EE 604
Digital Image Processing

IIT KANPUR
Indian Institute of Technology, Kanpur

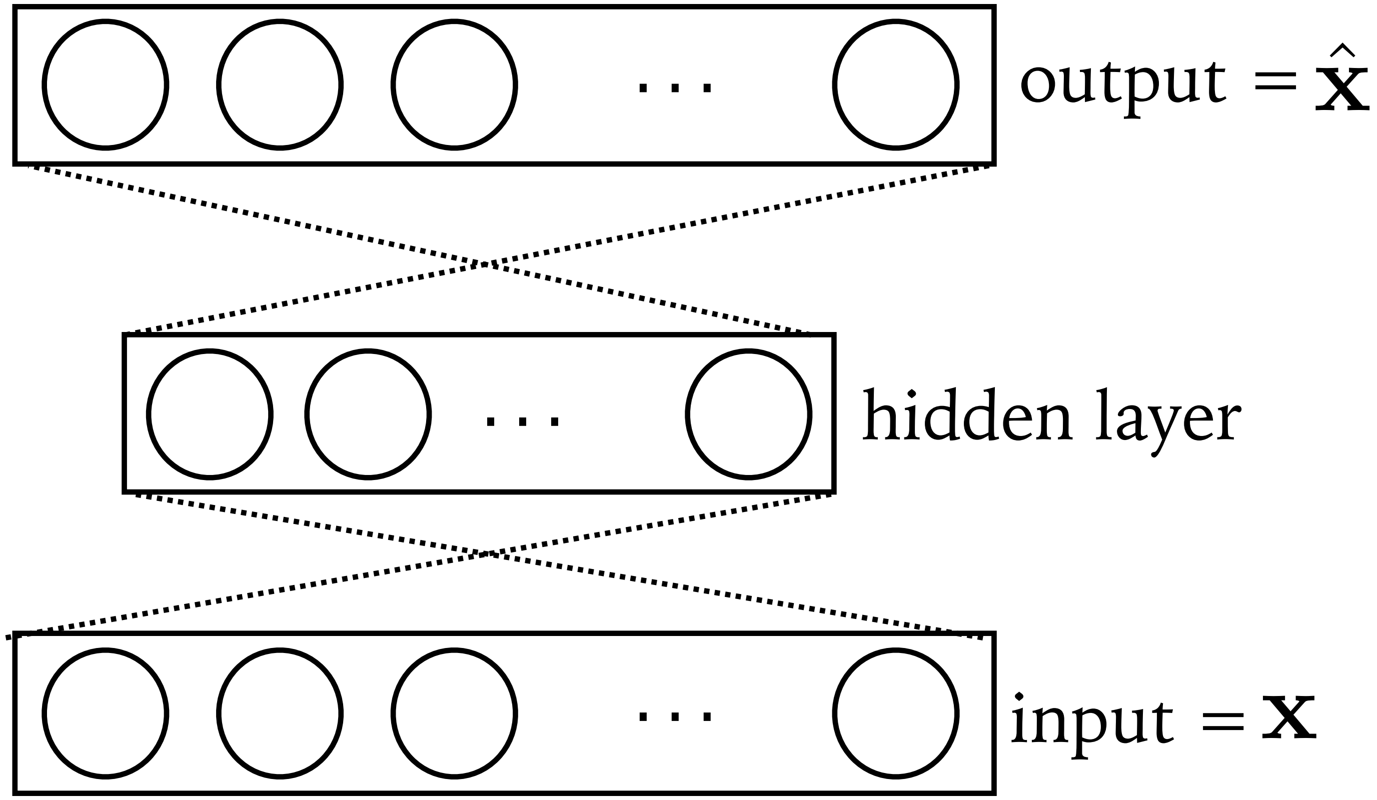
Tanaya Guha
Aug - Nov 2017

Denoising Autoencoder

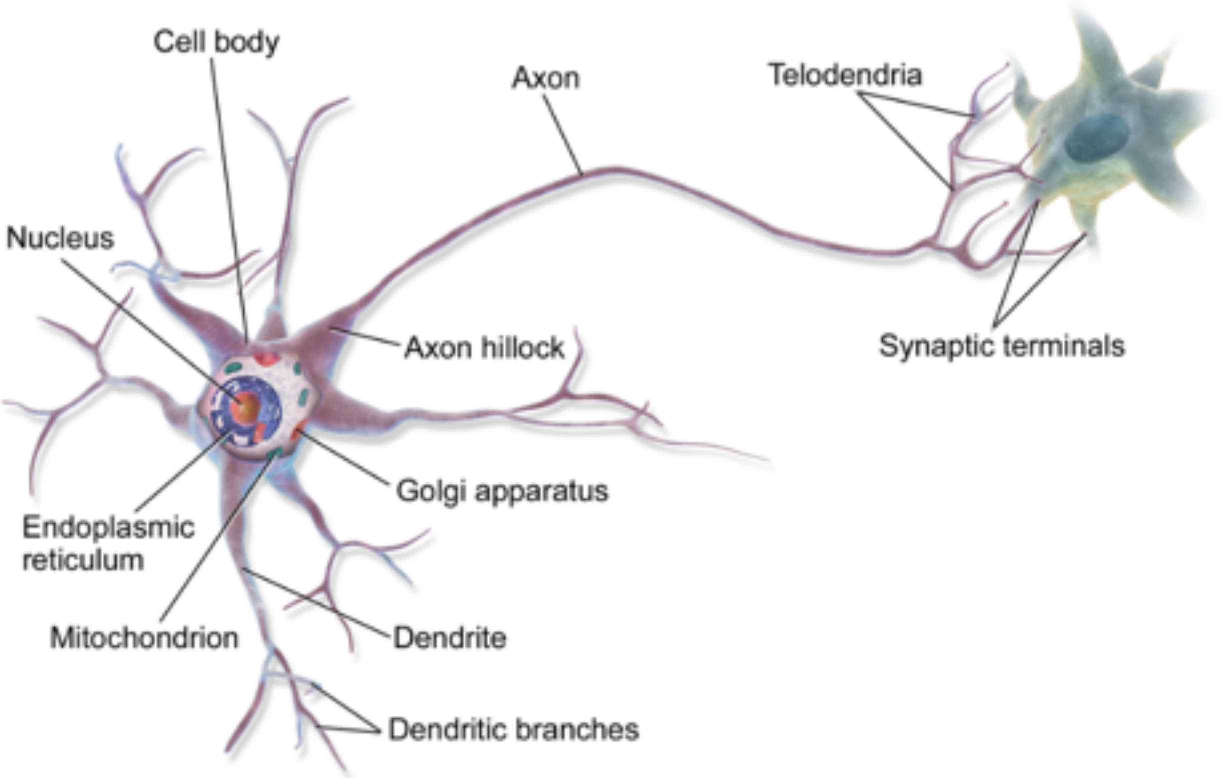
- A completely data-driven approach to denoising
- Achieves state-of-the-art results [Vincent’08]
- An extension of the classical auto encoder
- A building block of deep neural networks

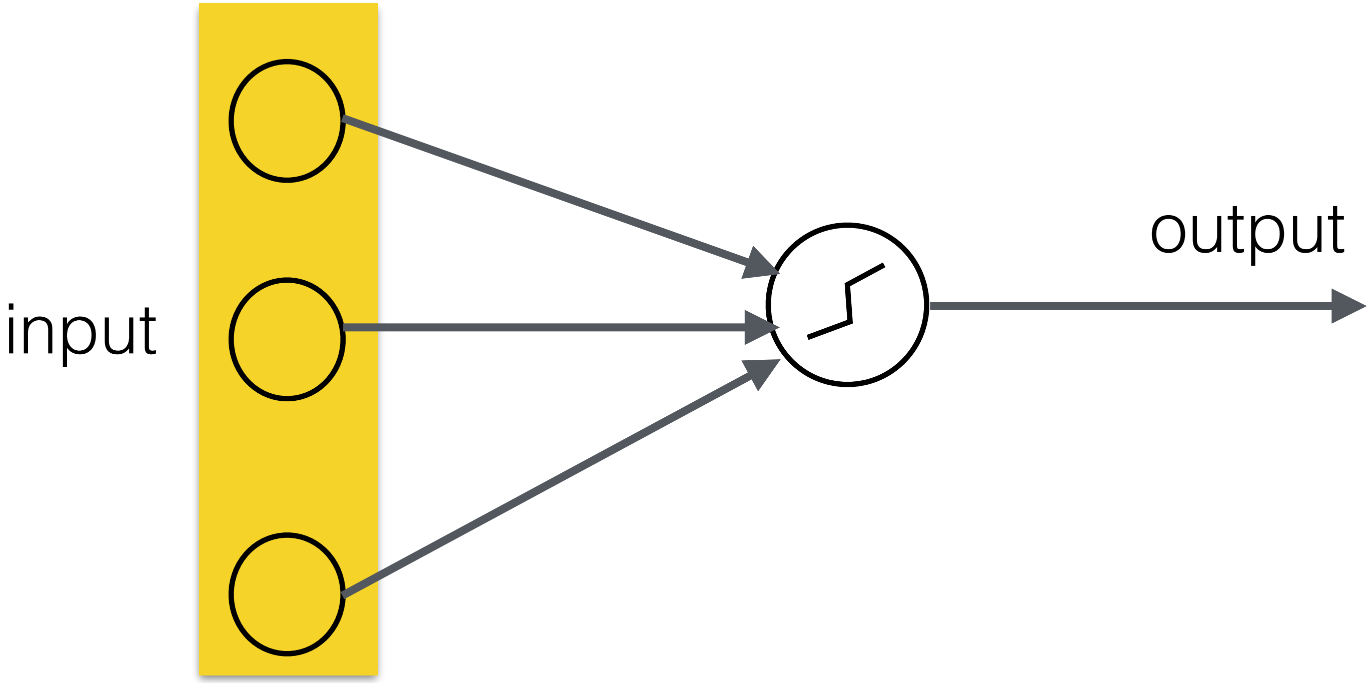
Autoencoder

- **Unsupervised** learning method
- A **neural network** that learns to map the input with itself

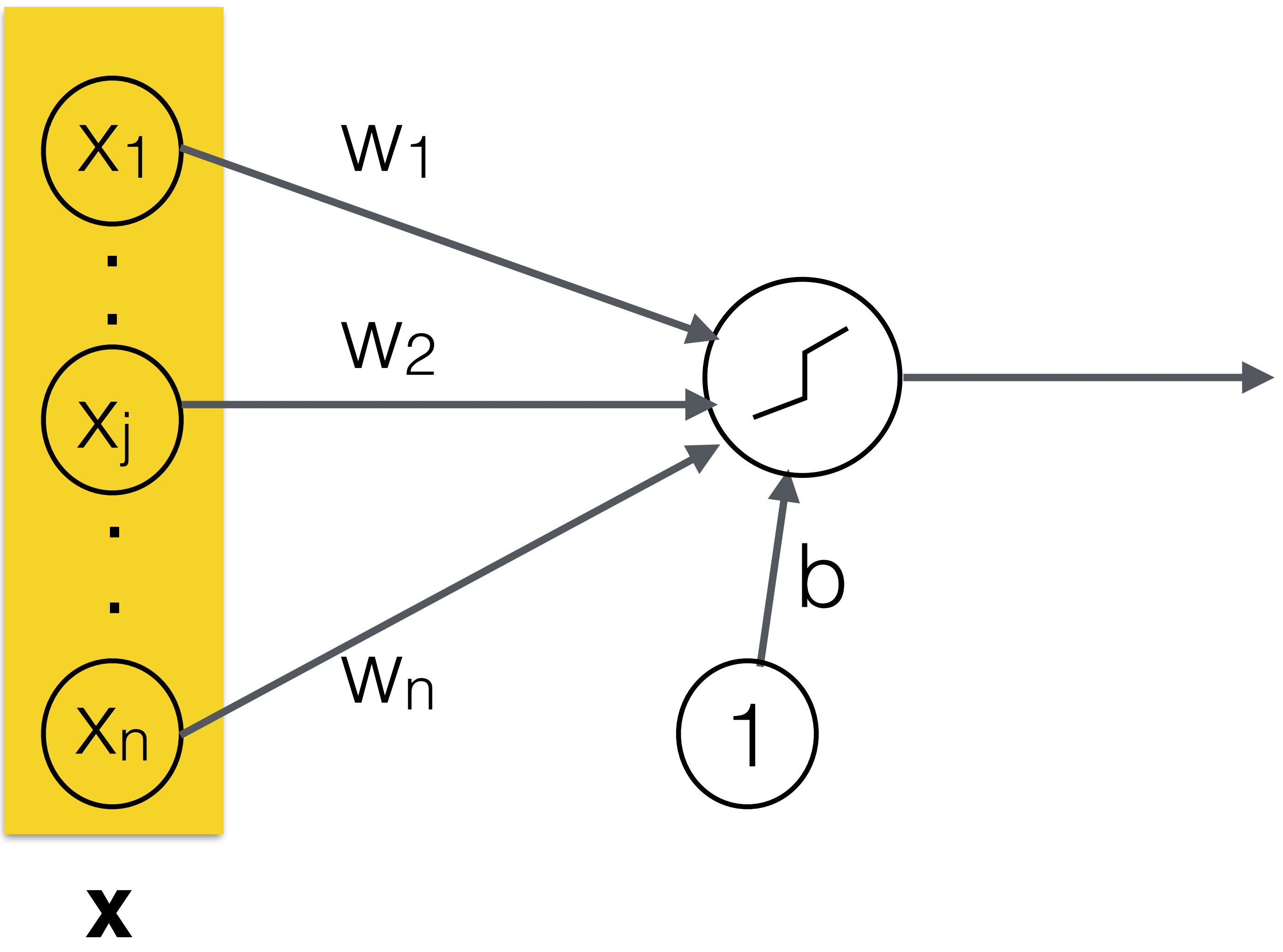


A single neuron



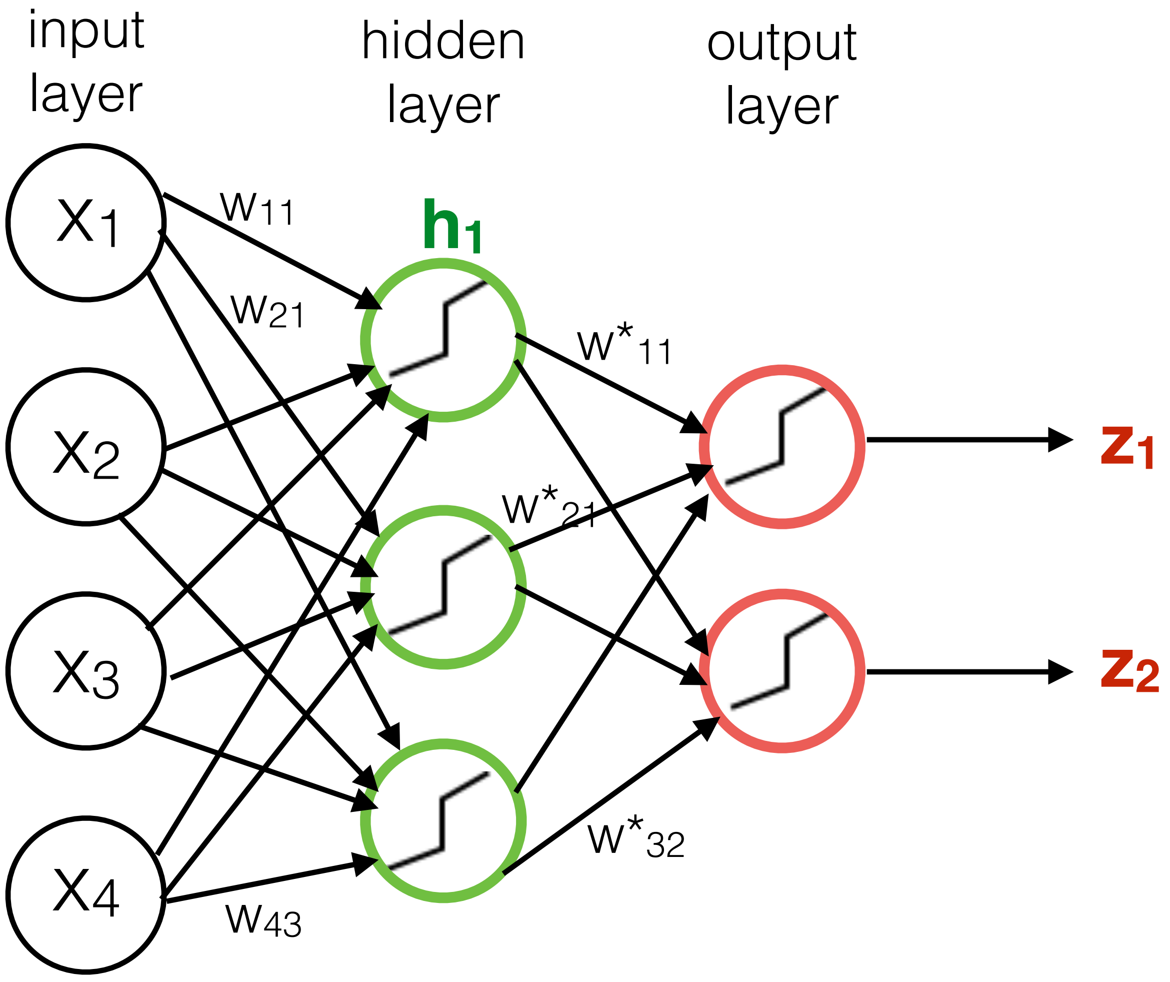


A single neuron

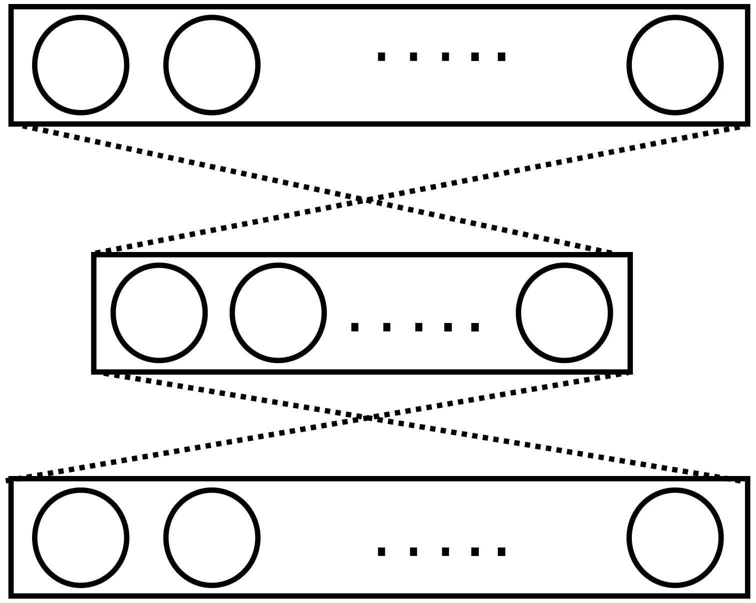


$$h = g(b + \sum_{i=1}^{n} w_i x_i)$$

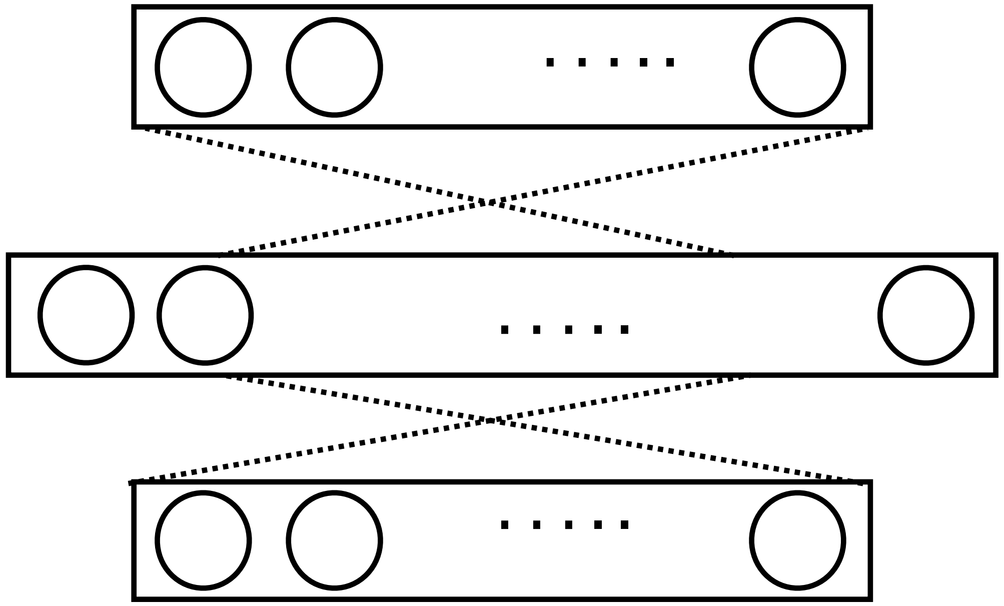
A simple neural net



Autoencoder architecture

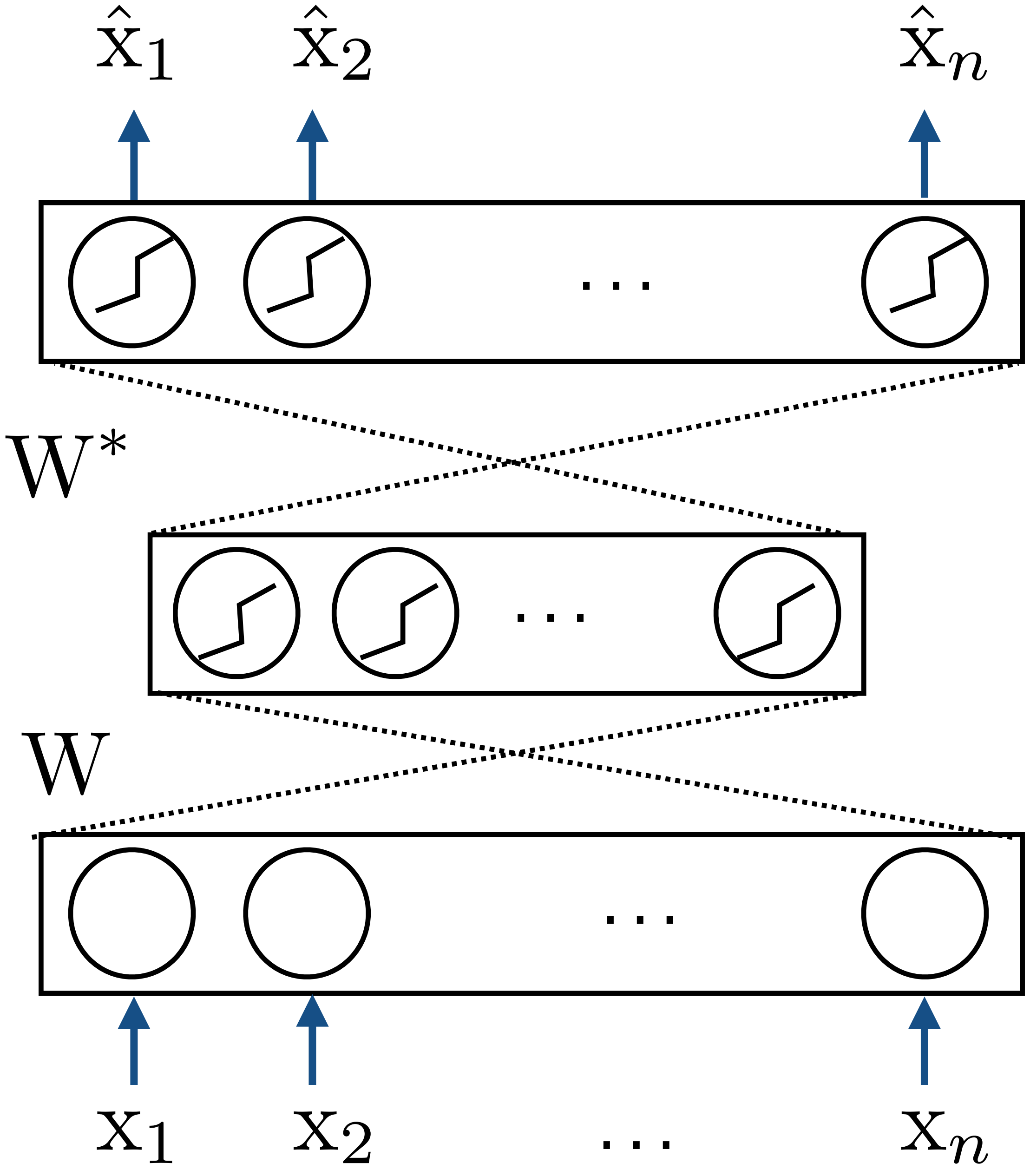


Undercomplete



Overcomplete

Autoencoder



$$\hat{\mathbf{x}} = g_2(\mathbf{b}^* + \mathbf{W}^*\mathbf{h})$$

$$\mathbf{h} = g_1(\mathbf{b} + \mathbf{W}\mathbf{x})$$

$$\text{Loss} = \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2$$

Autoencoder

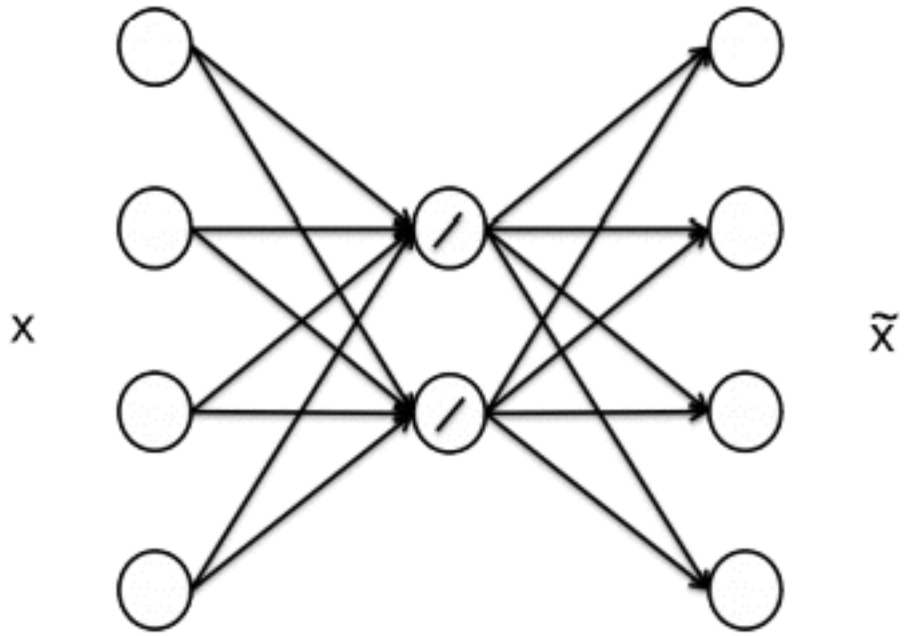
- Consider inputs to be 10x10 images
- **x** are the pixel intensities
- **n** = 100 in the input layer
- Hidden layer => 50 nodes/neurons (say)
- Output layer = 100 nodes

- Network is forced to learn a “compressed” representation of the input.
- What happens if input values are independent?

Autoencoder

- A common practice: $\mathbf{W}^* = \mathbf{W}^T$

- What if we have 1 hidden layer and linear activations?



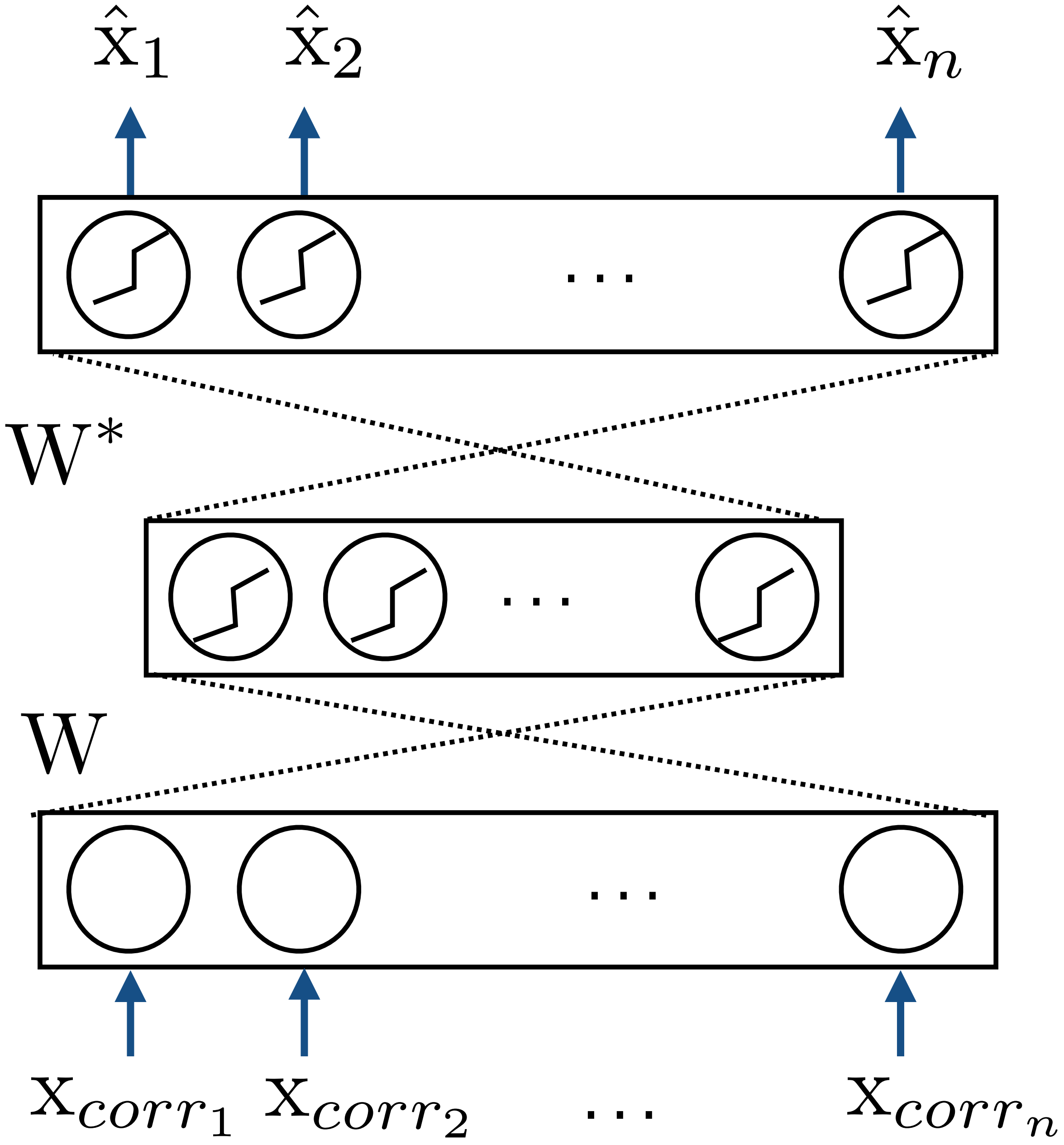
Training an autoencoder

- Initialize all $\mathbf{W}, \mathbf{b}$
- For every input (training) image
  - encode: $\mathbf{h} = \mathrm{g}_1(\mathbf{b} + \mathbf{W}\mathbf{x})$
  - decode: $\hat{\mathbf{x}} = \mathrm{g}_2(\mathbf{b}^* + \mathbf{W}^*\mathbf{h})$
- Compute **loss** for all training images
  - Determine $\mathbf{W}, \mathbf{b}$ by gradient descent

Denoising autoencoder

- **Idea:** corrupt the input before feeding to the network, try to reconstruct the clean image
- Hidden layer learns representation "robust" to noise
- [Vincent 2008]
  - Input is randomly corrupted by setting pixels to 0$\mathbf{x}_{corr}$
  - Loss is computed w.r.t the clean image
- Works well for other noise too.

Autoencoder



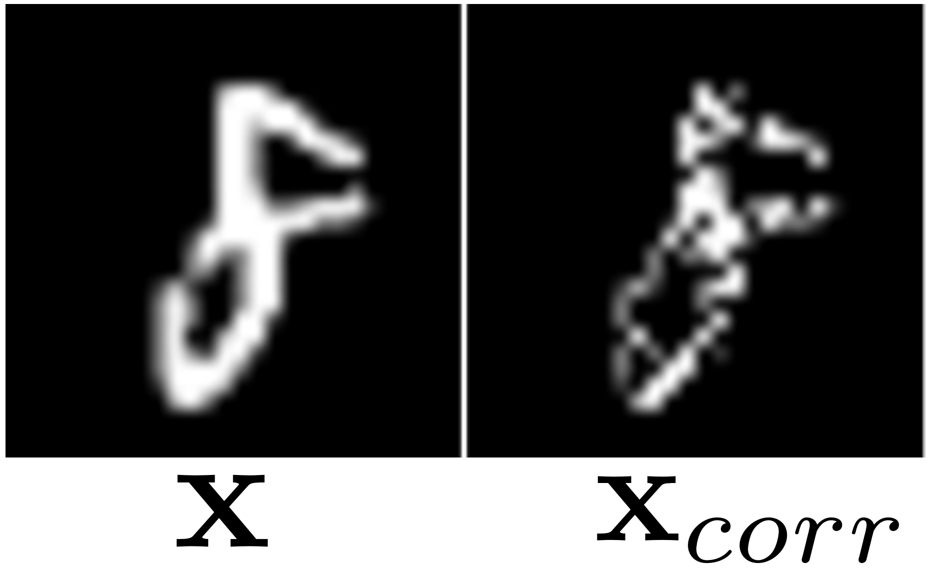
$$\hat{\mathbf{x}} = g_2(\mathbf{b}^* + \mathbf{W}^*\mathbf{h})$$

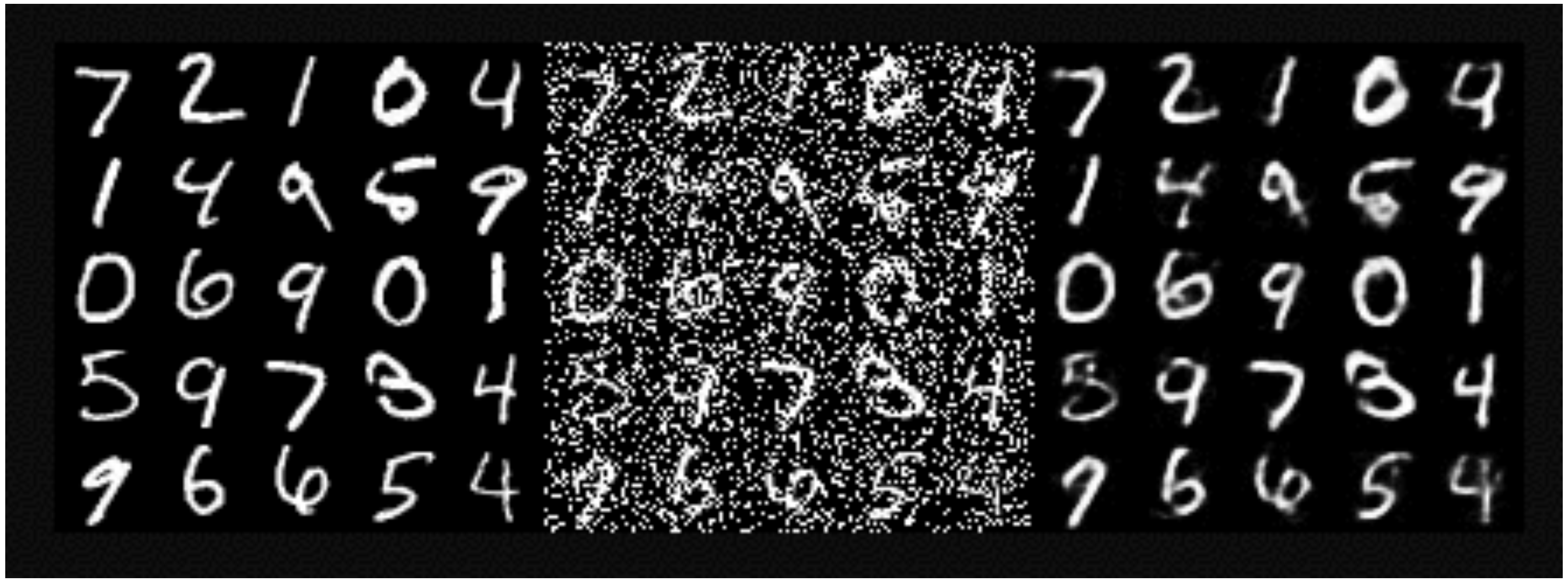
$$\mathbf{h} = g_1(\mathbf{b} + \mathbf{W}\mathbf{x}_{corr})$$

$$\text{Loss} = \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2$$

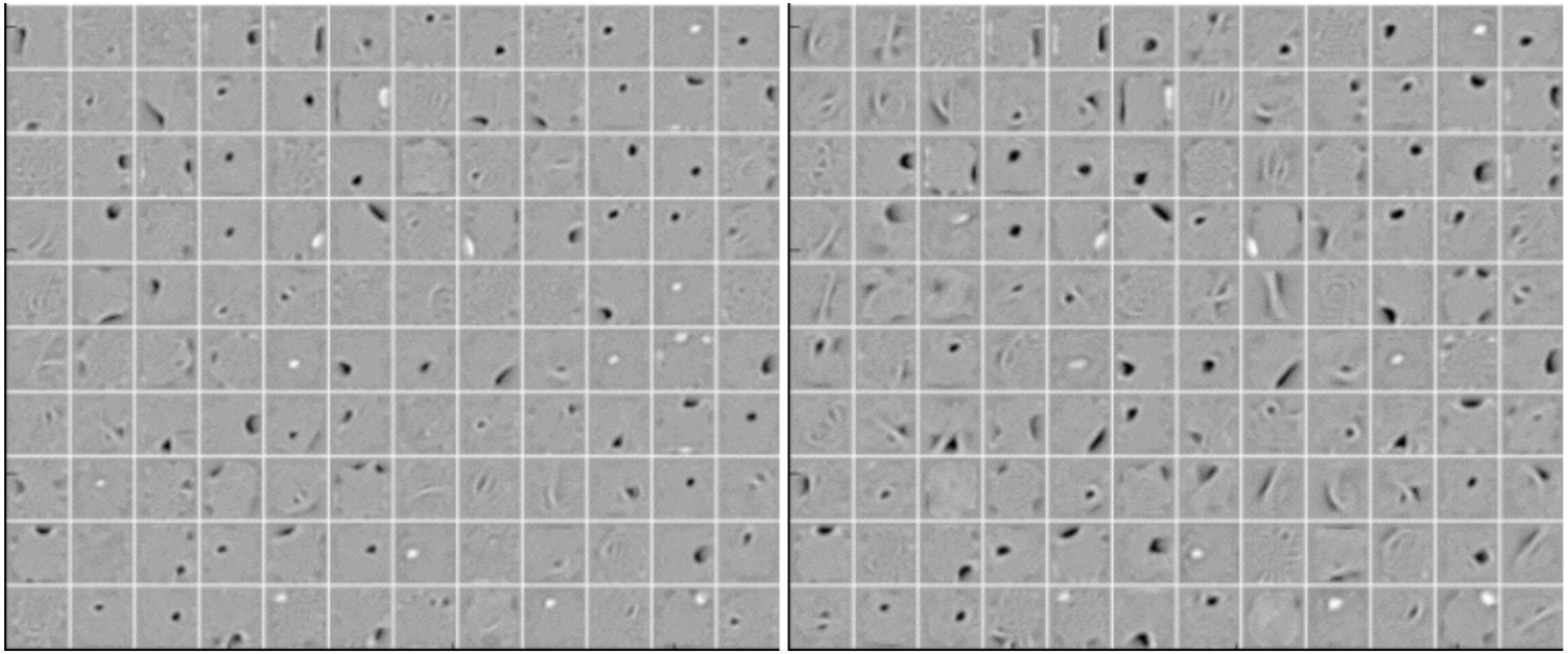
Note that the loss is computed w.r.t the clean image.

Denoising autoencoder





Denoising autoencoder



Visualization of weights for denoising auto encoders trained with 25% and 50% corrupted images as inputs. Note that as noise increases, the weights (filters) resemble edge-like structures.