Autoencoder

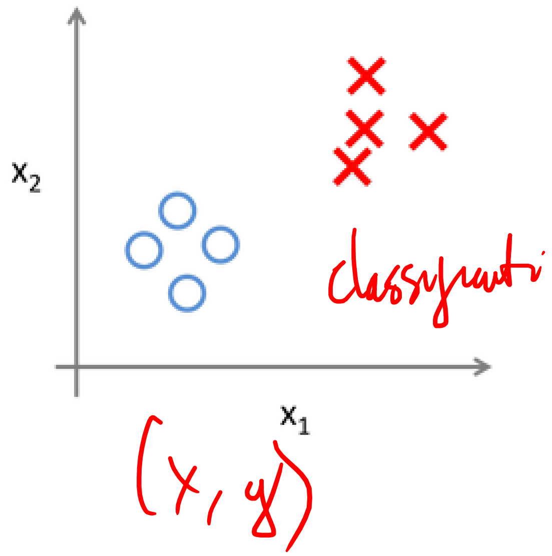
Tuan Nguyen - AI4E

Outline

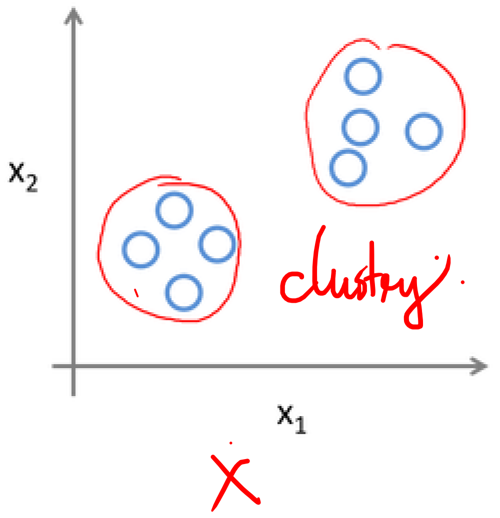
- Unsupervised Learning (Introduction)
- Autoencoder (AE)
- Autoencoder application
- Convolutional AE
- Denoising AE

Supervised vs Unsupervised

Supervised Learning



Unsupervised Learning



Supervised Learning

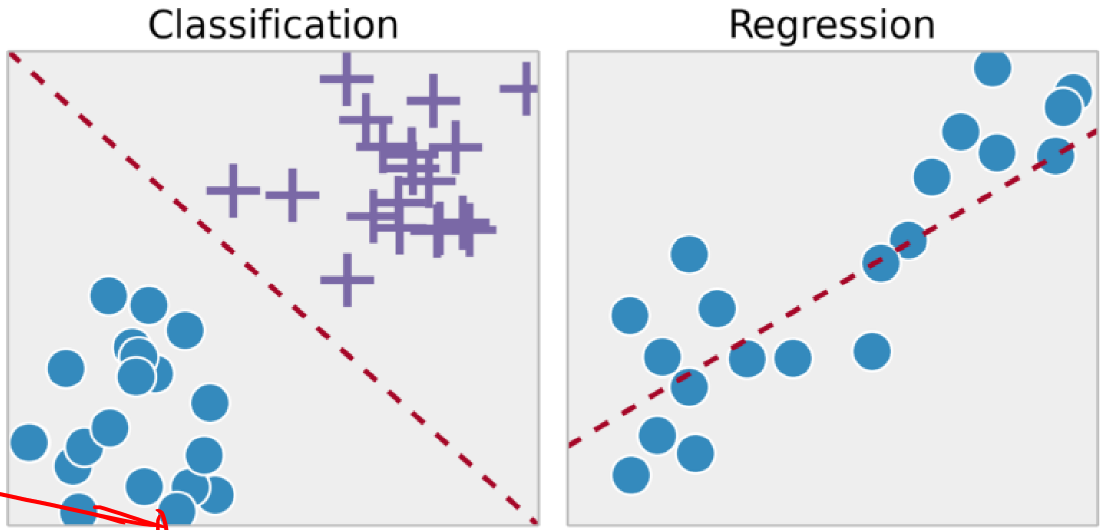
- Data: (X,Y)
- Goal: Learn a Mapping Function f where:

$$f(X) = Y$$

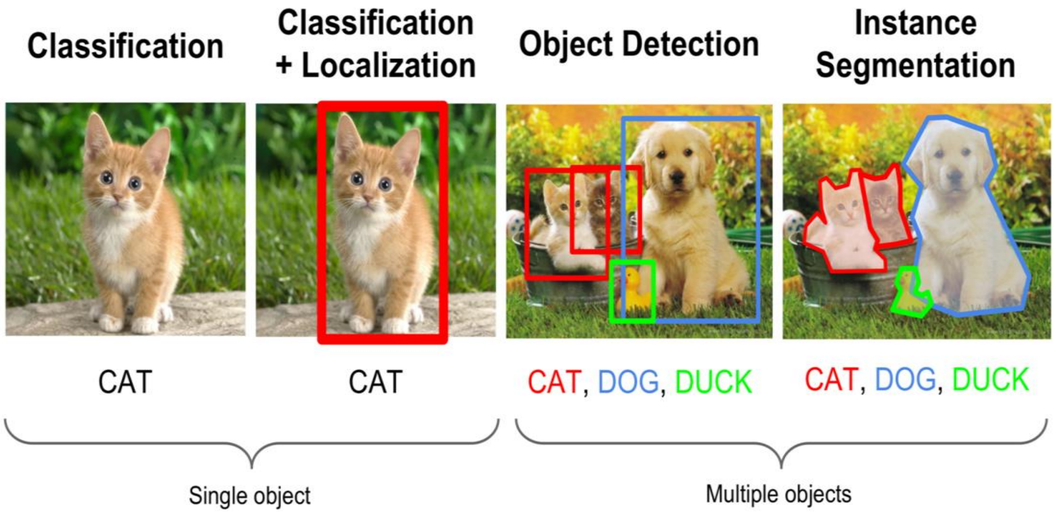
$f(x) = ax + b.$

$f(x) = f_1(f_2(\ldots f_n(x)))$

$f(x) = \text{conv}(\text{pool} \ldots (f(f\ldots(x))))$



Supervised Learning



Label data?

01

What happens when our labels are noisy?

- Missing values.
- Labeled incorrectly.

02

What happens where we don’t have labels for training **at all**?

Unsupervised Learning

Up until now we have encountered in this course mostly **Supervised Learning** problems and algorithms.

Let's talk about **Unsupervised Learning**

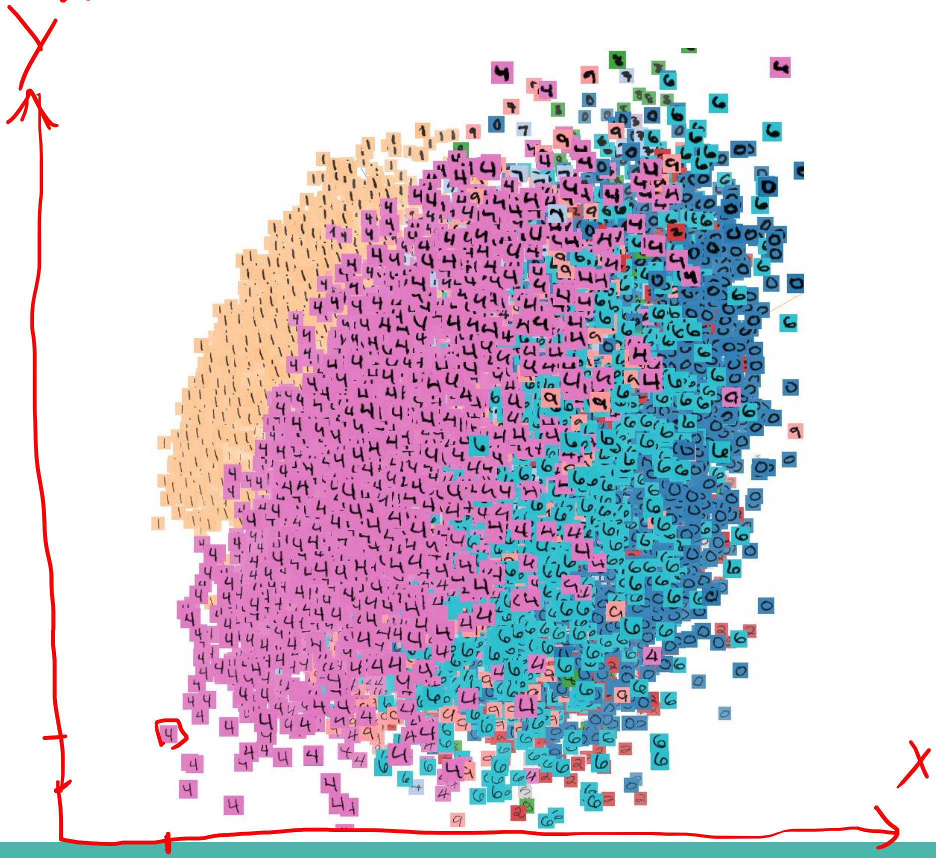
Unsupervised Learning

: unsupervsid..

Unsupervised Learning

Data: X no label

Goal: Learn the structure of the data learn correlations between features



Unsupervised Learning

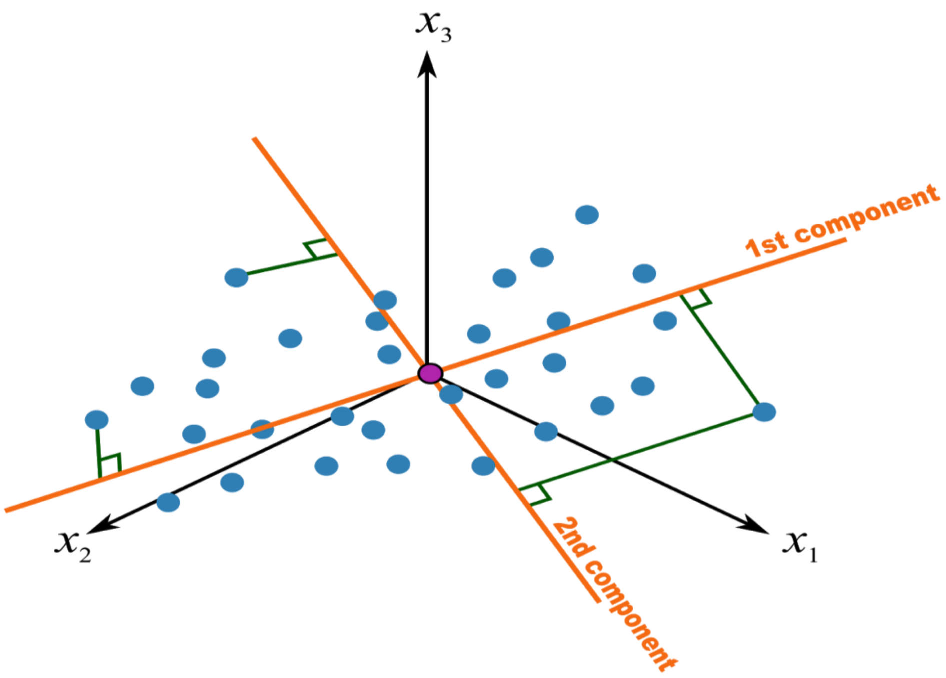
Examples: Clustering, **Compression**, **Feature & Representation learning, Dimensionality reduction**, Generative models ,etc.

GAN(s).

VAE(s)

PCA – Principal Component analysis

- Statistical approach for data compression and visualization
- Invented by Karl Pearson in 1901
- Weakness: linear components only.



Autoencoder

(x, y)

$x \xrightarrow{f} \hat{y},\ y \rightarrow \text{Loss}$

$x \xrightarrow{f} z\ (\mathbb{R}^2) \xrightarrow{g} \hat{x}$

- The autoencoder idea was a part of NN history for decades (LeCun et al, 1987).
- Traditionally an autoencoder is used for dimensionality reduction and feature learning.
- Recently, the connection between autoencoders and latent space modeling has brought autoencoders to the front of generative modeling.

Simple Idea

- Given data x (no labels) we would like to learn the functions f (encoder) and g (decoder) where:

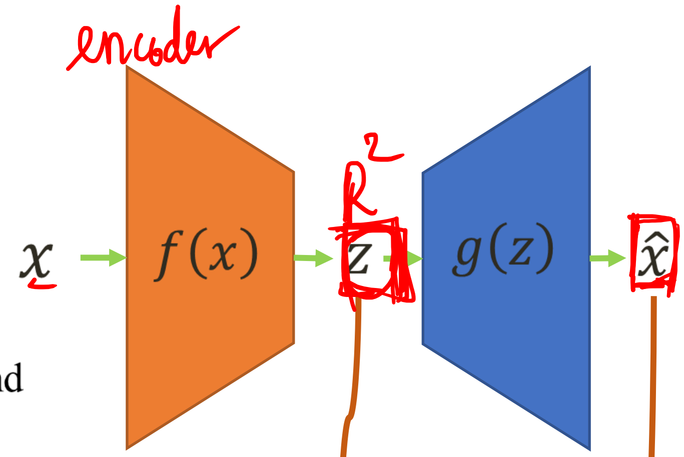
$$f(x) = s(wx + b) = z$$

and

$$g(z) = s(w'z + b') = \hat{x}$$

$$\text{s.t } h(x) = g(f(x)) = \hat{x}$$

where h is an **approximation** of the identity function.



(z is some **latent** representation or **code** and s is a non-linearity such as the sigmoid)

($\hat{x}$ is x's reconstruction)

Training Autoencoder

Using **Gradient Descent** we can simply train the model as any other FC NN with:

- Traditionally with *squared error* loss function

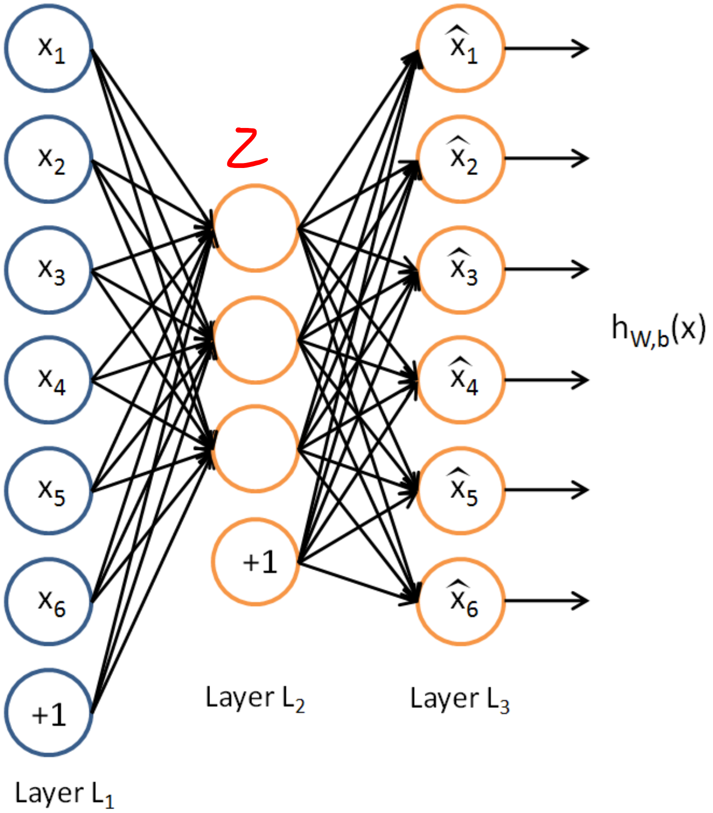
$$L(x, \hat{x}) = \boxed{\|x - \hat{x}\|^2}$$

$0 \to 1$.

- If our input is interpreted as bit vectors or vectors of bit probabilities the *cross entropy* can be used

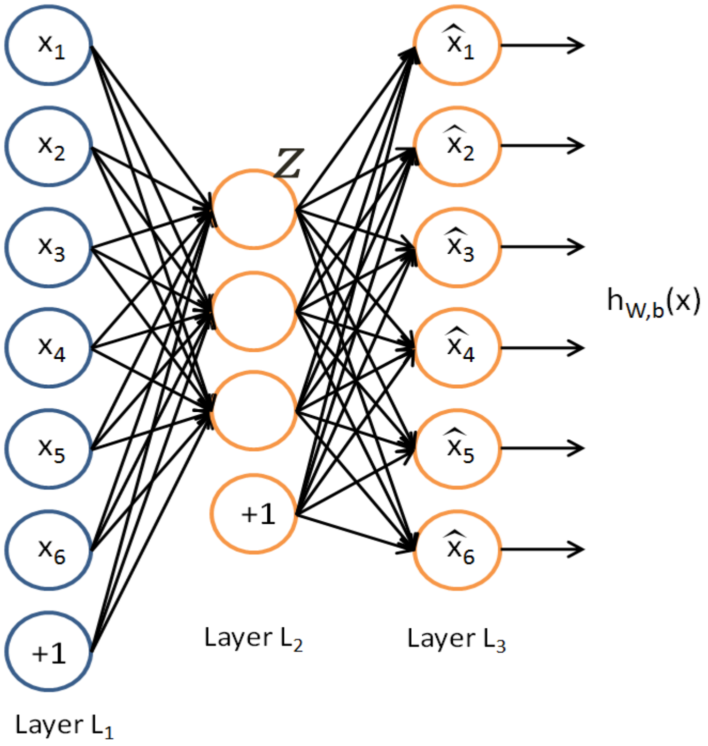
$$H(p, q) = -\sum_{x} p(x) \log q(x)$$

Traditional Autoencoder

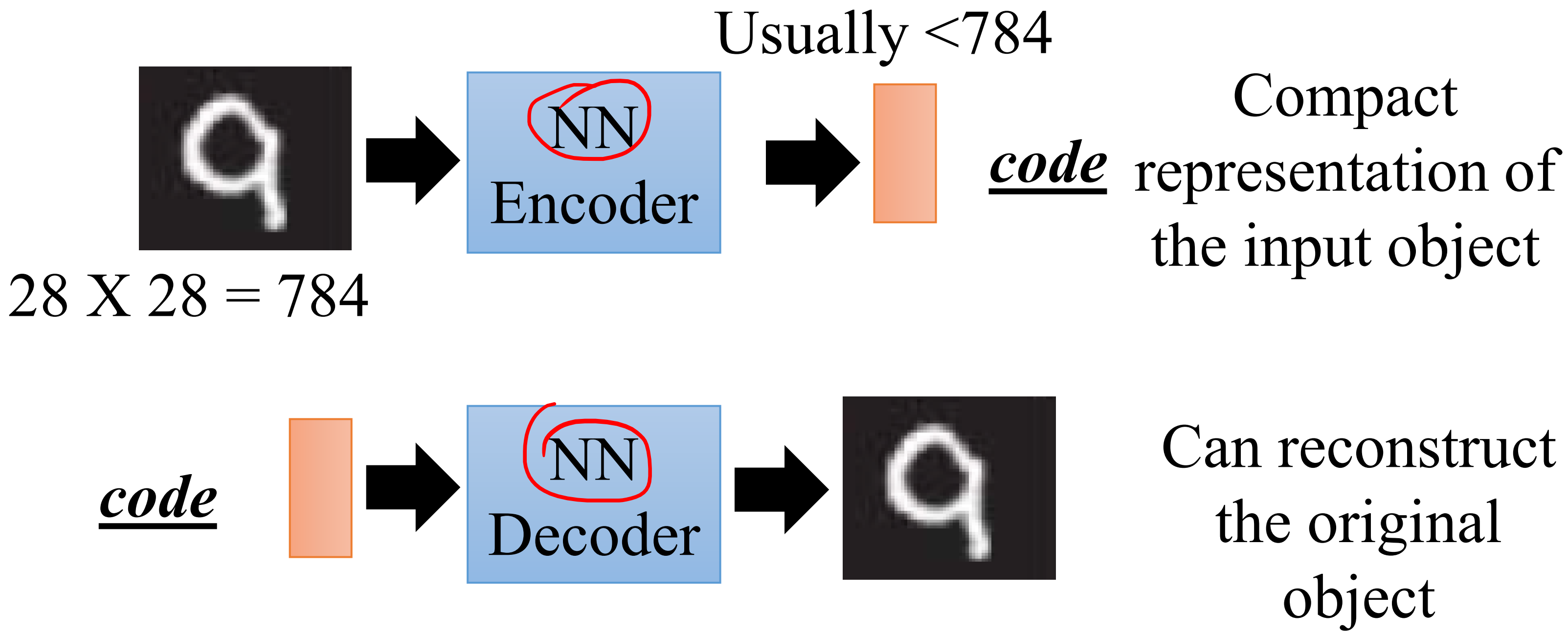


Traditional Autoencoder

- Unlike the **PCA** now we can use activation functions to achieve non-linearity.
- It has been shown that an AE without activation functions achieves the **PCA** capacity.

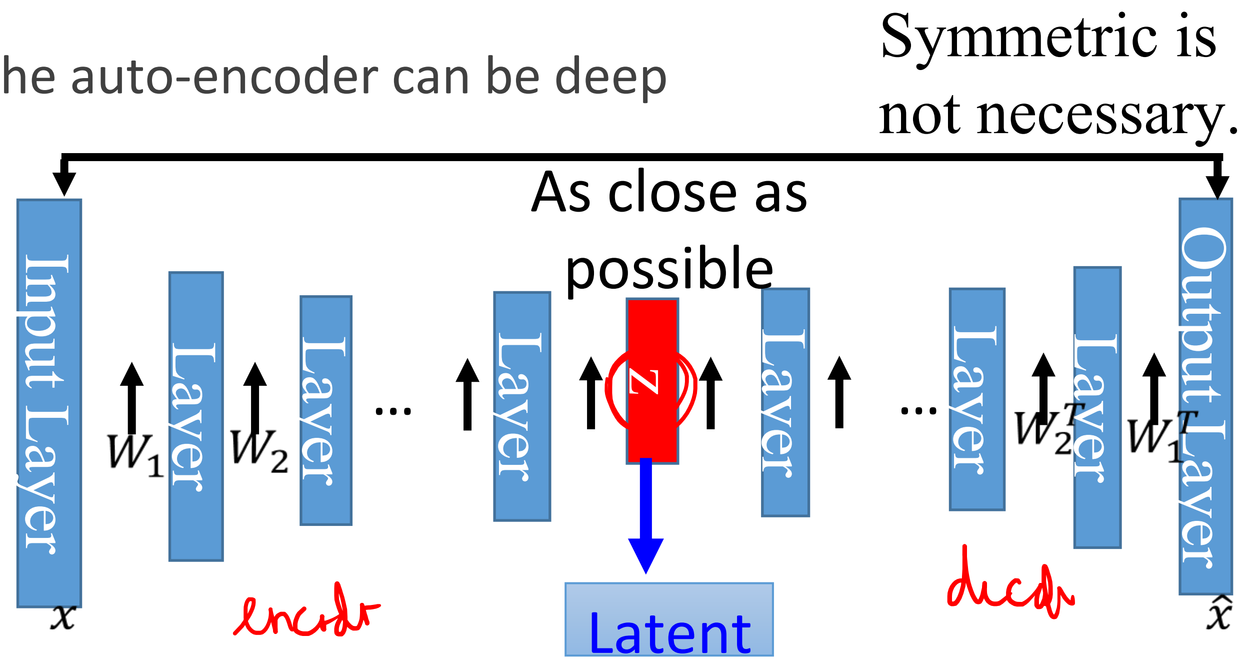


Auto-encoder



Deep Autoencoder

Of course, the auto-encoder can be deep

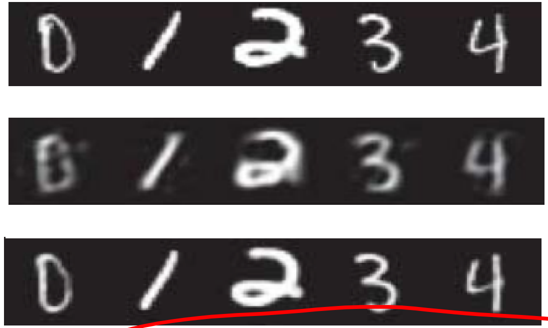


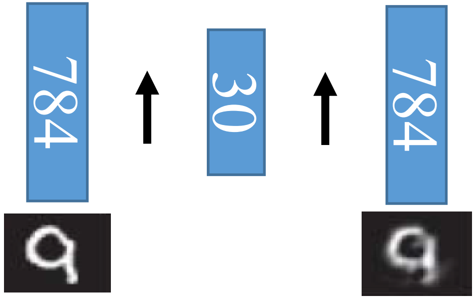
Deep Autoencoder

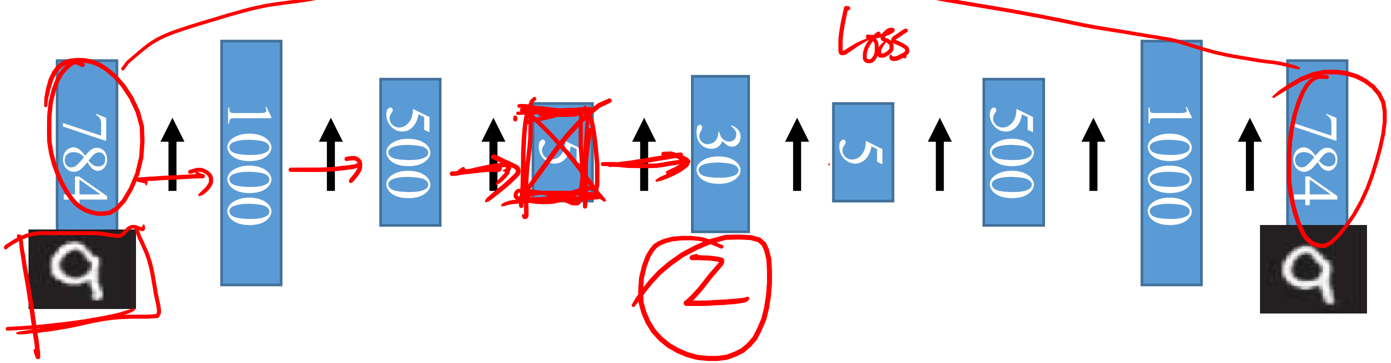
Original Image

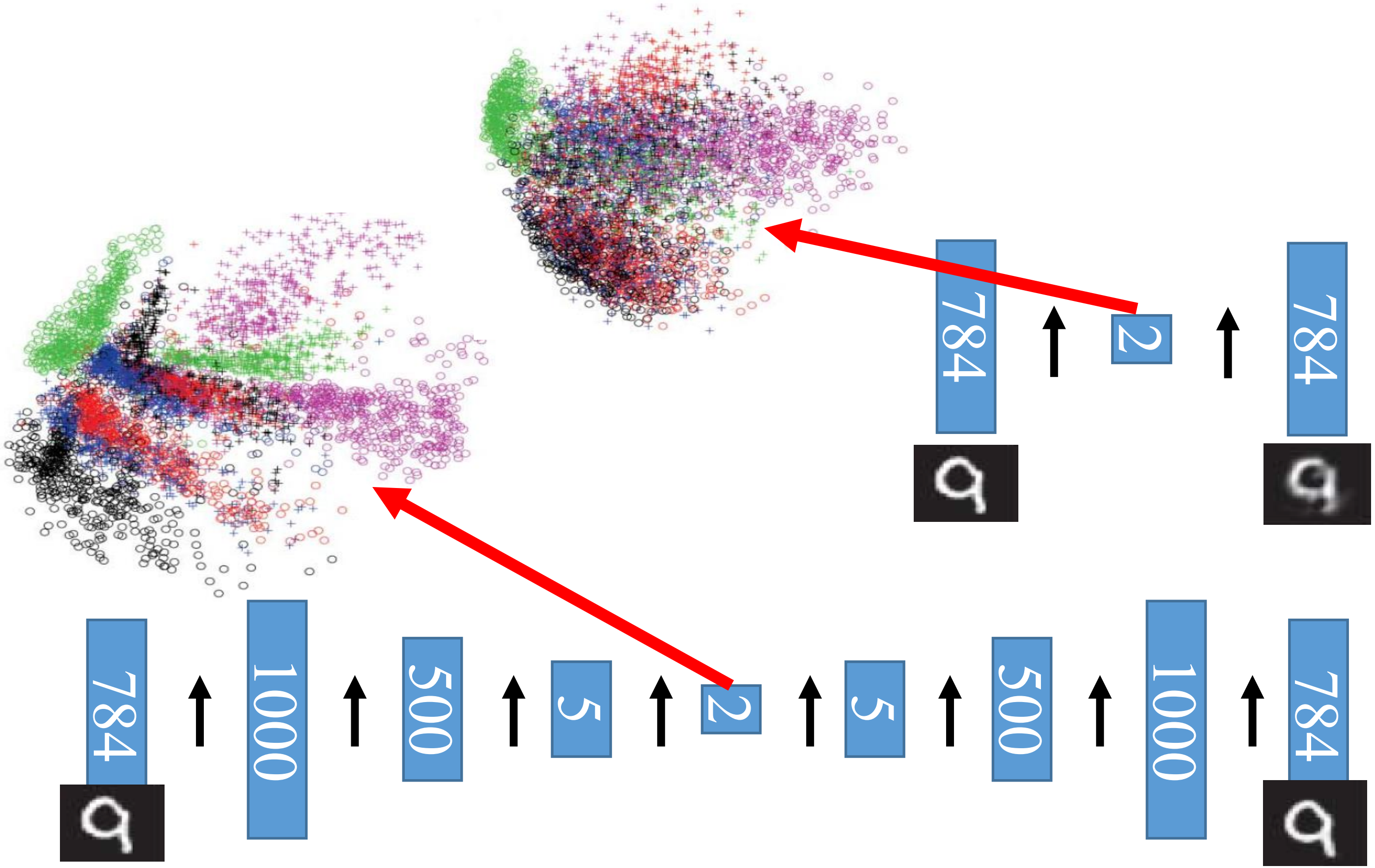
PCA

Deep Auto-encoder



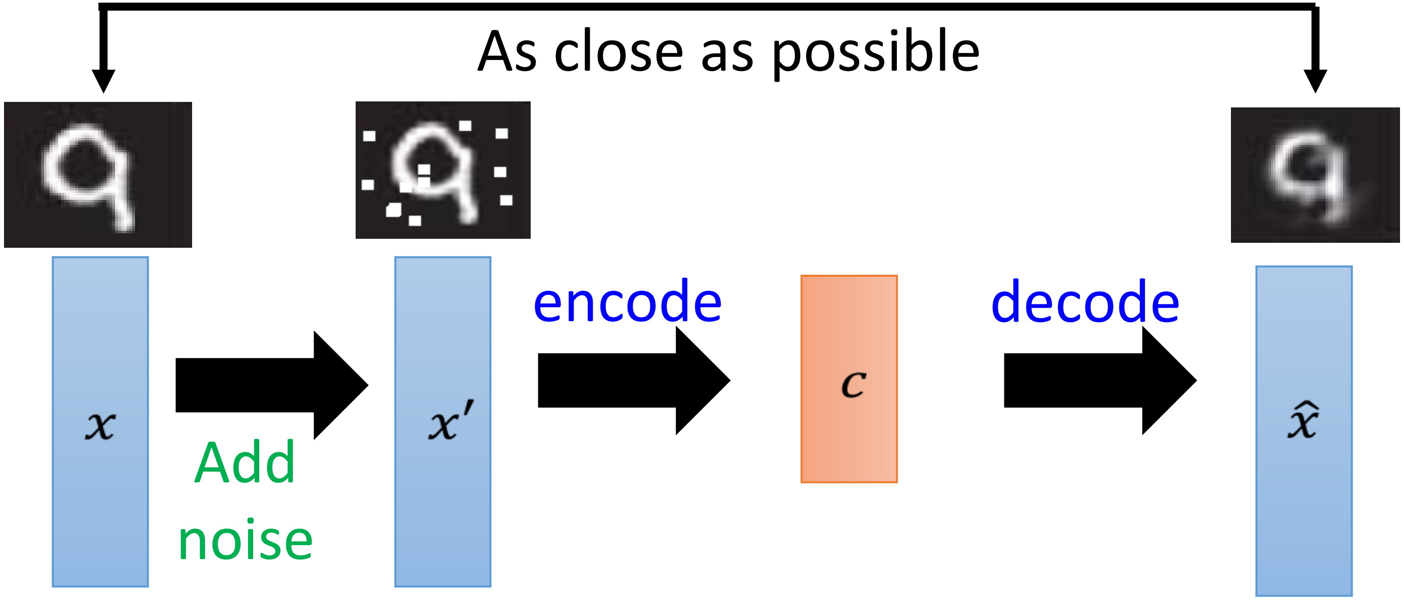






784
2
784
784
1000
500
5
2
5
500
1000
784

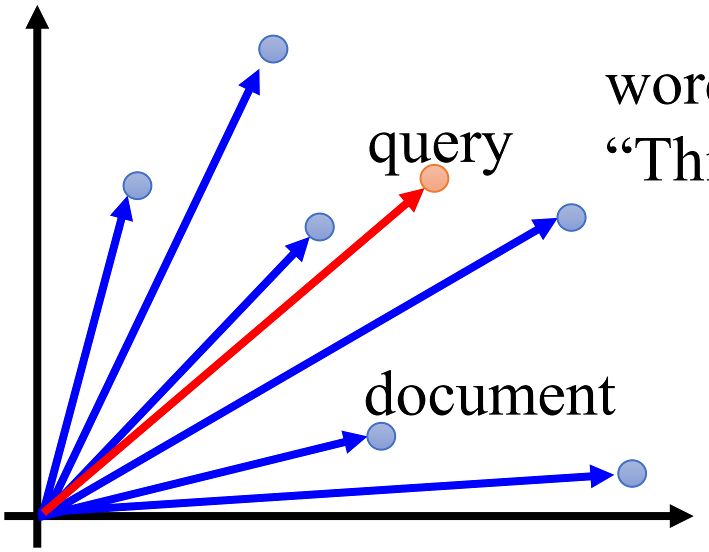
Denoise



Vincent, Pascal, et al. "Extracting and composing robust features with denoising autoencoders." *ICML*, 2008.

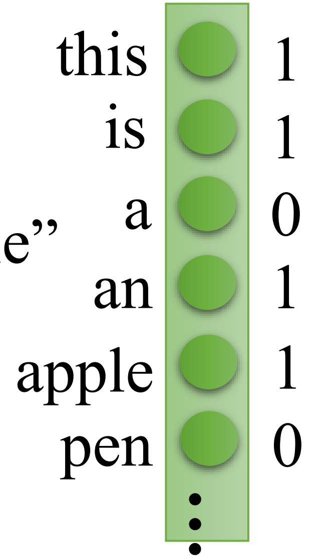
Text Retrieval

Vector Space Model



word string:
"This is an apple"

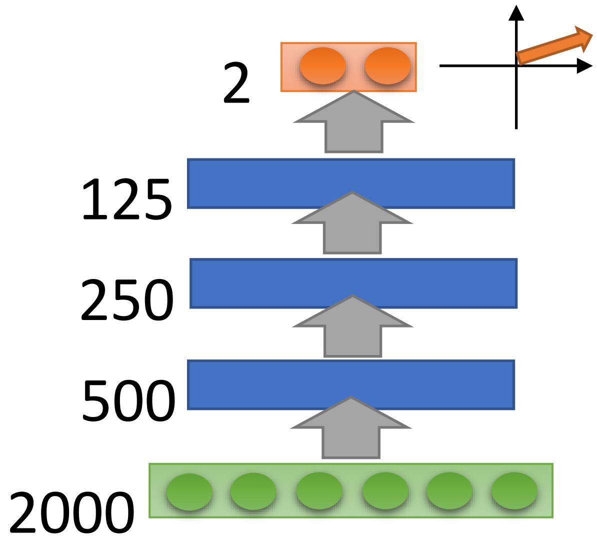
Bag-of-word



Semantics are not considered.

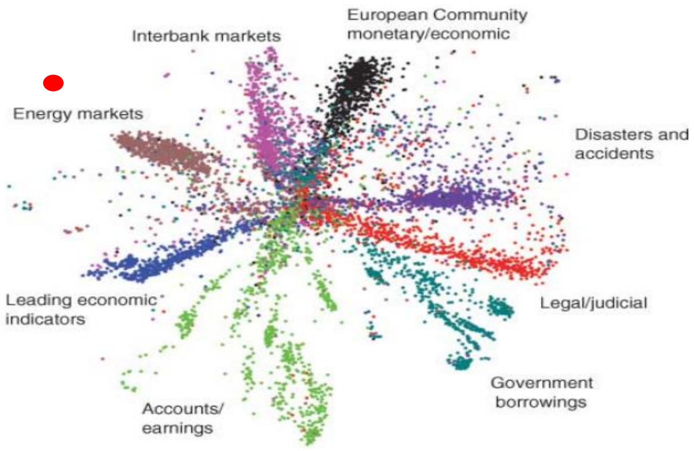
Text Retrieval

The documents talking about the same thing will have close code.



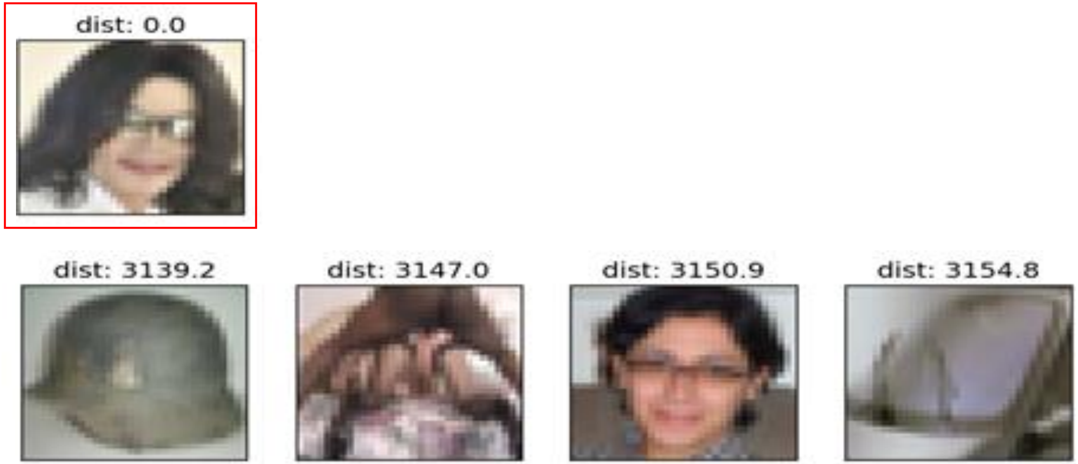
Bag-of-word
(document or query)

query



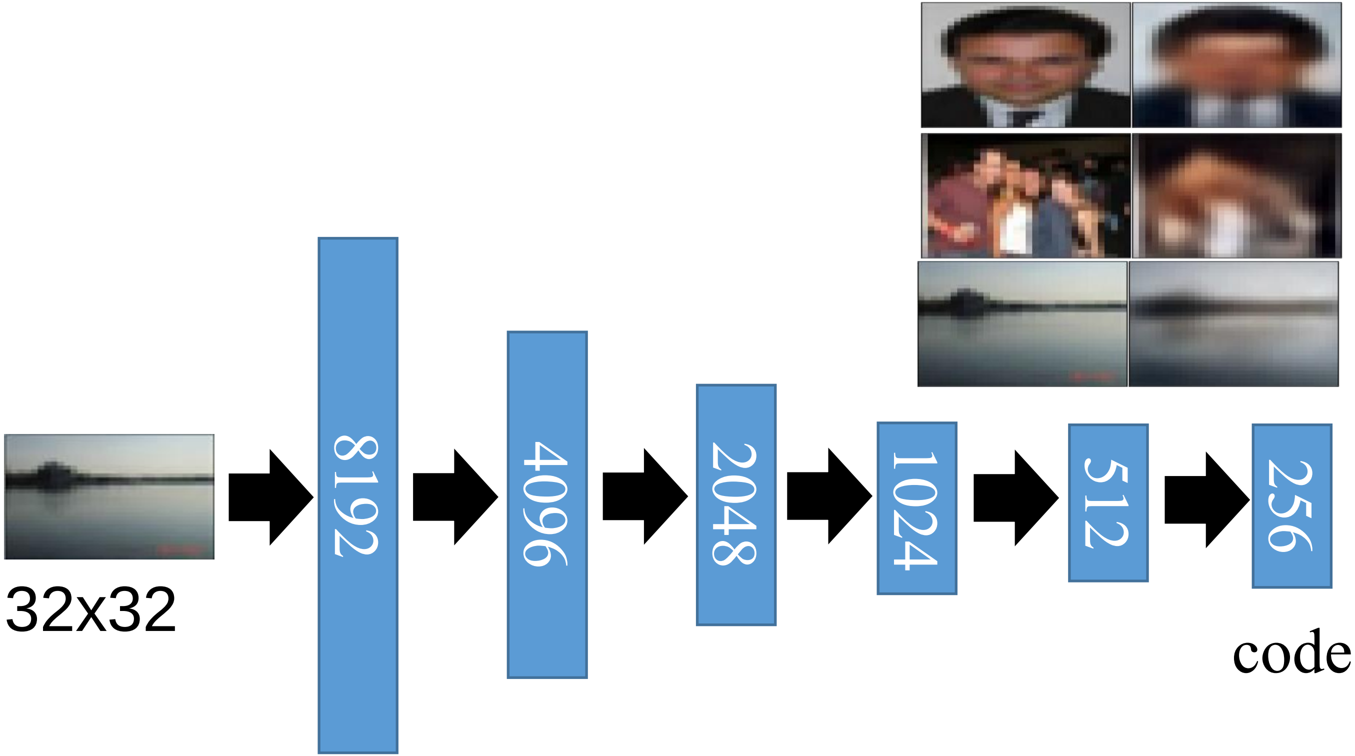
Similar image search

Retrieved using Euclidean distance in pixel intensity



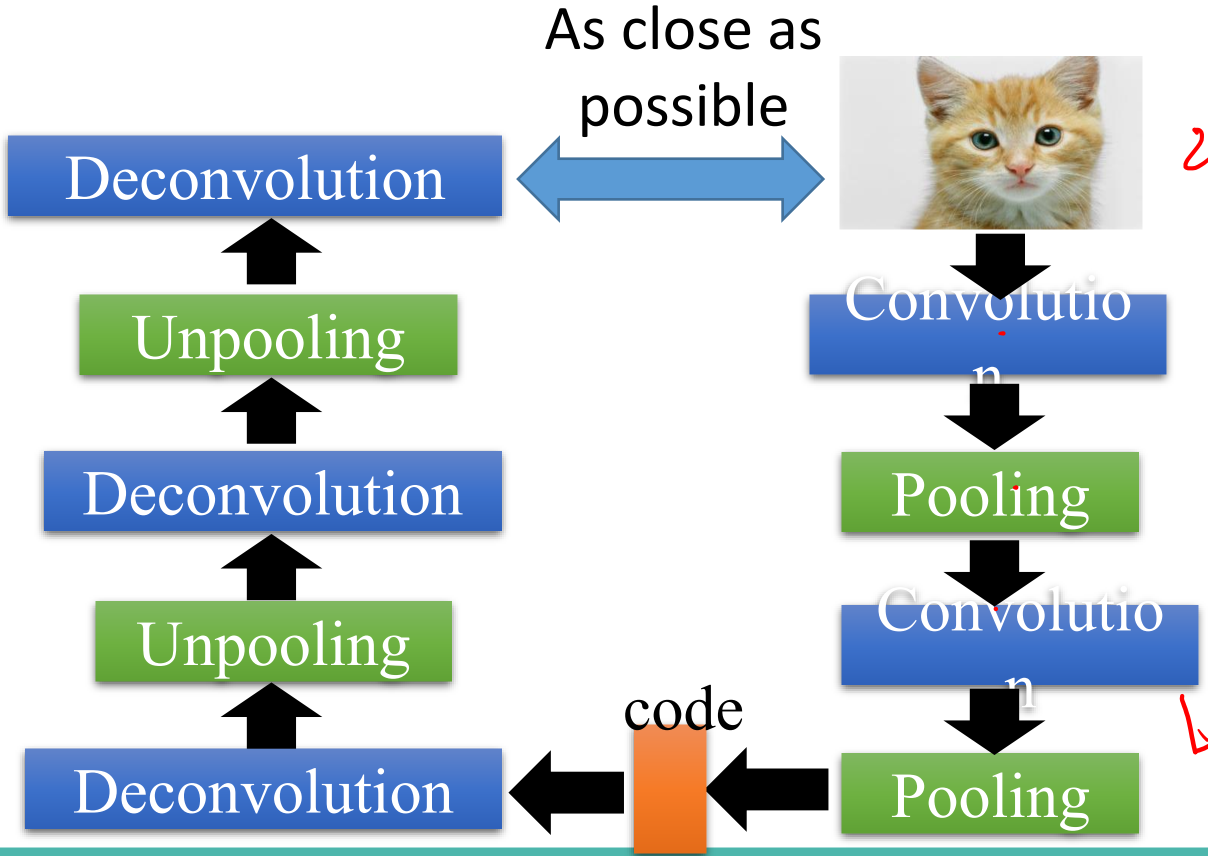
Reference: Krizhevsky, Alex, and Geoffrey E. Hinton. "Using very deep autoencoders for content-based image retrieval." *ESANN*. 2011.

Similar image search



max pooling, Conv s=2.

Auto-encoder for CNN



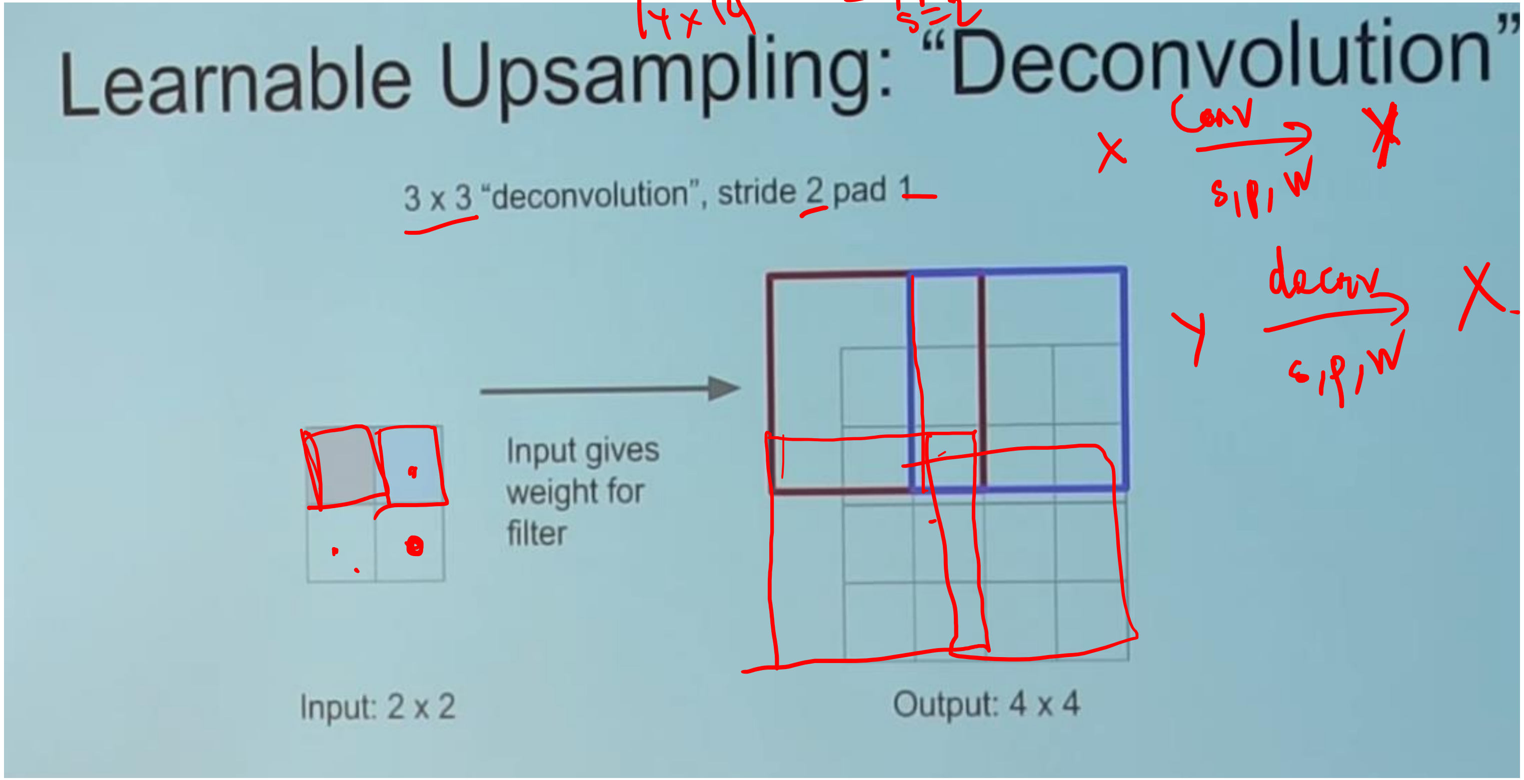
28×28

7×7.

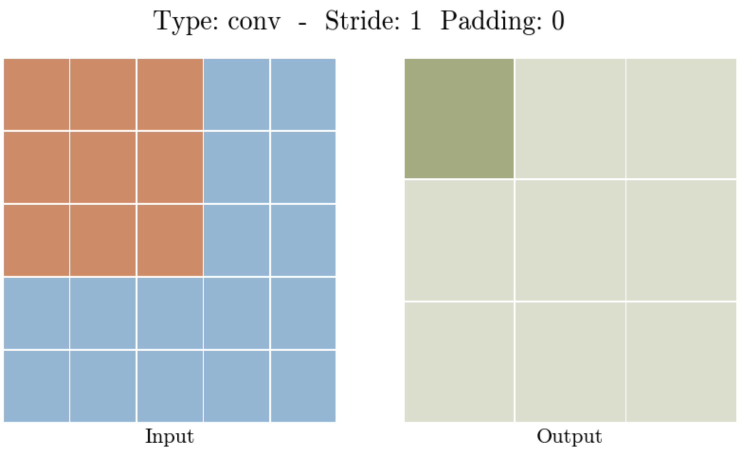
Deconvolution :

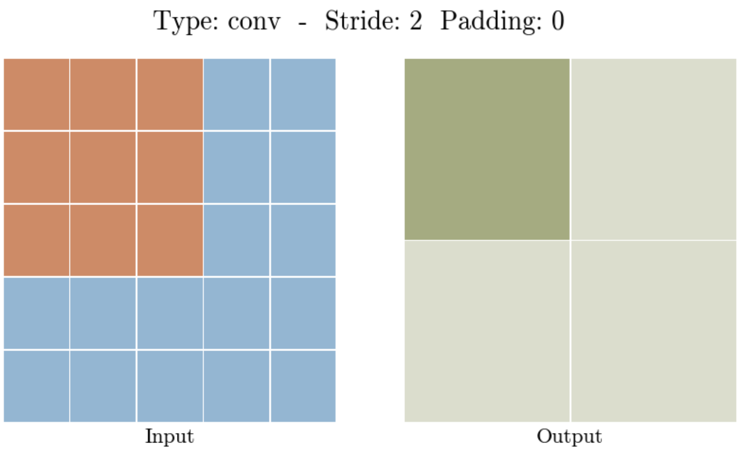
28×28 $m\times m$ $\xrightarrow[\cancel{s=2,\ p=1}\ s,\ p,\ 3\times 3]{\text{conv}}$ n. 14×14

$n\times n$ 14×14 $\xrightarrow[s,\ p,\ 3\times 3 \quad s=2]{\text{deconv}}$ $m\times m$. 28×28

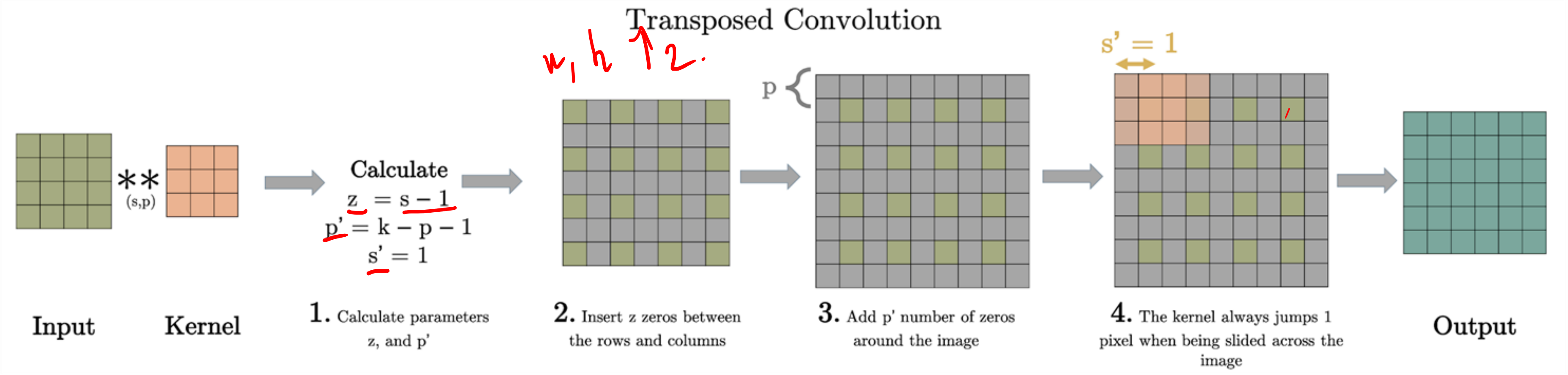


Convolution

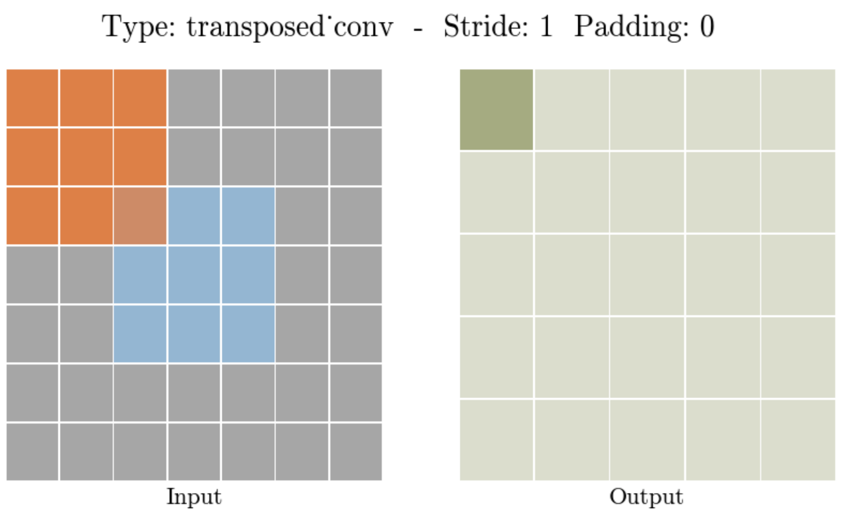


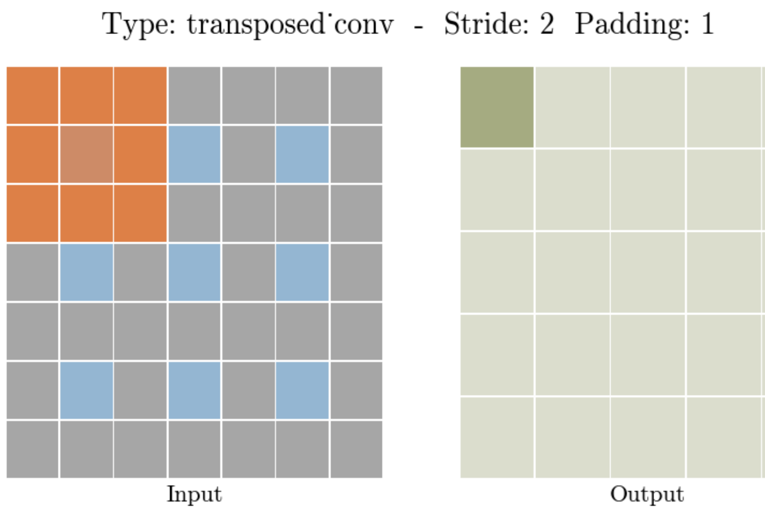


Transposed convolution

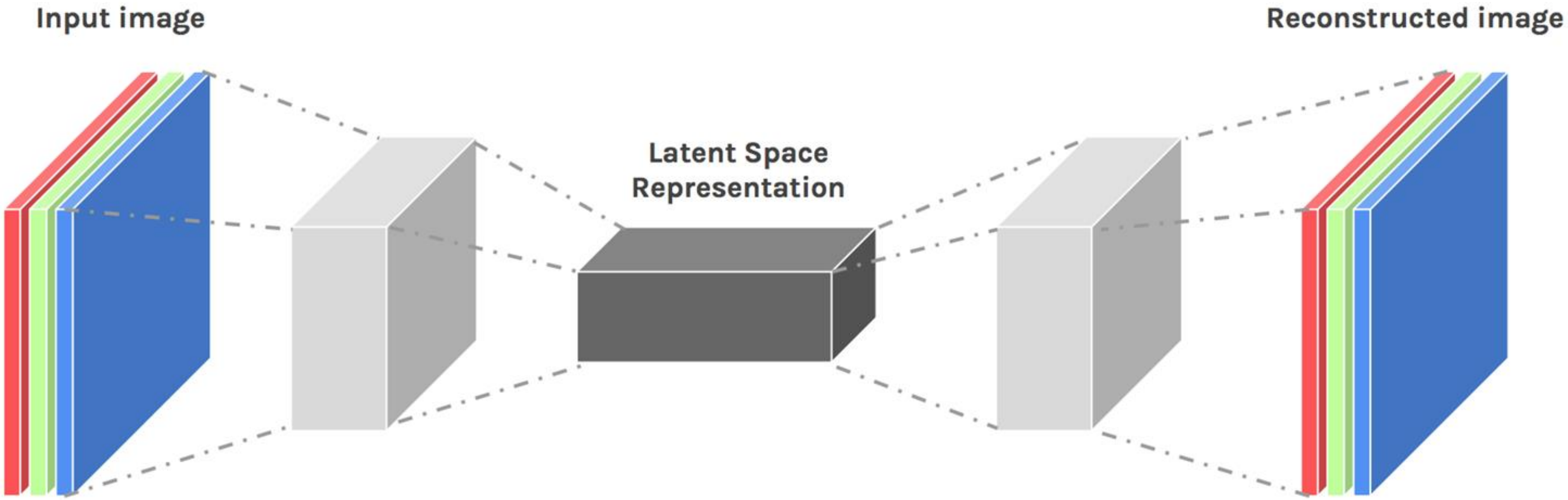


Transposed convolution

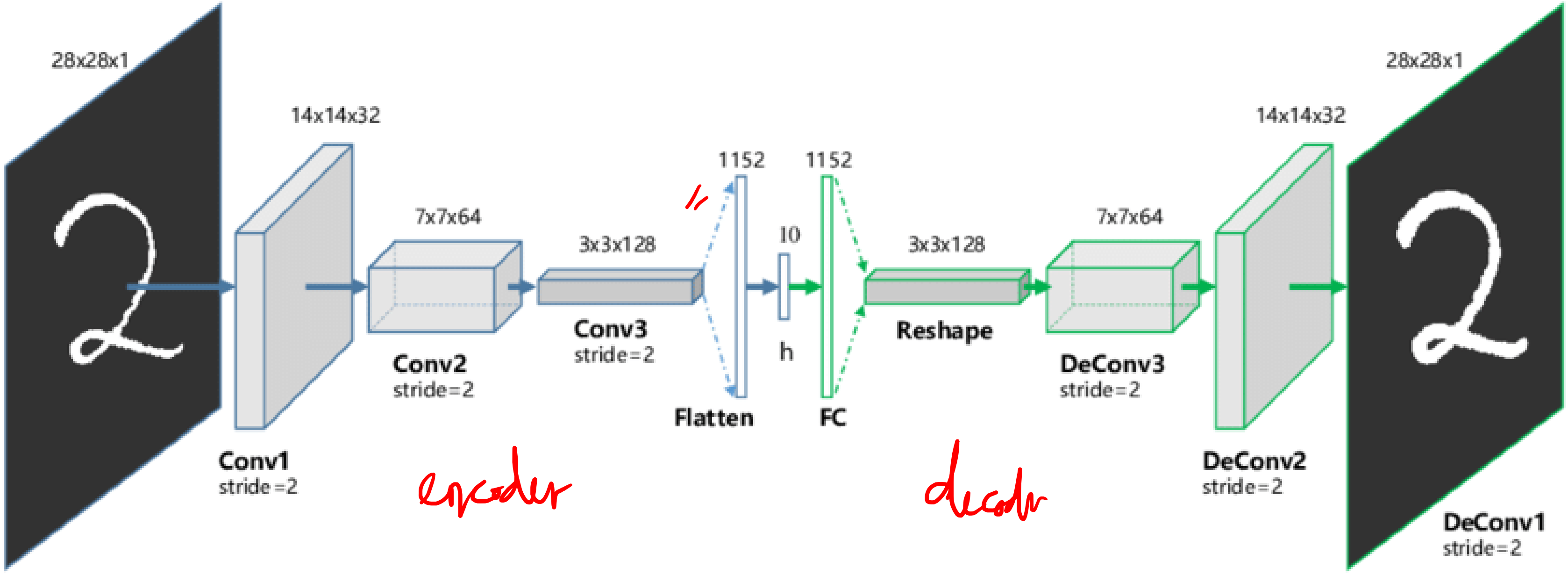




Convolutional AE



Convolutional AE

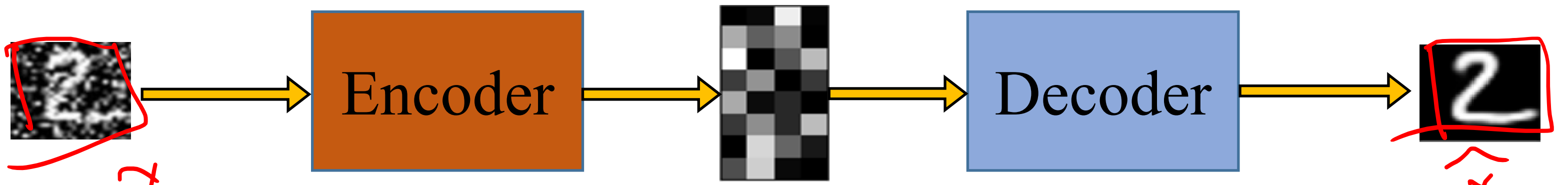


Denoising AE

Intuition:

- We still aim to encode the input and to NOT mimic the identity function.
- We try to undo the effect of *corruption* process stochastically applied to the input.

A more robust model



Noisy Input

Latent space representation

Denoised Input

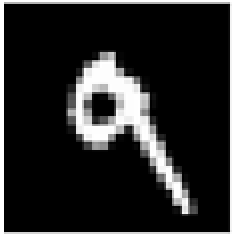
Process

denoise : $\boxed{9} \rightarrow \boxed{9}$ supervised.

Taken some input x

Apply Noise

$\tilde{x}$



y

x

Q&A


@LateNightSeth
THANK YOU!