

《神经网络与深度学习》



前馈神经网络

<https://nndl.github.io/>

复习

- ▶ 请简述神经网络、深度学习两者间的关联。
- ▶ 将有监督学习、无监督学习、强化学习填写到下面的空白处：
 - ▶ 通过海量对局构建实现王者荣耀的自动对战系统，这属于（ ）技术的应用；
 - ▶ 某电商平台根据购买频次、用户年均消费总金额、消费类目等自动将客户归为几个群体，这属于（ ）；
 - ▶ 某水果生产基地根据日均光照小时数、单位面积施肥量等预测苹果的单位面积产量，这属于（ ）。
- ▶ 什么是表示学习？表示学习与深度学习有着怎样的关联？

复习

- ▶ 对于特征的表示，什么是局部表示？什么是分布式表示？试各举一例。
- ▶ 试述特征提取与表示学习的区别。
- ▶ 神经网络的设计需要考虑哪三个方面？
- ▶ 试述人工神经元是如何模拟生物神经元的工作机理的。
- ▶ 列举 ≥ 5 个常见的深度学习计算框架。

腾讯开悟 教育 赛事 开放服务 平台动态 关于我们

赛事 > 2023 腾讯开悟人工智能全国公开赛

2023 腾讯开悟人工智能全国公开赛

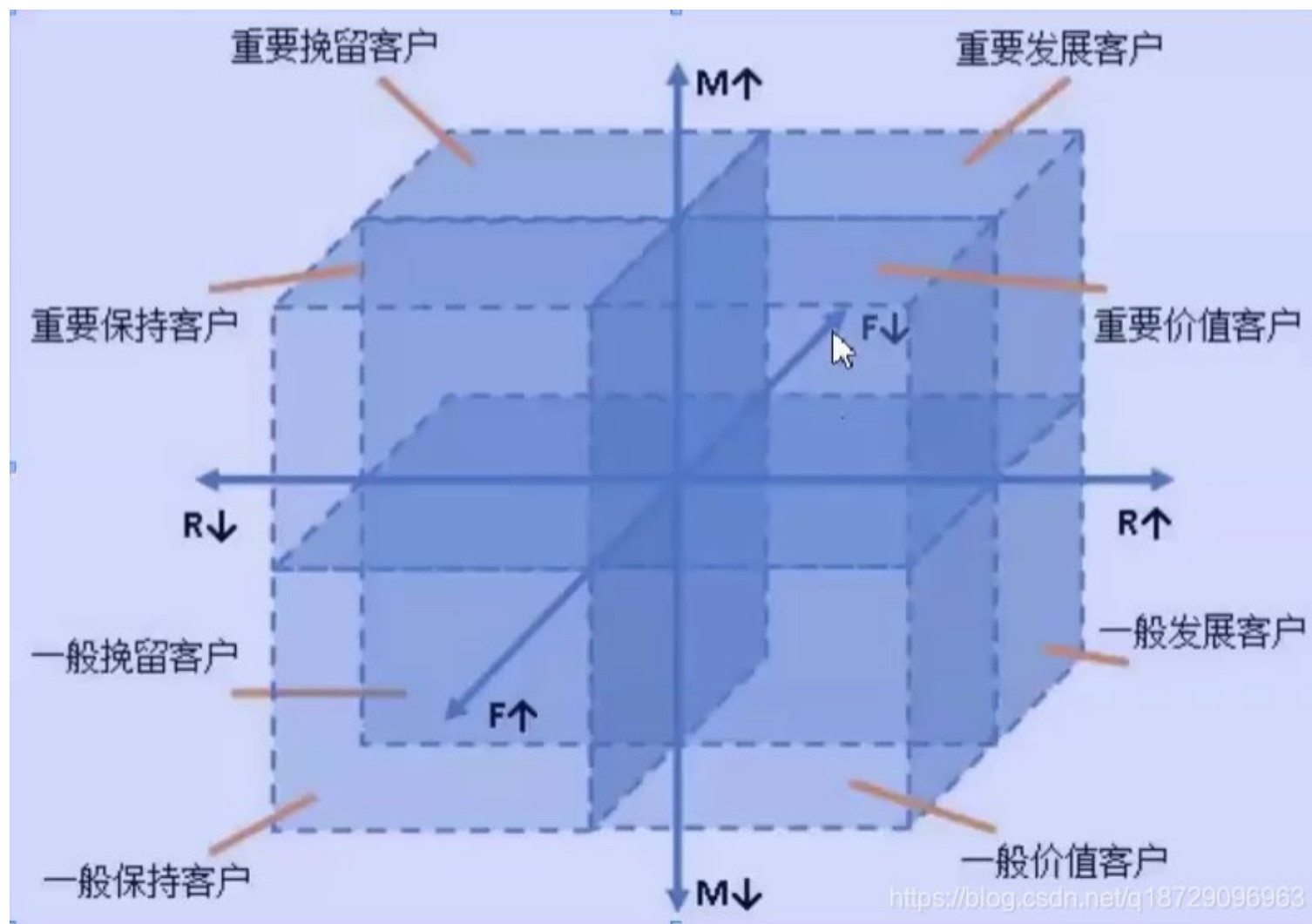
奖金金额 ¥930000 比赛状态 已结束

赛事介绍 排行榜

赛事介绍

2023 腾讯开悟人工智能全国公开赛首次公开接受全国高等院校在校学生报名参赛。赛事集科研资源、专业培训、创新赛事、行业资源于一体，旨在“以赛促学”助力建设全国高校人工智能专业人才培养，建立产学研合作新生态，推动人工智能研究的发

复习



复习



内容

- ▶ 分类问题
- ▶ 感知机
- ▶ 神经网络
 - ▶ 神经元
 - ▶ 网络结构
- ▶ 机器学习的四个要素
- ▶ 前馈神经网络
 - ▶ 参数学习
 - ▶ 计算图与自动微分
 - ▶ 优化问题



分类问题

示例：图像分类

▶ 数据集：CIFAR-10

▶ 60000张32x32色彩图像，共10类

▶ 每类6000张图像

airplane



automobile



bird



cat



deer



dog



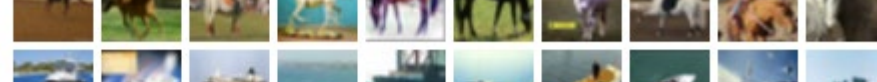
frog



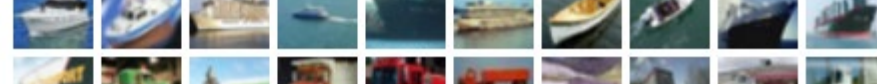
horse



ship



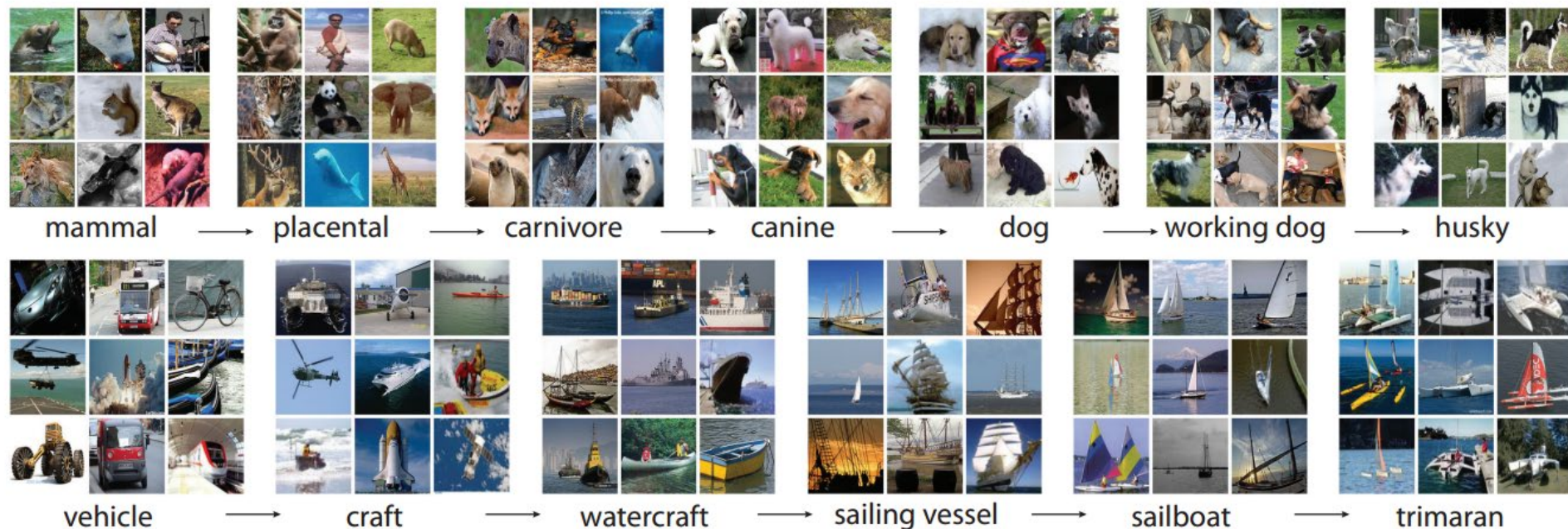
truck



示例：图像分类

► 数据集：ImageNet

► 14,197,122 images, 21841 synsets



示例：图像分类

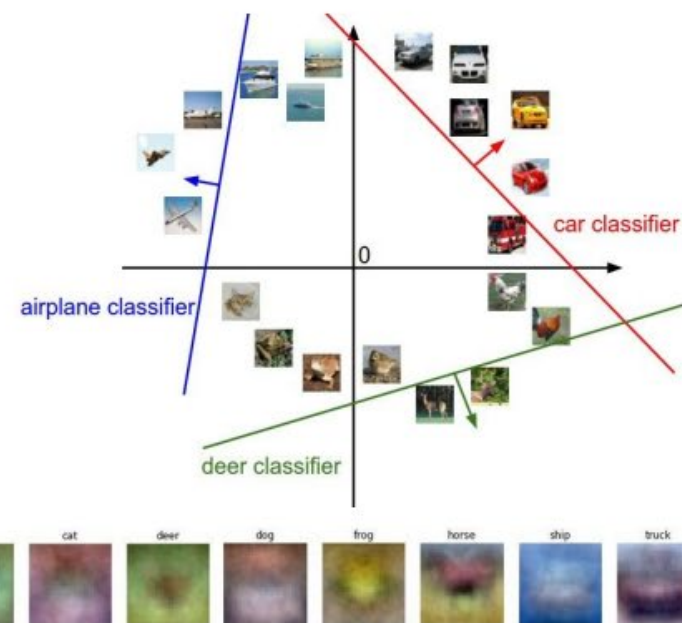
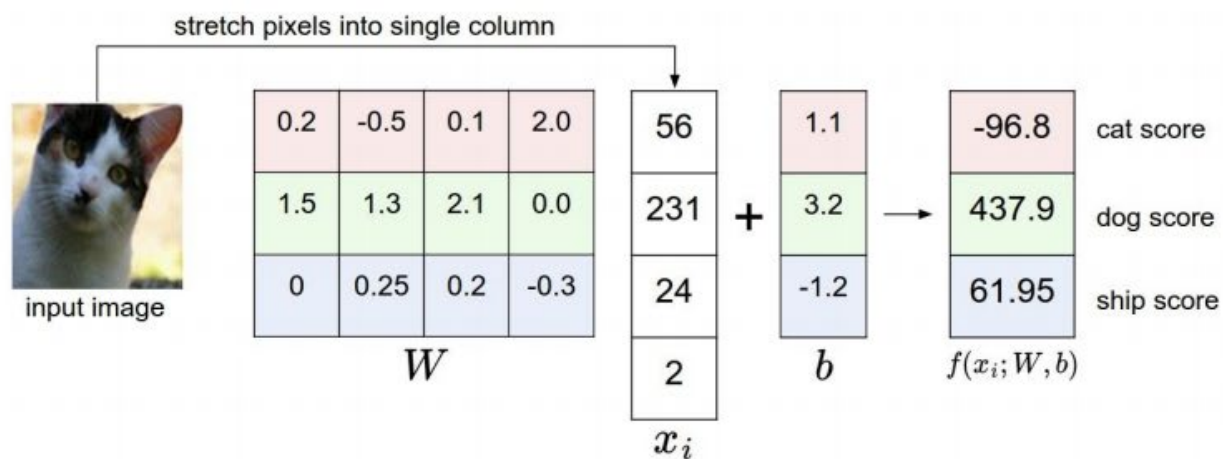


[32x32x3]

array of numbers 0...1
(3072 numbers total)

image parameters
 $f(\mathbf{x}, \mathbf{W})$

10 numbers, indicating
class scores



示例：图像分类、目标检测、实例分割

Classification



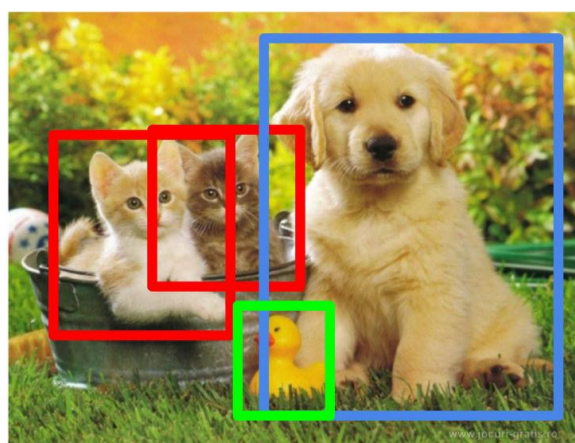
CAT

**Classification
+ Localization**



CAT

Object Detection



CAT, DOG, DUCK

**Instance
Segmentation**



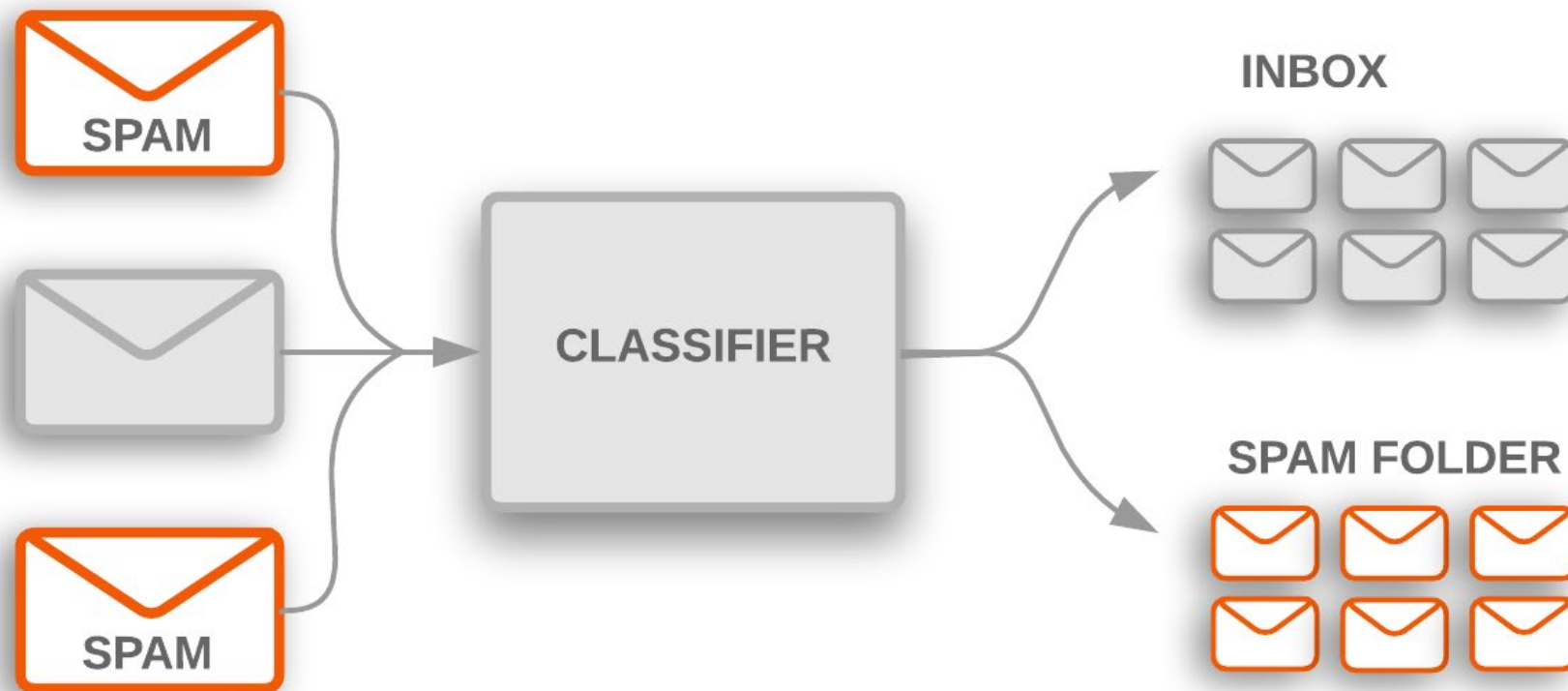
CAT, DOG, DUCK

Single object

Multiple objects

<https://medium.com/zylapp/review-of-deep-learning-algorithms-for-object-detection-c1f3d437b852>

示例：垃圾邮件过滤

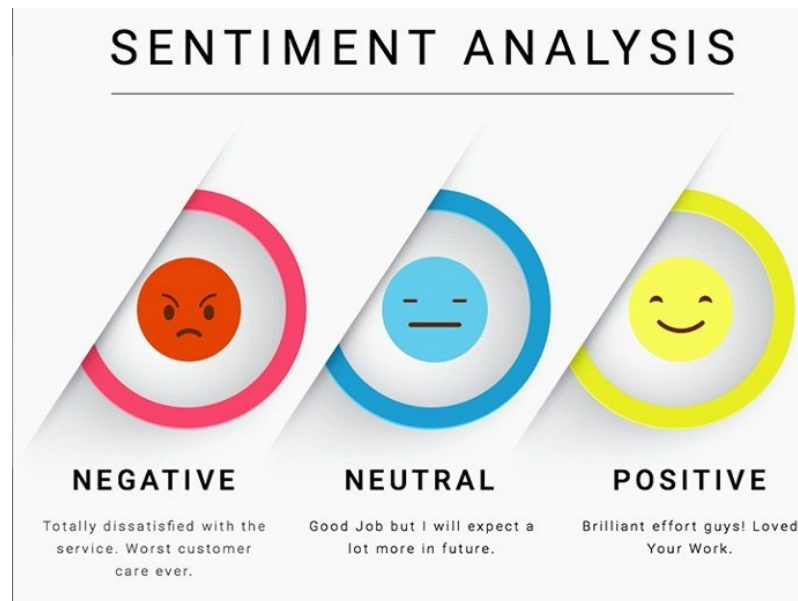


示例：文档归类



<https://towardsdatascience.com/automated-text-classification-using-machine-learning-3df4f4f9570b>

示例：情感分类



Review (X)

Rating (Y)

"This movie is fantastic! I really like it because it is so good!"



"Not to my taste, will skip and watch another movie"



"This movie really sucks! Can I get my money back please?"



示例：文本分类

- ▶ 将样本 x 从文本形式转为向量形式
- ▶ 词袋模型 (Bag-of-Words, BoW) 模型

the dog is on the table

0	0	1	1	0	1	1	1
are	cat	dog	is	now	on	table	the

比如两个文本“我 喜欢 读书”和“我 讨厌 读书”中共有“我”、“喜欢”、“讨厌”、“读书”四个词，它们的BoW表示分别为

$$\mathbf{v}_1 = [1 \ 1 \ 0 \ 1]^T,$$

$$\mathbf{v}_2 = [1 \ 0 \ 1 \ 1]^T.$$

示例：文本情感分类

根据文本内容来判断文本的相应类别

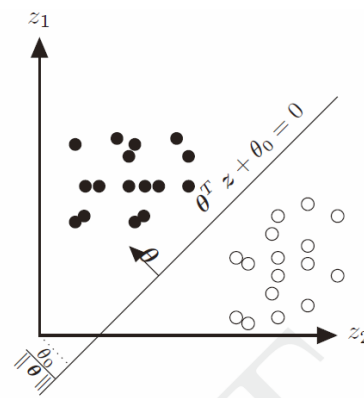
D_1 : “我喜欢读书”

D_2 : “我讨厌读书”

	我	喜欢	讨厌	读书
D_1	1	1	0	1
D_2	1	0	1	1

+

-





感知器

感知器

Psychological Review
Vol. 65, No. 6, 1958

THE PERCEPTRON: A PROBABILISTIC MODEL FOR INFORMATION STORAGE AND ORGANIZATION IN THE BRAIN¹

F. ROSENBLATT

Cornell Aeronautical Laboratory

HAVING told you about the giant digital computer known as I.B.M. 704 and how it has been taught to play a fairly creditable game of chess, we'd like to tell you about an even more remarkable machine, the perceptron, which, as its name implies, is capable of what amounts to original thought. The first perceptron has yet to be built,

The New Yorker, December 6, 1958 P. 44

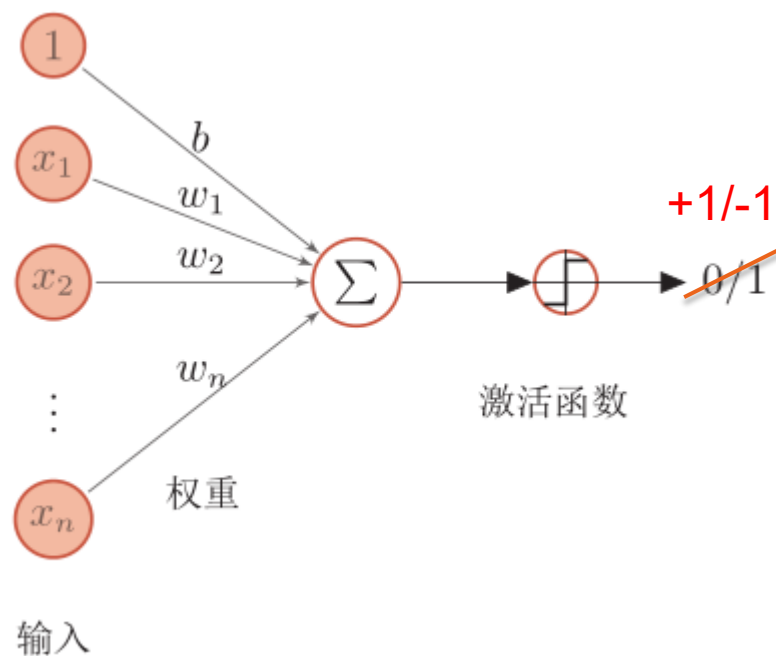


The IBM 704 computer

感知器

- ▶ 模拟生物神经元行为的机器，有与生物神经元相对应的部件，如权重（突触）、偏置（阈值）及激活函数（细胞体），输出为+1或-1。

$$\hat{y} = \begin{cases} +1 & \text{当 } \mathbf{w}^T \mathbf{x} > 0 \\ -1 & \text{当 } \mathbf{w}^T \mathbf{x} \leq 0 \end{cases},$$



感知器

▶ 学习算法

- ▶ 一种错误驱动的在线学习算法：
- ▶ 先初始化一个权重向量 $\mathbf{w} \leftarrow \mathbf{0}$ （通常是全零向量）；
- ▶ 每次分错一个样本 (\mathbf{x}, y) 时，即

$$y\mathbf{w}^T \mathbf{x} < 0$$

- ▶ 用这个样本来更新权重

$$\mathbf{w} \leftarrow \mathbf{w} + y\mathbf{x}$$

- ▶ 根据感知器的学习策略，可以反推出感知器的损失函数为

$$\mathcal{L}(\mathbf{w}; \mathbf{x}, y) = \max(0, -y\mathbf{w}^T \mathbf{x})$$

感知器的学习过程

算法 3.1 两类感知器的参数学习算法

输入: 训练集 $\mathcal{D} = \{(\mathbf{x}^{(n)}, y^{(n)})\}_{n=1}^N$, 最大迭代次数 T

```
1 初始化:  $\mathbf{w}_0 \leftarrow 0, k \leftarrow 0, t \leftarrow 0$ ;  
2 repeat  
3   对训练集  $\mathcal{D}$  中的样本随机排序;  
4   for  $n = 1 \cdots N$  do  
5     选取一个样本  $(\mathbf{x}^{(n)}, y^{(n)})$ ;  
6     if  $\mathbf{w}_k^\top (y^{(n)} \mathbf{x}^{(n)}) \leq 0$  then  
7        $\mathbf{w}_{k+1} \leftarrow \mathbf{w}_k + y^{(n)} \mathbf{x}^{(n)}$ ;  
8        $k \leftarrow k + 1$ ;  
9     end  
10     $t \leftarrow t + 1$ ;  
11    if  $t = T$  then break;  
12  end  
13 until  $t = T$ ;  
    输出:  $\mathbf{w}_k$ 
```

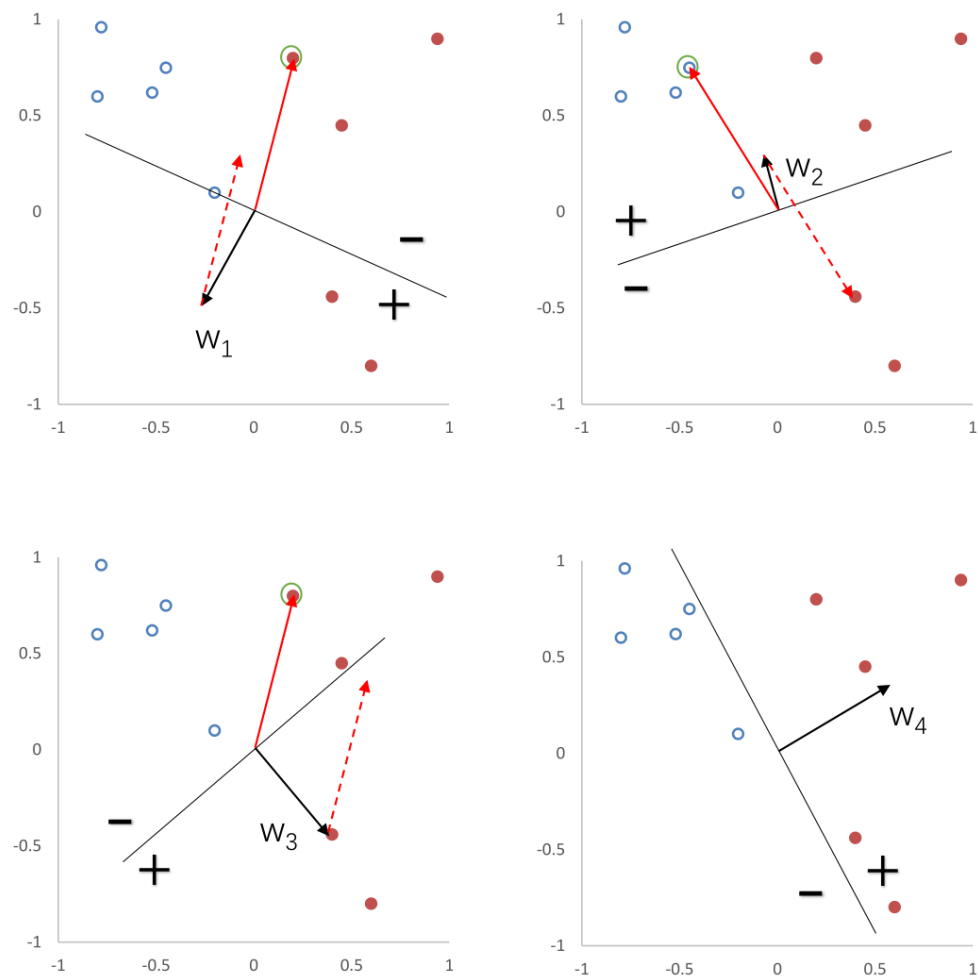
// 达到最大迭代次数

表示分错

对比Logistic回归的更新方式:

$$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + \alpha \frac{1}{N} \sum_{n=1}^N \mathbf{x}^{(n)} (y^{(n)} - \hat{y}_{\mathbf{w}_t}^{(n)})$$

感知器参数学习的更新过程



收敛性

当数据集是两类线性可分时,对于训练集 $\mathcal{D} = \{(\mathbf{x}^{(n)}, y^{(n)})\}_{n=1}^N$, 其中 $\mathbf{x}^{(n)}$ 为样本的增广特征向量, $y^{(n)} \in \{-1, 1\}$, 那么存在一个正的常数 $\gamma (\gamma > 0)$ 和权重向量 \mathbf{w}^* , 并且 $\|\mathbf{w}^*\| = 1$, 对所有 n 都满足 $(\mathbf{w}^*)^\top (y^{(n)} \mathbf{x}^{(n)}) \geq \gamma$. 我们可以证明如下定理.

定理 3.1 – 感知器收敛性: 给定训练集 $\mathcal{D} = \{(\mathbf{x}^{(n)}, y^{(n)})\}_{n=1}^N$, 令 R 是训练集中最大的特征向量的模, 即

$$R = \max_n \|\mathbf{x}^{(n)}\|.$$

如果训练集 \mathcal{D} 线性可分, 两类感知器的参数学习算法 3.1 的权重更新次数不超过 $\frac{R^2}{\gamma^2}$.

证明过程

- ▶ 感知器的权重向量的更新方式为

$$\boldsymbol{w}_k = \boldsymbol{w}_{k-1} + y^{(k)} \boldsymbol{x}^{(k)}$$

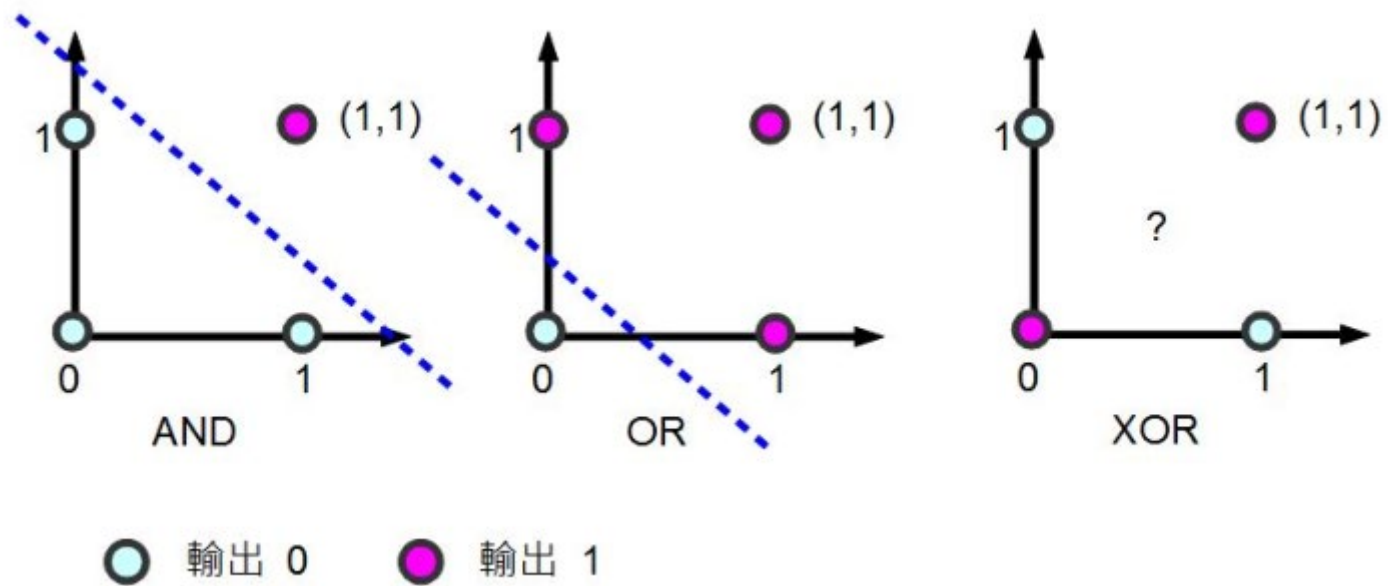
- ▶ 在第 K 次更新时感知器的权重向量为

$$\boldsymbol{w}_K = \sum_{k=1}^K y^{(k)} \boldsymbol{x}^{(k)}$$

- ▶ $\|\boldsymbol{w}_K\|^2$ 的上界

- ▶ $\|\boldsymbol{w}_K\|^2$ 的下界

XOR问题





神经网络

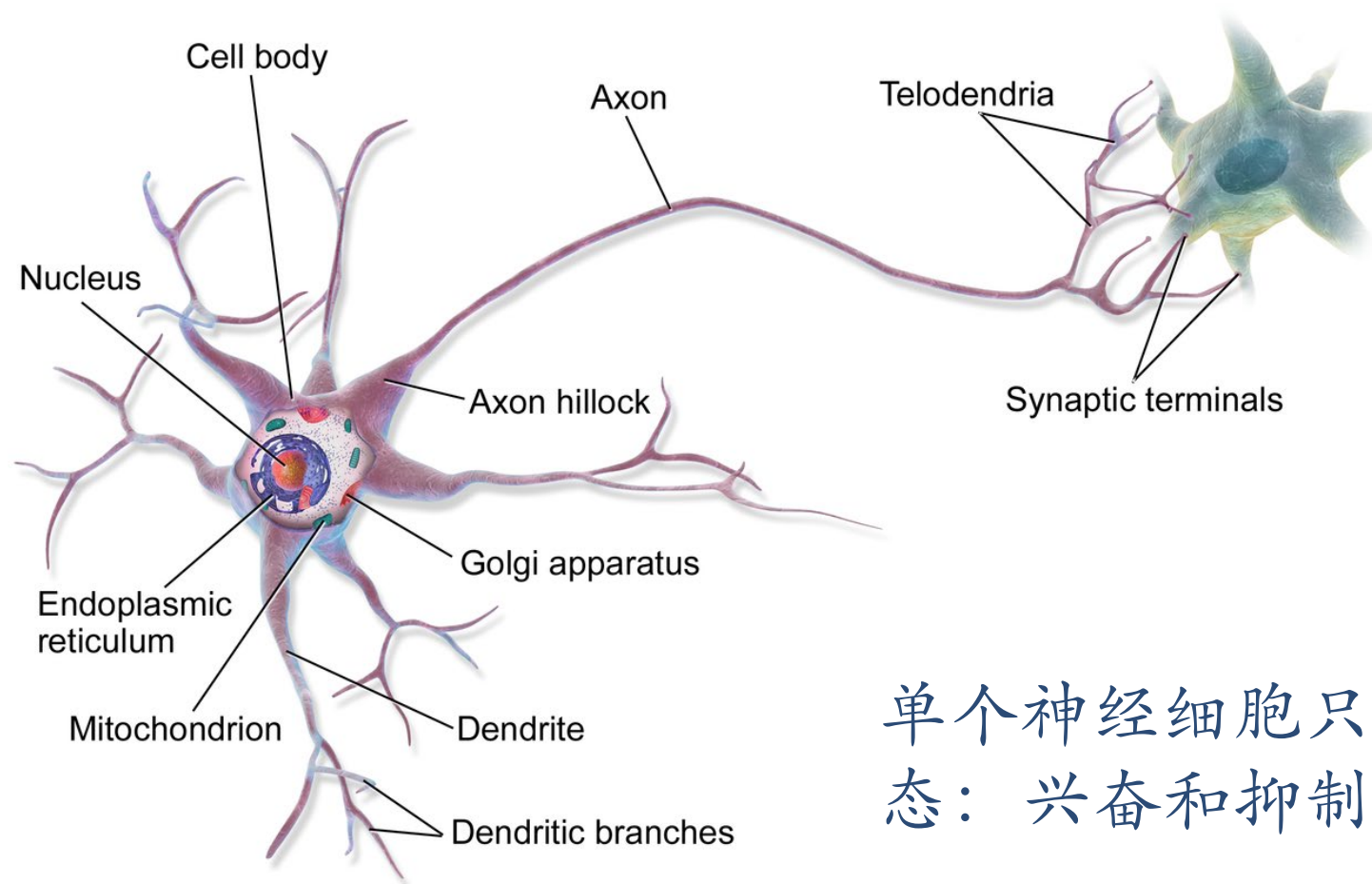
神经网络

- ▶ 神经网络最早是作为一种主要的连接主义模型。
- ▶ 20世纪80年代后期，最流行的一种连接主义模型是分布式并行处理（Parallel Distributed Processing, PDP）网络，其有3个主要特性：
 - ▶ 1) 信息表示是分布式的（非局部的）；
 - ▶ 2) 记忆和知识是存储在单元之间的连接上；
 - ▶ 3) 通过逐渐改变单元之间的连接强度来学习新的知识。
- ▶ 引入误差反向传播来改进其学习能力之后，神经网络也越来越多地应用在各种机器学习任务上。



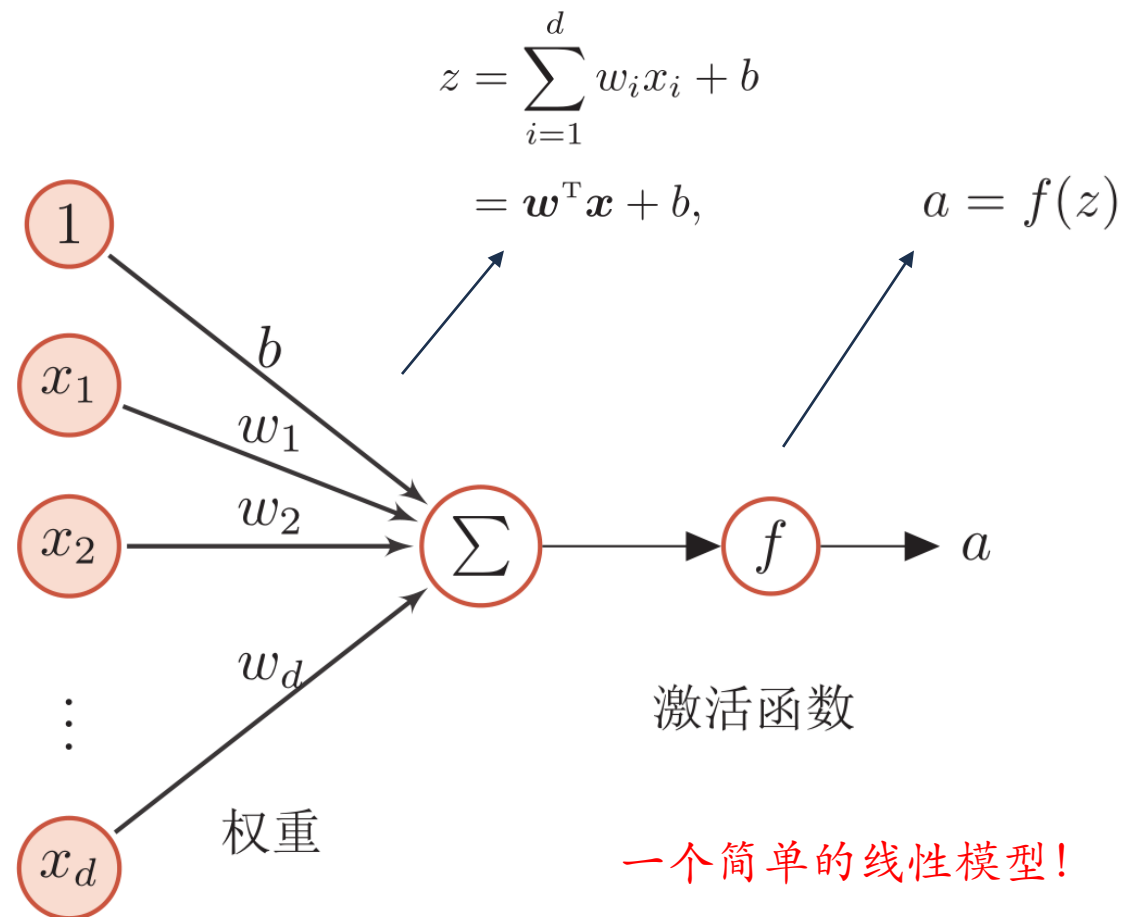
神经元

生物神经元



单个神经细胞只有两种状态：兴奋和抑制

人工神经元



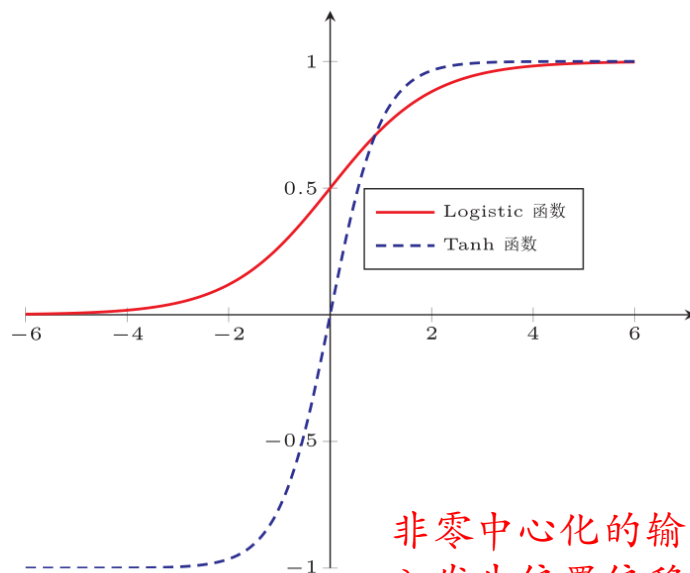
激活函数的性质

- ▶ 连续并可导（允许少数点上不可导）的非线性函数。
 - ▶ 可导的激活函数可以直接利用数值优化的方法来学习网络参数。
- ▶ 激活函数及其导函数要尽可能的简单
 - ▶ 有利于提高网络计算效率。
- ▶ 激活函数的导函数的值域要在一个合适的区间内
 - ▶ 不能太大也不能太小，否则会影响训练的效率和稳定性。
- ▶ 单调递增
 - ▶ ???

常见激活函数

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

$$\tanh(x) = \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)}$$



非零中心化的输出会使得其后的神经元的输入发生偏置偏移 (bias shift)，并进一步使得梯度下降的收敛速度变慢。

► 性质：

► 光滑、严格单调的饱和函数

► Tanh函数是零中心化的，而logistic函数的输出恒大于0

常见激活函数

$$\text{ReLU}(x) = \begin{cases} x & x \geq 0 \\ 0 & x < 0 \end{cases}$$

$$= \max(0, x).$$

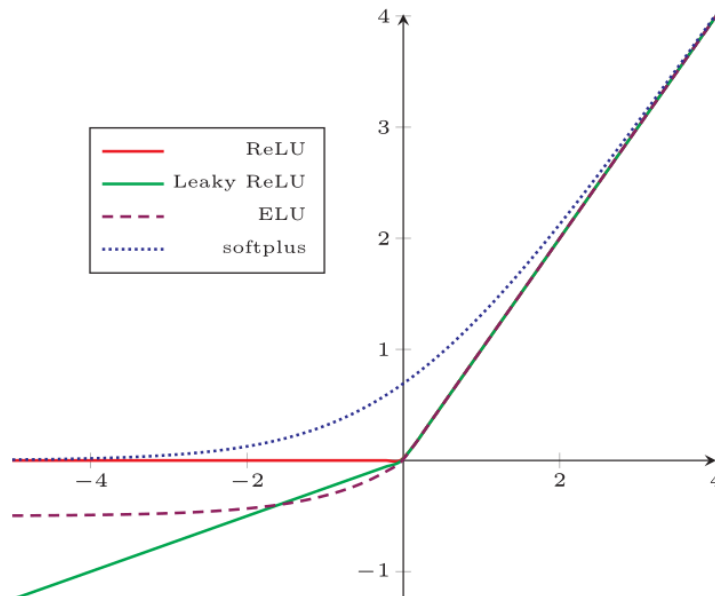
$$\text{LeakyReLU}(x) = \begin{cases} x & \text{if } x > 0 \\ \gamma x & \text{if } x \leq 0 \end{cases}$$

$$\text{PReLU}_i(x) = \begin{cases} x & \text{if } x > 0 \\ \gamma_i x & \text{if } x \leq 0 \end{cases}$$

$$\text{ELU}(x) = \begin{cases} x & \text{if } x > 0 \\ \gamma(\exp(x) - 1) & \text{if } x \leq 0 \end{cases}$$

$$= \max(0, x) + \min(0, \gamma(\exp(x) - 1))$$

$$\text{softplus}(x) = \log(1 + \exp(x))$$

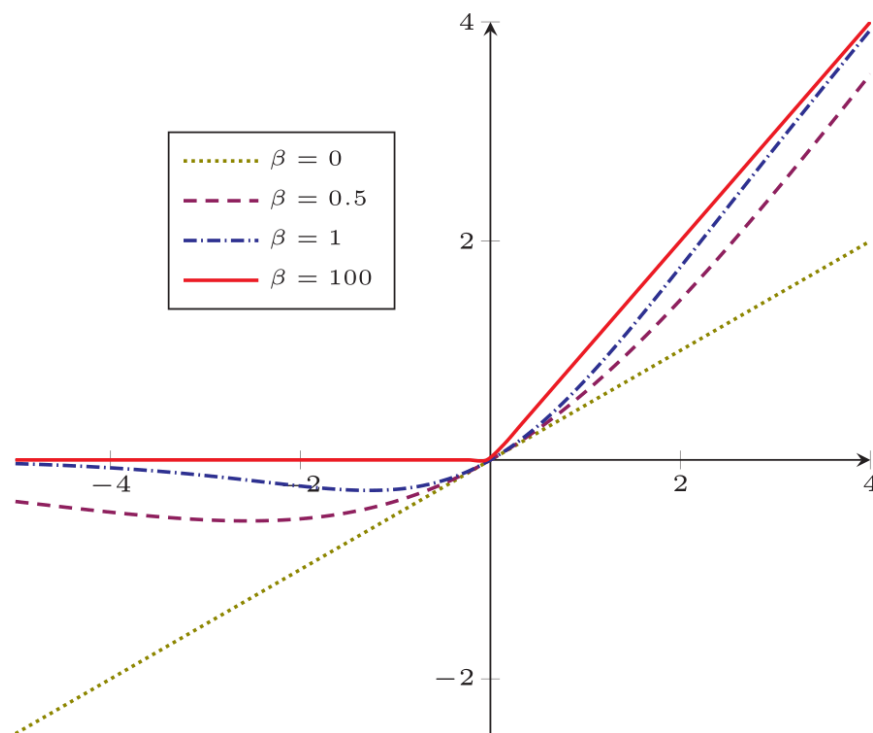


- ▶ 计算上更加高效
- ▶ 生物学合理性
 - ▶ 单侧抑制、宽兴奋边界
- ▶ 在一定程度上缓解梯度消失问题

死亡ReLU问题 (Dying ReLU Problem)

常见激活函数

Swish函数 $\text{swish}(x) = x\sigma(\beta x)$



常见激活函数

► 高斯误差线性单元 (Gaussian Error Linear Unit, GELU)

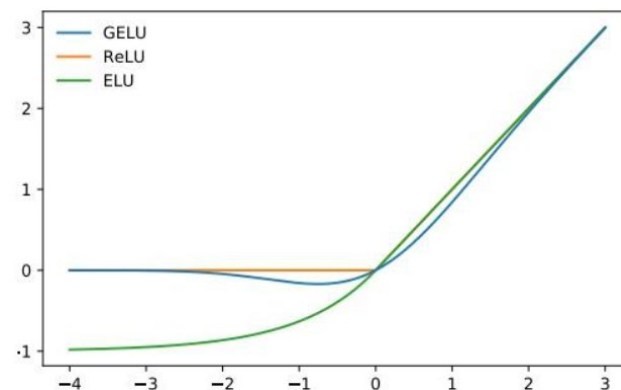
$$\text{GELU}(x) = xP(X \leq x)$$

► 其中 $P(X \leq x)$ 是高斯分布 $N(\mu, \sigma^2)$ 的累积分布函数，其中 μ, σ 为超参数，一般设 $\mu = 0, \sigma = 1$ 即可

► 由于高斯分布的累积分布函数为S型函数，因此GELU可以用Tanh函数或Logistic函数来近似

$$\text{GELU}(x) \approx 0.5x \left(1 + \tanh \left(\sqrt{\frac{2}{\pi}} (x + 0.044715x^3) \right) \right)$$

或 $\text{GELU}(x) \approx x\sigma(1.702x).$



常见激活函数及其导数

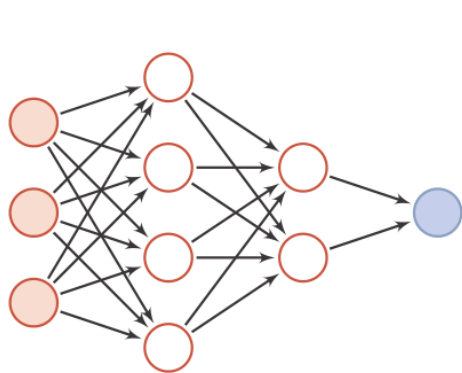
激活函数	函数	导数
Logistic 函数	$f(x) = \frac{1}{1+\exp(-x)}$	$f'(x) = f(x)(1 - f(x))$
Tanh 函数	$f(x) = \frac{\exp(x)-\exp(-x)}{\exp(x)+\exp(-x)}$	$f'(x) = 1 - f(x)^2$
ReLU 函数	$f(x) = \max(0, x)$	$f'(x) = I(x > 0)$
ELU 函数	$f(x) = \max(0, x) + \min(0, \gamma(\exp(x) - 1))$	$f'(x) = I(x > 0) + I(x \leq 0) \cdot \gamma \exp(x)$
SoftPlus 函数	$f(x) = \log(1 + \exp(x))$	$f'(x) = \frac{1}{1+\exp(-x)}$

人工神经网络

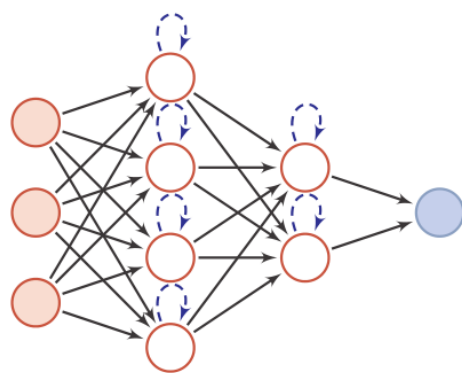
- ▶ 人工神经网络主要由大量的神经元以及它们之间的有向连接构成。因此考虑三方面：
 - ▶ 神经元的激活规则
 - ▶ 主要是指神经元输入到输出之间的映射关系，一般为非线性函数。
 - ▶ 网络的拓扑结构
 - ▶ 不同神经元之间的连接关系。
 - ▶ 学习算法
 - ▶ 通过训练数据来学习神经网络的参数。

网络结构

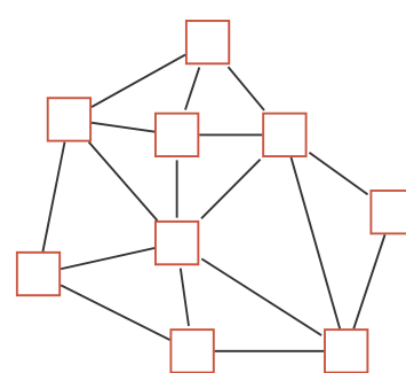
- ▶ 人工神经网络由神经元模型构成，这种由许多神经元组成的信息处理网络具有并行分布结构。



(a) 前馈网络



(b) 记忆网络



(c) 图网络

圆形节点表示一个神经元，方形节点表示一组神经元。