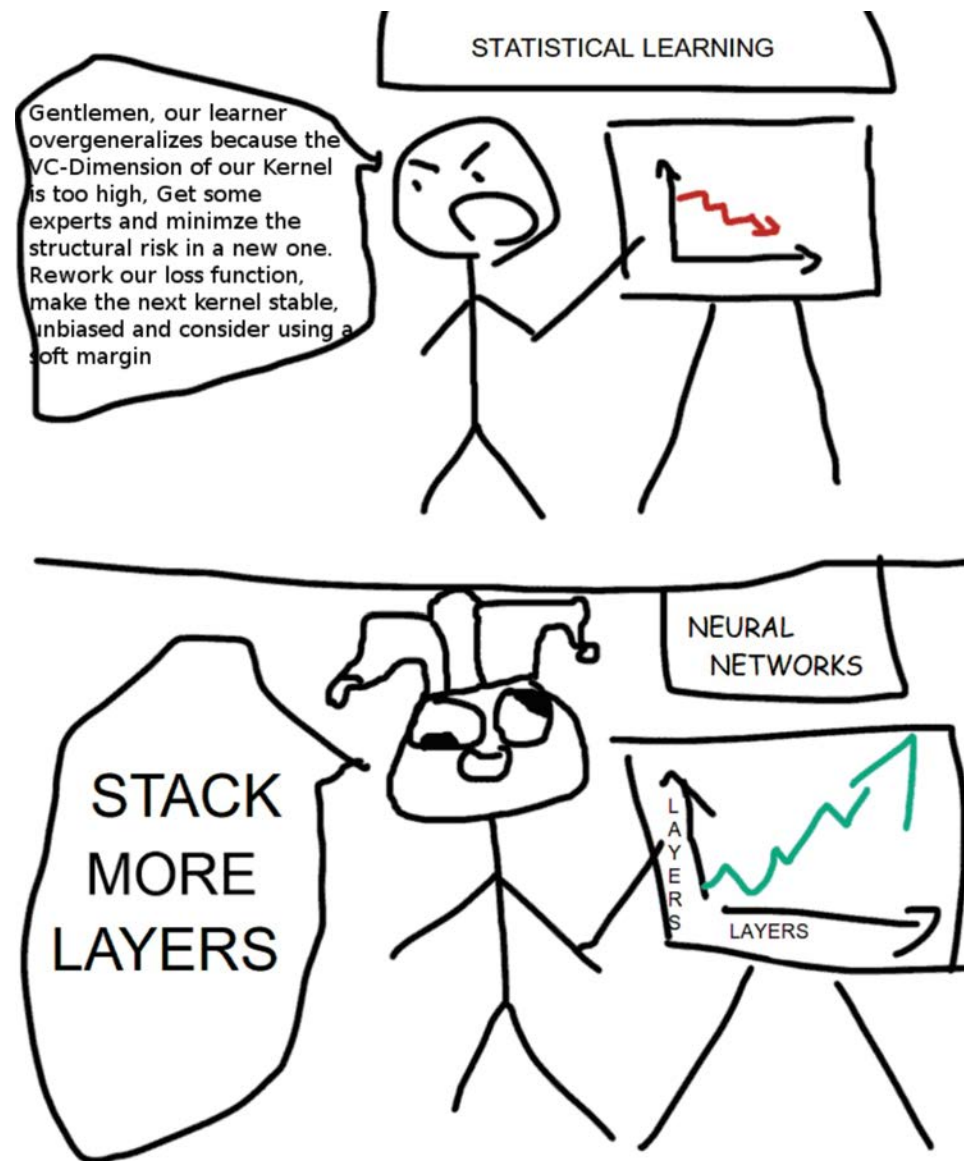




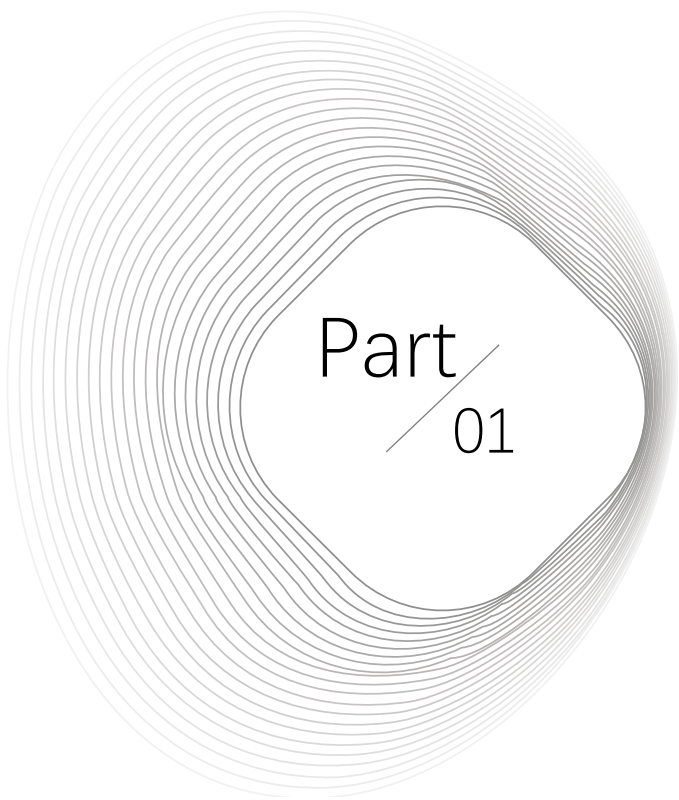
$2^{0.2}$
3

欢迎来到深度学习

我们会学些什么？



- 一些主流网络
 - 浅层网络
 - CNN、RNN、Auto Encoder、GAN
 - Attention、Transformer、BERT、GPT
- 掌握原理、回避数学
- 关键性的思想方法、欣赏智慧
- 在金融中的应用，但还是要靠想象力
- 编程？

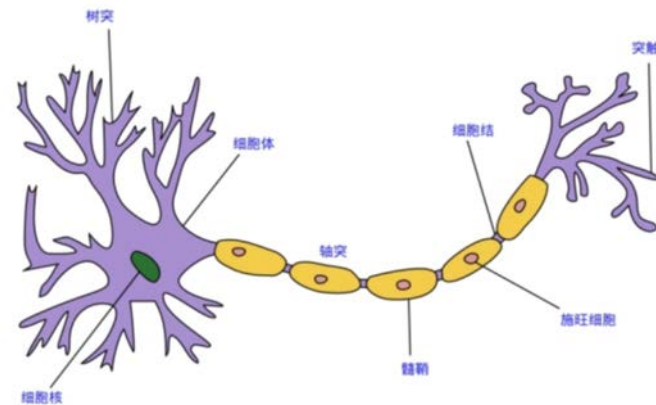


Part
01

简单神经网络

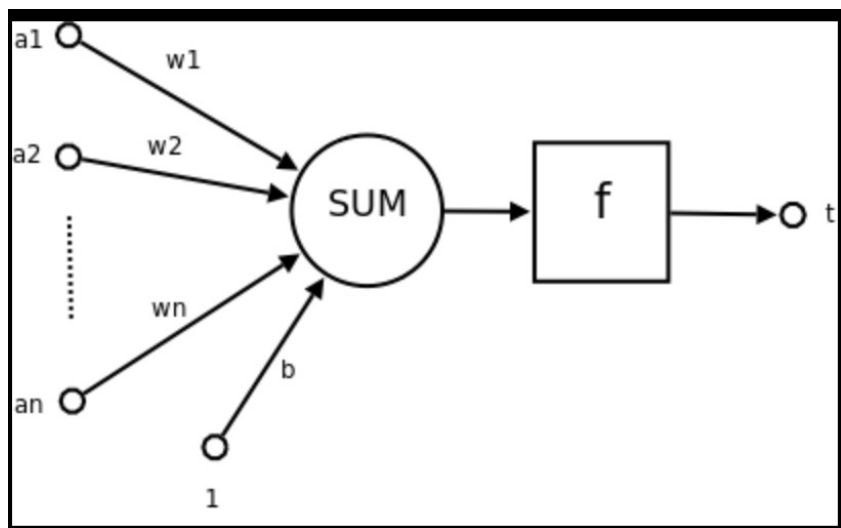
- 从感知机到神经网络
- 如何训练
- 应用：Word2Vec

感知机：一个老朋友

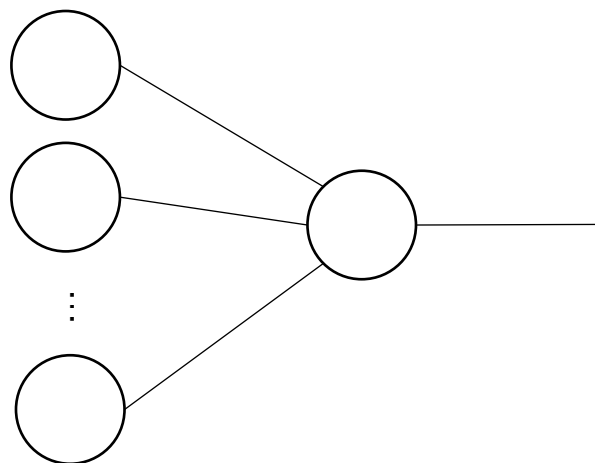


- 人工神经网络最初在1943年提出，1949年赫布型学习
- Frank Rosenblatt (1957) 发明的一种简单线性分类器

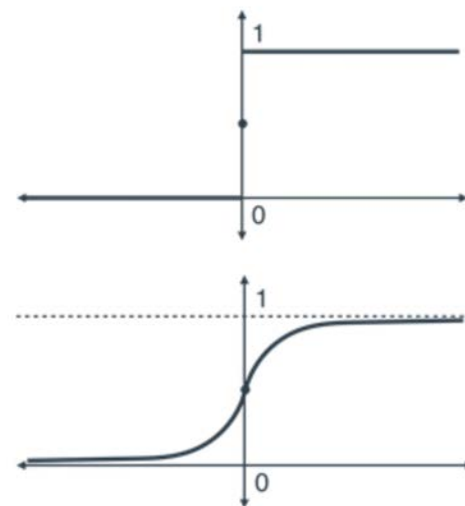
我们可以假定，反射活动的持续与重复会导致神经元稳定性的持久性提升……当神经元A的轴突与神经元B很近并参与了对B的重复持续的兴奋时，这两个神经元或其中一个便会发生某些生长过程或代谢变化，致使A作为能使B兴奋的细胞之一，它的效能增强了。



$$t = f\left(\sum_{i=1}^n w_i x_i + b\right) = f(\mathbf{w}^T \mathbf{x})$$



$$f(n) = \begin{cases} +1 & \text{if } n \geq 0 \\ -1 & \text{otherwise} \end{cases}$$

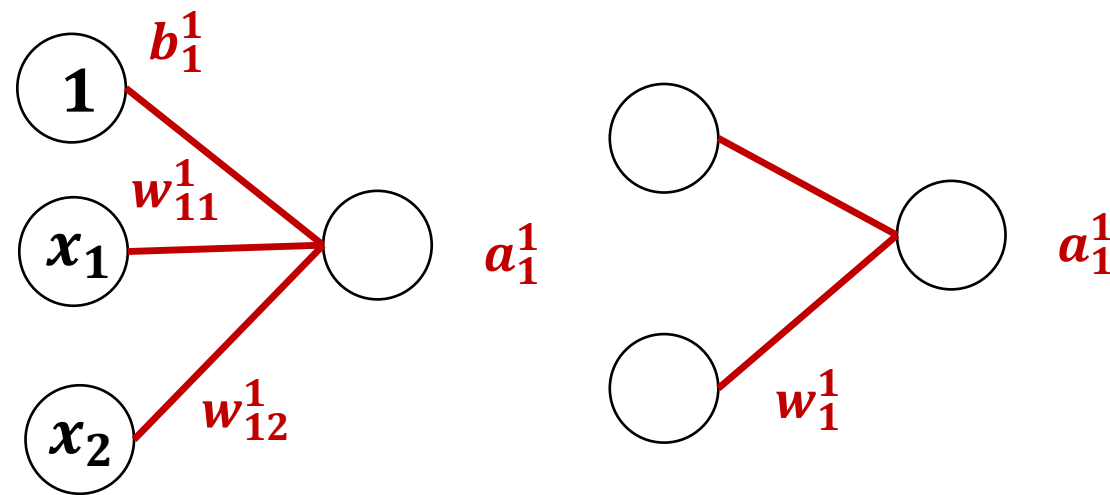
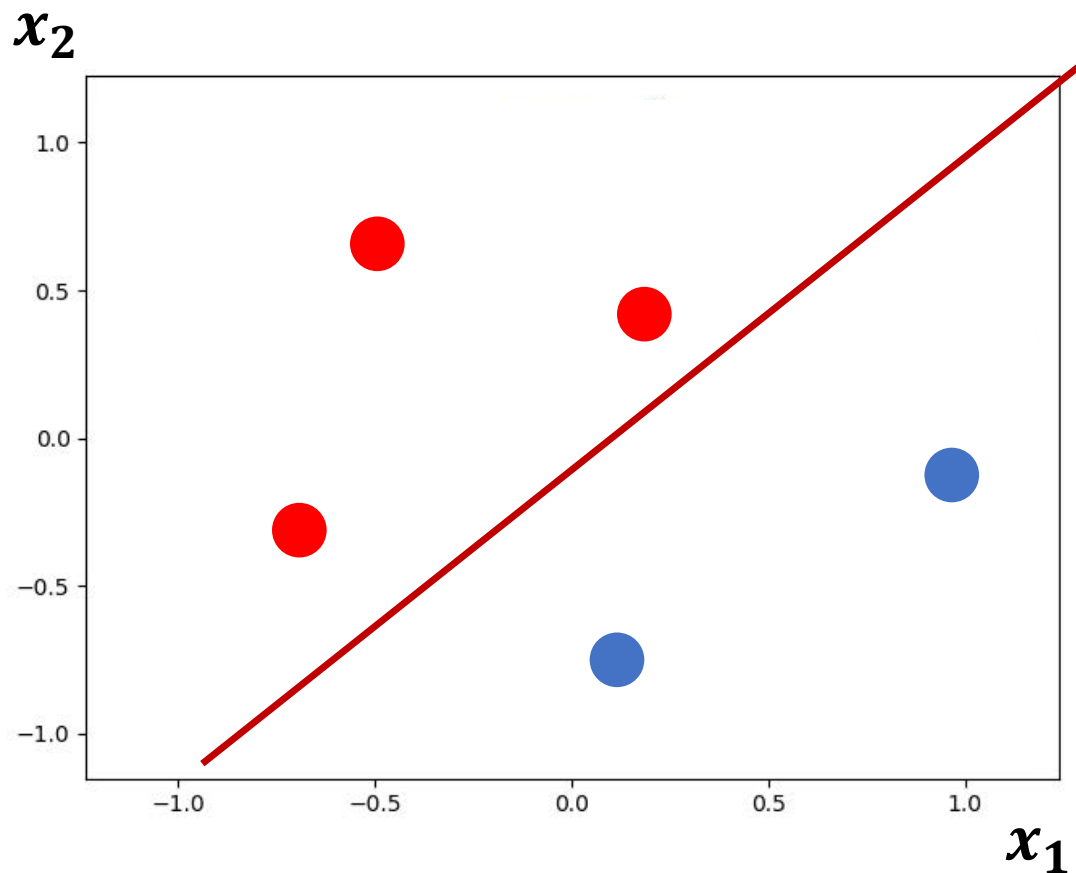


$$f(x) = \frac{1}{1 + e^{-x}}$$

sigmoid

1.1

感知机：分割平面

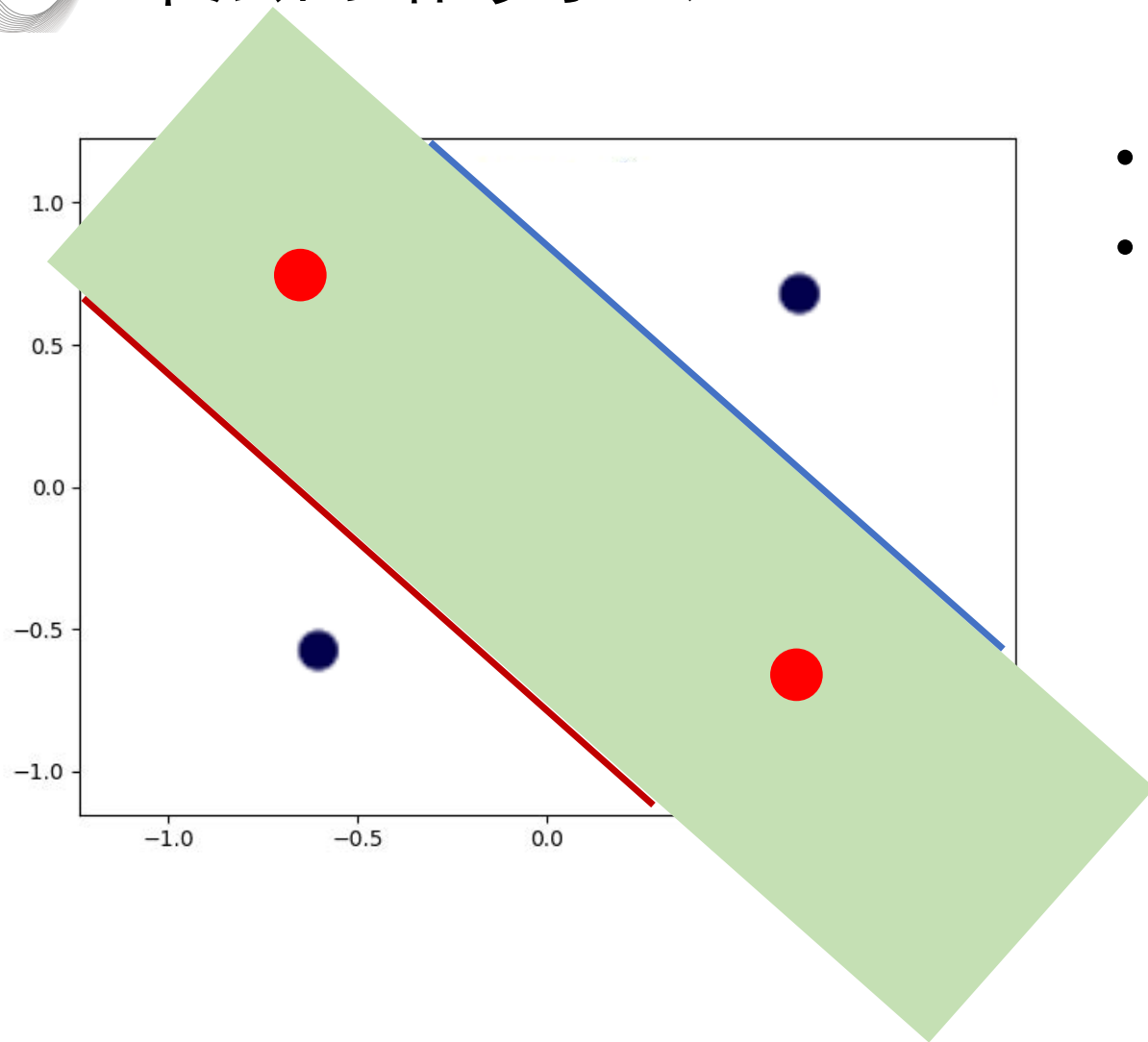


$$a_1^1 = \text{sigmoid}(w_{11}^1 x_1 + w_{12}^1 x_2 + b_1^1)$$

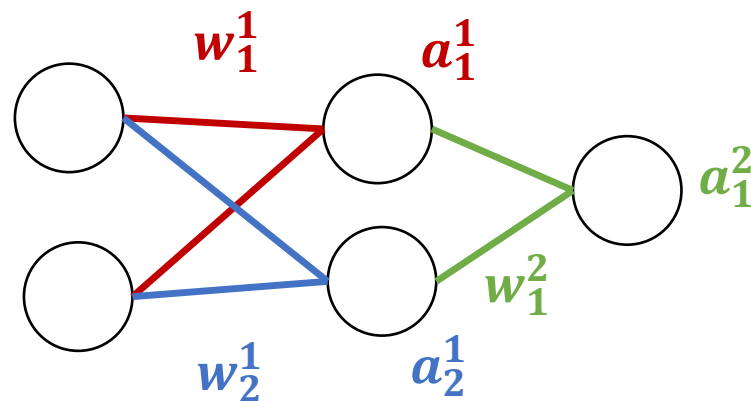
$$a_1^1 = \text{sigmoid}(w_1^1 x)$$

1.1

单层网络够了么？



- 异或问题，线性不可分
- 将红色和蓝色的结果，组合判断

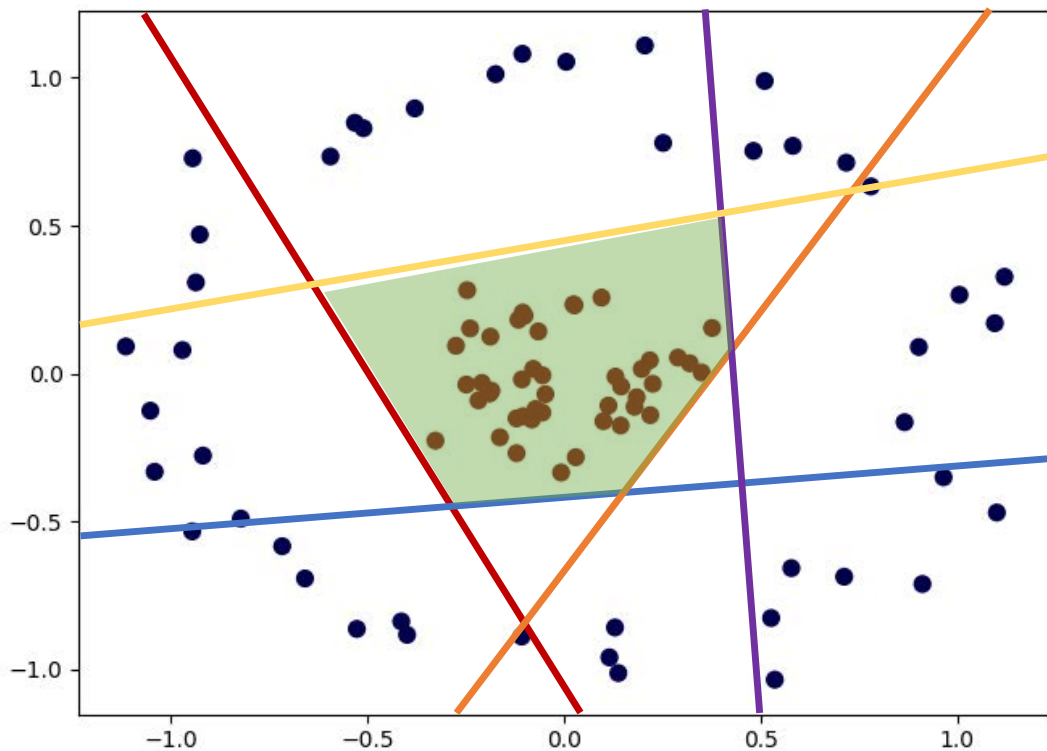


$$a_1^1 = \text{sigmoid}(w_1^1 x)$$

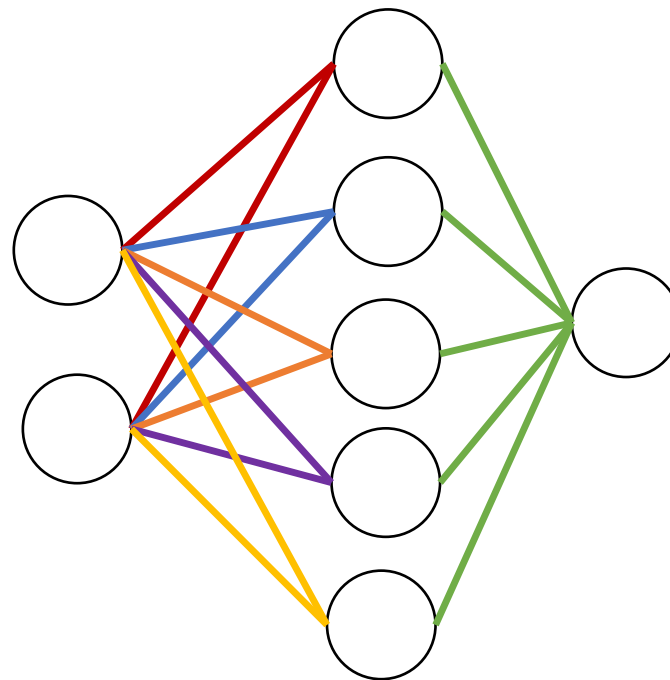
$$a_2^1 = \text{sigmoid}(w_2^1 x)$$

$$a_1^2 = \text{sigmoid}(w_1^2 a^1)$$

集成学习+感知机 = ?



- 并联多个感知机，切分原本难以切分的
- 而且，似乎多多益善？



$$a_1^1 = \text{sigmoid}(\mathbf{w}_1^1 \mathbf{x})$$

$$a_2^1 = \text{sigmoid}(\mathbf{w}_2^1 \mathbf{x})$$

$$a_3^1 = \text{sigmoid}(\mathbf{w}_3^1 \mathbf{x})$$

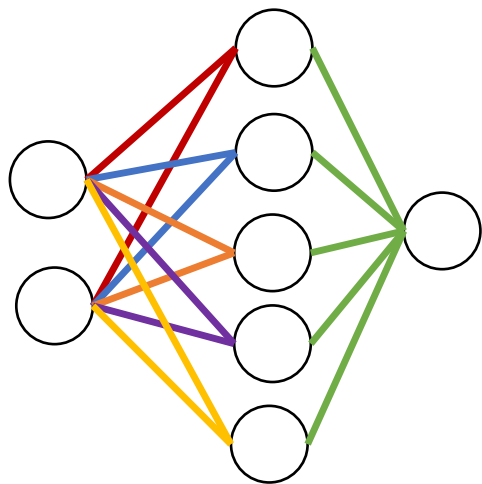
$$a_4^1 = \text{sigmoid}(\mathbf{w}_4^1 \mathbf{x})$$

$$a_5^1 = \text{sigmoid}(\mathbf{w}_5^1 \mathbf{x})$$

$$a_1^2 = \text{sigmoid}(\mathbf{w}_1^2 a^1)$$

1.1

更广、更深、更强



input layer

hidden layer 1

hidden layer 2

hidden layer 3

output layer

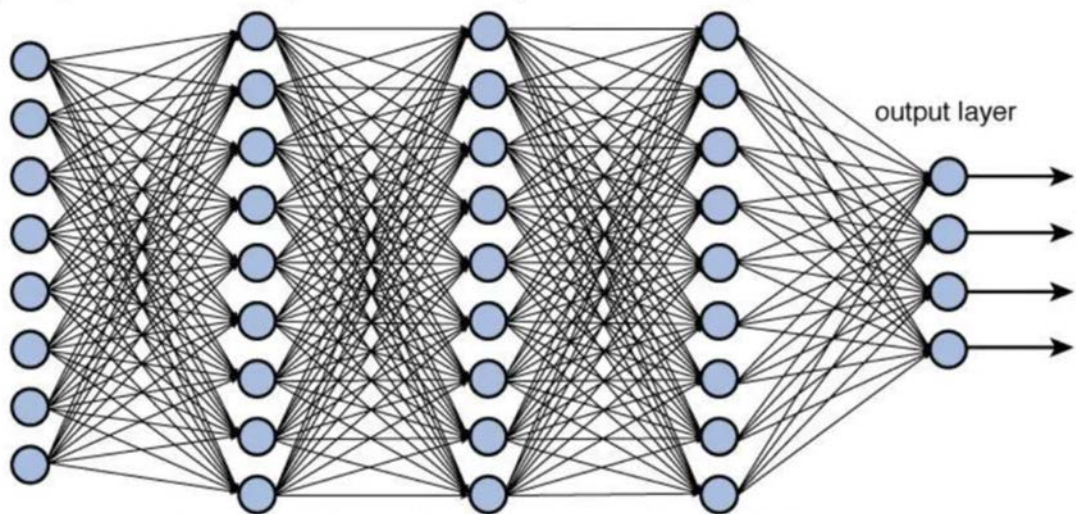
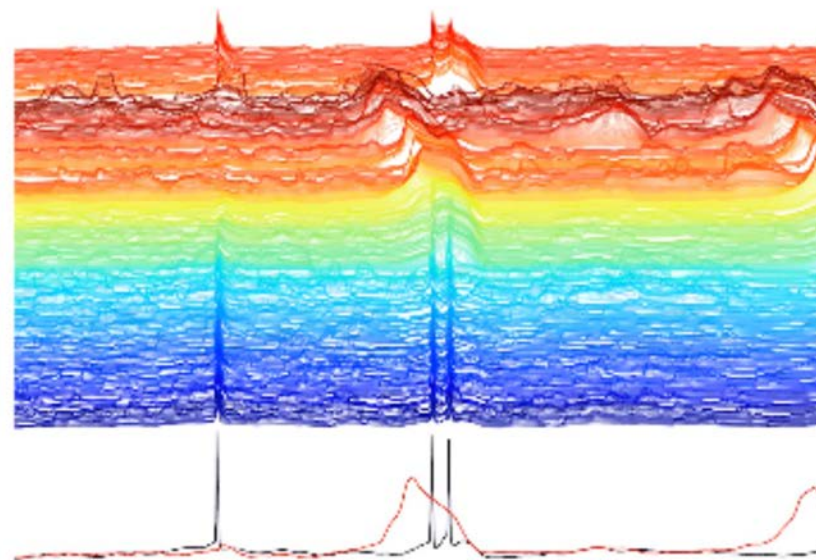


Figure 12.2 Deep network architecture with multiple layers.



"If the human brain were so simple that we could understand it, we would be so simple that we couldn't."

— Emerson M. Pugh



建网一时爽，推导火葬场

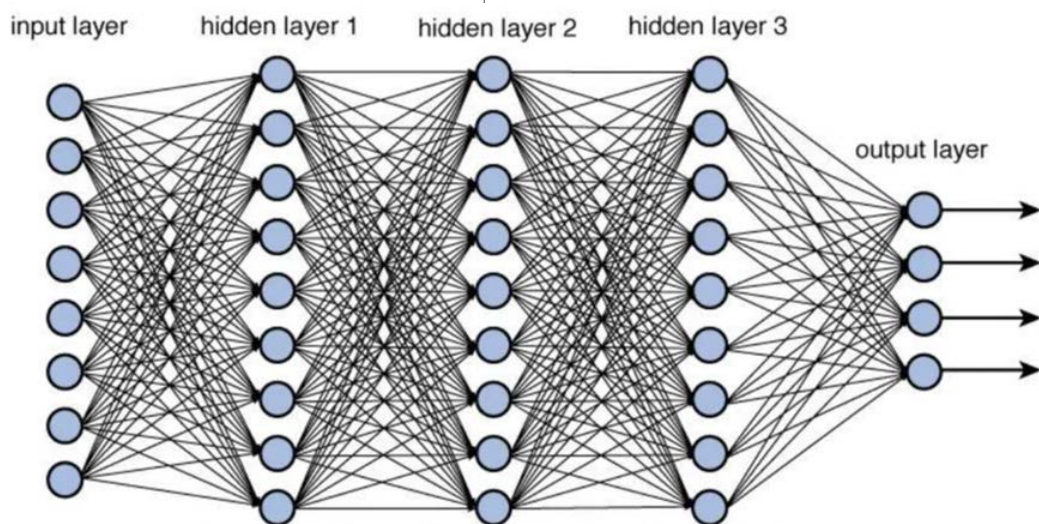
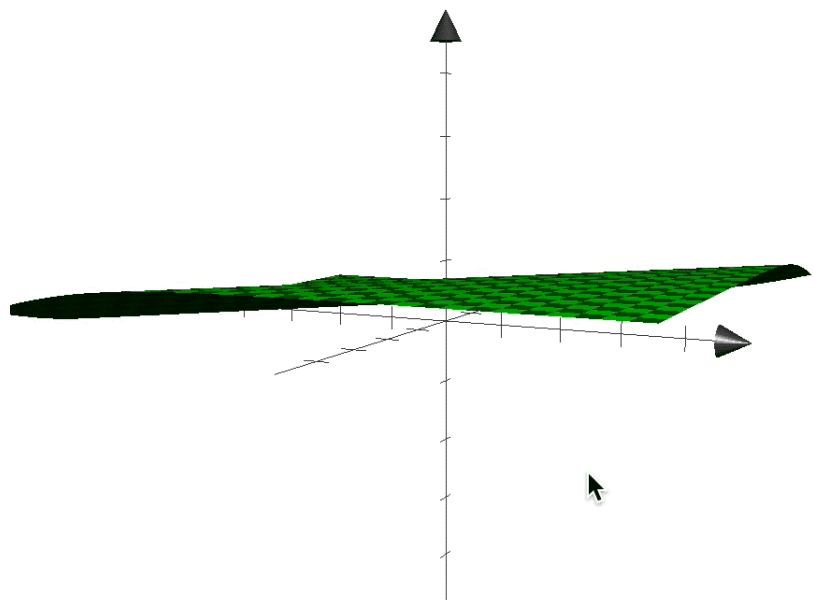
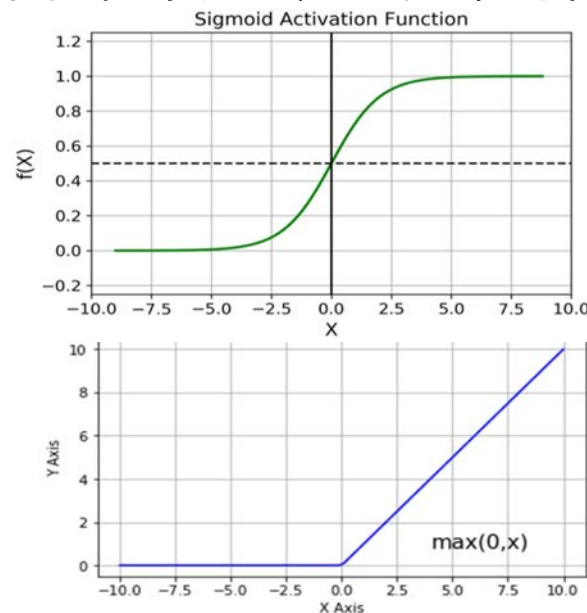


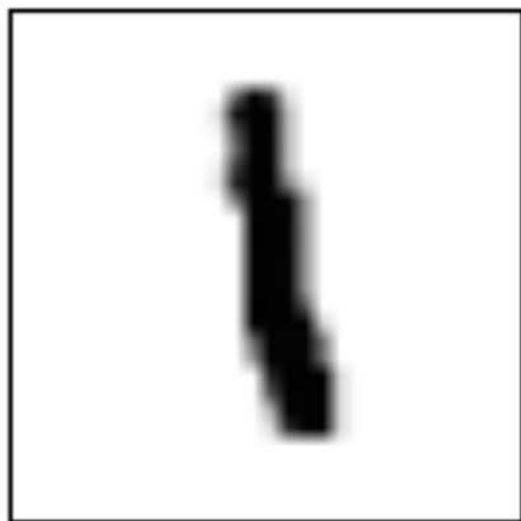
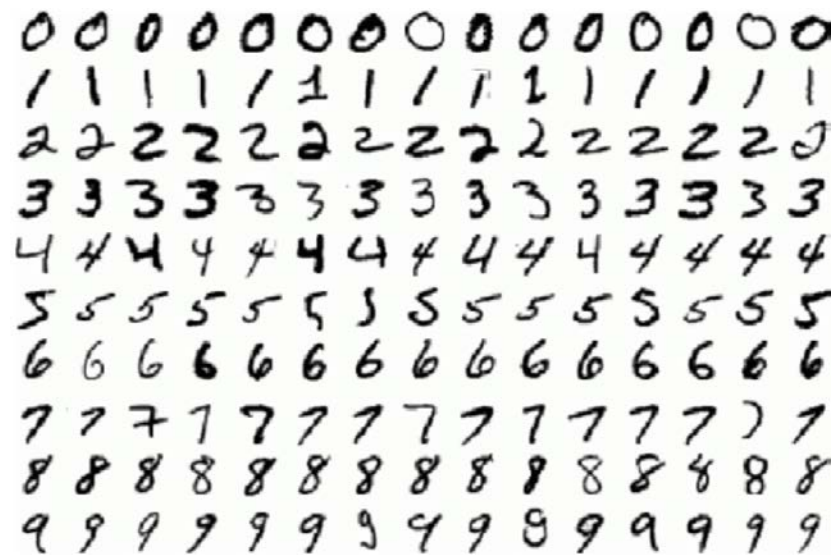
Figure 12.2 Deep network architecture with multiple layers.

- 在构想中，我们可以由前向后构建一个网络，计算权重，但是现实呢？
- 类似之前，我们可以根据最终预测结果的误差，回头来进行优化
- 需要计算的：
 - w 权重
 - $s^n = w^n a^n$ 输入信号
 - $a^n = \text{sigmoid}(s^n)$ 输出信号
- 为什么要加一个sigmoid函数？
- $\text{sigmoid}(x) = \frac{1}{1+e^{-x}}$
- $\text{Relu}(x) = \max(0, x)$
- $\text{softmax}: \sigma(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}}$; \mathbf{z} is a vector of K real numbers



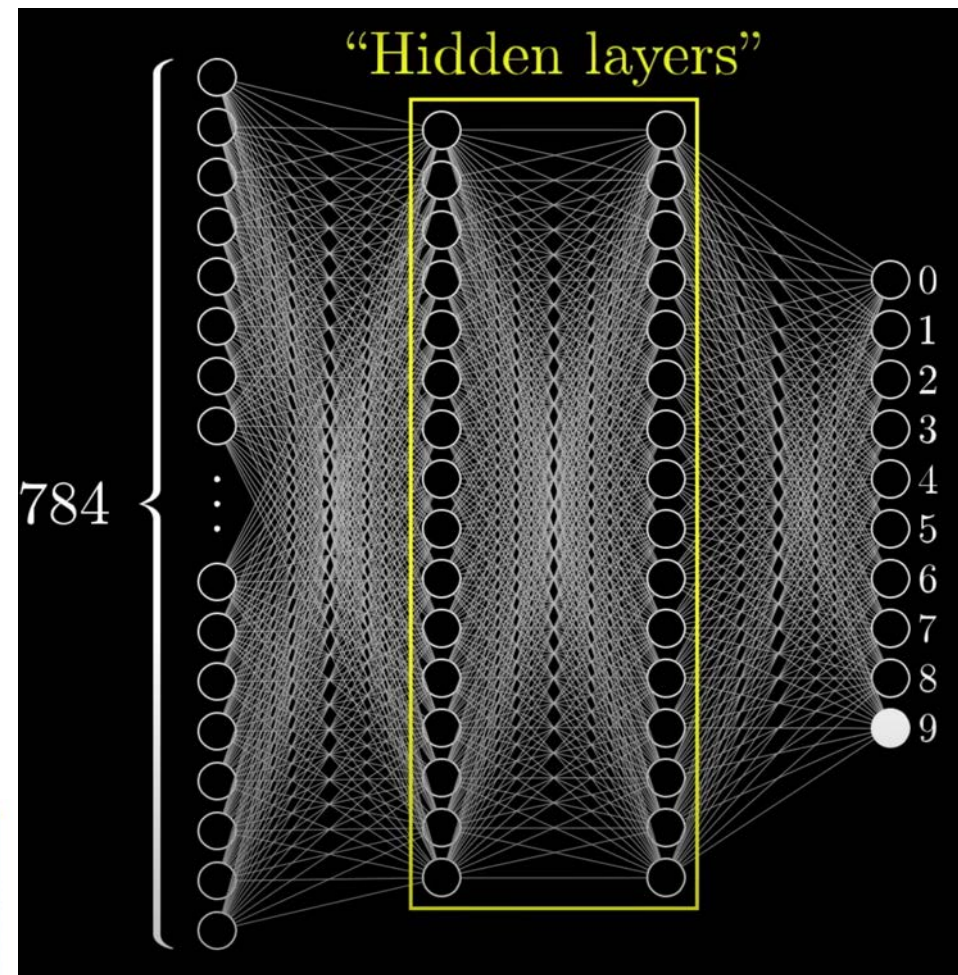
1.2

神经网络：手写体识别



12

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|----|----|----|----|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | .6 | .8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | .7 | .1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | .7 | .1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | .5 | .1 | .4 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | .1 | .4 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | .1 | .4 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | .1 | .7 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | .1 | .1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | .9 | .1 | .1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | .3 | .1 | .1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |



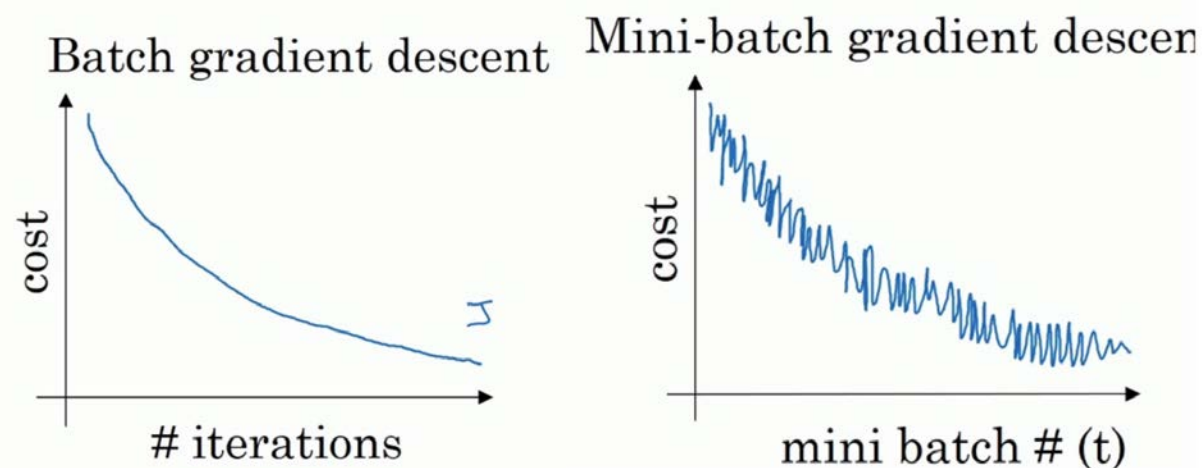
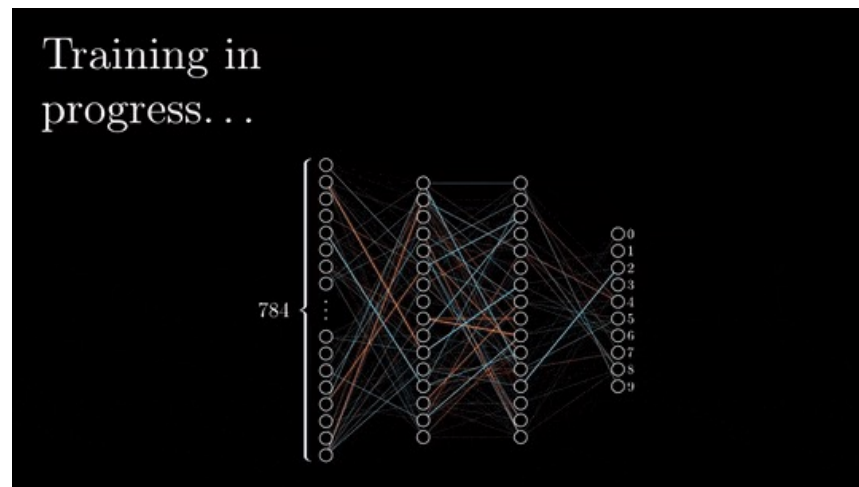
3B1B

<https://www.bilibili.com/video/BV16x411V7Qg>

1.2

反向传播 (Back Propagation) 算法

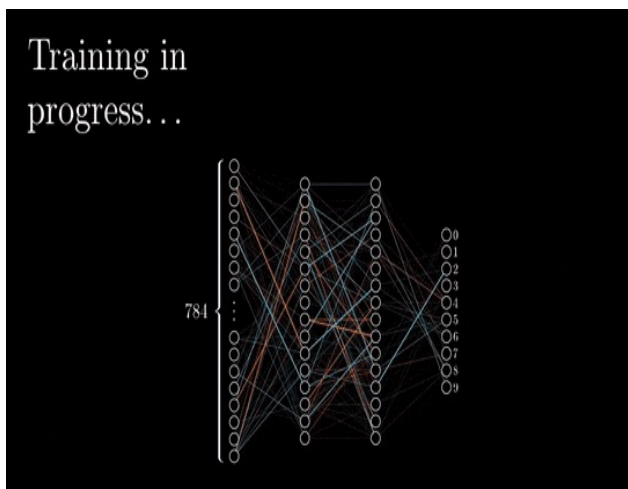
- Rumelhart, Hinton & Williams (1986)
- 算法流程
- 初始化权重 \mathbf{w} (整张网络)
- 训练过程分为 $t = 0, 1, 2, \dots, T$ 期
 - 1. 随机挑选: 随机挑选一组数据 $\mathbf{x}_{(n)}, y_{(n)}$
 - 2. 前向传播: 挑选数据 $\mathbf{x}_{(n)}$ 作为输入, 并向前传播直至算出网络总输出
 - 3. 反向传播: 将输出与真实值 $y_{(n)}$ 进行比较, 并根据链式法则将残差对某一个 \mathbf{w}_{ij}^l 求导
 - 4. 梯度下降: 按照减少残差的方向 (残差求导的负方向) 更新 \mathbf{w}_i^l
- 迭代多次后, 将最终的 \mathbf{w}_{ij}^l 作为权重进行构建网络
- 多数情况下, 1-3步会 (并行) 一起做多次 **mini-batch**
- 优化 \mathbf{w} 的过程道阻且长, 充满不确定性
- work but hard, 做了许多许多年的机器学习“守门员”



神经网络的正则

- 限制权重
 - 网络的能力来自于权重 w ，对于 $|w|$ 的限制我们做了很多次
 - $\min(L(w)) \rightarrow \min(L(w) + \frac{1}{2}|w|^2)$
- 加入白噪音
 - 白噪音 $N(0,1)$ ，如果 $L(w)$ 是 MSE ?
- 在更多位置加入噪音?
 - 输入层可以看做是一种特殊的隐藏层
 - 在隐藏层上加噪音?
 - Dropout 将一些隐藏层手动归零（丢失）
- 对于梯度下降的优化
 - Mini batch
 - Early stopping (epoch not iteration)
 - Learning rate





- 网络中的边，权重
- 网络连在一起，矩阵计算
- 矩阵很大，计算量很大，计算/存储需求
- 怎样计算呢？



从只需要1个面。二者变化的节点，便是所需渲染的草丛与视点距离，预设一个阈值，当草丛距离小于它的时候，使用垂直BillBoard；大于它的时候，使用简单平面。那么，针对不同性能环境的复杂度调节，一个重要方式就是改变复杂度变化的阈值。

绘制策略的另一个重要方面是避免重复性带来的问题，由于总体的大面积草坪是由数量众多的草丛重复拼接而成，所以若不加以处理直接绘制，则会



本章将介绍进行结果。此外，案。

5.1 实例化应用

由于大面积出，如果不使用在应用了实例化：在本文的实用一次DrawIndex序仍可流畅运行图5.1、图5.2为i



图 5.1 半空视角观察草坪

图 5.3 实例化技术性能比较

在目前的漫游程序中，仅有草地的绘制使用了实例化算法，这一方面是因

Encoding的局限：没有额外的信息表达

只是一个编号，有没有办法让这些信息有更多的意义？



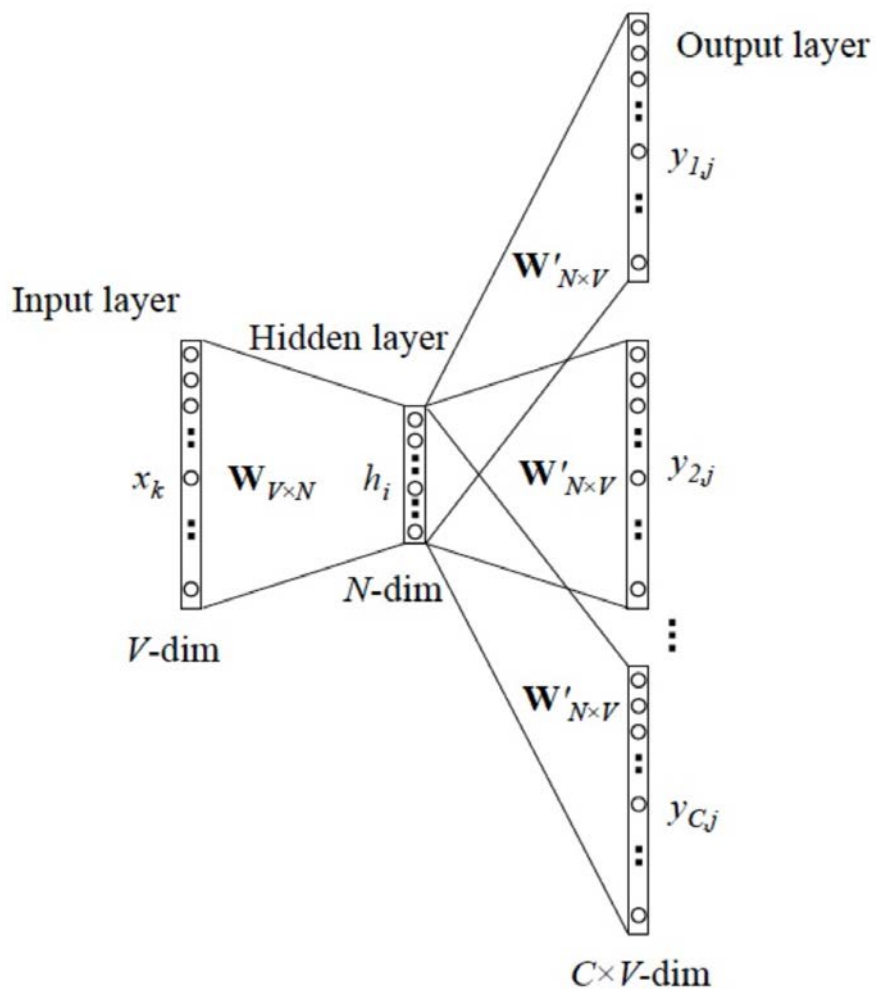
My name is Li Hua . I'm thirteen. I'm a middle school student . I am 1 Class Five, Grade One. My English teacher 2 Mr Lin. He is 3 old teacher. I 4 a pen, a ruler and two 5 in my pencil-box. I have a bike 6.

Liu Ping is in my class. She is a girl. We 7 good friends . She is not 8 today. I think she is at home. Look ! Here is a 9 . But it is not my book, I think it is 10 book.

- | | | | |
|---------------------|--------------|---------------|---------------|
| () 1. A. in | B. at | C. do | D. not |
| () 2. A. are | B. am | C. is | D. × |
| () 3. A. a | B. an | C. this | D. very |
| () 4. A. am | B. think | C. know | D. have |
| () 5. A. boxes | B. pencils | C. buses | D. desk |
| () 6. A. too | B. or | C. much | D. very |
| () 7. A. have | B. am | C. are | D. all |
| () 8. A. right | B. where | C. at home | D. at school |
| () 9. A. licence | B. book | C. picture | D. map |
| () 10. A. Liu Ping | B. Liu Pings | C. Liu Ping's | D. Liu Pings' |

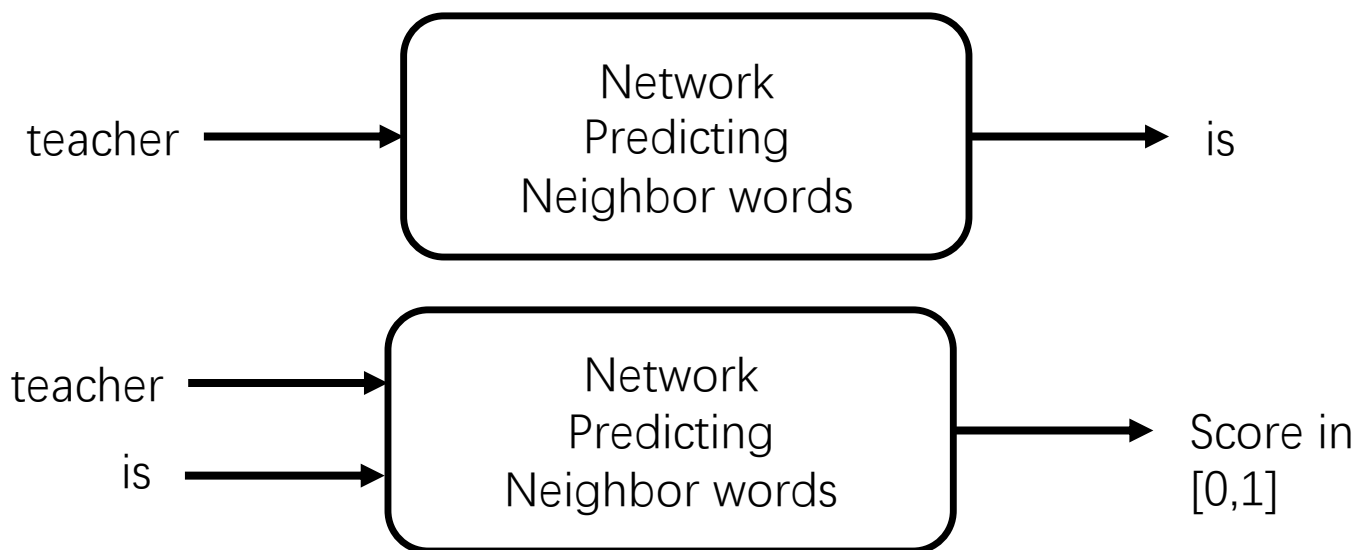
将上下文信息用于词语理解

• My English teacher _____ Mr. Lin. He is



- 输入一个单词的one-hot编码, x_{ik} 为一个 V 维向量
- 输出为 C 个 V 维向量, 表示该单词的上下文 C 个单词
- 中间隐藏层为人为确定的 N 维
- 训练完成之后, 输入某个单词的one-hot, 此时隐藏层的 N 维向量就是词向量
- 我们可以认为词向量是该单词的一个好的Embedding
- Embedding (嵌入) 比Encoding (编码) 保留更多信息
- 直觉解释: 从隐藏层恢复上下文

- My English teacher **is** Mr. Lin. He is



| Now word | Next word | score |
|----------|-----------|-------|
| Teacher | Is | 1 |
| Teacher | Mr. | 0 |
| Teacher | Lin | 0 |

- 输入一个词，输出它的下一个词
- 词使用one hot编码 $\text{size}=V$
- 但是 V 可能很大
- 改为两个输入，计算匹配分数
- 那么训练样本怎么来？
- **负采样 Negative Sampling**
- 负采样多少个？ 5
- 为什么？
- 古圣先贤，习惯就好

词向量的应用

Mikolov, Tomas, et al. "Efficient estimation of word representations in vector space." *arXiv preprint arXiv:1301.3781* (2013).

王靖一,黄益平.金融科技媒体情绪的刻画与对网贷市场的影响[J].经济学(季刊),2018,17(04):1623-1650.DOI:10.13821/j.cnki.ceq.2018.03.15.
《北京大学互联网金融情绪指数》 2016.9

- 把词表示成向量，一系列优秀的性质：
- 表示复杂度首先（从V到N）
- 向量维度：50--300
- 计算相似度？
 - 找近义词、反义词、几个词组合
- 计算加减法
 - 国王-男人+女人 = 女王

• “庞氏骗局”

| rank | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|-------------|----------|----------------|--------|-----------------------|---------------|-----------------------|-----------------|------------|-------|------------------|
| word | 骗局 | 自融 | 旁氏 | 拆东墙补西墙 | 拆标 | 传销 | 击鼓传花 | 阴谋 | 谎言 | 圈钱 |
| similarity | 0.564 | 0.536 | 0.524 | 0.509 | 0.500 | 0.494 | 0.482 | 0.480 | 0.478 | 0.472 |
| translation | cheating | Self-financing | Pond's | rob Peter to pay Paul | Loan dividing | Multi-level marketing | pass the parcel | conspiracy | lie | Money collecting |

P2P网贷术语

俗语

错别字





北京大学互联网金融研究中心
Institute of Internet Finance, Peking University

北京大学互联网金融发展指数

(第三期, 2014年1月-2016年3月)

北京大学互联网金融研究中心课题组 2016年7月



北京大学互联网金融研究中心
Institute of Internet Finance, Peking University

北京大学数字普惠金融指数

(2011年-2015年)

北京大学互联网金融研究中心课题组 2016年7月



北京大学互联网金融研究中心
Institute of Internet Finance, Peking University

北京大学互联网金融情绪指数

(2013年-2016年)

北京大学互联网金融研究中心课题组 2016年7月



2⁰₂3

THANKS