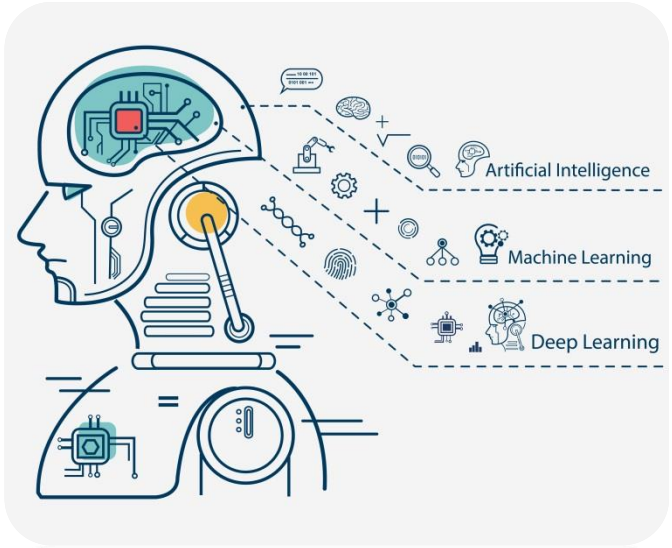
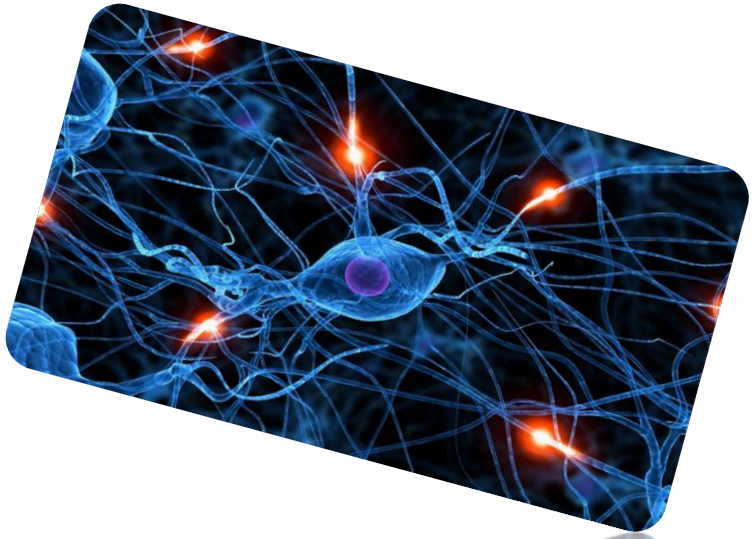
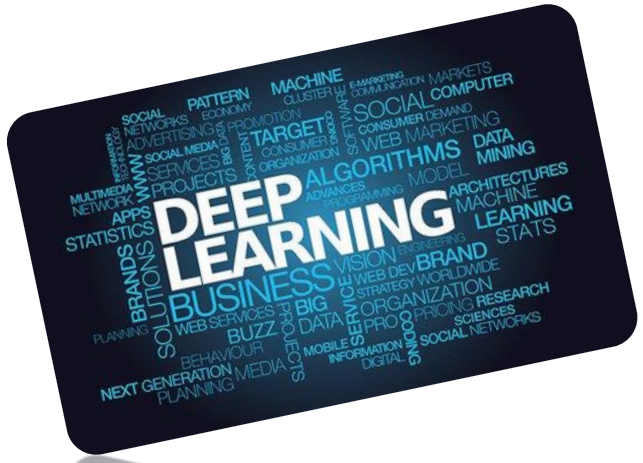
NUS National University of Singapore

School of Computing





神经网络与深度学习简介



赵沄堃

人机对弈：AlphaGo击败人类棋手，然后...

I

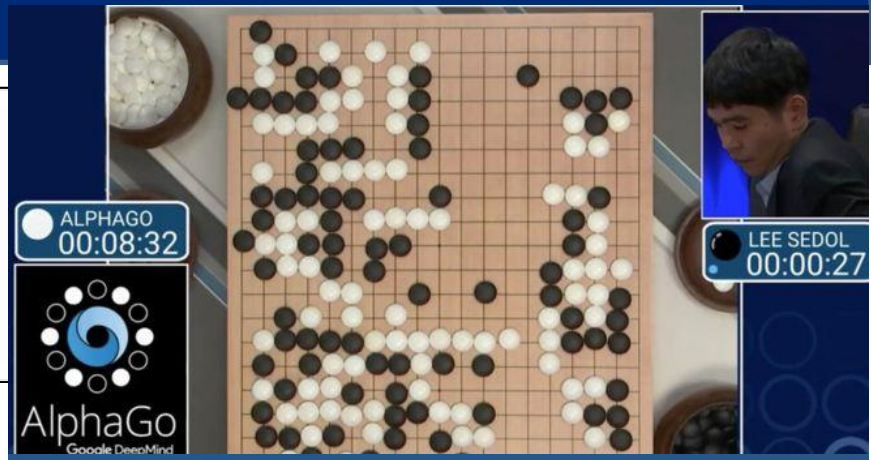
- 2016年03月
- AlphaGo(1.0) **4 : 1** 李世石

II

- 2017年05月
- AlphaGo(2.0) **3 : 0** 柯洁

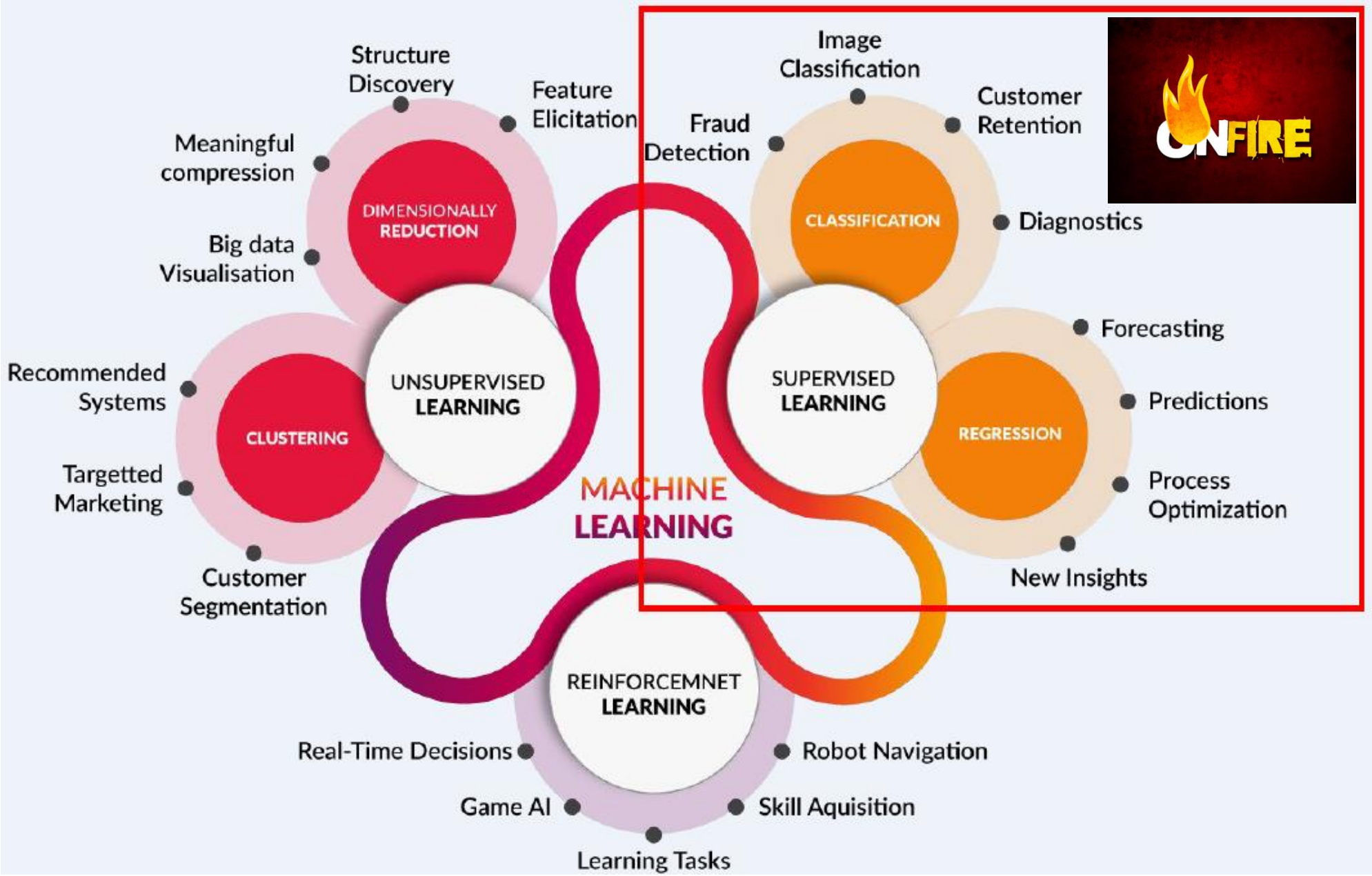
III

- 2017年10月
- AlphaGo(2.0) **0 : 100** AlphaGo Zero



百度面部识别AI在《最强大脑》中击败人类

机器学习的应用


MACHINE LEARNING
UNSUPERVISED LEARNING
DIMENSIONALLY REDUCTION
Structure Discovery
Feature Elicitation
Meaningful compression
Big data Visualisation
CLUSTERING
Recommended Systems
Targetted Marketing
Customer Segmentation
SUPERVISED LEARNING
CLASSIFICATION
Image Classification
Customer Retention
Fraud Detection
Diagnostics
REGRESSION
Forecasting
Predictions
Process Optimization
New Insights
REINFORCEMNET LEARNING
Real-Time Decisions
Robot Navigation
Game AI
Skill Aquisition
Learning Tasks
ONFIRE

机器是如何学习的？

➢ 例子: 用线性回归模型预测房价

模型：$Y = f(X) + \varepsilon = X\beta + \varepsilon$

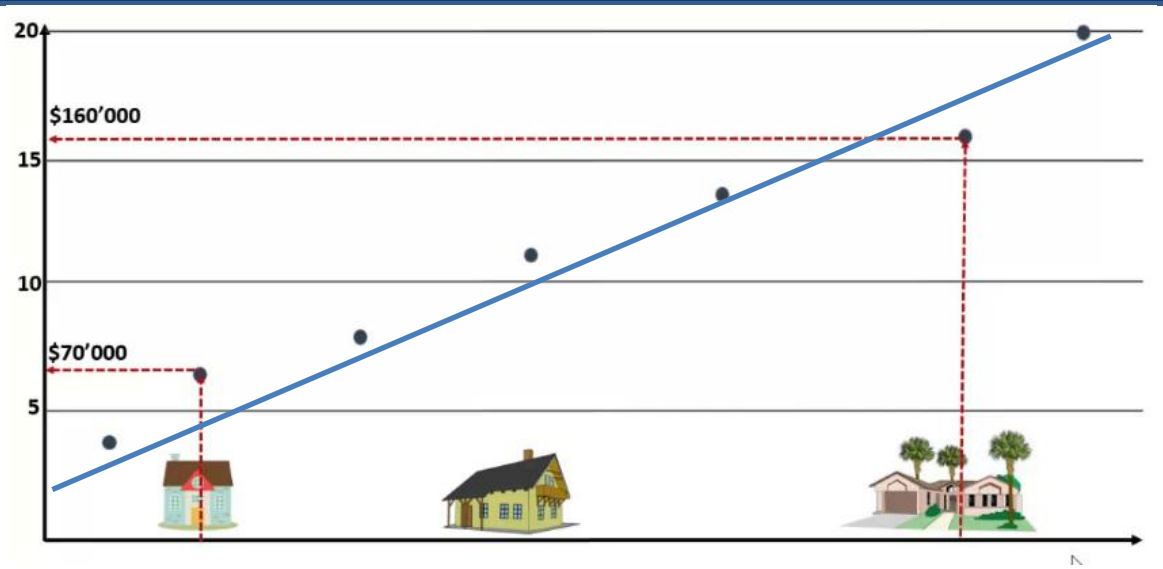
面积　#房间　#卫生间　房价

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix}, \beta = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{bmatrix}, Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{bmatrix}$$

$$\min_{\beta} Loss(\beta) = \varepsilon^T \varepsilon = (Y - X\beta)^T (Y - X\beta)$$

$\rightarrow$ 一阶求导：$-2X^T Y + 2X^T X\beta = 0$

$\rightarrow \beta_{OLS} = (X^T X)^{-1} X^T Y$



ID	面积 (m²)	#房间	#卫生间	房价 (万元)
1	65	2	2	900
2	72	3	2	120
……	……	……	……	……
n	50	1	1	500

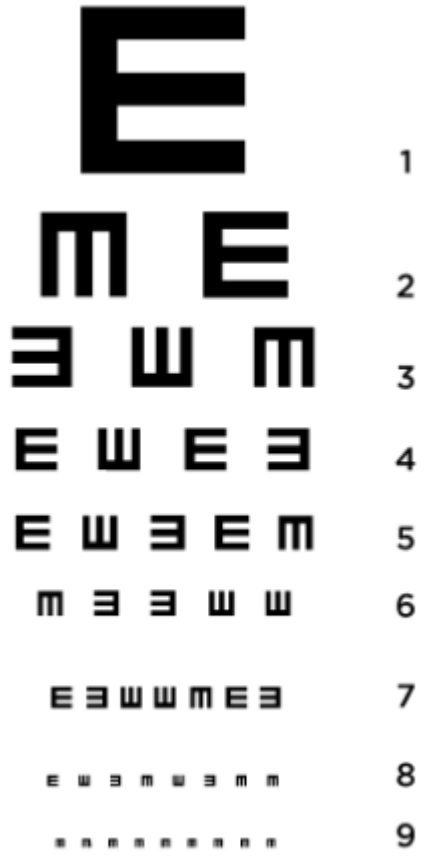
机器是如何学习的？

Machine

Learning (Not Memorizing)

Use algorithms to discover patterns in the data

Resource



机器学习：规范模式

输入： $\boldsymbol{X} \in \mathbb{R}^d$ (房屋信息，如：面积，地段，等)

输出： 回归: $y \in (-\infty, \infty)$ (如：房价)

分类: $y \in (0,1)$

目标函数： $f : X \to y$ (真实的预测模型)

f 实际未知

数据： $D = \{(x_1, y_1), \ldots (x_N, y_N)\}$ (训练数据集)

拟合模型 $\hat{f} \in F$ ： $\hat{f} : X \to y$ (最小化训练误差)

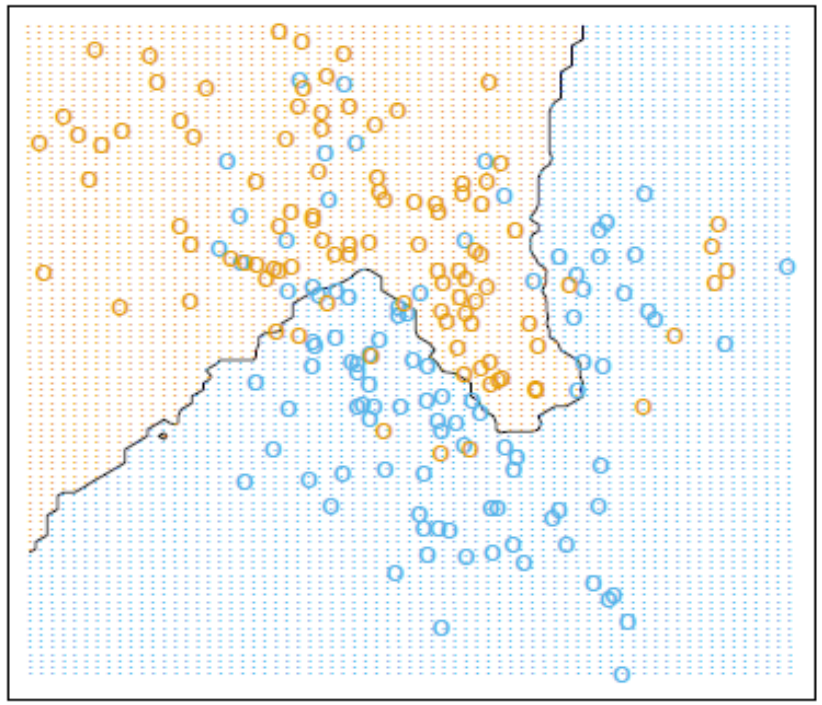
其中训练误差： $\frac{1}{N} \sum_{n=1}^{N} Loss(\hat{f}(x_n) \neq f(x_n))$

机器学习目标： $\min_{\hat{f}} E[Loss(\hat{f}(x_{new}) \neq f(x_{new}))]$ (最小化预测/测试误差)

传统机器学习的问题

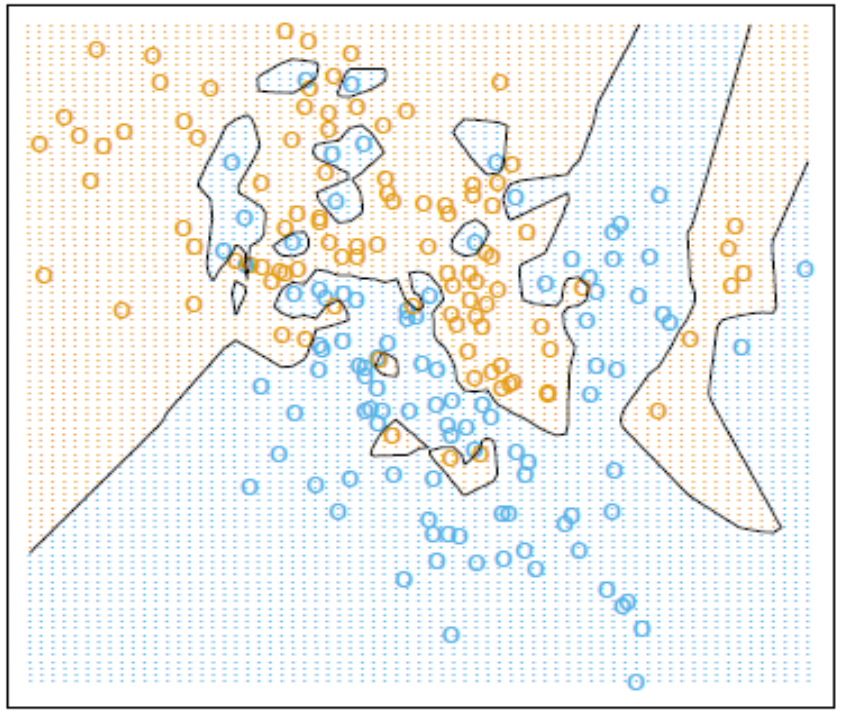
- 例子：用传统机器学习模型，对非线性数据集进行分类

欠拟合：训练偏差



From ESL

过拟合：预测方差



From ESL

传统机器学习的问题

➢ 测试数据集上的泛化误差：

$$\Pr\{|\frac{1}{N}\sum_{n=1}^{N} Loss(\hat{f}(x_n) \neq f(x_n)) - E[Loss(\hat{f}(x_{new}) \neq f(x_{new}))]| > \varepsilon\}$$

$$= \Pr\{|\text{训练误差} - \text{预测误差}| > \varepsilon\}$$

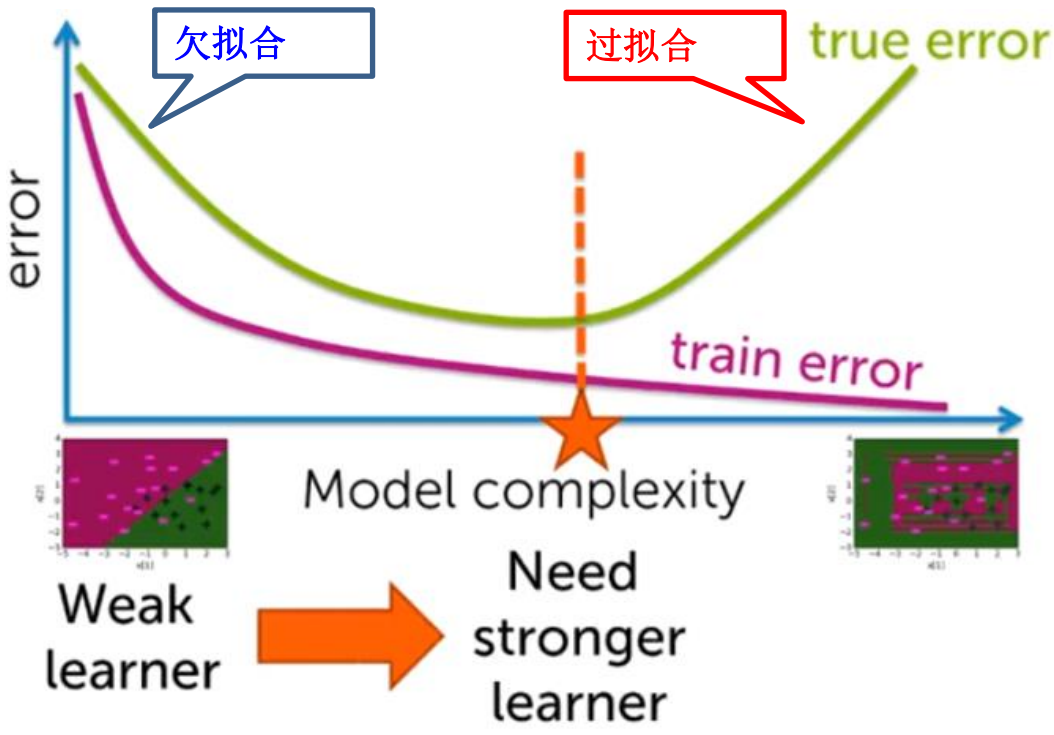
$$\leq \frac{2M}{e^{2N\varepsilon^2}}$$

- M：反映模型的复杂程度
- N：训练数据集的大小
- ε：对预测误差偏离训练误差多少的接受程度

♥ 理想模型 $\hat{f}$ ：训练误差 ≈ 0

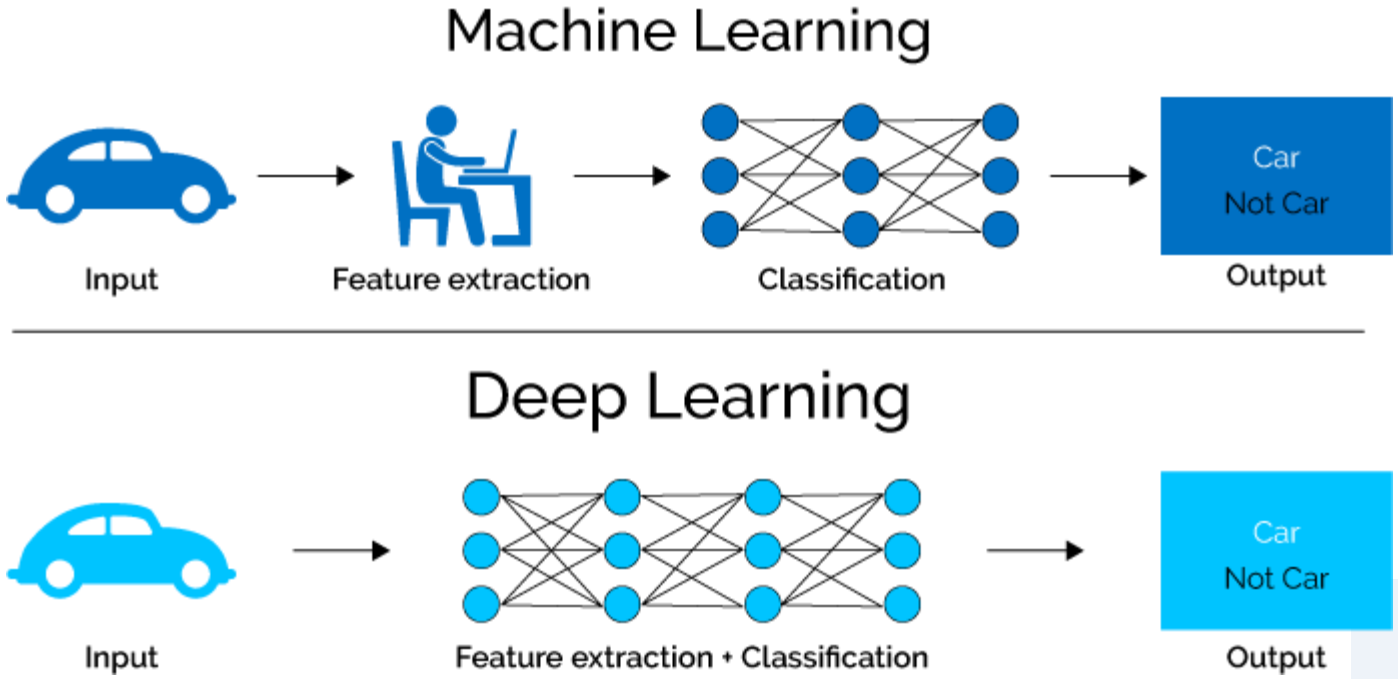
并且：训练误差 $\approx$ 预测误差

偏差-方差权衡



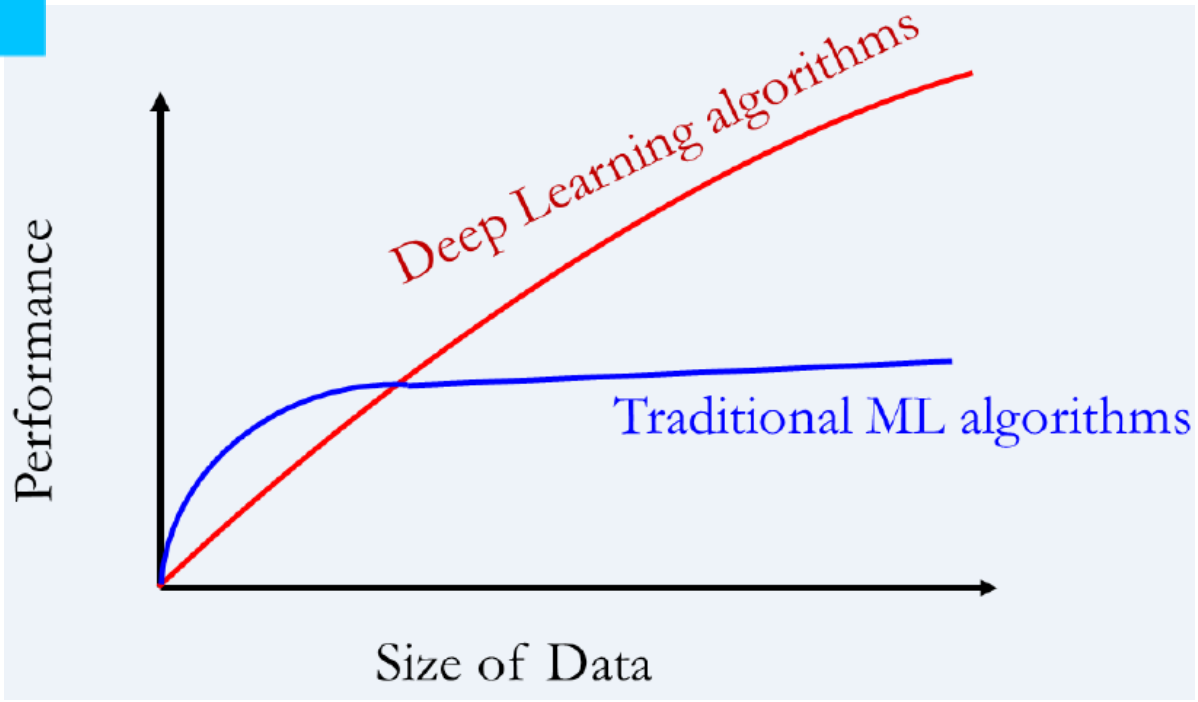
$$\mathrm{E}\left[(y-\hat{f}(x))^2\right] = \left(\mathrm{Bias}\left[\hat{f}(x)\right]\right)^2 + \mathrm{Var}\left[\hat{f}(x)\right] + \sigma^2$$

从传统机器学习到深度学习



- 人类大脑的学习能力这么强，为什么不让机器学习模仿人类大脑？

- “深度学习，并非学的东西不同，而是学习方式不同”
- 深度学习是自适应的

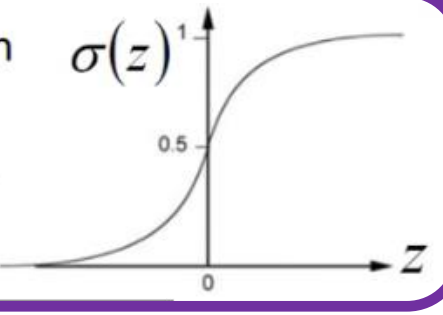


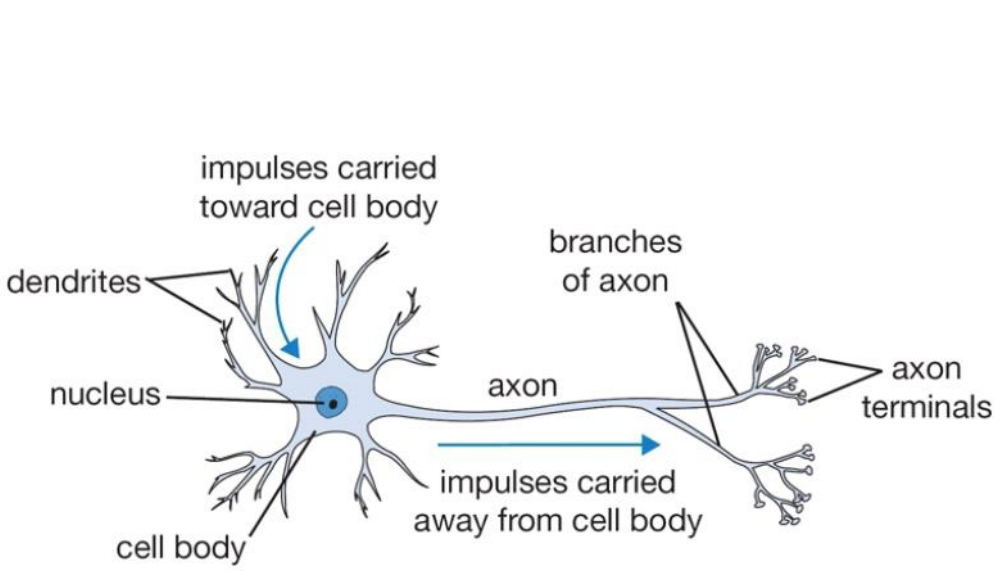
深度学习：基本认识

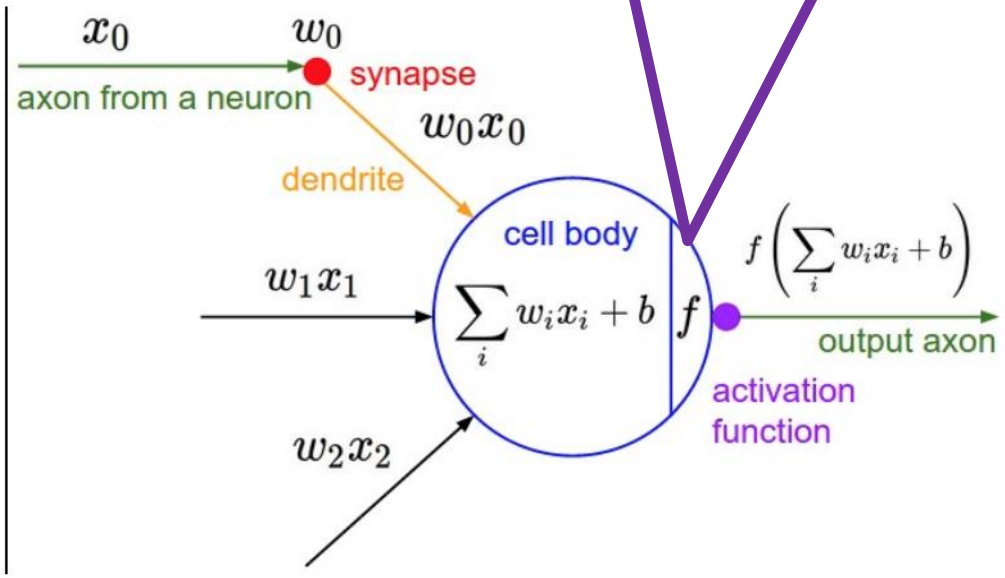
- 机器学习如何模仿人类大脑？
- 生物学理论
  - 神经元细胞(或感知器)：信息接收、处理与传递
  - 多层结构：上一层神经元的输出信息被下一层神经元接收并处理
  - **激活函数(非线性)：对输入信息进行(非线性)处理**

Sigmoid Function

$$\sigma(z)=\frac{1}{1+e^{-z}}$$



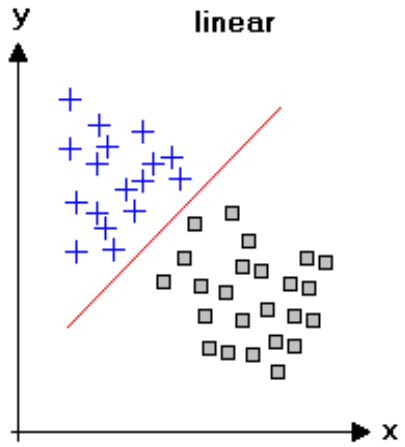


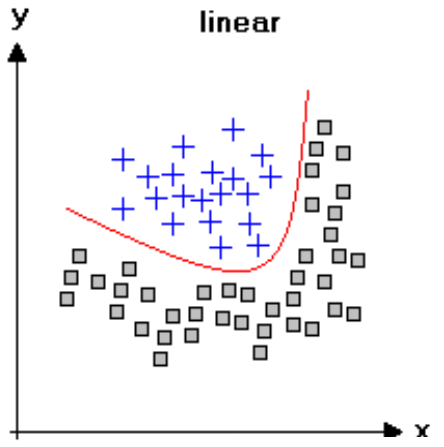


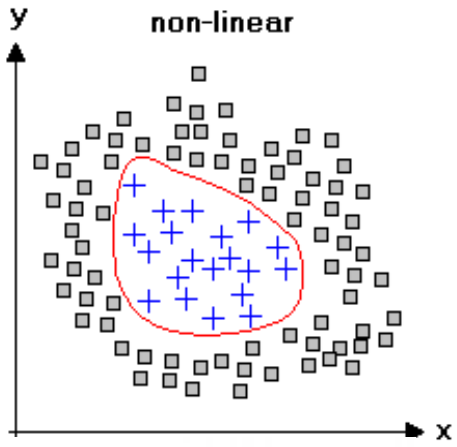
A cartoon drawing of a biological neuron (left) and its mathematical model (right).

神经网络：激活函数

- 神经网络模型：如何更好地学习到数据的非线性特性？
- 非线性激活函数

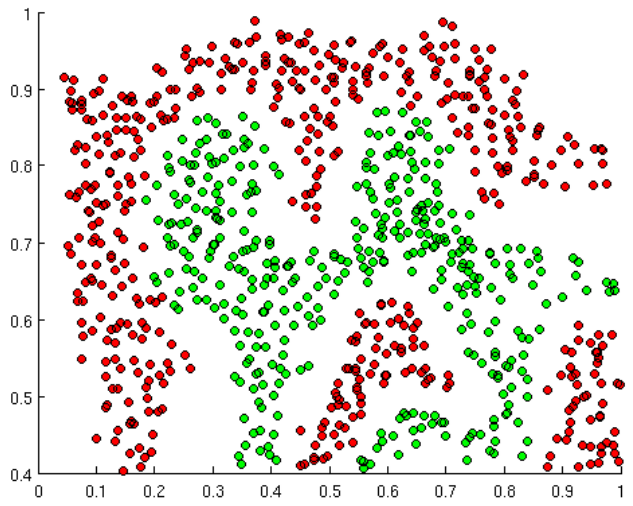


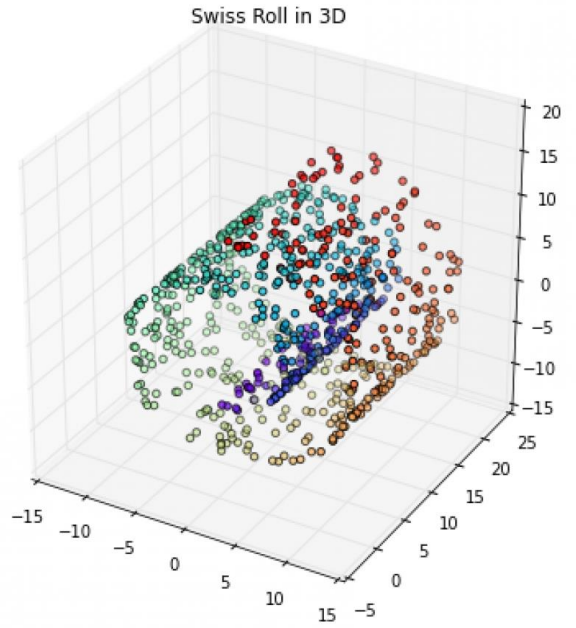




非线性激活函数：

赋予神经网络模型对**复杂非线性数据特性**的学习能力

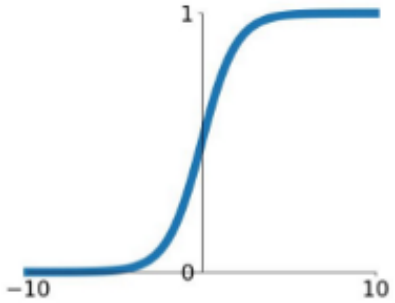




Activation Functions

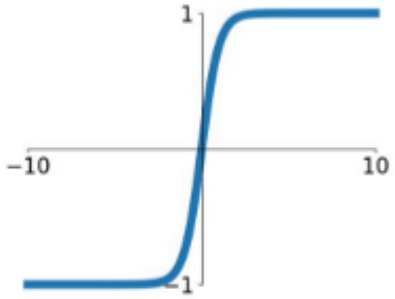
Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



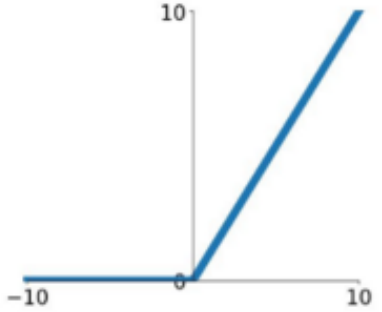
tanh

$$\tanh(x)$$



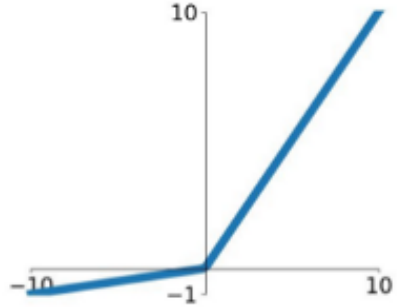
ReLU

$$\max(0, x)$$



Leaky ReLU

$$\max(0.1x, x)$$

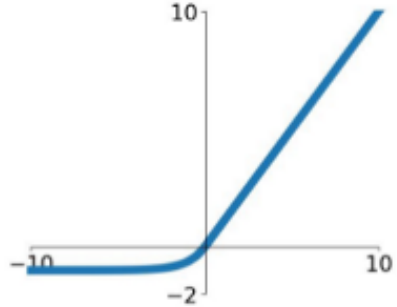


Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

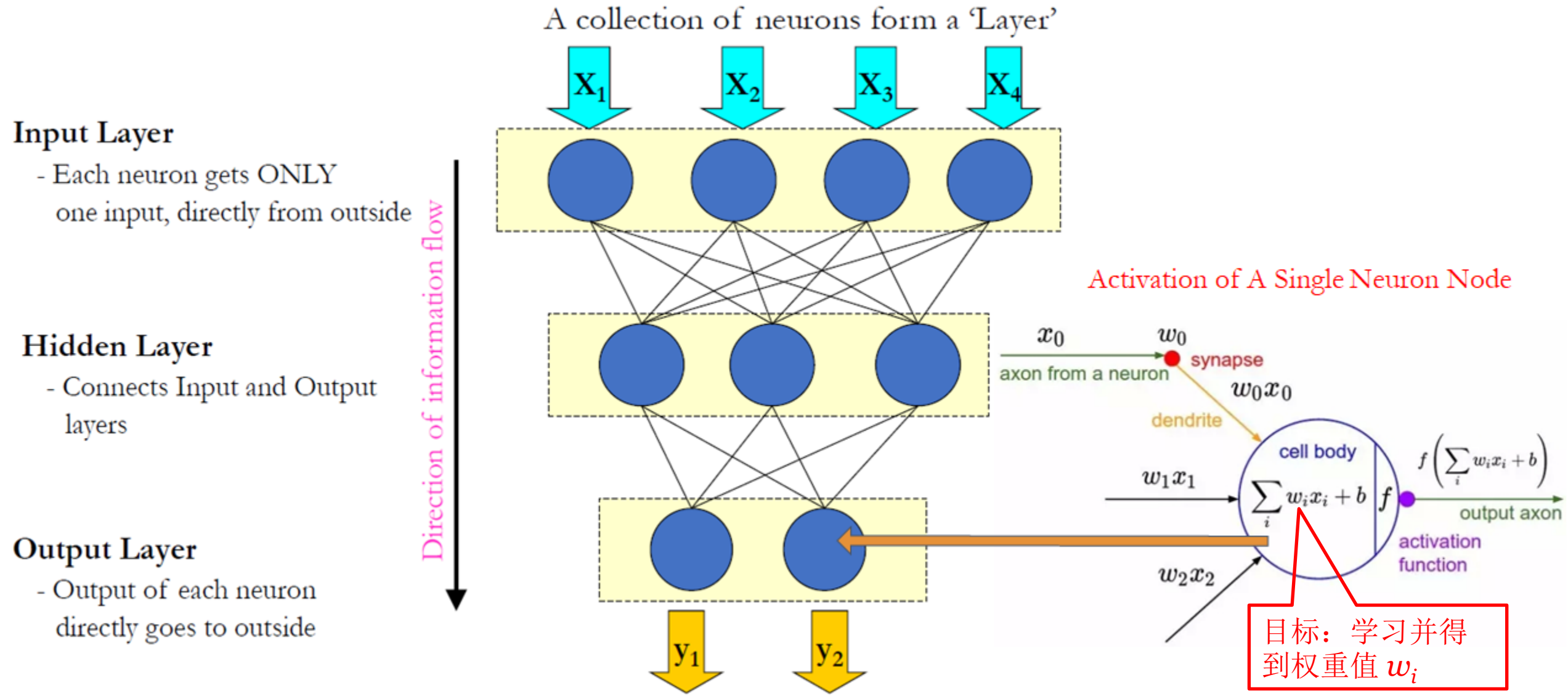
ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



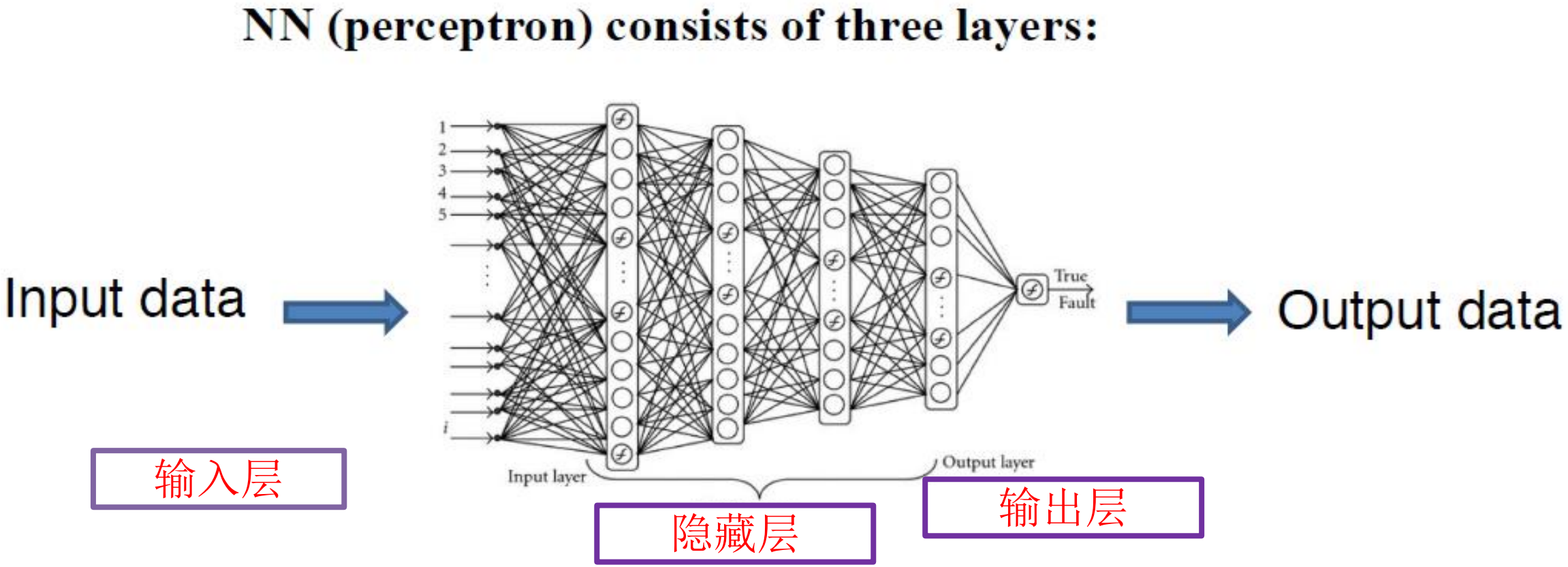
神经网络：基本结构

- 基本结构：使多层神经元细胞(或感知器)连接在一起

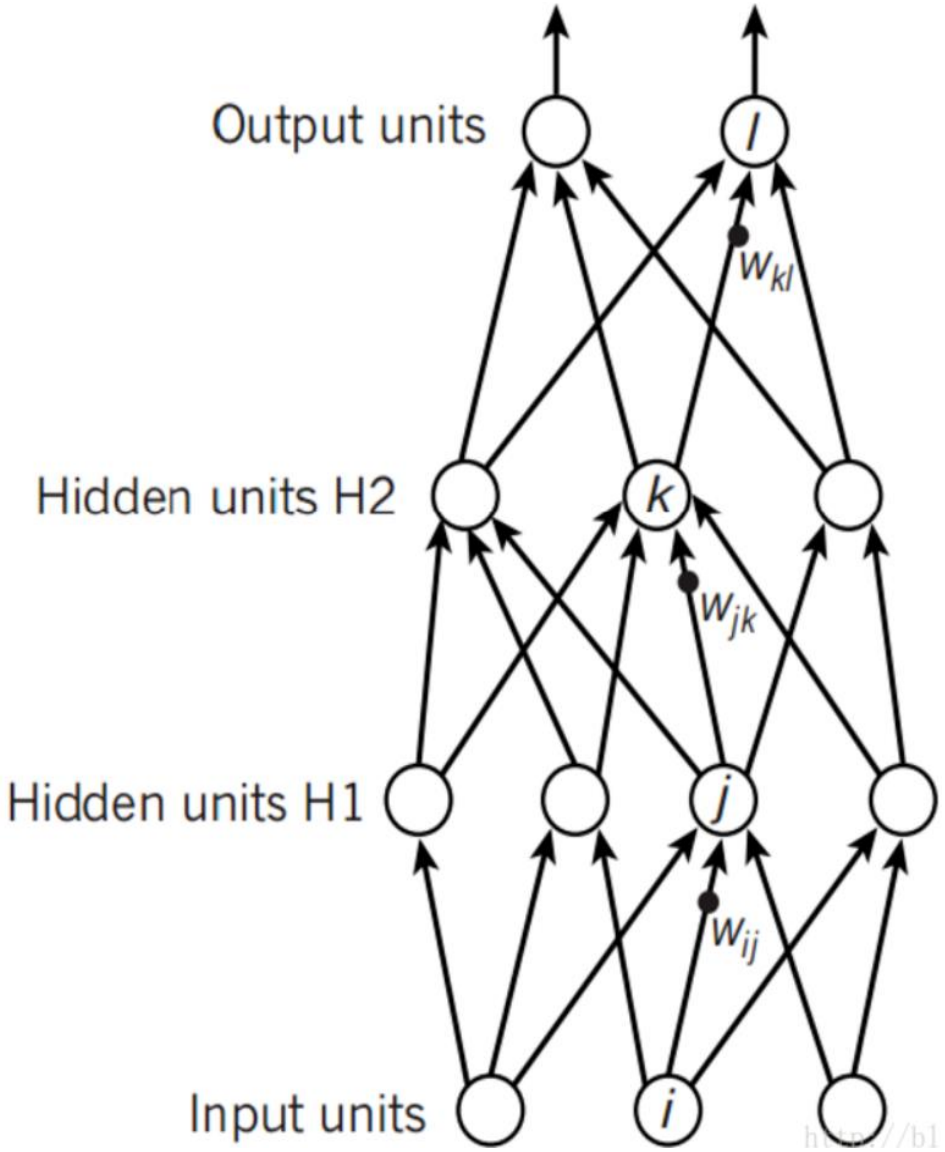


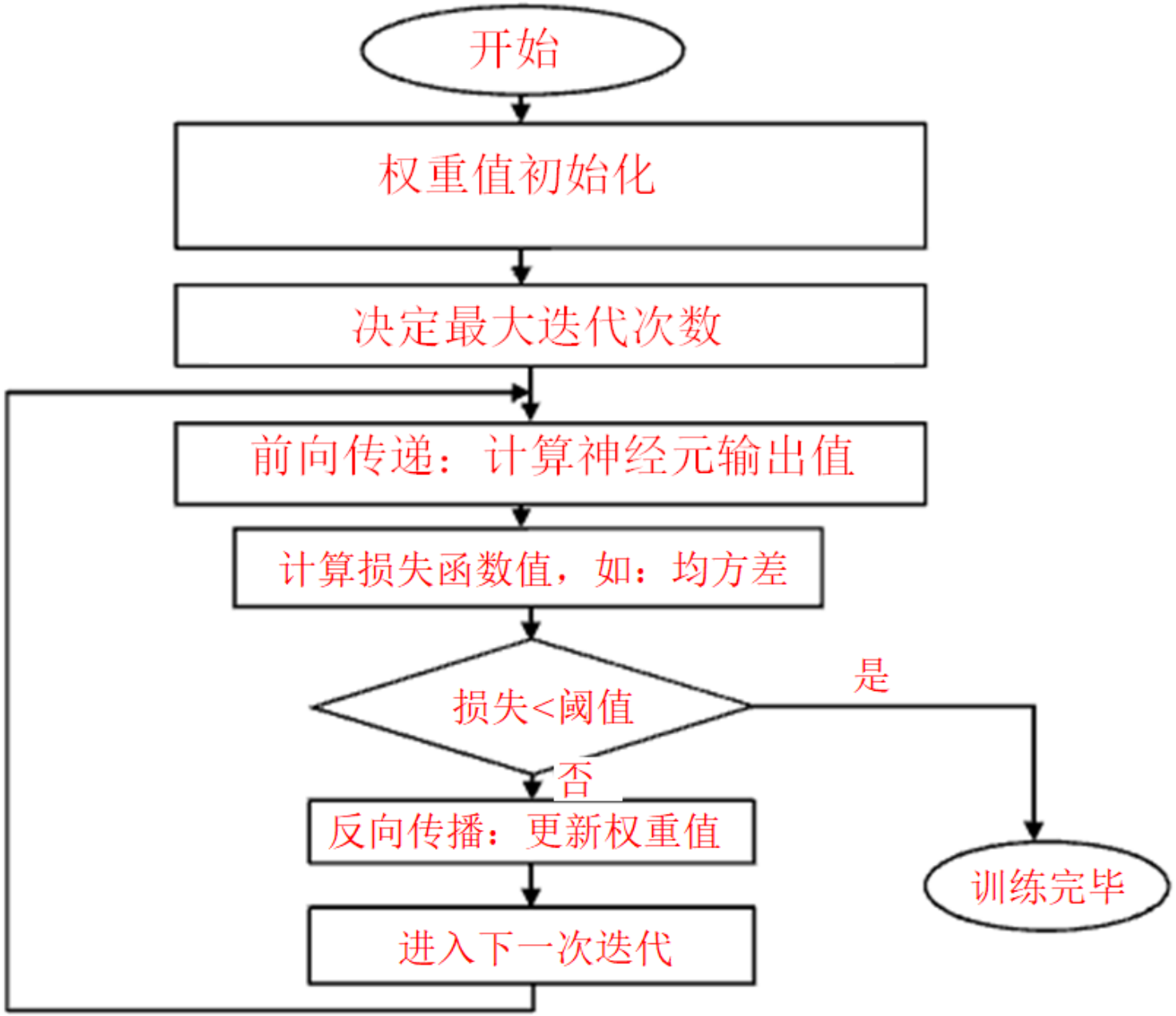
神经网络：深层神经网络

- 深层神经网络
  - 基本结构：使多层神经元细胞(或感知器) 连接在一起
  - 很多很多隐藏层

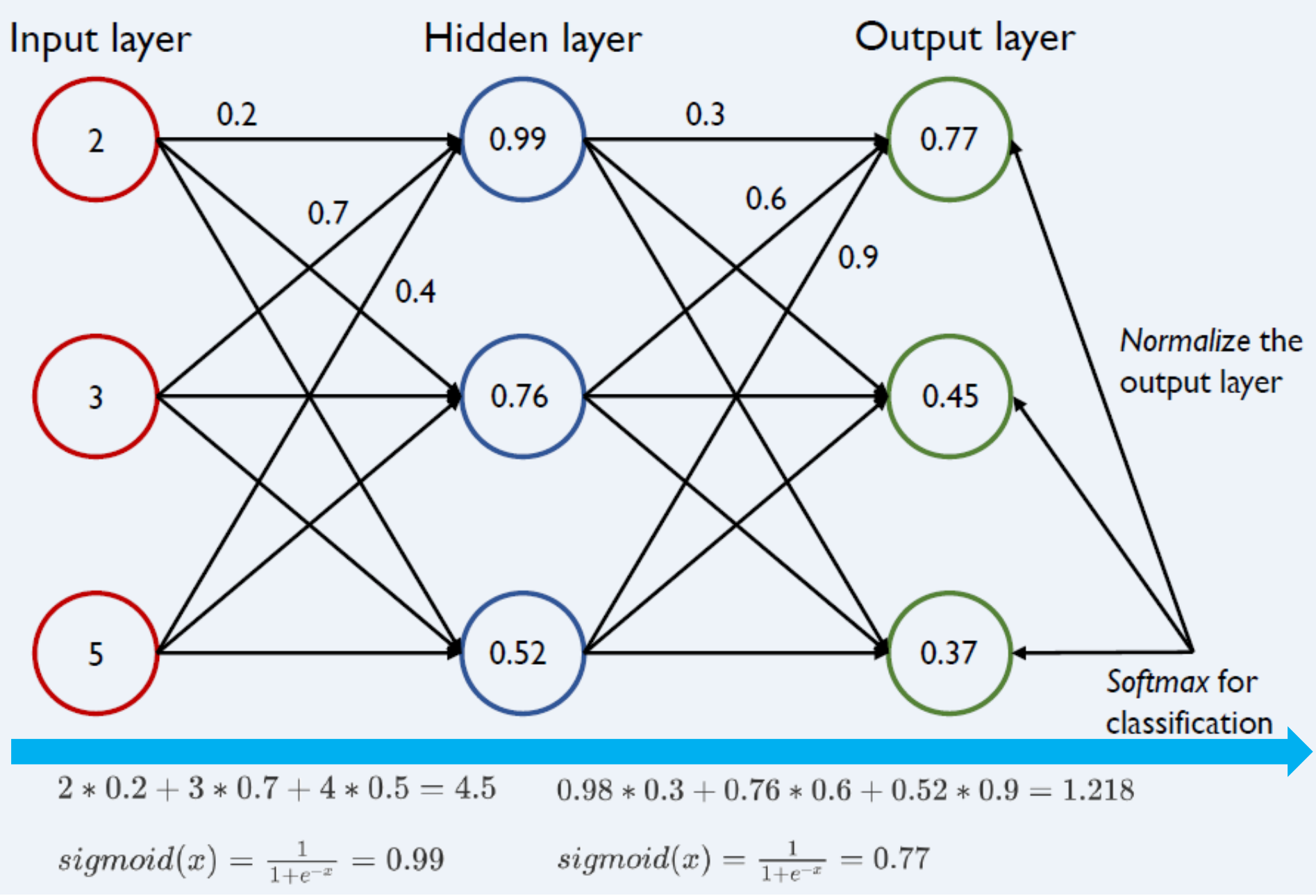


神经网络：模型训练的基本流程

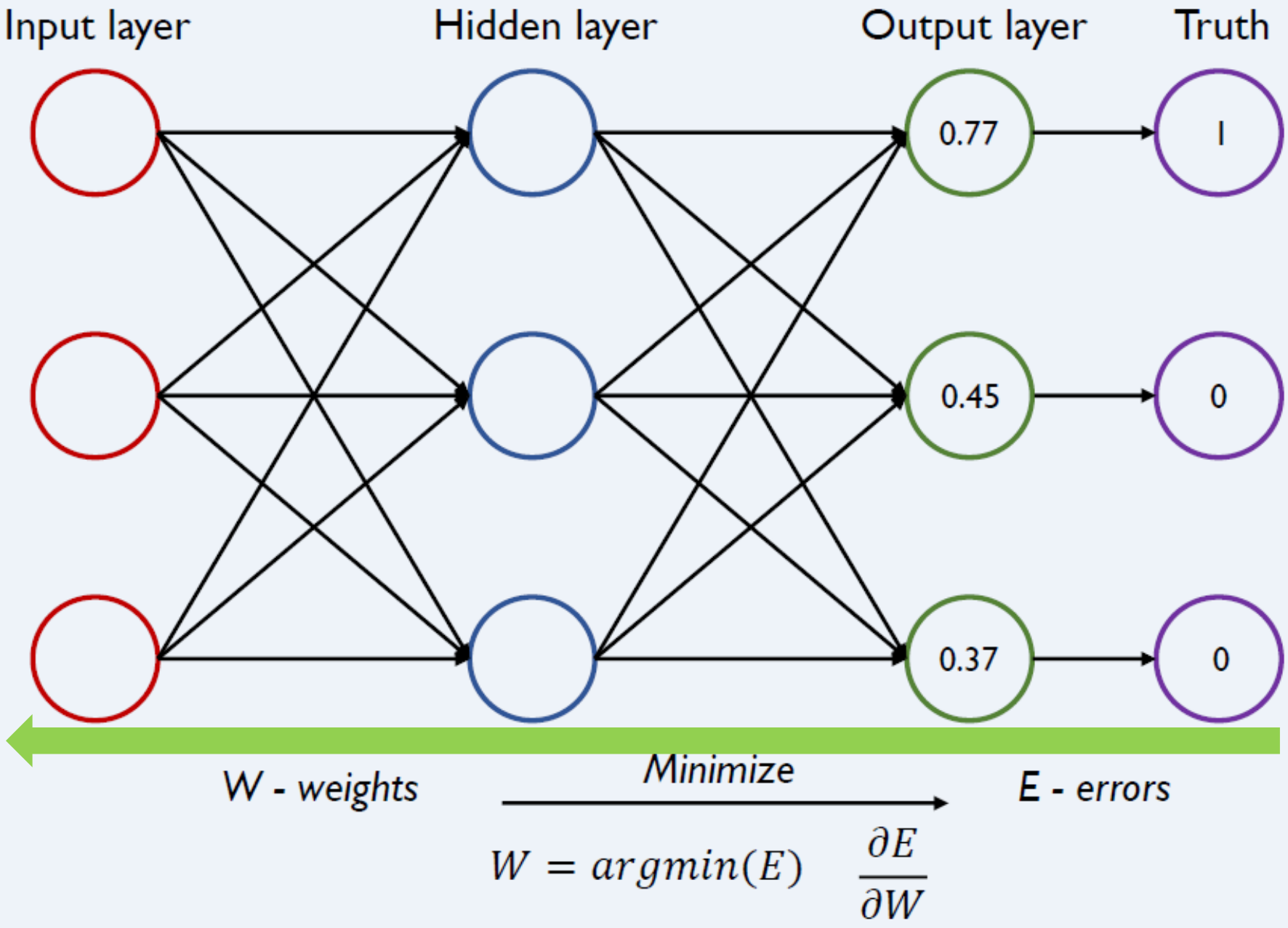




神经网络：前向传递


Input layer
Hidden layer
Output layer
2
3
5
0.2
0.7
0.4
0.99
0.76
0.52
0.3
0.6
0.9
0.77
0.45
0.37
Normalize the output layer
Softmax for classification
$2 * 0.2 + 3 * 0.7 + 4 * 0.5 = 4.5$
$0.98 * 0.3 + 0.76 * 0.6 + 0.52 * 0.9 = 1.218$
$sigmoid(x) = \frac{1}{1+e^{-x}} = 0.99$
$sigmoid(x) = \frac{1}{1+e^{-x}} = 0.77$

神经网络：反向传播



问题：为什么要反向传播？目的是什么？

- 产生误差的原因在于权值有偏差，所以需要返回更新权值

- 返回更新权值的方法：梯度下降算法(Gradient Descent)

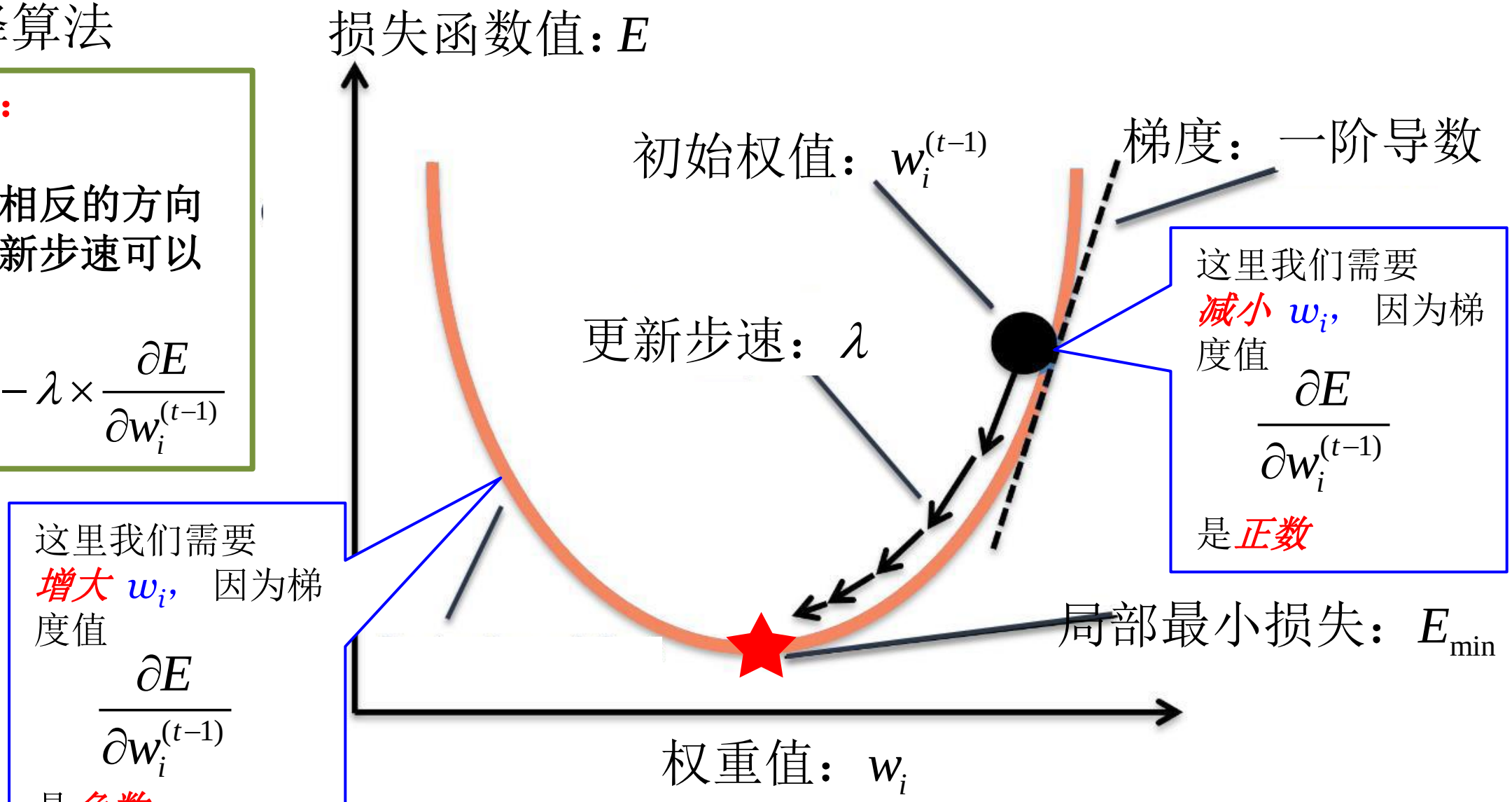
神经网络：反向传播

➢ 梯度下降算法

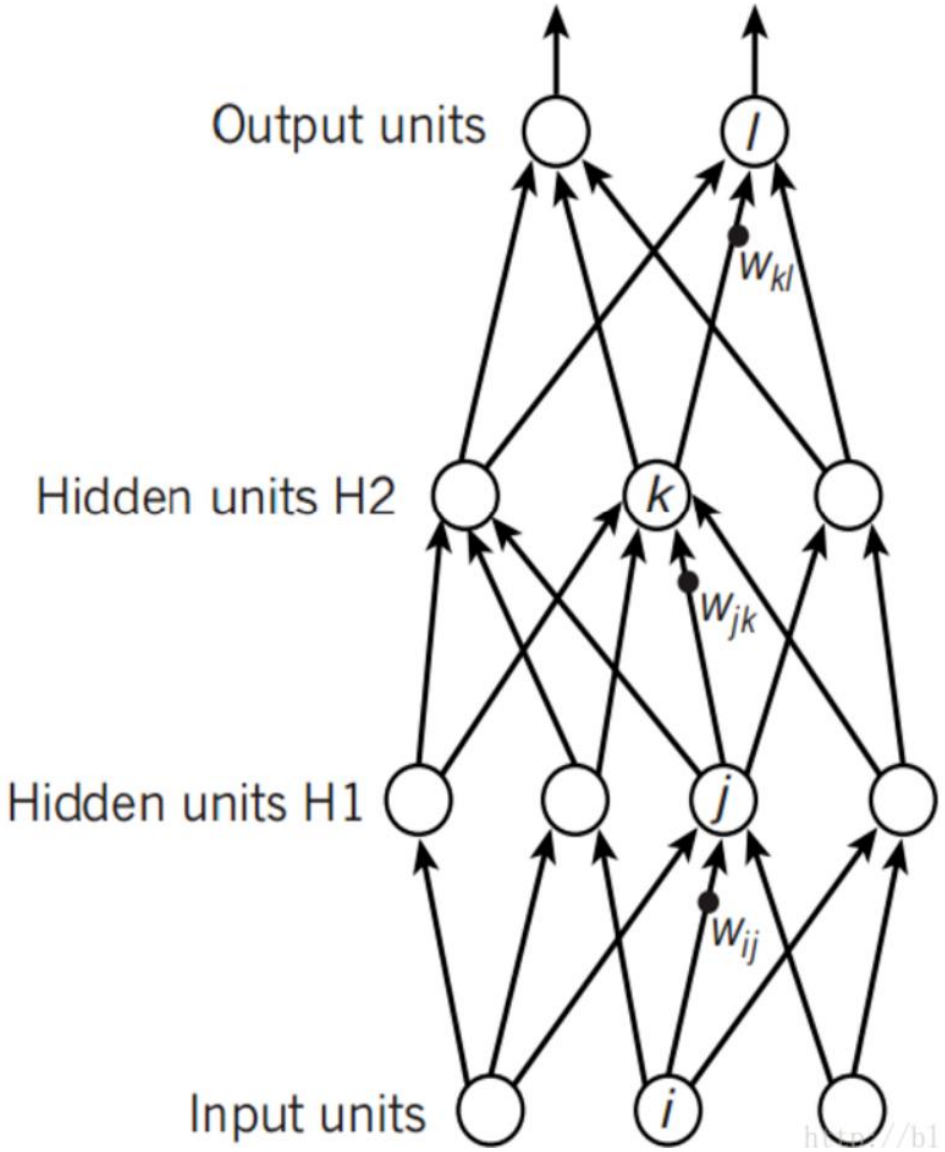
权值更新规则：

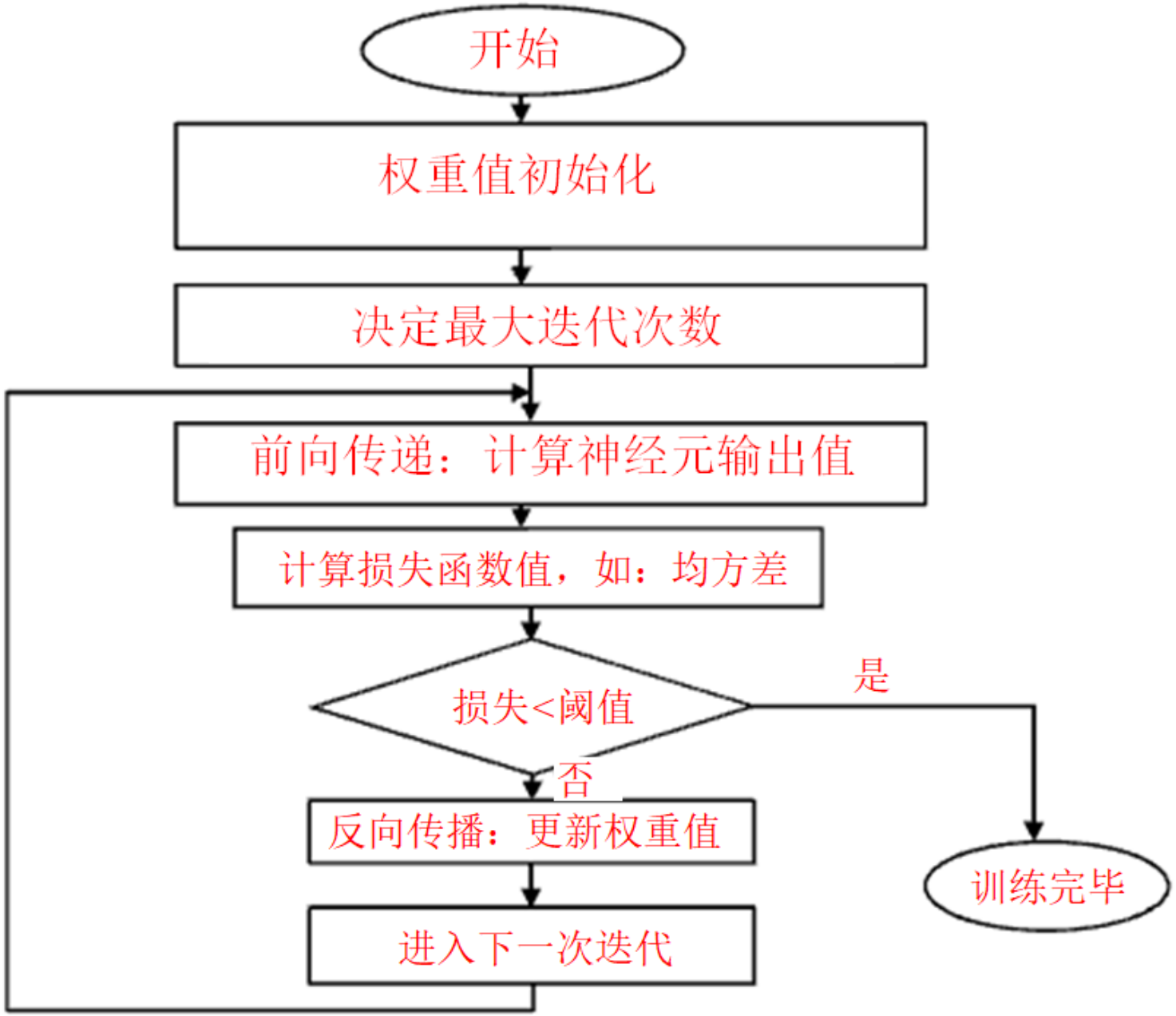
每次往梯度值相反的方向移动更新，更新步速可以设置为 λ：

$$w_i^{(t)} \leftarrow w_i^{(t-1)} - \lambda \times \frac{\partial E}{\partial w_i^{(t-1)}}$$



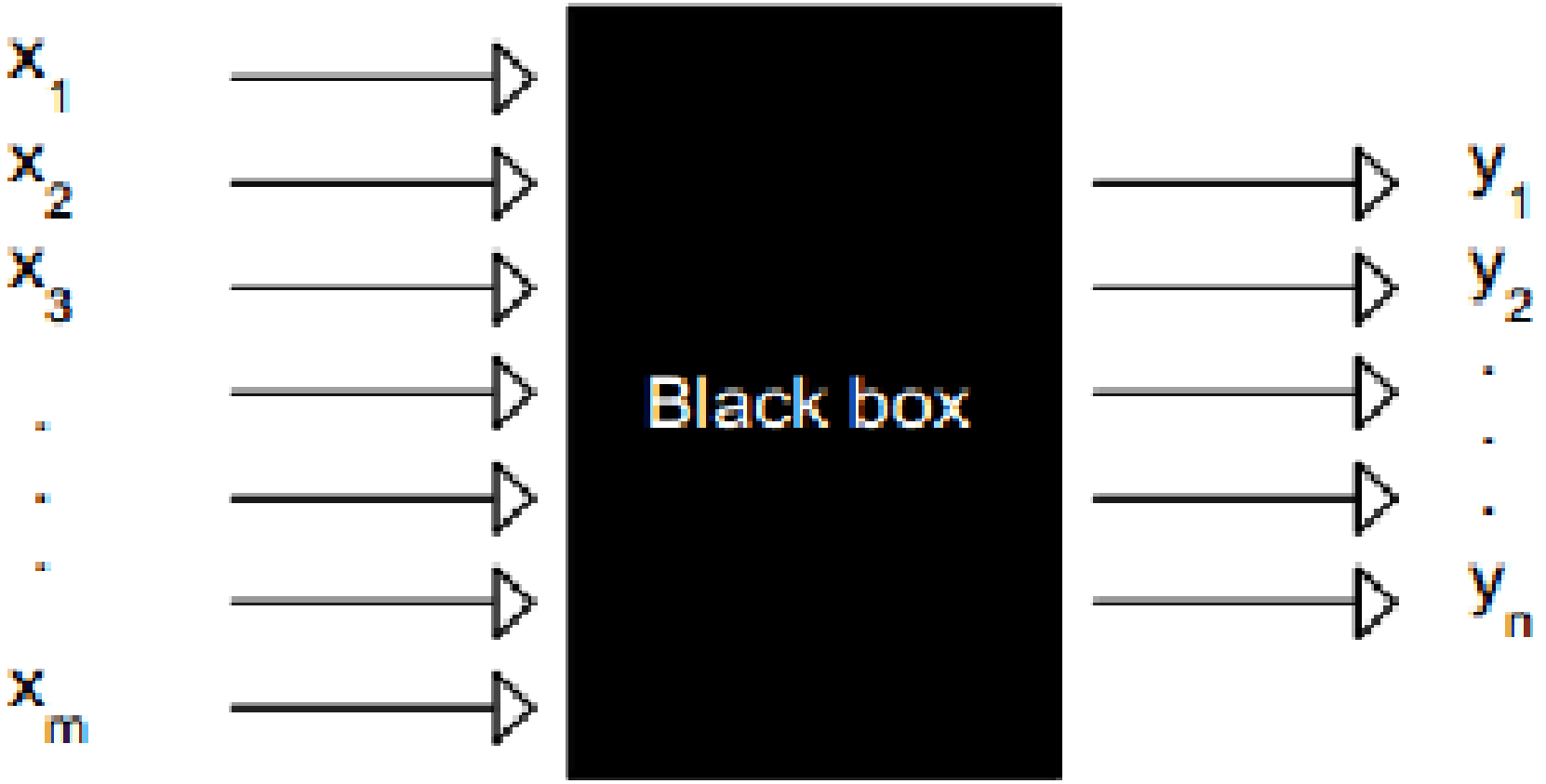
神经网络：快速总结


Output units
Hidden units H2
Hidden units H1
Input units
l
k
j
i
w_{kl}
w_{jk}
w_{ij}


开始
权重值初始化
决定最大迭代次数
前向传递：计算神经元输出值
计算损失函数值，如：均方差
损失<阈值
是
否
反向传播：更新权重值
进入下一次迭代
训练完毕

神经网络模型：缺陷与不足

神经网络：缺陷 I

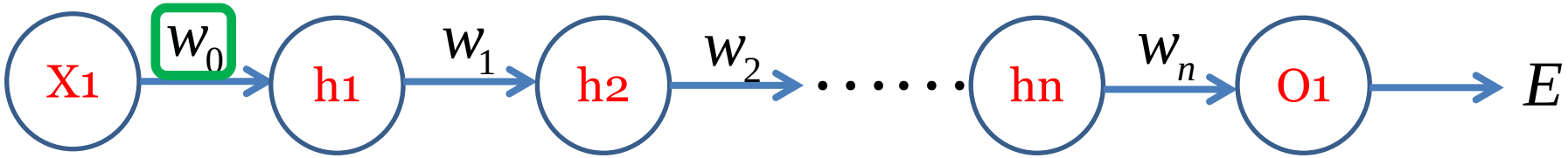


- 神经网络模型强在预测与泛化能力，但 解释能力不足
- 神经网络模型的隐藏层就像“黑箱”

神经网络：缺陷 II

- 梯度消失与梯度爆炸问题

权值更新：$w_i^{(t)} \leftarrow w_i^{(t-1)} - \lambda \times \frac{\partial E}{\partial w_i^{(t-1)}}$



- **计算梯度的链式法则：**

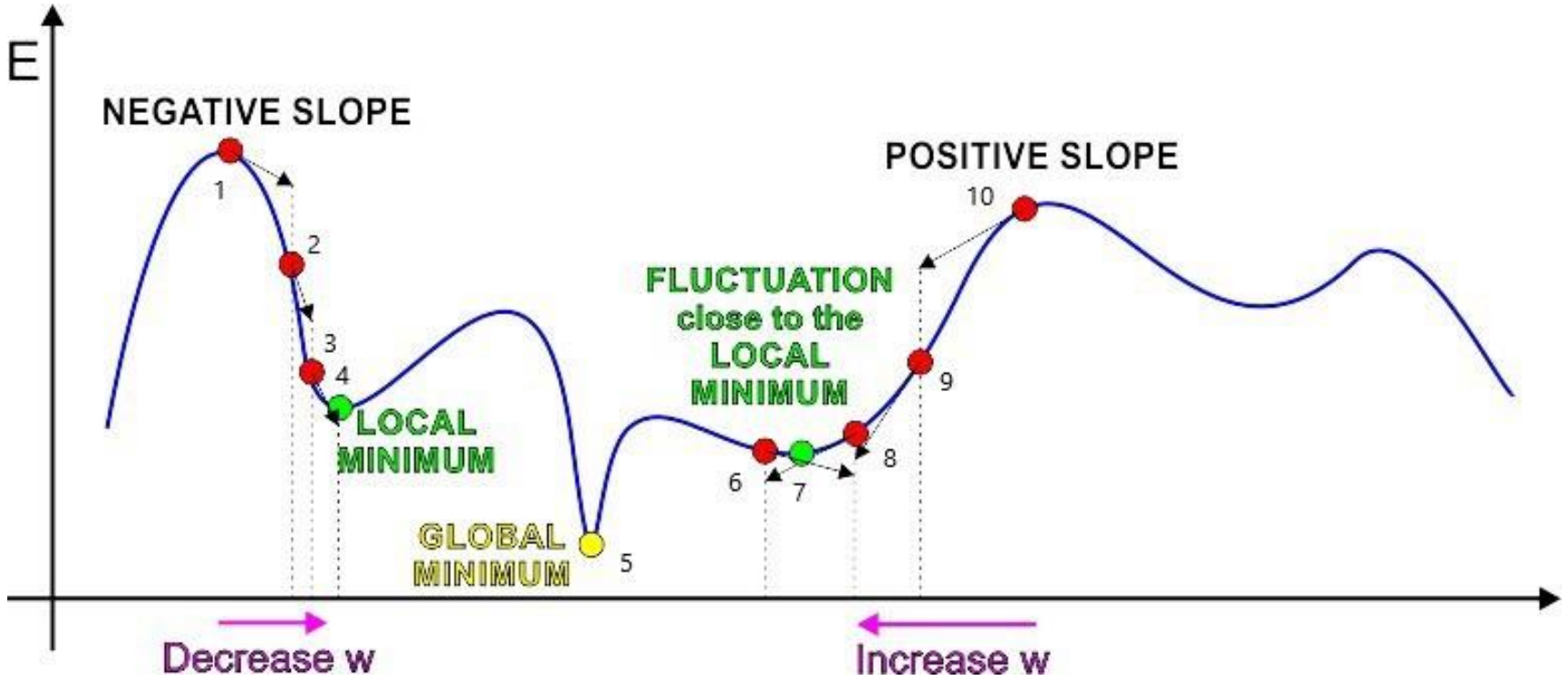
$$\frac{\partial E}{\partial w_0} = \frac{\partial E}{\partial O_{1\_out}} \times \frac{\partial O_{1\_out}}{\partial O_{1\_in}} \times \frac{\partial O_{1\_in}}{\partial h_{n\_out}} \times \frac{\partial h_{n\_out}}{\partial h_{n\_in}} \times \cdots \times \frac{\partial h_{1\_out}}{\partial h_{1\_in}} \times \frac{\partial h_{1\_in}}{\partial w_0}$$

$$= \frac{\partial E}{\partial O_{1\_out}} \times f'(O_{1\_in}) \times w_n \times f'(h_{n\_in}) \times w_{n-1} \times \cdots \times f'(h_{1\_in}) \times x_1$$

- 如果随意选择激活函数，
  - 如果梯度小于1 → 梯度消失：$\frac{\partial Sigmoid(x)}{\partial x} \in (0,1)$ so $\frac{\partial E}{\partial w_0} \to 0$
    - 解决办法：想清楚该不该选这个激活函数
  - 如果梯度大于1 → 梯度爆炸：$\frac{\partial E}{\partial w_0} \to \infty$
    - 解决办法：梯度裁剪(Gradient Clipping)；对权值进行正则化(Regularization)

神经网络：缺陷 III

- 用“梯度下降算法”进行权值优化：



注意：

梯度下降本质上是一种“贪心算法”

- 改进梯度下降：随机梯度下降算法(Stochastic Gradient Descent)
- 其他优化算法：模拟退火算法(Simulated Annealing)，用于玻尔兹曼机(Boltzmann Machine)

神经网络：缺陷 IV

- 测试数据集上的泛化误差：

$$\Pr\{|\frac{1}{N}\sum_{n=1}^{N} Loss(\hat{f}(x_n) \neq f(x_n)) - E[Loss(\hat{f}(x_{new}) \neq f(x_{new}))]| > \varepsilon\}$$

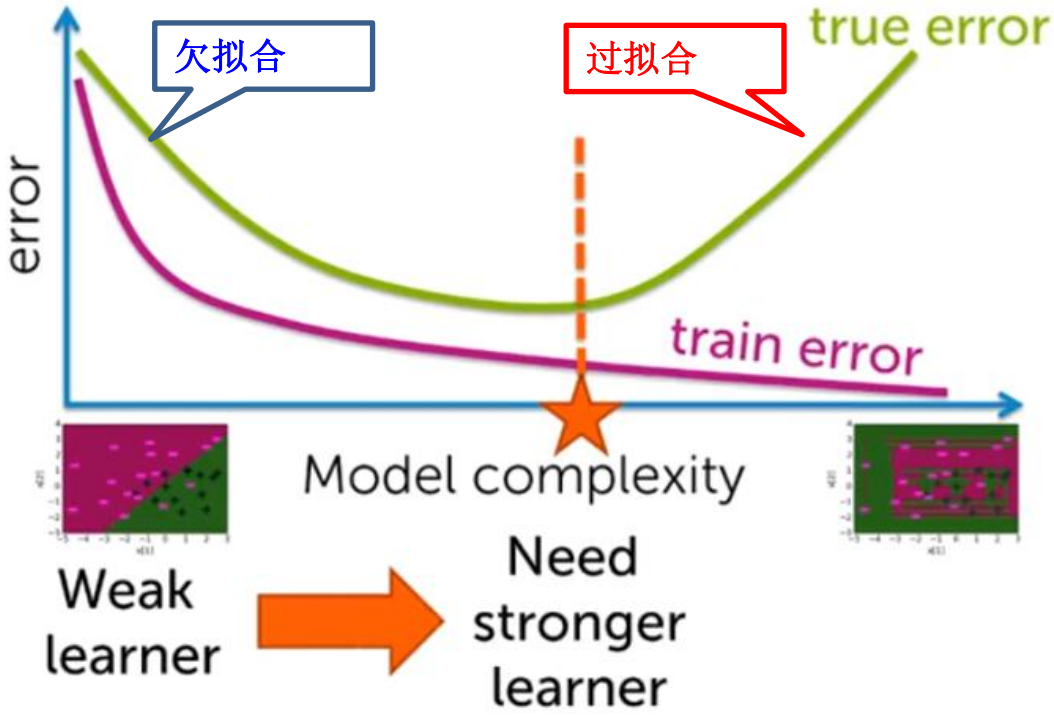
$$= \Pr\{|\text{训练误差} - \text{预测误差}| > \varepsilon\}$$

$$\leq \frac{2M}{e^{2N\varepsilon^2}}$$

♥ 理想模型 $\hat{f}$ ：训练误差 ≈ 0

并且：训练误差 $\approx$ 预测误差

偏差-方差权衡

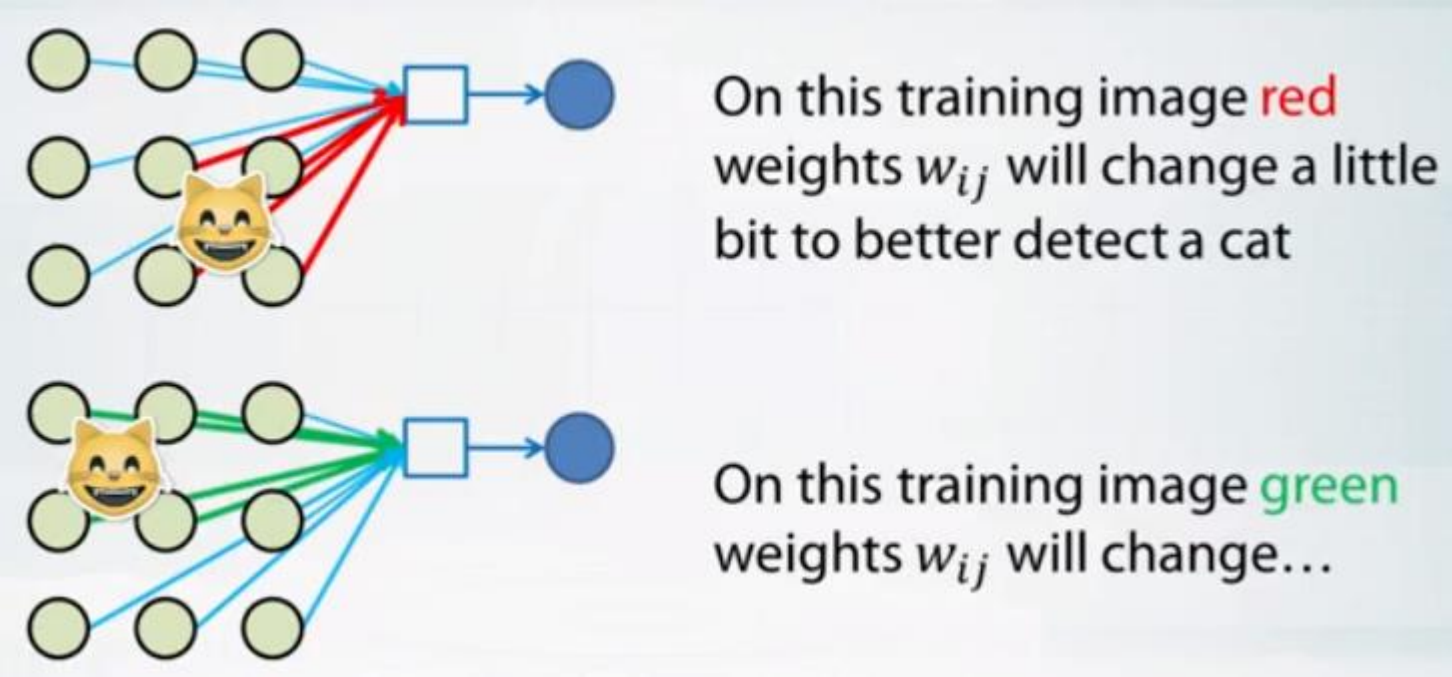


- 神经网络模型仍可能过拟合！
- 通常的解决办法：
  - 对权值进行正则化(Regularization)
  - 批标准化(Batch normalization)
  - 神经元随机失活(Dropout neurons randomly in each layer)

“天下没有免费午餐”

神经网络：缺陷 IV

- 图像分类：
  - 如果用普通的深层神经网络对图片进行分析，会发生什么？
  - 首先把图片分割成独立的像素(pixel)作为输入值，每个像素看作一个变量，取值在0-255之间，衡量像素的颜色深浅
  - 这有什么问题呢？



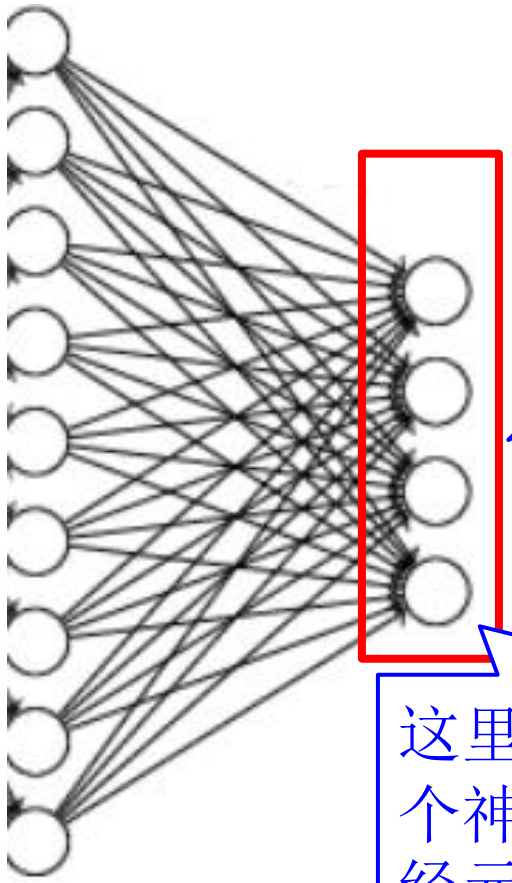
- 普通神经网络并不能完全利用数据集的所有重要信息
- 如果在测试数据集中，猫的位置改变了，怎么办？

神经网络：缺陷 IV

- 图像分类：

普通神经网络

300*300
维度



300*300*4+1

大约有360,001
权值

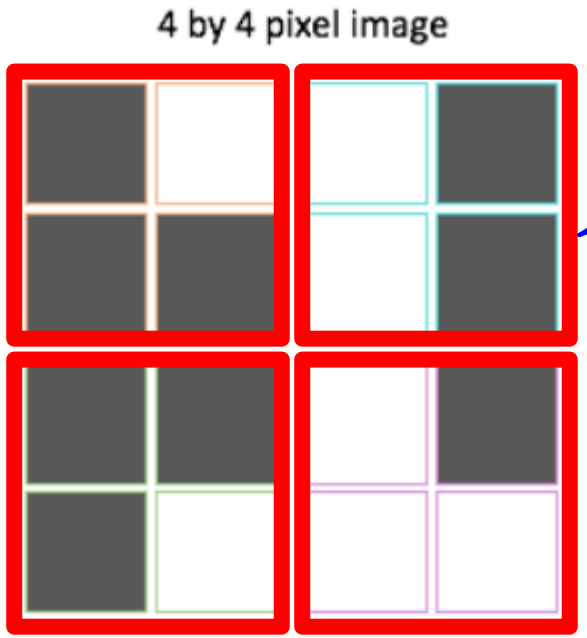
这里假设隐藏层只有 4 个神经元，如果增加神经元数量呢？

如果将普通神经网络应用于图像分析：

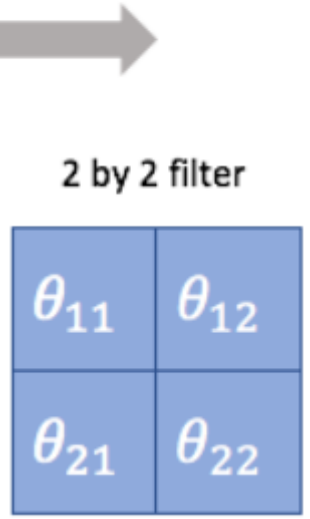
- 训练速度慢
- 容易过拟合

卷积神经网络

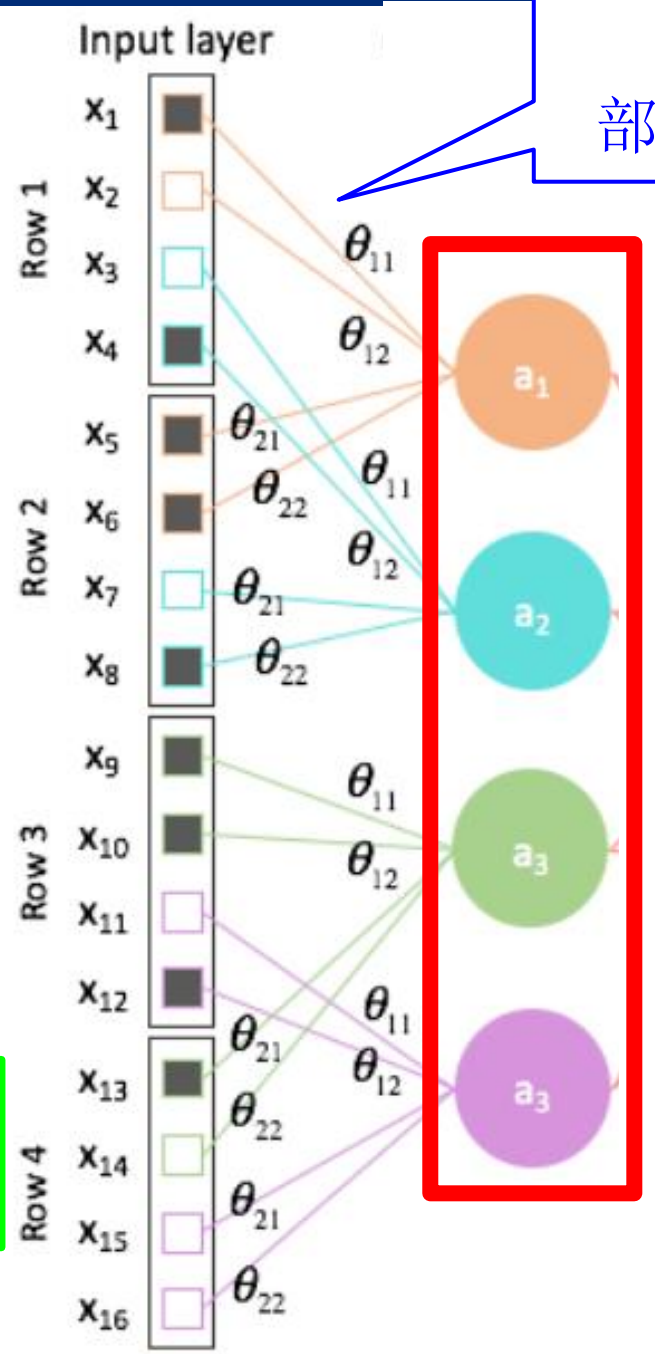
- Convolutional Neural Networks:



考虑空间信息



卷积层Convolution Layer (Filter/Kernel):
假设卷积层每次移动2个单位



利用空间信息：
部分变量(像素)共享权值

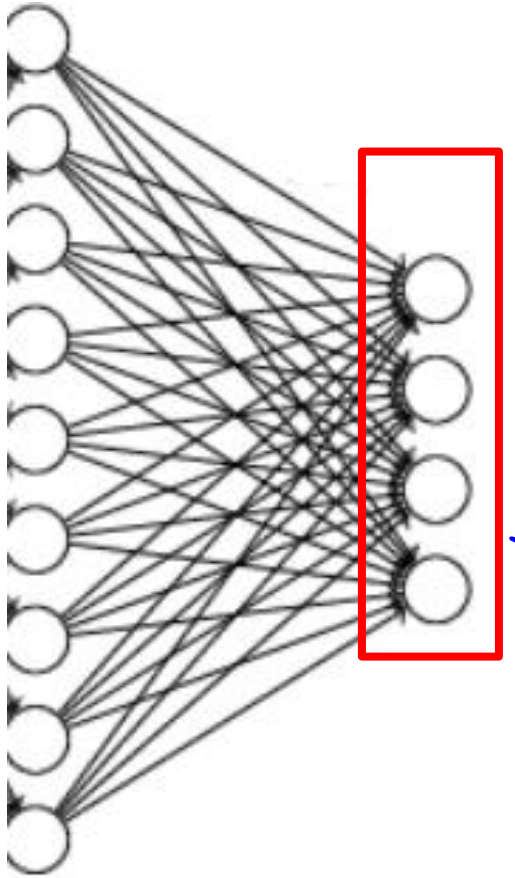
Because interesting features (edges) can happen at anywhere in the image.

输出Feature Map:
with new “pixels”

- 普通神经网络 V.S. 卷积神经网络

普通神经网络

300*300
维度

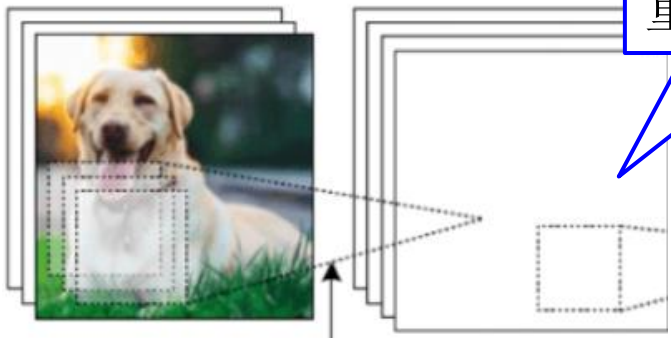


300*300*4+1
大约有360,001权重值

这里假设隐藏层只有 4个神经元，如果增加神经元数量呢？

卷积神经网络

300*300
维度



(5*5+1)*4
只需要104权重值

这里假设隐藏层只有4个5*5维度的卷积层filter/kernel

神经网络：其他模型

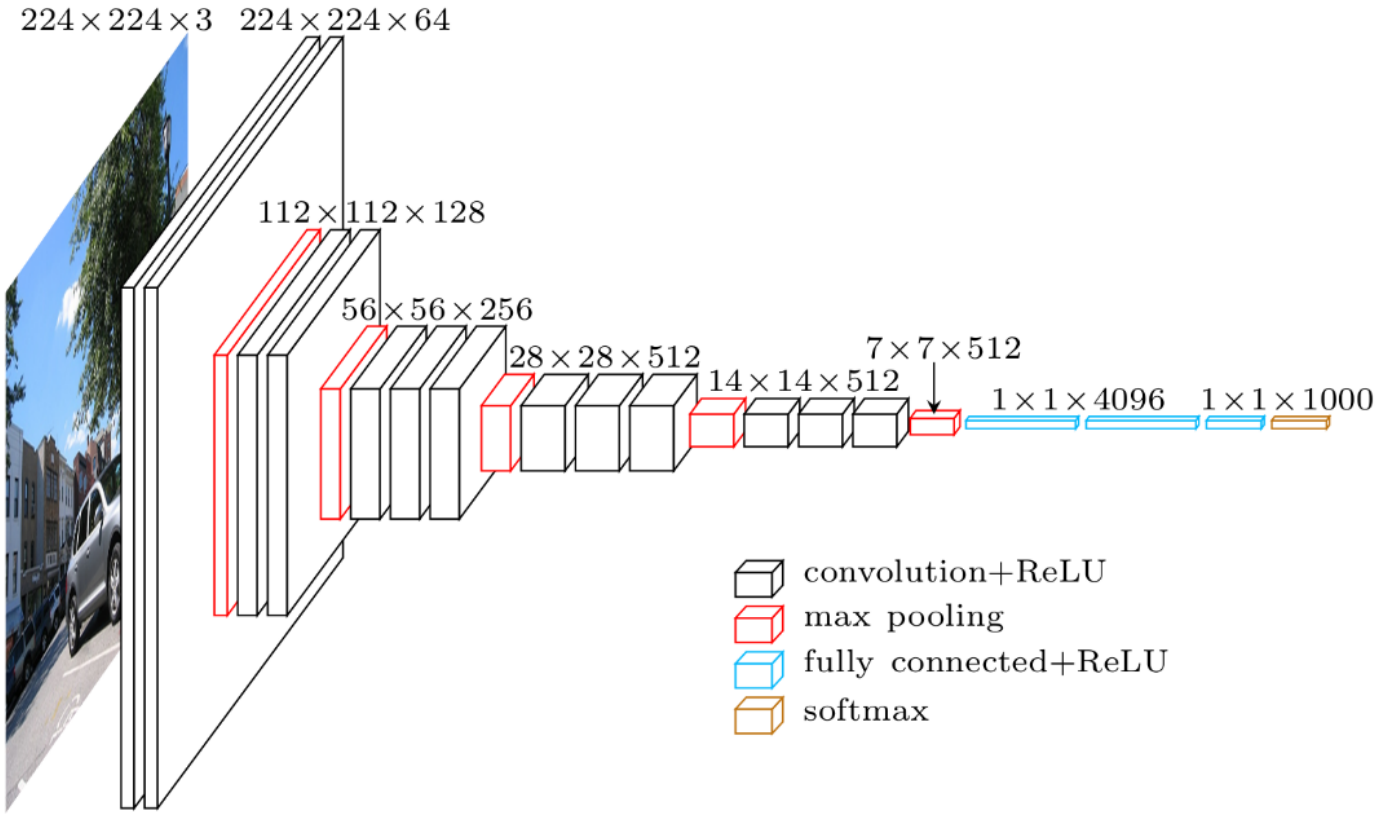
- 循环神经网络 Recurrent Neural Networks (RNN)
  - 神经元的输出值有时间依赖性
  - Long-Short-Term-Memory (LSTM)
- 受限玻尔兹曼机 Restricted Boltzmann Machine (RBM)
- 深度信念网络 Deep Belief Network (DBN)
- 自编码神经网络 Auto-Encoder
- 其他更多…

神经网络：IS学术与实际应用

神经网络：学术应用

➢ Academic Research

- Shunyuan Zhang, Dokyun Lee, Param Vir Singh, Kannan Srinivasan. *How Much is an Image Worth? Airbnb Property Demand Analytics Leveraging A Scalable Image Classification Algorithm*. Working Paper. (https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2976021)



- 研究问题：What is the effect of joining Airbnb photography program (i.e., verified photos)?
- 模型测试：Label image quality on Amazon Mechanical Turk
- 卷积神经网络(VGG-16): Image quality and (12) interpretable image features

神经网络：实际应用

- 计算机视觉
  - 图像分类，面部识别，目标对象检测
  - 视频流媒体挖掘
- 自然语言处理
  - 文本挖掘 (如，内容挖掘，情感分析，语义分析等)
- 时间序列预测
- 语音识别
- 其他更多…

注意：
需要赚更多钱，买GPU跑模型

In God we trust, all others bring data

------William Edwards Deming (1900-1993)

From "The Elements of Statistical Learning" (ESL) by Trevor Hastie, Robert Tibshirani and Jerome Friedman

非常感谢各位！