Lecture 18: Autoencoders
CS480/680 Intro to Machine Learning

2023-3-16

Pascal Poupart
David R. Cheriton School of Computer Science

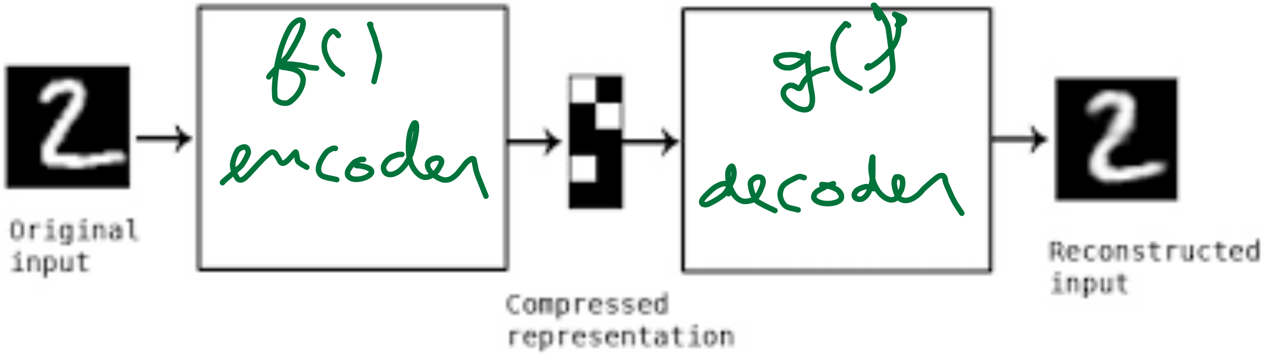
UNIVERSITY OF WATERLOO

Autoencoder

- Special type of feed forward network for
  - Compression
  - Denoising
  - Sparse representation
  - Data generation

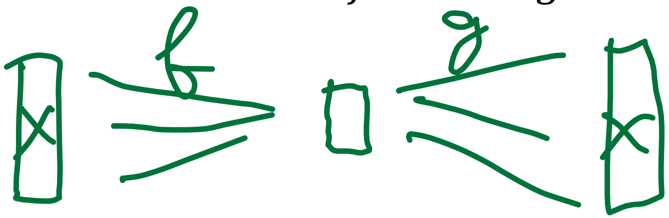
Autoencoder

- Encoder: $f(\quad)$
- Decoder: $g(\quad)$
- Autoencoder: $g\left(f(\boldsymbol{x})\right) = \boldsymbol{x}$



Linear Autoencoder

- f and g are linear
  - Matrix representations: $\boldsymbol{W}_f$ and $\boldsymbol{W}_g$
- Picture:



Linear Autoencoder

- Objective: find weights $\boldsymbol{W}_f$ and $\boldsymbol{W}_g$ that minimize reconstruction error

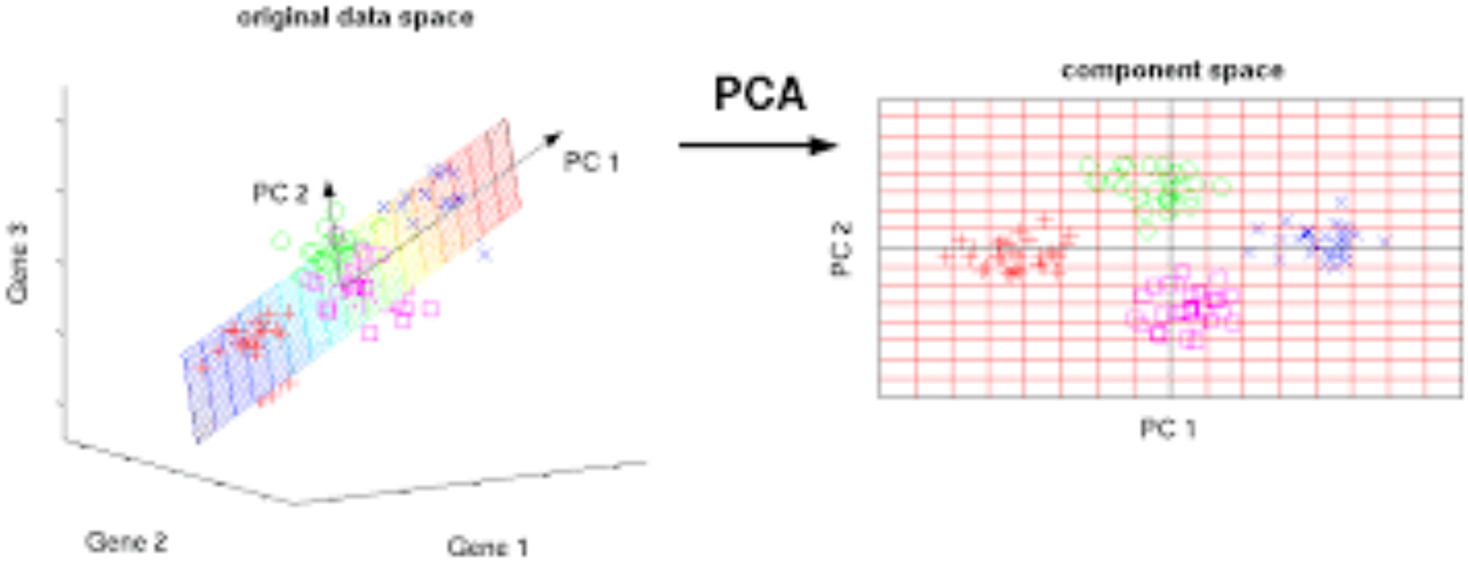
$$\min_{\boldsymbol{W}} \frac{1}{2} \sum_n \left|\left|\boldsymbol{W}_g \boldsymbol{W}_f \boldsymbol{x}_n - \boldsymbol{x}_n\right|\right|_2^2$$

- Algorithm: backpropagation
  - Gradient descent

- When using Euclidean norm (i.e., squared loss), solution is the same as principal component analysis (PCA)

Principal Component Analysis

- Hidden nodes: compressed representation



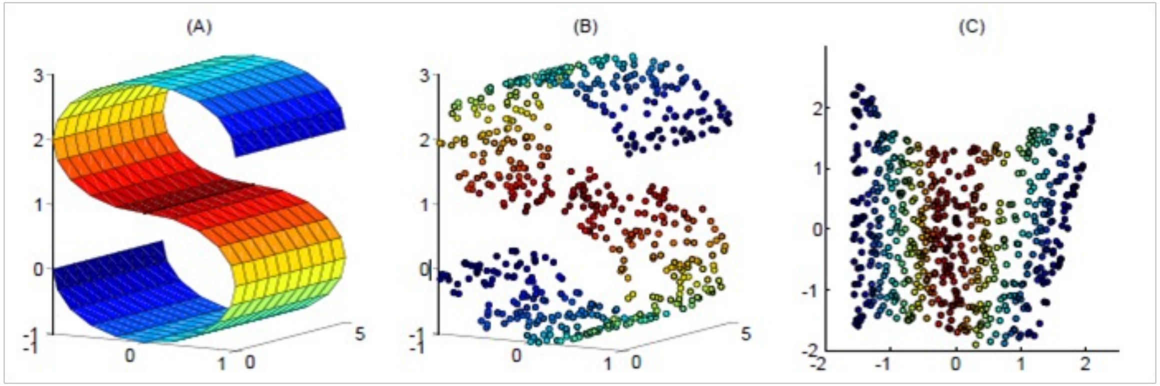
UNIVERSITY OF WATERLOO

Nonlinear Autoencoder

- f and g are non-linear functions

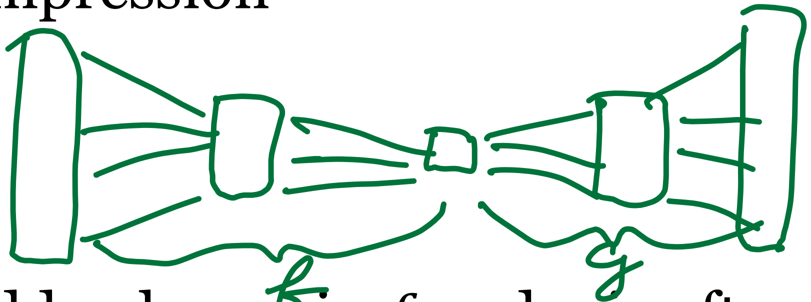
$$\min_{W} \frac{1}{2} \sum_{n} \left|\left| g(f(\boldsymbol{x}_n; \boldsymbol{W}_f); \boldsymbol{W}_g) - \boldsymbol{x}_n \right|\right|_2^2$$

- Hidden nodes: non-linear manifold

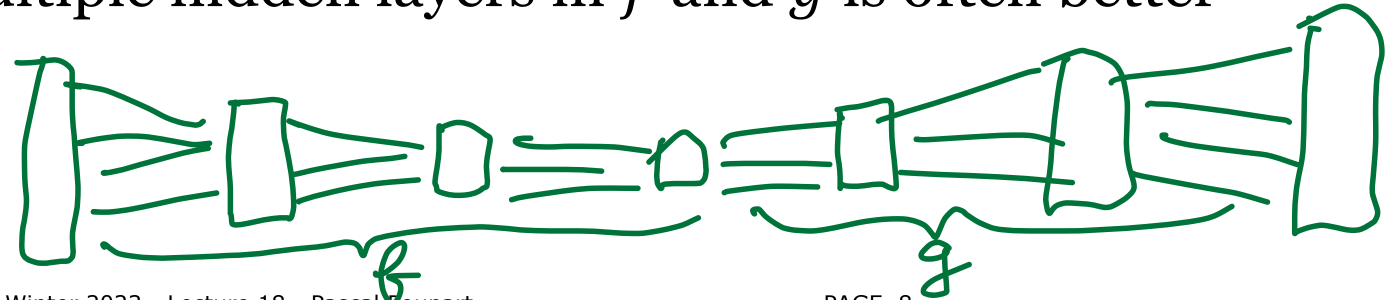


Deep Autoencoders

- f and g often consist of multiple layers
- In theory, one hidden layer in f and g is sufficient to represent any possible compression



- Multiple hidden layers in f and g is often better



UNIVERSITY OF WATERLOO

Sparse Representations

- When more hidden nodes than inputs, use regularization to constrain autoencoder
- Example: force hidden nodes to be sparse

$$\min_{\boldsymbol{W}} \frac{1}{2} \sum_n \left|\left| g\left(f\left(\boldsymbol{x}_n; \boldsymbol{W}_f\right); \boldsymbol{W}_g\right) - \boldsymbol{x}_n \right|\right|_2^2 + \mathrm{c}\, \underbrace{nnz\left(f\left(\boldsymbol{x}_n; \boldsymbol{W}_f\right)\right)}_{\textbf{Sparse hidden nodes}}$$

where $nnz\left(f\left(\boldsymbol{x}_n; \boldsymbol{W}_f\right)\right)$ is the number of non-zero entries in the vector produced by f.

- Approximate objective: L1 regularization

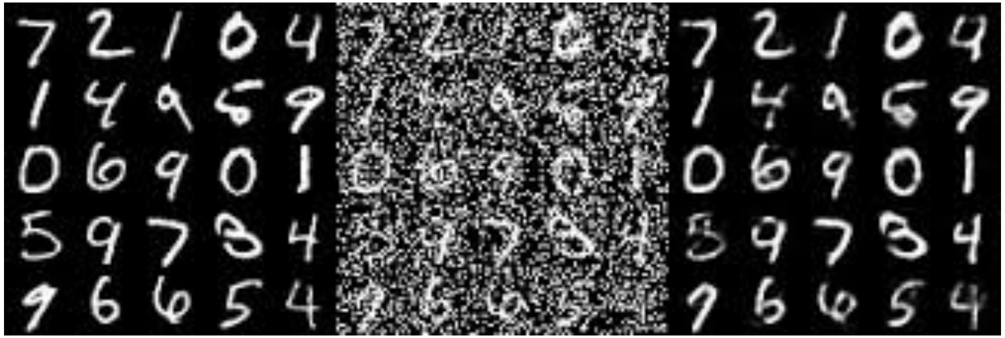
$$\min_{\boldsymbol{W}} \frac{1}{2} \sum_n \left|\left| g\left(f\left(\boldsymbol{x}_n; \boldsymbol{W}_f\right); \boldsymbol{W}_g\right) - \boldsymbol{x}_n \right|\right|_2^2 + \mathrm{c} \left|\left| f\left(\boldsymbol{x}_n; \boldsymbol{W}_f\right) \right|\right|_1$$

Denoising Autoencoder

- Consider noisy version $\tilde{\boldsymbol{x}}$ of the input $\boldsymbol{x}$
- Data denoising

$$\min_{\boldsymbol{W}} \frac{1}{2} \sum_n \left|\left| g(f(\tilde{\boldsymbol{x}}_n; \boldsymbol{W}_f); \boldsymbol{W}_g) - \boldsymbol{x}_n \right|\right|_2^2 + c \left|\left| f(\tilde{\boldsymbol{x}}_n; \boldsymbol{W}_f) \right|\right|_1$$

$\boldsymbol{x}$ $\tilde{\boldsymbol{x}}$ $g(f(\tilde{\boldsymbol{x}}))$



original perturbed reconstructed

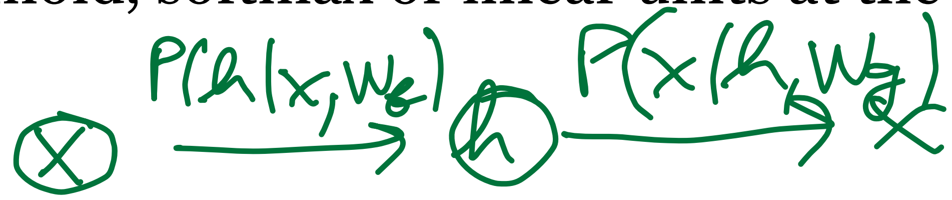
Probabilistic Autoencoder

- Let f and g represent conditional distributions

$$f: \Pr(\boldsymbol{h}|\boldsymbol{x};\boldsymbol{W}_f) \quad \text{and} \quad g: \Pr(\boldsymbol{x}|\boldsymbol{h};\boldsymbol{W}_g)$$

by using sigmoid, softmax or linear units at the hidden and output layers

- Picture



Sigmoid : Bernoulli $P(h) : h \in \{0,1\}$

Softmax : Categorical $P(h) : h \in$ one hot vectors

Linear : Gaussian $P(h|\mu,\sigma^2)$ → variance; mean

Generative Model

- Sample $\boldsymbol{h}$ from some distribution $\Pr(\boldsymbol{h})$
- Sample $\boldsymbol{x}$ from decoder $\Pr(\boldsymbol{x}|\boldsymbol{h};\boldsymbol{W}_g)$