行为识别小组周报

王晋东

2016.11.10

深度学习的一些概念

- 参考资料
  - Hinton在Coursera的公开课
  - LeCun Y, Bengio Y, Hinton G. Deep learning[J]. Nature, 2015, 521(7553): 436-444.
  - Bengio Y, Goodfellow I J, Courville A. Deep learning[J]. MIT. 2015.



目录

- Coursera公开课介绍
  - 公开课设置
  - 目前我的学习进度
- 一些关键点
  - 神经网络的学习模式
  - 深度网络的分类与发展
  - MP神经元、感知机
  - 损失函数与激活函数
  - BP
  - 学习率、冲量等

COURSERA公开课

- Coursera
  - Neural Networks for Machine Learning
  - 16学时，每学时4-5段10分钟左右的视频，每节课1个quiz，约7个问题。每3课1个编程作业，10个问题。

序号	内容	序号	内容
1	介绍	9	规范化
2	感知机	10	联合网络
3	BP	11	Hopfield/Boltzmann
4	预测词汇	12	RBM
5	CNN与物体识别	13	DBN
6	如何使学习更快	14	预训练
7-8	RNN	15	层次化模型

- Coursera was co-founded by Andrew Ng.

学习模式

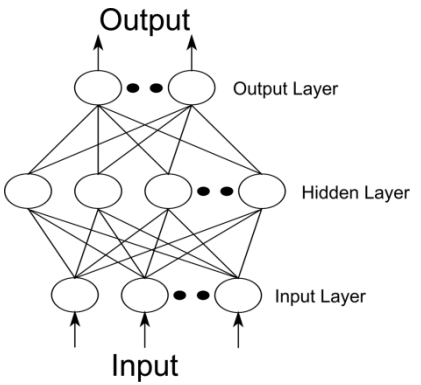
- 从机器学习的视角看神经网络
  - 终极目标：$f(x;\theta) \rightarrow f^{\star}(x;\theta)$
  - 学习 f 有多种方法，复合形式：

$$f(x) = f^{(3)}\left(f^{(2)}\left(f^{(1)}(x)\right)\right)$$

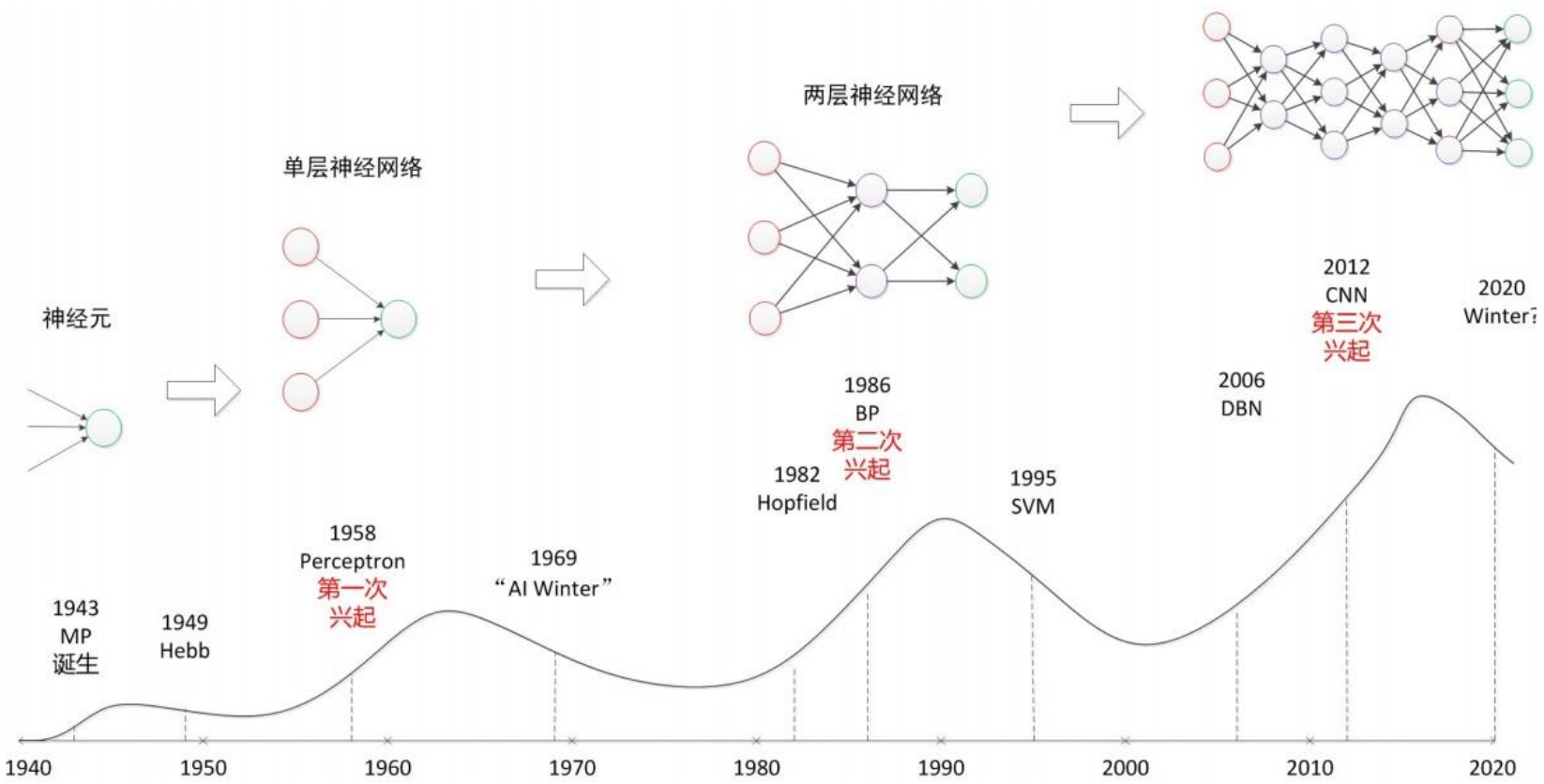
  - 恰恰是一种神经网络的形式！
  - 把不同功能的函数按照一定层次进行复合
  - Directed Acyclic Graph (DAG)

$$\textit{线性模型}(LR) \rightarrow x$$

$$\textit{非线性模型} \rightarrow \theta(x)$$



分类与发展


神经元
单层神经网络
两层神经网络
1943
MP
诞生
1949
Hebb
1958
Perceptron
第一次
兴起
1969
“AI Winter”
1982
Hopfield
1986
BP
第二次
兴起
1995
SVM
2006
DBN
2012
CNN
第三次
兴起
2020
Winter?
1940
1950
1960
1970
1980
1990
2000
2010
2020

近年大动作

2010年，美国国防部DARPA计划首次资助深度学习项目。

2011年，微软研究院和谷歌的语言识别研究人员先后采用DNN技术降低语音识别错误率20%-30%，是该领域10年来最大突破

2012年，Hinton将ImageNet图片分类问题的Top5错误率由26%降低至15%。同年Andrew Ng与Jeff Dean搭建Google Brain项目，用包含16000个CPU核的并行结算平台训练超过10亿个神经元的深度网络，在玉莹识别和图像识别领域取得突破性进展。

2013年，Hinton创立的DNN Research公司被Google收购，Yann LeCun加盟Facebook的人工智能实验室。

2014年，G[illegible]精准度从2012年的84%提升到如今的98%，移动端Android系统的语言识别正确率提高了25%。人脸识别方面，Google的[illegible]别系统Fa[illegible]在LFW上达到99[illegible]3%的准确率。

2015年，[illegible]soft采用[illegible]经网络的残差学习方[illegible]将ImageNet的分类错误率降低至[illegible]7%，已低于同类试验中人眼识别的错误率5.1%，其采用的[illegible]层。

2016年，DeepMind使用了1920个CPU集群和280个GPU的深度学习围棋软件AlphaGo战胜人类围棋冠军李世石。

国内对深度学习的研究也在不断加速：

2012年，华为在香港成立"诺亚方舟实验室"从事自然语言处理、数据挖掘与机器学习、媒体社交、人际交互等方面的研究。

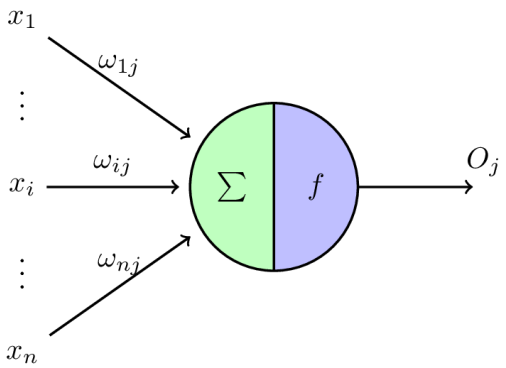
2013年，百度成立"深度学习研究院"（IDL），将深度学习应用于语言识别和图像识别、检索，2014年，Andrew Ng加盟百度。

2013年，腾讯着手建立深度学习平台Mariana，Mariana面向识别、广告推荐等众多应用领域，提供默认算法的并行实现。

2015年，阿里发布包含深度学习开放模块的DTPAI人工智能平台。

MP神经元与感知机

- MP（McCulloch-Pitts）模型1943



$$z = b + \sum_{i=1} w_i x_i$$

$$y = \begin{cases} 1 & \text{if } z \geqslant 0 \\ 0 & \text{otherwise} \end{cases}$$

- 感知机（Perceptron）1958
  - 遇到不匹配则调整权重

$$w(j) := w(j) + \alpha(y - f(x))x(j) \quad (j = 1, \ldots, n)$$

  - 只能进行线性二类分类

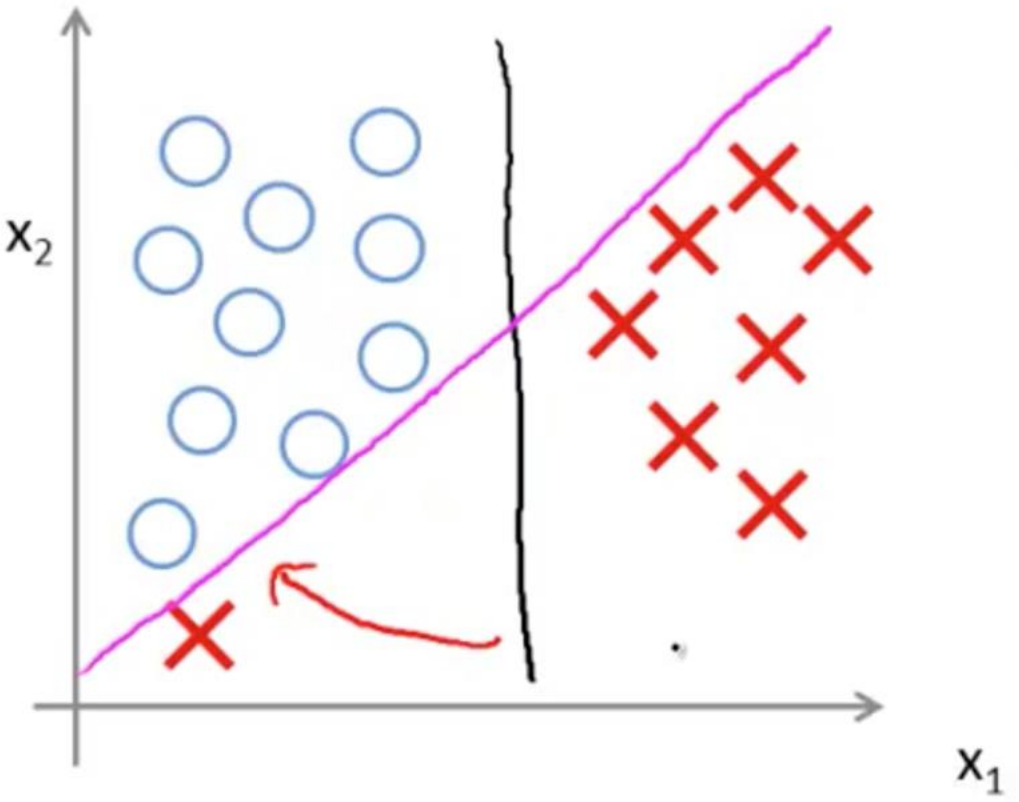
感知机

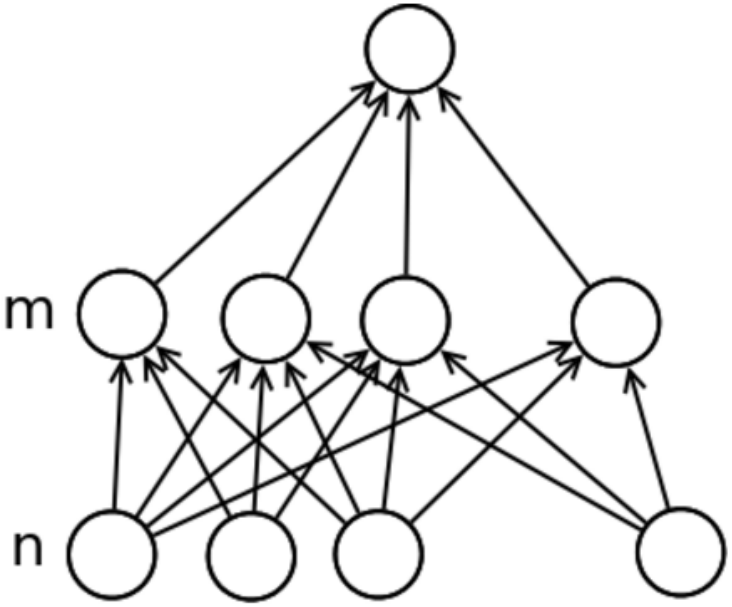
- 感知机权重调整

u)

- SVM

- Wha
  - 损失
  - 分类





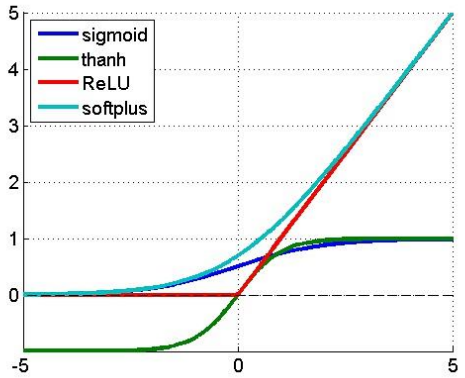
- [ ] There is a value (of at least 1) for m, such that there are functions that this network cannot learn to compute and that a network without a hidden layer (with the same inputs) can learn to compute.
- [ ] Any function that can be computed by such a network can also be computed by a network without a hidden layer.
- [ ] A network with $m > n$ can learn functions that a network with $m \leq n$ cannot learn.
- [ ] A network with $m > n$ has more learnable parameters than a network without any hidden layers (with the same inputs).

激活函数

- 映射（编码）输出的函数
  - Linear
  - Binary
  - **ReLU**（Rectified Linear Unit）
  - Sigmoid
  - Stochastic binary neurons
  - **Softmax**
- Why ReLU and Softmax popular?

WHY RELU AND SOFTMAX

- Why ReLU
  - Tanh和sigmoid都会出现梯度消失的问题
  - 计算速度快（只有线性、比较操作）
  - ReLU会使一部分结果为0，使网络更稀疏
- Why Softmax
  - 很多分类问题需要求概率映射
  - 梯度消失的问题（1-0.0000001？）



$$\phi_i = \frac{e^{\eta_i}}{\sum_{j=1}^{k} e^{\eta_j}}$$

损失函数

- 衡量模型预测与实际值的误差
  - 平方误差

$$E = \frac{1}{2}\sum_i (t_i - f_i)^2$$

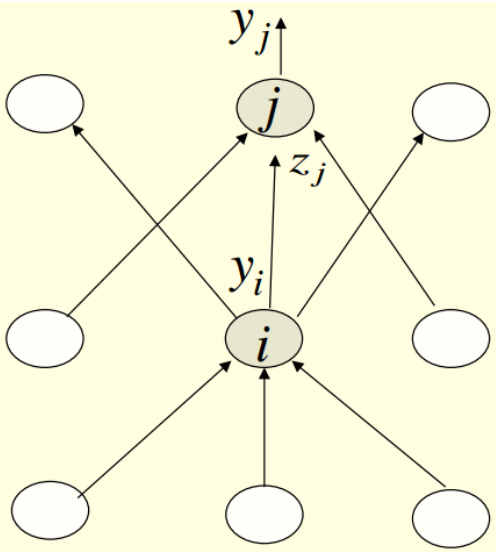
  - Cross-entropy

$$H(p,q) = -\sum_i p_i \log q_i = -y\log\hat{y} - (1-y)\log(1-\hat{y})$$

  - Why Cross-entropy?
    - 当预测值和真实值的差异巨大时，E将会有特别大的梯度
    - 对后面的BP很方便

BP

- Why BP?
  - FP过程当然可以，然而代价太大



$$\frac{\partial E}{\partial z_j} = \frac{dy_j}{dz_j}\frac{\partial E}{\partial y_j} = y_j(1-y_j)\frac{\partial E}{\partial y_j}$$

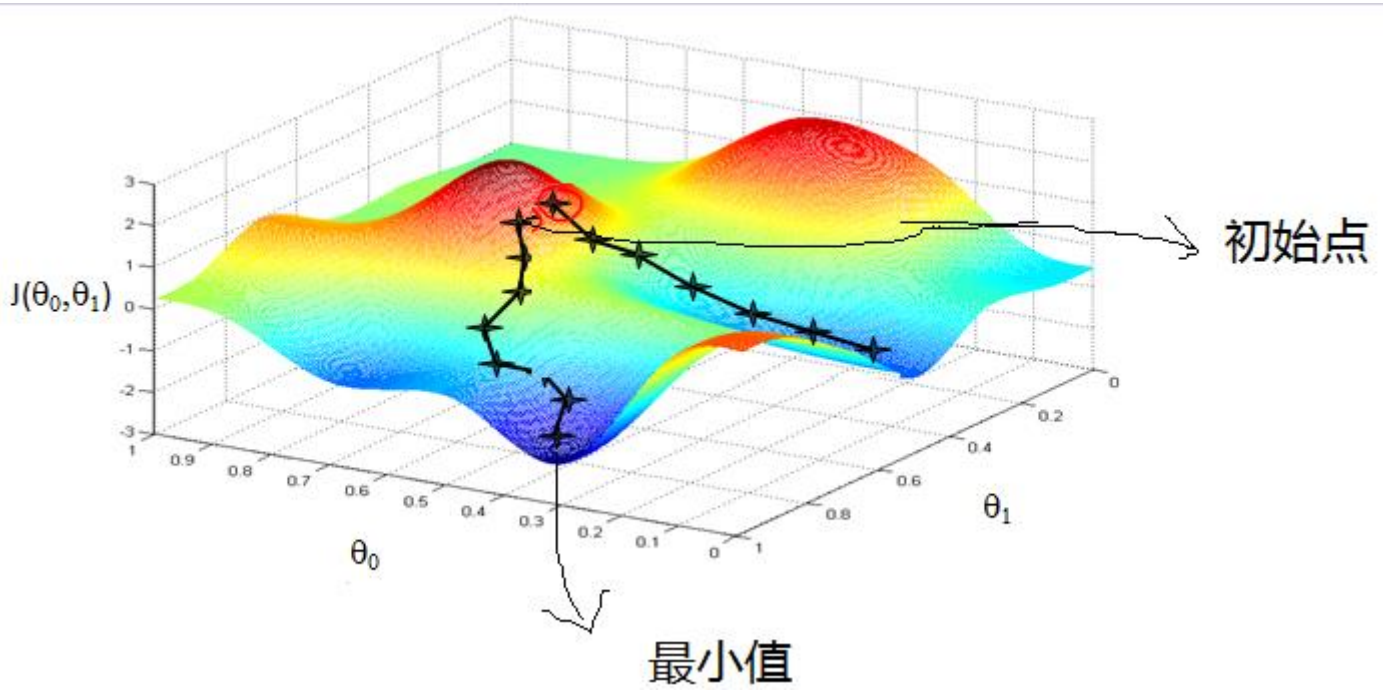
$$\frac{\partial E}{\partial y_i} = \sum_j \frac{dz_j}{dy_i}\frac{\partial E}{\partial z_j} = \sum_j w_{ij}\frac{\partial E}{\partial z_j}$$

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial z_j}{\partial w_{ij}}\frac{\partial E}{\partial z_j} = y_i\frac{\partial E}{\partial z_j}$$

- Cross-entropy: $\frac{\partial C}{\partial z_i} = \sum_j \frac{\partial C}{\partial y_j}\frac{\partial y_j}{\partial z_i} = y_i - t_i$

梯度下降

- Gradient decent
  - Stochastic gradient decent (SGD)
  - Batch gradient decent (BGD)
  - Mini-batch gradient decent (BGD)



MOMENTUM

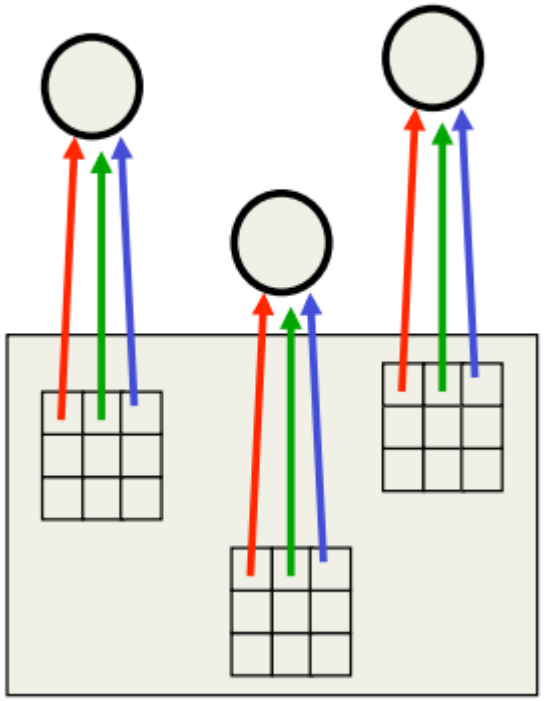
- 类似于物理学中的冲量
- 牛顿第二定律 $F = m \cdot a$
  - 把GD想像成小球下山，一开始是往下落的，然而，当小球有速度之后，由于冲量的作用，但不一定会一直沿着梯度下降的方向落
- 人为加入momentum来控制方向

$$v = \gamma v + \alpha \nabla_\theta J(\theta; x^{(i)}, y^{(i)})$$
$$\theta = \theta - v$$

CNN

- 什么是卷积？ $$y(t) = x(t) * h(t) = \sum_{\tau=-\infty}^{\infty} x(\tau)h(t-\tau)$$
  - 局部感受野
  - 权值共享
  - 池化

The red connections all have the same weight.



$1_{\times 1}$	$1_{\times 0}$	$1_{\times 1}$	0	0
$0_{\times 0}$	$1_{\times 1}$	$1_{\times 0}$	1	0
$0_{\times 1}$	$0_{\times 0}$	$1_{\times 1}$	1	1
0	0	1	1	0
0	1	1	0	0

Image

4		

Convolved Feature