生物统计学：
生物信息中的概率统计模型

2019年秋


Yann LeCun
Geoffrey Hinton
Yoshua Bengio
Andrew Ng

有关信息

- 授课教师： 宁康
  - Email: ningkang@hust.edu.cn
  - Office: 华中科技大学东十一楼504室
  - Phone: 87793041, 18627968927

2019生物统计学
扫一扫二维码，加入该群。

- 课程网页
  - http://www.microbioinformatics.org/teach/#
  - QQ群： 882140516

课程安排

- 生物背景和课程简介
- 传统生物统计学及其应用
- 生物统计学和生物大数据挖掘
  - Hidden Markov Model (HMM)及其应用
    - Markov Chain
    - HMM理论
    - HMM和基因识别 (Topic I)
    - HMM和序列比对 (Topic II)
  - 进化树的概率模型 (Topic III)
  - Motif finding中的概率模型 (Topic IV)
    - EM algorithm
    - Markov Chain Monte Carlo (MCMC)
  - 基因表达数据分析 (Topic V)
    - 聚类分析-Mixture model
    - Classification-Lasso Based variable selection
  - 基因网络推断 (Topic VI)
    - Bayesian网络
    - Gaussian Graphical Model
  - 基因网络分析 (Topic VII)
    - Network clustering
    - Network Motif
    - Markov random field (MRF)
  - Dimension reduction及其应用 (Topic VIII)
- 面向生物大数据挖掘的深度学习

研究对象:
生物序列,
进化树,
生物网络,
基因表达
...

方法:
生物计算与生物统计

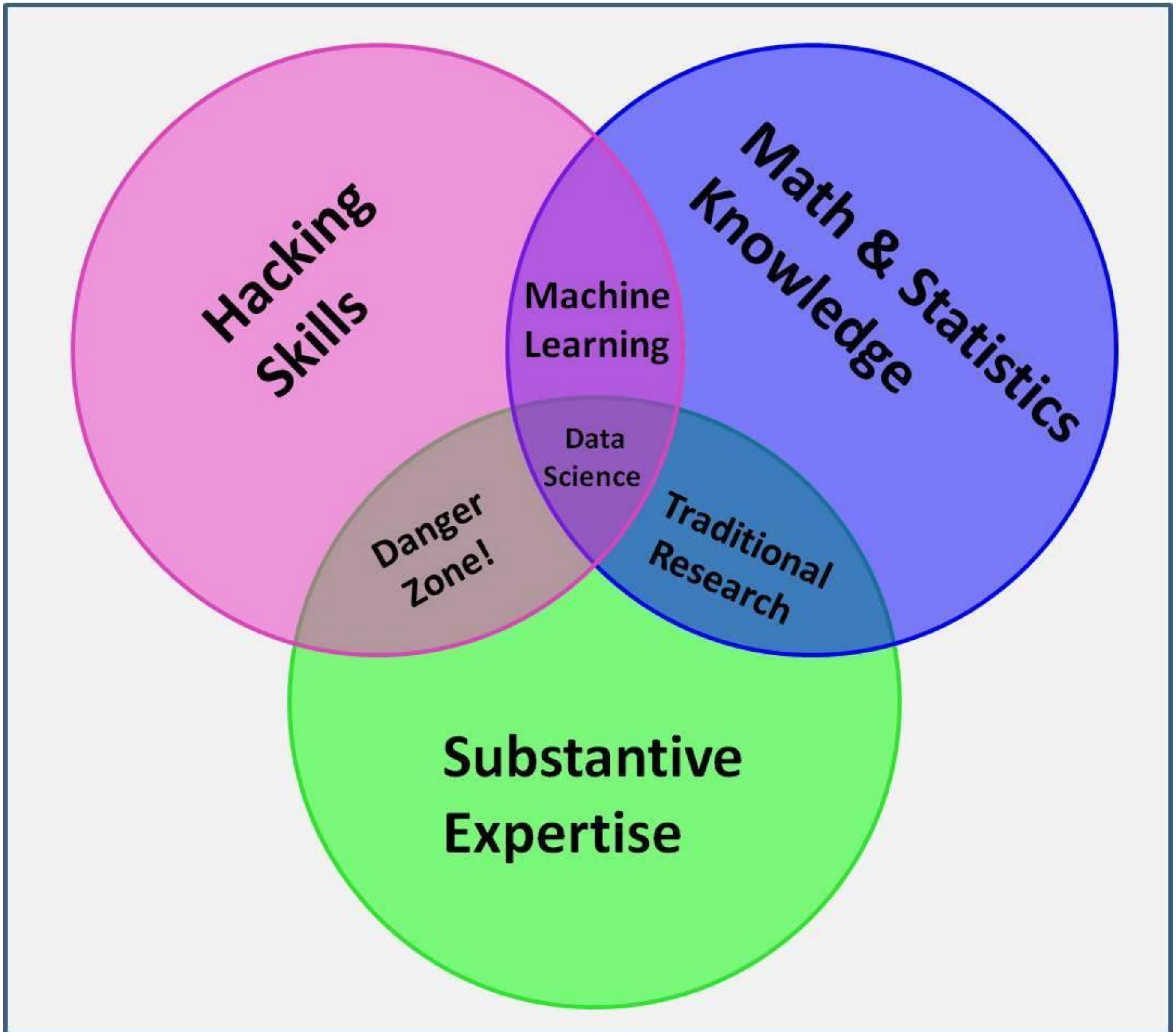
第10章：面向生物大数据挖掘的深度学习

- 深度学习是什么
- 深度学习的基本方法（以及神经网络算法）
- 生物大数据的深度学习方法
- 生物大数据深度学习的应用案例

Part I

深度学习是什么

深度学习与数据科学

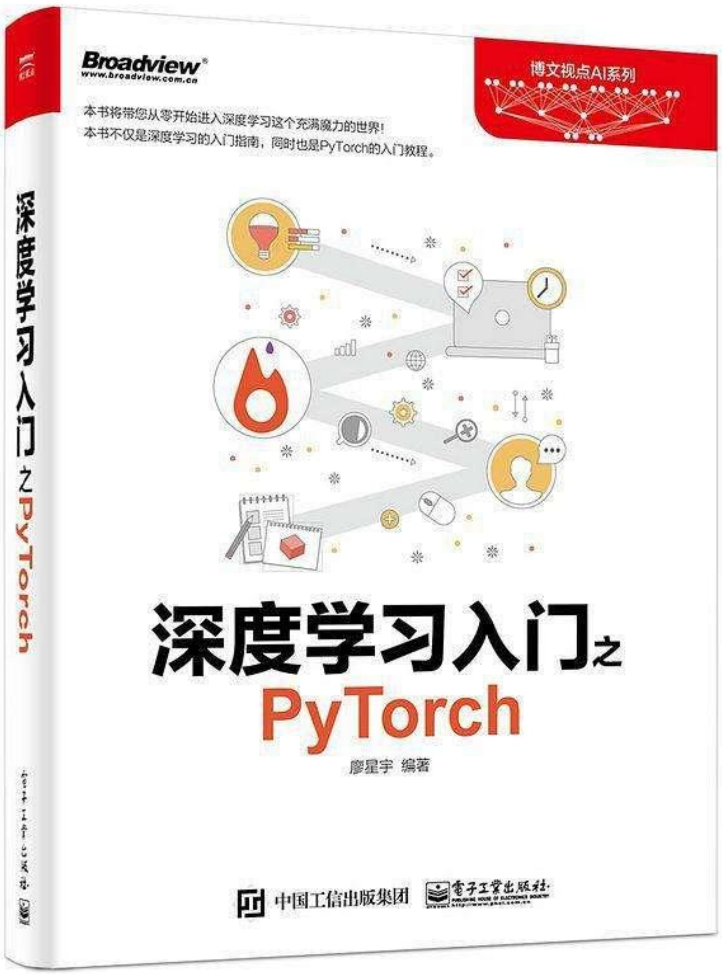

Hacking Skills
Math & Statistics Knowledge
Machine Learning
Data Science
Danger Zone!
Traditional Research
Substantive Expertise

深度学习与数据科学

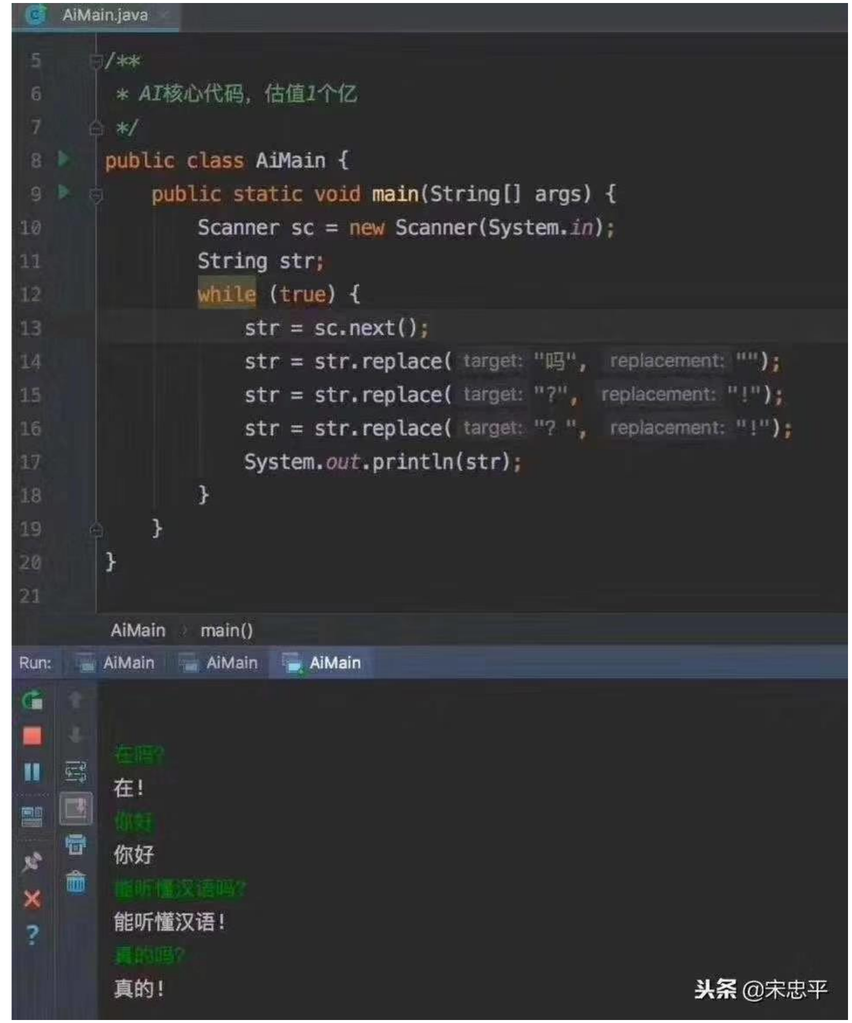
- 深度学习：一种基于无监督特征学习和特征层次结构的学习方法

- 可能的的名称：
  - 深度学习
  - 特征学习
  - 无监督特征学习

真假深度学习


Broadview
www.broadview.com.cn
博文视点AI系列
本书将带您从零开始进入深度学习这个充满魔力的世界!
本书不仅是深度学习的入门指南，同时也是PyTorch的入门教程。
深度学习入门之PyTorch
廖星宇 编著
中国工信出版集团
電子工業出版社

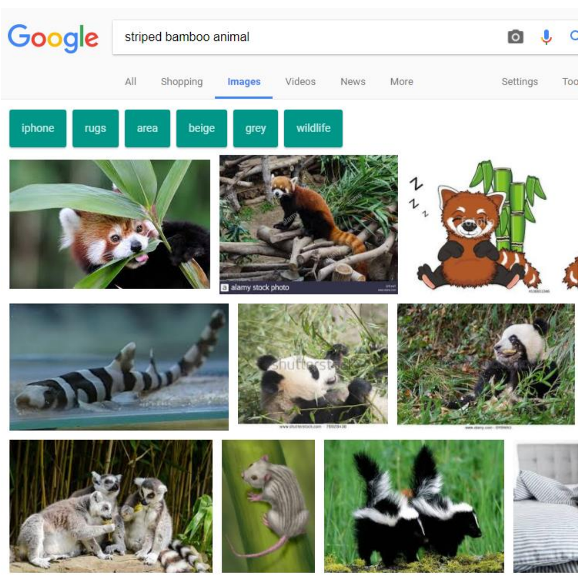
真


AiMain.java
/**
* AI核心代码，估值1个亿
*/
public class AiMain {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
String str;
while (true) {
str = sc.next();
str = str.replace(target: "吗", replacement: "");
str = str.replace(target: "?", replacement: "!");
str = str.replace(target: "? ", replacement: "!");
System.out.println(str);
}
}
}
AiMain
main()
Run:
AiMain
AiMain
AiMain
在吗?
在!
你好
你好
能听懂汉语吗?
能听懂汉语!
真的吗?
真的!
头条 @宋忠平

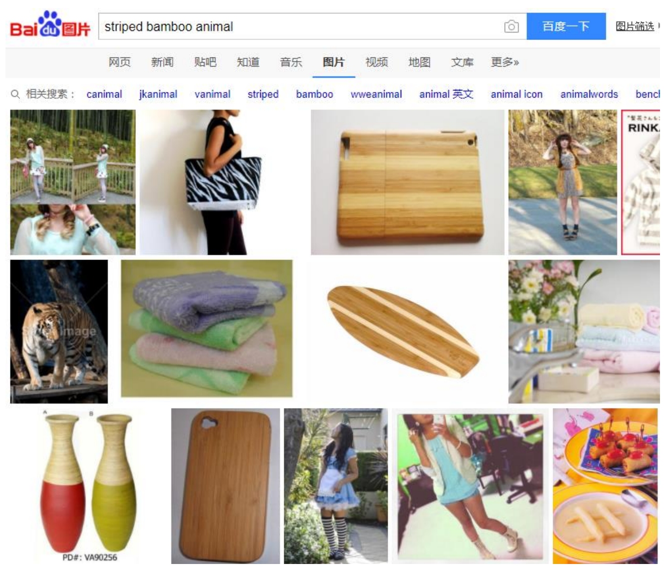
假

搜索引擎的区别

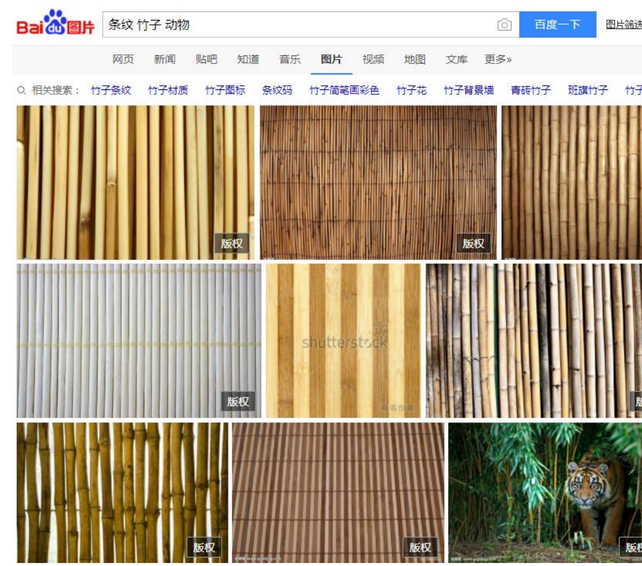
striped bamboo animal



striped bamboo animal

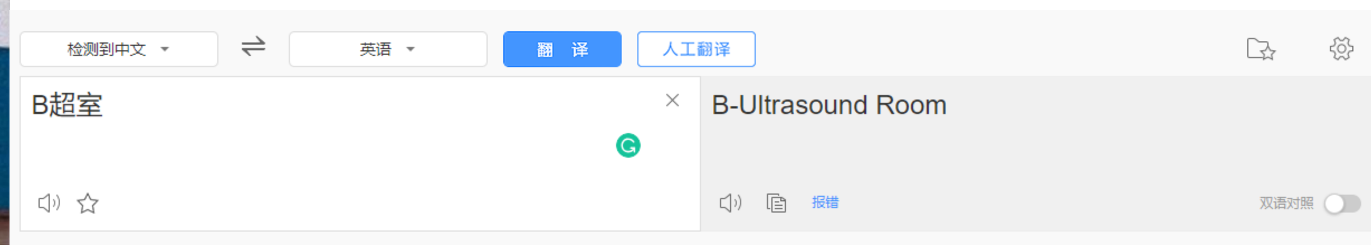


条纹 竹子 动物

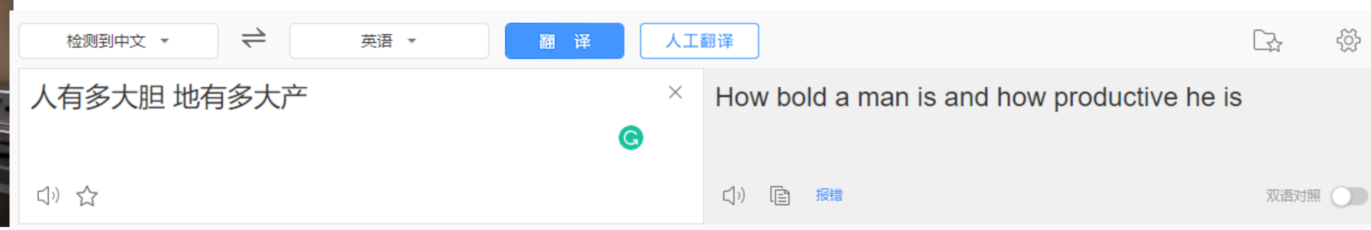


翻译引擎和人类的较量


宁德市医院
健康体检科
B 超 室
2
B super Room


检测到中文
英语
翻 译
人工翻译
B超室
B-Ultrasound Room
报错
双语对照

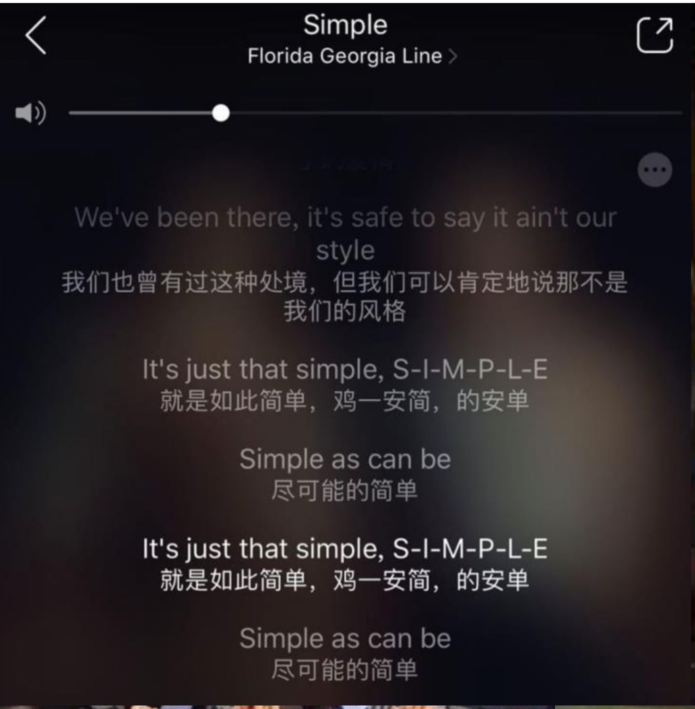

人有多大胆 地有多大产
No guts, no glory, man.


检测到中文
英语
翻 译
人工翻译
人有多大胆 地有多大产
How bold a man is and how productive he is
报错
双语对照


请在一米线外等候
PLEASE WAIT OUTSIDE A NOODLE
急诊神内科
Emergency God Medical


There's a fire starting in my heart
情不知所往，恨不知深长
舍不得孩子 套不着狼啊
You can't make an omelet without cracking a few eggs.

翻译引擎和人类的较量


Simple
Florida Georgia Line
We've been there, it's safe to say it ain't our style
我们也曾有过这种处境，但我们可以肯定地说那不是我们的风格
It's just that simple, S-I-M-P-L-E
就是如此简单，鸡一安简，的安单
Simple as can be
尽可能的简单
It's just that simple, S-I-M-P-L-E
就是如此简单，鸡一安简，的安单
Simple as can be
尽可能的简单


不为无益之事 何以遣有涯之生
What's life without whimsy?

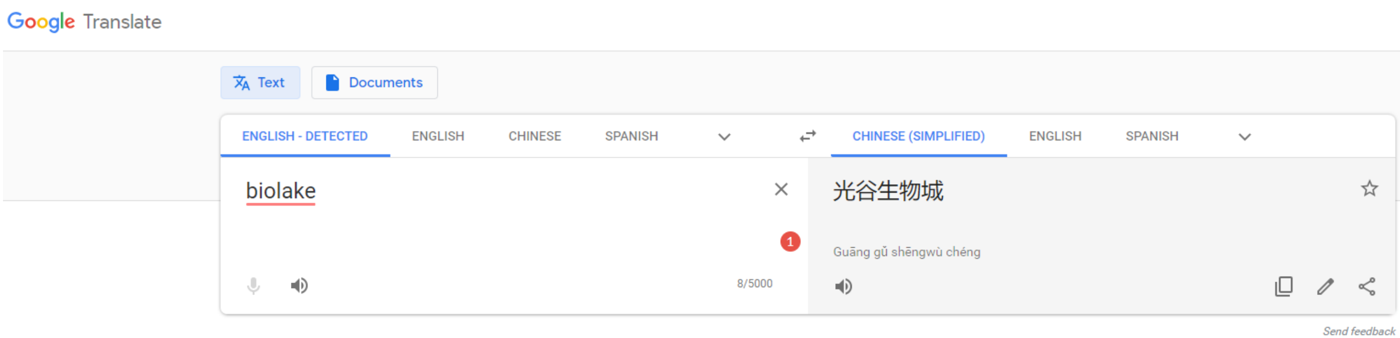

撒子
what


馄饨
chaos


灭 烟 台
Destroy Yantai
顺诚物业

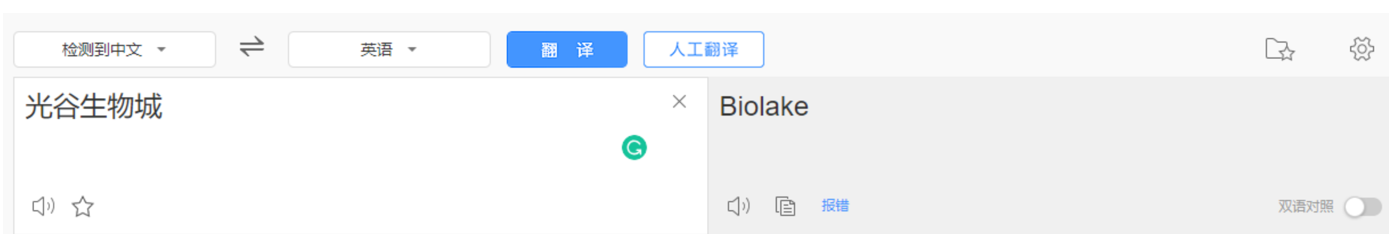
翻译引擎的区别

biolake

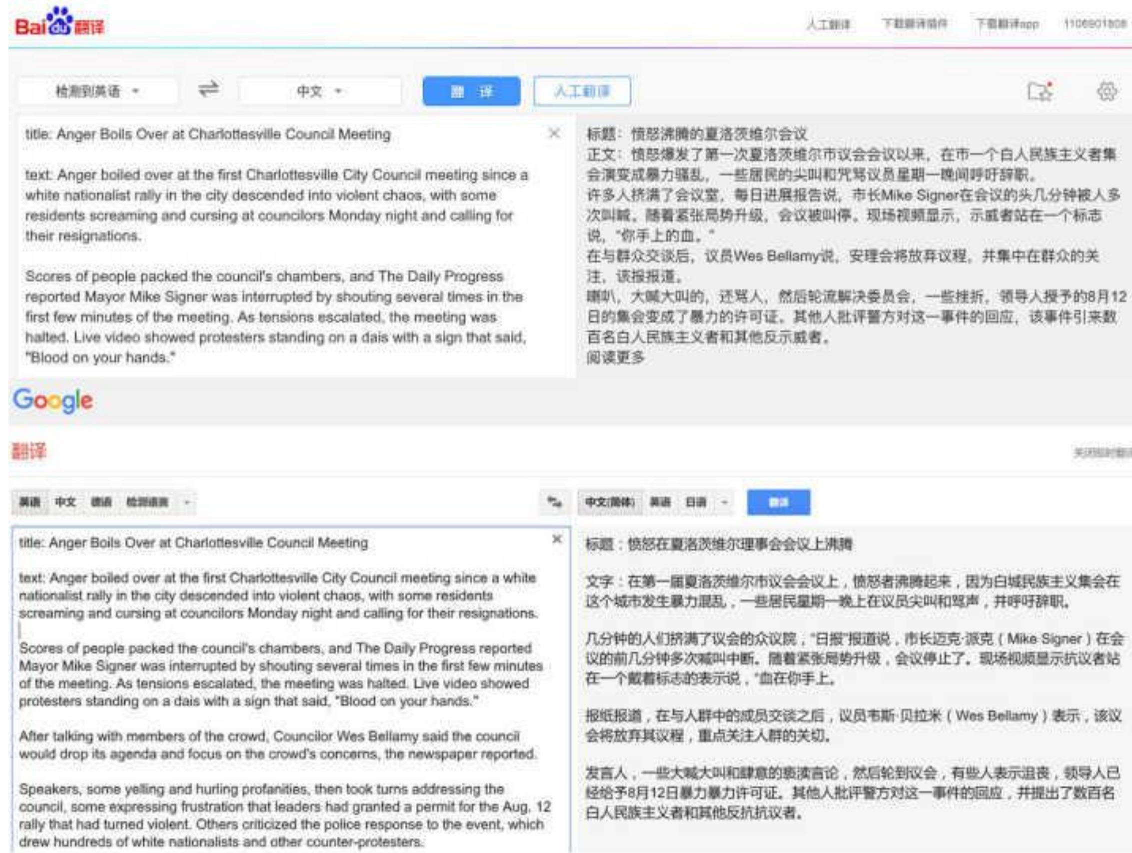


biolake

光谷生物城



翻译的区别（互有优劣）



它很难吃。

谷歌：It is hard to eat.

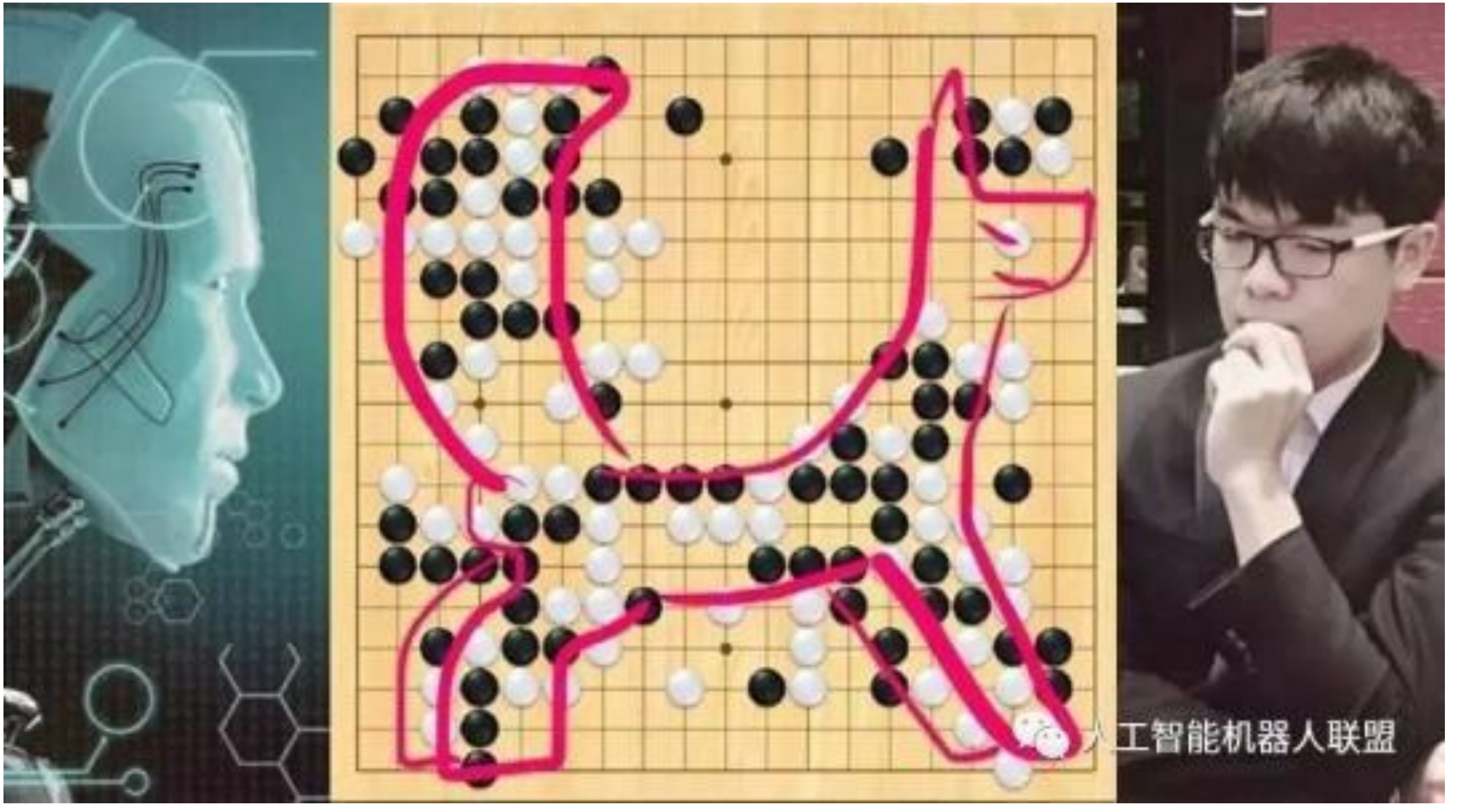
百度：It tastes terrible.

我要下班。

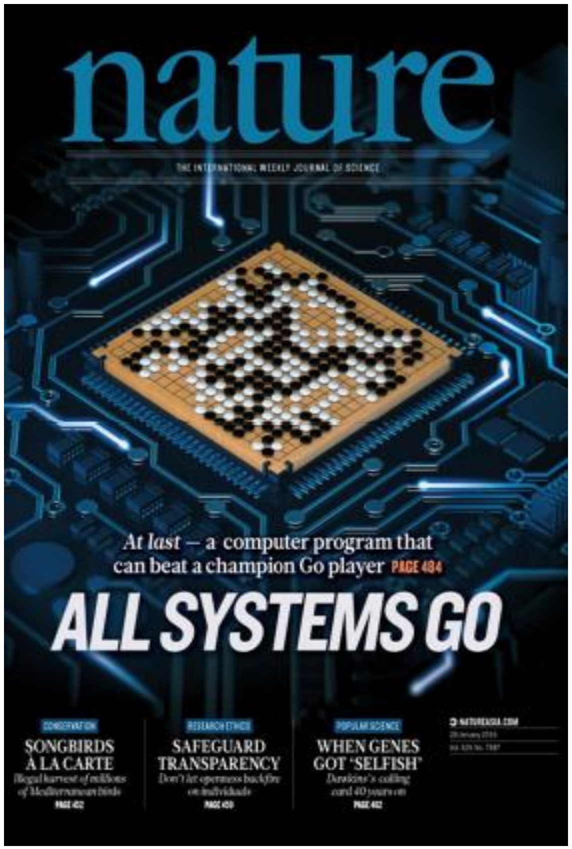
谷歌：I want to get off work from work.

百度：I'm going to get off work。

AlphaGo



AlphaGo

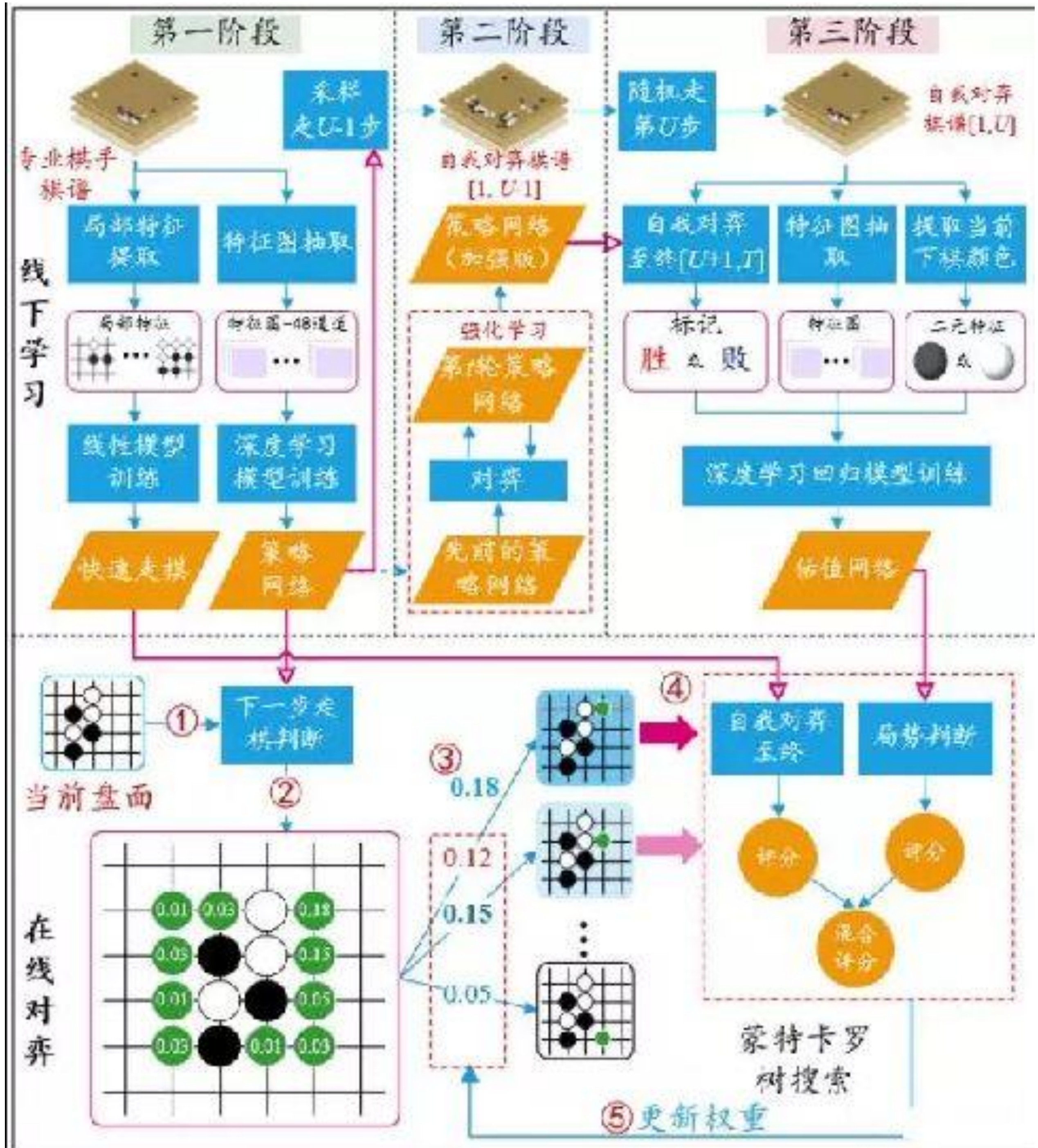


ARTICLE

doi:10.1038/nature16961

Mastering the game of Go with deep neural networks and tree search

David Silver[1*], Aja Huang[1*], Chris J. Maddison[1], Arthur Guez[1], Laurent Sifre[1], George van den Driessche[1], Julian Schrittwieser[1], Ioannis Antonoglou[1], Veda Panneershelvam[1], Marc Lanctot[1], Sander Dieleman[1], Dominik Grewe[1], John Nham[2], Nal Kalchbrenner[1], Ilya Sutskever[2], Timothy Lillicrap[1], Madeleine Leach[1], Koray Kavukcuoglu[1], Thore Graepel[1] & Demis Hassabis[1]



AlphaGo vs. AlphaGo Zero

数据很重要！

算法更重要！

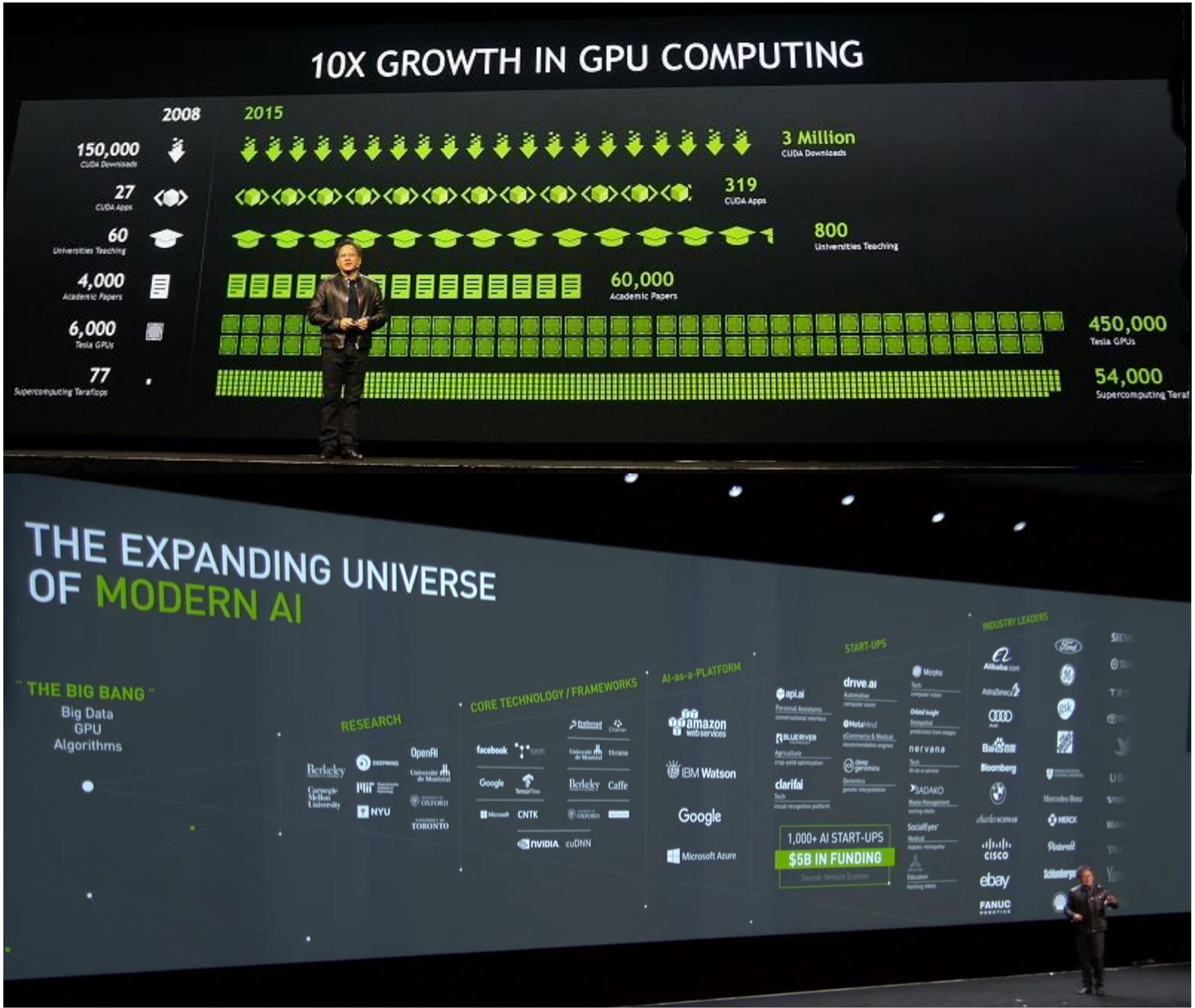
AlphaZero

Reinforcement Learning (强化学习)

Google DeepMind

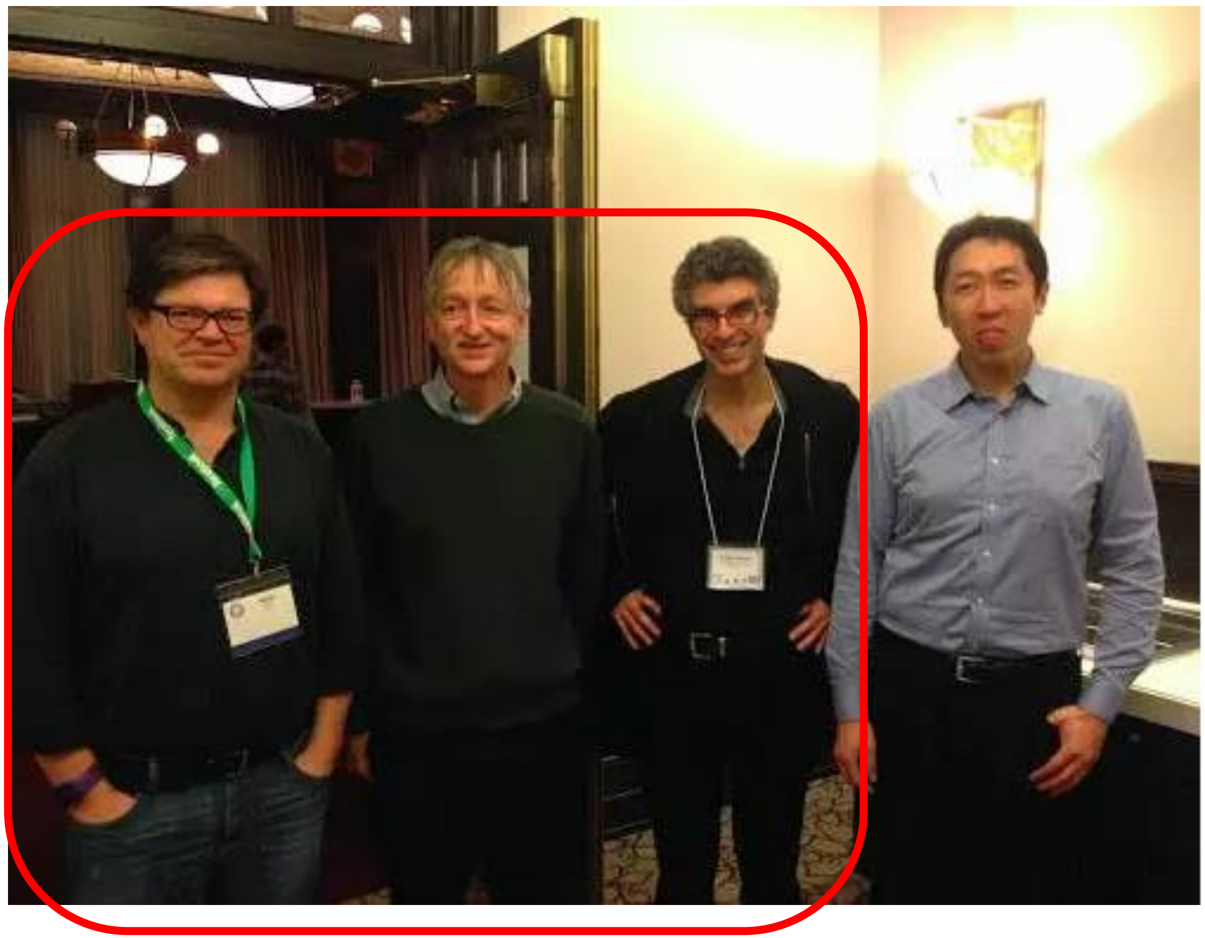
Transfer Learning (迁移学习)

Hardware: GPU, HPC...

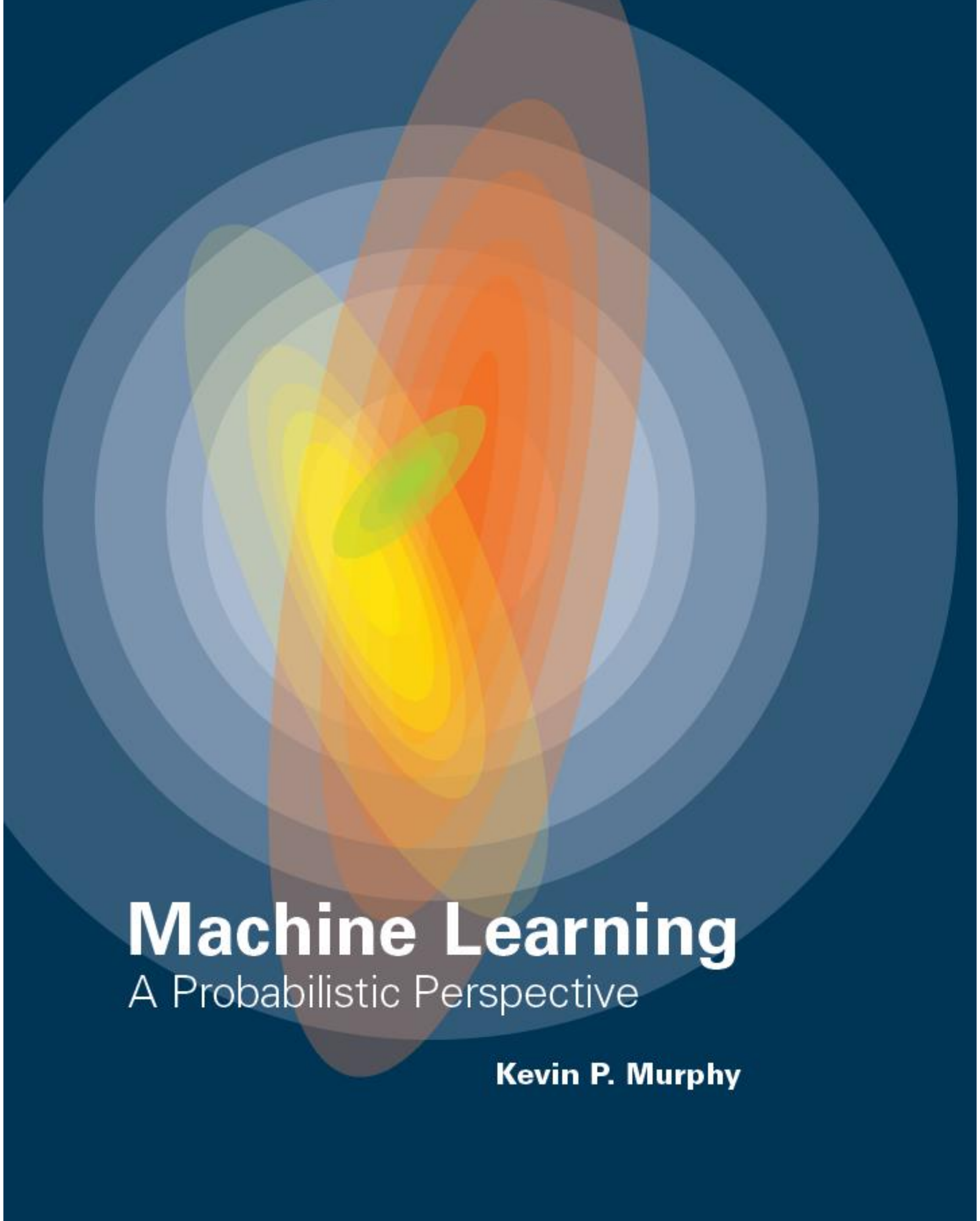


Software: TensorFlow, Caffe...

Turing Award, 2019



Yann LeCun
Geoffrey Hinton
Yoshua Bengio
Andrew Ng



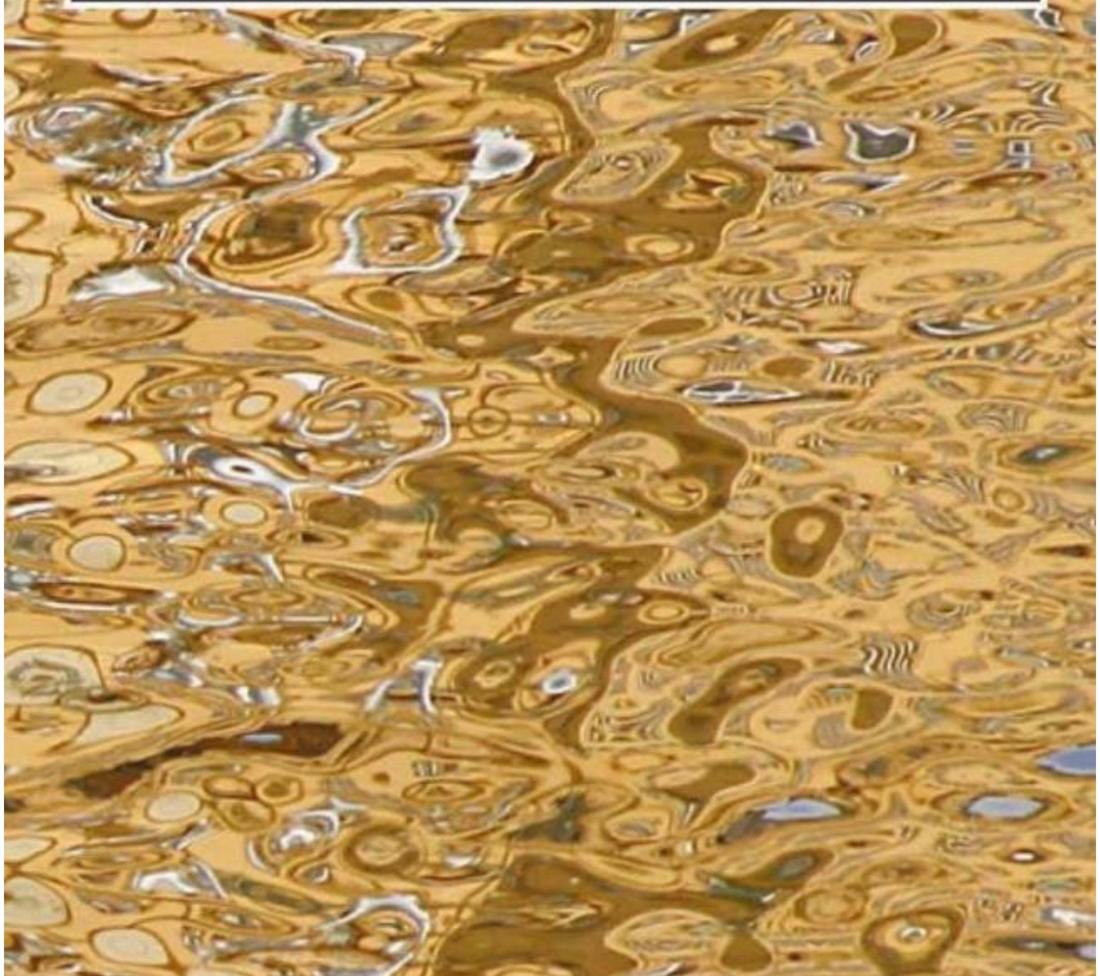
Machine Learning

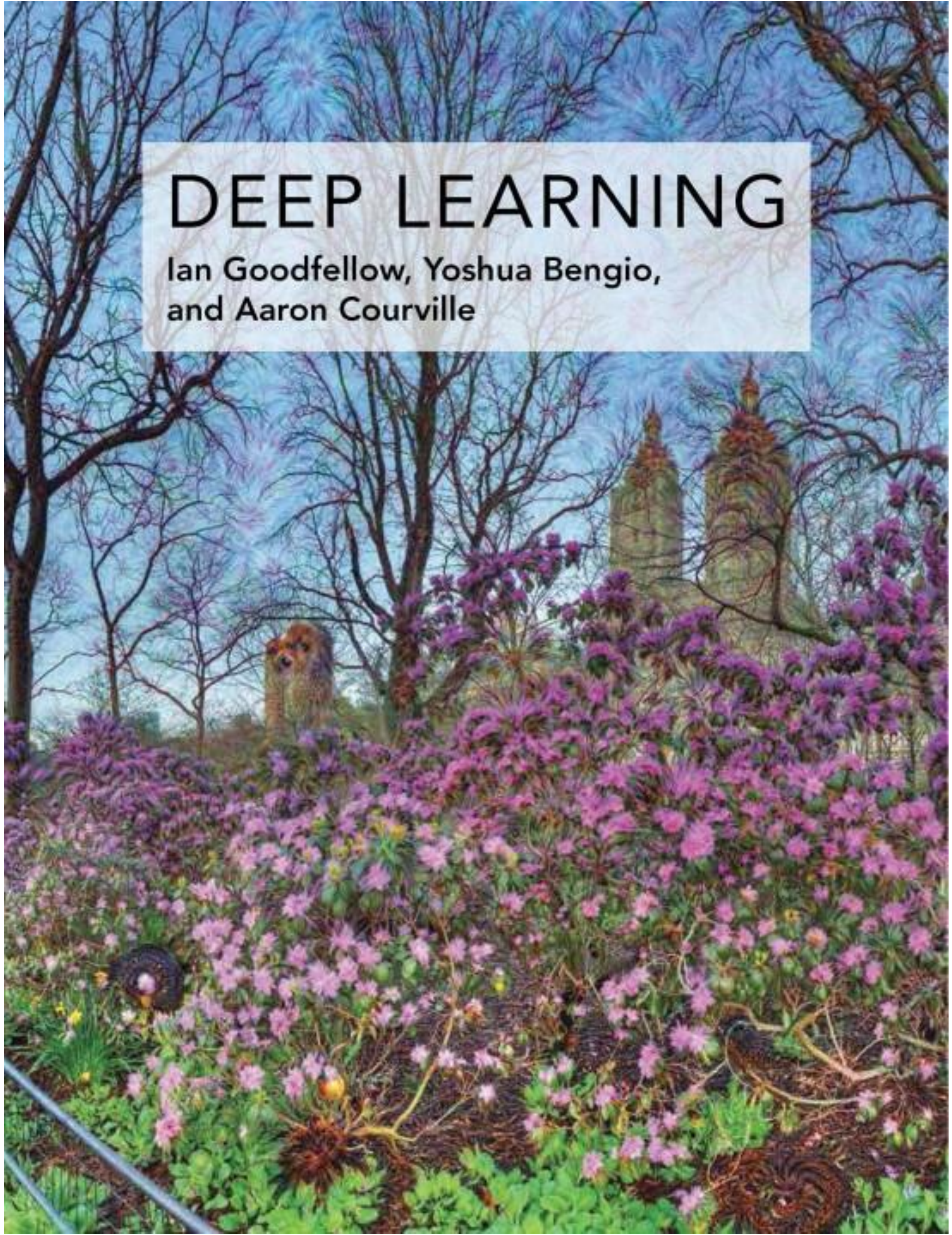
A Probabilistic Perspective

Kevin P. Murphy

PATTERN RECOGNITION AND MACHINE LEARNING

CHRISTOPHER M. BISHOP



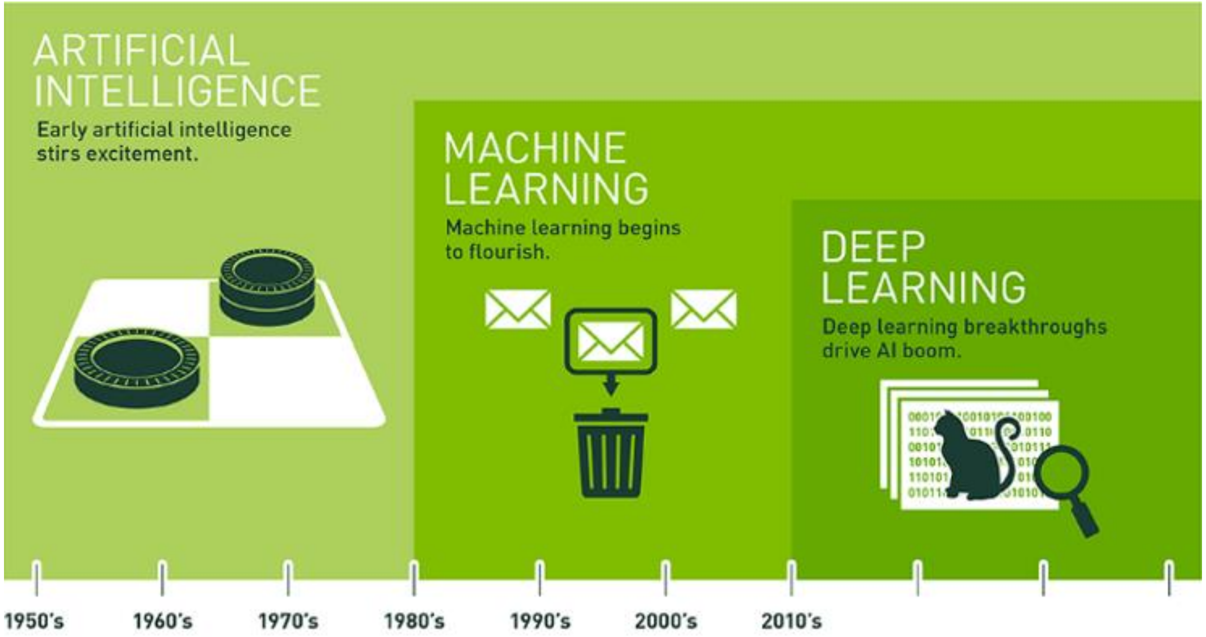

DEEP LEARNING
Ian Goodfellow, Yoshua Bengio,
and Aaron Courville

Introduction

- Artificial Intelligence

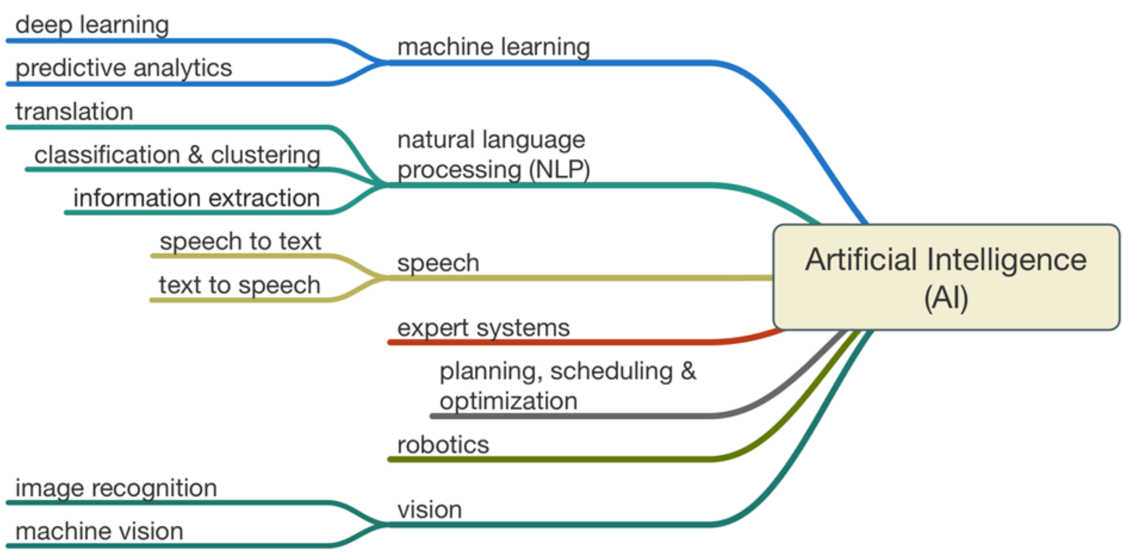
- Machine learning

- “Deep” learning



Artificial Intelligence

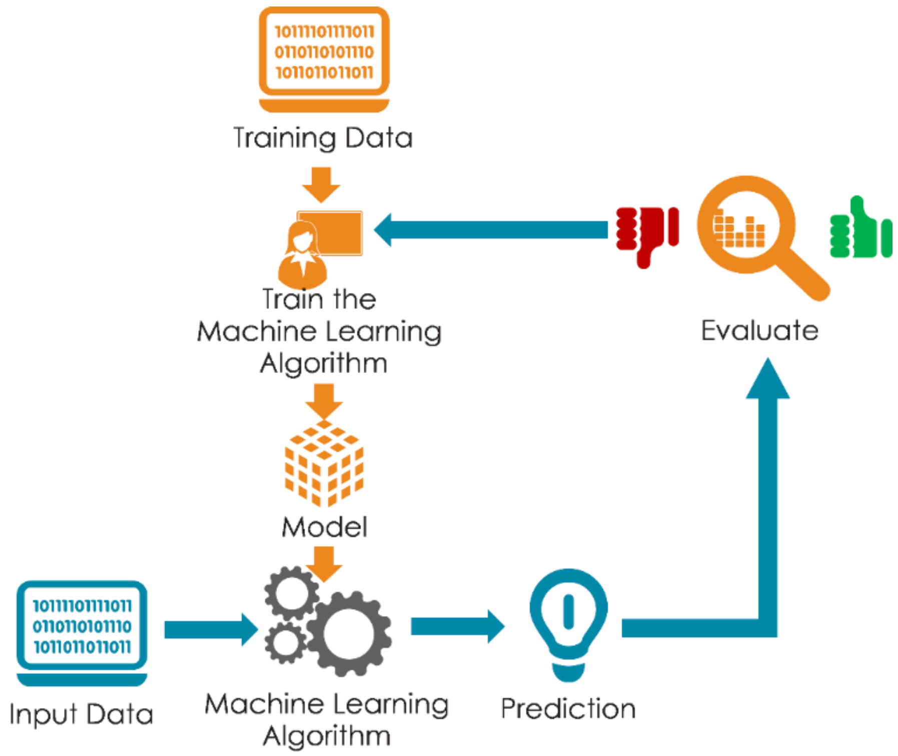
Artificial intelligence (AI, also machine intelligence, MI) is intelligence displayed by machines, in contrast with the natural intelligence (NI) displayed by humans and other animals.



Hatley, L. (2016). Presentation, New Designs for Learning: Games and Gamification
https://en.wikipedia.org/wiki/Applications_of_artificial_intelligence

Machine Learning

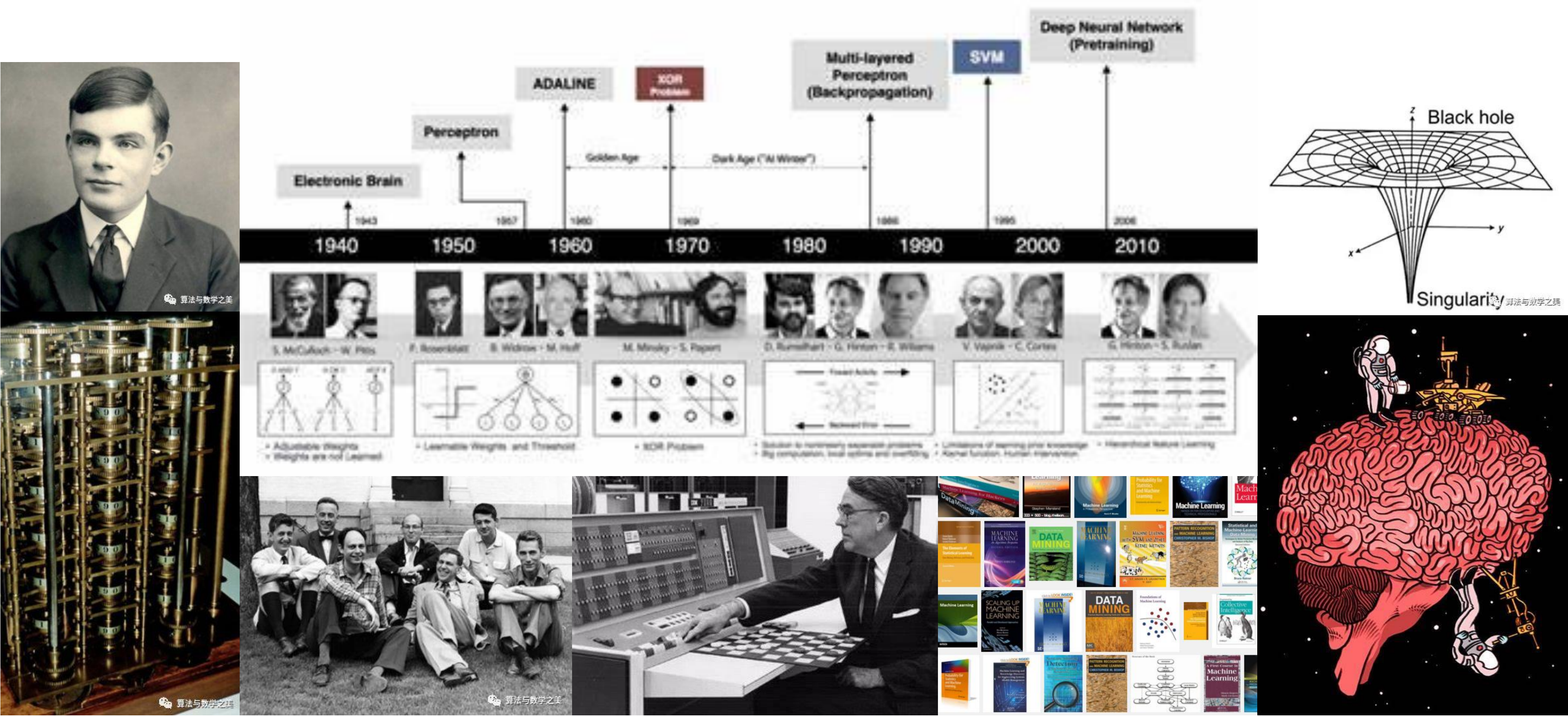
Machine learning is the science of getting computers to act without being explicitly programmed.



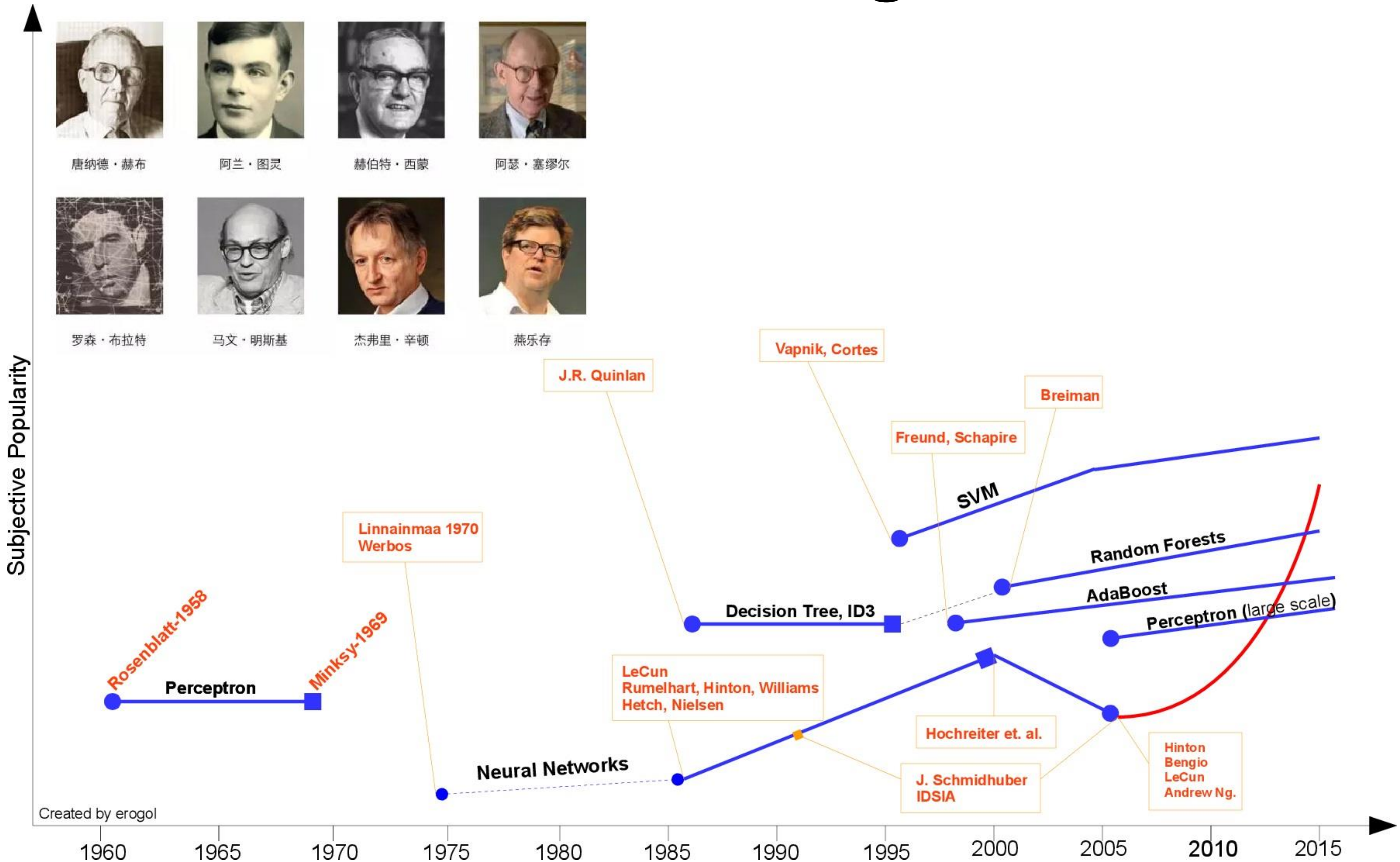
General workflow of Machine Learning

Pat Alvarado et al,2016,Building the Machine Learning Infrastructure

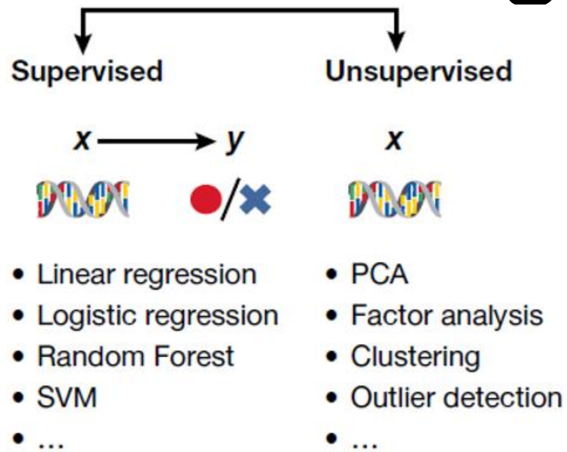
Machine Learning



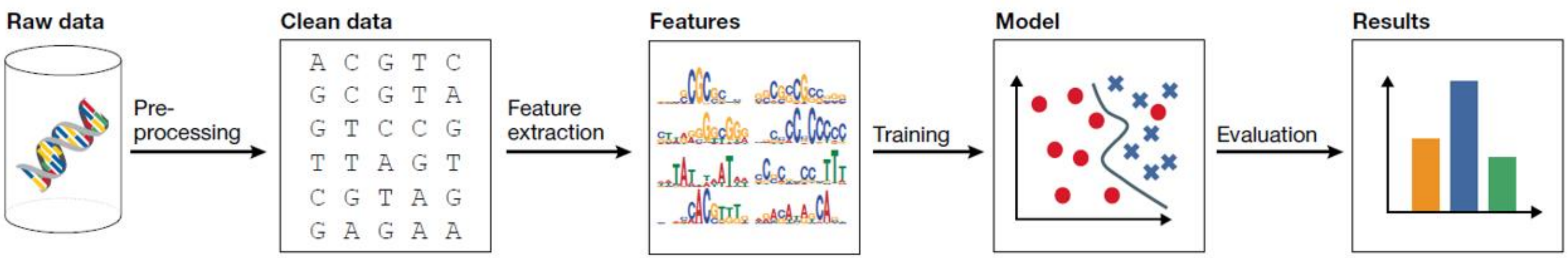
Machine Learning



Machine Learning in biology



Machine learning algorithm: supervised and unsupervised



An exemplification of Machine learning in biology : classification model

Angermueller C, et al, 2016, Deep learning for computational biology

Deep learning

Deep learning is a part of machine learning.

Deep:

Complex Model : Multi-layer characters, Many parameters (Curse of dimensionality)

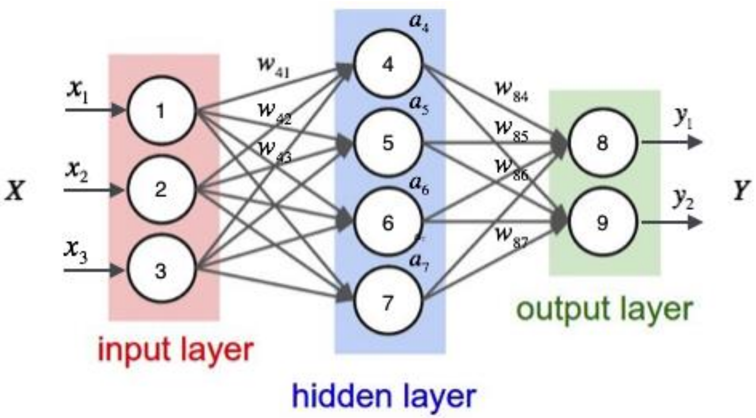
Training data & Testing data : a big volume of data

Adjustment: A single model construction could cost one week

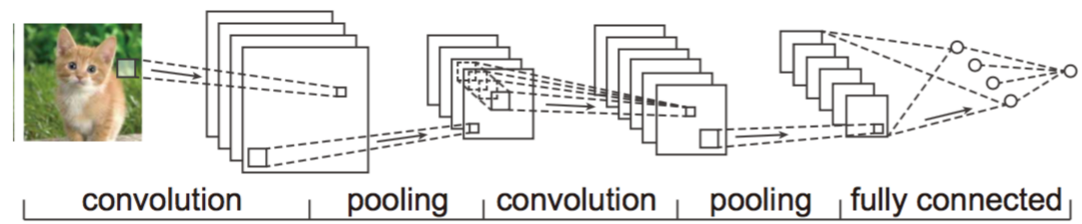
几乎所有深度学习算法都可以被描述为一个简单配方：
特定数据集、代价函数、优化过程和模型

Deep learning

Now, neural network and convolution neural network models that work best



Neural Network



Convolution Neural Network

IMAGENET Large Scale Visual Recognition Challenge

Year 2010

NEC-UIUC

Dense descriptor grid: HOG, LBP

↓

Coding: local coordinate, super-vector

↓

Pooling, SPM

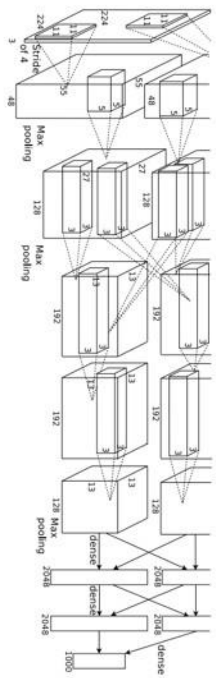
↓

Linear SVM

[Lin CVPR 2011]

Year 2012

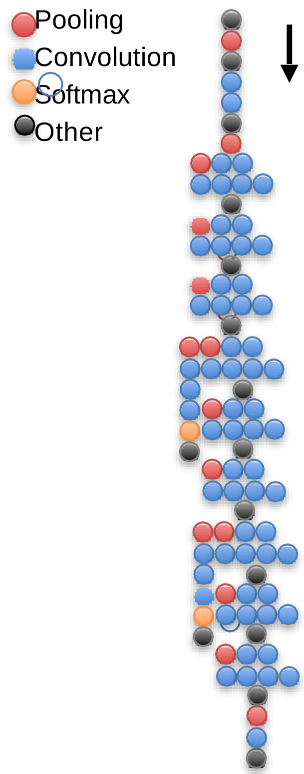
SuperVision



[Krizhevsky NIPS 2012]

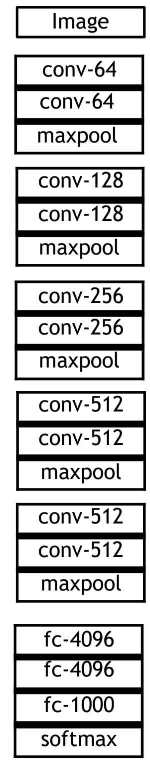
Year 2014

GoogLeNet



[Szegedy arxiv 2014]

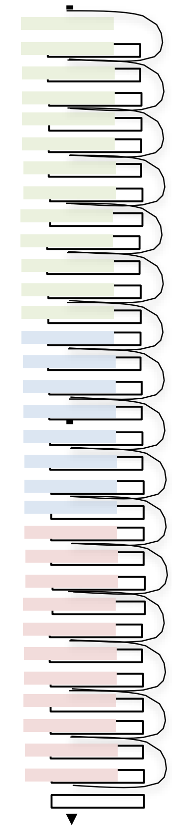
VGG



[Simonyan arxiv 2014]

Year 2015

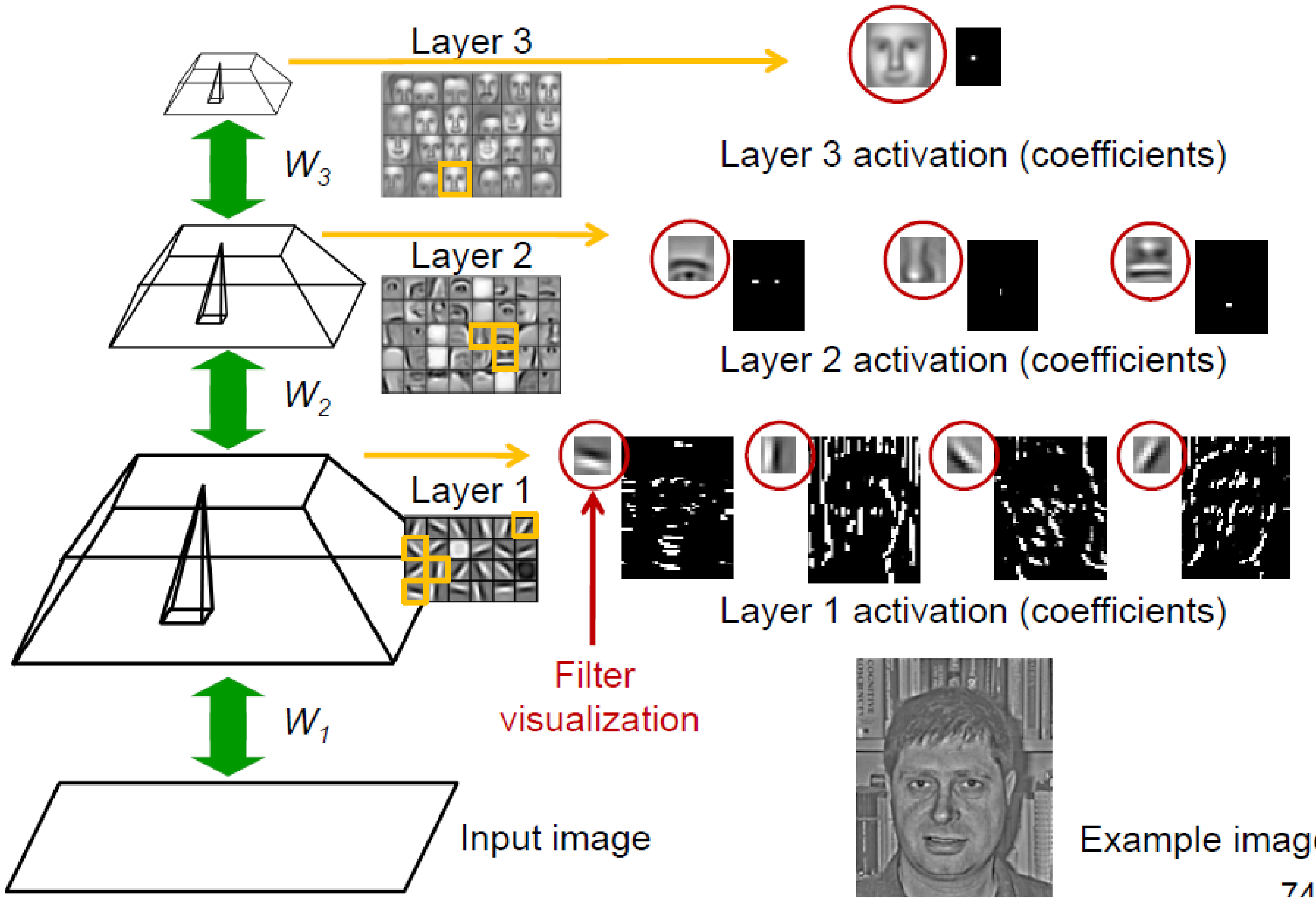
MSRA



[He ICCV 2015]

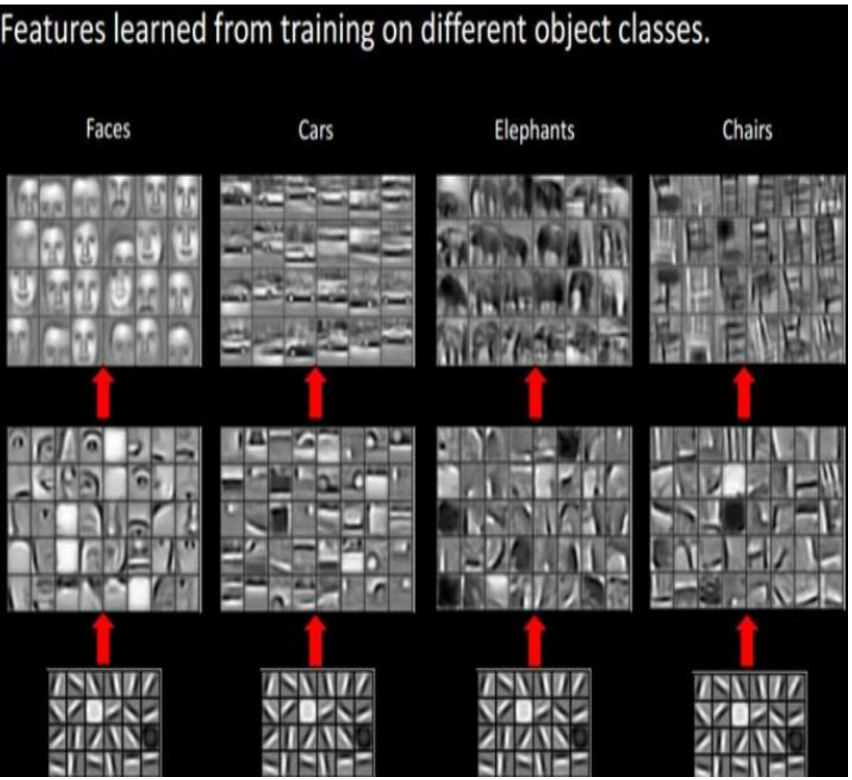
深度学习的应用

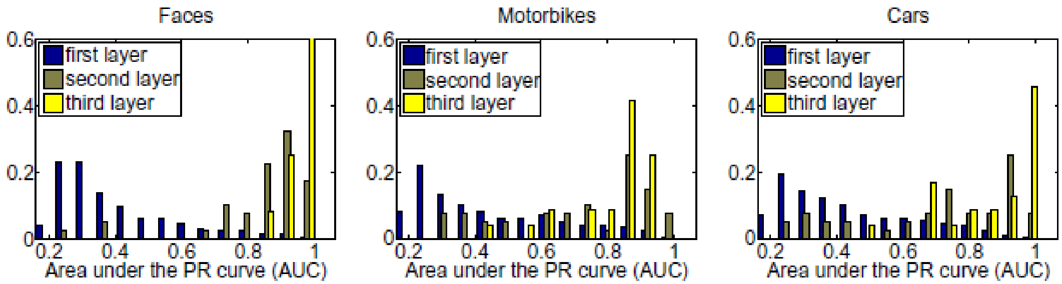
- 深度学习在图像识别上的应用



深度学习的应用

- 深度学习在图像识别上的应用

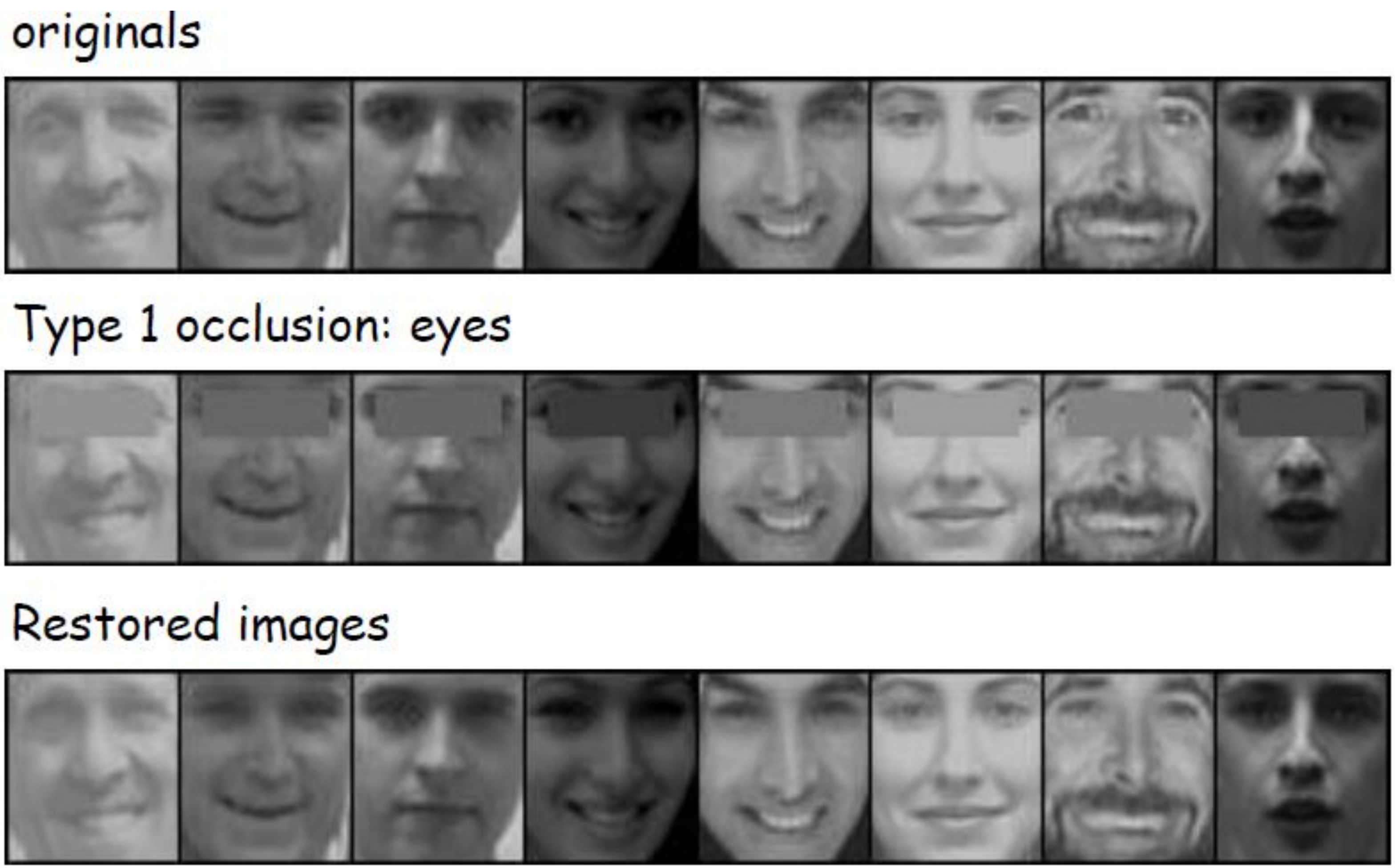




Features	Faces	Motorbikes	Cars
First layer	0.39±0.17	0.44±0.21	0.43±0.19
Second layer	0.86±0.13	0.69±0.22	0.72±0.23
Third layer	0.95±0.03	0.81±0.13	0.87±0.15

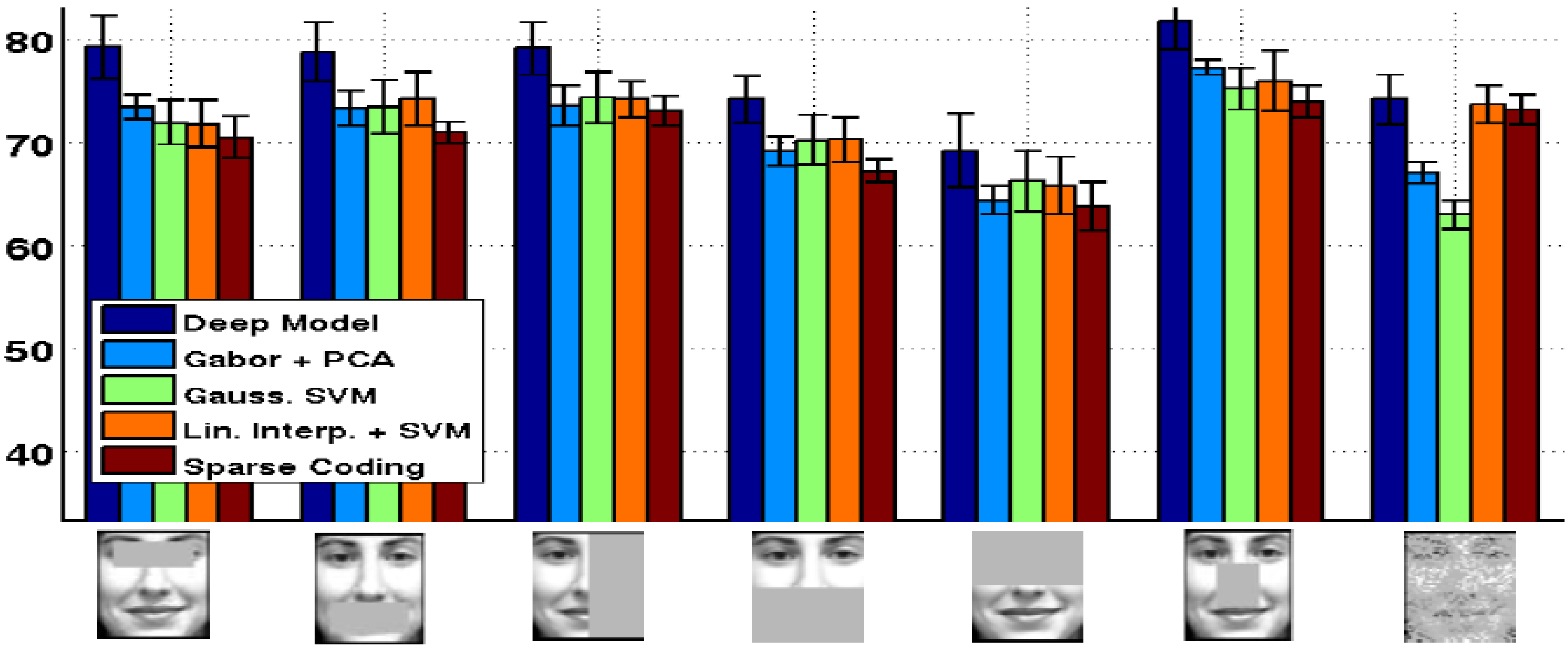
深度学习的应用

- 深度学习在图像识别上的应用



深度学习的应用

- 深度学习在图像识别上的应用



深度学习的应用

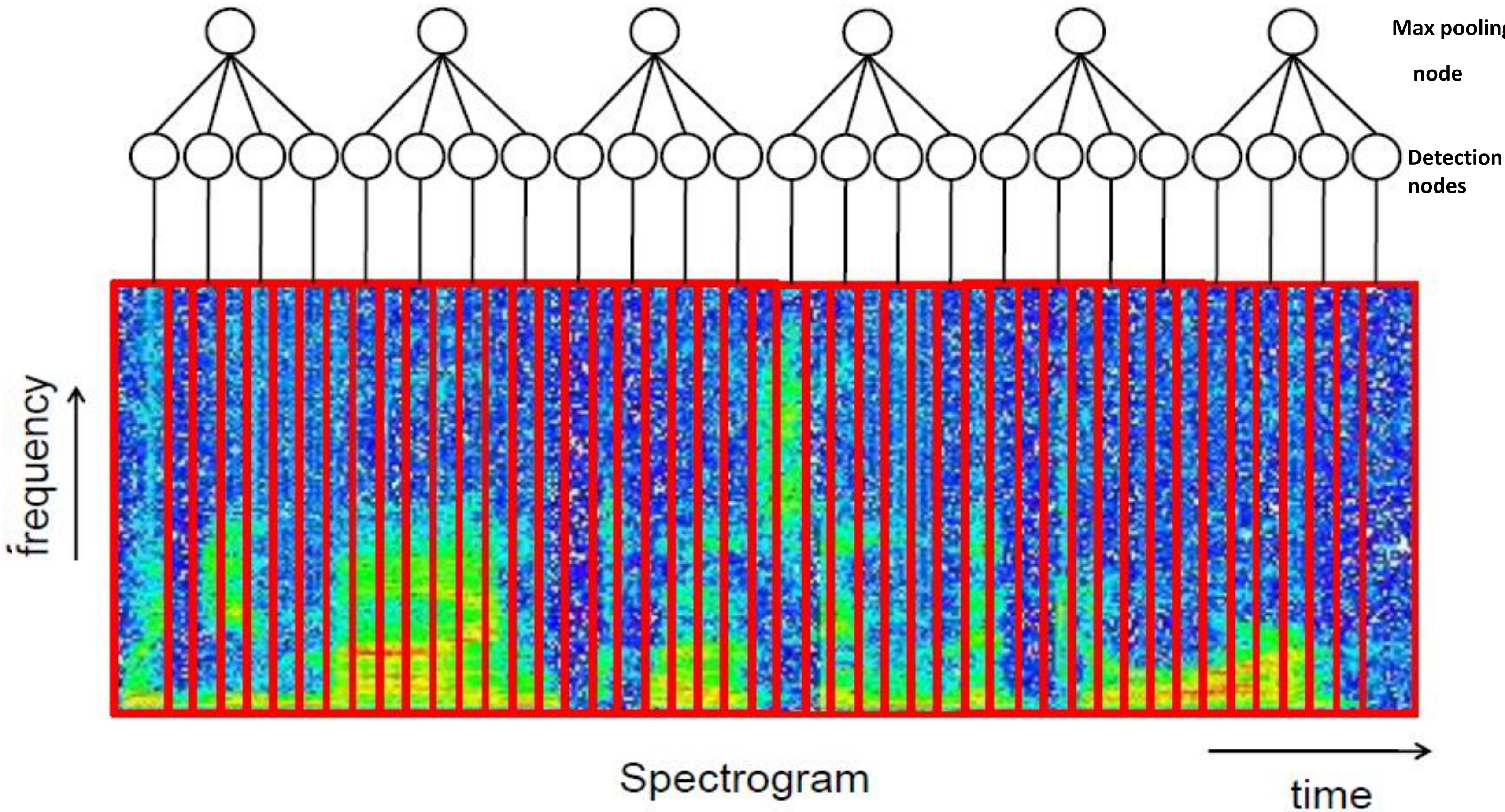
- 深度学习在图像识别上的应用

< Caltech 256 >

# of training images	30	60
Griffin et al. [2]	34.10	-
vanGemert et al., PAMI 2010	27.17	-
ScSPM [Yang et al., CVPR 2009]	34.02	40.14
LLC [Wang et al., CVPR 2010]	41.19	47.68
Sparse CRBM [Sohn et al., ICCV 2011]	42.05	47.94

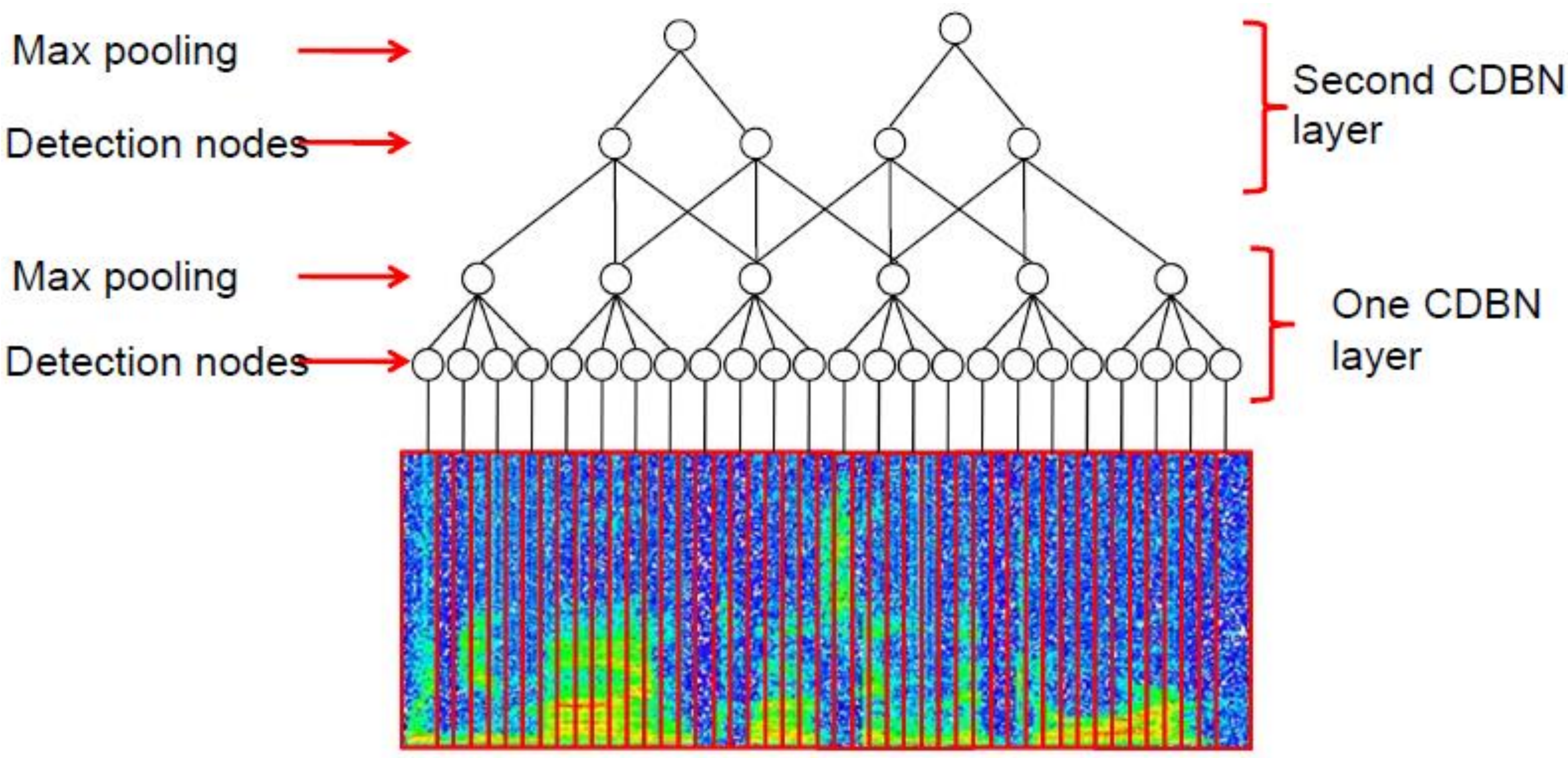
深度学习的应用

- 深度学习在音频识别上的应用



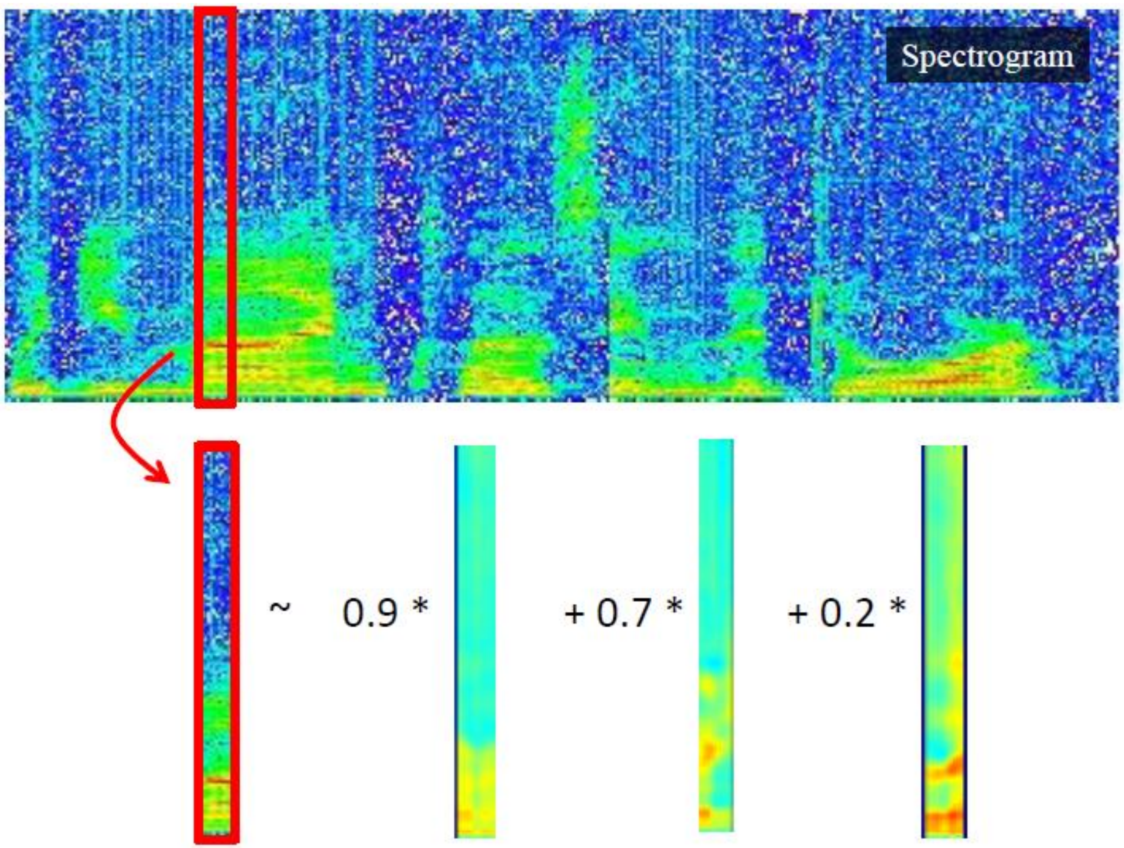
深度学习的应用

- 深度学习在音频识别上的应用



深度学习的应用

- 深度学习在音频识别上的应用



[Lee, Largman, Pham, Ng, NIPS 2009]

深度学习的应用

- 深度学习在音频识别上的应用
- Speaker identification

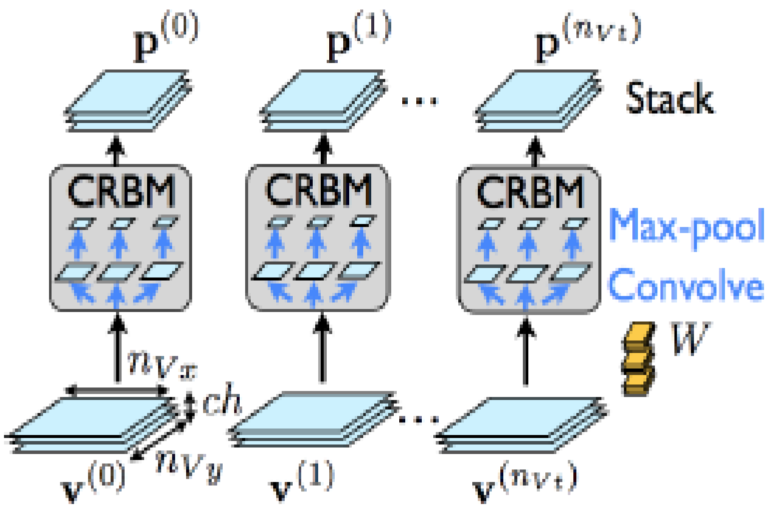
TIMIT Speaker identification	Accuracy
Prior art (Reynolds, 1995)	99.7%
Convolutional DBN	100.0%

- Phone classification

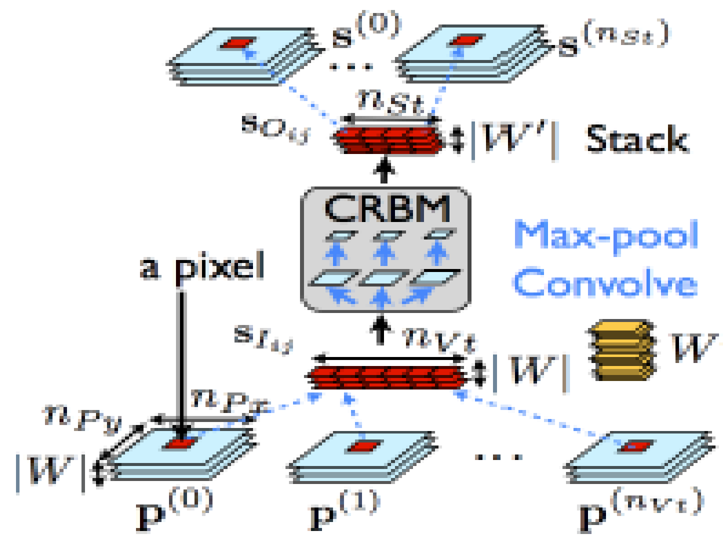
TIMIT Phone classification	Accuracy
Clarkson et al. (1999)	77.6%
Petrov et al. (2007)	78.6%
Sha & Saul (2006)	78.9%
Yu et al. (2009)	79.2%
Convolutional DBN	80.3%
Transformation-invariant RBM (Sohn et al., ICML 2012)	81.5%

深度学习的应用

- 深度学习在视频识别上的应用



Spatial pooling layer



Temporal pooling layer

深度学习的应用

- 深度学习在视频识别上的应用

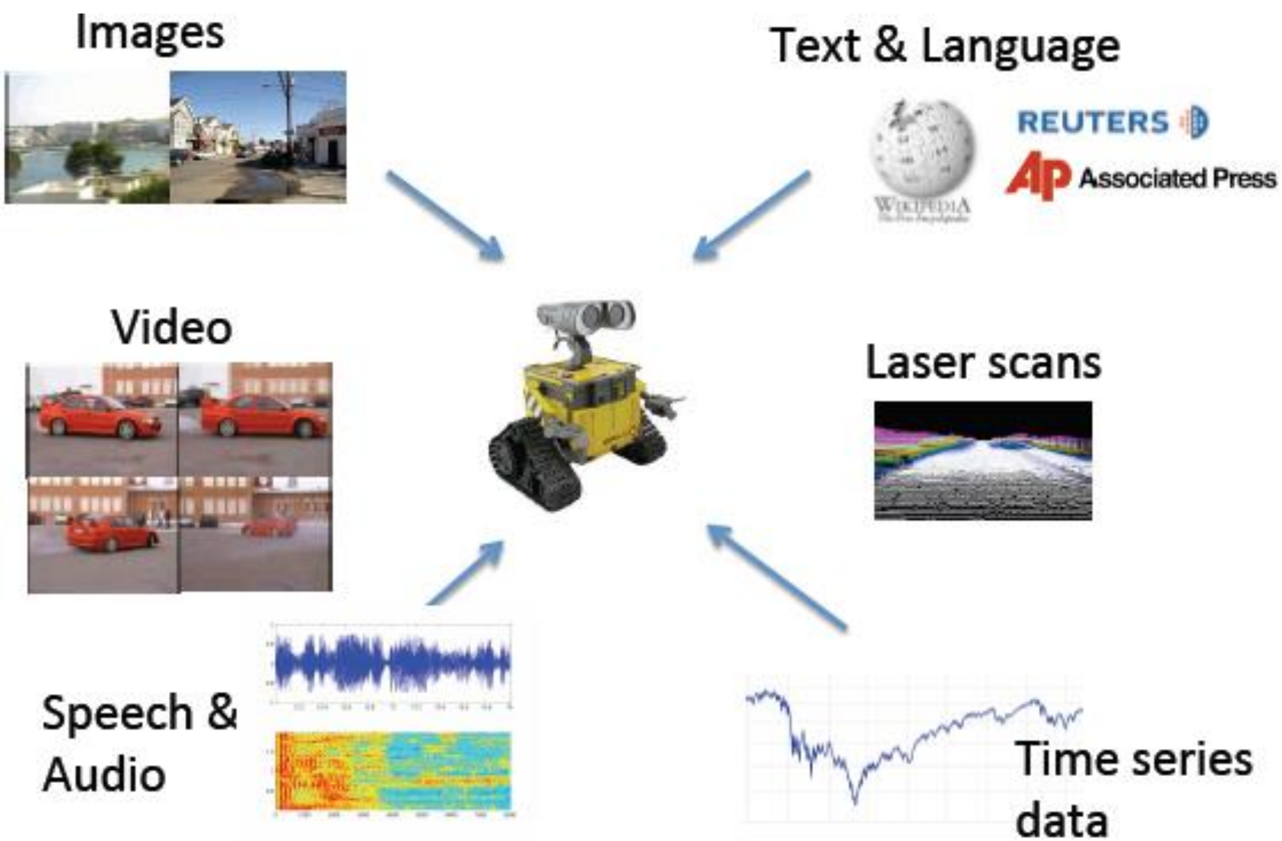
Video Activity recognition (Hollywood 2 benchmark)

Method	Accuracy
Hessian + ESURF [Williems et al 2008]	38%
Harris3D + HOG/HOF [Laptev et al 2003, 2004]	45%
Cuboids + HOG/HOF [Dollar et al 2005, Laptev 2004]	46%
Hessian + HOG/HOF [Laptev 2004, Williems et al 2008]	46%
Dense + HOG / HOF [Laptev 2004]	47%
Cuboids + HOG3D [Klaser 2008, Dollar et al 2005]	46%
Unsupervised feature learning (our method)	**52%**

Unsupervised feature learning significantly improves
on the previous state-of-the-art.

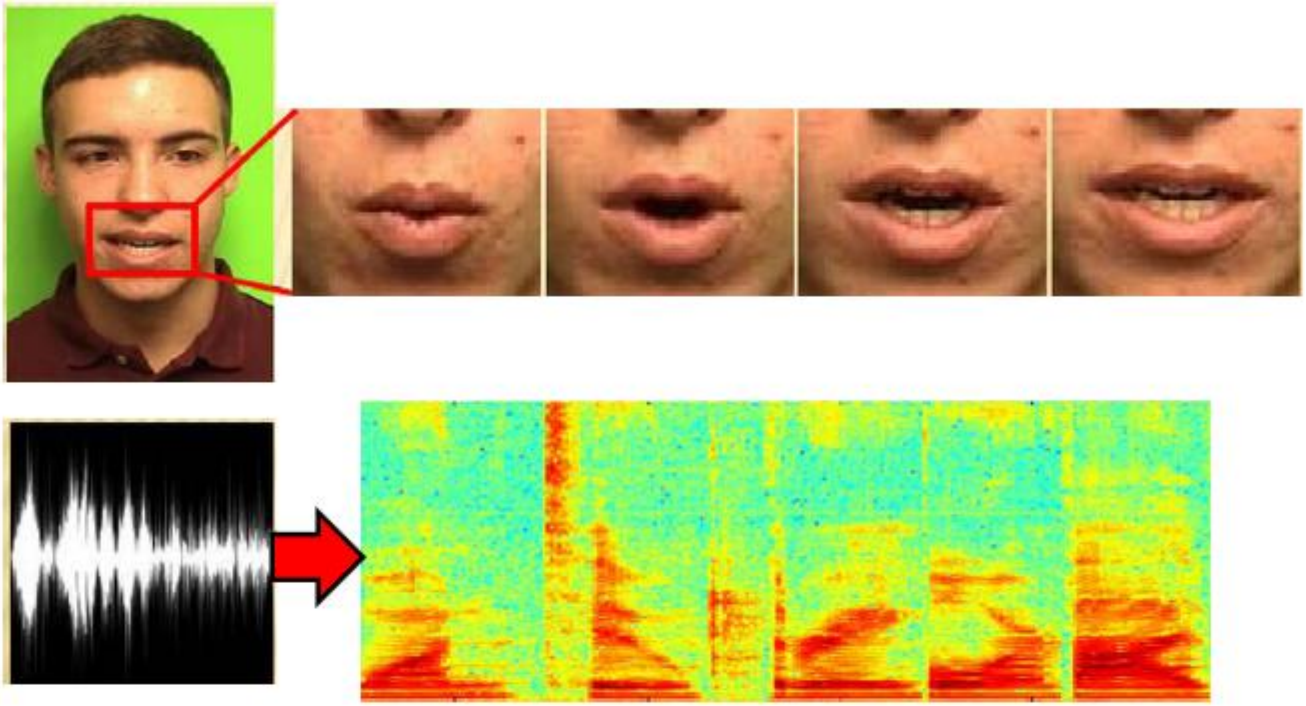
深度学习的应用

- 深度学习在多模态学习中的应用



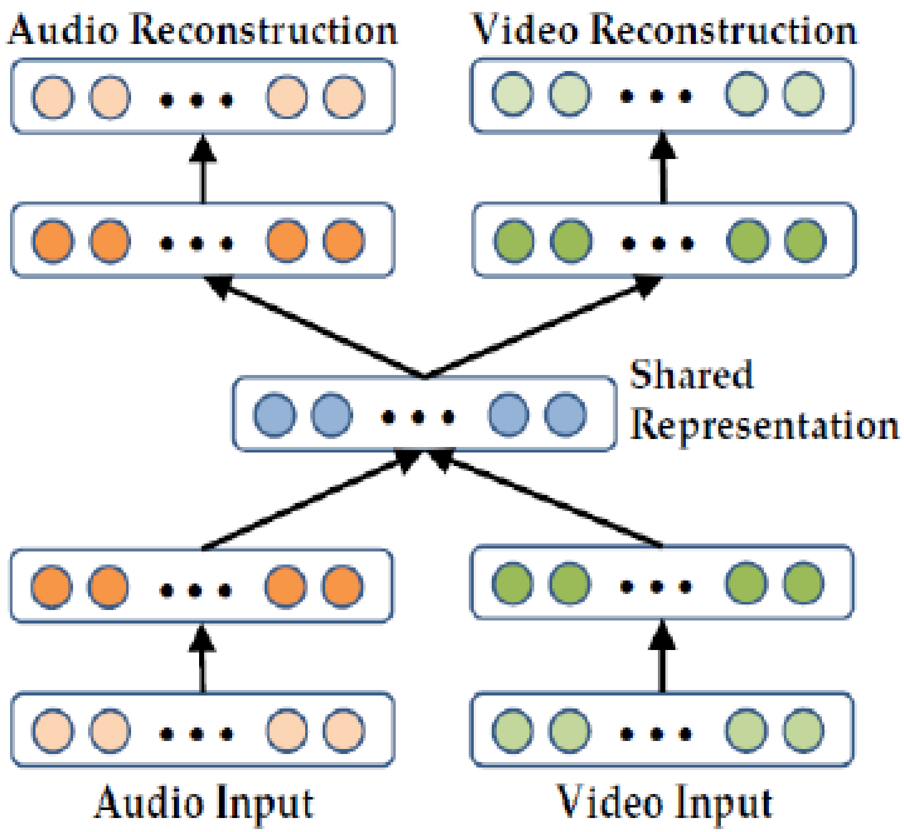
深度学习的应用

- 深度学习在多模态学习中的应用



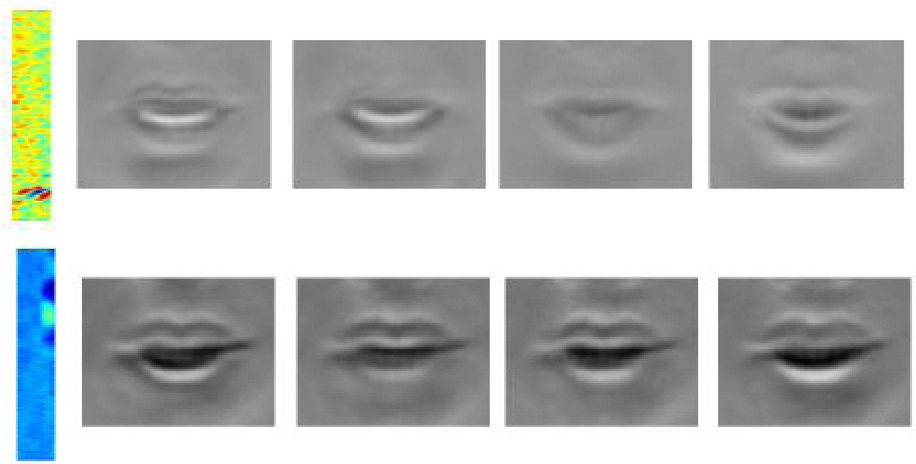
深度学习的应用

- 深度学习在多模态学习中的应用



深度学习的应用

- 深度学习在多模态学习中的应用
  - Visualization of learned filters



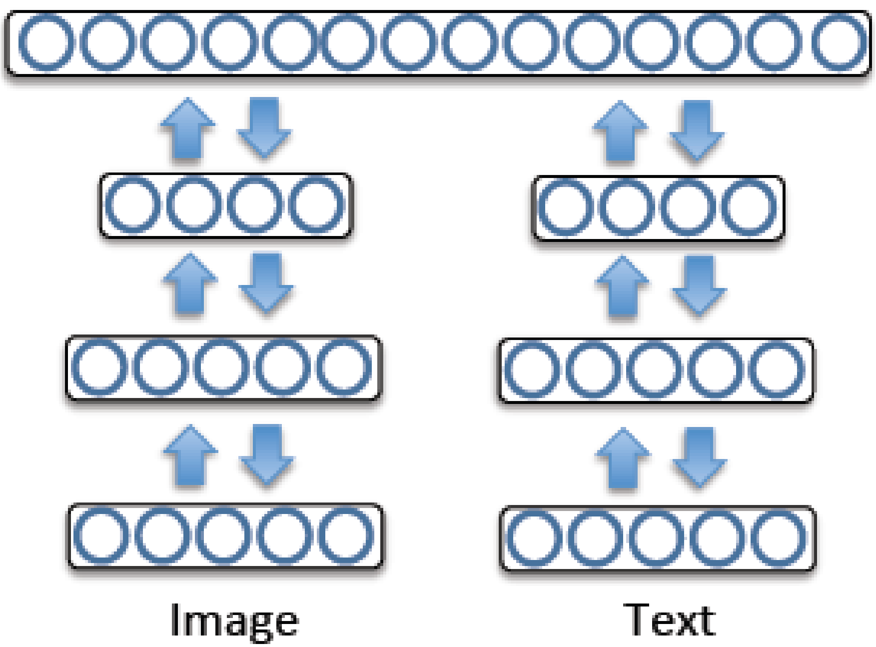
Audio(spectrogram) and Video features learned over 100ms windows

  - Results: AVLetters Lip reading dataset

Method	Accuracy
Prior art (Zhao et al., 2009)	58.9%
Multimodal deep autoencoder (Ngiam et al., 2011)	**65.8%**

深度学习的应用

- 深度学习在多模态学习中的应用

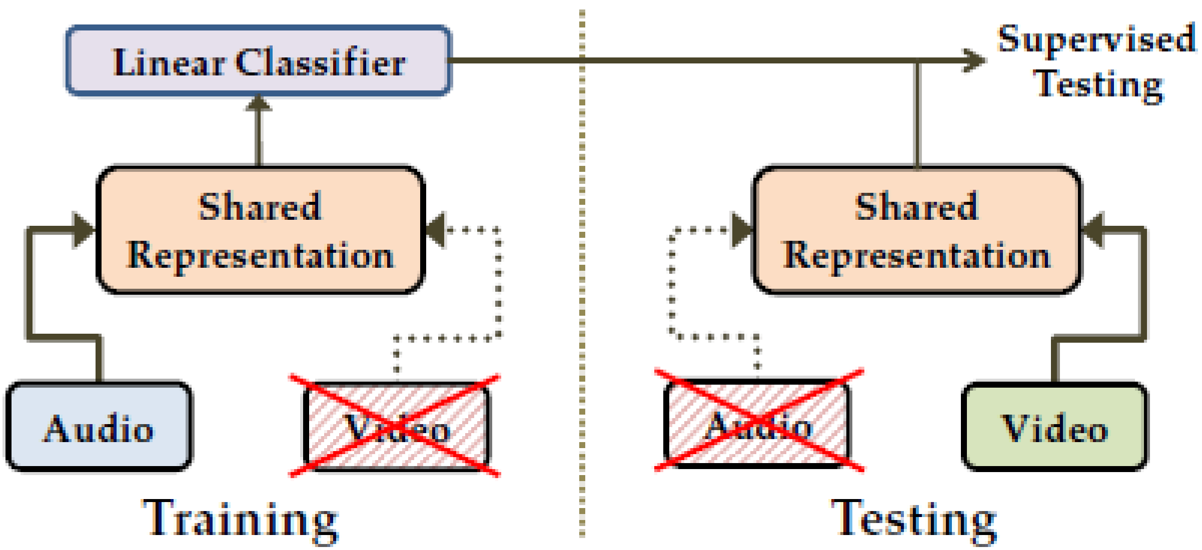


- Multimodal Inputs (images + text), 38 classes.

Learning Algorithm	Mean Average Precision
Image-text SVM	0.475
Image-text LDA	0.492
Multimodal DBN	0.566

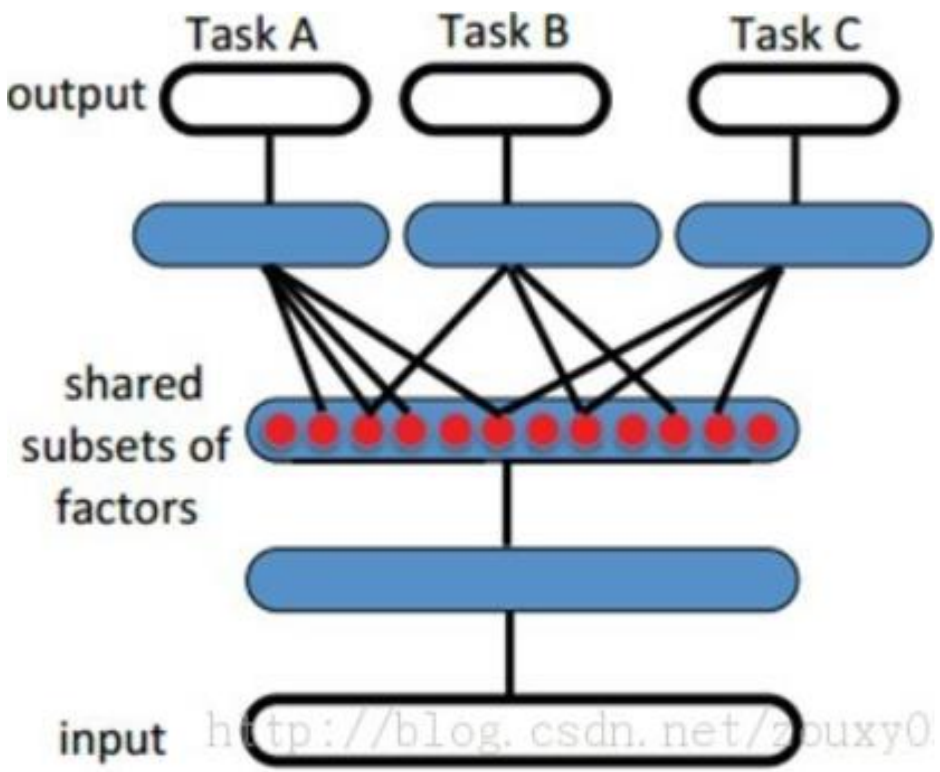
深度学习的应用

- 深度学习在多模态学习中的应用



深度学习的应用

- 深度学习在多任务学习中的应用



深度学习的应用

- 深度学习在多任务学习中的应用

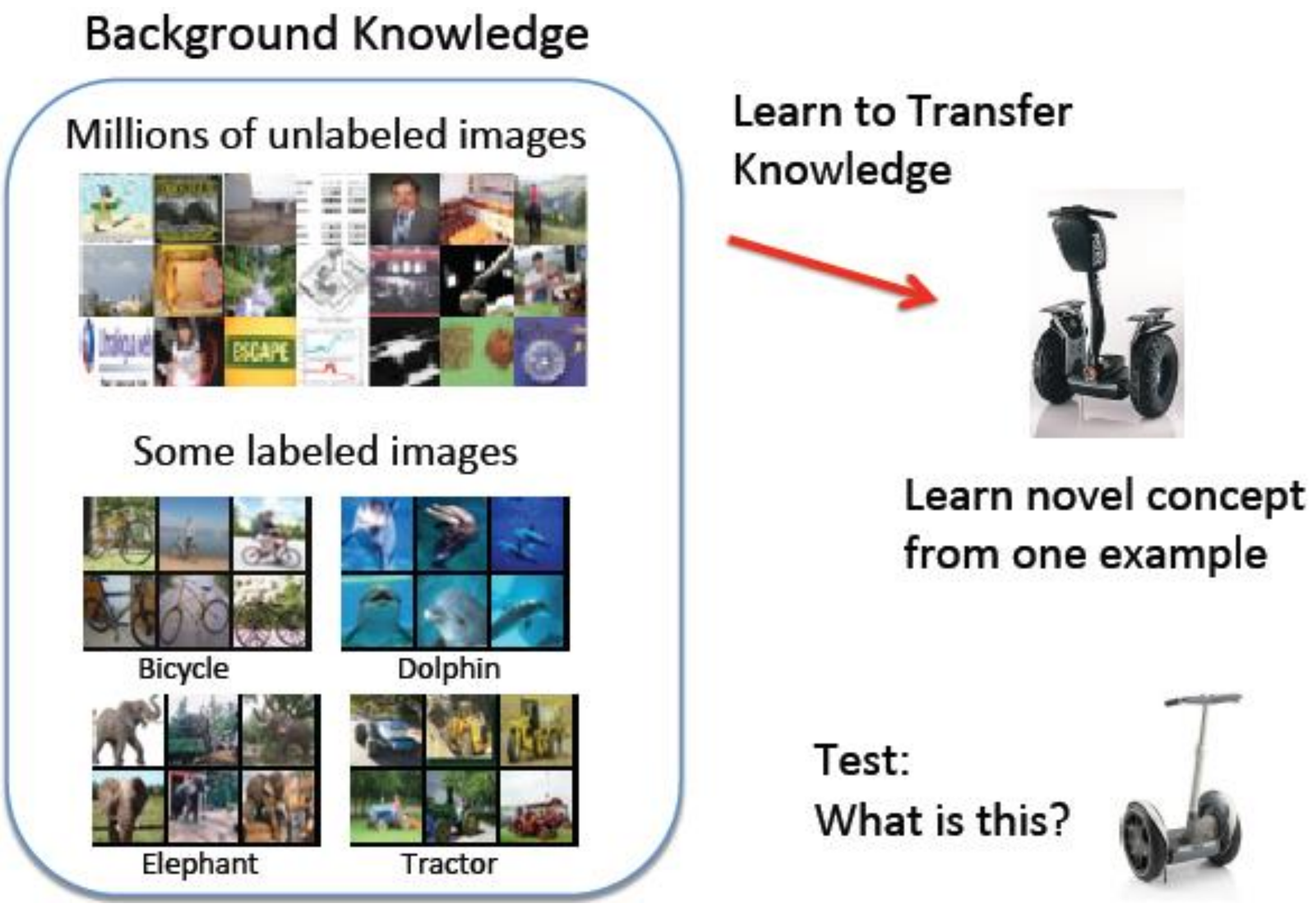
✓ 在深度学习模型中，对于相关任务的联合学习，往往会取得较好的特征表达；

✓ 多任务联合学习，能够增强损失函数的作用效能；

比如：单独进行人脸检测会比较难（光照、遮挡等因素），但是当人脸检测与人脸识别这两个相关的任务联合学习时，人脸检测的难度反而降低了。

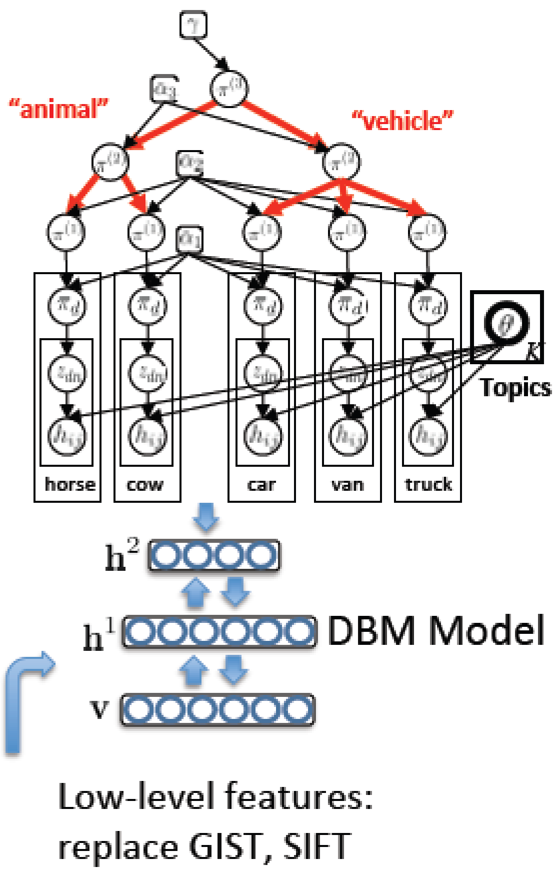
深度学习的应用

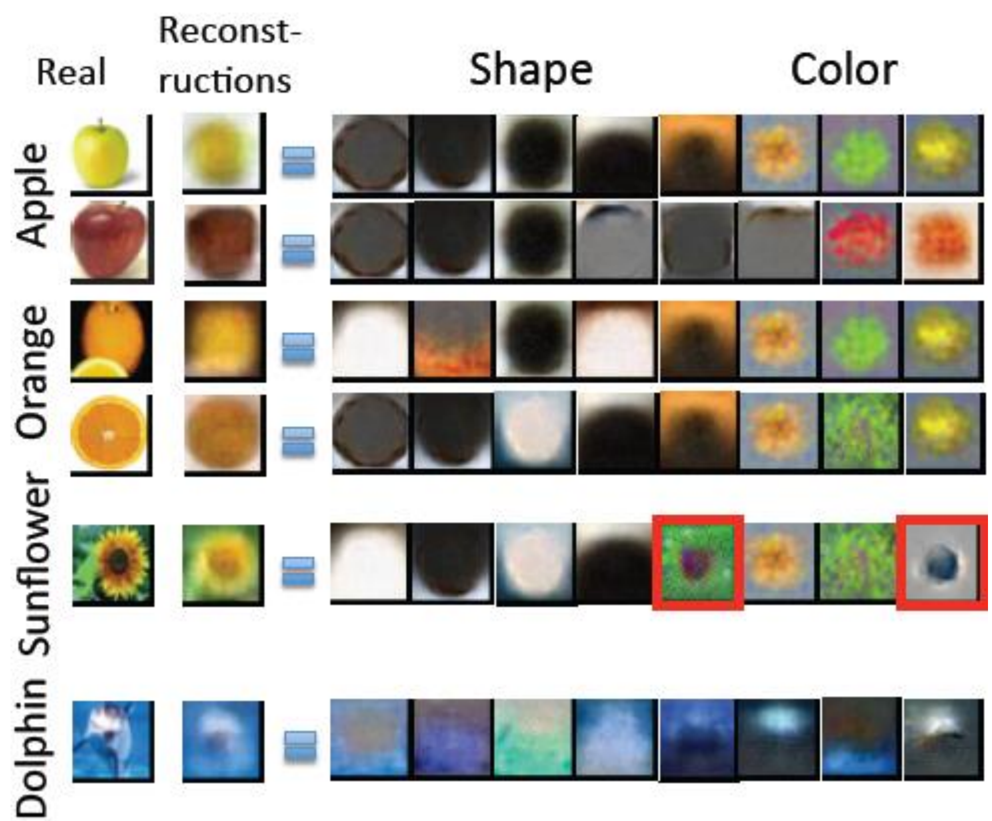
- 基于深度学习的迁移学习应用



深度学习的应用

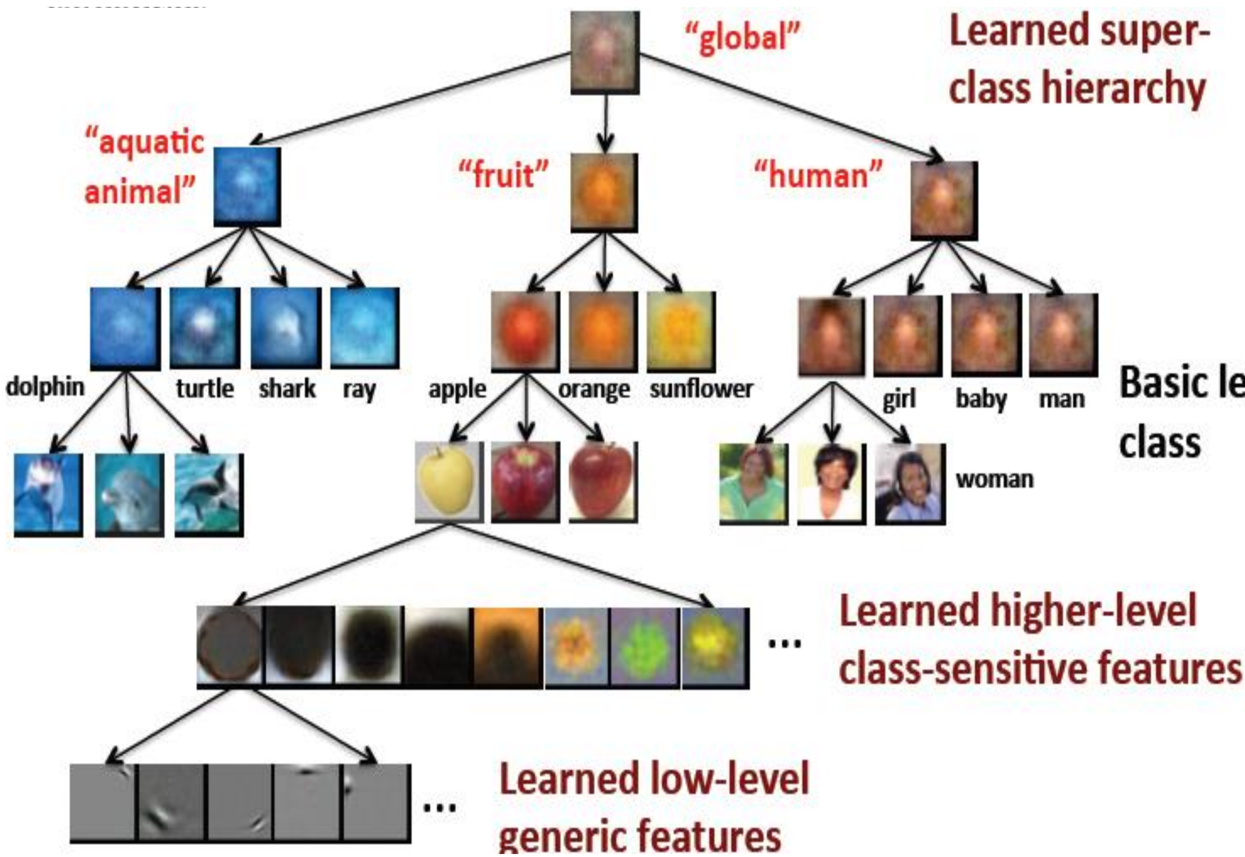
- 基于深度学习的迁移学习应用

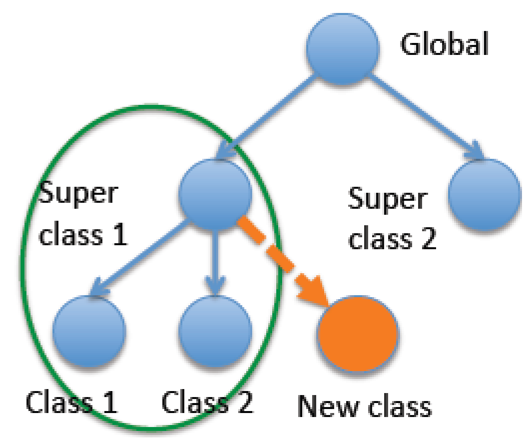




深度学习的应用

- 基于深度学习的迁移学习应用





深度学习的应用

- 深度学习在大尺度数据集上的应用

□大尺度数据集：

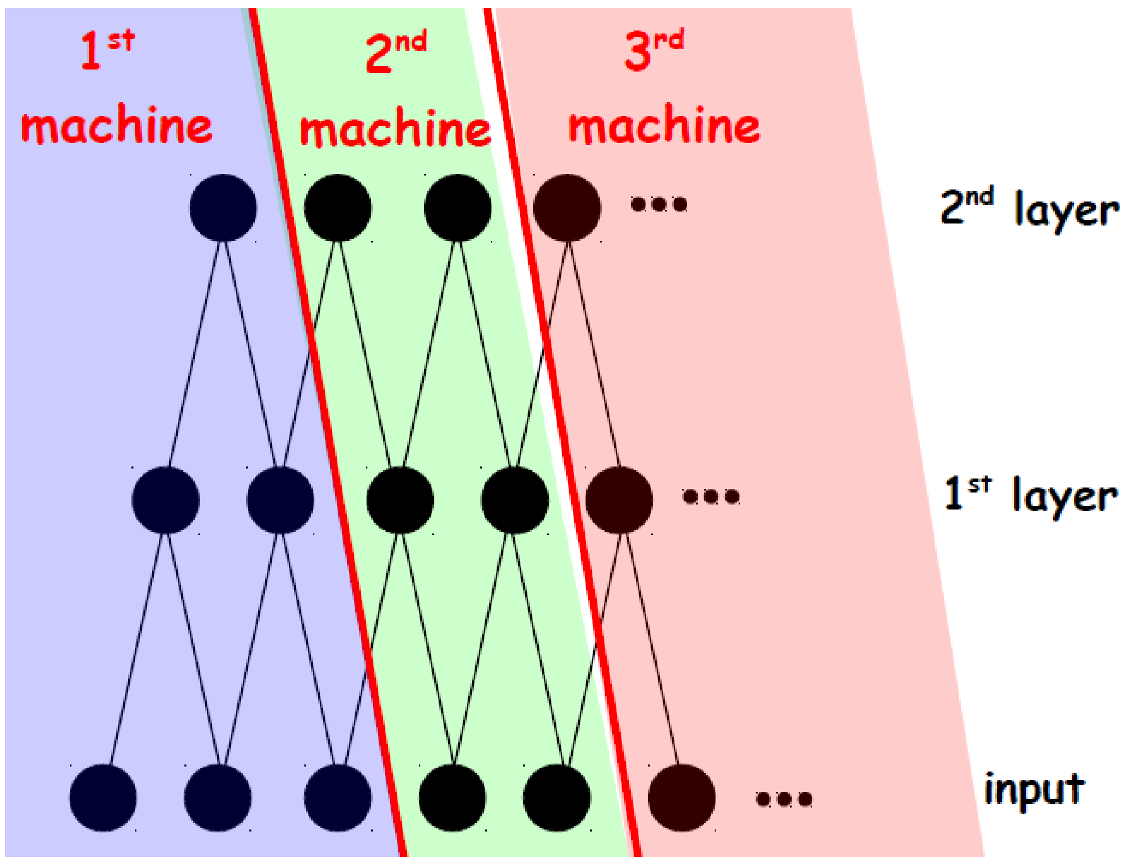
✓样本总数>100M,

✓类别总数>10K,

✓特征维度>10K

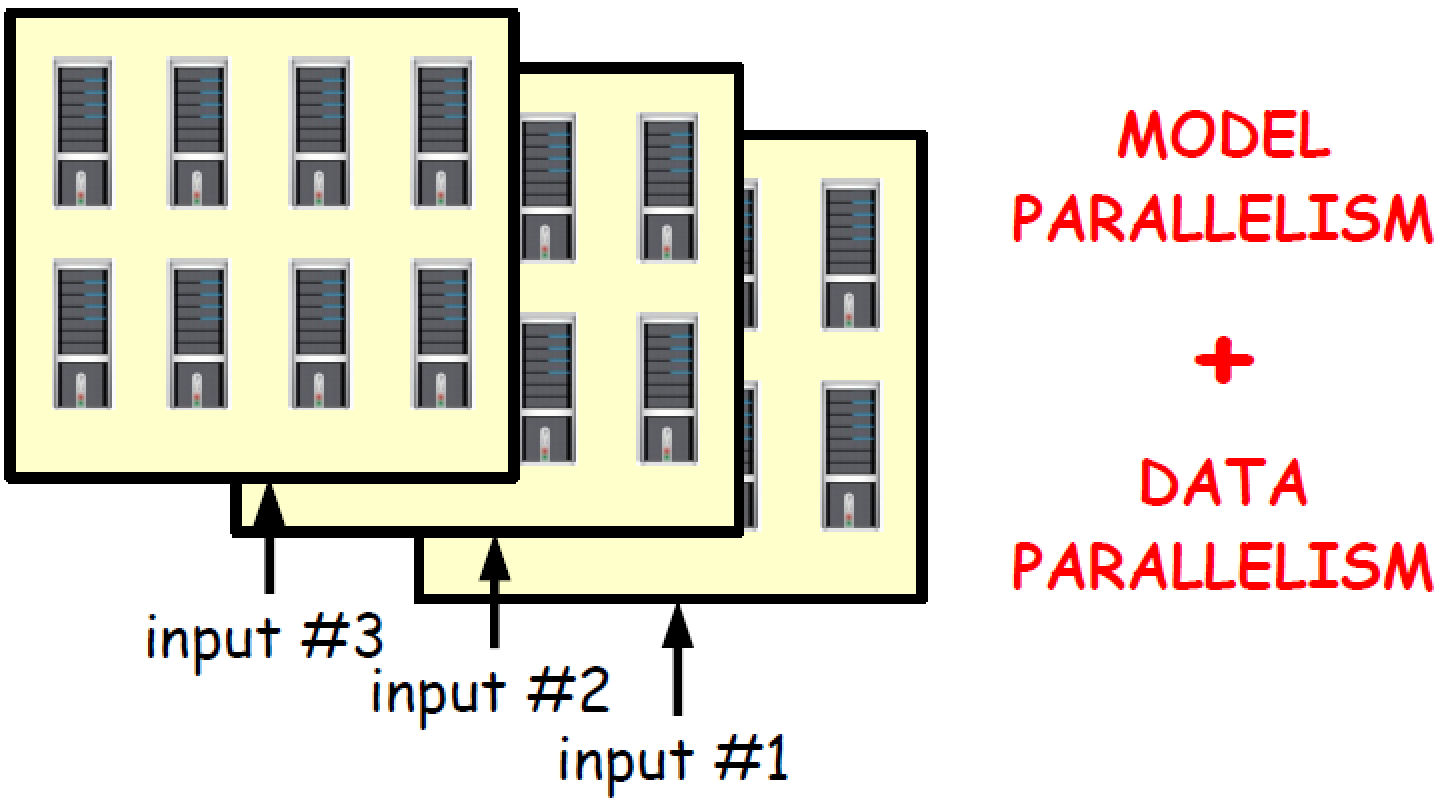
深度学习的应用

- 深度学习在大尺度数据集上的应用



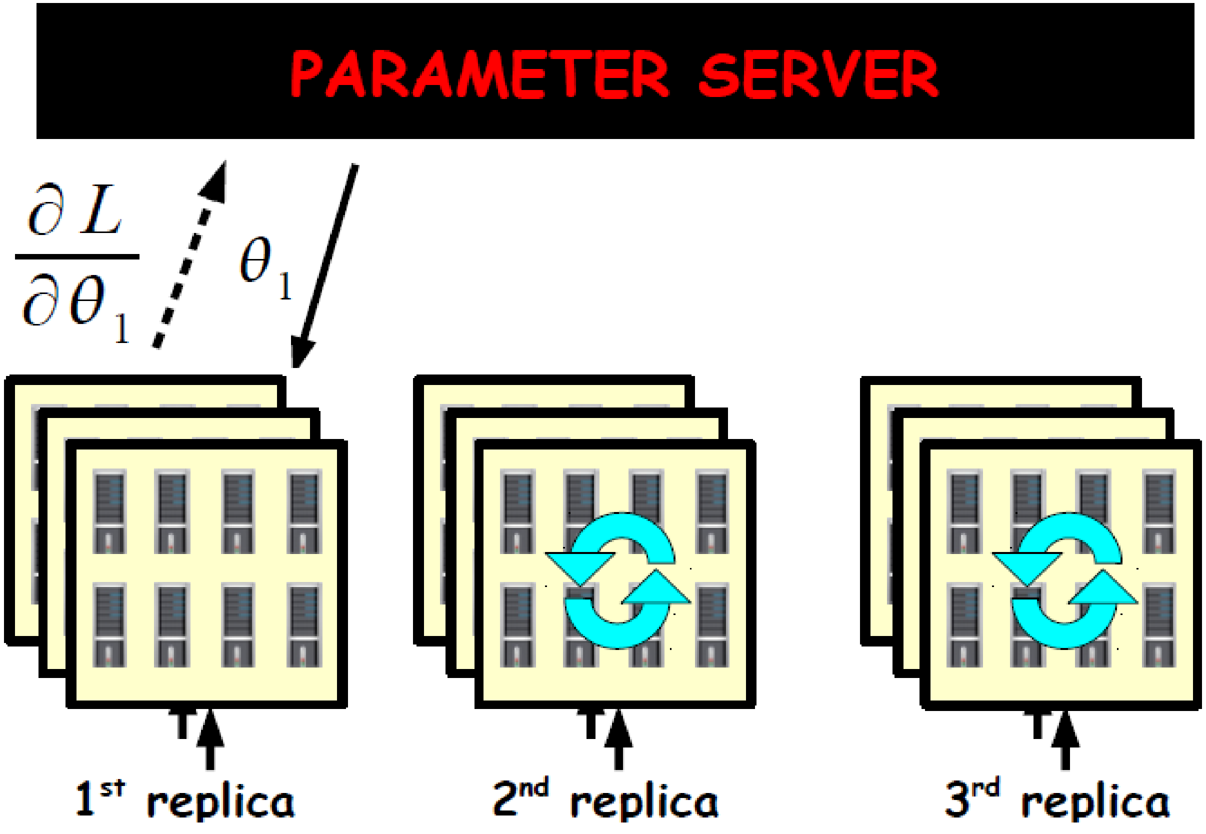
深度学习的应用

- 深度学习在大尺度数据集上的应用



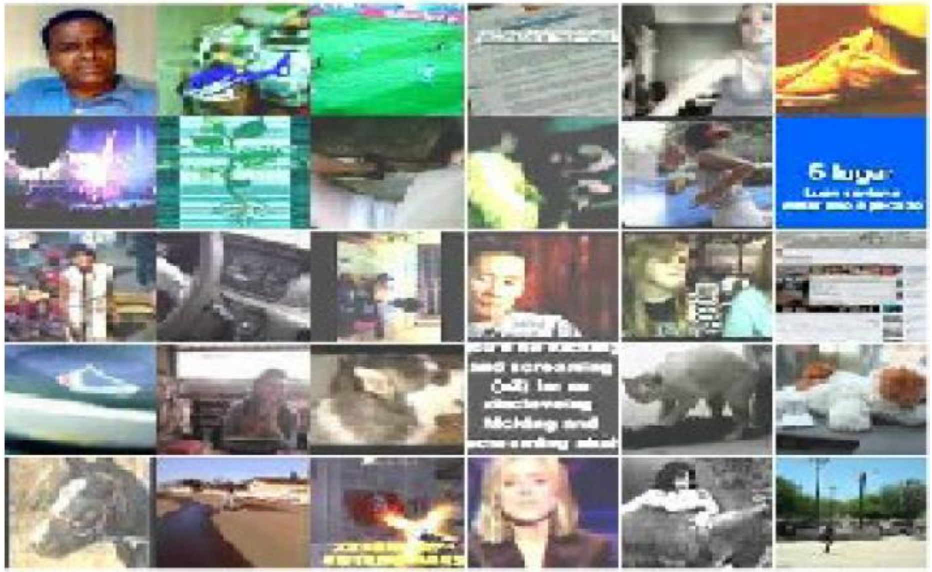
深度学习的应用

- 深度学习在大尺度数据集上的应用



深度学习的应用

- 深度学习在大尺度数据集上的应用



IMAGENET v.2011 (16M images, 20K categories)

METHOD	ACCURACY %
Weston & Bengio 2011	9.3
Linear Classifier on deep features	13.1
Deep Net (from random)	13.6
Deep Net (from unsup.)	15.8

深度学习的应用

- 深度学习的State-of-the-art

Images

CIFAR Object classification	Accuracy
Prior art (Ciresan et al., 2011)	80.5%
Stanford Feature learning	**82.0%**

NORB Object classification	Accuracy
Prior art (Scherer et al., 2010)	94.4%
Stanford Feature learning	**95.0%**

Video

Hollywood2 Classification	Accuracy
Prior art (Laptev et al., 2004)	48%
Stanford Feature learning	**53%**

YouTube	Accuracy
Prior art (Liu et al., 2009)	71.2%
Stanford Feature learning	**75.8%**

KTH	Accuracy
Prior art (Wang et al., 2010)	92.1%
Stanford Feature learning	**93.9%**

UCF	Accuracy
Prior art (Wang et al., 2010)	85.6%
Stanford Feature learning	**86.5%**

Text/NLP

Paraphrase detection	Accuracy
Prior art (Das & Smith, 2009)	76.1%
Stanford Feature learning	**76.4%**

Sentiment (MR/MPQA data)	Accuracy
Prior art (Nakagawa et al., 2010)	77.3%
Stanford Feature learning	**77.7%**

Multimodal (audio/video)

AVLetters Lip reading	Accuracy
Prior art (Zhao et al., 2009)	58.9%
Stanford Feature learning	**65.8%**

Other unsupervised feature learning records:
Pedestrian detection (Yann LeCun)
Speech recognition (Geoff Hinton)
PASCAL VOC object classification (Kai Yu)

Andrew

动 机

传统的模式识别方法：

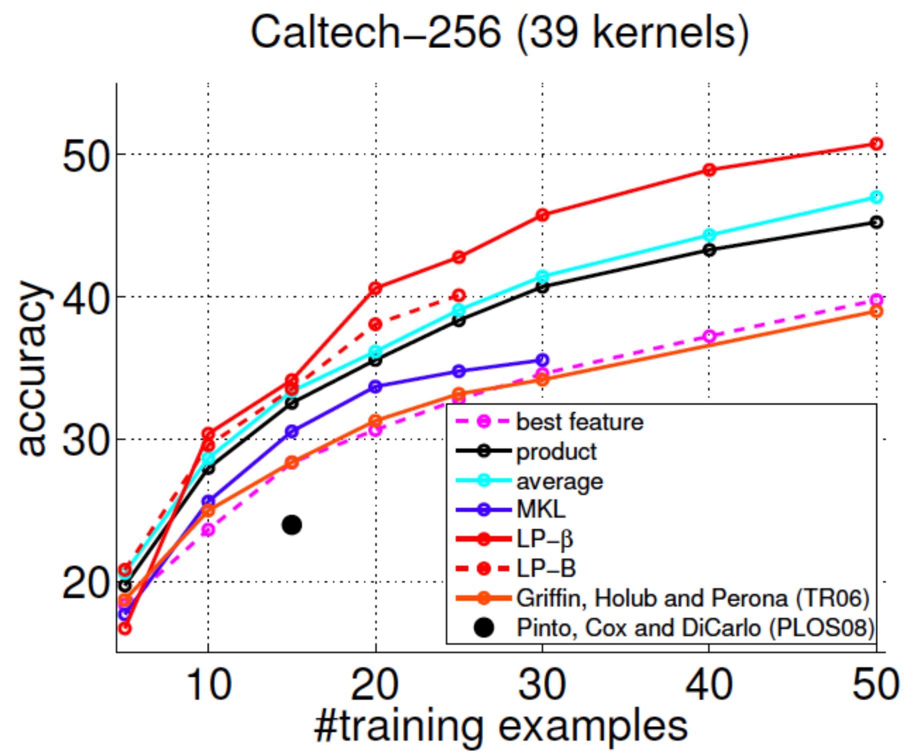
Low-level sensing → Pre-processing → Feature extract. → Feature selection → Inference: prediction, recognition

- 良好的特征表达，对最终算法的准确性起了非常关键的作用；
- 识别系统主要的计算和测试工作耗时主要集中在特征提取部分；
- 特征的样式目前一般都是人工设计的，靠人工提取特征。

动 机——为什么要自动学习特征

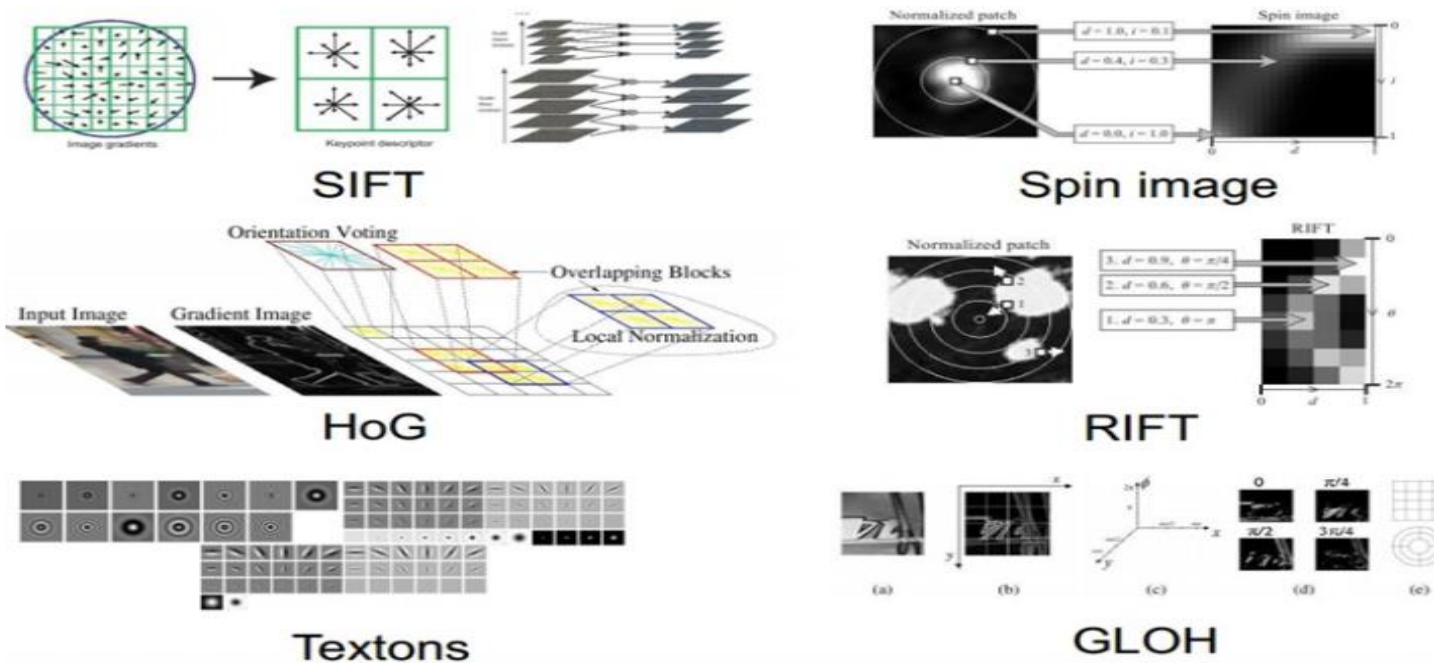
- 实验：LP-β Multiple Kernel Learning
  - Gehler and Nowozin, On Feature Combination for Multiclass Object Classification, ICCV’09
- 采用39 个不同的特征
  - PHOG, SIFT, V1S+, Region Cov. Etc.
- 在普通特征上MKL表现有限

结论：特征更重要



动 机——为什么要自动学习特征

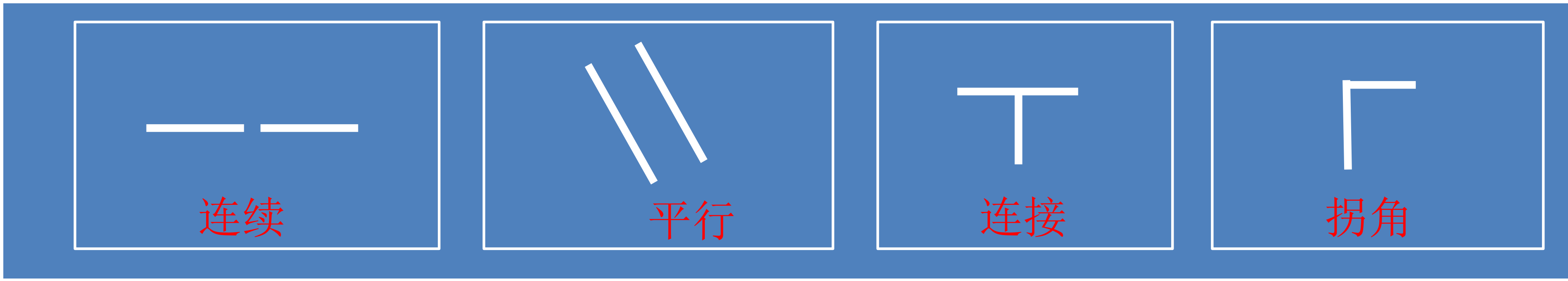
- 机器学习中，获得好的特征是识别成功的关键
- 目前存在大量人工设计的特征，不同研究对象特征不同，特征具有多样性，如：SIFT, HOG, LBP等
- 手工选取特征费时费力，需要启发式专业知识，很大程度上靠经验和运气
- 是否能自动地学习特征？



动 机——为什么要自动学习特征

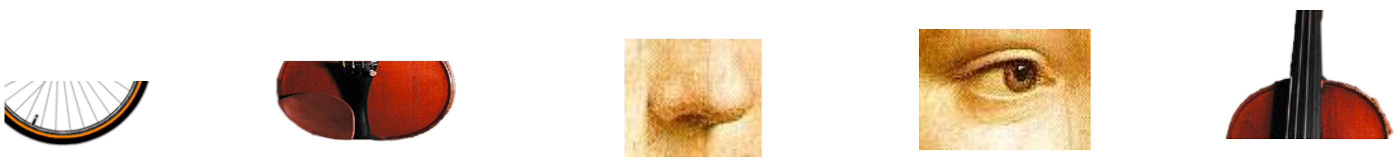
- 中层特征

✓ 中层信号：



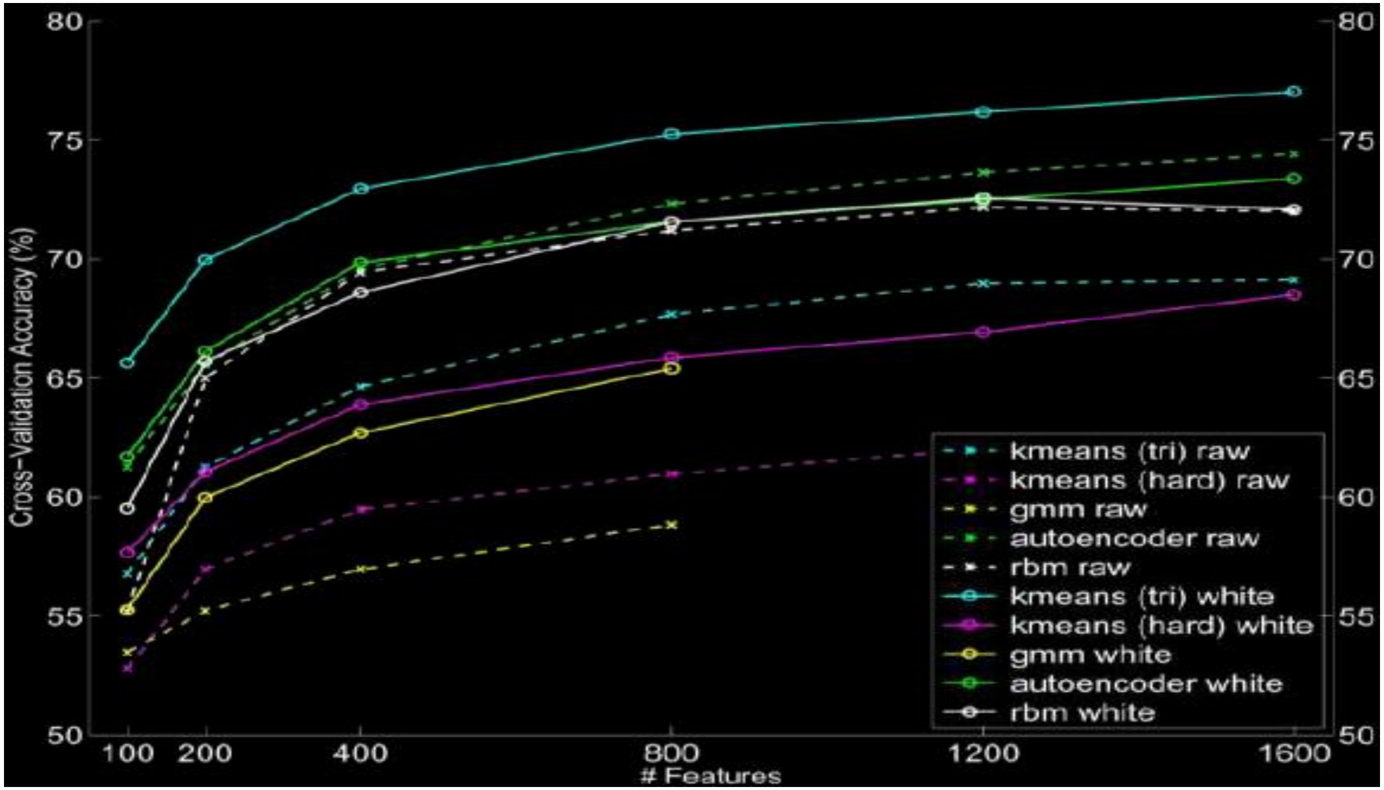
“Tokens” from Vision by D.Marr:

✓ 物体部件：



- 他们对于人工而言是十分困难的，那么如何学习呢？

动 机——为什么要自动学习特征



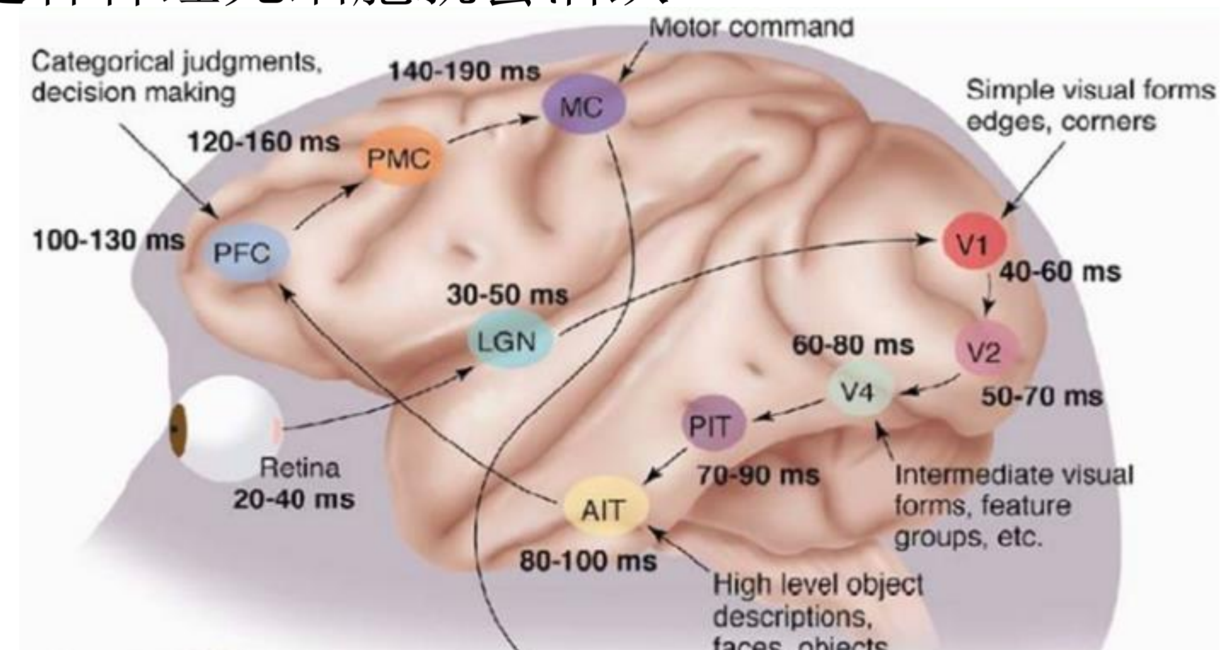
- 一般而言，特征越多，给出信息就越多，识别准确性会得到提升；
- 但特征多，计算复杂度增加，探索的空间大，可以用来训练的数据在每个特征上就会稀疏。
- **结论：不一定特征越多越好！需要有多少个特征，需要学习确定。**

动 机——为什么采用层次网络结构

- **人脑视觉机理**

✓ 1981年的诺贝尔医学奖获得者 David Hubel和TorstenWiesel发现了视觉系统的信息处理机制

✓ 发现了一种被称为“方向选择性细胞的神经元细胞，当瞳孔发现了眼前的物体的边缘，而且这个边缘指向某个方向时，这种神经元细胞就会活跃



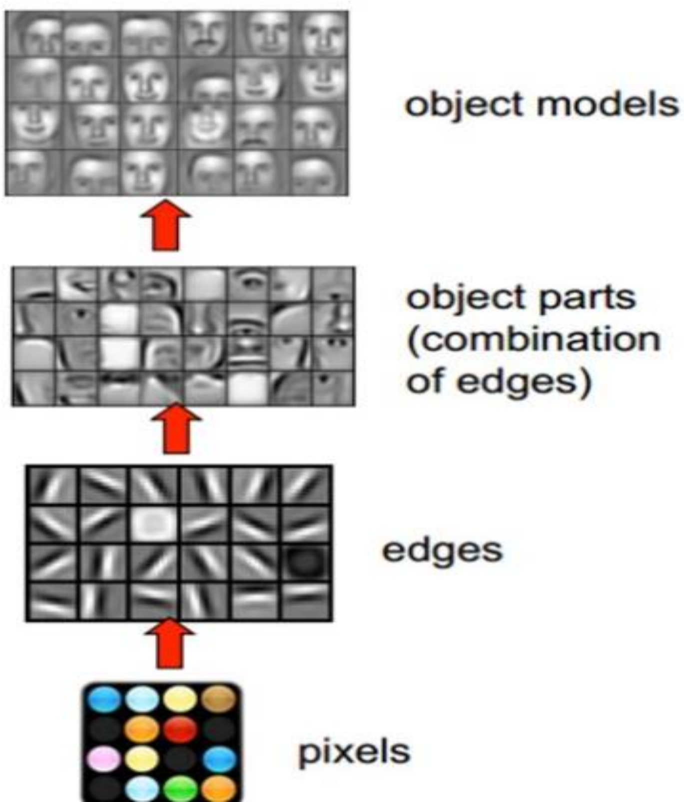
动 机——为什么采用层次网络结构

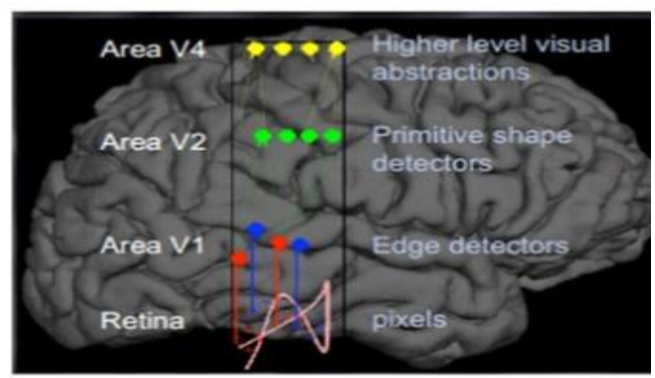
- **人脑视觉机理**

✓ 人的视觉系统的信息处理是分级的

✓ 高层的特征是低层特征的组合，从低层到高层的特征表示越来越抽象，越来越能表现语义或者意图

✓ 抽象层面越高，存在的可能猜测就越少，就越利于分类



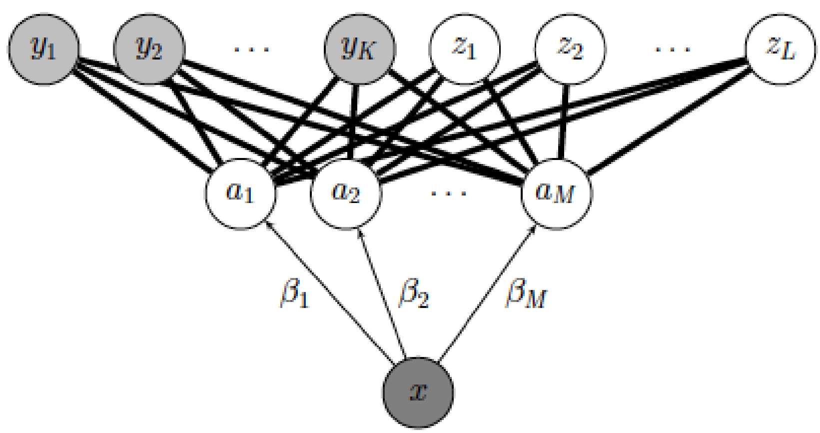


动 机——为什么采用层次网络结构

- 视觉的层次性

✓ 属性学习，类别作为属性的一种组合映射

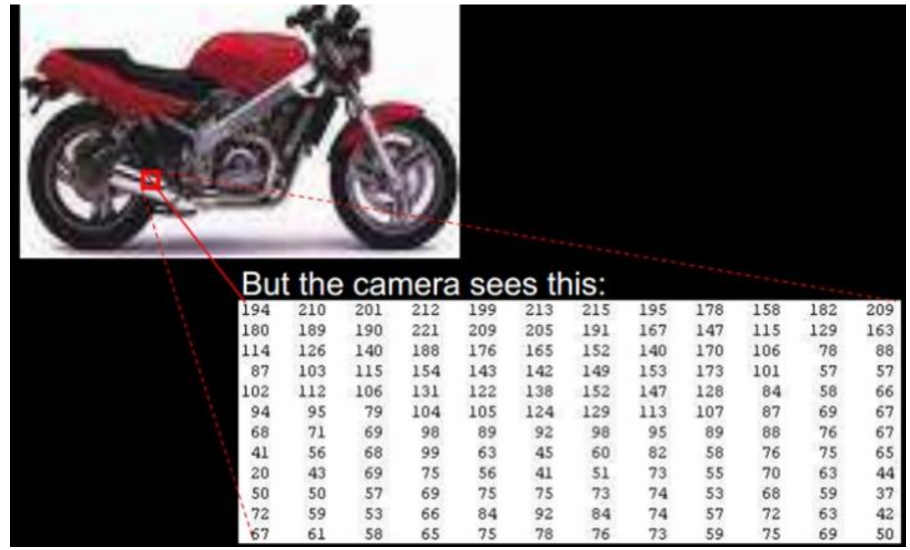
Lampert et al. CVPR’09

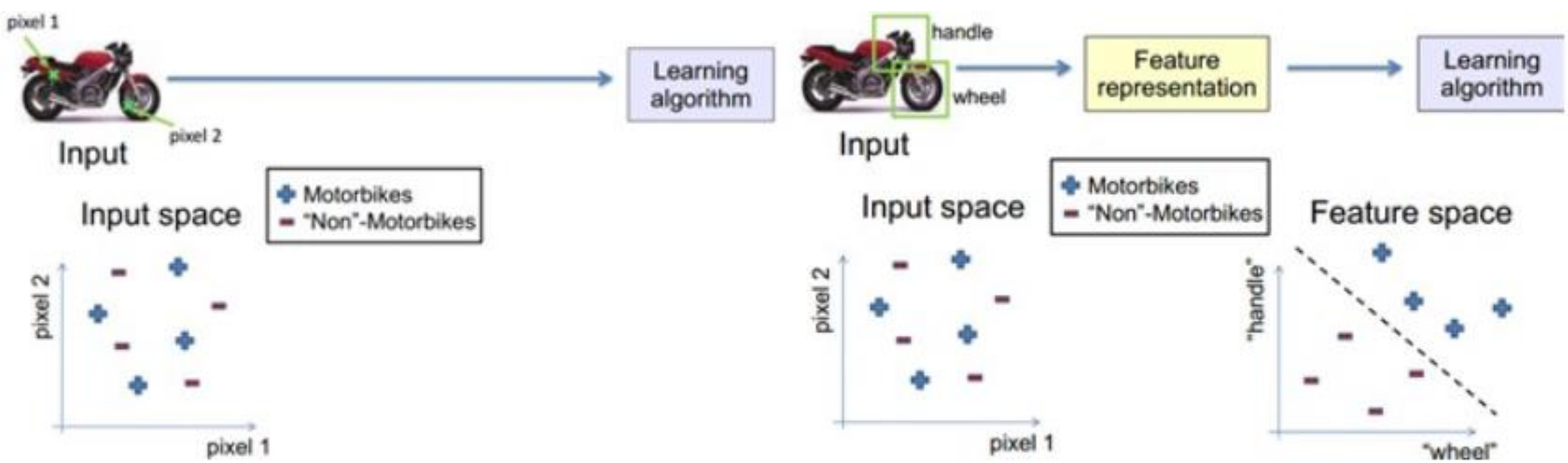


动 机——为什么采用层次网络结构

- 特征表示的粒度

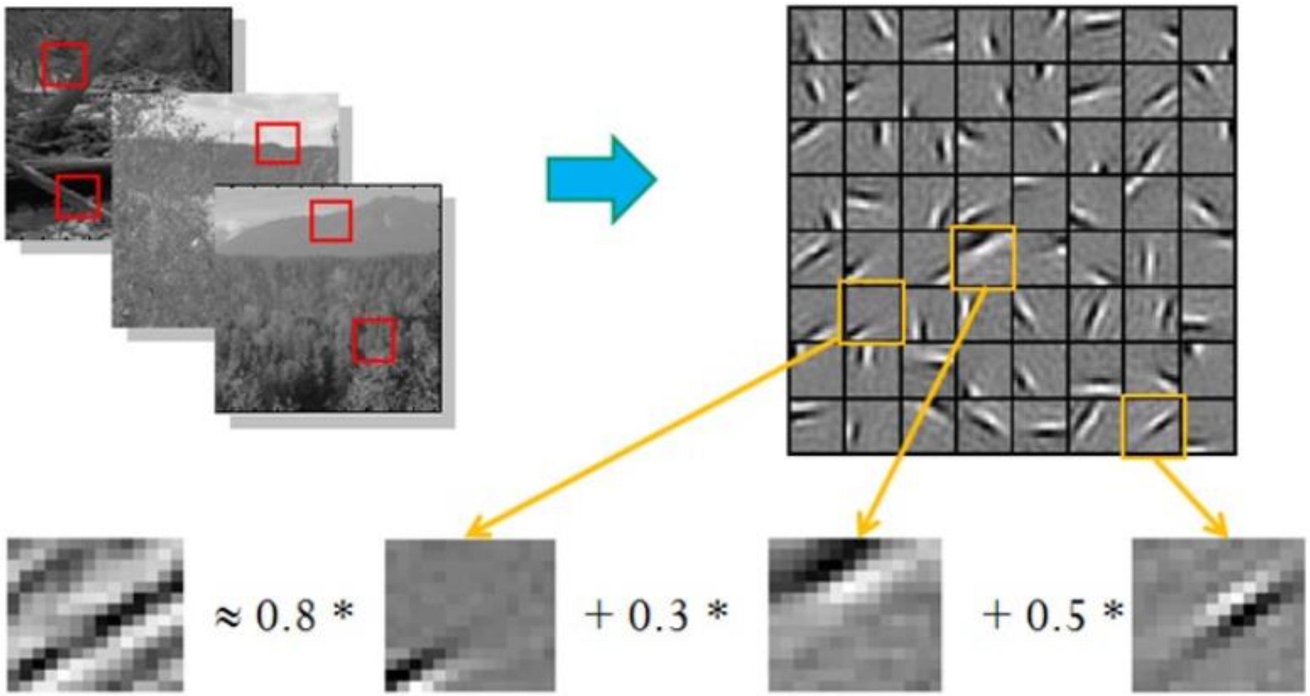
✓ 具有结构性（或者语义）的高层特征对于分类更有意义





动 机——为什么采用层次网络结构

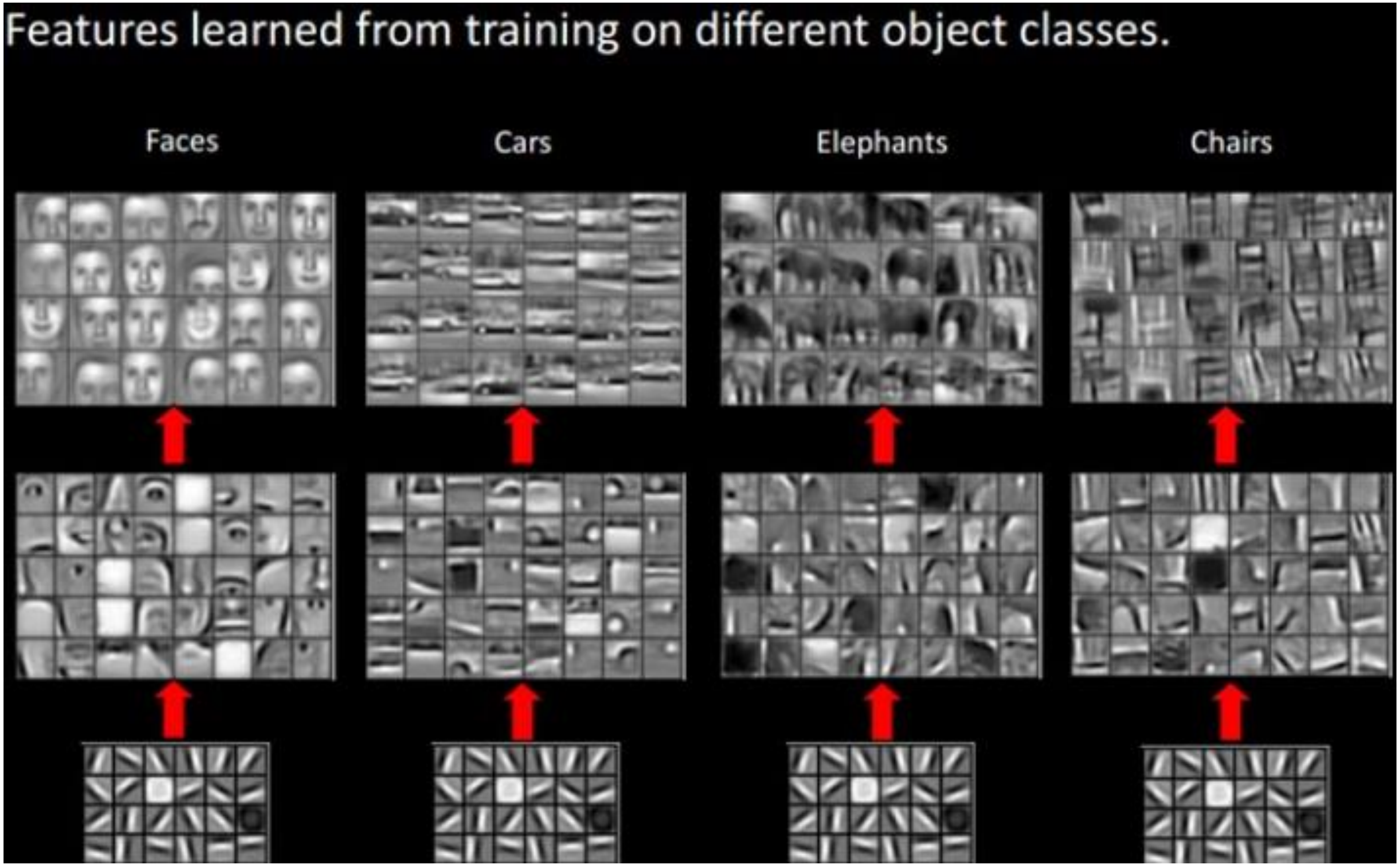
- 初级（浅层）特征表示



$[a_1, \ldots, a_{64}] = [0, 0, \ldots, 0, \mathbf{0.8}, 0, \ldots, 0, \mathbf{0.3}, 0, \ldots, 0, \mathbf{0.5}, 0]$
(feature representation)

动 机——为什么采用层次网络结构

- 结构性特征表示



动 机——为什么采用层次网络结构

- **浅层学习的局限**

✓人工神经网络（BP算法）

—虽被称作多层感知机，但实际是种只含有一层隐层节点的浅层模型

✓SVM、Boosting、最大熵方法（如LR，Logistic Regression）

—带有一层隐层节点（如SVM、Boosting），或没有隐层节点（如LR）的浅层模型

局限性：有限样本和计算单元情况下对复杂函数的表示能力有限，针对复杂分类问题其泛化能力受限。

深度学习

- **2006年，加拿大多伦多大学教授、机器学习领域的泰斗Geoffrey Hinton在《科学》上发表论文提出深度学习主要观点：**

  1）多隐层的人工神经网络具有优异的特征学习能力，学习得到的特征对数据有更本质的刻画，从而有利于可视化或分类；

  2）深度神经网络在训练上的难度，可以通过“逐层初始化”（layer-wise pre-training）来有效克服，逐层初始化可通过无监督学习实现的。

深度学习

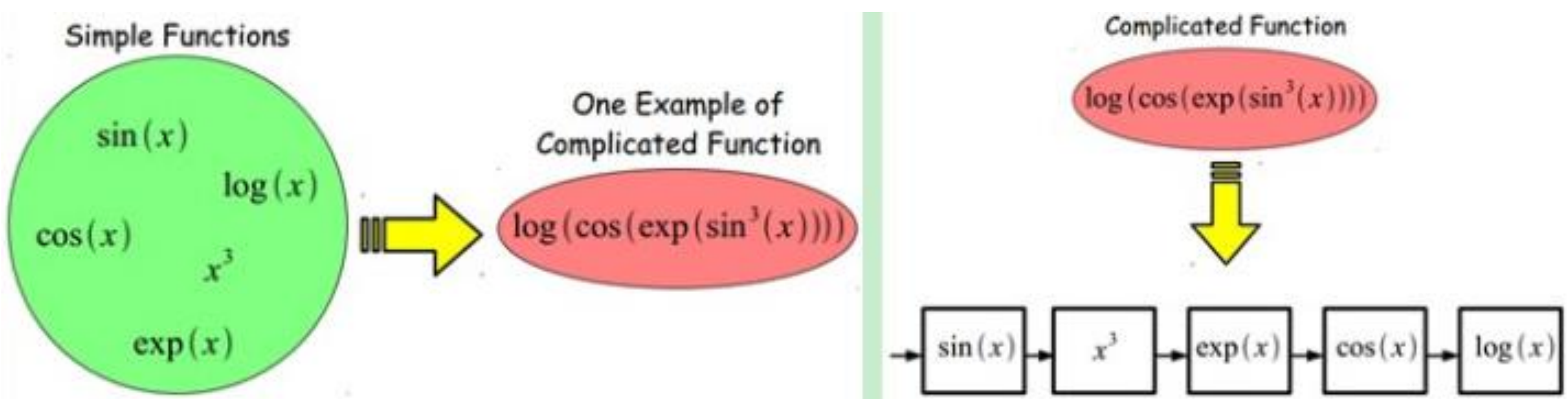
- 本质：通过构建多隐层的模型和海量训练数据（可为无标签数据），来学习更有用的特征，从而最终提升分类或预测的准确性。 “深度模型”是手段，“特征学习”是目的。
- 与浅层学习区别：

1）强调了模型结构的深度，通常有5-10多层的隐层节点；

2）明确突出了特征学习的重要性，通过逐层特征变换，将样本在原空间的特征表示变换到一个新特征空间，从而使分类或预测更加容易。与人工规则构造特征的方法相比，利用大数据来学习特征，更能够刻画数据的丰富内在信息。

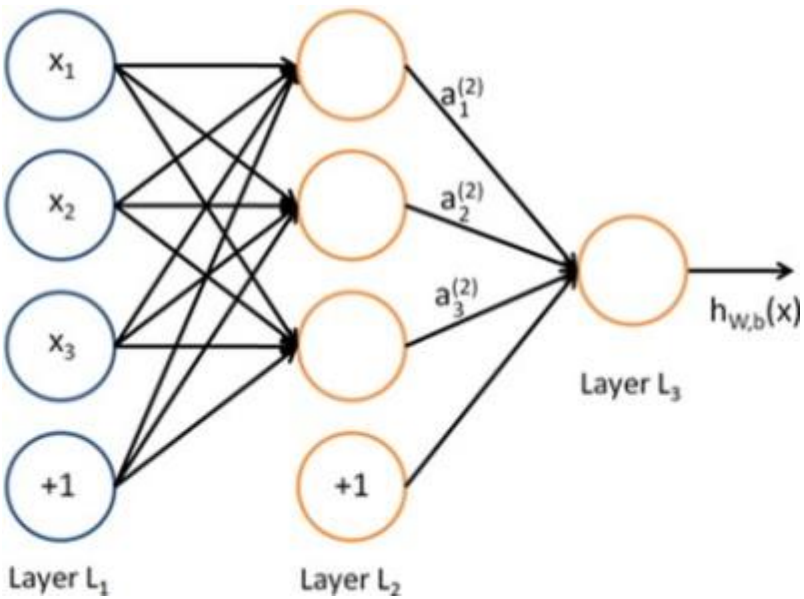
深度学习

- 好处：可通过学习一种深层非线性网络结构，实现复杂函数逼近，表征输入数据分布式表示。

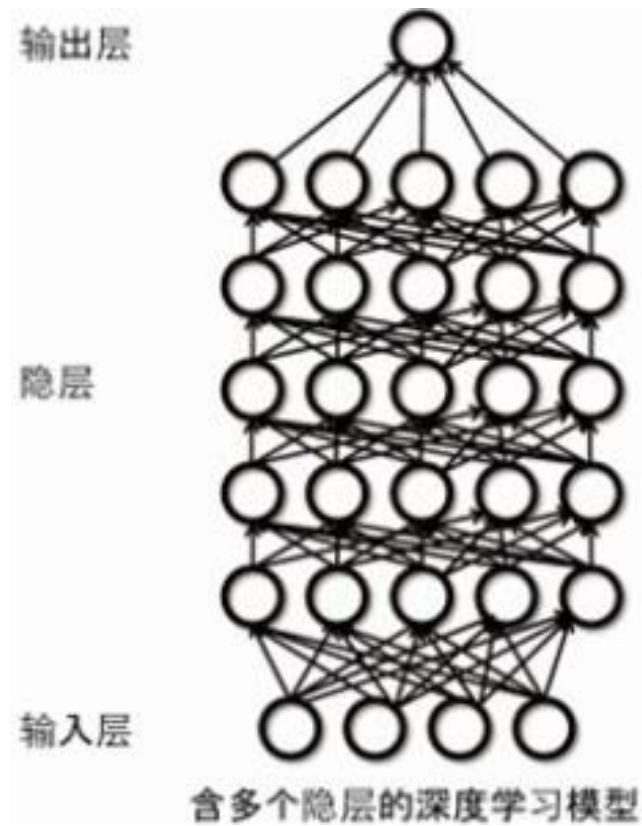


深度学习 vs. 神经网络

<u>神经网络</u>：



<u>深度学习</u>：



深度学习 vs. 神经网络

相同点： 二者均采用分层结构，系统包括输入层、隐层（多层）、输出层组成的多层网络，只有相邻层节点之间有连接，同一层以及跨层节点之间相互无连接，每一层可以看作是一个logistic 回归模型。

不同点：

神经网络：采用BP算法调整参数，即采用迭代算法来训练整个网络。随机设定初值，计算当前网络的输出，然后根据当前输出和样本真实标签之间的差去改变前面各层的参数，直到收敛；

深度学习：采用逐层训练机制。采用该机制的原因在于如果采用BP机制，对于一个deep network（7层以上），残差传播到最前面的层将变得很小，出现所谓的gradient diffusion（梯度扩散）。

深度学习 vs. 神经网络

- 神经网络的局限性：

1）比较容易过拟合，参数比较难调整，而且需要不少技巧；

2）训练速度比较慢，在层次比较少（小于等于3）的情况下效果并不比其它方法更优；

Deep learning

Yoshua Bengio:

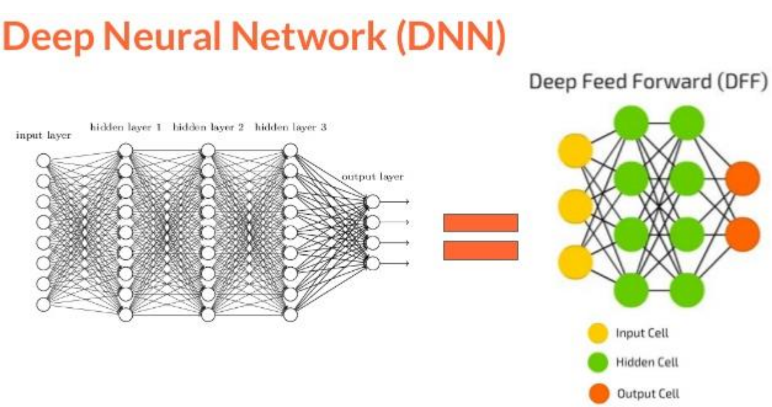
Science is NOT a battle, it is a collaboration. We all build on each other's ideas. Science is an act of love, not war. Love for the beauty in the world that surrounds us and love to share and build something together. That makes science a highly satisfying activity, emotionally speaking!

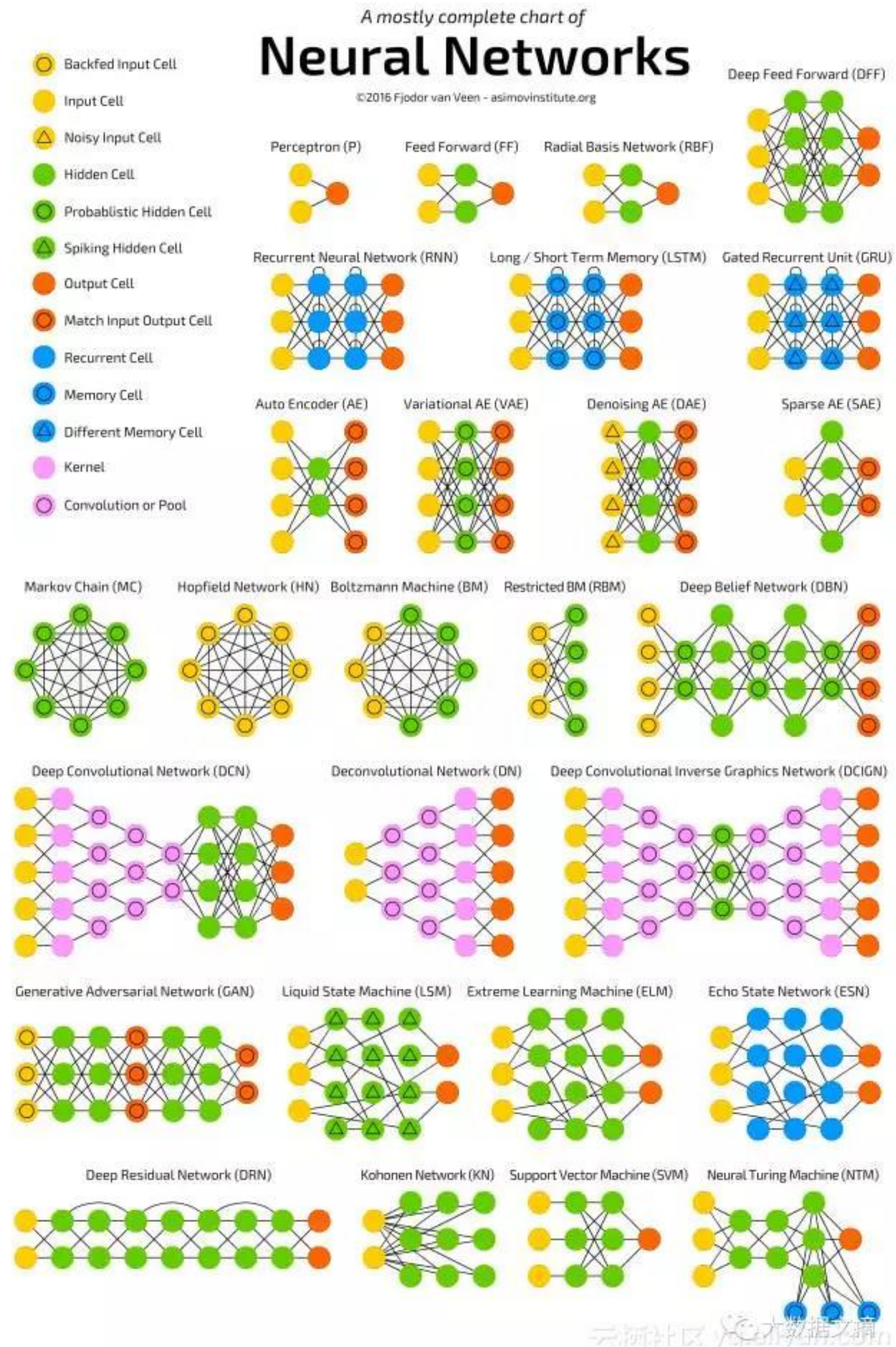
科学不是战争而是合作，任何学科的发展从来都不是一条路走到黑，而是同行之间互相学习、互相借鉴、博采众长、相得益彰，站在巨人的肩膀上不断前行。机器学习的研究也是一样，你死我活那是邪教，开放包容才是正道。

Part II

深度学习的基本方法
（以及神经网络算法）

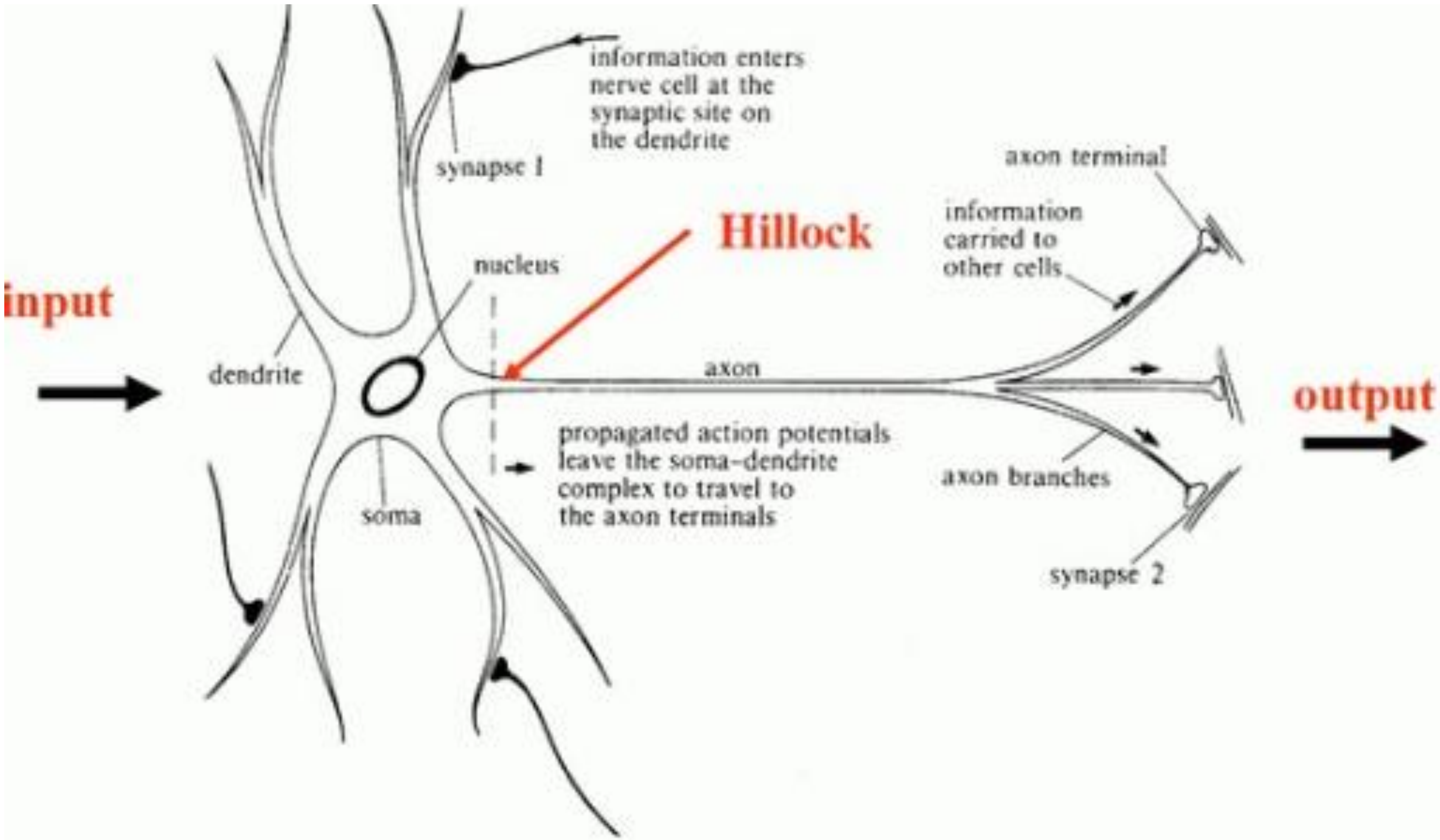
Neural network





Adapted from: http://www.asimovinstitute.org/neural-network-zoo/

Neural network

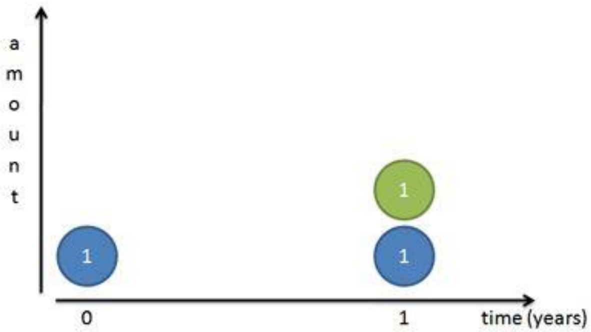


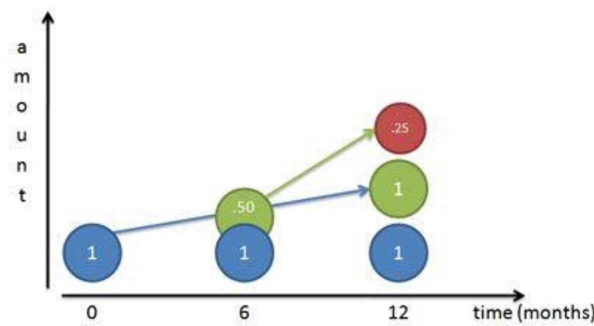
Neural network

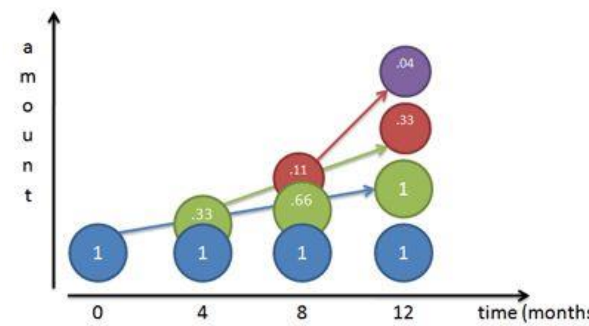
- 自然对数e
- 逻辑回归
- 神经网络
- 反向传播算法(BP算法)
- 示例

Neural network

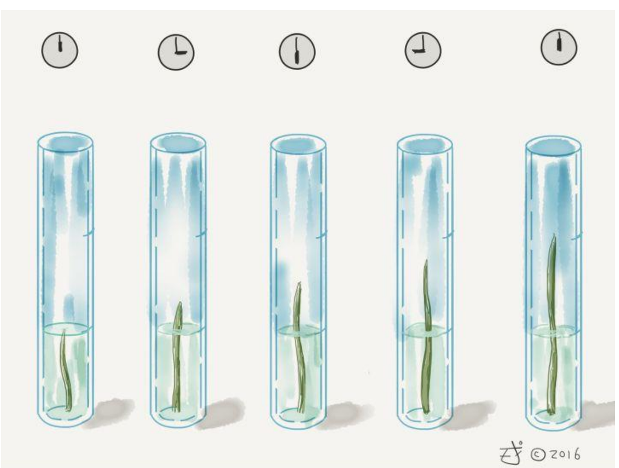
自然对数e



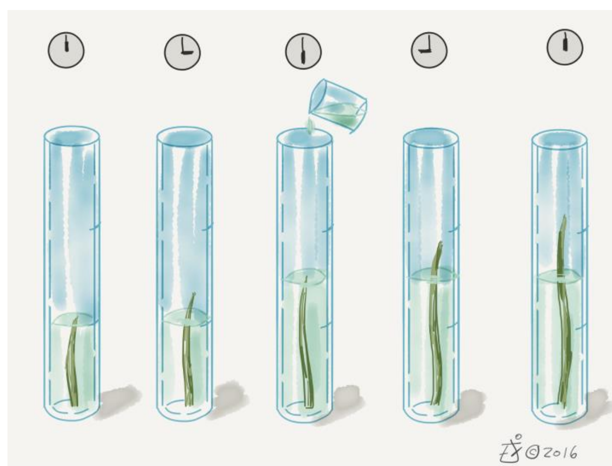




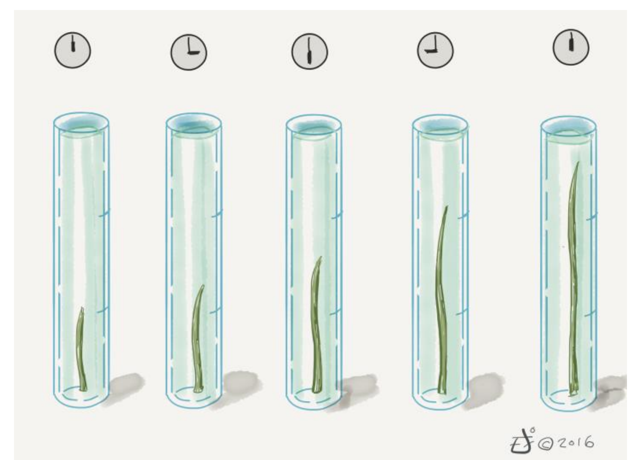
1元存1年，在年利率100%下，无论怎么利滚利，其余额总有一个上限，这个天花板就是e



有一种草齐头泡水半天变2倍



中间加一次水能长到2.25倍



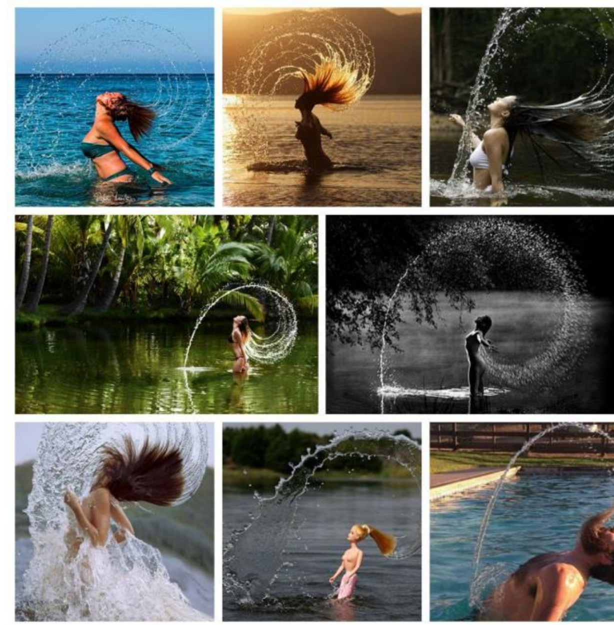
加足水最多能长到2.718...倍

Neural network

自然对数e

e = 1 + 1/1! + 1/2! + 1/3! + 1/4! + 1/5! + 1/6! + 1/7! + … = 1 + 1 + 1/2 + 1/6 + 1/24 + 1/120+ … ≈ 2.71828

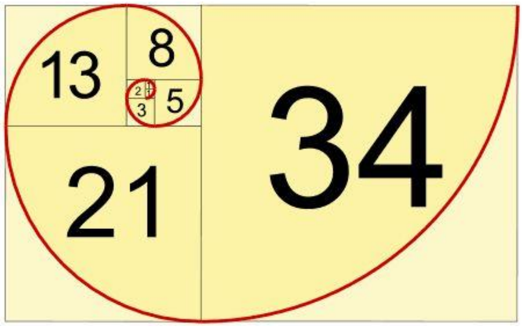
e=(1+1/x)^x≈ 2.71828



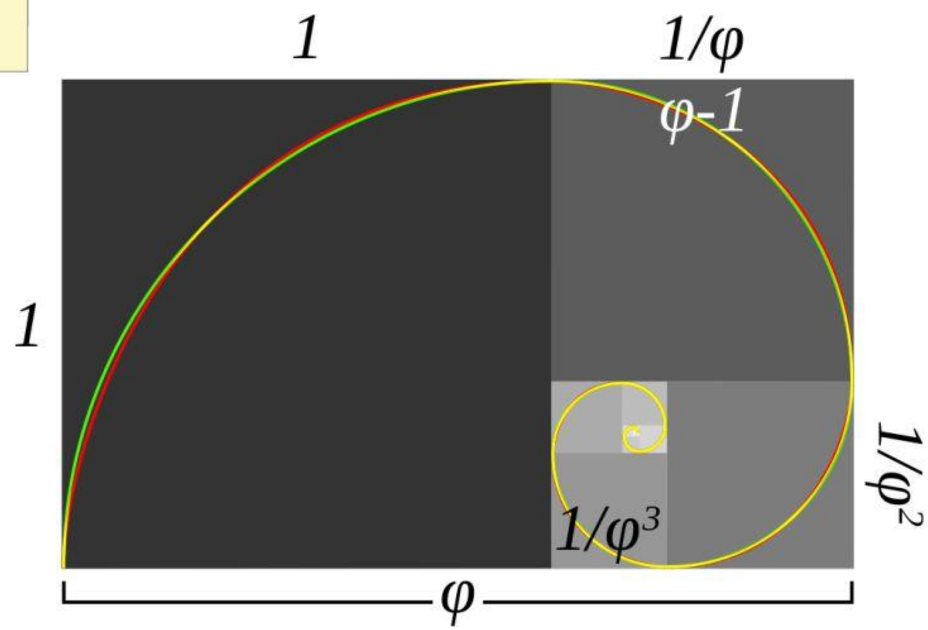
Reference: https://www.zhihu.com/question/20296247/answer/29370489

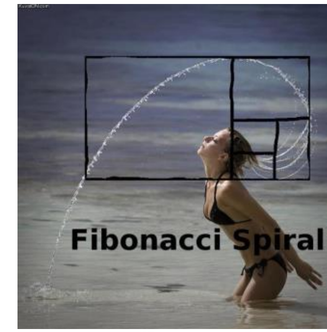
Neural network

自然对数e vs. 斐波拉切数列



e^{θ}



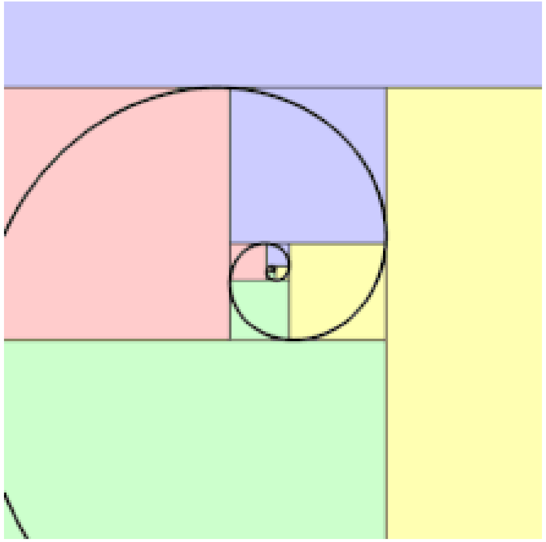


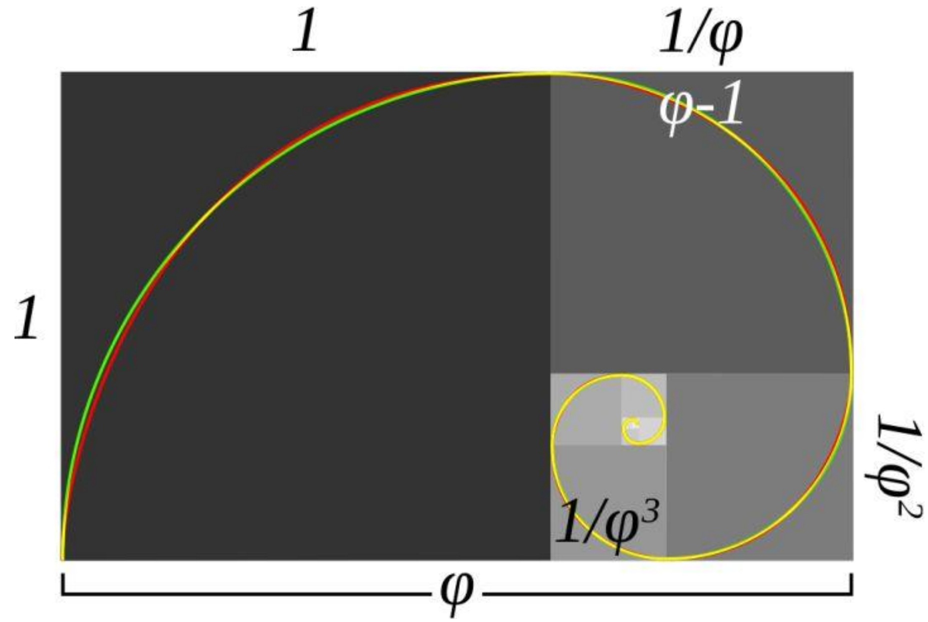
斐波那契螺线仅仅是对一种叫黄金螺线（Golden spiral）的近似，黄金螺线是一种内涵黄金分割比例的对数螺线。
红色的才是黄金曲线，绿色的是“假黄金螺线”（斐波那契螺线），近似却不重合。

Neural network

自然对数e & 黄金分割

$$e^{\theta}$$

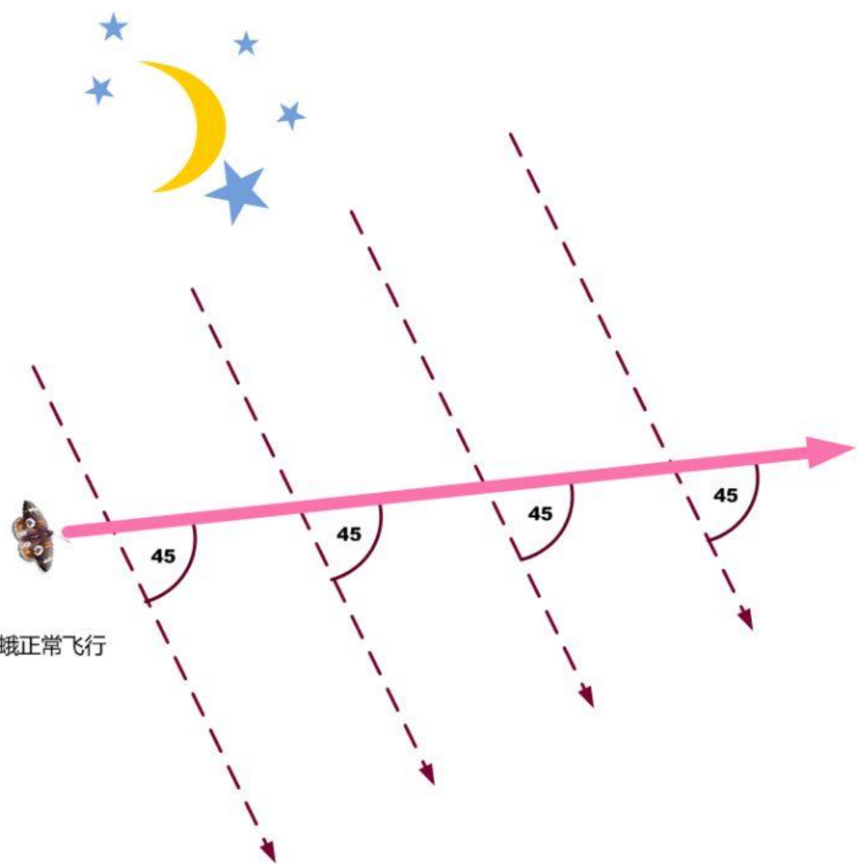




φ=1.6180339887498948482...

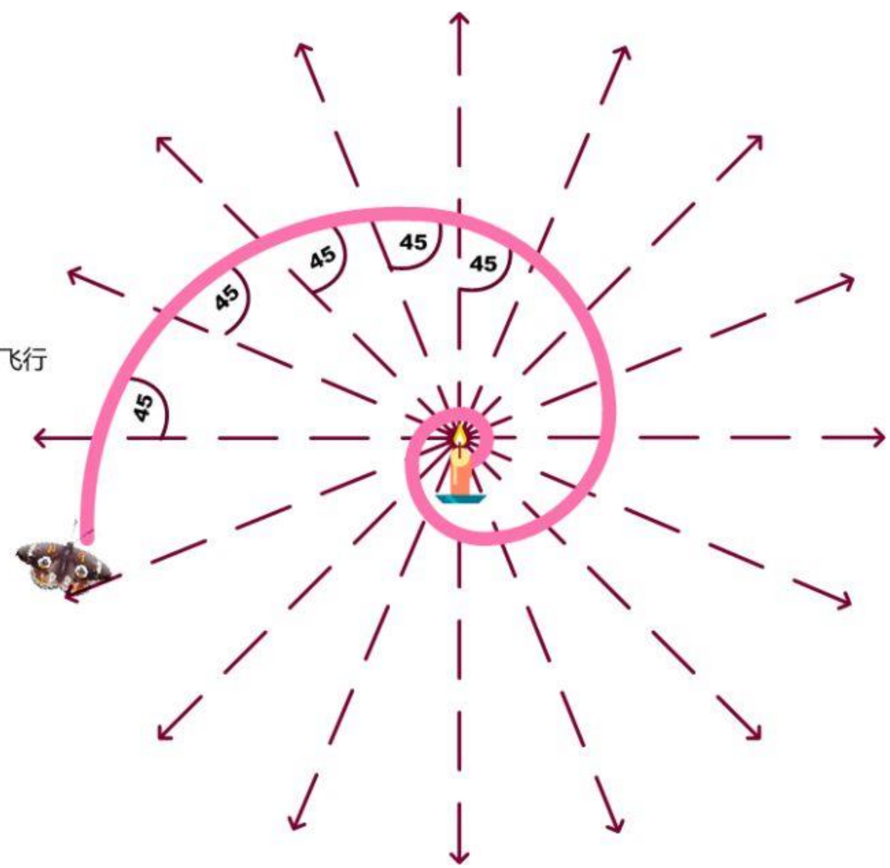
Neural network

自然对数e



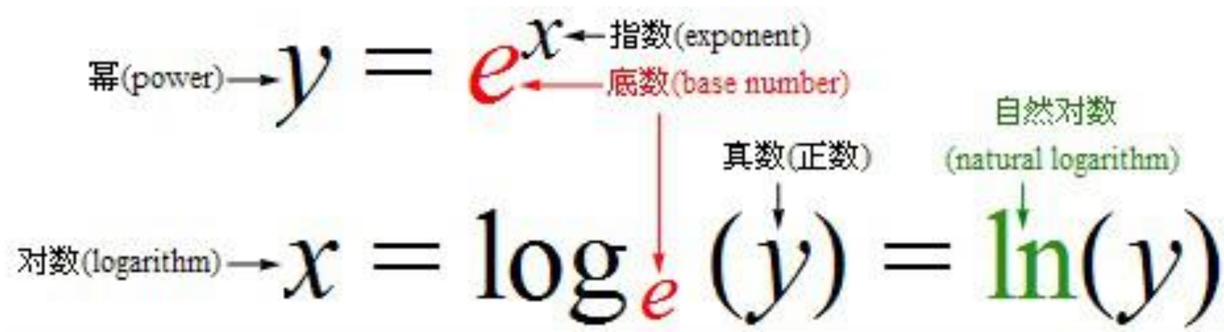
飞蛾正常飞行

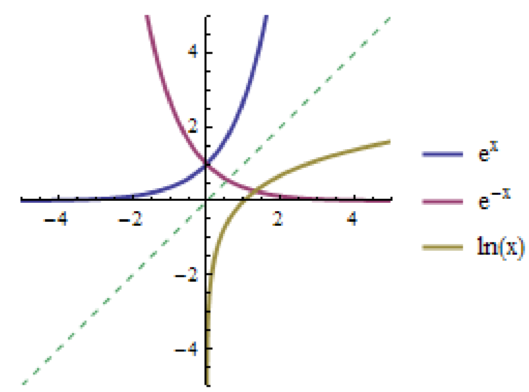
飞蛾扑火飞行



Neural network

自然对数e

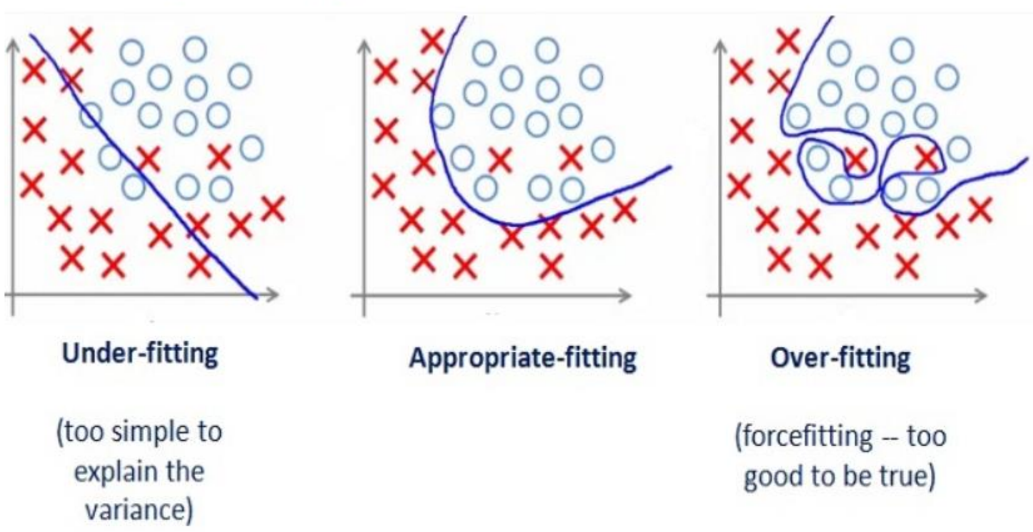


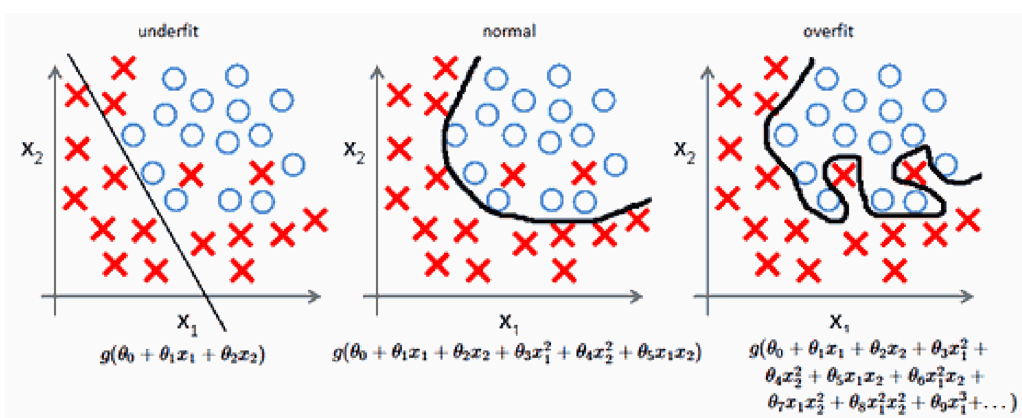


e^x和e^-x的图形是对称的；ln(x)是e^x的逆函数，它们呈45度对称。

Neural network

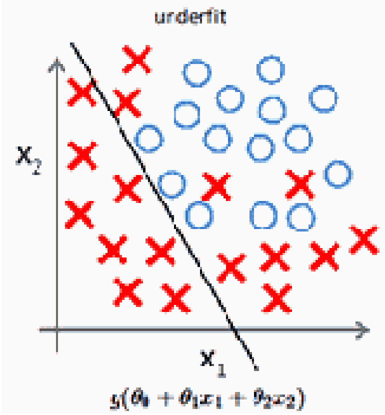
逻辑回归

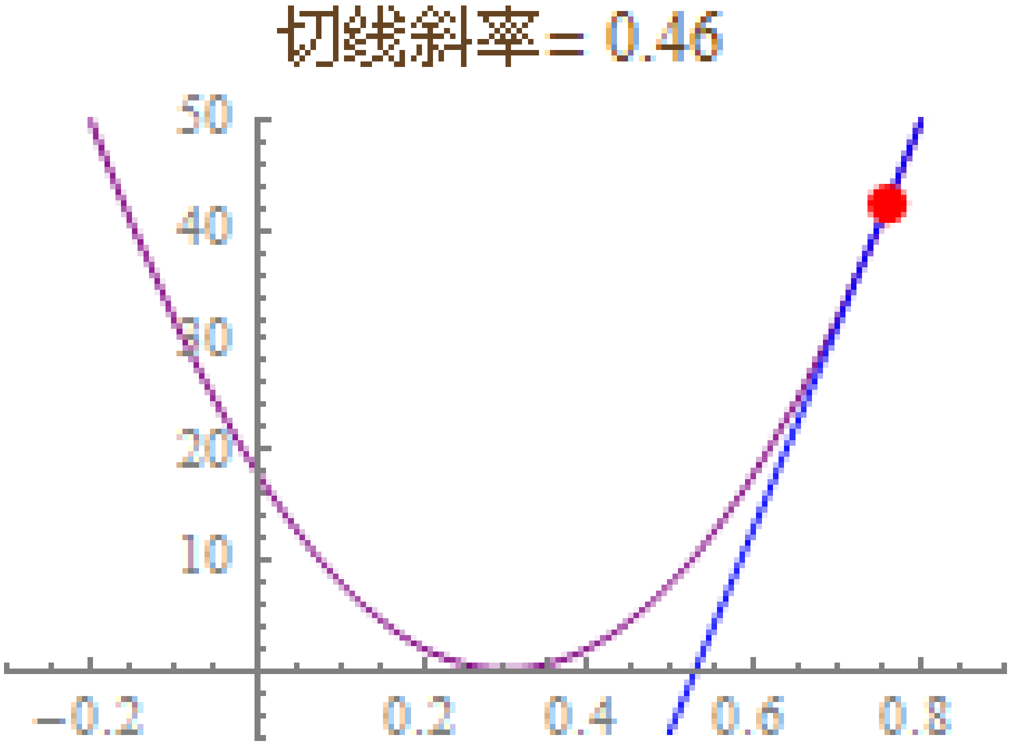




Neural network

逻辑回归

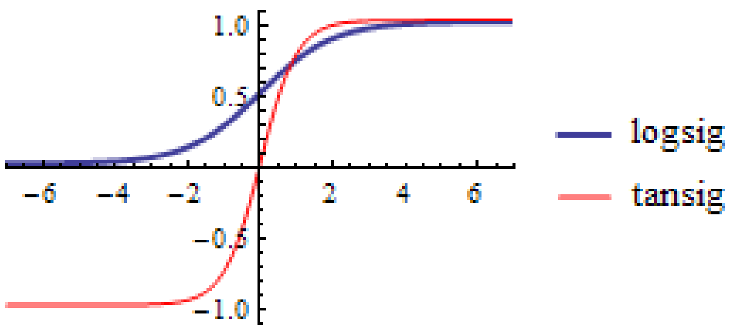




切线每次旋转的幅度叫做学习率(Learning Rate)，加大学习率会加快拟合速度，但是如果调得太大会导致切线旋转过度而无法收敛。[学习率其实是个预先设置好的参数，不会每次变化，不过可以影响每次变化的幅度。]

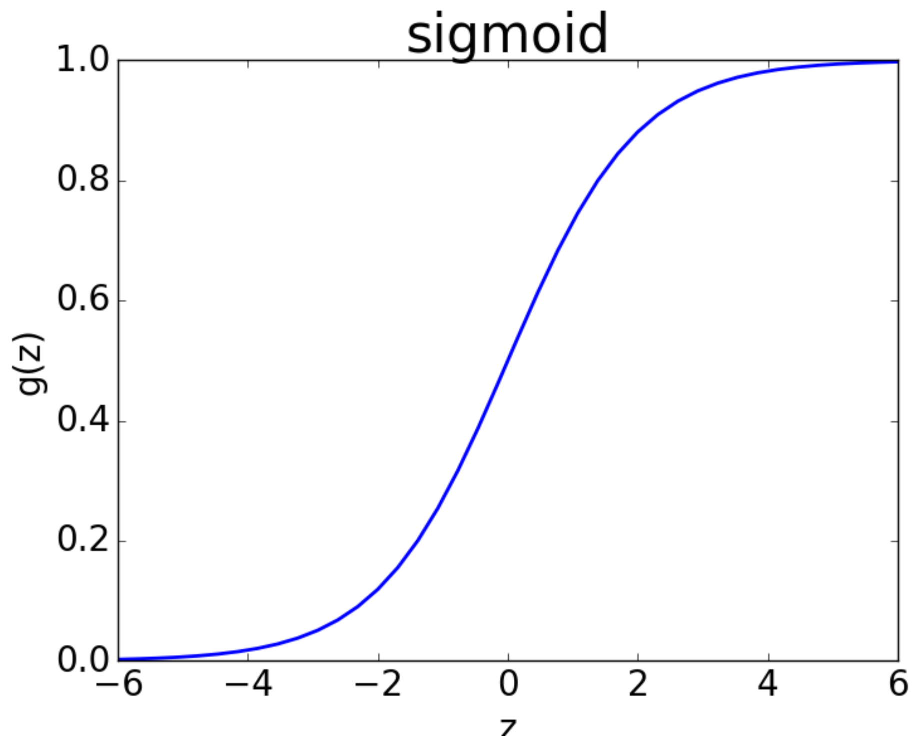
Neural network

逻辑回归(Logistic Regression，逻辑斯谛函数)



$$y = \mathrm{logsig}(x) = \frac{1}{1+e^{-x}}$$

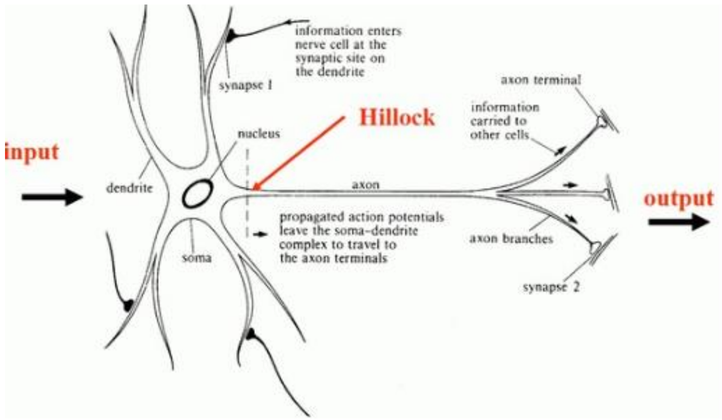
$$y = \mathrm{tansig}(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$$

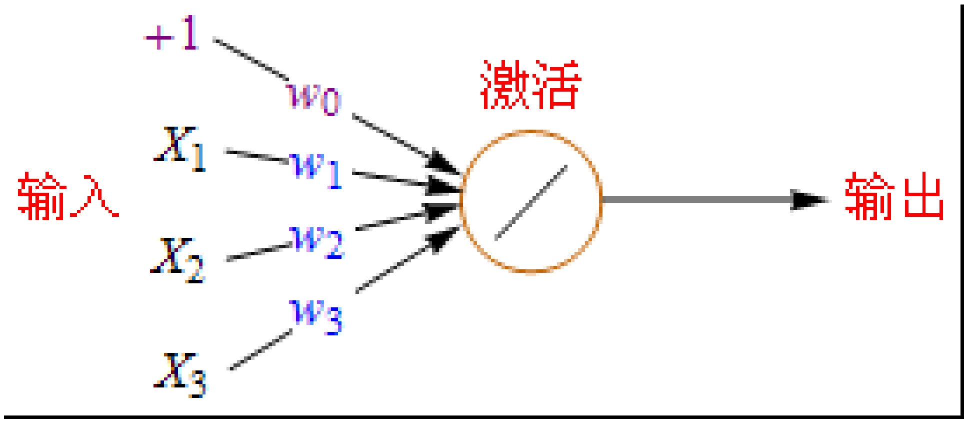


y的阈值处于（−∞，+∞），此时不能很好的给出属于某一类的概率，因为概率的范围是[0,1],我们需要一个更好的映射函数，能够将分类的结果很好的映射成为[0,1]之间的概率，并且这个函数能够具有很好的可微分性。

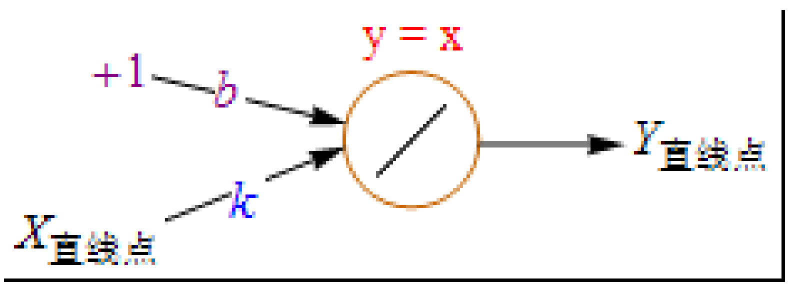
Neural network

逻辑回归

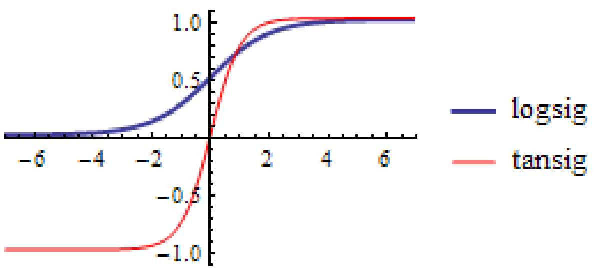




Purelin激活函数

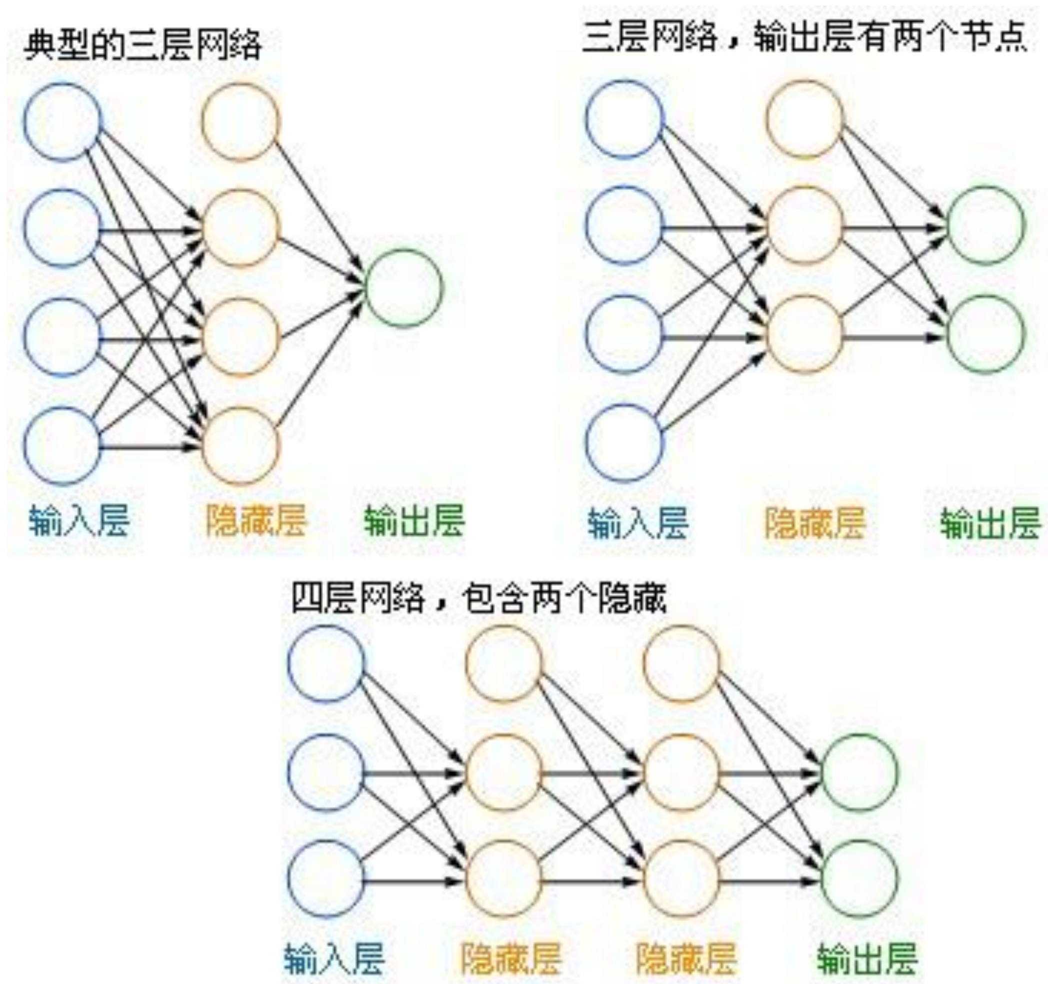


Sigmoid激活函数



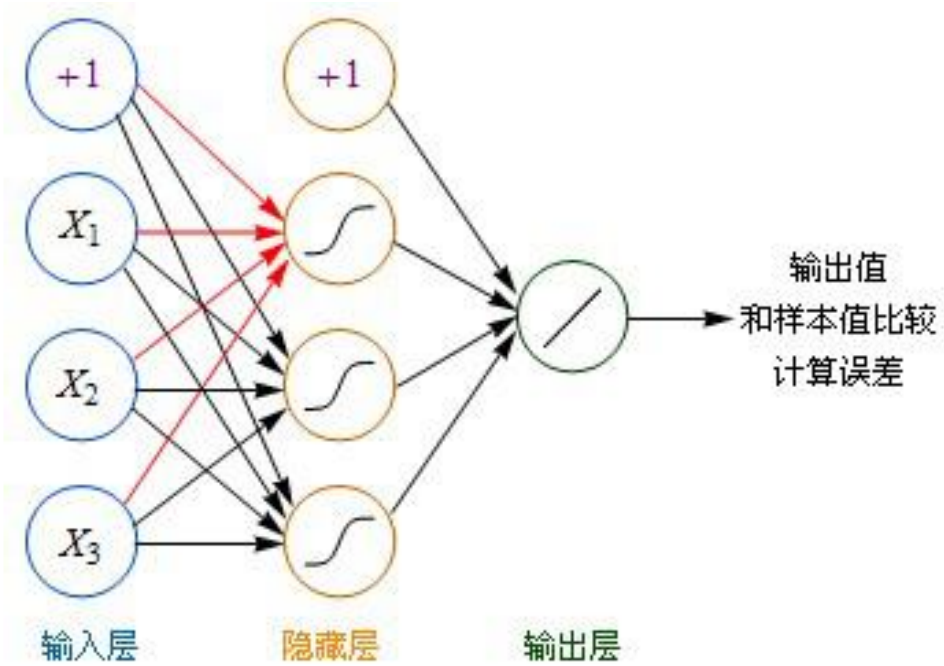
Neural network

神经网络



Neural network

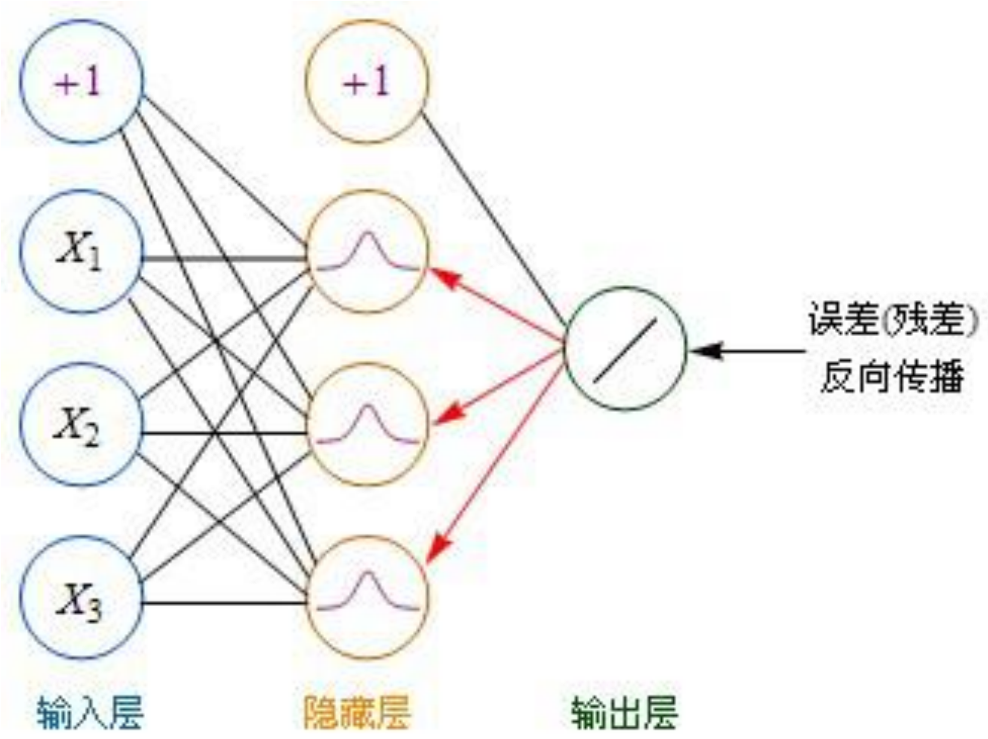
神经网络



- 隐藏层用都是用Sigmoid作激活函数，而输出层用的是Purelin。这是因为Purelin可以保持之前任意范围的数值缩放，便于和样本值作比较，而Sigmoid的数值范围只能在0~1之间。
- 起初输入层的数值通过网络计算分别传播到隐藏层，再以相同的方式传播到输出层，最终的输出值和样本值作比较，计算出误差，这个过程叫前向传播(Forward Propagation)。

Neural network

反向传播算法(BP算法)

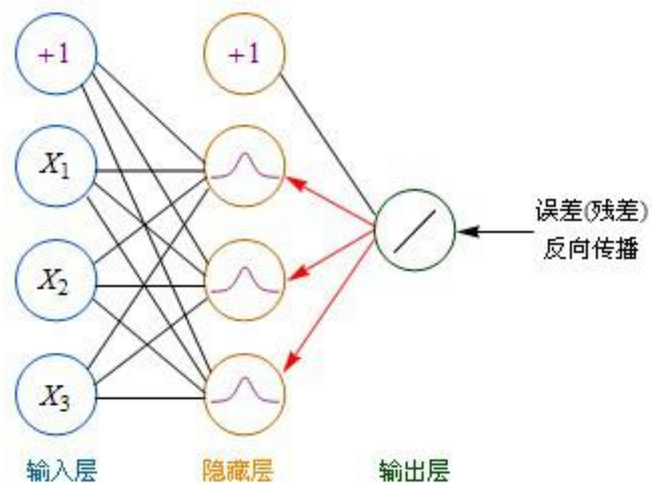


利用前向传播最后输出的结果来计算误差的偏导数，再用这个偏导数和前面的隐藏层进行加权求和，如此一层一层的向后传下去，直到输入层(不计算输入层)，最后利用每个节点求出的偏导数来更新权重

Neural network

反向传播算法(BP算法)

前馈神经网络(FeedForward Neural Network)，也叫BP神经网络(Back Propagation Neural Network)。



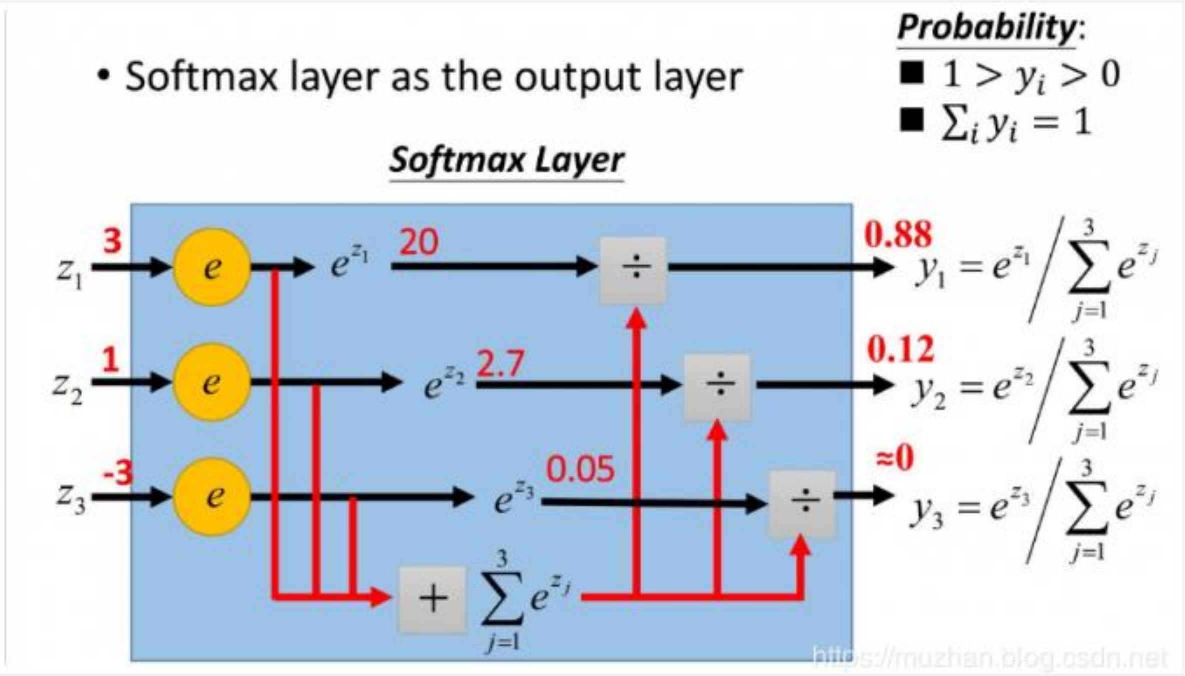
如果输出层用Purelin作激活函数，Purelin的导数是1，输出层→隐藏层：残差 = -(输出值-样本值)

如果用Sigmoid(logsig)作激活函数，那么：Sigmoid导数 = Sigmoid*(1-Sigmoid)
输出层→隐藏层：残差 = -(Sigmoid输出值-样本值) * Sigmoid*(1-Sigmoid) = -(输出值-样本值)输出值(1-输出值)
隐藏层→隐藏层：残差 = (右层每个节点的残差加权求和)* 当前节点的Sigmoid*(1-当前节点的Sigmoid)

Neural network

softmax分类器

$$softmax(x_0) = \frac{e^{x_0}}{e^{x_0} + e^{x_1} + e^{x_2}}$$



softmax对神经元的输出信号进行加工，输出为分类的概率值。

Neural network

示例

图例：

logsig激活函数

残差(误差偏导数)

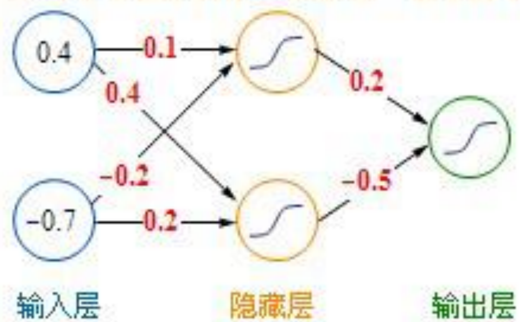
加权求和

训练集的数据，首先对第一行进行处理

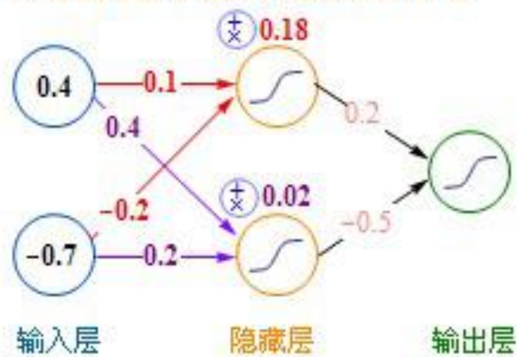
X1	X2	样本值
0.4	-0.7	0.1
0.3	-0.5	0.05
0.6	0.1	0.3
0.2	0.4	0.25
0.1	-0.2	0.12

为了适应输出层的logsig变换，X1,X2的数值范围规定在至0~1之间

1.每个节点之间的初始权重一般是随机生成的



2.对输入层节点进行加权求和计算



$0.4 \times 0.1 + (-0.7) \times (-0.2) = 0.18$

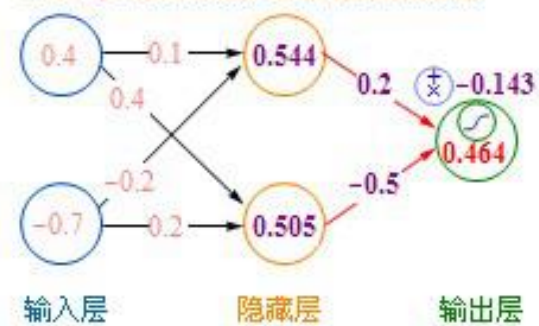
$0.4 \times 0.4 + (-0.7) \times (0.2) = 0.02$

3.执行Sigmoid激活

$\mathrm{logsig}(0.18) = \dfrac{1}{1+e^{-0.18}} = 0.544$

$\mathrm{logsig}(0.02) = \dfrac{1}{1+e^{-0.02}} = 0.505$

4.用相同的方法计算出输出层的值



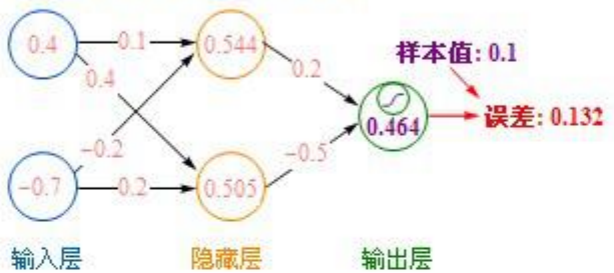
$0.544 \times 0.2 + 0.505 \times (-0.5) = -0.143$

$\mathrm{logsig}(-0.143) = \dfrac{1}{1+e^{0.143}} = 0.464$

Neural network

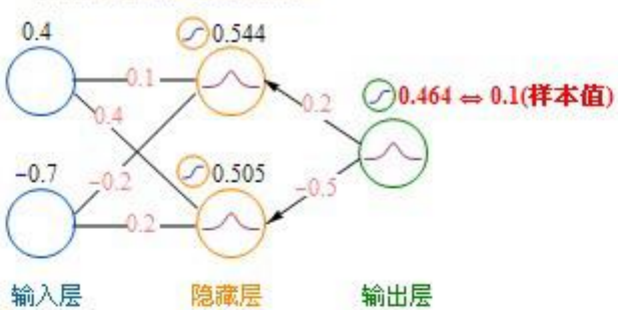
示例

5.计算误差，误差接近0时收敛

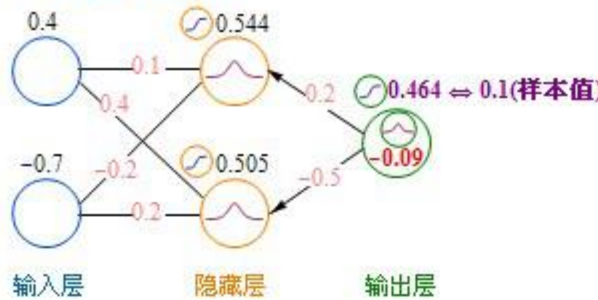


误差=(0.464 - 0.1)²=0.132 (误差<0.0001,可以收敛)

6.开始准备误差的反向传播



7.计算输出层的残差



残差=-(0.464 - 0.1)×0.464×(1 - 0.464)=-0.09

8.输出层节点的残差加权求和

-0.09×0.2=-0.018 (输出层只有一个节点,不用求和)

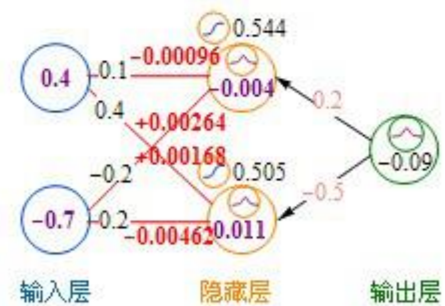
-0.09×0.5=-0.045

9.继续求隐藏层的残差

残差1=(-0.018)×(0.544)×(1-0.544)=-0.004

残差2=0.045×0.505×(1-0.505)=0.011

10.开始准备更新第一层权重，设学习率为0.6



0.4×(-0.004)×0.6=-0.0016×0.6=-0.00096

0.4× 0.011 ×0.6= 0.0044×0.6= 0.00264

-0.7×(-0.004)×0.6= 0.0028×0.6= 0.00168

-0.7× 0.011 ×0.6=-0.0077×0.6=-0.00462

Neural network

示例

- **Step 15 :**

  With the updated weights **[V]** and **[W]**, error is calculated again and next training set is taken and the error will then get adjusted.

- **Step 16 :**

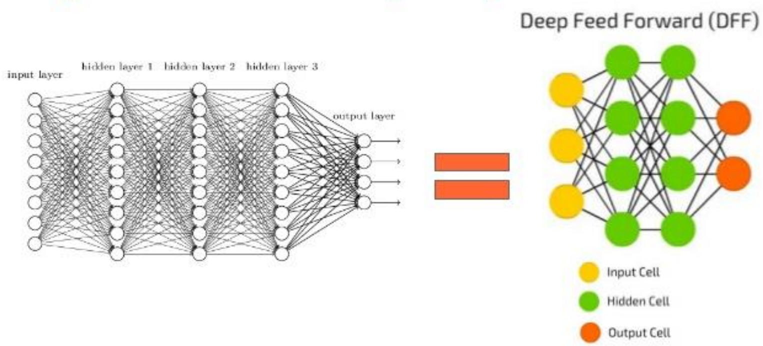
  Iterations are carried out till we get the error less than the tolerance.

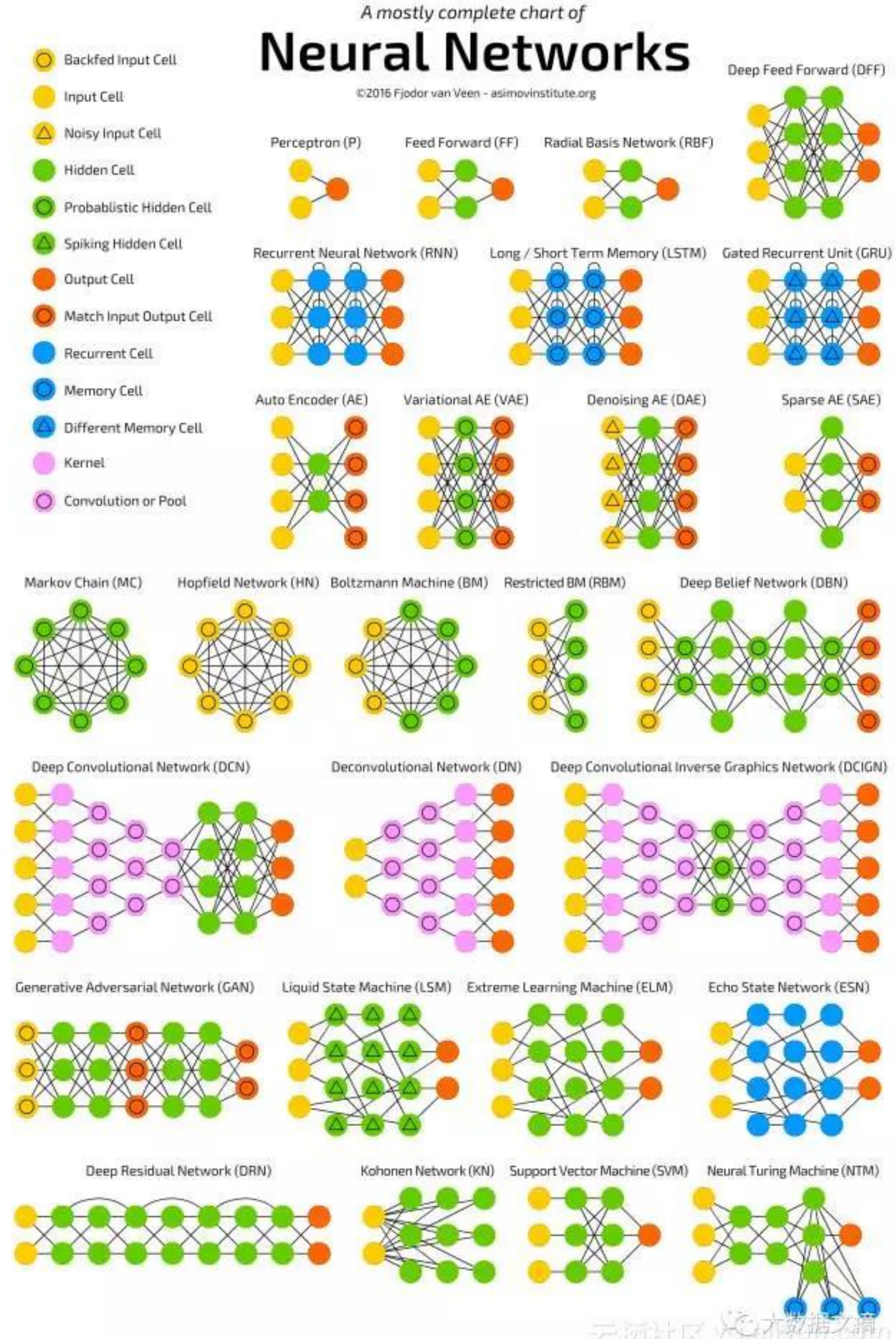
- **Step 17 :**

  Once the weights are adjusted the network is ready for inferencing new objects .

Neural network

Deep Neural Network (DNN)

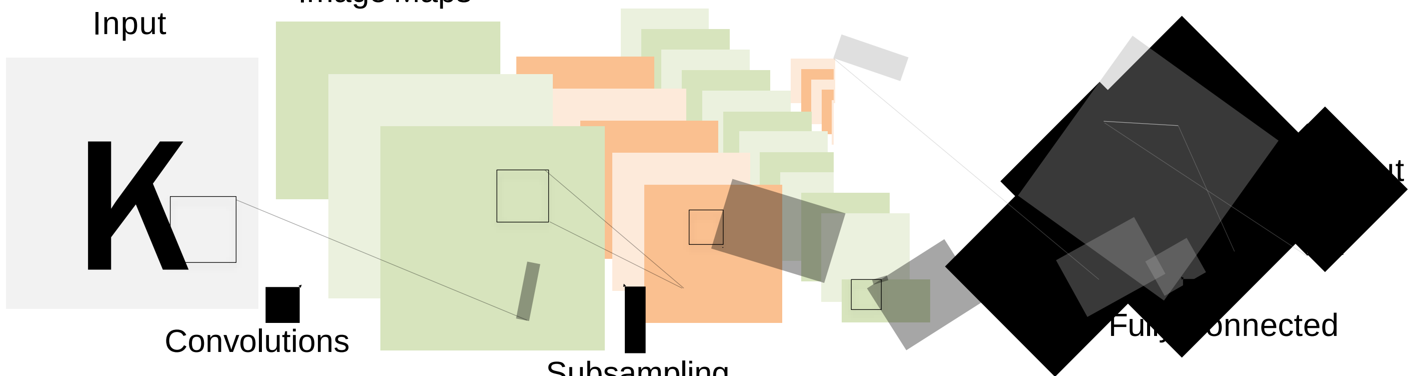




Adapted from: http://www.asimovinstitute.org/neural-network-zoo/

1998

LeCun et al.



# of transistors

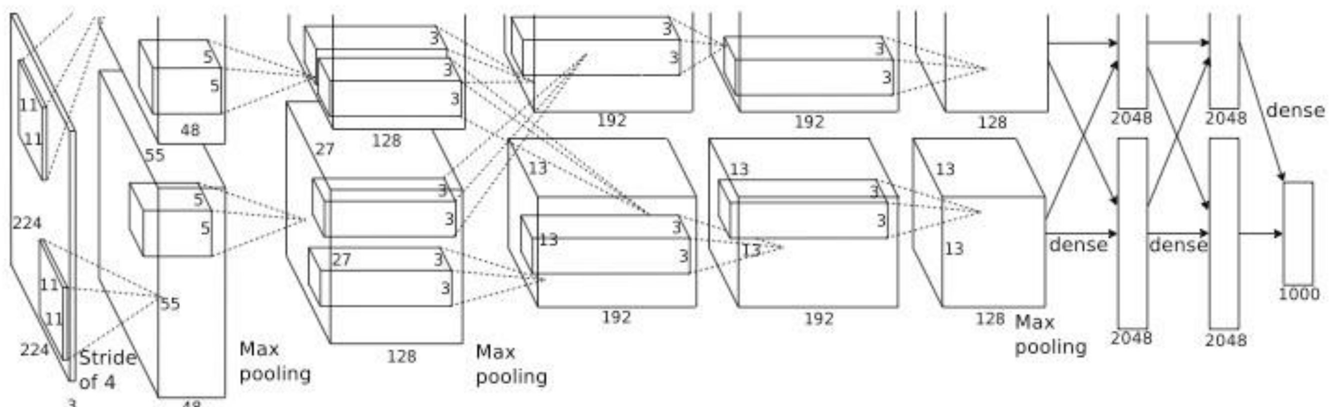
10^6

# of pixels used in training

10^7 NIST

2012

Krizhevsky et al.



# of transistors

10^9

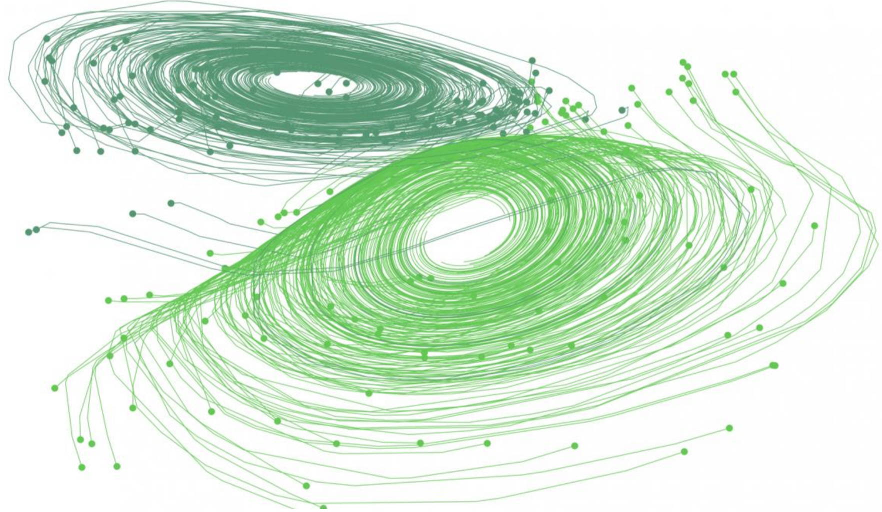
GPUs

# of pixels used in training

10^{14} IMAGENET

Convolutional Neural Networks (CNN) were NOT invented overnight!

Neural network



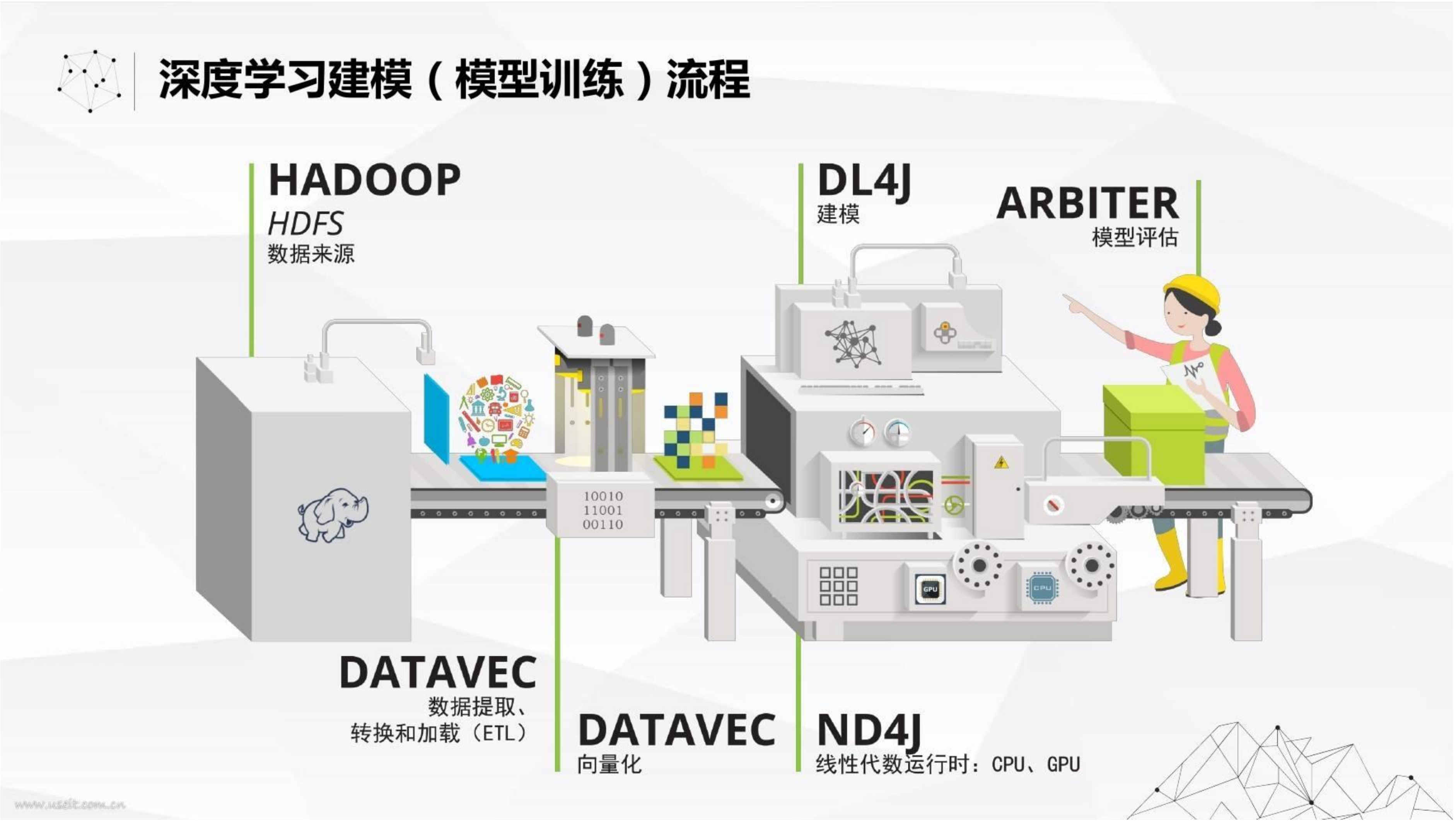
Intelligent Machines

A radical new neural network design could overcome big challenges in AI

Researchers borrowed equations from calculus to redesign the core machinery of deep learning so it can model continuous processes like changes in health.

by Karen Hao   December 12, 2018

深度学习的基本过程



深度学习建模（模型训练）流程
HADOOP
HDFS
数据来源
DL4J
建模
ARBITER
模型评估
10010
11001
00110
DATAVEC
数据提取、
转换和加载（ETL）
DATAVEC
向量化
ND4J
线性代数运行时：CPU、GPU

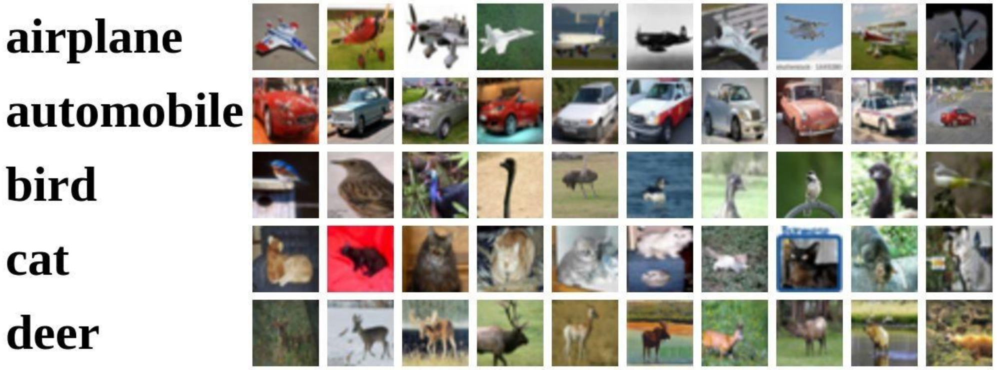
Data-Driven Approach

1. Collect a dataset of images and labels
2. Use Machine Learning to train a classifier
3. Evaluate the classifier on new images

```
def train(images, labels):
  # Machine learning!
  return model
```

```
def predict(model, test_images):
  # Use model to predict labels
  return test_labels
```

Example training set



What does this look like?

Parametric Approach: Linear Classifier

3072x1

$$\boxed{f(x,W) = \boxed{W}\boxed{x}}$$

10x1 **10x3072**

Image

Array of **32x32x3** numbers
(3072 numbers total)

f(**x**,**W**)

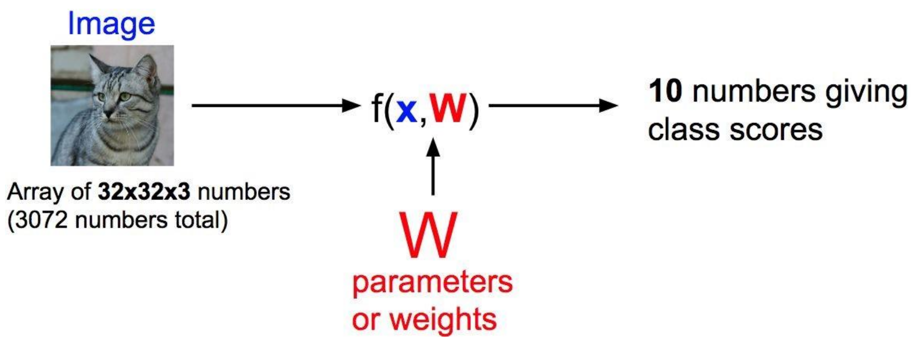
10 numbers giving
class scores

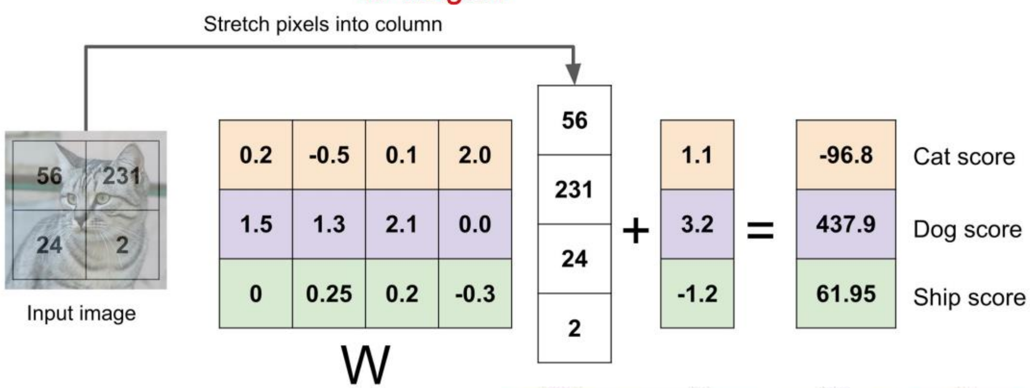
W

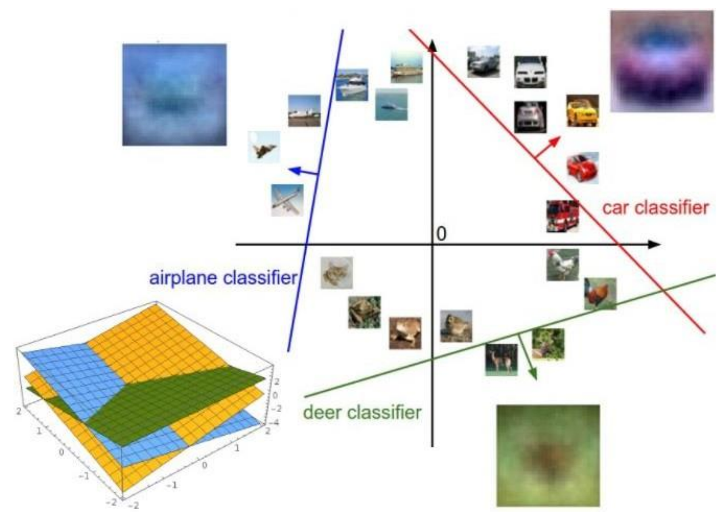
parameters
or weights

Parametric Approach: Linear Classifier

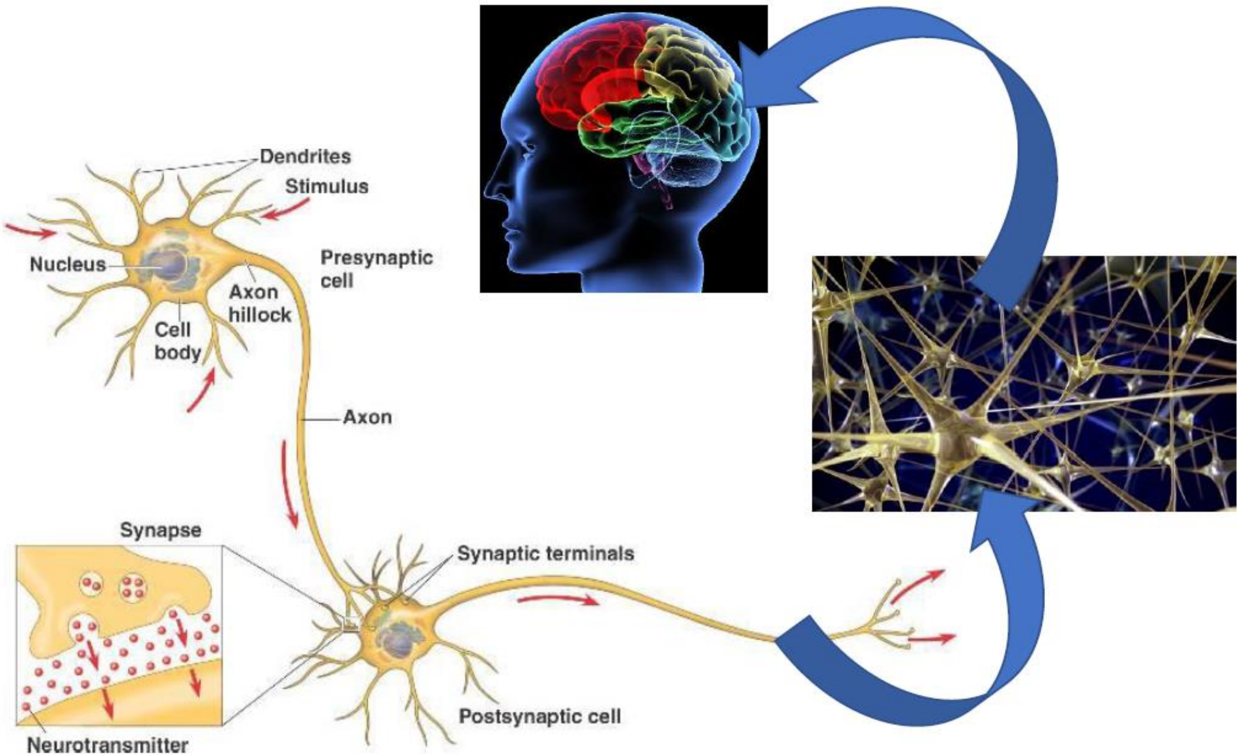
$$f(x,W) = Wx + b$$







Neural Network introduction

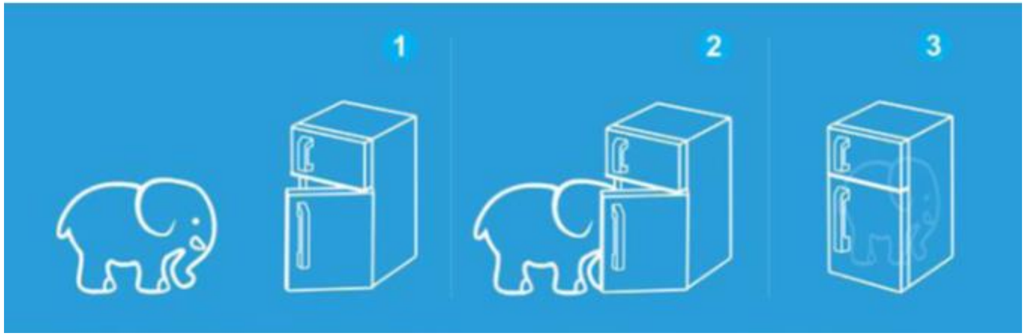


Neural network model is inspired by human neuron
Each neuron is a judgement unit

Hubel & Wiesel,1959

Neural Network workflow

Neural network is so simple……



Feature Selection → Model Construction → Pick the best model

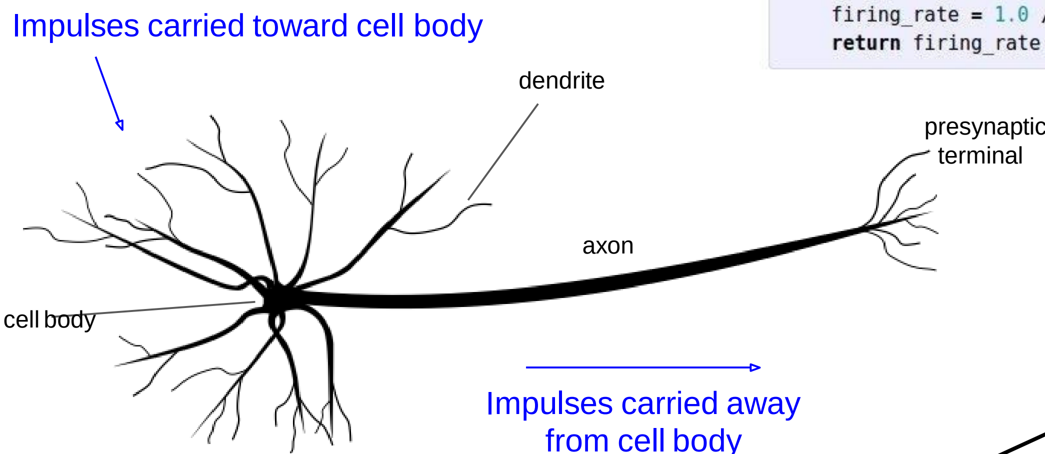
The key point
Deep learning should be a problem-oriented tool.

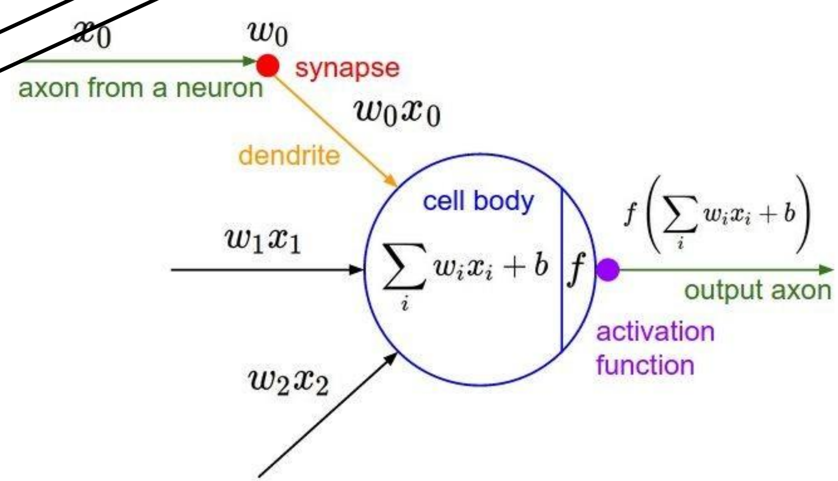
Number of layers
Functions Selection

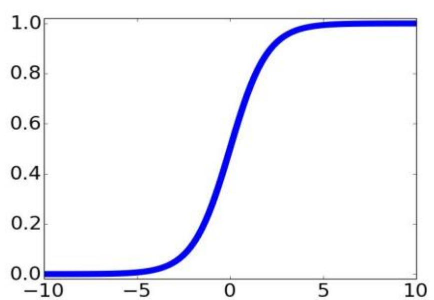
Parameters Adjustment

Neuron Network

```python
class Neuron:
  # ...
  def neuron_tick(inputs):
    """ assume inputs and weights are 1-D numpy arrays and bias is a number """
    cell_body_sum = np.sum(inputs * self.weights) + self.bias
    firing_rate = 1.0 / (1.0 + math.exp(-cell_body_sum)) # sigmoid activation function
    return firing_rate
```







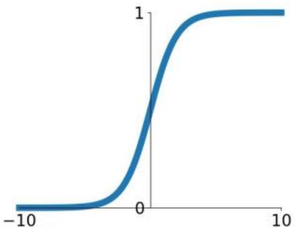
sigmoid activation function

$$\frac{1}{1+e^{-x}}$$

Activation Functions

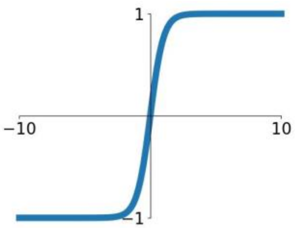
Sigmoid

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



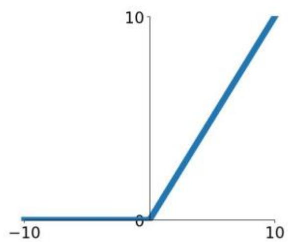
tanh

$\tanh(x)$



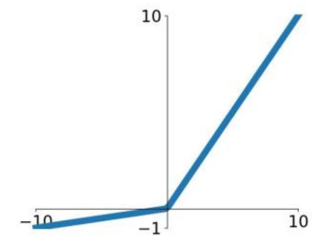
ReLU

$\max(0, x)$



Leaky ReLU

$\max(0.1x, x)$

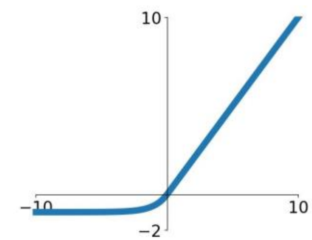


Maxout

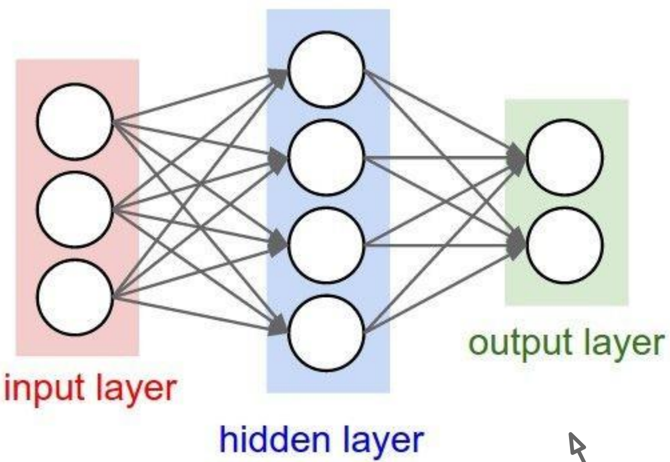
$\max(w_1^T x + b_1, w_2^T x + b_2)$

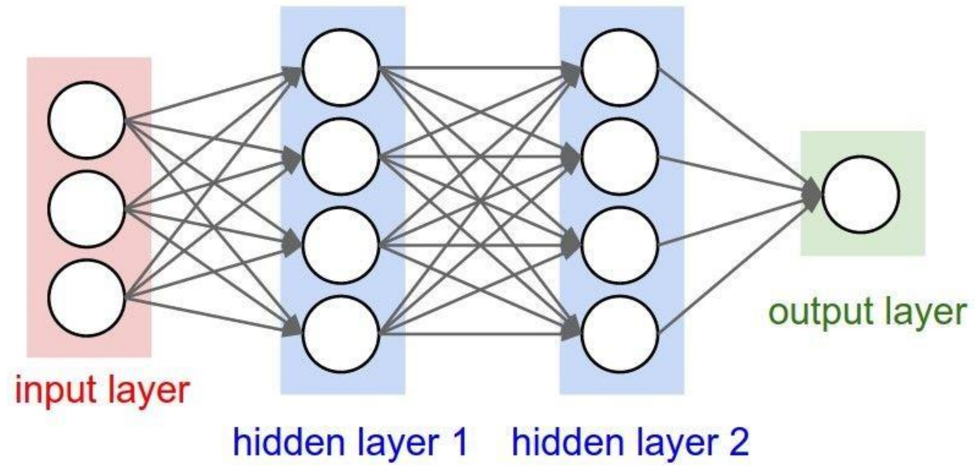
ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



Neural networks: Architectures





"2-layer Neural Net", or
"1-hidden-layer Neural Net"

"3-layer Neural Net", or
"2-hidden-layer Neural Net"

"Fully-connected" layers

Convolutional Neural Networks

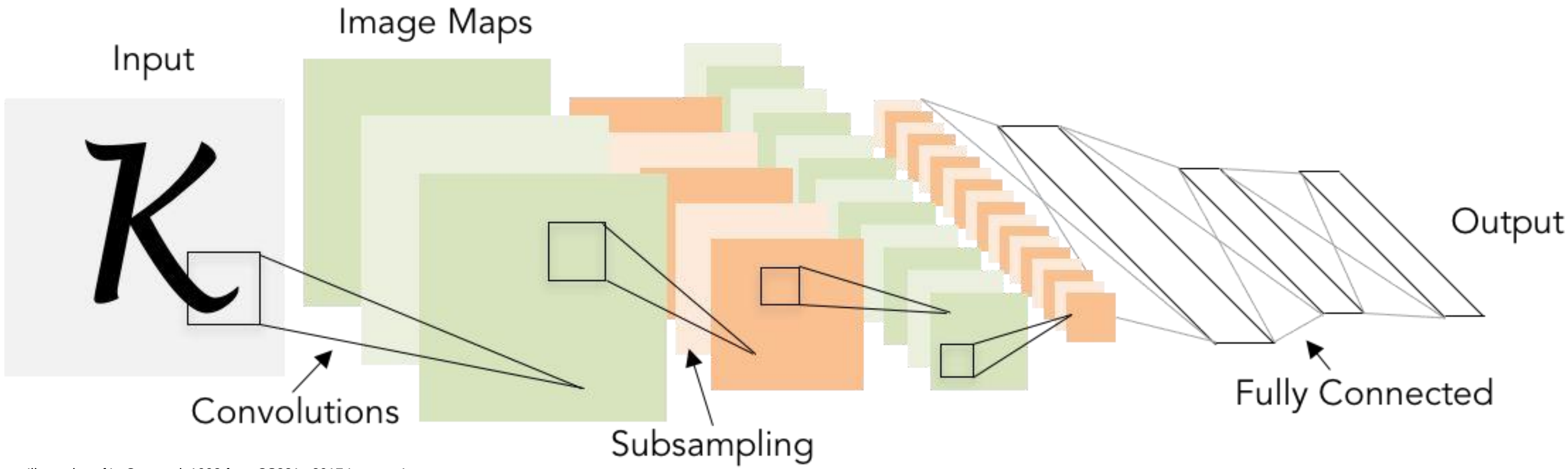
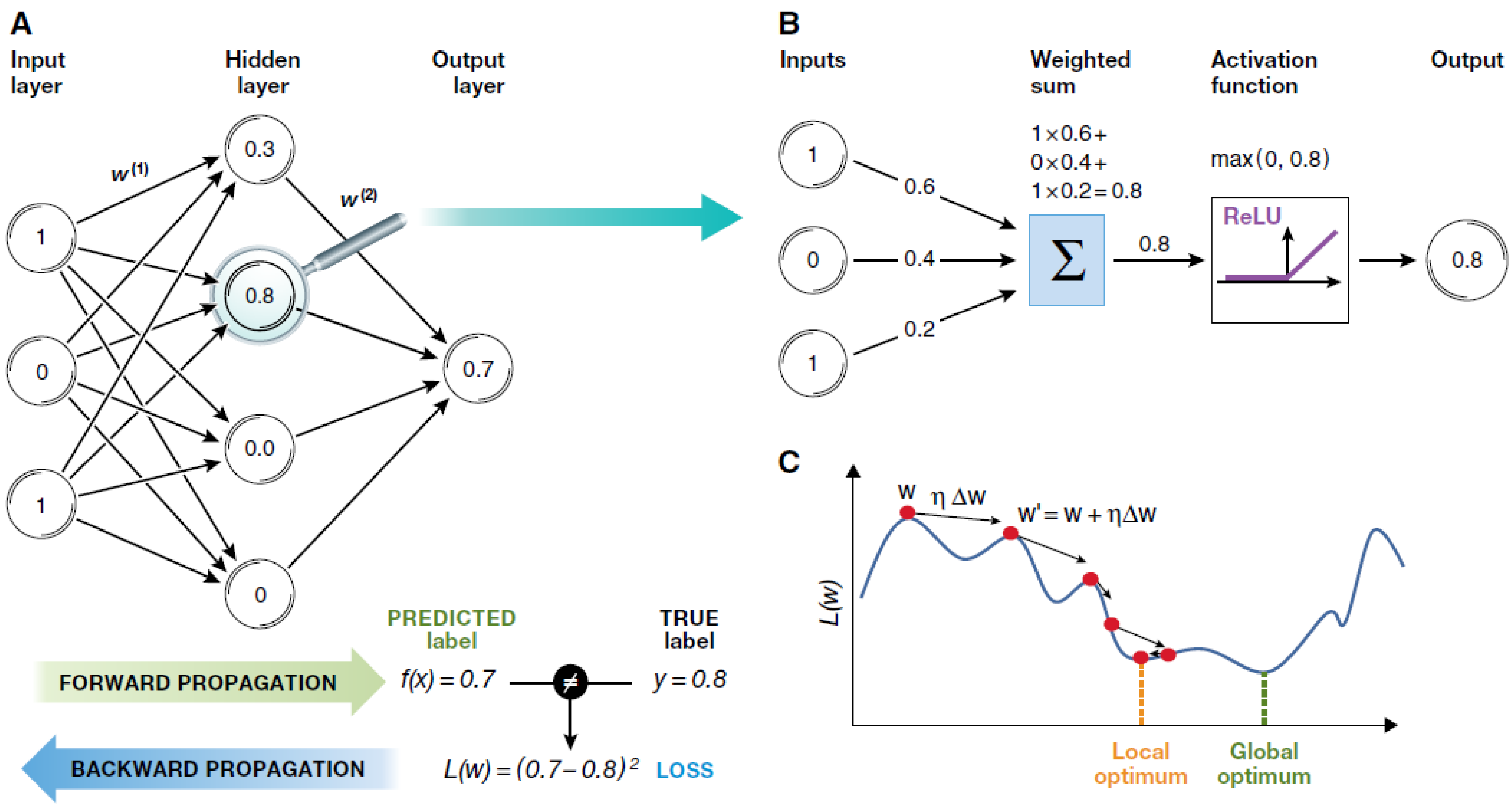


Illustration of LeCun et al. 1998 from CS231n 2017 Lecture 1

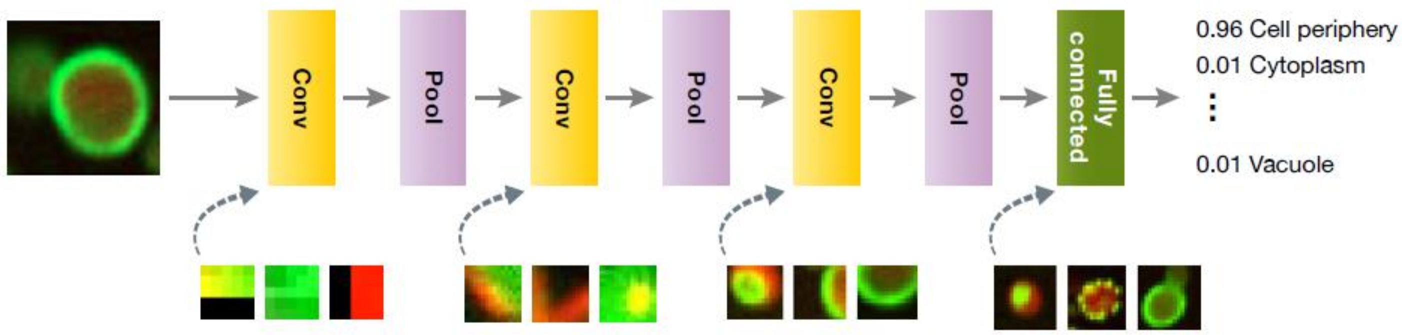
Artificial Neural Network



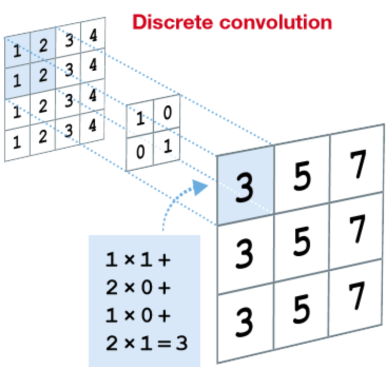
Adapted from “Deep learning for computational biology. Molecular Systems Biology, 2016.”

Convolution Neural Network

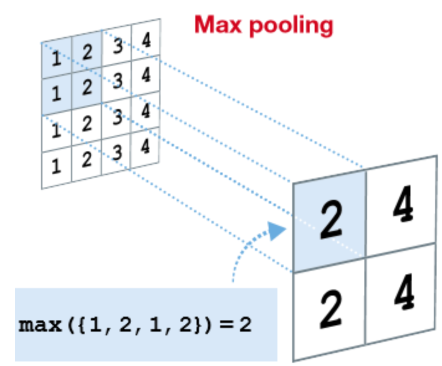
Convolution neural network is similar to neural network, it is a good approach to process figure



Convolution and pooling process : edge features extraction

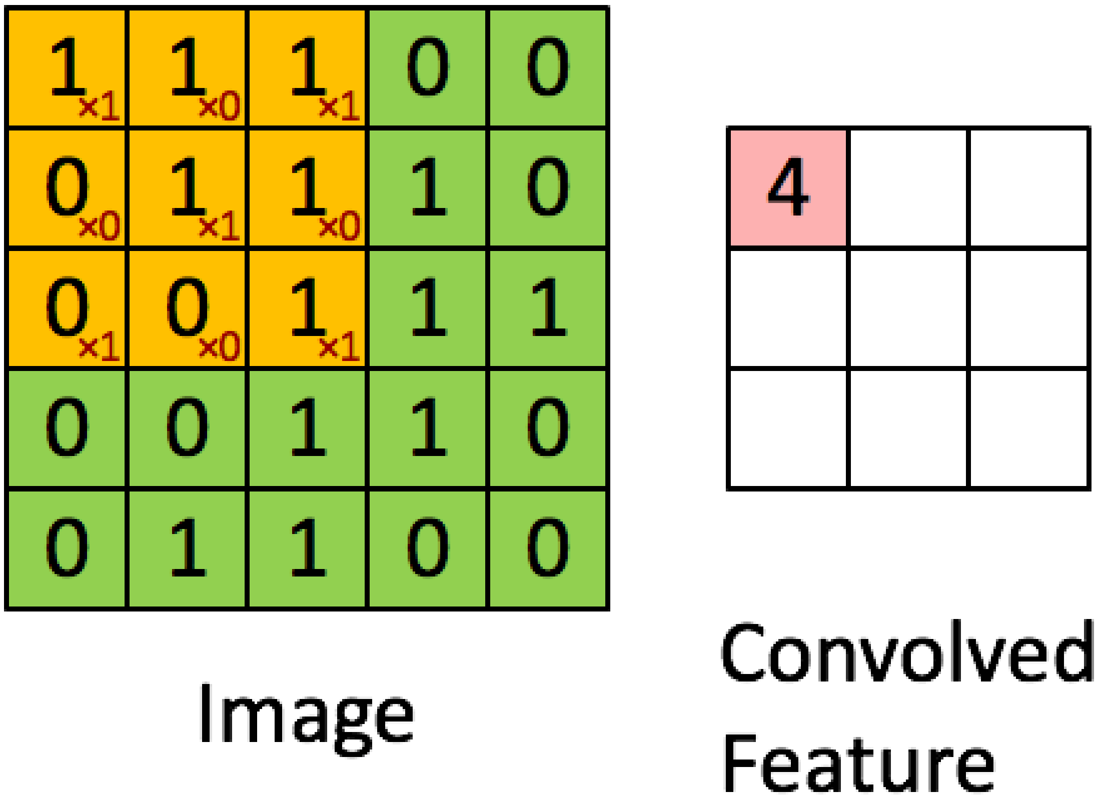


Convolution

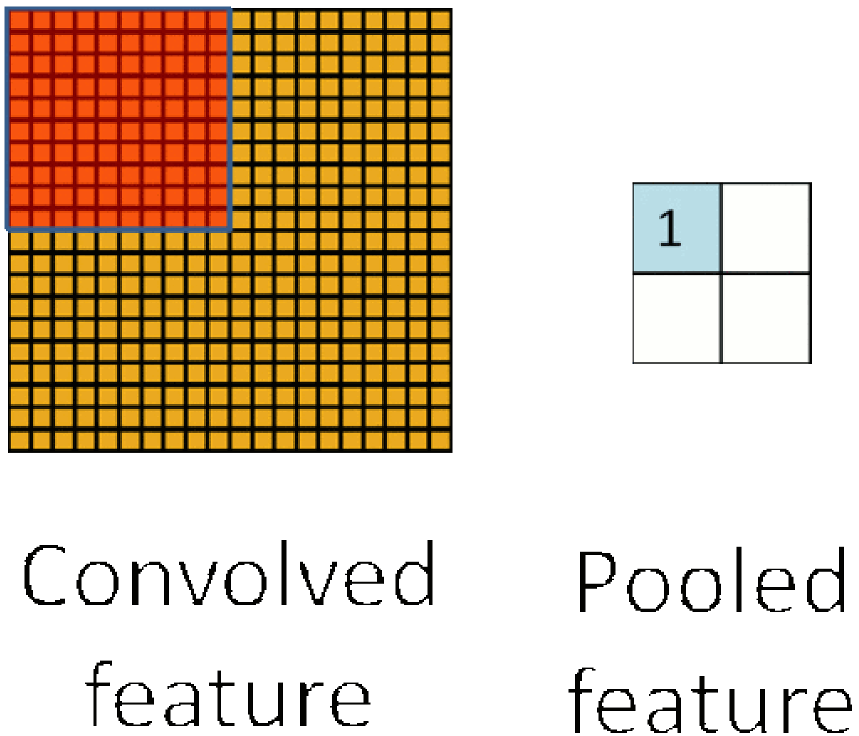


Pooling

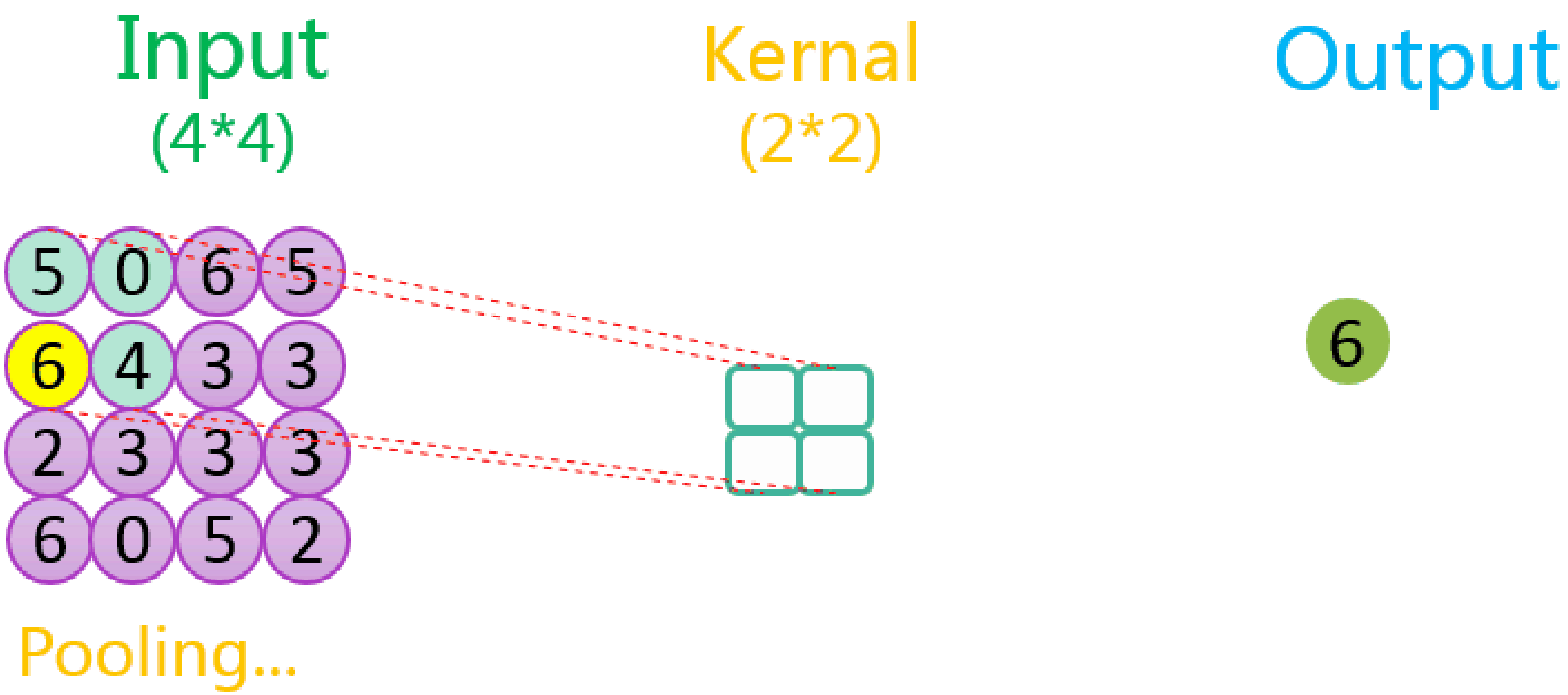
Convolution Neural Network



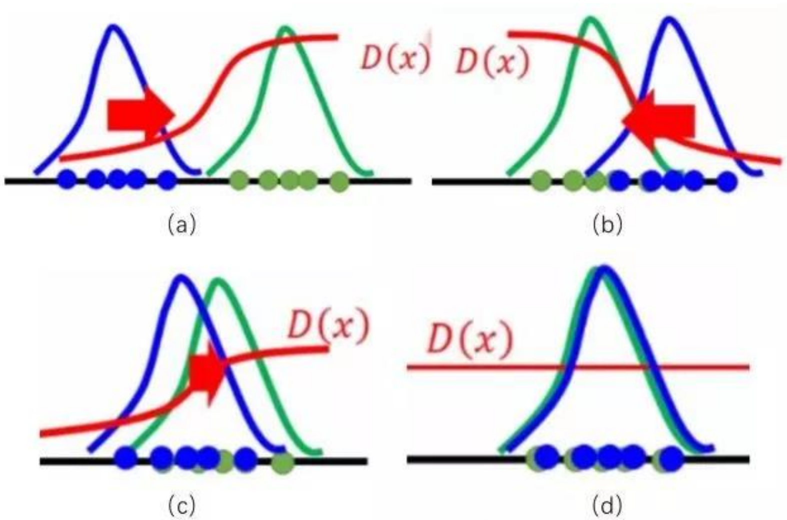
Convolution Neural Network

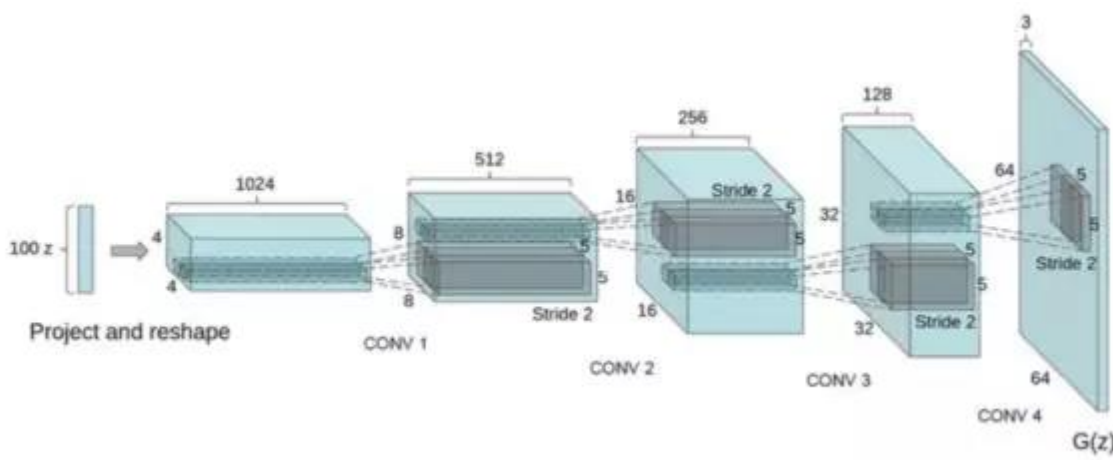


Convolution Neural Network

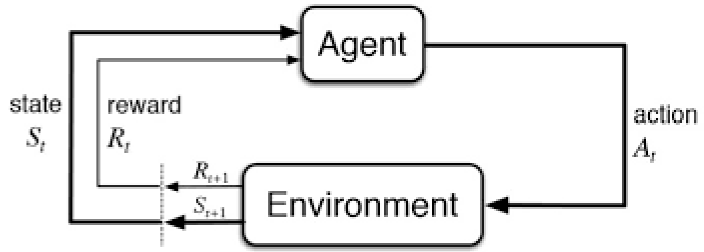


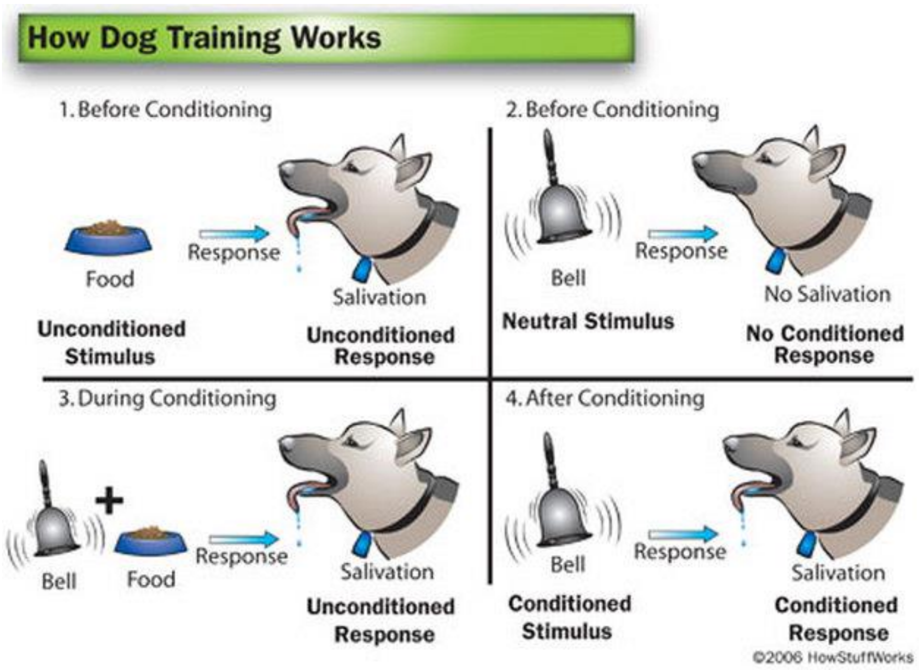
Generative adversarial network, GAN

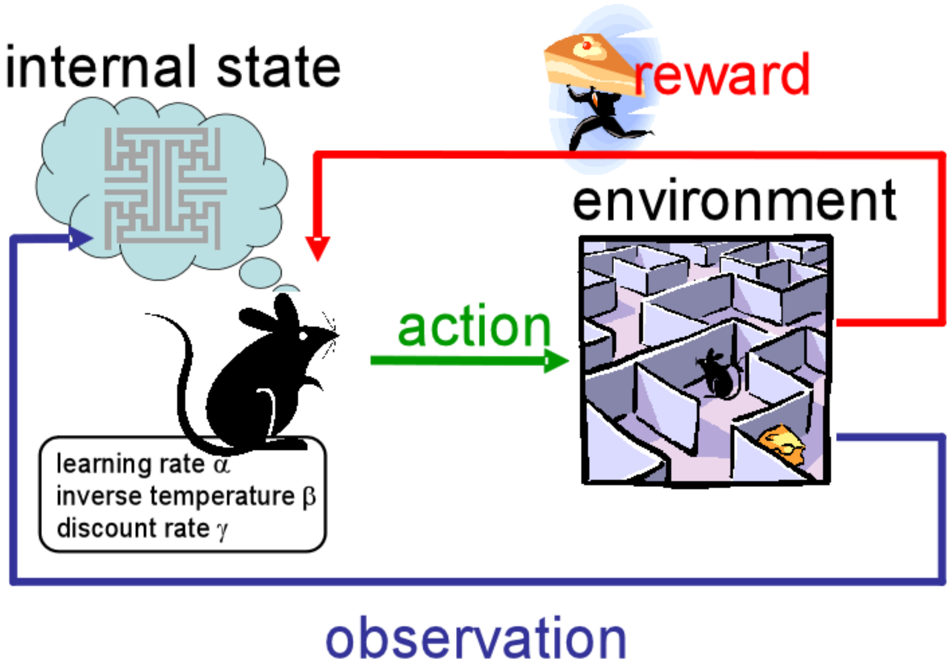




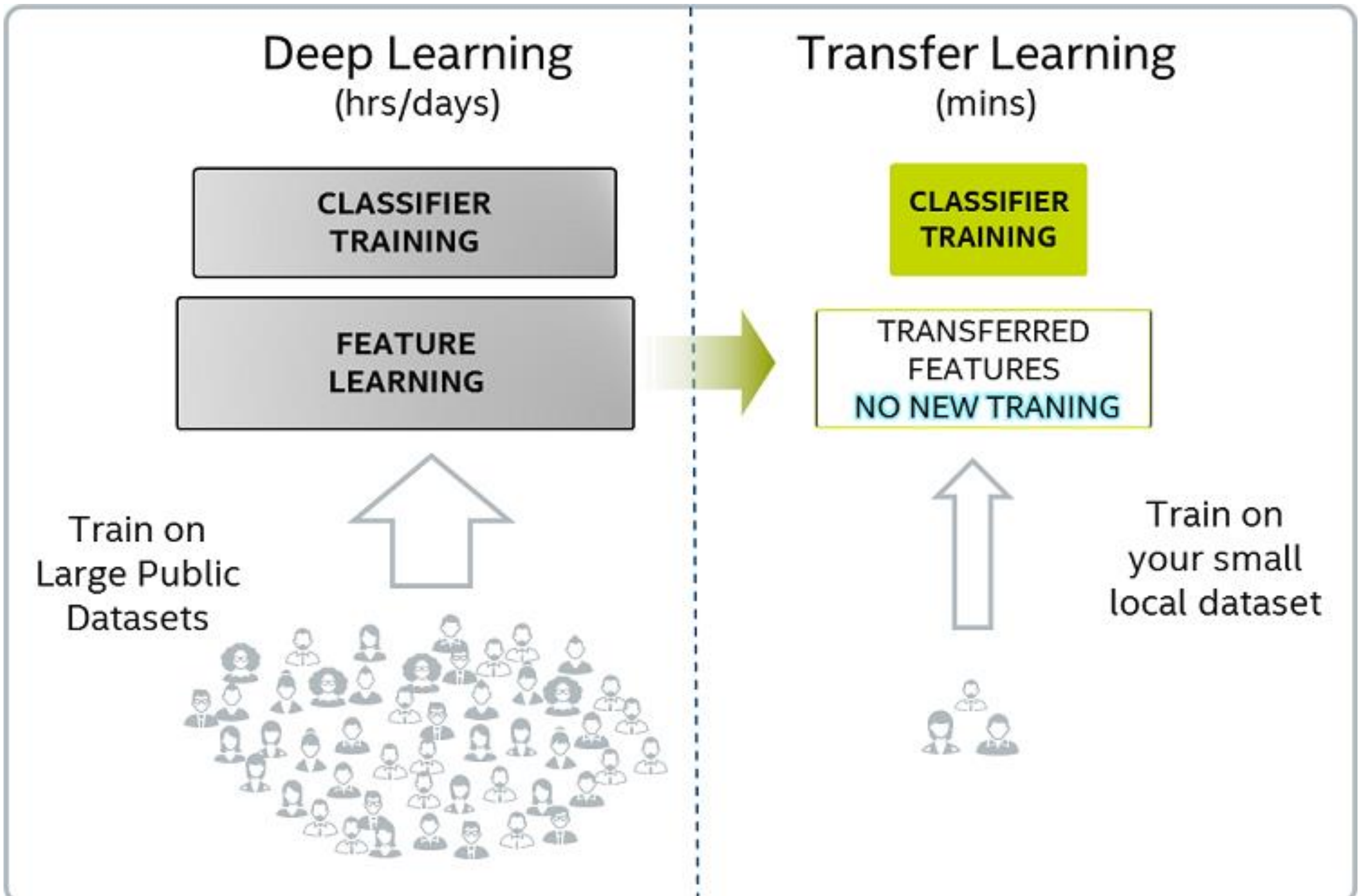
Reinforcement learning







Transfer learning



深度学习的特点

深度学习常用算法介绍

深度学习常用框架介绍

深度学习

- 2006年，加拿大多伦多大学教授、机器学习领域的泰斗Geoffrey Hinton在《科学》上发表论文提出深度学习主要观点：

1）多隐层的人工神经网络具有优异的特征学习能力，学习得到的特征对数据有更本质的刻画，从而有利于可视化或分类；

2）深度神经网络在训练上的难度，可以通过“逐层初始化”（layer-wise pre-training）来有效克服，逐层初始化可通过无监督学习实现的。

深度学习

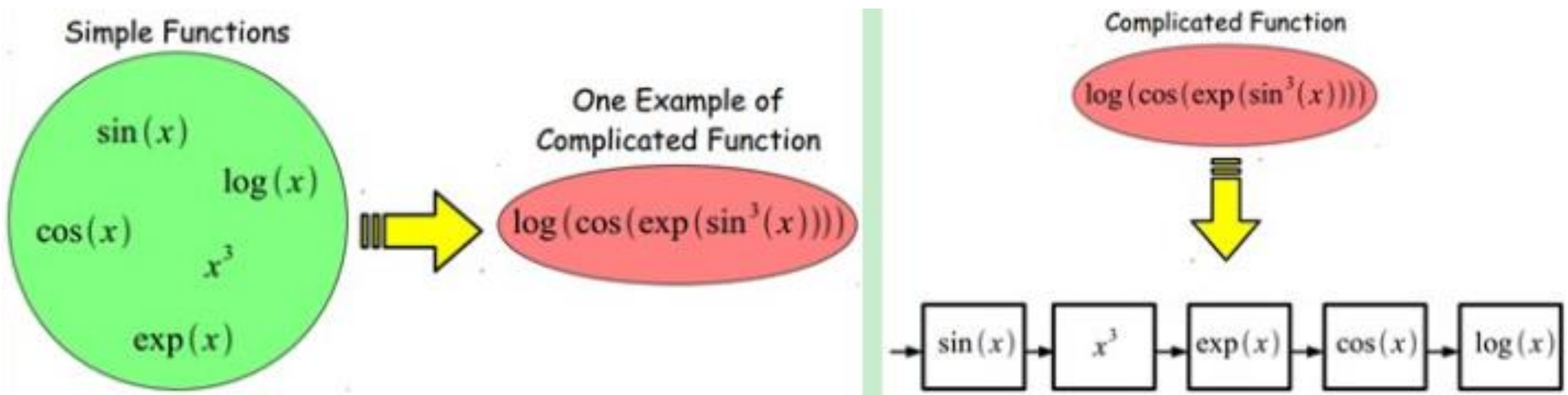
- 本质：通过构建多隐层的模型和海量训练数据（可为无标签数据），来学习更有用的特征，从而最终提升分类或预测的准确性。“深度模型”是手段，“特征学习”是目的。
- 与浅层学习区别：

1）强调了模型结构的深度，通常有5-10多层的隐层节点；

2）明确突出了特征学习的重要性，通过逐层特征变换，将样本在原空间的特征表示变换到一个新特征空间，从而使分类或预测更加容易。与人工规则构造特征的方法相比，利用大数据来学习特征，更能够刻画数据的丰富内在信息。

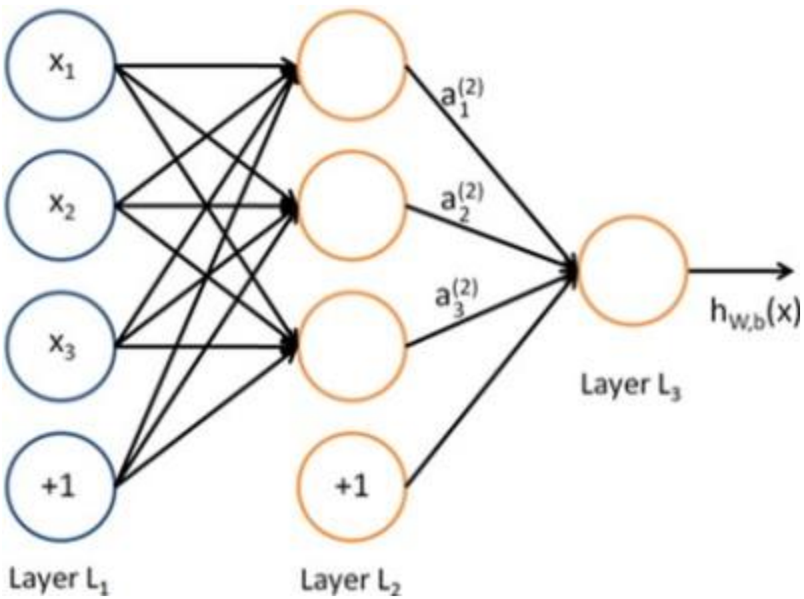
深度学习

- 好处：可通过学习一种深层非线性网络结构，实现复杂函数逼近，表征输入数据分布式表示。

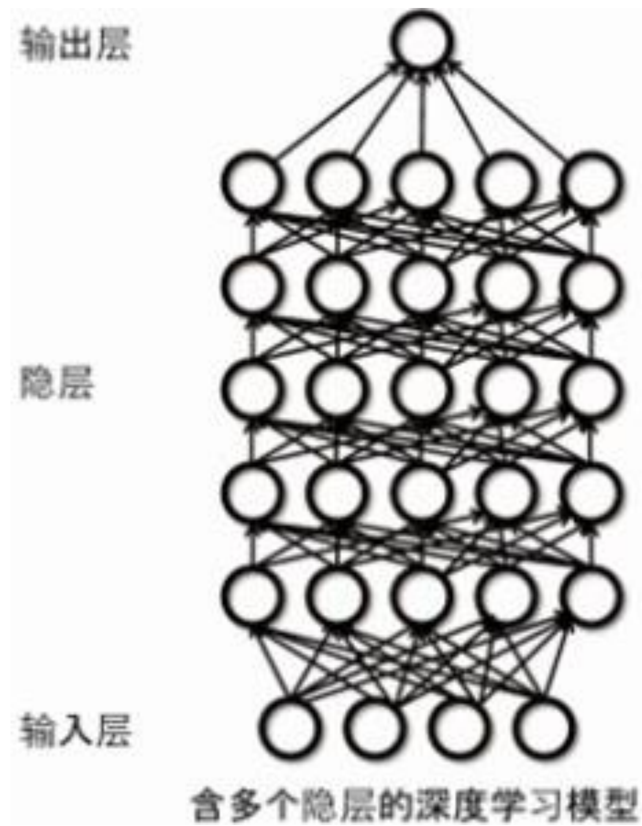


深度学习 vs. 神经网络

<u>神经网络</u>：



<u>深度学习</u>：



深度学习 vs. 神经网络

相同点：二者均采用分层结构，系统包括输入层、隐层（多层）、输出层组成的多层网络，只有相邻层节点之间有连接，同一层以及跨层节点之间相互无连接，每一层可以看作是一个logistic 回归模型。

不同点：

神经网络：采用BP算法调整参数，即采用迭代算法来训练整个网络。随机设定初值，计算当前网络的输出，然后根据当前输出和样本真实标签之间的差去改变前面各层的参数，直到收敛；

深度学习：采用逐层训练机制。采用该机制的原因在于如果采用BP机制，对于一个deep network（7层以上），残差传播到最前面的层将变得很小，出现所谓的gradient diffusion（梯度扩散）。

深度学习 vs. 神经网络

- 神经网络的局限性：

1）比较容易过拟合，参数比较难调整，而且需要不少技巧；

2）训练速度比较慢，在层次比较少（小于等于3）的情况下效果并不比其它方法更优；

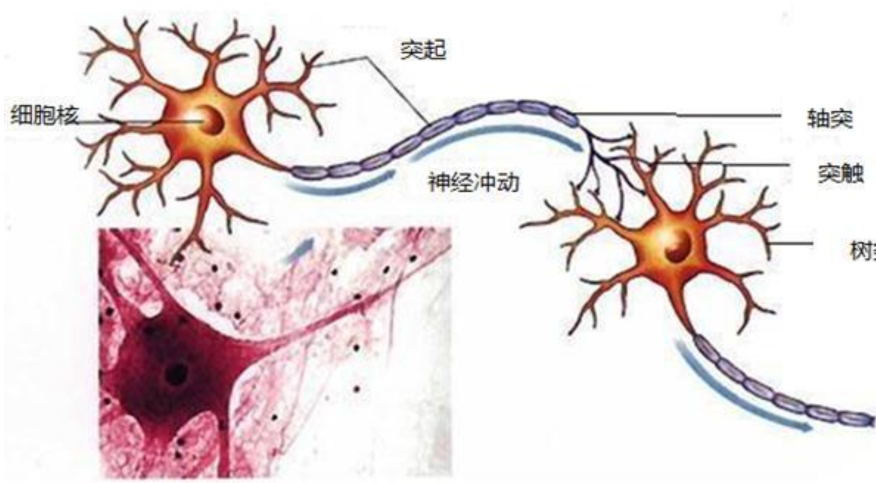
深度学习的特点

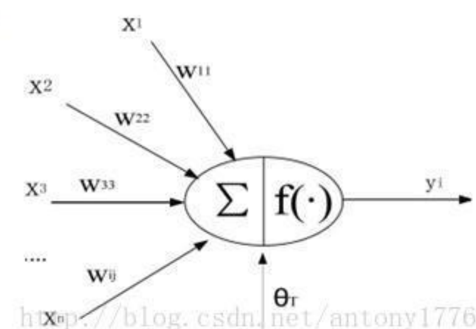
深度学习常用算法介绍

深度学习常用框架介绍

人工神经网络（ANN）

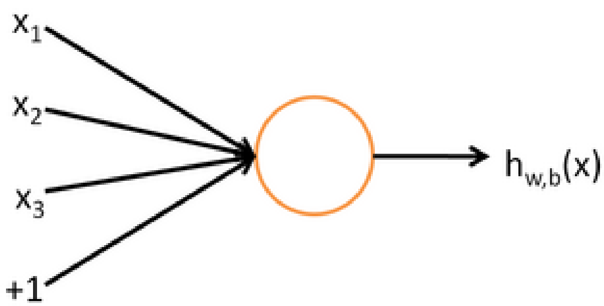
- 人工神经网络（Artificial Neural Networks）是一种模仿生物神经网络行为特征，进行分布式并行信息处理的算法数学模型。这种网络依靠系统的复杂程度，通过调整内部大量节点（神经元）之间相互连接的权重，从而达到处理信息的目的。



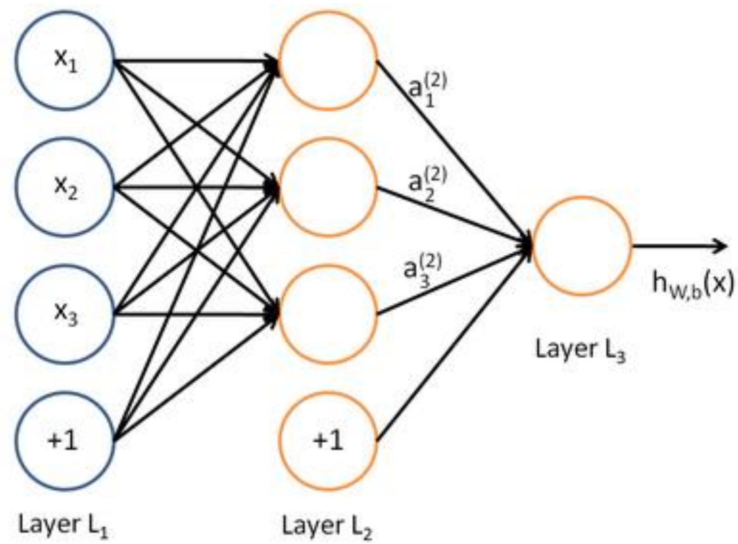


人工神经网络（ANN）

- 神经网络



$$h_{W,b}(x) = f(W^T x) = f(\textstyle\sum_{i=1}^{3} W_i x_i + b)$$



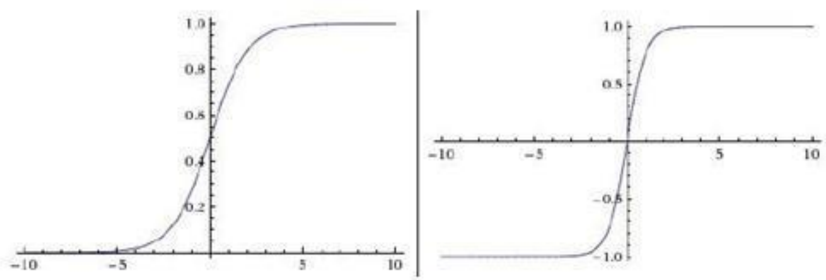
$$a_1^{(2)} = f(W_{11}^{(1)}x_1 + W_{12}^{(1)}x_2 + W_{13}^{(1)}x_3 + b_1^{(1)})$$
$$a_2^{(2)} = f(W_{21}^{(1)}x_1 + W_{22}^{(1)}x_2 + W_{23}^{(1)}x_3 + b_2^{(1)})$$
$$a_3^{(2)} = f(W_{31}^{(1)}x_1 + W_{32}^{(1)}x_2 + W_{33}^{(1)}x_3 + b_3^{(1)})$$
$$h_{W,b}(x) = a_1^{(3)} = f(W_{11}^{(2)}a_1^{(2)} + W_{12}^{(2)}a_2^{(2)} + W_{13}^{(2)}a_3^{(2)} + b_1^{(2)})$$

人工神经网络（ANN）

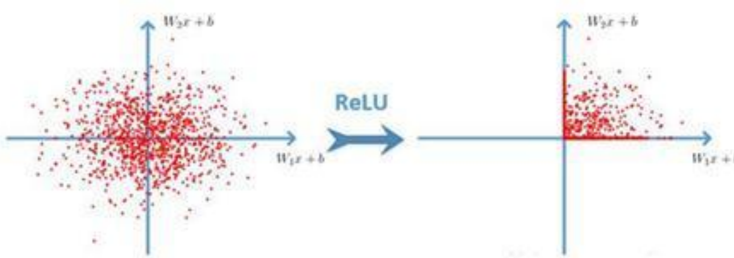
- 人工神经网络的重要概念：

  1 权值矩阵：相当于神经网络的记忆！在训练的过程中，动态调整和适应。

  2 激励函数：

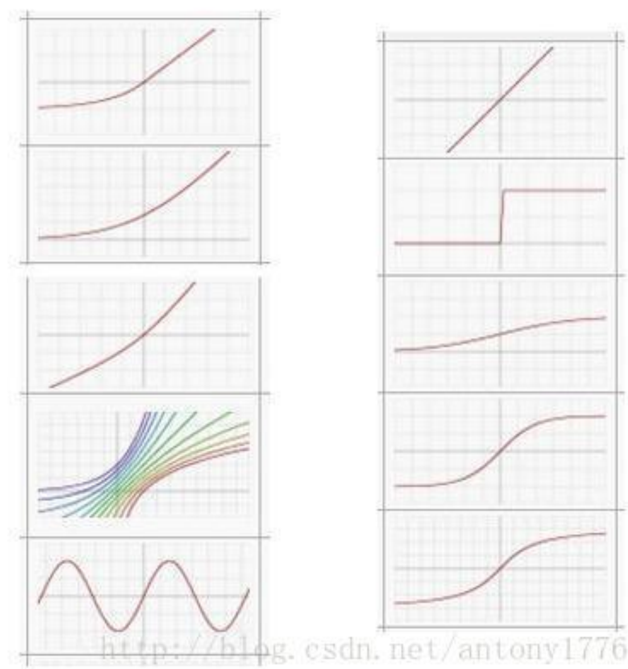


Sigmoid



Relu

Others



人工神经网络（ANN）

- 人工神经网络的重要概念：

　激励函数很重要，无论是对建立神经网络的模型，还是理解神经网络。首先要了解，它有以下几个影响：

1 如何能更好的求解目标函数的极值！——高等数学中求解函数极值的知识！可微，单调！

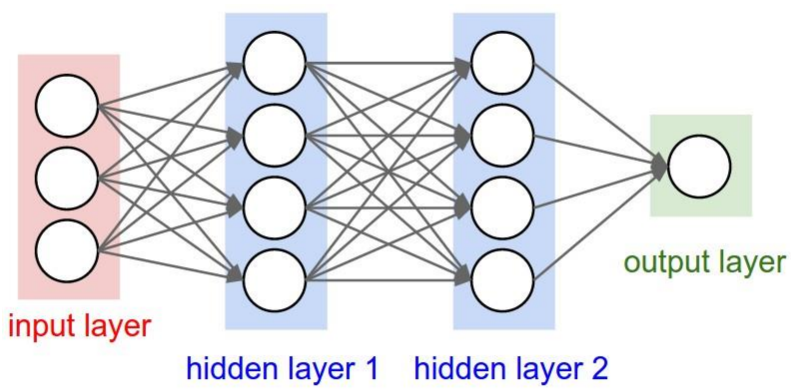
2 如何提升训练效率，让梯度的优化方法更稳定；

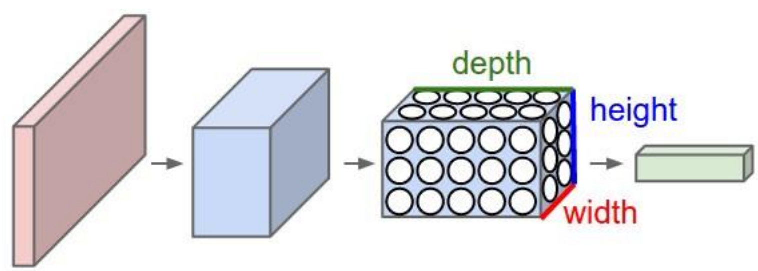
3 权值的初始值，不影响训练结果！

卷积神经网络（CNN）

- 卷积神经网络（Convolutional Neural Networks / CNNs / ConvNets）与普通神经网络非常相似，它们都由具有可学习的权重和偏置常量(biases)的神经元组成。每个神经元都接收一些输入，并做一些点积计算，输出是每个分类的分数，普通神经网络里的一些计算技巧到这里依旧适用。

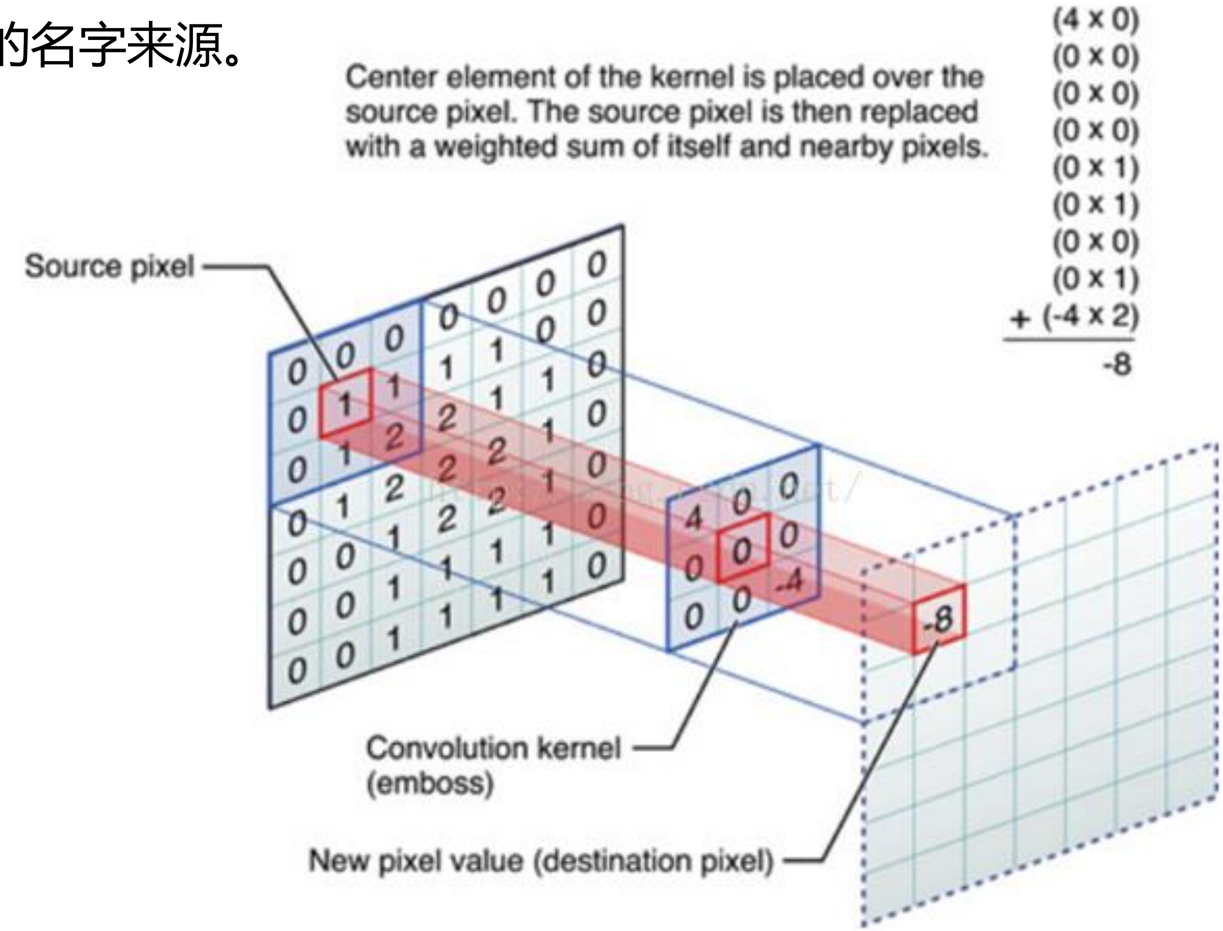
- 与普通神经网络不同之处：卷积神经网络默认输入是图像，可以让我们把特定的性质编码入网络结构，使是我们的前馈函数更加有效率，并减少了大量参数。



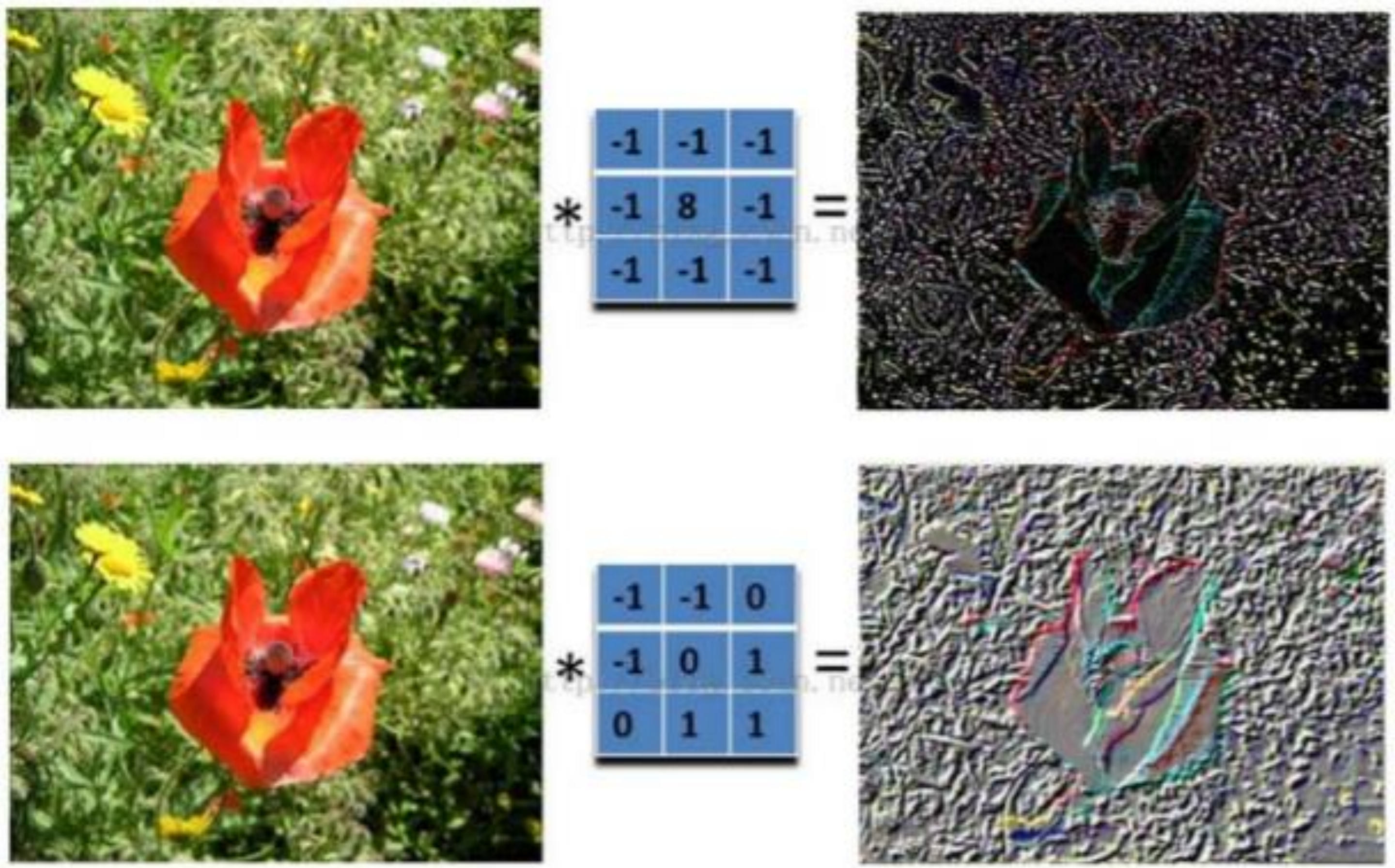


卷积神经网络（CNN）

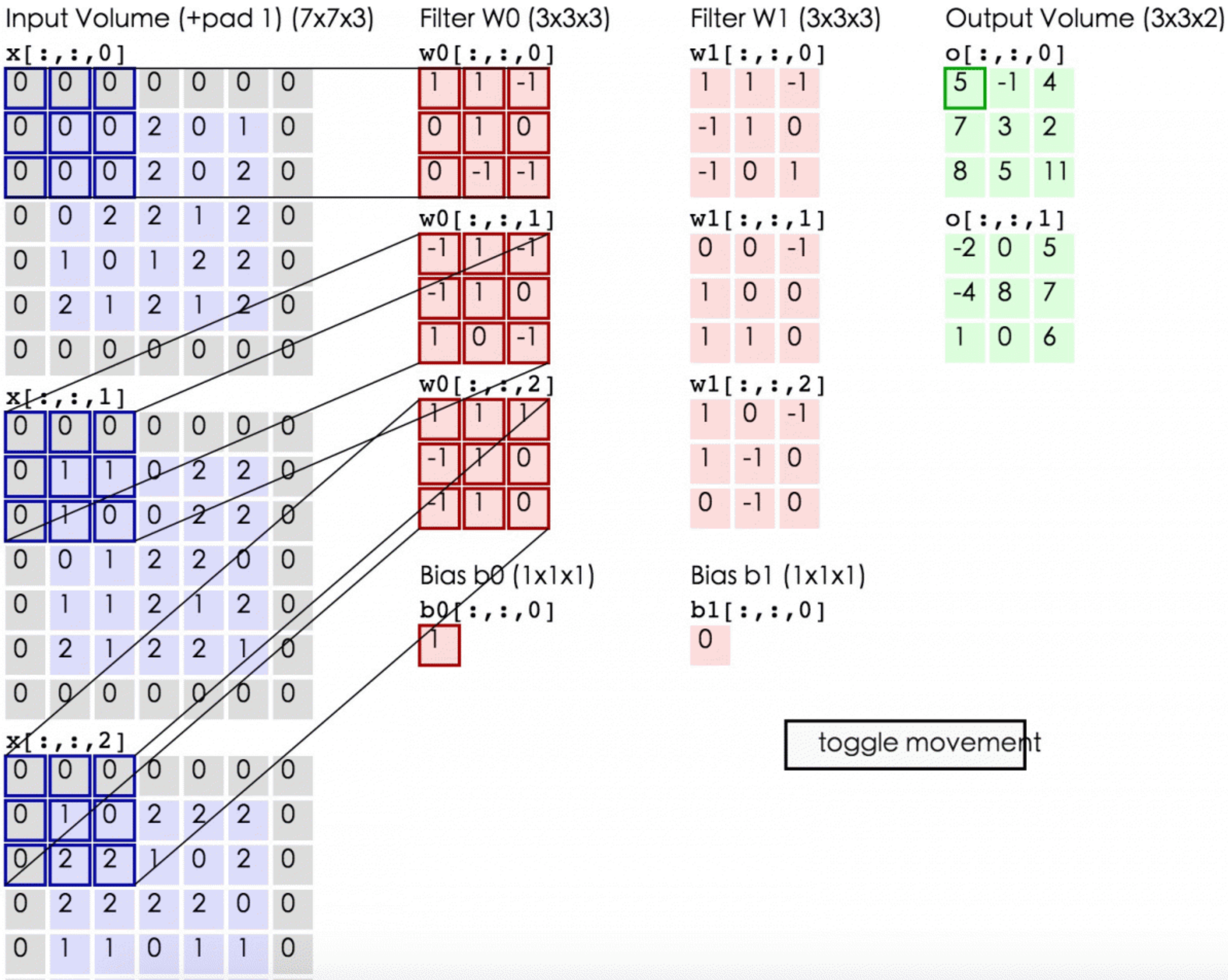
- 卷积操作：对图像（不同的数据窗口数据）和滤波矩阵（一组固定的权重：因为每个神经元的权重固定，所以又可以看做一个恒定的滤波器filter）做内积（逐个元素相乘再求和）的操作就是所谓的『卷积』操作，也是卷积神经网络的名字来源。



卷积神经网络（CNN）



卷积神经网络（CNN）

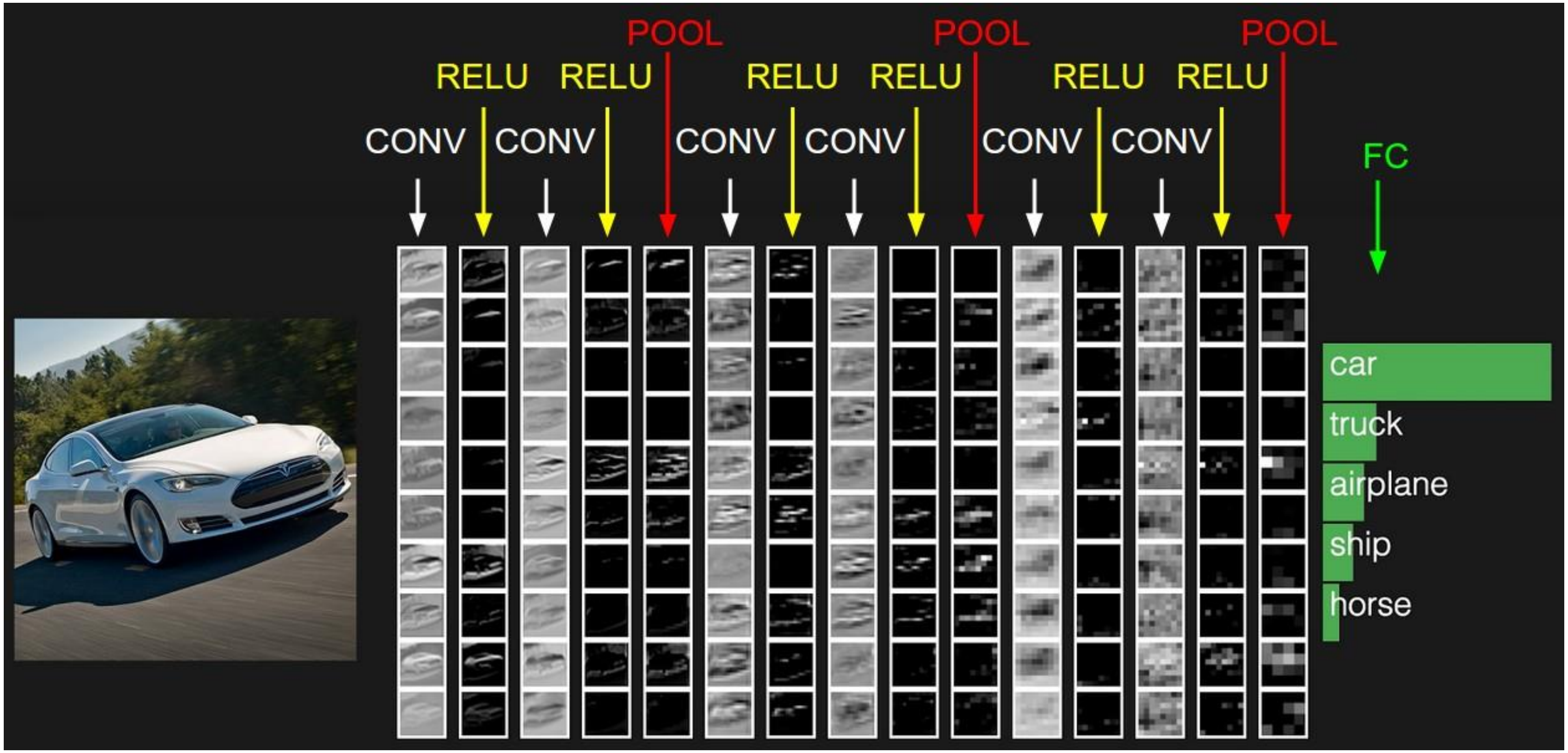


Input Volume (+pad 1) (7x7x3)
x[:,:,0]
x[:,:,1]
x[:,:,2]
Filter W0 (3x3x3)
w0[:,:,0]
w0[:,:,1]
w0[:,:,2]
Filter W1 (3x3x3)
w1[:,:,0]
w1[:,:,1]
w1[:,:,2]
Output Volume (3x3x2)
o[:,:,0]
o[:,:,1]
Bias b0 (1x1x1)
b0[:,:,0]
Bias b1 (1x1x1)
b1[:,:,0]
toggle movement

卷积神经网络（CNN）

- 卷积层（Convolutional layer），卷积神经网路中每层卷积层由若干卷积单元组成，每个卷积单元的参数都是通过反向传播算法优化得到的。卷积运算的目的是提取输入的不同特征，第一层卷积层可能只能提取一些低级的特征如边缘、线条和角等层级，更多层的网络能从低级特征中迭代提取更复杂的特征。
- 线性整流层（Rectified Linear Units layer, ReLU layer），这一层神经的激励函数（Activation function）使用线性整流（Rectified Linear Units, ReLU）f(x)=max(0,x)。
- 池化层（Pooling layer），通常在卷积层之后会得到维度很大的特征，将特征切成几个区域，取其最大值或平均值，得到新的、维度较小的特征。
- 全连接层（ Fully-Connected layer）, 把所有局部特征结合变成全局特征，用来计算最后每一类的得分。

卷积神经网络（CNN）



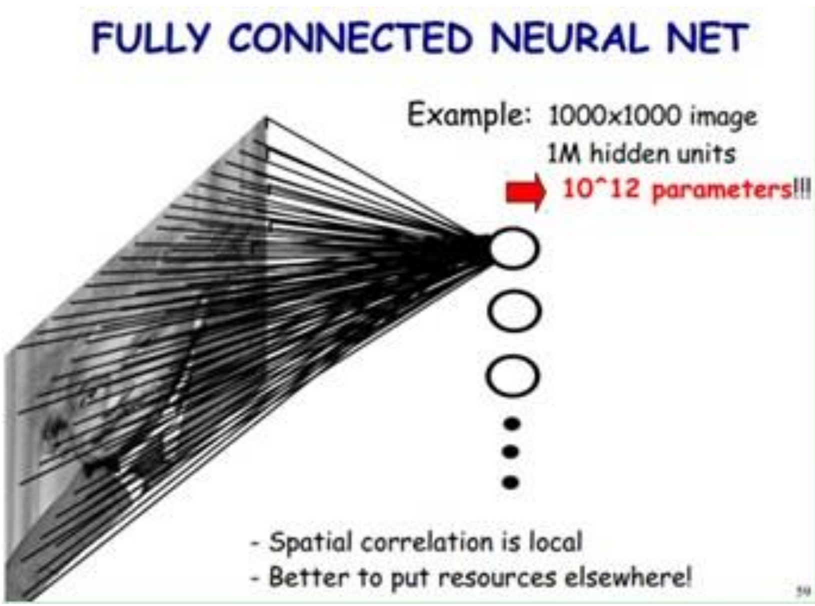
POOL
POOL
POOL
RELU
RELU
RELU
RELU
RELU
RELU
CONV
CONV
CONV
CONV
CONV
CONV
FC
car
truck
airplane
ship
horse

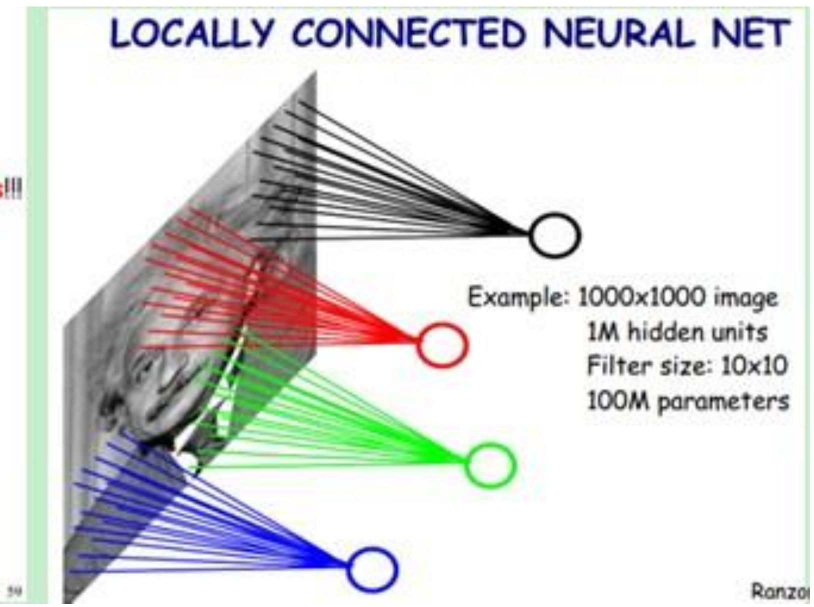
卷积神经网络（CNN）

- 在图像处理中，往往把图像表示为像素的向量，比如一个1000×1000的图像，可以表示为一个1000000的向量。在上一节中提到的神经网络中，如果隐含层数目与输入层一样，即也是1000000时，那么输入层到隐含层的参数数据为1000000×1000000=10^12，这样就太多了，基本没法训练。所以图像处理要想练成神经网络大法，必先减少参数加快速度。就跟辟邪剑谱似的，普通人练得很挫，一旦自宫后内力变强剑法变快，就变的很牛了。

卷积神经网络（CNN）

- 卷积神经网络有两种神器可以降低参数数目，第一种神器叫做局部感知。
- 在下方右图中，假如每个神经元只和10×10个像素值相连，那么权值数据为1000000×100个参数，减少为原来的万分之一。而那10×10个像素值对应的10×10个参数，其实就相当于卷积操作。



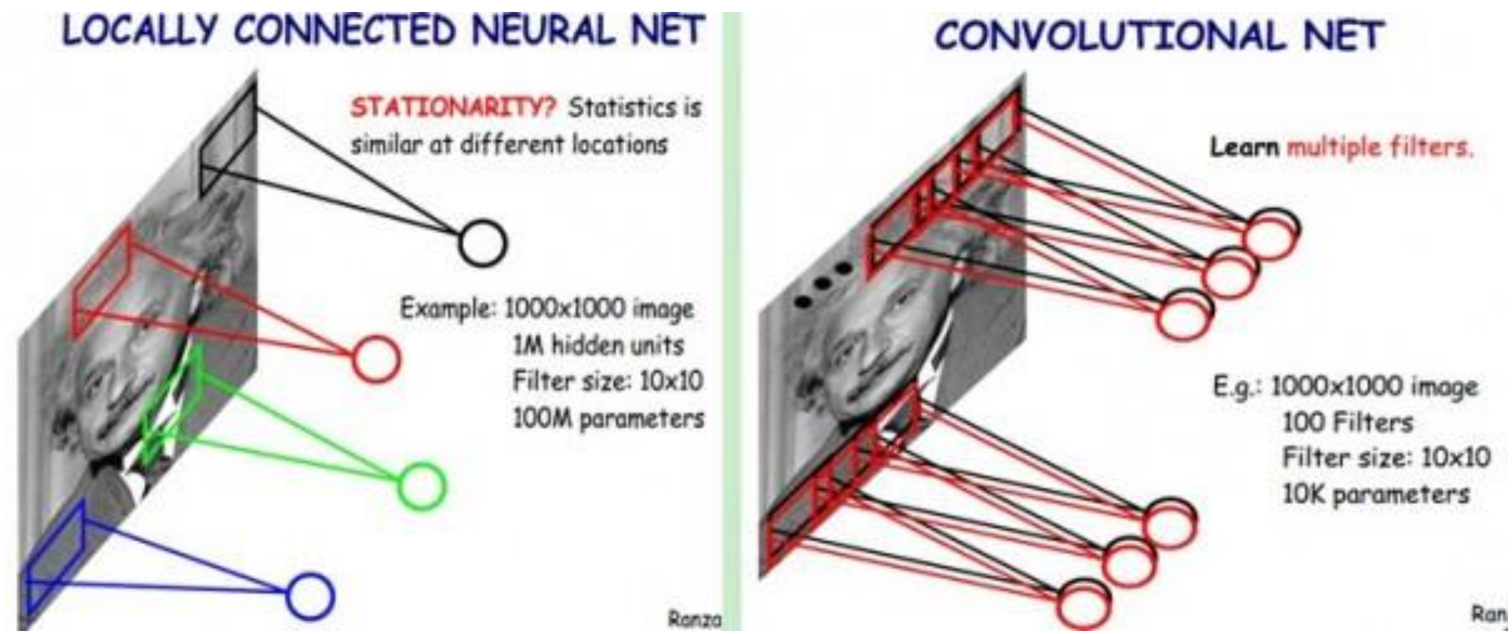


卷积神经网络（CNN）

- 这样的话参数仍然过多，那么就启动第二级神器，即权值共享。在上面的局部连接中，每个神经元都对应100个参数，一共1000000个神经元，如果这1000000个神经元的100个参数都是相等的，那么参数数目就变为100了。
- 怎么理解权值共享呢？我们可以这100个参数（也就是卷积操作）看成是提取特征的方式，该方式与位置无关。这其中隐含的原理则是：图像的一部分的统计特性与其他部分是一样的。这也意味着我们在这一部分学习的特征也能用在另一部分上，所以对于这个图像上的所有位置，我们都能使用同样的学习特征。

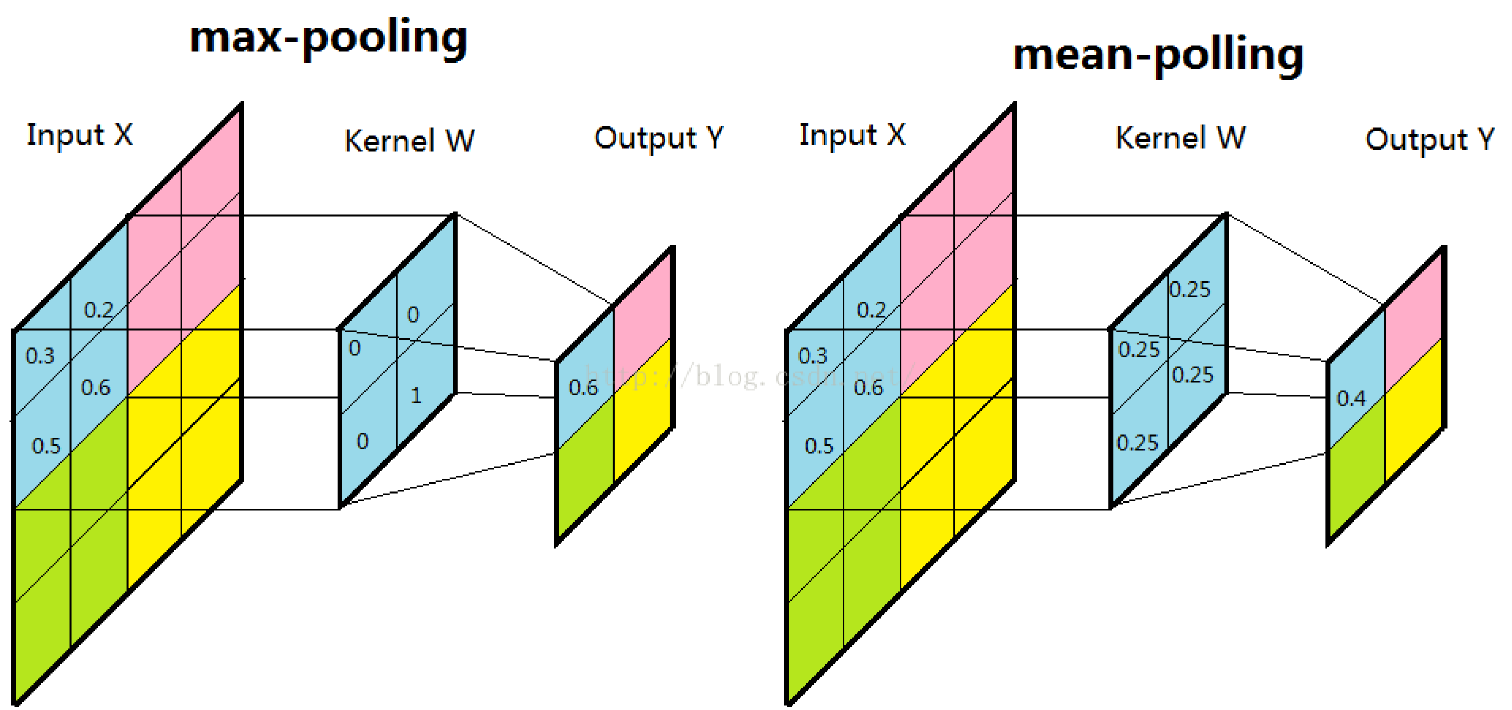
卷积神经网络（CNN）

- 上面所述只有100个参数时，表明只有1个100*100的卷积核，显然，特征提取是不充分的，我们可以添加多个卷积核，比如32个卷积核，可以学习32种特征。在有多个卷积核时，如下图所示：



卷积神经网络（CNN）

- 池化，也称作下采样，可以实现降维。常用有最大值池化和均值池化。

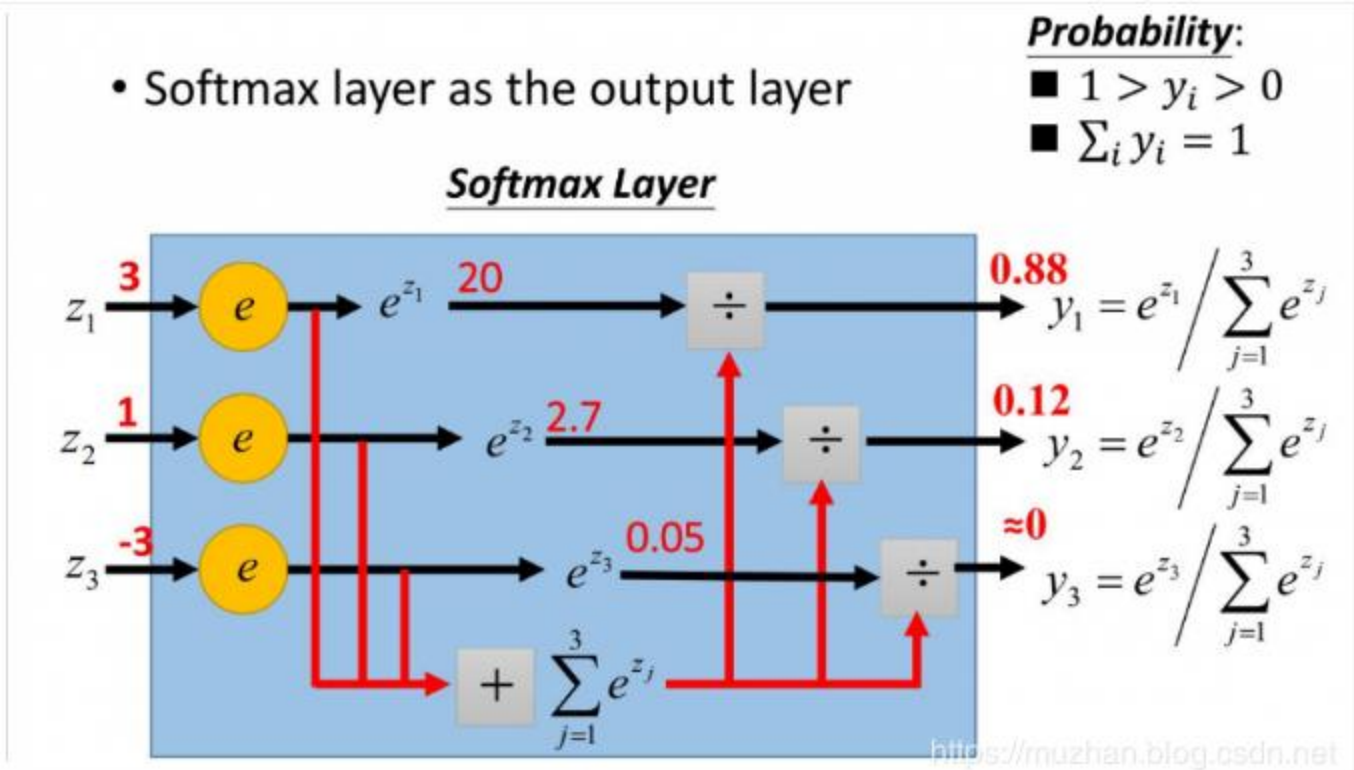


卷积神经网络（CNN）

- 全连接层：连接所有的特征，将输出值送给分类器（如softmax分类器），最终得出识别结果。

softmax分类器

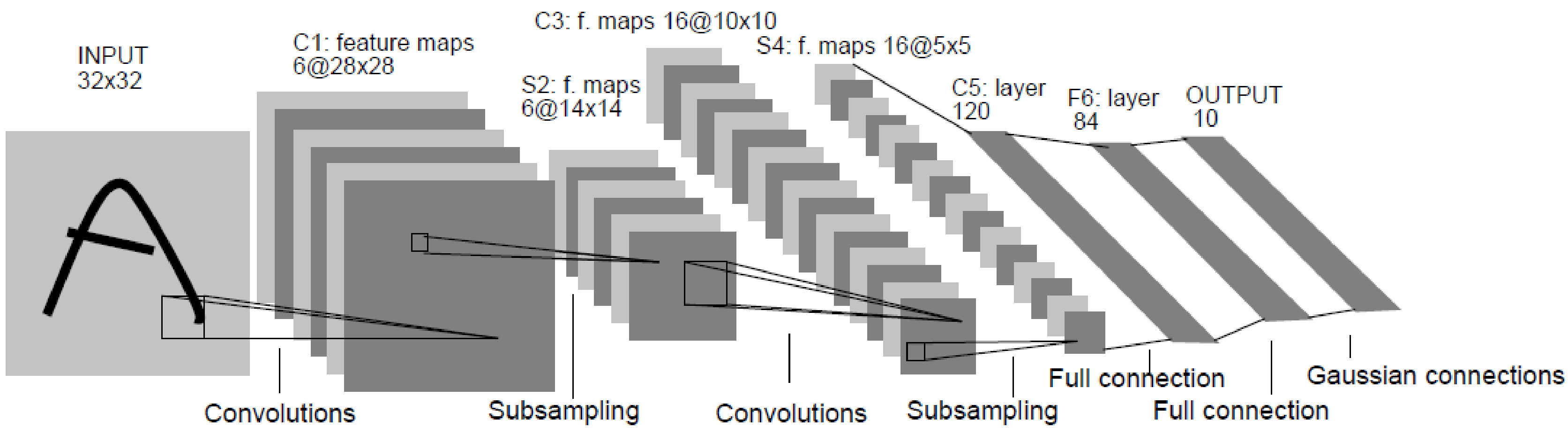
$$softmax(x_0) = \frac{e^{x_0}}{e^{x_0} + e^{x_1} + e^{x_2}}$$



softmax对神经元的输出信号进行加工，输出为分类的概率值。

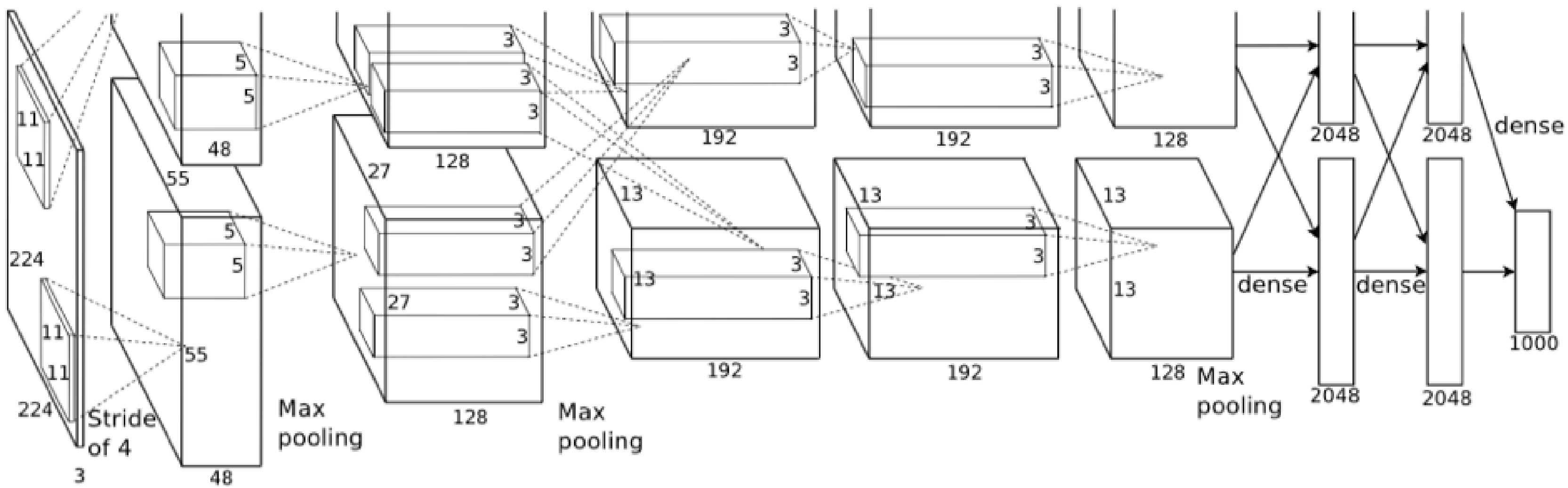
常见网络模型

- LeNet



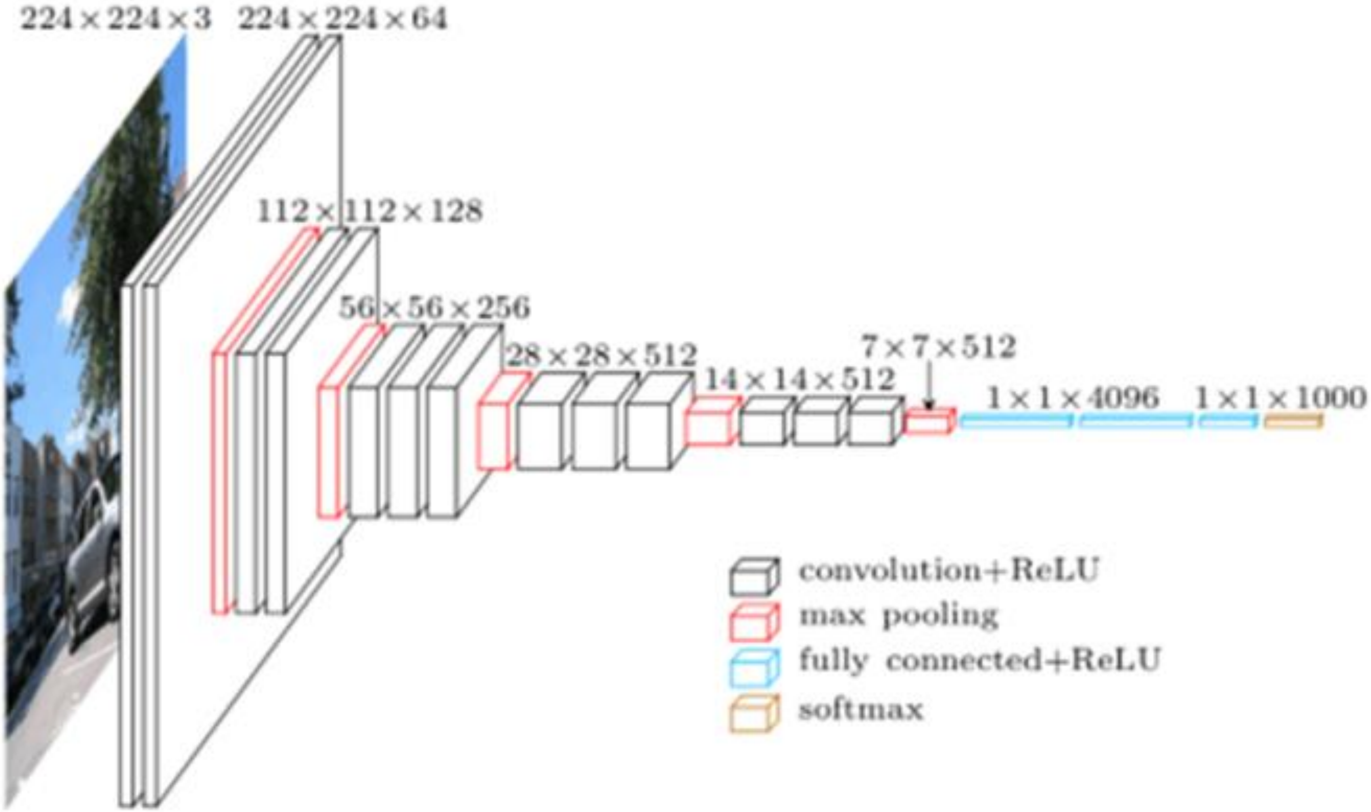
常见网络模型

- AlexNet



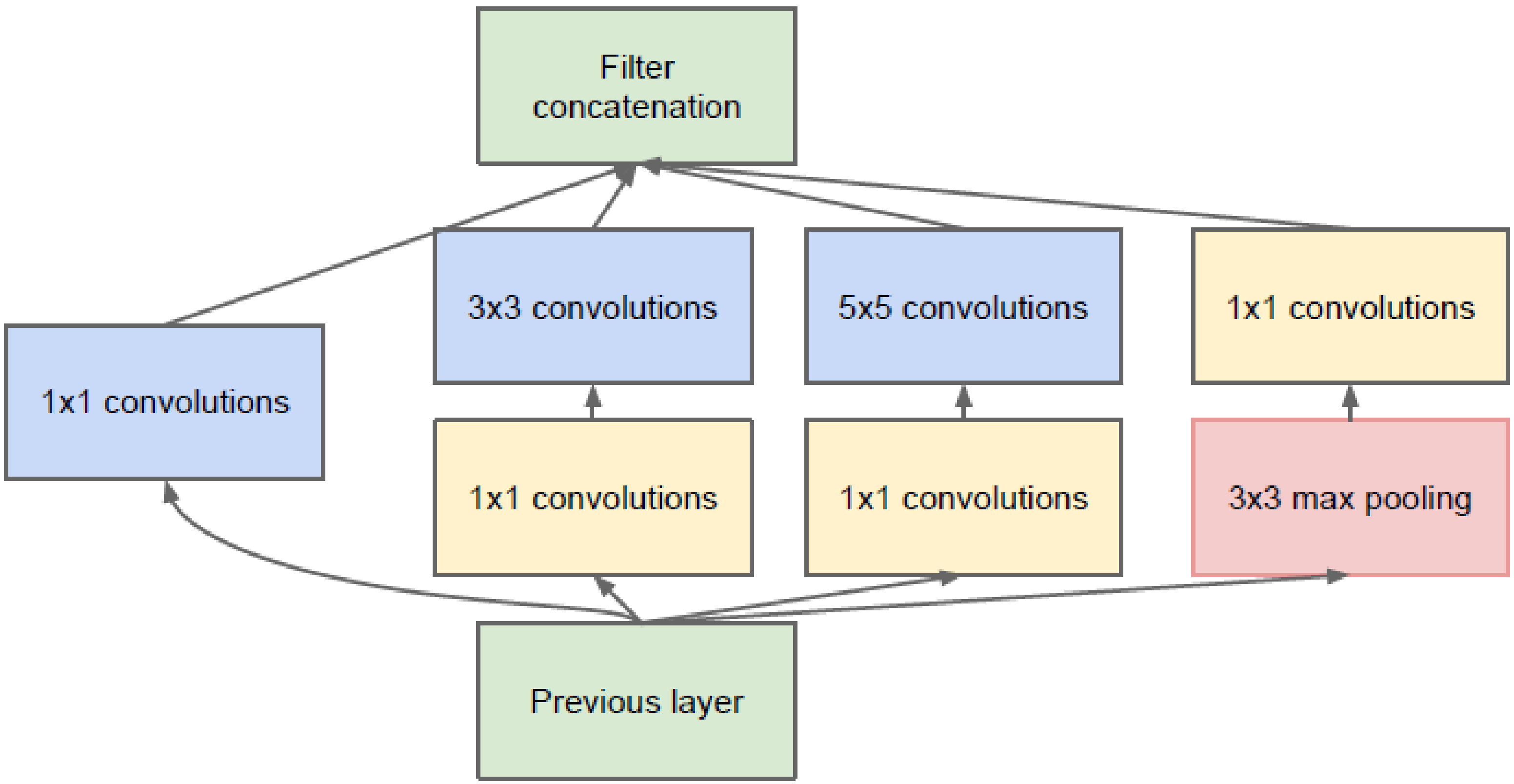
常见网络模型

- VGG16



常见网络模型

- GoogleNet （InceptionV4）



常见网络模型

CNN模型比较.doc

- 比较

模型名	AlexNet	VGG	GoogLeNet	ResNet
初入江湖	2012	2014	2014	2015
层数	8	19	22	152
Top-5错误	16.4%	7.3%	6.7%	3.57%
Data Augmentation	+	+	+	+
Inception(NIN)	–	–	+	–
卷积层数	5	16	21	151
卷积核大小	11,5,3	3	7,1,3,5	7,1,3,5
全连接层数	3	3	1	1
全连接层大小	4096,4096,1000	4096,4096,1000	1000	1000
Dropout	+	+	+	+
Local Response Normalization	+	–	+	–
Batch Normalization	–	–	–	+

其他深度学习算法

- 自动编码器（AutoEncoder）
- 稀疏编码（Sparse Coding）
- 限制玻尔兹曼机（RBM）

深度学习的特点

深度学习常用算法介绍

深度学习常用框架介绍

开源框架概述

- 深度学习研究的热潮持续高涨，各种开源深度学习框架也层出不穷，其中包括TensorFlow、Caffe、Keras、CNTK、Torch7、MXNet、Leaf、Theano、DeepLearning4、Lasagne、Neon等等。下图是各个开源框架在GitHub上的数据统计（2017年初）。

框　架	机　构	支持语言	Stars	Forks	Contributors
TensorFlow	Google	Python/C++/Go/...	41628	19339	568
Caffe	BVLC	C++/Python	14956	9282	221
Keras	fchollet	Python	10727	3575	322
CNTK	Microsoft	C++	9063	2144	100
MXNet	DMLC	Python/C++/R/...	7393	2745	241
Torch7	Facebook	Lua	6111	1784	113
Theano	U. Montreal	Python	5352	1868	271
Deeplearning4J	DeepLearning4J	Java/Scala	5053	1927	101
Leaf	AutumnAI	Rust	4562	216	14
Lasagne	Lasagne	Python	2749	761	55
Neon	NervanaSystems	Python	2633	573	52

开源框架概述

- Google、Microsoft、Facebook等巨头都参与了这场深度学习框架大战，此外，还有毕业于伯克利大学的贾扬清主导开发的Caffe，蒙特利尔大学Lisa Lab团队开发的Theano，以及其他个人或商业组织贡献的框架。下表是主流深度学习框架在各个维度的评分。

	模型设计	接　口	部　署	性　能	架构设计	总体评分
TensorFlow	80	80	90	90	100	88
Caffe	60	60	90	80	70	72

续表

	模型设计	接　口	部　署	性　能	架构设计	总体评分
CNTK	50	50	70	100	60	66
Theano	80	70	40	50	50	58
Torch	90	70	60	70	90	76
MXNet	70	100	80	80	90	84
DeepLearning4J	60	70	80	80	70	72

TensorFlow

- TensorFlow最初是由研究人员和Google Brain团队针对机器学习和深度神经网络进行研究所开发的，目前开源之后可以在几乎各种领域适用。
- TensorFlow灵活的架构可以部署在一个或多个CPU、GPU的台式以及服务器中，或者使用单一的API应用在移动设备中。

TensorFlow™

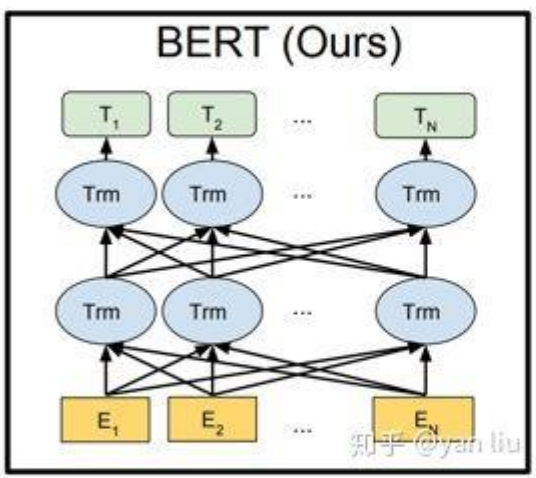
Torch

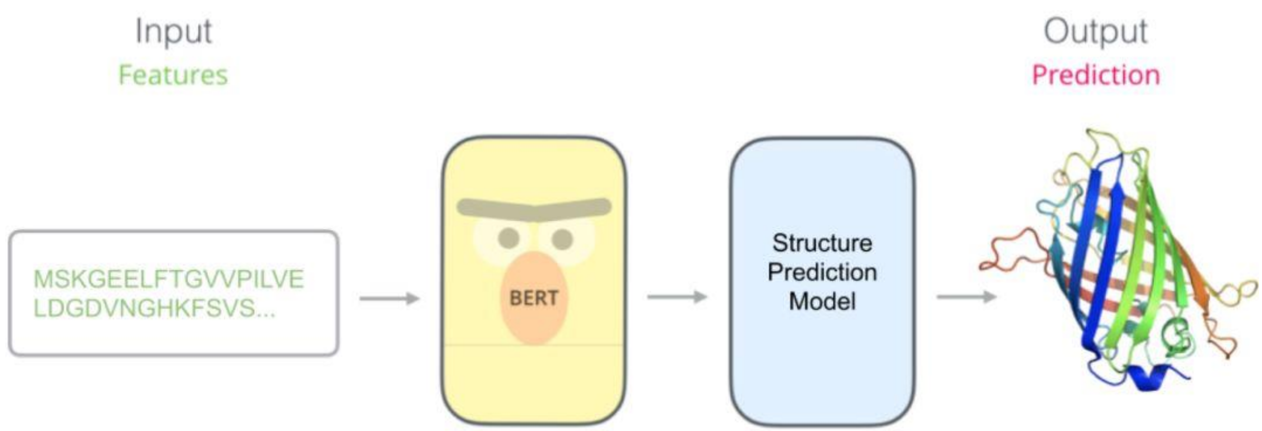
- Torch是一个有大量机器学习算法支持的科学计算框架，其诞生已经有十年之久，但是真正起势得益于Facebook开源了大量Torch的深度学习模块和扩展。Torch另外一个特殊之处是采用了编程语言Lua(该语言曾被用来开发视频游戏)。

- PyTorch是基于Torch的衍生，支持Python语言，实现了机器学习框架 Torch 在 Python 语言环境的执行。

Bert

- 使用Transformer的结构将已经走向瓶颈期的Word2Vec带向了一个新的方向，并再一次炒火了《Attention is All you Need》这篇论文；
- 11个NLP任务的精度大幅提升足以震惊整个深度学习领域；
- 无私的开源了多种语言的源码和模型，具有非常高的商业价值；
- 迁移学习又一次胜利，而且这次是在NLP领域的大胜，狂胜。





Caffe

- Caffe由加州大学伯克利的PHD贾扬清开发，全称Convolutional Architecture for Fast Feature Embedding，是一个清晰而高效的开源深度学习框架，目前由伯克利视觉学中心（Berkeley Vision and Learning Center，BVLC）进行维护。（贾扬清曾就职于MSRA、NEC、Google Brain，他也是TensorFlow的作者之一，目前任职于Facebook FAIR实验室。）
- Caffe2脸书 (Facebook) 出品，为生产环境设计，提供在各种平台（包括移动设备）的运行。

Theano

- 2008年诞生于蒙特利尔理工学院,Theano派生出了大量深度学习Python软件包，最著名的包括Blocks和Keras。Theano的核心是一个数学表达式的编译器，它知道如何获取你的结构。并使之成为一个使用numpy、高效本地库的高效代码，如BLAS和本地代码（C++）在CPU或GPU上尽可能快地运行。它是为深度学习中处理大型神经网络算法所需的计算而专门设计的，是这类库的首创之一（发展始于2007年），被认为是深度学习研究和开发的行业标准。

Deeplearning4j

- Deeplearning4j是“for Java”的深度学习框架，也是首个商用级别的深度学习开源库。Deeplearning4j由创业公司Skymind于2014年6月发布，使用Deeplearning4j的不乏埃森哲、雪弗兰、博斯咨询和IBM等明星企业。DeepLearning4j是一个面向生产环境和商业应用的高成熟度深度学习开源库，可与Hadoop和Spark集成，即插即用，方便开发者在APP中快速集成深度学习功能。

DL4J Deep Learning for Java

MXNet

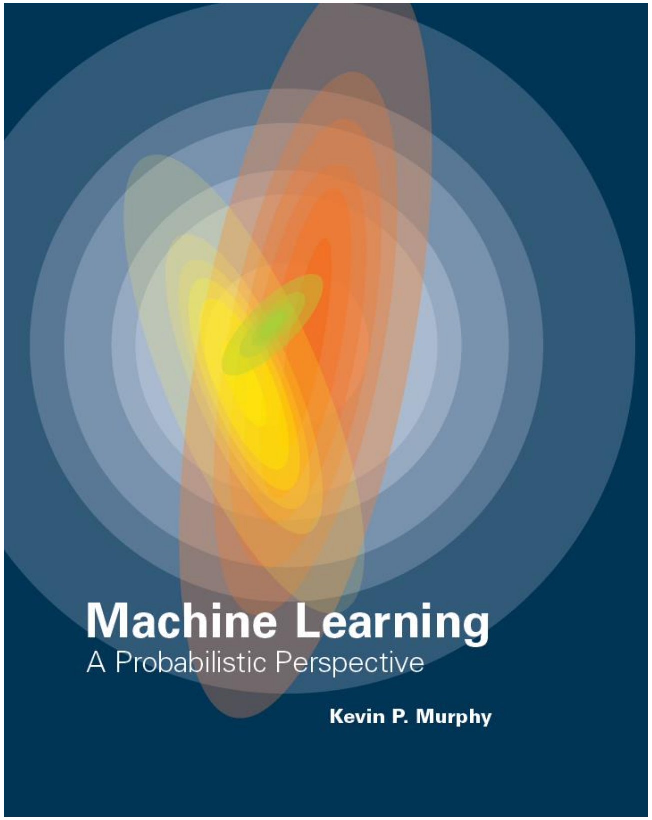
- 出自CXXNet、Minerva、Purine 等项目的开发者之手，主要用C++ 编写。MXNet 强调提高内存使用的效率，甚至能在智能手机上运行诸如图像识别等任务。

CNTK

- CNTK（Computational Network Toolkit）是微软研究院（MSR）开源的深度学习框架。它最早由start the deep learning craze的演讲人创建，目前已经发展成一个通用的、跨平台的深度学习系统，在语音识别领域的使用尤其广泛。

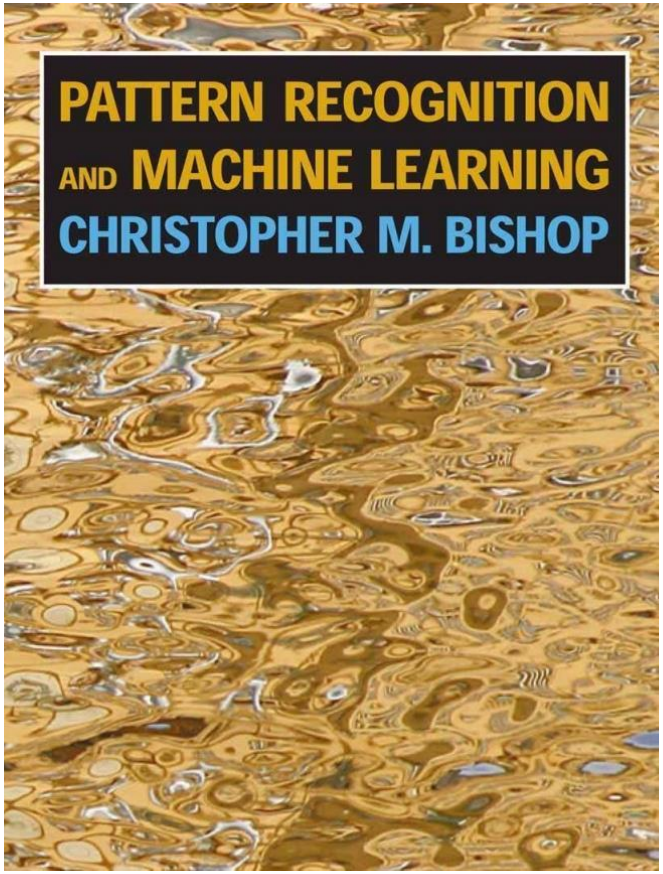
Microsoft
CNTK

References

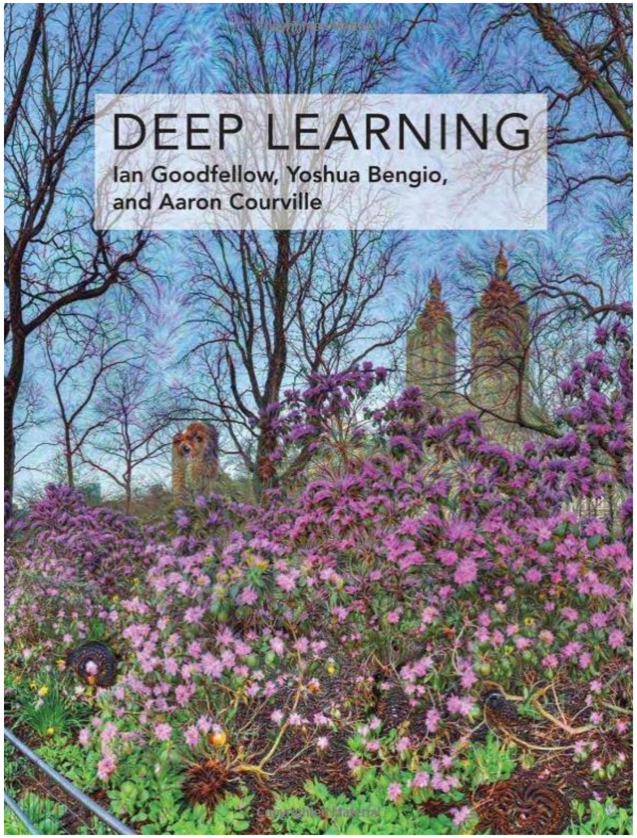


https://github.com/probml/pyprobml

References


PATTERN RECOGNITION
AND MACHINE LEARNING
CHRISTOPHER M. BISHOP

References



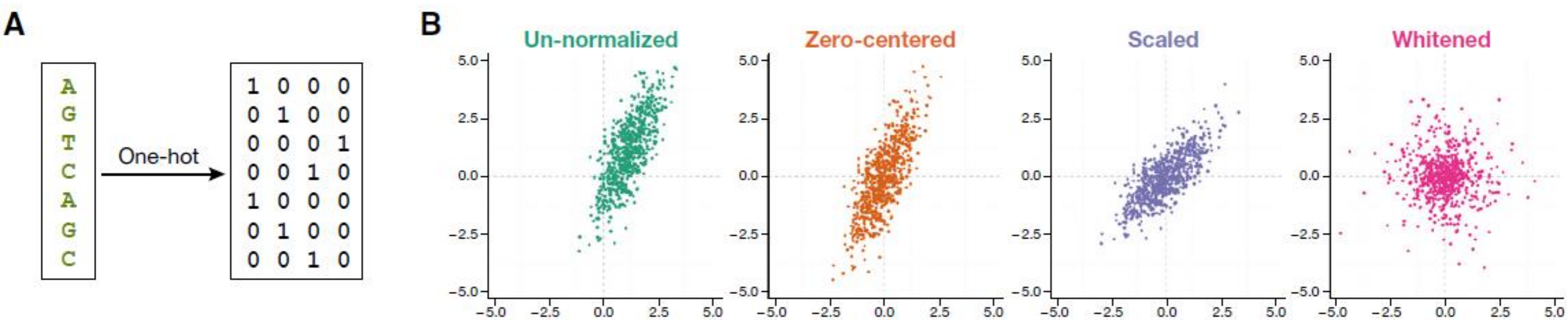
https://github.com/exacity/deeplearningbook-chinese

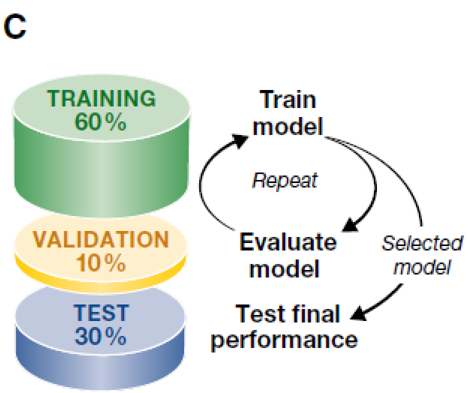
Part III

生物大数据的深度学习方法

Feature Selection

Collection→Partition→Normalization
Data label



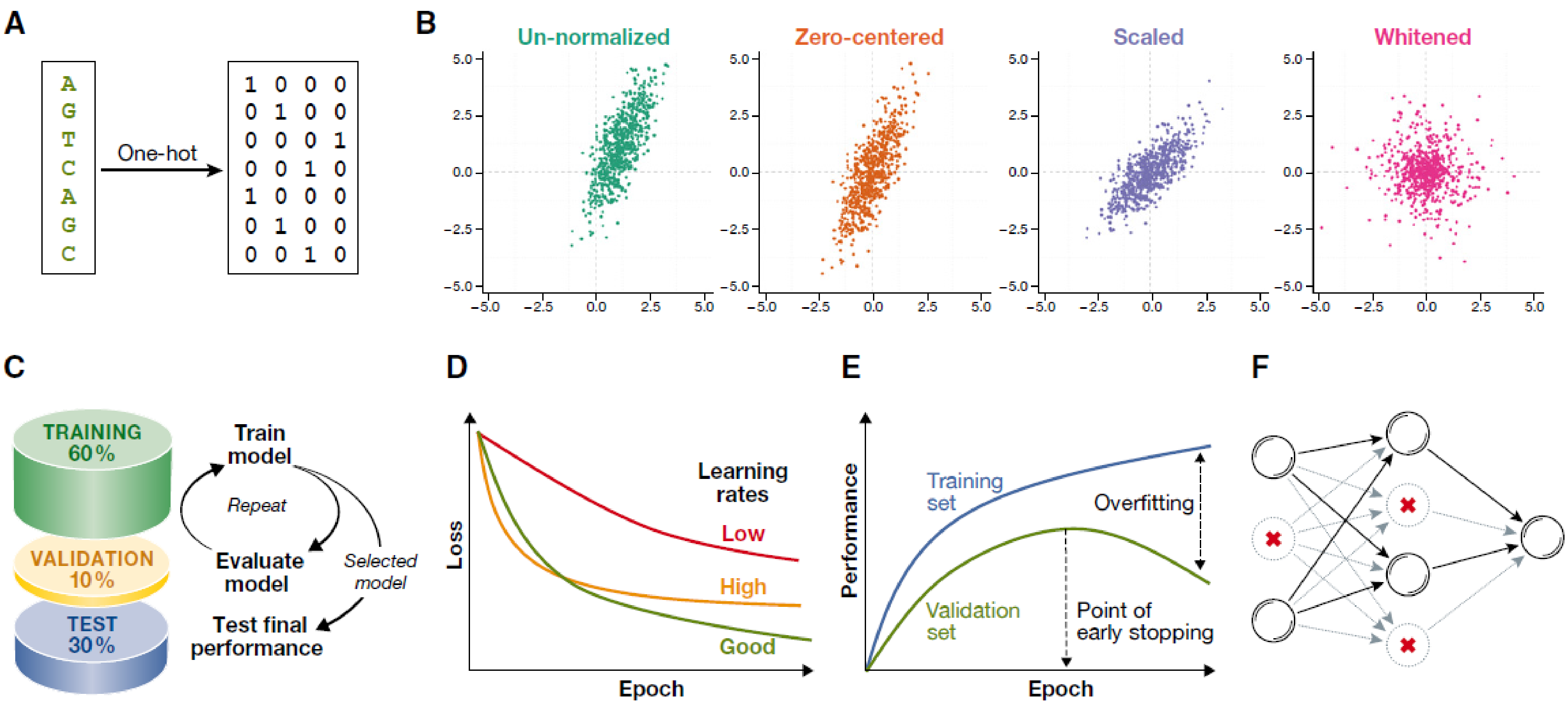


(A) Data One-hot Coding

(B) Different data normalization methods

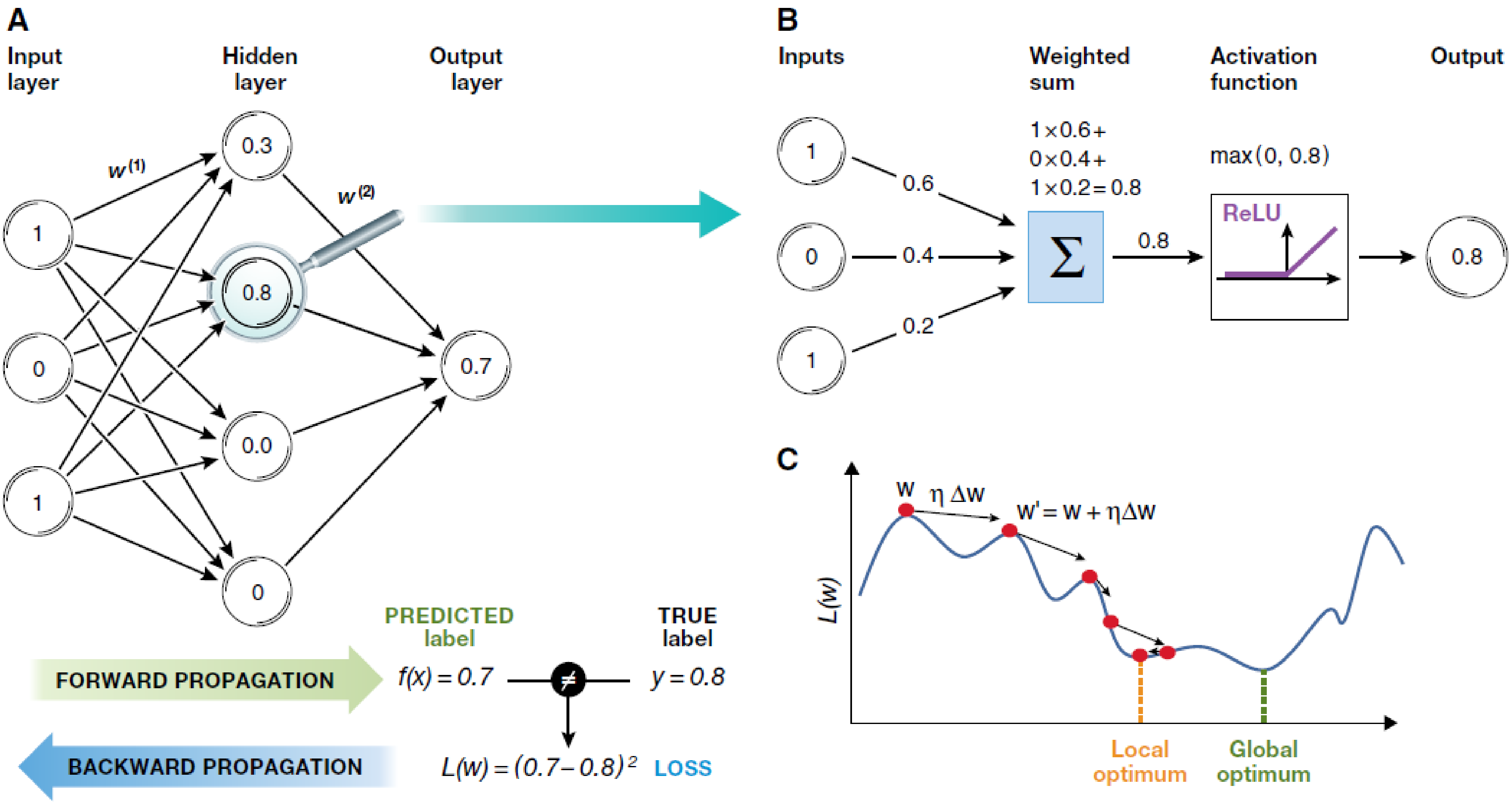
(C) Training data, validation data(adjust model structure),Test data

Data normalization



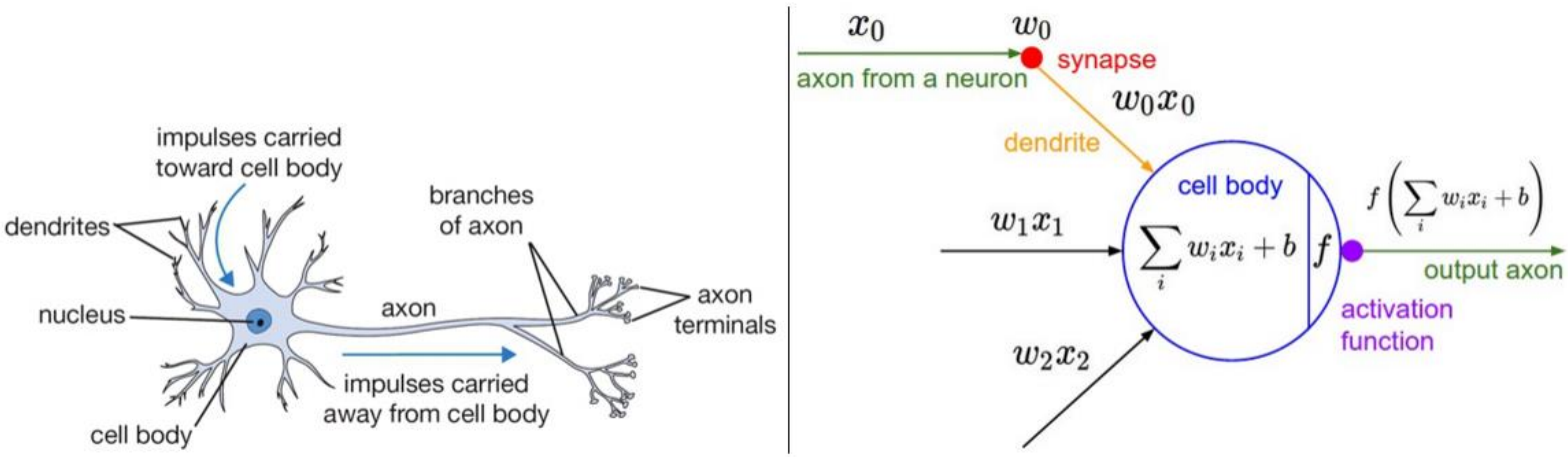
Adapted from “Deep learning for computational biology. Molecular Systems Biology, 2016.”

Artificial Neural Network



Adapted from "Deep learning for computational biology. Molecular Systems Biology, 2016."

Model Construction



A Cartoon drawing of a biological neuron(left) and its mathematical model(right)

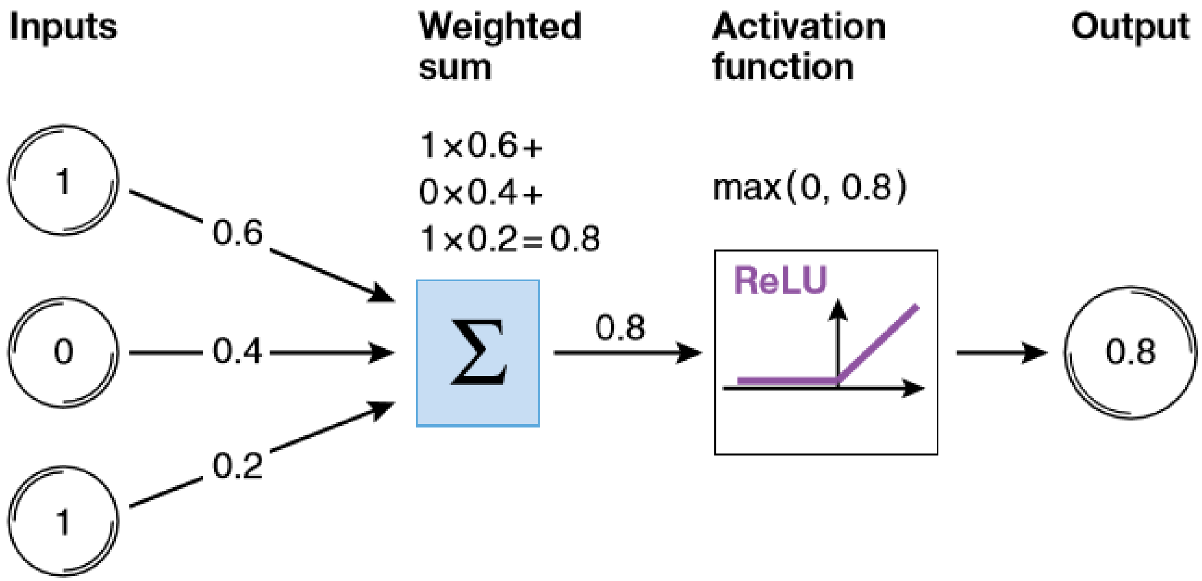
$$f\left(\sum_i w_i x_i + b\right)$$

Activation function

Translation

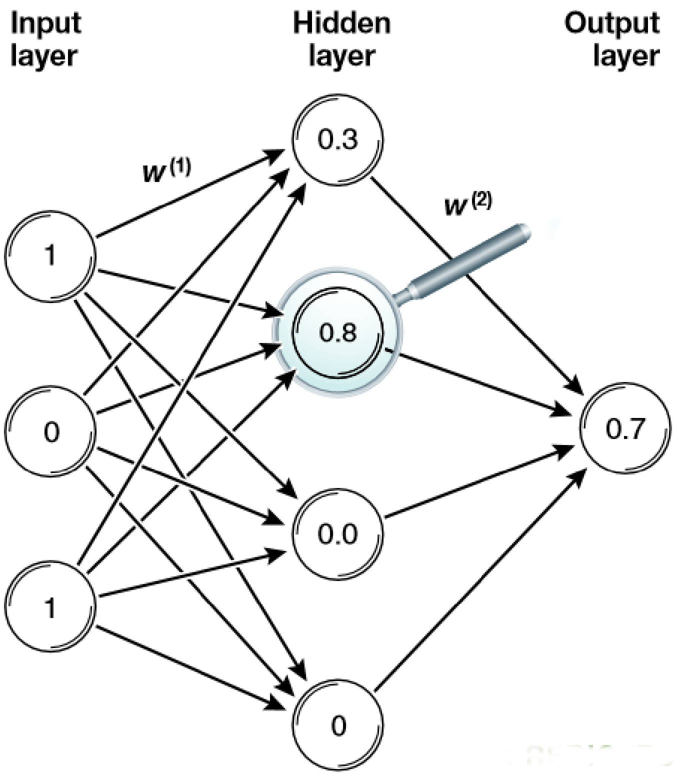
Input x_i and corresponded weight w_i

Model Construction



An example for a “transmission of neural signal”

Model Construction



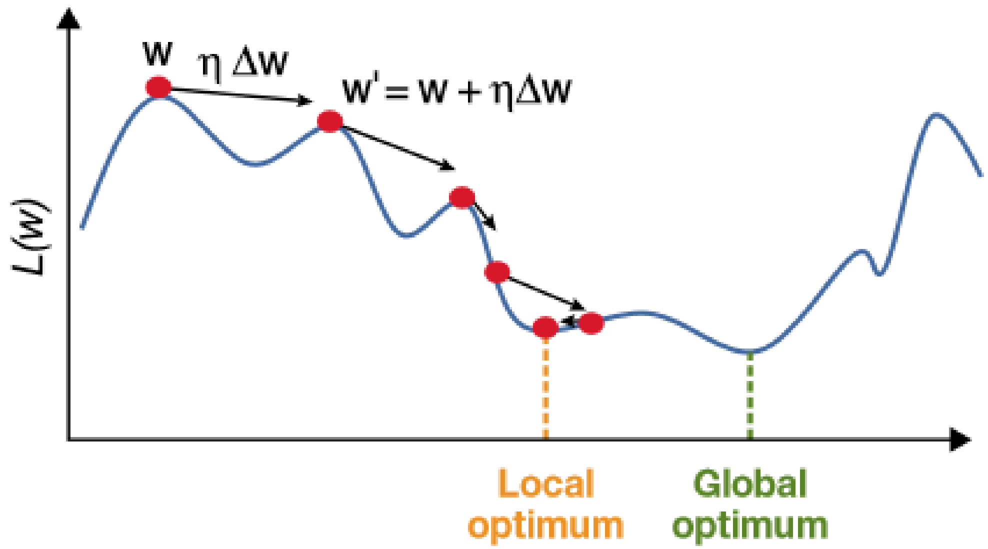
Neuron network construction process

Model construction：
Input label:1,0,1
Predicted label: 0.7

Ture data label:0.8

Pick the best model

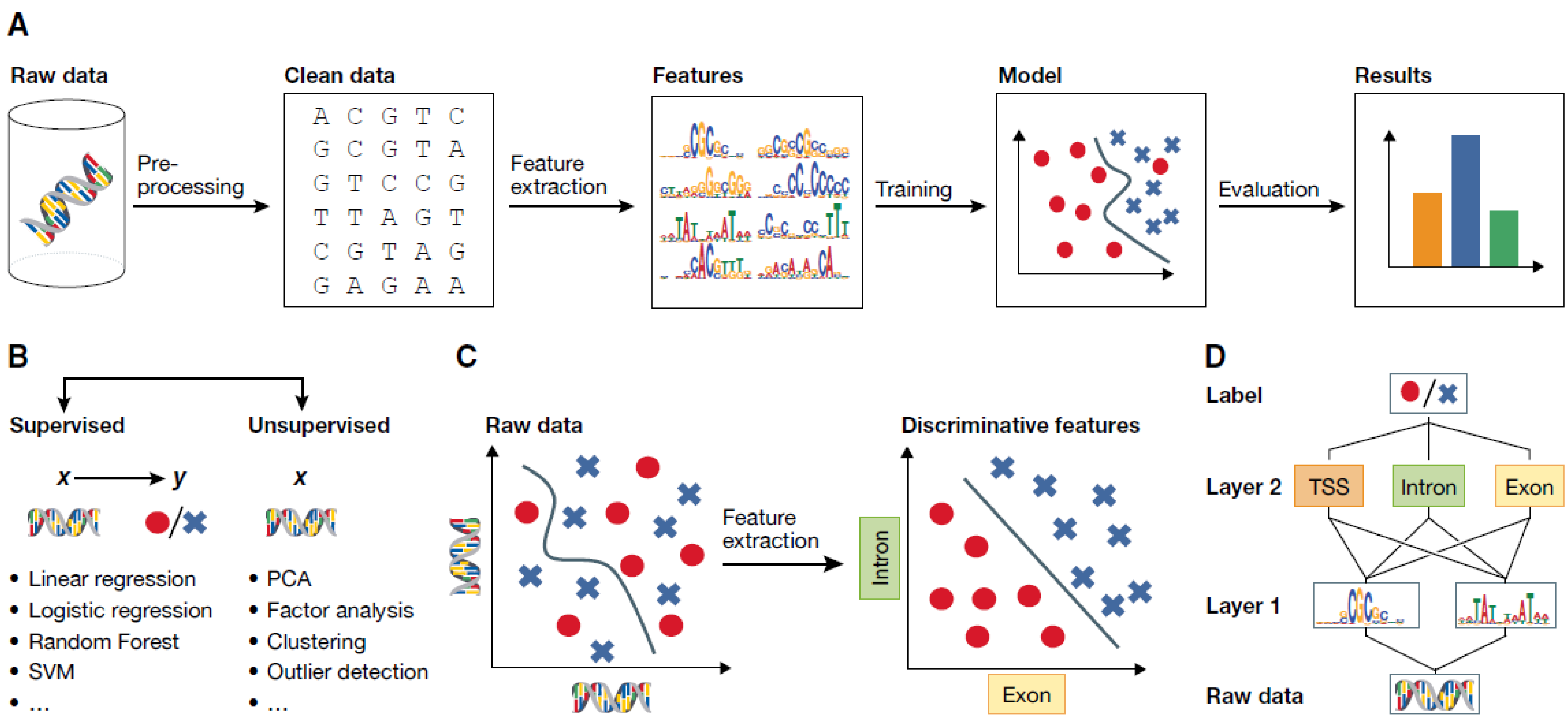
loss function: L(w)=(predicted label-true label)^2



Many parameters are adjusted to find the global minimums then update the neuron weight

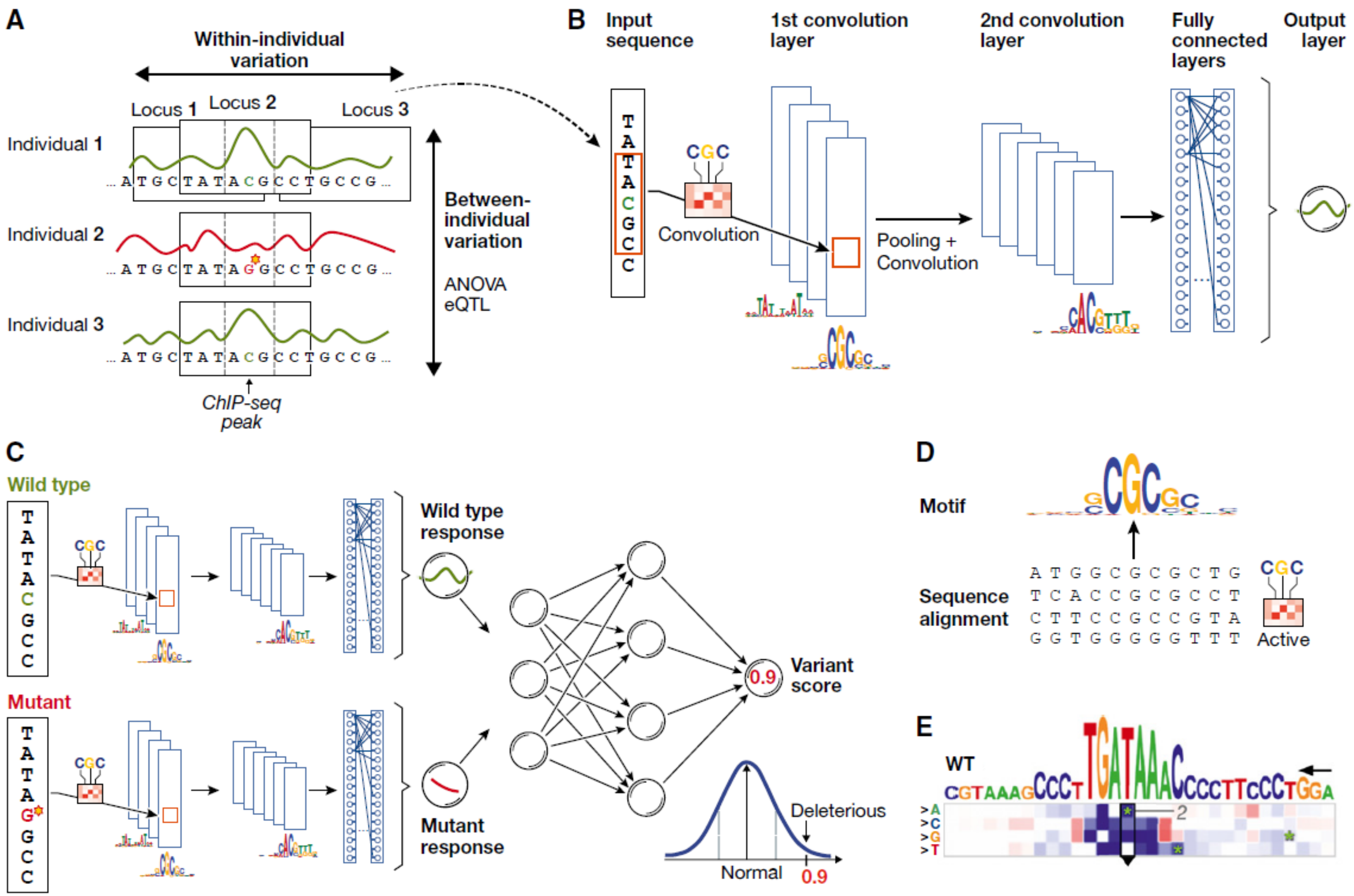
Predicted label:0.8 → True label:0.8

Machine learning and representation learning



Adapted from "Deep learning for computational biology. Molecular Systems Biology, 2016."

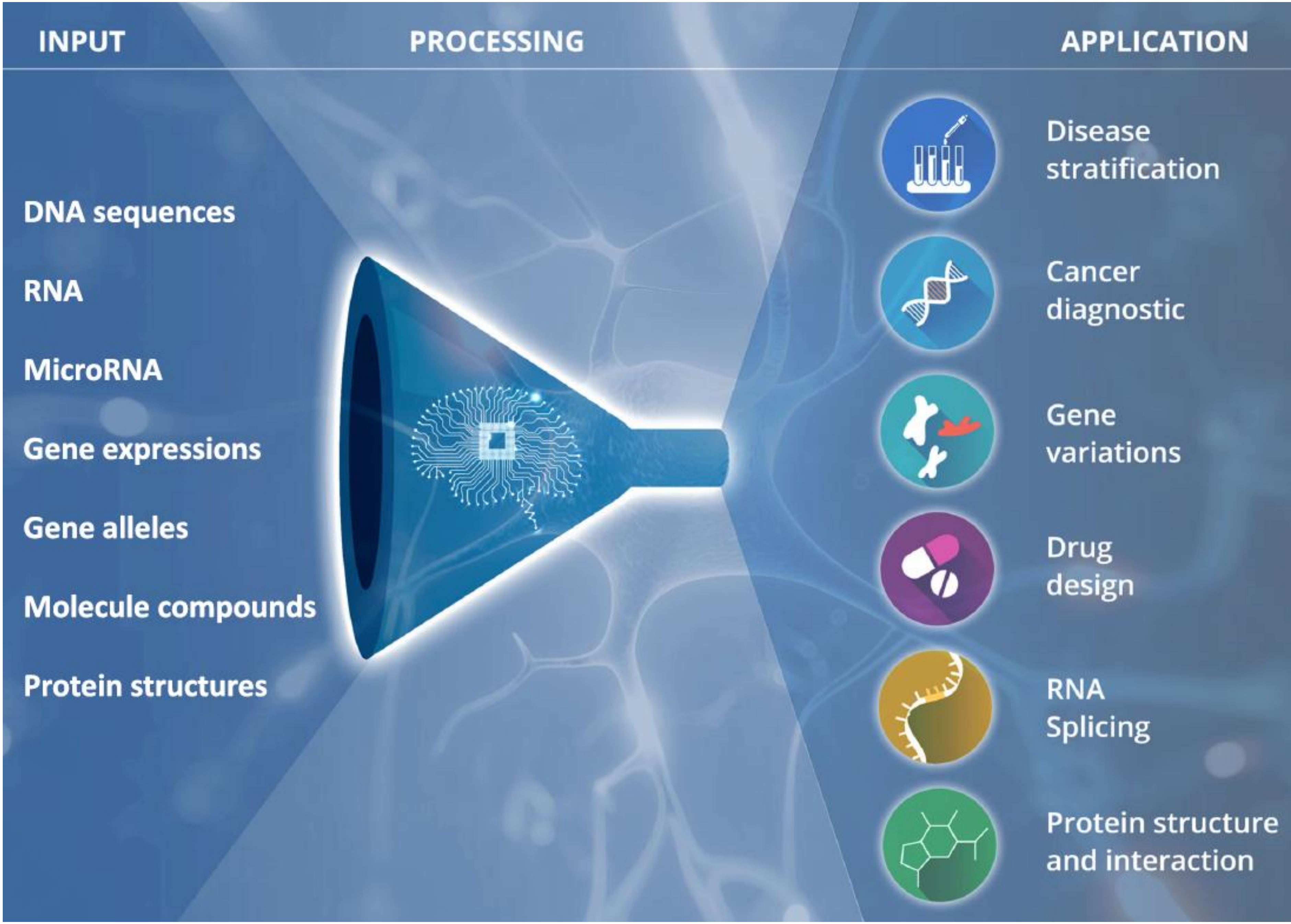
Neural Network for DNA sequences



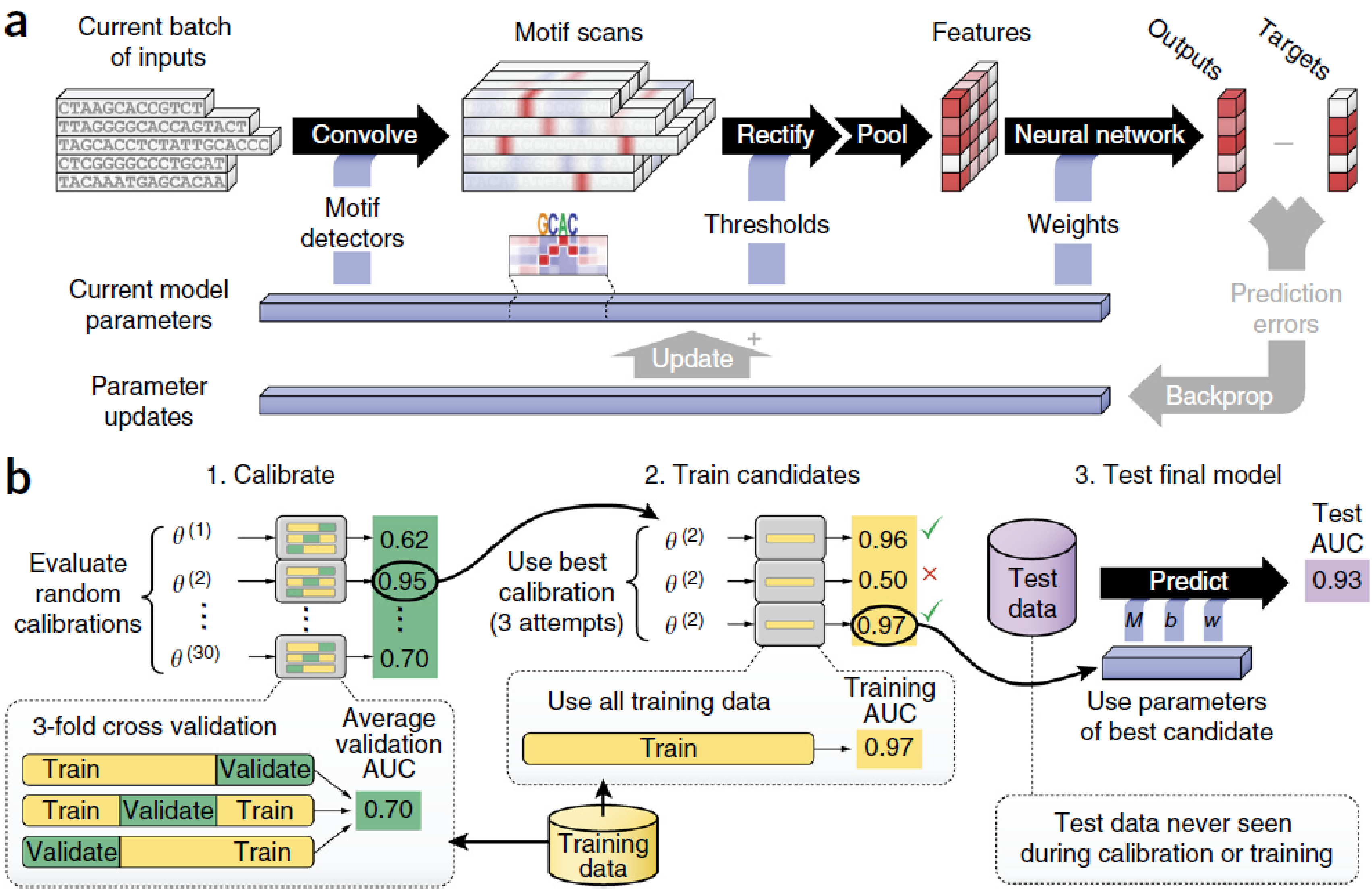
Adapted from “Deep learning for computational biology. Molecular Systems Biology, 2016.”

Part IV

生物大数据深度学习的应用案例


INPUT
PROCESSING
APPLICATION
DNA sequences
RNA
MicroRNA
Gene expressions
Gene alleles
Molecule compounds
Protein structures
Disease stratification
Cancer diagnostic
Gene variations
Drug design
RNA Splicing
Protein structure and interaction

DNA- and RNA-binding protein prediction

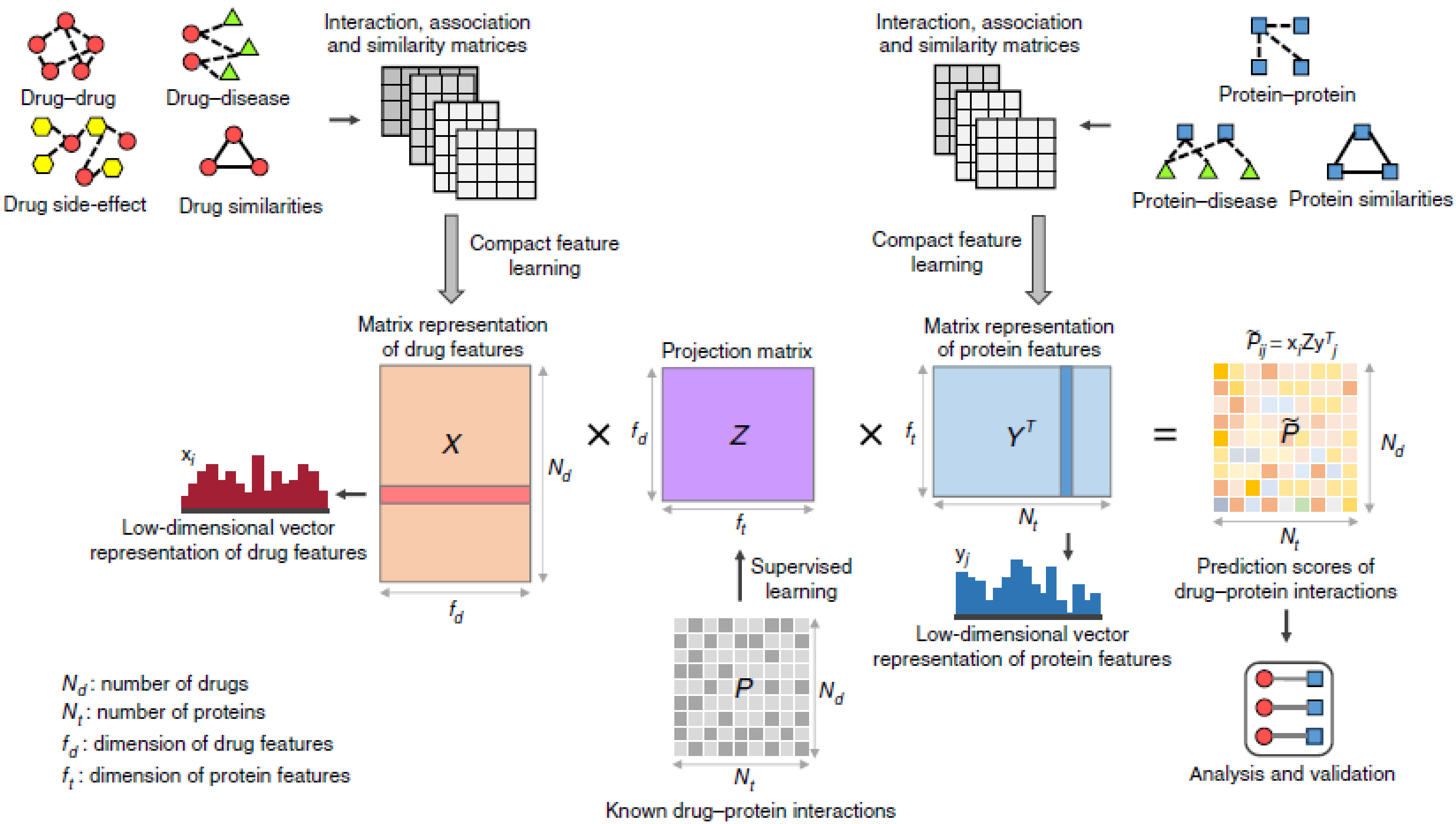


Adapted from "Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. Nature Biotechnology, 2015."

DeepBind总结

- CNN算法自动学习了模版M的参数，这个模版就是能够决定蛋白质是否Binding的特征！
- 而CNN网络就是在前端自动总结出可能的模版，在后端根据这些模版在序列中出现与否而进行判断；
- CNN算法巧妙地将统计归纳和MLP的学习功能相结合，减少了MLP中要学习的参数；

Drug-target interaction prediction

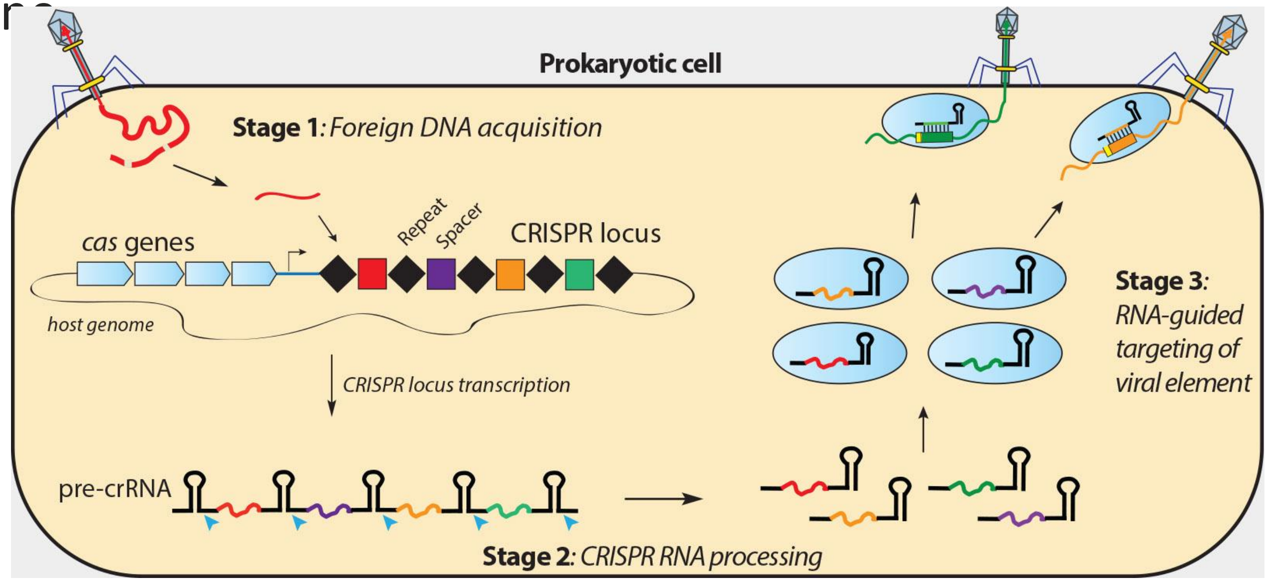


Adapted from "A network integration approach for drug-target interaction prediction and computational drug repositioning from heterogeneous information. Nature Comunications, 2017"

What is Cas?

- Cas (CRISPR-associated system) genes

bacteria immune system: cleavage the foreign gene



CRISPR-Cas system workflow: foreign DNA acquisition→CRISPR RNA processing→RNA-guided targeting of viral element

The existing method for Cas genes prediction

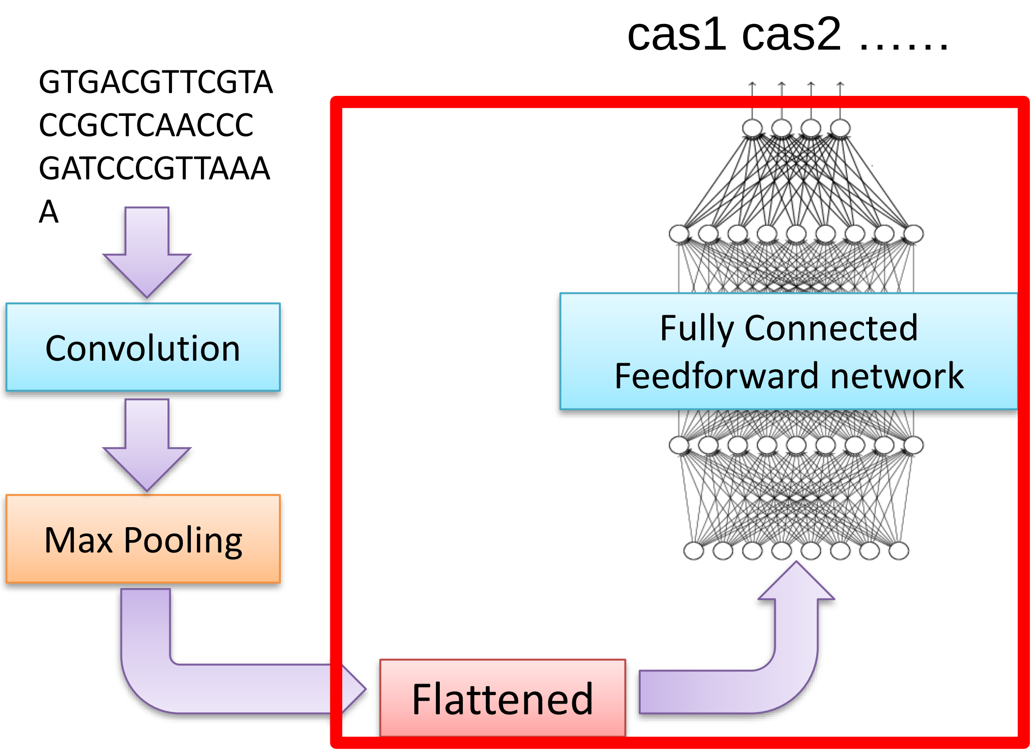
- The existing method for Cas genes prediction

Existing tools	Description
HMMCAS	A web tool for Cas protein identification and domain annotation
CRISPRone	A website tool to predict CRISPR arrays of repeat-spacer units, and cas genes
BLAST	A website tool to find regions of local similarity between sequences
CD-search	A tool allows the conserved domain annotation for large sets of protein queries
Custom HMMs	HMMs deposited in the TIGRFAMs and Pfam protein family databases for known protein families

- Limitations
  - Scan for known Cas proteins.
  - Identify or predict based on the “best hits” of homology searches against existing databases.
  - Work well for detecting known and highly conserved types of Cas genes but may fail to detect novel Cas genes.

The whole neural network to predict Cas genes

Mainly based on Python Tensorflow package



Data preparation

10 core family,75 family,7500 Cas genes in total

Core family	**Structural features**
Cas1	N-terminal β stranded domain and catalytic C-terminal α-helical domain
Cas2	RuvC-like nuclease domains
Cas3	Helicase and HD domain
Cas4	RecB-like nuclease homolog with three-cysteine C-terminal cluster
Cas5	RRM (ferredoxin) fold, RAMP superfamily
Cas6	Double RRM (ferredoxin) fold, RAMP superfamily
Cas7	RRM (ferredoxin) fold with subdomains, RAMP superfamily
Cas8	Subunit of Cascade complex, involved in PAM recognition
Cas9	RuvC-like (RNAse H fold) and HNH nuclease domains
Cas10	Two domains homologous to Palm domain polymerases and cyclases

Data preparation

one-hot encoding, the 4 nucleotides as binary vectors A = [1, 0, 0, 0], T=[0, 1, 0, 0], G=[0, 0, 1, 0] and C=[0, 0, 0, 1].

- Raw data format:

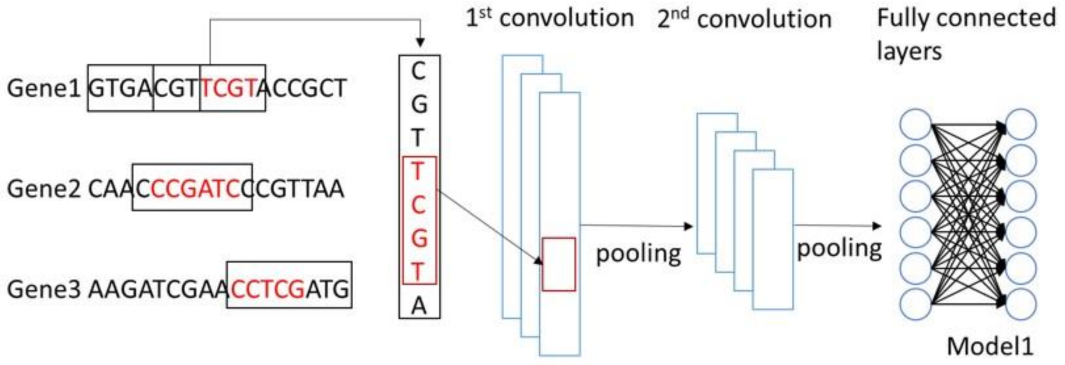
```
>CP001154.1_cds_ACO74068.1_1072
GTGGAGCTGCTGGTCGCCCCGGACCTCAAAACCATCCTGCATTCCGGGCGGGCCAGCCTTCATGGCCTGGAATACTGCCG
>CP010338.1_cds_AMC82843.1_3024
ATGCGTACCGAAGCCATGAACCCGCAACTCATCGAGGCCATAATCGGCTGCCTCTTGCACGACATTGGCAAACCGGTCCA
>CP002567.1_cds_AFI63574.1_1345
ATGAGCAAGATTCCCGGAGTAAAACCACCAGAATTGAAATCGTTGGTGCGGGTGCAGGATAGGCTGACGTTCCTGTATGC
>CP018217.1_cds_APG74590.1_777
ATGTCATGGAGATATGTATTTGTCACTAATCCGTGTAGGATAACGTATACCAACAATAGTATAGTTGTTCAAACTTTCGA
>CP012396.1_cds_AOC85297.1_899
GTGACGTTCGTACCGCTCAACCCGATCCCGTTAAAAGATCGAACCTCGATGATCTTCCTCCAGTACGGTCAAATTGACGT
```

- After change to one_hot:

```
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,1
1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,1
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,1
1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,1
```

Sequence label，save in an individual file

Convolution Neural Networks model



- Design a neural network that will classify cas proteins.
- Given a cas gene sequence, we want our neural net to let us know which cas gene that it belong to.
- The sequence is be made up of A, T, C,G, and fasta format:
- the neural net will have xxx inputs, each one representing a particular sequence and a hidden layer consisting of a number of neurons (more on this later) all feeding their output into just one neuron in the output layer.

A convolutional layer

A CNN is a neural network with some convolutional layers (and some other layers). A convolutional layer has a number of filters that does convolutional operation.

1	0	0	0
0	1	0	0
0	0	0	1
1	0	0	0
⋮	⋮	⋮	⋮
0	0	1	0

80 x 4

These are the network parameters to be learned

1	-1	-1	-1
-1	1	-1	-1
-1	-1	-1	1
1	-1	-1	-1

Filter 1

-1	1	-1	-1
-1	-1	1	-1
-1	-1	-1	1
1	-1	-1	-1

Filter 2

......

Each filter detects a small pattern (4 x 4).

A convolutional layer

stride=1

1	0	0	0
0	1	0	0
0	0	0	1
1	0	0	0
.	.	.	.
.	.	.	.
0	0	1	0

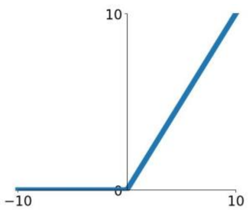
80 x 4

Dot product

4

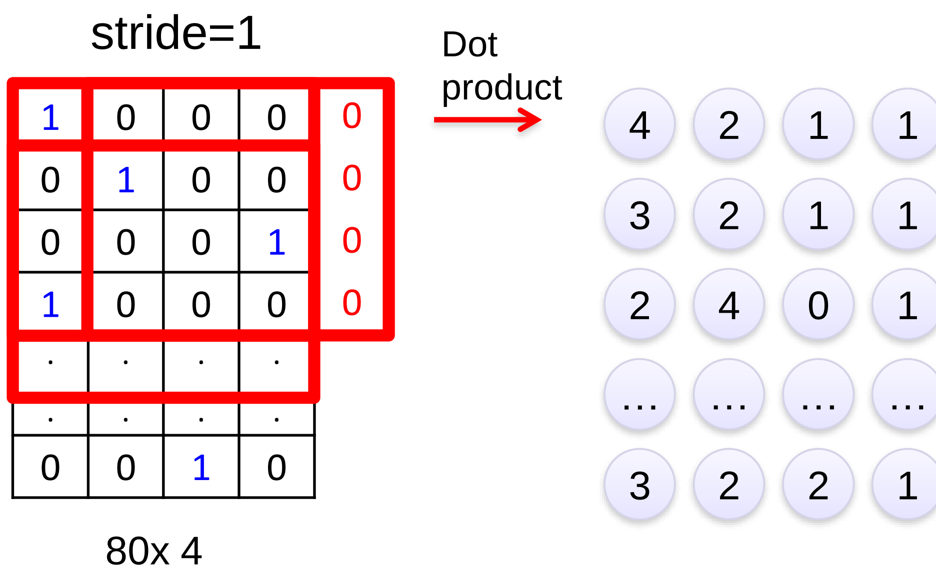
1	-1	-1	-1
-1	1	-1	-1
-1	-1	-1	1
1	-1	-1	-1

Filter 1



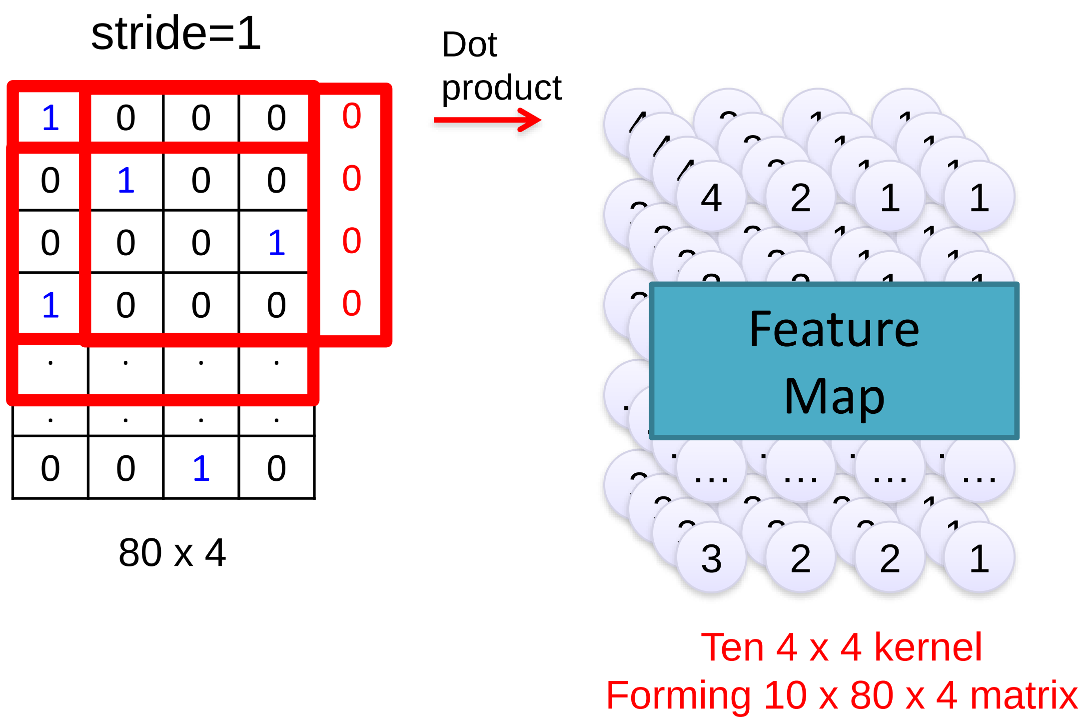
$$\mathrm{ReLU}(x) = \max(0, x)$$

A convolutional layer



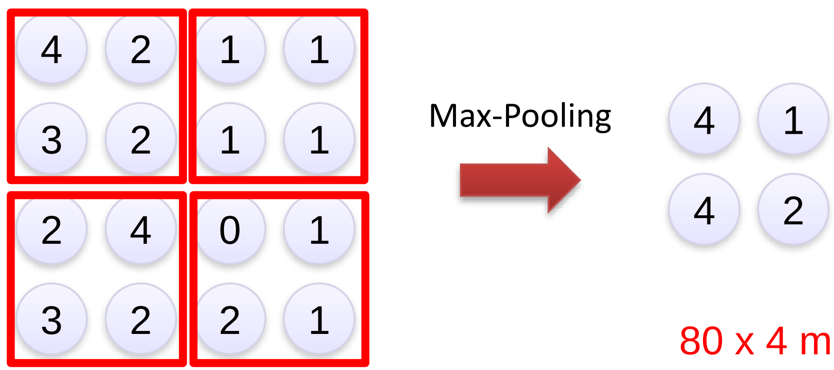
Repeat this for each filter

A convolutional layer



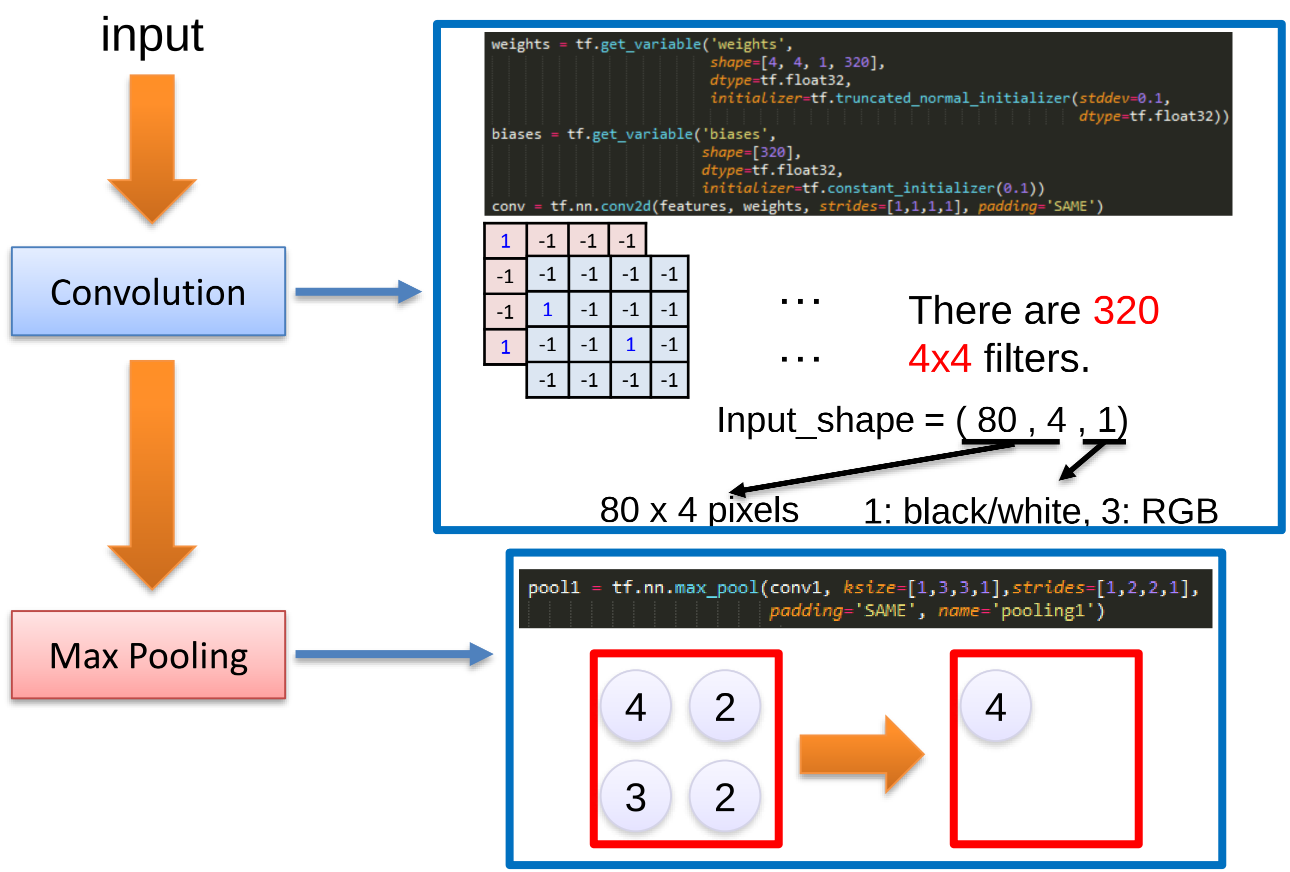
A convolutional layer

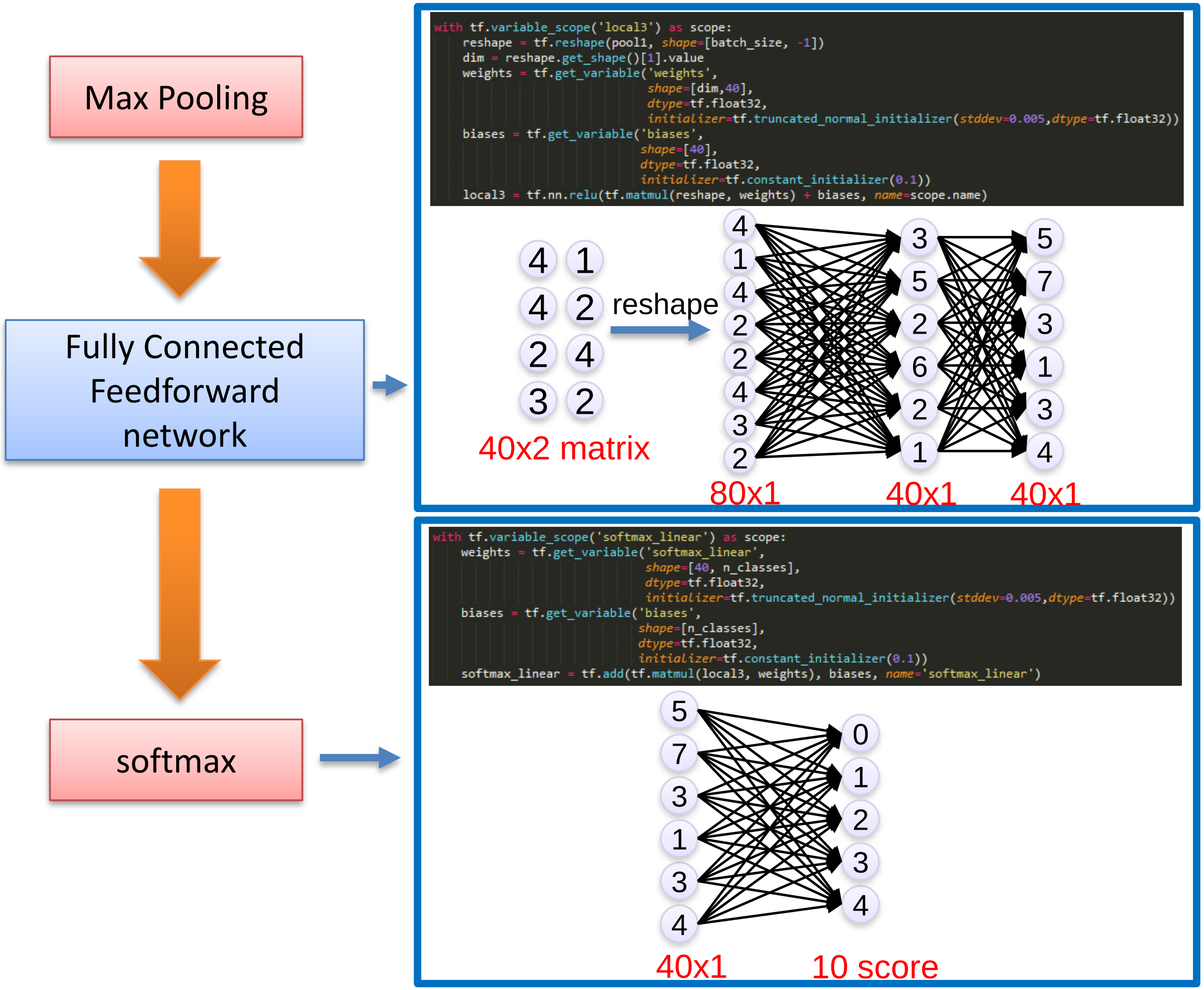
$$p_{nf\,i} = \max_{|k|<P/2}\left(a_{nf,i+k}\right)$$



80 x 4 matrix
Forming 40 x 2 matrix

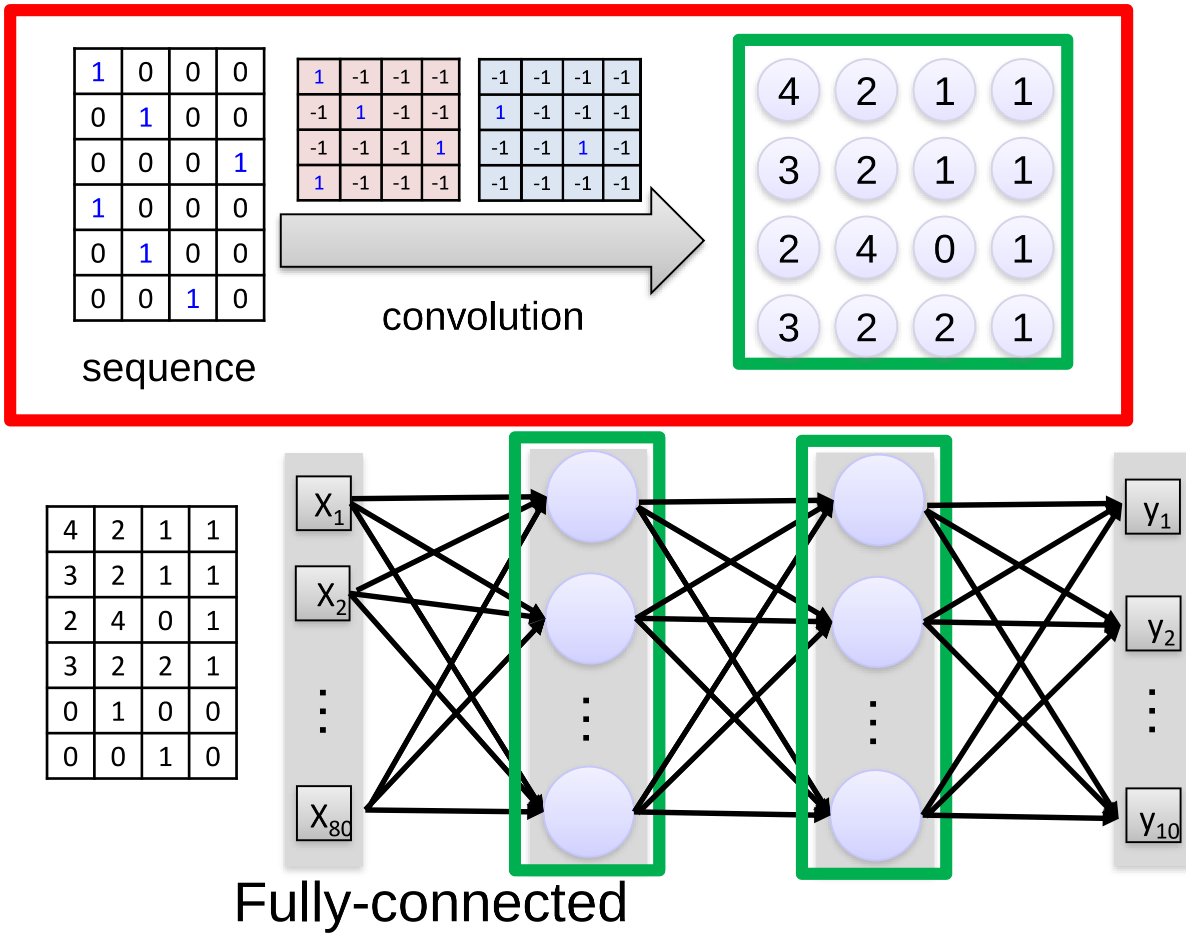
***CNN model construction in* Tensorflow**





Max Pooling
Fully Connected Feedforward network
softmax
with tf.variable_scope('local3') as scope:
    reshape = tf.reshape(pool1, shape=[batch_size, -1])
    dim = reshape.get_shape()[1].value
    weights = tf.get_variable('weights',
                              shape=[dim,40],
                              dtype=tf.float32,
                              initializer=tf.truncated_normal_initializer(stddev=0.005,dtype=tf.float32))
    biases = tf.get_variable('biases',
                             shape=[40],
                             dtype=tf.float32,
                             initializer=tf.constant_initializer(0.1))
    local3 = tf.nn.relu(tf.matmul(reshape, weights) + biases, name=scope.name)
reshape
40x2 matrix
80x1
40x1
40x1
with tf.variable_scope('softmax_linear') as scope:
    weights = tf.get_variable('softmax_linear',
                              shape=[40, n_classes],
                              dtype=tf.float32,
                              initializer=tf.truncated_normal_initializer(stddev=0.005,dtype=tf.float32))
    biases = tf.get_variable('biases',
                             shape=[n_classes],
                             dtype=tf.float32,
                             initializer=tf.constant_initializer(0.1))
    softmax_linear = tf.add(tf.matmul(local3, weights), biases, name='softmax_linear')
40x1
10 score

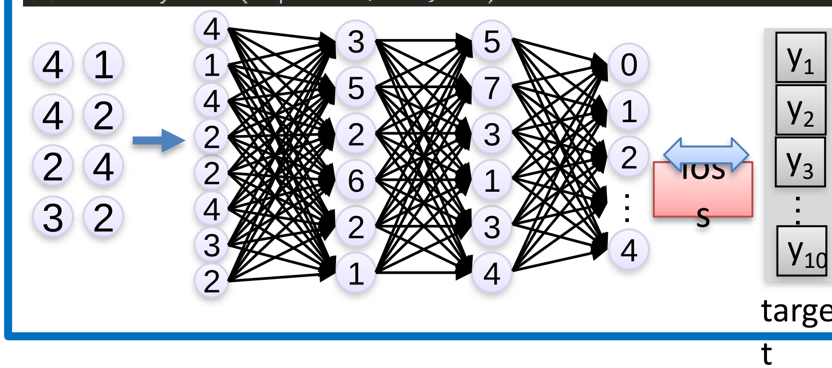
Convolution v.s. Fully Connected



Pick the best model

loss

```
with tf.variable_scope('loss') as scope:
    cross_entropy = tf.nn.sparse_softmax_cross_entropy_with_logits(logits=logits,
      labels=labels, name='xentropy_per_example')
    loss = tf.reduce_mean(cross_entropy, name='loss')
    tf.summary.scalar(scope.name+'/loss', loss)
```



Randomly initialize network parameters

Pick the 1st batch

$L' = l1 + l2 + \cdots$

Update parameters once

Pick the 2nd batch

$L'' = l1 + l2 + \cdots$

Update parameters once

…

Until all mini-batches have been picked

training

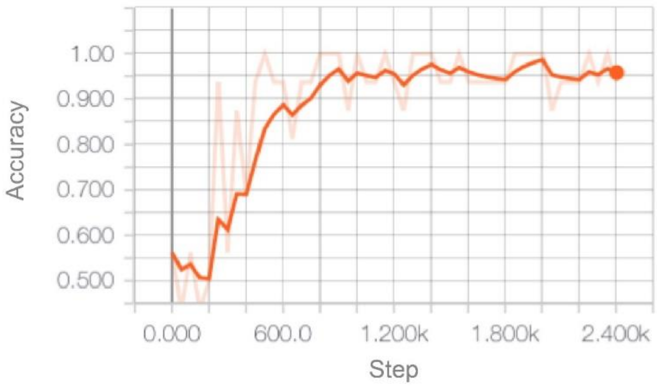
```
N_CLASSES = 2
seq_W = 20  # resize the seq, if the input seq is too large, training will be very slow.
seq_H = 80
BATCH_SIZE = 16
CAPACITY = 2000
MAX_STEP = 10000 # with current parameters, it is suggested to use MAX_STEP>10k
learning_rate = 0.0001 # with current parameters, it is suggested to use learning rate<0.0001
n = 1601
```

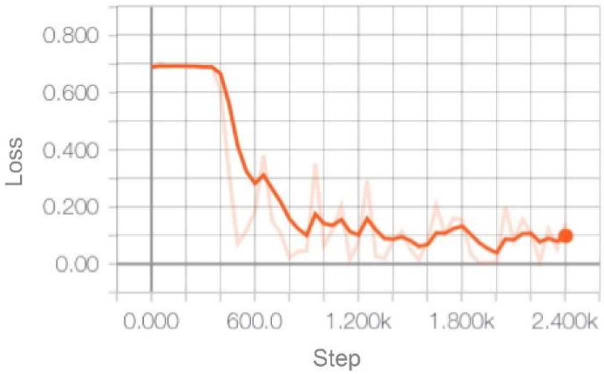
Output:

```
Step 0, train loss = 0.69, train accuracy = 62.50%
Step 50, train loss = 0.69, train accuracy = 56.25%
Step 100, train loss = 0.69, train accuracy = 43.75%
Step 150, train loss = 0.69, train accuracy = 56.25%
Step 200, train loss = 0.68, train accuracy = 93.75%
Step 250, train loss = 0.68, train accuracy = 43.75%
Step 300, train loss = 0.57, train accuracy = 68.75%
Step 350, train loss = 0.45, train accuracy = 68.75%
Step 400, train loss = 0.36, train accuracy = 75.00%
Step 450, train loss = 0.41, train accuracy = 87.50%
```

Training Module1

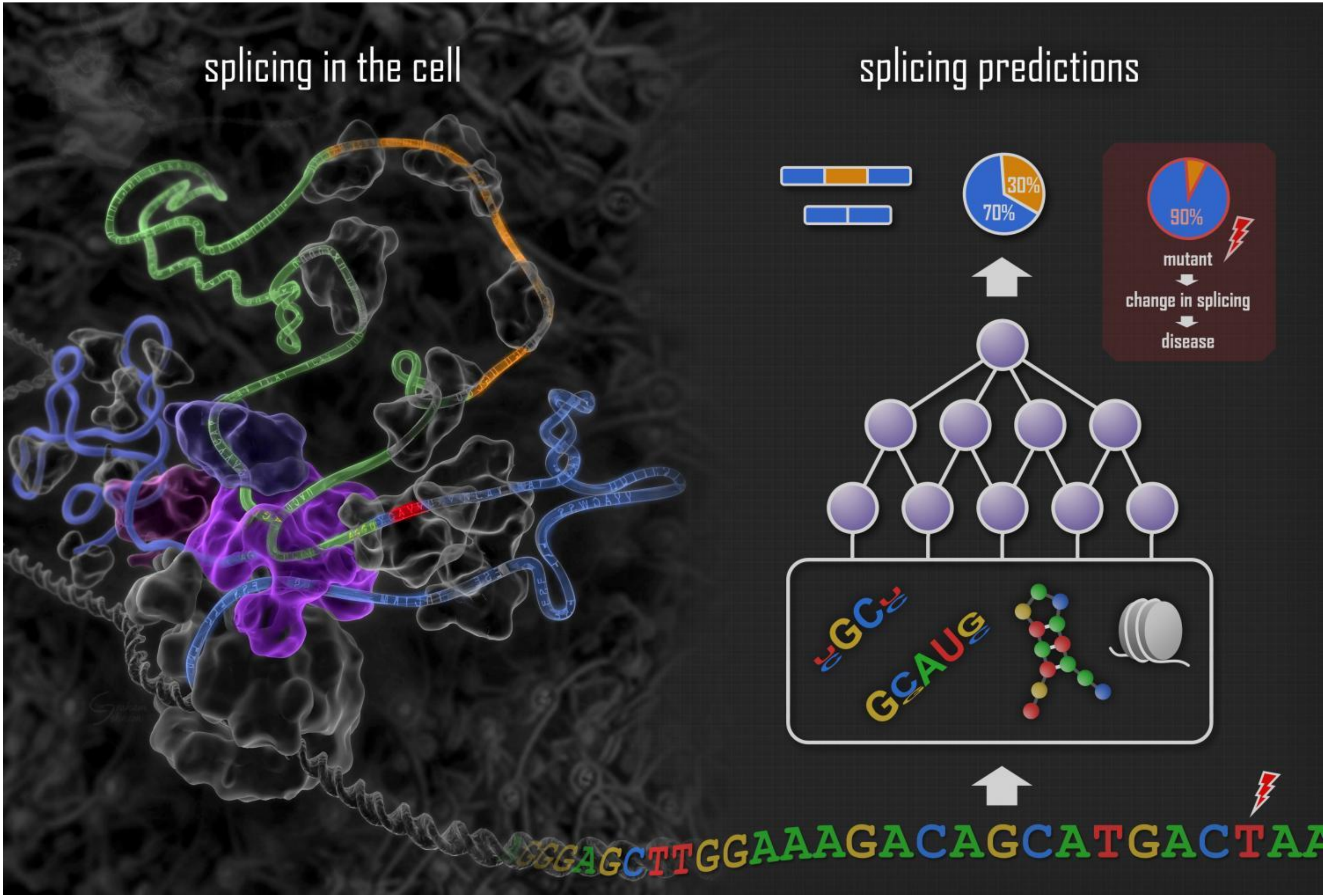
- Data: 5000 Cas9 sequences with domain sites and 5000 without domain sites



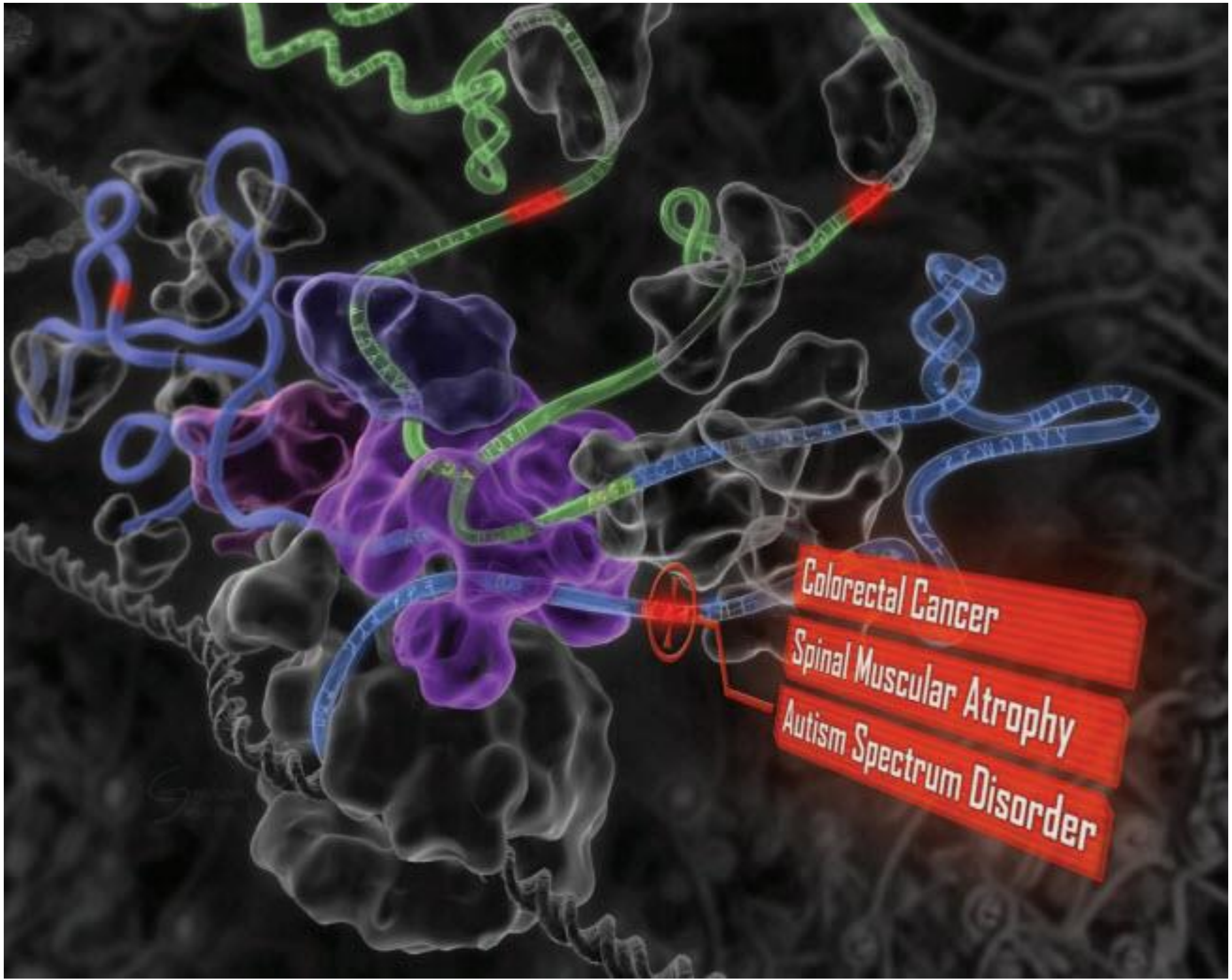


Accuracy： 0.80225

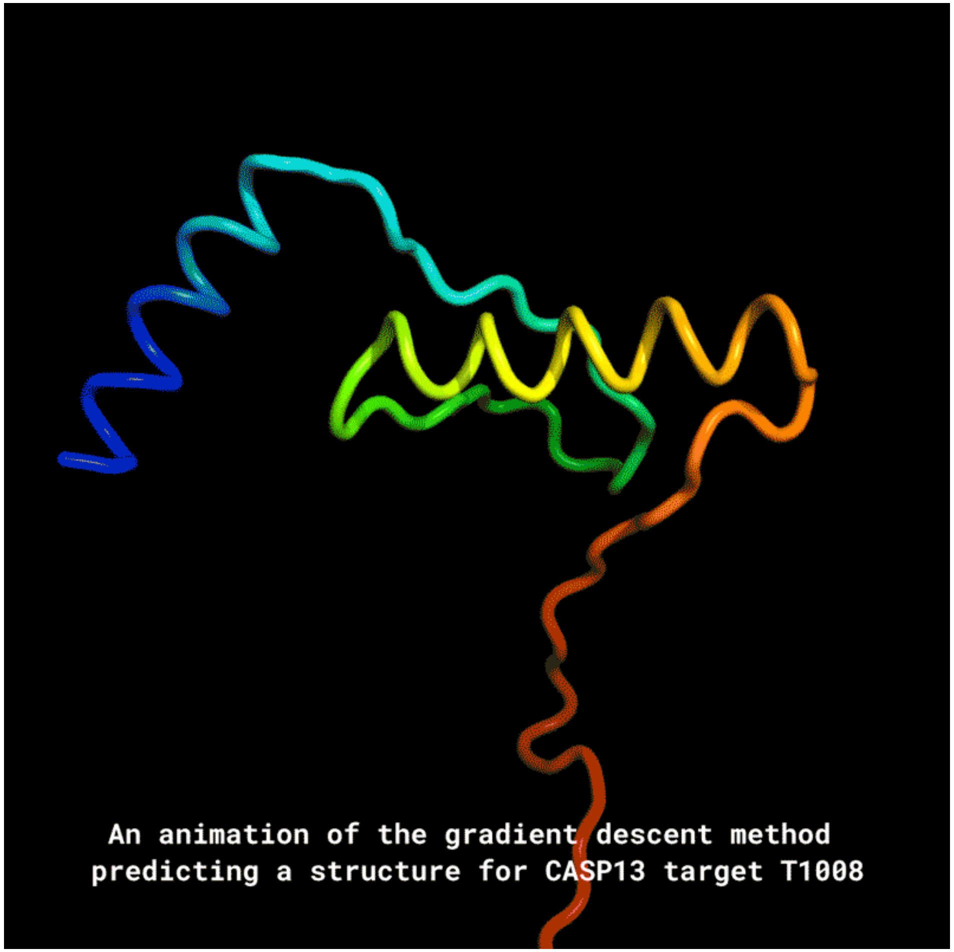
New variant discovery



New variant discovery



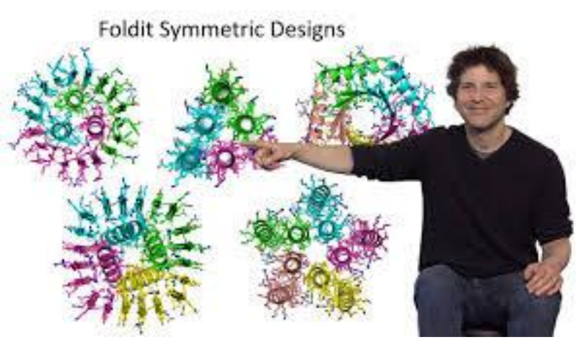
CASP
(Critical Assessment of Techniques for Protein Structure Prediction)



CASP
(Critical Assessment of Techniques for Protein Structure Prediction)

Rosetta Commons
The hub for Rosetta modeling software

I-TASSER
Protein Structure & Function Predictions



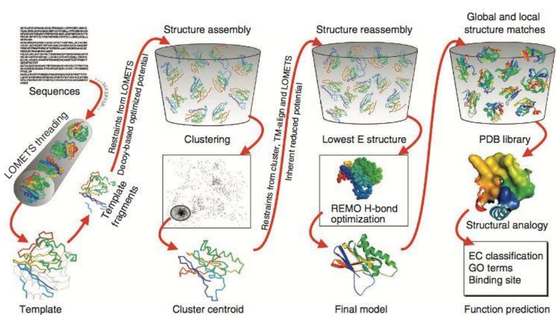
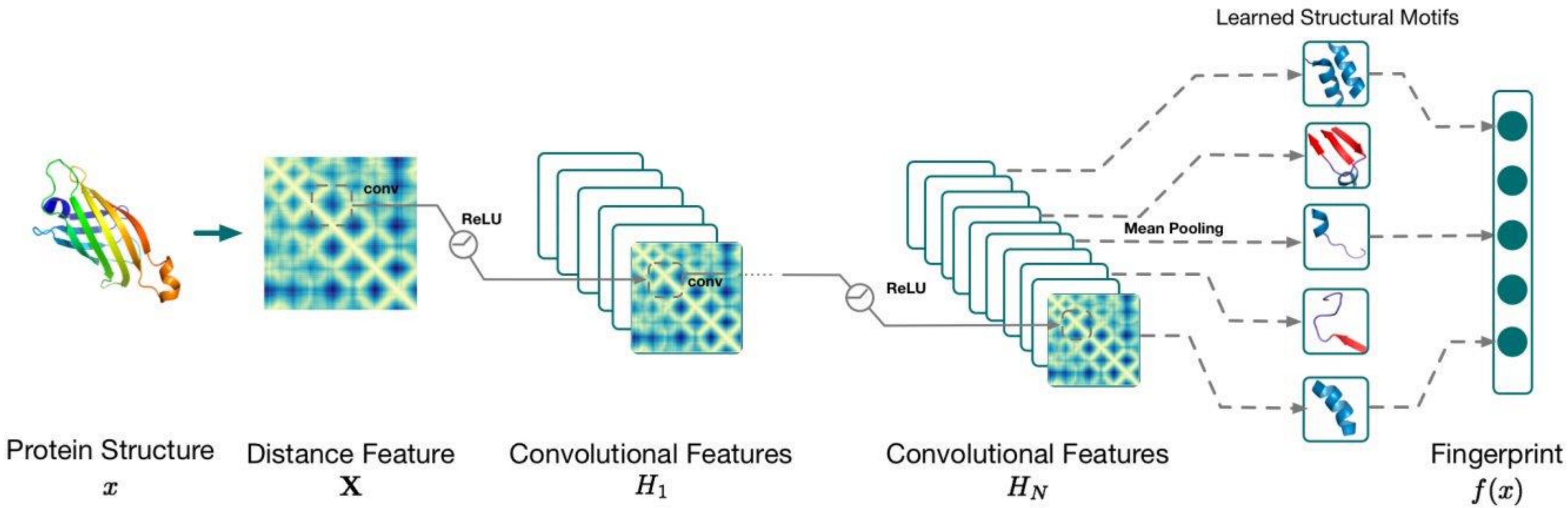


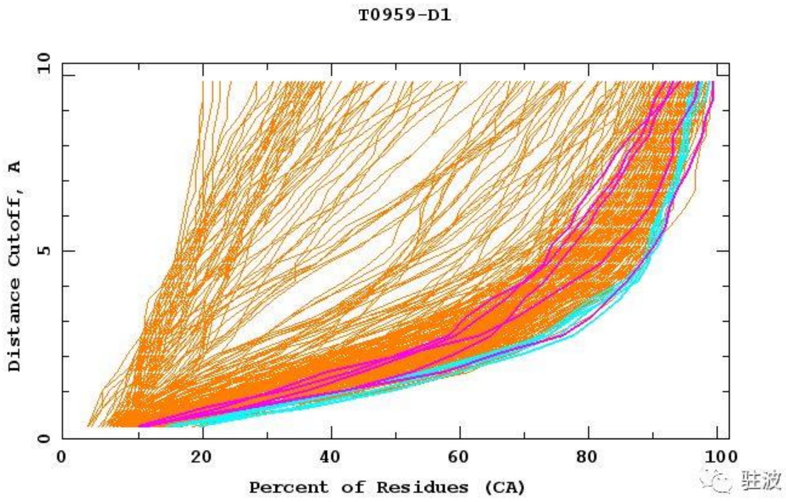
Figure 1 | A schematic representation of the I-TASSER protocol for protein structure and function predictions. The protein chains are colored from blue at the N-terminus to red at the C-terminus.

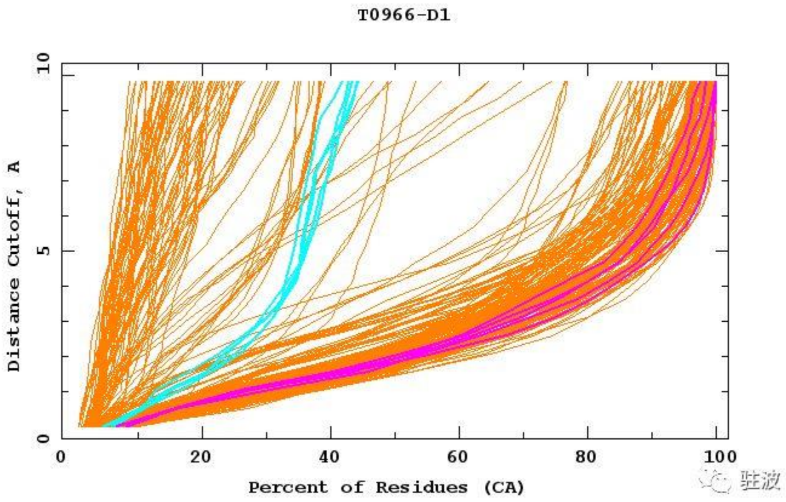
CASP
(Critical Assessment of Techniques for Protein Structure Prediction)

DeepFold: overall winner of CASP18



CASP
(Critical Assessment of Techniques for Protein Structure Prediction)





Recap (知识点总结)

- 深度学习：
  - 概念和区别
  - 模型和方法
  - 应用领域

- 生物大数据的深度学习：
  - 序列和特征
  - 特征学习和模型构建
  - 应用案例

References

- Angermueller, C., et al., Deep learning for computational biology. Molecular systems biology, 2016. 12(7): p. 878.
- Jurtz, V.I., et al., An introduction to deep learning on biological sequence data: examples and solutions. Bioinformatics, 2017. 33(22): p. 3685-3690.
- Wang, S., et al., Protein secondary structure prediction using deep convolutional neural fields. Scientific reports, 2016. 6.
- Angermueller, C., et al., DeepCpG: accurate prediction of single-cell DNA methylation states using deep learning. Genome biology, 2017. 18(1): p. 67.
- Yuan, L., et al., Applications of Deep Learning in Biological and Medical Data Analysis. PROGRESS IN BIOCHEMISTRY AND BIOPHYSICS, 2016. 43(5): p. 472-483.
- Chai, G., et al., HMMCAS: a web tool for the identification and domain annotations of Cas proteins. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2017.

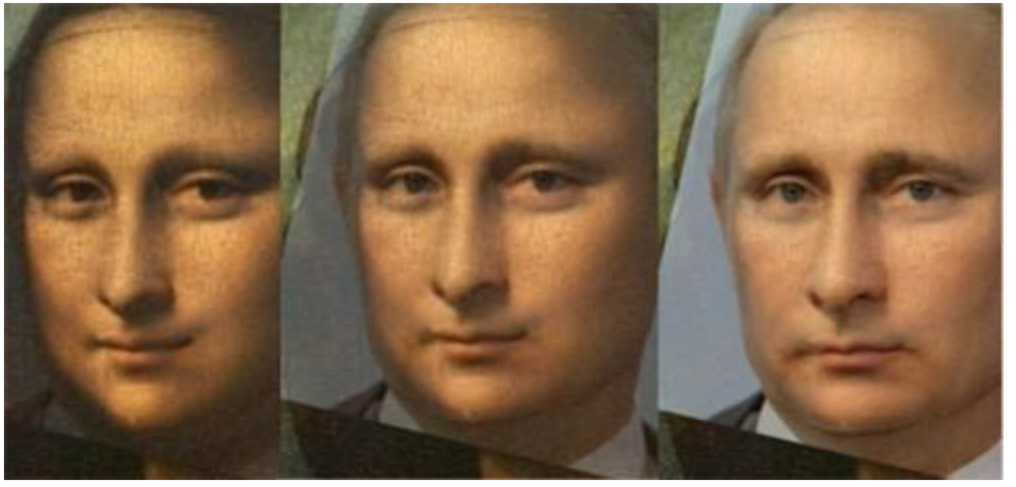
深度学习的问题。。。

Feature extraction? Or D&C

Case 1

为何对基因的分析结果难以达到共识？“其中一个原因是遗传变异太常见。一个典型的基因包含了数以百计的变异，有些变异可能对健康没有任何影响。有些疾病可能是多个基因变异相互作用的结果，这种作用可能经常被算法所遗漏”，威康信托基金会桑格研究所遗传学家 Matthew Hurles说。 Hurles目前正引领一个关于破解发育障碍的项目，该项目分析了1400个家庭中未确诊重症儿的外显子。他说，“即使当一个单一变异能解释疾病，也还需要进行统计调查、分析数据以及开展临床试验，以提供最终的确诊。”

深度学习的问题。。。



Majority vote? Or rely on expert?

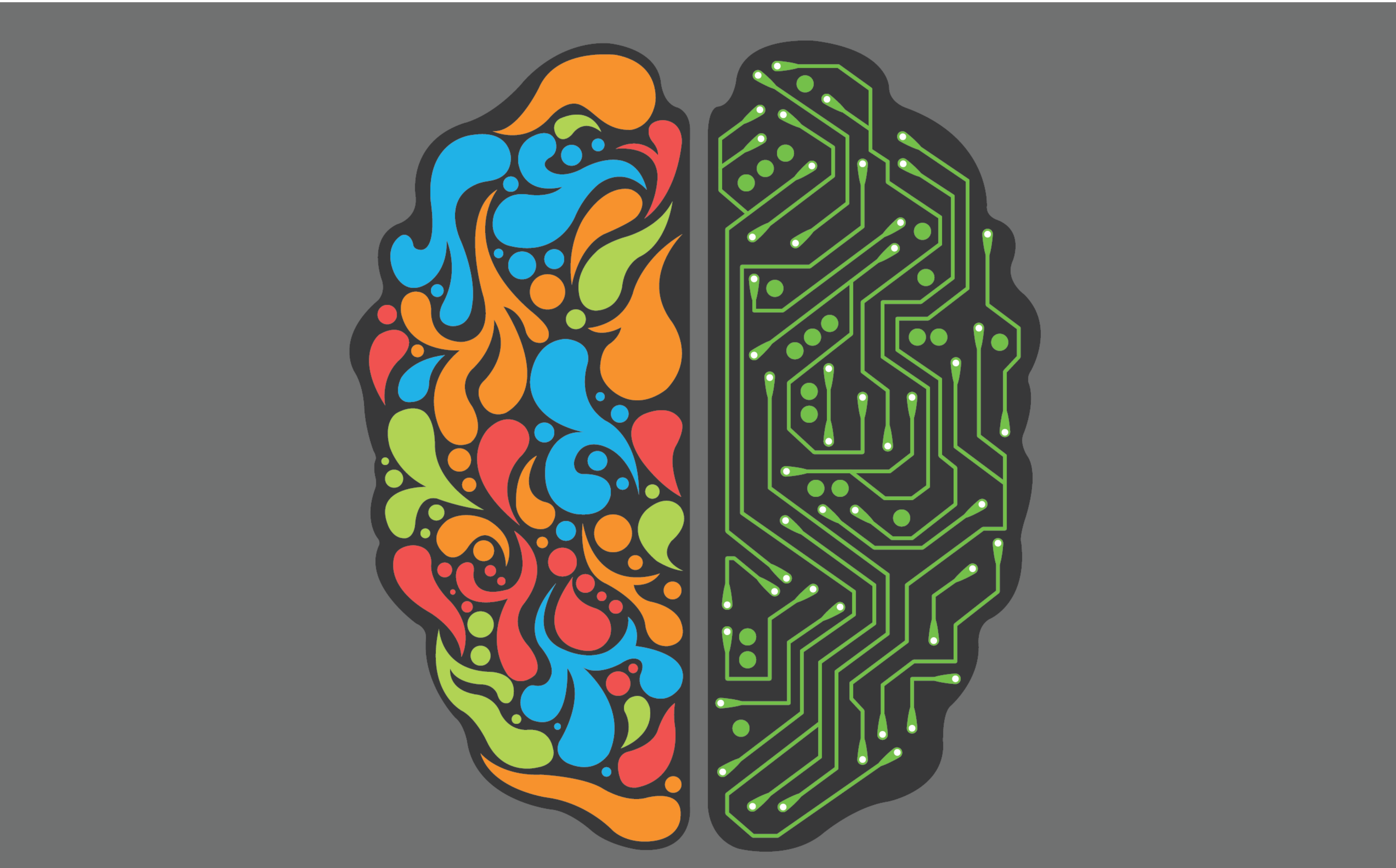
Case 1

一名40岁女子从小被诊断为重型地中海贫血及血小板过多症，后来并被切除脾臟及长期服用抑制血小板功能的药物，进一步做全外显子基因体定序，才发现真正的致病机转是因细胞内质网相关的基因突变引起「先天性红血球生成异常性贫血」，输血治疗地中海贫血反受其害。

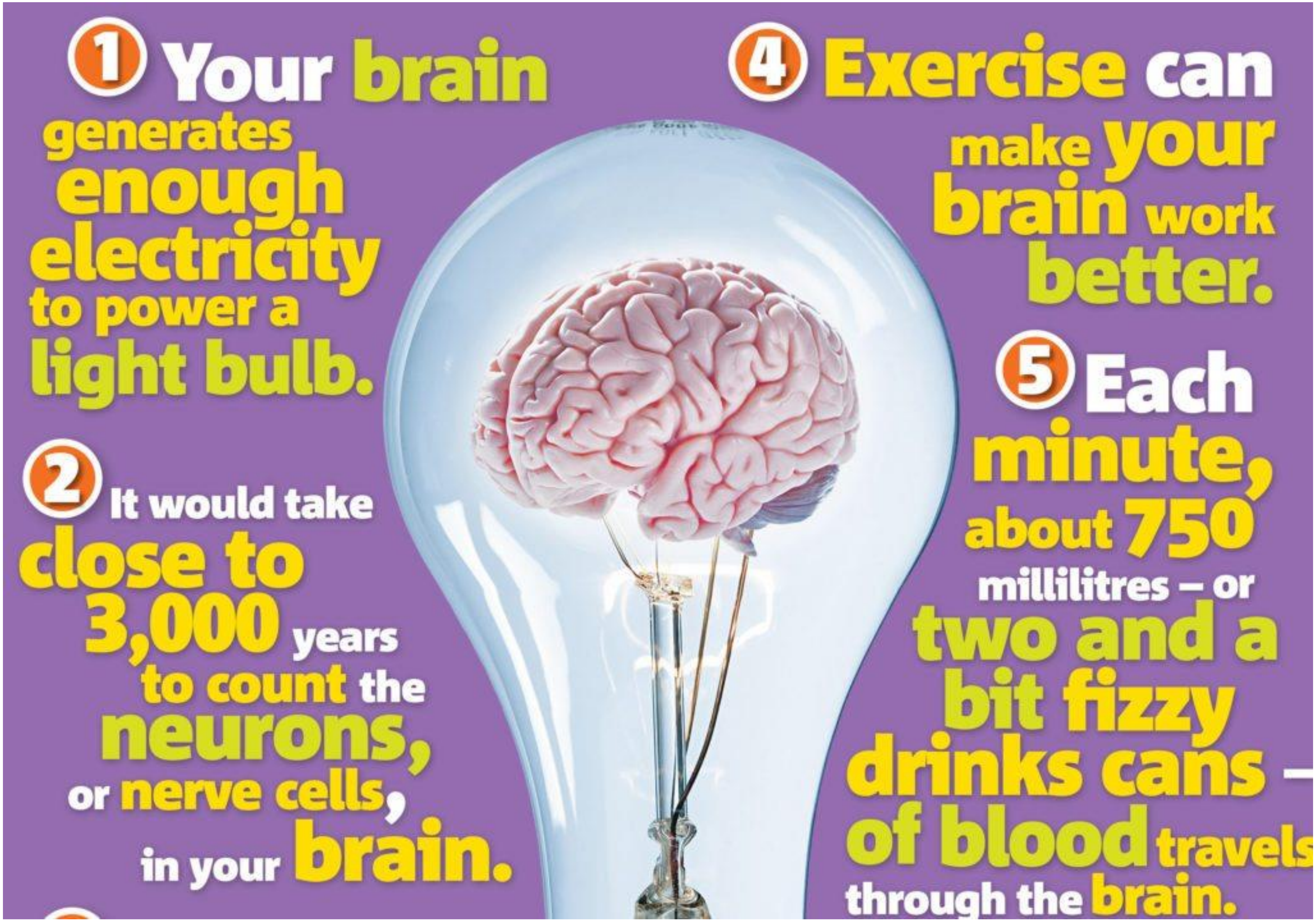
Case 2

2007年，美国最大的分子诊断公司Quest Diagnostics子公司——Athena Diagnostics 给患儿Christian Millare进行了SCN1A基因突变检测。2008年5月1日，两岁的Christian疾病发作并不幸去世。此后死亡患儿母亲Amy Williams 通过咨询专家和查阅发表的文献，再结合Christian的病历，她坚信儿子的生命不应该这样过早的结束。8年后，Williams对Quest 诊断公司和Athena诊断公司提出了诉讼。

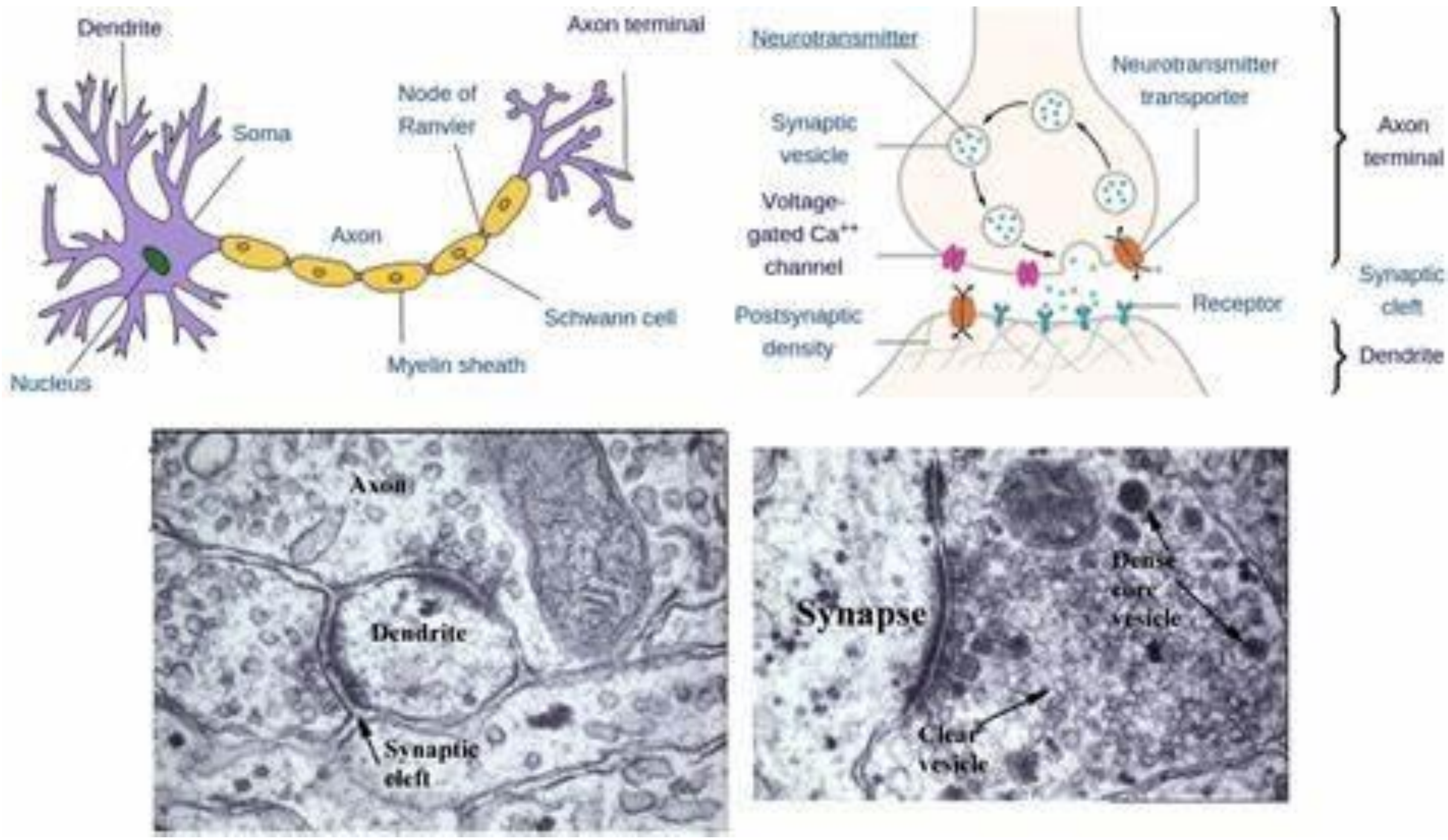
深度学习的问题：脑科学。。。



深度学习的问题：脑科学。。。


1 Your brain generates enough electricity to power a light bulb.
2 It would take close to 3,000 years to count the neurons, or nerve cells, in your brain.
4 Exercise can make your brain work better.
5 Each minute, about 750 millilitres – or two and a bit fizzy drinks cans – of blood travels through the brain.

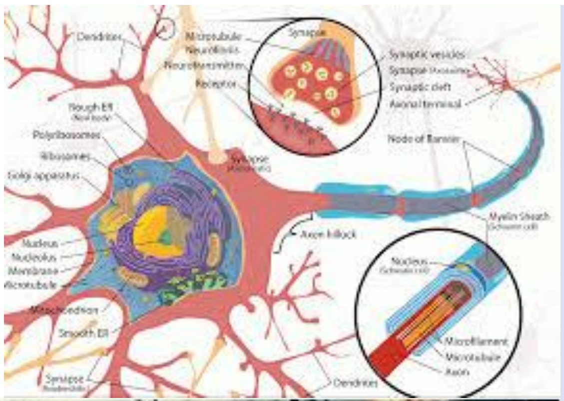
深度学习的问题：脑科学。。。

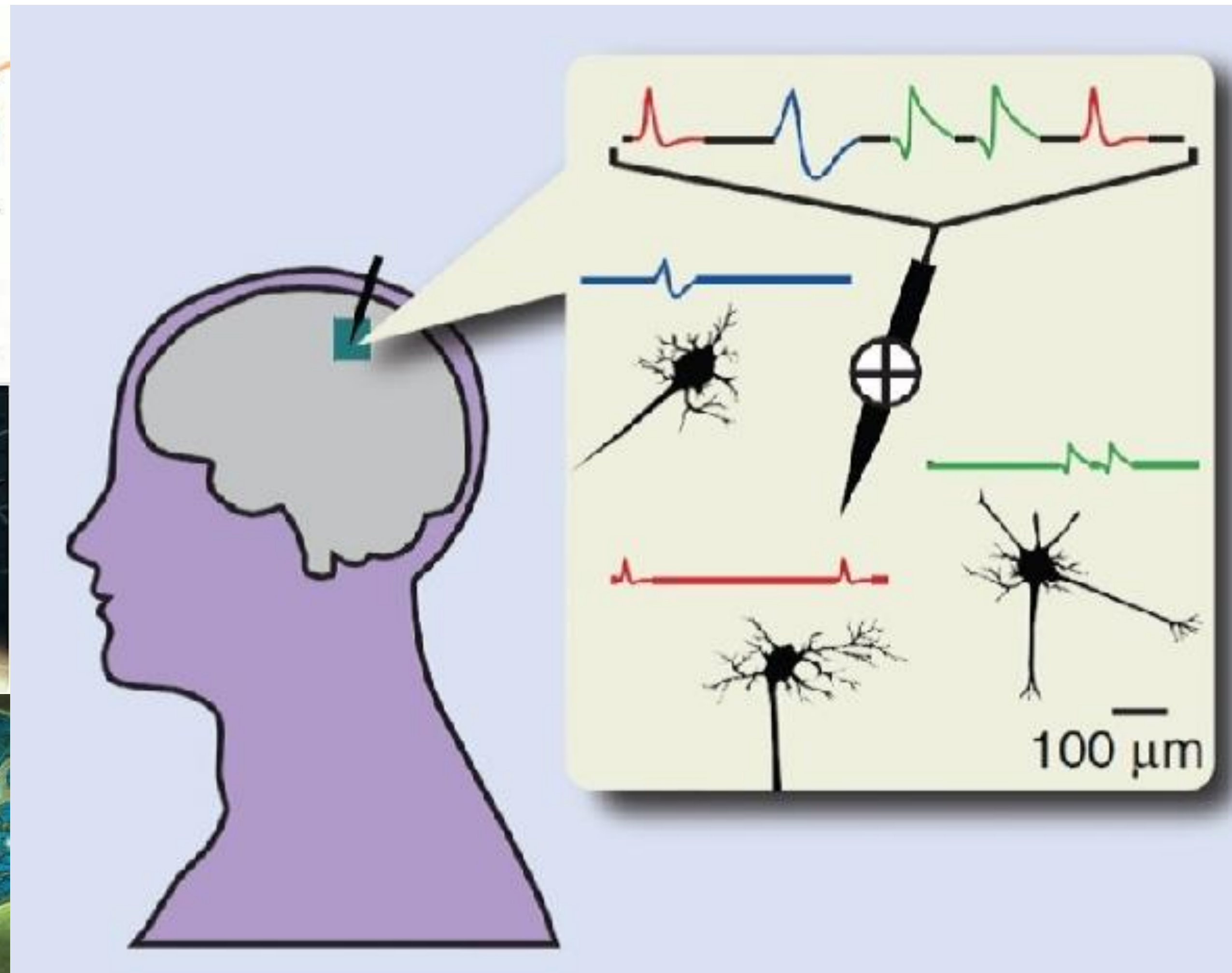


The Brain vs. Deep Learning vs. Singularity

深度学习的问题：脑科学。。。

We are actually HMM (or Deep learning) animal...





深度学习的问题。。。

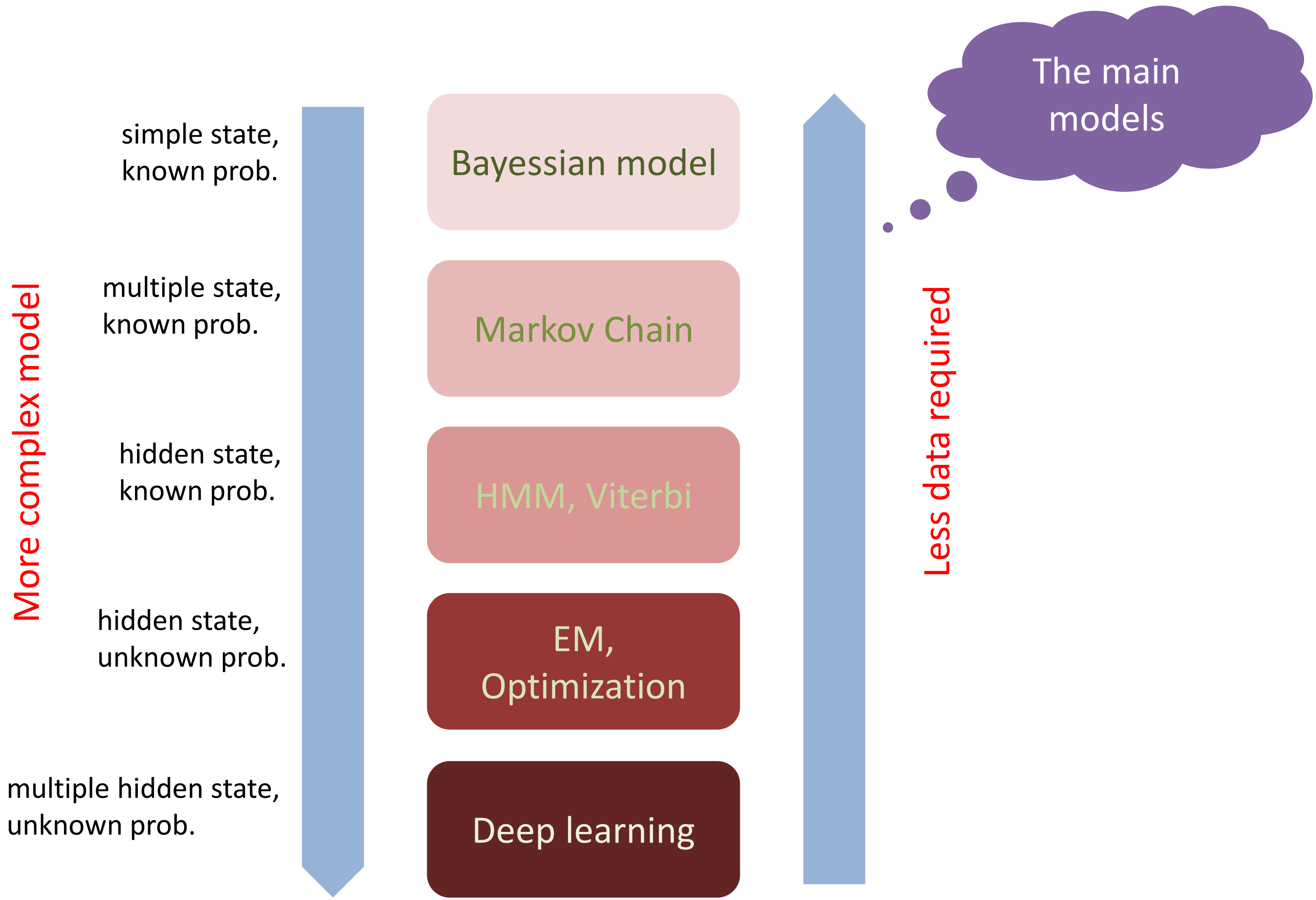
More problems that you can think of?

不建议用简单的深度学习来解决的问题。。。

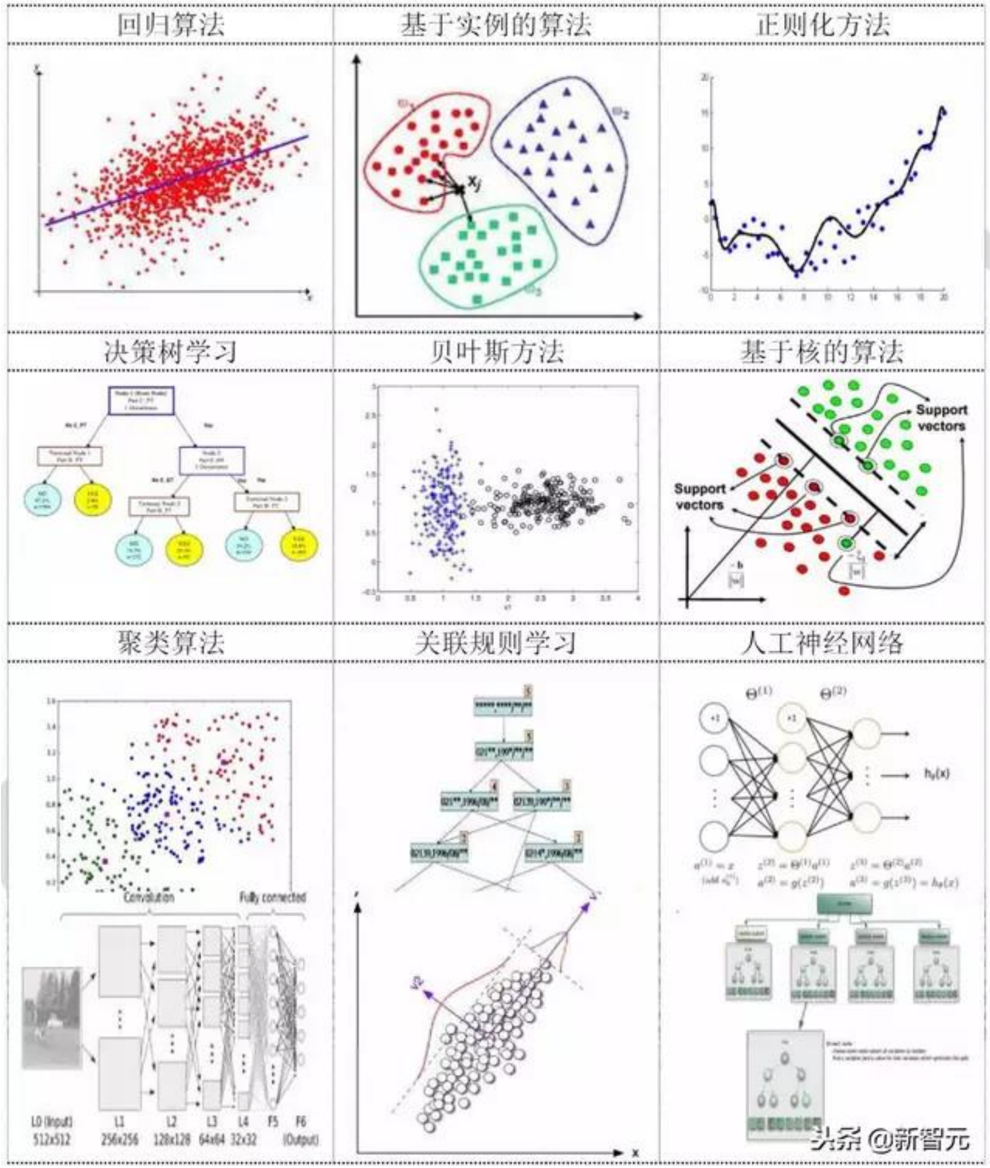
Statistical modeling

Distribution, Modeling, Prediction

Statistical modeling



Statistical modeling



课程安排

- 生物背景和课程简介
- 传统生物统计学及其应用
- 生物统计学和生物大数据挖掘
  - Hidden Markov Model (HMM)及其应用
    - Markov Chain
    - HMM理论
    - HMM和基因识别 (Topic I)
    - HMM和序列比对 (Topic II)
  - 进化树的概率模型 (Topic III)
  - Motif finding中的概率模型 (Topic IV)
    - EM algorithm
    - Markov Chain Monte Carlo (MCMC)
  - 基因表达数据分析 (Topic V)
    - 聚类分析-Mixture model
    - Classification-Lasso Based variable selection
  - 基因网络推断 (Topic VI)
    - Bayesian网络
    - Gaussian Graphical Model
  - 基因网络分析 (Topic VII)
    - Network clustering
    - Network Motif
    - Markov random field (MRF)
  - Dimension reduction及其应用 (Topic VIII)
- 面向生物大数据挖掘的深度学习

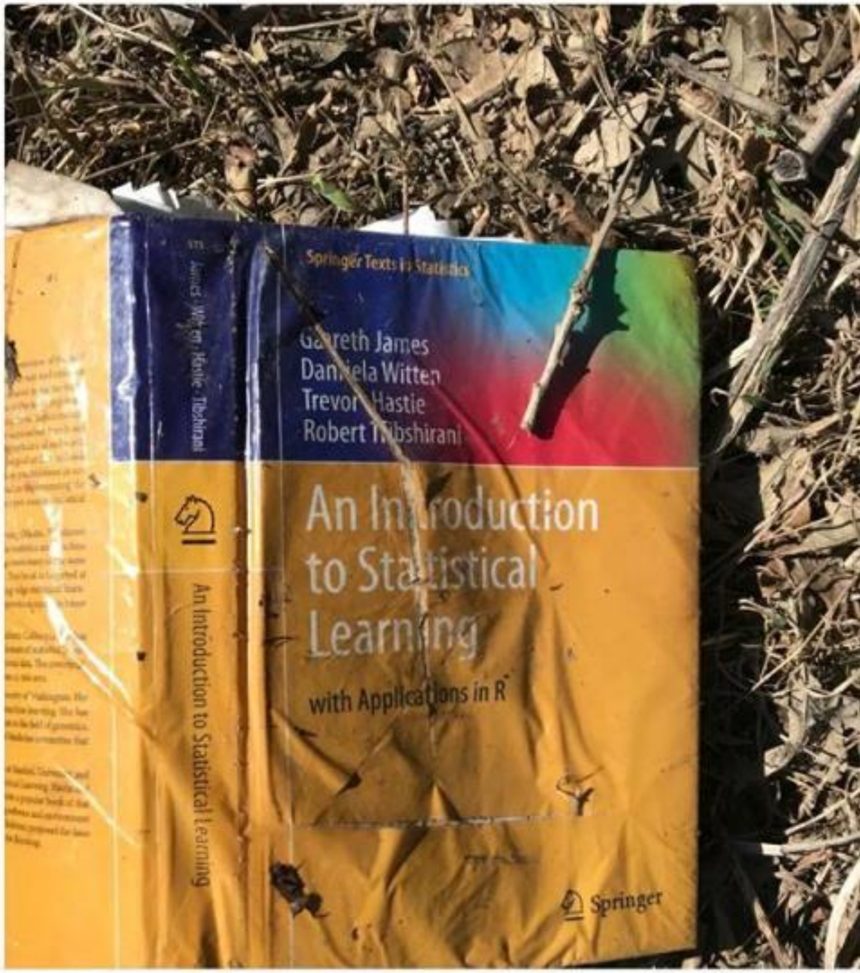
研究对象:
生物序列,
进化树,
生物网络,
基因表达
...

方法:
生物计算与生物统计

Good luck in biostatistics!

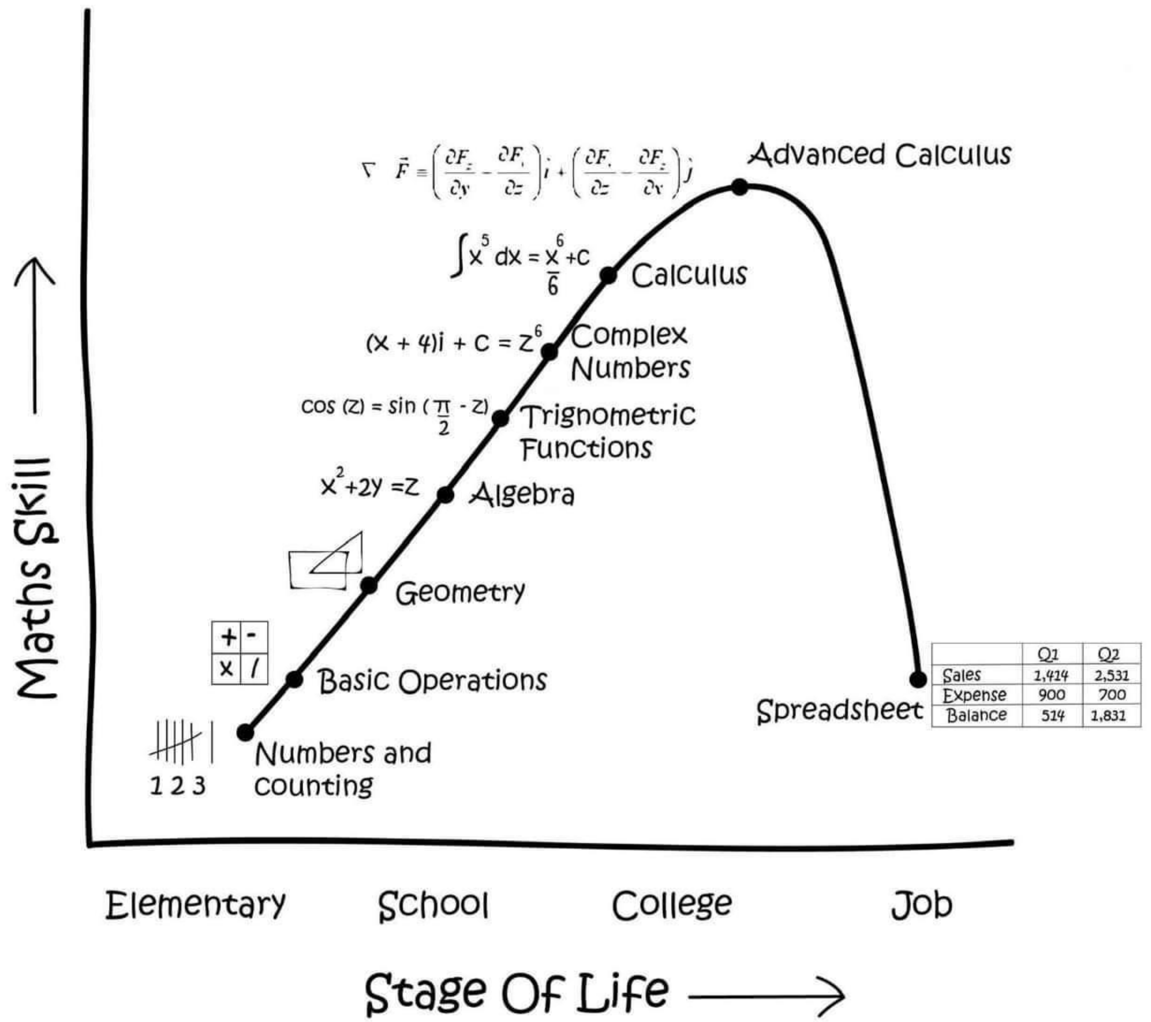
Noah Williams
@Bellmanequation

Found on the roadside:
@SpringerStats yellow book, beside empty vodka bottles & cigarette packs

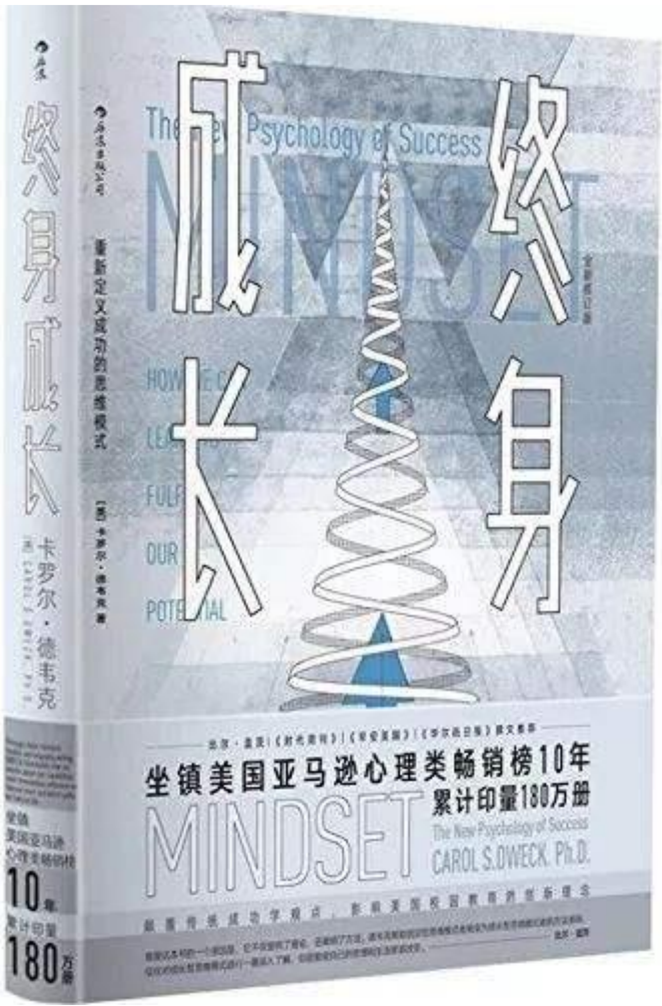


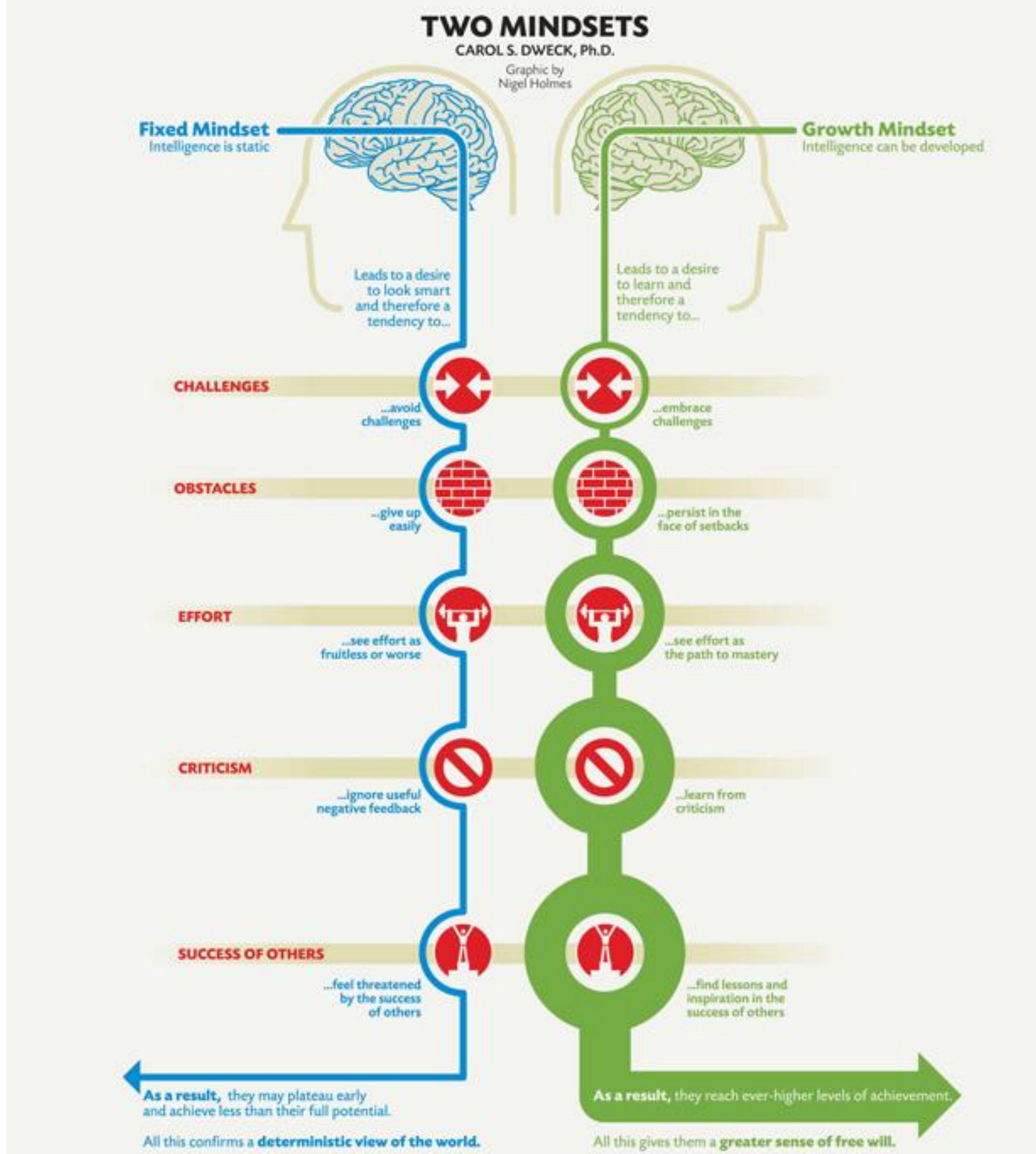
$p<0.05$

Maybe you will not face these many statistical problems...



But you have the knowledge





Statistical modeling

More to learn:

- Fast Fourier Transformation: https://zhuanlan.zhihu.com/p/19763358

- Causality analysis: https://zhuanlan.zhihu.com/p/33860572

- …

Statistical modeling

More to learn:

- 生物统计与生物信息的区别与联系？（https://www.zhihu.com/question/30284194）
  - 生物学的生物信息学：首先是讲进化，做序列比对，blast，构建进化树；然后，讲基因功能，基因功能富集分析等。侧重的是，将进化中的生物学原理，参数如何设置，软件如何使用，如何将这些软件应用到生物学问题。
  - 计算机系的生物信息学：首先讲的也是进化，但是，讲的是算法设计；然后讲了很多马尔科夫链，HMM模型，序列比对中的blosum62矩阵是怎么来的，如何加快计算效率，如何降低空间存储等等。侧重的是，如何设计合理的算法，给生物学的人使用。
  - 数学系的系统生物学：讲了回归中的penalty function，然后我学了LASSO回归；后来学了SVM中的核函数；还有Gibbs samping，MCMC；还有时间序列中的Granger因果推断等等。
  - 生统对数学要求更高，生信基本不需要。国内生统大多是挂XXX卖XXX

- Yoshua Bengio:
  - **你死我活那是邪教，开放包容才是正道。**

- What do you think?

Biostatistics in the market...

[公司文件] 加强与国内大学合作，吸纳全球优秀人才，共同推动中国基础研究——任总与中国科学技术大学包信和校长座谈的讲话 荐

2018-12-13 17:59 6799 113 只看楼主

总 裁 办 电 子 邮 件

电邮讲话【2018】128号 签发人：任正非

加强与国内大学合作，吸纳全球优秀人才，共同推动中国基础研究

——任总与中国科学技术大学包信和校长座谈的讲话

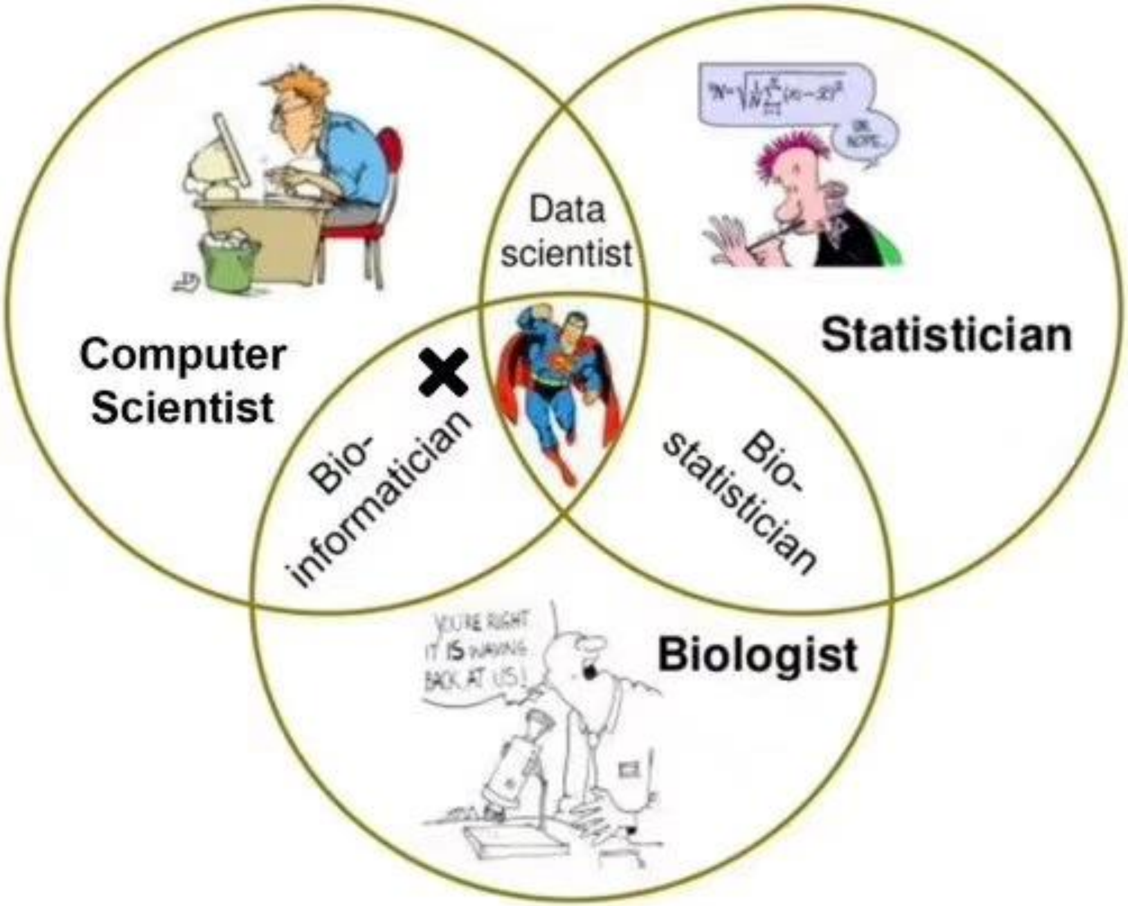
2018年11月19日

在高校学科设置上，我特别支持你们重视**统计学**。计算机科学不仅仅是技术，还应该以**统计学**为基础。大数据需要**统计学**，信息科学需要**统计学**，**生命科学也需要统计学**。国家要搞人工智能，更要重视**统计学**。**统计学**不是一个纯粹的学科，而是每一个学科都要以**统计学**为基础。

Or

Good luck in final exam!

Superman/Wonder woman



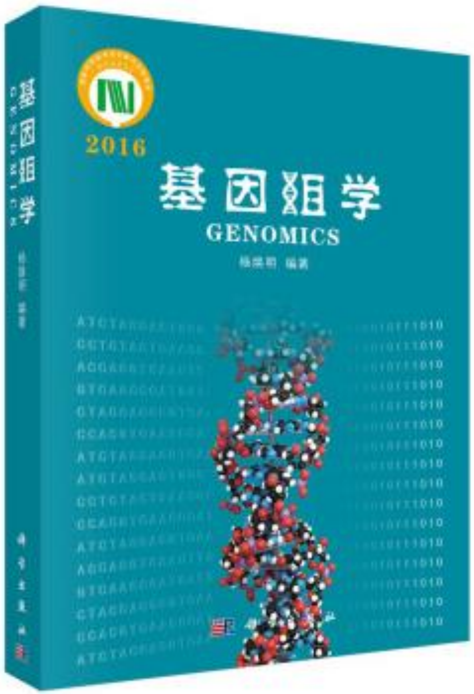
Good luck in your life!

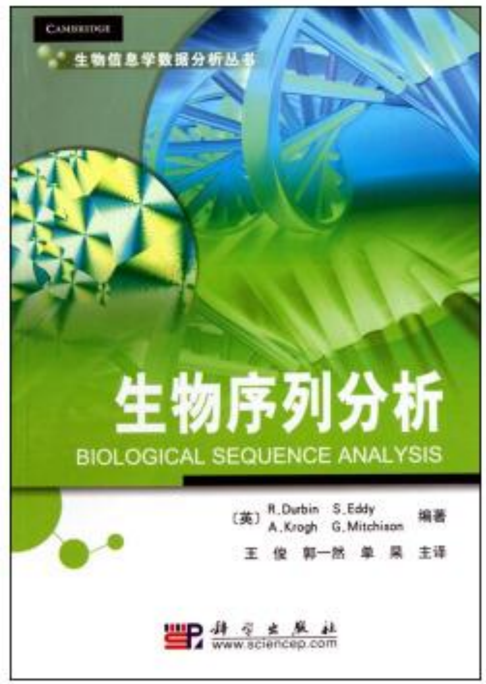
谢谢大家！

OTHER WAYS TO SAY
“THANK YOU”
I thank you from the bottom of my heart
That's so kind of you
I am very thankful
You are great
I'm really grateful
Thanks a million
I'm in your debt
You are the best
I owe you one
Thank you so much
I really appreciate it
Please accept my deepest thanks
www.englishstudypage.com

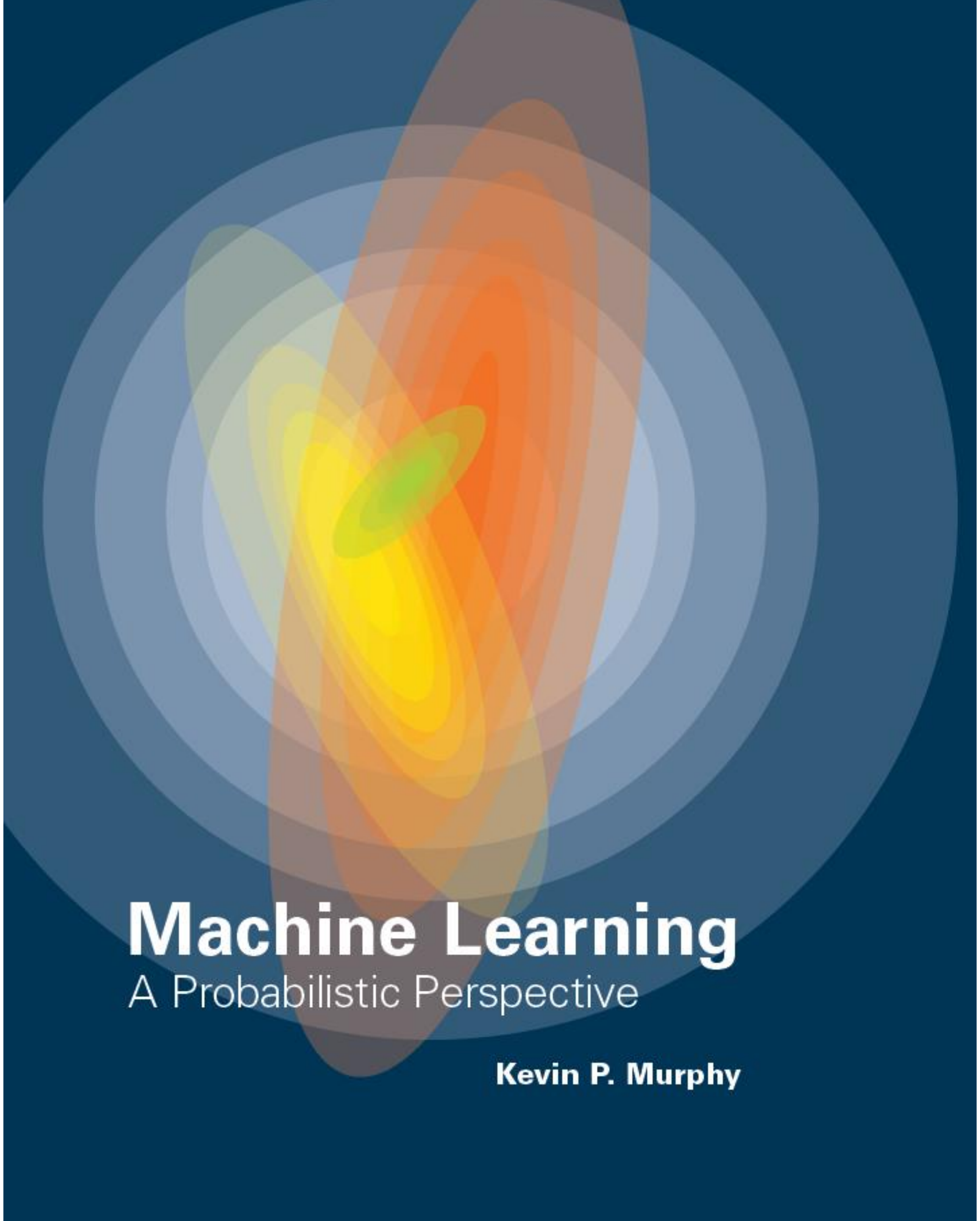
希望大家和我一起，
终身学习！

References


2016
基因组学
GENOMICS


CAMBRIDGE
生物信息学数据分析丛书
生物序列分析
BIOLOGICAL SEQUENCE ANALYSIS
(英) R.Durbin S.Eddy A.Krogh G.Mitchison 编著
王俊 郭一然 单杲 主译
www.sciencep.com


生物统计学
(第五版)



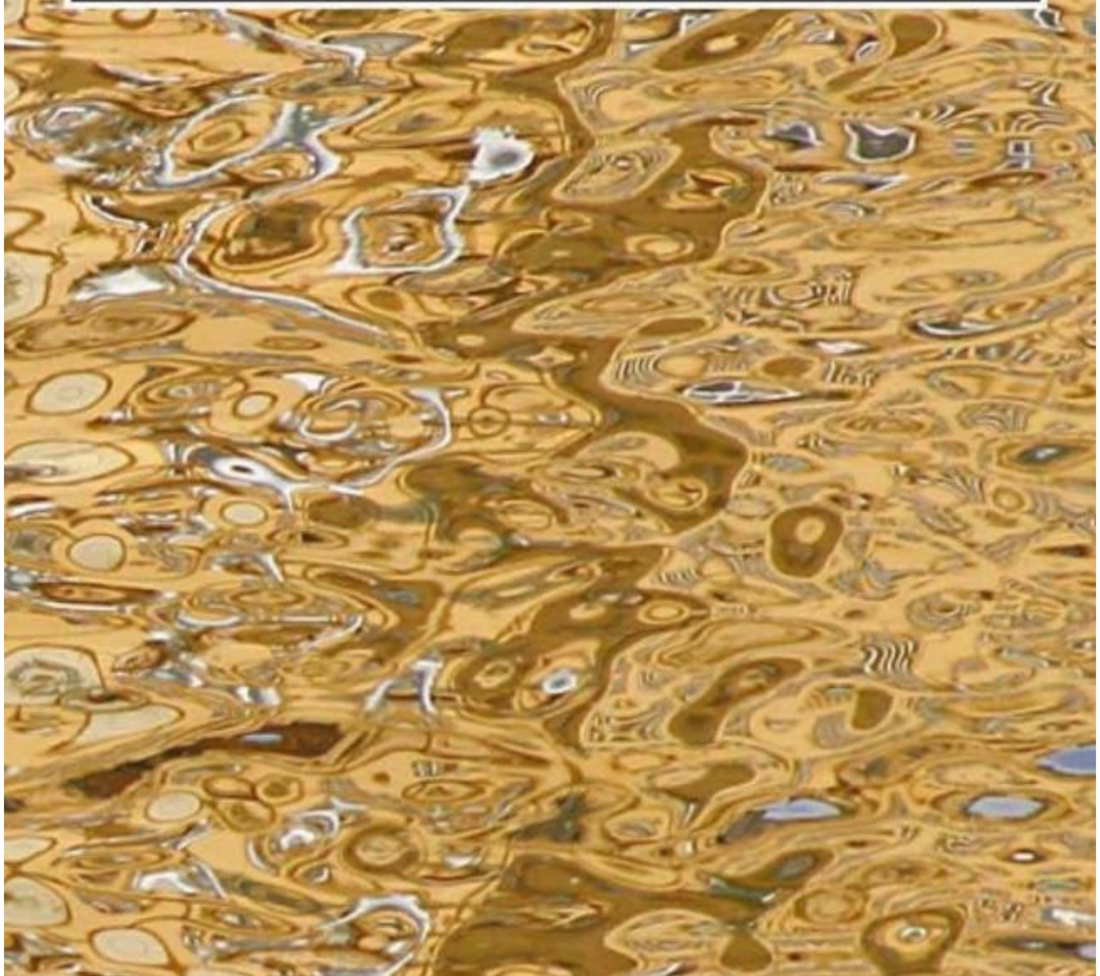
Machine Learning

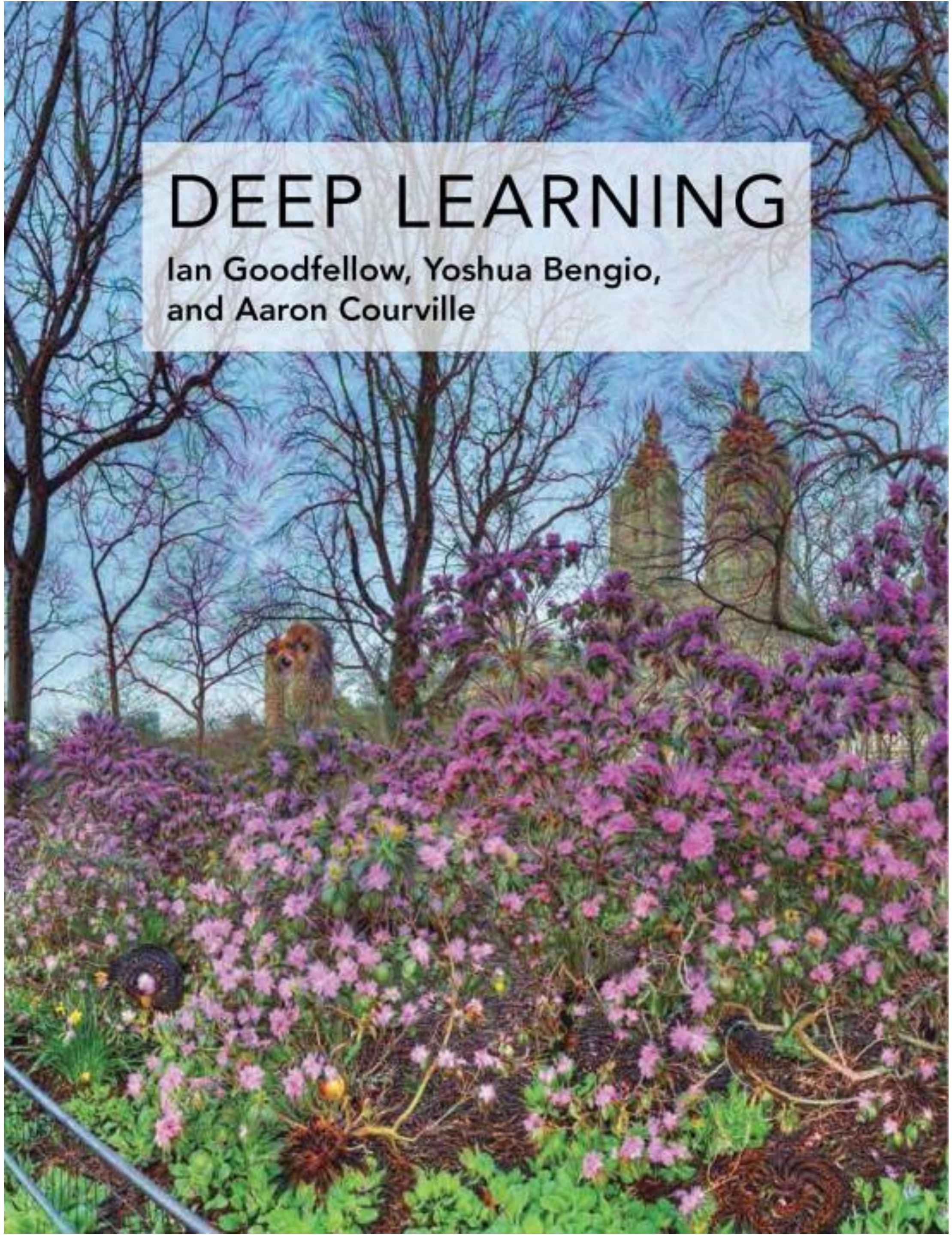
A Probabilistic Perspective

Kevin P. Murphy

PATTERN RECOGNITION AND MACHINE LEARNING

CHRISTOPHER M. BISHOP




DEEP LEARNING
Ian Goodfellow, Yoshua Bengio,
and Aaron Courville