Artificial Intelligence

School of Electronic and Computer Engineering
Peking University

Wang Wenmin

Models in Machine Learning

School of Electronic and Computer Engineering
Peking University

Wang Wenmin

Contents:

Contents:

Objectives 教学目的

- This chapter will discuss in detail about the models that have been used in machine learning.
  这一章我们将详细讨论用于机器学习的一些模型。

What are Learning Models 什么是学习的模型

- The learning models are used to denote the approaches that can handle to fulfil a learning task.

  学习模型用于表示可以处理完成一个学习任务的方法。

- It is the algorithm-level but not data-level models (approaches).

  它是属于算法层面而不是数据层面的模型 (方法)。

Why Study Learning Models 为什么要研究学习的模型

- The result of machine learning is heavily dependent on the choice of an approach for solving the learning task.

  机器学习的效果在很大程度上取决于解决该学习任务时所选用的方法。

Typical Models for Machine Learning 机器学习的代表性模型

Models 模型	Brief Statements 简短描述	Sub-models 子模型	Typical Algorithm 典型算法
Probabilistic 概率	Use probabilistic models to denote the conditional dependence between random variables. 采用概率模式来表示随机变量之间的条件相关性。	Bayes 贝叶斯	Bayesian Network 贝叶斯网络
		Generative 生成	Probabilistic Program. 概率规划
		Statistic 统计	Linear Regression 线性回归
Geometric 几何	Use geometric models such as line, plane, distance or manifold to construct learning algorithms. 采用线、面、距离或流行等几何图形模型来构建学习算法。	Line 线	Linear Regression 线性回归
		Plane 面	SVM 支撑向量机
		Distance 距离	*k*-NN *k*-近邻
		Manifold 流行	Isomap 等距映射
Logical 逻辑	Use logical models to construct learning algorithms. 采用逻辑模型来构建学习算法。	Logic 逻辑	Inductive Logic Program. 归纳逻辑编程
		Rule 规则	Association Rule 相关规则
Networked 网络	Use networked models to construct learning algorithms. 采用网络模式构建机器学习算法。	Shallow 浅层	Perceptron 感知机
		Deep 深层	CNN 卷积神经网络

Contents:

- 12.1. Probabilistic Models
- 12.2. Geometric Models
- 12.3. Logical Models
- 12.4. Networked Models

Probabilistic Models

School of Electronic and Computer Engineering
Peking University

Wang Wenmin

What are Probabilistic Models 什么是概率模型

- Probability theory is used to study the uncertainty problems, such as:
  概率论用于研究不确定性问题，例如：
  - what is the best prediction given some data?
    什么是给定某些数据的最佳预测？
  - what is the best model given some data?
    什么是给定某些数据的最优模型？
  - what measurement should I perform next?
    接下来应该执行什么度量？
- Probabilistic model is to use probability theory to express all forms of uncertainty.
  概率模型是采用概率论来表示所有不确定性的形式。
- Bayes rule allows us to infer unknown quantities, adapt our models, make predictions and learn from data.
  贝叶斯规则允许我们推断未知量，适配我们的模型，做出预测并从数据中学习。

Typical Probabilistic Methods 典型的概率方法

- Naive Bayes and Bayes Network 朴素贝叶斯和贝叶斯网络
- Probabilistic program 概率规划
- Gaussian processes 高斯过程
- Hidden Markov models 隐藏马可夫模型

Probability Used in Machine Learning 用于机器学习的概率

- Often the goal of machine learning is to determine the probability of an event, e.g.,
  机器学习的目标常常是要确定某个事件的概率，例如：

  P (a person is at risk for a disease)
  某人患某种疾病病的风险

- There are two main perspectives:
  有两个主要的学派：
  - Frequentist: probabilities are long-run frequencies, flip a coin a million times to determine if it’s fair.
    频率学派：概率是重复多次的频率，如投掷硬币一百万次以确定正反面出现频率是否相等。
  - Bayesian: probabilities quantify our uncertainty in events designed to get the closest to the truth given a specific set of data.
    贝叶斯学派：给定一组特定数据，概率对事件的不确定性进行量化，旨在最接近事实真值。

Bayes Rule 贝叶斯规则

Thomas Bayes (1701-1761),
an English statistician, philosopher
托马斯·贝叶斯，英国统计学家、哲学家

- Bayes rule tells us how to do inference about hypotheses from data.
  贝叶斯规则告诉我们如何从数据推断假设。
- Learning and prediction can be seen as forms of inference.
  学习和预测可以看作是推理的形式。
- Bayes rule allows us to 贝叶斯规则允许我们

infer unknown quantities，推断未知的量、 → **adapt our models 适配我们的模型、** → **make predictions, and 做出预测、以及** → **learn from data. 从数据中学习。**

$$P(\text{hypothesis} \mid \text{data}) = \frac{P(\text{data} \mid \text{hypothesis})\, P(\text{hypothesis})}{P(\text{data})}$$

Bayesian Machine Learning 贝叶斯机器学习

Everything follows from two simple rules:
任何事物都遵循如下**两个简单规则**：

Sum rule 求和规则： $P(x) = \sum_y P(x, y)$

Product rule 乘积规则： $P(x , y) = P(x)\, P(y \mid x)$

$P(\theta \mid \text{ƛ}, m)$ ： posterior probability of given data ƛ 给定数据D的后验概率

$P(\text{ƛ} \mid \theta, \mathrm{m})$ ： likelihood of parameters θ in model m 模型m中参数θ的似然性

$P(\theta \mid \mathrm{m})$ ： prior probability of θ θ的先验概率

$P(\text{ƛ} \mid \mathrm{m})$ ： evidence 证据

$$P(\theta \mid \text{ƛ}, m) = \frac{P(\text{ƛ} \mid \theta, m)\, P(\theta \mid m)}{P(\text{ƛ} \mid m)}$$

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{evidence}}$$

Parametric vs Non-parametric Methods 参数与非参数方法

☐ Parametric Methods 参数方法

- Assume some *finite set of parameters* θ. Given the parameters, future predictions *x*, are independent of the observed data D:
  设有限参数集 θ。给定该参数，则特征预测 x 独立于观测数据 D，即：

$$P(x \mid \theta, \lambda) = P(x \mid \theta)$$

  therefore capture everything there is to know about the data.
  因此，捕捉任何事物在于要知晓这些数据。
- So, the model's complexity is bounded even if the amount of data is unbounded.
  故，即使数据量是无限的，模型的复杂性也是有界的。
- This makes them not very flexible.
  这使得参数方法**不是很灵活**。

Parametric vs Non-parametric Methods 参数与非参数方法

Non-parametric Methods **非参数方法**

- Assume that the data distribution cannot be defined in terms of such a finite set of parameters. But they can often be defined by assuming an *infinite dimensional* θ.

  设数据分布不能用这样一个有限参数集来定义。但通常能够由假设的无限维度 θ 来定义。

  Usually we think of θ as a function.

  通常，我们把 θ 看作是一个函数。

- The amount of information that θ can capture about the data D can grow as the amount of data grows.

  θ 可以捕捉数据 D 的信息量可以随着数据量的增长而增长。

- This makes them more flexible.

  这使得非参数方法**更加灵活**。

Parametric vs Non-parametric Methods 参数与非参数方法

	Parametric 参数	Non-parametric 非参数
Brief Statements 简短陈述	The methods have a fixed number of parameters. 该方法具有**固定的**参数个数。	The number of parameters grow with the amount of training data. 参数个数随着训练数据的量而**增长**。
Typical Algorithm 典型算法	Support Vector Machine 支撑向量机	*k*-nearest neighbors *k* 近邻

Parametric vs Non-parametric Methods 参数与非参数方法

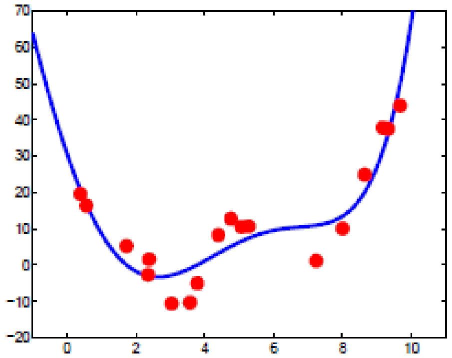
	Parametric 参数	Non-parametric 非参数
Feature 特征	Having a fixed number of parameters. 具有**固定的**参数个数	The number of parameters grow with the amount of training data. 参数的个数随着训练数据的量而**增长。**
Advantage 优点	Often being faster to use 通常**易于使用**	More flexible. 更加**灵活**
Disadvantage 缺点	Making stronger assumptions about the nature of the data distributions. 对数据分布的性质做出**严格的假设。**	Often computationally intractable for large datasets. 时常在计算上**难以应付大的数据集。**

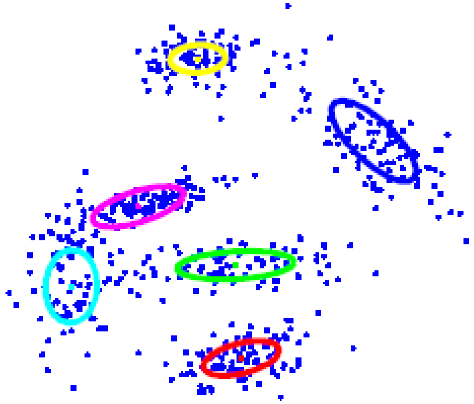
Why Non-parametric 为什么非参数

- Flexibility 灵活性
- Better predictive performance 更好的预测性能
- More realistic 更现实

Most methods are non-parametric 大多数算法是非参数的

- Kernel methods 核方法
  (SVM, Gaussian process 高斯过程)
- Deep networks 深度网络
- *k*-nearest neighbors *k* 近邻
- etc.





Non-parametric Methods and Uses 非参数方法及用途

- Bayesian non-parametric has many uses.
  贝叶斯非参数有许多用途。

Some modelling goals and examples of associated non-parametric Bayesian models.
一些建模目标和与贝叶斯非参数模型关联的实例

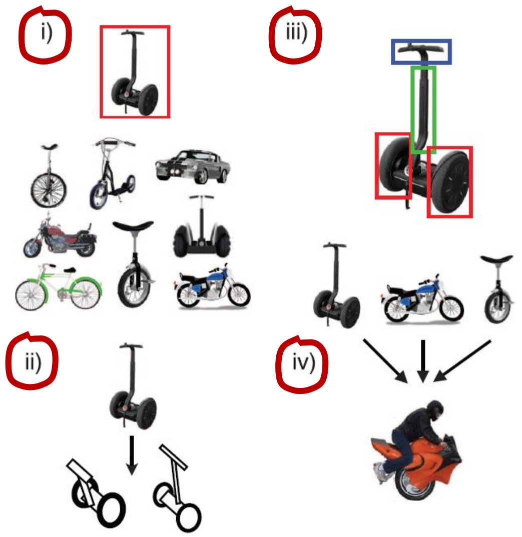
Non-parametric Methods 非参数方法	Examples 示例
Distributions on functions 功能分布	Gaussian process 高斯过程
Clustering 聚类	Chinese restaurant process 中餐馆过程
Sparse binary matrices 稀疏二元矩阵	Indian buffet processes 印度自助餐过程
Survival analysis 生存分析	Beta processes 贝塔过程
Distributions on measures 措施分布	Completely random measures 完全随机措施

Case Study: Probabilistic Program Induction 概率规划归纳

Brenden Lake, Ruslan Salakhutdinov, Joshua Tenenbaum, “Human-level concept learning through probabilistic program induction” 凭借概率规划归纳法的人类层级概念学习, *Science*, Vol. 350, Dec. 2015.

A single example of a new concept can be enough information to support:
一个新概念的单一实例可以有足够的信息来支持：

i. classification of new examples,
新样本的分类，
ii. generation of new examples,
新样本的生成，
iii. parsing an object into parts and relations,
将一个物体分成部件和关系，
iv. generation of new concepts from related concepts.
从相关概念生成新的概念。



People can learn rich concepts from limited data.
人们可以从有限数据中学习丰富的概念。

Case Study: Probabilistic Program Induction 概率规划归纳

☐ Bayesian Program Learning (BPL) **贝叶斯规划学习 (BPL)**

- BPL learns simple stochastic programs to represent concepts, and builds new types of concepts compositionally from parts, sub-parts, and spatial relations.

  BPL学习简单的随机规划来表示概念，并从部件、子部件、以及空间关系的组合来构建新类型的概念。

- Each new type is also represented as a generative model, and this lower-level generative model produces new examples (or tokens) of the concept.

  每个新类型还可以表示为一个生成模型，并且这个低层的生成模型产生概念的新样本 (或记号)。

- The joint distribution on types ψ, a set of *M* tokens $\theta^{(1)}, \ldots, \theta^{(M)}$, and the corresponding binary images $I^{(1)}, \ldots, I^{(M)}$ factors as

  联合分布，其参数：类型 ψ，一组 *M* 个记号 $\theta^{(1)}, \ldots, \theta^{(M)}$ ，和对应的二值图像 $I^{(1)}, \ldots, I^{(M)}$因子，即：

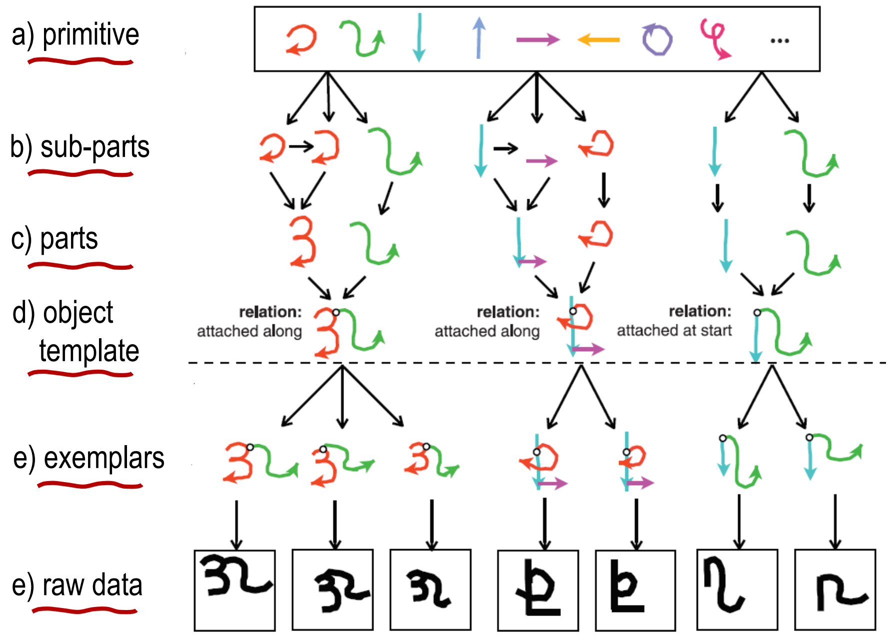
$$P(\psi, \theta^{(1)}, \ldots, \theta^{(M)}, I^{(1)}, \ldots, I^{(M)}) = P(\psi)\prod_{m=1}^{M} P(I^{(m)}|\theta^{(m)})P(\theta^{(m)}|\psi)$$

Case Study: Probabilistic Program Induction 概率规划归纳

□ One-shot learning 一次性学习

It incorporates three principles: compositionality, causality, and learning to learn.

它包含三个基本原理：**组合性、因果性、和学会学习。**



A generative model of handwritten characters.

一个手写字符的生成模型

a) generate new types by primitive,
   由原语生成新的类型
b) extract sub-parts,
   提取子部件
c) make parts,
   生成部件
d) combine parts with relations to define object template,
   根据关系组合部件来定义对象模板
e) generate new exemplars by running these templates,
   通过运行这些模板生成新的样例
f) render them as raw data.
   将其渲染成原始数据

Thank you for your attention!

AI

Geometric Models

School of Electronic and Computer Engineering
Peking University

Wang Wenmin

Contents:

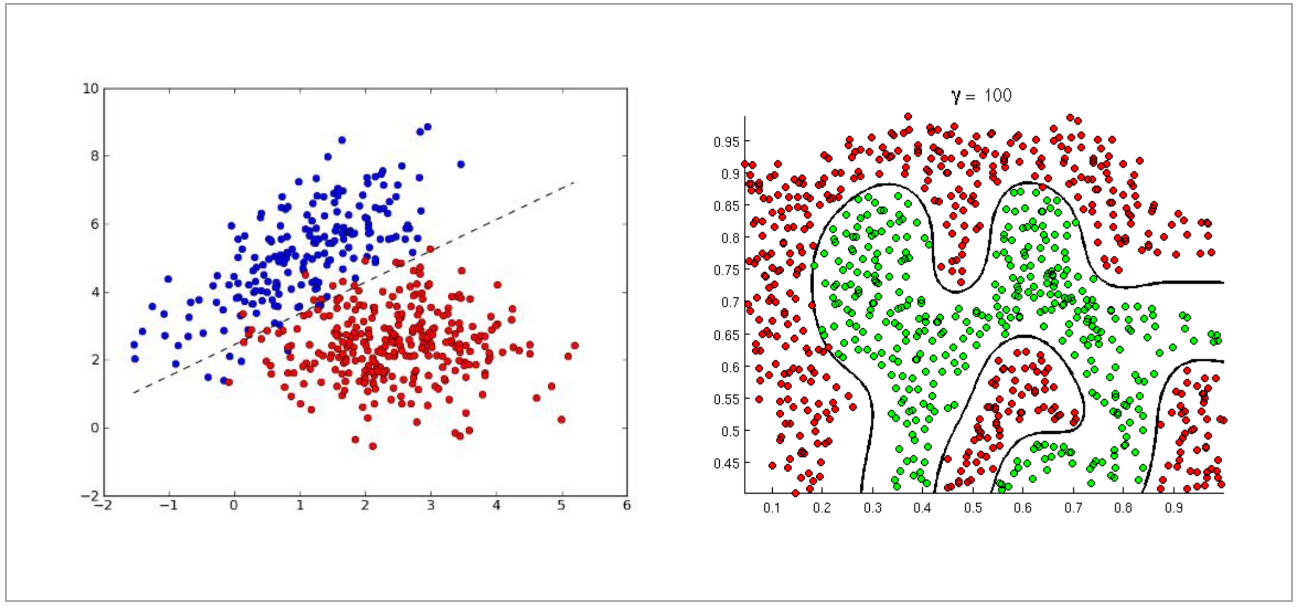
What are Geometric Models 什么是几何模型

- A geometric model is constructed directly in instance space. The space can be thought as the geometric concepts, such as,
  几何模型直接在**实例空间**构建。空间可以被认为是几何的概念，例如:
  - Euclidean geometry: e.g., lines, planes, and Euclidean distances.
    欧式几何：如线、面、以及欧式距离。
  - Riemannian geometry: e.g., manifold, and Riemannian distances.
    黎曼几何：如流形、黎曼距离。
- The geometric model keeping to two or three dimensions is easy to visualize.
  二维或三维的几何模型**易于可视化**。
- Geometric concepts that potentially apply to high-dimensional spaces are usually prefixed with ‘hyper’: e.g., hyper-plane.
  可能适用于高维空间的几何概念通常冠以hyper (超)，例如：hyper-plane (超平面)。

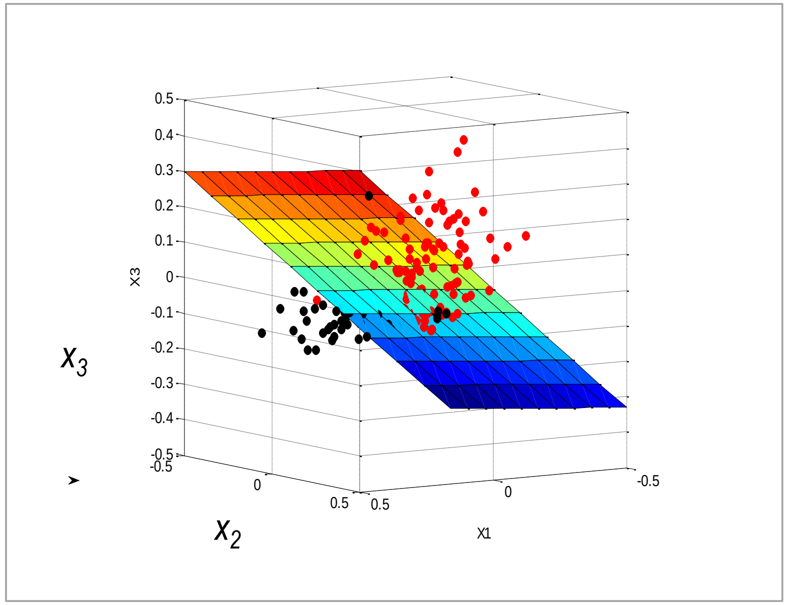
Line and Plane 线与面

- Using such geometric models as line or plane to construct some algorithms for machine learning.

  使用线或平面这样的几何模型来构建一些机器学习算法。



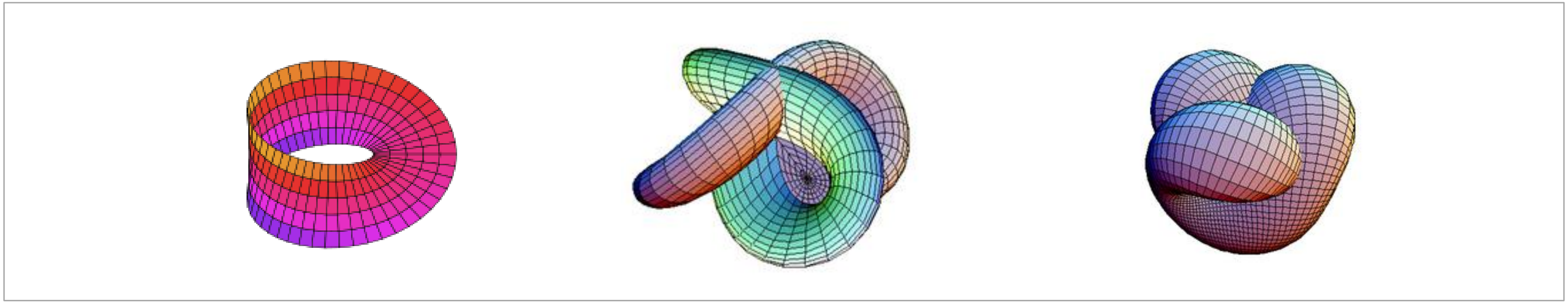
A line in two dimension
二维中的线



A plane in three dimensions
三维中的平面

Manifold 流形

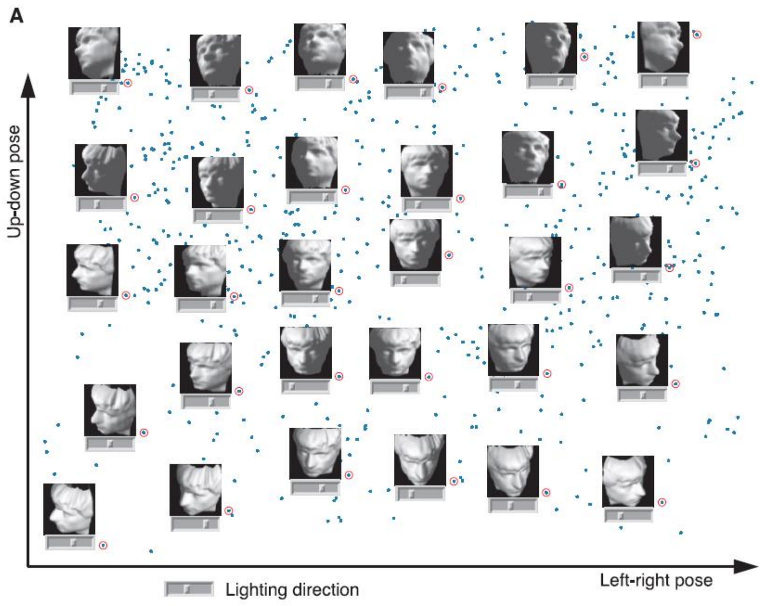
- What is Manifold 什么是流形
  - A topological space that resembles Euclidean space near each point.
    一种拓扑空间，其每个点附近存在近似的**欧几里得空间**。
  - More precisely, each point of an *n*-dimensional manifold has a neighborhood that is homeomorphic to the Euclidean space of dimension *n*.
    更精确地说，一个*n*维流形的每个点都有一个相邻点，它与维度为*n*的欧几里得空间同胚。

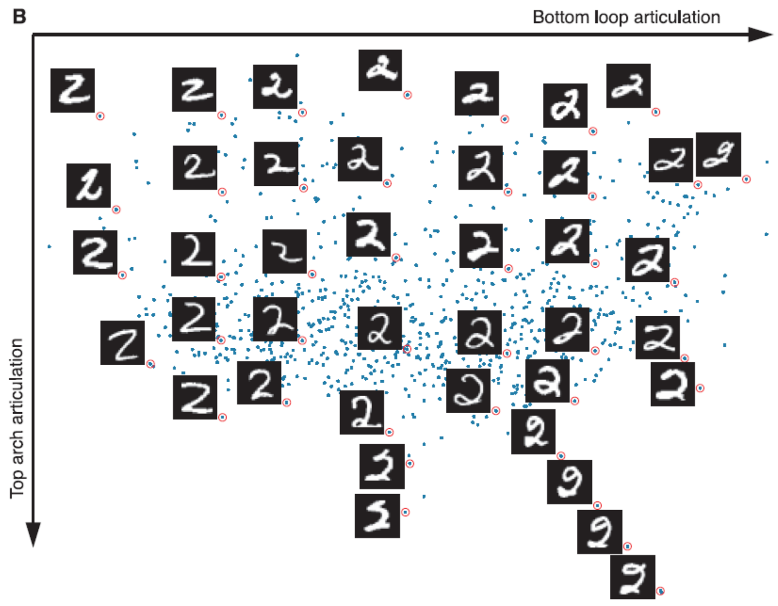


Manifold 流形

☐ Why need Manifold 为什么需要流形

Source: Science, 290, 2000





(A) A sequence of 4096-dimensional vectors, representing the brightness values of 64x64 pixel images of a face rendered with different *poses* and *lighting directions*.

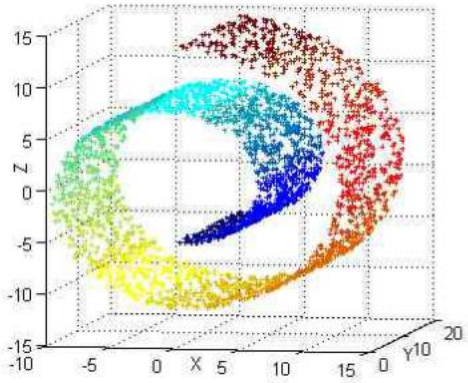
一个4096维的向量序列，表示64×64像素人脸图像的亮度值，呈现不同姿势和光照角度。

(B) N = 1000 handwritten "2". The two most significant dimensions articulate the major features of the "2": bottom loop articulation (x axis) and top arch articulations (y axis).

N = 1000的手写体"2"。两个最重要的维度刻画了"2"的主要特征："2"的底部变化(x轴)和顶部变化 (y轴)。

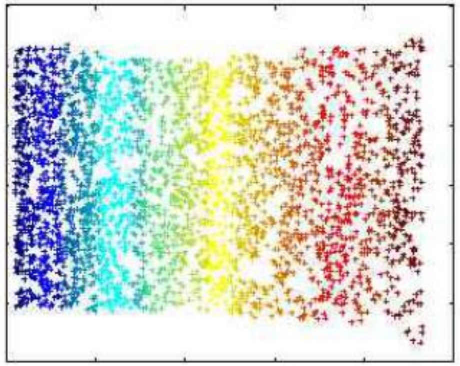
Manifold 流形

- Low-dimensional Structure of Data 数据的低维结构
  - Intrinsic low-dimensional properties can be found in high-dimensional input data.
    高维输入数据中可以发现其**固有的低维特性**。
- Low-dimensional Manifold Assumption 低维流形假设
  - High-dimensional input data is lying on a low-dimensional manifold.
    高维输入数据依附于一个**低维流形**。



Data points in a spiraling band (Swiss roll)
螺旋带（瑞士卷）上的数据点

Mapping



Unrolled manifold
拉平后的流形

Typical Algorithms of Manifold Learning 典型的流形学习的算法

- **Isomap** (Isometric mapping 等距映射)
  - Tenenbaum et al, "A Global Geometric Framework for Nonlinear Dimension Reduction" —种非线性降维的全局几何框架, *Science*, 290, 2000.
- **LLE** (Locally Linear Embeddings 局部线性嵌入)
  - Roweis et al, "Nonlinear Dimensionality Reduction by Locally Linear Embeddings" 凭借局部线性嵌入的非线性降维, *Science*, 290, 2000.
- **LE** (Laplacian Eigenmaps 拉普拉斯特征映射)
  - Belkin et al, "Laplacian Eigenmaps for Dimensionality Reduction and Data Representation" 用于降维和数据表达的特征映射, *NIPS* 2001.

Typical Algorithms of Manifold Learning 典型的流形学习的算法

- LTSA (Local Tangent Space Alignment 局部切空间对齐)
  - Zhang et al, “Principal Manifolds and Nonlinear Dimensionality Reduction via Local Tangent Space Alignment” 主要流形与采用局部切空间对齐的非线性降维, *SIAM Journal on Scientific Computing*, 26(1), 2005.
- Inductive Manifold Learning 归纳流形学习
  - Kim et al, “Inductive Manifold Learning Using Structured Support Vector Machine” 采用结构化支撑向量机的归纳流形学习, *Pattern Recognition*, 27(1), 2014.

Case Study 1: Isomap (Isometric mapping) 等距映射

- Isomap algorithm preserves geodesic distances but not the Euclidean distance.
  Isomap算法保留数据之间的测地线距离而不是欧几里得距离。
- A high-level description of the algorithm: 算法的简要描述

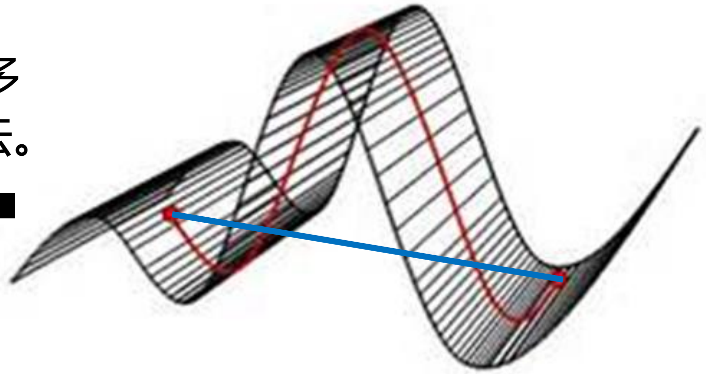
Source: Science, 290, 2000

确定每个点的相邻点。
Determine neighbors of each point.

构建一个邻接图。
Construct a neighborhood graph.

计算两点之间的最短距离。
Compute shortest path between two nodes.

计算低维嵌入，采用多元尺度分析(MDS)方法。
Compute lower-dimensional embedding using Multi-dimensional Scaling (MDS).



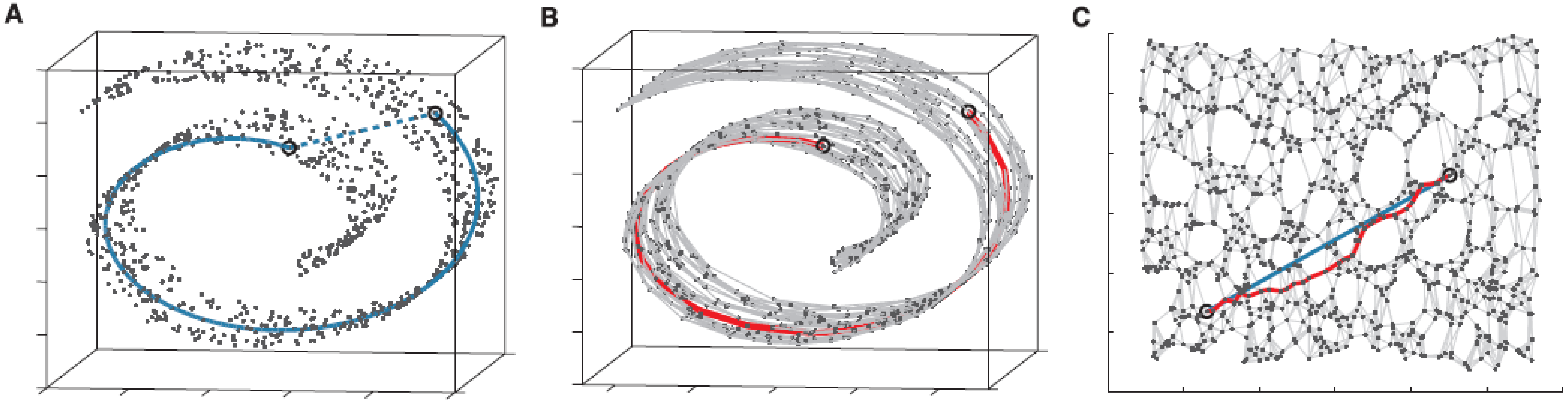
Geodesic distances
测地线距离

Euclidean distance
欧氏距离

Case Study 1: Isomap (Isometric mapping) 等距映射

- Using the "Swiss roll" data set to illustrate how Isomap exploits geodesic paths for nonlinear dimensionality reduction.

  采用"瑞士卷"数据集来说明Isomap如何利用测地线路径进行非线性降维。



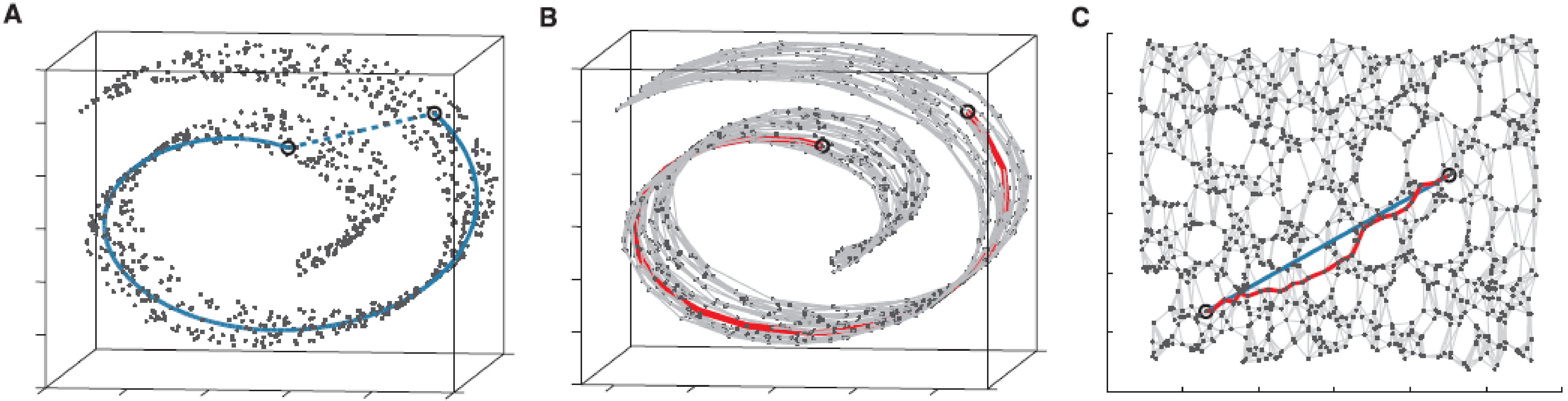
A. For two arbitrary points on a nonlinear manifold, their Euclidean distance is not equal to their geodesic distance.

图A：在非线性流形上任意两点间，其欧氏距离不等于其测地线距离。

Case Study 1: Isomap (Isometric mapping) 等距映射

- Using the “Swiss roll” data set to illustrate how Isomap exploits geodesic paths for nonlinear dimensionality reduction.

  采用“瑞士卷”数据集来说明Isomap如何利用测地线路径进行非线性降维。

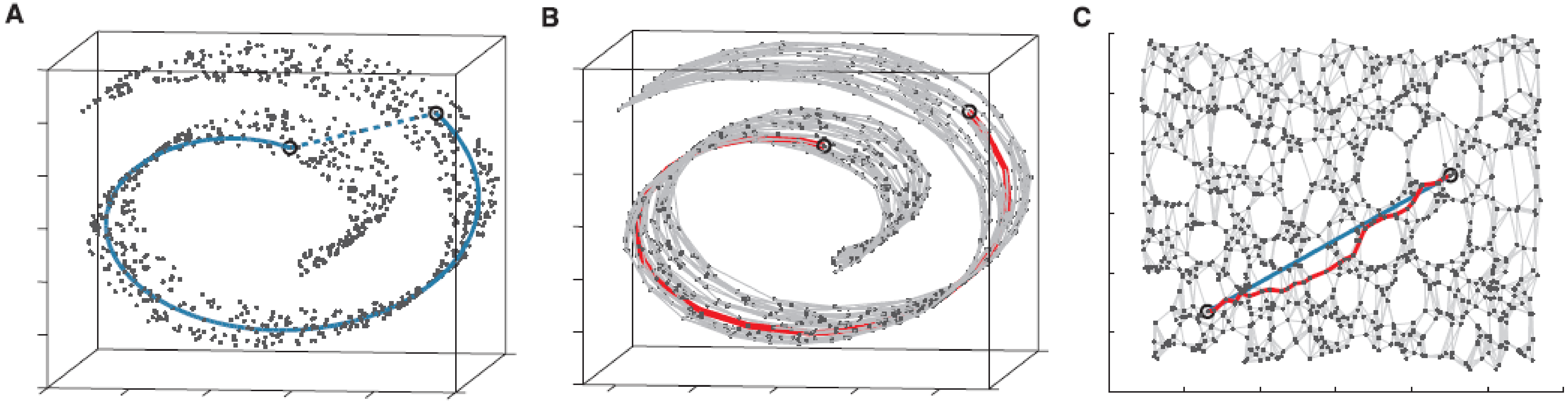


B. The neighborhood graph constructed by Isomap allows an approximation to the true geodesic path.
图B：由Isomap构建的邻接图，得到一条近似于真正的测地线路径。

Case Study 1: Isomap (Isometric mapping) 等距映射

- Using the “Swiss roll” data set to illustrate how Isomap exploits geodesic paths for nonlinear dimensionality reduction.

  采用“瑞士卷”数据集来说明Isomap如何利用测地线路径进行非线性降维。

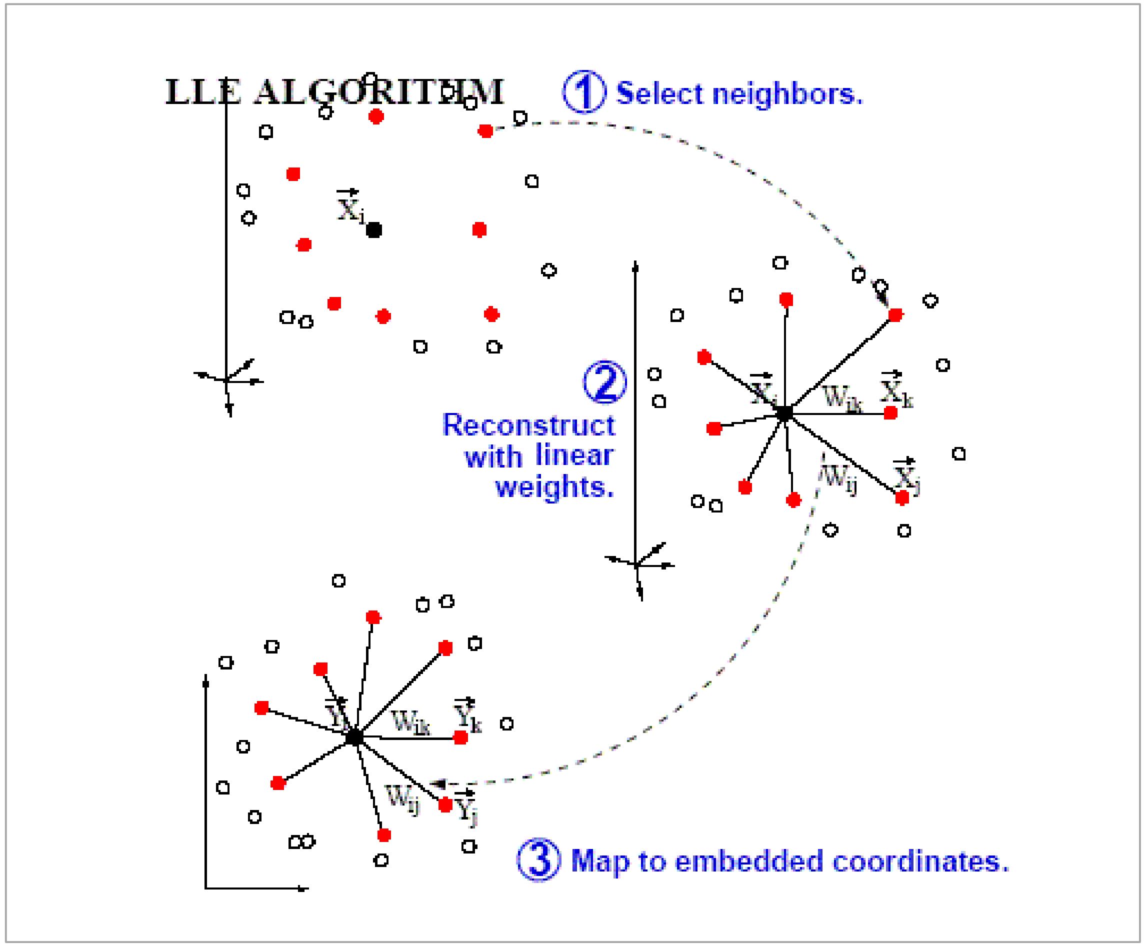


C. The two-dimensional embedding by Isomap best preserves the shortest path distances in the neighborhood graph.
图C：由Isomap生成的二维嵌入，很好地保持了邻接图中最短路径的距离。

Case Study 2: LLE (Locally Linear Embedding) 局部线性嵌入

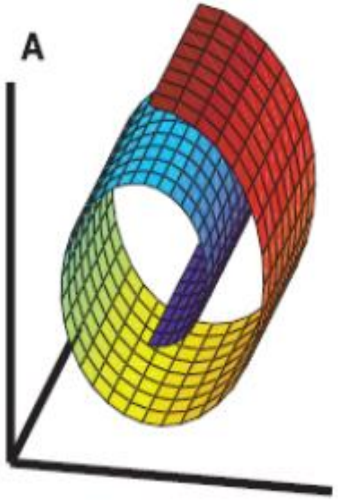
Source: Science, 290, 2000

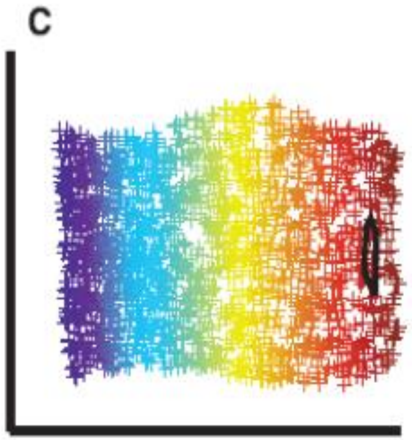
- LLE embeds data points in a low-dimensional space by finding the optimal linear reconstruction in a small neighborhood.
  LLE通过在小邻域中找到最佳线性重构，将数据点嵌入在低维空间中。
- It computes the reconstruction weights for each point, and then minimizes the embedding cost by solving an eigenvalue problem.
  它计算每个点的重建权重，然后通过求解特征值问题对嵌入成本进行最小化。



Case Study 2: LLE (Locally Linear Embedding) 局部线性嵌入

- Using the “Swiss roll” data set to illustrate how LLE exploits neighborhood for nonlinear dimensionality reduction. 采用“瑞士卷”数据集来说明LLE如何利用邻域来处理非线性降维。





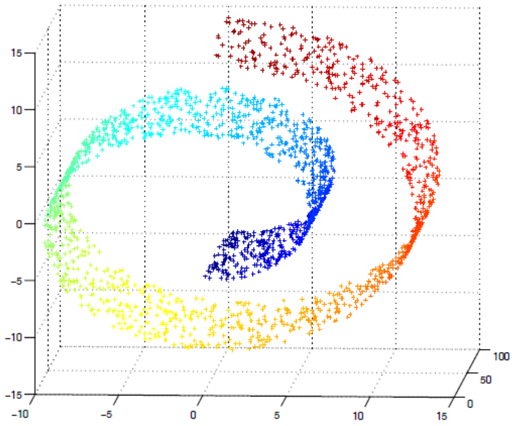
(A) The color coding illustrates the neighborhood preserving mapping discovered by LLE.
颜色编码说明由LLE获得的邻域保留映射。

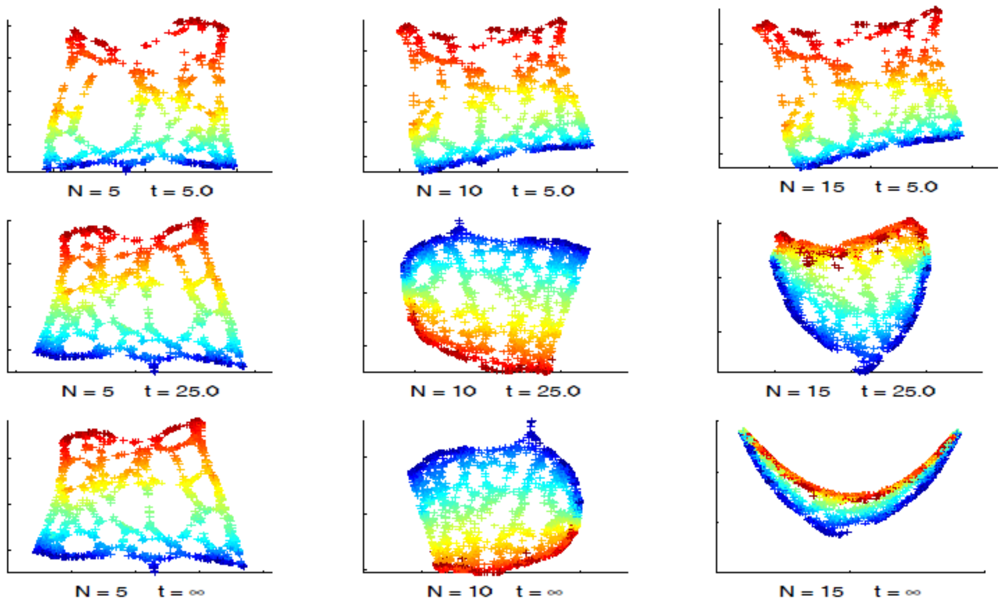
(B) Black outlines show the neighborhood of a single point on the “Swiss roll” .
黑色轮廓显示出在“瑞士卷”上单个点的邻域。

(C) Black outlines show the neighborhood of a single point on the unrolled roll”.
黑色轮廓显示出在拉平的“瑞士卷”上单个点的邻域。

Case Study 3: LE (Laplacian Eigenmaps) 拉普拉斯特征映射

- LE restates the nonlinear mapping problem as an embedding of vertices in a graph, and uses the graph Laplacian to derive a smooth mapping.
  LE重申将非线性映射问题作为一种图的顶点嵌入，并且采用图拉普拉斯来生成一个平滑的映射。

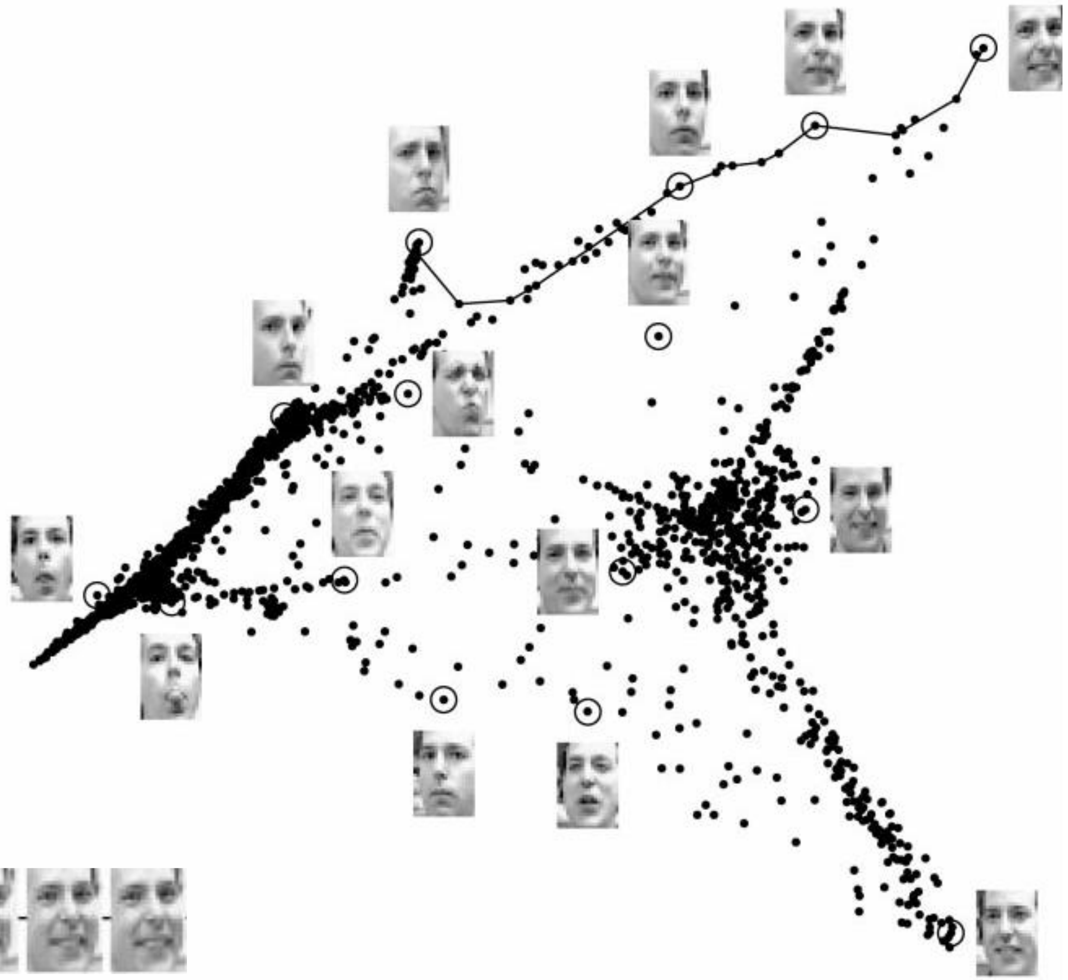




N denotes number of nearest neighbors, and *t* denotes kernel parameters.
*N*表示最近邻接点的数量，而*t*表示核参数。

Typical Applications of Manifold Learning 典型的流形学习应用

Image processing	☐	图像处理
Gene expression profiles	☐	基因表达图谱
Handwriting recognition	☐	手写体识别
Face recognition	☐	人脸识别



Source: Science, 290, 2000

The right and bottom images show the two-dimensional embeddings of faces discovered by LLE algorithm.

右侧和下面的图像展示由LLE算法获得的人脸二维线性嵌入。

Thank you for your attention!

AI

Logical Models

School of Electronic and Computer Engineering
Peking University

Wang Wenmin

Contents:

What are Logical Models 什么是逻辑模型

- The 'logical' models are defined in terms of easily interpretable logical expressions, or can be easily translated into rules that are understandable by humans.

  "逻辑"模型被定义为**易于解释的逻辑表达式**，或者**易于转换成人类能够理解的规则**。

Typical Logical Models 典型的逻辑模型

- First-order logic 一阶逻辑
- Association rules 关联规则
- Decision tree 决策树

Case Study: Association Rule 关联规则

- It is a method for discovering interesting relations between variables in databases.
  是一种用于在大型数据库中变量之间发现有趣关系的方法。
- It is intended to identify *strong rules* discovered in databases using different measures of interestingness.
  它采用差异趣味性度量方式，旨在识别数据库中发现的强规则。
- It can be used for discovering regularities between products in supermarkets, called **Market Basket Analysis**.
  可用于发现超市产品之间的规律，称之为**购物篮分析**。

$$\{\text{onions, potatoes}\} \Rightarrow \{\text{burger}\}$$

- It is also applicable to other application domain such as: bioinformatics, medical diagnosis, Web mining, and scientific data analysis.
  还可以用于其它应用领域，例如：生物信息学、医学诊断、Web挖掘、以及科学数据分析。

Case Study: Association Rule 关联规则

- Definition 定义
  - $I = \{i_1, i_2, \ldots, i_n\}$,
    a set of *n* attributes called *items*. 称之为项的*n*个属性集。
  - $T = \{t_1, t_2, \ldots, t_m\}$,
    a set of transactions called transactions database. 称之为转换数据库的转换集
  - Each t_i is a *subset* of the items in *I*, i.e., 每个t_i是*I*中项的子集
    $$t_i \subseteq I.$$
  - $X \Rightarrow Y$,
    an *association rule*, 一个关联规则
    where $X, Y \subseteq I$, and $X \cap Y = \emptyset$.

Case Study: Association Rule 关联规则

- Example 举例
  - I = {bread, cheese, milk, apple, eggs, salt, yogurt, biscuit, butter, beer, peanut}
  - i_1 = {bread, cheese, milk},

    i_2 = {apple, eggs, salt, yogurt},

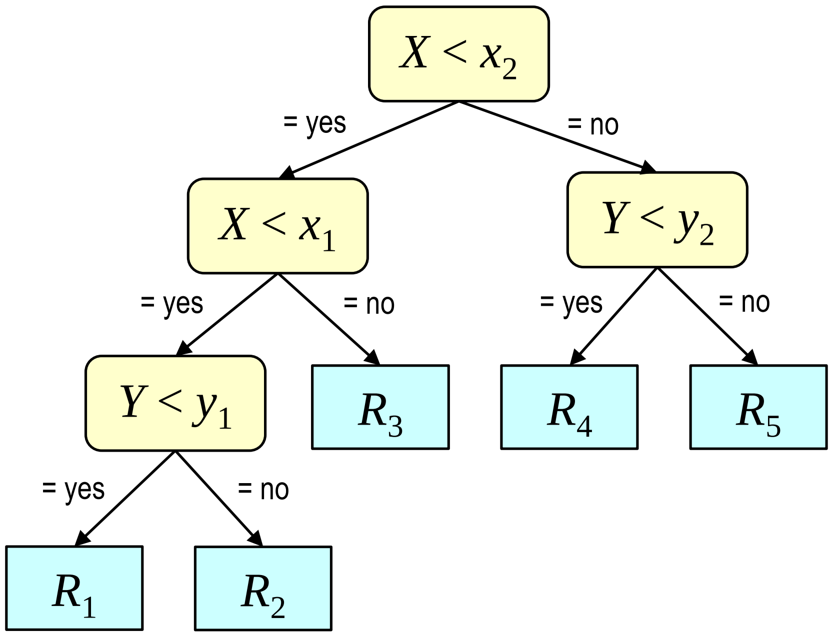
    ……,

    i_n = {beer, peanuts, eggs, milk}.
  - $X \Rightarrow Y$

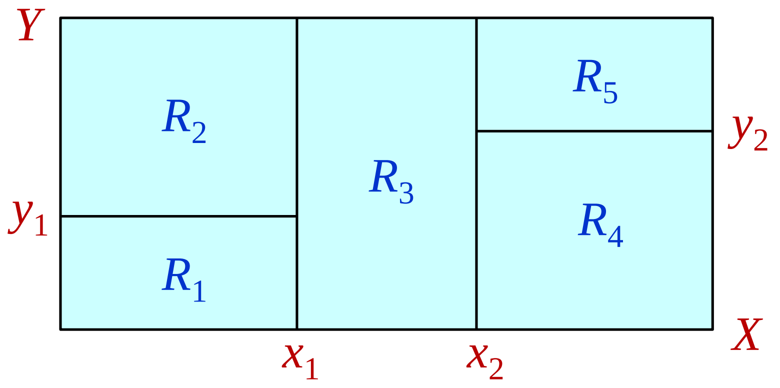
    {butter, bread} $\Rightarrow$ {milk}

Case Study: Decision Tree 决策树

- It uses a decision tree as a **predictive model** which maps observations about an item to conclusions about the item's target value.
  使用决策树作为**预测模型**，将项的观察结果映射到关于项的目标值。
- Tree models where target variable can take a finite set of values are called **classification trees**; where leaves represent class labels, branches represent conjunctions of features that lead to those class labels.
  目标变量可以取一组有限值的树模型称为**分类树**；其中叶节点表示类标签，分支表示通往这些类标签的特征连接。



A decision tree



Thank you for your attention!

AI

Networked Models

School of Electronic and Computer Engineering
Peking University

Wang Wenmin

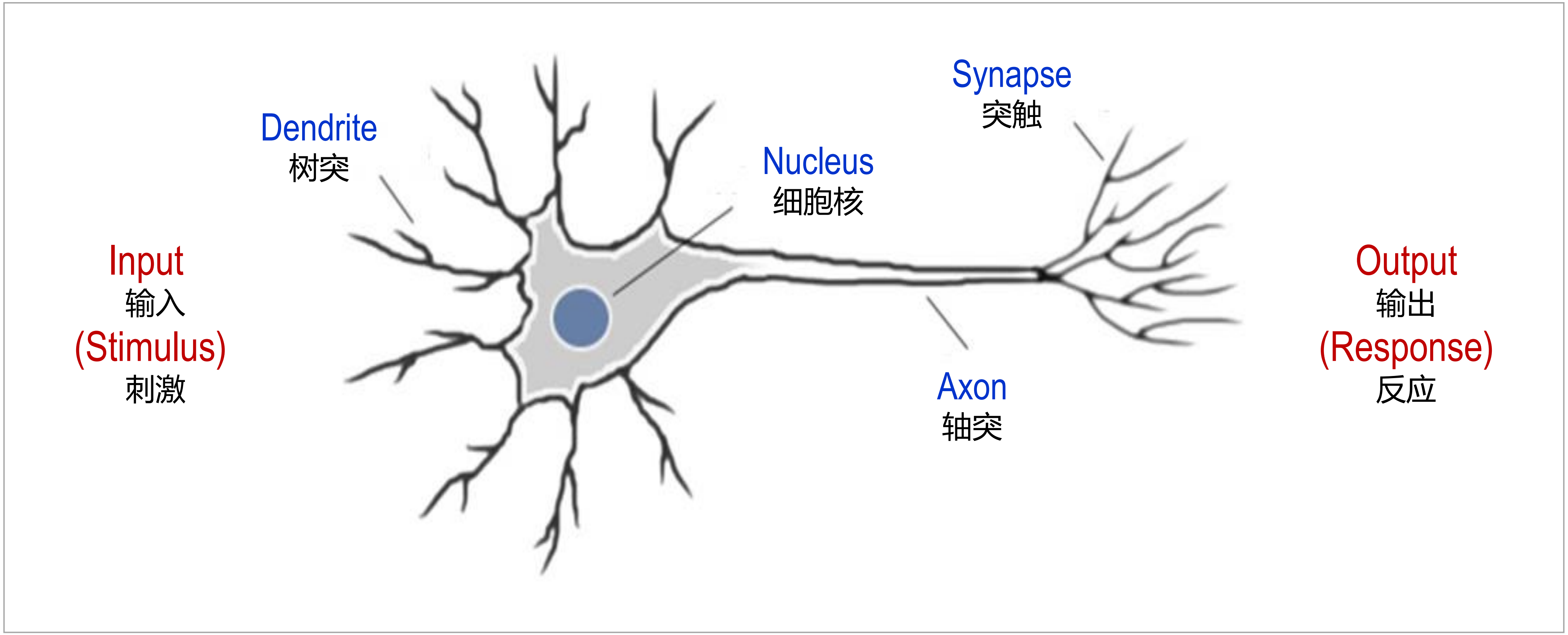
Contents:

What are Networked Models 什么是网络化模型

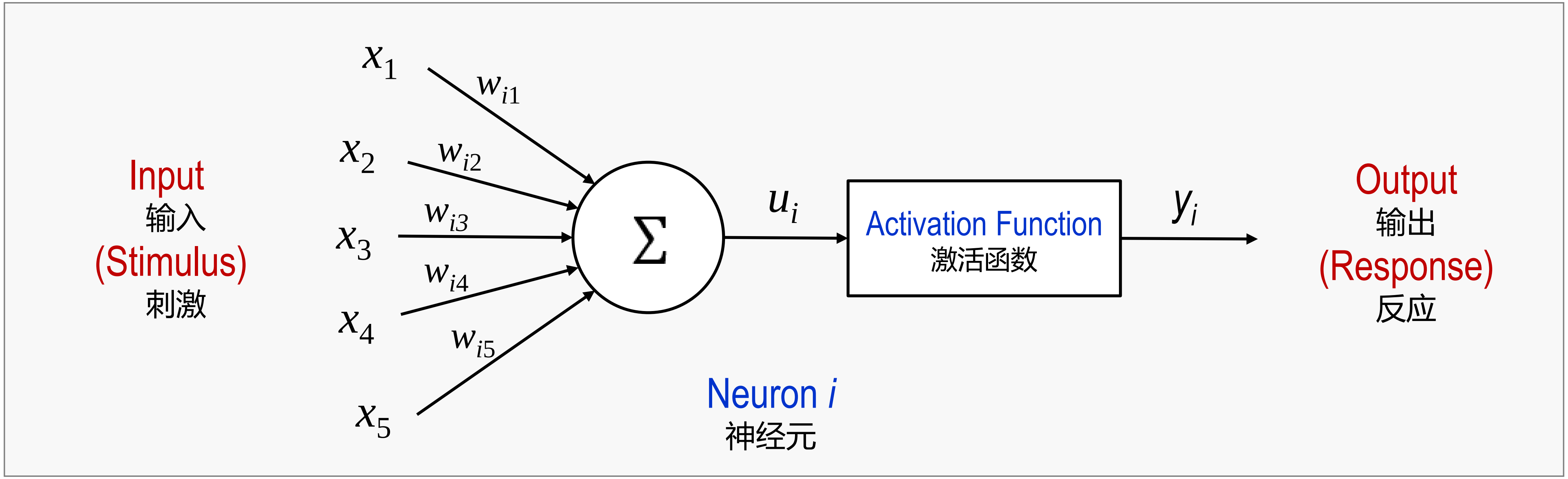
- The networked models here refer to as the models of artificial neural network (ANN).
  这里的网络化模型指的是**人工神经网络模型 (ANN)**。
- An ANN is an artificial representation of the human brain that tries to simulate its learning processing.
  一个ANN是**人脑的一种人工表征**，试图模拟人类的学习过程。
- ANN can be constructed a system by interconnected “neurons” which send messages to each other.
  ANN可以通过互联的“神经元”构建一个系统，神经元之间相互发送消息。
- The connections between neurons have numeric weights that can be tuned based on experience, making ANN adaptive to inputs and capable of learning.
  神经元之间的连接具有数值权重，可以通过经验调整，使ANN适应输入并且能够学习。

Contents:

Biological Neuron 生物神经元



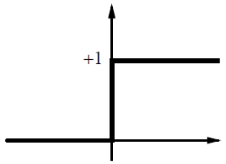
Artificial Neuron 人工神经元

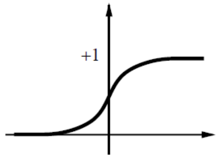


$$u_i = \sum_j w_{ij} \cdot x_j$$

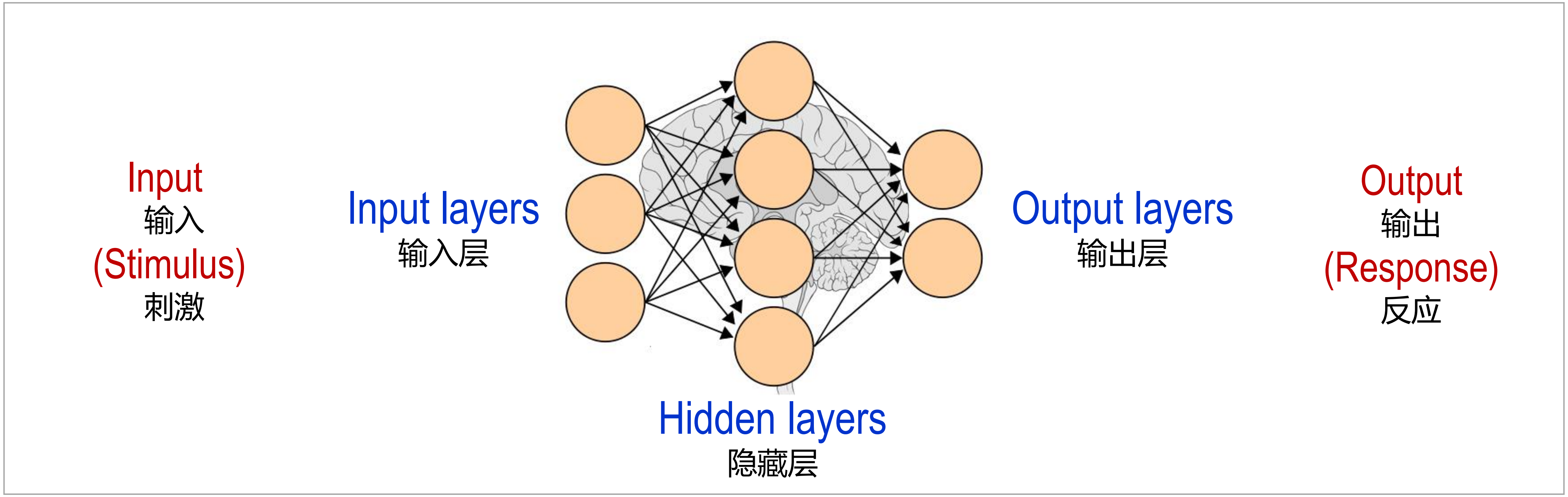
$$y_i = f(u_i) = \begin{cases} +1, \text{ if } u_i \ge \omega_0 \\ -1, \text{ if } u_i < \omega_0 \end{cases} \quad \text{or} \quad \frac{1}{1 + e^{-u_i}}$$

Activation function:
激活函数





Artificial Neural Network (ANN) 人工神经网络



- ANN is a family of learning models inspired by biological neural networks ■ ANN是受生物神经网络启发的一系列学习模型
- The interconnection between the different layers of neurons ■ 不同的神经元层次之间互联
- The learning process for updating the weights of the interconnections ■ 学习过程是为了更新互联权重
- The activation function that converts a neuron's weighted input to its output ■ 激活函数将神经元的加权输入转换为其输出

History of Artificial Neural Networks 人工神经网络的发展史

1943, McCulloch and Pitts
马卡洛和匹茨

- created a **computational model for neural networks** based on mathematics and algorithms called threshold logic.
  基于称之为阈值逻辑的数学和算法创建了**神经网络的计算模型**。

1954, Farley and Clark
法利和克拉克

- first used computational machines, then called calculators, to simulate a **Hebbian network**.
  首次利用计算的机器、后来称其为计算器，来仿真**赫布网络**。

1958, Rosenblatt
罗森布莱特

- created **perceptron**, an algorithm for pattern recognition, which is with only one output layer, so also called “single layer perceptron”.
  创建了**感知机**，一种模式识别算法，它仅有一个输出层，也被称为“单层感知机”。

History of Artificial Neural Networks 人工神经网络的发展史

1969, Minsky and Papert
明斯基和帕伯特

- Published a famous book entitled “Perceptrons”.
  出版了一本名为“感知机”的著名书籍。
- It pointed in this book that the single layer perceptrons are only capable of learning **linearly separable patterns**, but not possible to learn an XOR function.
  书中指出，单层感知机仅能学习**线性可分模式**，而不能用于学习异或功能。

1974, Werbos
韦伯斯

- Proposed the **back-propagation** algorithm, a method for training ANNs and used in conjunction with an optimization method such as gradient descent.
  提出了**反向传播算法**，一种用于训练ANNs的方法，并且与梯度下降等优化方法结合使用。
- Regenerates interest in the 1980s.
  1980年代才引起重视。

History of Artificial Neural Networks 人工神经网络的发展史

1989, Yann LeCun et al
雅恩·勒昆等人

- Published LeNet-5, a **pioneering 7-level convolutional neural network** (CNN) is applied to recognize hand-written numbers on checks.
  发表了LeNet-5，一种**开拓性的7层卷积神经网络** (CNN)，用于检查支票上的手写数字。

1992, Schmidhuber
施米德胡贝

- Proposed **recurrent neural network** (RNN), this creates an internal state which allows it to exhibit dynamic temporal behavior.
  提出了**循环神经网络**，它创建网络的内部状态，得以展现动态时间行为。

2006, Hinton and Salakhutdinov
辛顿和萨拉赫丁诺夫

- Renewed interest in **neural nets** was sparked by the advent of deep learning.
  深度学习的出现，再次引发了对**神经网络**的兴趣。

History of Artificial Neural Networks 人工神经网络的发展史

2012, Andrew Ng and Jeff Dean 吴恩达和杰夫·迪恩

- Google Brain team created a neural network that learned to recognize higher-level concepts, such as cats, from watching unlabeled images.
  Google大脑团队创建了一个神经网络，学会观看未标注图像来识别高层次概念，例如猫。

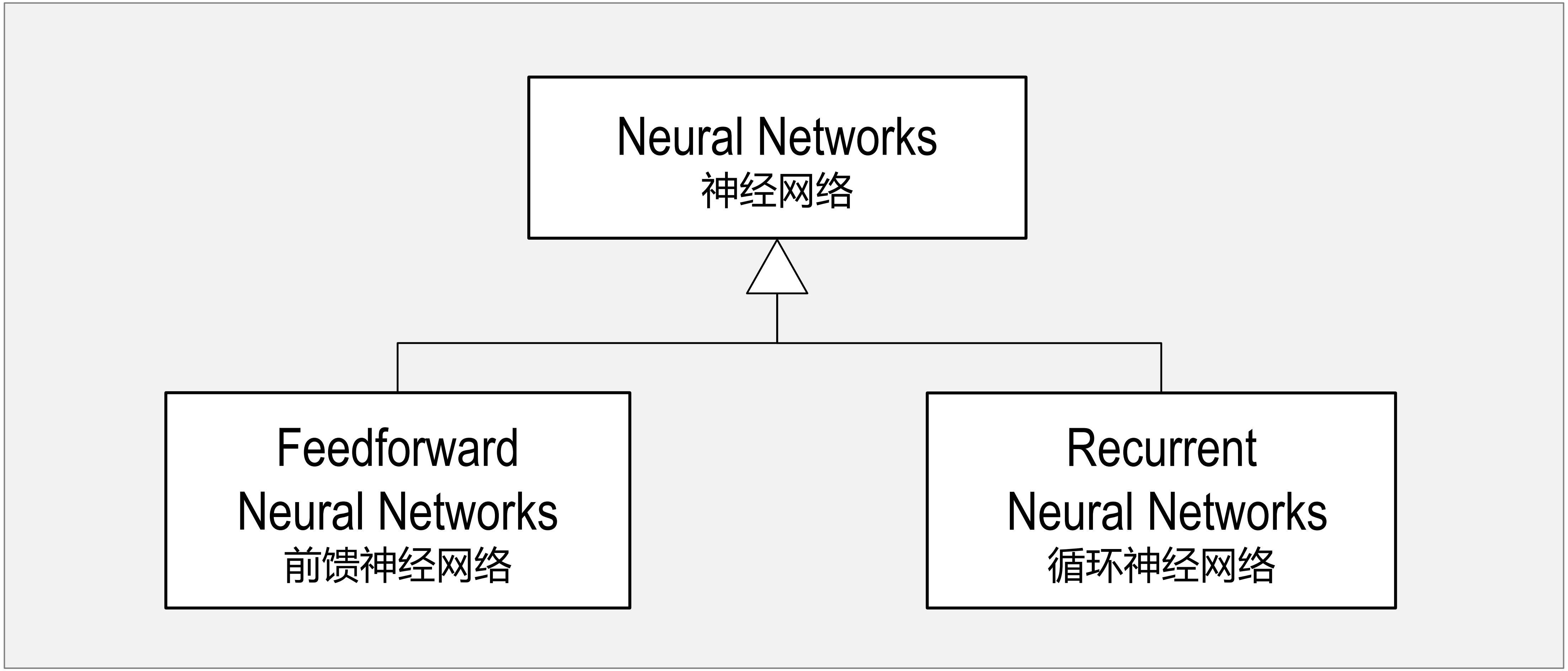
2012, Krizhevsky et al 克利则夫斯基等

- With Deep CNNs won the large-scale ImageNet competition by a significant margin over shallow machine learning methods.
  采用深度CNNs获得了大规模ImageNet比赛的胜利，比浅层学习方法有显著优势。

2014, Ian Goodfellow et al 伊恩·古德菲勒等

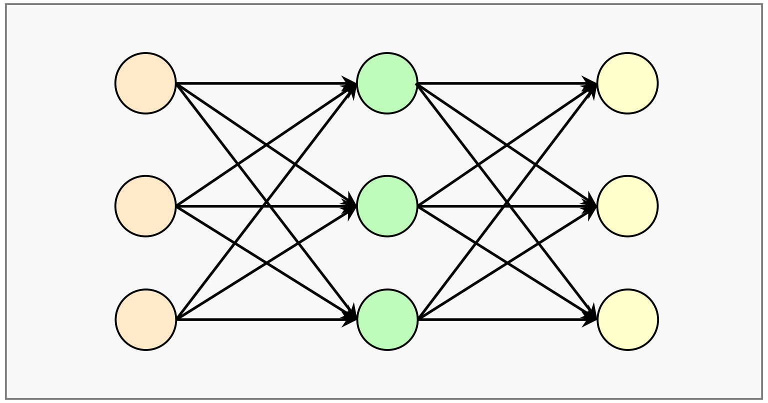
- Proposed **generative adversarial network** (GAN) which has two neural networks competing against each other in a zero-sum game framework.
  提出了**生成对抗网络** (GAN)，其中有两个神经网络，彼此以“零和”博弈方式相互竞争。

Structures of Neural Networks 神经网络的结构



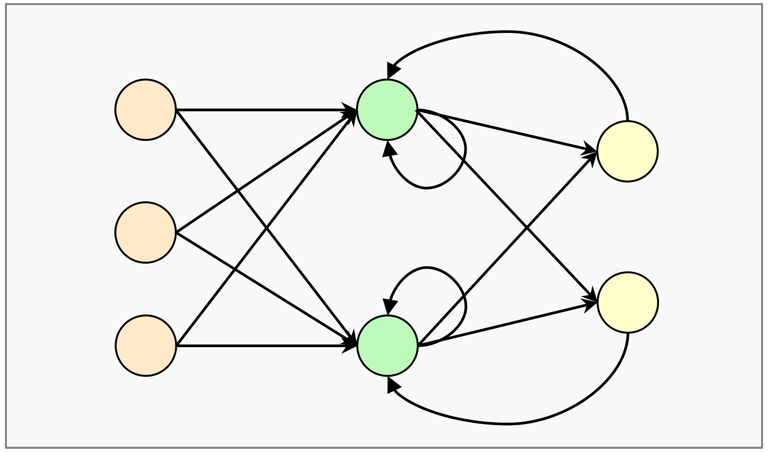
Structures of Neural Network Models 神经网络模型的结构

- Feedforward neural network 前馈神经网络
  - information moves in only one direction, forward, from input nodes, through hidden nodes and to the output nodes.
    信息从输入结点仅仅以一个方向，即前进方向，穿过隐藏层并抵达输出节点。



Feedforward neural network 前馈神经网络

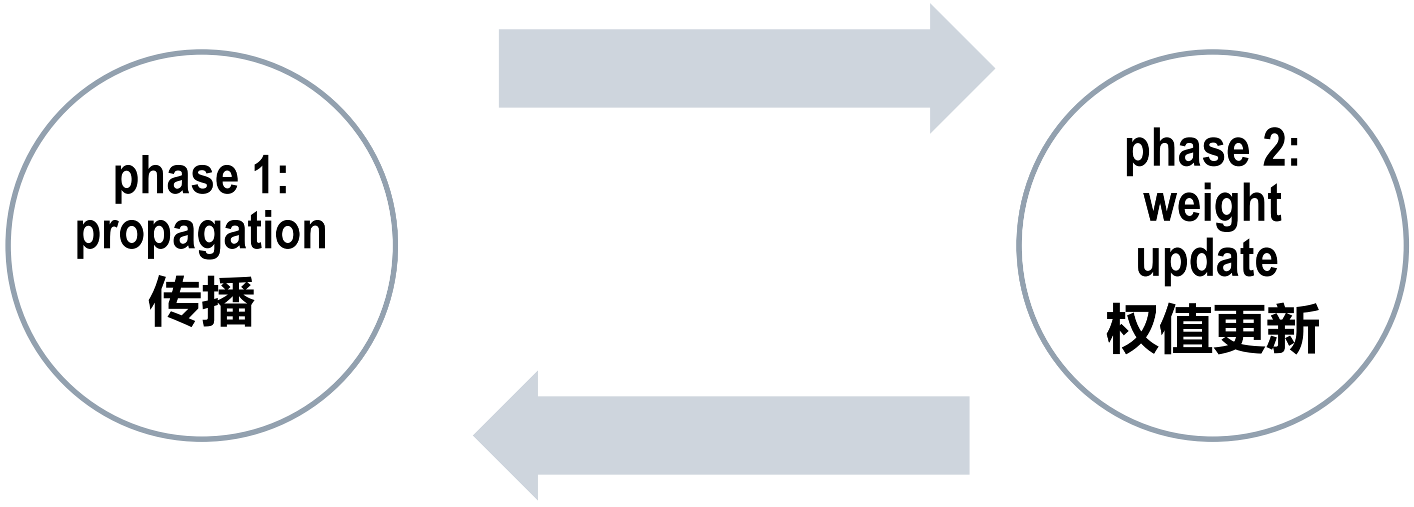
- Recurrent neural network 循环神经网络
  - connections form a directed cycle.
    连接形成有向循环。
  - creating an internal state of the network which allows it to exhibit dynamic temporal behavior.
    建立网络的内部状态，使之展现动态的时间特性。



Recurrent neural network 循环神经网络

Back-propagation 反向传播

- Back-propagation (BP) is an abbreviation for "**Backward propagation of errors**".
  反向传播 (BP) 是"**反向误差传播**"的缩略语。
- It is a common method of training Artificial Neural Networks, and used in conjunction with an optimization method such as gradient descent.
  是训练人工神经网络的常用方法，与梯度下降优化方法结合使用。
- The algorithm repeats a two phase cycle: 该算法重复两个阶段的循环：



Repeat phase 1 and phase 2 until the performance of the network is satisfactory.
重复阶段1和阶段2的操作，直到网络的性能得到满足。

Algorithm of Back-propagation 反向传播算法

☐ Phase 1: **Propagation** 第1阶段：传播

- **Feedforward propagation** 前馈传播

  the input of training data through the neural network in order to generate *output* **activations**.

  输入的训练数据穿过神经网络，从而生成输出**激活值**。

- **Back-propagation** 反向传播

  the output activations through the neural network using the training data target in order to generate the **deltas** of all output and hidden neurons.

  输出激活再使用训练数据目标穿过神经网络，生成所有的输出层和隐藏层神经元的差值。

  - deltas = expected output - actual output values

    差值 = 期待输出 - 实际输出

Algorithm of Back-propagation 反向传播算法

☐ Phase 2: **Weight update** 第2阶段：权值更新

For each weight: 对每个权值：

- Multiply its output delta and input activation, to get the gradient of the weight.
  将其输出差值与输入激活相乘，以便得到该权值梯度。
- Subtract a ***ratio*** (percentage) of the gradient from the weight. The ratio is called *learning rate*.
  从权值中减去梯度的比值（百分比）。该比值被称为学习率。
  - The greater the ratio, the faster the neuron trains;
    比值越大，神经元训练越快；
  - the lower the ratio, the more accurate the training is.
    比值越低，训练精度越高。

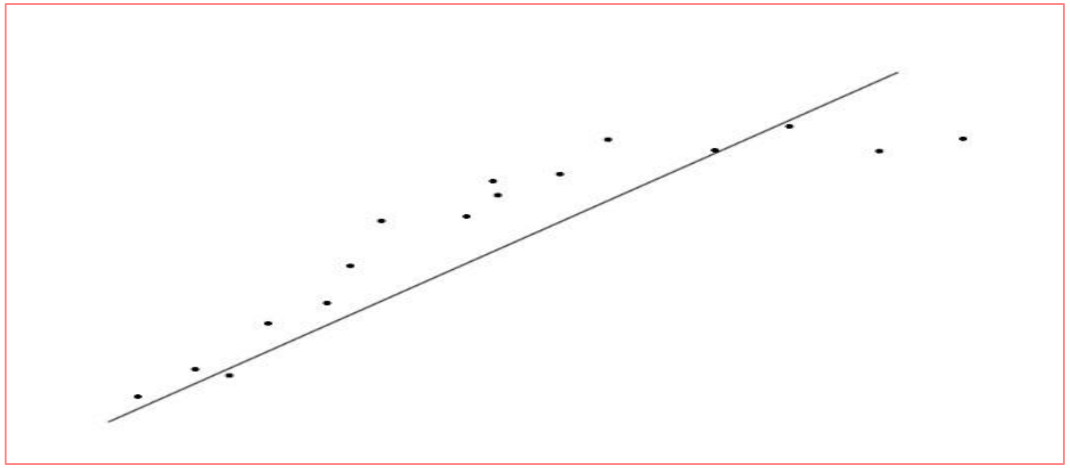
A Stochastic Gradient Descent Algorithm 随机梯度下降算法

```
function STOCHASTIC-GRADIENT-DESCENT() return the network
    initialize network weights (often small random values)
    do
        for each training example named ex
            prediction = neural-net-output(network, ex)  // forward pass
            actual = teacher-output(ex)
            compute error (prediction - actual) at the output units
            compute Δw_h for all weights from hidden layer to output layer  // backward pass
            compute Δw_i for all weights from input layer to hidden layer
            update network weights  // input layer not modified by error estimate
    until all examples classified correctly or another stopping criterion satisfied
    return the network
```

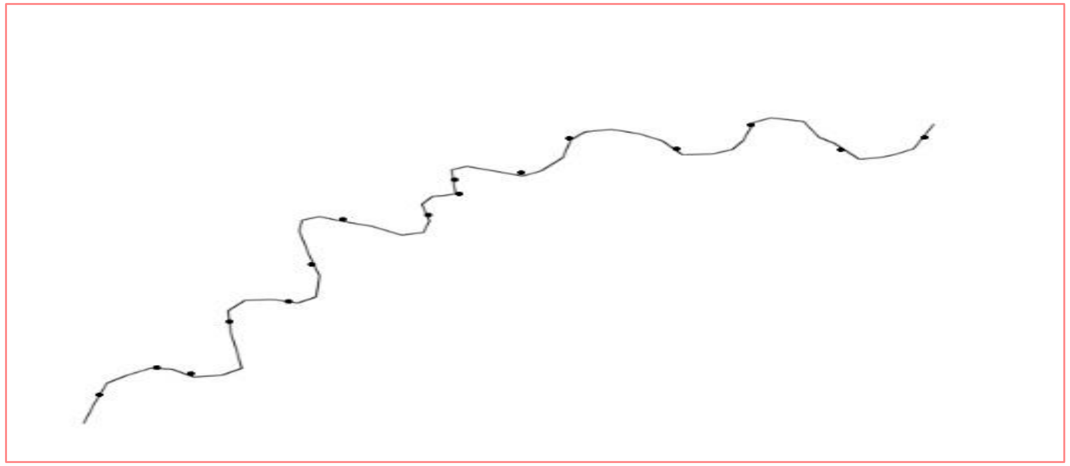
For training a three-layer network (only one hidden layer)

用于训练一个三层网络（仅有一个隐藏层）

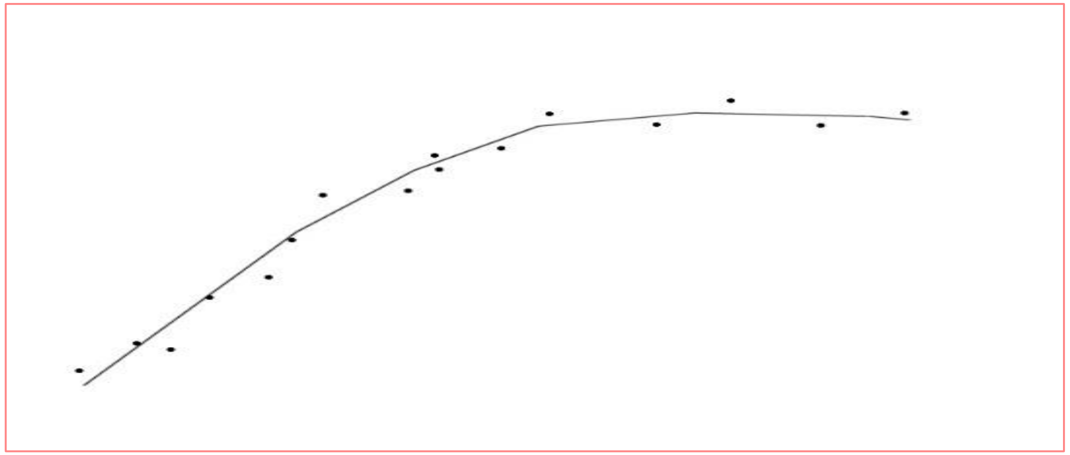
Comparison of Training Results 训练结果的比较



(a) Under-fit of the data
数据低拟合



(b) Over-fit of the data
数据过拟合

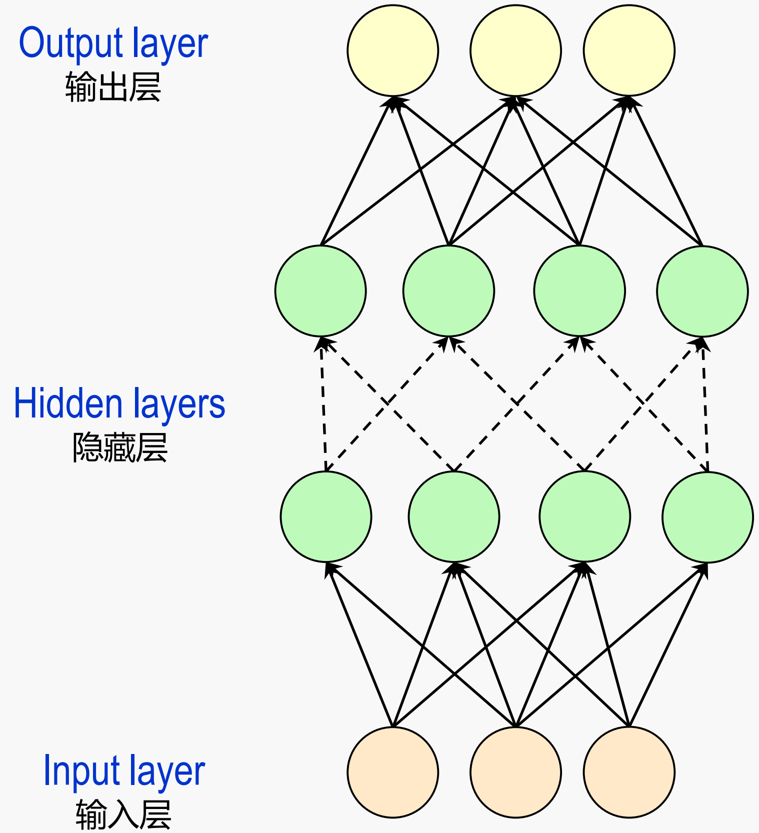


(c) Good fit of the data
数据良拟合

Shallow vs. Deep Neural Network 浅层与深层神经网络

- There is no universally agreed upon threshold of depth dividing shallow neural networks from deep neural networks.
  就划分浅层神经网络与深层神经网络的深度而言，尚未有公认的观点。

- But most researchers agree that **deep neural networks have more than 2 of hidden layers, and hidden layers > 10 to be very deep neural networks**.
  但大多数研究人员认为，**深度神经网络的隐藏层超过2、而隐藏层大于10的为超深度神经网络**。

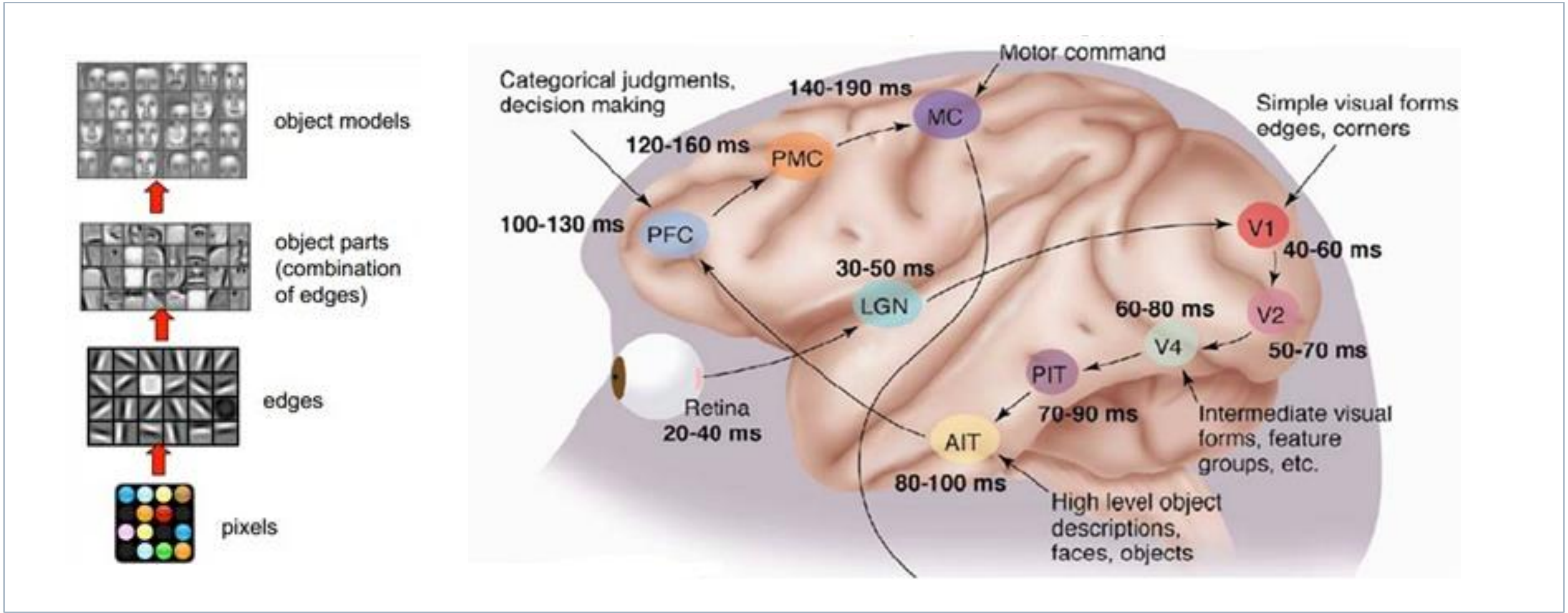


Artificial Neural Network
人工神经网络

Contents:

Why Deep Hierarchy 为什么深度层次

- Biological: Visual cortex is Deep Hierarchical
  生物学：视觉皮层是深层次的



Deep Neural Networks (DNNs) 深度神经网络

- DNNs use many layers of nonlinear processing units for **feature extraction and transformation**.

  DNNs使用许多层非线性处理单元，用于**特征提取和转换**。

- Able to learn multiple levels of features or representations of the data. Higher level features are derived from lower level features.

  能够学习数据的**多层特征或表征**。高层特征来自于低层特征。

- Be part of the broader machine learning field: **learning representations of data**.

  成为更广泛的机器学习领域的一部分：**学习数据表征**。

- Learning multiple levels of representations that correspond to different levels of abstraction; the levels form a hierarchy of concepts.

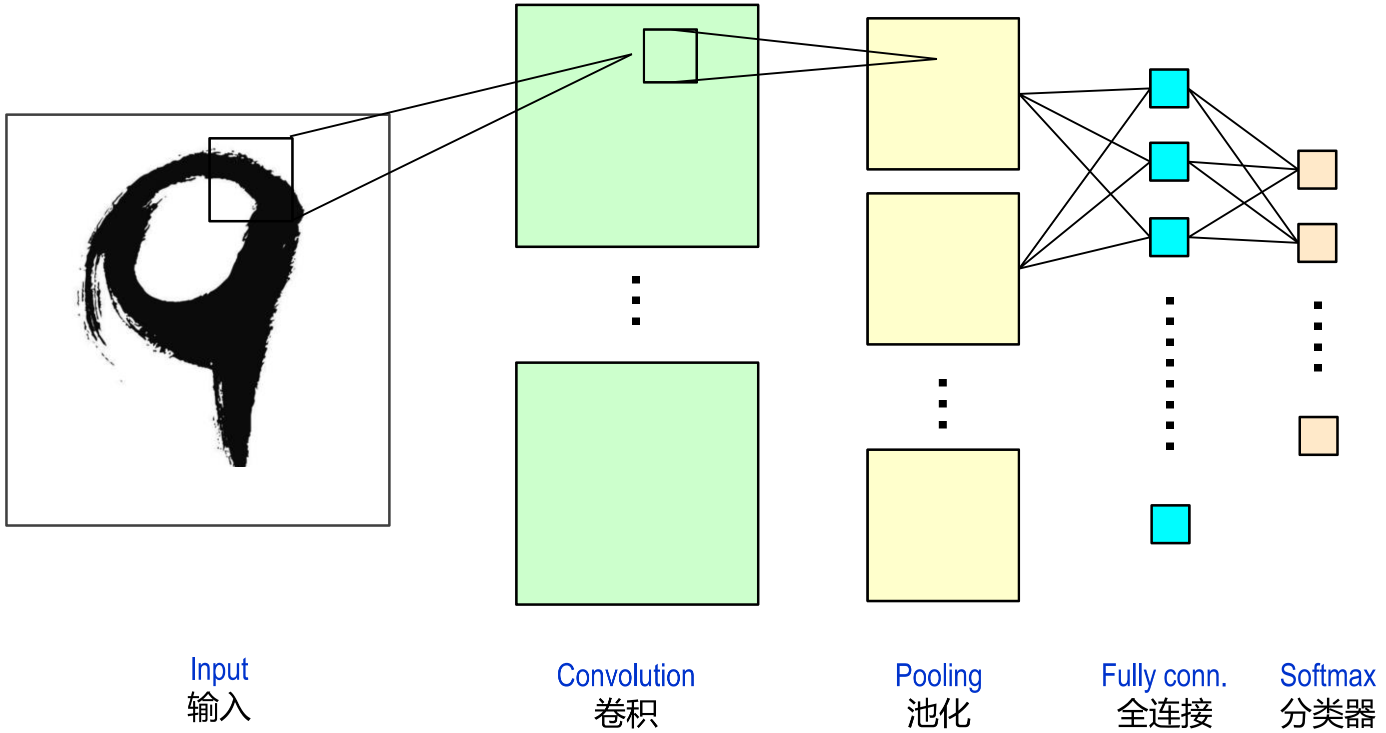
  学习多层级表征，对应于不同的抽象层级；这种层级形成了一种概念的层次结构。

Typical Deep Neural Networks 代表性的深度神经网络

Deep belief networks (DBN)	深度信念网络 (DBN)
Convolutional neural networks (CNN)	卷积神经网络 (CNN)
Deep Boltzmann machines (DBM)	深度波兹曼机 (DBM)
Recurrent neural networks (RNN)	循环网络 (RNN)
Long short-term memory (LSTM)	长短期记忆 (LSTM)
Auto-Encoders	自动编码器
Generative Adversarial Network (GAN)	生成对抗网络 (GAN)

Case Study: Convolutional neural network (CNN) 卷积神经网络

- CNN is a type of feed-forward artificial neural network that uses at least one of convolution in place of general matrix multiplication.
  CNN是一种前馈式人工神经网络，使用至少一个卷积层来代替一般的矩阵乘法。
- Four key ideas: 四个关键思想：
  - local connections (convolution)
    局部连接（卷积）
  - shared weights
    共享权值
  - pooling (sampling)
    池化（采样）
  - many layers.
    多层



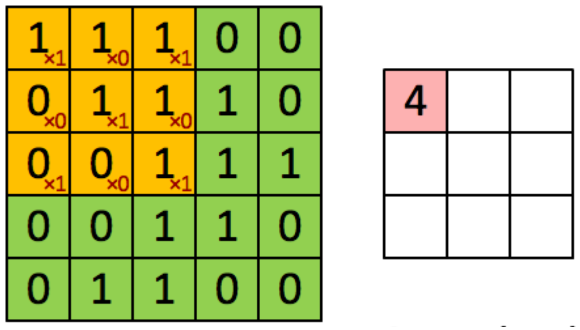
Case Study: Convolutional neural network (CNN) 卷积神经网络

☐ Convolution layer 卷积层

Consist of a set of learnable filters, each filter is convolved across the width and height of the input volume, computing the *dot product* between the entries of the filter and the input, and producing a 2-dimensional activation map of that filter.

包含一组学习滤波器，每个滤波器对输入的宽和高进行卷积，计算滤波器和输入之间的点积，生成一个该滤波器的2维活动图。

Source: http://deeplearning.net/tutorial/lenet.html

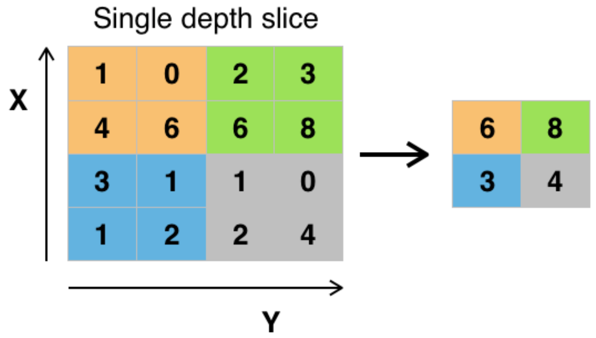


Convolution with a 3x3 filter
用3x3滤波器进行卷积

☐ Pooling layer 池化层

A form of non-linear down-sampling. It partitions the input image into a set of non-overlapping rectangles and, for each such sub-region, outputs the maximum.

是一种非线性下采样。它将输入图像分割成一组不重叠的矩形，对每个这样的子区域，再产出其最大值。



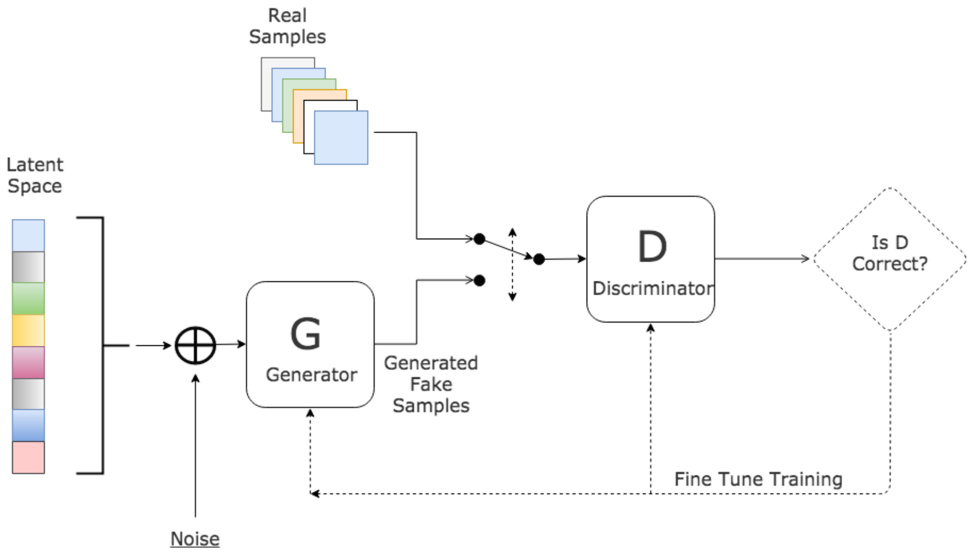
Max pooling with a 2x2 filter and stride = 2
用2x2滤波器进行最大池化，步长=2

Case Study: Generative Adversarial Network (GAN) 生成对抗网络

- GAN is pioneered by Ian Goodfellow et al at University of Montreal in 2014.
  GAN是由Goodfellow等人于2014年在蒙特利尔大学开创的。

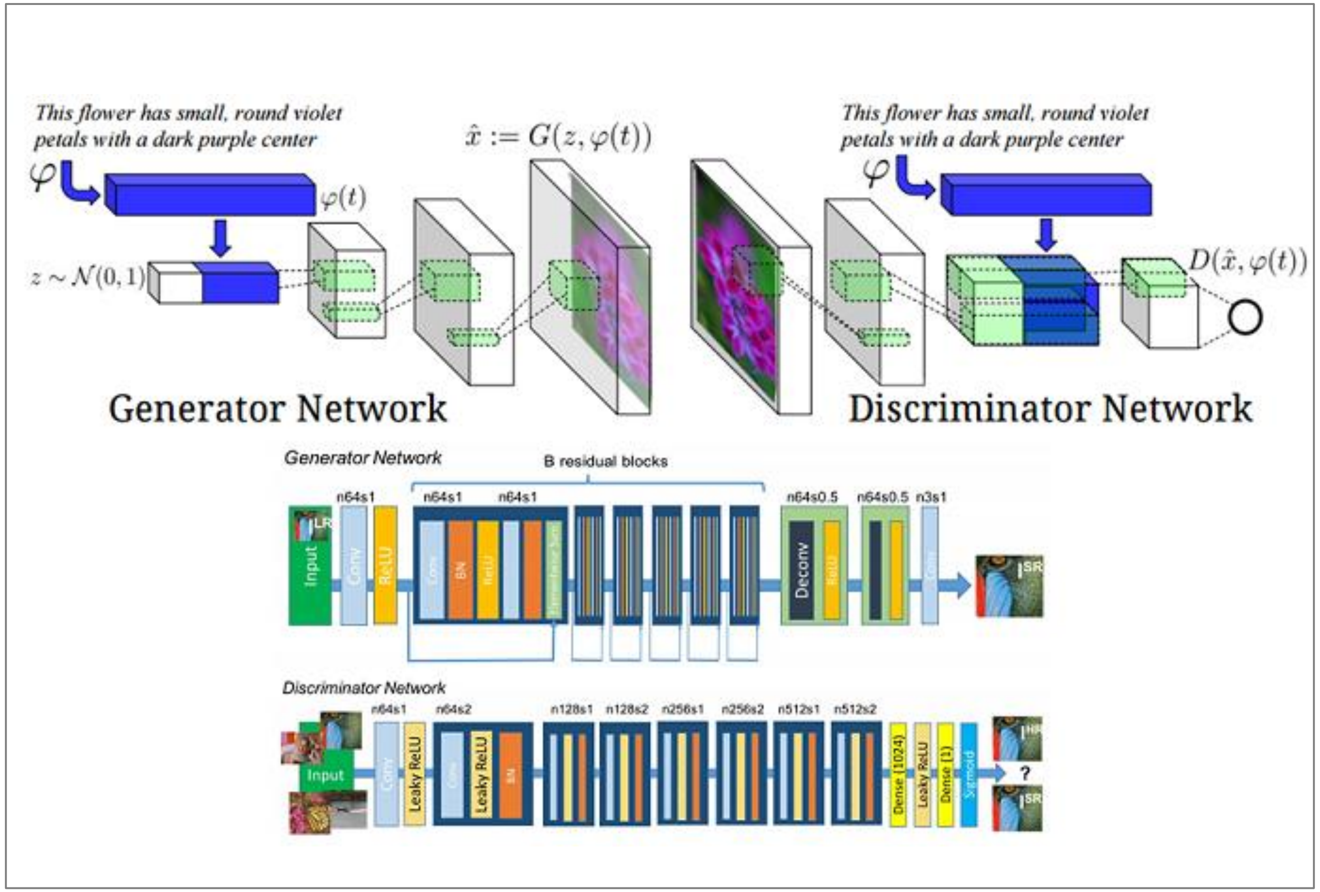
Source: Goodfellow @NIPS 2016 Workshop

- Adversarial Network, inspired by *Adversarial game.*
  对抗网络，灵感源于对抗博弈。
- Generator maps from a latent space to a particular data distribution of interest.
  生成器从潜在空间映射到所关注的特定数据分布。
- Discriminator discriminate between real samples and samples produced by generator.
  判别器在真实样本和生成样本之间进行判别。



Training a model in a worst-case scenario, with inputs chosen by an adversary.
在最坏场景下训练模型，由adversary选择输入。

Case Study: Generative Adversarial Network (GAN) 生成对抗网络



Generative Adversarial Network (GAN)

生成对抗网络

The generator worked well with digits (a) and faces (b), but it created very fuzzy and vague images (c) and (d) when using the CIFAR-10 dataset.

该生成器对数字(a)和人脸(b)效果好，但使用CIFAR-10数据集时，却生成了非常模糊和含糊的图像(c)和(d)。

Typical Applications of Deep Neural Networks 深度神经网络的主要应用

- Speech recognition 语音识别
- Object recognition 物体识别
- Image retrieval 图像检索
- Image understanding 图像理解
- Natural language processing 自然语言处理
- Recommendation systems 推荐系统
- Drug discovery 药物发现
- Biomedical informatics 生物医学信息学

Postscript 后记

Deep Learning is a new area of Machine Learning research, which has been introduced with the objective of moving Machine Learning closer to one of its original goals: Artificial Intelligence.

深度学习是机器学习研究的一个新领域，其研究目的是使机器学习更接近其原始目标之一：**人工智能。**

Source: http://deeplearning.net/

Thank you for your attention!

AI