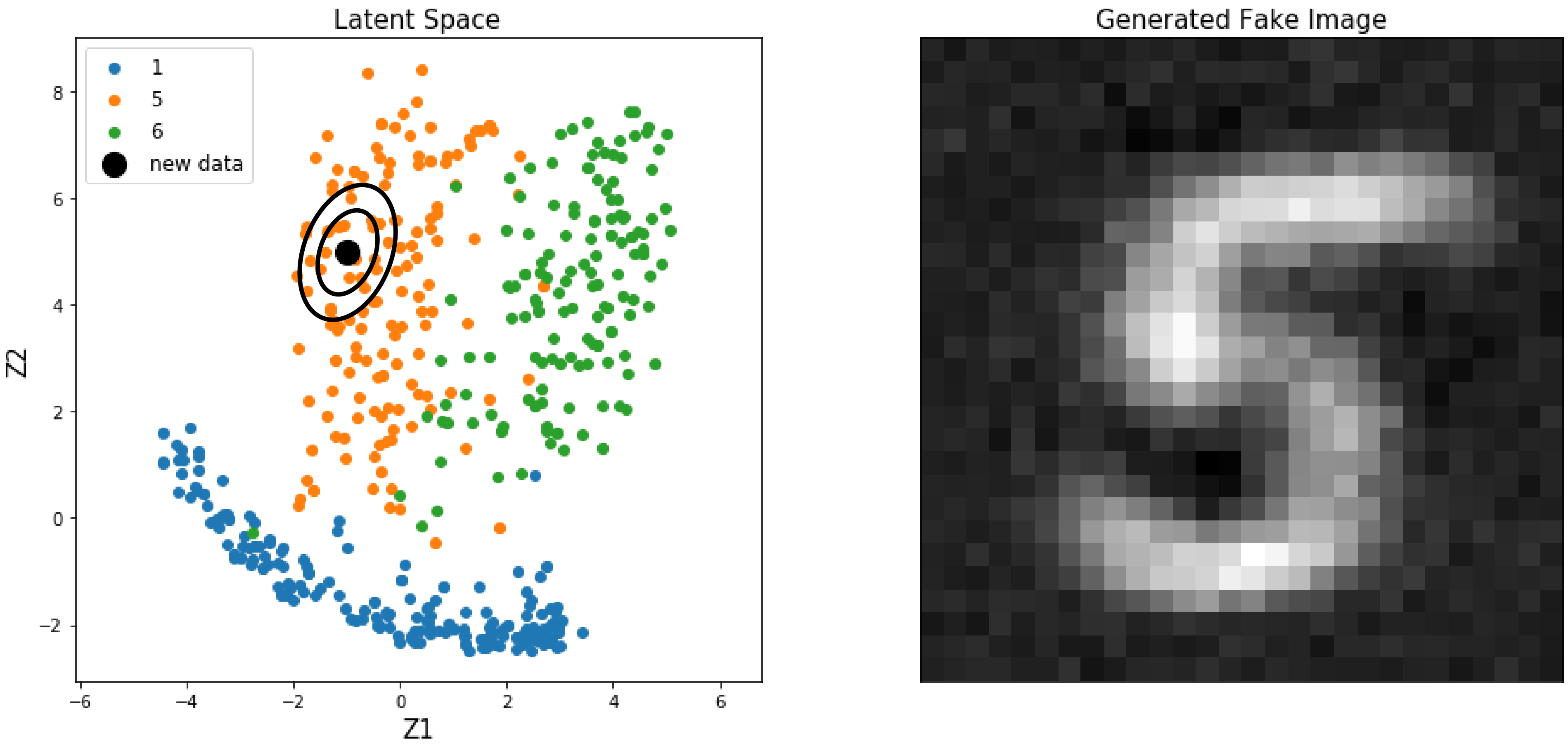
Autoencoder as Generative Model

Generative Capabilities

- We can assess the generative capabilities of the decoder g by introducing a [simple] density model q^Z over the latent space $\mathcal{F}$, sample there, and map the samples into the image space $\mathcal{X}$ with g.

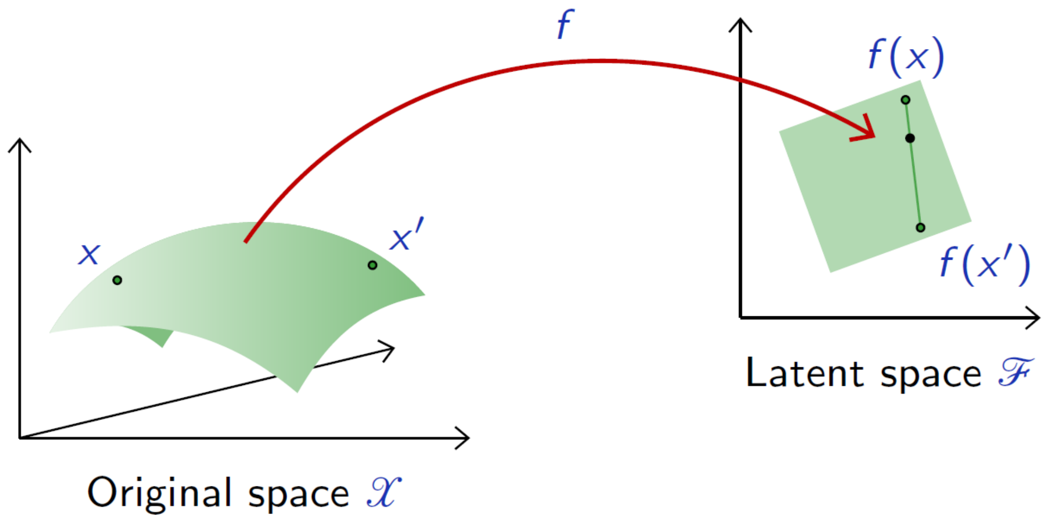


MNIST Example



Latent Representation

- To get an intuition of the latent representation, we can pick two samples x and x' at random and interpolate samples along the line in the latent space

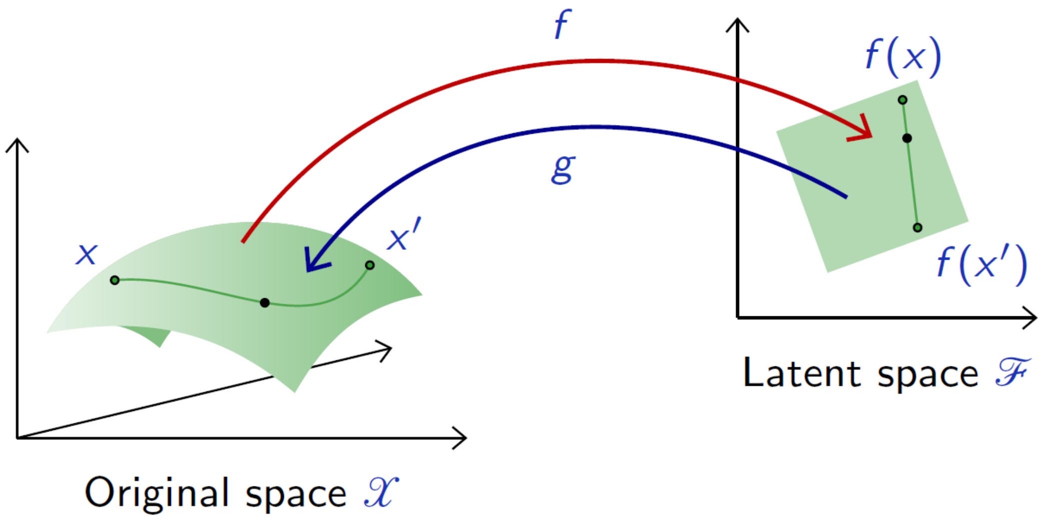


Source: Dr. Francois Fleuret at EPEL

Latent Representation

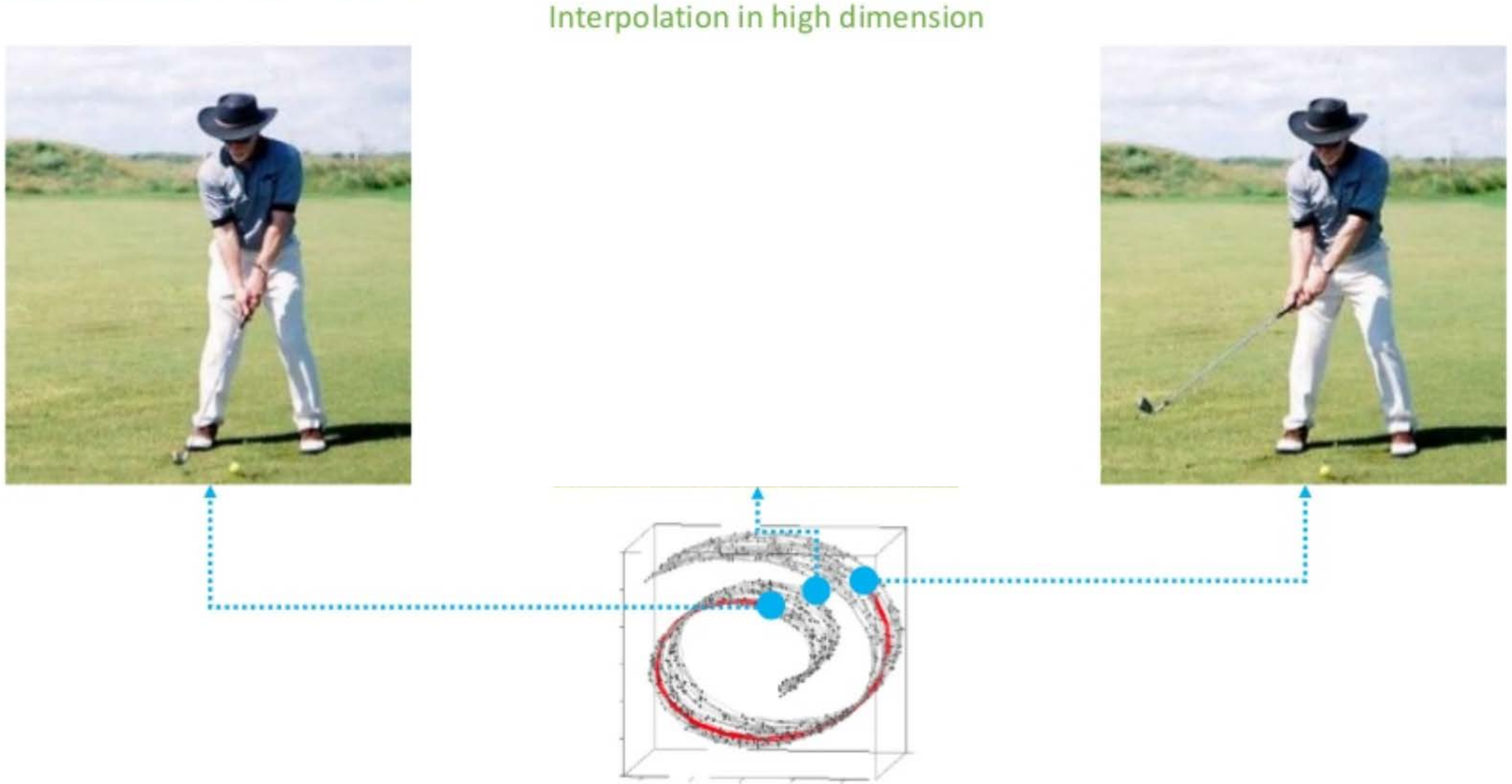
- To get an intuition of the latent representation, we can pick two samples x and x' at random and interpolate samples along the line in the latent space

$$g\left((1-\alpha)f(x)+\alpha f(x')\right)$$



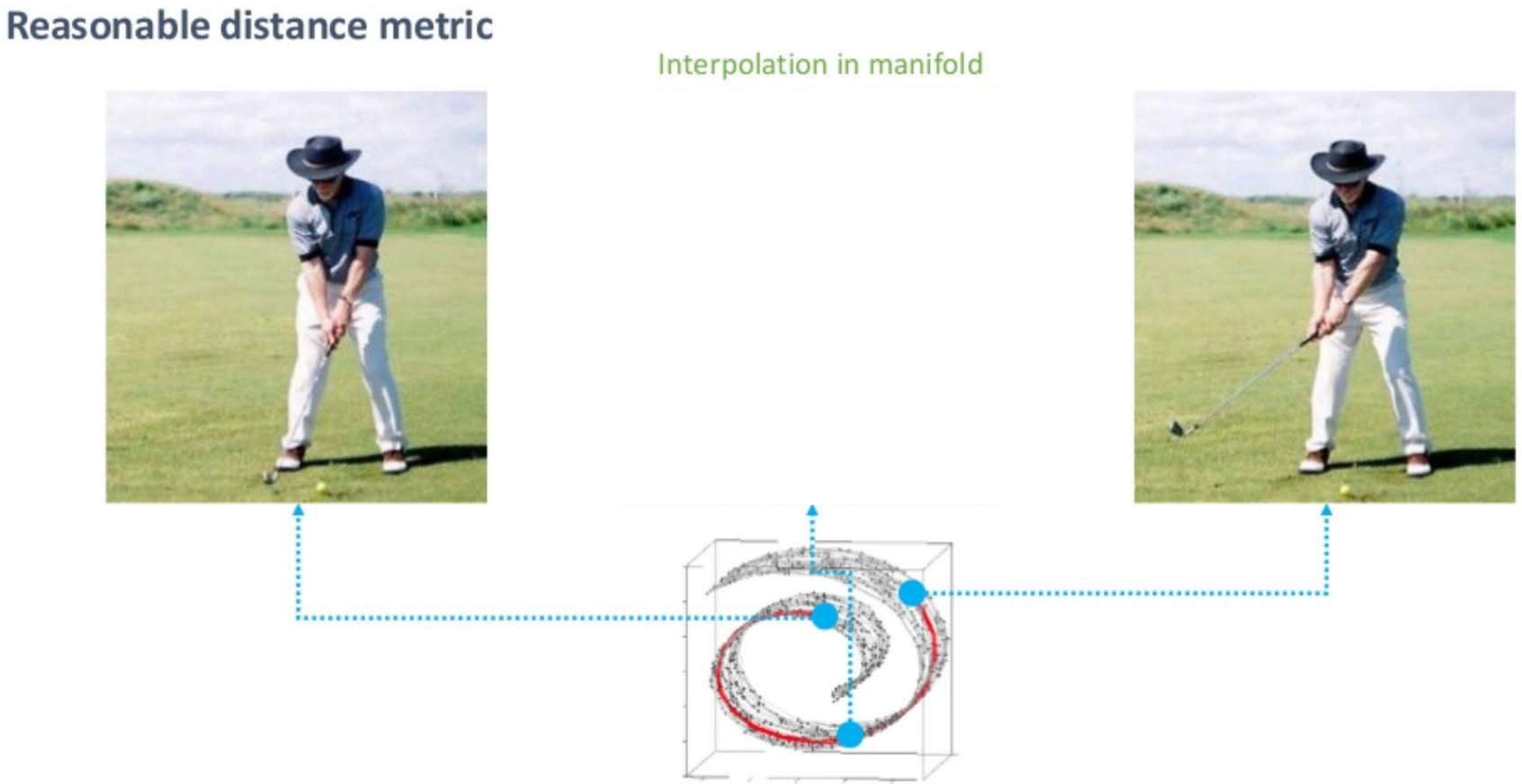
Interpolation in High Dimension

Reasonable distance metric



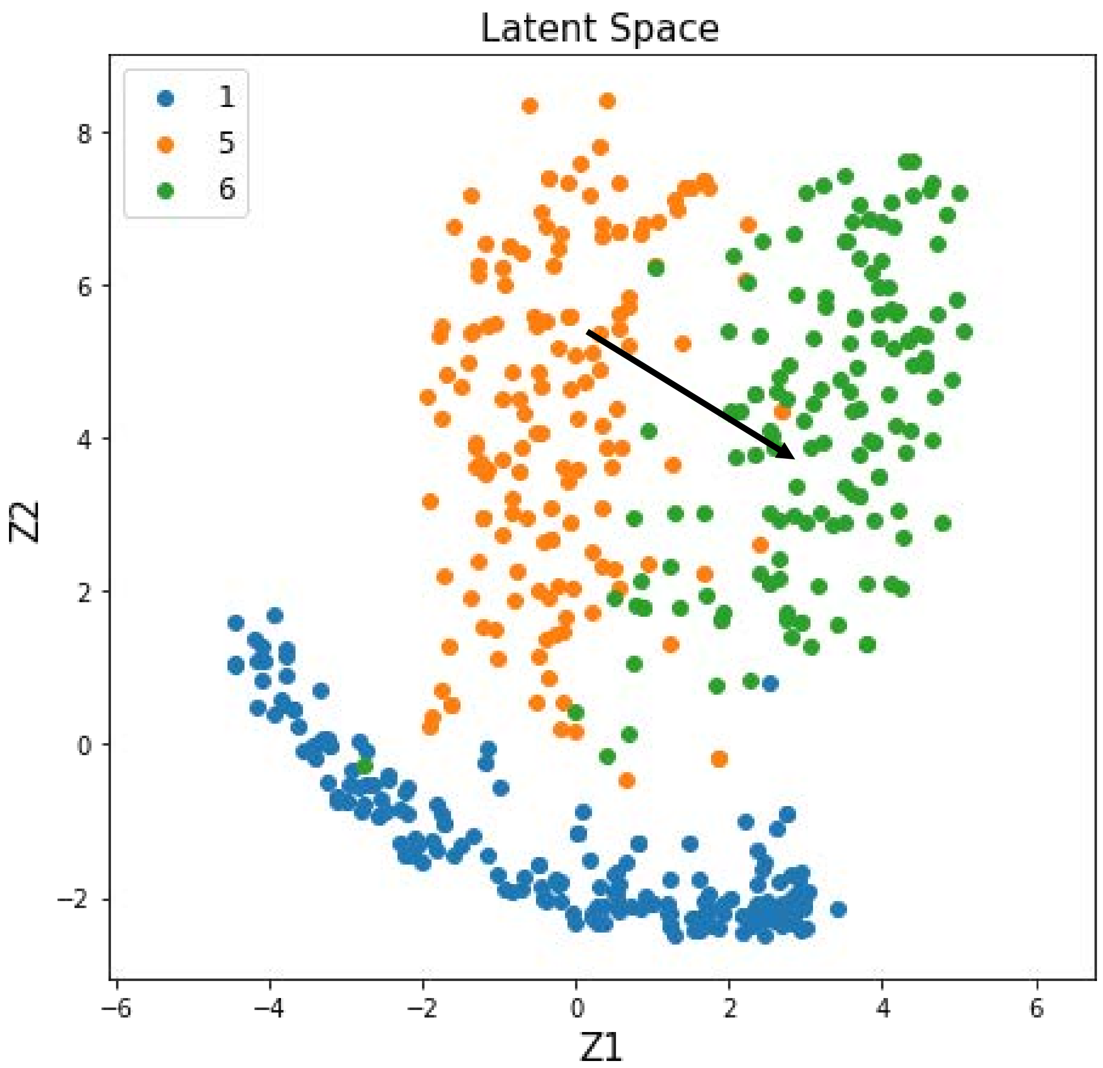
https://www.cs.cmu.edu/~efros/courses/AP06/presentations/ThompsonDimensionalityReduction.pdf

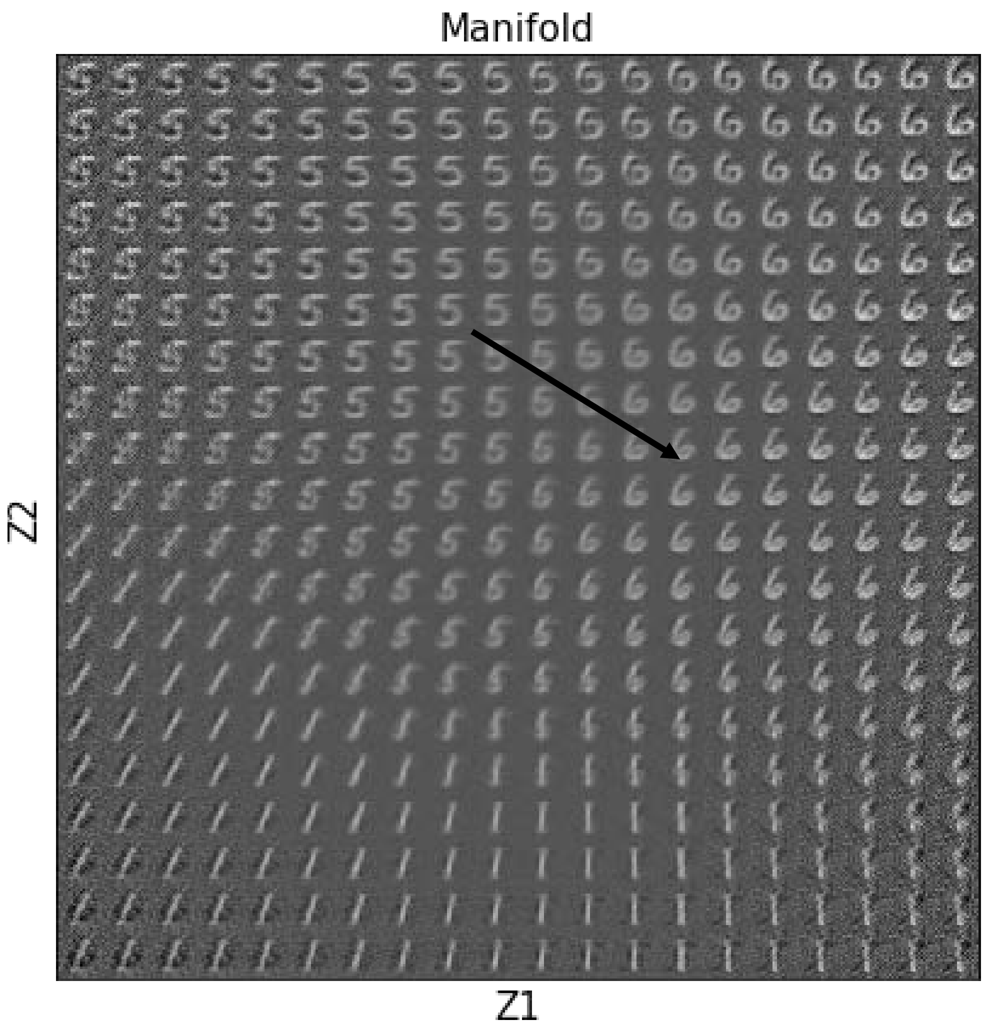
Interpolation in Manifold



https://www.cs.cmu.edu/~efros/courses/AP06/presentations/ThompsonDimensionalityReduction.pdf

MNIST Example: Walk in the Latent Space





Generative Models

- It generates something that makes sense.
- These results are unsatisfying, because the density model used on the latent space $\mathcal{F}$ is too simple and inadequate.
- Building a “good” model amounts to our original problem of modeling an empirical distribution, although it may now be in a lower dimension space.
- This is a motivation to VAE or GAN.