2023-2024学年第一学期本科生课程

《神经网络与深度学习》

第八节：无监督学习与自编码器

主讲人：戴金晟(副教授，博士生导师)

daijincheng@bupt.edu.cn

神经网络与深度学习课程组

北京邮电大学

Beijing University of Posts and Telecommunications

内容导览

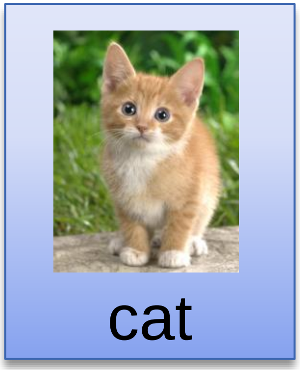
- 学习方法分类
- 主成分分析：线性自编码器
- 非线性深度自编码器(AE)
- AE的重要变型与应用

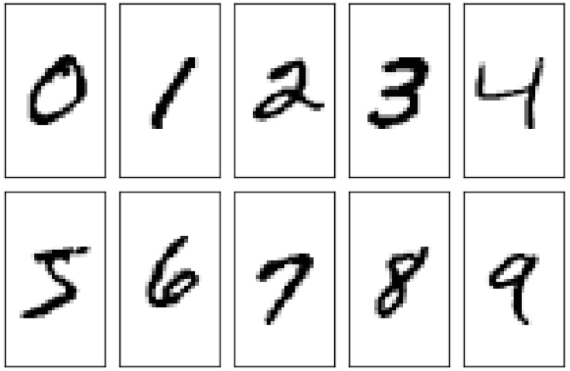
内容导览

- 学习方法分类
- 主成分分析：线性自编码器
- 非线性深度自编码器(AE)
- AE的重要变型与应用

有监督学习

有标签
数据





$$(\tilde{\boldsymbol{x}}_1, \tilde{y}_1), \ldots, (\tilde{\boldsymbol{x}}_m, \tilde{y}_m) \in \mathcal{X} \times \mathcal{Y}$$

Machine Learning: Formulation

Given a dataset $\mathcal{D} = \{x_i, y_i\}$, find a possibly existed mapping $f^*: x \to y$ to fit $\mathcal{D}$. To formulate the problem, choose a hypothetical space $\mathfrak{F}$, a loss function l, and a regularizer p, we find the approximate mapping f_{θ^*} such that

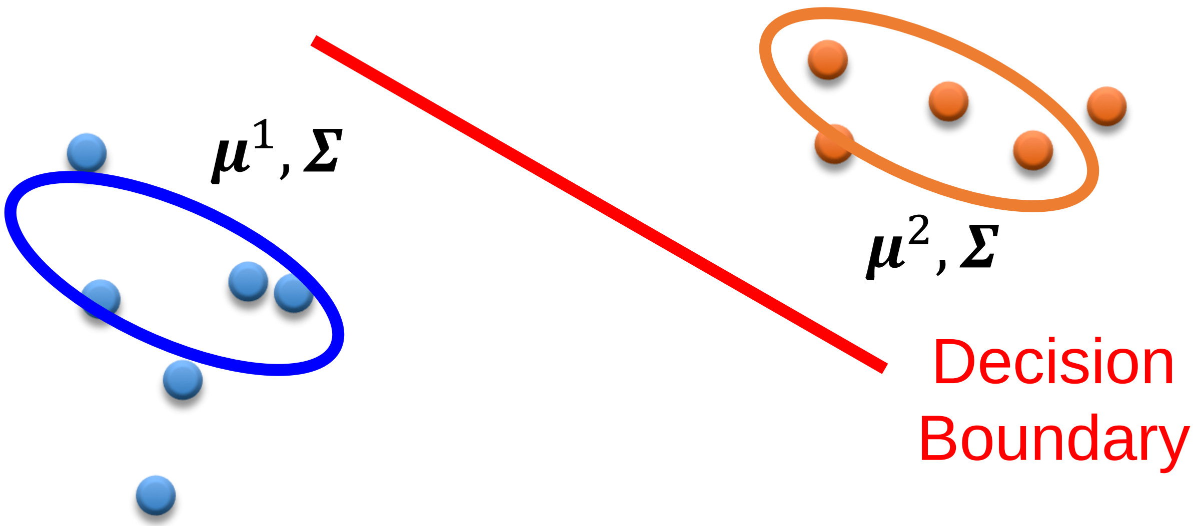
Optim. Algorithm · hypothetical space · loss fun. · data · regularizer

$$f_{\theta^*} = \operatorname{argmin}_{f_\theta \in \mathfrak{F}} \{\mathbb{E}_{\mathcal{D}}[l(f_\theta(x), y)] + \lambda p(f_\theta)\}$$

where f_θ: A function in $\mathfrak{F}$ with parameter θ

有监督学习(Supervised Learning)

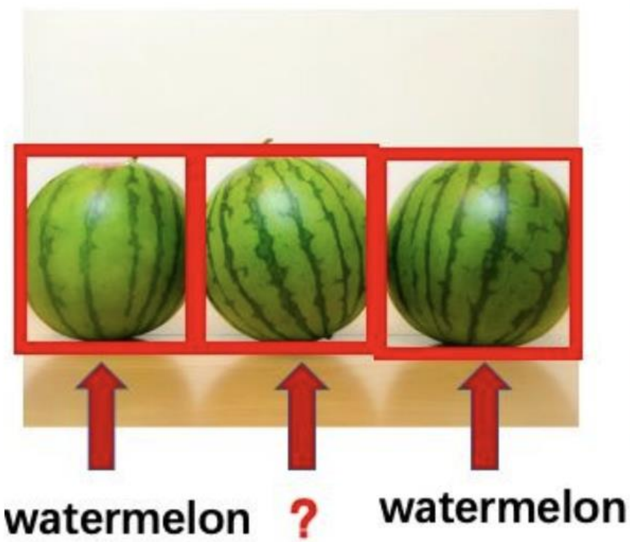
- 给定带有标签的训练数据$\boldsymbol{x} \in C_1, C_2$
  - 寻求最有可能的先验概率$P(C_i)$和类相关概率$P(\boldsymbol{x}|C_i)$
  - $P(\boldsymbol{x}|C_i)$是由$\boldsymbol{\mu}^i$和$\boldsymbol{\Sigma}$参数化的高斯分布



$$P(C_1|\boldsymbol{x}) = \frac{P(\boldsymbol{x}|C_1)P(C_1)}{P(\boldsymbol{x}|C_1)P(C_1) + P(\boldsymbol{x}|C_2)P(C_2)}$$

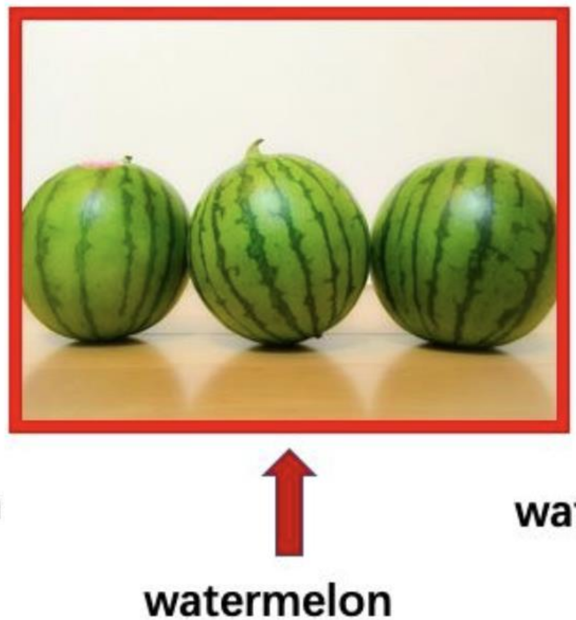
弱监督学习(Weak Supervised Learning)

不完全监督
Incomplete supervision



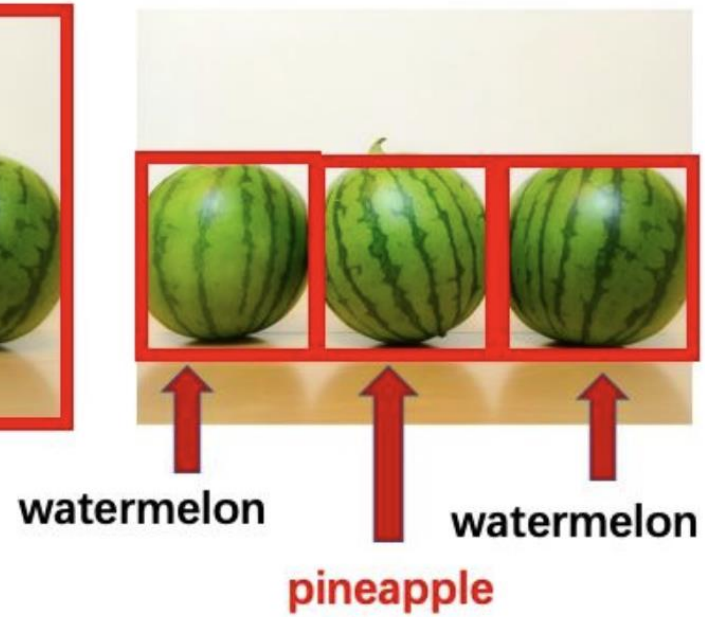
仅标记训练
数据子集

不确切监督
Inexact supervision



训练数据带有标签
但不如预期精确

不准确监督
Inaccurate supervision



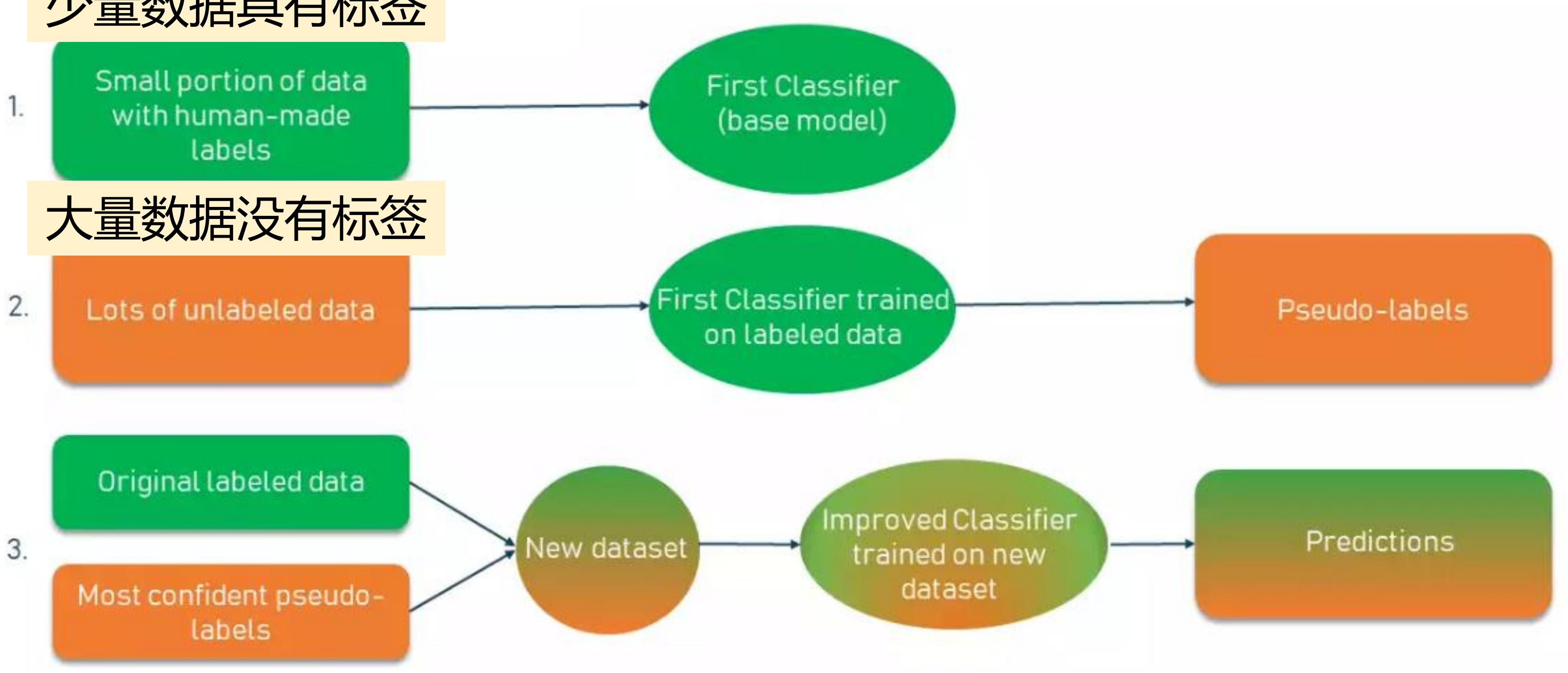
训练数据中一些
标签有错误

半监督学习(Semi-supervised Learning)

- 训练数据$\{(x^r,\hat{y}^r)\}_{r=1}^{R}$, $\{x^u\}_{u=R}^{R+U}$，通常U >> R

少量数据具有标签

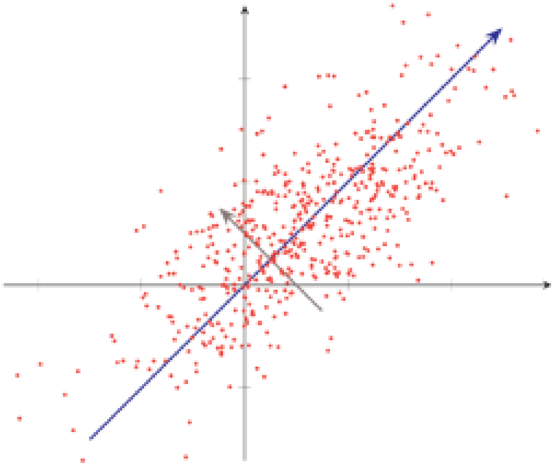
大量数据没有标签

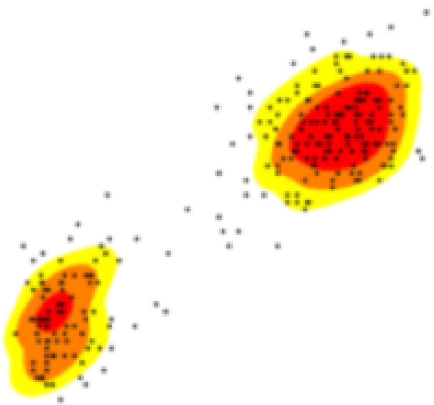


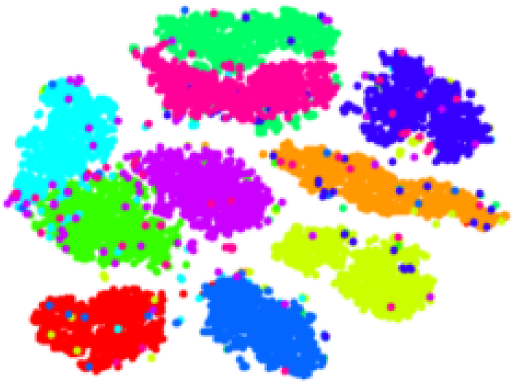
既减少人工标注数据的成本，同时适用于分类、回归、聚类和关联登多种问题

无监督学习(Unsupervised Learning)

- 训练数据由一组输入向量组成，但没有相应的目标值
  - 背后思想是根据相似性、模式和差异对信息进行分组


特征学习


密度估计


聚类

无监督学习：不借助于任何人工给出标签或者反馈等指导信息

无监督学习(Unsupervised Learning)

数据降维(化繁为简)

only having function input

function

内容生成(无中生有)

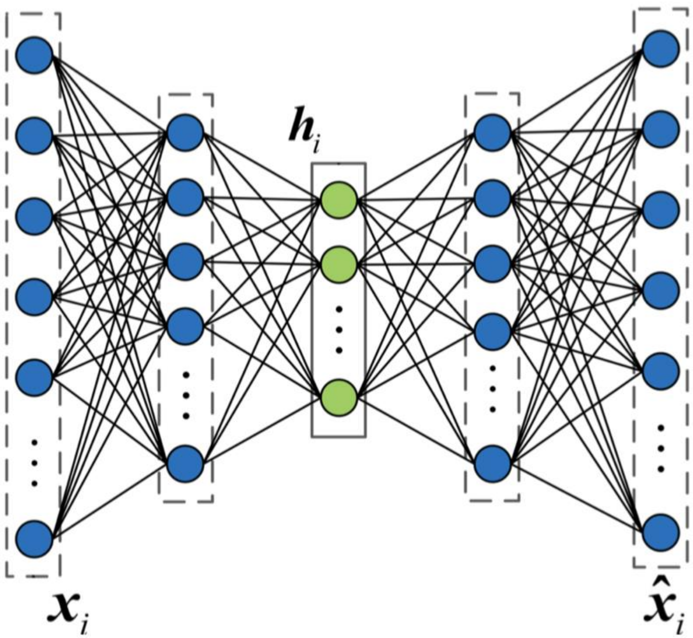
only having function output

function

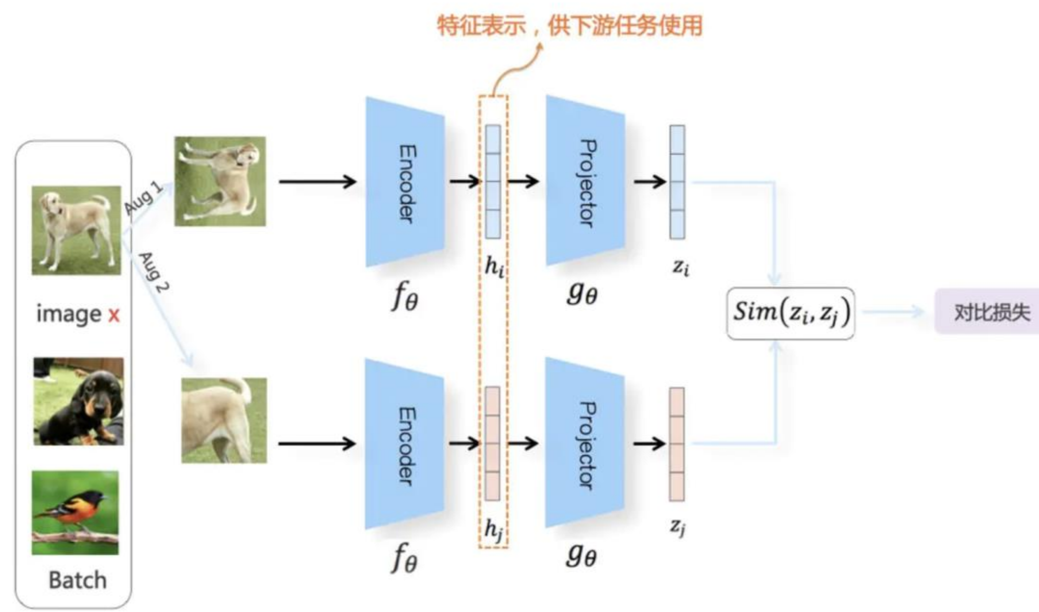
Random variables

自监督学习(Self-supervised Learning)

- 自监督学习(self-supervised learning)可以被看作是机器学习的一种“理想状态”，模型**直接从无标签数据中自行学习，无需标注数据**
  - 自监督学习的核心，在于**如何自动为数据产生标签**
  - 有很多监督信号在训练过程中充当反馈



生成式模型



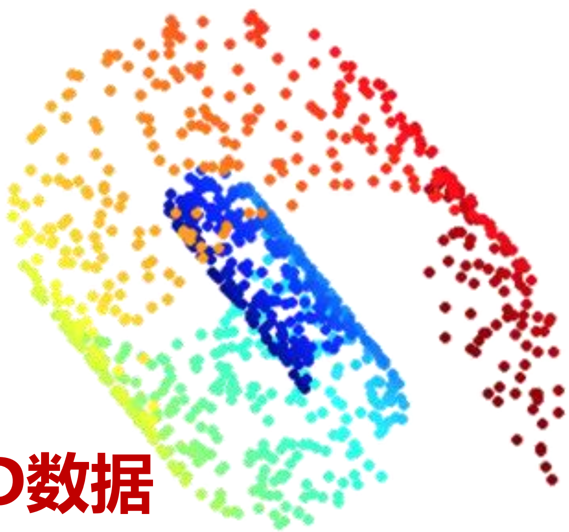
判别式模型

内容导览

- 学习方法分类
- **主成分分析：线性自编码器**
- 非线性深度自编码器(AE)
- AE的重要变型与应用

数据降维(Dimension Reduction)

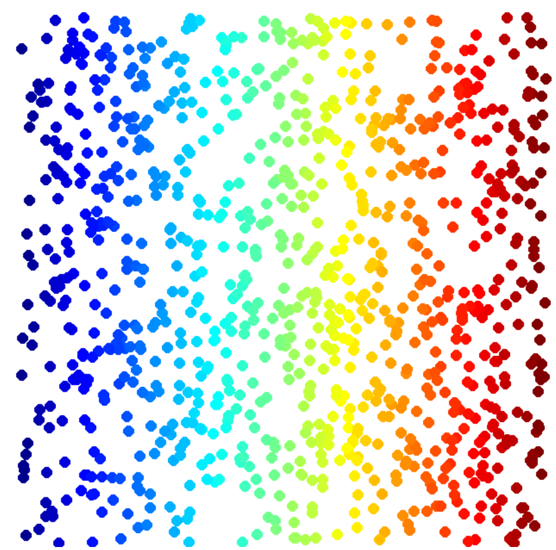
□ **表征学习的核心任务：**选择合适的方式，可以获得更加直观的数据表示



3D数据

数据降维

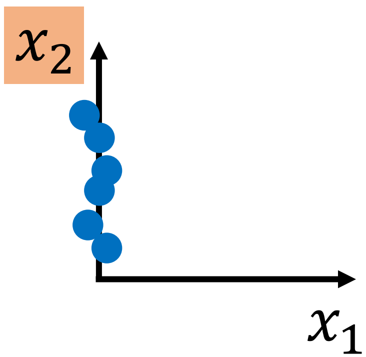
实际是2D数据

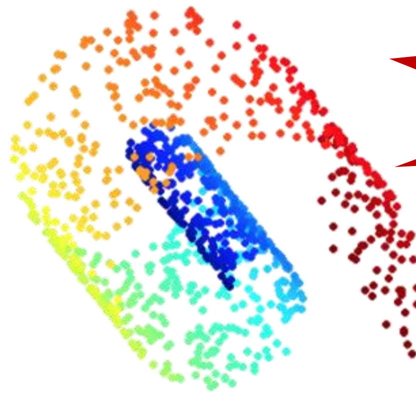


数据降维(Dimension Reduction)

$\boldsymbol{x}$ (高维度) → **函数 f** → $\boldsymbol{z}$ (低维度)

□特征选择





实际情况可能每一个特征都不能舍去

□主成分分析 (PCA)

核心：根据给定的 x 找到 z

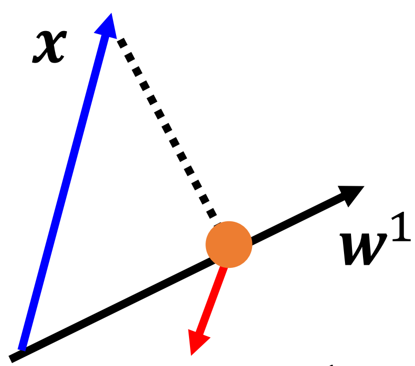
➢ 假设此处 f 是一个简单的线性函数，那么向量 $\boldsymbol{x}$ 和向量 $\boldsymbol{z}$ 之间的关系可以表示为 $\boldsymbol{z} = \boldsymbol{W}\boldsymbol{x}$

主成分分析 (PCA)

$\boldsymbol{z} = \boldsymbol{W}\boldsymbol{x}$

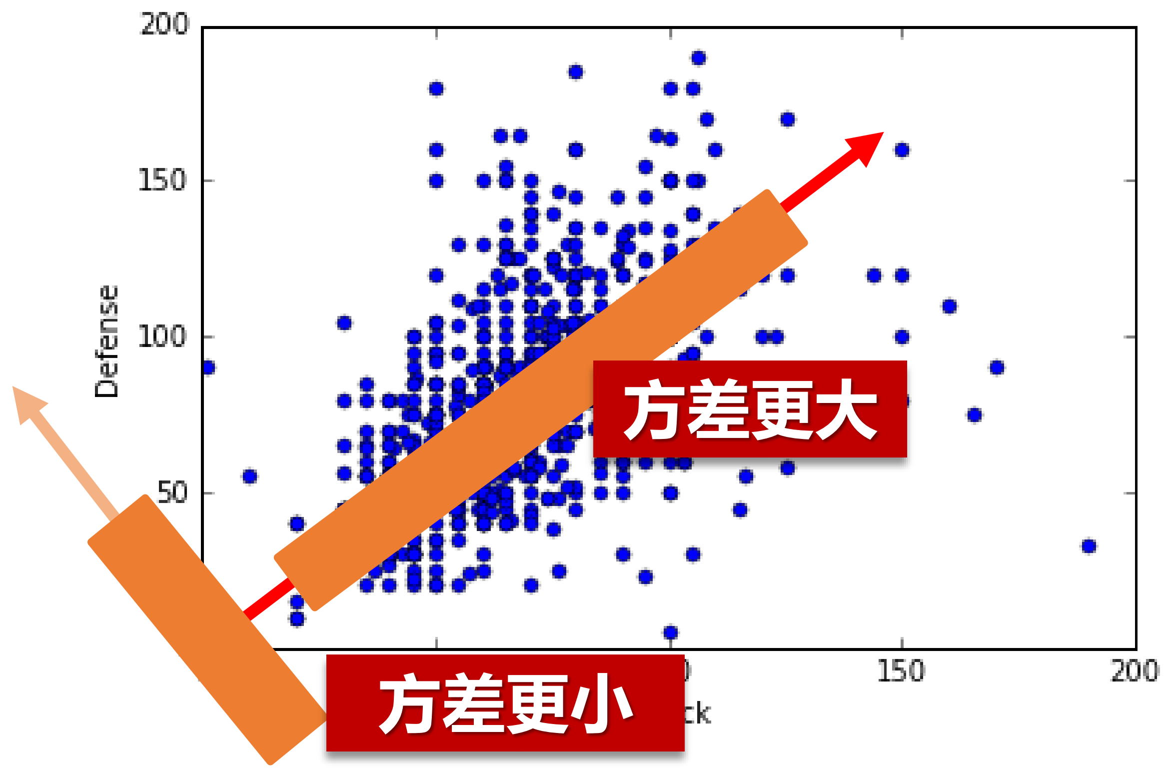
- 降至一维

$$\boldsymbol{z}_1 = \boldsymbol{w}^1 \cdot \boldsymbol{x}$$



$$\boldsymbol{z}_1 = \boldsymbol{w}^1 \cdot \boldsymbol{x}$$

基于向量 $\boldsymbol{w}^1$ 对所有的数据 $\boldsymbol{x}$ 进行投影，得到一系列的 $\boldsymbol{z}_1$



我们希望 z_1 的方差尽可能地大

$$Var(\boldsymbol{z}_1) = \frac{1}{N}\sum_{z_1}(\boldsymbol{z}_1 - \bar{\boldsymbol{z}}_1)^2 \quad \|\boldsymbol{w}^1\|_2 = 1$$

主成分分析 (PCA)

$\boldsymbol{z} = \boldsymbol{W}\boldsymbol{x}$

口降至一维

$\boldsymbol{z}_1 = \boldsymbol{w}^1 \cdot \boldsymbol{x}$

$\boldsymbol{z}_2 = \boldsymbol{w}^2 \cdot \boldsymbol{x}$

$$\boldsymbol{W} = \begin{bmatrix} (\boldsymbol{w}^1)^T \\ (\boldsymbol{w}^2)^T \\ \vdots \end{bmatrix}$$

正交矩阵

基于向量 $\boldsymbol{w}^1$ 对所有的数据 $\boldsymbol{x}$ 进行投影，得到一系列的 $\boldsymbol{z}_1$

希望 z_1 的方差尽可能地大

$$Var(\boldsymbol{z}_1) = \frac{1}{N}\sum_{\boldsymbol{z}_1}(\boldsymbol{z}_1 - \bar{\boldsymbol{z}}_1)^2 \quad \|\boldsymbol{w}^1\|_2 = 1$$

希望 z_2 的方差也尽可能地大

$$Var(\boldsymbol{z}_2) = \frac{1}{N}\sum_{\boldsymbol{z}_2}(\boldsymbol{z}_2 - \bar{\boldsymbol{z}}_2)^2 \quad \|\boldsymbol{w}^2\|_2 = 1$$

$$\boldsymbol{w}^1 \cdot \boldsymbol{w}^2 = 0$$

PCA的数学原理

$$\boldsymbol{z}_1 = \boldsymbol{w}^1 \cdot \boldsymbol{x} \quad \bar{\boldsymbol{z}}_1 = \frac{1}{N}\sum \boldsymbol{z}_1 = \frac{1}{N}\sum \boldsymbol{w}^1 \cdot \boldsymbol{x} = \boldsymbol{w}^1 \cdot \frac{1}{N}\sum \boldsymbol{x} = \boldsymbol{w}^1 \cdot \bar{\boldsymbol{x}}$$

$$Var(\boldsymbol{z}_1) = \frac{1}{N}\sum_{\boldsymbol{z}_1}(\boldsymbol{z}_1 - \bar{\boldsymbol{z}}_1)^2 = \frac{1}{N}\sum_{\boldsymbol{x}}(\boldsymbol{w}^1 \cdot \boldsymbol{x} - \boldsymbol{w}^1 \cdot \bar{\boldsymbol{x}})^2$$

$$= \frac{1}{N}\sum\left(\boldsymbol{w}^1 \cdot (\boldsymbol{x} - \bar{\boldsymbol{x}})\right)^2$$

$$= \frac{1}{N}\sum(\boldsymbol{w}^1)^T(\boldsymbol{x} - \bar{\boldsymbol{x}})(\boldsymbol{x} - \bar{\boldsymbol{x}})^T\boldsymbol{w}^1$$

$$= (\boldsymbol{w}^1)^T \boxed{\frac{1}{N}\sum(\boldsymbol{x} - \bar{\boldsymbol{x}})(\boldsymbol{x} - \bar{\boldsymbol{x}})^T} \boldsymbol{w}^1$$

$$= (\boldsymbol{w}^1)^T Cov(\boldsymbol{x})\boldsymbol{w}^1$$

$$\begin{aligned}(\boldsymbol{a} \cdot \boldsymbol{b})^2 &= (\boldsymbol{a}^T\boldsymbol{b})^2 \\ &= \boldsymbol{a}^T\boldsymbol{b}\boldsymbol{a}^T\boldsymbol{b} \\ &= \boldsymbol{a}^T\boldsymbol{b}(\boldsymbol{a}^T\boldsymbol{b})^T \\ &= \boldsymbol{a}^T\boldsymbol{b}\boldsymbol{b}^T\boldsymbol{a}\end{aligned}$$

在 $\|\boldsymbol{w}^1\|_2 = (\boldsymbol{w}^1)^T\boldsymbol{w}^1 = 1$ 的条件下找到使得 $(\boldsymbol{w}^1)^T\boldsymbol{S}\boldsymbol{w}^1$ 取最大值的$\boldsymbol{w}^1$

$$\boldsymbol{S} = Cov(\boldsymbol{x})$$

PCA的数学原理

求解目标： 在 $\|\boldsymbol{w}^1\|_2 = (\boldsymbol{w}^1)^T\boldsymbol{w}^1 = 1$ 的条件下找到使得 $(\boldsymbol{w}^1)^T\boldsymbol{S}\boldsymbol{w}^1$ 取最大值的 $\boldsymbol{w}^1$

$\boldsymbol{S} = Cov(\boldsymbol{x})$ 对称半正定矩阵

使用拉格朗日乘数法：$g(\boldsymbol{w}^1) = (\boldsymbol{w}^1)^T\boldsymbol{S}\boldsymbol{w}^1 - \alpha\left((\boldsymbol{w}^1)^T\boldsymbol{w}^1 - 1\right)$

$$\left.\begin{array}{l} \partial g(\boldsymbol{w}^1)/\partial w_1^1 = 0 \\ \partial g(\boldsymbol{w}^1)/\partial w_2^1 = 0 \\ \vdots \end{array}\right\}$$

$$\boldsymbol{S}\boldsymbol{w}^1 - \alpha\boldsymbol{w}^1 = 0$$

$$\boldsymbol{S}\boldsymbol{w}^1 = \alpha\boldsymbol{w}^1$$

$\boldsymbol{w}^1$：**特征向量**

$$(\boldsymbol{w}^1)^T\boldsymbol{S}\boldsymbol{w}^1 = \alpha(\boldsymbol{w}^1)^T\boldsymbol{w}^1 = \alpha$$

选择最大值

$\boldsymbol{w}^1$ 是协方差矩阵 $\boldsymbol{S}$ 的特征向量，对应最大的特征值 λ_1

PCA的数学原理

求解目标： 在$(\boldsymbol{w}^2)^T\boldsymbol{w}^2 = 1$和$(\boldsymbol{w}^2)^T\boldsymbol{w}^1 = 0$的条件下找到使得 $(\boldsymbol{w}^2)^T\boldsymbol{S}\boldsymbol{w}^2$ 取最大值的$\boldsymbol{w}^2$

$$g(\boldsymbol{w}^2) = (\boldsymbol{w}^2)^T\boldsymbol{S}\boldsymbol{w}^2 - \alpha\left((\boldsymbol{w}^2)^T\boldsymbol{w}^2 - 1\right) - \beta\left((\boldsymbol{w}^2)^T\boldsymbol{w}^1 - 0\right)$$

$$\partial g(\boldsymbol{w}^2)/\partial w_1^2 = 0$$

$$\partial g(\boldsymbol{w}^2)/\partial w_2^2 = 0$$

$$\vdots$$

$$\boldsymbol{S}\boldsymbol{w}^2 - \alpha\boldsymbol{w}^2 - \beta\boldsymbol{w}^1 = 0$$

$$\boxed{0} - \alpha\boxed{0} - \beta\boxed{1} = 0$$

$$= \left((\boldsymbol{w}^1)^T\boldsymbol{S}\boldsymbol{w}^2\right)^T = (\boldsymbol{w}^2)^T\boldsymbol{S}^T\boldsymbol{w}^1$$

$$= (\boldsymbol{w}^2)^T\boldsymbol{S}\boldsymbol{w}^1 = \lambda_1(\boldsymbol{w}^2)^T\boldsymbol{w}^1 = 0$$

$$\boldsymbol{S}\boldsymbol{w}^1 = \lambda_1\boldsymbol{w}^1$$

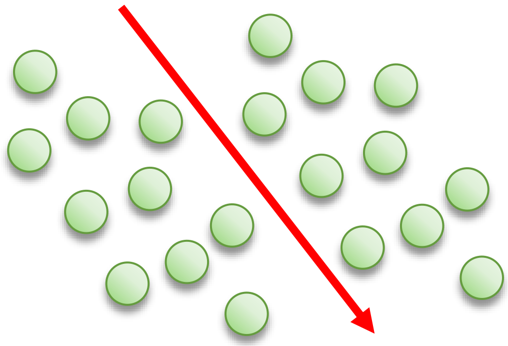
$\beta = 0$: $\quad \boldsymbol{S}\boldsymbol{w}^2 - \alpha\boldsymbol{w}^2 = 0 \qquad \boldsymbol{S}\boldsymbol{w}^2 = \alpha\boldsymbol{w}^2$

$\boldsymbol{w}^2$ 是协方差矩阵 $\boldsymbol{S}$ 的特征向量，对应第二大的特征值 λ_2

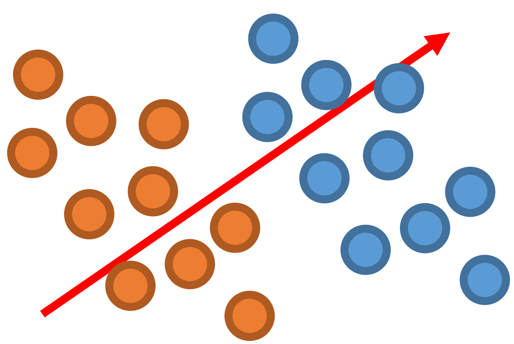
PCA的缺点

- 无监督

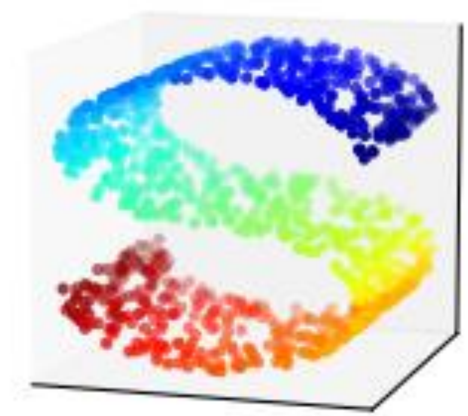
PCA

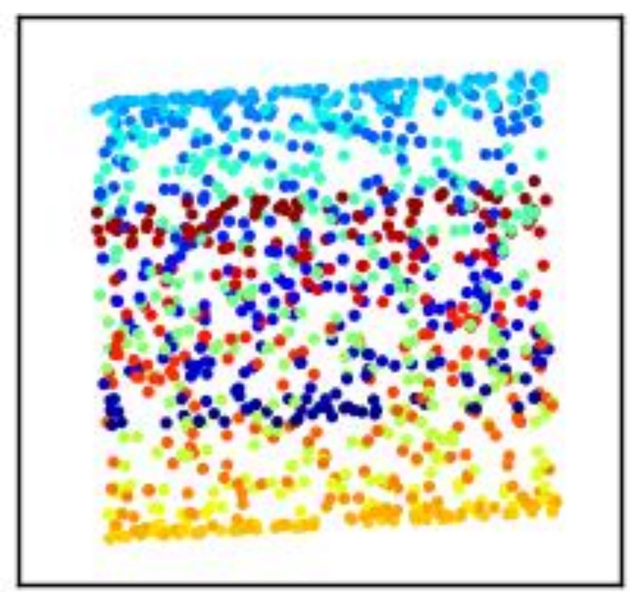


LDA



- 只能进行线性处理

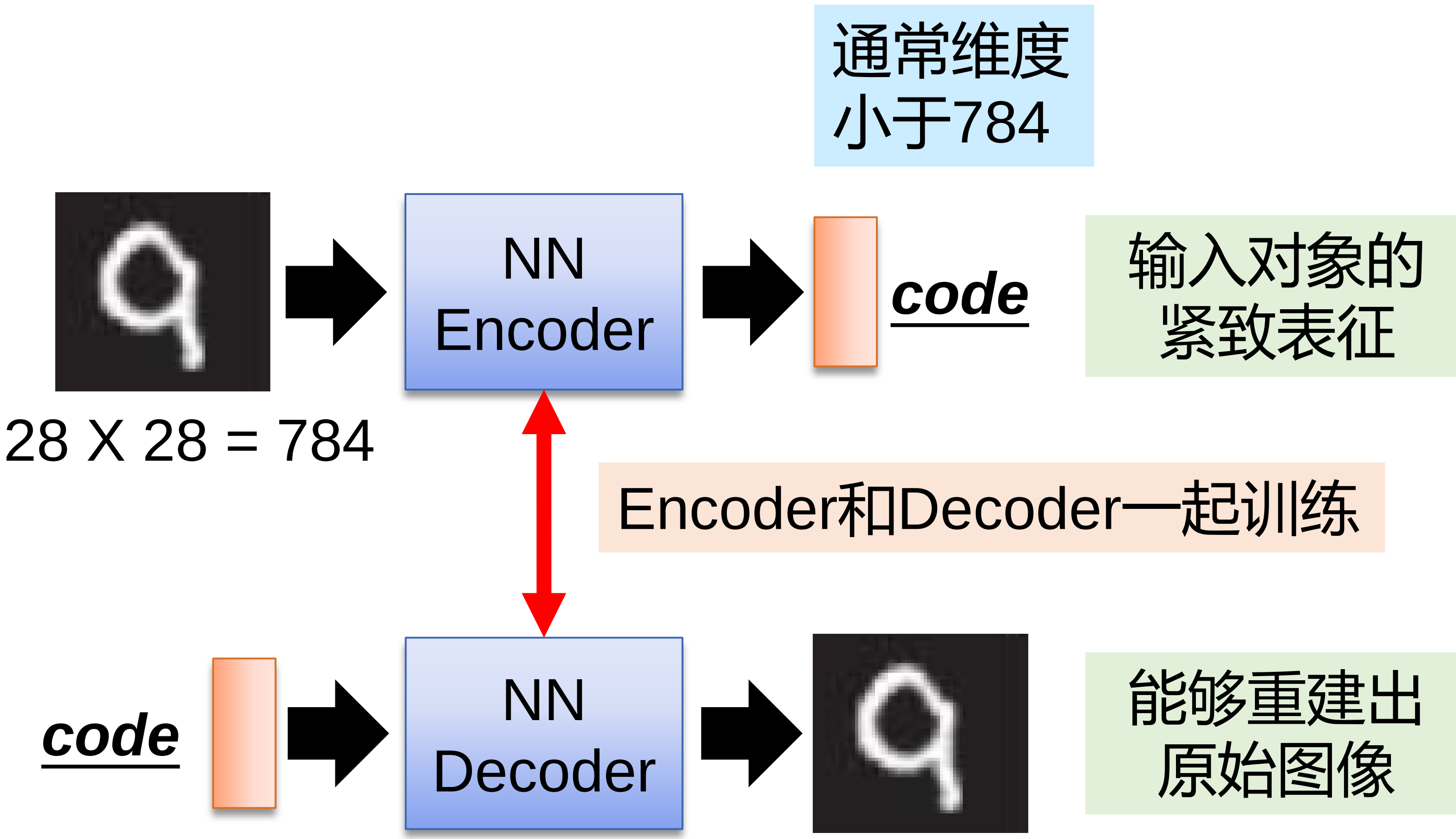




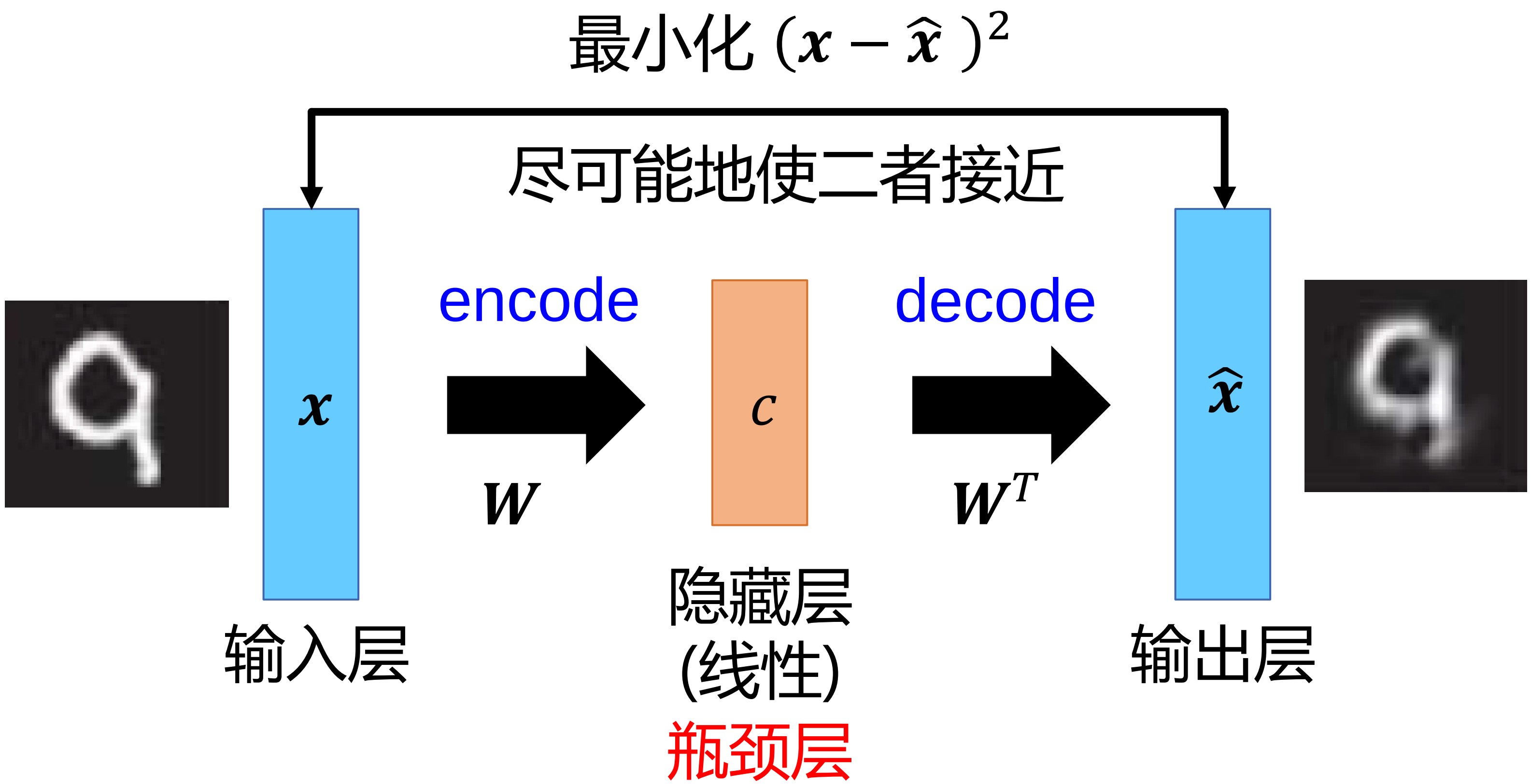
内容导览

- 学习方法分类
- 主成分分析：线性自编码器
- 非线性深度自编码器(AE)
- AE的重要变型与应用

自编码器(Autoencoder, AE)



从 AE 角度再看 PCA



隐藏层的输出就是 AE 中的 ***latent code***
(embedding, latent representation)

从神经网络角度认识PCA

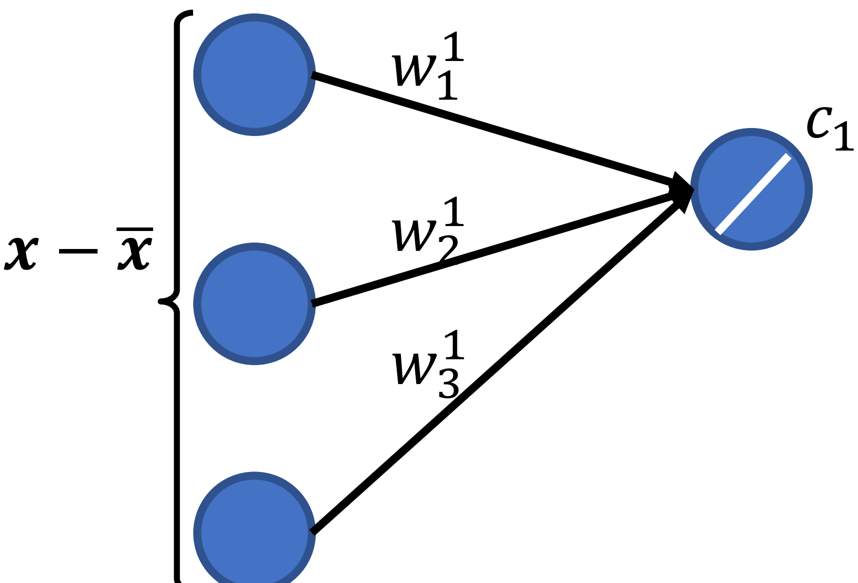
- PCA可以当作是一个只有一层具有线性激活函数隐藏层的神经网络

Autoencoder

如果 $\{w^1, w^2, \ldots w^K\}$ 是 $\{u^1, u^2, \ldots u^K\}$

$$\hat{\boldsymbol{x}} = \sum_{k=1}^{K} c_k \boldsymbol{w}^k \Leftrightarrow \boldsymbol{x} - \bar{\boldsymbol{x}}$$ 为了最小化重建：$c_k = (\boldsymbol{x} - \bar{\boldsymbol{x}}) \cdot \boldsymbol{w}^k$

$K = 2$:



从神经网络角度认识PCA

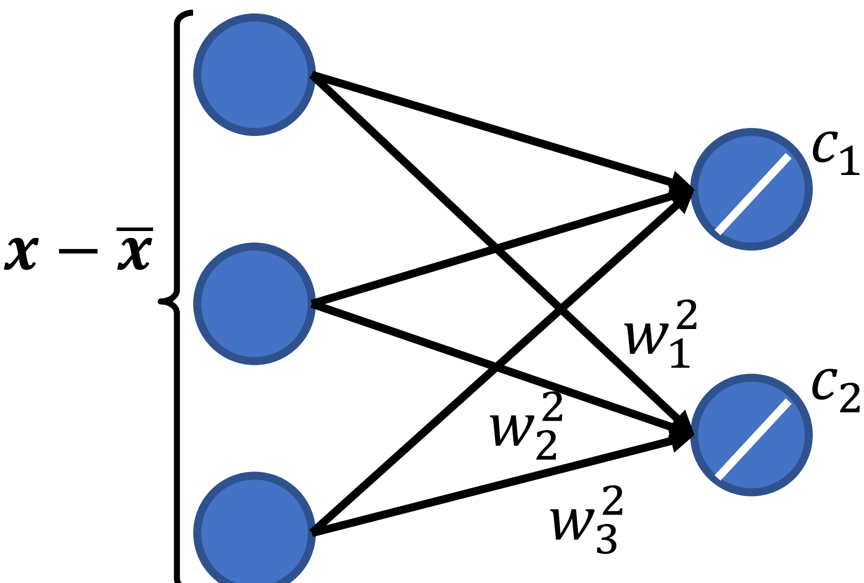
- PCA可以当作是一个只有一层具有线性激活函数隐藏层的神经网络

Autoencoder

如果 $\{w^1, w^2, \ldots w^K\}$ 是 $\{u^1, u^2, \ldots u^K\}$

$$\hat{\boldsymbol{x}} = \sum_{k=1}^{K} c_k \boldsymbol{w}^k \Leftrightarrow \boldsymbol{x} - \bar{\boldsymbol{x}}$$ 为了最小化重建：$c_k = (\boldsymbol{x} - \bar{\boldsymbol{x}}) \cdot \boldsymbol{w}^k$

$K = 2$:



从神经网络角度认识PCA

- PCA可以当作是一个只有一层具有线性激活函数隐藏层的神经网络

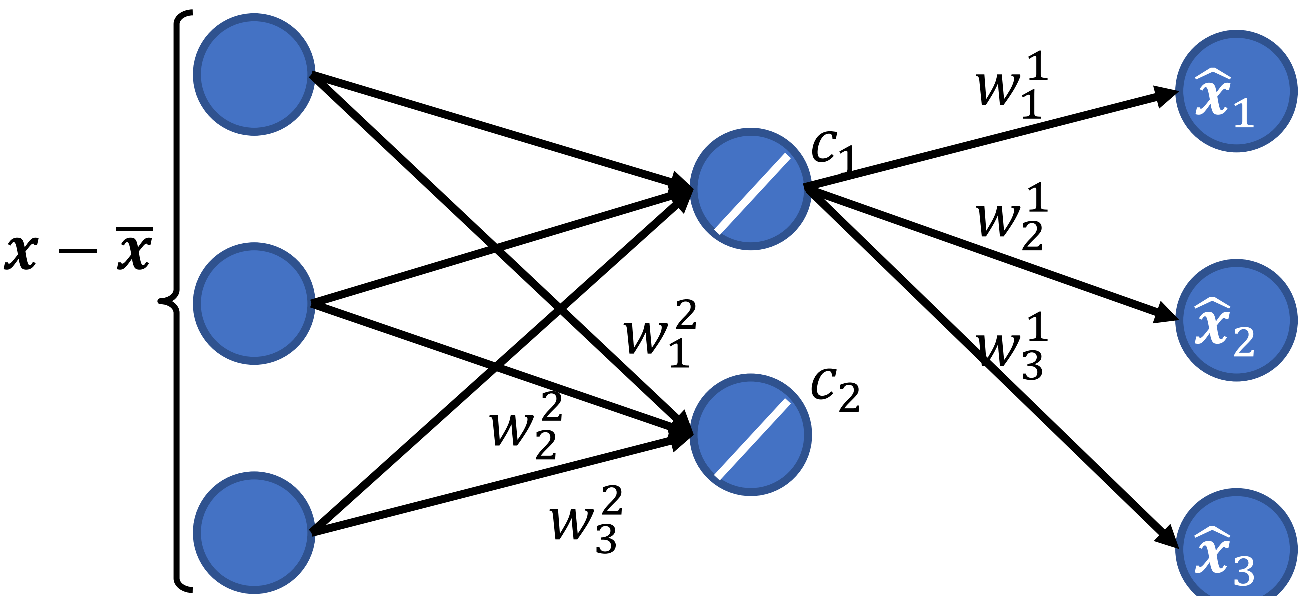
Autoencoder

如果 $\{w^1, w^2, \dots w^K\}$ 是 $\{u^1, u^2, \dots u^K\}$

$$\hat{\boldsymbol{x}} = \sum_{k=1}^{K} c_k \boldsymbol{w}^k \Leftrightarrow \boldsymbol{x} - \bar{\boldsymbol{x}}$$

为了最小化重建：$c_k = (\boldsymbol{x} - \bar{\boldsymbol{x}}) \cdot \boldsymbol{w}^k$

$K = 2$:



从神经网络角度认识PCA

- PCA可以当作是一个只有一层具有线性激活函数隐藏层的神经网络

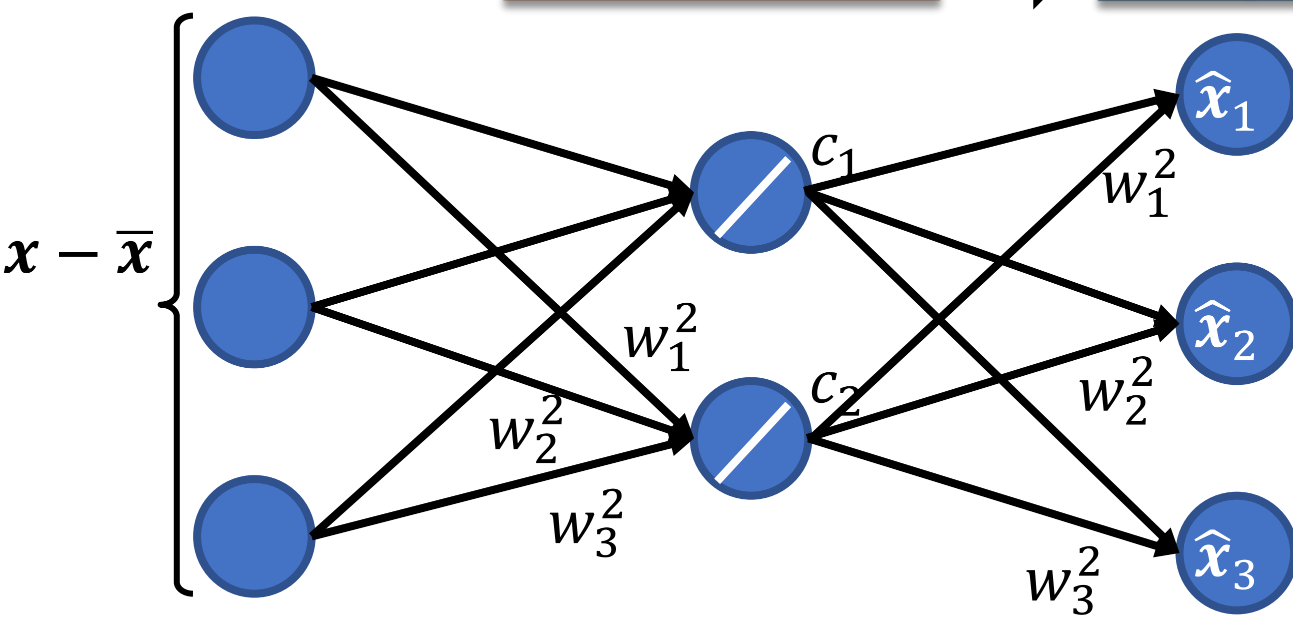
Autoencoder

如果 $\{w^1, w^2, \dots w^K\}$ 是 $\{u^1, u^2, \dots u^K\}$

$\hat{\boldsymbol{x}} = \sum_{k=1}^{K} c_k \boldsymbol{w}^k \Leftrightarrow \boldsymbol{x} - \bar{\boldsymbol{x}}$ 为了最小化重建：$c_k = (\boldsymbol{x} - \bar{\boldsymbol{x}}) \cdot \boldsymbol{w}^k$

$K = 2$:

网络可以更深 → Deep Autoencoder

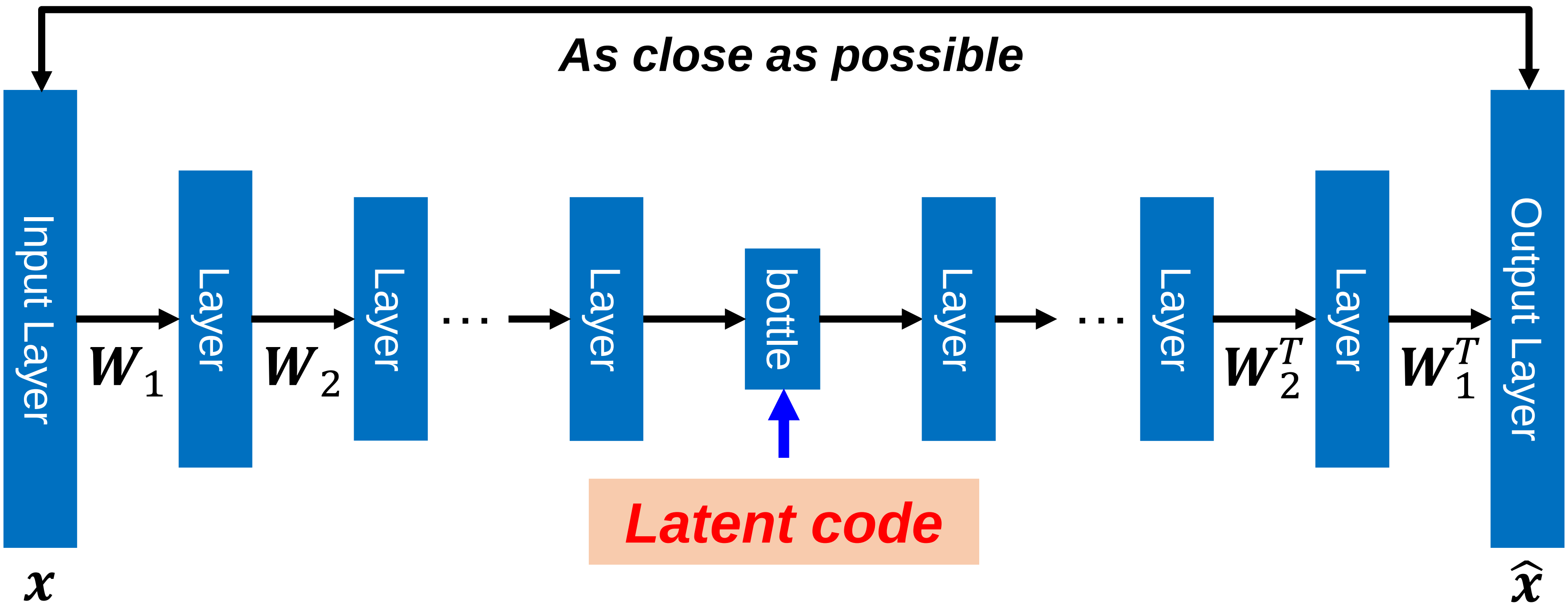


最小化重建误差 ⟷ $\boldsymbol{x} - \bar{\boldsymbol{x}}$

能够使用梯度下降法吗？

深度自编码器(Deep Autoencoder)

- Autoencoder可以使用深度神经网络



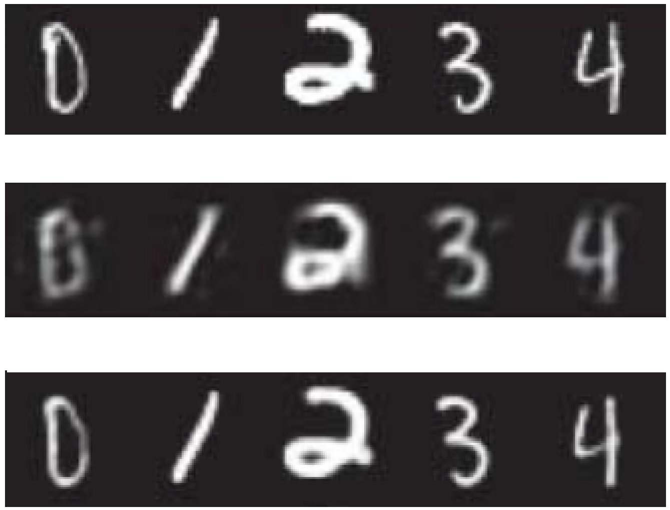
Hinton, Geoffrey E., and Ruslan R. Salakhutdinov. "Reducing the dimensionality of data with neural networks." *Science* 313.5786 (2006): 504-507

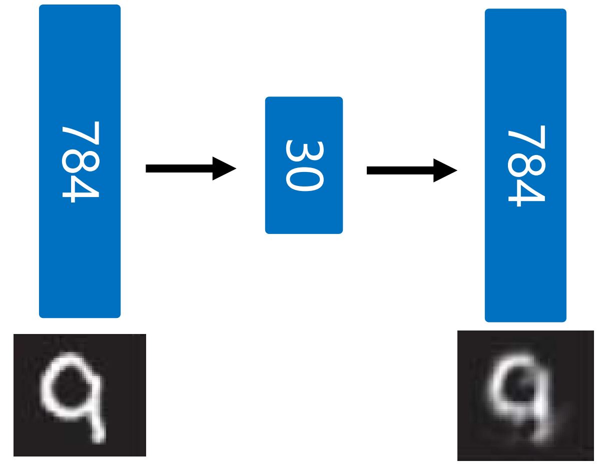
深度自编码器(Deep Autoencoder)

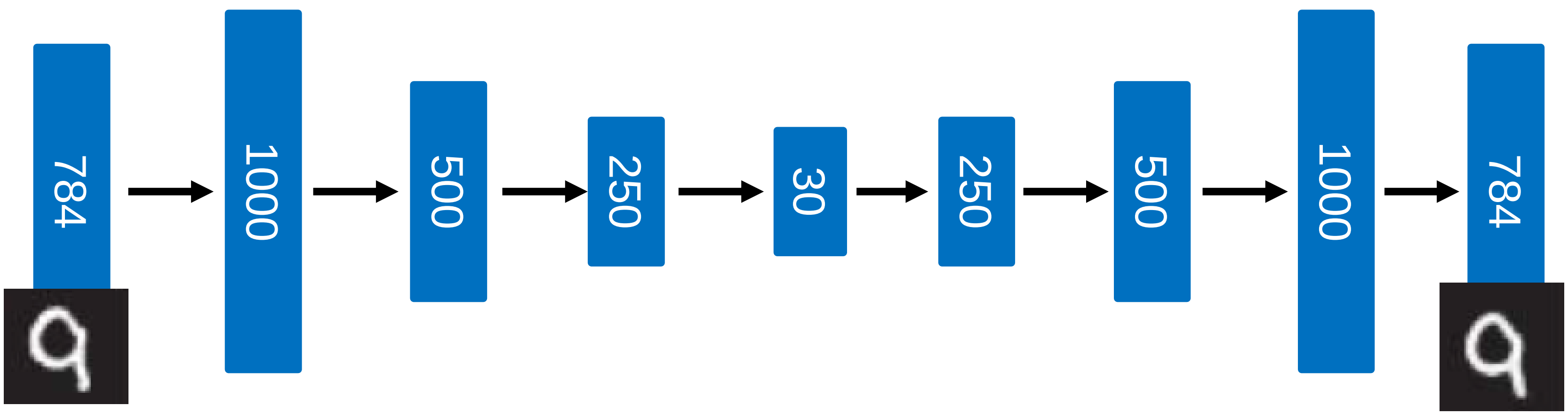
原始图像

PCA

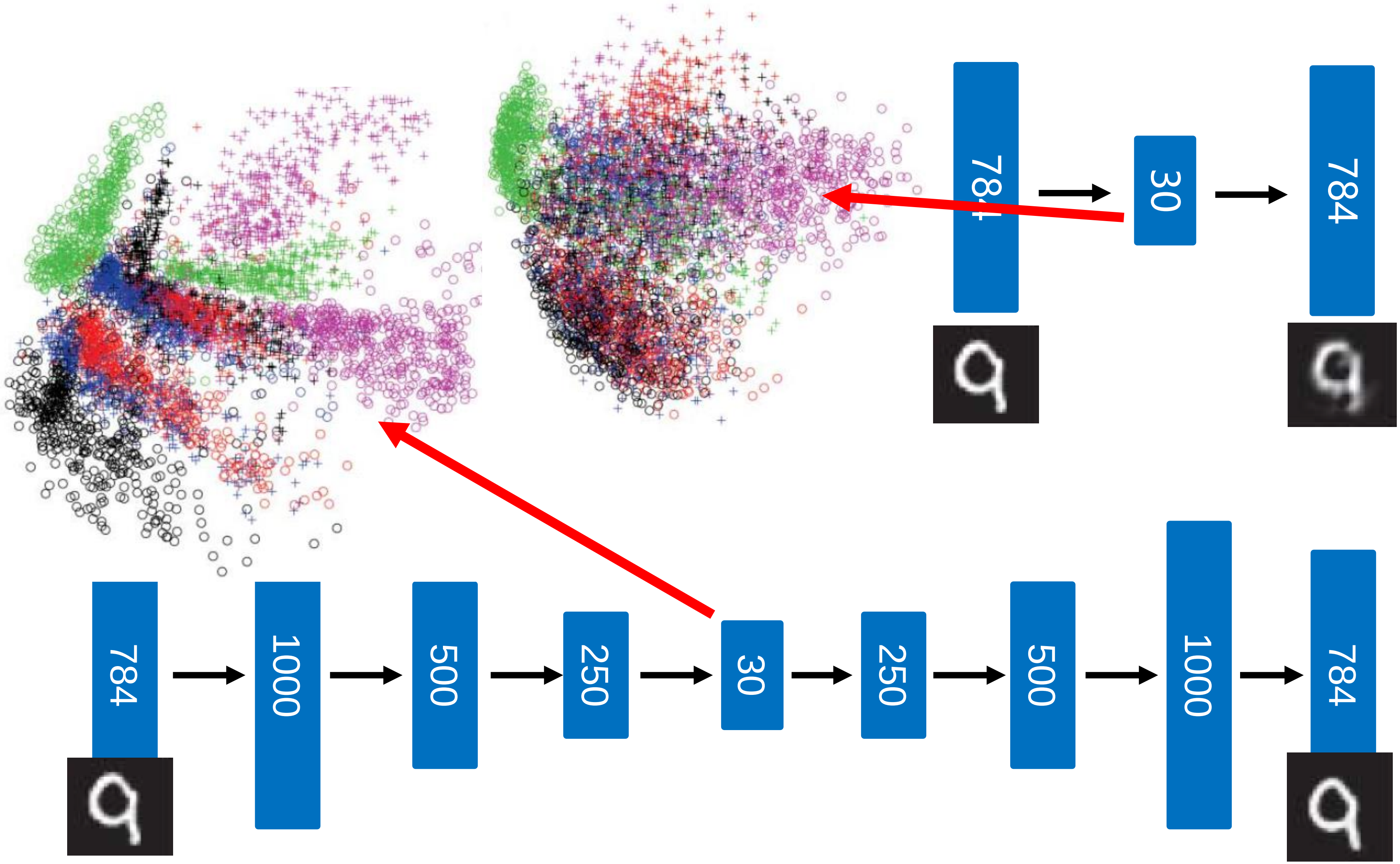
Deep
Auto-encoder







深度自编码器(Deep Autoencoder)



内容导览

- 学习方法分类
- 主成分分析：线性自编码器
- 非线性深度自编码器(AE)
- AE的重要变型与应用

正则自编码器

- 损失函数为**重构误差**和**编码层的惩罚项** $\Omega(h)$：

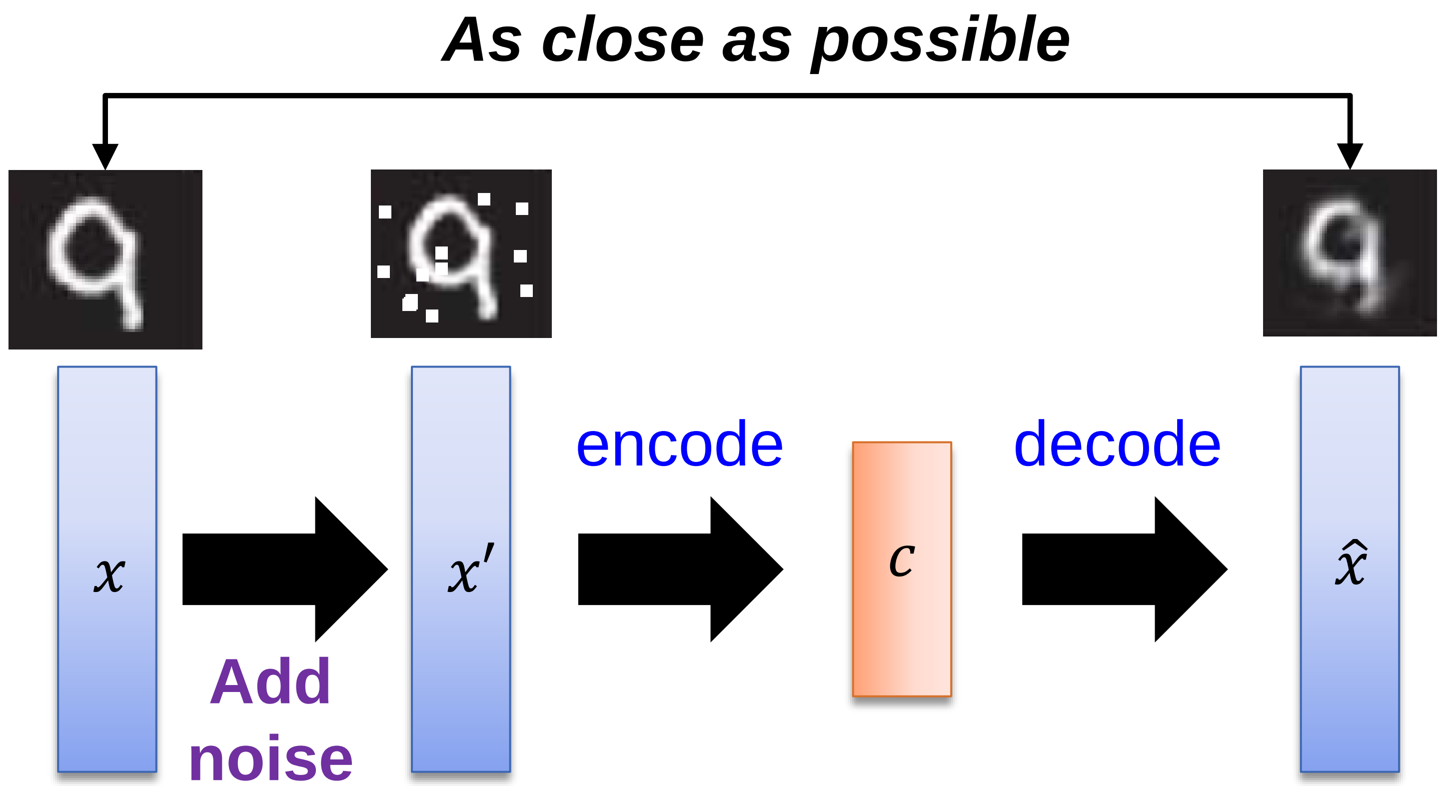
$$L\left(x, g\left(f(x)\right)\right) + \Omega(h)$$

Sensitive enough to inputs so that it can accurately reconstruct input data	Able to generalize well even when evaluated on unseen data

- 增加稀疏惩罚：稀疏自编码器 ⇨ 可以很好的重构输入数据
- 增加去噪惩罚：去噪自编码器 ⇨ 对输入数据一定程度下的扰动具有不变形
- 增加收缩惩罚：收缩自编码器 ⇨ 对输入数据一定程度下的扰动具有不变形

去噪自编码器(De-noising autoencoder)

□De-noising autoencoder (去噪自编码器)



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

为什么需要Denoising?

- 图像越清晰，越容易理解
- 去噪在医疗和自动驾驶领域意义重大
- 自编码器在去噪方便的应用使得它在特征提取和数据元素理解方面具有巨大潜力

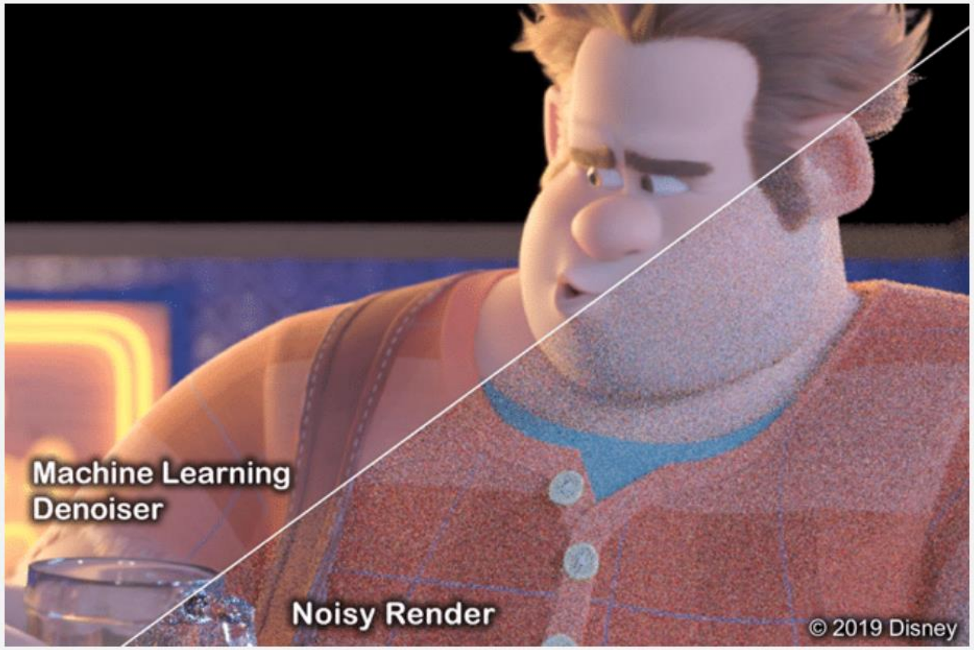
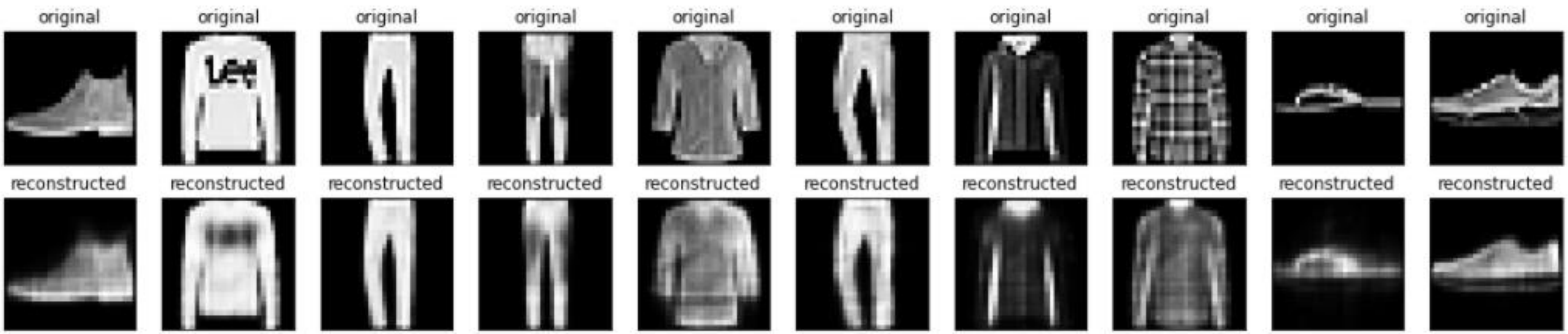


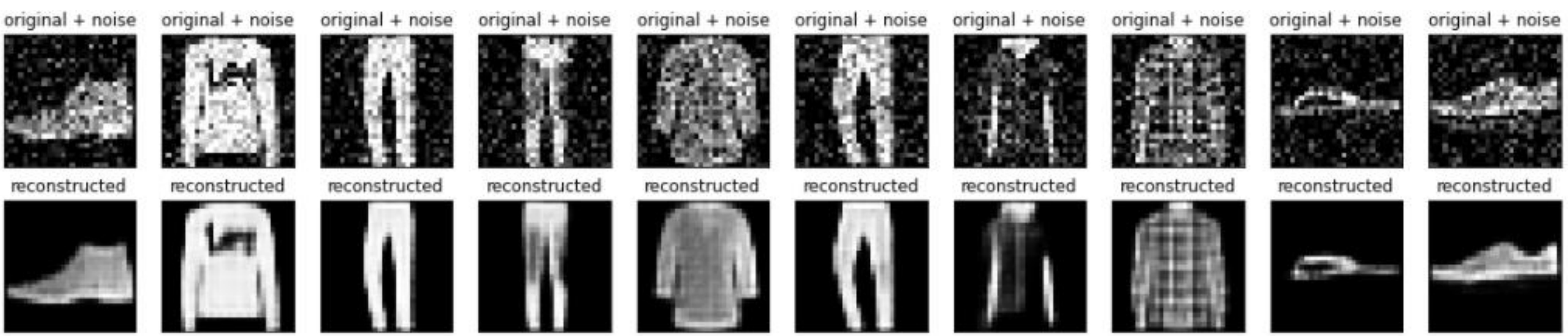
Photo credit: disneyanimation.com

去噪自编码器效果

Basic AutoEncoder



图像去噪



稀疏自编码器(Sparse Autoencoder)

- 稀疏自动编码器只是一个训练标准涉及**稀疏性惩罚**的自动编码器
  - 在大多数情况下，通过惩罚隐藏层节点的激活情况来构建损失函数，当将单个样本输入网络时仅鼓励少数节点激活。

intuition behind method

expert in mathematics	→	**shallow knowledge**
↓		↓
devoted to mathematics	→	**useful insights**

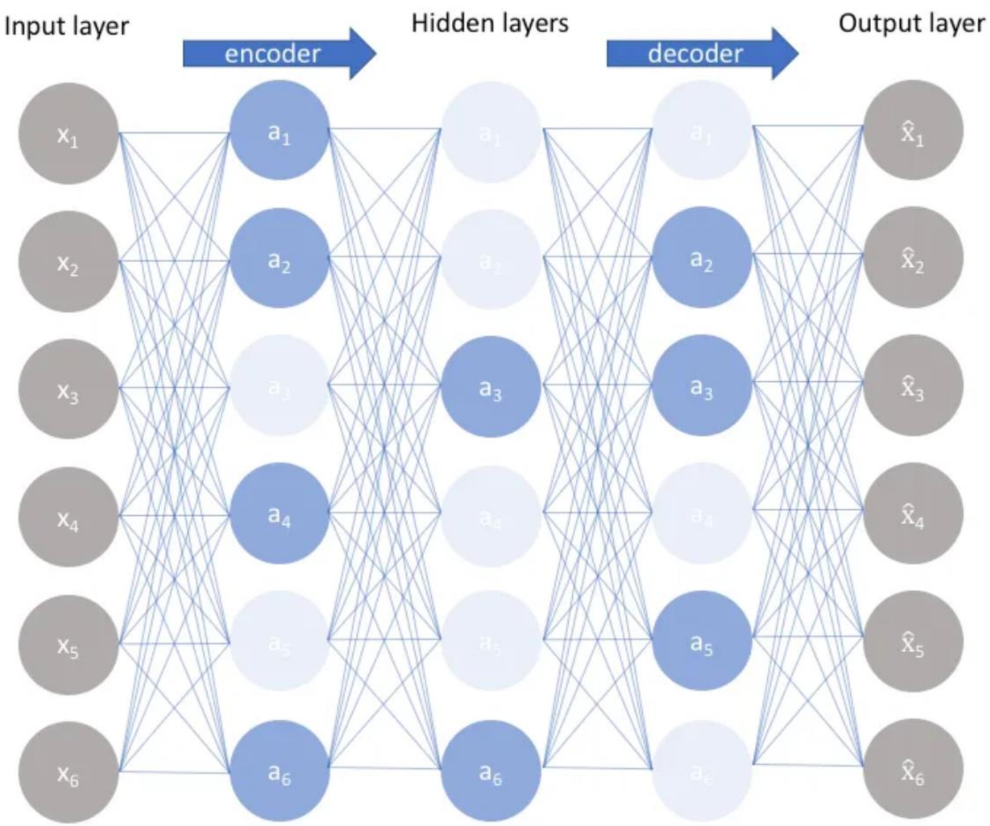
保证性能同时仅有少数节点激活 → 保证自动编码器**实际在学习潜在表示**而非输入数据中冗余信息

https://medium.com/@syoya/what-happens-in-sparse-autencoder-b9a5a69da5c6

稀疏自编码器(Sparse Autoencoder)

☐ 构建稀疏性限制的方法

- ➢ **给自编码器的隐藏层加入 L0/L1 正则化(稀疏惩罚)**
- ➢ 损失函数为重构误差、编码层的稀疏惩罚 $\Omega_{sparse}(h)$



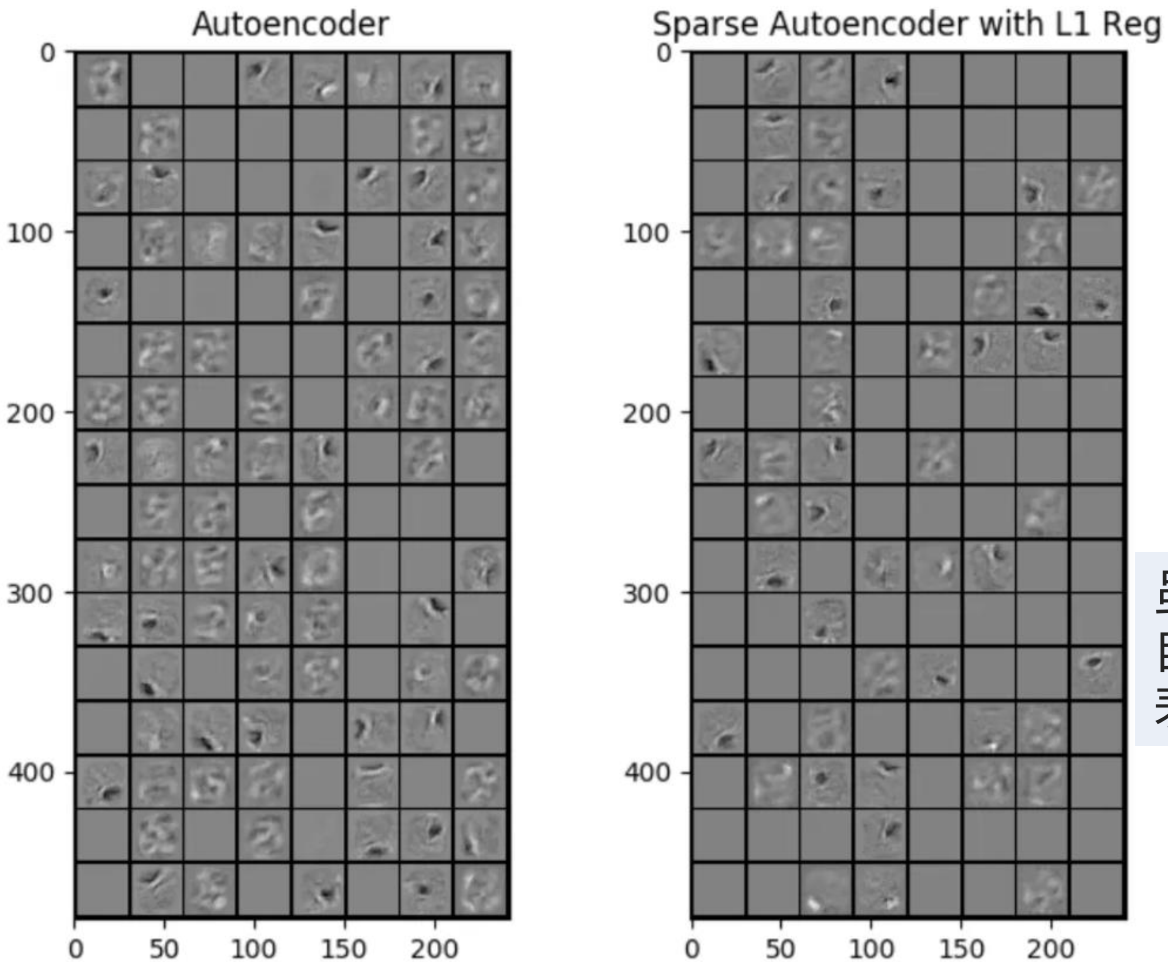
“active”：
神经元输出**接近1**
“inactive”：
神经元输出**接近0**

稀疏性限制：
使得神经元大部分的时间都是被抑制的状态

https://medium.com/@syoya/what-happens-in-sparse-autencoder-b9a5a69da5c6

使用L1正则化的效果

- MNIST数据集上epoch为100，batch大小为128，使用Adam优化器



实验结果

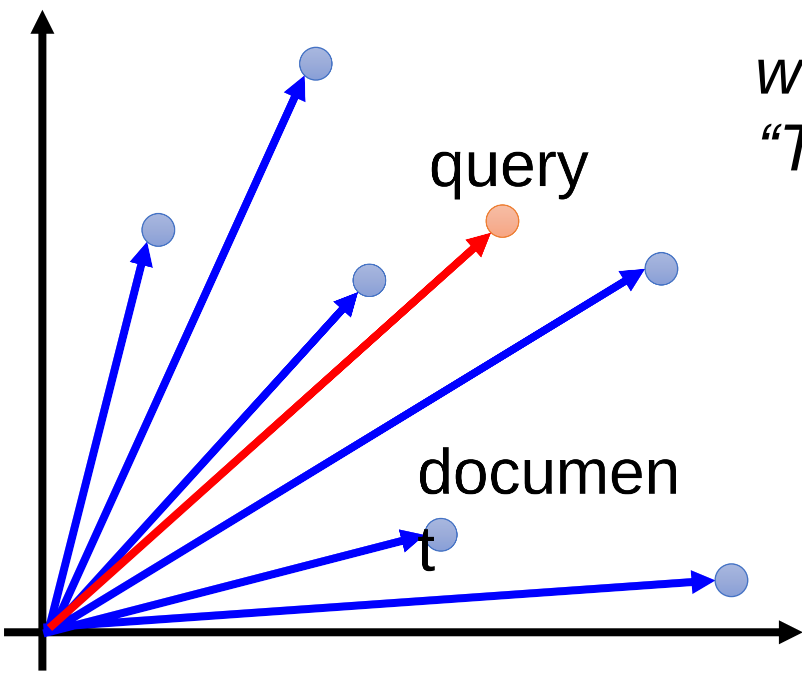
方法	最佳MSE损失
Simple	0.0318
Sparse	**0.0301**

虽然只是微小的改进，但证明稀疏自编码器比自编码器学到了更好的表示。

https://medium.com/@syoya/what-happens-in-sparse-autencoder-b9a5a69da5c6

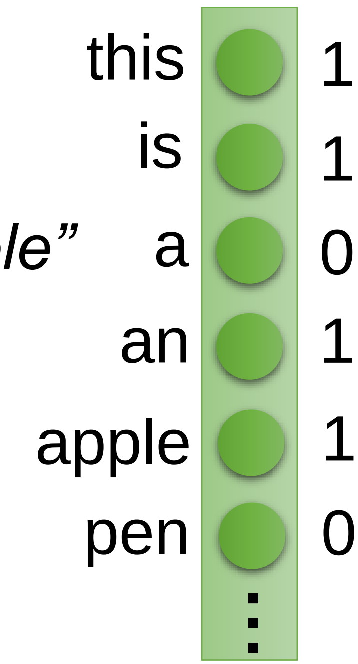
自编码器应用：Text Retrieval

Vector Space Model



Bag-of-word

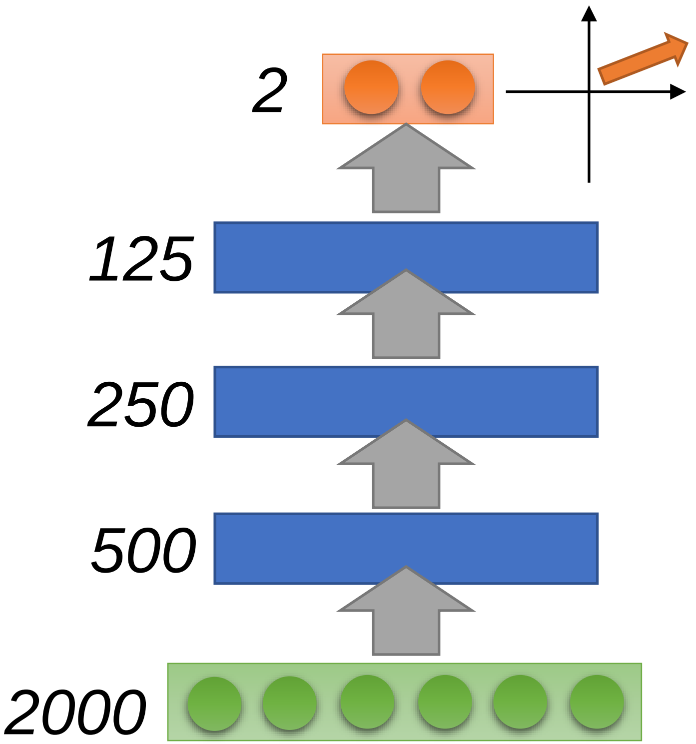
word string:
“This is an apple”



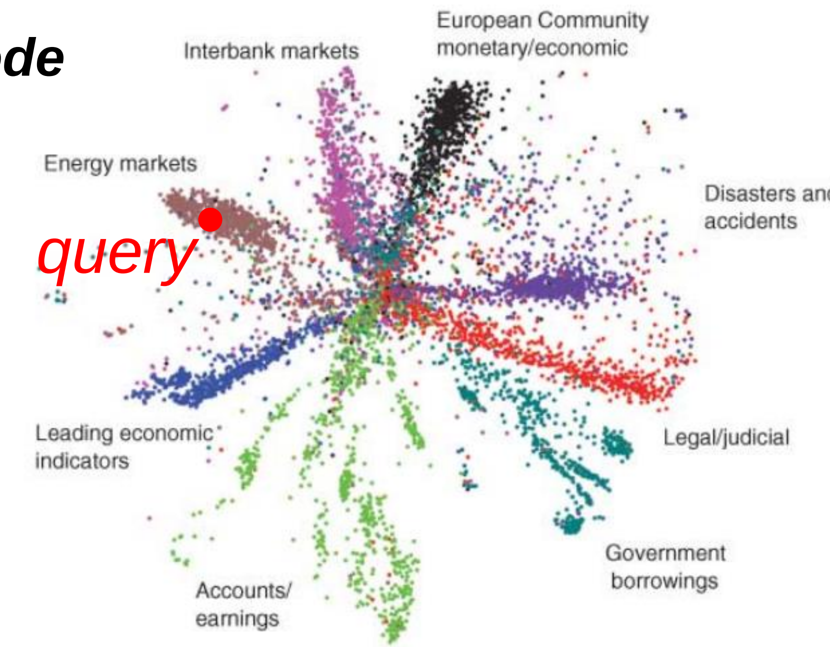
没有考虑语义

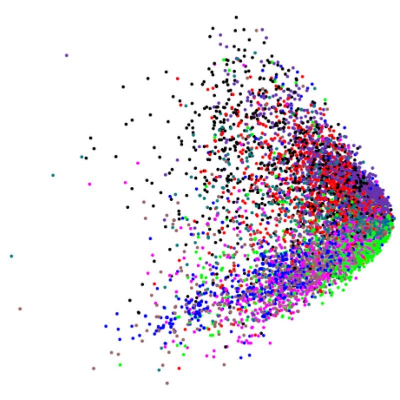
自编码器应用：Text Retrieval

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



Bag-of-word
(document or query)





LSA: project documents to 2 latent topics

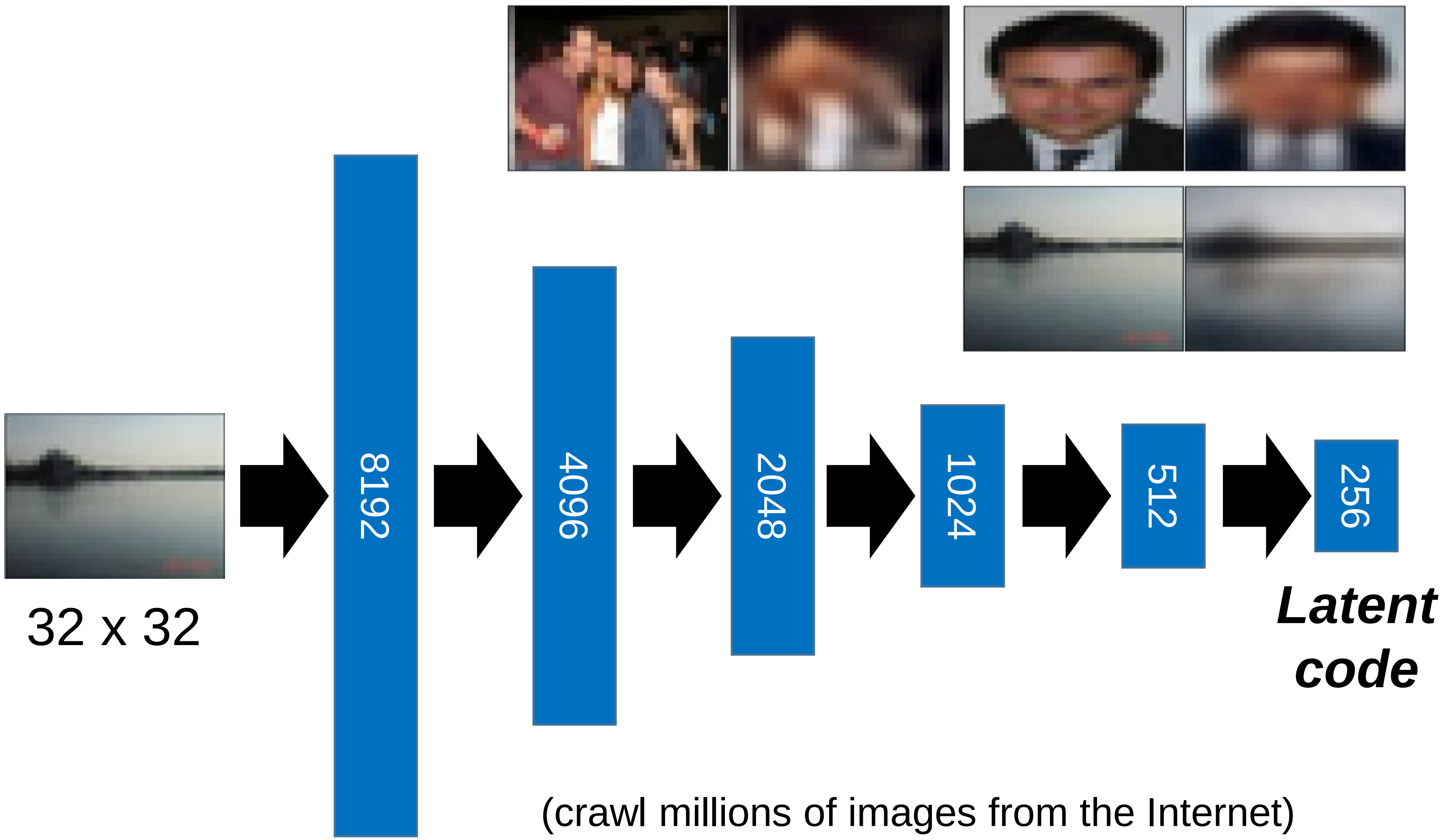
自编码器应用：Similar Image Search

原图像空间检索：Retrieved using Euclidean distance in pixel intensity space

(Images from Hinton’s slides on Coursera)

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

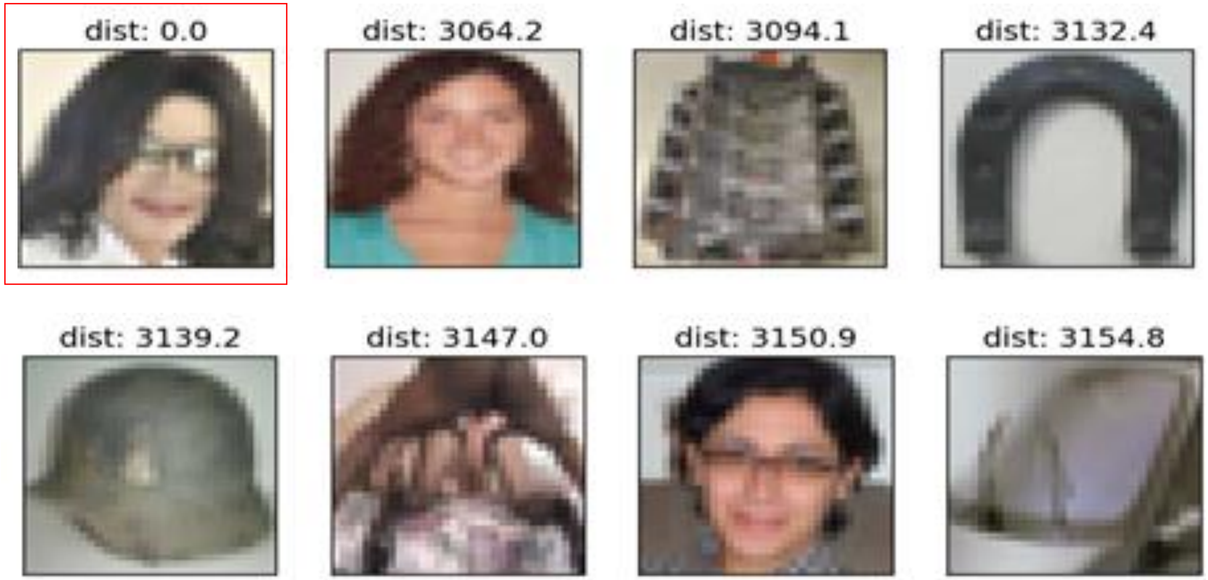
自编码器应用： Similar Image Search



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

自编码器应用： Similar Image Search

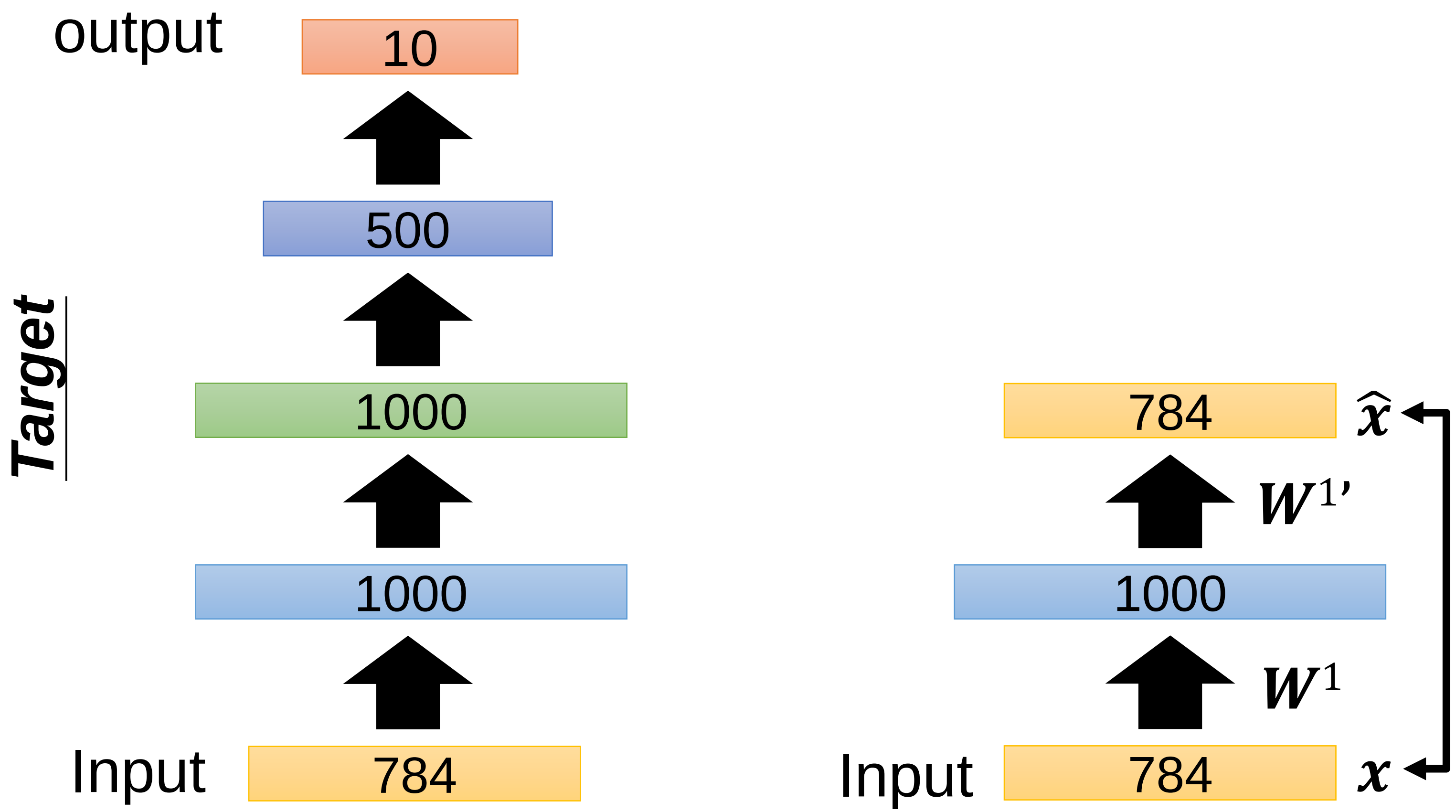
Retrieved using Euclidean distance in pixel intensity space



Retrieved using 256 codes

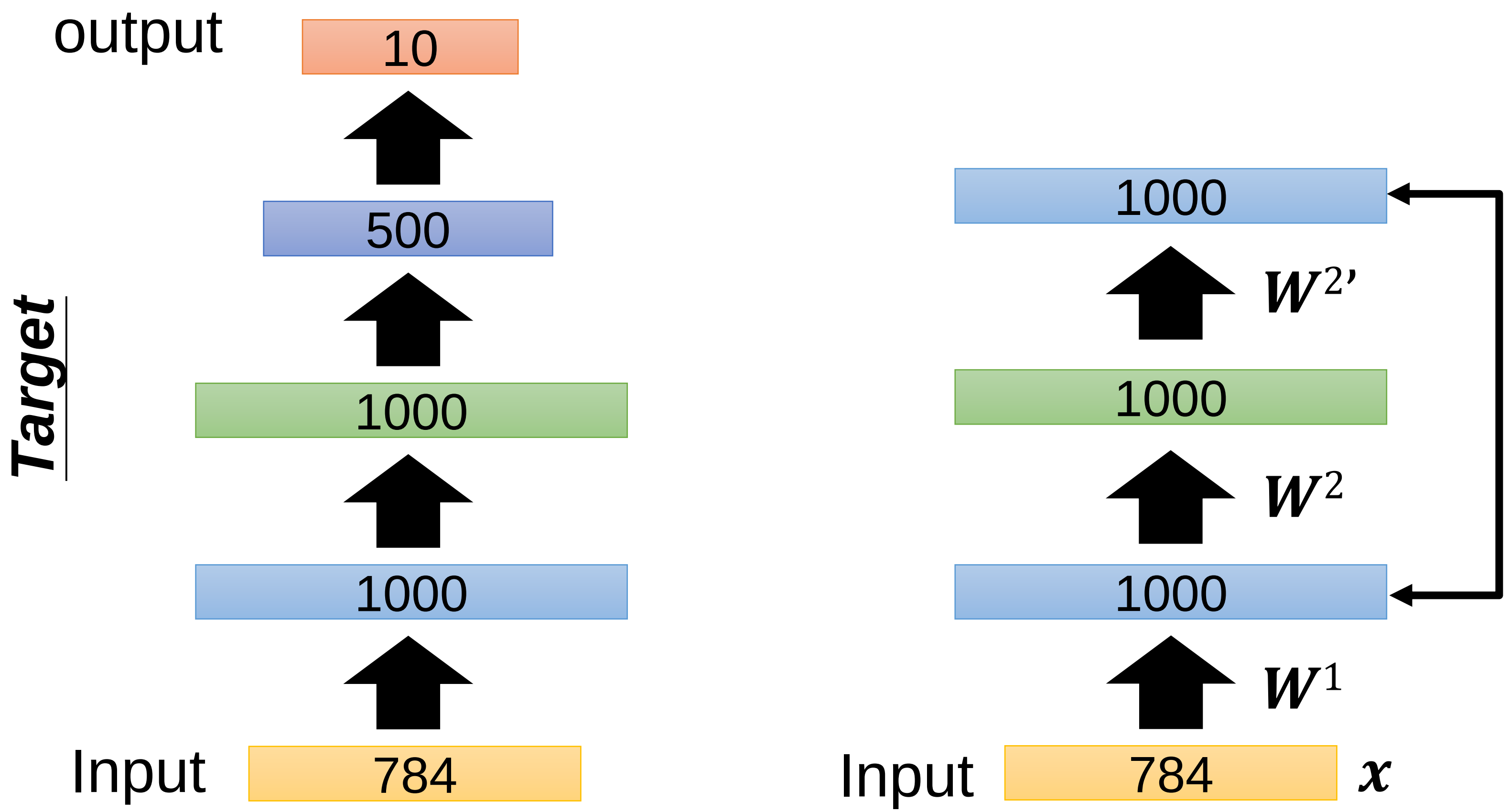
自编码器应用：预训练深度网络

- Greedy layer-wise pre-training *again*



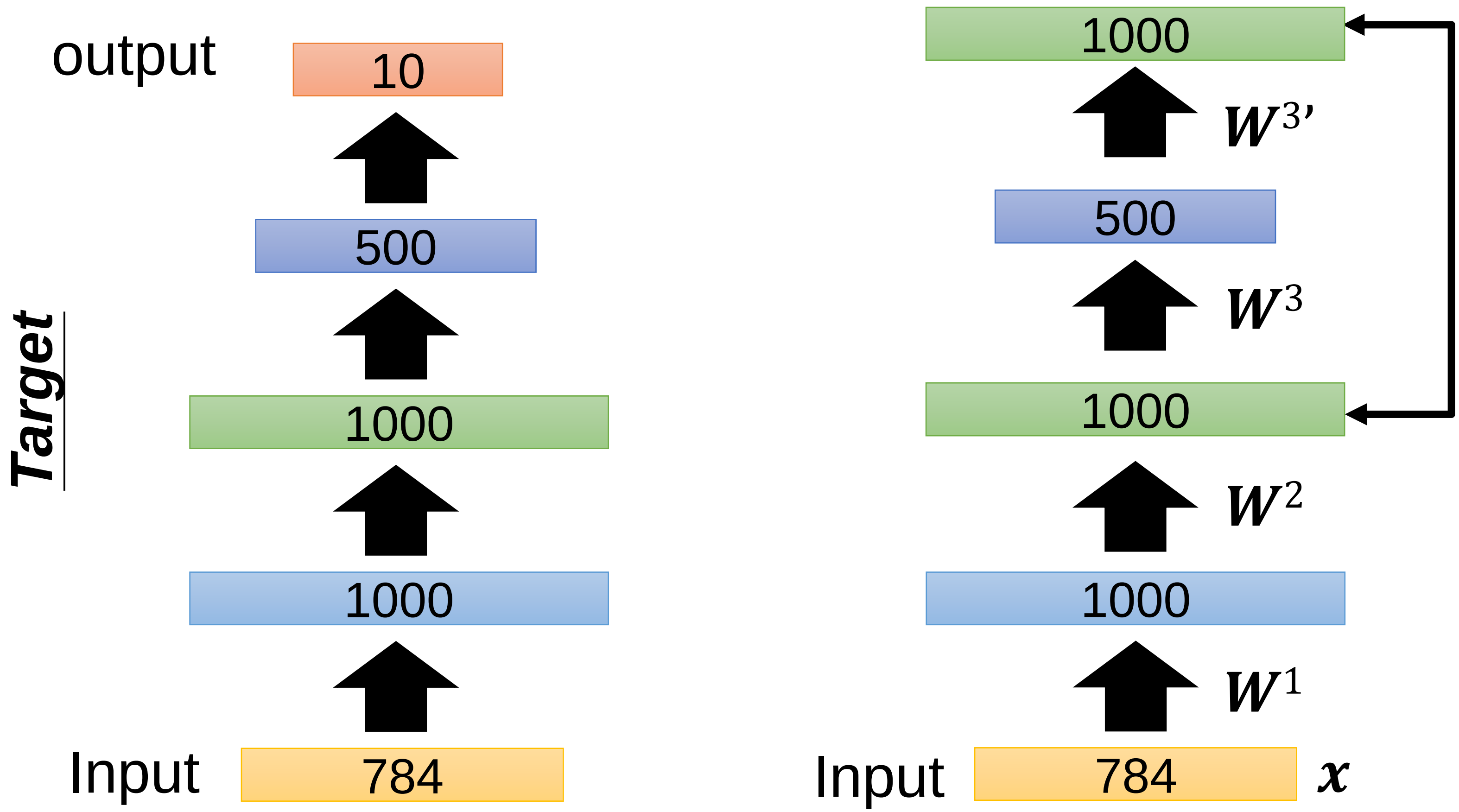
自编码器应用：预训练深度网络

- Greedy layer-wise pre-training *again*



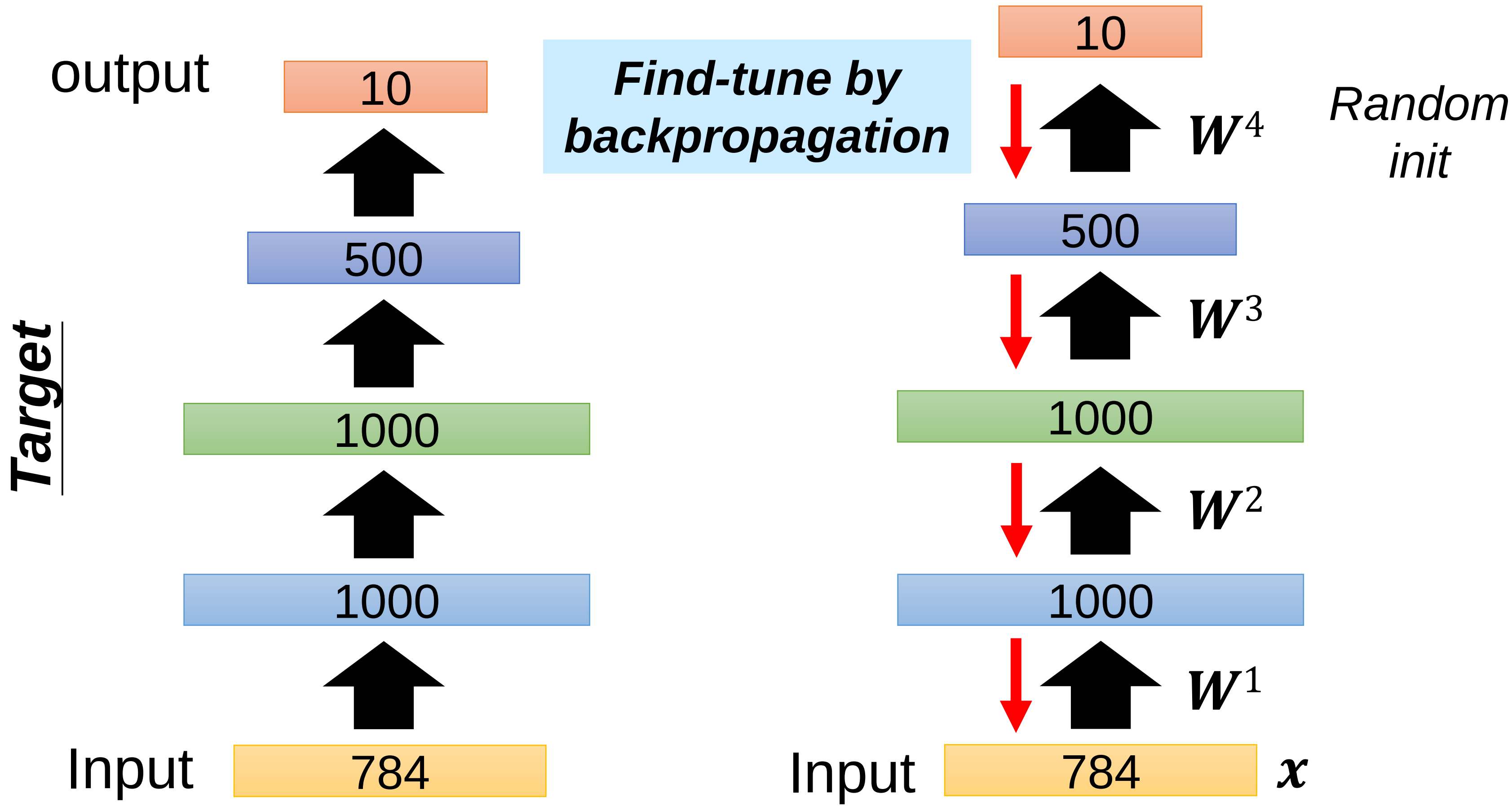
自编码器应用：预训练深度网络

- Greedy layer-wise pre-training *again*



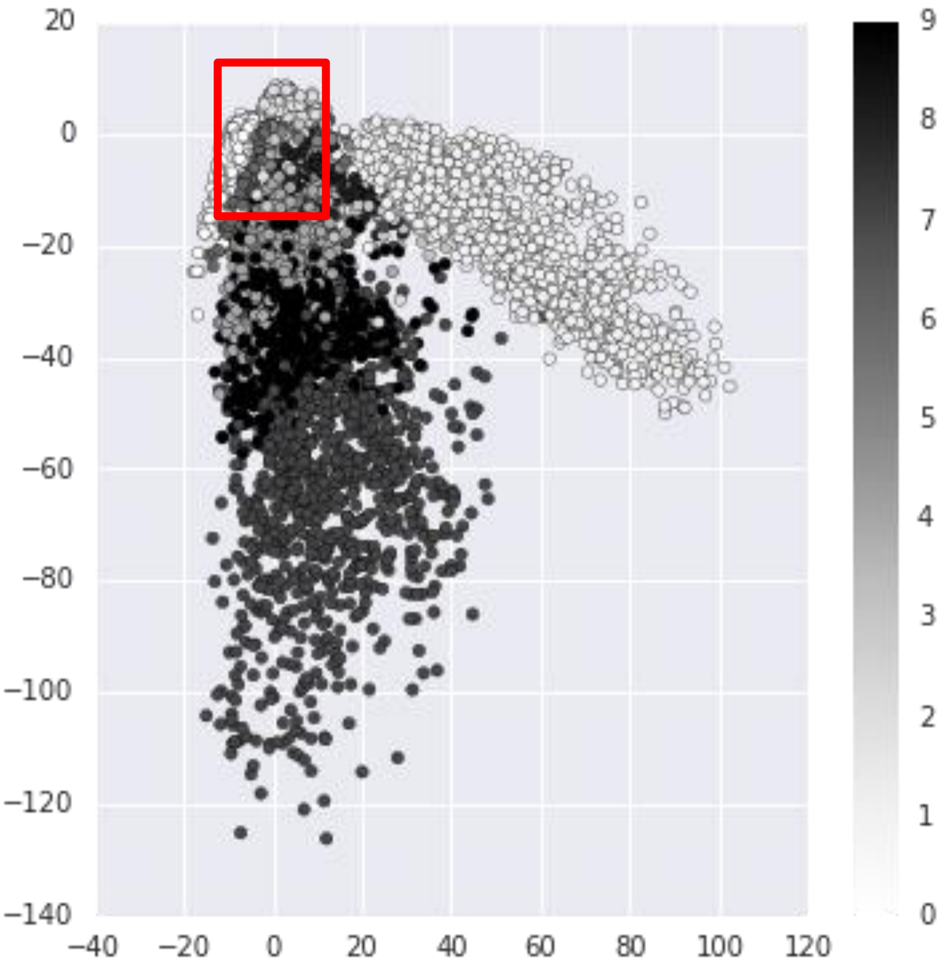
自编码器应用：预训练深度网络

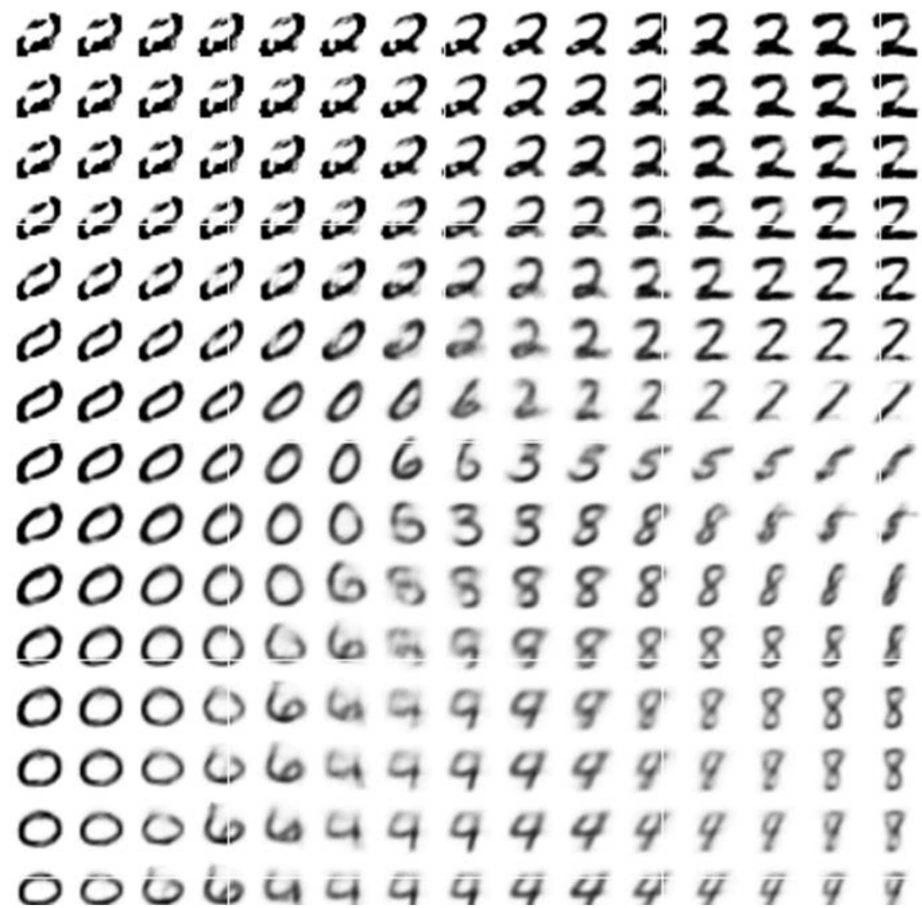
- Greedy layer-wise pre-training *again*



思考：AE是否有生成能力?

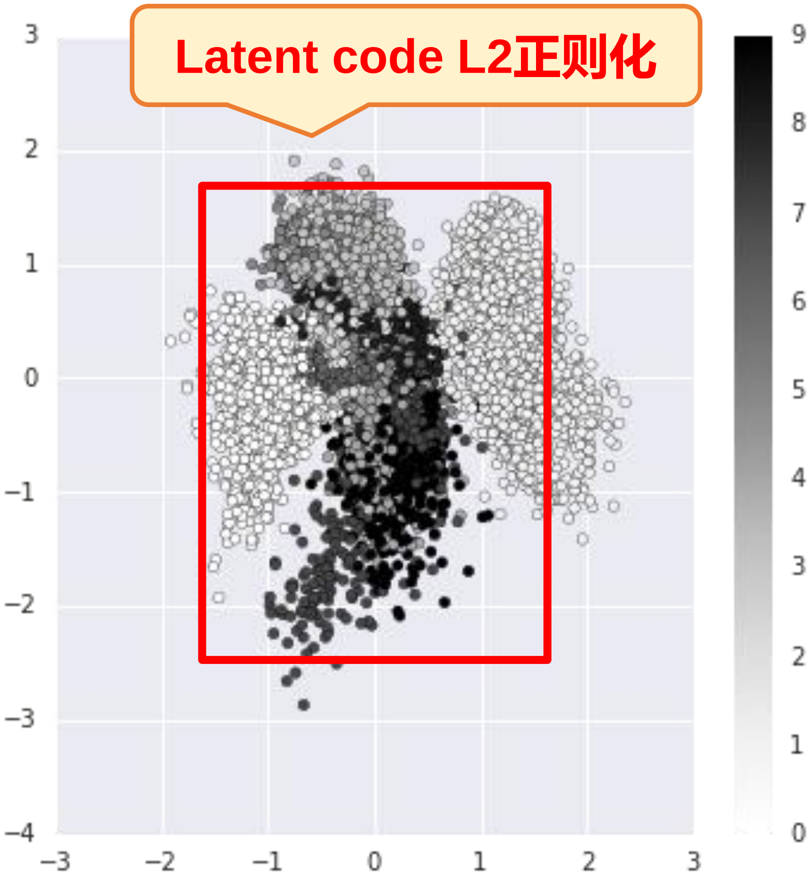
- *Can we use decoder to generate something?*

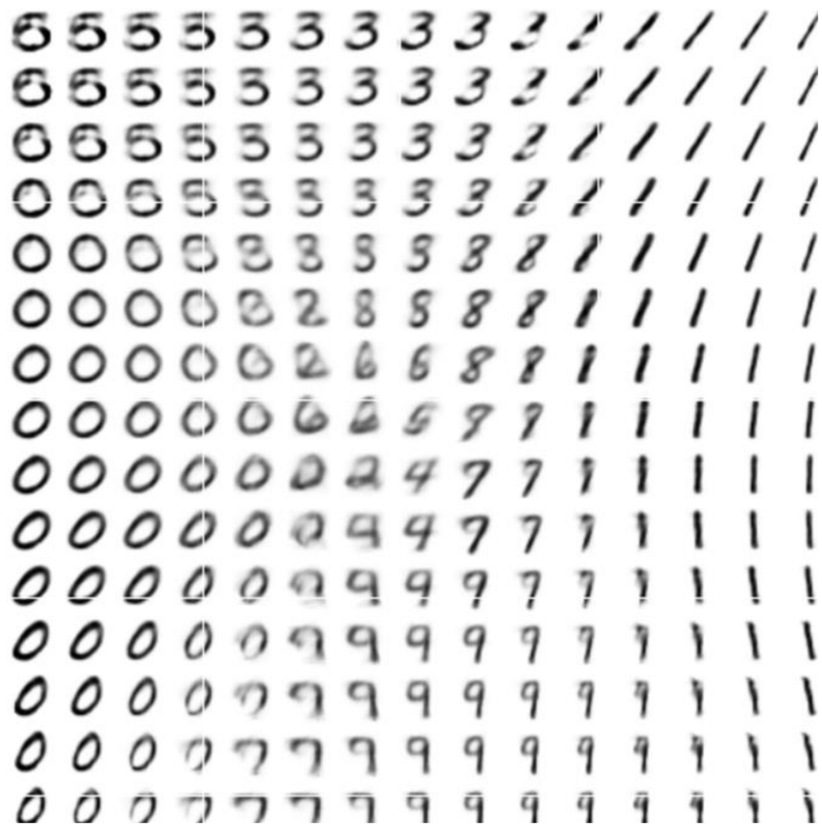




思考：AE是否有生成能力?

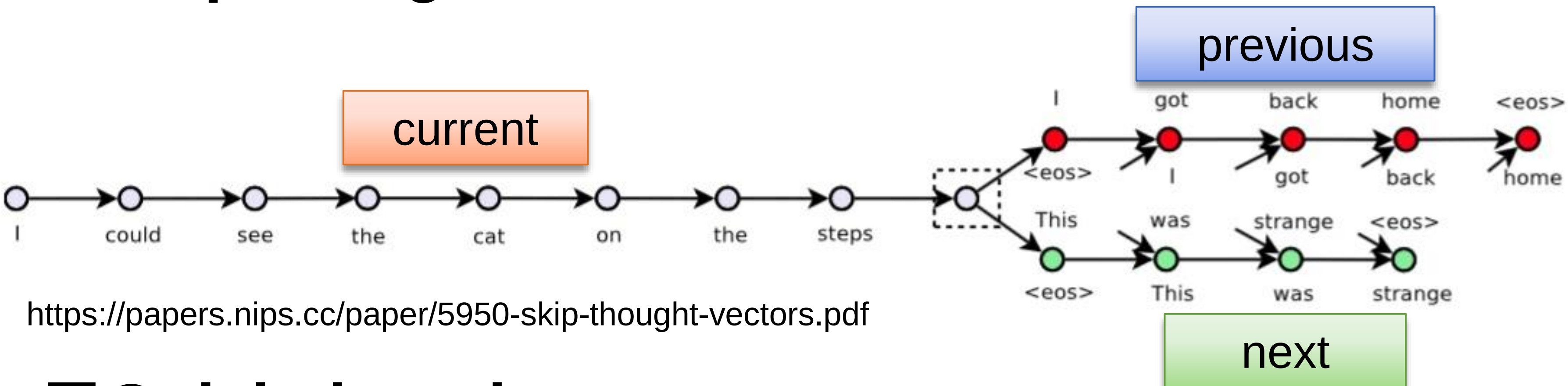
- *Can we use decoder to generate something?*





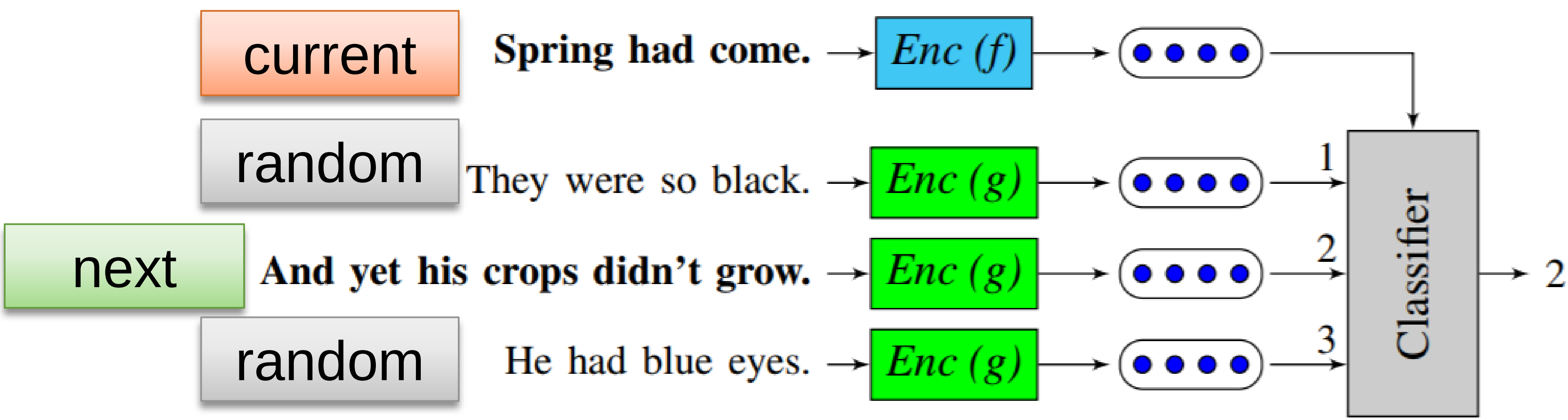
AE处理序列数据(Sequential Data)

☐ Skip thought　一段文本就是句子的一个序列



https://papers.nips.cc/paper/5950-skip-thought-vectors.pdf

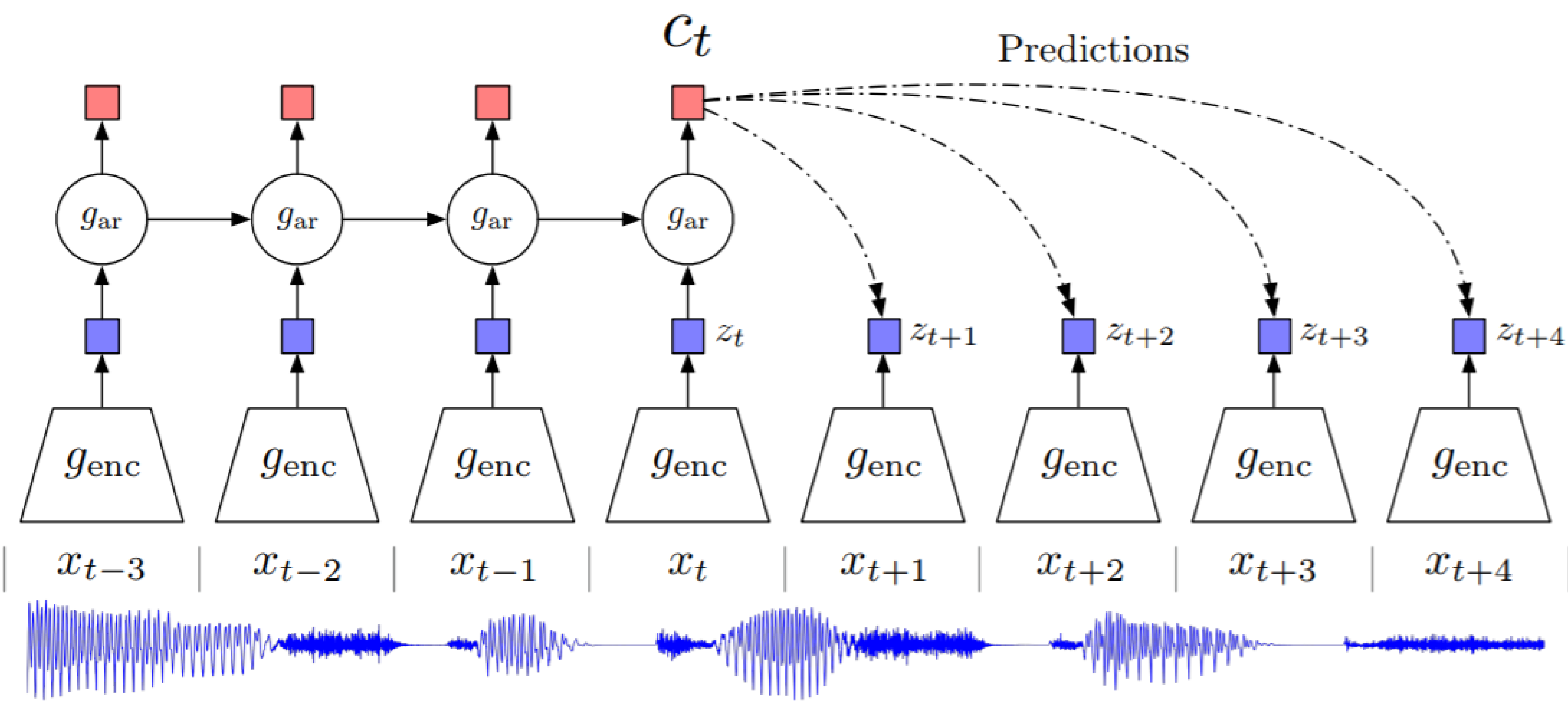
☐ Quick thought



https://arxiv.org/pdf/1803.02893.pdf

AE处理序列数据(Sequential Data)

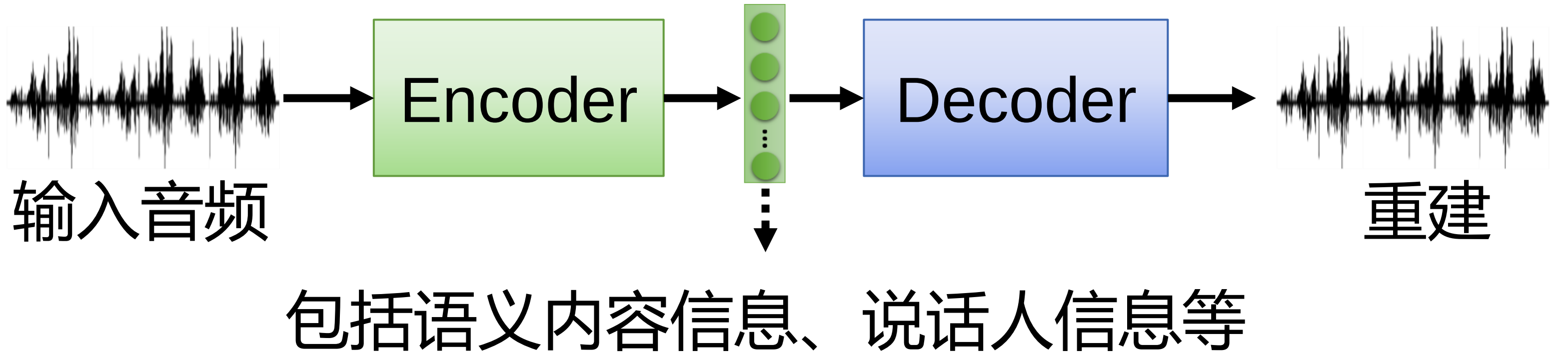
- 对比预测编码 (Contrastive Predictive Coding, CPC)

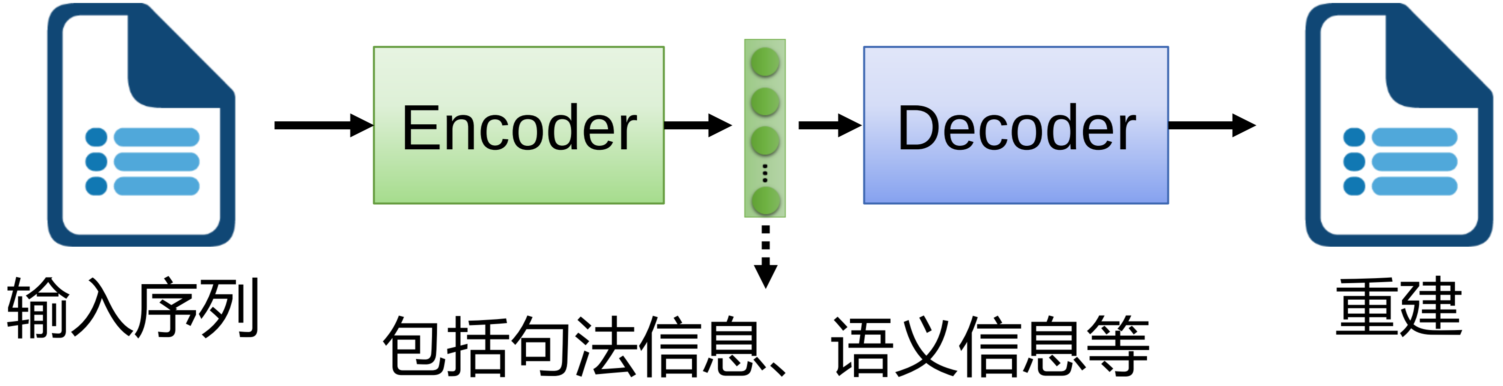


https://arxiv.org/pdf/1807.03748.pdf

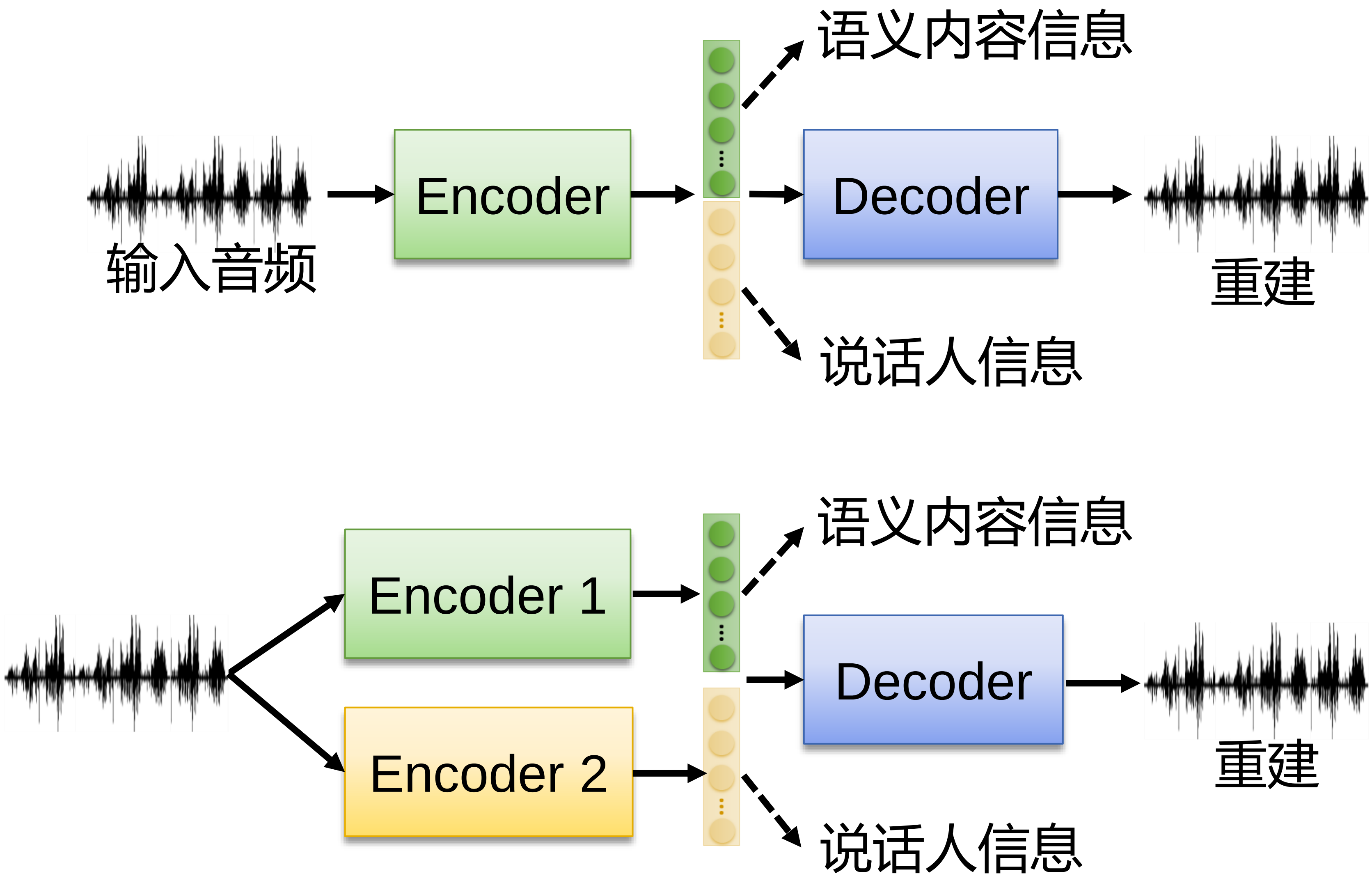
AE特征解耦(Feature Disentangle)

◻一个对象包含多个方面信息

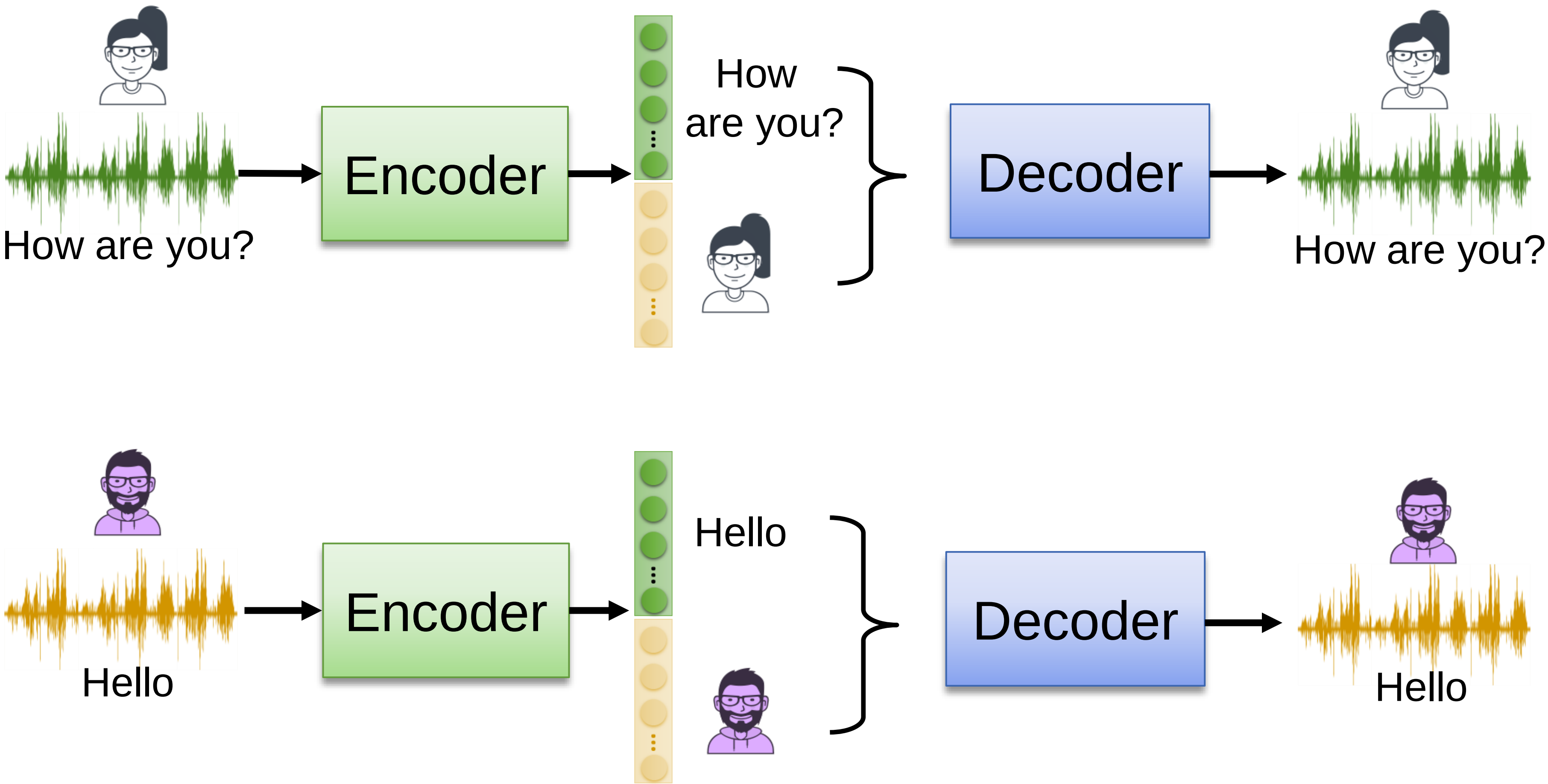




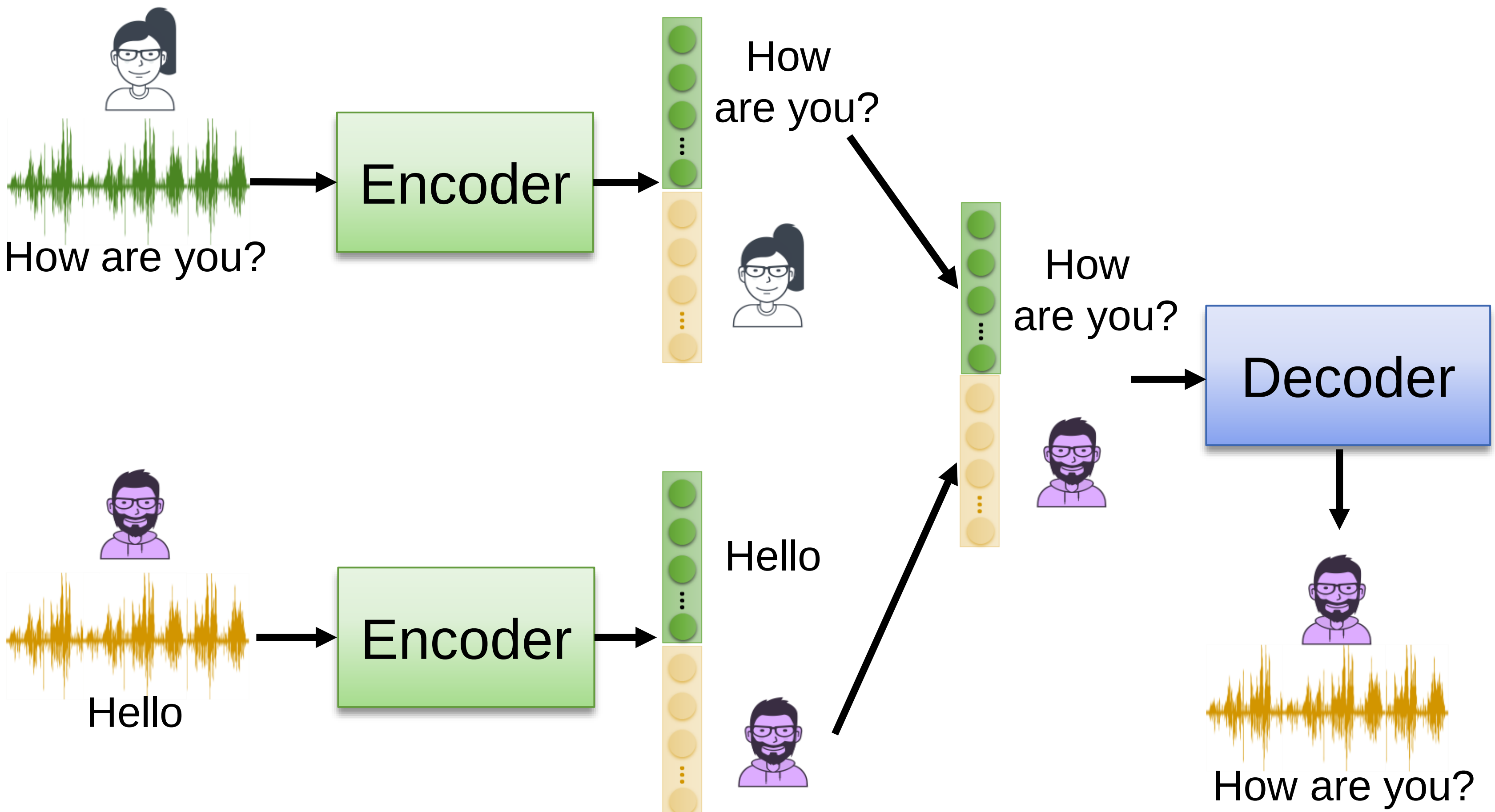
AE特征解耦(Feature Disentangle)



特征解耦示例：语音对话

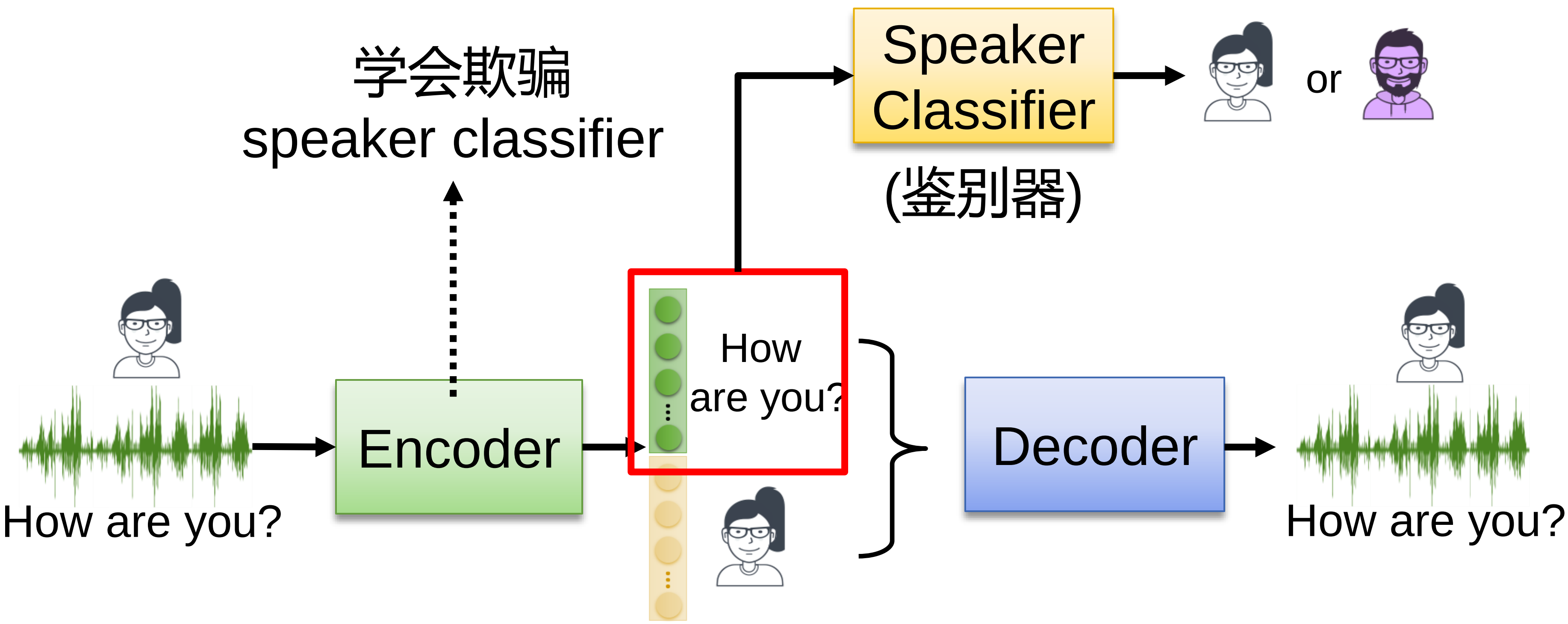

How are you?
Encoder
How
are you?
Decoder
How are you?
Hello
Encoder
Hello
Decoder
Hello

AE特征解耦(Feature Disentangle)



AE特征解耦(Feature Disentangle)

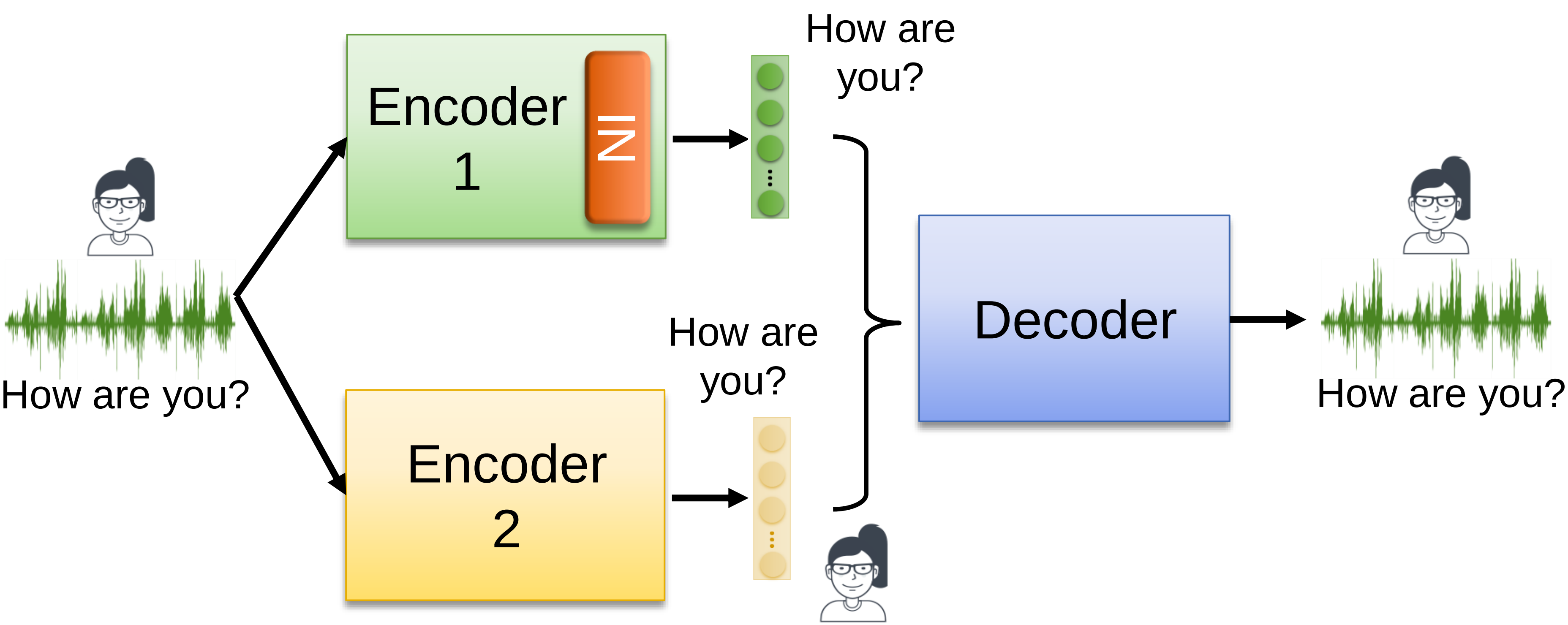
□对抗训练



说话人分类器和编码器迭代学习

AE特征解耦(Feature Disentangle)

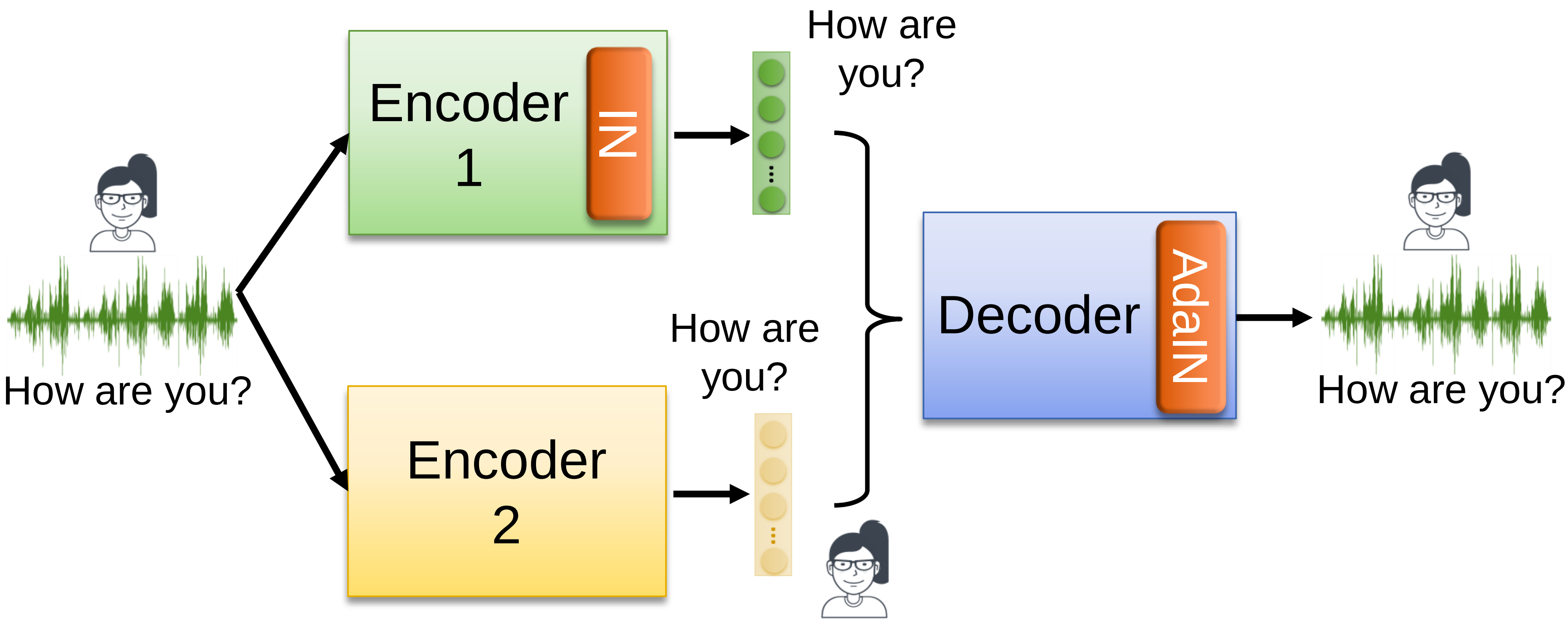
◻设计的网络架构



IN = 实例归一化 (instance normalization) ⇒ (移除全局信息)

AE特征解耦(Feature Disentangle)

☐ 设计的网络架构

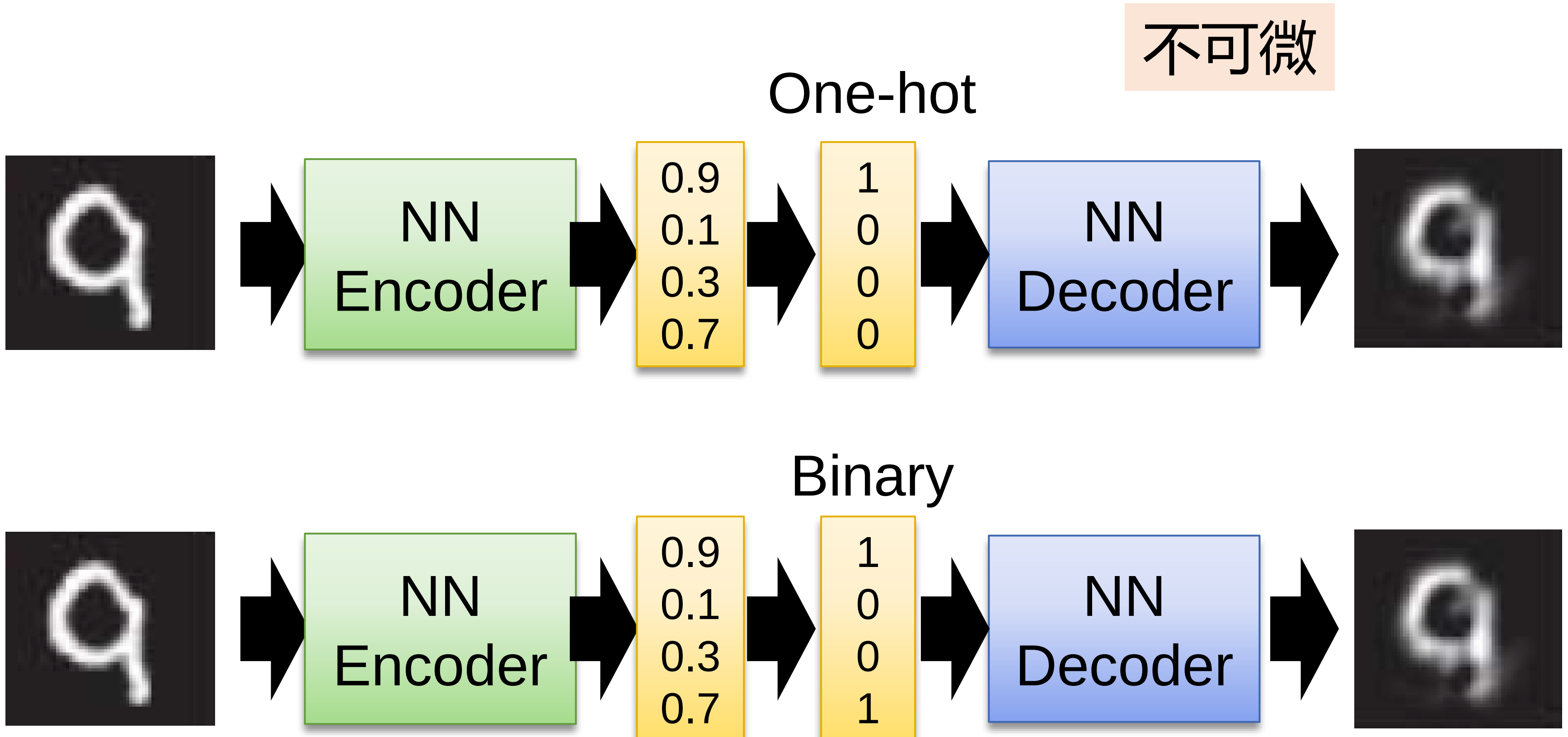


IN = 实例归一化 (instance normalization) ⇨ (移除全局信息)

AdaIN = 自适应实例归一化 (adaptive instance normalization) ⇨ (只影响全局信息)

Latent code离散表示

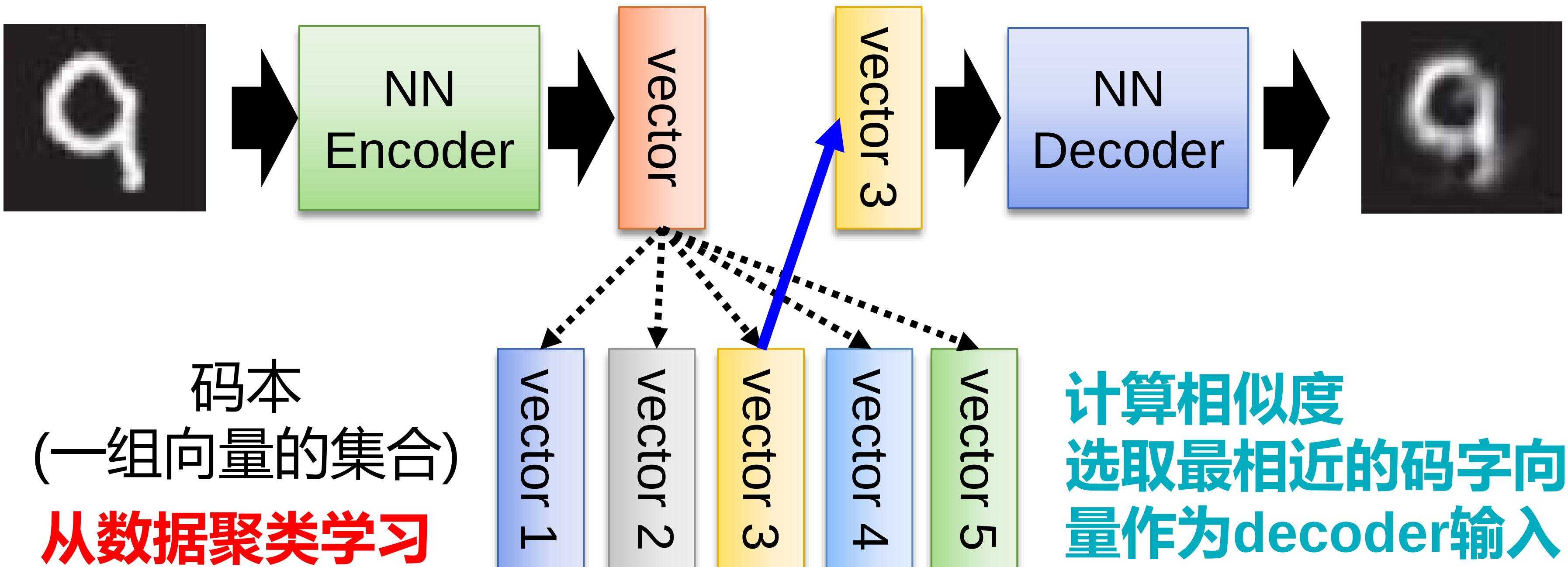
- 更易于解释AE或聚类



https://arxiv.org/pdf/1611.01144.pdf

Latent code离散表示

- 矢量量化自编码器(Vector Quantized Variational Auto-encoder, VQVAE)



https://arxiv.org/abs/1711.00937