



中国科学院大学
University of Chinese Academy of Sciences

深度学习基础





目录



AI DISCOVERY

更多AI知识，请关注



1

深度学习方法（上）

深度学习发展历程、为什么使用深度学习、如何使用深度学习

2

深度学习方法（下）

深度学习使用方法使用步骤、正向传播、反向传播

3

PaddlePaddle入门

概述、开发环境、使用入门

4

课程实践

实践：房价预测、手写数字识别



AI DISCOVERY



目录



AI DISCOVERY

更多AI知识，请关注



1

深度学习方法（上）

深度学习发展历程、为什么使用深度学习

2

深度学习方法（下）

深度学习方法使用步骤、正向传播、反向传播

3

PaddlePaddle入门

概述、开发环境、使用入门

4

课程实践

实践：房价预测、手写数字识别



AI DISCOVERY



深度学习方法（上）



AI DISCOVERY

深度学习发展历程

为什么使用深度学习

如何使用深度学习



AI DISCOVERY



深度学习与图像识别



AI DISCOVERY

深度学习起源与发展



1958 Perceptron

1974 Backpropagation



Convolution Neural Networks for
Handwritten Recognition



1998



Google Brain Project on
16k Cores

2012

awkward silence (AI Winter)

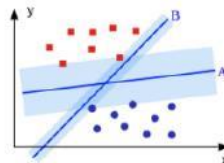
1969

Perceptron criticized



1995

SVM reigns



2006

Restricted
Boltzmann
Machine



2012

AlexNet wins
ImageNet
IMAGENET



AI DISCOVERY



深度学习与图像识别



AI DISCOVERY

深度学习在图像领域的应用



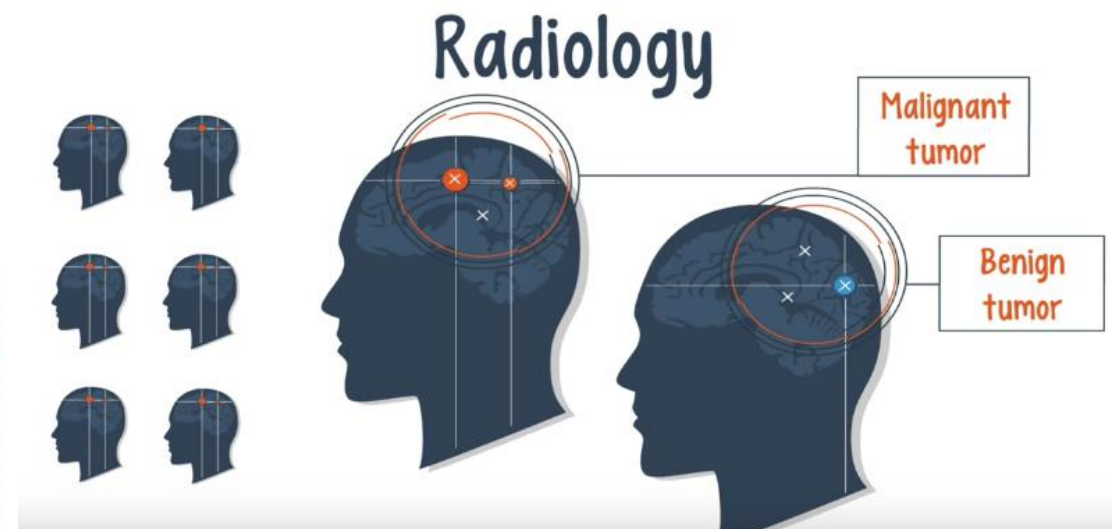
Predicted Tags

no person table elegant
indoors luxury furniture
fashion decoration tableware
party

Similar Images



图片检索



识别异常的肿瘤或者癌细胞



AI DISCOVERY



深度学习与图像识别



AI DISCOVERY

深度学习在图像领域的应用



图片描述



图片着色



AI DISCOVERY



深度学习方法（上）



AI DISCOVERY

深度学习发展历程

为什么使用深度学习

如何使用深度学习



AI DISCOVERY



图像识别为什么使用深度学习

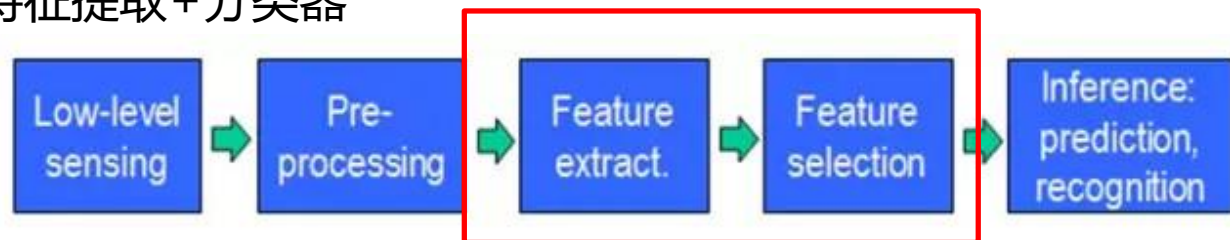
AI DISCOVERY

测量空间 $\xrightarrow{\text{red arrow}}$ 特征空间 $\xrightarrow{\text{blue arrow}}$ 类别空间

核心任务

传统方法

- 人工特征提取+分类器



深度学习

- 手工地选取特征是一件非常费力、启发式（需要专业知识）的方法，能不能选取好很大程度上靠经验和运气，而且它的调节需要大量的时间

AI DISCOVERY



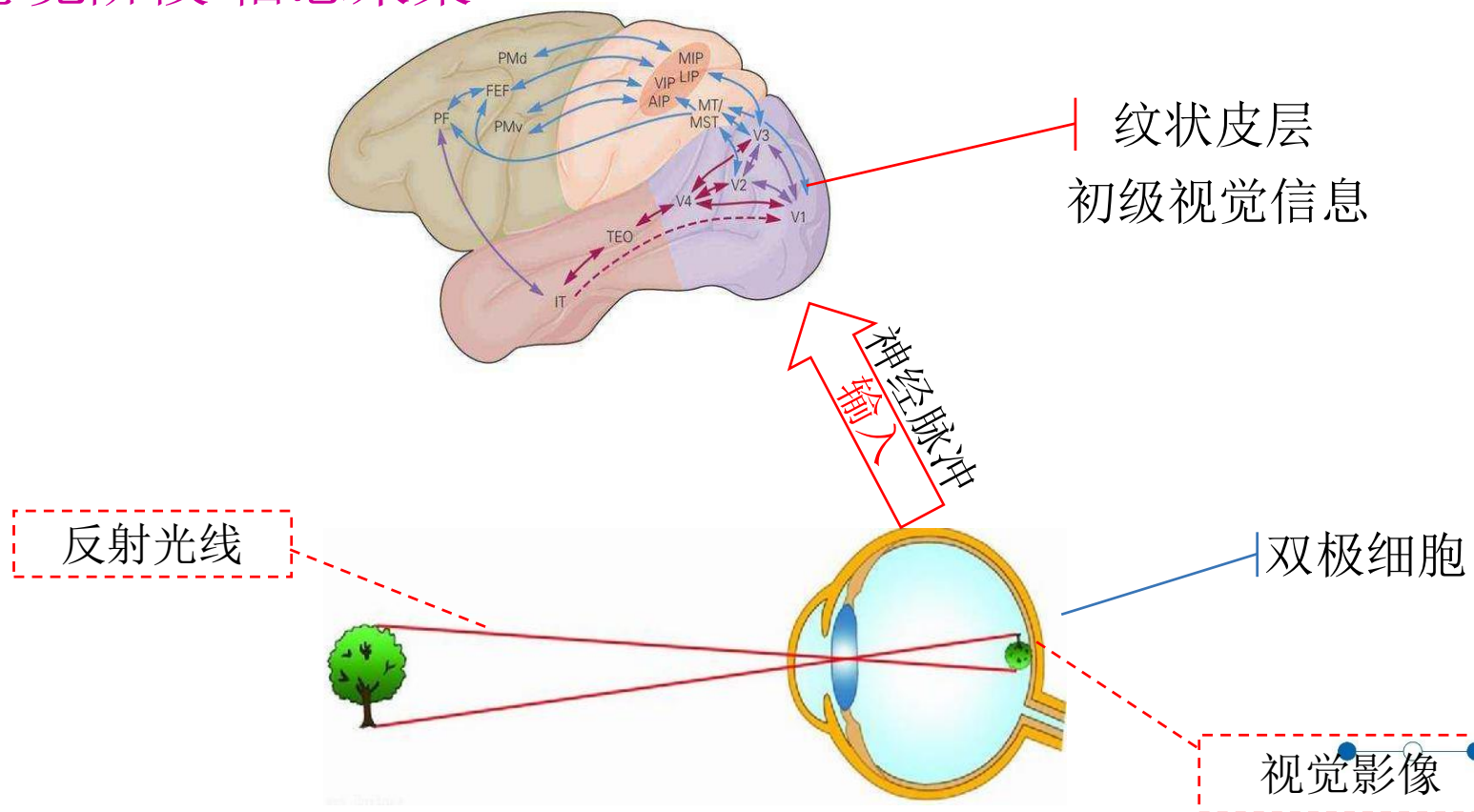
图像识别为什么用深度学习



AI DISCOVERY

人脑视觉机理

i. 视感觉阶段-信息采集



AI DISCOVERY



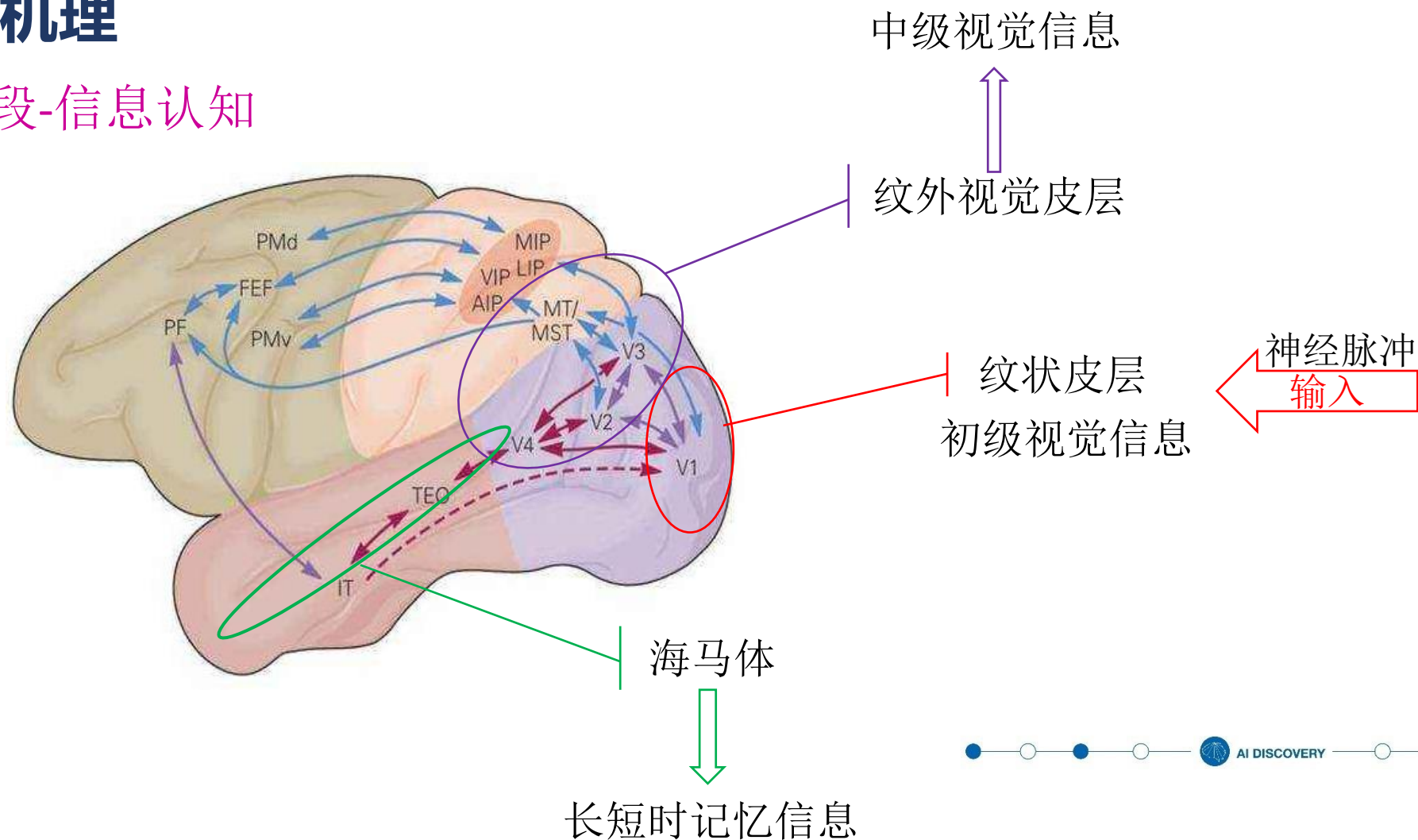
图像识别为什么用深度学习



AI DISCOVERY

人脑视觉机理

ii. 视知觉阶段-信息认知



AI DISCOVERY



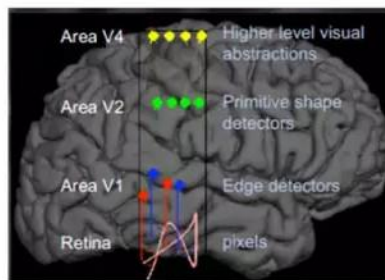
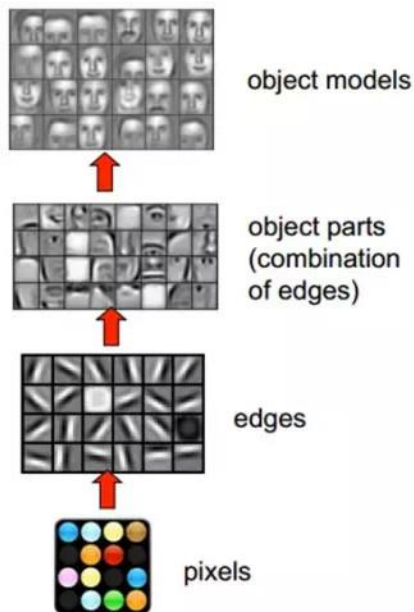
图像识别为什么用深度学习



AI DISCOVERY

人脑视觉机理：神经-中枢-大脑

- 从原始信号摄入（瞳孔摄入像素 Pixels）→初步处理（大脑皮层某些细胞发现边缘和方向）→抽象（大脑判定，眼前的物体的形状，是圆形的）→进一步抽象（大脑进一步判定该物体是人脸）
- 从原始信号，做低级抽象，逐渐向高级抽象迭代



特征空间（概念）



测量空间（表象）



AI DISCOVERY

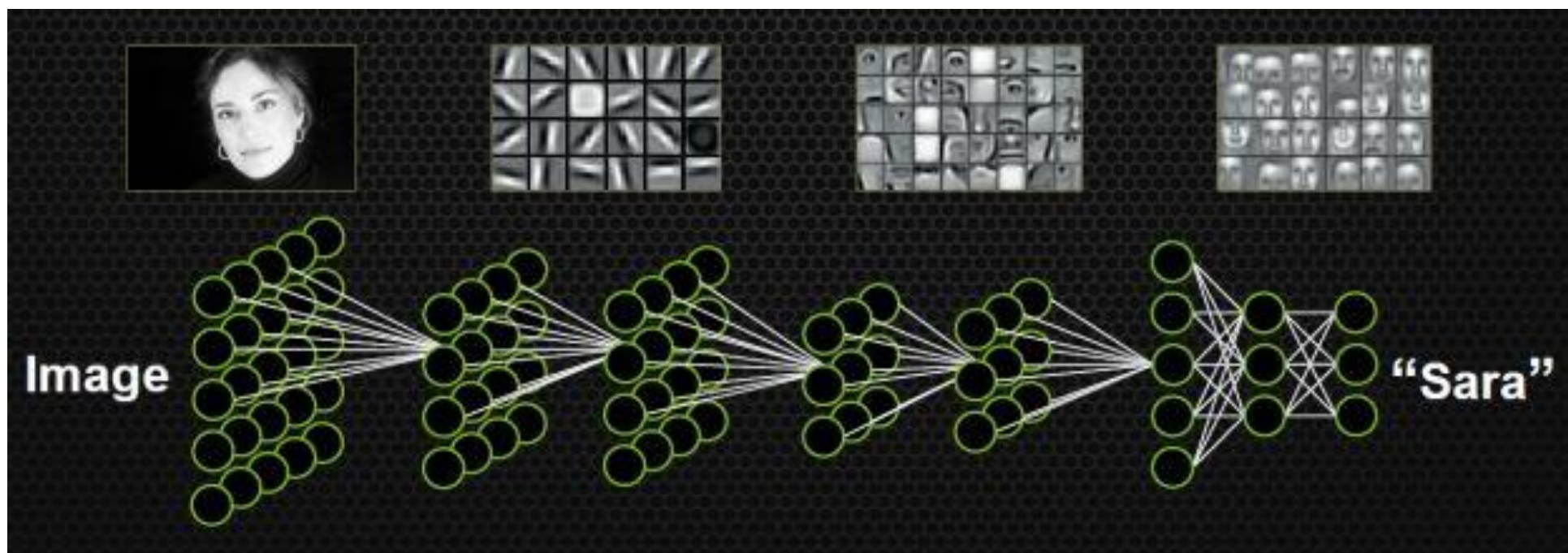


图像识别为什么用深度学习



AI DISCOVERY

深度学习模仿人类的视觉系统



AI DISCOVERY



深度学习方法（上）



AI DISCOVERY

深度学习发展历程

为什么使用深度学习

如何使用深度学习



AI DISCOVERY





如何使用深度学习解决图像识别



AI DISCOVERY

◆ 使用机器学习（深度学习）的目的：寻找一个合适的函数

- 数字识别

$$f(\text{3}) = \text{"3"}$$

- 猫狗识别

$$f(\text{cat}) = \text{"cat"}$$

- 人脸识别

$$f(\text{隔壁老王}) = \text{"隔壁老王"}$$



AI DISCOVERY



如何使用深度学习解决图像识别

◆ 图像识别

学习任务: 寻找手写识别函数, 候选函数集合为

$f, g, h \dots$

$$f\left(\boxed{\text{2}}\right) = \text{"烫烫烫烫"}$$

$$g\left(\boxed{\text{2}}\right) = \text{"8"}$$

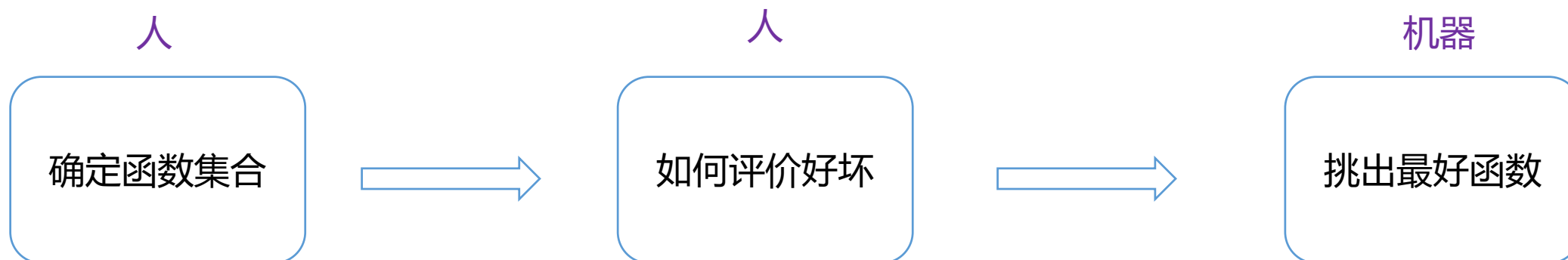
$$h\left(\boxed{\text{2}}\right) = \text{"2"}$$



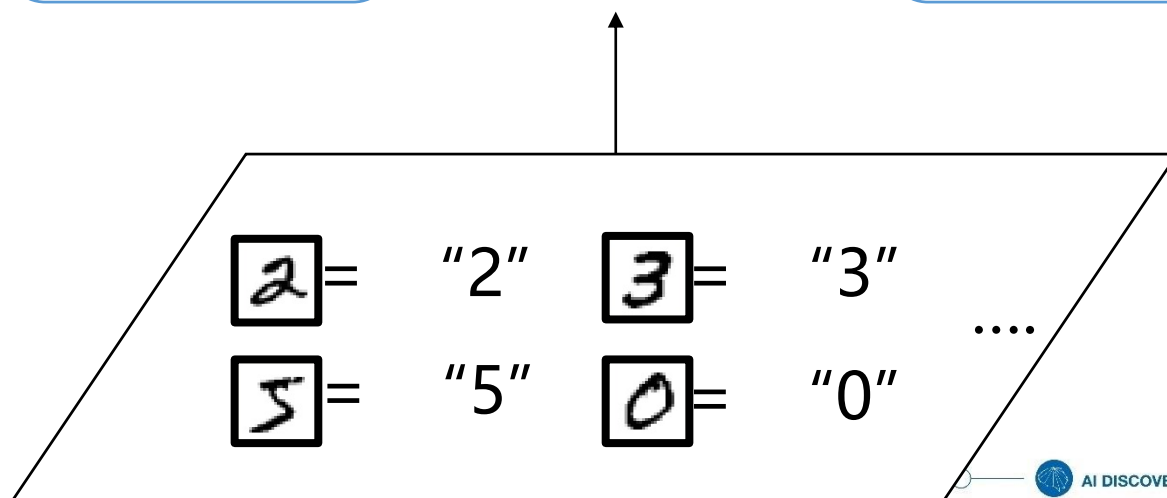
如何使用深度学习解决图像识别

AI DISCOVERY

◆ 学习过程



怎么做：对答案



答案

AI DISCOVERY



如何使用深度学习解决图像识别

◆ 学习过程

人

建立模型



人

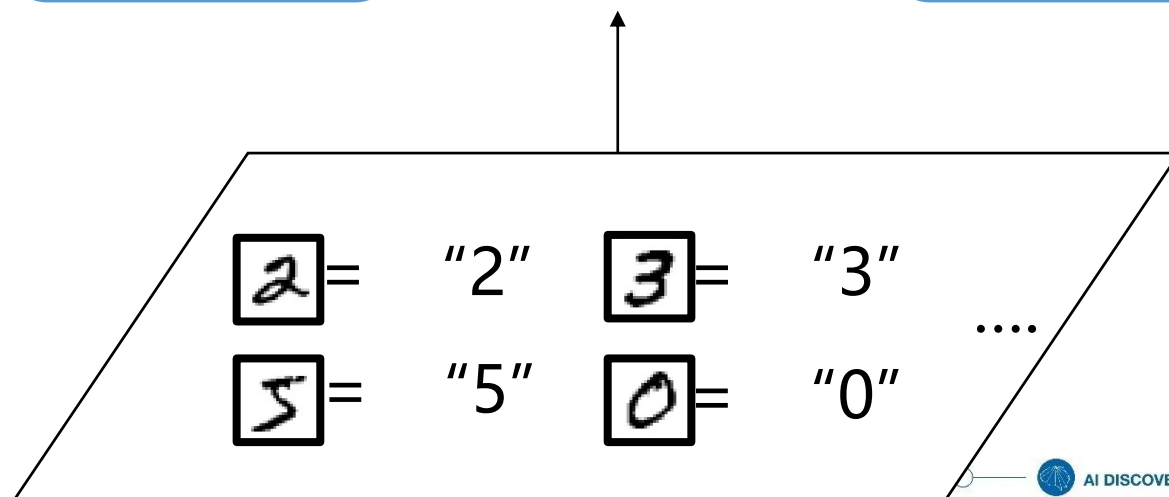
损失函数



机器

参数学习

监督学习 (Supervised Learning)



训练数据



如何使用深度学习解决图像识别

AI DISCOVERY

◆ 图像识别

人

建立模型

$$\{f(x) = kx | k = 1, 2, 3, \dots\}$$

人

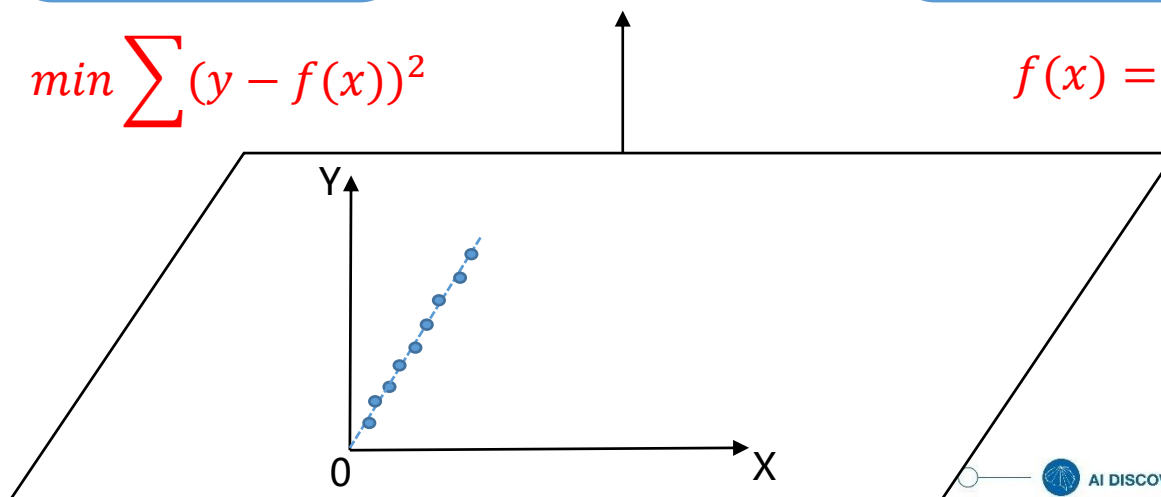
损失函数

$$\min \sum (y - f(x))^2$$

机器

参数学习

$$f(x) = 2x$$



答案

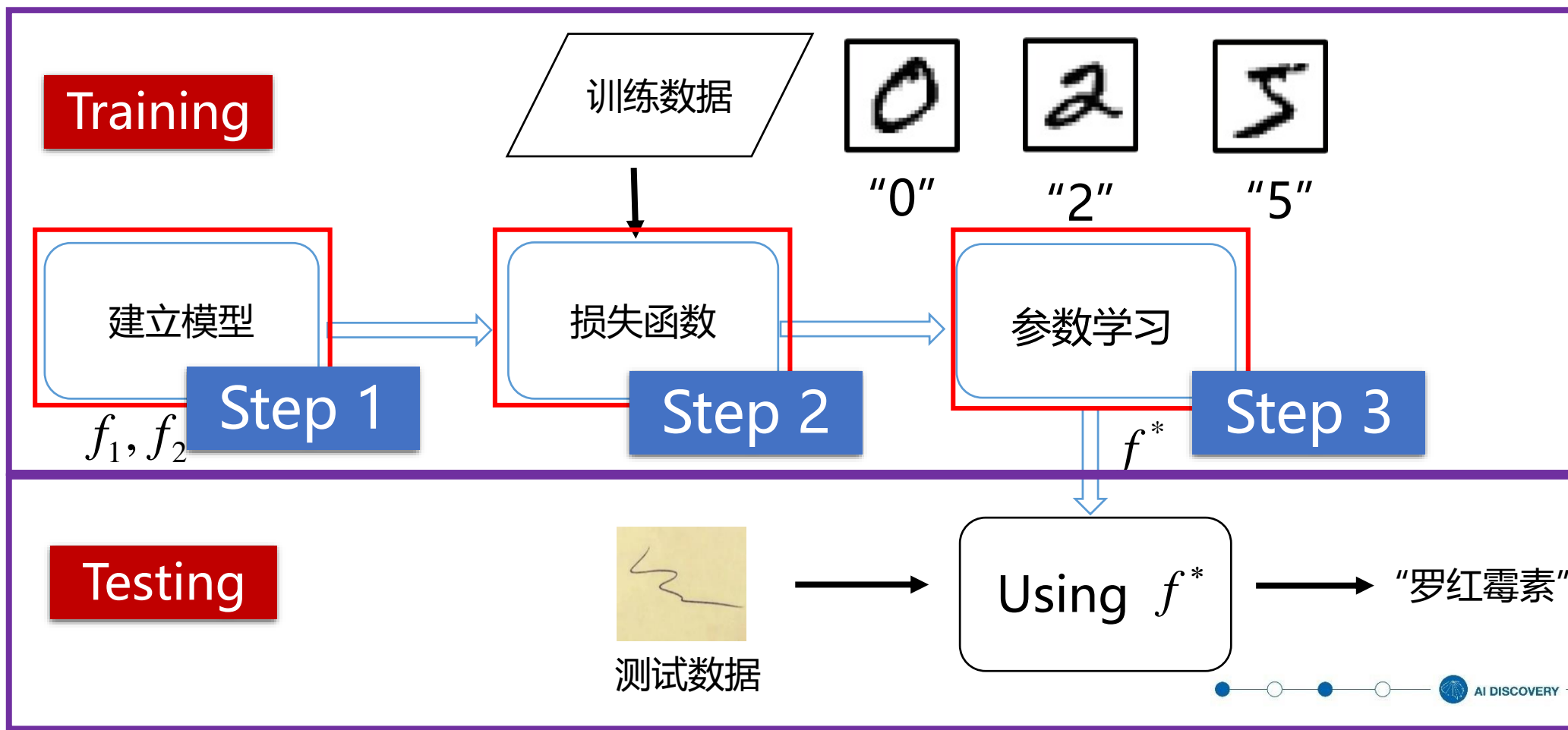
AI DISCOVERY



如何使用深度学习解决图像识别

◆ 图像识别

监督学习任务: 手写识别





目录



AI DISCOVERY

更多AI知识，请关注



1

深度学习方法（上）

深度学习发展历程、为什么使用深度学习

2

深度学习方法（下）

深度学习使用方法使用步骤、正向传播、反向传播

3

PaddlePaddle入门

概述、开发环境、使用入门

4

课程实践

实践：房价预测、手写数字识别



AI DISCOVERY



深度学习方法（下）



AI DISCOVERY

深度学习方法使用步骤

正向传播

反向传播



AI DISCOVERY



使用步骤

AI DISCOVERY

★ 建立模型

- 选择什么样的网络结构
- 选择多少层数，每层选择多少神经元

★ 损失函数

- 选择常用损失函数，平方误差，交叉熵....

★ 参数学习

- 梯度下降
- 反向传播算法

AI DISCOVERY



使用步骤

AI DISCOVERY

★ 建立模型

- 选择什么样的网络结构
- 选择多少层数，每层选择多少神经元

★ 损失函数

- 选择常用损失函数，平方误差，交叉熵....

★ 参数学习

- 梯度下降
- 反向传播算法

AI DISCOVERY

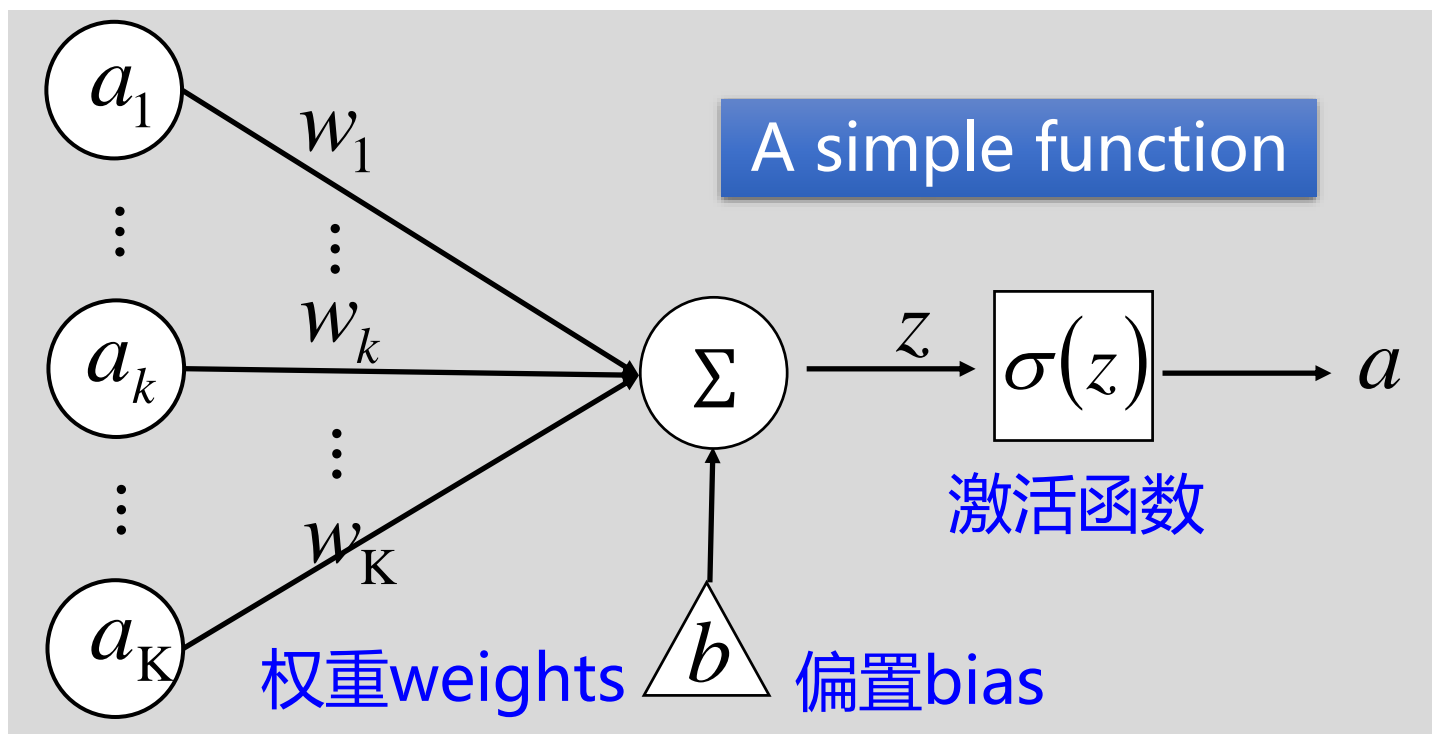


建立模型

AI DISCOVERY

★ 神经元

$$z = a_1 w_1 + \cdots + a_k w_k + \cdots + a_K w_K + b$$



理解：简单线性函数

$$f(x) = kx + b$$

(k是斜率, b是截距)

AI DISCOVERY

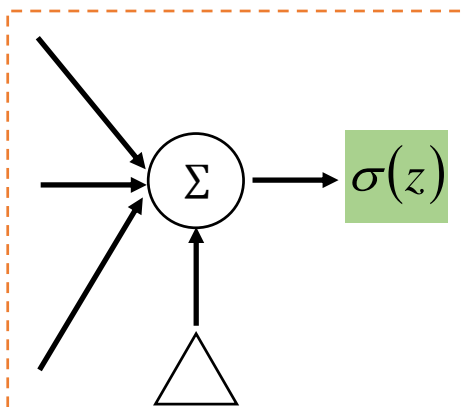
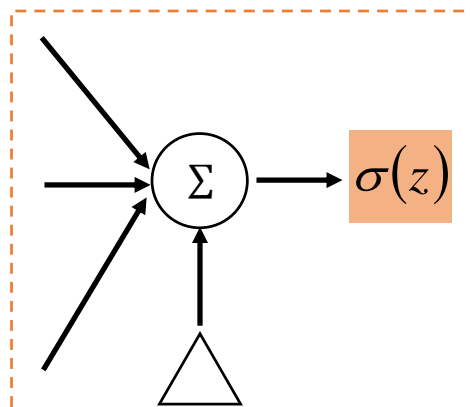
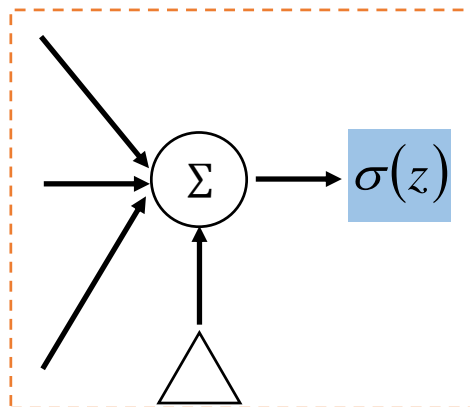
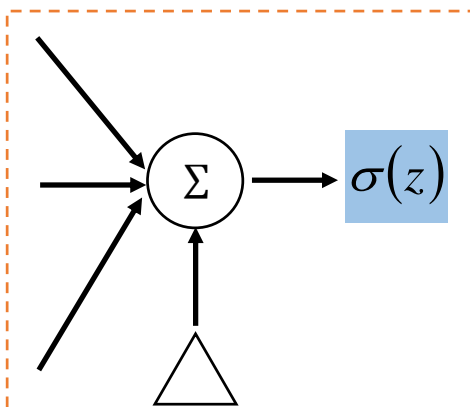


建立模型

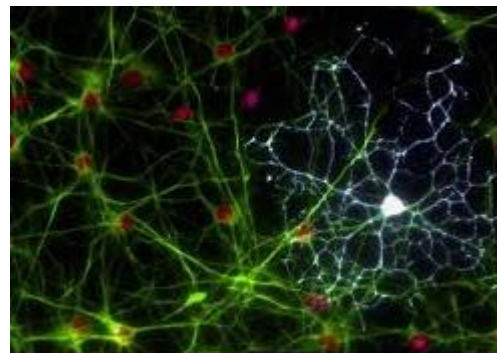
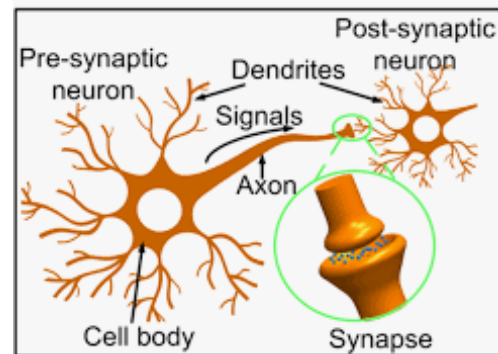
AI DISCOVERY

★ 网络结构

◆ 神经元不同的连接方式构成不同的网络结构



每个神经元都有自己的权重和偏置参数



AI DISCOVERY



建立模型

AI DISCOVERY

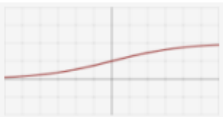
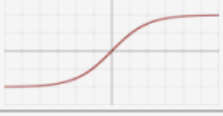


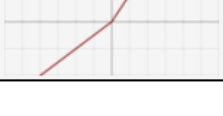


建立模型

神经元

◆ 为什么引入激活函数

- 为了增强网络的表达能力，我们需要激活函数来将线性函数->非线性函数
- 非线性的激活函数需要有连续性。因为连续非线性激活函数可以可导的，所以可以用最优化的方法来求解

Logistic (a.k.a Soft step)		$f(x) = \frac{1}{1 + e^{-x}}$	$f'(x) = f(x)(1 - f(x))$
TanH		$f(x) = \tanh(x) = \frac{2}{1 + e^{-2x}} - 1$	$f'(x) = 1 - f(x)^2$
ArcTan		$f(x) = \tan^{-1}(x)$	$f'(x) = \frac{1}{x^2 + 1}$
Rectified Linear Unit (ReLU) ^[7]		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Parameteric Rectified Linear Unit (PReLU) ^[8]		$f(x) = \begin{cases} \alpha x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} \alpha & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$

常用激活函数示例

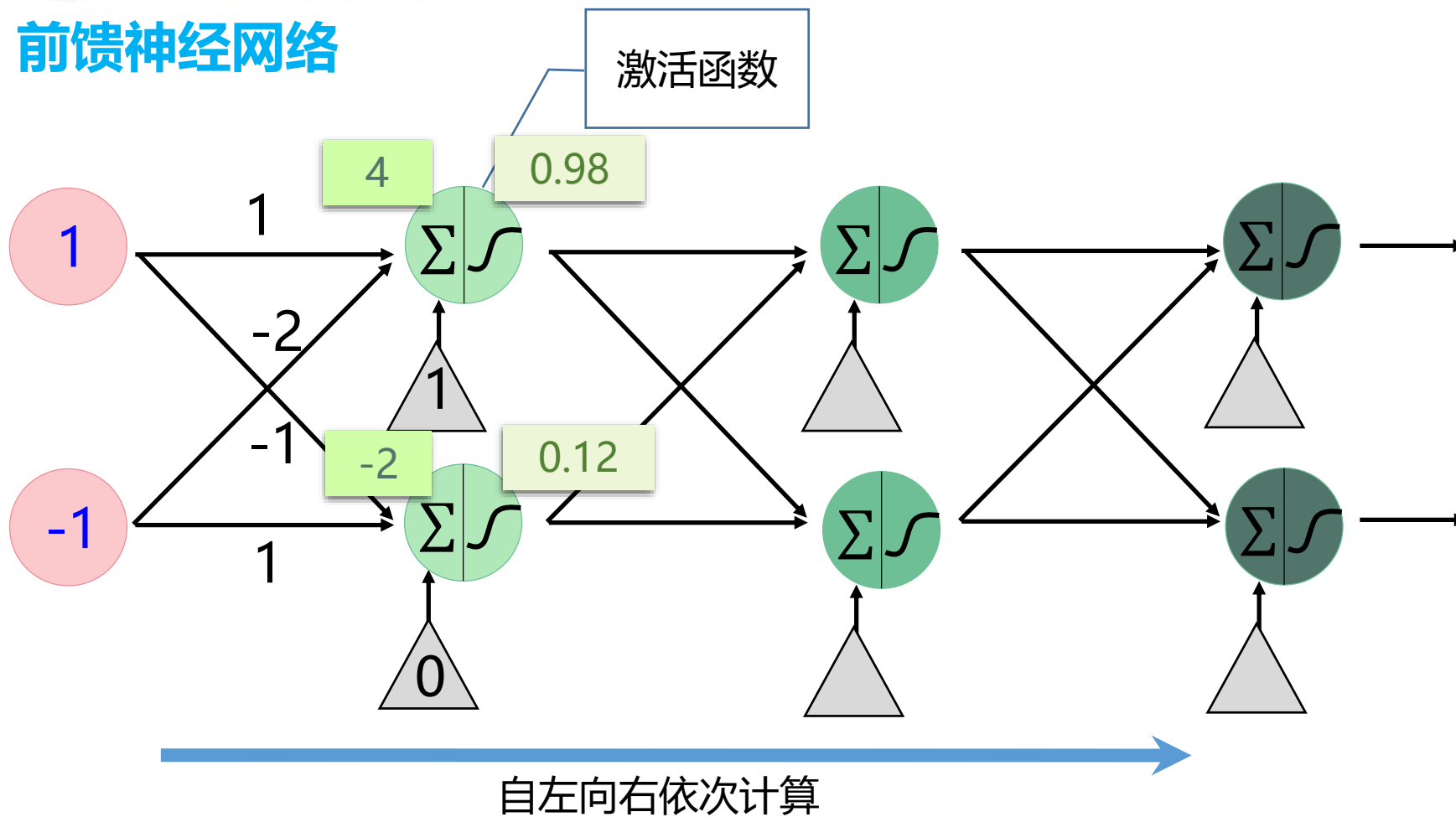
AI DISCOVERY



建立模型

AI DISCOVERY

★ 前馈神经网络



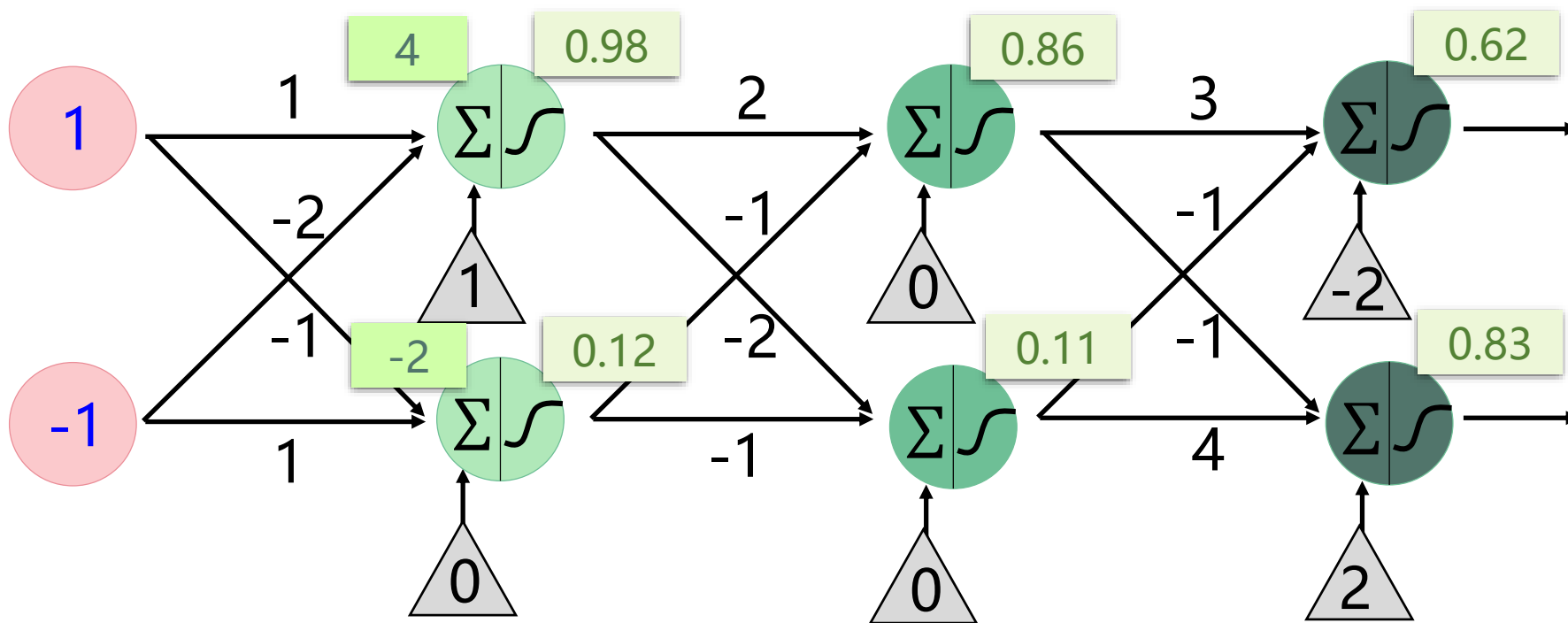
AI DISCOVERY



建立模型

AI DISCOVERY

★ 前馈神经网络



$$\text{函数: } f\left(\begin{bmatrix} 1 \\ -1 \end{bmatrix}\right) = \begin{bmatrix} 0.62 \\ 0.83 \end{bmatrix}$$

AI DISCOVERY



建立模型



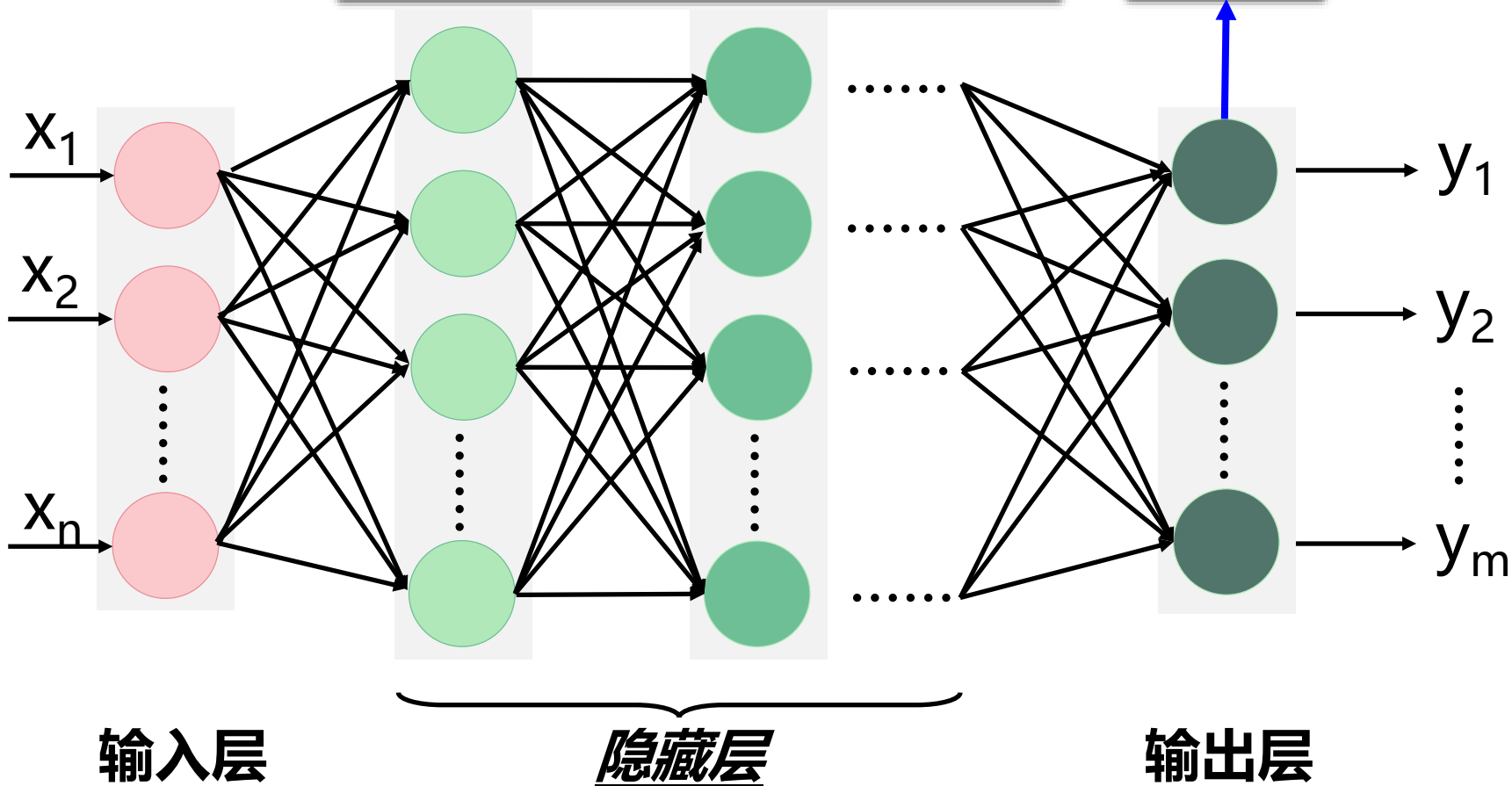
AI DISCOVERY



前馈神经网络

隐藏层数很多意味着网络越深

神经元



AI DISCOVERY

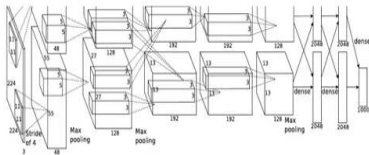


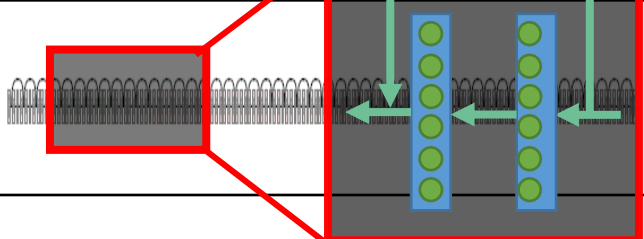


建立模型

AI DISCOVERY



例子

模型	层数	结构	错误率
AlexNet (2012)	8层		16.4%
VGG (2014)	19层		6.7%
GoogleNet (2014)	22层		6.7%
Residual Net (2015)	152层		3.57%

AI DISCOVERY



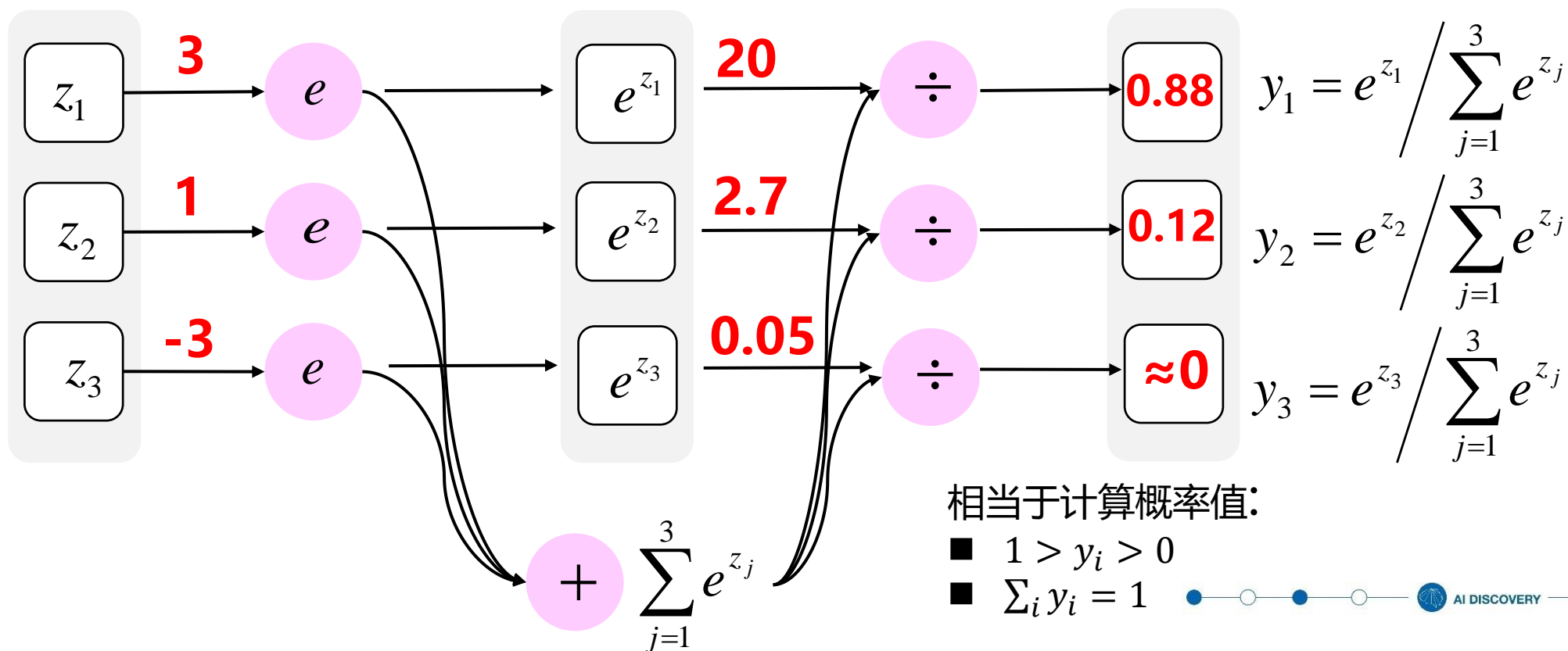
建立模型

AI DISCOVERY



输出层

- 常用softmax函数作为输出层激活函数：容易理解、便于计算



AI DISCOVERY

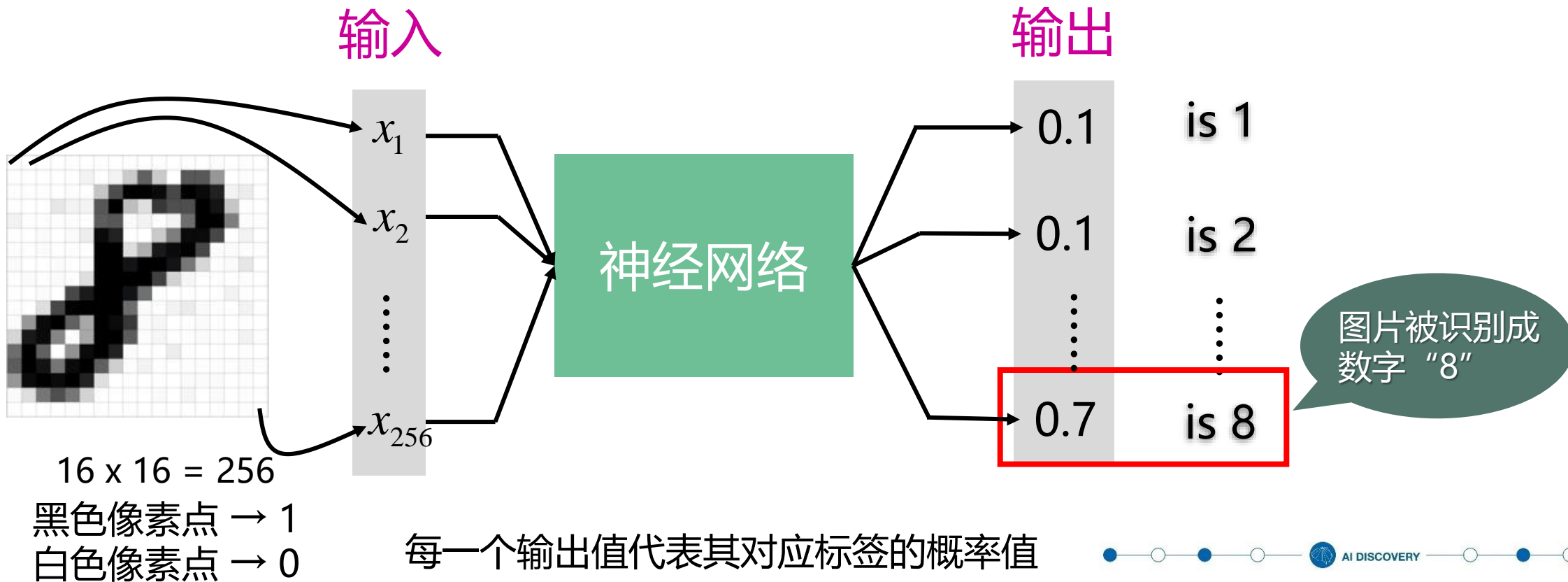


建立模型

AI DISCOVERY



应用示例：手写识别



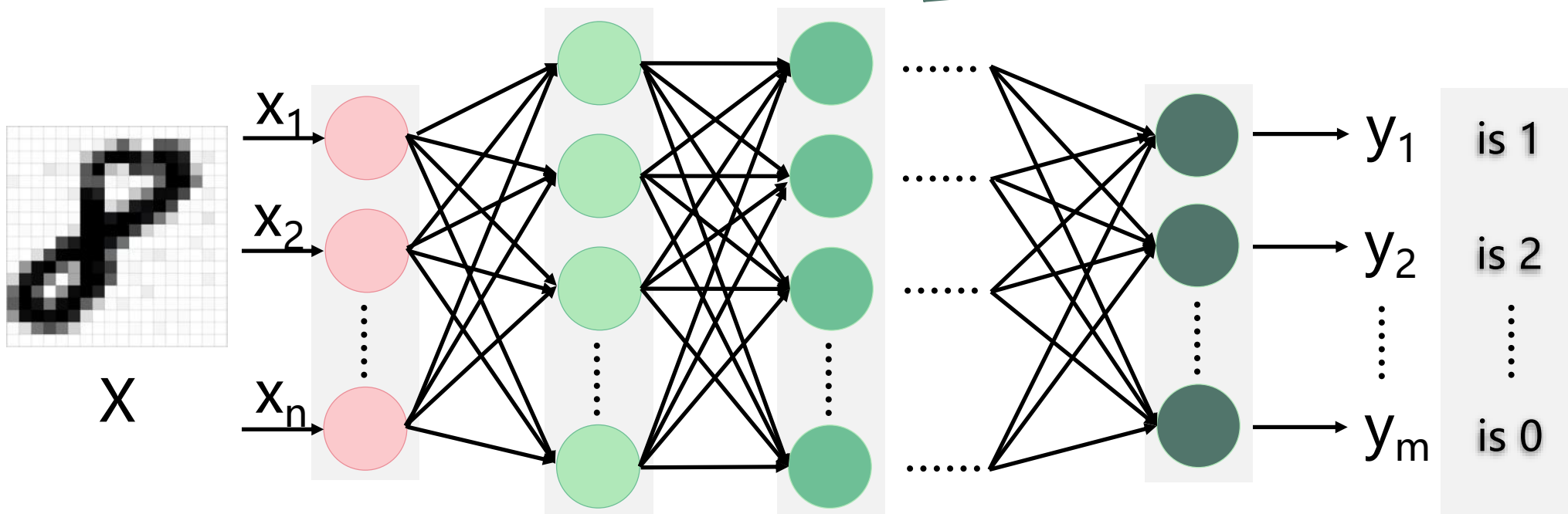
AI DISCOVERY



建立模型

AI DISCOVERY

★ 应用示例：手写体识别



问题：应该设置多少层，多少结点？
是否需要选择其他网络结构如
CNN/RNN？

设置合适的网络结构：层数和结点个数、激活函数



使用步骤

AI DISCOVERY

★ 建立模型

- 选择什么样的网络结构
- 选择多少层数，每层选择多少神经元

★ 损失函数

- 选择常用损失函数，平方误差，交叉熵....

★ 参数学习

- 梯度下降
- 反向传播算法

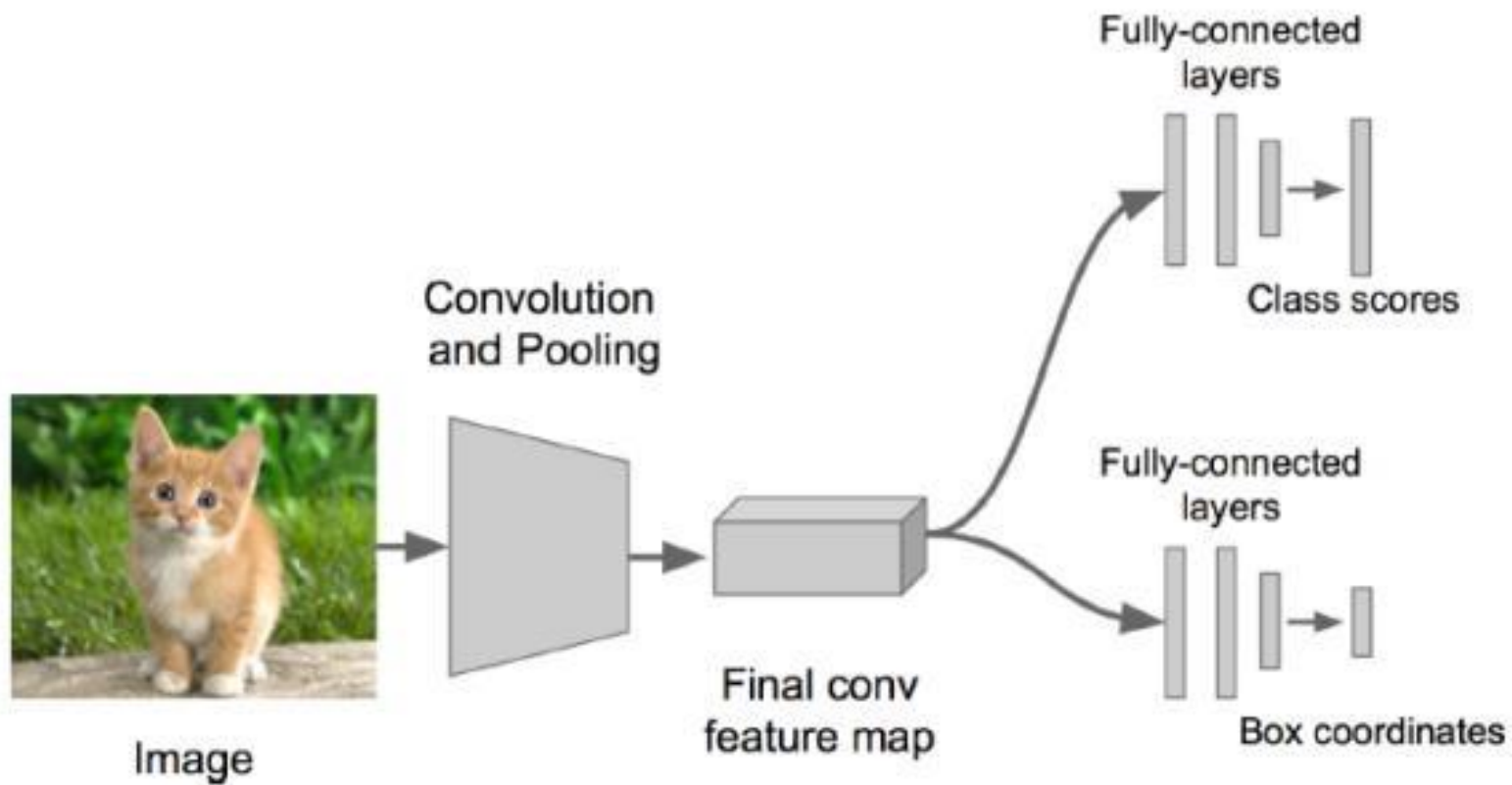
AI DISCOVERY



损失函数



AI DISCOVERY



损失函数的设计依赖于具体的任务



AI DISCOVERY

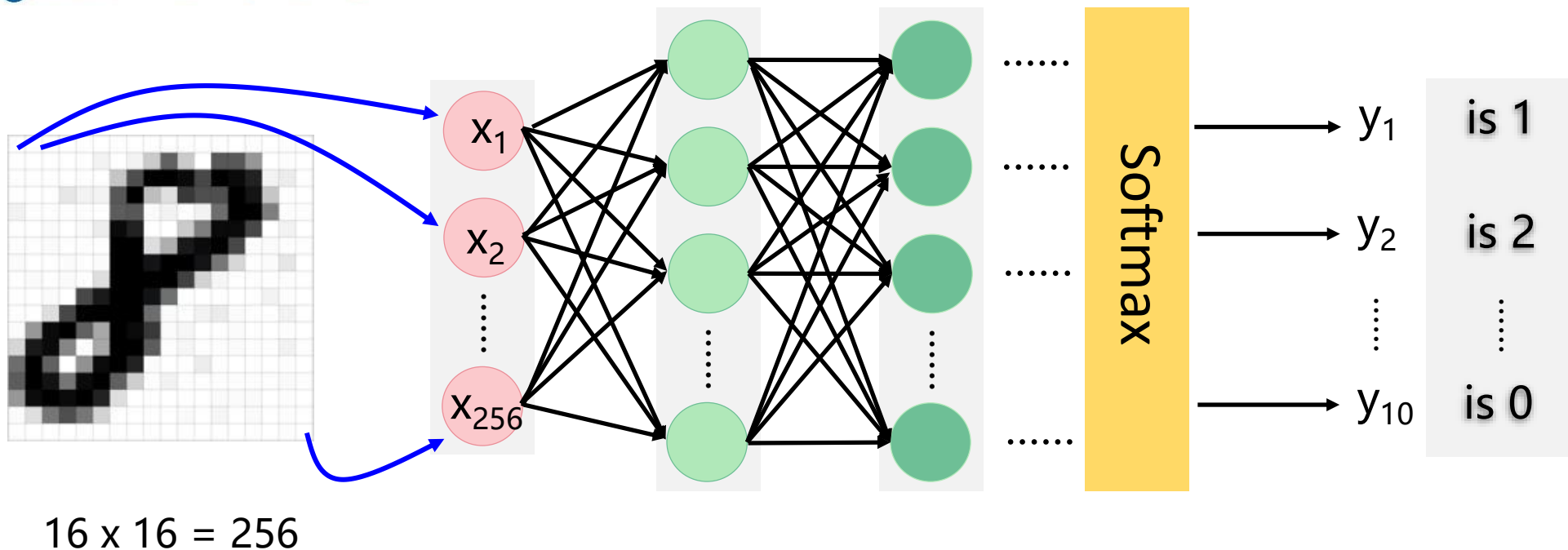




损失函数



AI DISCOVERY



对于数字识别任务，设计用于分类的损失函数，使得学习目标变为：

输入： $\rightarrow y_1$ 值最大；输入： $\rightarrow y_8$ 值最大



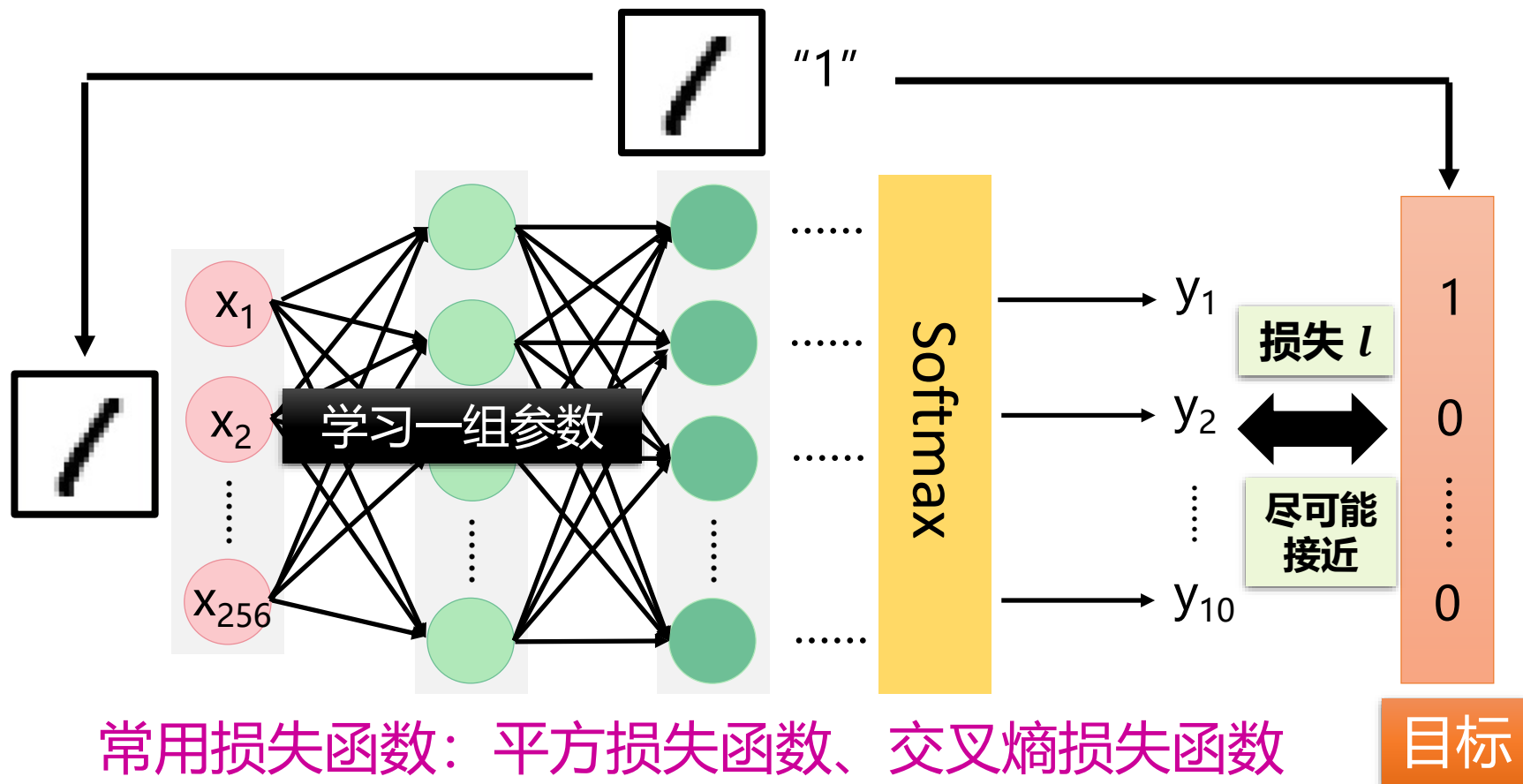
AI DISCOVERY





损失函数

好的参数使得所有训练数据的损失越小越好





损失函数

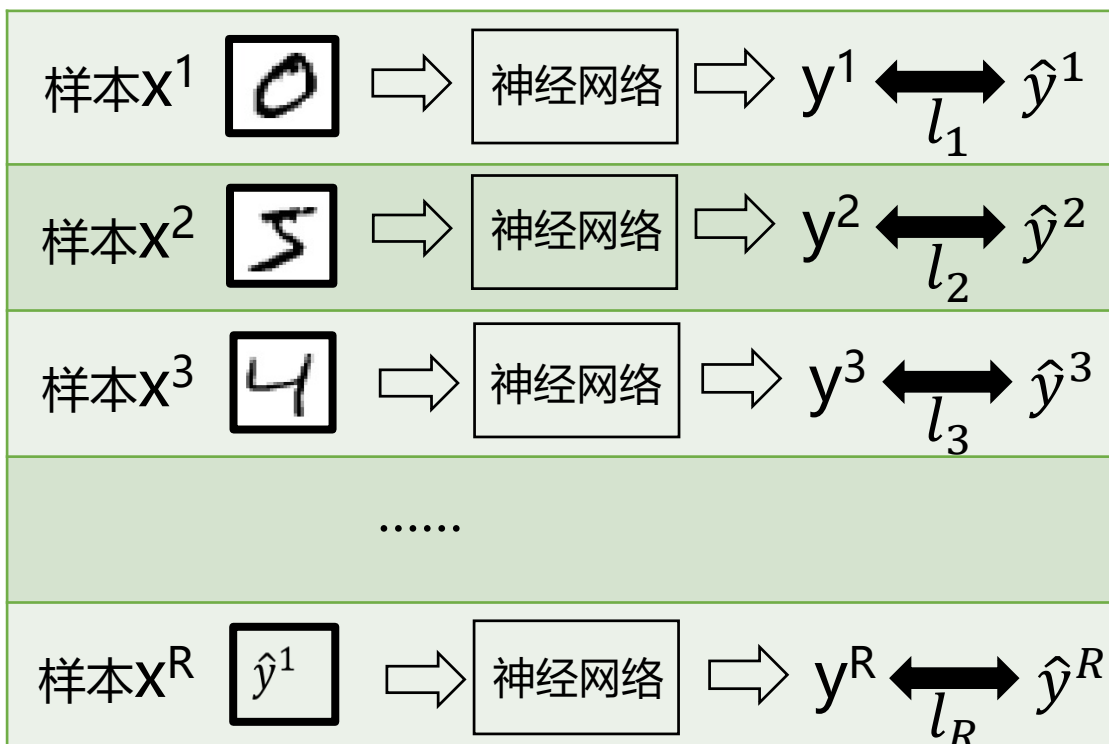


AI DISCOVERY

总损失:

$$L = \sum_{r=1}^R l_r$$

对所有训练数据:



尽可能小

找到一个函数使得总损失 L 最小

即确定参数使得总损失 L 最小



AI DISCOVERY



使用步骤

AI DISCOVERY

★ 建立模型

- 选择什么样的网络结构
- 选择多少层数，每层选择多少神经元

★ 损失函数

- 选择常用损失函数，平方误差，交叉熵....

★ 参数学习

- 梯度下降
- 反向传播算法

AI DISCOVERY



参数学习

➤ 枚举所有可能的取值

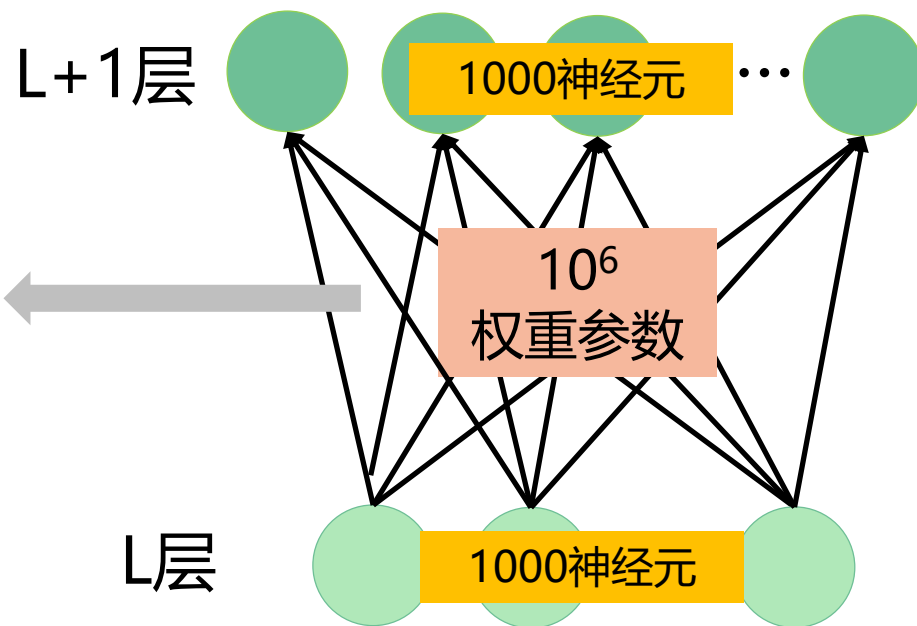


网络参数 $\theta =$

$$\{w_1, w_2, w_3, \dots, b_1, b_2, b_3, \dots\}$$

参数个数巨大

➤ 寻找模型参数 θ^* 使得总损失 L 最小



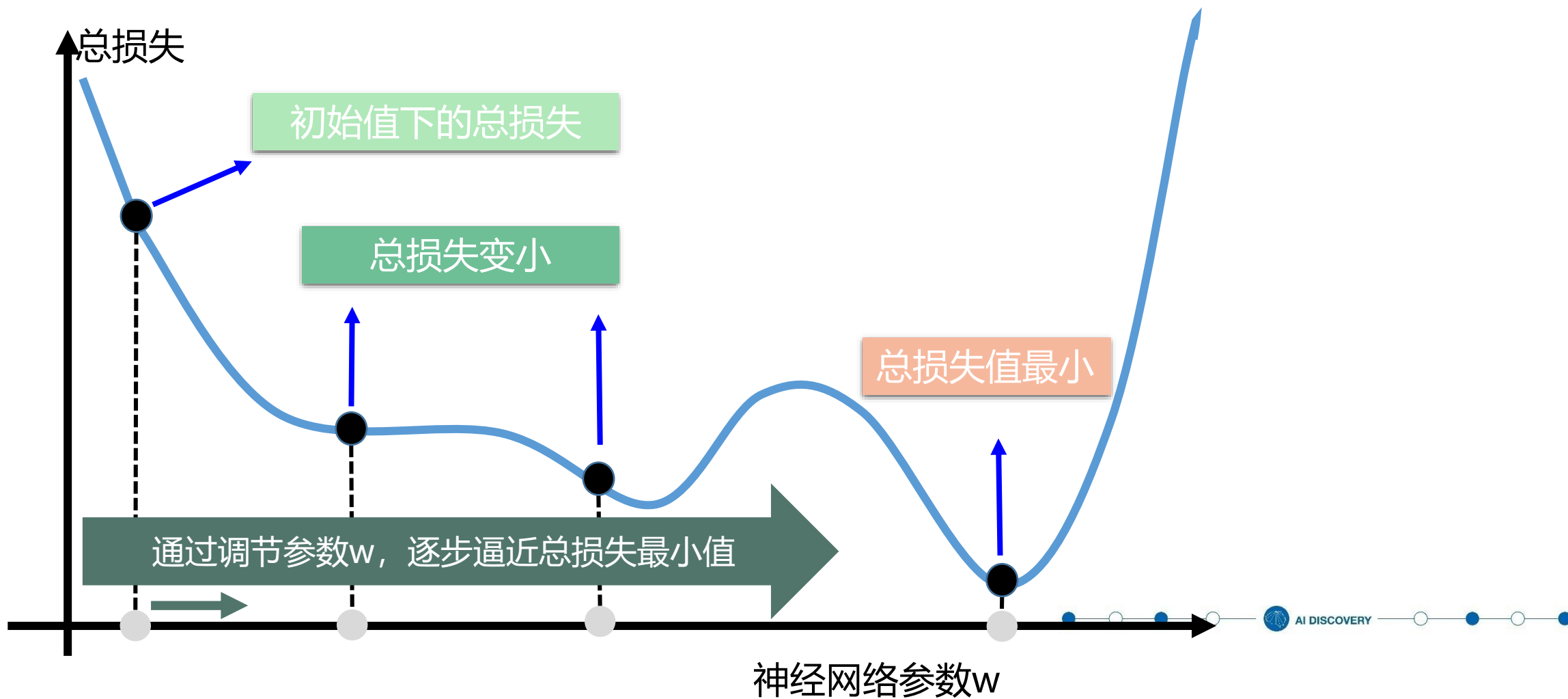
例如：语音识别模型有 8 层，每层 1000 神经元



参数学习

AI DISCOVERY

★ 梯度下降法





参数学习

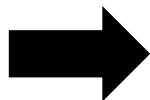
AI DISCOVERY



梯度下降法

- 选择一个初始值 w , Random, RBM pre-train
- 计算梯度值 $\partial L / \partial w$

梯度为负



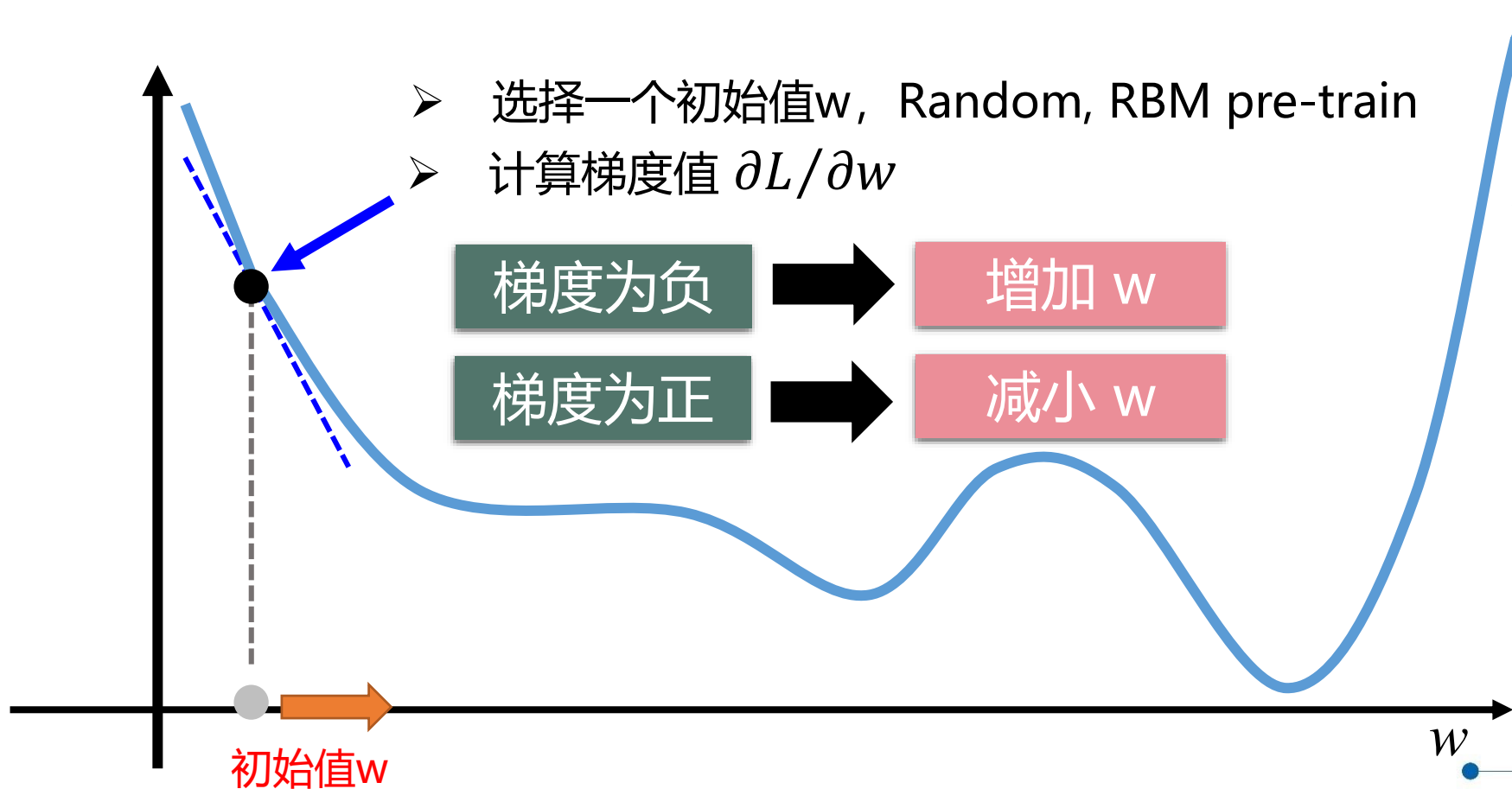
增加 w

梯度为正



减小 w

神经网络参数 $\theta = \{w_1, w_2, \dots, b_1, b_2, \dots\}$



AI DISCOVERY



参数学习

AI DISCOVERY



梯度下降法

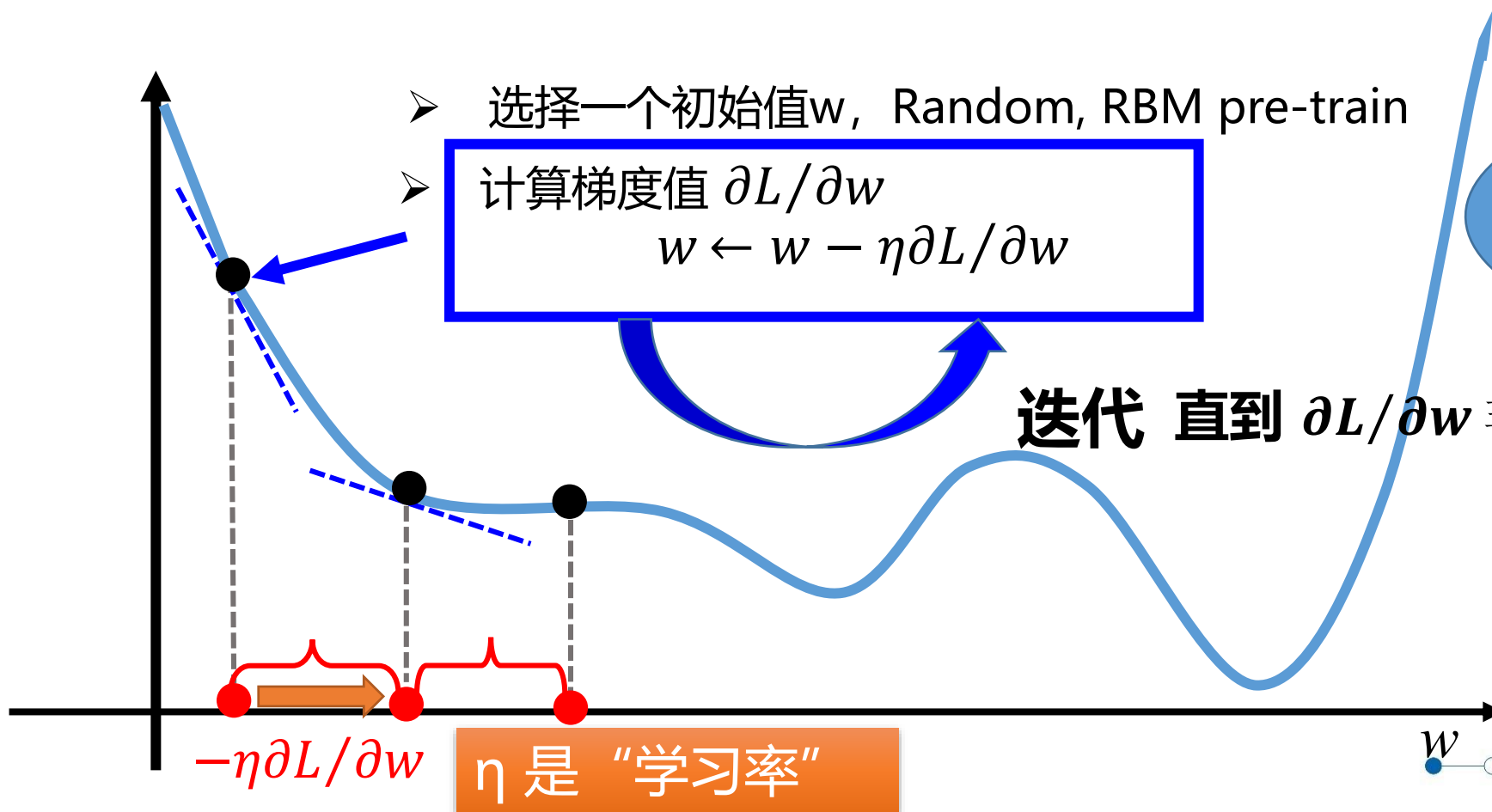
➤ 选择一个初始值 w , Random, RBM pre-train

➤ 计算梯度值 $\partial L / \partial w$

$$w \leftarrow w - \eta \partial L / \partial w$$

神经网络参数 $\theta = \{w_1, w_2, \dots, b_1, b_2, \dots\}$

迭代 直到 $\partial L / \partial w$ 非常小 (更新很小)



AI DISCOVERY

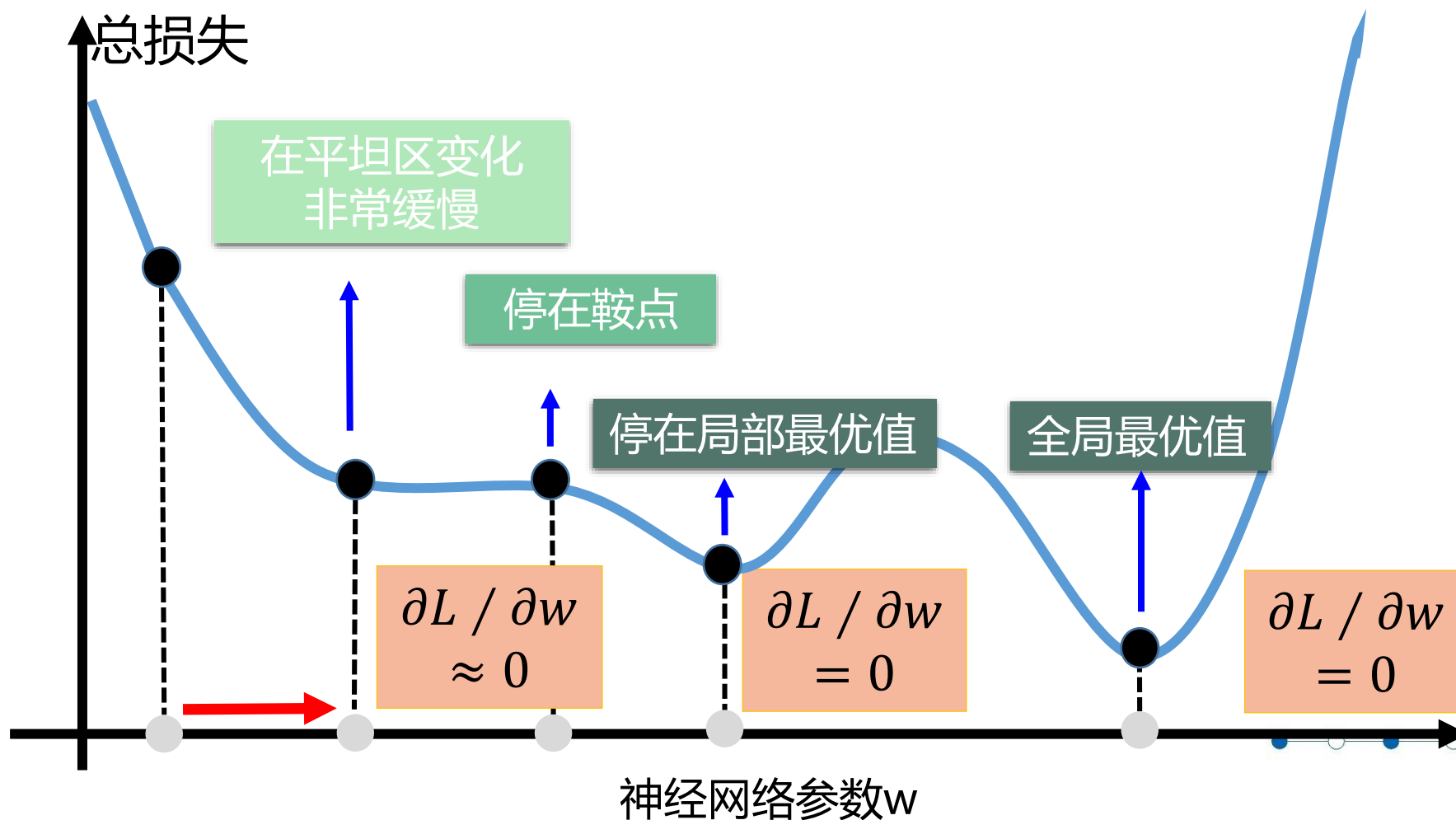


参数学习

AI DISCOVERY



梯度下降法



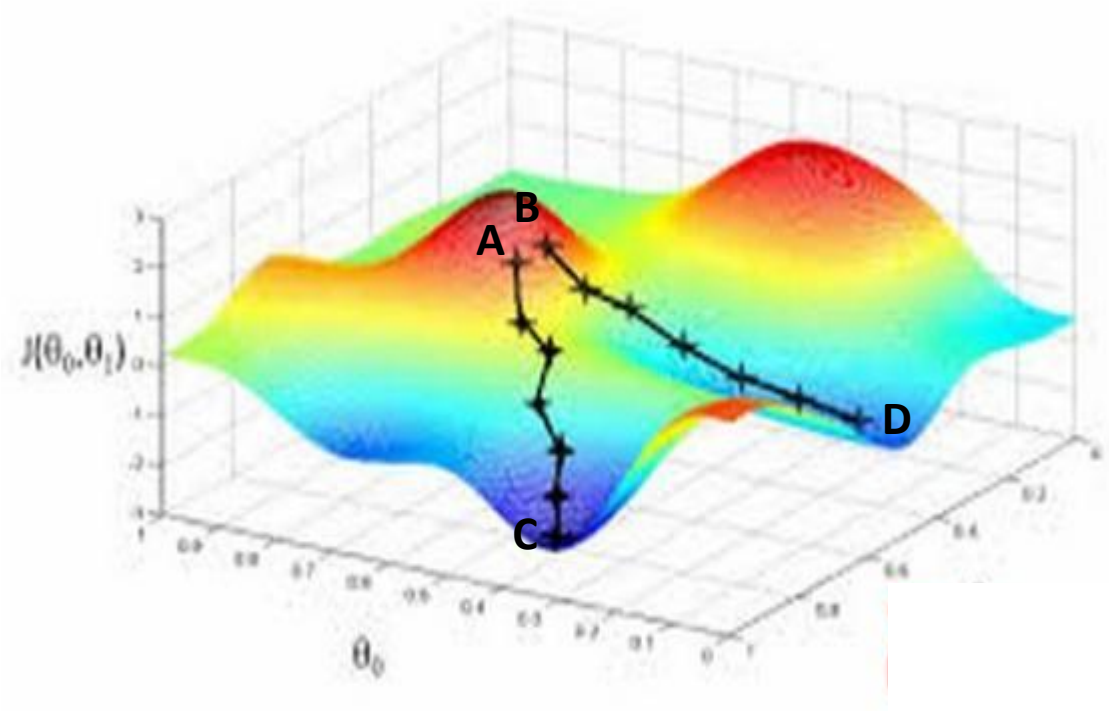
AI DISCOVERY



参数学习

AI DISCOVERY

★ 初始值影响



例如：

A → C

B → D

选取不同的初始值，可能到达不同的局部最小值

AI DISCOVERY



使用步骤

AI DISCOVERY

★ 建立模型

- 选择什么样的网络结构
- 选择多少层数，每层选择多少神经元

★ 损失函数

- 选择常用损失函数，平方误差，交叉熵....

★ 参数学习

- 梯度下降
- 反向传播算法

AI DISCOVERY



参数学习



反向传播算法



只剩一个问题：
怎么求 梯度 $\partial L / \partial w$?

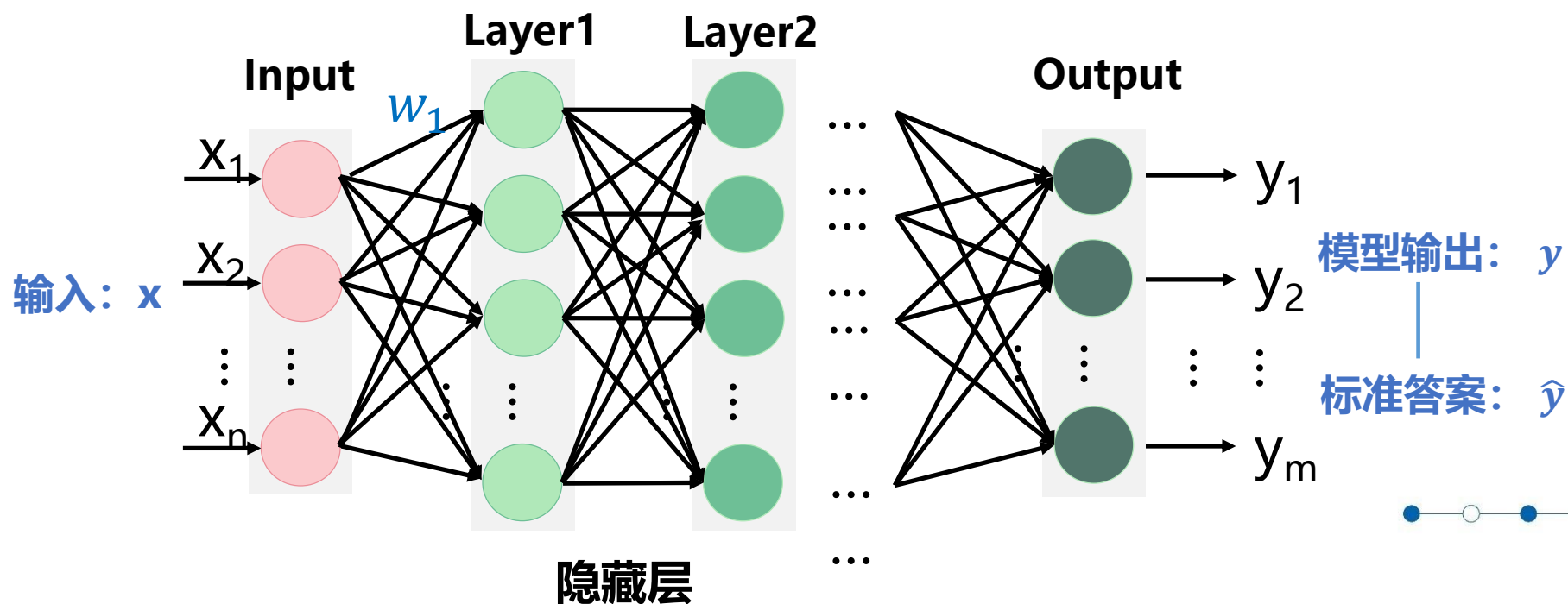
$$w_1 - \eta \partial L / \partial w_1$$

更新 w_1

计算 $\partial L / \partial w_1$

总误差 L

以此类推更新 w_2 w_3





深度学习方法（下）



AI DISCOVERY

深度学习方法使用步骤

正向传播

反向传播



AI DISCOVERY



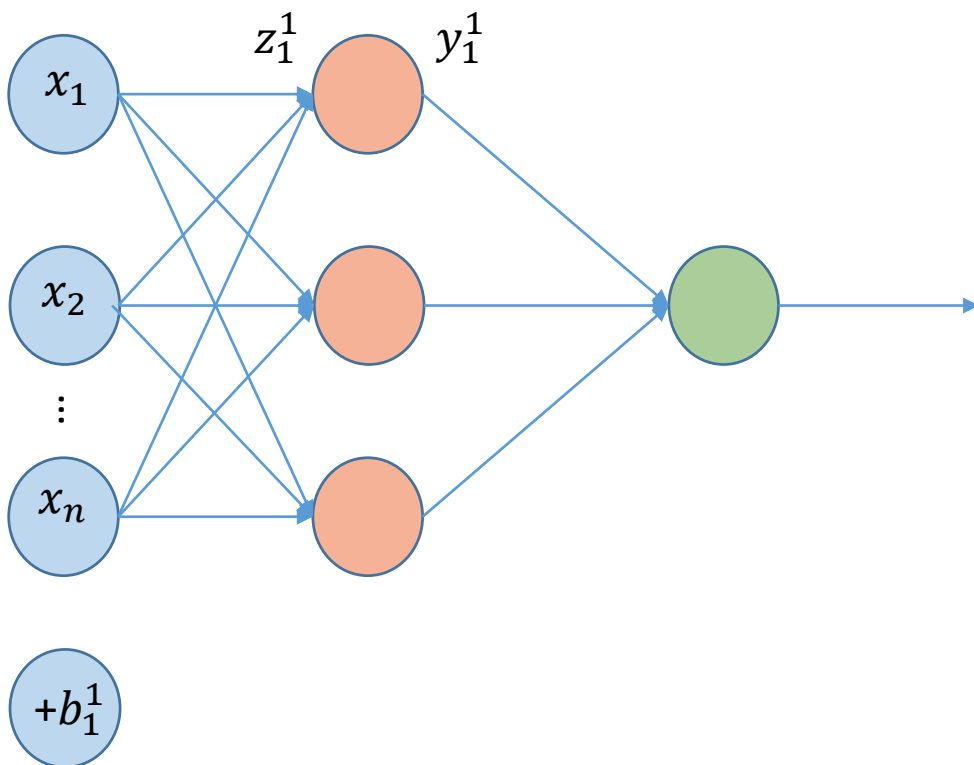
参数学习

AI DISCOVERY



信号正向传播

➤ 假定第 l 层神经元状态为 z^l ，经激活函数后的输出值为 y^l



AI DISCOVERY



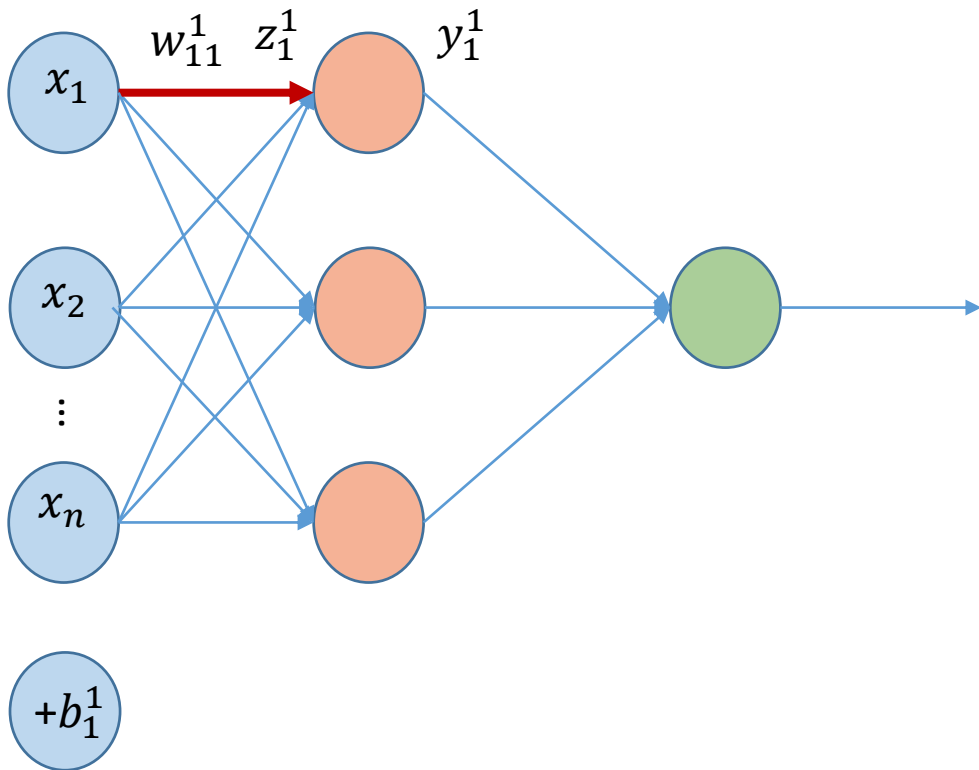
参数学习

AI DISCOVERY



信号正向传播

➤ 假定第 l 层神经元状态为 z^l ，经激活函数后的输出值为 y^l



$$\bullet z_1^1 = w_{11}^1 x_1$$

AI DISCOVERY



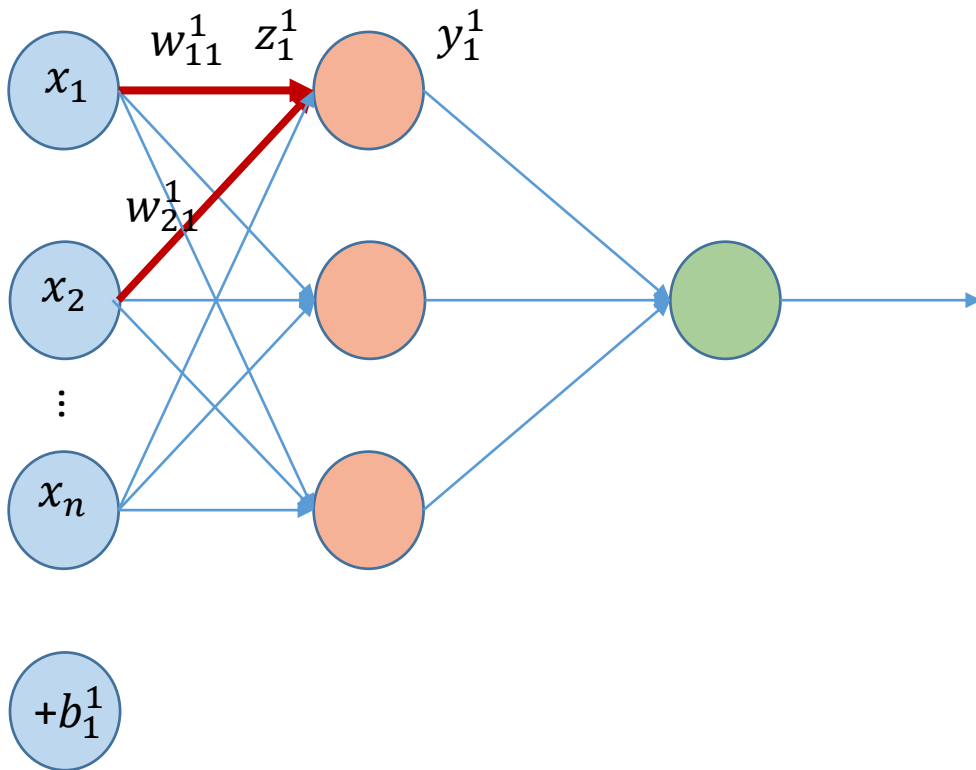
参数学习

AI DISCOVERY



信号正向传播

➤ 假定第 l 层神经元状态为 z^l ，经激活函数后的输出值为 y^l



$$\bullet z_1^1 = w_{11}^1 x_1 + w_{21}^1 x_2$$

AI DISCOVERY



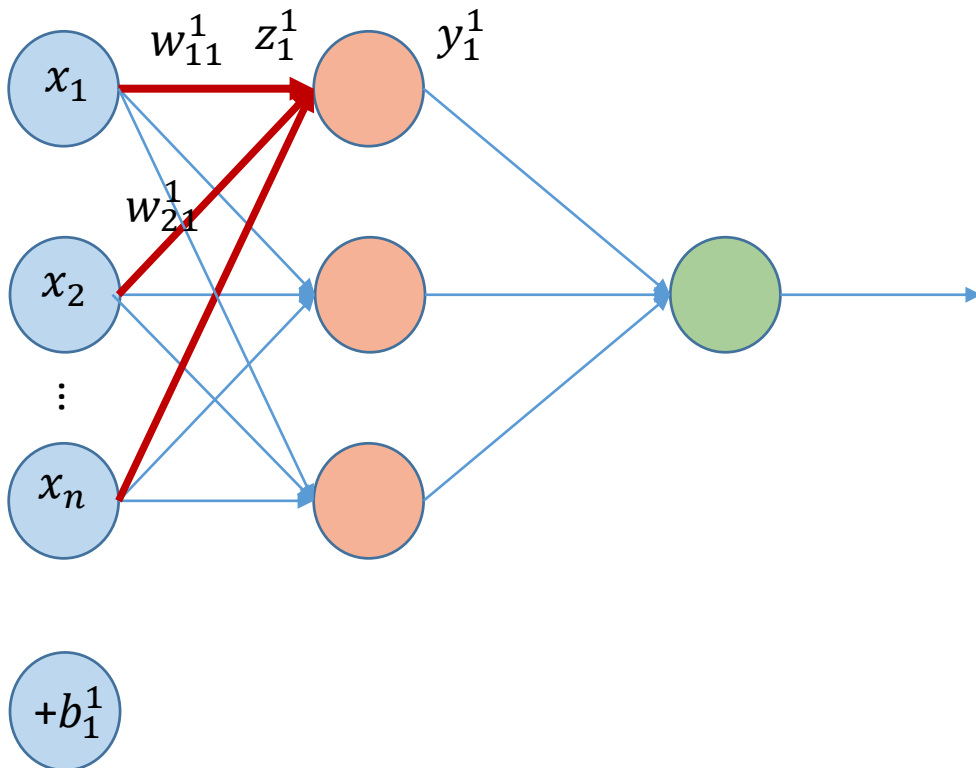
参数学习

AI DISCOVERY



信号正向传播

➤ 假定第 l 层神经元状态为 z^l ，经激活函数后的输出值为 y^l



$$z_1^1 = w_{11}^1 x_1 + w_{21}^1 x_2 + \dots + w_{n1}^1 x_n$$

AI DISCOVERY



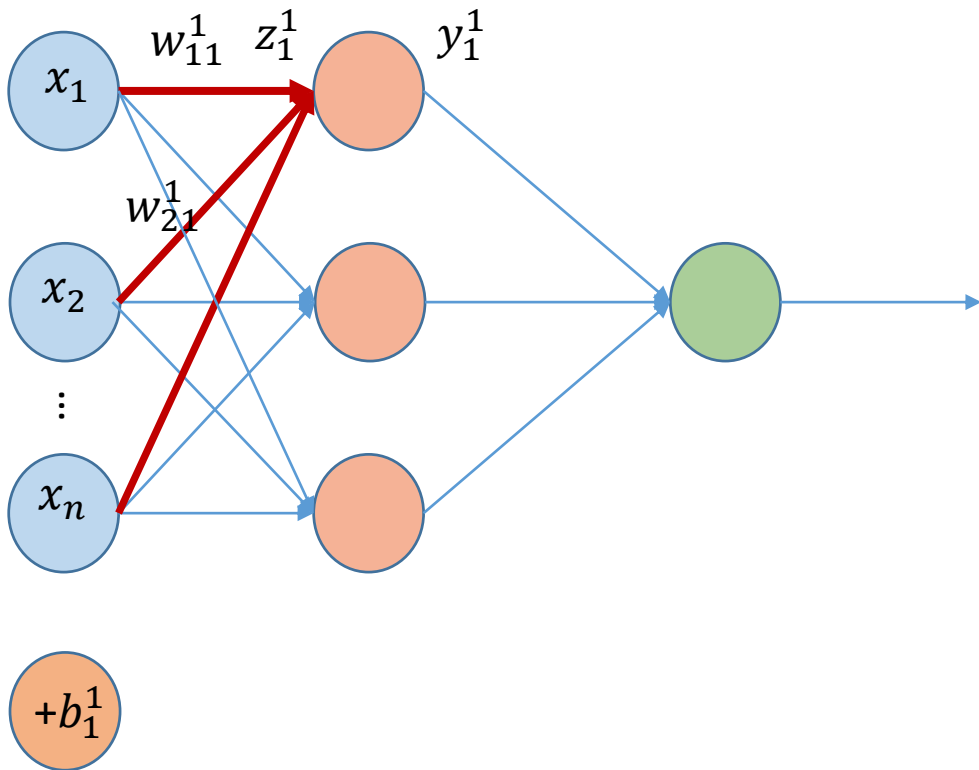
参数学习

AI DISCOVERY



信号正向传播

➤ 假定第 l 层神经元状态为 z^l ，经激活函数后的输出值为 y^l



$$\bullet \quad z_1^1 = w_{11}^1 x_1 + w_{21}^1 x_2 + \dots + w_{n1}^1 x_n + b_1^1$$

AI DISCOVERY



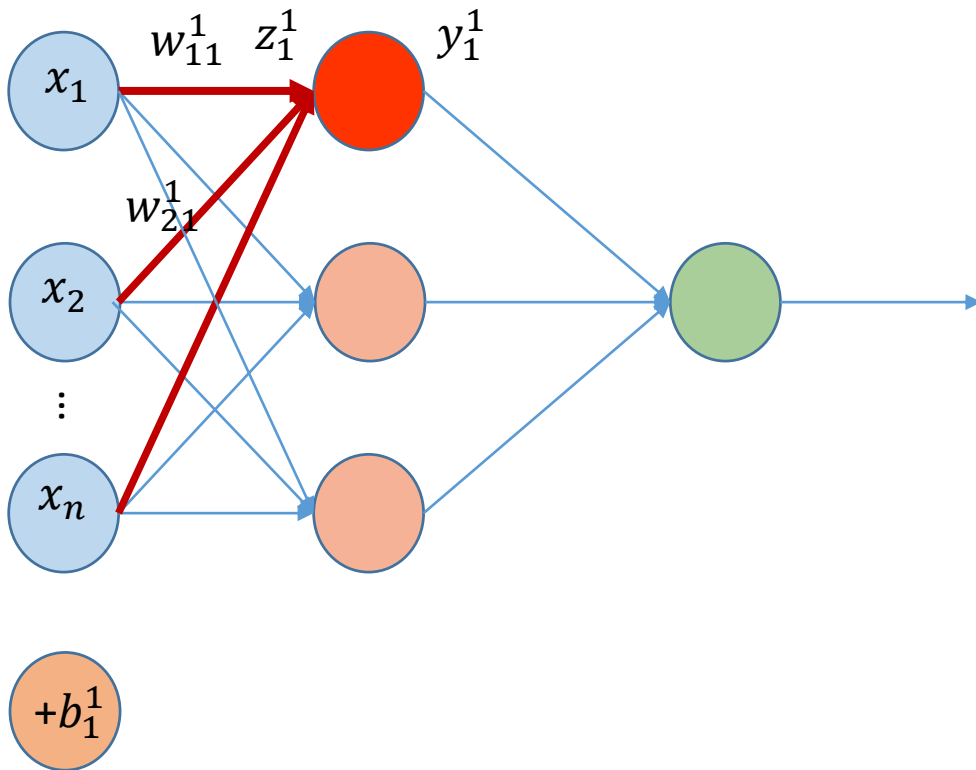
参数学习

AI DISCOVERY



信号正向传播

➤ 假定第 l 层神经元状态为 z^l ，经激活函数后的输出值为 y^l



- $z_1^1 = w_{11}^1 x_1 + w_{21}^1 x_2 + \dots + w_{n1}^1 x_n + b_1^1$
- $y_1^1 = f(z_1^1) = f(w_{11}^1 x_1 + w_{21}^1 x_2 + \dots + w_{n1}^1 x_n + b_1^1)$

AI DISCOVERY



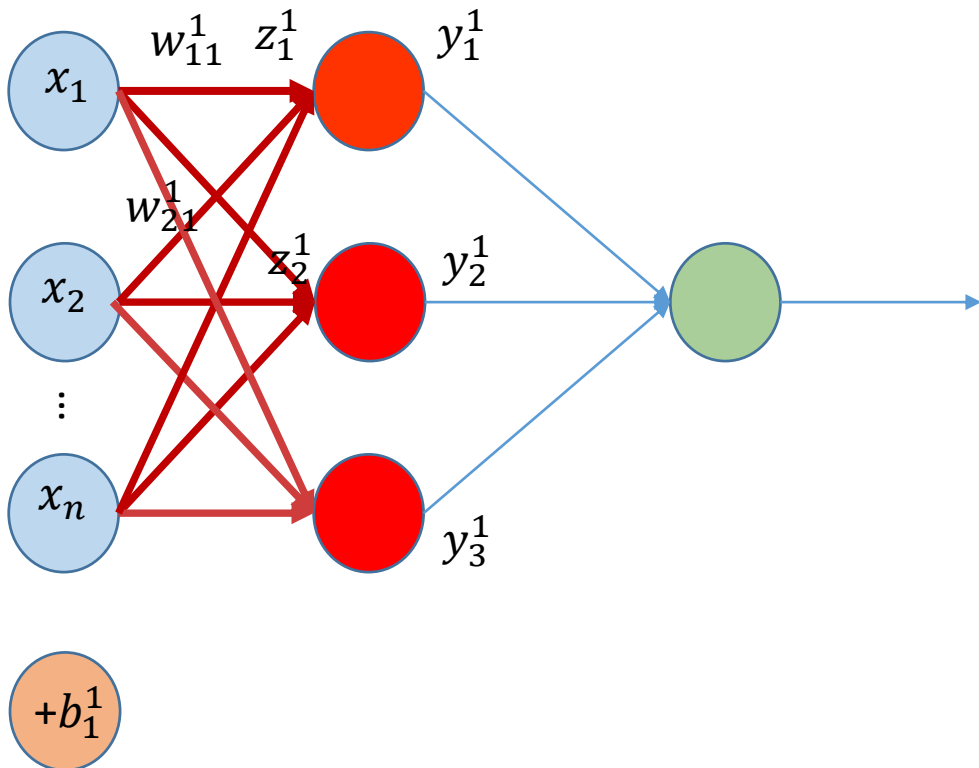
参数学习

AI DISCOVERY



信号正向传播

➤ 假定第 l 层神经元状态为 z^l ，经激活函数后的输出值为 y^l



- $z_1^1 = w_{11}^1 x_1 + w_{21}^1 x_2 + \cdots + w_{n1}^1 x_n + b_1^1$
- $y_1^1 = f(z_1^1) = f(w_{11}^1 x_1 + w_{21}^1 x_2 + \cdots + w_{n1}^1 x_n + b_1^1)$
- $y_2^1 = f(z_2^1) = f(w_{12}^1 x_1 + w_{22}^1 x_2 + \cdots + w_{n2}^1 x_n + b_2^1)$
- $y_3^1 = f(z_3^1) = f(w_{13}^1 x_1 + w_{23}^1 x_2 + \cdots + w_{n3}^1 x_n + b_2^1)$

AI DISCOVERY



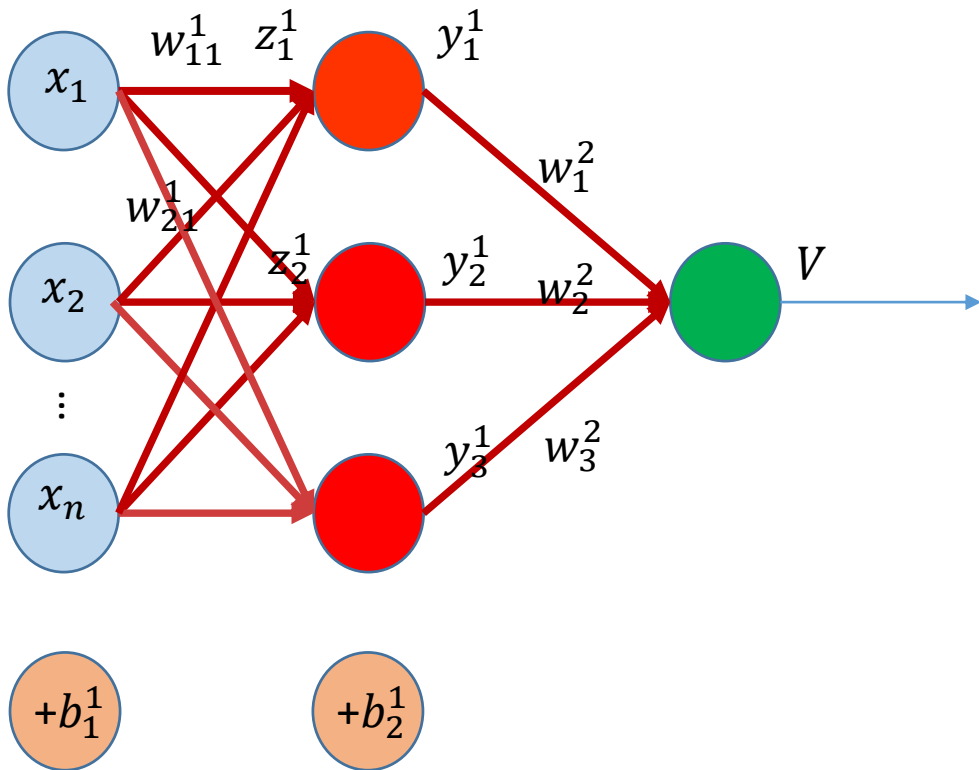
参数学习

AI DISCOVERY



信号正向传播

➤ 假定第 l 层神经元状态为 z^l ，经激活函数后的输出值为 y^l



$$V = f(w_1^2 y_1^1 + w_2^2 y_2^1 + \dots + w_n^2 y_n^1 + b_2)$$

其中：

$$y_1^1 = f(z_1^1) = f(w_{11}^1 x_1 + w_{21}^1 x_2 + \dots + w_{n1}^1 x_n + b_1^1)$$

$$y_2^1 = f(z_2^1) = f(w_{12}^1 x_1 + w_{22}^1 x_2 + \dots + w_{n2}^1 x_n + b_2^1)$$

$$y_3^1 = f(z_3^1) = f(w_{13}^1 x_1 + w_{23}^1 x_2 + \dots + w_{n3}^1 x_n + b_3^1)$$

AI DISCOVERY



深度学习方法（下）



AI DISCOVERY

深度学习方法使用步骤

正向传播

反向传播



AI DISCOVERY



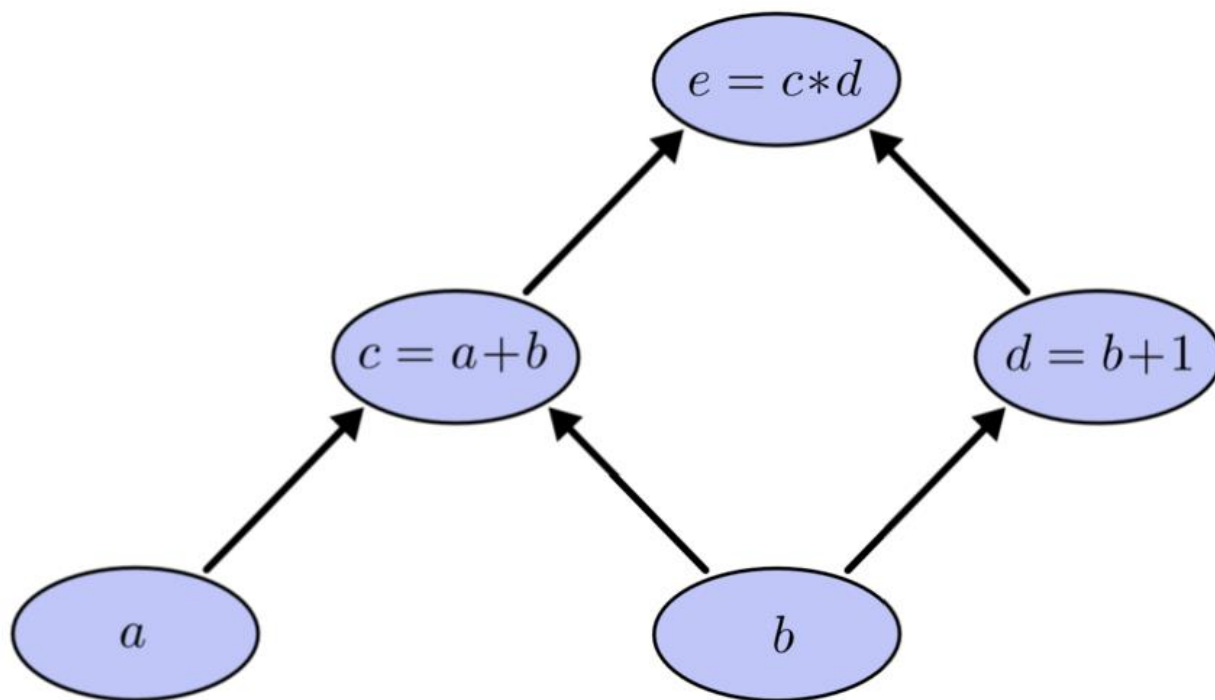
参数学习

AI DISCOVERY

★ 反向传播算法

➤ 算法示例： $e = (a+b)*(b+1)$, 求 $\partial e / \partial a$, $\partial e / \partial b$

引入两个中间变量 c 和 d : $c = a+b$, $d = b+1$, $e = c*d$



AI DISCOVERY

ref: <http://colah.github.io/posts/2015-08-Backprop/>



参数学习

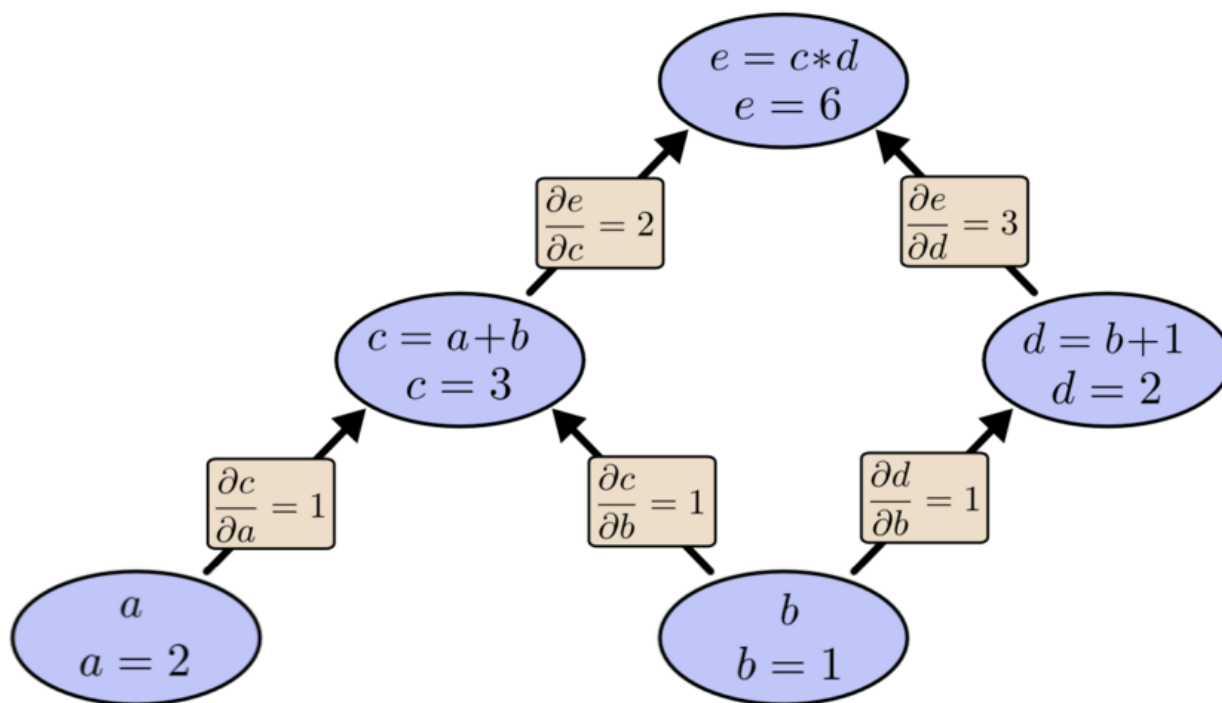
AI DISCOVERY



反向传播算法

➤ 算法示例: $e = (a+b)*(b+1)$, 求 $\partial e / \partial a$, $\partial e / \partial b$

引入两个中间变量 c 和 d : $c = a+b$, $d = b+1$, $e = c*d$



AI DISCOVERY

ref: <http://colah.github.io/posts/2015-08-Backprop/>



参数学习



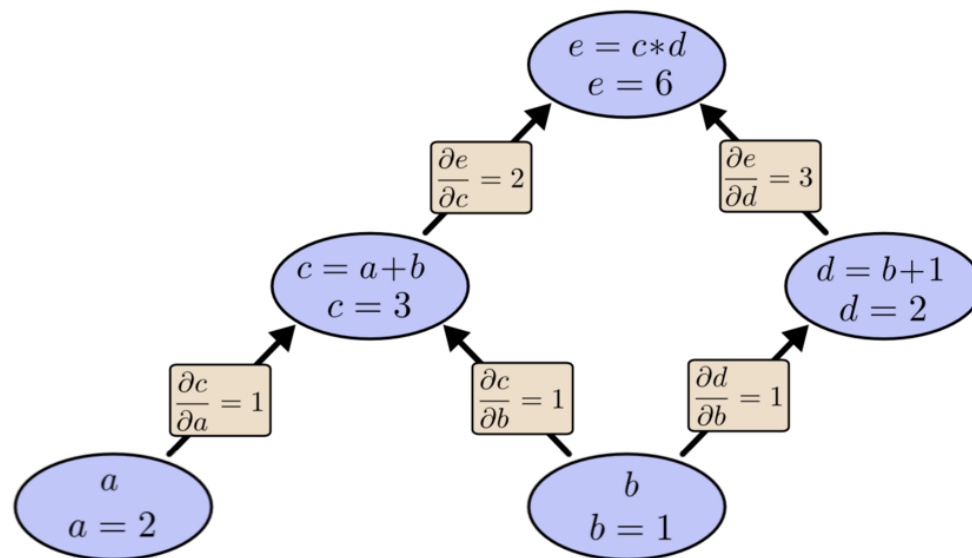
AI DISCOVERY



反向传播算法

◆ 链式法则

- $\frac{\partial e}{\partial a} = \frac{\partial e}{\partial c} \cdot \frac{\partial c}{\partial a}$, 图中 $\frac{\partial e}{\partial a}$ 的值等于从a到e的路径上的偏导值的乘积
- $\frac{\partial e}{\partial b} = \frac{\partial e}{\partial c} \cdot \frac{\partial c}{\partial b} + \frac{\partial e}{\partial d} \cdot \frac{\partial d}{\partial b}$, 上图中 $\frac{\partial e}{\partial b}$ 的值等于从b到e的路径 (b-c-e)上的偏导值的乘积加上路径 (b-d-e)上的偏导值的乘积。
- 若自下而上求解, 很多**路径被重复访问了**。
比如图中, 求 $\frac{\partial e}{\partial a}$ 需要计算路径a-c-e, 求 $\frac{\partial e}{\partial b}$ 都需要计算路径b-c-e和b-d-e, **路径c-e被访问了两次**。



AI DISCOVERY



参数学习



AI DISCOVERY



反向传播算法

◆ 链式法则

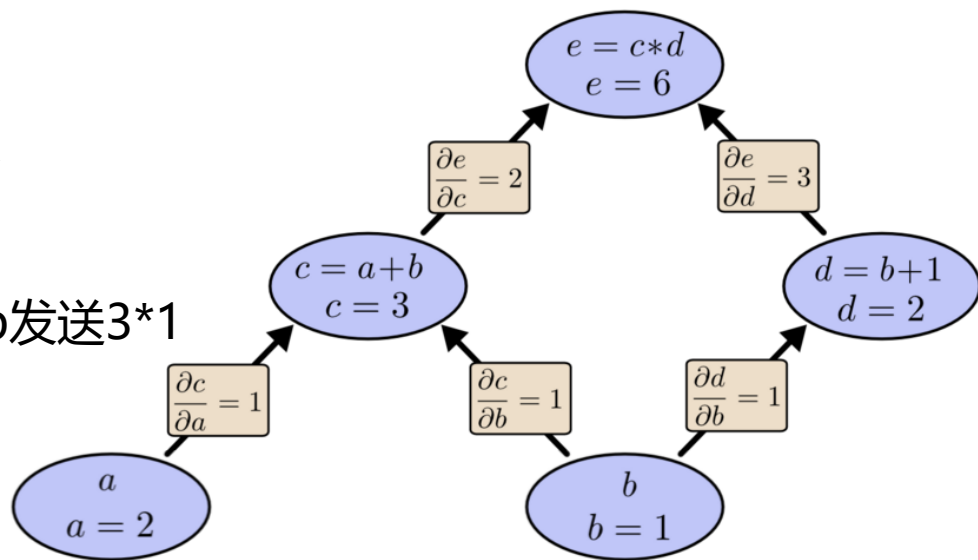
自上而下：从最上层的节点e开始，对于e的下一层的所有子节点，将e的值（e是最顶点，值=1）乘以e到某个节点路径上的偏导值，并将结果发送到该子节点中。该子节点的值被设为“发送过来的值”，继续此过程向下传播

第一层：节点e初始值为1

第二层：节点e向节点c发送 $1*2$ ，节点e向节点d发送 $1*3$ ，
节点c值为2. 节点d值为3.

第三层：节点c向a发送 $2*1$ ，节点c向b发送 $2*1$ ，节点d向b发送 $3*1$
节点a值为2. 节点b值为为 $2*1+3*1=5$.

即顶点e对a的偏导数为2 顶点e对b的偏导数为5





目录



AI DISCOVERY

更多AI知识，请关注



1

深度学习方法（上）

深度学习发展历程、为什么使用深度学习

2

深度学习方法（下）

深度学习方法使用步骤、正向传播、反向传播

3

PaddlePaddle入门

概述、开发环境、使用入门

4

课程实践

实践：房价预测、手写数字识别



AI DISCOVERY



PaddlePaddle入门



AI DISCOVERY

概述

开发环境

使用入门



AI DISCOVERY



深度学习工具



AI DISCOVERY

深度学习在很多机器学习领域都有非常出色的表现，在图像识别、语音识别、自然语言处理、机器人、网络广告投放、医学自动诊断和金融等各大领域均有广泛应用。

深度学习工具的优势：

◆易用性

屏蔽底层，用户只需关注模型结构。
同时，深度学习工具简化了计算，降低了深度学习入门门槛。

◆高效性

灵活的移植性，可将代码部署到CPU或GPU上。选择具有分布式性能的深度学习工具会使模型训练更高效。

选择一个合适的深度学习工具可以事半功倍。



AI DISCOVERY





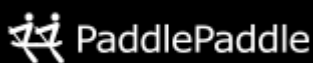
主流的深度学习开源工具

AI DISCOVERY

PaddlePaddle是百度提供的国内首个开源深度学习框架，是基于“深度学习编程语言”的新一代深度学习框架，在兼具性能的同时，极大的提升了框架对模型的表达能力。框架本身具有易学、易用、安全、高效四大特性，是最适合中国开发者和企业的深度学习工具。

GitHub项目地址：

<https://github.com/PaddlePaddle/Paddle>



Caffe是由神经网络中的表达式、速度及模块化产生的深度学习框架。Caffe是一个基于C++/CUDA架构框架，开发者能够利用它自由的组织网络，目前支持卷积神经网络和全连接神经网络（人工神经网络）。在Linux上，C++可以通过命令行来操作接口，运算上支持CPU和GPU直接无缝切换。

但官方文档和教程支持不是很充分，对初学者来说门槛较高。

GitHub项目地址：

<https://github.com/BVLC/caffe>

Caffe

TensorFlow是谷歌基于C++开发、发布的第二代机器学习系统。开发目的是用于进行机器学习和深度神经网络的研究。目前Google的Google App的语音识别、Gmail的自动回复功能、Google Photos的图片搜索等都在使用TensorFlow。

但内部概念众多，结构复杂，API繁重上手困难，版本迭代快API兼容性问题。

GitHub项目地址：

<https://github.com/tensorflow/tensorflow>



Keras是基于Python开发的极其精简并高度模块化的神经网络库，在TensorFlow或Theano上都能够运行，是一个高度模块化的神经网络库，支持GPU和CPU运算。Keras侧重于开发快速实验，用可能最少延迟实现从理念到结果的转变，即为做好一项研究的关键。

GitHub项目地址：

<https://github.com/fchollet/keras>





PaddlePaddle深度学习平台



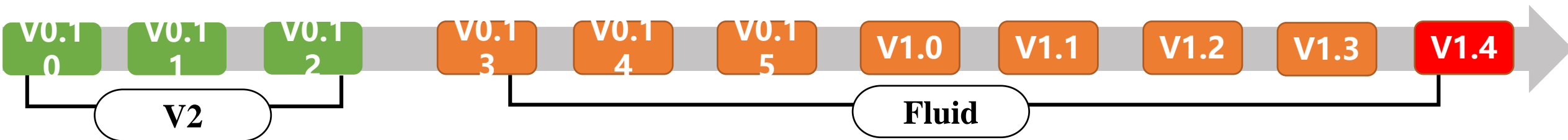
PaddlePaddle (飞桨) -源于产业实践的开源深度学习平台，是集深度学习核心框架、工具组件和服务平台为一体的技术先进、功能完备的开源深度学习平台，已被中国企业广泛使用，深度契合企业实际应用需求，并拥有活跃的开发社区生态。提供丰富的官方支持模型集合，并推出全类型的高性能部署和集成方案供开发者使用。。

网址: <http://www.paddlepaddle.org>

PaddlePaddle前身是2013年百度自主研发的深度学习平台，一直为百度内部工程师使用。

2016年9月1日百度世界大会上，百度首席科学家Andrew Ng（吴恩达）首次宣布将百度深度学习平台对外开放，命名PaddlePaddle。

截止2017年11月 PaddlePaddle 同其他框架比，前13个月的活跃度是最高。



PaddlePaddle一直在持续更新版本!





PaddlePaddle全景



AI DISCOVERY

PaddlePaddle不仅包含深度学习框架，还提供了一整套紧密关联、灵活组合的完整工具组件和服务平台，有利于深度学习技术的应用落地。





PaddlePaddle核心框架-开发



AI DISCOVERY

PaddlePaddle已开源60+ 经过真实业务场景验证的官方模型涵盖视觉、NLP、推荐等 AI核心技术领域，成为官方**支持模型最多**的深度学习平台。从而可以快速构建强大的应用，满足各种场景需求。

PaddleCV

◆ 图像分类

- ✓ 根据图像的语义信息对不同类别图像进行区分。
- ✓ 常用模型：AlexNet、VGG、GoogleNet、ResNet等，方便用户使用。

◆ 视频识别

- ✓ 为开发者提供解决视频理解、视频编辑、视频生成等一系列任务。
- ✓ **业界首个视频识别工具集**。已开放7个视频分类经典模型，覆盖视频识别方向的主流领先模型。
包括：TSN、Non-Local、st-Net、TSM、Attention LSTM、Attention Cluster、NeXtVLAD。



AI DISCOVERY



PaddlePaddle核心框架-开发



AI DISCOVERY

PaddleNLP-面向工业应用的中文NLP开源工具集

基于PaddlePaddle深度学习框架打造的工业级NLP开源工具集，**最懂中文、完善易用、灵活插拔。**

◆全面丰富的能力

提供全面丰富的中文处理任务，涵盖了**文本分类、序列标注、语义表示、语义匹配**等多种NLP任务，可根据业务需求或实验需求快速选择合适的预训练模型进行使用。

◆灵活插拔的网络

基于PaddlePaddle深度学习框架构建的基础NLP网络和NLP应用任务的**灵活解耦**，网络可灵活调整，场景可高效迁移，真正高效易用。

◆工业级应用效果

拥有当前业内效果最好的**中文语义表示模型**和基于用户大数据训练的应用任务模型，模型效果调整机制**源于产业实践**，效果更突出。



AI DISCOVERY



PaddlePaddle核心框架-开发

AI DISCOVERY

 强大的百度NLP 工具集



AI DISCOVERY

 点击我播放



PaddlePaddle核心框架-开发



AI DISCOVERY

PaddleRec

◆智能推荐在当前的互联网服务中正在发挥越来越大的作用，目前大部分**电子商务系统、社交网络、广告推荐、搜索引擎**，都不同程度的使用了各种形式的个性化推荐技术，帮助用户快速找到他们想要的信息。

◆PaddleRec提供多种推荐场景下的召回、排序经典算法，如**GRU4Rec、DeepCTR、Multiview-Simnet**等模型。

除此之外，PaddlePaddle还为开发者们提供了**语音识别、强化学习**等方面的一些模型库，可以满足各种场景需求。



AI DISCOVERY





PaddlePaddle核心框架-训练



AI DISCOVERY



大规模数据分布式训练

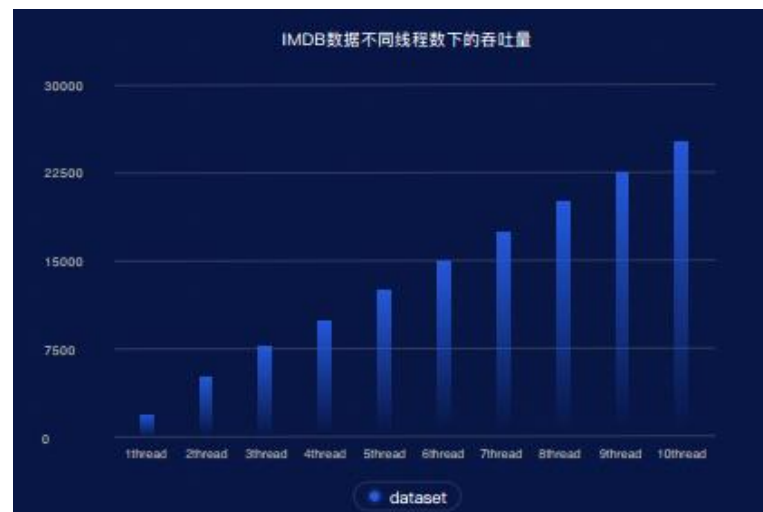
- ◆ 首先多机多卡的全面高效支持，提升了速度；
- ◆ 基于百度海量规模的业务场景实践，同时支持稠密参数和稀疏参数场景的超大规模深度学习并行训练，支持千亿规模参数、数百个节点的高效并行训练；



工业级数据处理

- ◆ 分布式文件系统IO支持
- ◆ 高性能多生产者-多消费者设计

更快的训练速度！





PaddlePaddle核心框架-预测



AI DISCOVERY



端到端的全流程部署方案

◆高速的推理引擎

- ✓ 基于多硬件的支持，PaddlePaddle提供性能全面领先的底层加速库和推理引擎。
- ✓ 全新发布的Paddle Serving支持服务器端的快速部署。

◆模型压缩库

- ✓ 为了部署在更多的硬件上往往需要做模型压缩，模型体积压缩库PaddleSlim针对体积已经很小的MobileNet模型，仍能在模型效果不损失的前提下实现70%以上的体积压缩。

更快的预测速度！



AI DISCOVERY





PaddlePaddle全景



AI DISCOVERY

PaddlePaddle不仅包含深度学习框架，还提供了一整套紧密关联、灵活组合的完整工具组件和服务平台，有利于深度学习技术的应用落地。





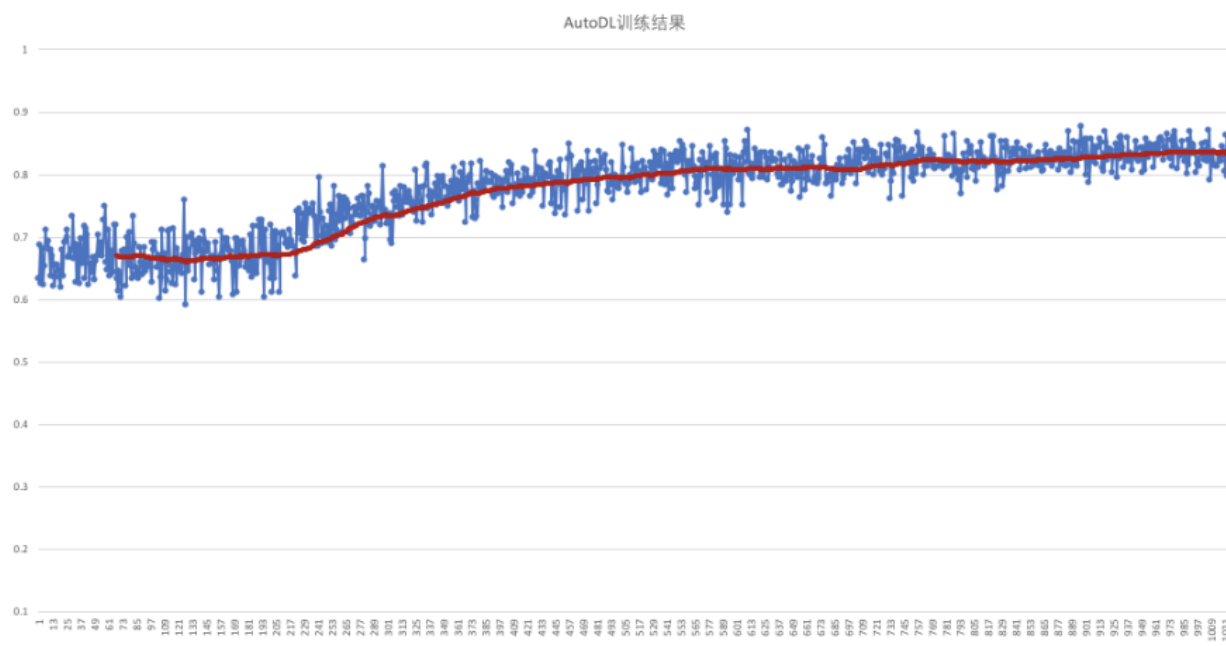
丰富的工具组件

◆ AutoDL Design

开源的**网络结构自动化设计**技术。AutoDL是一种高效的自动搜索构建最佳网络结构的方法，通过增强学习在不断训练过程中得到定制化高质量的模型。

使用文档： <https://github.com/PaddlePaddle/AutoDL/tree/master/AutoDL%20Design>

AutoDL Design设计的图像分类网络在CIFAR10数据集上准确率
达到98%，效果全面超过人类专家，
居于业内领先地位。



通过不断的迭代搜索，使得自动构建模型的效果在不断增强。



丰富的工具组件



简单易用的**预训练模型管理框架**，通过命令行接口，便捷获取PaddlePaddle生态下的预训练模型，无需编写代码，一键使用预训练模型进行预测。

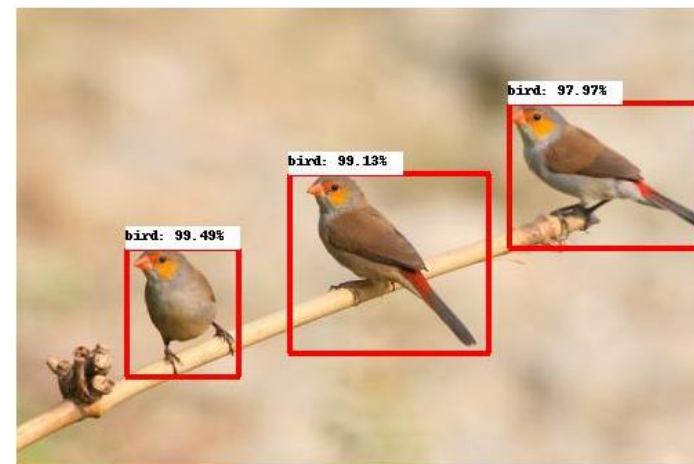
使用文档：<https://github.com/PaddlePaddle/PaddleHub>

安装成功后，执行下面的命令，可以快速体验PaddleHub无需代码、一键预测的命令行功能：

```
# 使用百度LAC词法分析工具进行分词
$ hub run lac --input_text "今天是个好日子"

# 使用百度Senta情感分析模型对句子进行预测
$ hub run senta_bilstm --input_text "今天是个好日子"

# 使用SSD检测模型对图片进行目标检测，检测结果如下图所示
$ wget --no-check-certificate https://paddlehub.bj.bcebos.com/resources/test_img_bird.jpg
$ hub run ssd_mobilenet_v1_pascal --input_path test_img_bird.jpg
```



想了解更多PaddleHub已经发布的模型，请使用hub search命令查看所有已发布的模型。





丰富的工具组件



AI DISCOVERY

◆ **PARL: 一个高性能、灵活的强化学习框架。**

具有灵活性和可扩展性，支持可定制的并行扩展，覆盖DQN、DDPG、PPO、A2C等主流**强化学习**算法。

使用文档: <https://github.com/paddlepaddle/parl>

- ✓ **可复现性保证。** 提供了高质量的主流强化学习算法实现，严格地复现了论文对应的指标。
- ✓ **大规模并行支持。** 最高可支持上万个CPU的同时并发计算，并且支持多GPU强化学习模型的训练。
- ✓ **可复用性强。** 用户无需自己重新实现算法，通过复用框架提供的算法可以轻松地把经典强化学习算法应用到具体的场景中。
- ✓ **良好扩展性。** 当用户想调研新的算法时，可以通过继承PARL提供的基类快速实现自己的强化学习算法。



AI DISCOVERY





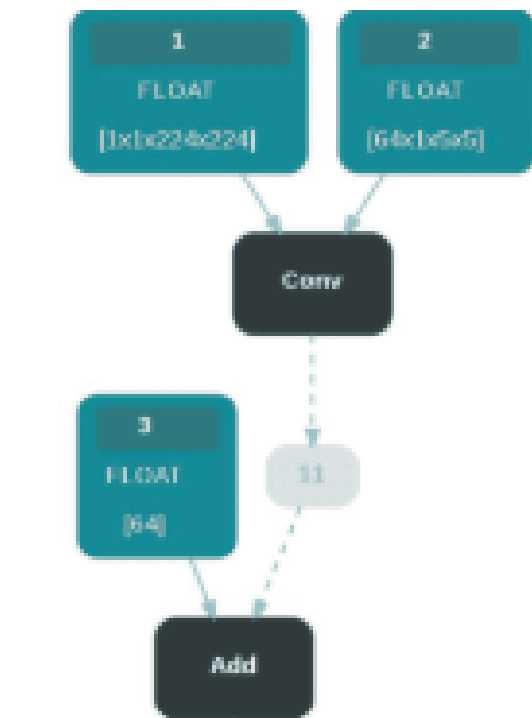
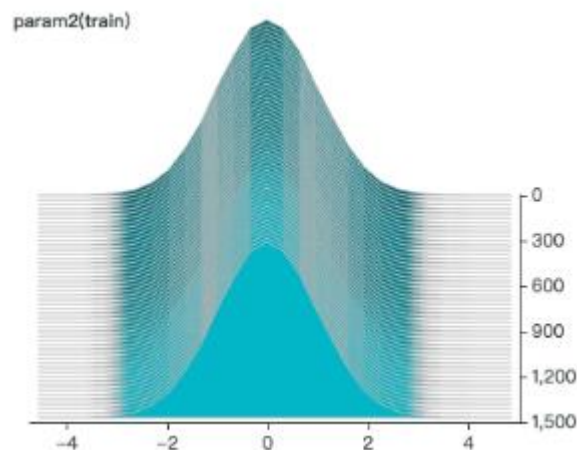
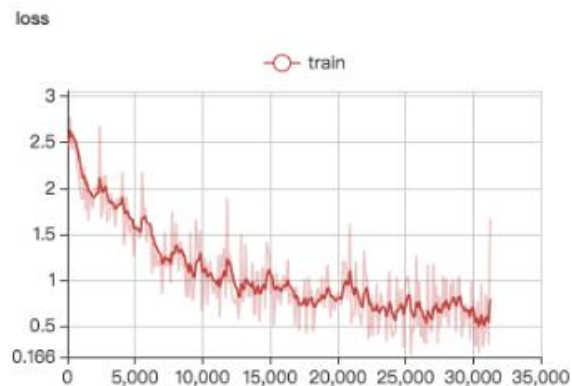
丰富的工具组件

◆ VisualDL

VisualDL 是一款开源的、支持多个深度学习框架的数据可视化工具库；可以将神经网络的各种训练结果以及网络的结构以可视化的形式呈现。

使用文档: <http://visualdl.paddlepaddle.org/>

VisualDL提供scalar、参数分布、模型结构、图像可视化等功能。





丰富的工具组件

◆X2Paddle

为了帮助用户快速从其他框架迁移，PaddlePaddle开源了**模型转换工具X2Paddle**。

- ✓ tensorflow2fluid: 可将TensorFlow、Caffe 的模型转换为Fluid可加载的格式。
- ✓ caffe2fluid: 可将TensorFlow、Caffe 的模型转换为Fluid可加载的格式。
- ✓ onnx2fluid: 还支持ONNX格式的模型转换，比如PyTorch、MXNet、CNTK等。

提供了相应API对比文档，可帮助用户通过现有框架的使用经验快速上手PaddlePaddle的使用。

Caffe2fluid: <https://github.com/PaddlePaddle/X2Paddle/tree/master/caffe2fluid/doc>

Tensorflow2fluid: <https://github.com/PaddlePaddle/X2Paddle/tree/master/tensorflow2fluid/doc>

序号	TensorFlow接口	PaddlePaddle接口	备注
1	<code>tf.abs</code>	<code>fluid.layers.abs</code>	功能一致
2	<code>tf.add</code>	<code>fluid.layers.elementwise_add</code>	功能一致



AI DISCOVERY



PaddlePaddle全景



AI DISCOVERY

PaddlePaddle不仅包含深度学习框架，还提供了一整套紧密关联、灵活组合的完整工具组件和服务平台，有利于深度学习技术的应用落地。





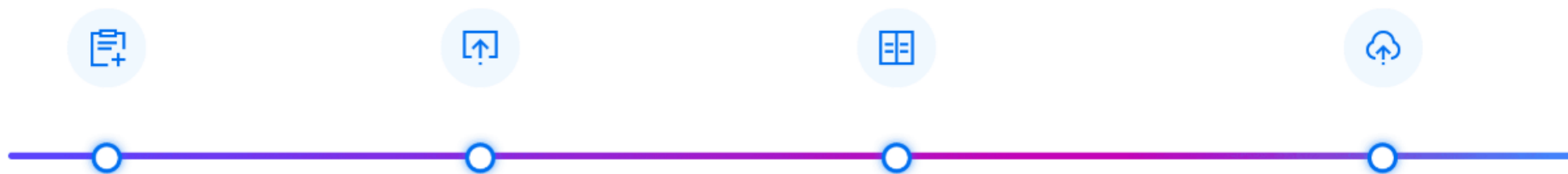
专业的服务平台

◆ EasyDL

致力于为零算法基础的企业用户和
开发者提供高精度的AI模型定制服务。

官网: <http://ai.baidu.com/easydl/>

平台使用流程:



1. 创建模型

2. 上传并标注数据

3. 训练模型并校验效果

4. 上线模型获取API或离线SDK

模型中心

我的模型

创建模型

训练模型

校验模型

发布模型

数据中心

数据集管理

创建数据集

云服务调用数据

模型列表 > 创建模型

模型类别: 图像分类

* 模型名称:

模型归属:

公司

个人

* 所属行业:

请选择行业

* 应用场景:

请选择应用场景

* 邮箱地址:

* 联系方式:

* 功能描述:



专业的服务平台



AI DISCOVERY

◆ AI Studio：一站式深度学习开发平台

集开放数据、开源算法、免费算力三位一体，为开发者提供高效学习和开发环境、高价值高奖金竞赛项目，支撑高校老师轻松实现AI教学，并助力企业加速落地AI业务场景。

官网：<https://aistudio.baidu.com/>

实践平台！

创建项目

X

* 项目环境

python 3.5



* 预加载项目框架

PaddlePaddle v1.3.2



* 配置资源

单机



* 项目名称

项目名称最多输入20字

* 项目描述

项目描述最多可输入100字

数据集

+ 添加数据集

创建数据集

创建

取消

选择运行环境

X

您当前算力卡总余额为0，获取更多算力卡？

☒ CPU: 2 Cores 8GB Memory (免费使用)

☐ GPU: V100 16GB VMem, CPU: 2 Cores 8GB Memory (1算力卡/小时)

确定

取消



AI DISCOVERY



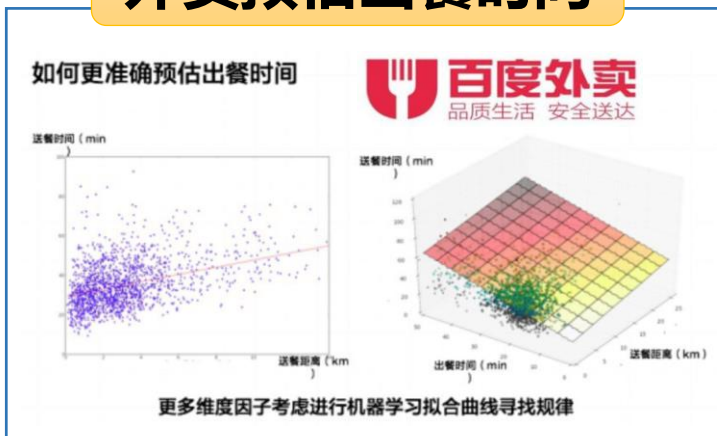
工业应用



AI DISCOVERY

PaddlePaddle已在百度30多项主要产品和服务之中发挥着巨大的作用。

外卖预估出餐时间



精准推荐用户所需信息

Baidu 百度

[PaddlePaddle_易学易用的全开源深度学习平台](#)

百度深度学习平台PaddlePaddle—是并行分布式全功能深度学习框架,易学易用,高效灵活,支持海量图像识别分类,机器翻译和自动驾驶等多个领域的业务需求全面开源!

· 用科技让复杂的世界更简单

技术服务: 客户身份:

联系方式: [我要咨询](#)

字符识别(OCR)

机器翻译



百度翻译手机版提供中英日韩等10多个语种的文本互译;使用OCR技术实现拍照整屏实时翻译、菜单翻译等功能,无需键盘输入,提升用户体验。

病毒和垃圾信息检测

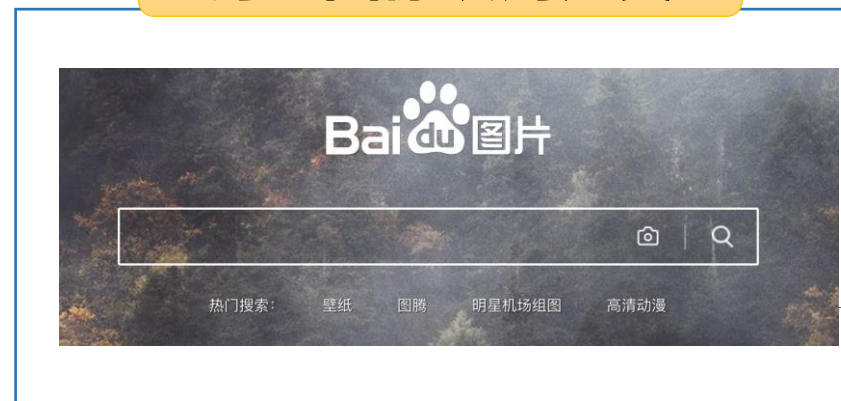
百度卫士

更快,更安全

自动驾驶



海量图像识别分类





工业应用



AI DISCOVERY

通过PaddlePaddle可以实现这些有趣的项目。

高尔夫球场的遥感检测



中科院遥感与地球研究所应用PaddlePaddle**基于遥感数据进行语义分割**，从而达到对高尔夫球场的识别，辅助国家对国土资源进行有效管控。

深度学习质检一体机



大恒图像正在使用PaddlePaddle进行多种产品**外观质检模型**的研发并部署至质检一体机中，实现工业质检效率的显著提升。

耕地地块识别



中科赛诺应用PaddlePaddle对**农业遥感数据进行图像分割**，对耕地面积进行提取，从而达到进行有效的估产并辅助相关其它农事活动。

机器听诊大师



硕橙将深度学习用于**机器设备噪声识别**，实现了设备生产量的实时统计，并评估设备的运行状态。



AI DISCOVERY



工业应用：AI识虫



AI DISCOVERY

北京林业大学智能感知实验室遇到的两大难题：一是虫子太小，辨别难度大。基层工作人员一般不会鉴别，只靠老师和研究生。二是林区面积大、地形复杂，老师加研究生就8个人，一个暑假也只能跑一个林区，想要监测整片树林实在太难了。

中国人工智能学会教育工作委员会及多位高校人工智能学科建设代表与百度展开了一场关于深度学习校企合作的探讨启发了他们，为什么不能用国人自己的深度学习框架，开发一个“AI识虫”系统？

目前“AI识虫”系统的识别准确率达到**90%**，与专业人士识别准确率相当。而且效率大大提高，以往人工检测需要一周的工作量，现在用自动检测一小时内就能处理完。



AI识虫系统运行后，不需要上山就能获得整个林区的规律性数据。



AI DISCOVERY



PaddlePaddle入门



AI DISCOVERY

概述

开发环境

使用入门



AI DISCOVERY



本地环境搭建



AI DISCOVERY

目前PaddlePaddle支持以下环境:

- Ubuntu 14.04 / 16.04 / 18.04
- CentOS 7 / 6
- MacOS 10.11 / 10.12 / 10.13 / 10.14
- Windows7 / 8/ 10 (专业版/企业版)

PaddlePaddle支持三种形式的安装, pip安装、docker安装、源码安装, **本教程以Windows7下pip安装为例。**

若需要docker或源码安装, 请参考:

http://paddlepaddle.org/documentation/docs/zh/1.3/beginners_guide/install/install_Docker.html



本地环境搭建



安装Python

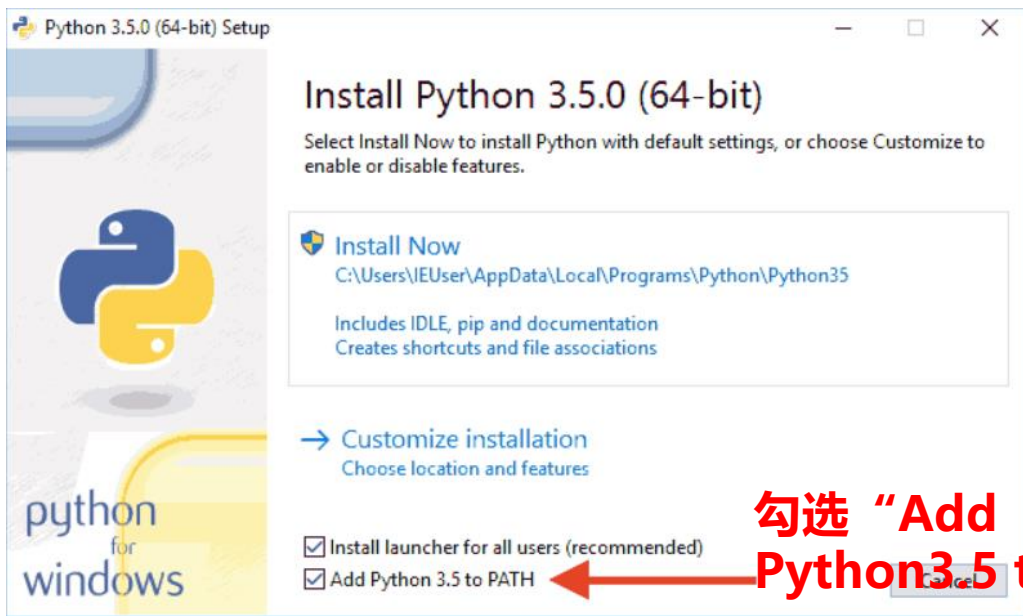
◆ 下载Python3.5

- 1) 自行百度下载
- 2) 百度网盘https://pan.baidu.com/s/1t_mBGiwre7eGTio-0MRqmQ, 提取码: 4efy

◆ 安装Python3.5

◆ 安装是否成功

cmd中执行python命令





本地环境搭建



AI DISCOVERY



安装PaddlePaddle

◆在线pip安装

cmd中，执行pip install paddlepaddle,开始安装

```
C:\Users\wenxiaoxin>pip install paddlepaddle
```

```
python-4.0.0.21 paddlepaddle-1.3.0 protobuf-3.1.0 pyparsing-2.3.1 python-dateutil-2.8.0 pytz-2018.9 pyyaml-3.13 rarfile-3.0 recordio-0.1.7 requests-2.9.2 scipy-1.2.1 singledispatch-3.4.0.3 six-1.12.0
You are using pip version 9.0.1, however version 19.0.3 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.
C:\Users\wenxiaoxin>
```

◆升级pip版本

如果提示pip版本过低，按照提示，执行python -m pip install --upgrade pip, 升级pip



本地环境搭建

安装PyCharm

◆下载PyCharm

- 1) 自行百度下载
- 2) 百度网盘https://pan.baidu.com/s/1t_mBGiwre7eGTio-0MRqmQ, 提取码: 4efy

◆安装PyCharm

默认安装即可





本地环境搭建

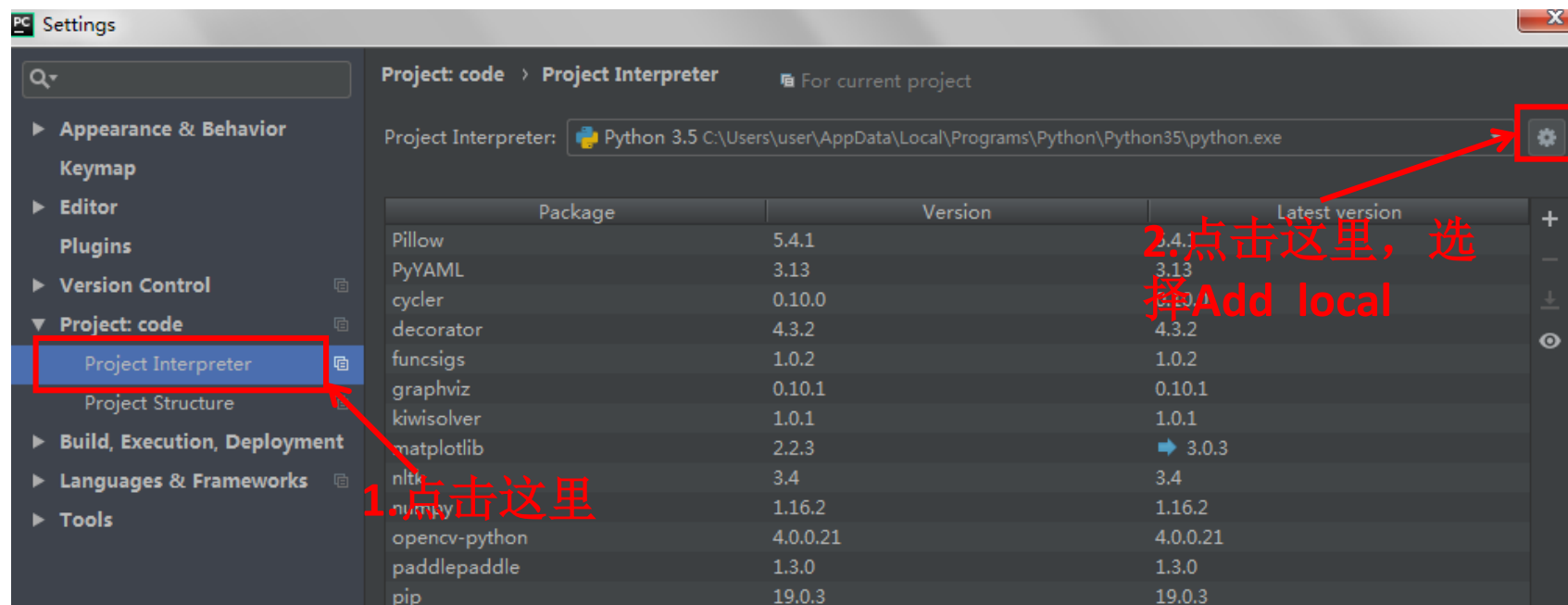


AI DISCOVERY

◆配置Python解释器

1) 点击file->Settings

2) 点击Project Interpreter,选择Add Local



AI DISCOVERY



本地环境搭建



AI DISCOVERY

◆配置Python解释器

3) 点击Virtualenv Environment->Existing environment, 选择Python安装路径, 勾选“Make available to all projects”, 配置完成。





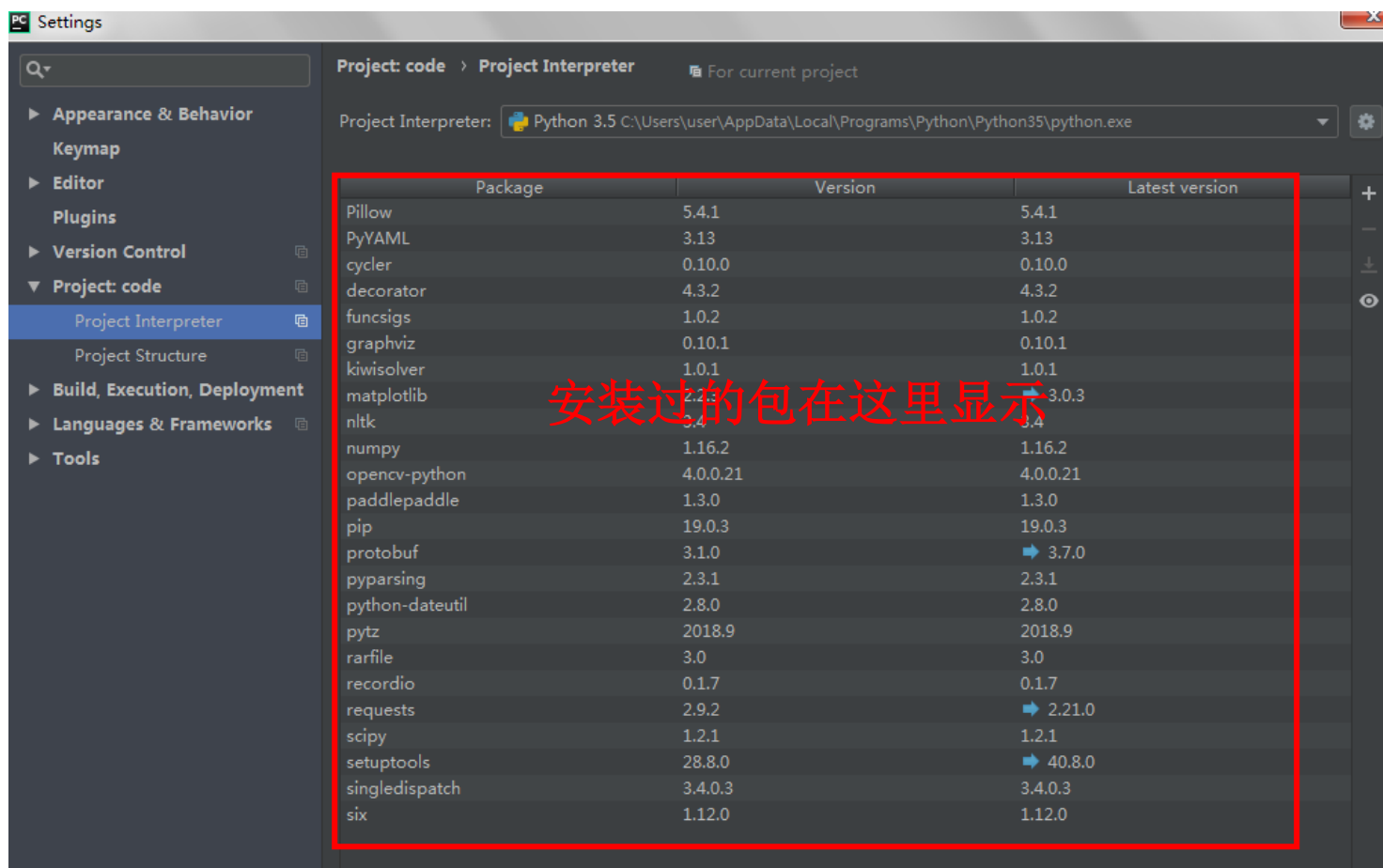
本地环境搭建



AI DISCOVERY

◆配置Python解释器

4) 加载完成后，会显示出Python安装过的所有包，证明PyCharm调试已经成功了。



AI DISCOVERY



本地环境搭建

AI DISCOVERY

◆代码测试



test.py

双击我！！

将测试代码复制到PyCharm中，点击运行，出现下图左侧所示信息，则运行成功

```
Pass:0, Batch:0, Cost:2.70508, Accuracy:0.06250
Pass:0, Batch:100, Cost:0.38185, Accuracy:0.86719
Pass:0, Batch:200, Cost:0.23436, Accuracy:0.89844
Pass:0, Batch:300, Cost:0.44820, Accuracy:0.84375
Pass:0, Batch:400, Cost:0.26247, Accuracy:0.90625
Test:0, Cost:0.21892, Accuracy:0.93305
save models to /home/aistudio/data/hand_inference.model
Pass:1, Batch:0, Cost:0.27801, Accuracy:0.91406
Pass:1, Batch:100, Cost:0.28942, Accuracy:0.92188
Pass:1, Batch:200, Cost:0.16124, Accuracy:0.95312
Pass:1, Batch:300, Cost:0.21073, Accuracy:0.92188
Pass:1, Batch:400, Cost:0.22340, Accuracy:0.90625
Test:1, Cost:0.14514, Accuracy:0.95342
save models to /home/aistudio/data/hand_inference.model
```

```
Run: hand_write x
W:\pycharm\venv\Scripts\python.exe W:/pycharm/paddlepaddle/hand_write.py
Traceback (most recent call last):
  File "W:/pycharm/paddlepaddle/hand_write.py", line 2, in <module>
    import numpy as np
  ImportError: No module named 'numpy'
Process finished with exit code 1
```

注：若出现上图右侧所示信息，则是由于缺少Numpy安装包引起的，cmd中执行pip install numpy命令即可。

AI DISCOVERY



AI Studio平台



AI DISCOVERY

百度AI Studio一站式开发平台：集开放数据、开源算法、免费算力三位一体，为开发者提供高效学习和开发环境、高价值高奖金竞赛项目，支撑高校老师轻松实现AI教学，并助力企业加速落地AI业务场景。。**官网地址：** <http://aistudio.baidu.com/>

AI Studio特性

云端集成

计算资源

比赛平台

样例模型

视频教程

简单易用

项目管理

项目共享

数据集成

无需安装

运行高效

资源配置

集群支撑

多项目并行

P40 GPU

资源免费

视频教程

空间资源

计算资源

代码管理

百度AI Studio平台已经为用户**预置**了Python语言环境, 以及百度PaddlePaddle深度学习开发框架. 同时用户可以在其中**自行加载**Scikit-Learn等机器学习库.

AI Studio平台

AI Studio个人版

AI Studio教育版



AI Studio个人版



项目列表

- ◆ 百度AI学习项目 样例工程，经典应用场景
- ◆ 开发则共享项目 项目共享功能，相互学习
- ◆ 我的项目 开发者项目列表



AI Studio个人版



AI DISCOVERY

◆ 创建你自己的单机项目

Step1.创建项目: 点击按钮创建项目

Step2.项目环境: Python2.7或Python3.5

Step3.预加载模块: PaddlePaddle-v1.3.2

Step4.配置资源: 选择单机 (可选单机或远程集群)

Step5.项目名称/项目描述: 用来标识项目, 便于日后进行查找和管理, 创建后支持修改.

创建项目 ×

* 项目环境

python 3.5

* 预加载项目框架

PaddlePaddle v1.3.2

* 配置资源

单机

* 项目名称

项目名称最多输入20字

* 项目描述

项目描述最多可输入100字

数据集

+ 添加数据集

创建数据集

创建

取消



AI DISCOVERY



AI Studio个人版



AI DISCOVERY

◆ 创建你自己的单机项目

✓如果项目涉及到数据集, 可以考虑直接使用系统预置的数据集, 点击**添加数据集**按钮

注意: 每个项目最多可以引入两个数据集, 如无合适的数据集, 用户也可以**自行上传创建新数据集**.

✓点击“创建”, “查看”, 可进入项目详情页



< 返回

创建项目

公开数据集

个人数据集

公开/个人数据集切换



房价预测Fluid
房价预测Fluid



经典MNIST数据集
手写数字, 训练集数量60000, 测试集数量10000. 参考: <http://yann.lecun...>

勾选



CIFAR10数据集
包含60,000张32x32的彩色图片, 10个类别, 每个类包含6,000张。其中...



房价预测
房价预测



数字识别CNNFluid
数字识别CNNFluid



1

2

3

4

5

...

13



添加(1/2)

点击添加



AI Studio个人版

AI Studio 项目 **数据集** 比赛 教育 帮助文档

数据集

汇聚海量开放数据, 链接真实业务场景

创建数据集 创建数据集

公开数据集 查看更多> 我收藏的数据集

当前还没有收藏数据集

房	房价预测Fluid 房价预测Fluid
经	经典MNIST数据集 手与数字, 训练集数量60000, 测试集数量10000. 参考: http://yann.lecun.com/exdb/mnist/
鸢	鸢尾花数据集 鸢尾花数据集
C	CIFAR10数据集 包含60,000张32x32的彩色图片, 10个类别, 每个类包含6,000张。其中50,000张图片作为训练集, 10

我创建的数据集 查看更多>

鸢尾花数据集
SVHN
SVHN数据集
55
人脸识别数据集

数据集列表

- ◆ 公开数据集
- ◆ 收藏数据集
- ◆ 个人数据集



AI Studio个人版



AI DISCOVERY

◆ 创建数据集

创建数据集

* 数据集名称

鸢尾花数据集

* 数据集简介

每一行数据由 4 个特征值及一个目标值组成。
4 个特征值分别为：萼片长度、萼片宽度、花瓣长度、花瓣宽度
目标值为三种不同类别的鸢尾花，分别为：Iris Setosa，Iris
Versicolour，Iris Virginica。

* 上传数据集文件

iris.data



最多输入50个字



上传文件

最多上传10个文件，单个文件不超过10GB

下一步

取消



AI DISCOVERY



AI Studio个人版



AI DISCOVERY

◆ fork公开项目

如果不熟悉操作, 则可以直接fork百度AI学习项目, 或者其他开发者共享的项目, 加快学习速度.



Notebook

数字识别CNNFluid

项目作者: Hangjiweilacloud

编辑时间:

2018-08-13 09:44:20

状态: 已公开

★ 收藏

fork

Step1. 点击

fork项目

fork来源: 数字识别CNNFluid

Step2. 填写信息

* 项目名称

项目描述最多可输入20字

* 项目描述

项目描述最多可输入100字

创建

取消

Step3. 点击创建



AI DISCOVERY



Notebook环境



AI DISCOVERY

◆ 项目详情页

✓ 进入项目详情页

项目大厅 > 我的项目 > test

点击进入项目代码编辑页面

✓ 点击进入，项目环境初始化

Notebook test

项目作者: 彩虹糖wrj

编辑时间: 2019-03-02 15:34:59

状态: 未公开

删除

修改

设置为公开

项目简介:
test

引用数据集:
- CIFAR10数据集

版本内容

Fork记录

在线服务

运行项目

创建预测服务

In []:

查看当前挂载的数据集目录
!ls /home/aistudio/data/

In []:

查看个人持久化工作区文件
!ls /home/aistudio/work/

公开项目

删除项目

修改项目名称、数据集等



AI DISCOVERY



Notebook环境



AI DISCOVERY

◆ 代码编辑页面

test

文件 编辑 运行 Kernel

数据集

CIFAR10数据集
cifar-10-batches-py.zip

数据集

Cell操作 Notebook操作 文件操作 Kernel操作 设置操作

操作区

新增cell类型: Code 新增cell | 运行当前cell | 上移当前cell | 下移当前cell | 上方插入cell | 下方插入cell

1 # 查看当前挂载的数据集目录
2 !ls /home/aistudio/data/

2
In[] 运行 删除 切换cell类型
1 # 查看个人持久化工作区文件
2 !ls /home/aistudio/work/

内容编辑区



AI DISCOVERY



Notebook环境



AI DISCOVERY

◆ 新建块

✓选择块类型：Code或Markdown,分别插入代码块或文字块

新增cell类型:

Code



新增cell

✓代码块，输入代码，点击下方
输出运行结果

```
1 print("Welcome to AIStudio")
```

运行输出

Welcome to AIStudio

✓文字块，支持Markdown格式, 点击【预览】则出现渲染后的效果

预览

删除

切换cell类型

B

I

“

🔗

H

</>

≡

≡

🔗

🔗

🔗

****Welcome to AIStudio****

编辑

删除

切换cell类型

Welcome to AIStudio



AI DISCOVERY



Notebook环境

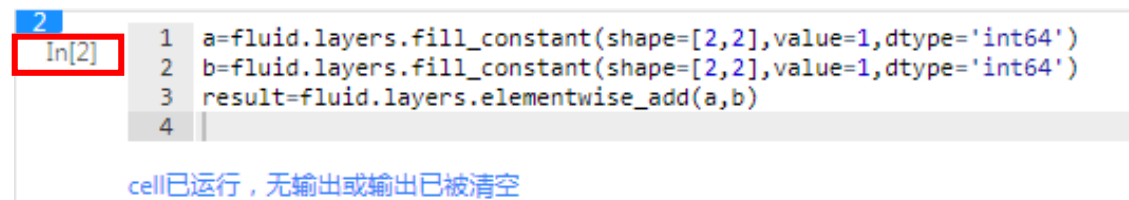


AI DISCOVERY

◆ 工具栏



◆ In提示符



In[]:程序未运行

In[num]:程序运行后

In[*]:程序正在运行



AI DISCOVERY



Notebook环境



AI DISCOVERY

◆ Linux命令

```
5 In[3]: 1 !ls /home/aistudio/
```

运行输出

```
59313.ipynb data work
```

Linux命令前加!

运行Linux命令的方式是在Linux命令前加一个!, 就可以在块里运行。

◆ 文件夹

文件夹		
	名称	
数据集	操作	
	work	
数据集	data	

按照树形结构展示
/home/aistudio路径下的文件夹和文件。可以在该目录下文件夹和文件的上传、删除、重命名等操作。

◆ 数据集

文件夹	数据集	
	infer_3	
数据集	infer_3.png	
	infer_9	
数据集	infer_9.jpg	

复制路径

可以复制数据集文件的路径, 并置于代码中



AI DISCOVERY



AI Studio教育版



AI DISCOVERY

◆ AI Studio教育版入口一：



◆ AI Studio教育版入口二：

57 15
我的项目 我的数据集

AI Studio教育

帮助学校、机构建立自己的线上教学班级，提供从教学项目、AI在线实训环境、教学管理的全流程在线一站式解决方案。[了解更多>](#)

使用AI Studio教育专版

◆ AI Studio教育版使用文档



教育

教师-基础介绍
教师-教学班级管理
教师-学生管理
教师-教学专区
教师-班级信息
学生-基础介绍
学生-身份认证
学生-项目学习



AI DISCOVERY



PaddlePaddle入门



AI DISCOVERY

概述

开发环境

使用入门



AI DISCOVERY



PaddlePaddle计算模型

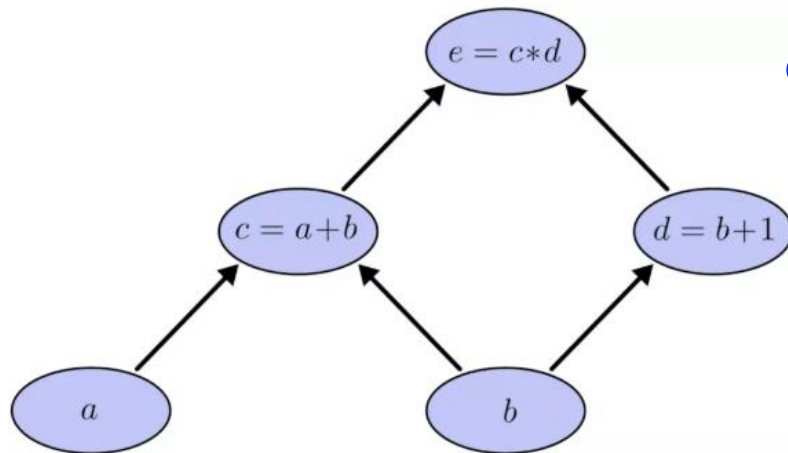


AI DISCOVERY

计算图

描述计算的经典方式：

- ✓节点代表计算
- ✓边代表计算之间的依赖关系



$$e=(a+b)*(b+1)$$

PaddlePaddle Fluid去掉静态计算图的概念，采用Program的形式动态描述计算过程。

在Fluid程序实例中，区分编译时和运行时：

- ✓编译时：用户像写程序一样描述计算。
- ✓运行时：执行规划好的计算

Fluid中不再有模型的概念，
提倡像写程序一样描述计算



AI DISCOVERY



PaddlePaddle计算模型



AI DISCOVERY



Program

在Fluid中，一个神经网络任务（训练/预测）被描述为一段Program

Fluid通过提供顺序、分支和循环三种执行结构的支持，让用户可以通过组合描述任意复杂的模型。

➤ 顺序执行

```
x = fluid.layers.data(name='x', shape=[13], dtype='float32')
y_predict = fluid.layers.fc(input=x, size=1, act=None)
y = fluid.layers.data(name='y', shape=[1], dtype='float32')
cost = fluid.layers.square_error_cost(input=y_predict, label=y)
```

➤ 条件分支

```
with fluid.layers.control_flow.Switch() as switch:
    with switch.case(global_step == zero_var):
        fluid.layers.tensor.assign(input=one_var, output=lr)
    with switch.default():
        fluid.layers.tensor.assign(input=two_var, output=lr)
```

网络结构修改的灵活性、模型搭建的便捷性



PaddlePaddle计算模型



PaddlePaddle Fluid使用Operator表示对数据的操作。

```
a=fluid.layers.fill_constant(shape=[2,2],value=1,dtype='int64')
b=fluid.layers.fill_constant(shape=[2,2],value=1,dtype='int64')
result=fluid.layers.elementwise_add(a,b)
print(result)
```

开发者的所有 Operator
都将写入 Program。

一段PaddlePaddle Fluid程序中通常存在2段Program:

1) default_startup_program

定义了模型中可学习参数的初始化等各种操作

2) default_main_program

定义了由用户定义的模型，包括了前向、反向、优化及所有必要的计算

使用Fluid的核心就是构建起default_main_program





PaddlePaddle数据模型



AI DISCOVERY



张量(Tensor)是标量和向量的总称

```
3                                # 标量
[1., 2., 3.]                    # 一维数组, 也称一维向量, 形状为[3]。
[[1., 2., 3.], [4., 5., 6.]]    # 二维数组, 也称二维向量, 形状为[2, 3]。
```

功能上看, 张量可以简单理解为多维数组。

◆Tensor常量

```
a=fluid.layers.fill_constant(shape=[2,2],value=1,dtype='int64')
```

定义张量的常量a, 并指定它们的形状是[2,2], 并赋值为1铺满整个张量, 类型为int64。

◆Tensor变量

```
var1=fluid.layers.data(name='var1',shape=[2,2],dtype='int64')
```

name:张量的名字, 是张量的唯一标识符;

shape:张量的维度, shape=[2,2]表示var1为一个二维数组, 每维数组长度均为2;

dtype:类型, 每个张量会有一个唯一的类型。



AI DISCOVERY



PaddlePaddle数据模型



AI DISCOVERY

以向量加法为例

```
a=fluid.layers.fill_constant(shape=[2,2],value=1,dtype='int64')
b=fluid.layers.fill_constant(shape=[2,2],value=1,dtype='int64')
result=fluid.layers.elementwise_add(a,b)
print(result)
```

在PaddlePaddle中，张量中并没有真正保存数字，它保存的是如何得到这些数字的计算过程。result中并不会得到加法的结果，而会得到对结果的一个引用。

输出

[2,2]
[2,2]



```
name: "elementwise_add_0"
type {
  type: LOD_TENSOR
  lod_tensor {
    tensor {
      data_type: INT64
      dims: 2
      dims: 2
    }
    lod_level: 0
  }
}
persistable: false
```



PaddlePaddle中张量计算的结果不是一个具体的数字，而是一个张量的结构。



张量本身没有存储具体的数字，那如何获取这些具体的数字呢



AI DISCOVERY



PaddlePaddle运行模型



AI DISCOVERY

Executor

PaddlePaddle中使用Executor来执行定义好的运算。

向量加法中，当构建好 Program 后，需要创建 Executor，进行初始化 Program 和训练 Program。

#定义Executor

```
place = fluid.CPUPlace()
```

#定义运算场所为cpu

```
exe = fluid.Executor(place)
```

#创建执行器

```
exe.run(fluid.default_startup_program())#初始化Program
```

#训练Program，开始计算

```
myresult = exe.run(
```

```
    program=fluid.default_main_program(),#默认Program
```

```
    fetch_list=[result])
```

#定义输出

```
print(myresult)
```

输出

```
[array([[2, 2],  
        [2, 2]], dtype=int64)]
```

可将希望查看的变量添加在fetch_list中



AI DISCOVERY



目录

更多AI知识，请关注



1

深度学习方法（上）

深度学习发展历程、为什么使用深度学习

2

深度学习方法（下）

深度学习使用方法使用步骤、正向传播、反向传播

3

PaddlePaddle入门

概述、开发环境、使用入门

4

课程实践

实践：房价预测、手写数字识别



AI DISCOVERY



AI DISCOVERY



课程实践



AI DISCOVERY

实践：房价预测、手写数字识别



AI DISCOVERY