Learning to Reconstruct and Restore Images

David I. Inouye

Dynamic Content · Last Updated: November 4, 2026

Arc 7 · Image generation and evaluation

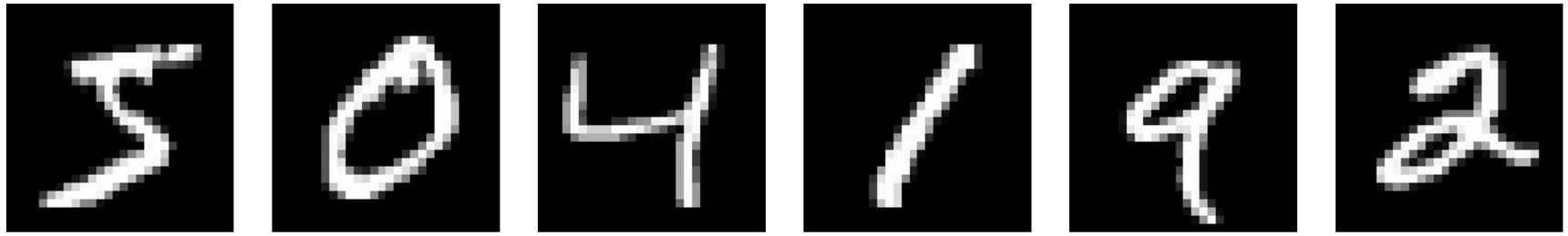
Vanilla autoencoders and denoising autoencoders

We will build from reconstructing images to generating new ones

We generated text one token at a time. Now we turn to **images**, starting with two simpler learning tasks.

1. **Reconstruct an image:** learn an encoder and decoder with an autoencoder. *(Today)*
2. **Restore a corrupted image:** train a denoising autoencoder to recover the clean image. *(Today)*
3. **Make the mappings probabilistic:** then add a generative model to develop VAEs.
4. **Evaluate the results:** compare image quality, diversity, usefulness and cost. *(Later)*

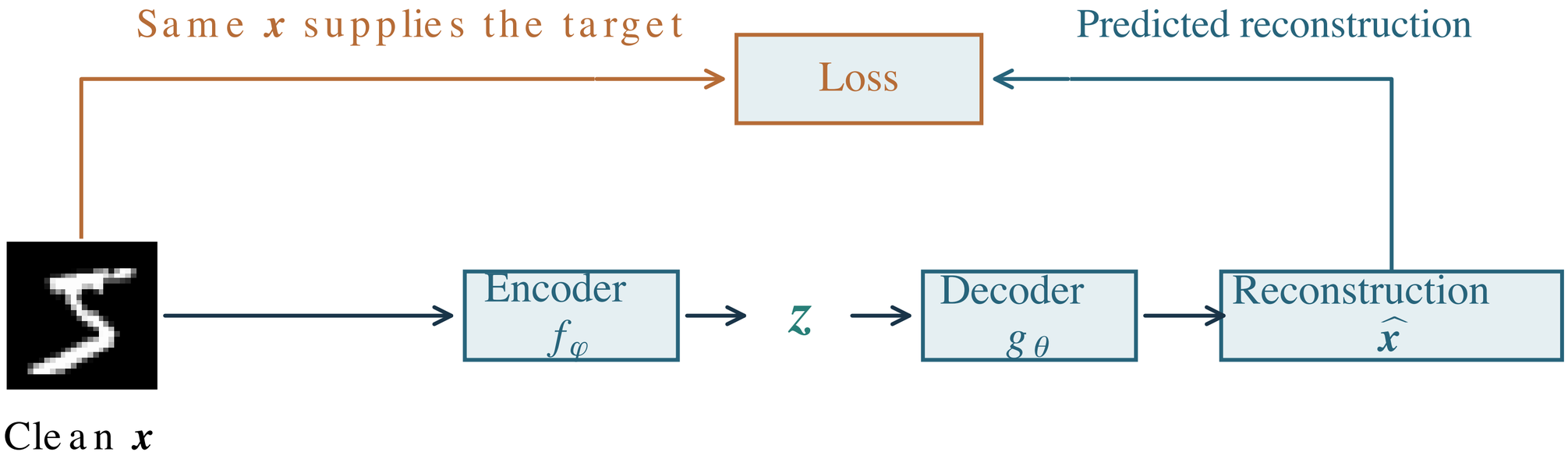
We will use handwritten digits and small color images as examples



- **MNIST:** trace a recognizable digit through encoding, corruption and reconstruction.
- **CIFAR-10:** compare models on 32 × 32 color images with varied objects and backgrounds.

Use the same data, evaluation protocol and declared compute budget for comparisons.

An autoencoder learns a code that supports reconstruction



MNIST training image 0; reconstruction is a symbol, not a trained output.

$$z = f_\phi(x), \qquad \hat{x} = g_\theta(z)$$

Reconstruction error trains both halves together

For images $\boldsymbol{x}_i \in \mathbb{R}^d$, one choice is mean squared error:

$$\min_{\phi,\theta} \frac{1}{nd} \sum_{i=1}^{n} \|\boldsymbol{x}_i - g_\theta(f_\phi(\boldsymbol{x}_i))\|_2^2 .$$

- The image supplies its own reconstruction target.
- Backpropagation updates encoder and decoder parameters through the same loss.

A small pixel error measures reconstruction, not every property we care about.

Perfect copying can solve the loss without a useful code

If both functions can be the identity,

$$f(\boldsymbol{x}) = \boldsymbol{x}, \qquad g(\boldsymbol{z}) = \boldsymbol{z} \quad \Longrightarrow \quad \|\boldsymbol{x} - g(f(\boldsymbol{x}))\|_2^2 = 0.$$

The reconstruction objective needs a constraint that makes learning useful structure worthwhile.

Our first choice: pass through a smaller code, $\boldsymbol{z} \in \mathbb{R}^k$ with $k < d$.

Remember: PCA encodes an input in fewer features and reconstructs it

We already know a **linear encoder and decoder**: PCA.

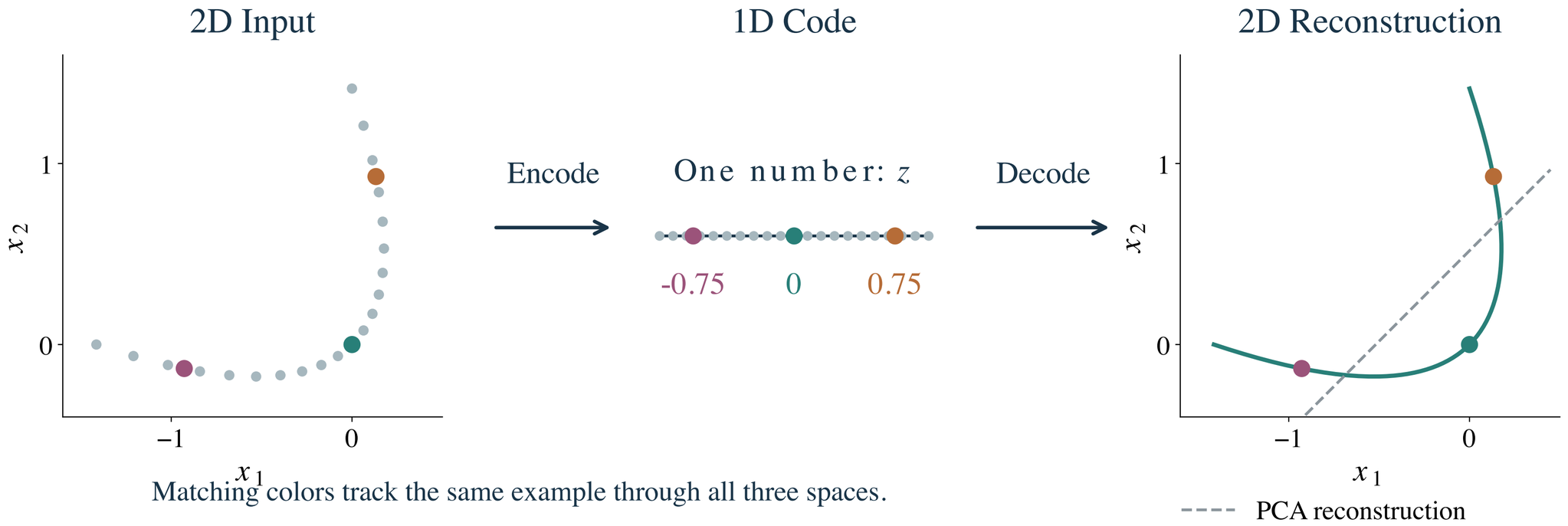
For a centered input $\boldsymbol{x}$ and principal directions $U_k \in \mathbb{R}^{d \times k}$:

$$\underbrace{\boldsymbol{z} = U_k^\top \boldsymbol{x}}_{\text{encode into } k \text{ features}}, \qquad \underbrace{\widehat{\boldsymbol{x}} = U_k \boldsymbol{z}}_{\text{decode back to } d \text{ features}}.$$

This gives an optimal rank-k linear reconstruction under squared error.

Learned linear AE factors can differ while producing the same reconstruction.

A nonlinear decoder maps a 1D code back to a 2D curve



Matching colors track the same example through all three spaces.

Encode: $z = (x_1 + x_2)/\sqrt{2}.$ **Decode:** $\widehat{x} = \frac{1}{\sqrt{2}}[z - z^2,\ z + z^2]^\mathsf{T}.$

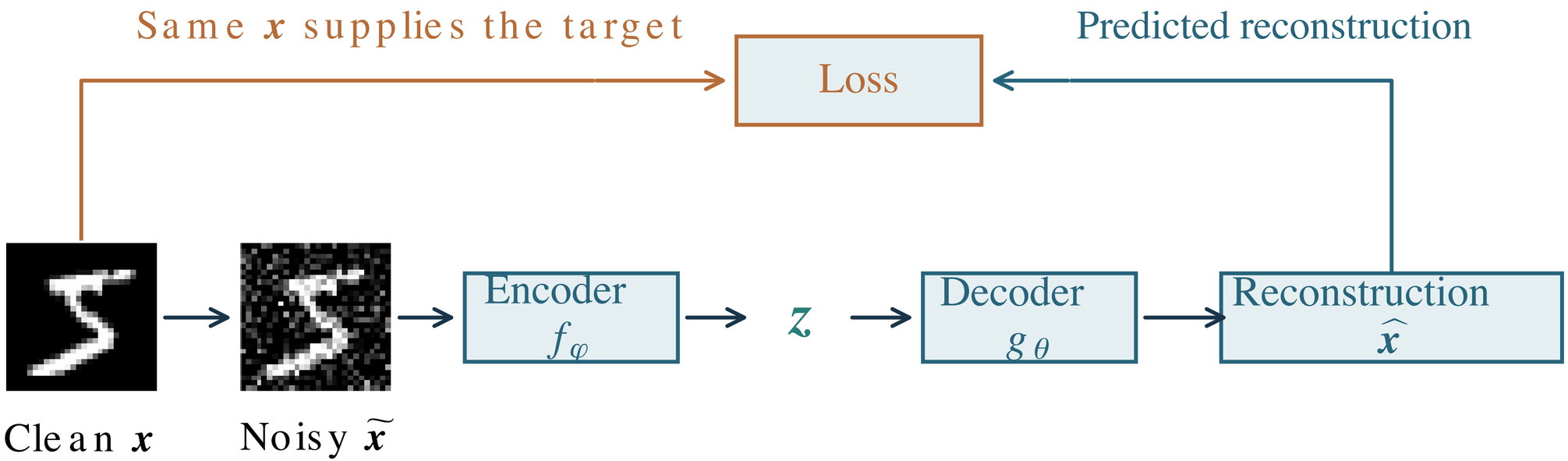
The code combines both input features. *Analytic example; no model was trained.*

What does a small reconstruction error tell us?

> **An AE reconstructs held-out images accurately.**
> **Which conclusion is supported?**
> **A. Its codes separate all useful classes.**
> **B. Its reconstructions preserve the measured image features well.**
> **C. Random Gaussian codes produce realistic images.**

B. Classification and generation need their own evidence.

A denoising AE receives corruption but targets the clean image



MNIST training image 0; reconstruction is a symbol, not a trained output.

Copying the corrupted input now disagrees with the clean target.

Denoising trains against the original, not the corrupted input

Construct $\widetilde{x} = c(x, e)$ using a sampled corruption:

$$\min_{\phi,\theta} \; \mathbb{E}_{\mathbf{x},\mathbf{e}} \left[\frac{1}{d} \| \mathbf{x} - g_\theta(f_\phi(c(\mathbf{x}, \mathbf{e}))) \|_2^2 \right].$$

Input sent to encoder	Target used in loss
Corrupted image $\widetilde{x}$	Clean image x

The chosen corruption shapes which structure helps solve the task.