算法

机器学习与人工智能（二）

陈一帅

yschen@bjtu.edu.cn

北京交通大学电子信息工程学院

内容

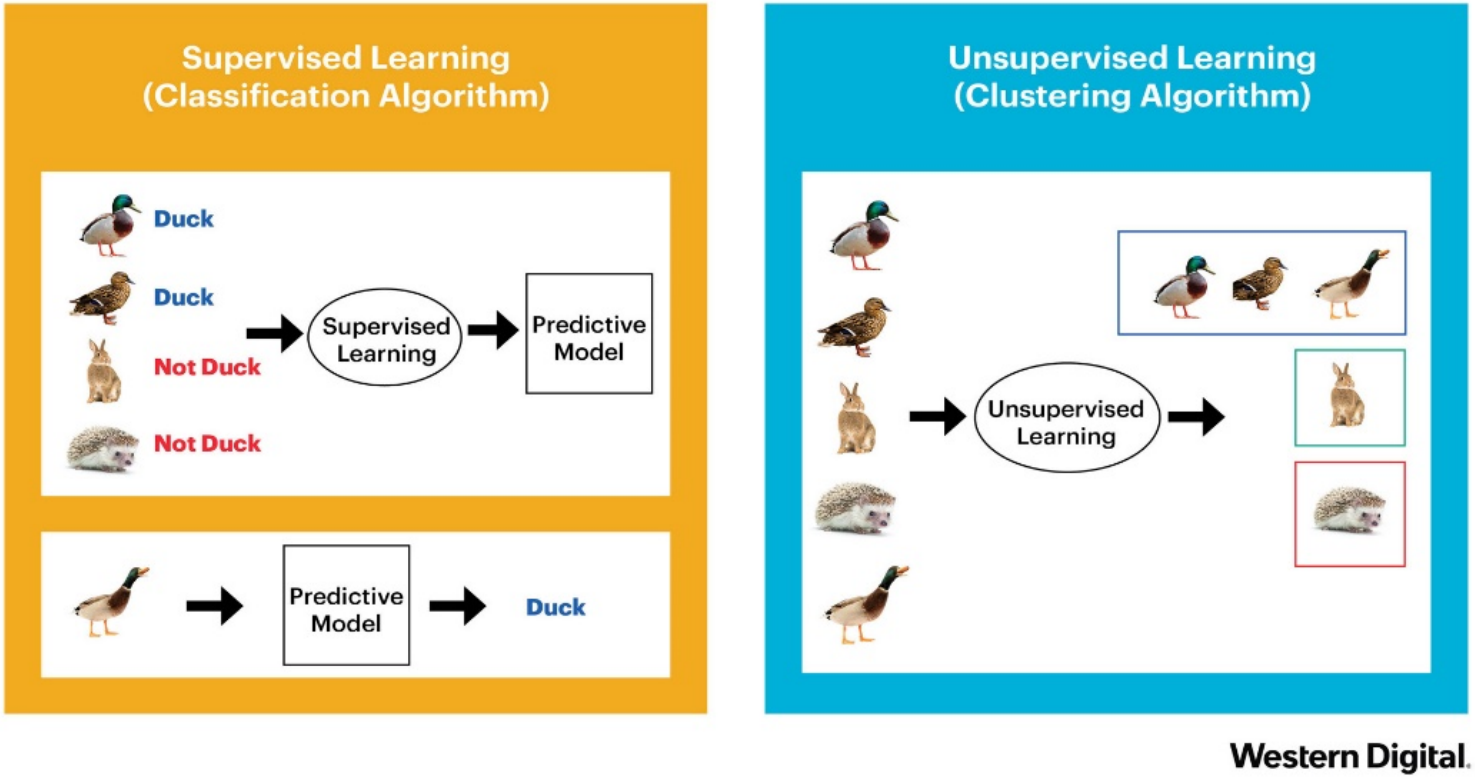
- 介绍
- 机器学习模型
- 深度学习模型
- 模型训练
- 模型选择

算法

- 有监督
- 无监督
- 半监督
- 增强学习

有监督和无监督

- 有监督：已知正确答案，比如图片类别
- 无监督：没有正确答案，比如只有图片



1）有监督学习

Supervised learning

已知正确答案

步骤

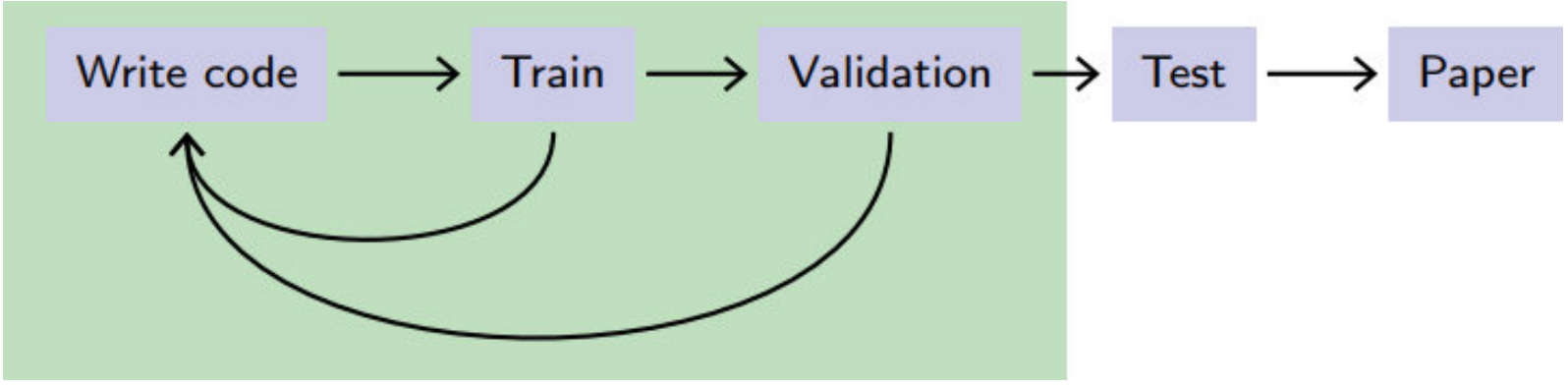
1. 打标
2. 训练
3. 测试

1）打标

1. 收集数据集
2. 打标
    - 给图片标注：“猫”，“狗”
3. 把数据分为三部分
    - 训练集：训练模型
    - 验证集：选择模型参数
    - 测试集：测试模型准确度

2）训练模型

1. 训练：训练模型
2. 验证：选择模型参数
3. 测试：在测试集上，对模型进行最终的评估



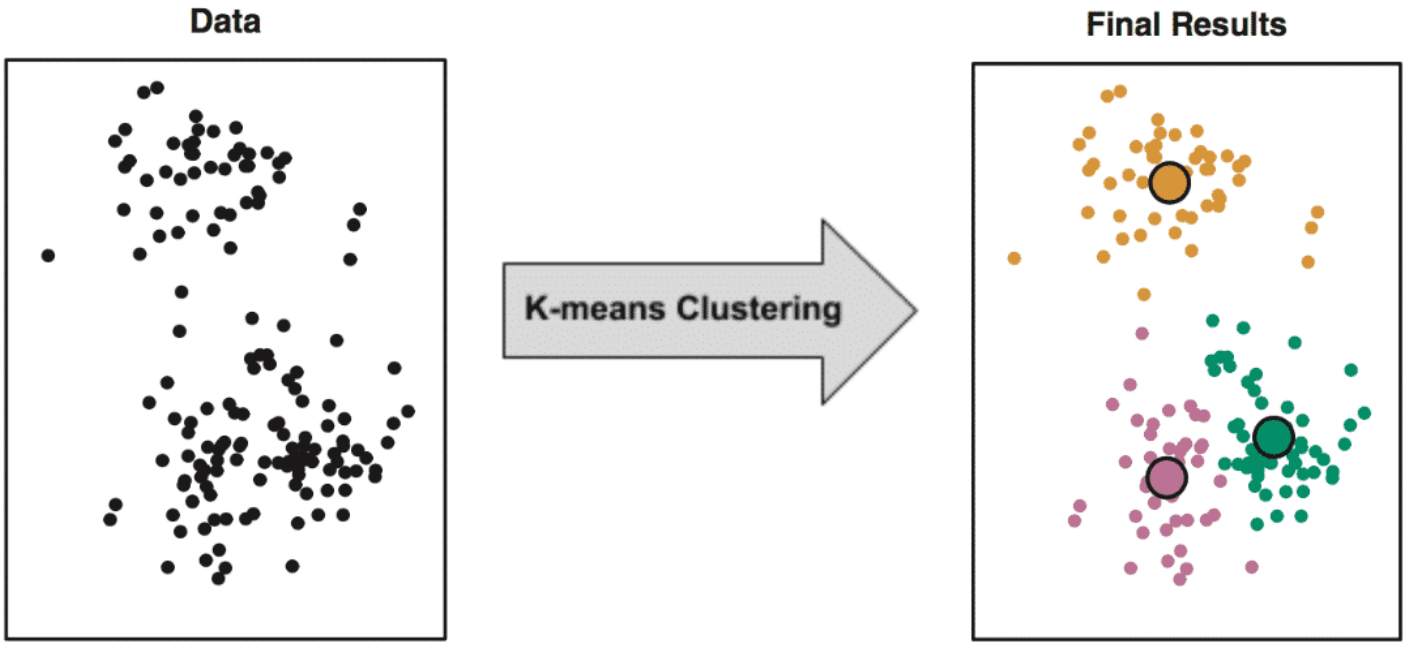
2）无监督学习

Unsupervised learning

有数据，无标签，就在数据上寻找规律

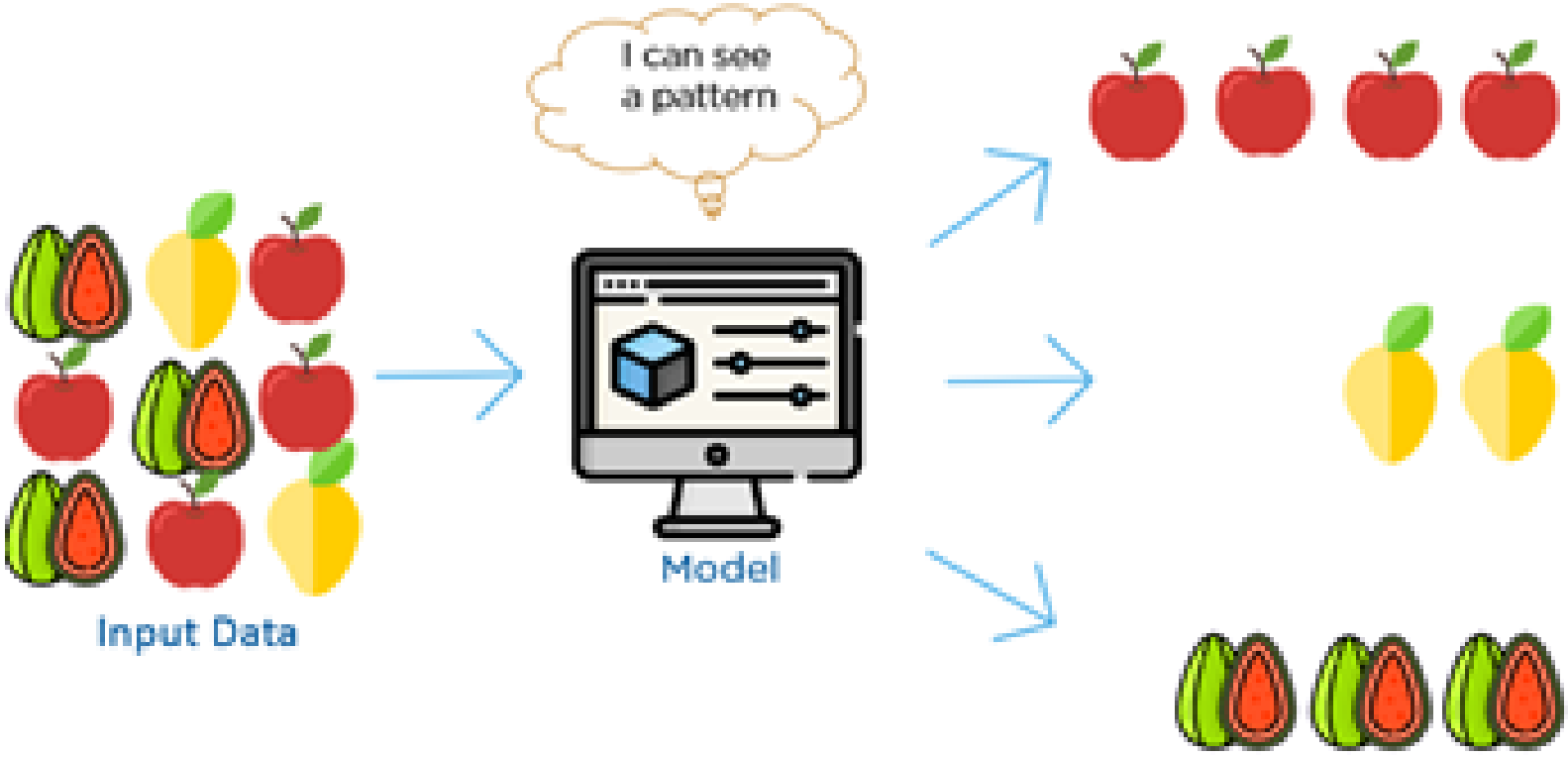
1）聚类

指定聚为3个簇



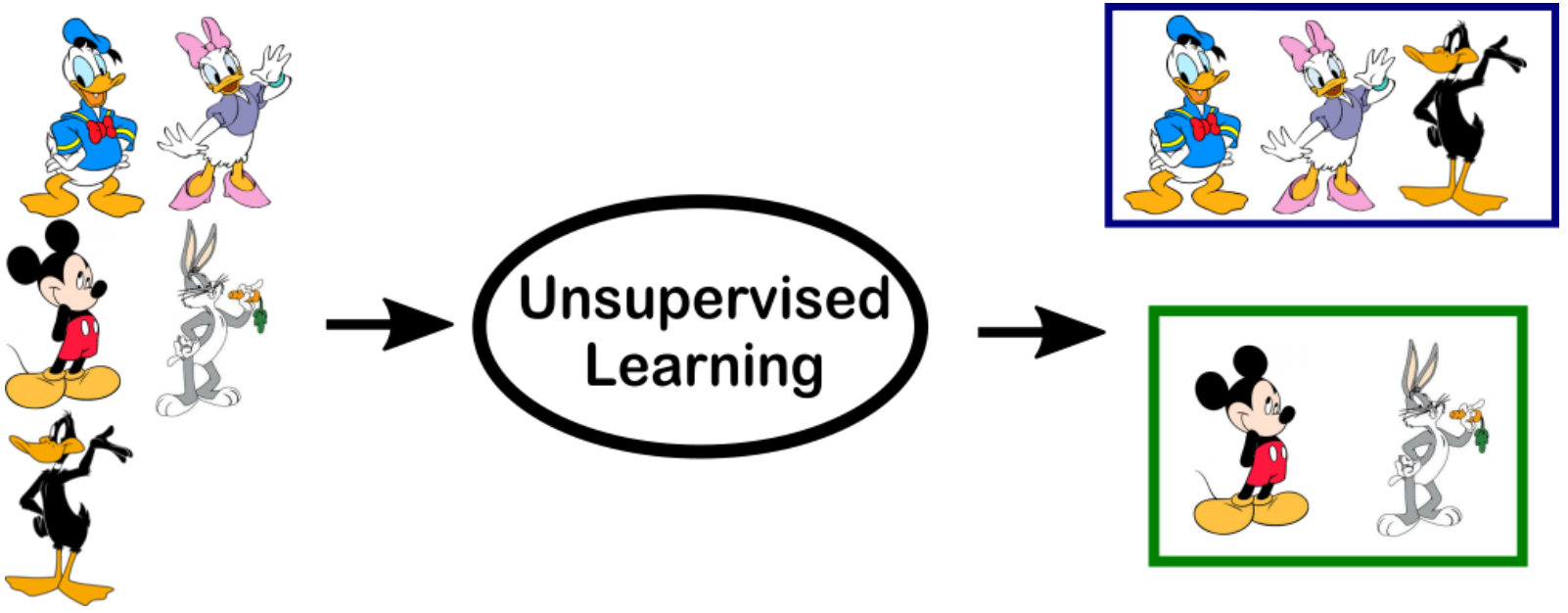
1）聚类

- 聚完类后，观察各簇，获得其物理意义
- 结果可能是这样的



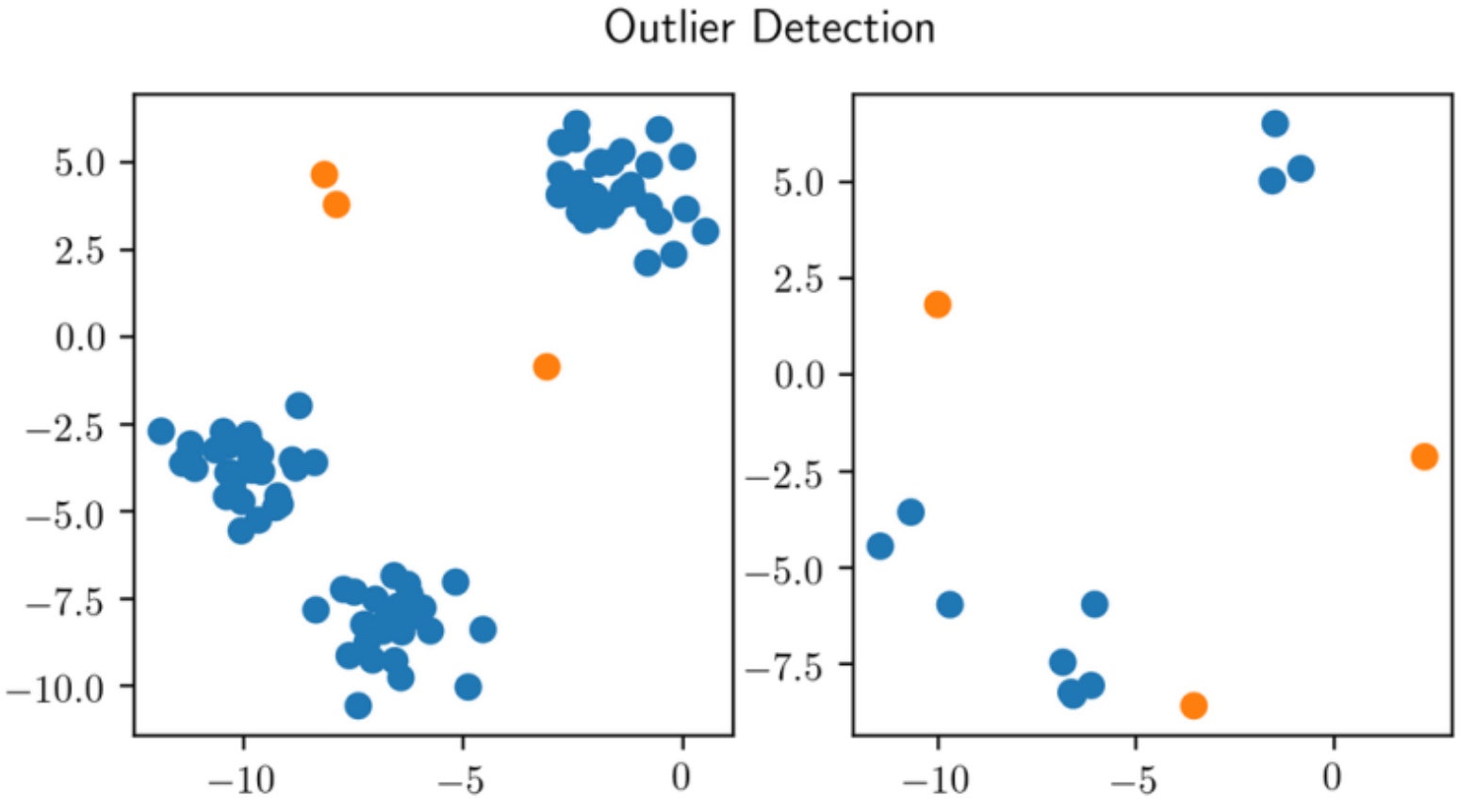
1）聚类

- 结果也可能是这样的



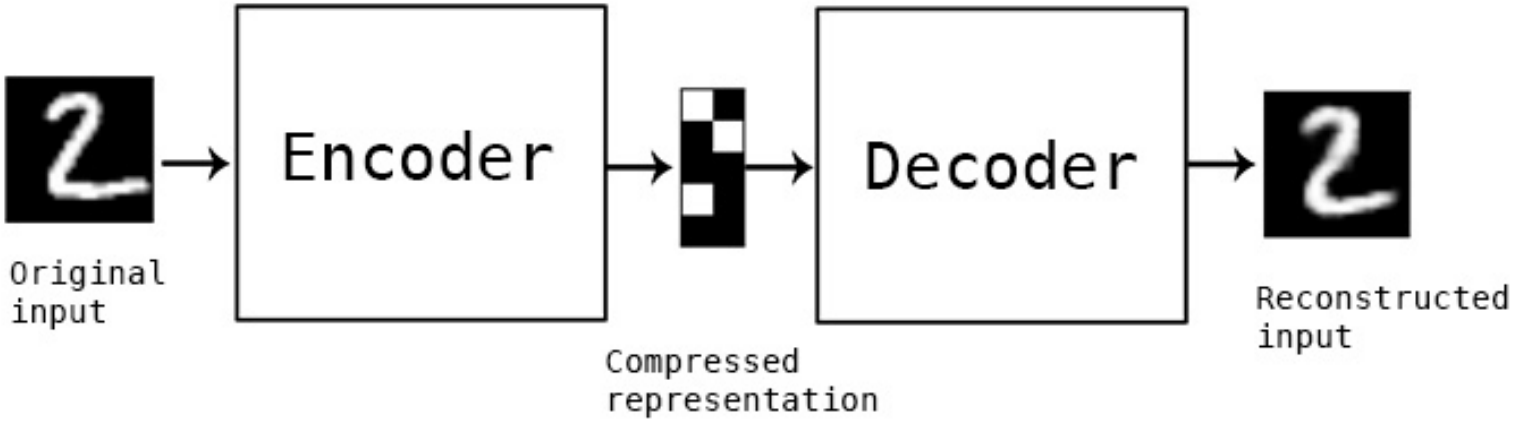
2）异常检测

发现离群的点，即异常点



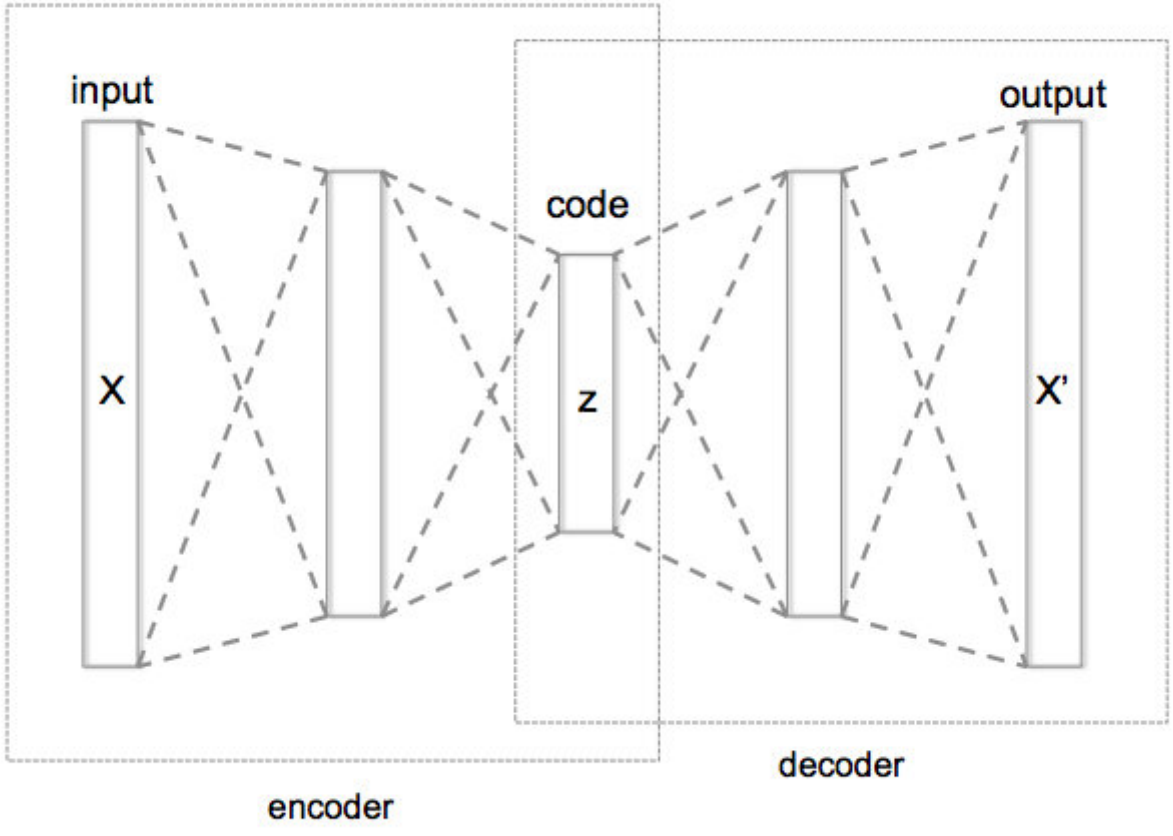
3）自编码

- 编码压缩，得到原始图像的压缩表征
- 译码根据此压缩表征，恢复原图像



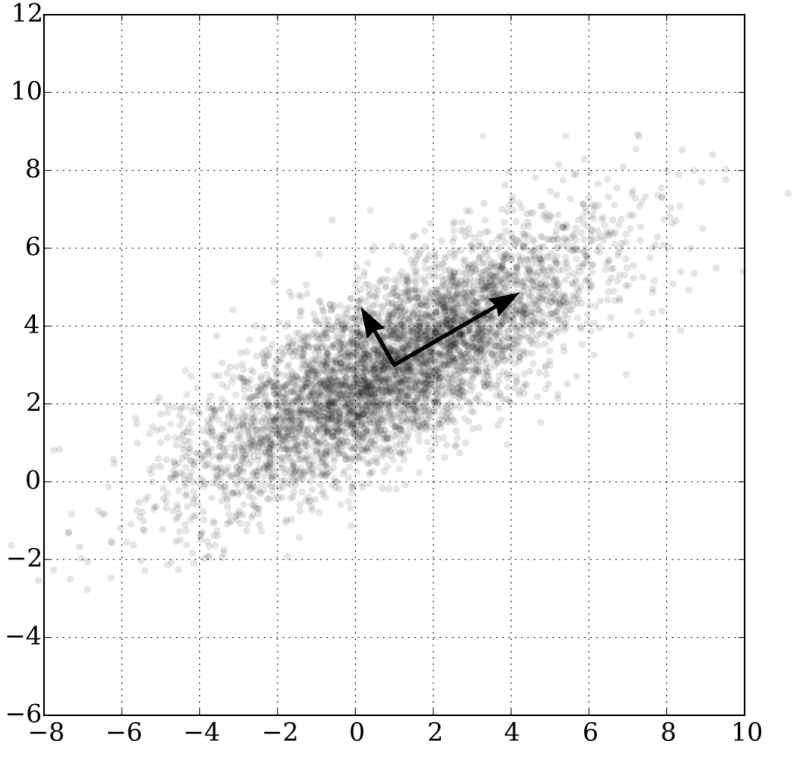
3）自编码

- 压缩后的结果，就是自编码得到的数据的Code
- 一般用深度神经元网络做编码器和解码器



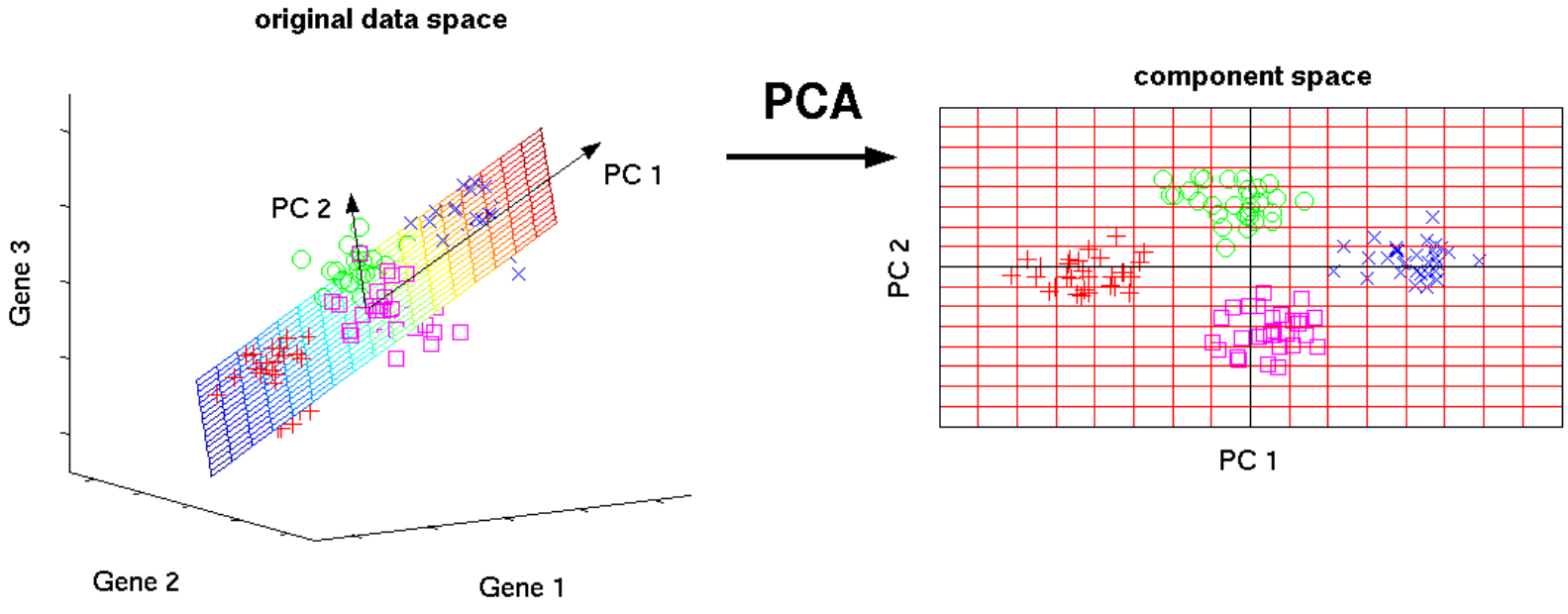
4）主元素分析

- PCA：Principal Component Analysis
- 数据信息主要在其主元素向量上



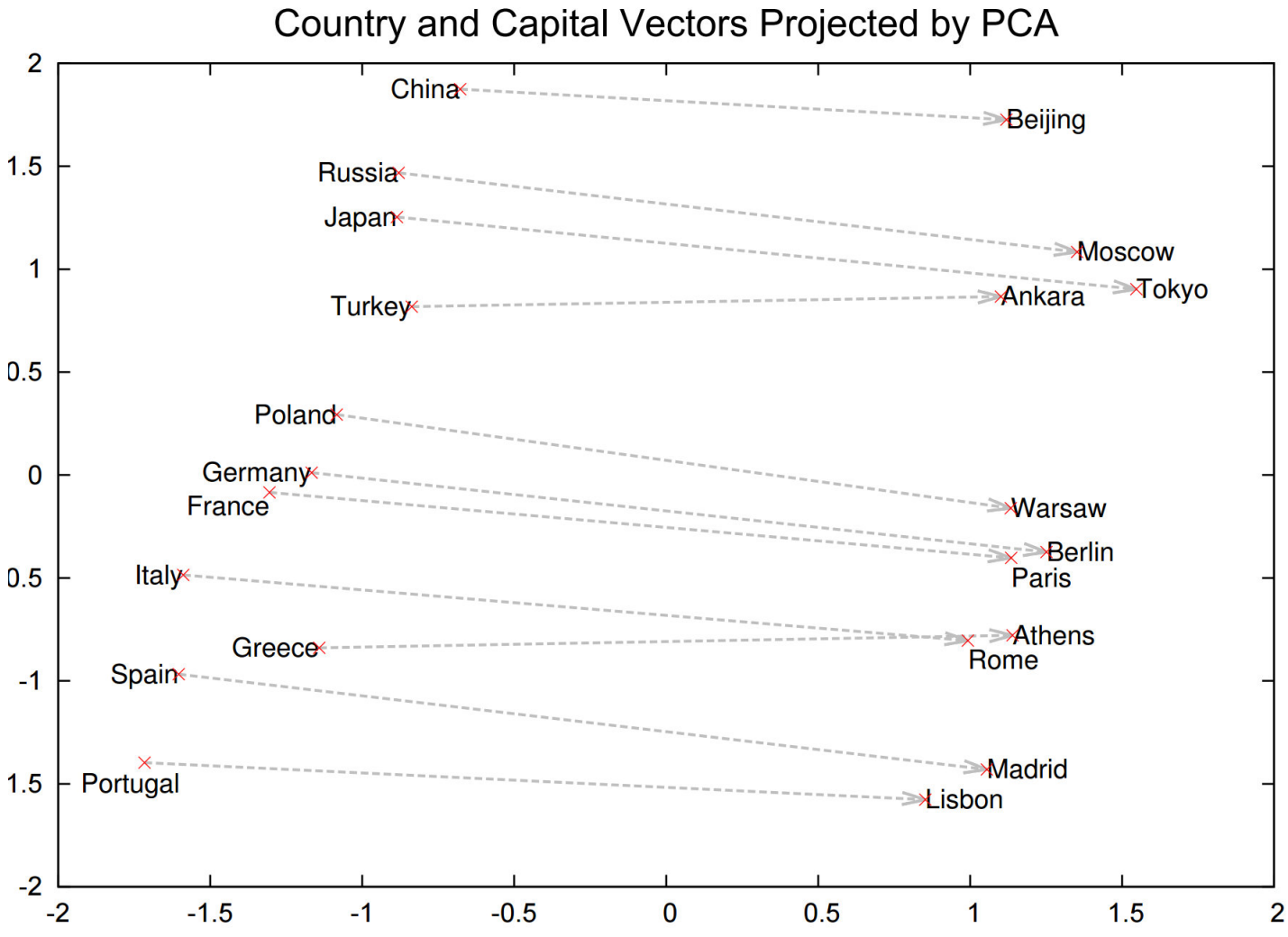
4）主元素分析

- 利用PCA，将3维数据表示在2维上
- 丢失信息不多，达到了降维效果



4）主元素分析

利用PCA得到的单词表征

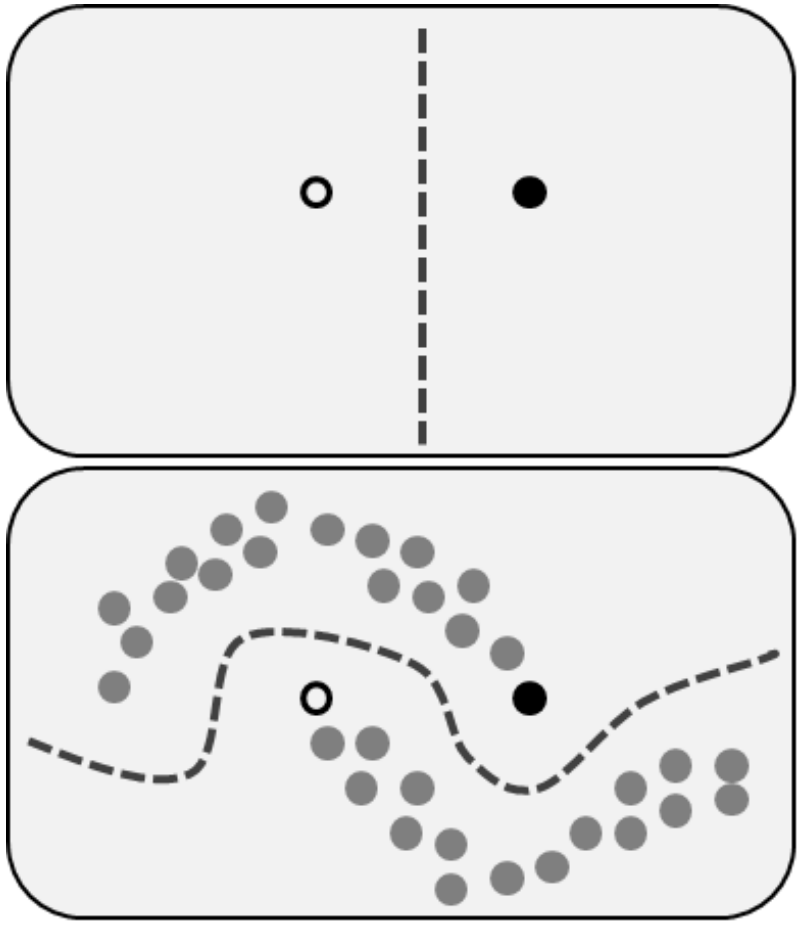


3）半监督学习

Semisupervised Learning

半监督学习

打标费时费力，利用大量没有打标的数据，结合少量打标数据，提高性能



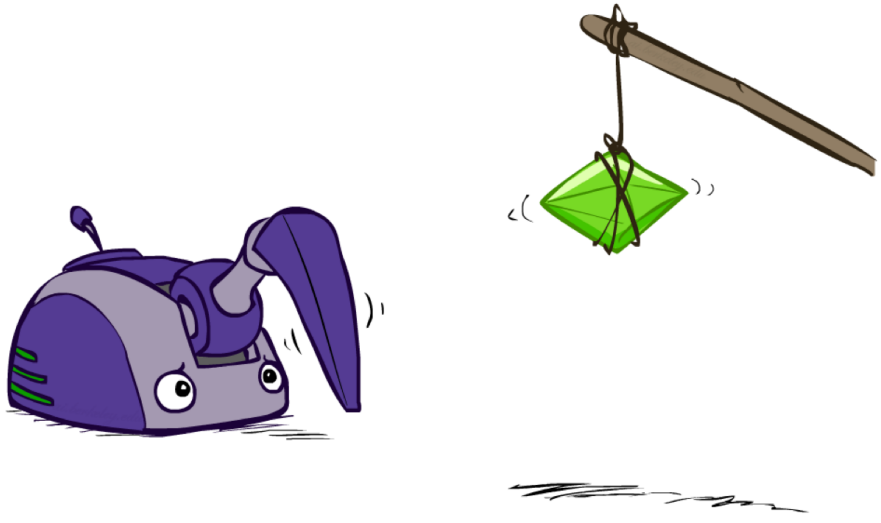
4）增强学习

Reinforcement Learning

根据获得的回报，进行学习

基于回报的学习

- 没有打标数据集
- 但能判断是否有回报reward
- 根据获得的回报，进行学习
- 目标：最大化收益



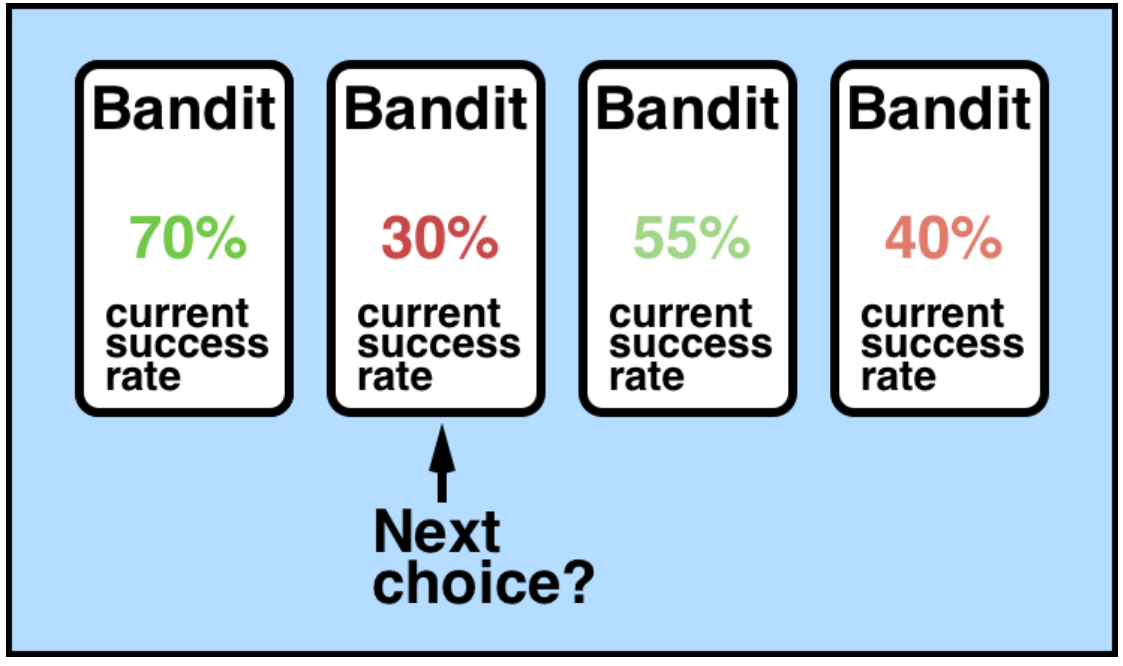
多臂老虎机

Multi-Arm Bandit

选择哪个机器玩呢?

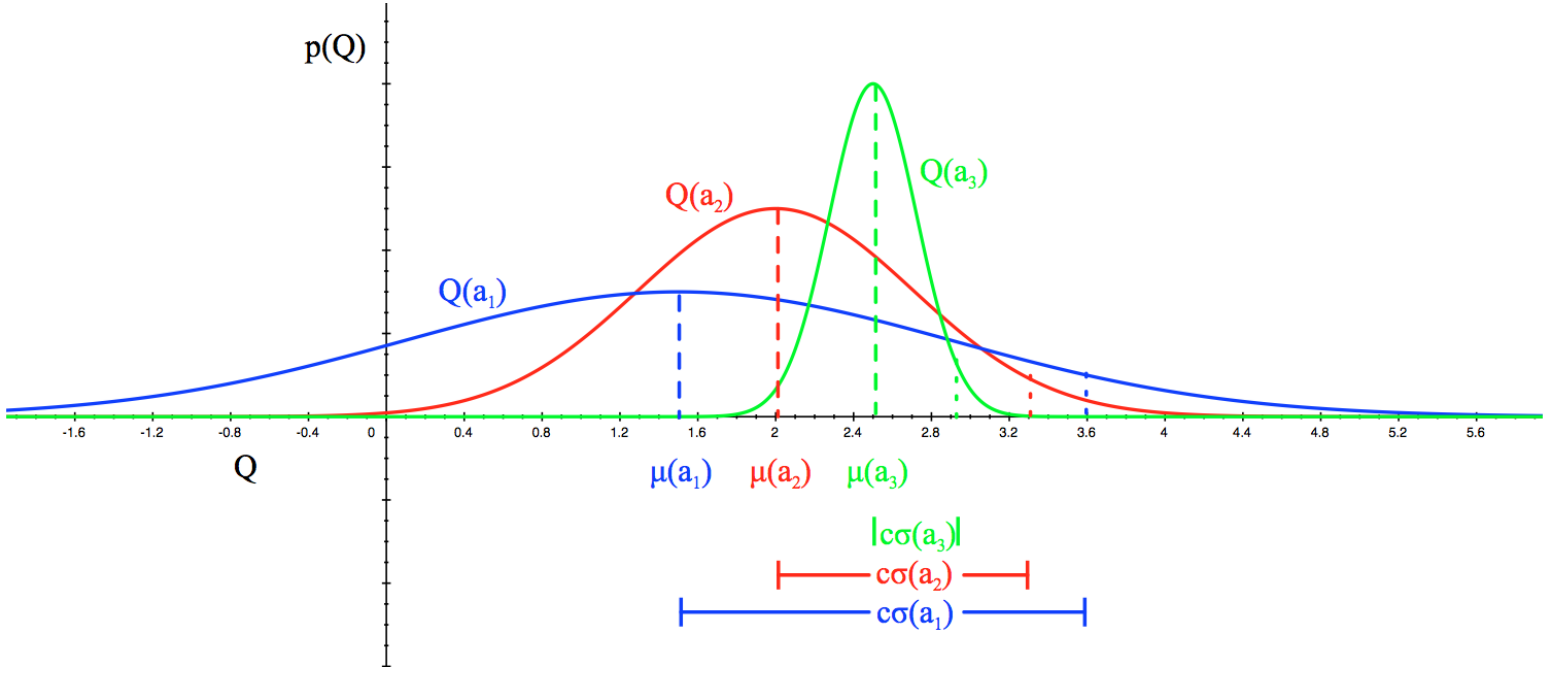
多臂老虎机

- “利用”：玩已经发现的赢率最高的机器
- “探索”：玩那些还没有充分探索的机器
- 关键：平衡好“利用”和“探索”的关系



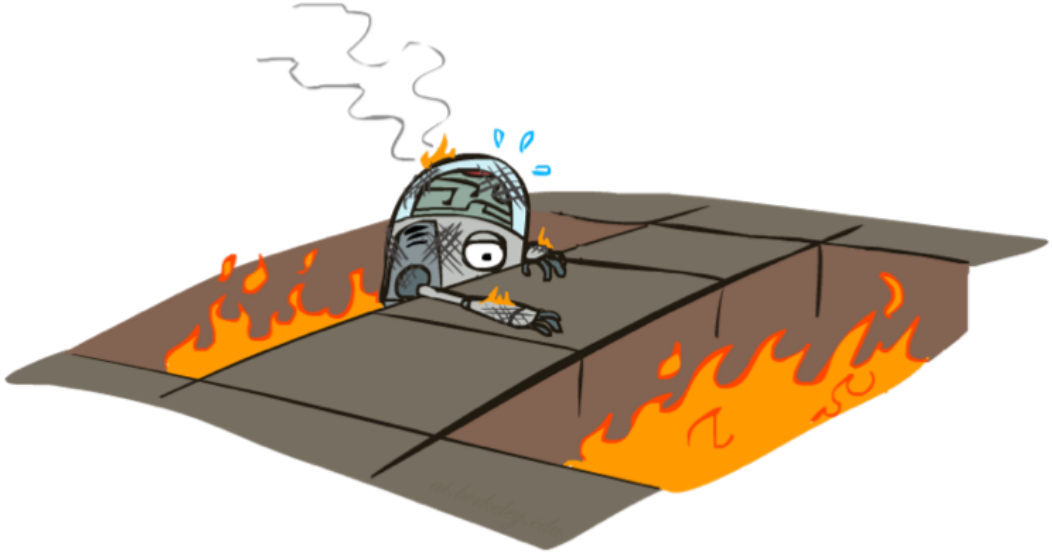
UCB算法

- Upper Confidence Bounds：置信区间上界方法
- 包括了平均赢率（均值）和探索空间（标准差）
- 综合“利用”和“探索”两种信息



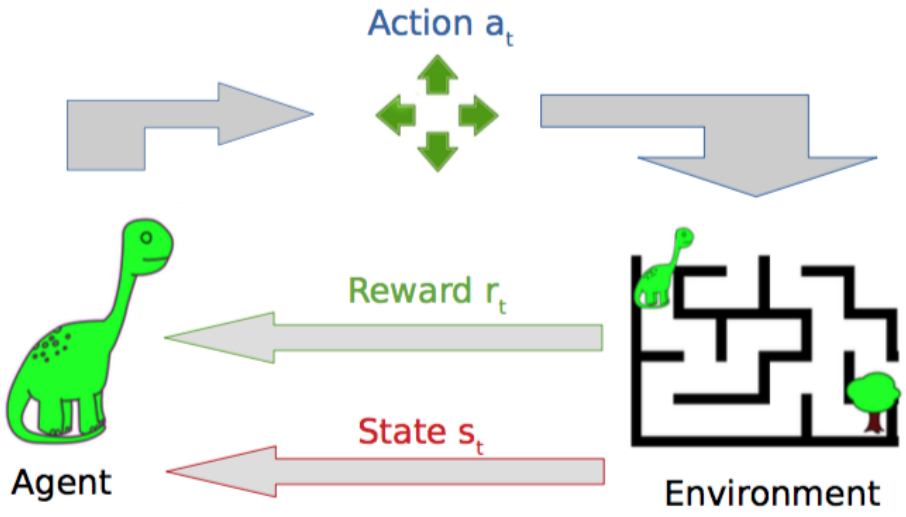
增强学习

- 进行大量尝试
- 跳进火坑也不怕



增强学习

- 不断尝试
- 得到每一个位置的“价值”
- 或者每一个位置下的最佳动作

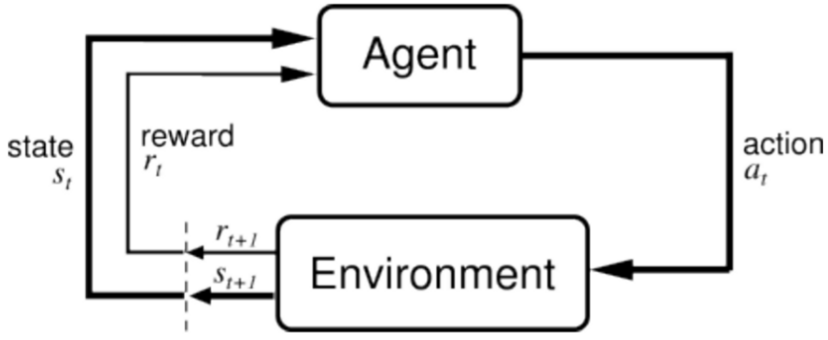


挑战

- 收益会有延时：期末才知道成绩
- 收益反馈稀疏：学了一个学期，才有一个期末考试

An MDP is defined by:

- Set of states S
- Set of actions A
- Transition function $P(s' \mid s, a)$
- Reward function $R(s, a, s')$
- Start state s_0
- Discount factor γ
- Horizon H



小结：学习类型

1. 有监督学习
    - 已知正确答案（标签）
2. 无监督学习
    - 从纯数据中发现规律
3. 半监督学习
    - 利用大量没有打标的数据
4. 增强学习
    - 通过尝试进行学习

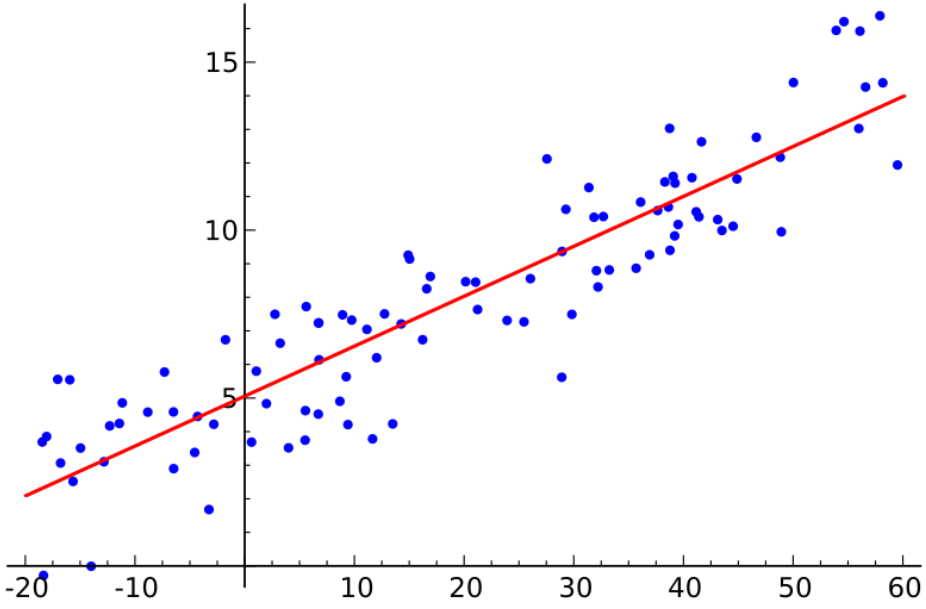
模型

Model

线性回归

Linear Regression

= +

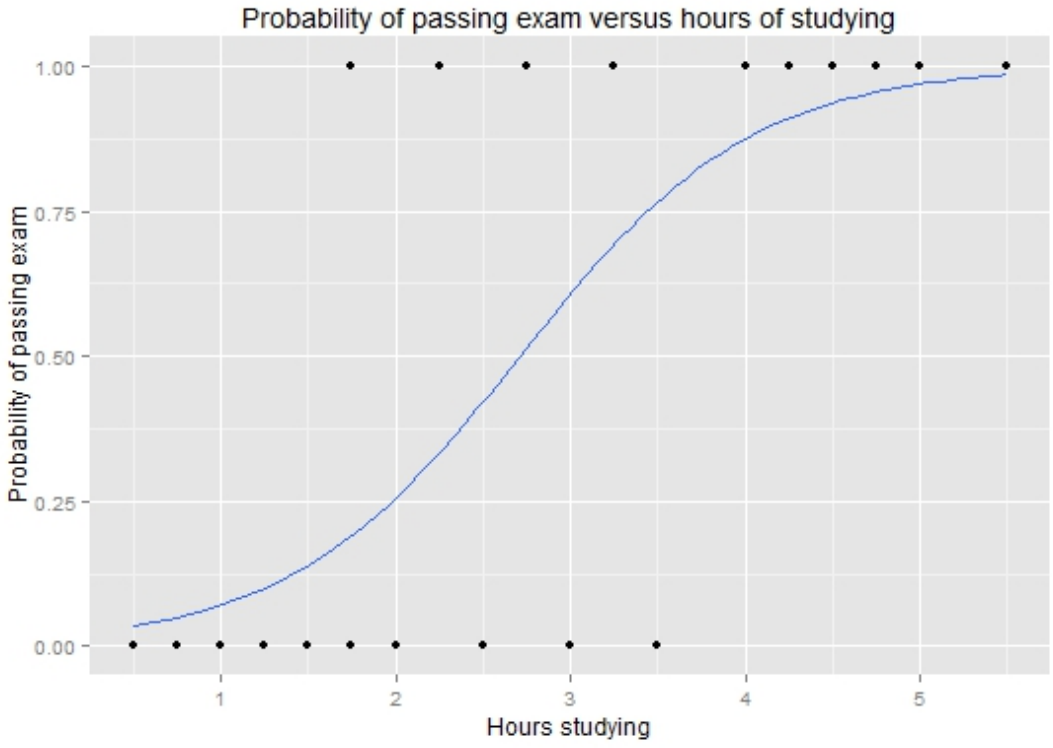


直线

Logistic回归

$$= \frac{1}{1 + \quad -(\quad + \ )}$$

考试通过与学习时间关系



$$(\qquad) = \frac{1}{1 + \quad -(1.5* \qquad -4)}$$

S曲线

感知机

模型人脑神经元

$$\vec{} * \vec{} \geq$$

神经元模型

- 神经元（脑细胞）通过突触连接
- 大脑会不断创建、强化、弱化这些连接

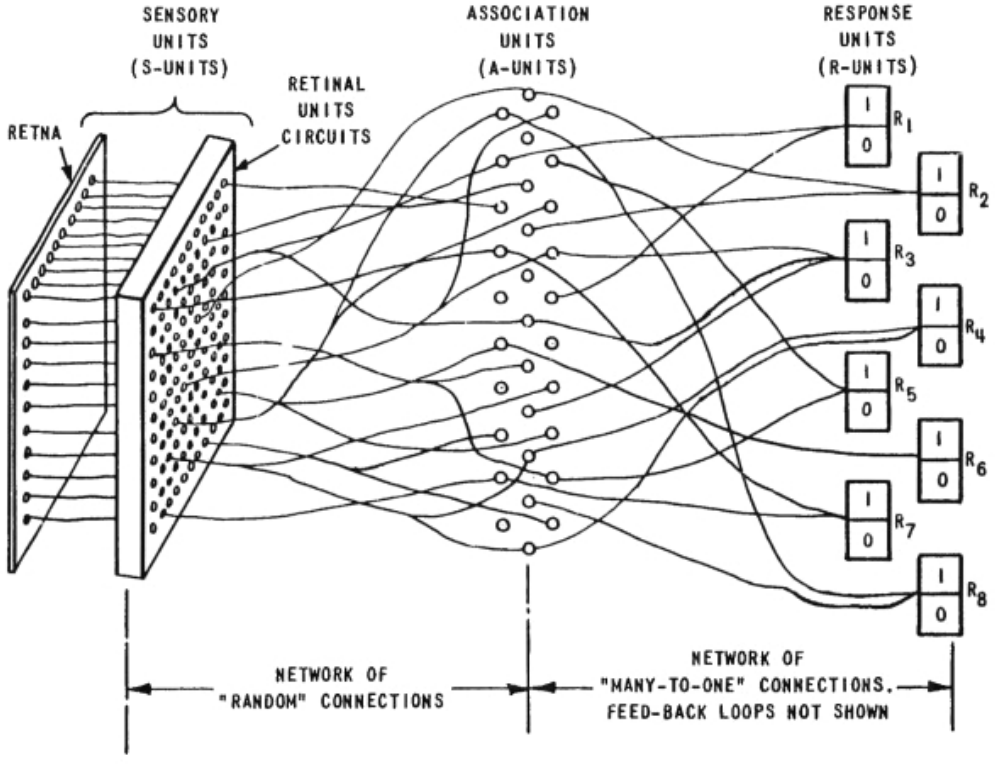
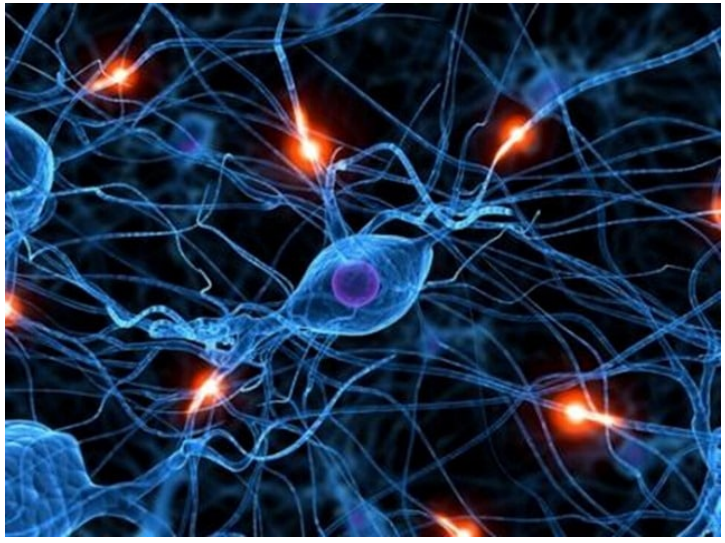


Figure 1 ORGANIZATION OF THE MARK 1 PERCEPTRON

神经元模型

输入的线性加权和

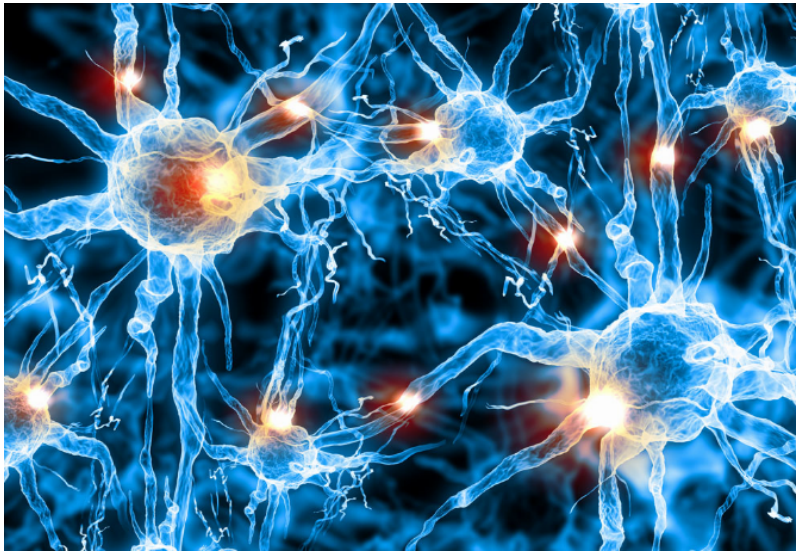
- ：神经元输入
- ：连接权重
- $1\ 1 + \ 2\ 2 +$ ：求和



神经元模型

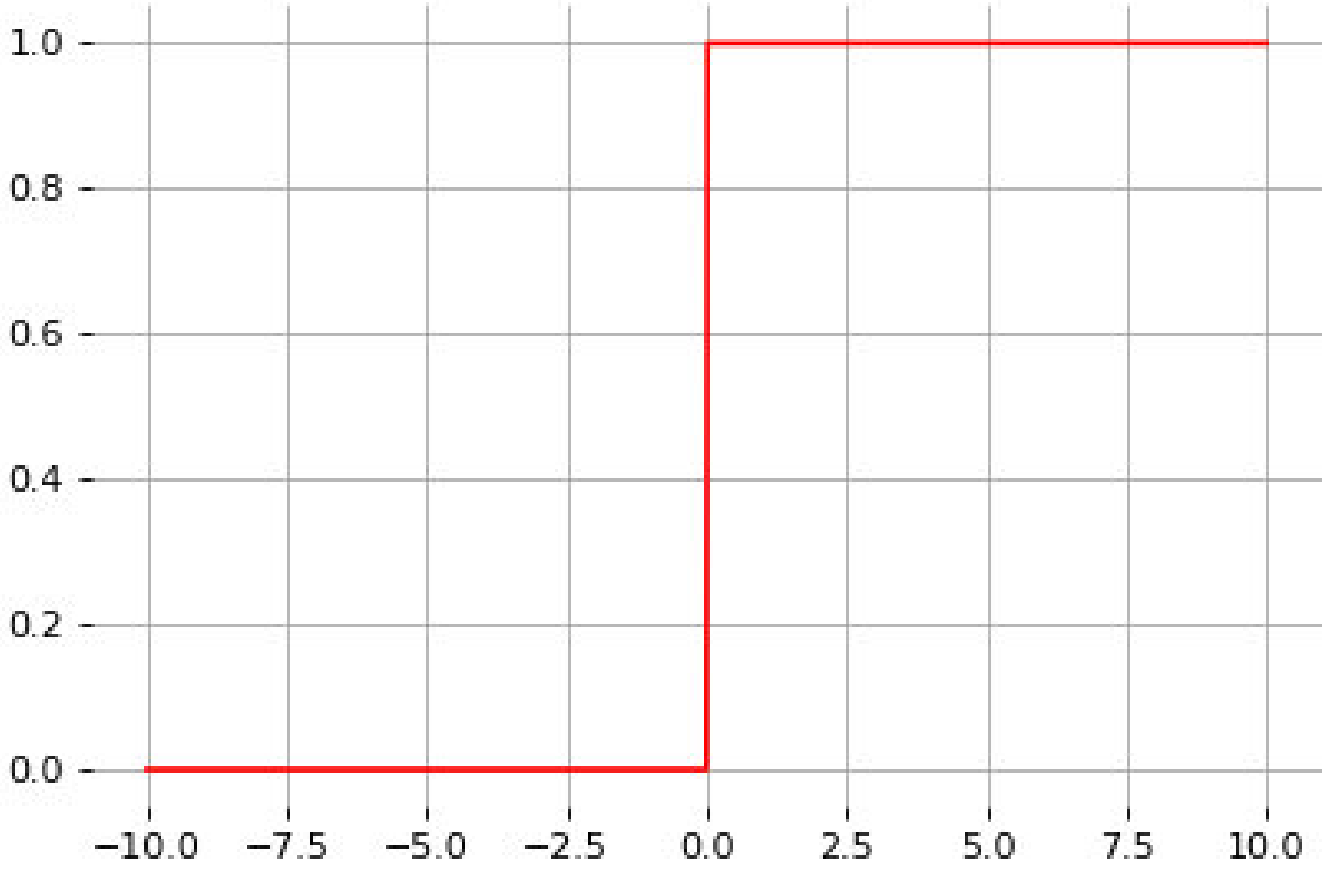
非线性激活函数

- $_1 \ _1 + \ _2 \ _2 + \ \geq 0$?



非线性激活函数 $(\ ) : \ \geq 0$

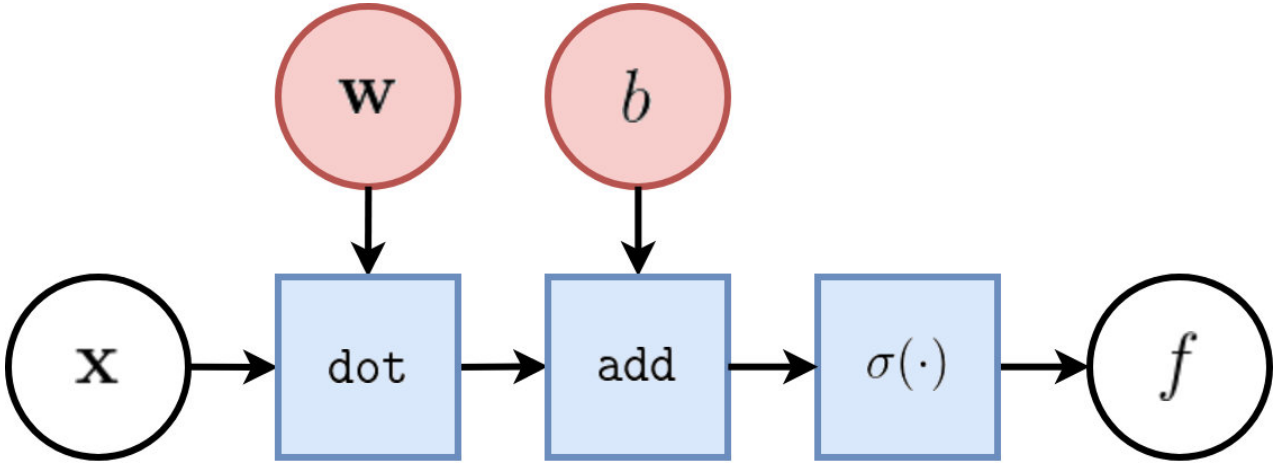
激活函数



非线性激活函数 $(\ )$：≥ 0

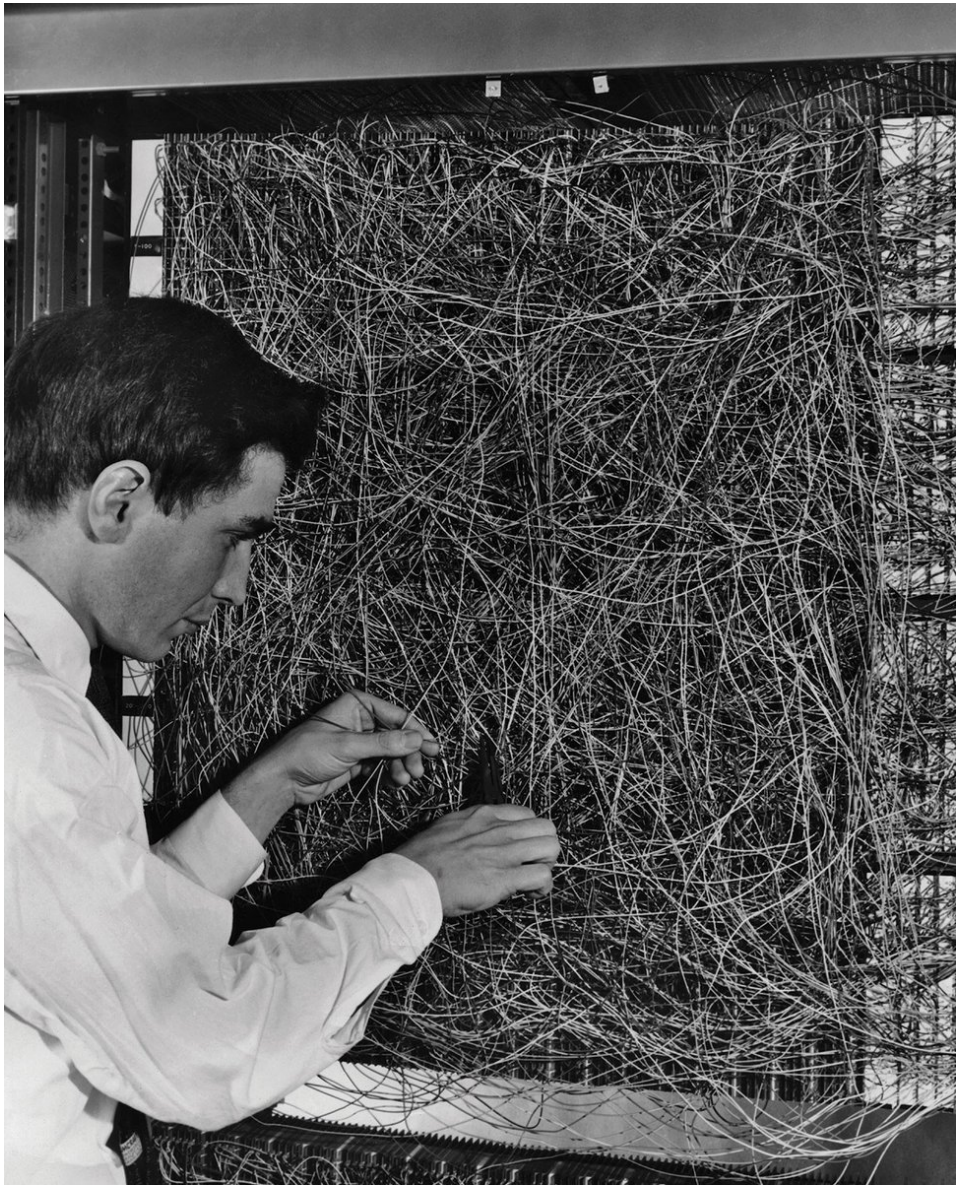
感知机

输入线性加权 + 非线性激活函数输出



$$= \left(\quad 1 \quad 1 + \quad 2 \quad 2 + \quad \right)$$

感知机



模型训练方法

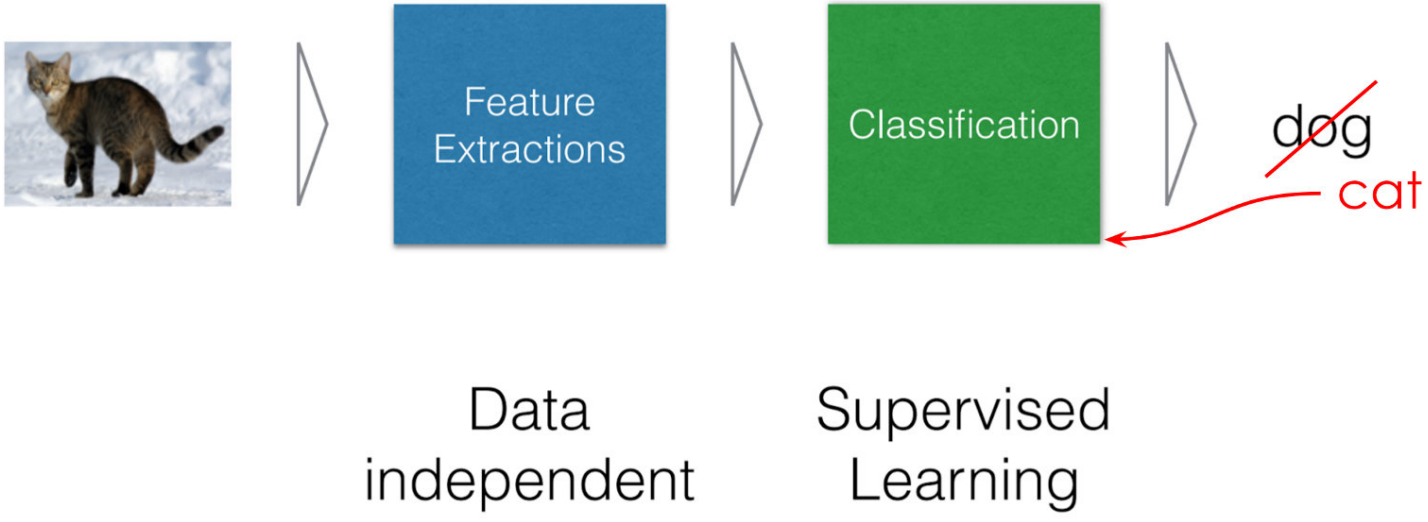
在错误中学习

大脑学习过程

- 根据实验结果，不断创建、强化、弱化神经元之间连接
- 也就是调整连接的权重：

机器学习的学习过程

- 出现错误，调整模型参数



感知机的学习过程

- 发现错误，调整权重 ，使错误减少

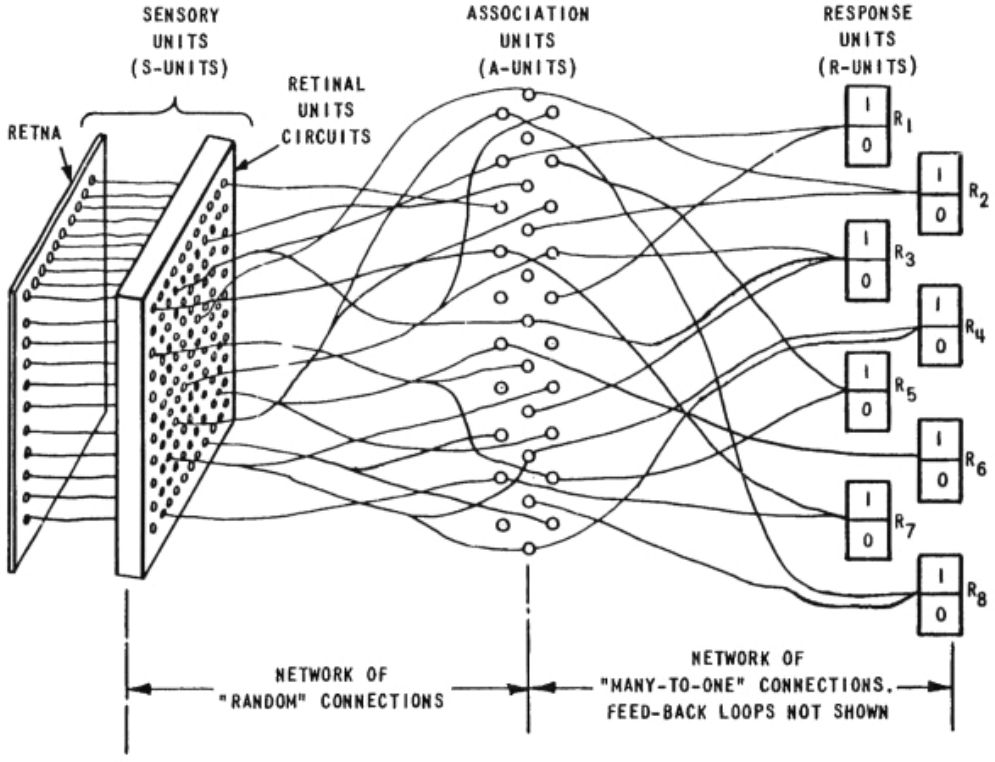
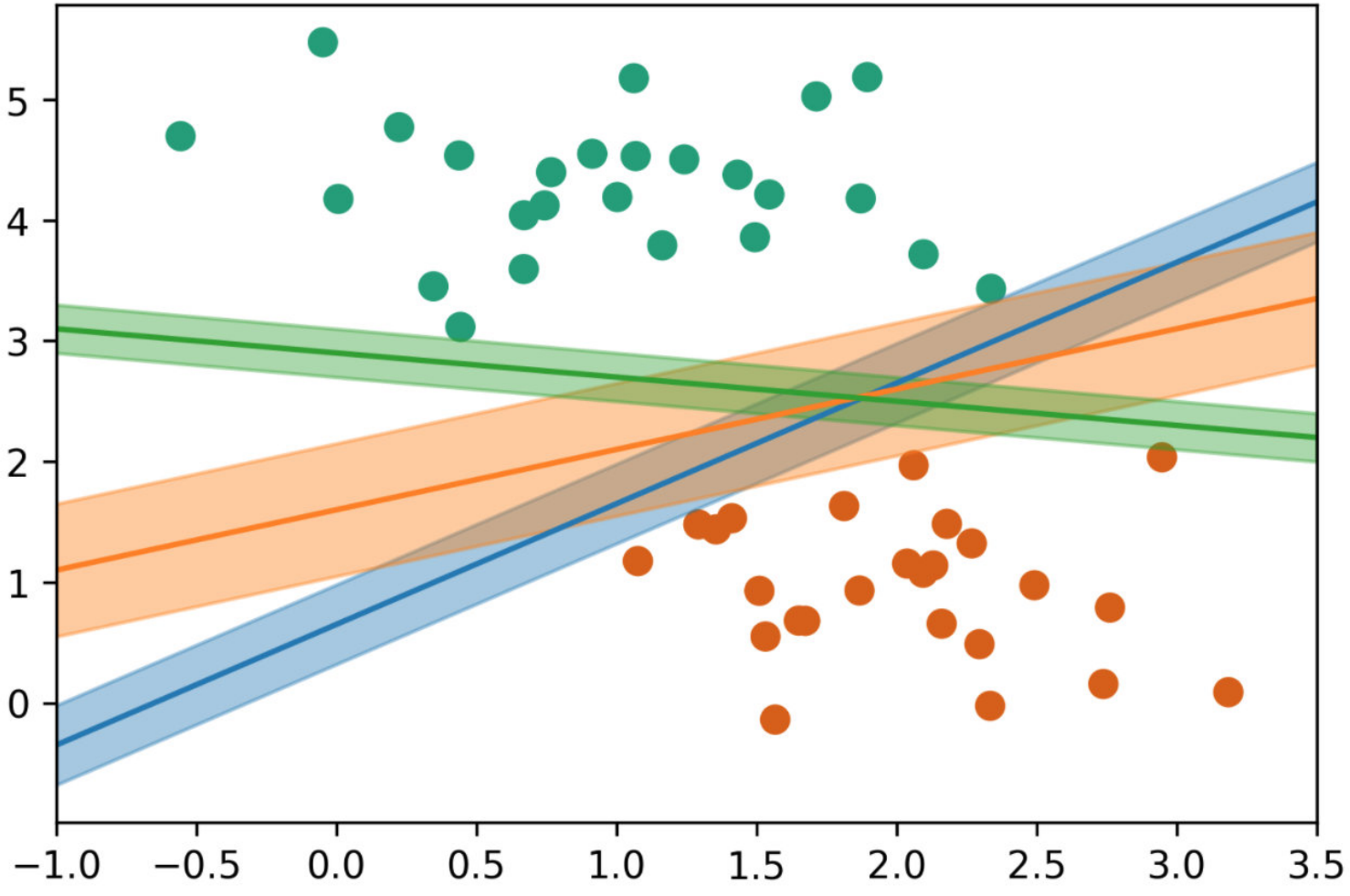


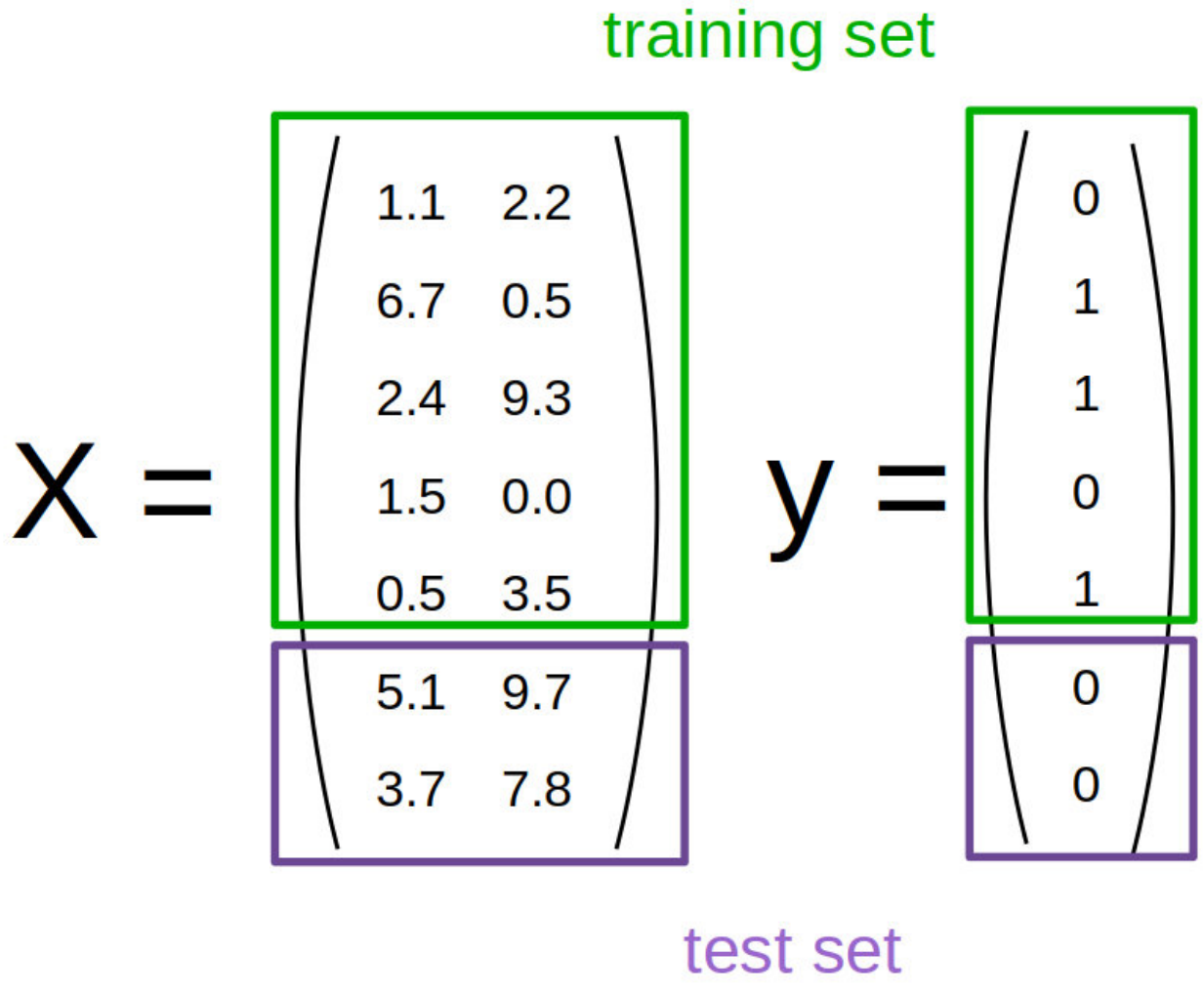
Figure 1 ORGANIZATION OF THE MARK 1 PERCEPTRON

感知机的学习过程

- 发现错误，调整　，调整决策边界

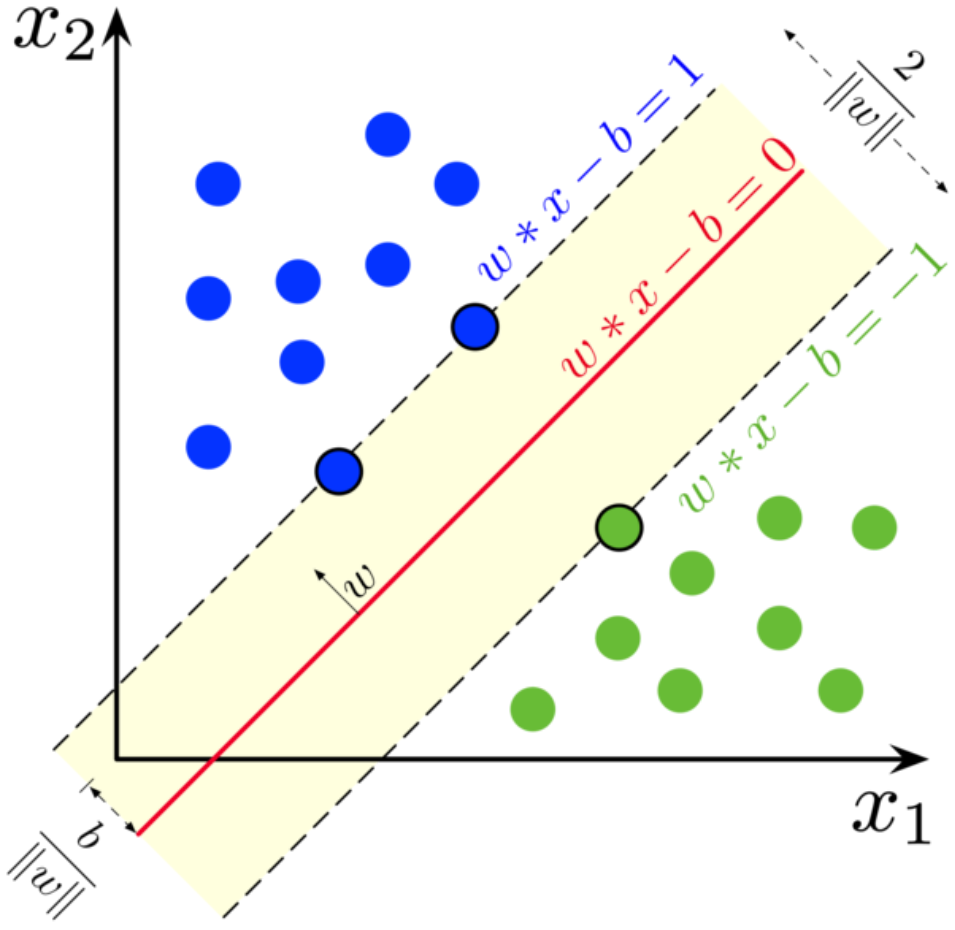


模型效果


training set
X =
1.1 2.2
6.7 0.5
2.4 9.3
1.5 0.0
0.5 3.5
5.1 9.7
3.7 7.8
y =
0
1
1
0
1
0
0
test set

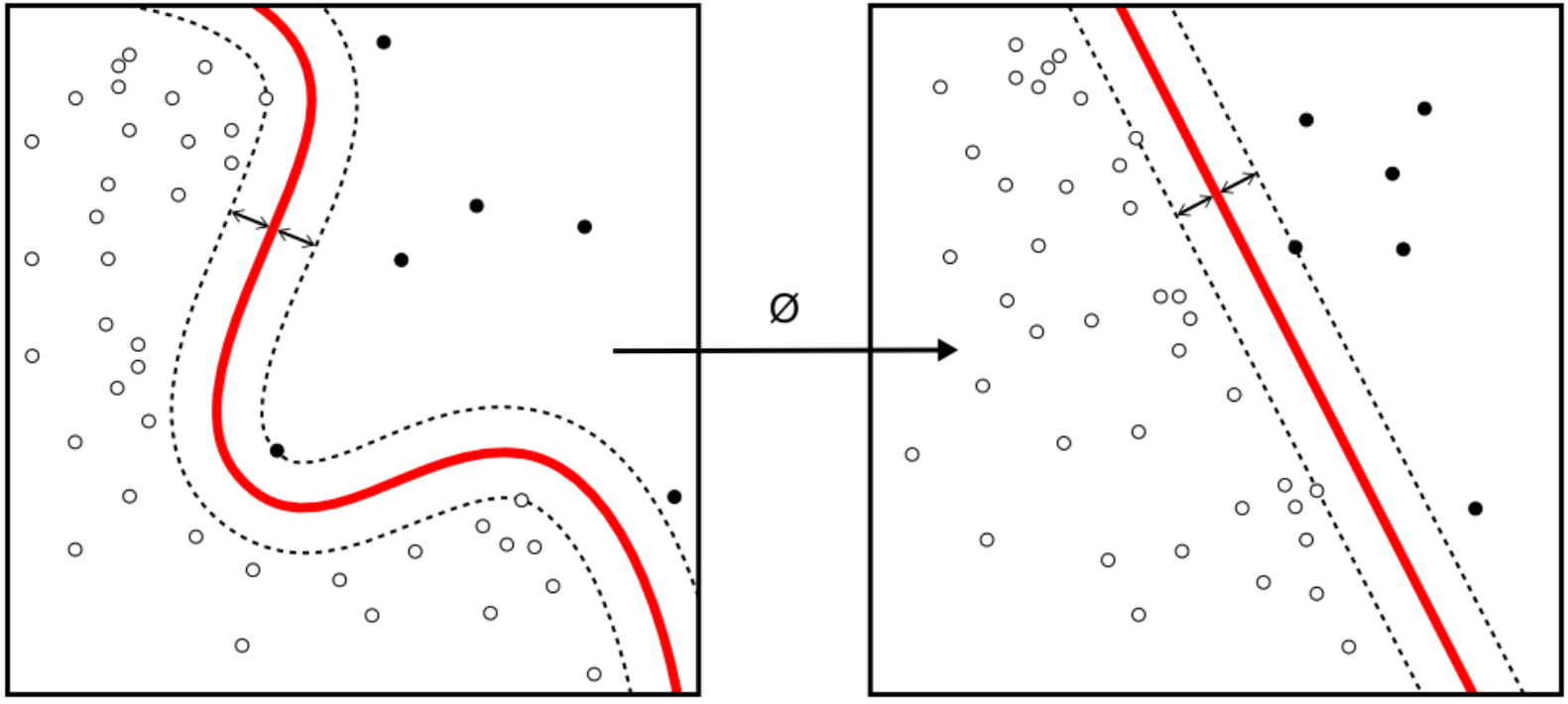
SVM：支持向量机

不仅避免错误，而且两边距离越远越好



核函数：支持非线性边界

用非线性核函数代替向量点积，支持曲线边界

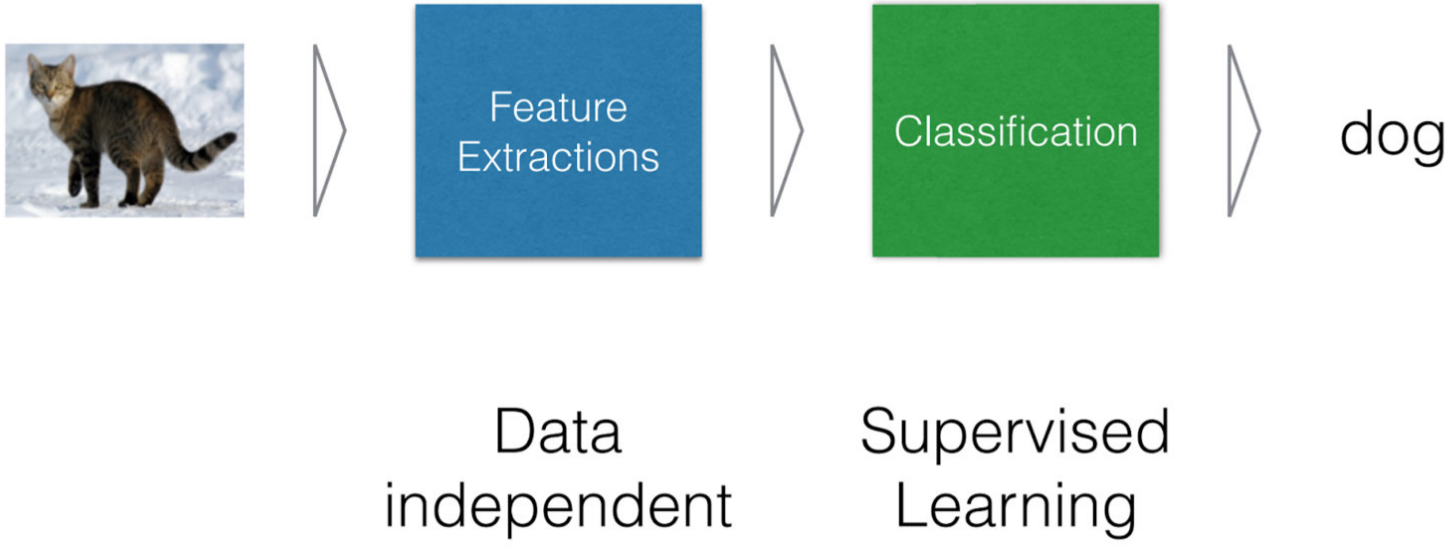


深度学习

Deep Learning

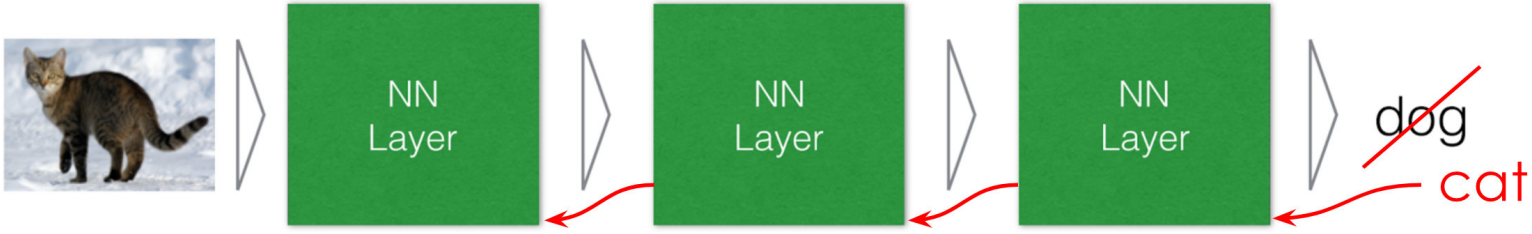
复习：机器学习

- 先提取图片特征
- 然后根据这些特征进行学习



深度学习

- 不专门提取数据特征
- 将原始数据直接送入多层神经元网络进行学习
- 出现错误，调整到底



Supervised
Learning

常用结构

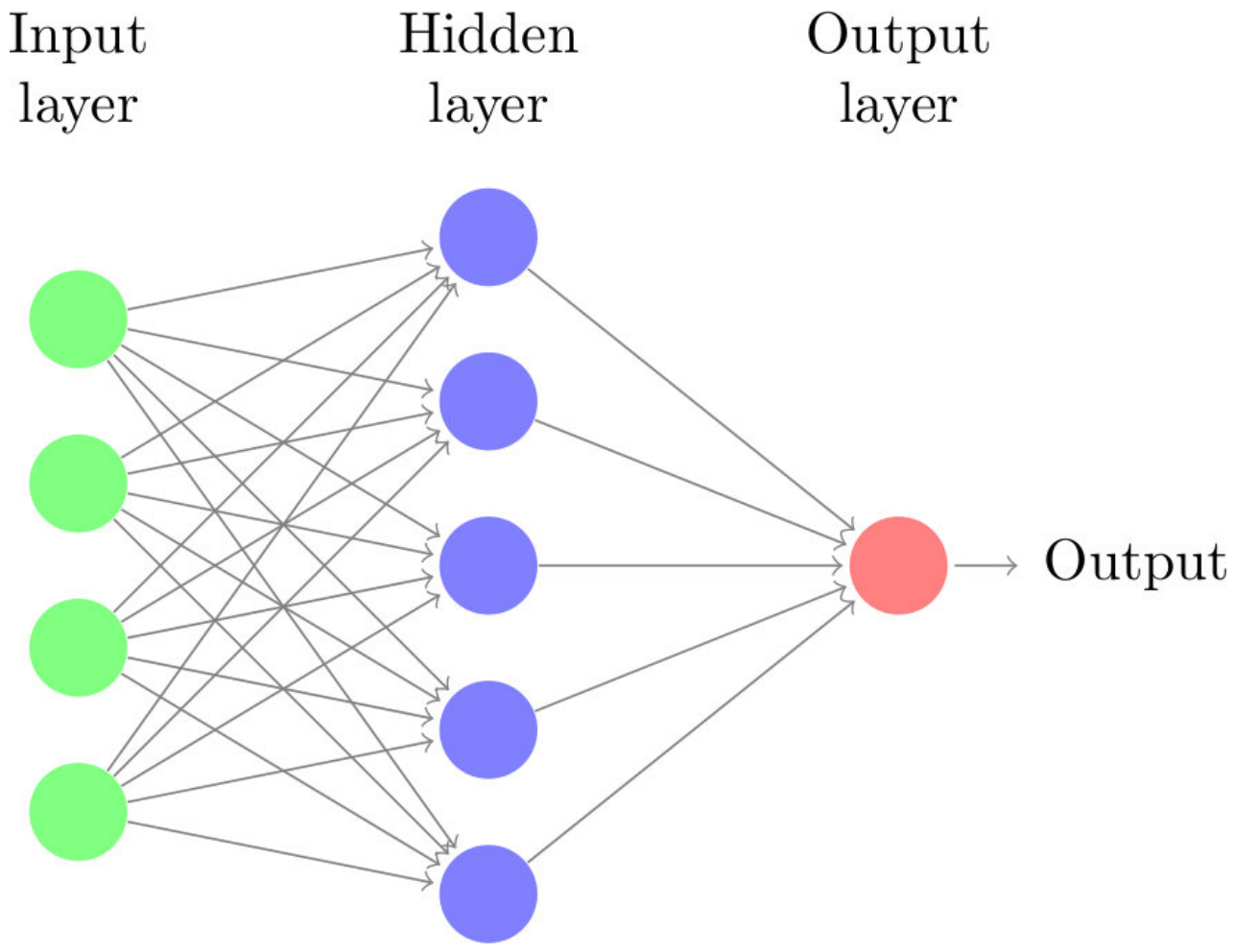
FFN、CNN、RNN

前向神经元网络

FFN：Feed Forward Network

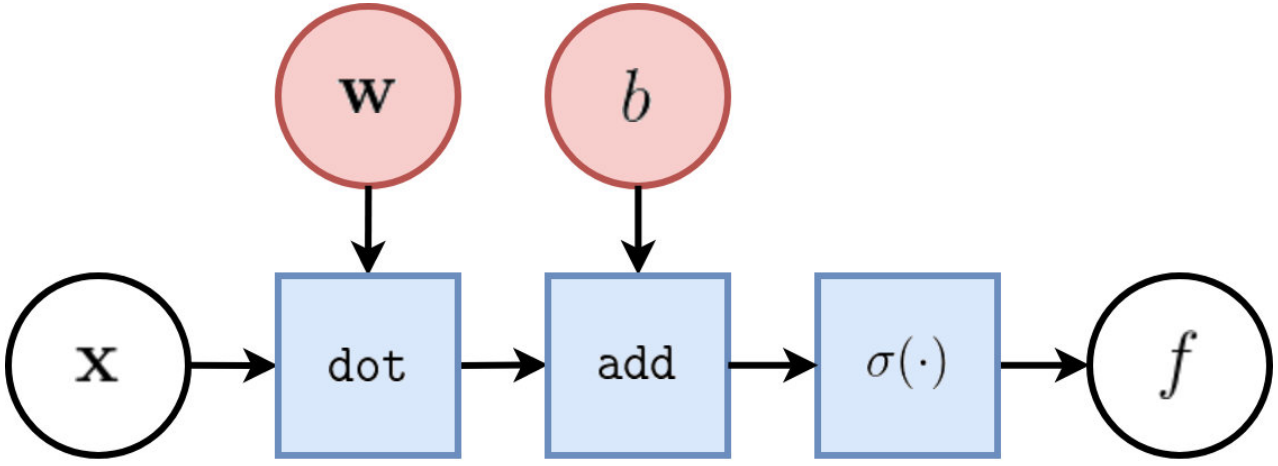
前向神经元网络

输入层，隐藏层，输出层



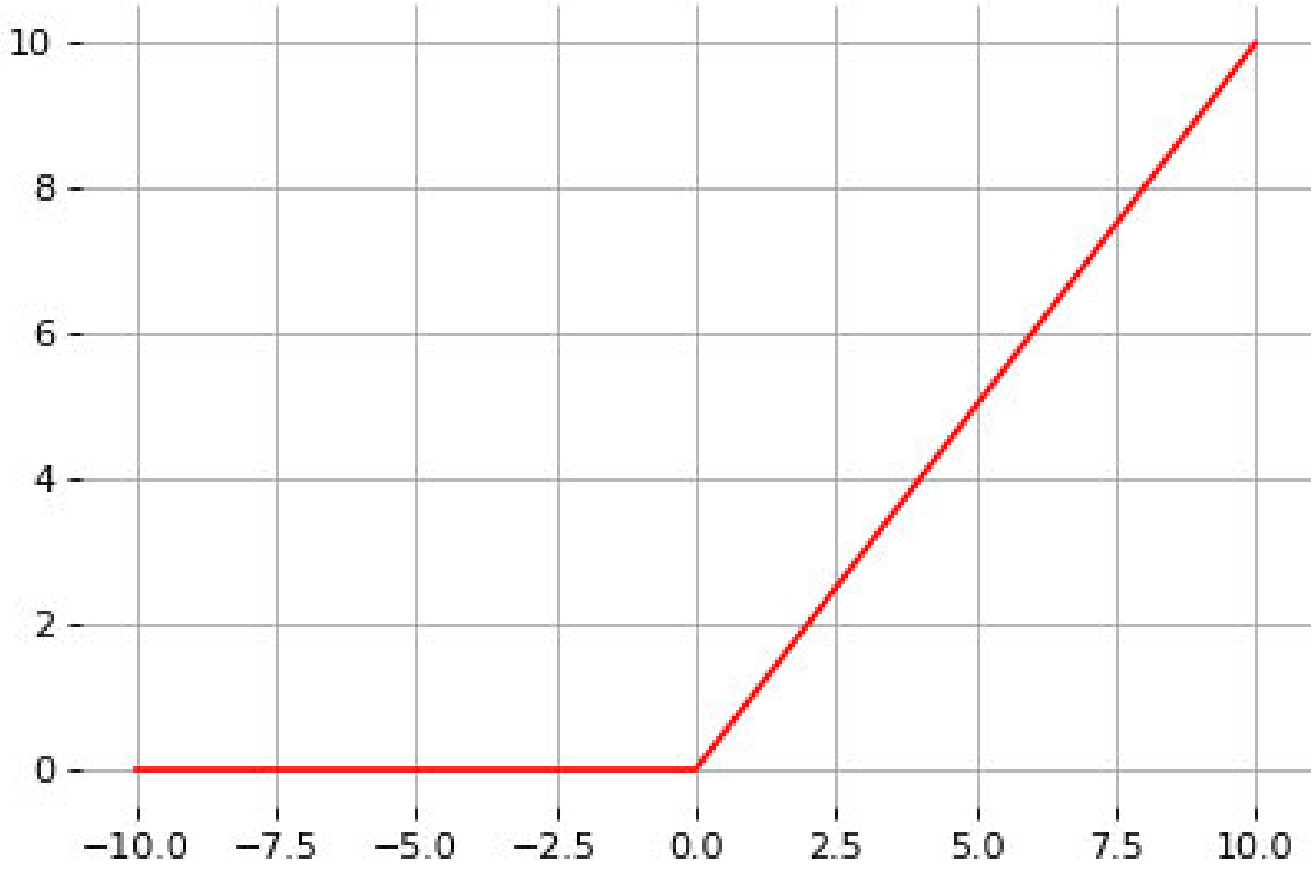
隐藏和输出层单元：感知机

输入线性加权 + 非线性激活函数输出



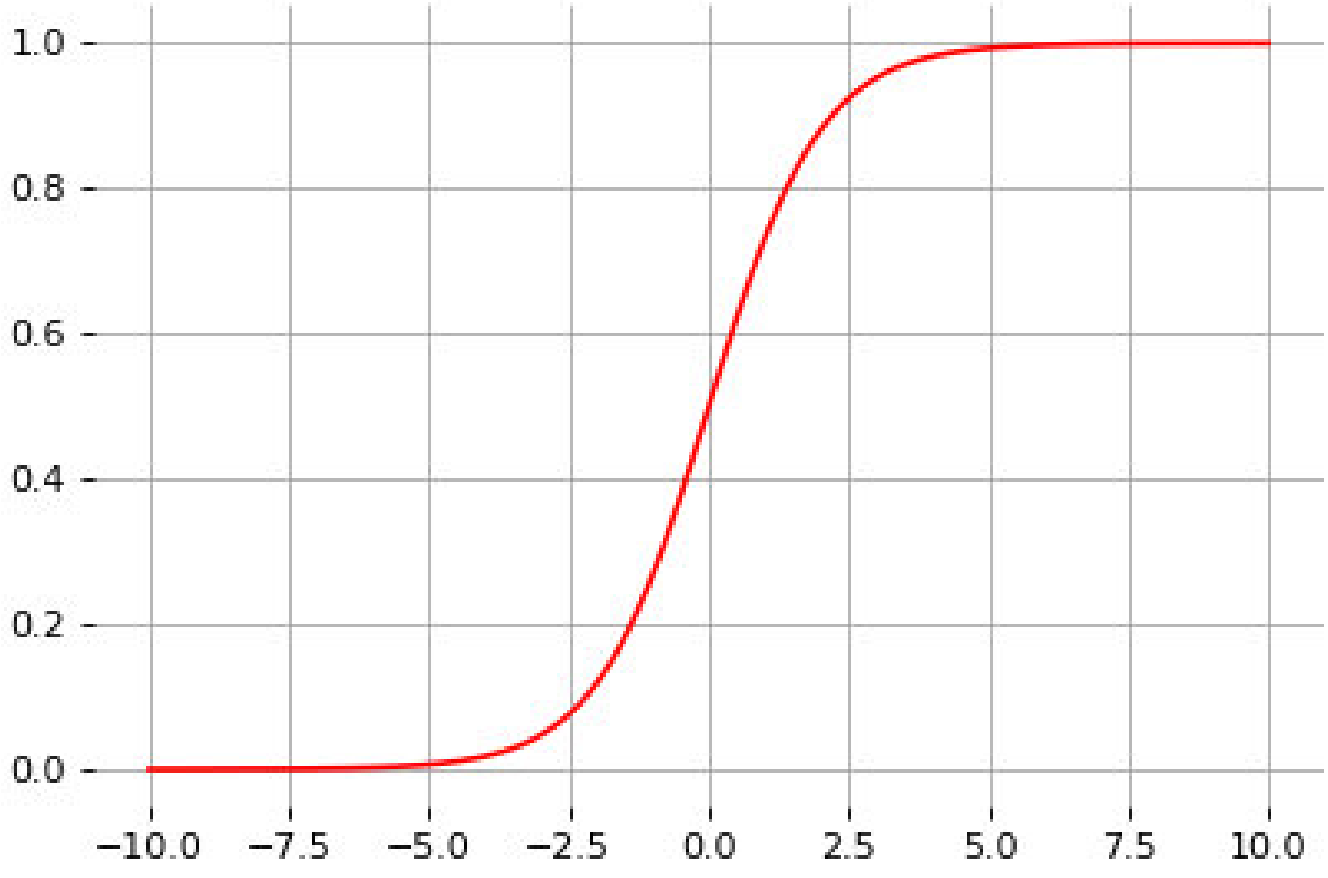
$$= \left(\quad 1 \quad 1 + \quad 2 \quad 2 + \quad \right)$$

常用激活函数：ReLU



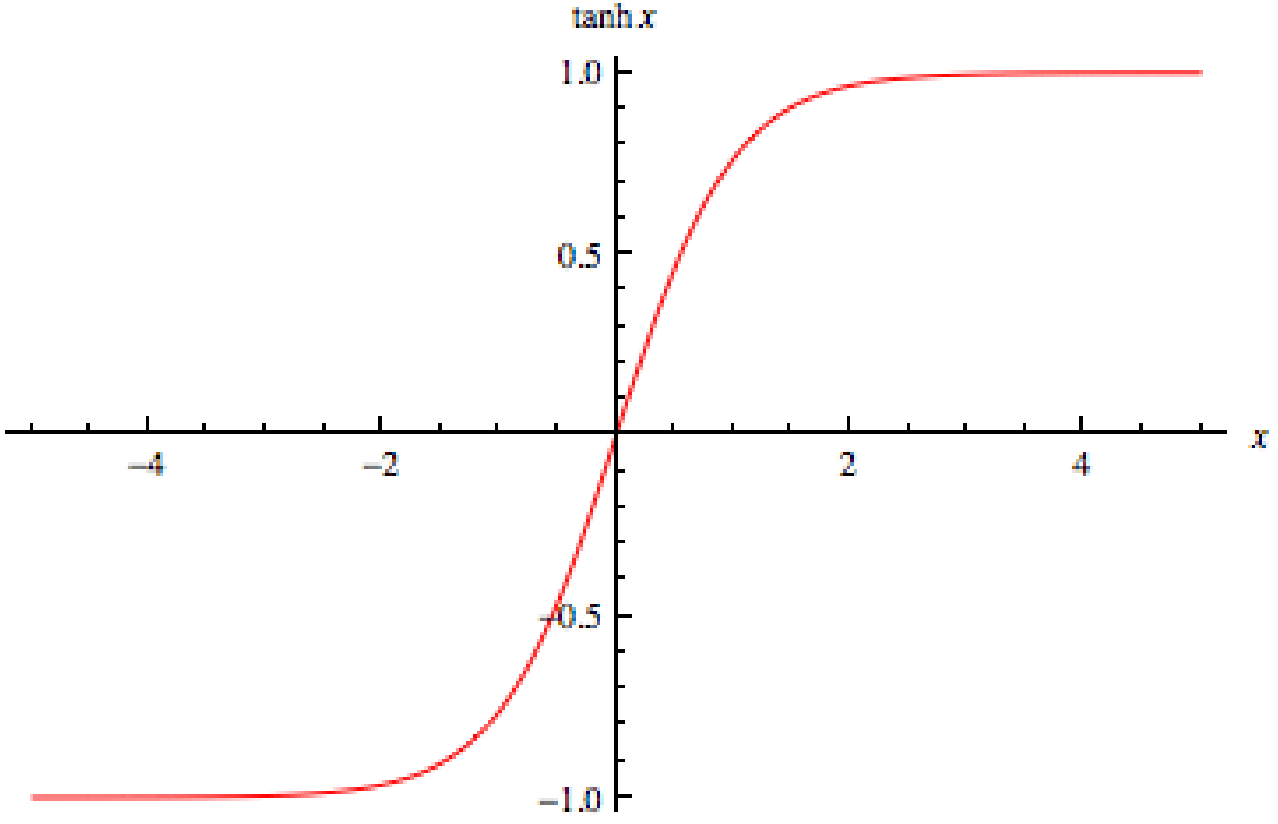
( ): Rectified Linear Unit

常用激活函数：Sigmoid



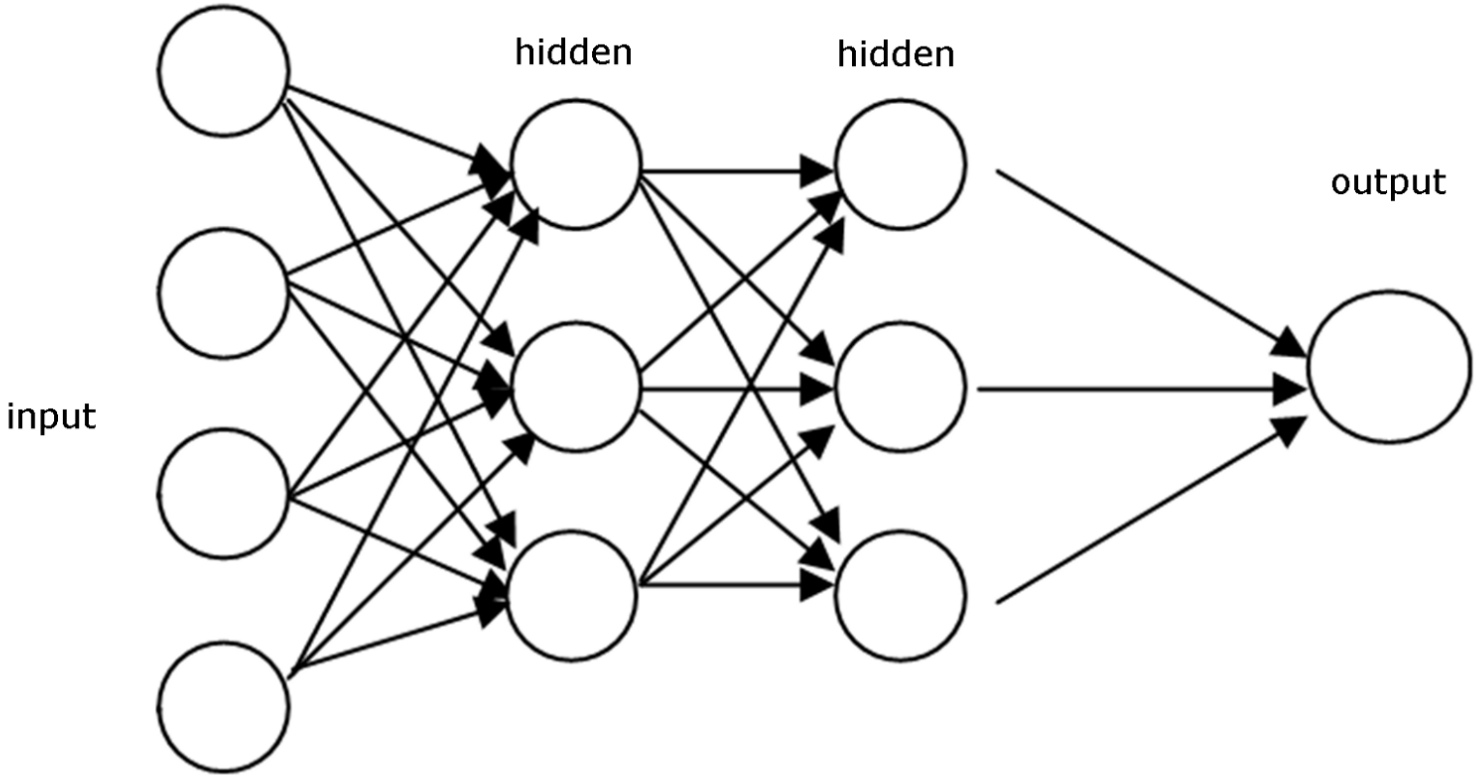
( ): S曲线

常用激活函数：Tanh



( )：Hyperboilic Tangent

深度神经元网络



多个隐藏层

深度的好处

越深，模型能力越强

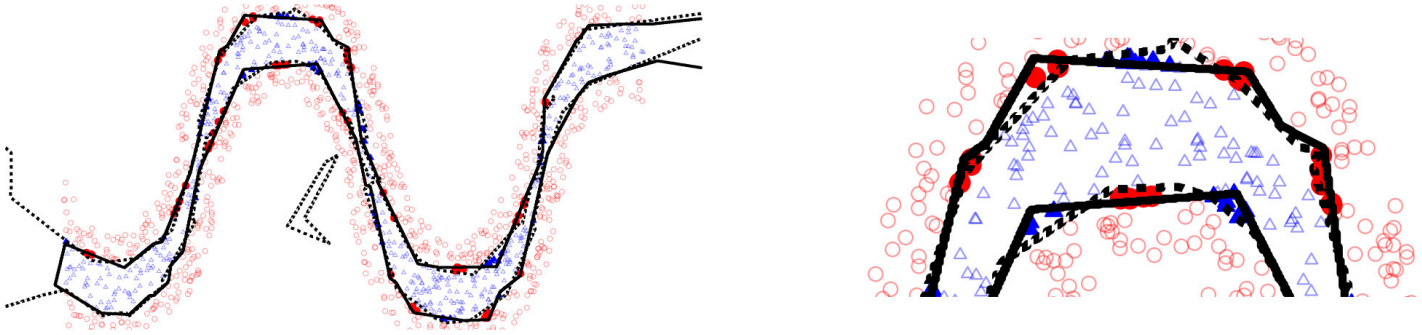
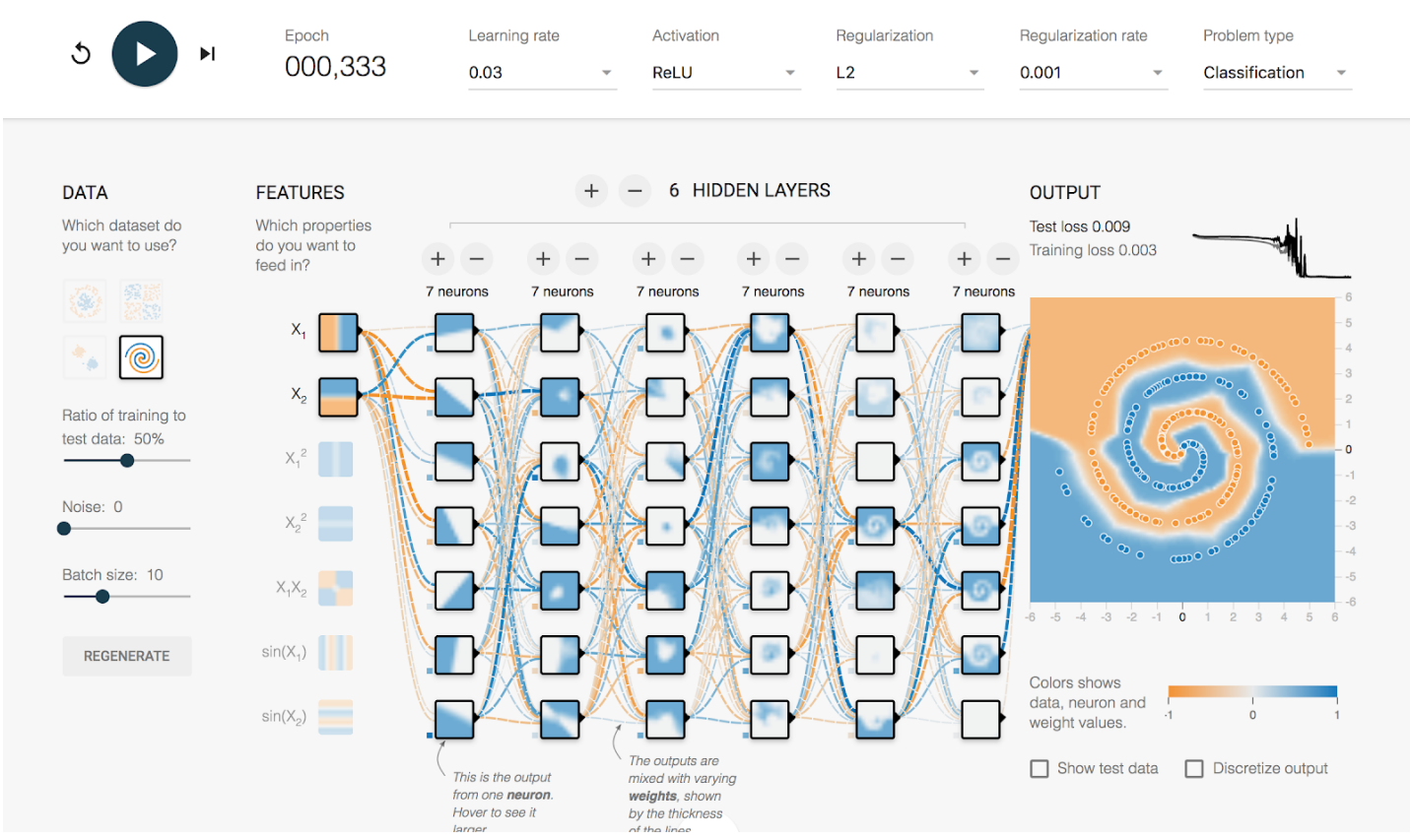


Figure 1: Binary classification using a shallow model with 20 hidden units (solid line) and a deep model with two layers of 10 units each (dashed line). The right panel shows a close-up of the left panel. Filled markers indicate errors made by the shallow model.

FNN练习

- 基于浏览器的TensorFlow实验平台
- http://playground.tensorflow.org

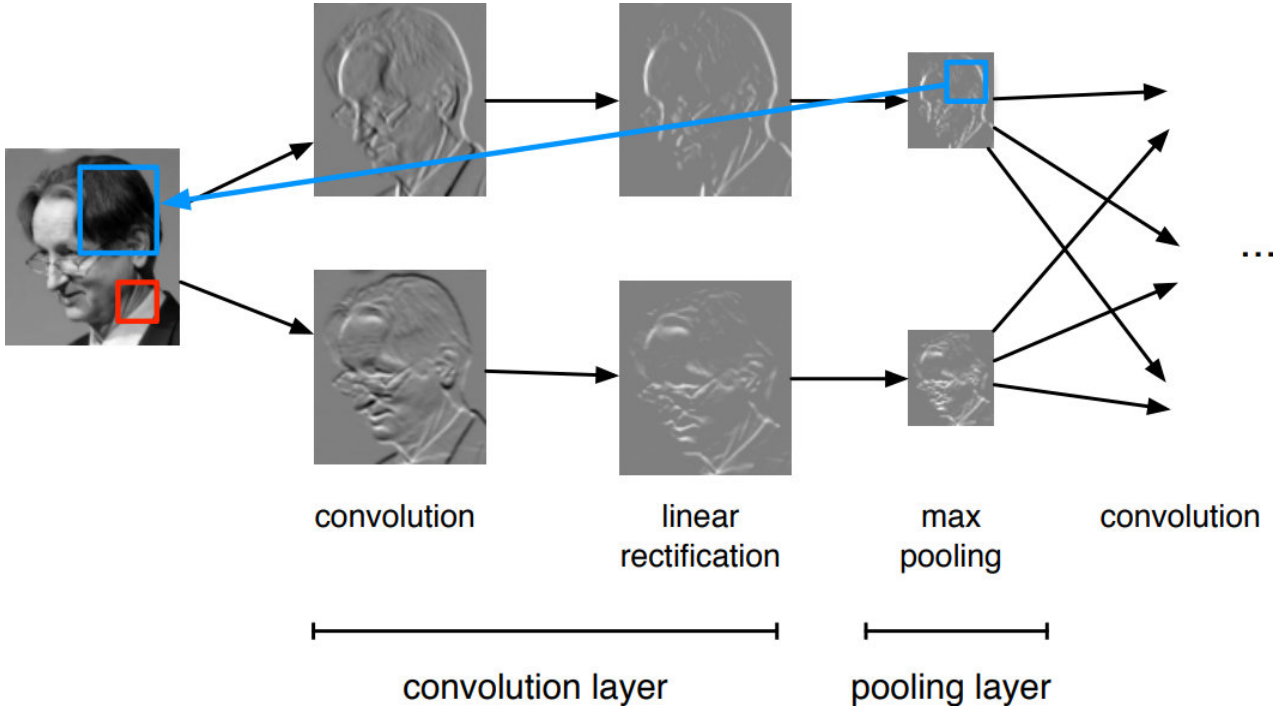


卷积神经元网络

CNN：Convolutional Neural Network

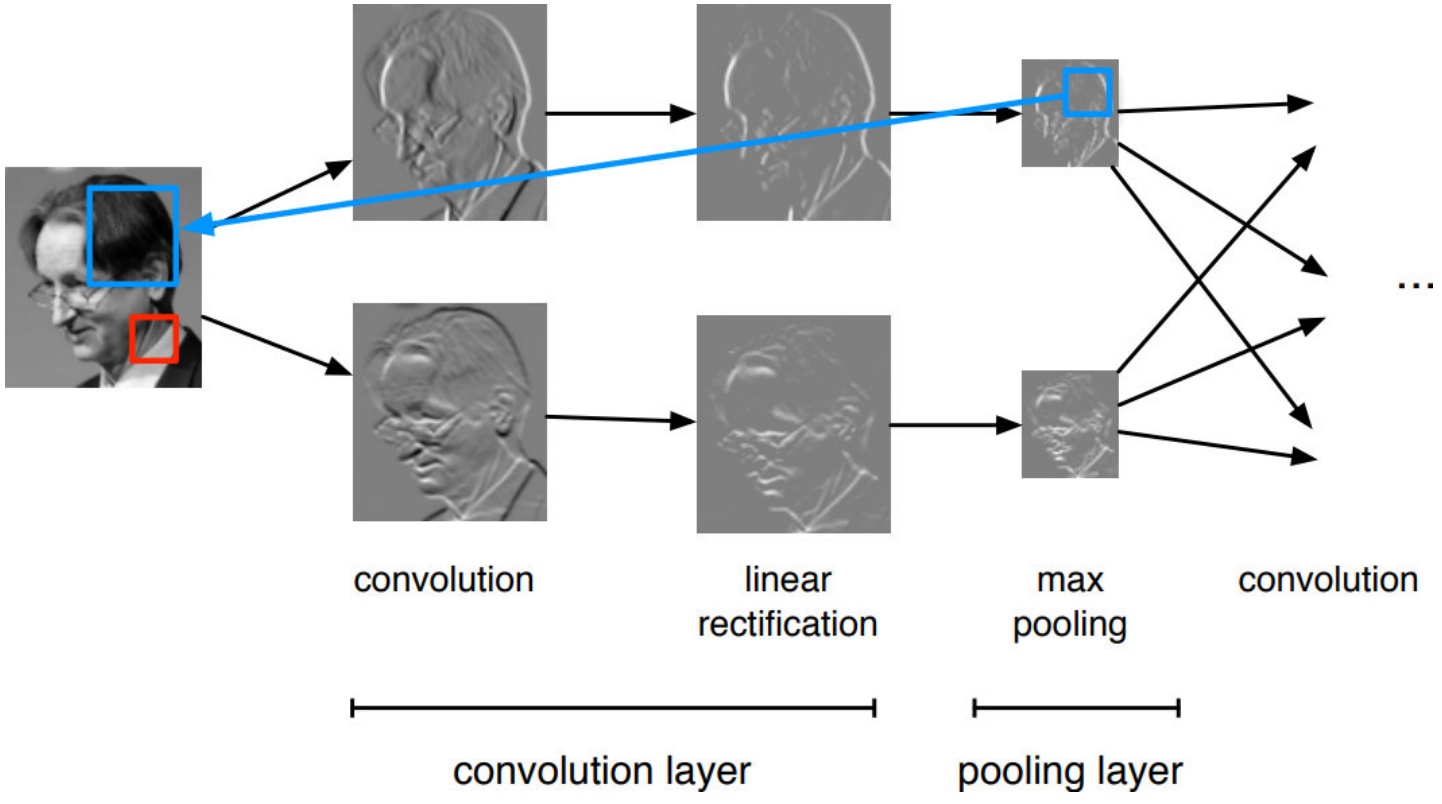
卷积神经元网络

- 一种特别的多层前向神经元网络
- 起源：手写体识别
- 常用于图像视觉应用、文本处理



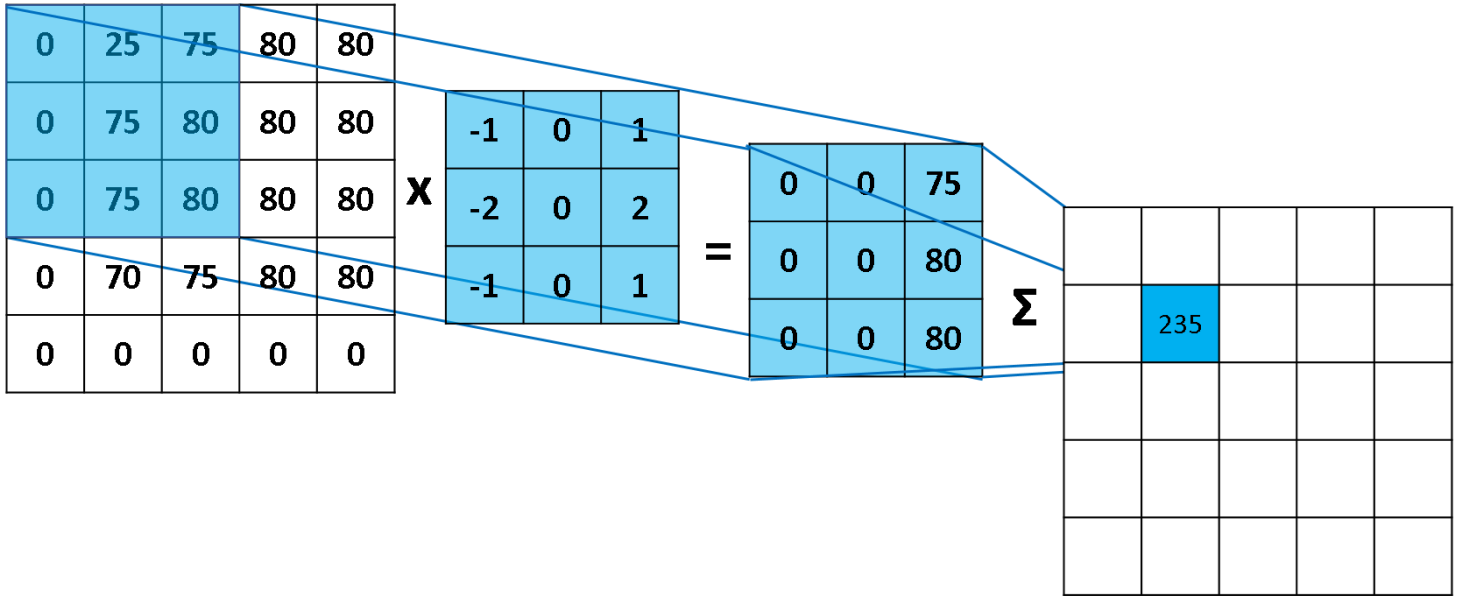
组成

- 卷积层
  - 卷积 + 非线性激活函数（如ReLU）
- 池化层



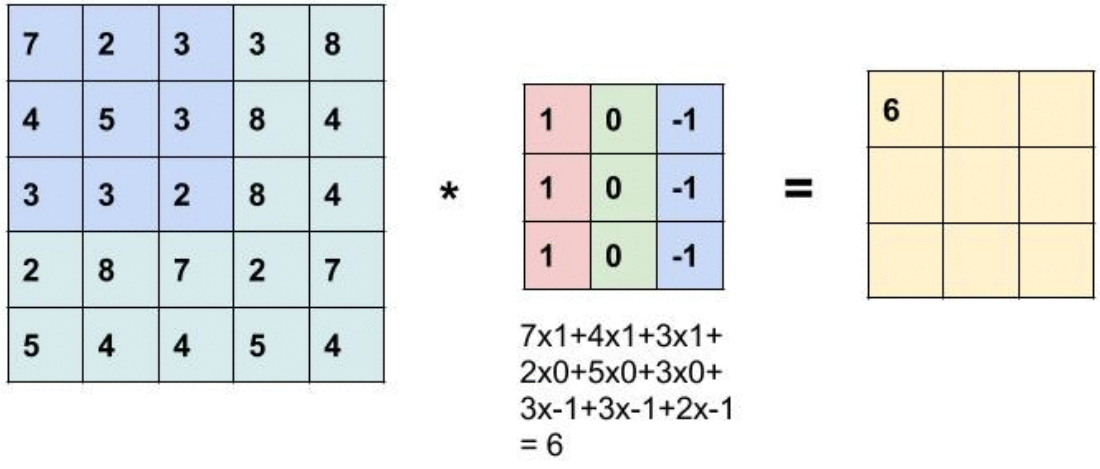
复习：卷积操作

二维卷积，对应位置相乘，然后相加



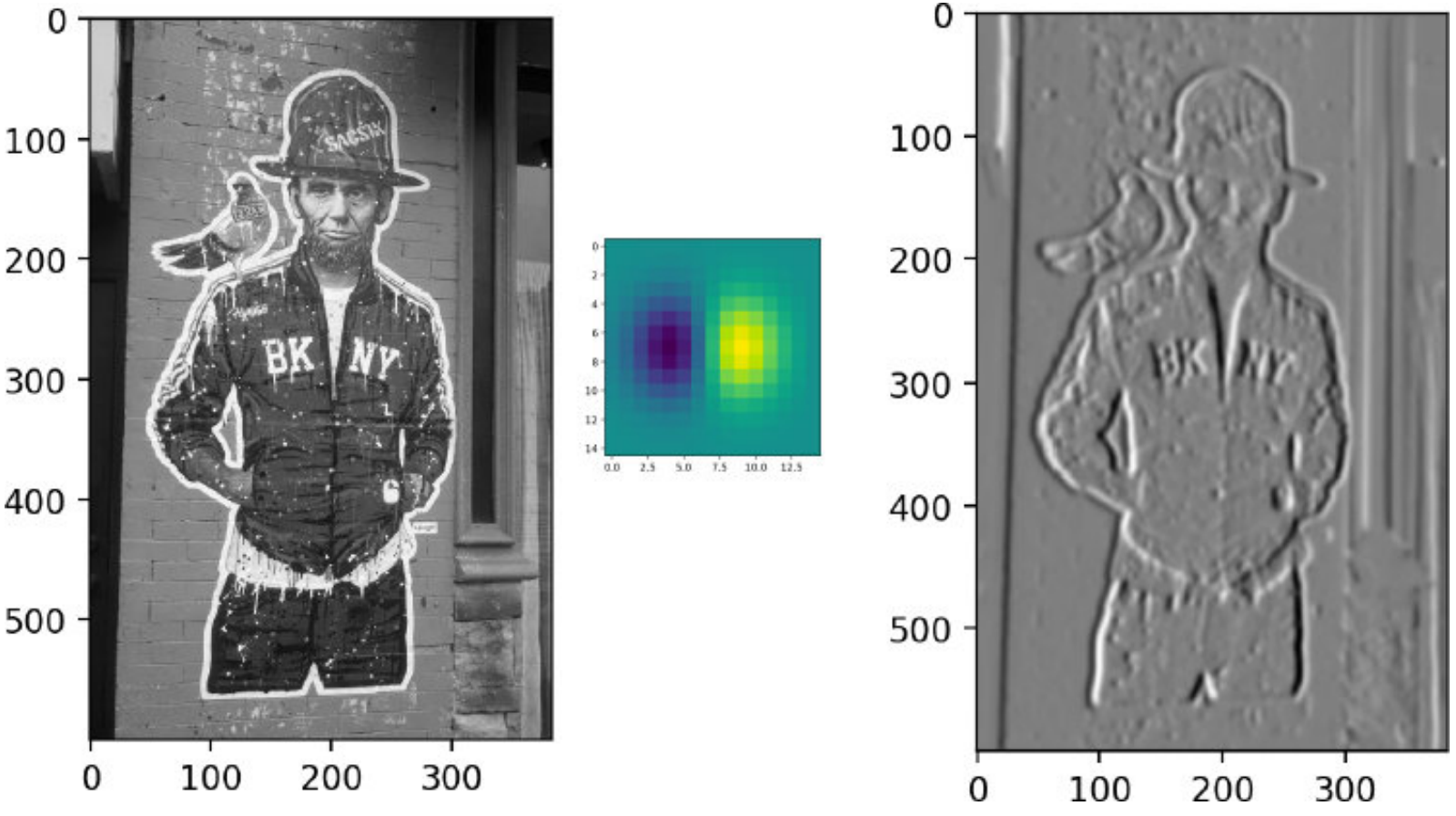
复习：图像卷积

滤波器在图片上滑动，进行卷积操作



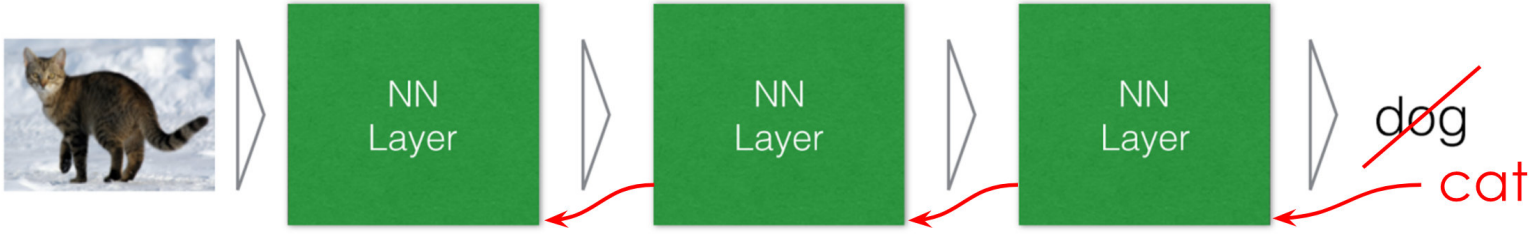
复习：图像卷积效果

选择合适卷积核（滤波器），卷积计算图像像素梯度



深度卷积神经元网络

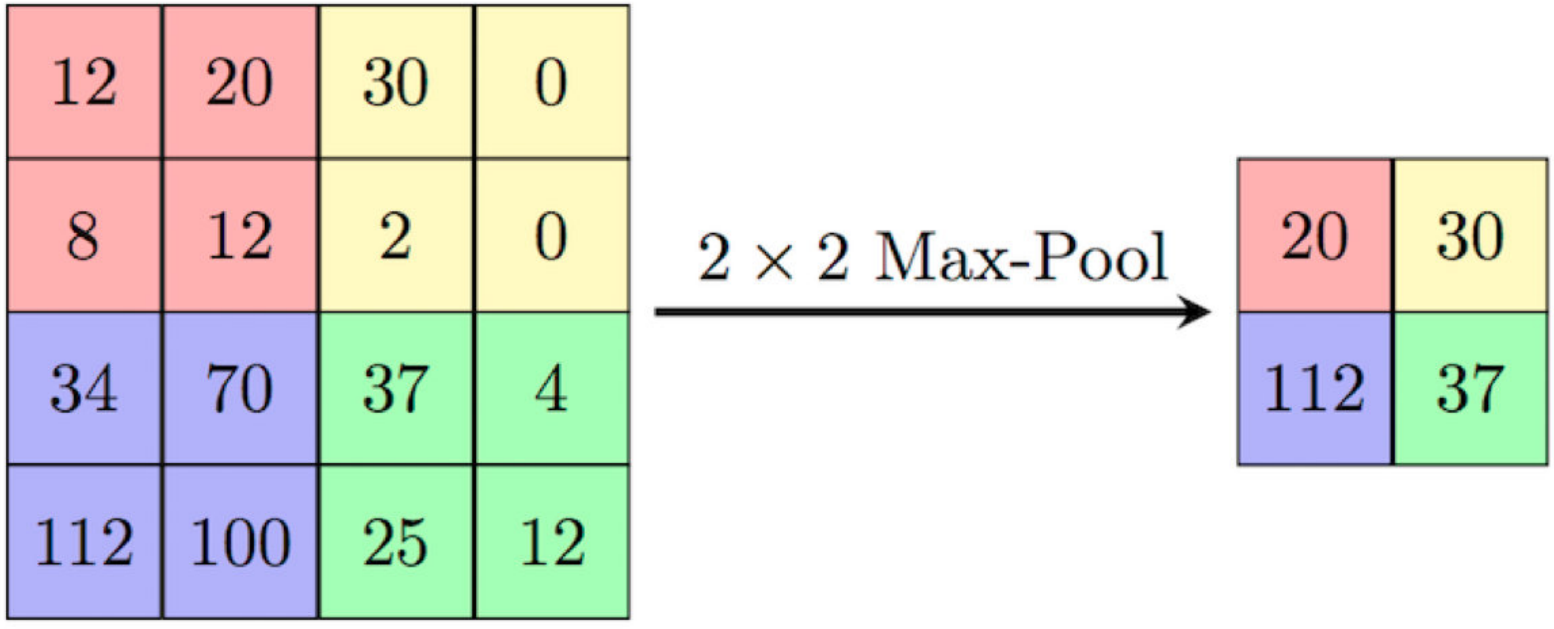
- 将原始数据直接送入多层神经元网络进行学习
- 多次卷积池化
- 出现错误，一路调整卷积核



Supervised
Learning

池化

采样降低数据量



最大池化：Max Pooling

LeNet

- 手写体识别
- 1988年，LeCun

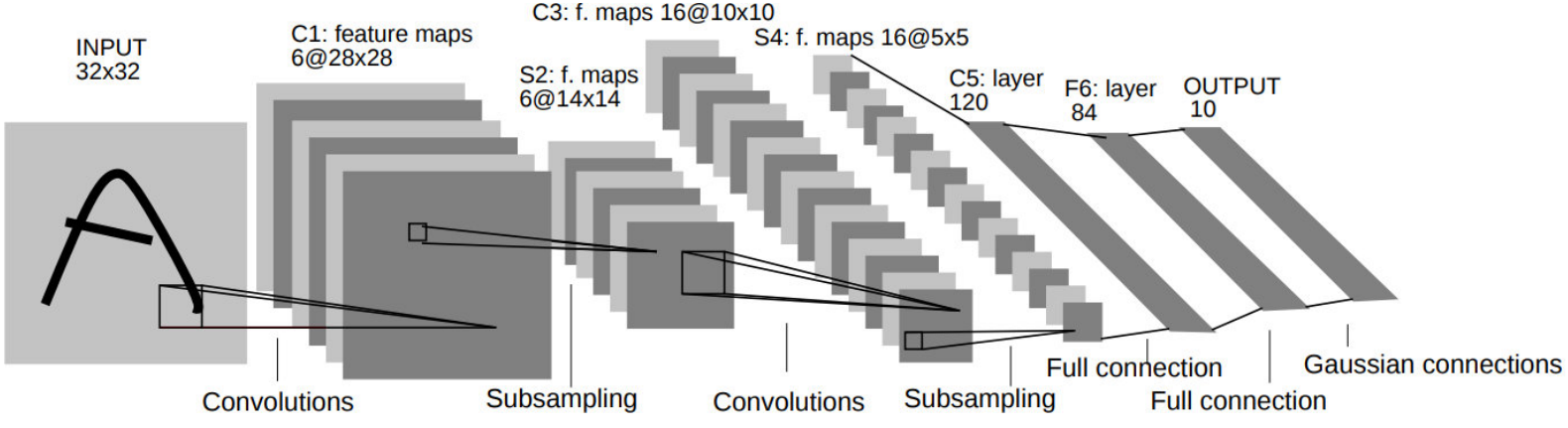
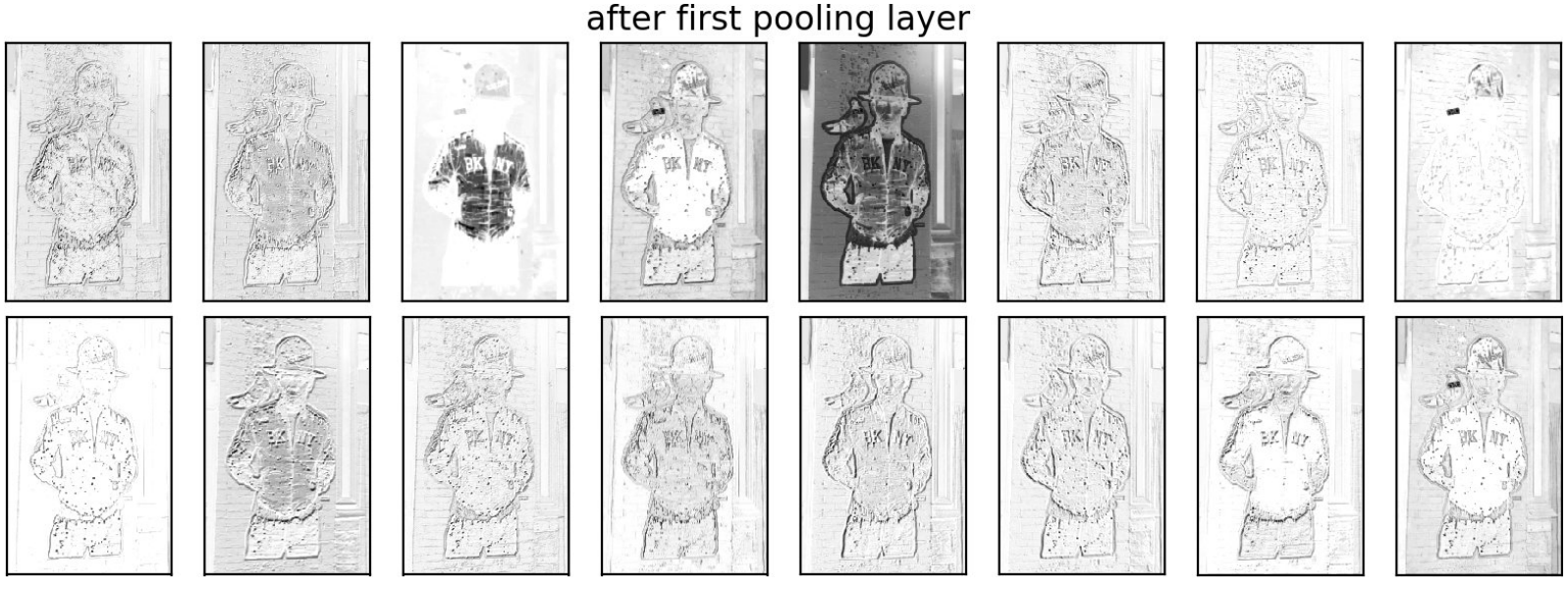


Fig. 2. Architecture of LeNet-5, a Convolutional Neural Network, here for digits recognition. Each plane is a feature map, i.e. a set of units whose weights are constrained to be identical.

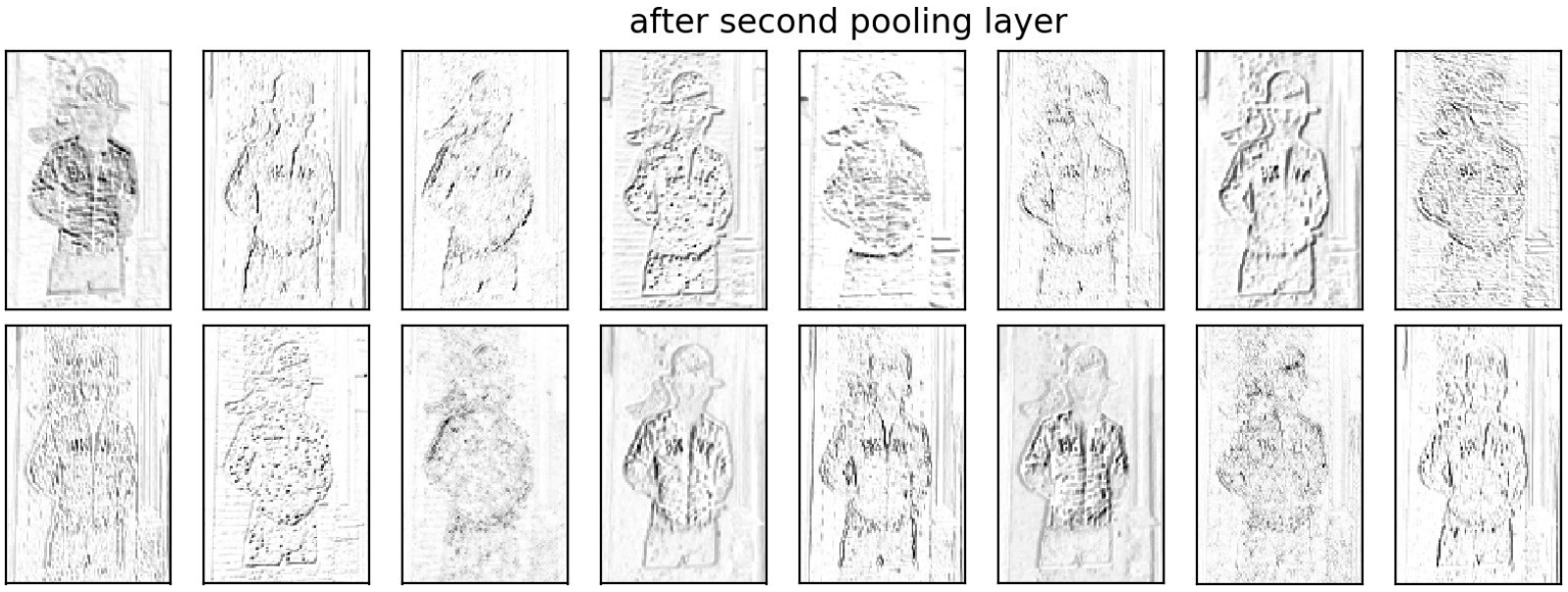
图像处理效果

经过第一层卷积和池化



图像处理效果

经过第二层卷积和池化



深度CNN

实际应用的模型层次非常多

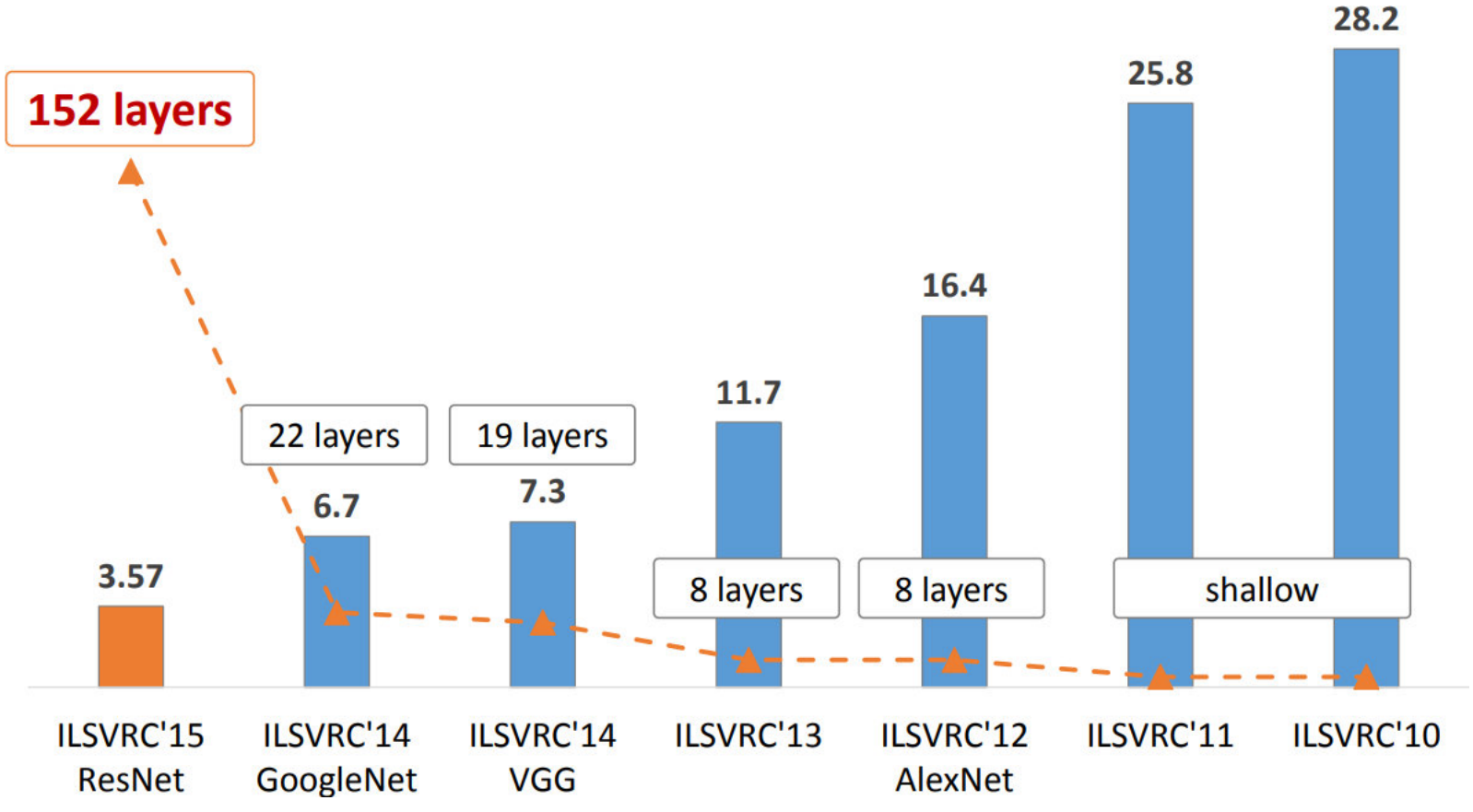
GoogleNet

GPU

- 几千万像素、上千万参数需要计算、调整
- 利用GPU的数千计算单元并行计算

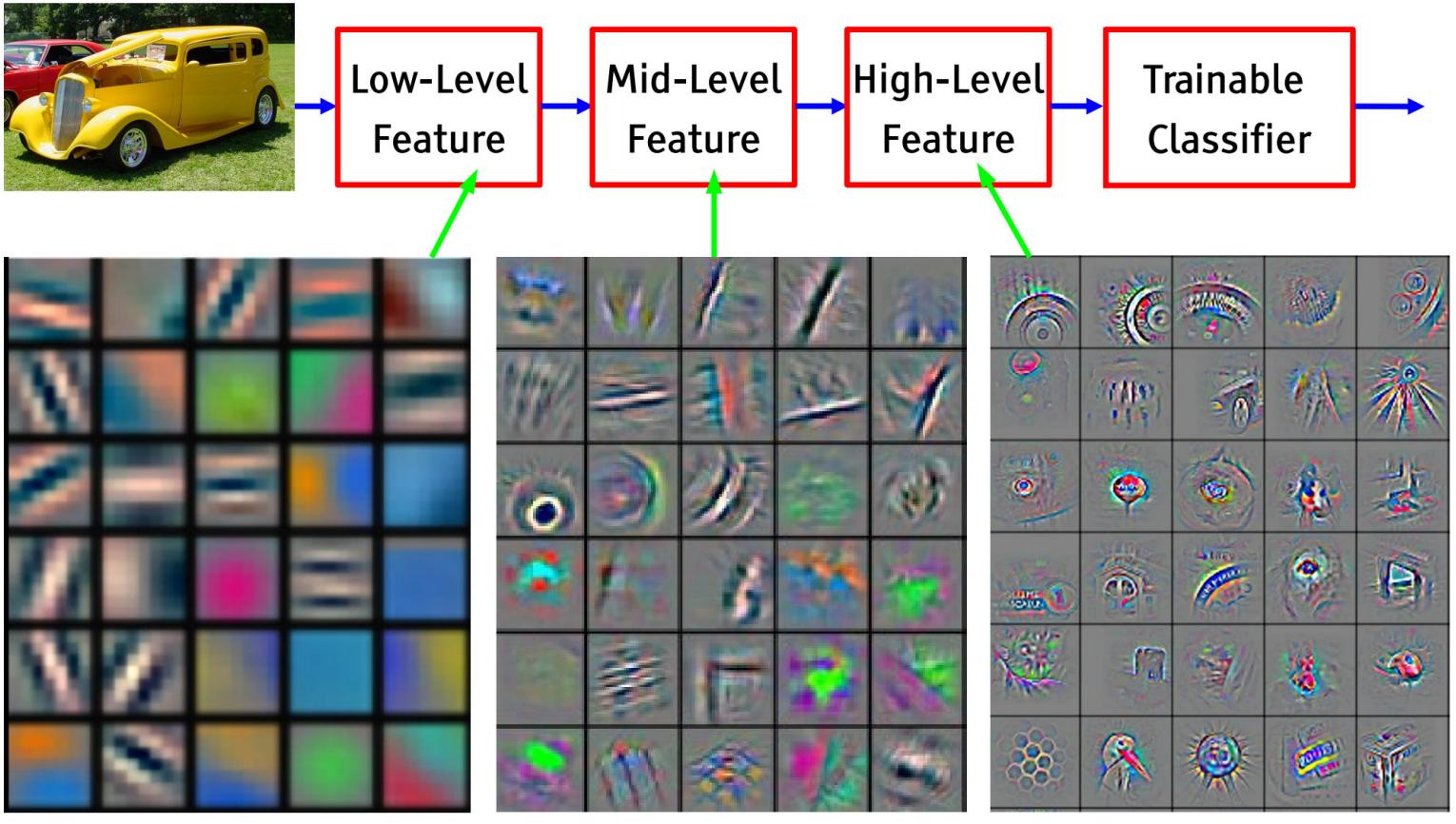
深度带来性能极大改善

ImageNet目标识别图像数据集



对各层卷积核的理解

- 底层提取简单特征，高层提取复杂特征



Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

CNN演示

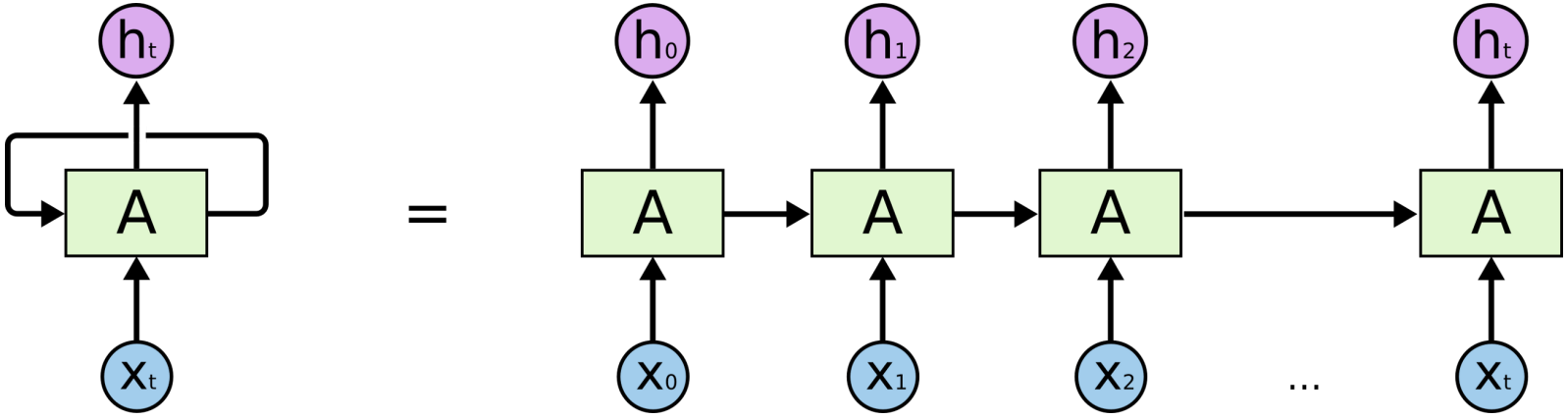
- Andrej Karpathy ConvNetJS
- https://cs.stanford.edu/people/karpathy/convnetjs/demo/mnist.html
- 在浏览器里训练CNN，实验MNIST手写体识别任务

循环神经元网络

RNN：Recurrent Neural Network

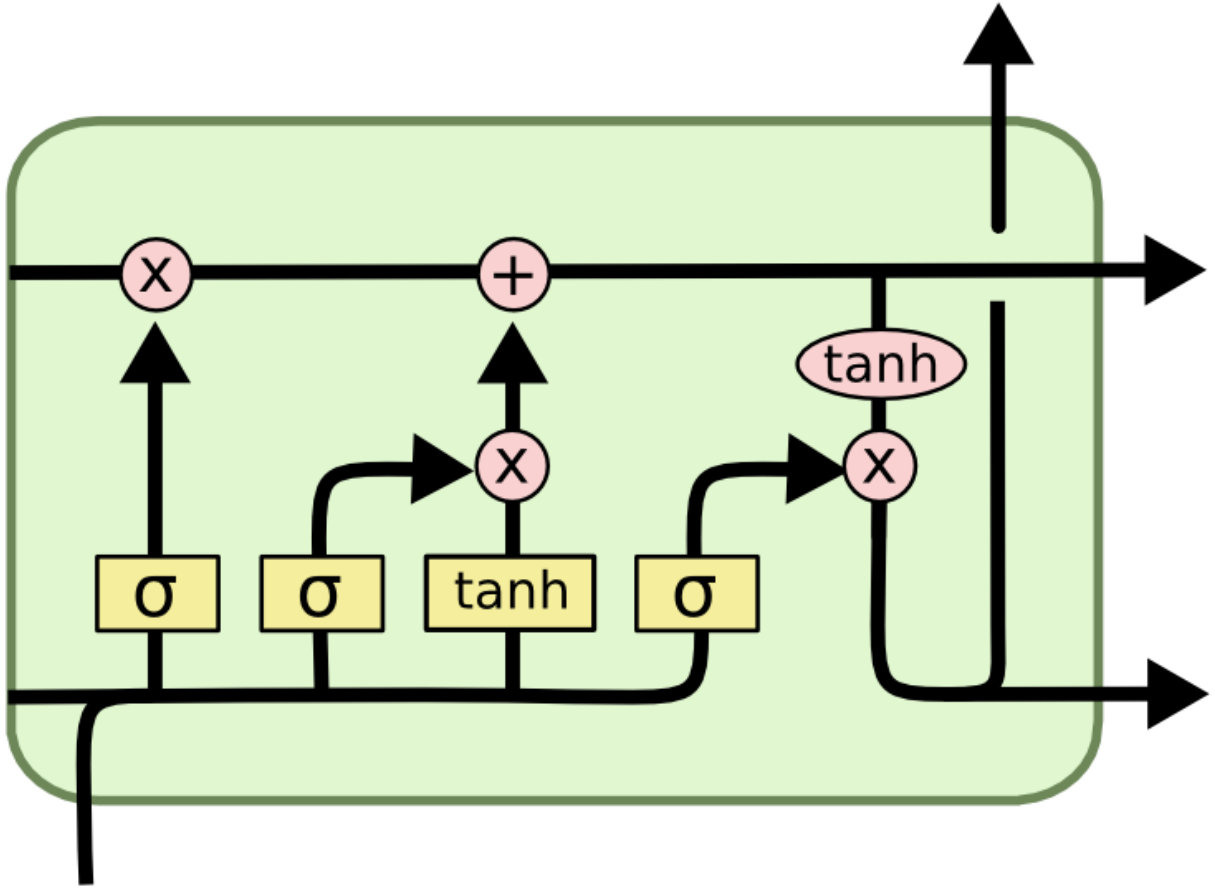
循环神经元网络

- “记忆单元”
- 适合处理时间序列数据、NLP任务
- 序列输入



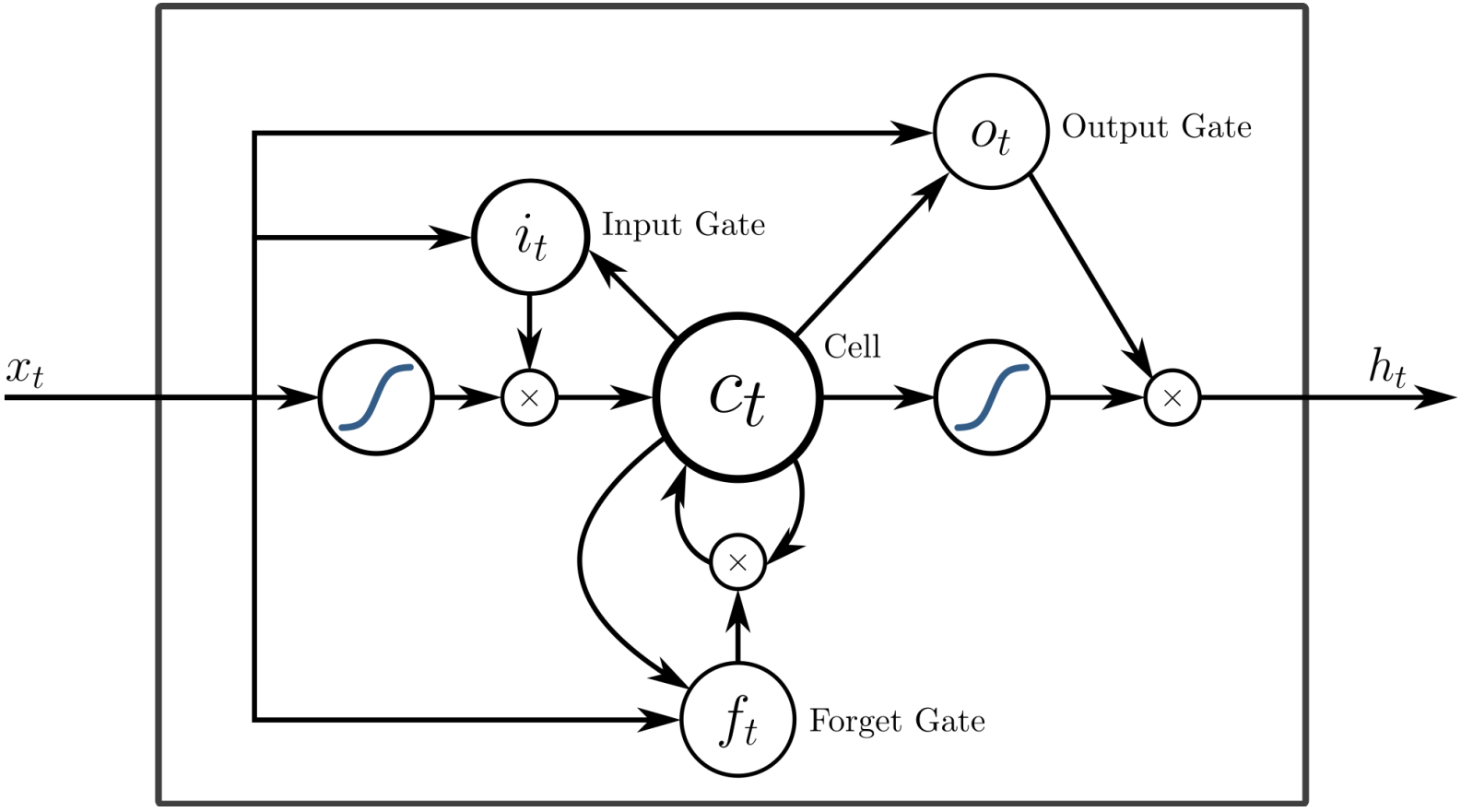
LSTM

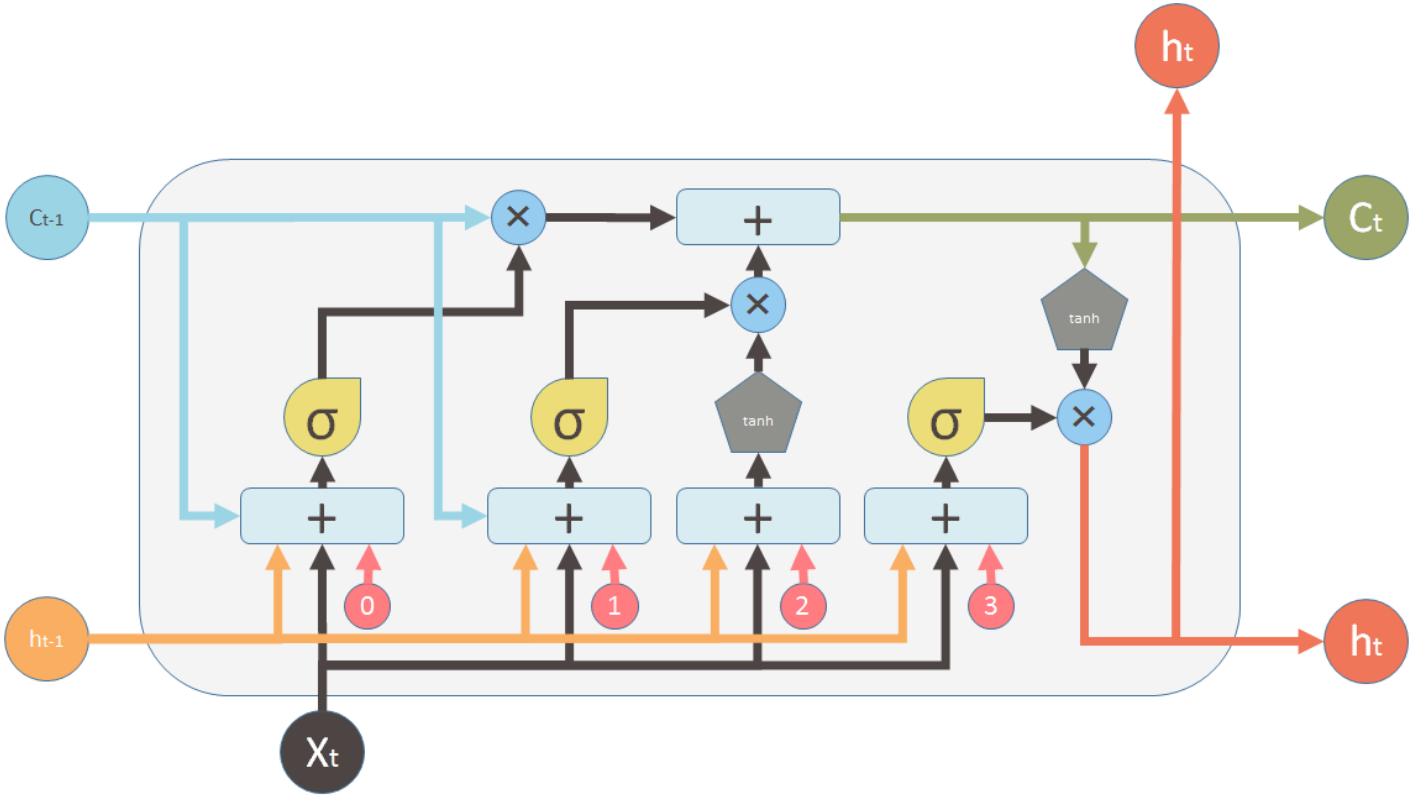
长短期记忆单元：Long short-term memory



LSTM

- 人的大脑会遗忘
- 输入门，输出门，遗忘门





Inputs:

- X_t Input vector
- C_{t-1} Memory from previous block
- h_{t-1} Output of previous block

outputs:

- C_t Memory from current block
- h_t Output of current block

Nonlinearities:

- σ Sigmoid
- tanh Hyperbolic tangent

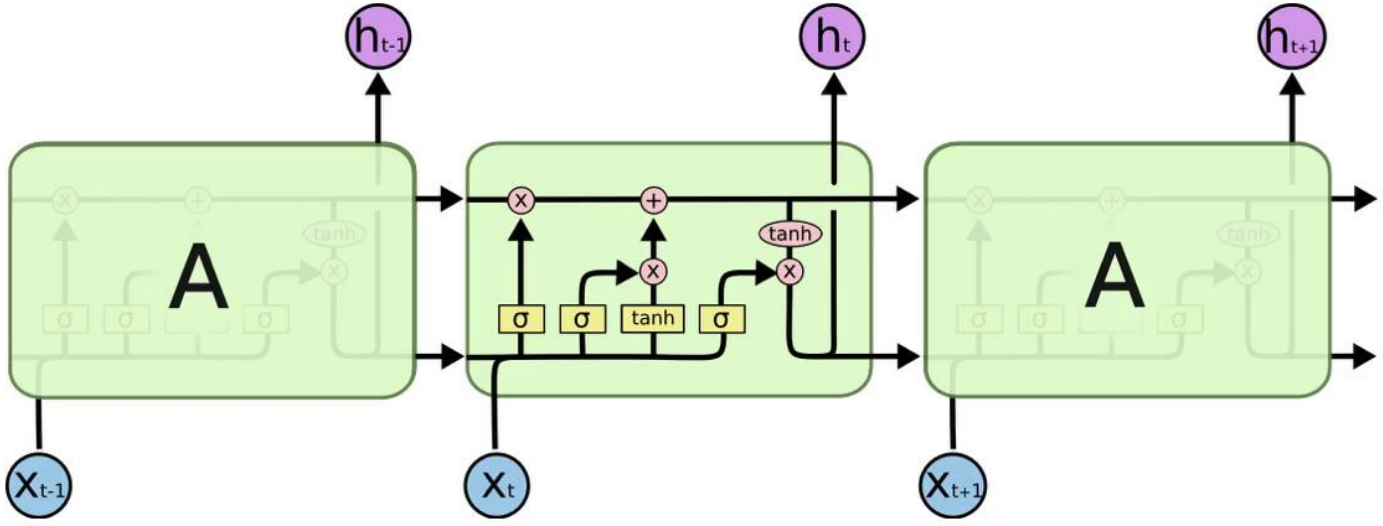
Bias: 0

Vector operations:

- × Element-wise multiplication
- \+ Element-wise Summation / Concatenation

基于LSTM的RNN

Long-Short Term Memory module: LSTM



long-short term memory modules used in an RNN

http://colah.github.io/posts/2015-08-Understanding-LSTMs/ Eugenio Culurciello © 2016

RNN的广泛应用

1. 语音识别
2. 机器翻译
3. 文本生成
4. 推荐系统
5. 时间序列预测

小结：深度学习模型三种结构

1. 前向神经元网络（FFN）
2. 卷积神经元网络（CNN）
3. 循环神经元网络（RNN）

进展

进展

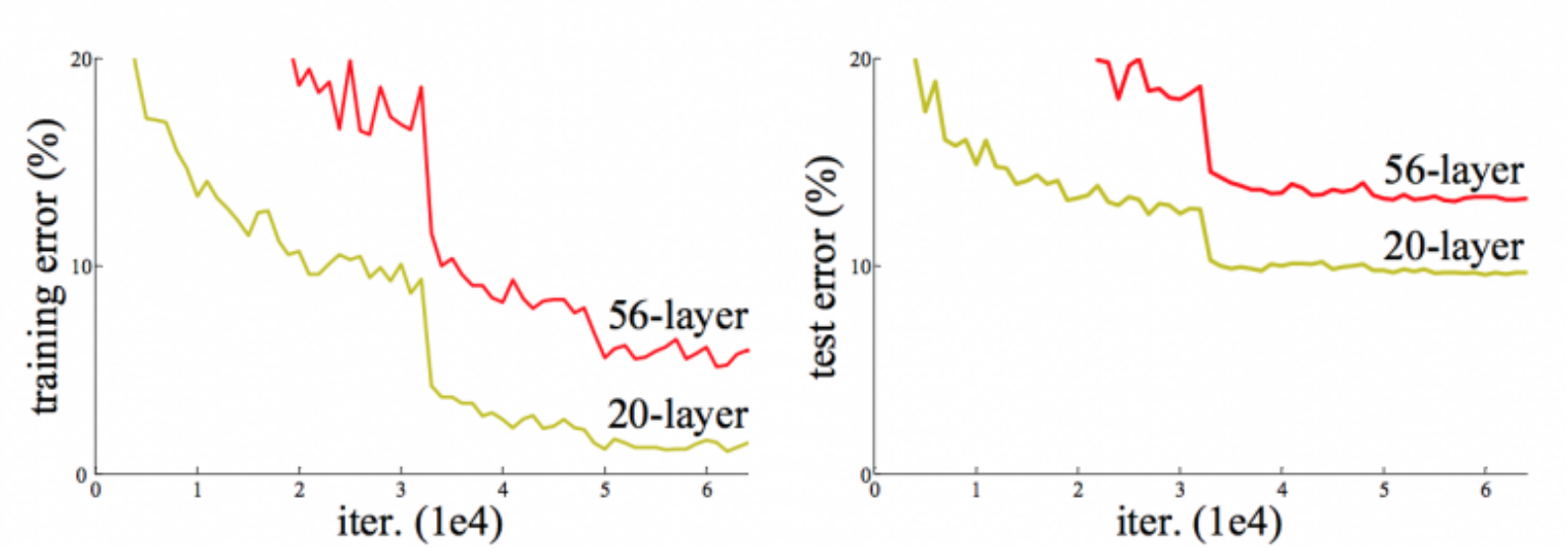
- 有许多不同类型的神经网络
- 每种神经网络都可用于解决特定的AI问题
- 这个领域正在迅速发展
  - Ian Goodfellow在2014年发明了GAN
  - Capsule network

ResNet

残差网络

ResNet

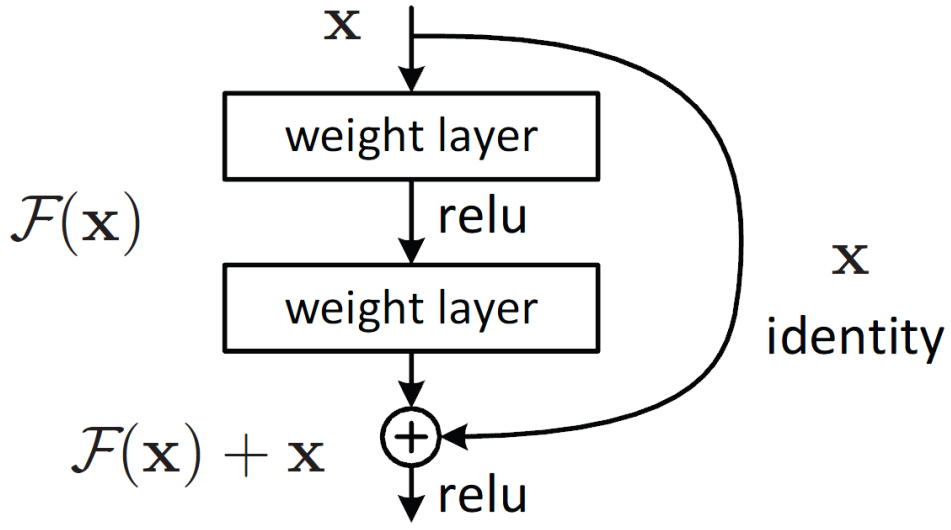
一般的深度神经元网络，超过一定的层数后，层数越多，越难优化，性能反而变差



CIFA-10数据集

ResNet

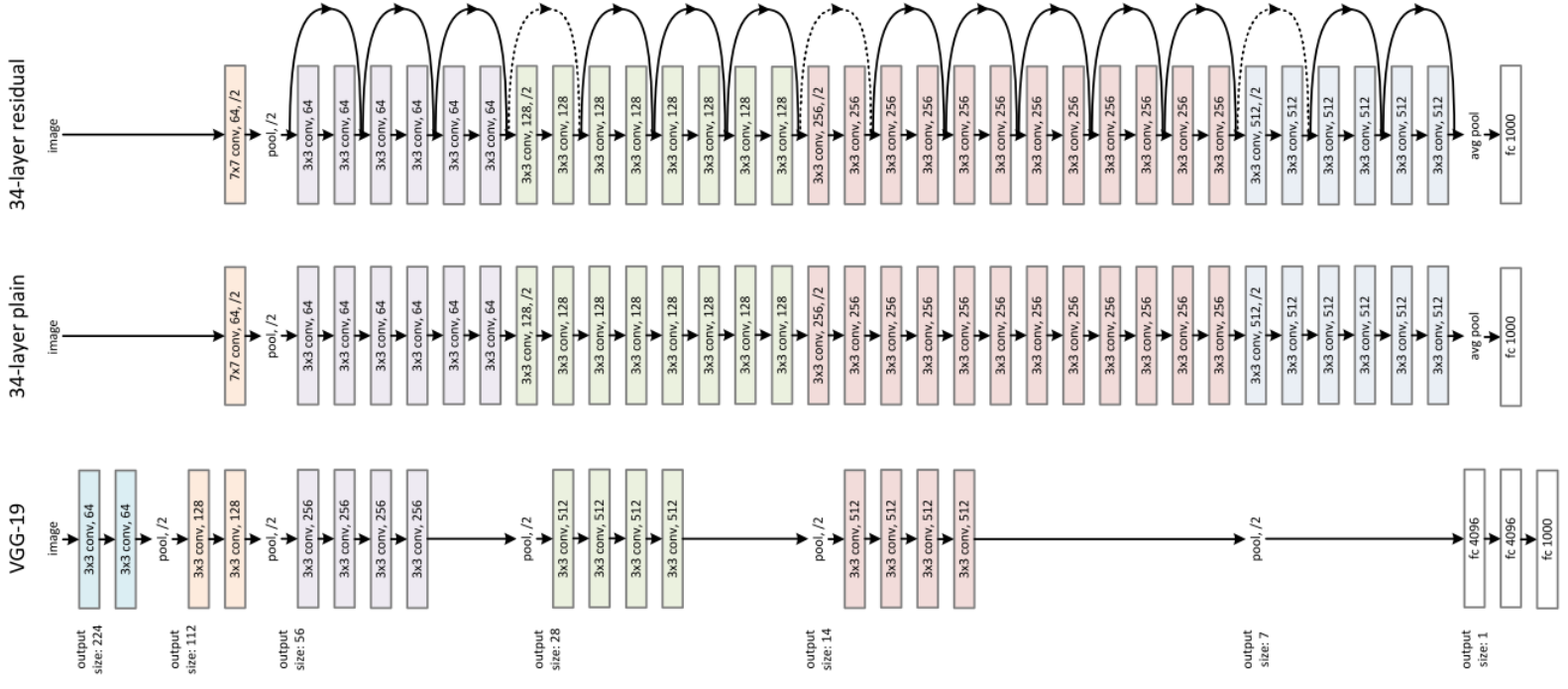
- 残差网络
- 加入直连链路



Residual Network

ResNet

支持更深的网络，获得更好的性能



Attention

注意力机制

Attention

- 人的注意力不是平均的
- 给不同元素不同注意力，能够改进性能

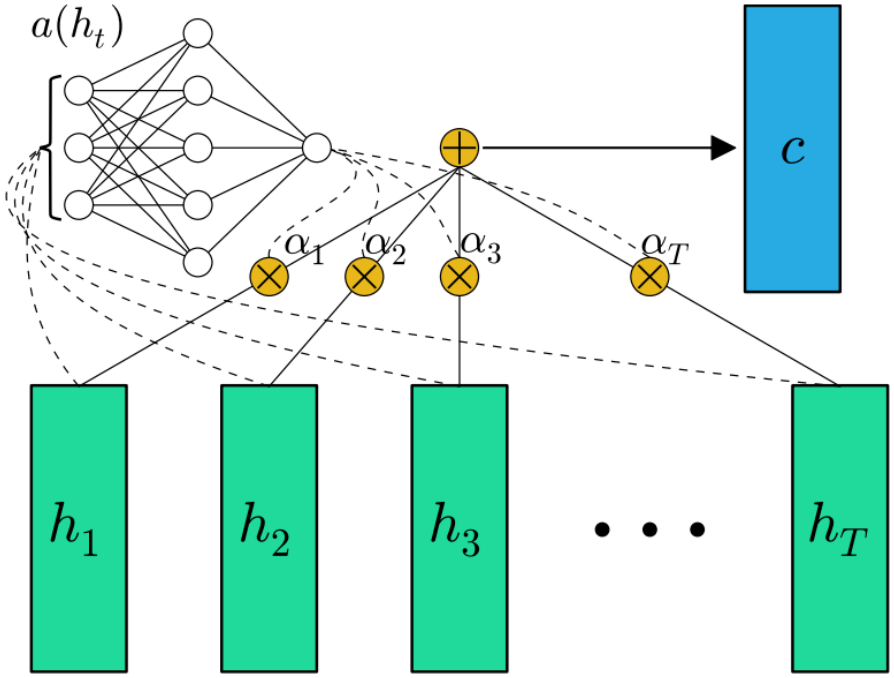
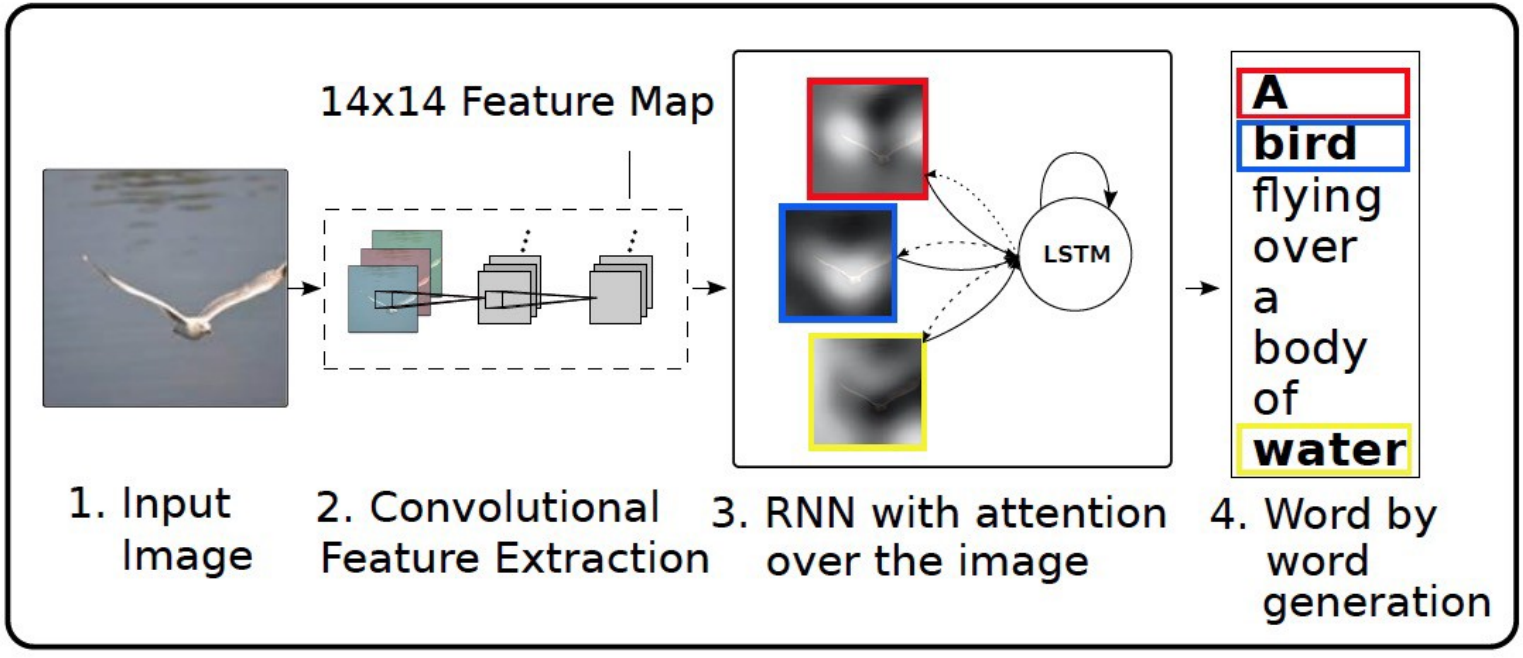


Figure 1: Schematic of our proposed "feed-forward" attention mechanism (cf. (Cho, 2015) Figure 1). Vectors in the hidden state sequence h_t are fed into the learnable function $a(h_t)$ to produce a probability vector α. The vector c is computed as a weighted average of h_t, with weighting given by α.

Attention在图像理解中的应用

生成图像的文字描述



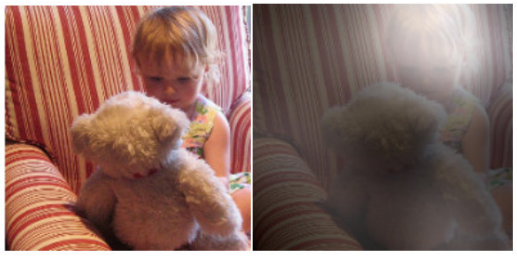
Attention在图像理解中的应用

将文字和图像中的目标匹配

A woman is throwing a frisbee in a park.

A dog is standing on a hardwood floor.

A stop sign is on a road with a mountain in the background.



A little girl sitting on a bed with a teddy bear.

A group of people sitting on a boat in the water.

A giraffe standing in a forest with trees in the background.

Transformer

避免RNN结构，利用Attention

Transformer

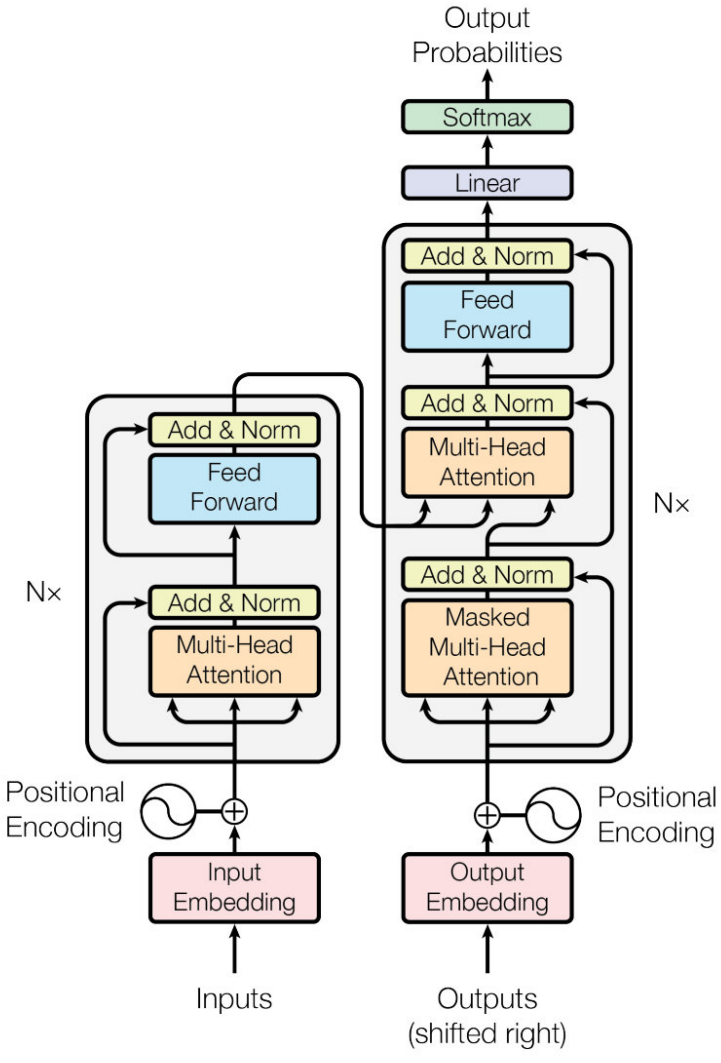


Figure 1: The Transformer - model architecture.

模型性能

Model Performance

模型性能

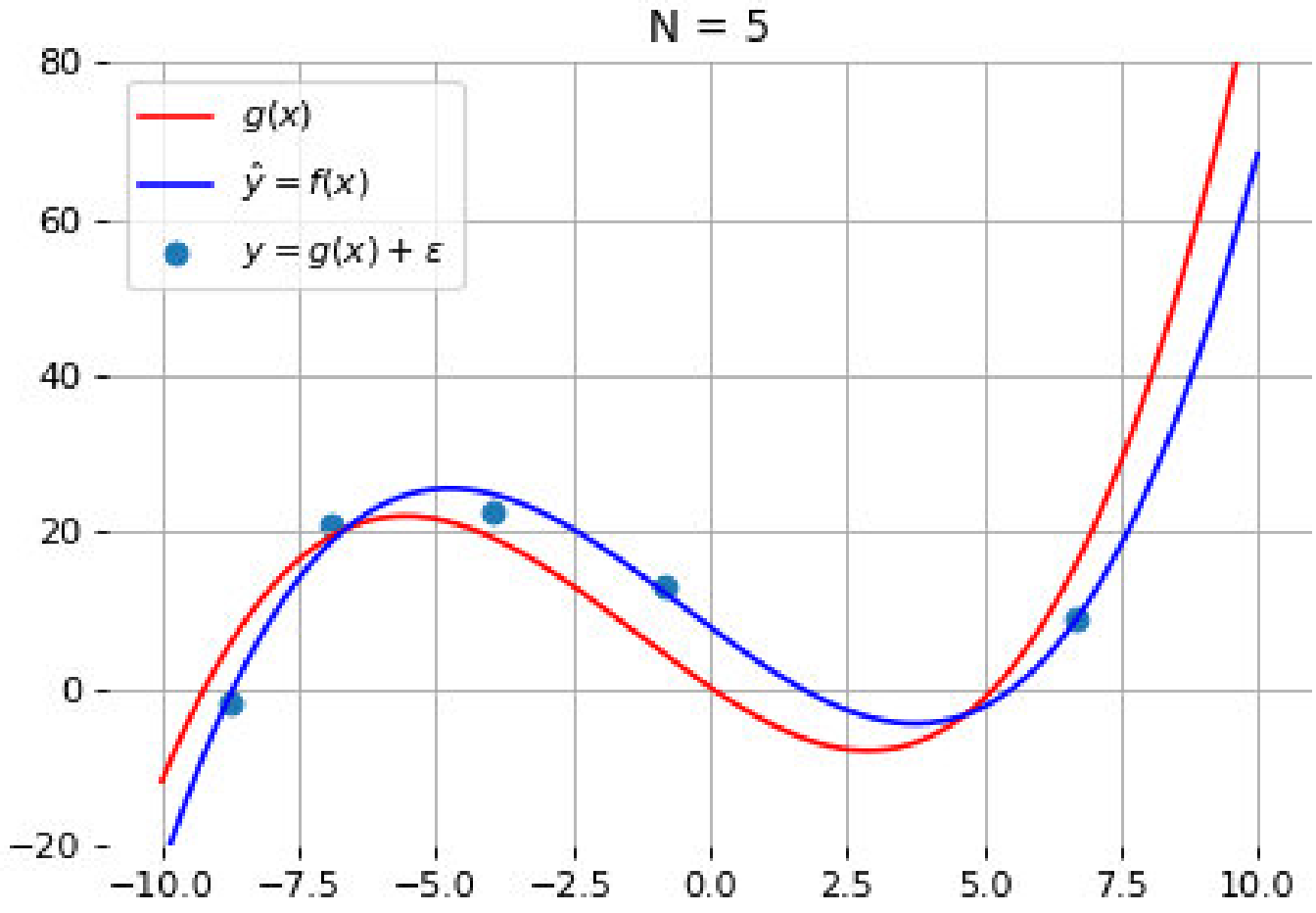
- 数据
- 模型
- 训练方法
- 优化
- 测试调优

1）数据

好的数据是成功的关键

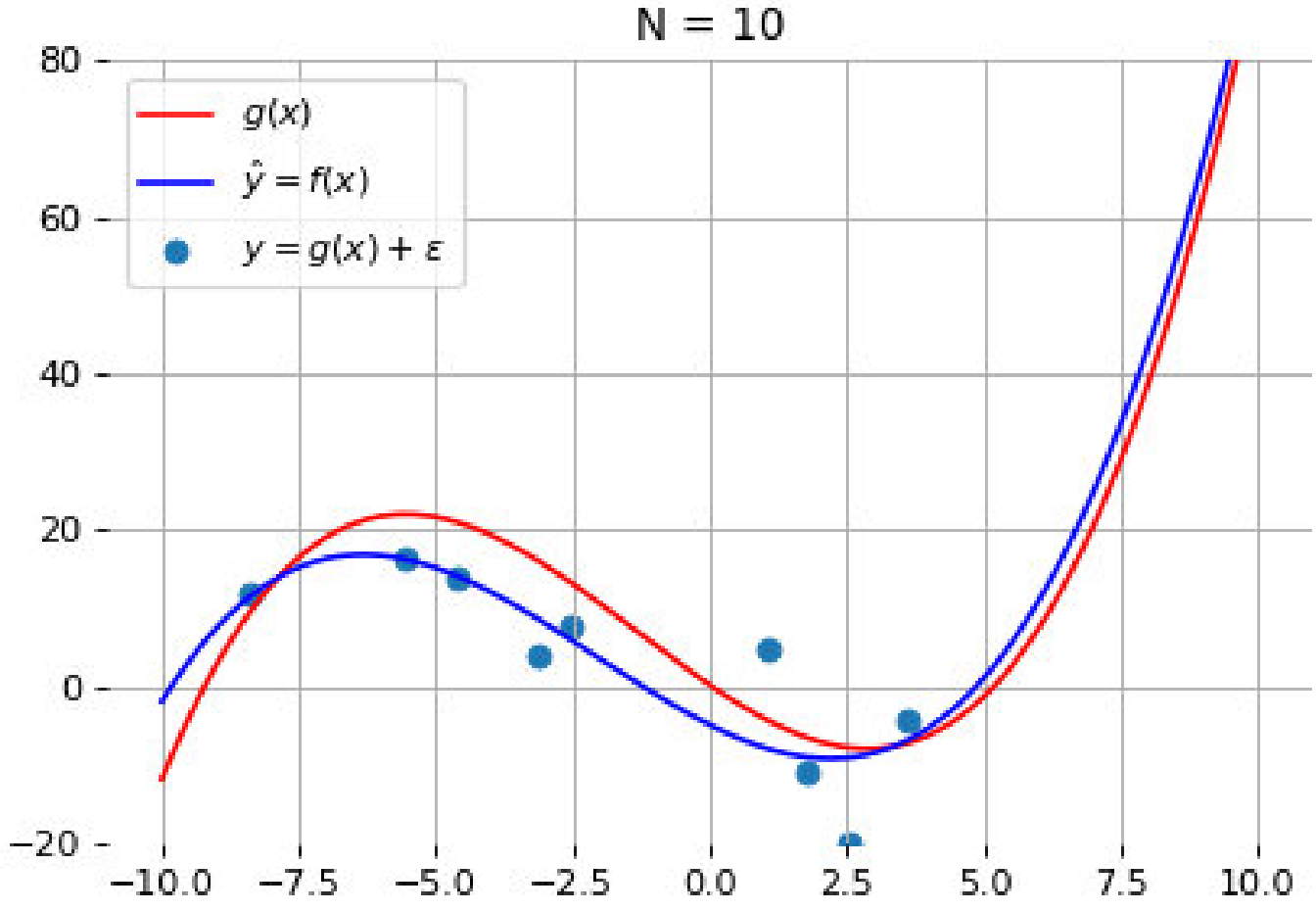
数据量对结果的影响

数据量少，模型误差大



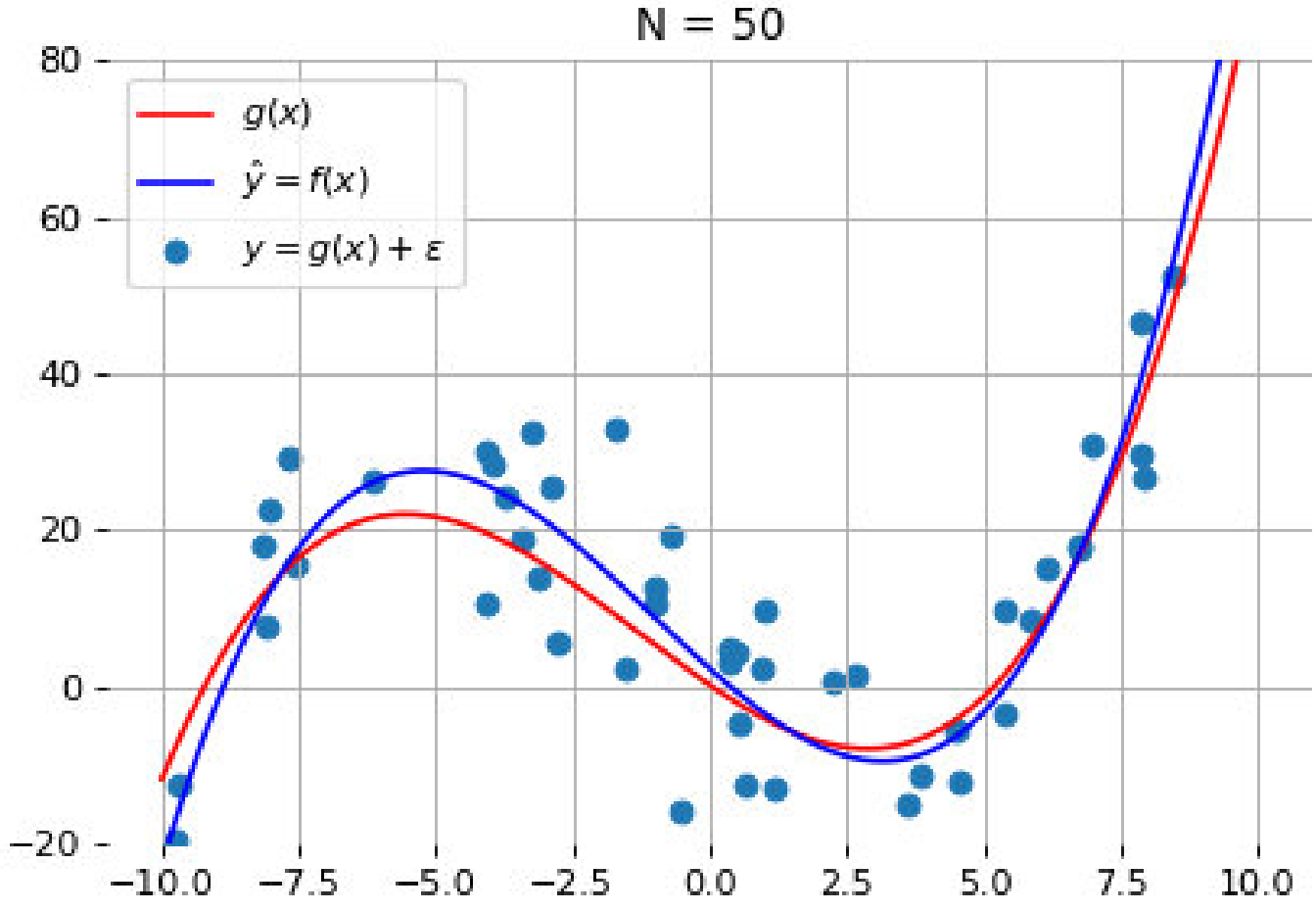
数据量对结果的影响

随着数据量增长，模型误差减少



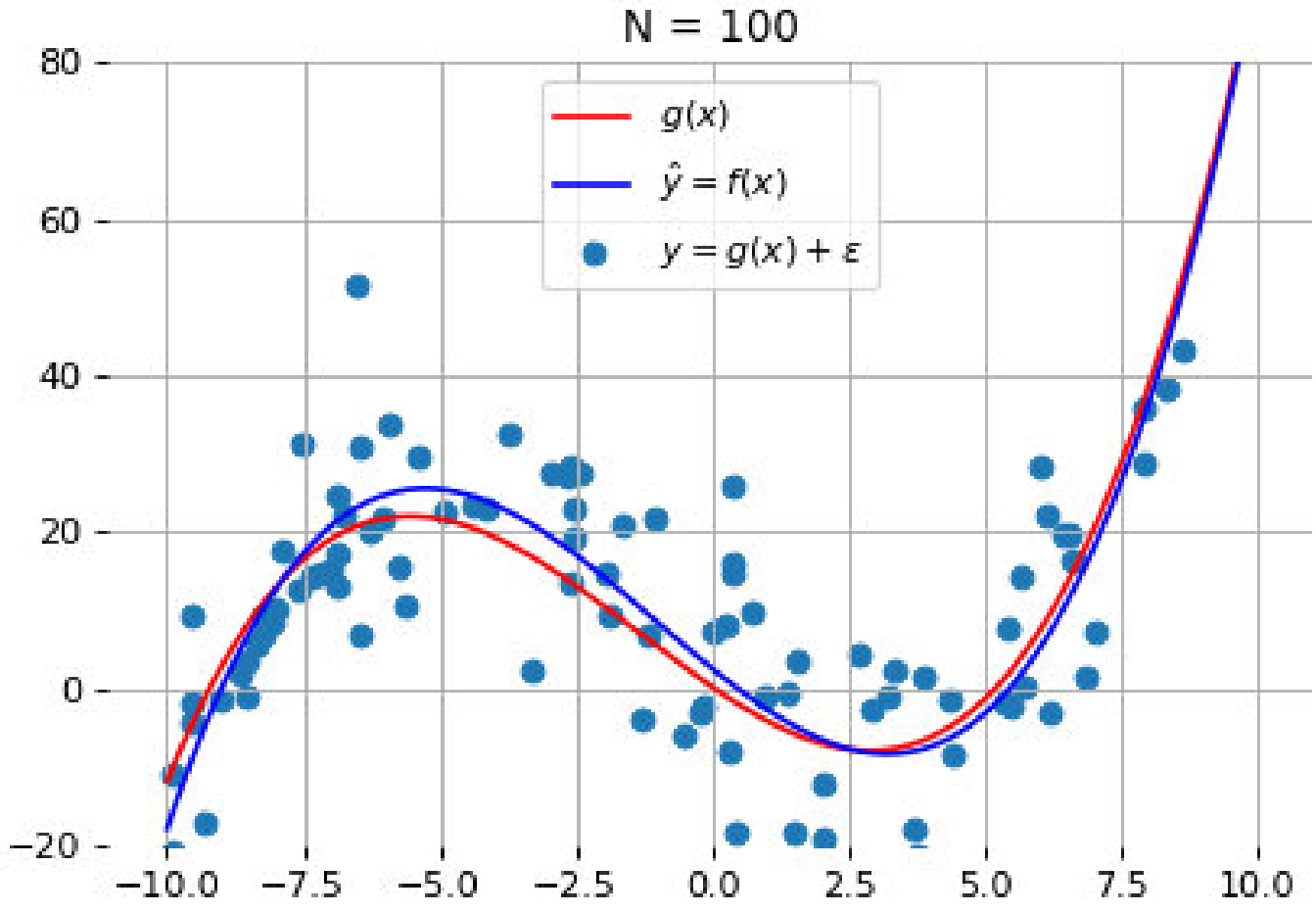
数据量对结果的影响

随着数据量增长，模型误差减少



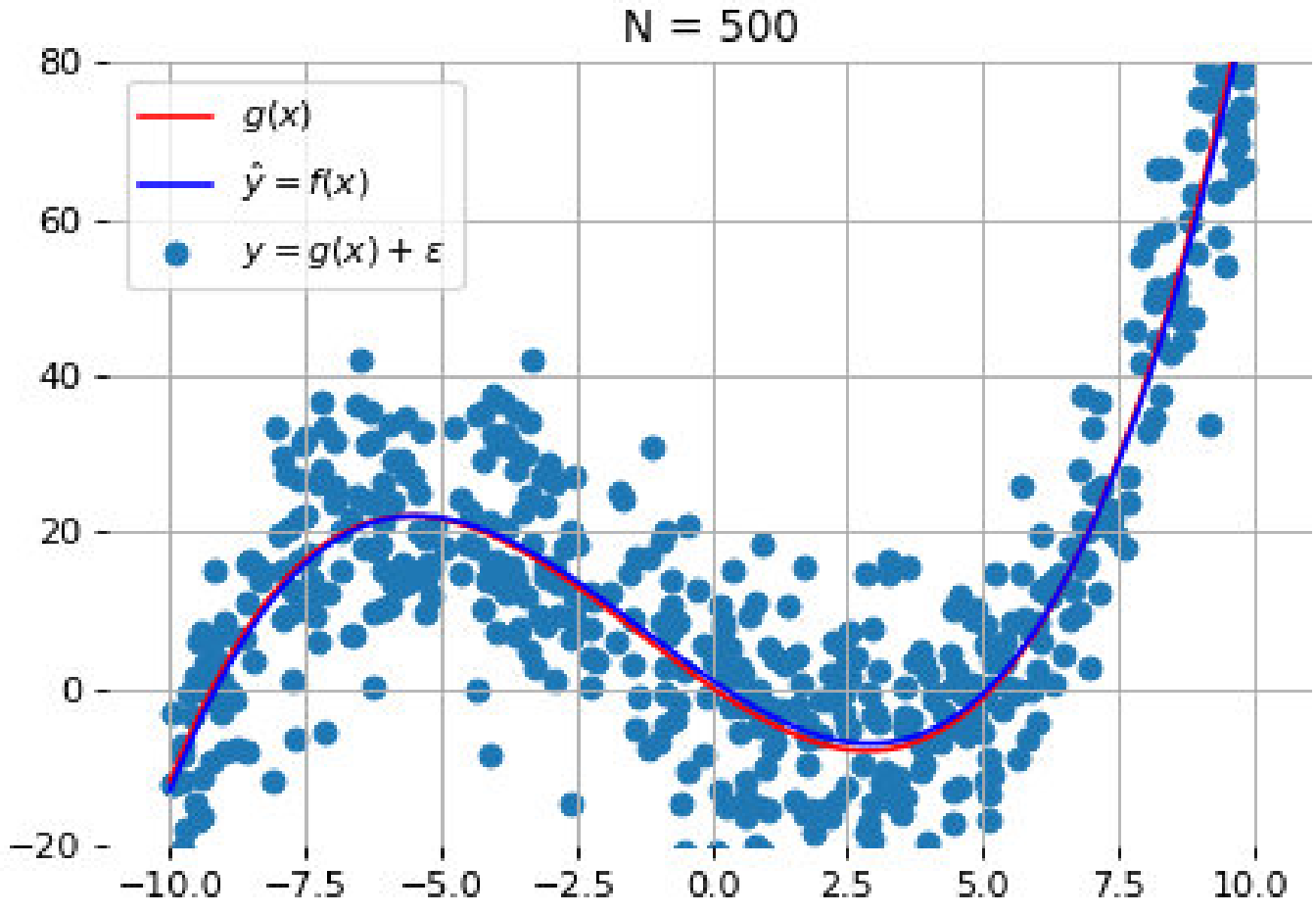
数据量对结果的影响

随着数据量增长，模型误差减少



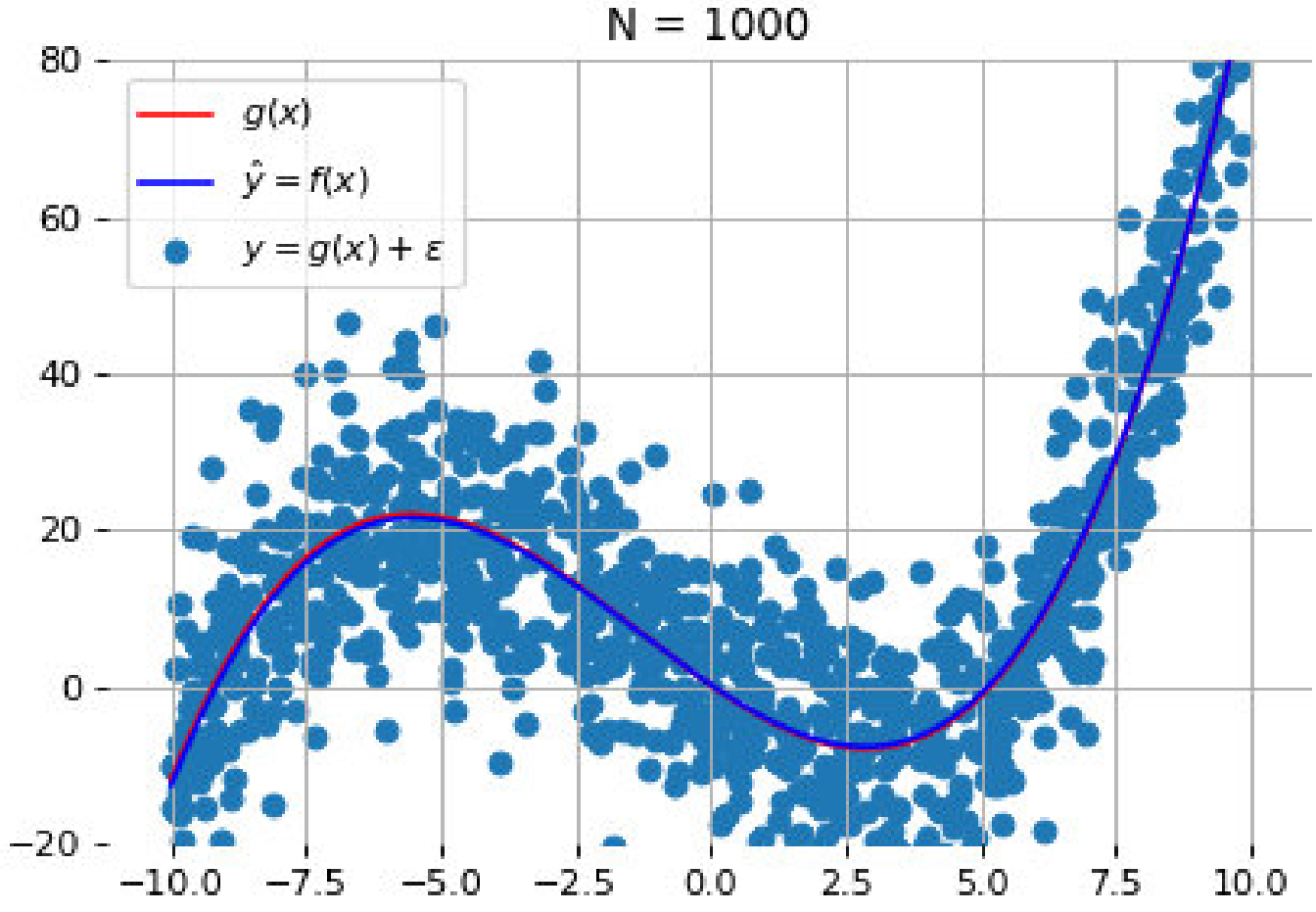
数据量对结果的影响

随着数据量增长，模型误差减少



数据量对结果的影响

随着数据量增长，模型误差减少

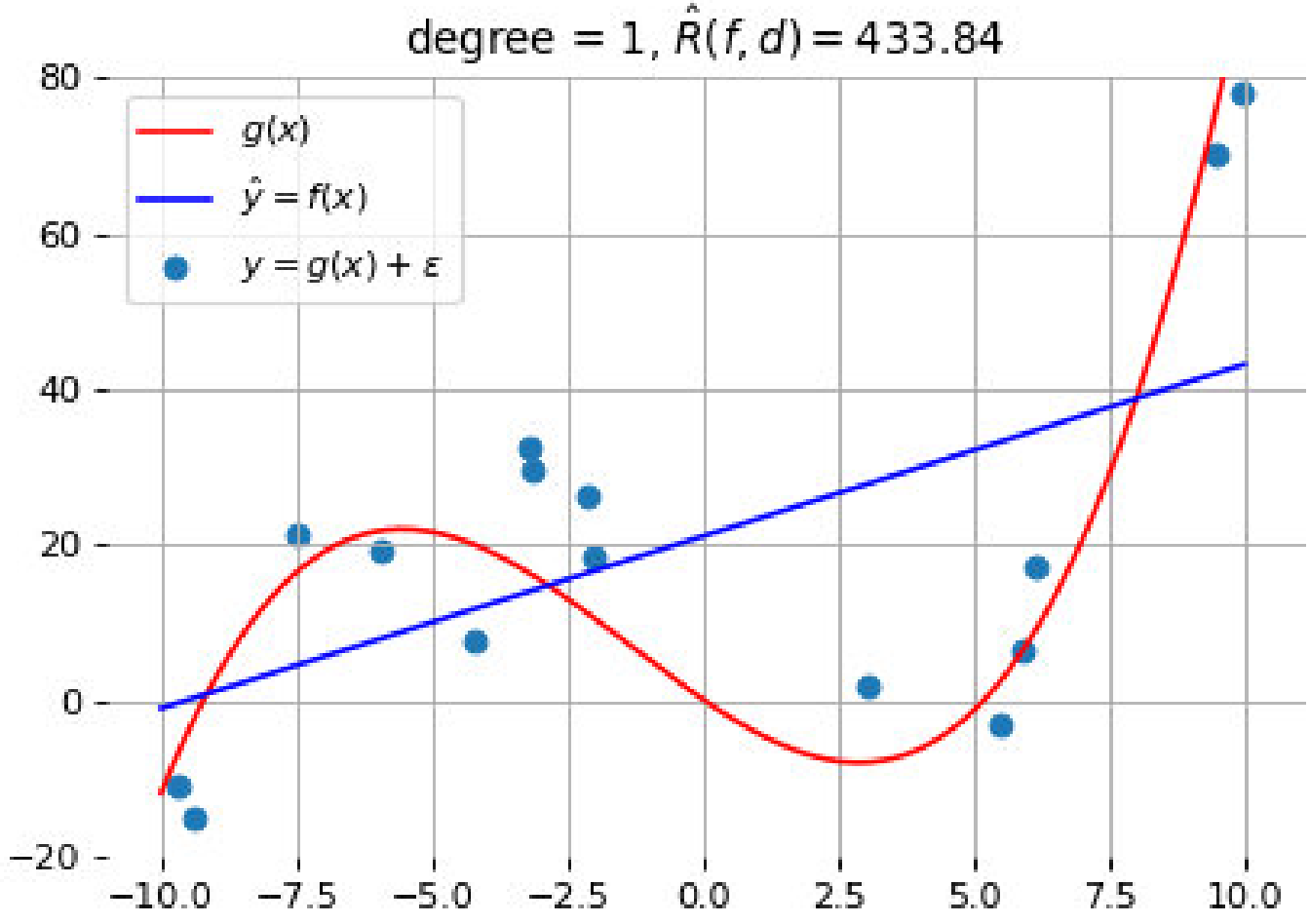


2）模型

模型选择非常重要

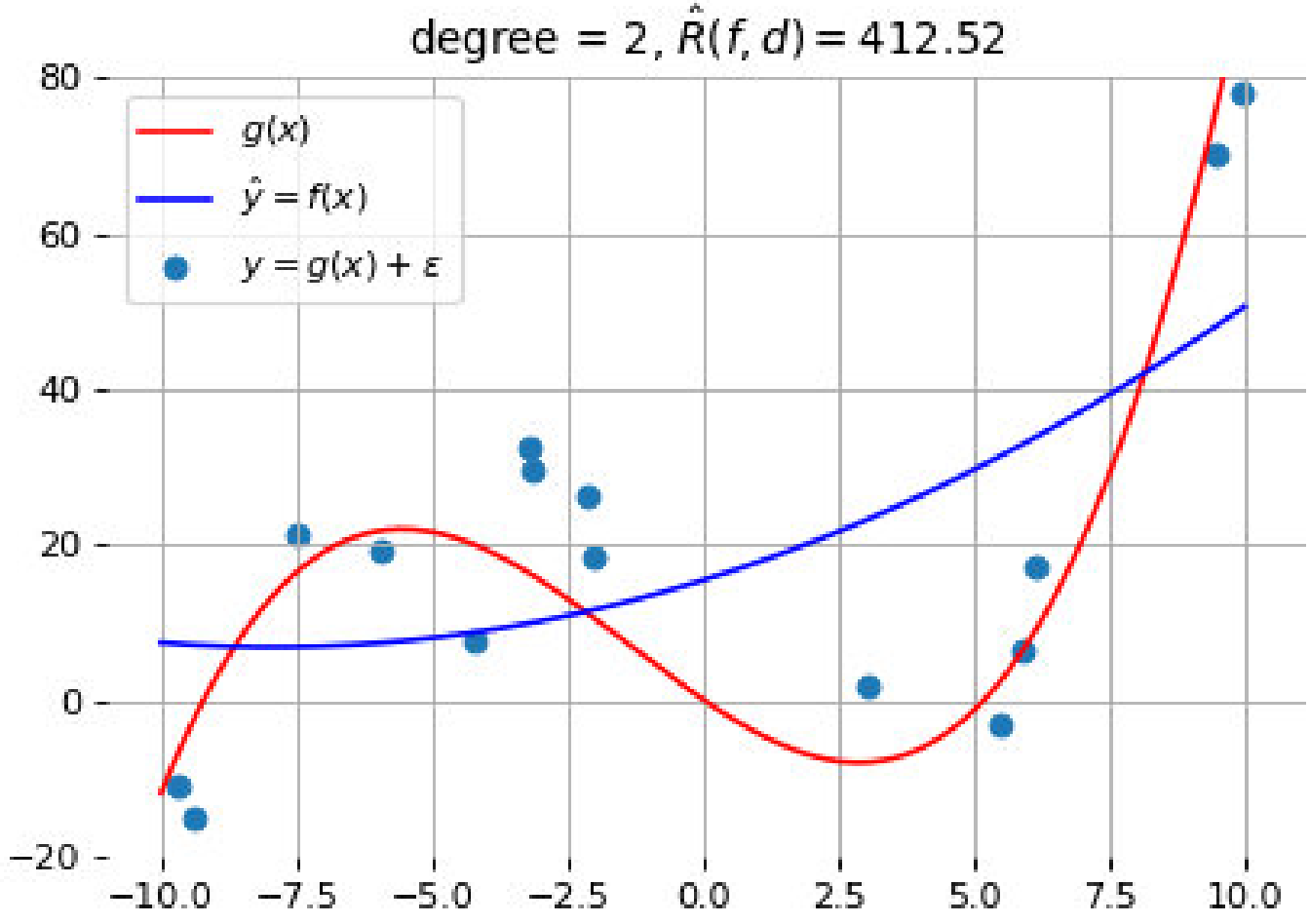
模型能力

模型能力不够，欠拟合



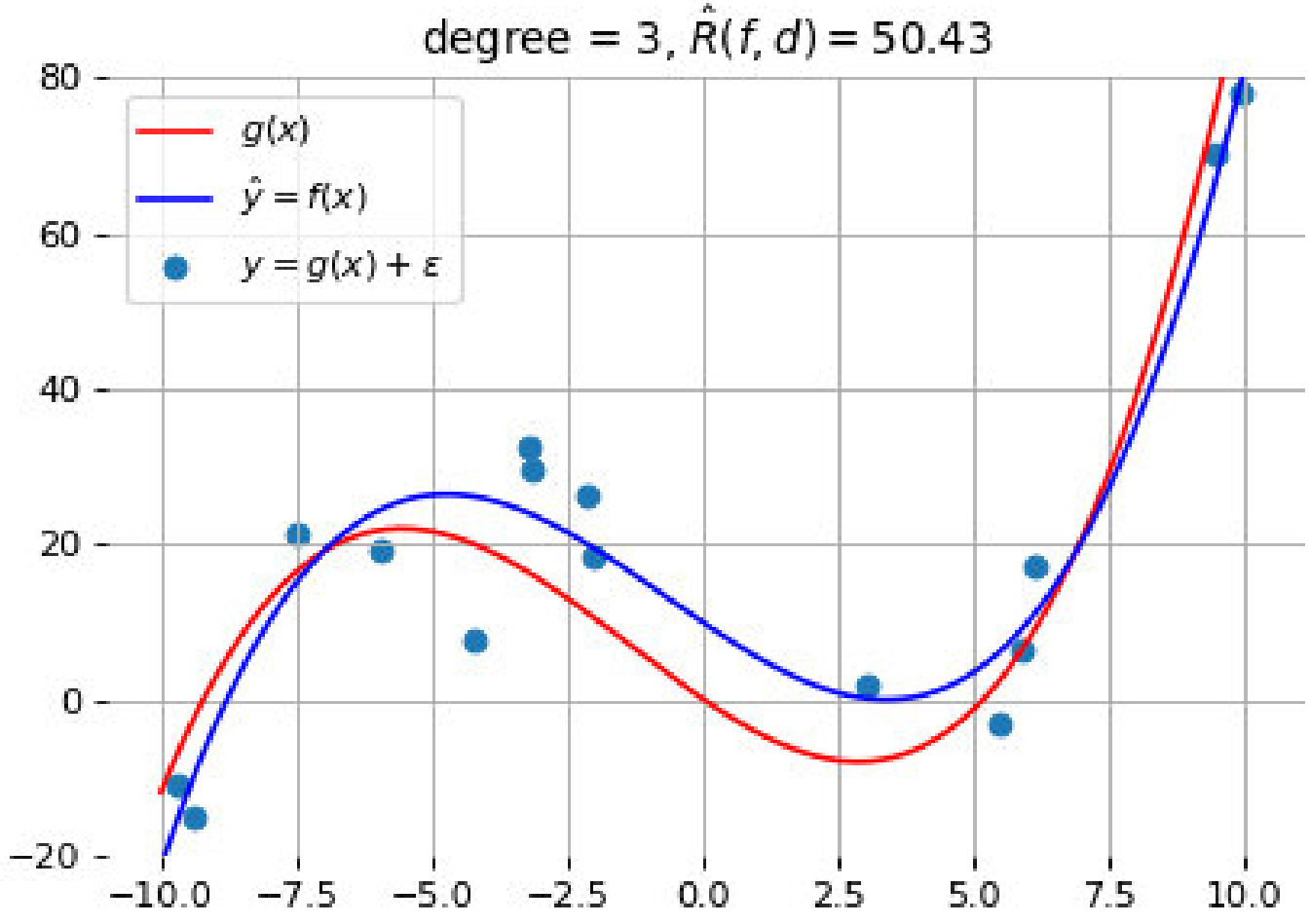
模型能力

模型能力不够，欠拟合



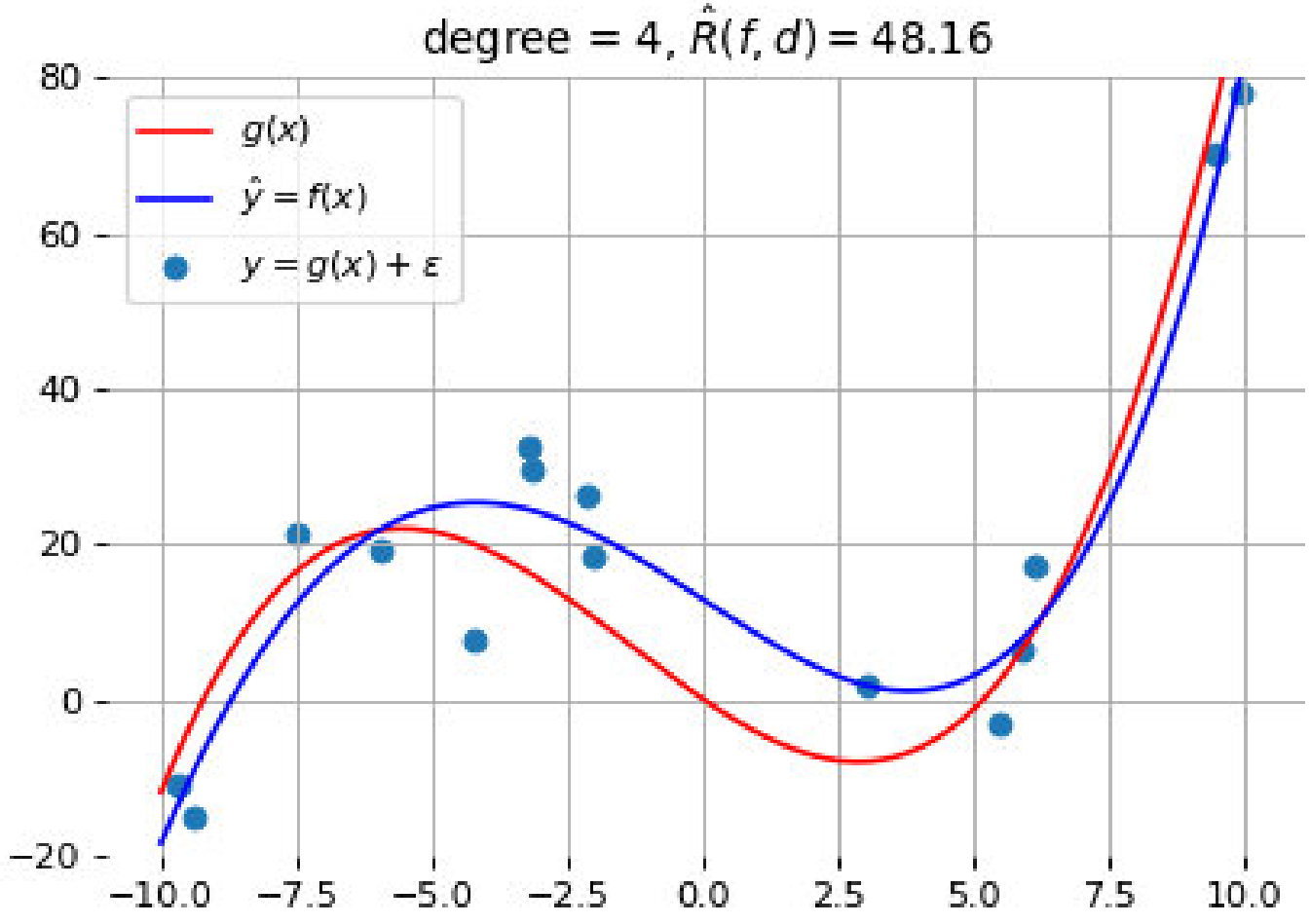
模型能力

模型能力适中


degree = 3, $\hat{R}(f, d) = 50.43$
$g(x)$
$\hat{y} = f(x)$
$y = g(x) + \varepsilon$
80
60
40
20
0
−20
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0

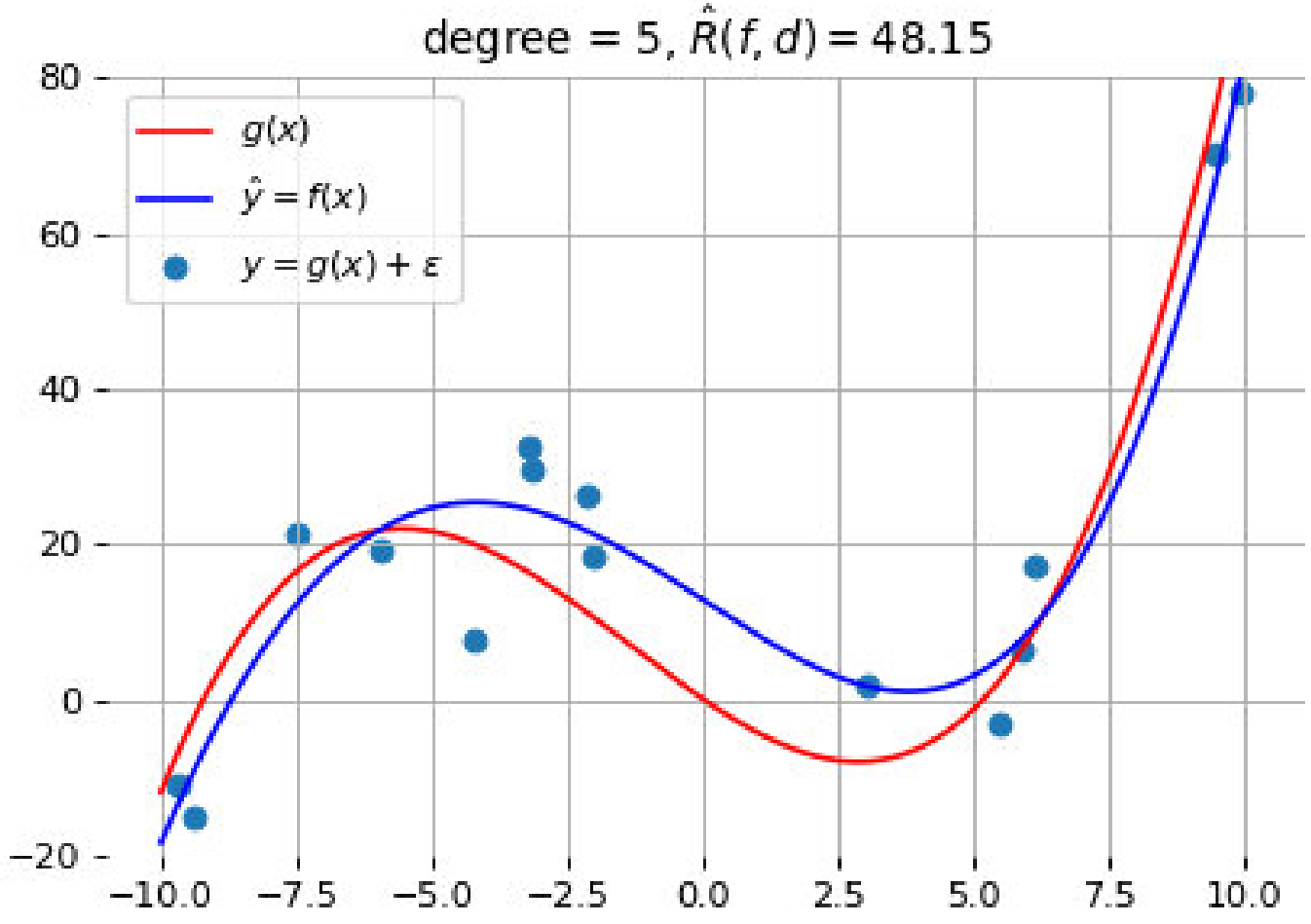
模型能力

模型能力适中


degree = 4, $\hat{R}(f, d) = 48.16$
$g(x)$
$\hat{y} = f(x)$
$y = g(x) + \varepsilon$
80
60
40
20
0
−20
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0

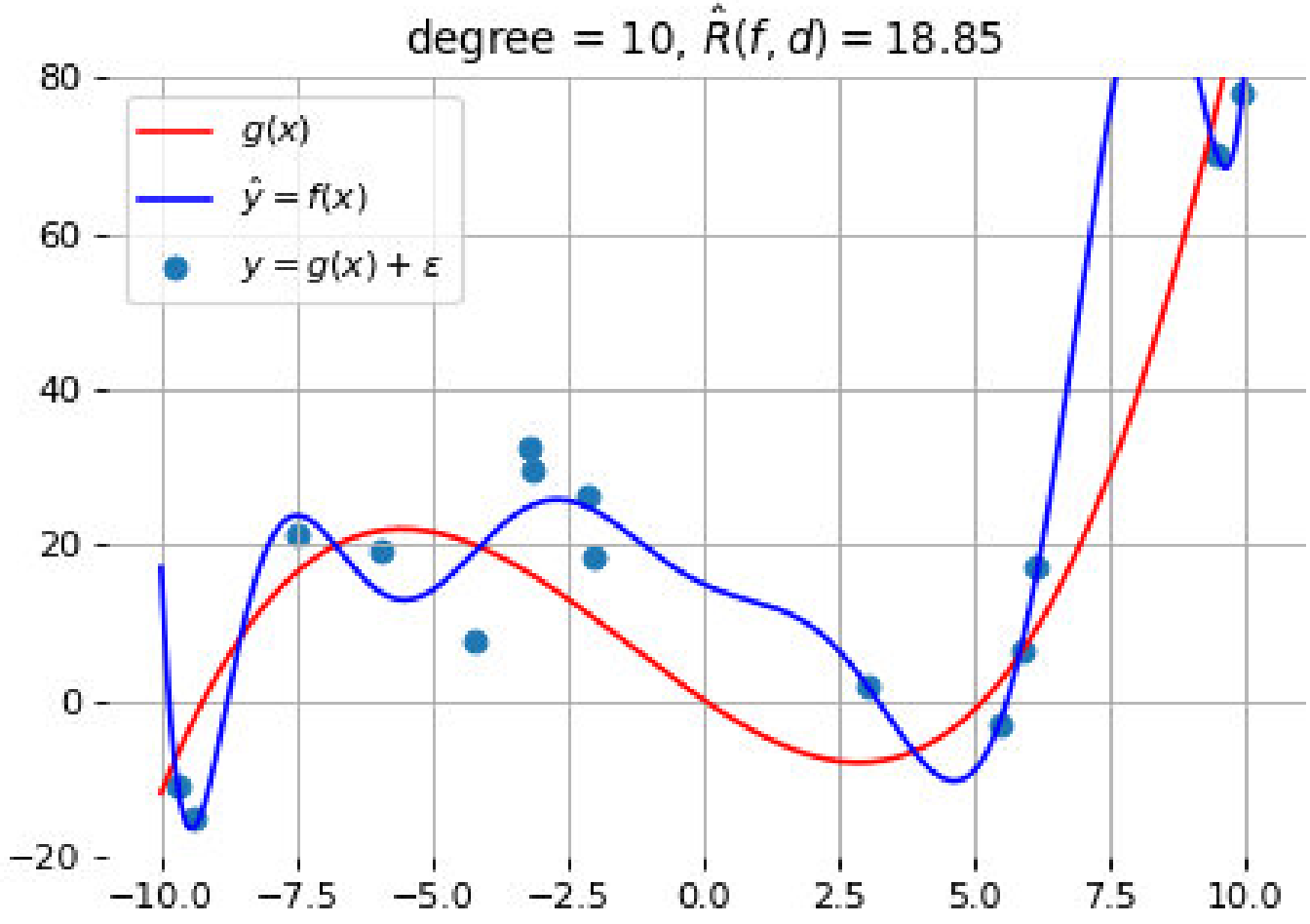
模型能力

模型能力适中


degree = 5, $\hat{R}(f, d) = 48.15$
$g(x)$
$\hat{y} = f(x)$
$y = g(x) + \varepsilon$
80
60
40
20
0
−20
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0

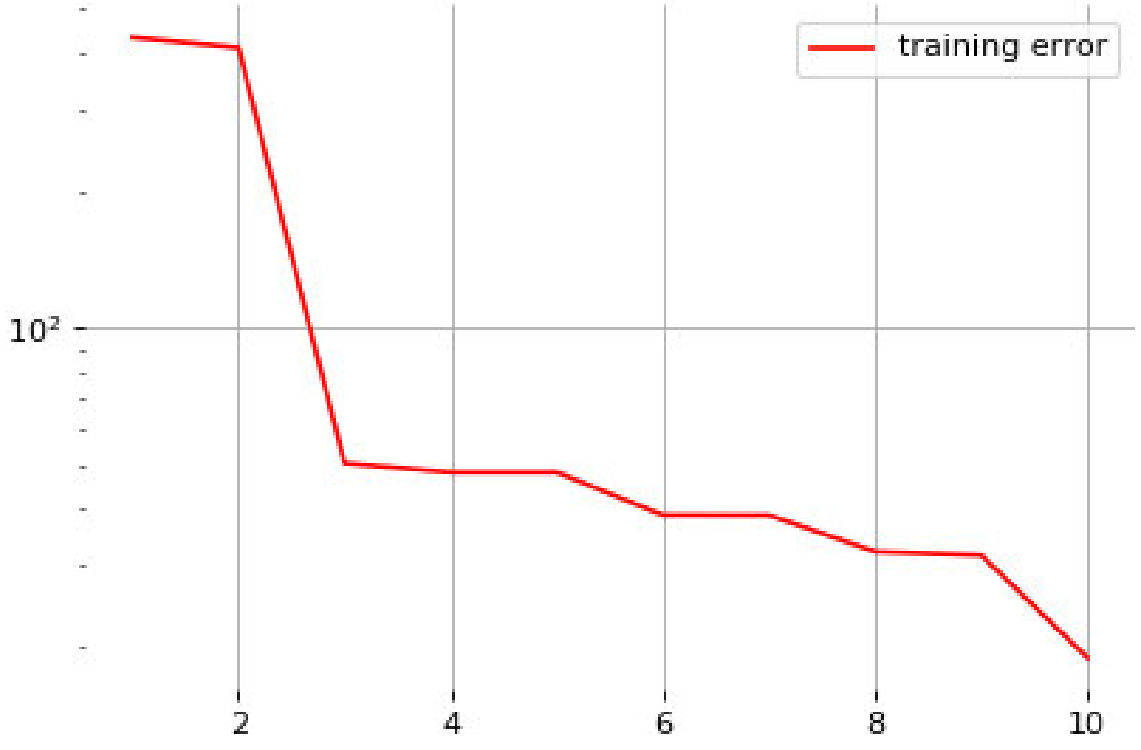
模型能力

模型能力太强，过拟合



模型能力

- 训练集上，模型错误随模型能力增长一直下降
- 但最后的下降，是过拟合了

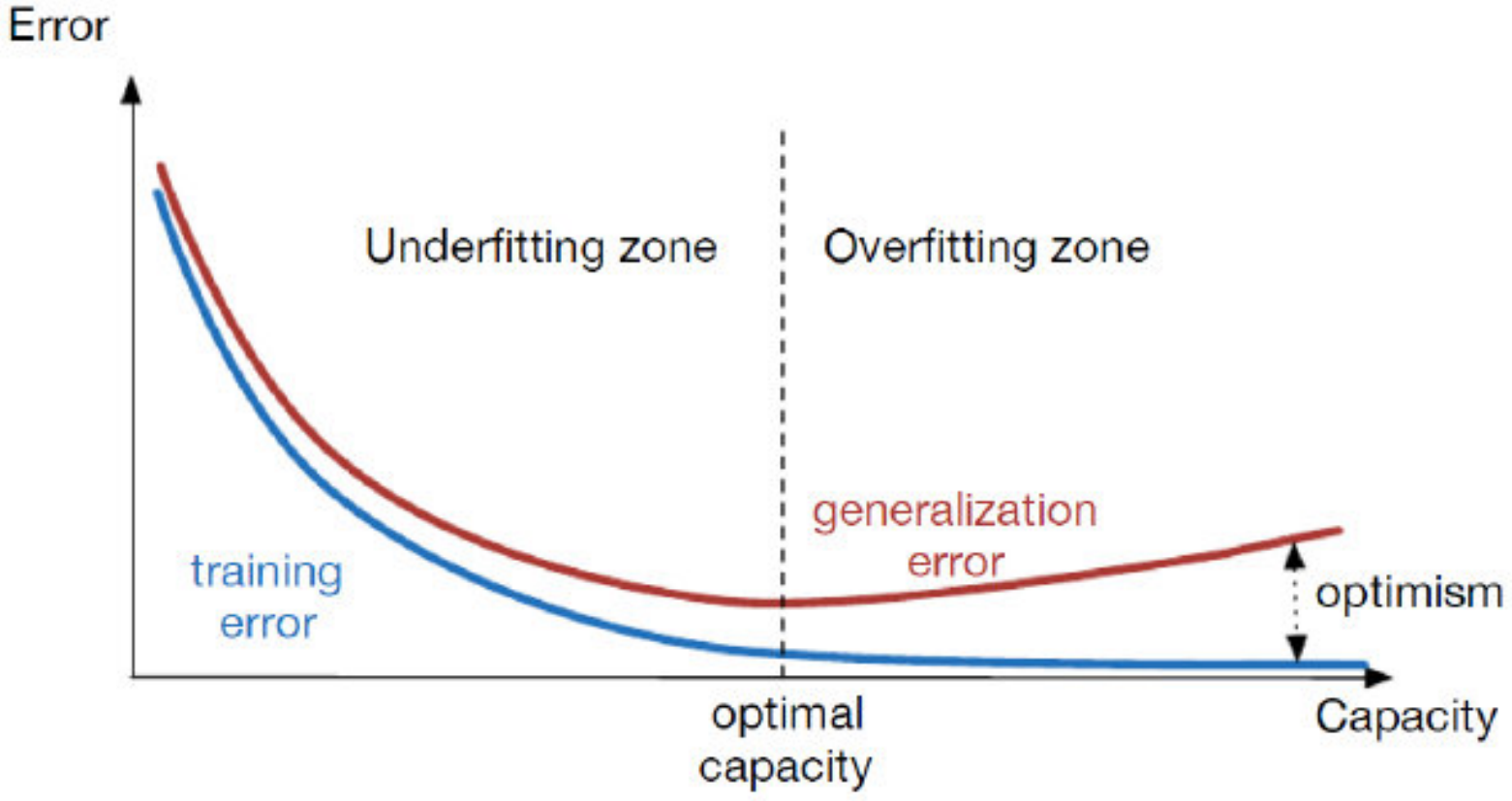


过拟合

- 过拟合导致模型在测试集上错误上升

模型能力

选择合适的模型非常重要



模型选择

- 深度神经网络不是唯一的机器学习算法
- 完全可以基于干净的数据集、更简单的算法（如线性回归）来解决问题
- 记住奥卡姆剃刀准则

奥卡姆剃刀准则

简约至上

“The explanation requiring the fewest assumptions is most likely to be correct”

"解释能力相同情况下，假设越少越好”

奥卡姆剃刀

- Occam's Razor
- 14世纪逻辑学家，奥卡姆的威廉（William of Occam）提出
- "切勿浪费较多东西，去做'用较少的东西，同样可以做好的事情'

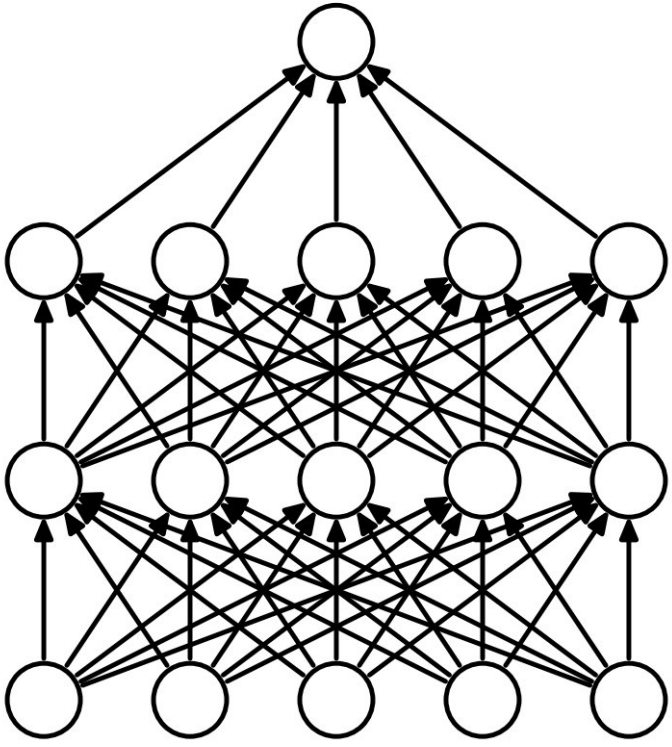
奥卡姆剃刀

- 关于同一个问题有许多种理论，每一种都能作出同样准确的预言，那么挑选其中使用假定最少的
- 尽管越复杂的方法通常能做出越好的预言，但是在不考虑预言能力（即结果大致相同）的情况下，假设越少越好
- 在结果大致相同的情况下，模型越简单越好

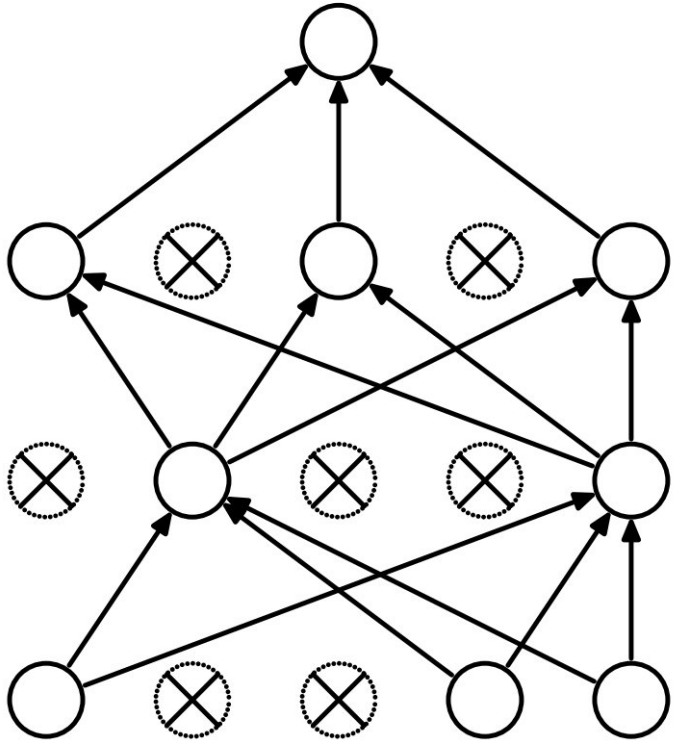
3）训练方法

Dropout

- 每轮优化中，随机选择部分神经元进行计算
- 防止某些神经元特别厉害，一股独大



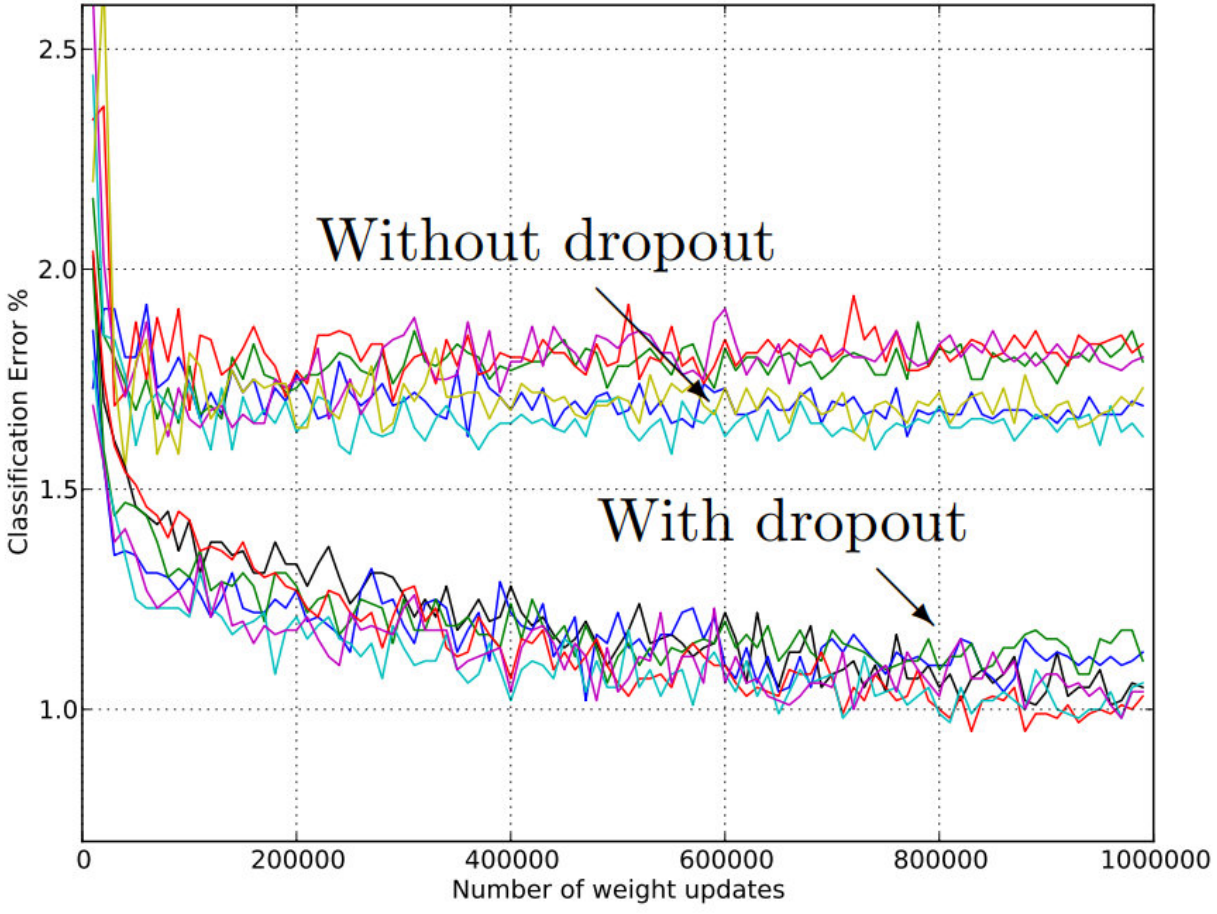
(a) Standard Neural Net



(b) After applying dropout.

Dropout

- 大家拾柴火焰高



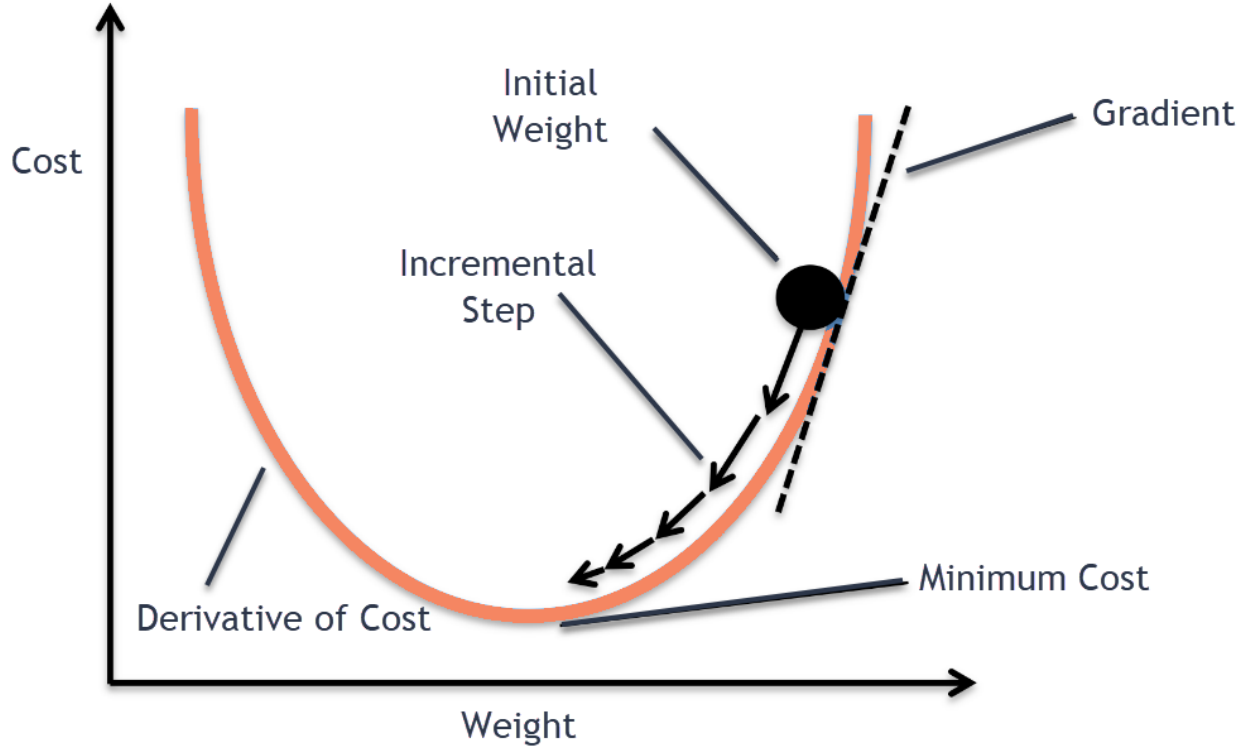
4）优化

Optimization

寻找合适的 ，令模型错误最小

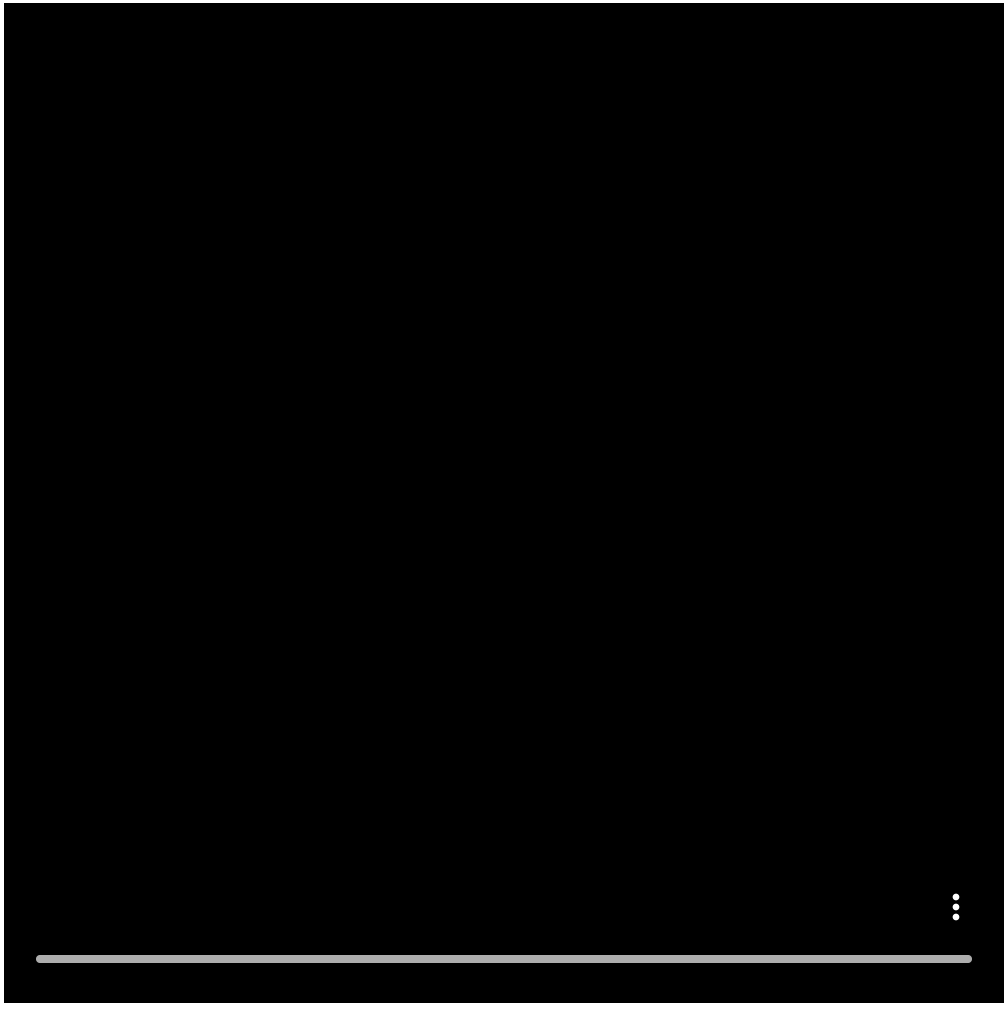
梯度下降寻找错误最小的模型权重

- 斜率为正，减少权重
- 斜率为负，增加权重

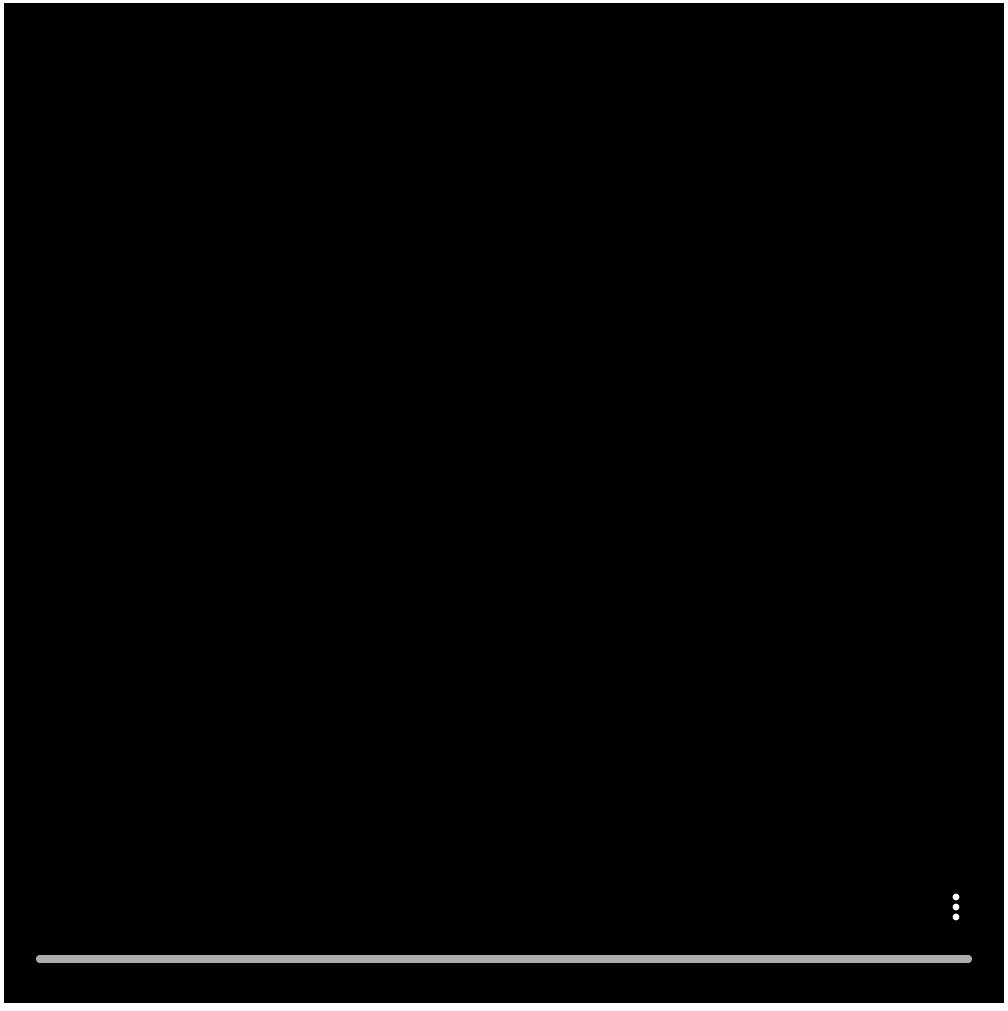


权重

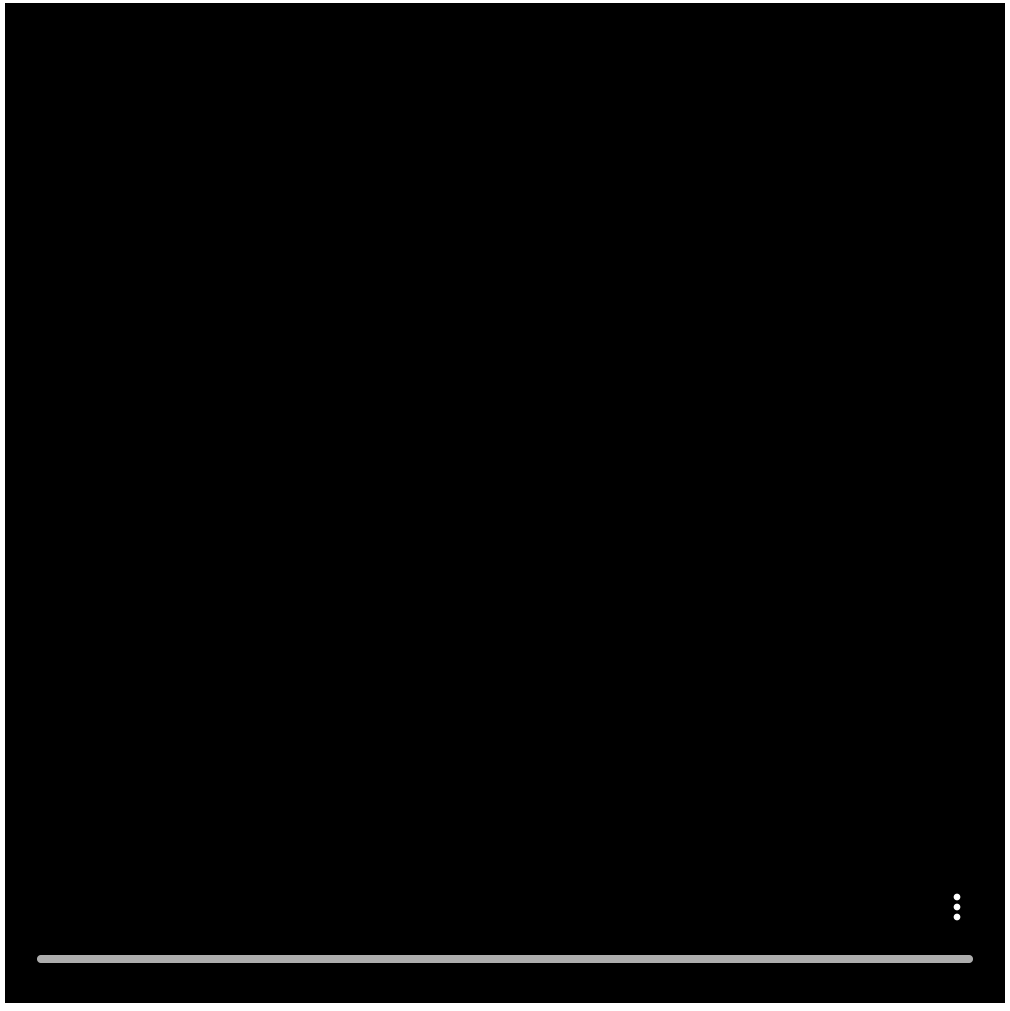
梯度下降过程



梯度下降过程



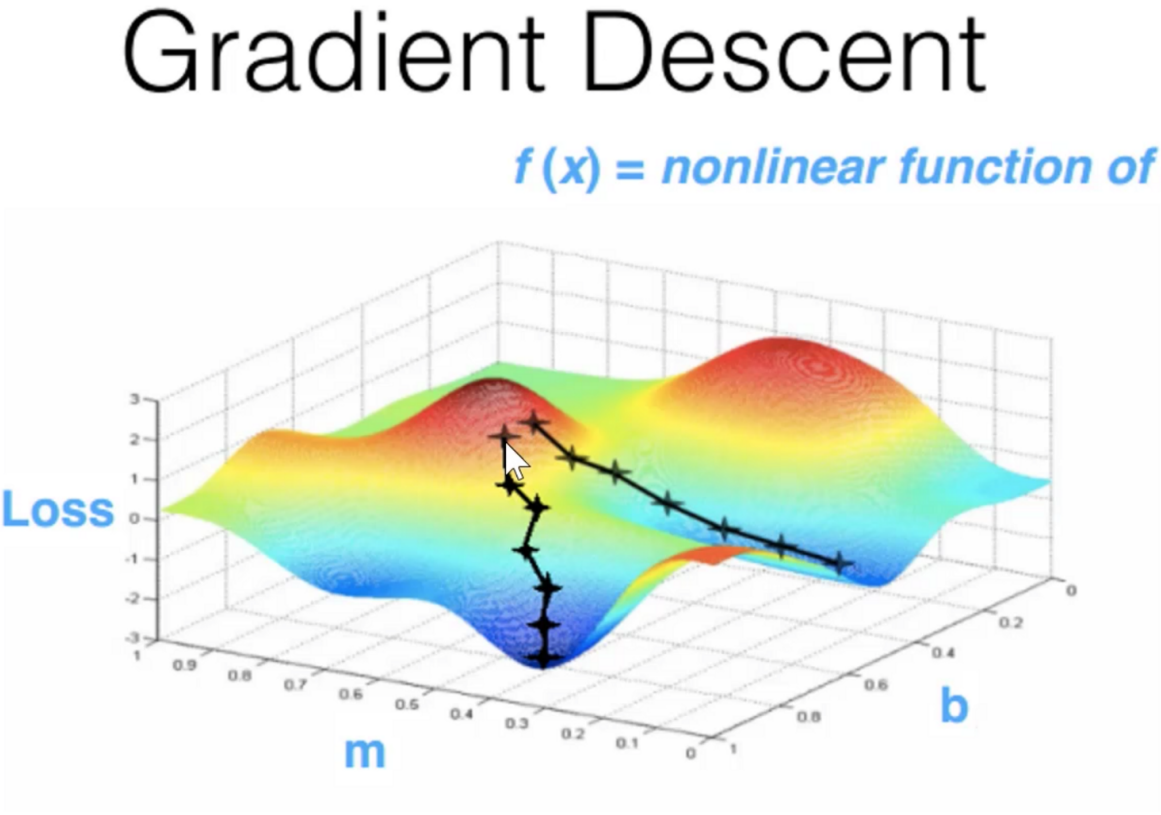
自适应步长选择



5）调参

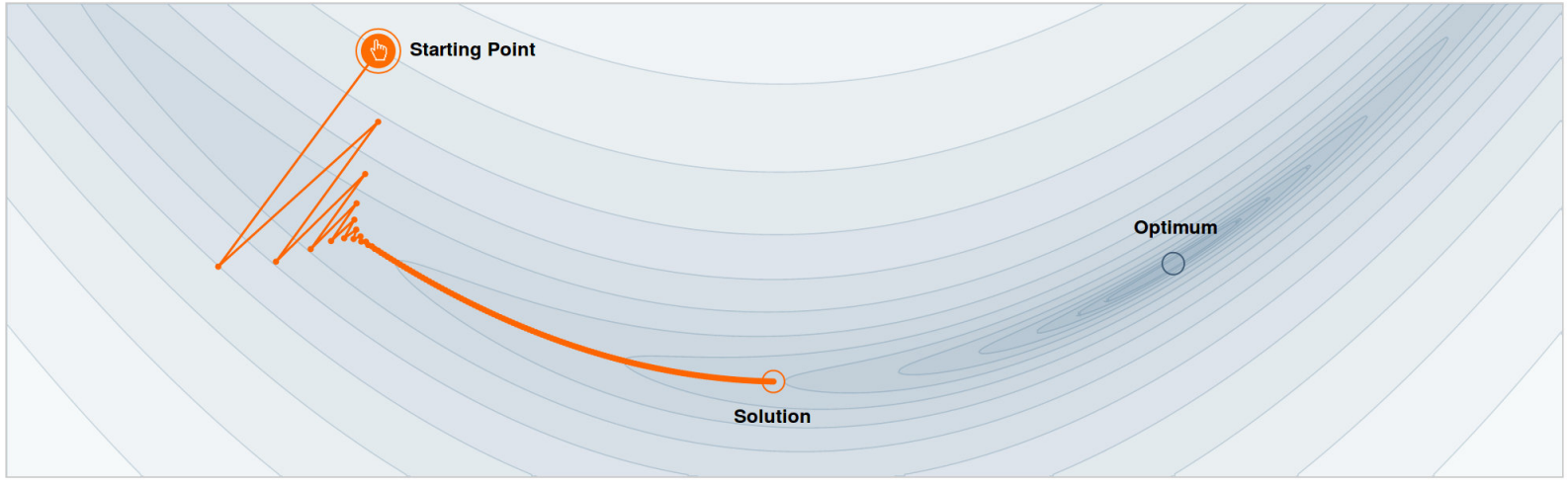
参数影响模型性能

高维下情况复杂



自适应步长参数选择

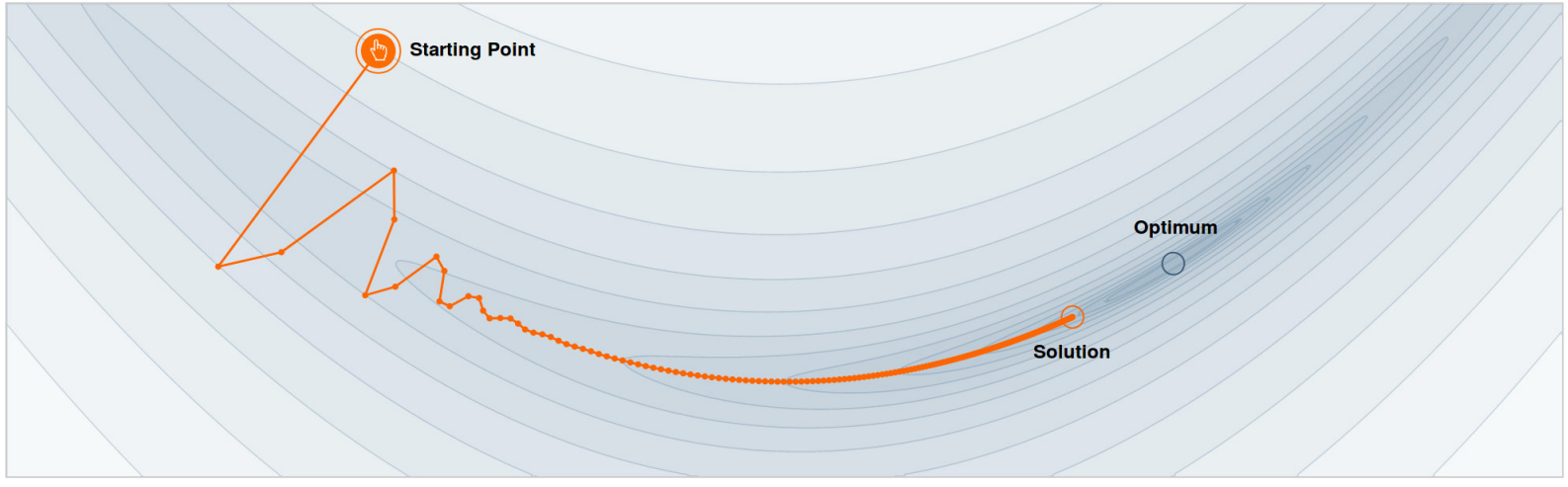
Momentum（冲量）



未达最优点

自适应步长参数选择

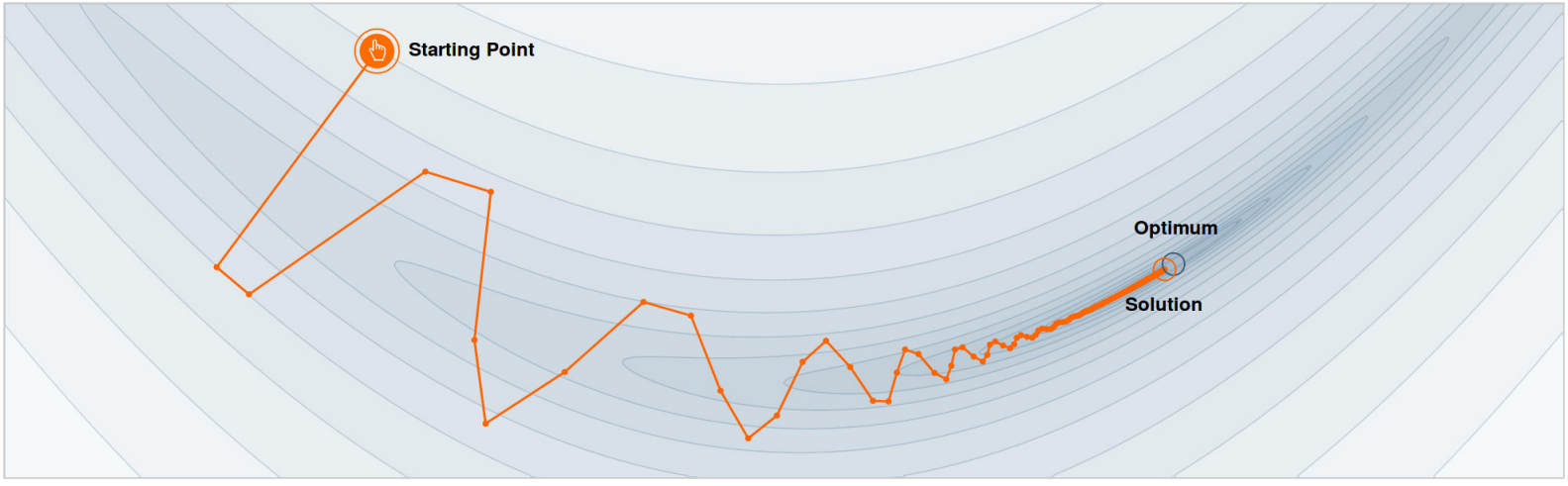
Momentum（冲量）



未达最优点

自适应步长参数选择

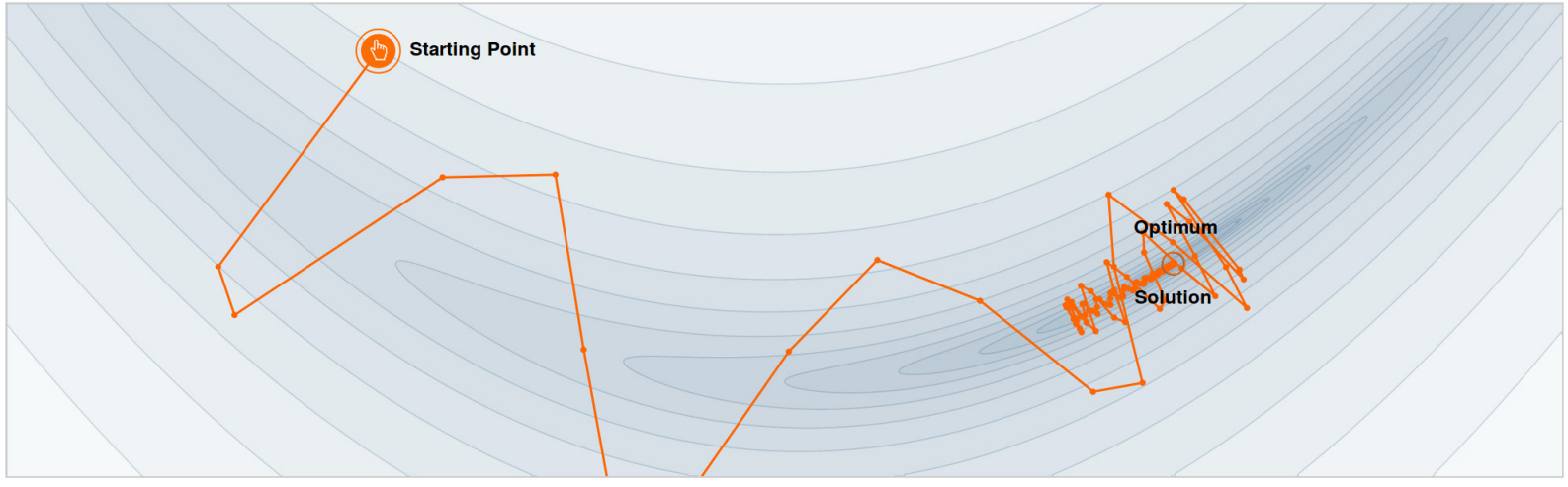
Momentum（冲量）



到达最优点

自适应步长参数选择

Momentum（冲量）



到达最优点，但震荡

调参非常重要

1. 随机初始化模型
2. 用该模型进行预测
3. 将预测结果和真实结果比对：如果错误，调整模型
4. 重复第2-3步，直到性能无法提升
5. 在验证集上验证，选择最好的模型参数

复习题I

- 什么是有监督学习？什么是无监督学习?
- 图片分类是有监督学习，还是无监督学习?
- 聚类是有监督学习，还是无监督学习?
- 线性回归模型是直线还是S曲线?
- Logistic回归模型的是直线还是S曲线?
- 感知机由哪两部分组成?

复习题II

- 三种最典型的深度神经元网络，分别是什么?
- 模型的能力不够，会过拟合还是欠拟合?
- 模型的能力太强，会过拟合还是欠拟合?
- 什么是奥卡姆剃刀原则?