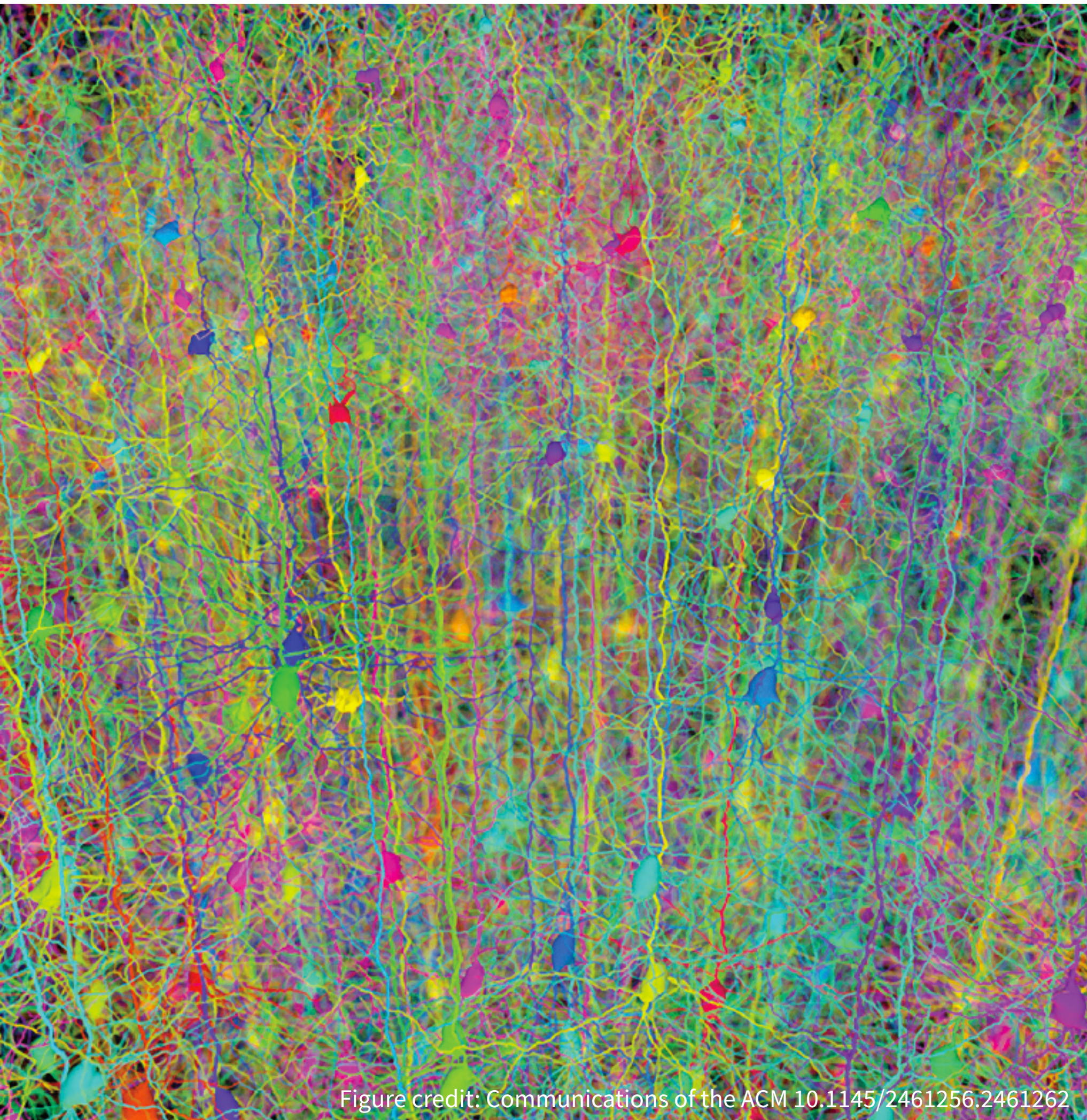


Autoencoders



GRAHAM TAYLOR

VECTOR INSTITUTE

SCHOOL OF ENGINEERING
UNIVERSITY OF GUELPH

CANADIAN INSTITUTE
FOR ADVANCED RESEARCH

UNIVERSITY
of GUELPH

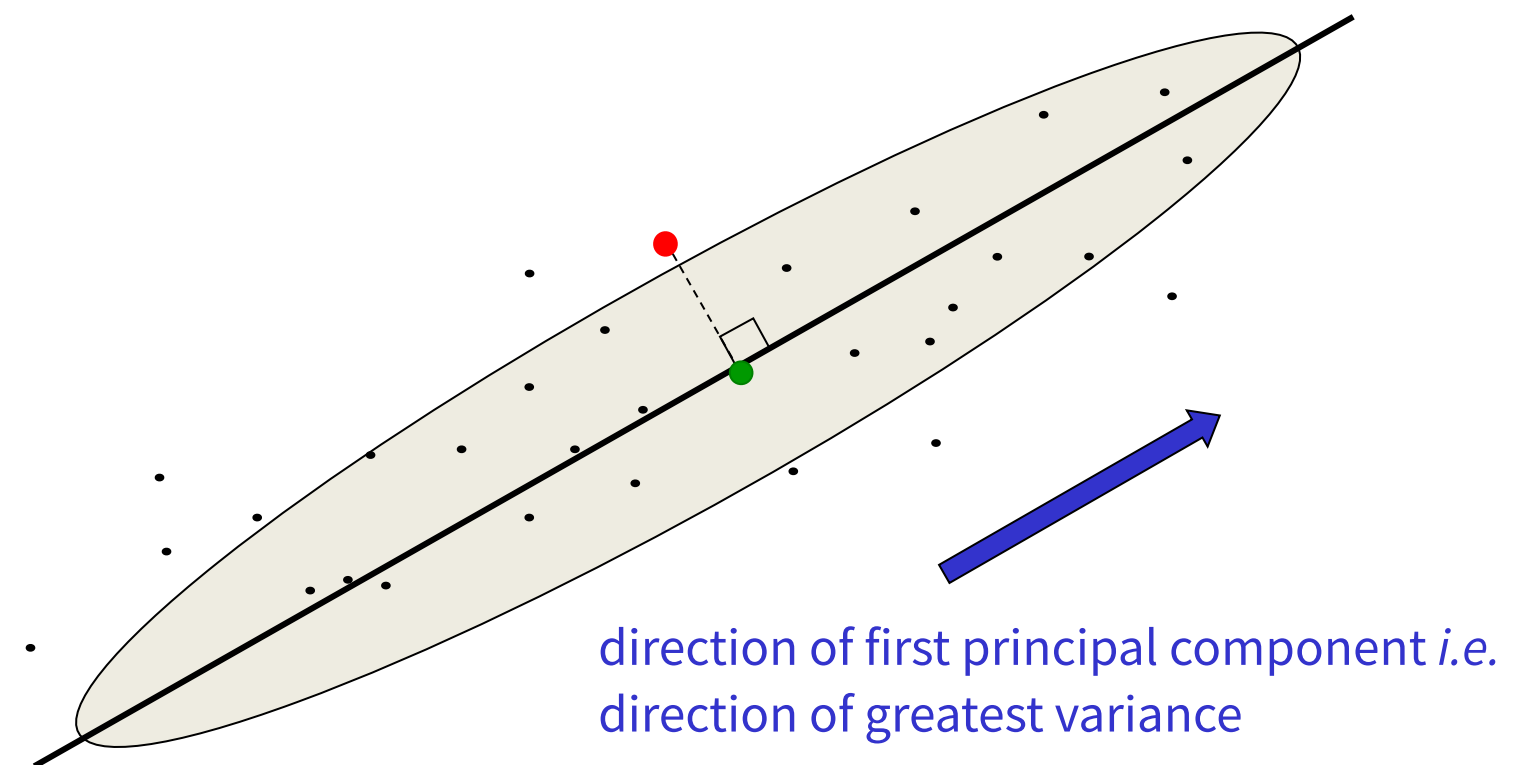
CHANGING LIVES
IMPROVING LIFE

CIFAR
CANADIAN
INSTITUTE
FOR
ADVANCED
RESEARCH

Figure credit: Communications of the ACM 10.1145/2461256.2461262

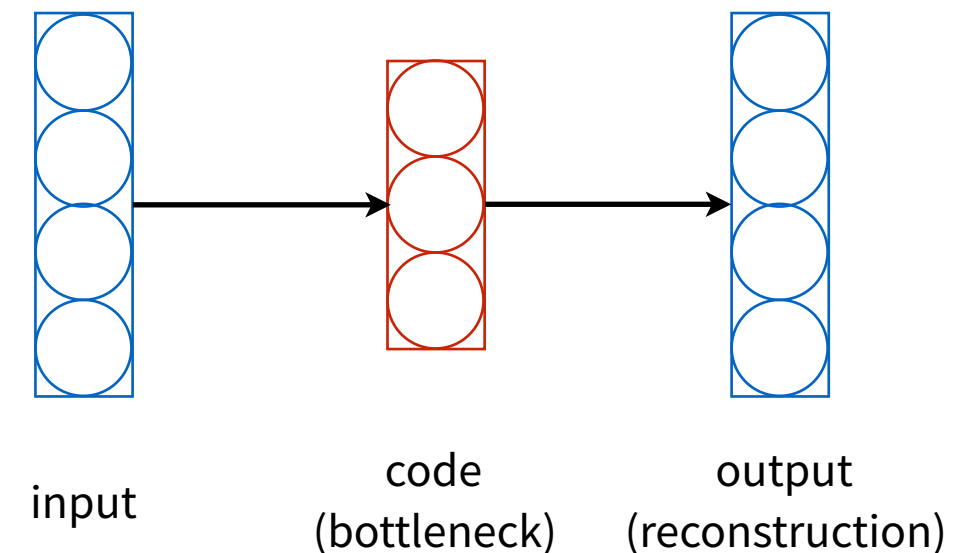
Principal Components Analysis

- PCA works well when the data is near a **linear manifold** in high-dimensional space
- **Project** the data onto this subspace spanned by principal components
- In dimensions orthogonal to the subspace the data has low variance



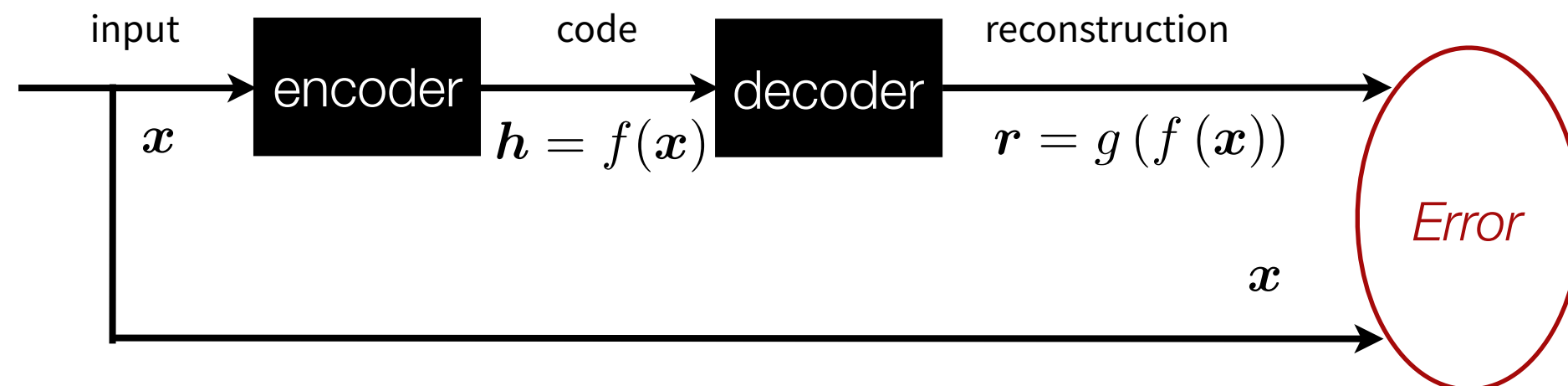
An inefficient way to fit PCA

- Train a neural network with a “bottleneck” hidden layer
- Try to make the output the same as the input



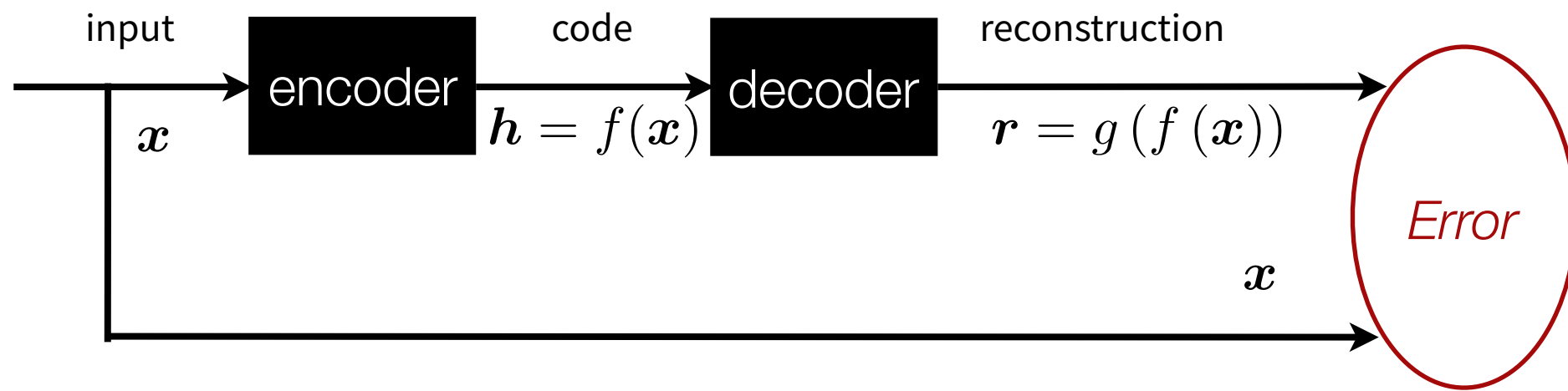
- If the hidden and output layers are linear, and we minimize squared reconstruction error:
 - The M hidden units will span the same space as the first M principal components
 - But their weight vectors will not be orthogonal
 - And they will have approximately equal variance

Why fit PCA inefficiently?



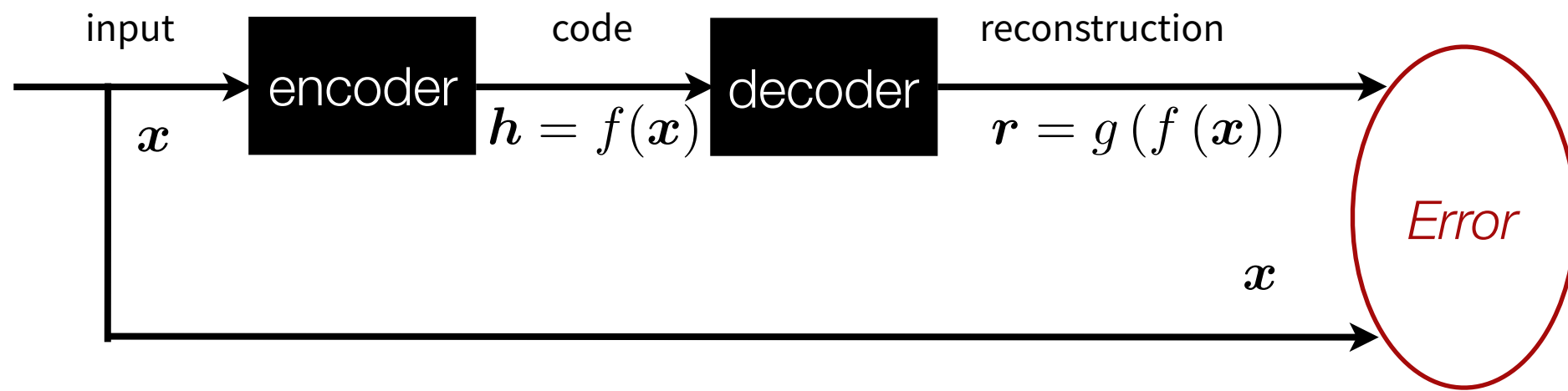
- With nonlinear layers before and after the code, it should be possible to represent data that lies on or near a nonlinear manifold
 - the encoder maps from data space to co-ordinates on the manifold
 - the decoder does the inverse transformation
- The encoder/decoder can be rich, multi-layer functions

Auto-encoder



- Feed-forward architecture
- Trained to minimize reconstruction error
 - bottleneck or regularization essential

Auto-encoder



- Feed-forward architecture
- Trained to minimize reconstruction error
 - bottleneck or regularization essential

Example: real-valued data

Encoder

$$h_i = \sigma \left(\sum_j W_{i,j} x_j \right)$$

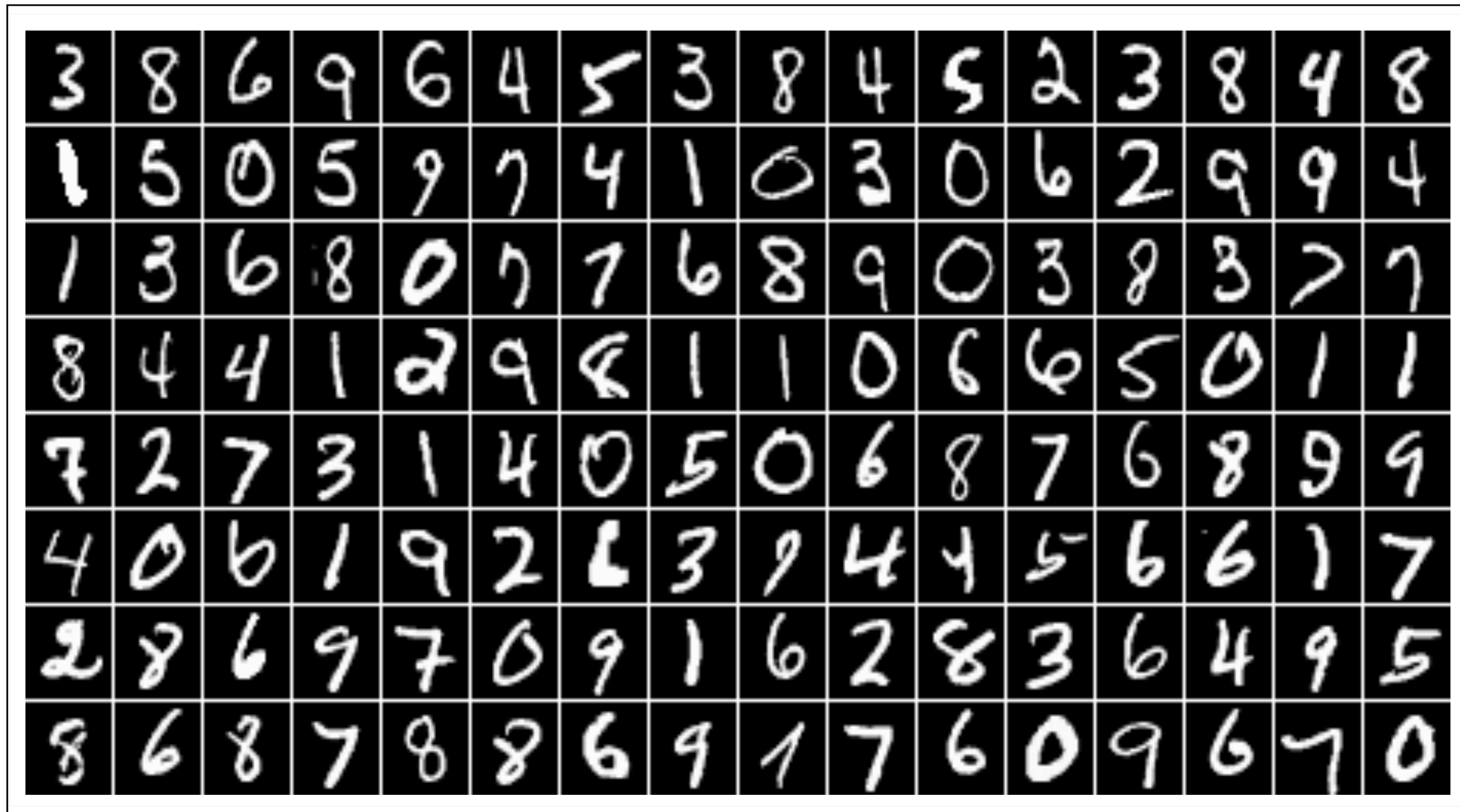
Decoder

$$r_j = \sum_i W_{i,j} h_i$$

Error

$$L = ||\mathbf{r} - \mathbf{x}||^2$$

Autoencoder: Example (MNIST)



Autoencoder: Filters

