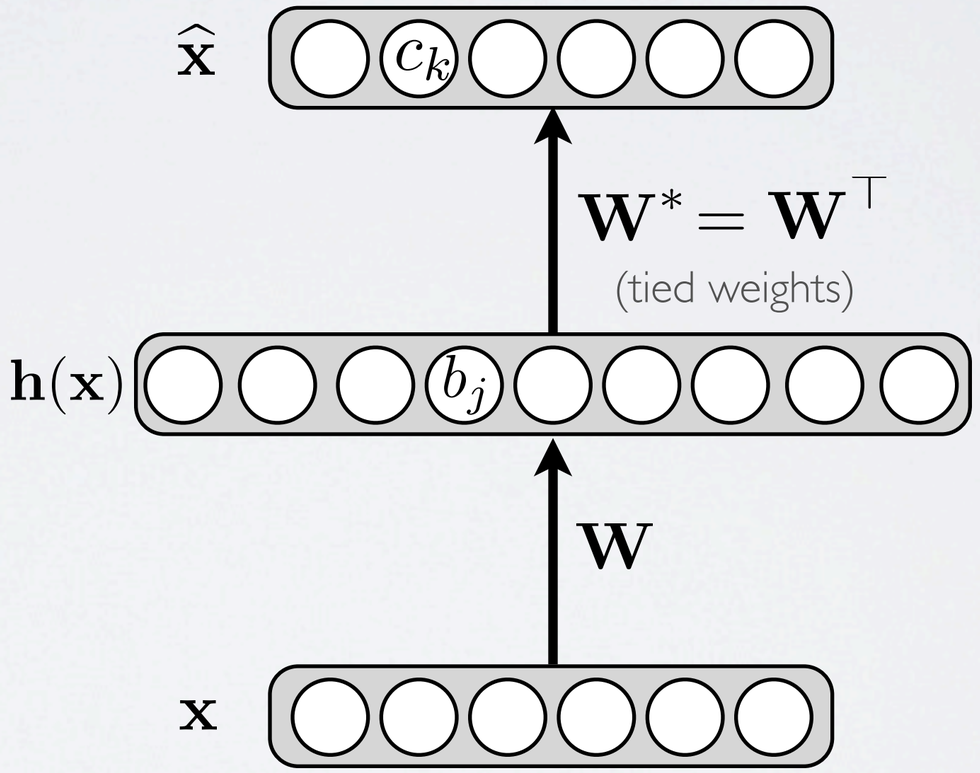
Neural networks

Autoencoder - denoising autoencoder

OVERCOMPLETE HIDDEN LAYER

Topics: overcomplete representation

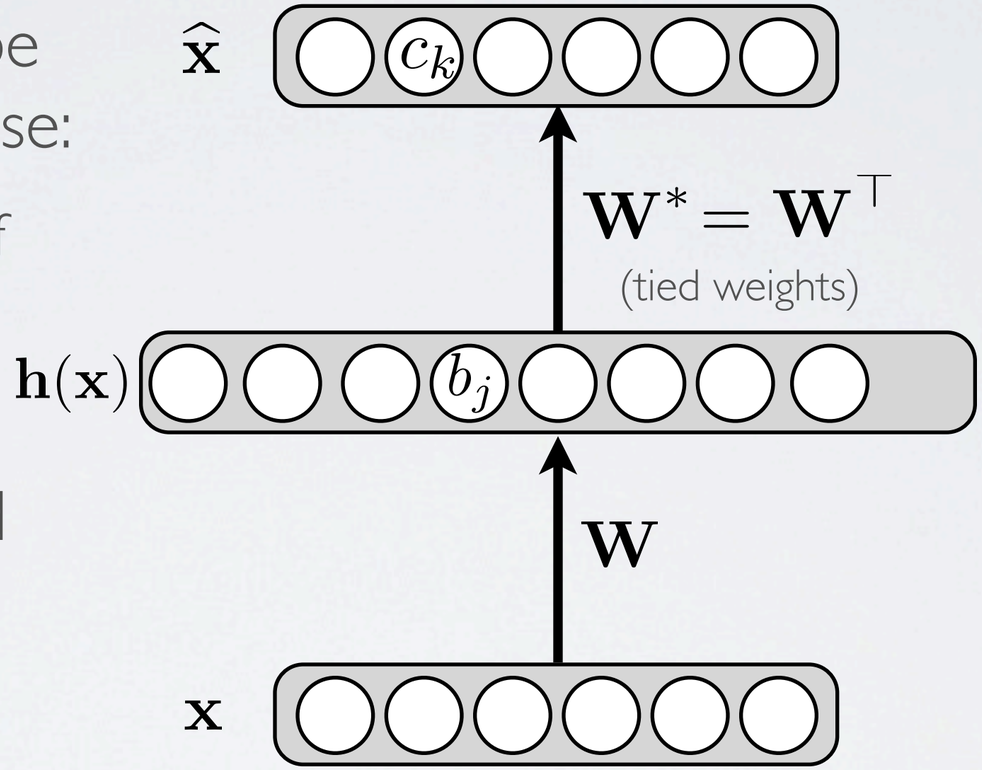
- Hidden layer is overcomplete if greater than the input layer
  - no compression in hidden layer
  - each hidden unit could copy a different input component
- No guarantee that the hidden units will extract meaningful structure



DENOISING AUTOENCODER

Topics: denoising autoencoder

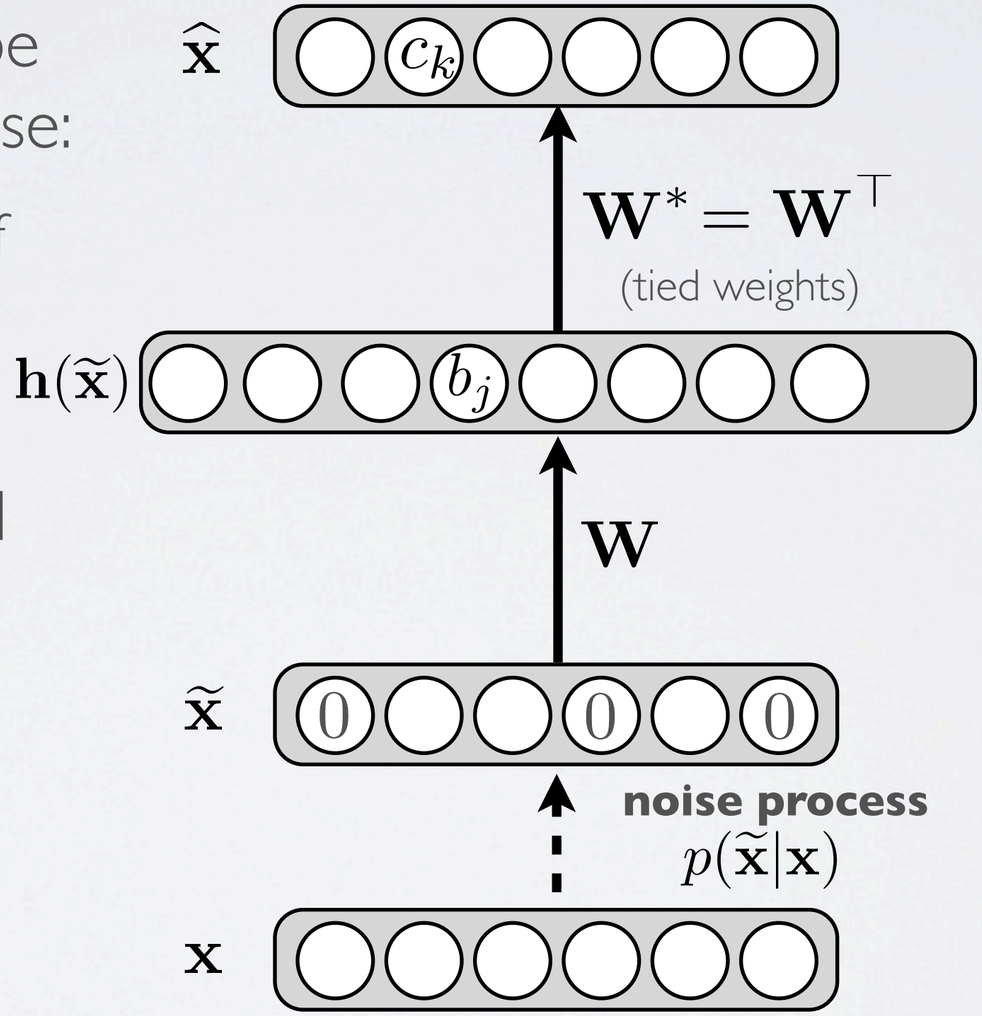
- Idea: representation should be robust to introduction of noise:
  - random assignment of subset of inputs to 0, with probability ν
  - Gaussian additive noise
- Reconstruction $\widehat{\mathbf{x}}$ computed from the corrupted input $\widetilde{\mathbf{x}}$
- Loss function compares $\widehat{\mathbf{x}}$ reconstruction with the **noiseless input $\mathbf{x}$**



DENOISING AUTOENCODER

Topics: denoising autoencoder

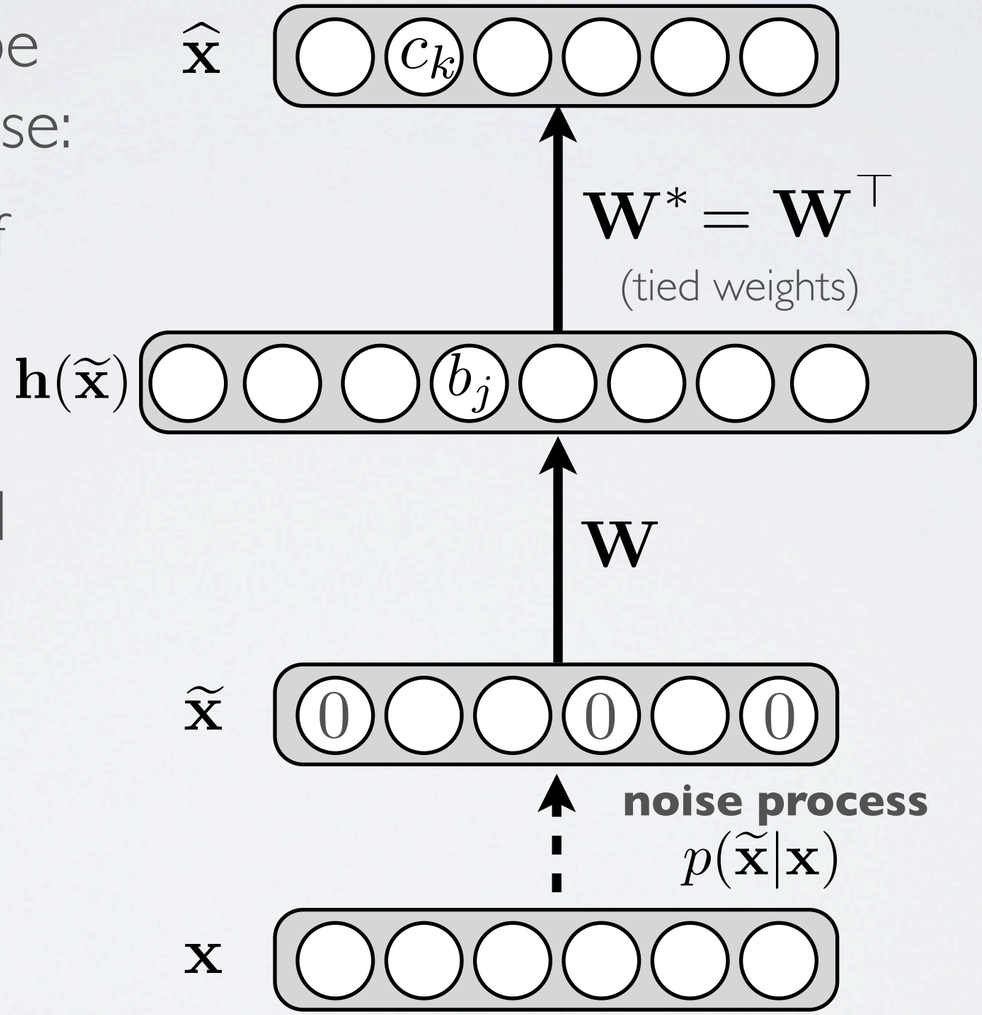
- Idea: representation should be robust to introduction of noise:
  - random assignment of subset of inputs to 0, with probability ν
  - Gaussian additive noise
- Reconstruction $\widehat{\mathbf{x}}$ computed from the corrupted input $\widetilde{\mathbf{x}}$
- Loss function compares $\widehat{\mathbf{x}}$ reconstruction with the **noiseless input $\mathbf{x}$**



DENOISING AUTOENCODER

Topics: denoising autoencoder

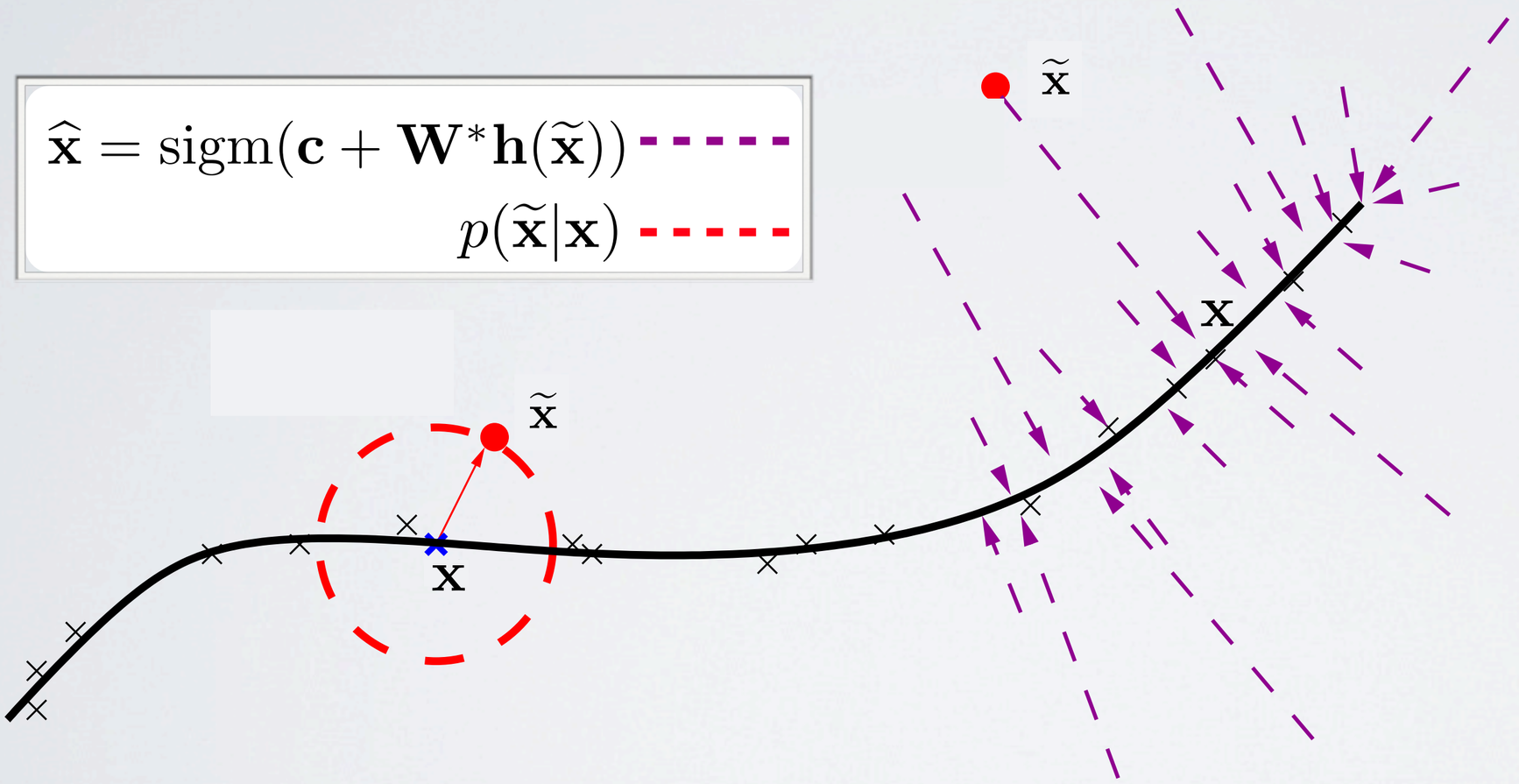
- Idea: representation should be robust to introduction of noise:
  - random assignment of subset of inputs to 0, with probability ν
  - Gaussian additive noise
- Reconstruction $\widehat{\mathbf{x}}$ computed from the corrupted input $\widetilde{\mathbf{x}}$
- Loss function compares $\widehat{\mathbf{x}}$ reconstruction with the **noiseless input $\mathbf{x}$**



DENOISING AUTOENCODER

$$\widehat{\mathbf{x}} = \mathrm{sigm}(\mathbf{c} + \mathbf{W}^* \mathbf{h}(\widetilde{\mathbf{x}}))$$

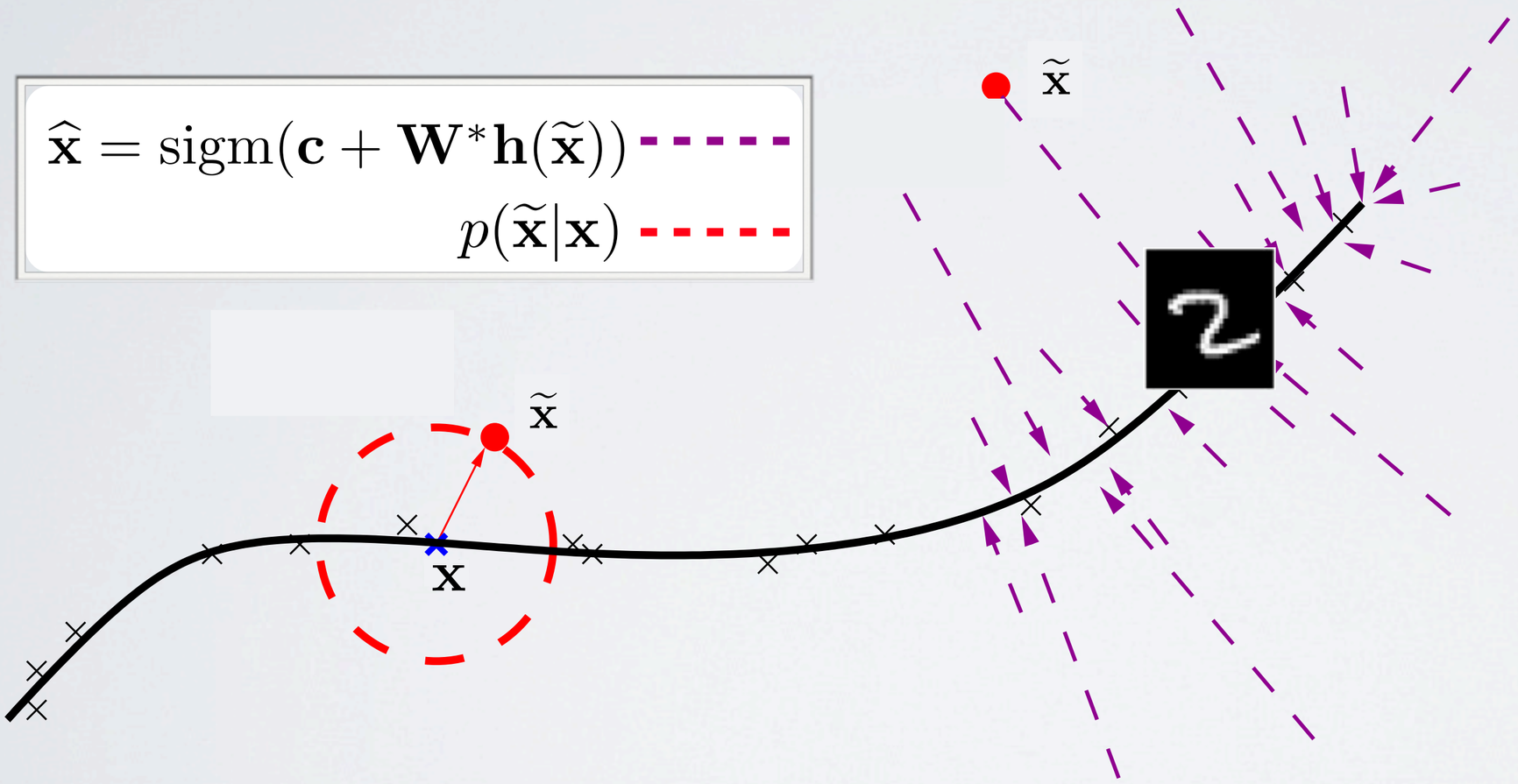
$$p(\widetilde{\mathbf{x}}|\mathbf{x})$$



DENOISING AUTOENCODER

$$\widehat{\mathbf{x}} = \mathrm{sigm}(\mathbf{c} + \mathbf{W}^* \mathbf{h}(\widetilde{\mathbf{x}}))$$

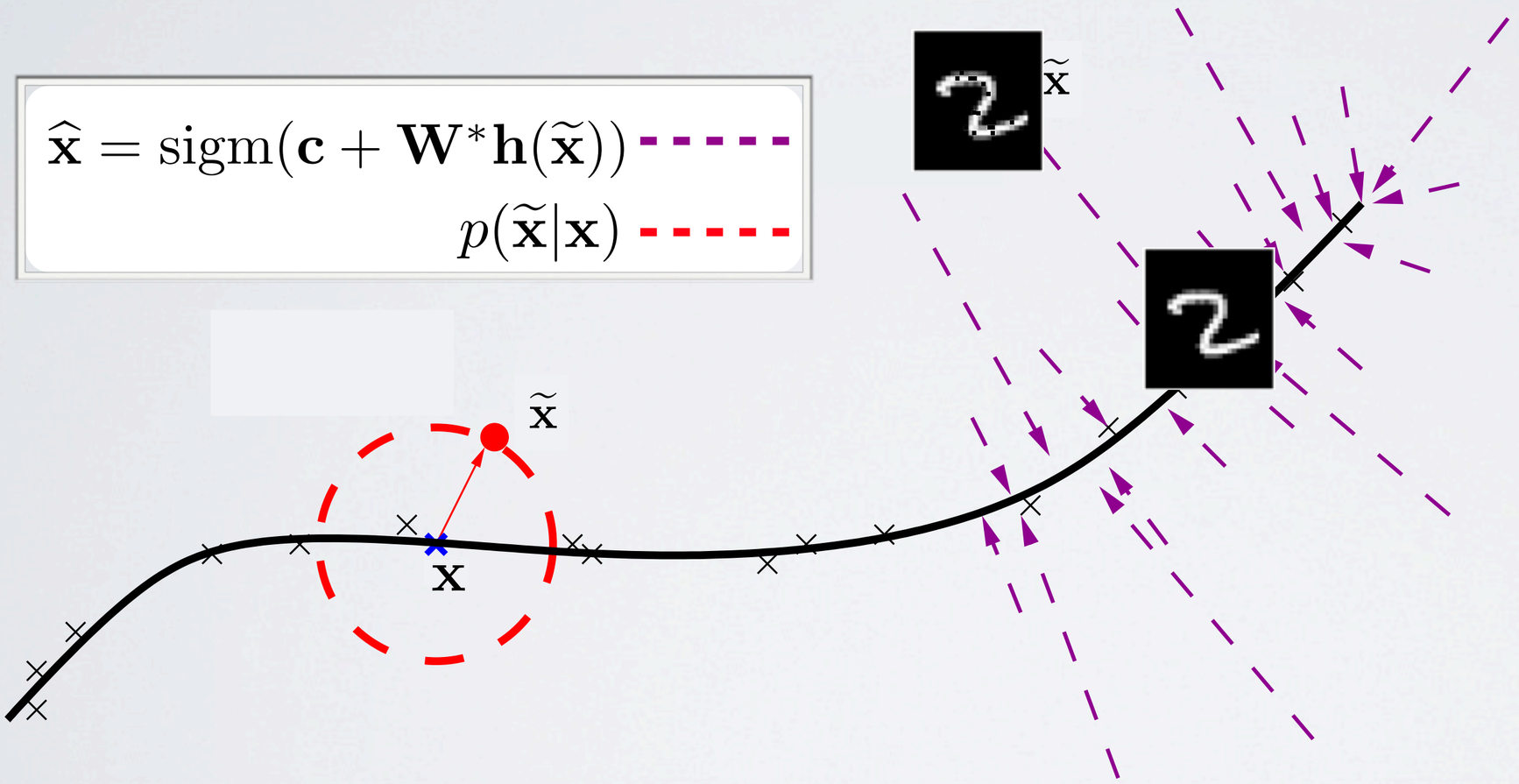
$$p(\widetilde{\mathbf{x}}|\mathbf{x})$$



DENOISING AUTOENCODER

$$\widehat{\mathbf{x}} = \mathrm{sigm}(\mathbf{c} + \mathbf{W}^* \mathbf{h}(\widetilde{\mathbf{x}}))$$

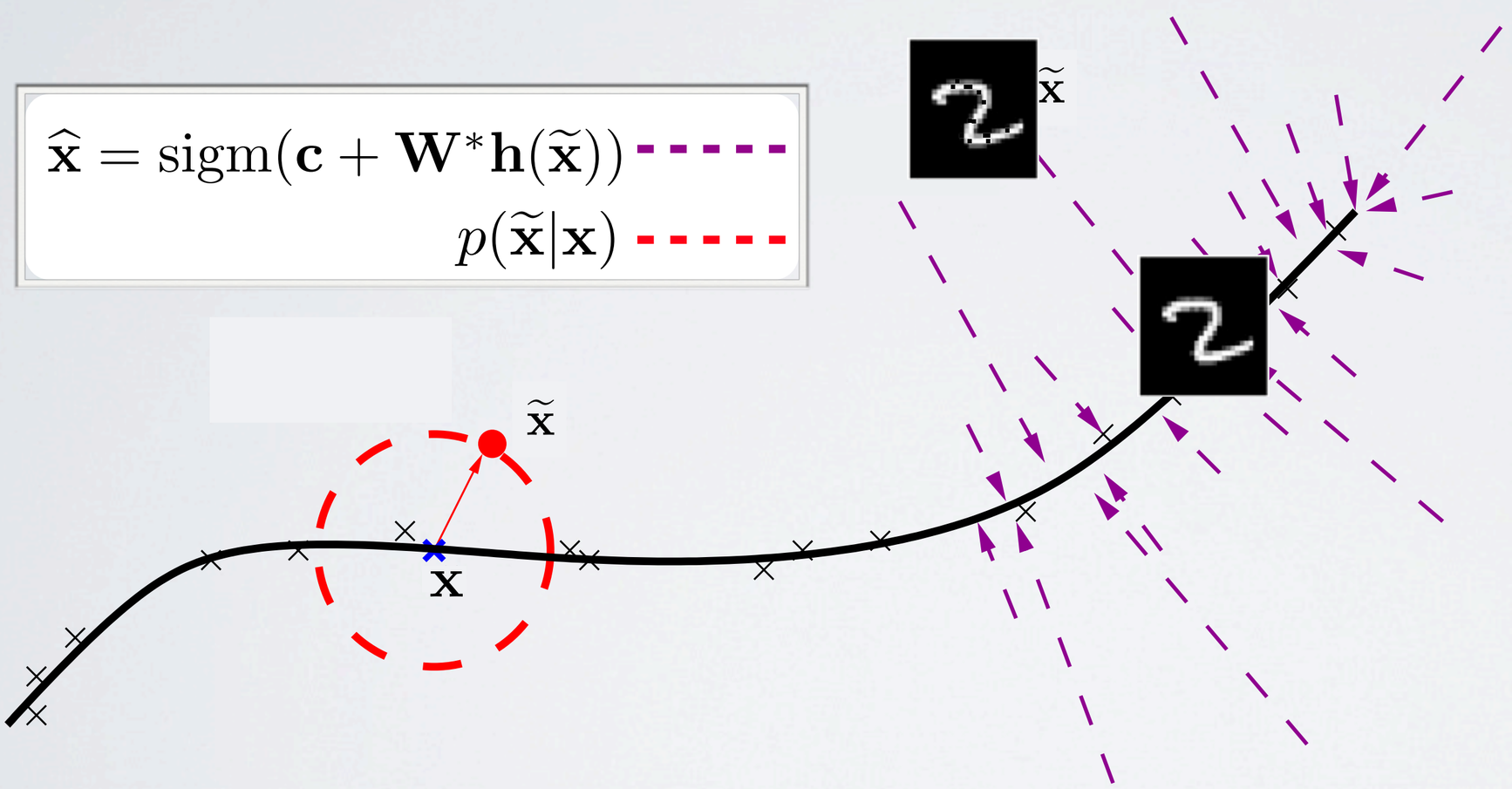
$$p(\widetilde{\mathbf{x}}|\mathbf{x})$$



DENOISING AUTOENCODER

$$\widehat{\mathbf{x}} = \mathrm{sigm}(\mathbf{c} + \mathbf{W}^* \mathbf{h}(\widetilde{\mathbf{x}}))$$

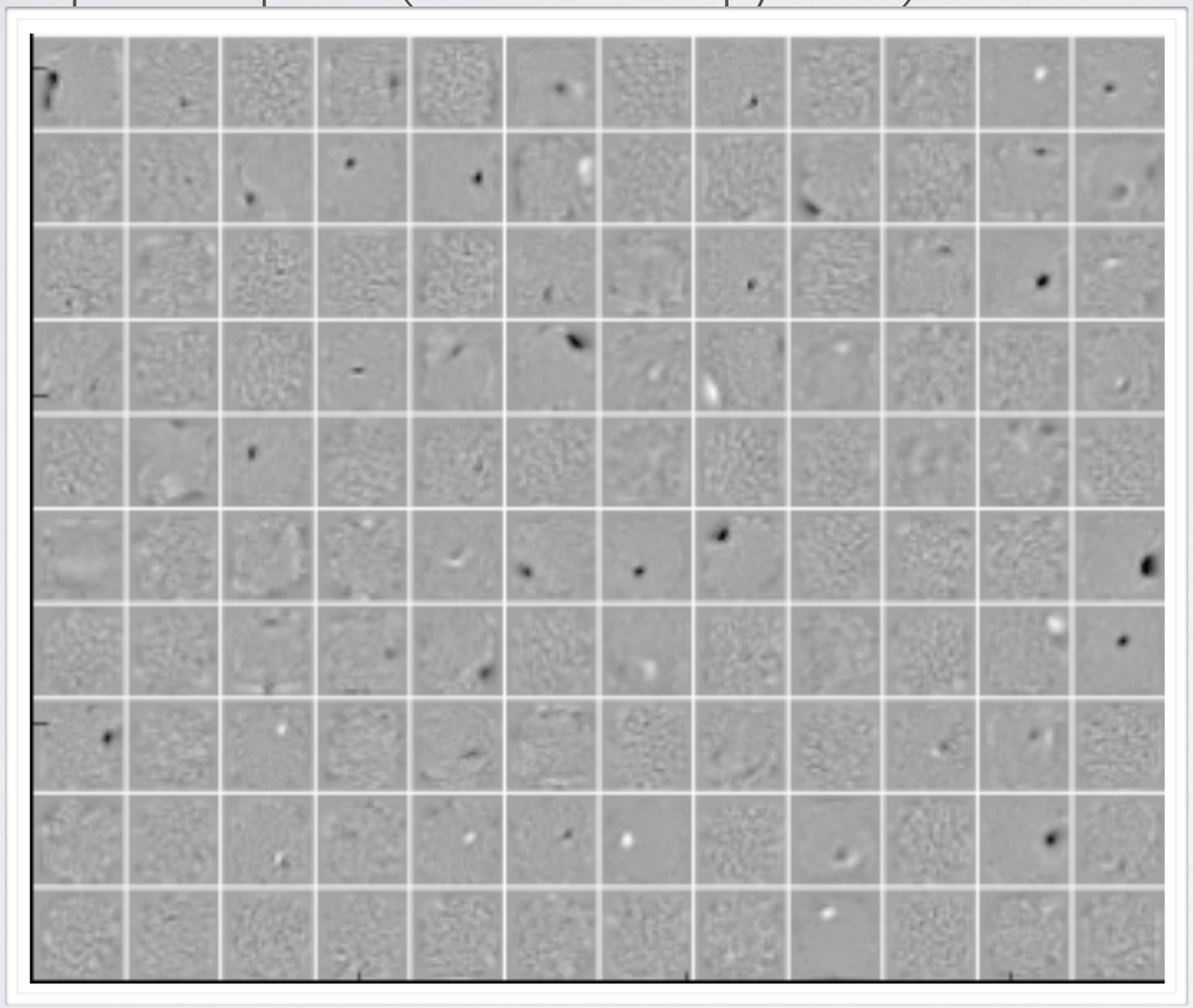
$$p(\widetilde{\mathbf{x}}|\mathbf{x})$$



FILTERS (DENOISING AUTOENCODER)

(Vincent, Larochelle, Bengio and Manzagol, ICML 2008)

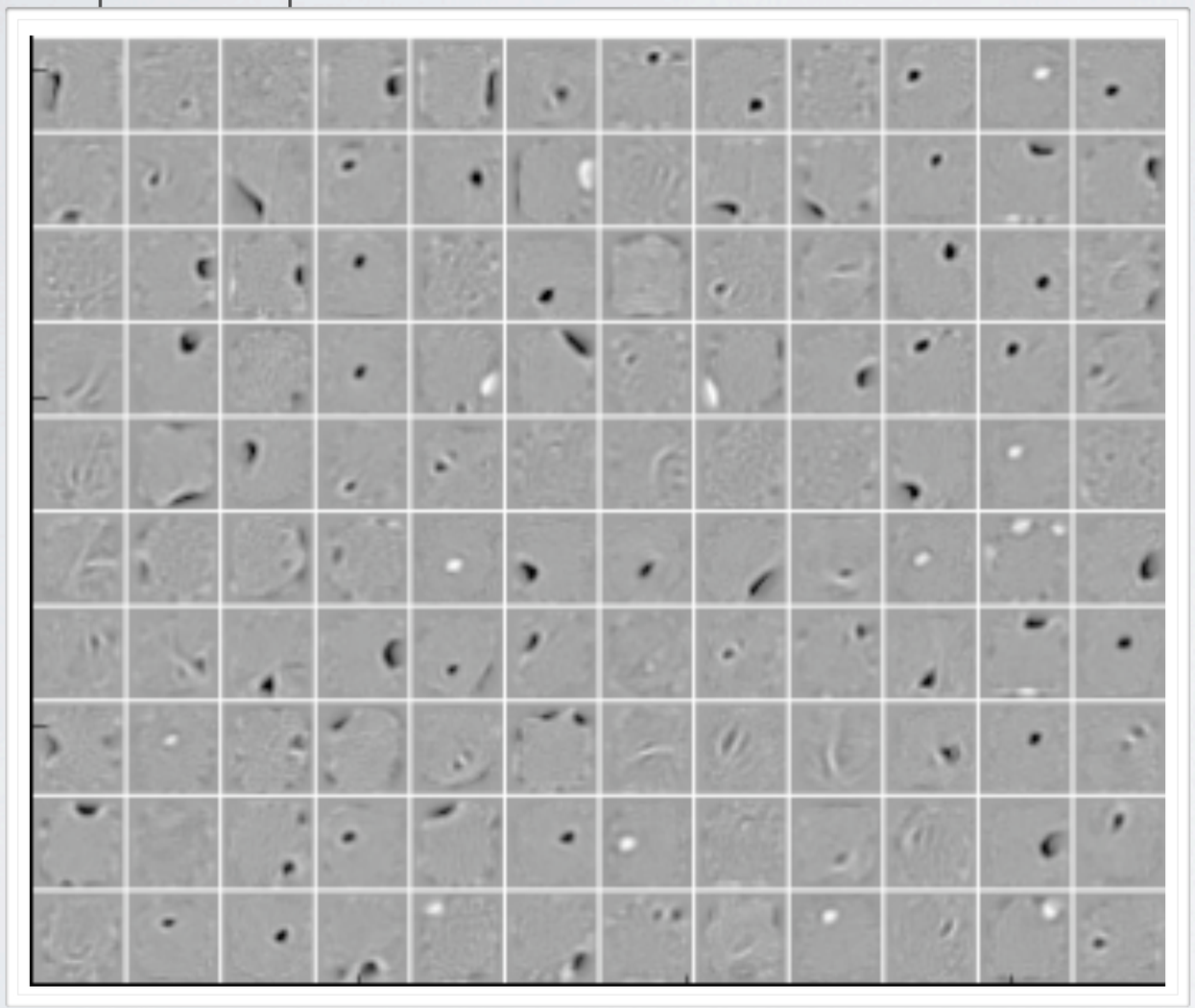
- No corrupted inputs (cross-entropy loss)



FILTERS (DENOISING AUTOENCODER)

(Vincent, Larochelle, Bengio and Manzagol, ICML 2008)

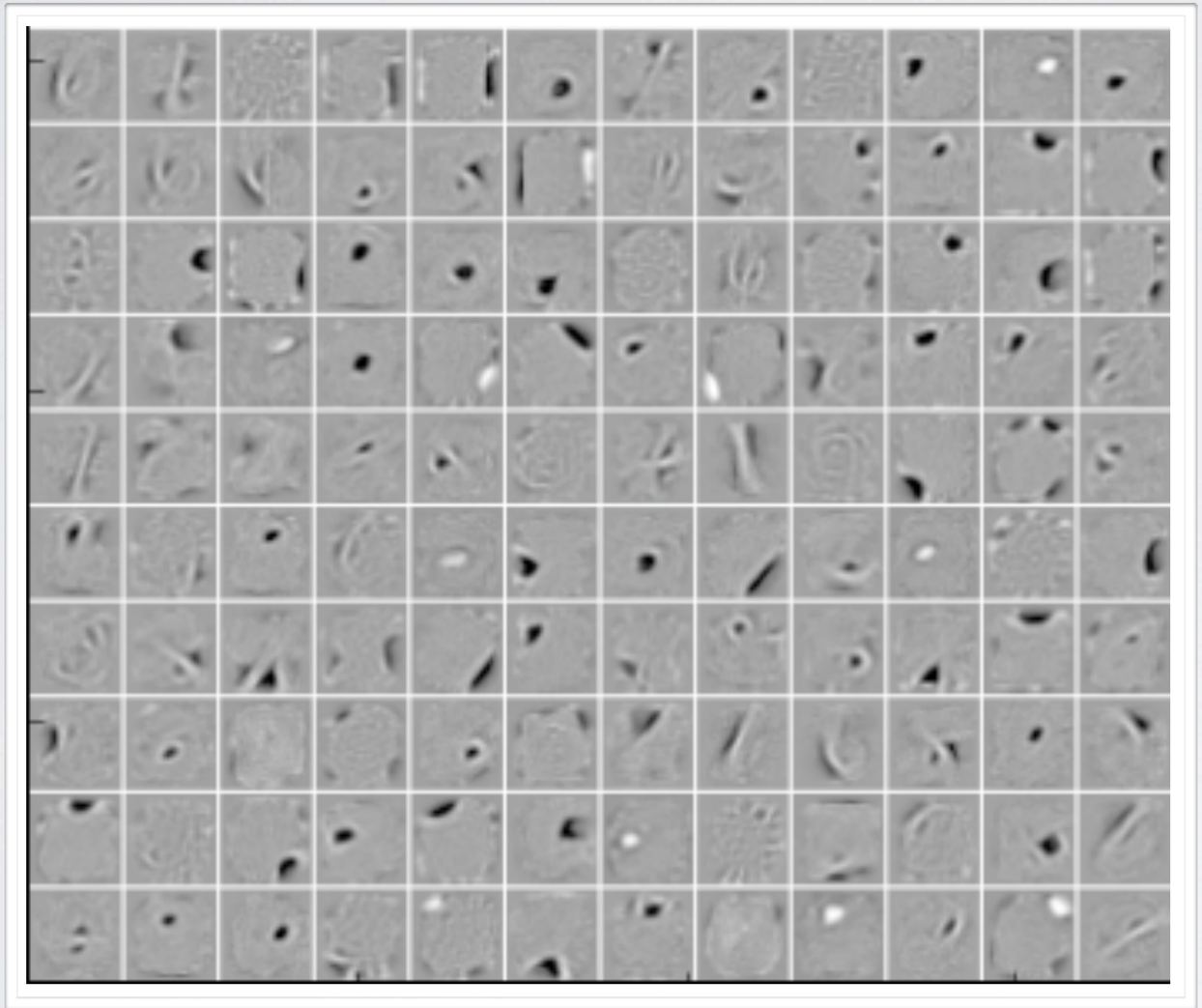
- 25% corrupted inputs



FILTERS (DENOISING AUTOENCODER)

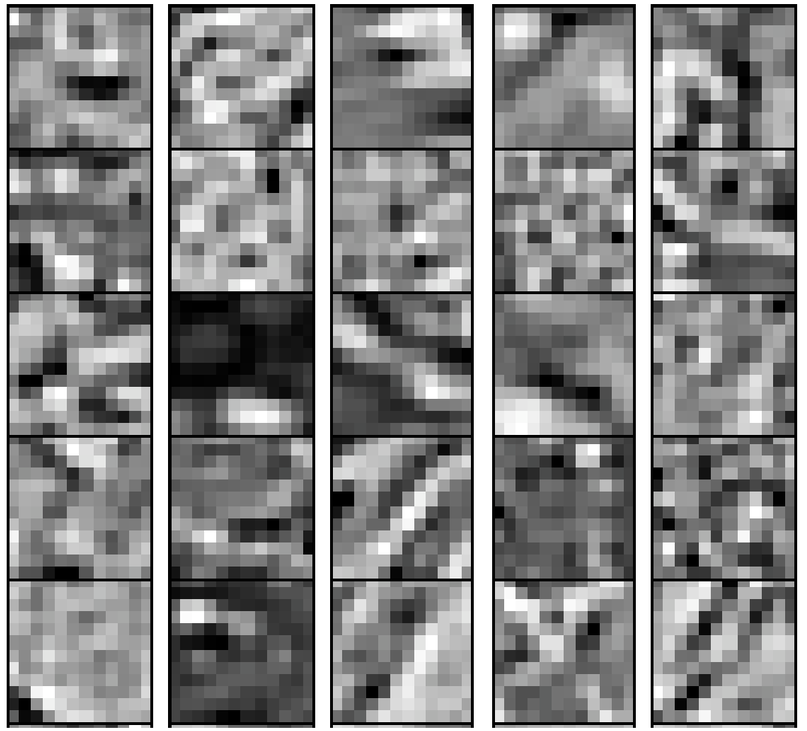
(Vincent, Larochelle, Bengio and Manzagol, ICML 2008)

- 50% corrupted inputs

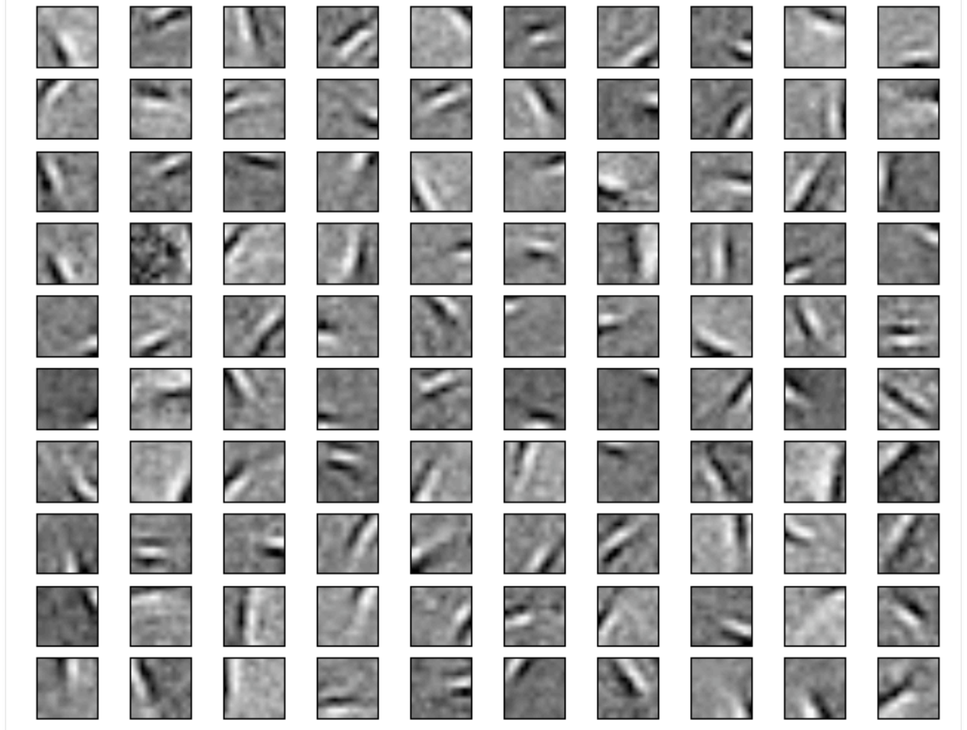


SQUARED ERROR LOSS

- Training on natural image patches, with squared-difference loss
  - PCA is not the best solution



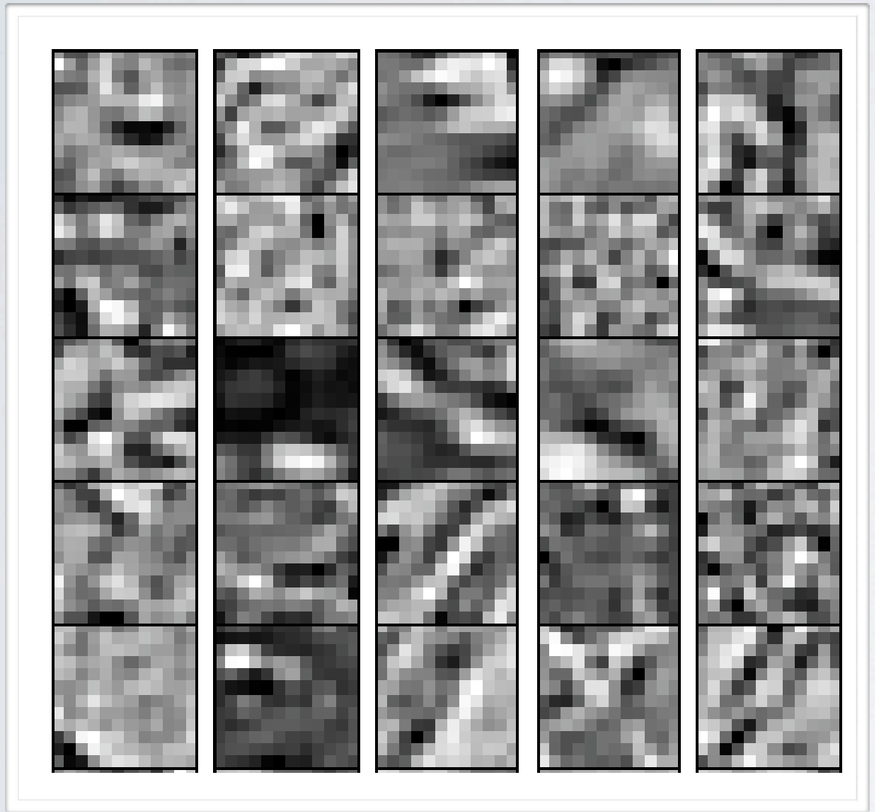
Data



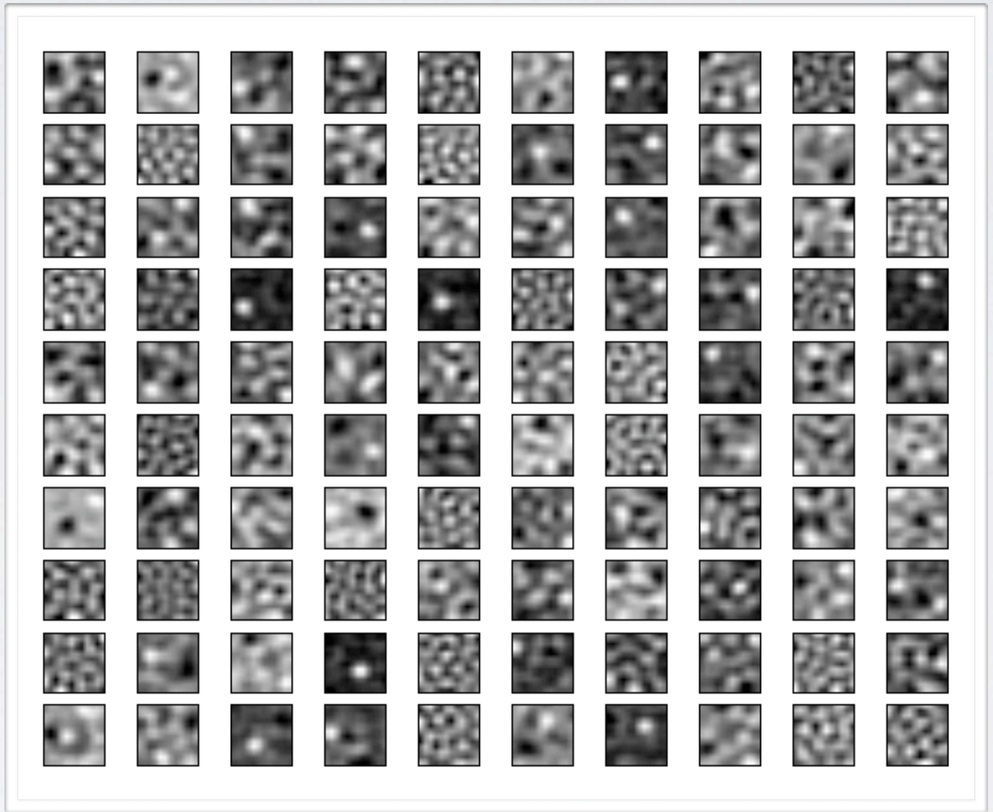
Filters

SQUARED ERROR LOSS

- Training on natural image patches, with squared-difference loss
  - Not equivalent to weight decay



Data



Filters