

# 分类模型

## 第二部分 非线性分类模型

### BP神经网络

张朝晖

2018-2019学年 20181009



## 1. 分类模块

### 产生式分类模型

A. 贝叶斯分类模型

### 判别式分类模型

#### 线性分类模型

- B. Fisher判别分类
- C. 感知器分类模型
- D. 大间隔分类模型(线性SVM)

#### 非线性分类模型

- E. 核SVM(非线性SVM)
- F. 核Fisher判别分类
- G. 神经网络

### 其它分类模型

- H. KNN分类模型
- I. 决策树分类模型
- J. Logistic回归
- K. Softmax回归

## 2. 聚类模块

- L. K-均值聚类
- M. 高斯混合聚类
- N. DBSCAN聚类
- O. 层次聚类

## 3. 回归模块

- P. KNN回归
- Q. 回归树
- R. 最小二乘线性回归
- S. 岭回归
- T. LASSO回归

## 4. 集成学习

- U. Bagging
- V. 随机森林
- W. Boosting

## 5. 特征工程

- X. 主成分分析(PCA)
- ...

## 6. 评价模块

混淆矩阵(及其相关指标)、ROC曲线、交叉验证

# 人工神经网络

## --前馈神经网络与BP算法

# 关键词：

人工神经网络

前馈神经网络、人工神经元、激活函数

BP神经网络：梯度下降法、导数链式法则、误差反向传播

自编码器

监督式学习、非监督式学习

分类、回归、表示学习

多层BP神经网络学习

第一阶段：基于逐层表示学习的参数初始化；

第二阶段：基于BP算法的参数精调

# 主要内容

1. 人工神经网络基本知识、神经元与感知器
2. 前馈神经网络、多层感知器、及非线性分类
3. BP神经网络
4. 基于前馈神经网络的自编码器(Autoencoders)

基本自编码器

栈式自编码器(Stacked Auto-Encoder)

稀疏自编码器(Sparse Auto-Encoder)

去噪自编码器(Denoising Auto-Encoder)

压缩自编码器(Contrative Auto-Encoder)

...

## (1)什么是自编码器

Auto-encoder(自编码器)

--又称Auto-associator(自联想器)

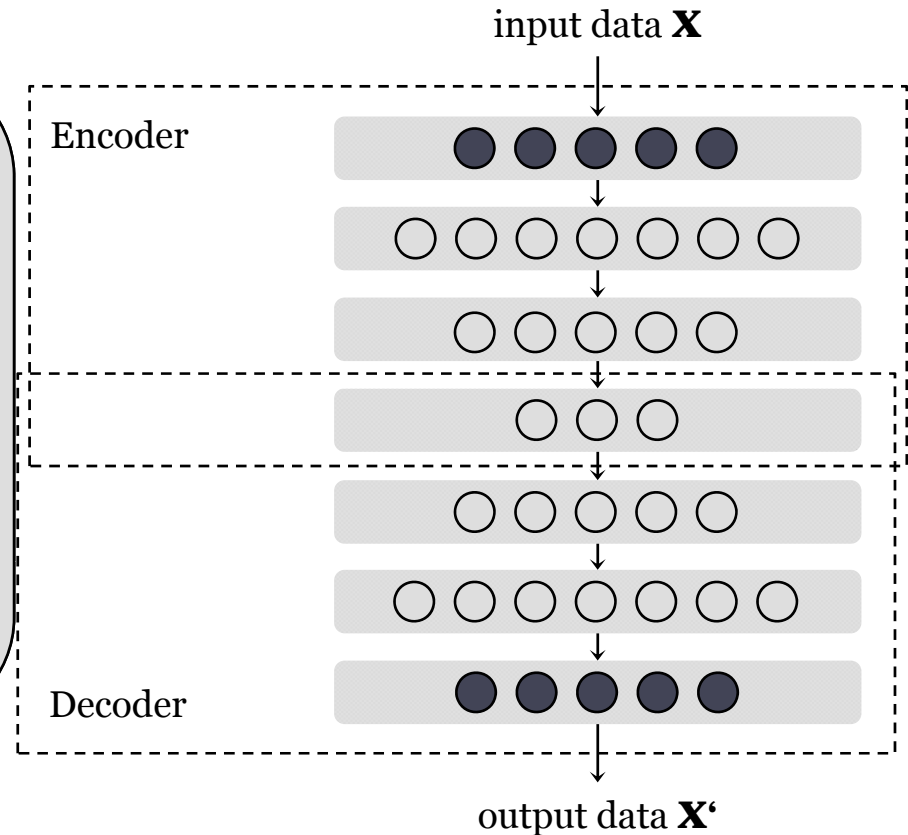
- Unsupervised learning
- A neural network that make the outputs approximate the inputs

# Autoencoders

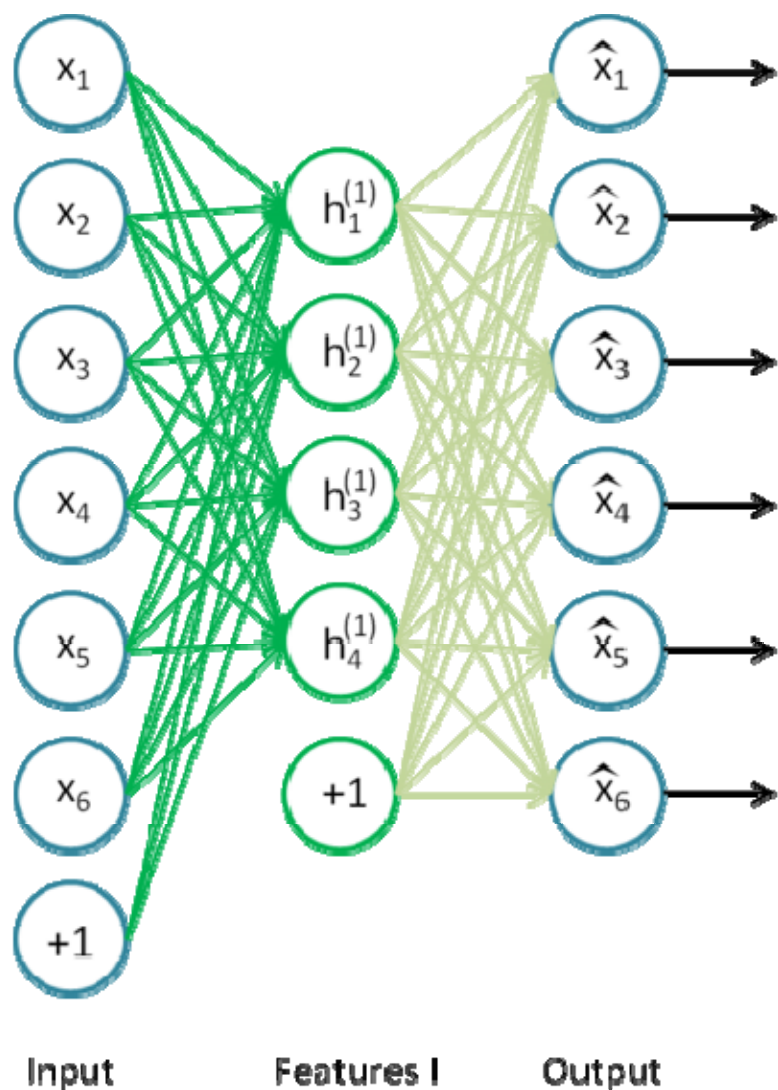
- Symmetric topology
- **Information bottleneck**
- **Encoder**
- **Decoder**

## Definition (*autoencoder*)

• Autoencoders are *deep networks* with a *symmetric topology* and an odd number of hidden layers, containing a *encoder*, a **low** dimensional representation and a *decoder*.



例：包含一个隐含层的Auto-encoders(AE)



三种功能层：

- 输入层
- 编码层、或表示层
- 输出层、或重构层



## (2) 基于BP算法的自编码器的训练

面向特征提取的预训练阶段

Use unlabeled instances to pre-train an autoencoder

- Layer-wise training
- Unsupervised training

结合分类问题需要的参数精调

- Fine-tuning (to makes the autoencoder discriminative)
- Supervised training

# Autoencoders

## Training

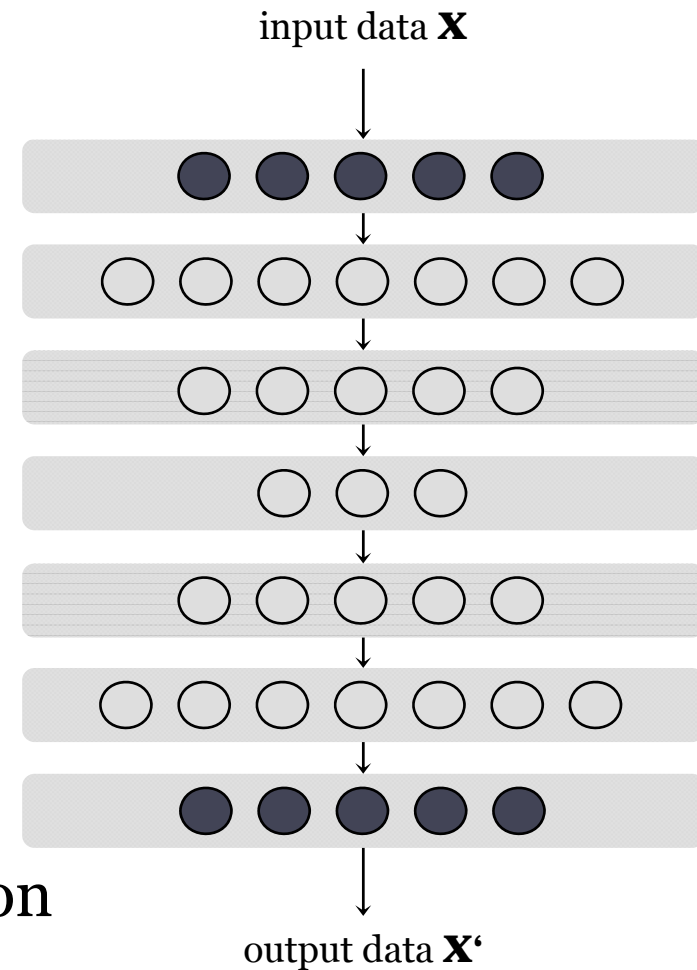
Backpropagation

**Problem:** Deep Network

- Very slow training
- Maybe bad solution

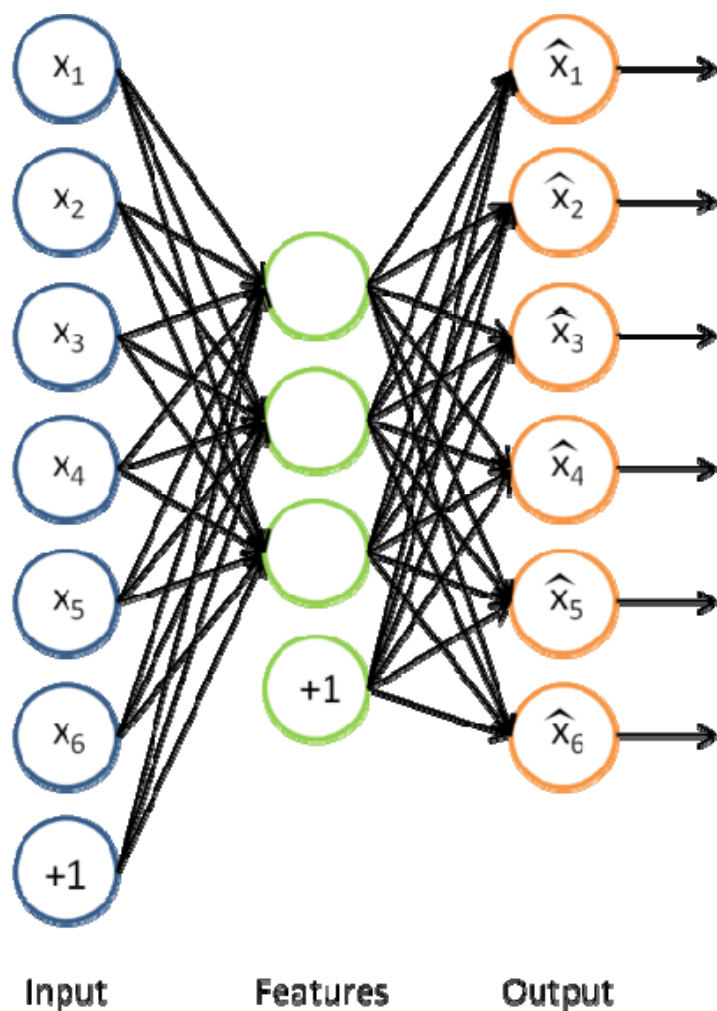
**Idea:** Initialize close to a good solution

- Pretraining
- Restricted Boltzmann Machines

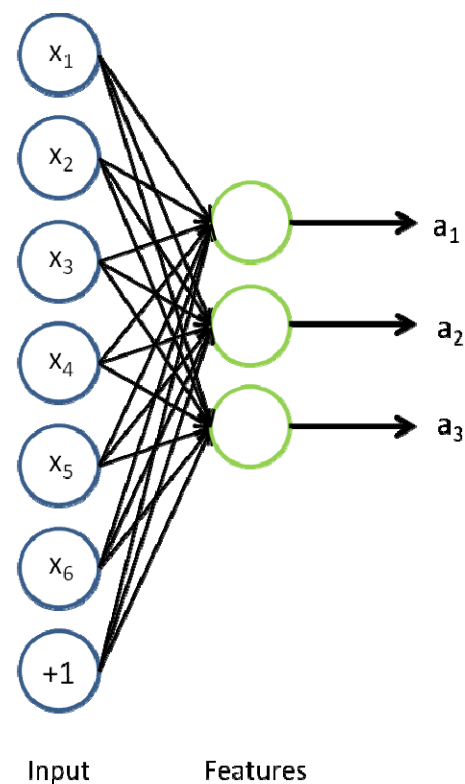
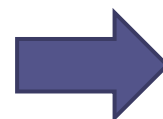


# Feature Extraction

无监督式训练

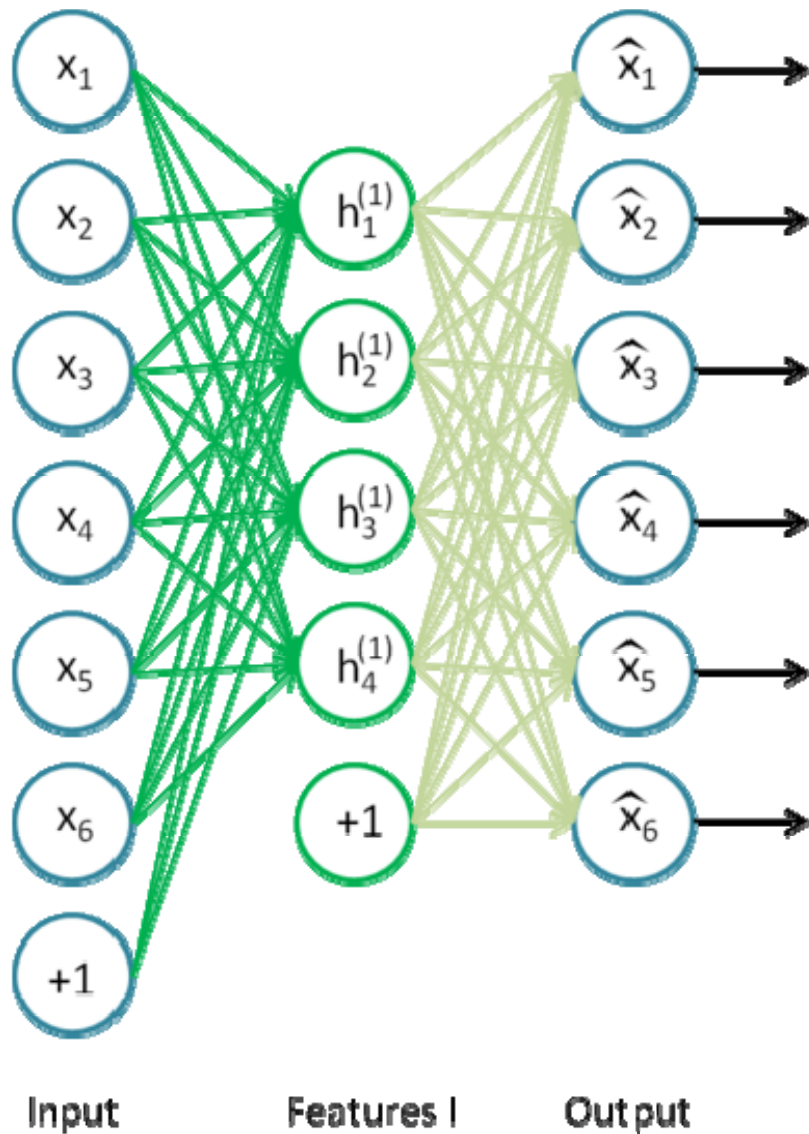


无监督式训练

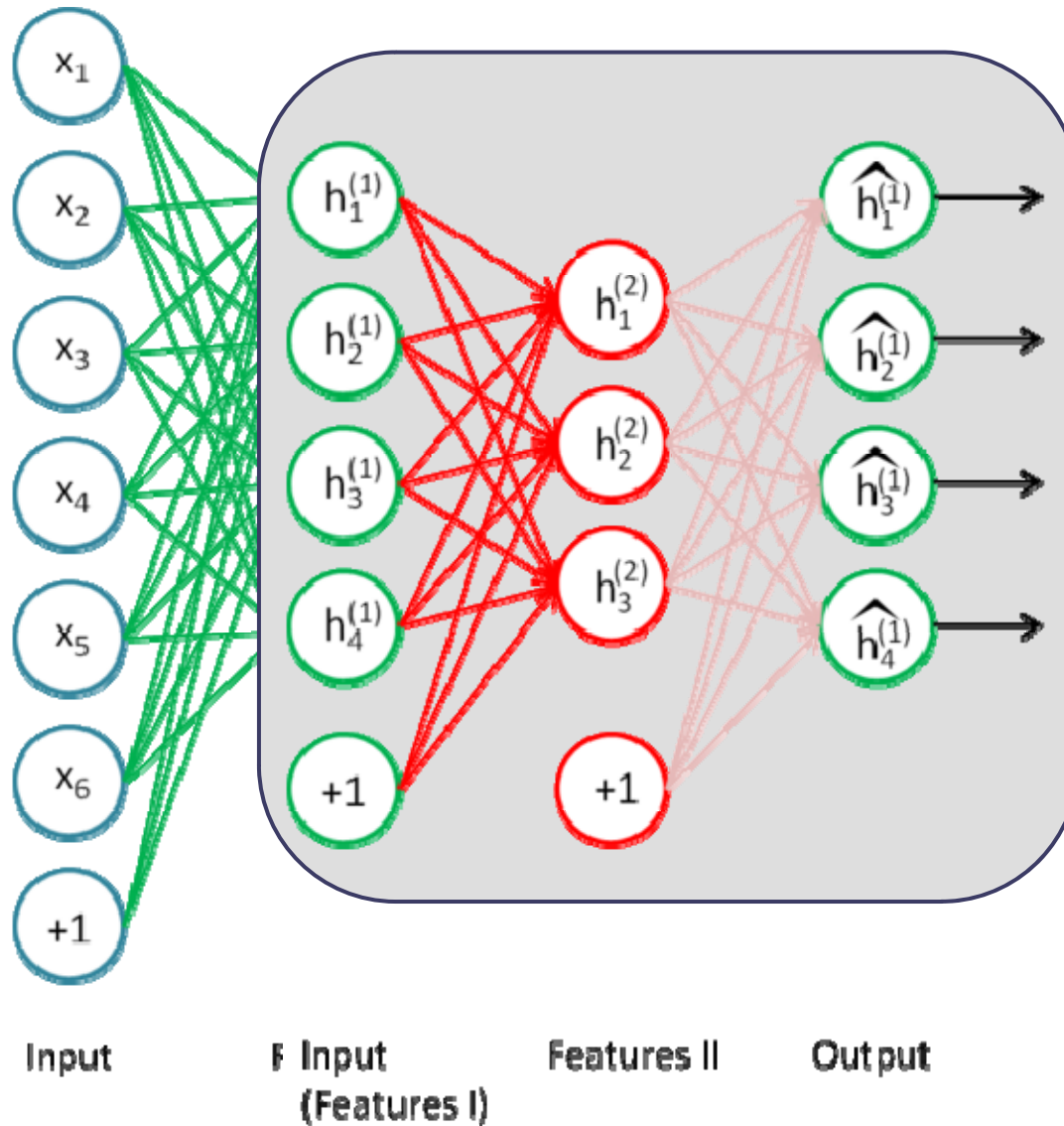


特征提取

# Stacked Auto-Encoders(SAE)



# Stacked Auto-Encoders(SAE)- 续1

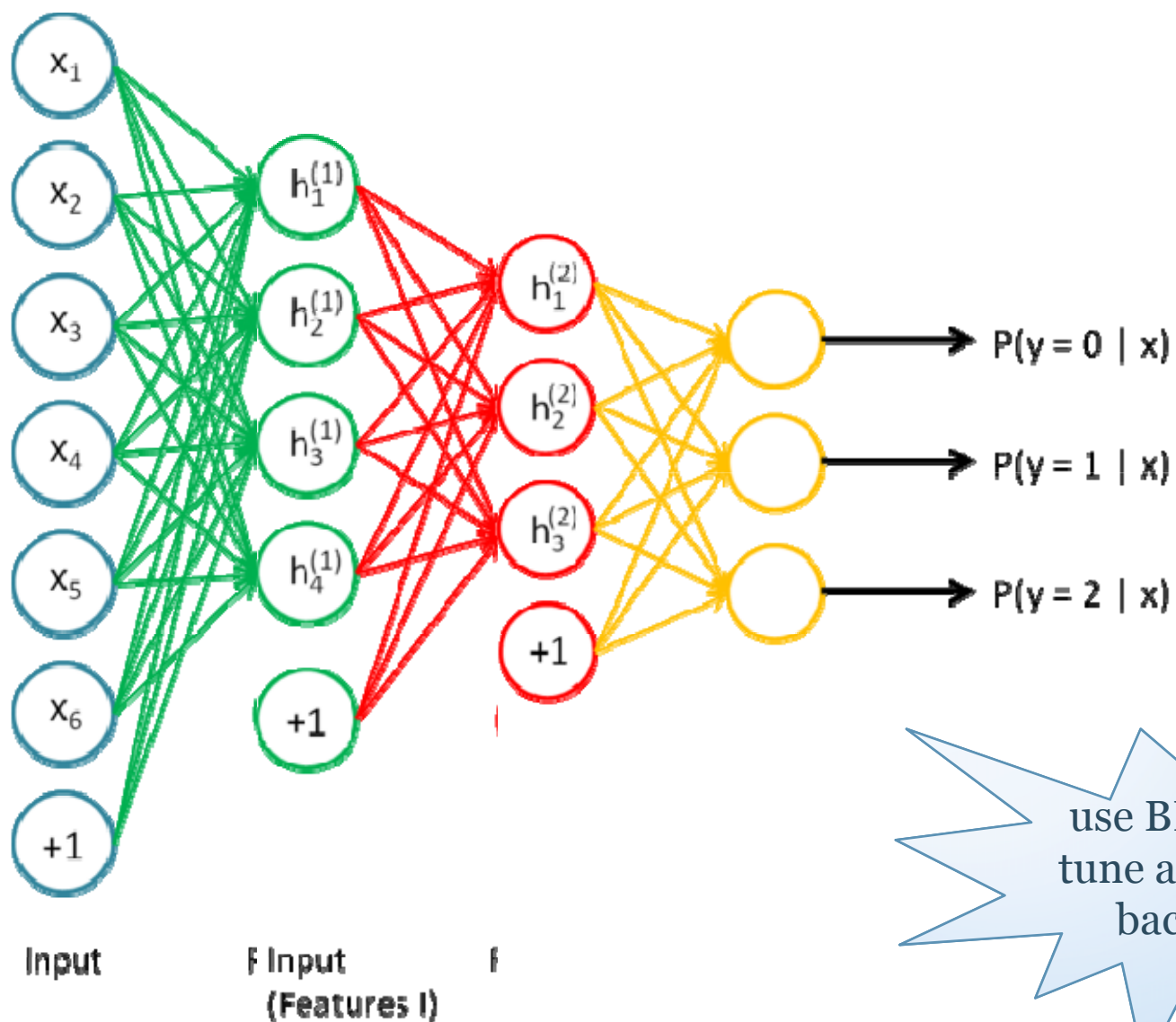


➤ 无监督式训练

➤ 更高级特征提取

# Stacked Autoencoders- 续2

## 监督式训练



如何将自编码器  
与分类需求结合，  
训练一个分类模  
型？

--参数精调

监督式训练  
误差反向传播

use BP to fine-  
tune all weights  
backward

### (3) 自编码器的典型形式

基于RBM的自编码器

Stacked Autoencoder

Denoising Autoencoder

Sparse Autoencoder

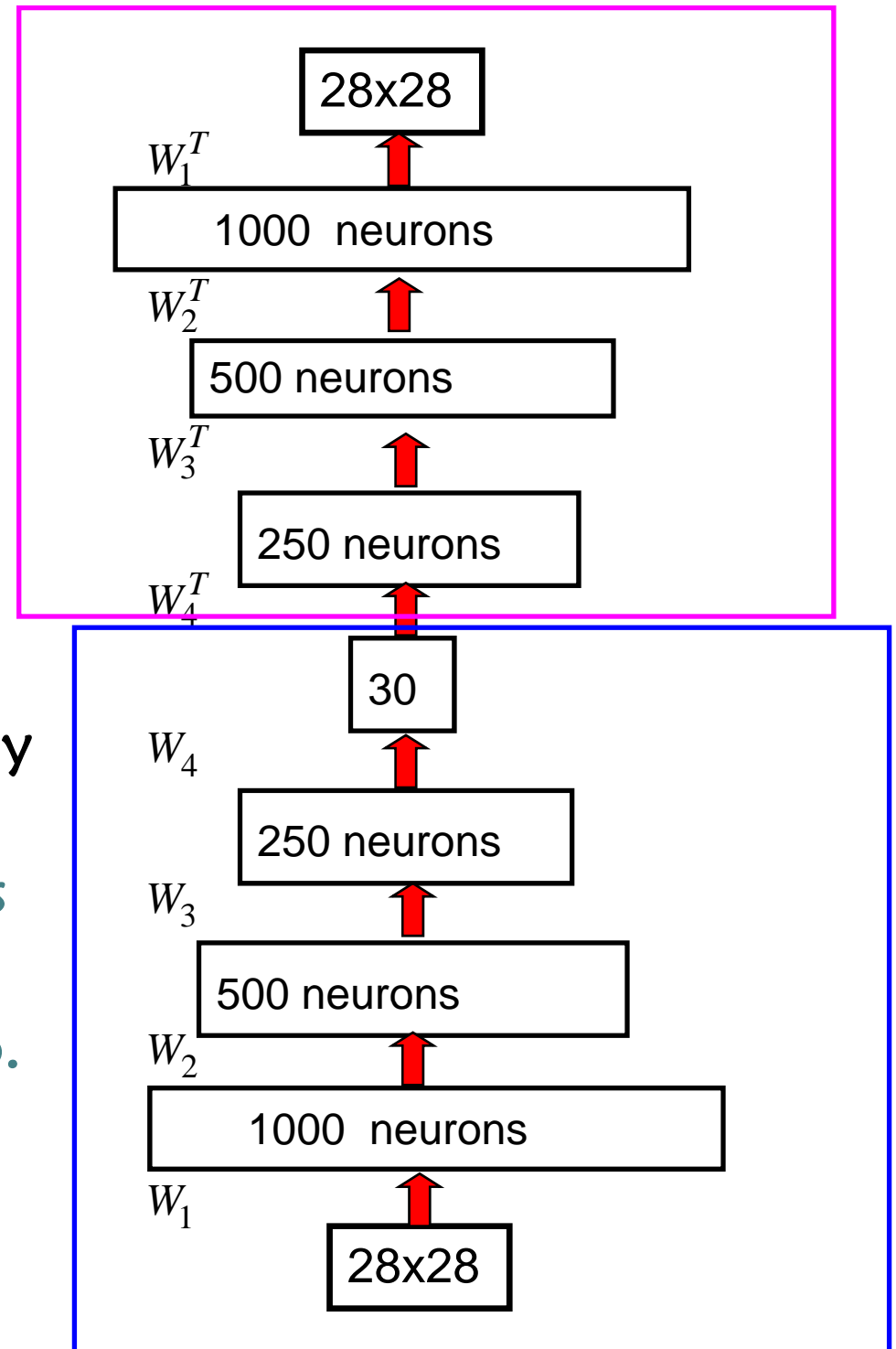
Non-Negative Autoencoder

及其上述各种组合或其他变种

# Deep Autoencoders

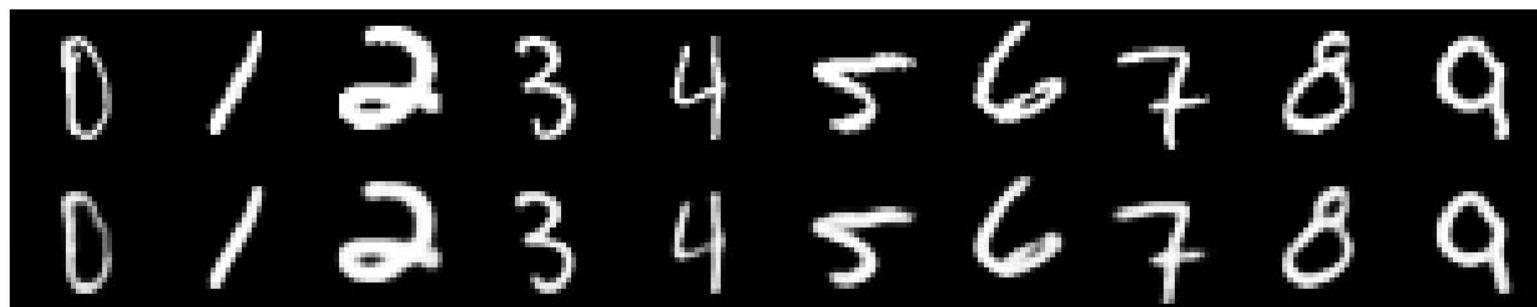
(with Ruslan Salakhutdinov)

- They always looked like a really nice way to do non-linear dimensionality reduction:
  - But it is **very** difficult to optimize deep autoencoders using backpropagation.
- We now have a much better way to optimize them:
  - First train a stack of 4 RBM's
  - Then “unroll” them.
  - Then fine-tune with backprop.





A comparison of methods for compressing digit images to 30 real numbers.



real  
data

30-D  
deep auto



30-D  
PCA

## (4)应用

特征提取(非线性PCA)

高维数据的降维、低维可视化;

样本数据结构描述

缺失数据复原

图像应用:

图像滤波、复原、超分辨率重构

分类、基于内容检索、图像标注

推荐

....

## 自编码器目标函数的构造：

$$E(W, b) = L(W, b) + \lambda \Omega(W)$$

$$\begin{cases} L(W, b) & \text{--关于训练样本集的学习能力} \\ \Omega(W) & \text{--正则项} \end{cases}$$

$L(W, b)$ --重构误差的平均平方和(MSE)

$$L(W, b) = \frac{1}{2N} \sum_{i=1}^N \|Y(x_i; W, b) - \mathbf{x}_i\|^2$$

$\Omega(W)$ --正则项

$$\Omega(W) = \|W\|_F^2 = \sum_{l,i,j} (\omega_{ij}^l)^2 \text{ --起权重衰减作用}$$

$$E(W, b) = L(W, b) + \lambda \Omega(W)$$

自编码器的学习： $[\hat{W}, \hat{b}] = \arg \min_{W, b} E(W, b | X)$

## 稀疏自编码器 (*sparse autoencoder*)

编码层节点数目明显高于其输入层节点数

编码层得到的特征向量是关于输入向量的稀疏表示

### 稀疏自编码器目标函数的构造：

$$E_{\text{sparse}}(\mathbf{W}, \mathbf{b}) = E(\mathbf{W}, \mathbf{b}) + \beta Q(\hat{\boldsymbol{\rho}})$$

$$= L(\mathbf{W}, \mathbf{b}) + \lambda \Omega(\mathbf{W}) + \beta Q(\hat{\boldsymbol{\rho}}(\mathbf{W}, \mathbf{b}))$$

$$= \frac{1}{2N} \sum_{i=1}^N \|\mathbf{Y}(\mathbf{x}_i; \mathbf{W}, \mathbf{b}) - \mathbf{x}_i\|^2 + \lambda \|\mathbf{W}\|_F^2 + \beta \sum_j KL(\boldsymbol{\rho} \parallel \hat{\boldsymbol{\rho}}_j)$$

## 稀疏自编码器的学习

$$E_{sparse}(\mathbf{W}, \mathbf{b}) = L(\mathbf{W}, \mathbf{b}) + \lambda \Omega(\mathbf{W}) + \beta Q(\hat{\rho}(\mathbf{W}, \mathbf{b}))$$
$$= \frac{1}{2N} \sum_{i=1}^N \|Y(\mathbf{x}_i; \mathbf{W}, \mathbf{b}) - \mathbf{x}_i\|^2 + \lambda \|\mathbf{W}\|_F^2 + \beta \sum_j KL(\rho \parallel \hat{\rho}_j)$$

$$[\hat{\mathbf{W}}, \hat{\mathbf{b}}] = \arg \min_{\mathbf{W}, \mathbf{b}} E_{sparse}(\mathbf{W}, \mathbf{b})$$

惩罚平均激活值关于  
目标值的偏离程度

$$KL(\rho \parallel \hat{\rho}_j) = \rho \cdot \log \frac{\rho}{\hat{\rho}_j} + (1 - \rho) \cdot \log \frac{1 - \rho}{1 - \hat{\rho}_j}$$

$\hat{\rho}_j$  —— 自编码器编码层第  $j$  个神经元节点关于训练样本集  
各样本的平均激活值

$\rho$  —— 稀疏参数, 为接近于 0 的小正数, 如  $\rho = 0.05$

$\beta$  —— 稀疏惩罚系数