

机器学习简史及分类

人工智能、机器学习、深度学习 区分

1、机器学习简史

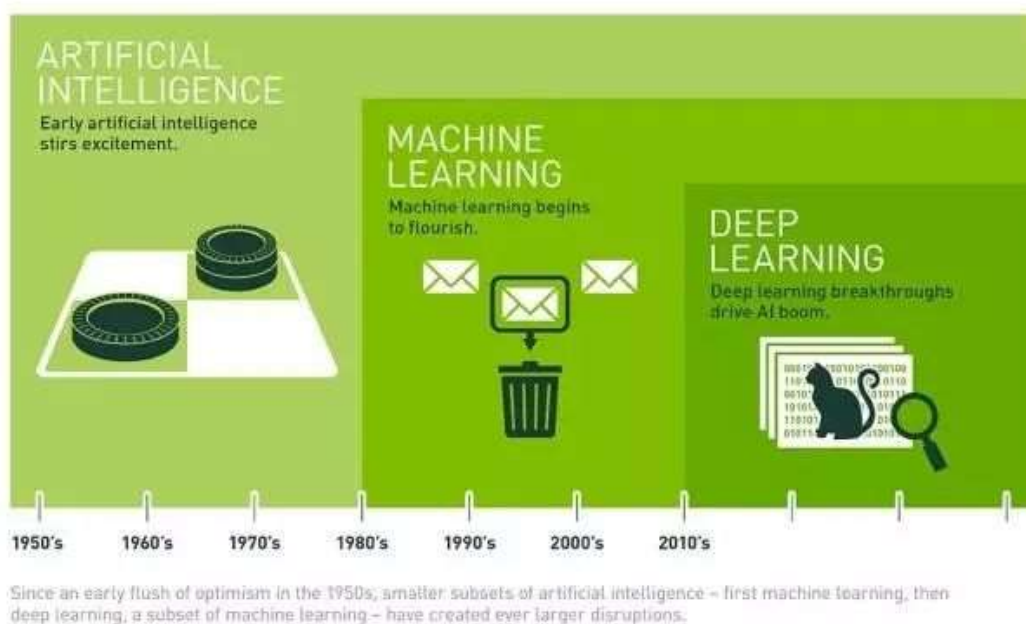
- 1.1 诞生并奠定基础时期 - 感知机, KNN
- 1.2 停滞不前的瓶颈时期
- 1.3 希望之光重新点亮 - MLP, BP, DT
- 1.4 现代机器学习的成型时期 - Boosting, SVM, RF
- 1.5 爆发时期 - Deep Learning
- 1.6 启示与未来的发展

2、机器学习基本概念

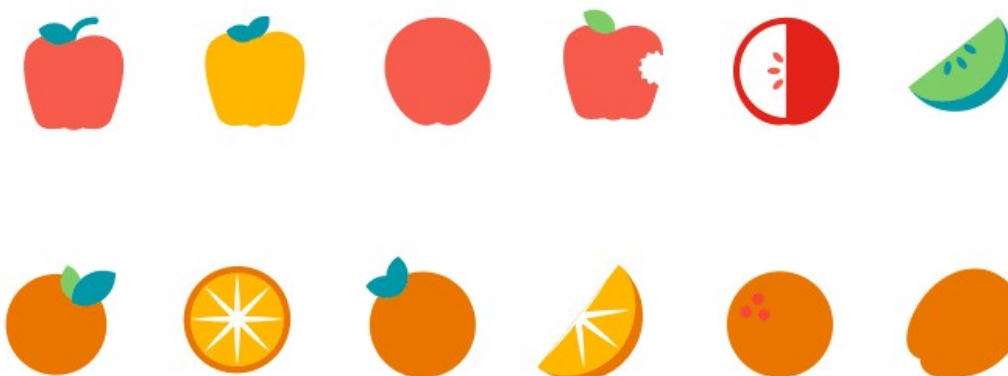
3、机器学习分类

- 3.1、监督式学习 Supervised Learning
 - 分类, 回归, 排序, 匹配学习
- 3.2 半监督学习 - 深度学习
- 3.3 无监督学习 Unsupervised Learning
 - 聚类, 关联规则
- 3.4 增强学习 Reinforcement Learning
- 3.5 其他 - 集成算法

有三个词，这两年出现的频率越来越高：人工智能（AI），机器学习（ML），深度学习（DL），到底他们哥仨是什么关系？今天小编化繁为简，用几张图带你迅速看明白。关系如图，从提出概念的时间上来看也很清楚：

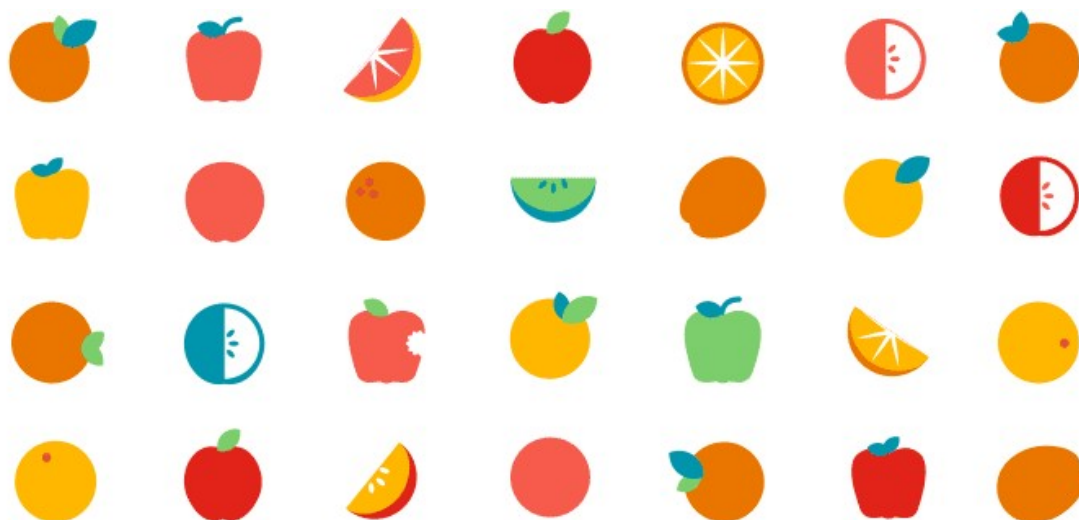


人工智能AI：模拟人脑，辨认哪个是苹果，哪个是橙子。



机器学习ML：根据特征在水果摊买橙子，随着见过的橙子和其他水果越来越多，辨别橙子的能力越来越强，不会再把香蕉当橙子。

机器学习强调“学习”而不是程序本身，通过复杂的算法来分析大量的数据，识别数据中的模式，并做出一个预测--不需要特定的代码。在样本的数量不断增加的同时，自我纠正完善“学习目的”，可以从自身的错误中学习，提高识别能力。



深度学习DL：超市里有3种苹果和5种橙子，通过数据分析比对，把超市里的品种和数据建立联系，通过水果的颜色、形状、大小、成熟时间和产地等信息，分辨普通橙子和血橙，从而选择购买用户需要的橙子品种。

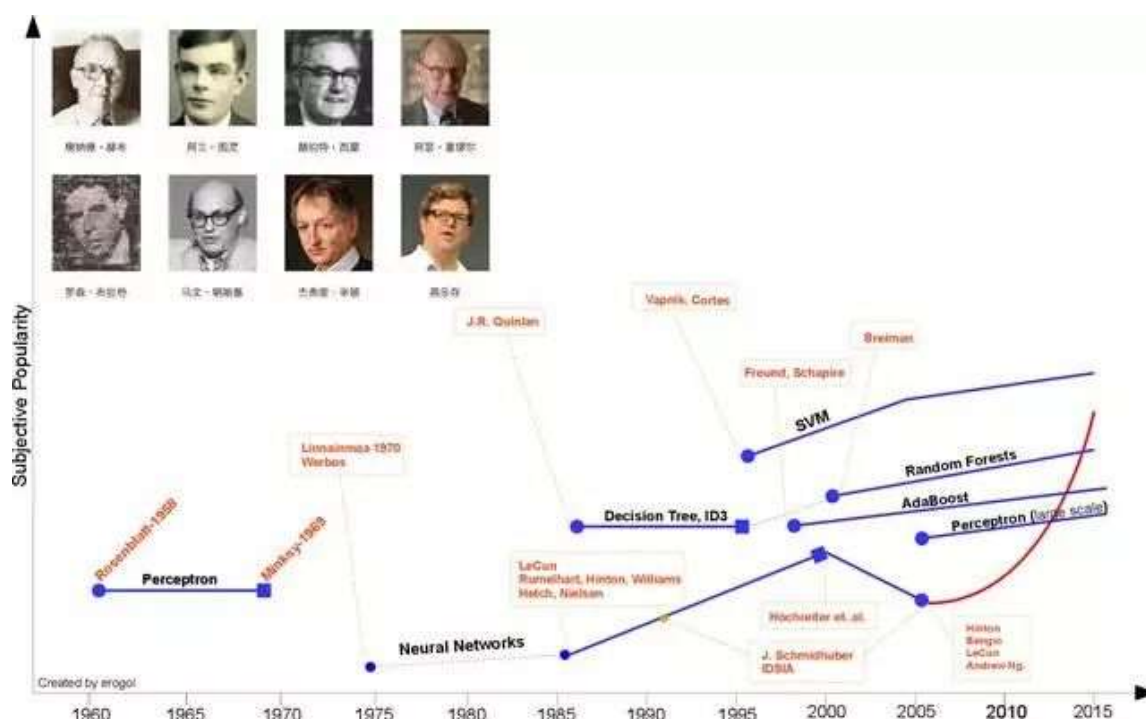
1、机器学习简史

哲学上的三大终极问题：是谁？从哪来？到哪去？用在任何地方都是有意义的。

——尼古拉斯·沃布吉·道格拉斯·硕德

虽然人工智能并不是最近几年才兴起的，但在它一直作为科幻元素出现在大众视野中。自从

AlphaGo战胜李世石之后，人工智能突然间成了坊间谈资，仿佛人类已经造出了超越人类智慧的机器。而人工智能的核心技术机器学习及其子领域深度学习一时间成了人们的掌上明珠。面对这个从天而降的“怪物”，乐观者有之，悲观者亦有之。但追溯历史，我们会发现机器学习的技术爆发有其历史必然性，属于技术发展的必然产物。而理清机器学习的发展脉络有助于我们整体把握机器学习，或者人工智能的技术框架，有助于从“道”的层面理解这一技术领域。这一节就先从三大究极哲学问题中的后两个——从哪来、到哪去入手，整体把握机器学习，而后再从“术”的角度深入学习，解决是谁的问题。（本来想加个纵时间轴，无奈查了半天没找到方法，如果有人知道，求分享）



图一 机器学习发展史（图片来源：Brief History of Machine Learning）

1.1 诞生并奠定基础时期

1949, Hebb, Hebbian Learning theory

赫布于1949年基于神经心理的提出了一种学习方式，该方法被称之为赫布学习理论。大致描述为：

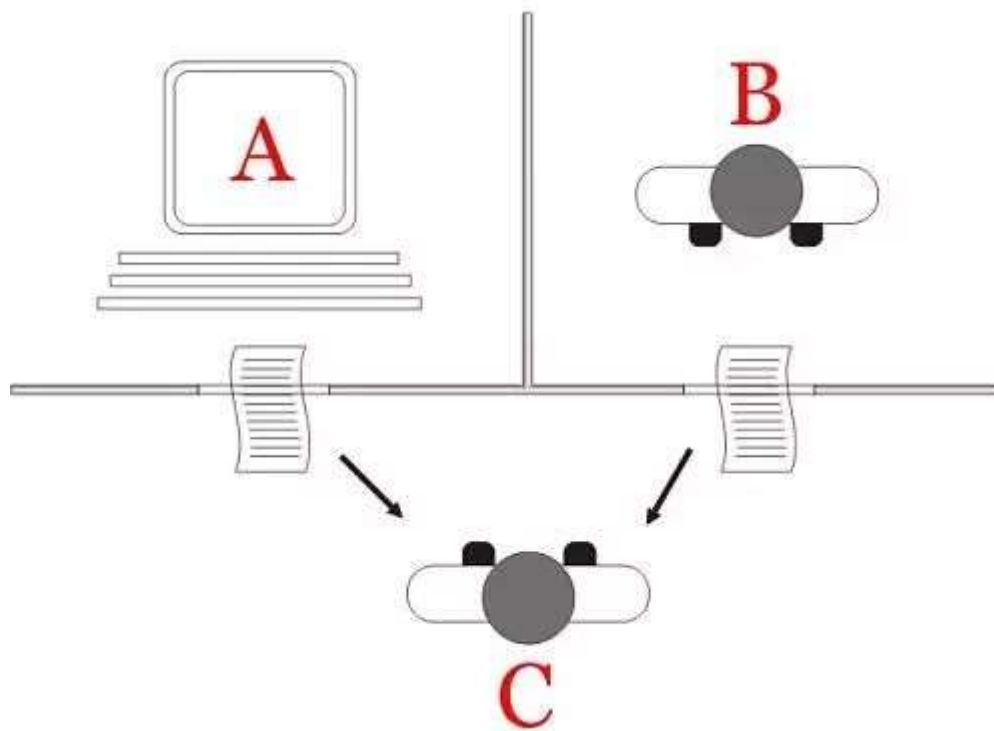
假设反射活动的持续性或反复性会导致细胞的持续性变化并增加其稳定性，当一个神经元A能

持续或反复激发神经元B时，其中一个或两个神经元的生长或代谢过程都会变化。

Let us assume that the persistence or repetition of a reverberatory activity (or “trace”) tends to induce lasting cellular changes that add to its stability.... When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased

从人工神经元或人工神经网络角度来看，该学习理论简单地解释了循环神经网络（RNN）中结点之间的相关性关系（权重），即：当两个节点同时发生变化（无论是positive还是negative），那么节点之间有很强的正相关性（positive weight）；如果两者变化相反，那么说明有负相关性（negative weight）。

1950, Alan Turing, The Turing test



图二 图灵测试（图片来源：维基百科）

1950年，阿兰·图灵创造了图灵测试来判定计算机是否智能。图灵测试认为，如果一台机器能够与人类展开对话（通过电传设备）而不能被辨别出其机器身份，那么称这台机器具有智能。这一简化使得图灵能够令人信服地说明“思考的机器”是可能的。

2014年6月8日，一个叫做尤金·古斯特曼的聊天机器人成功让人类相信它是一个13岁的男孩，成为有史以来首台通过图灵测试的计算机。这被认为是人工智能发展的一个里程碑事件。

1952, Arthur Samuel, “Machine Learning”



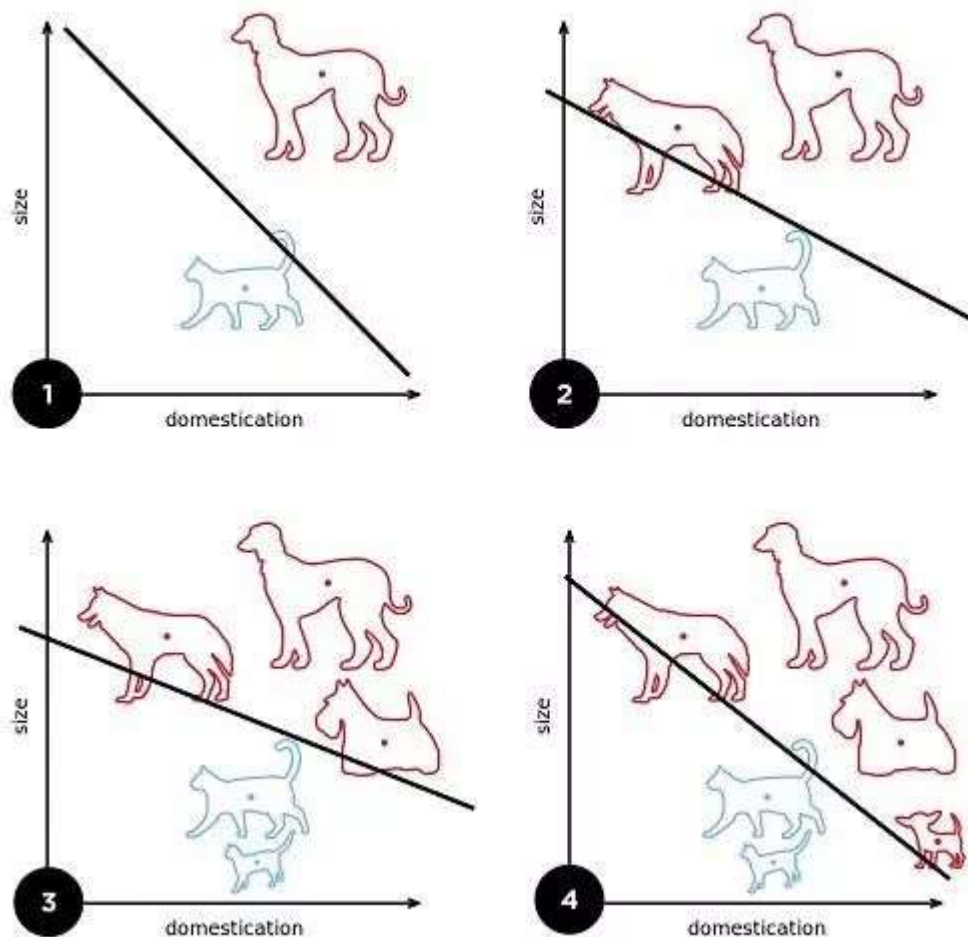
图三 塞缪尔（图片来源：Brief History of Machine Learning）

1952，IBM科学家亚瑟·塞缪尔开发了一个跳棋程序。该程序能够通过观察当前位置，并学习一个隐含的模型，从而为后续动作提供更好的指导。塞缪尔发现，伴随着该游戏程序运行时间的增加，其可以实现越来越好的后续指导。通过这个程序，塞缪尔驳倒了普罗维登斯提出的机器无法超越人类，像人类一样写代码和学习的模式。他创造了“机器学习”这一术语，并将它定义为：

可以提供计算机能力而无需显式编程的研究领域

a field of study that gives computer the ability without being explicitly programmed.

1957, Rosenblatt, Perceptron



图四 感知机线性分类器（图片来源：维基百科）

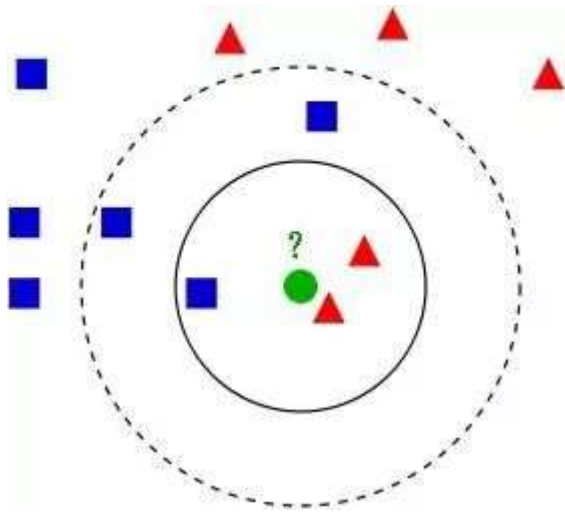
1957年，罗森·布拉特基于神经感知科学背景提出了第二模型，非常的类似于今天的机器学习模型。这在当时是一个非常令人兴奋的发现，它比赫布的想法更适用。基于这个模型罗森·布拉特设计出了第一个计算机神经网络——感知机（the perceptron），它模拟了人脑的运作方式。罗森·布拉特对感知机的定义如下：

感知机旨在说明一般智能系统的一些基本属性，它不会因为个别特例或通常不知道的东西所束缚住，也不会因为那些个别生物有机体的情况而陷入混乱。

The perceptron is designed to illustrate some of the fundamental properties of intelligent systems in general, without becoming too deeply enmeshed in the special, and frequently unknown, conditions which hold for particular biological organisms.

3年后，维德罗首次使用Delta学习规则（即最小二乘法）用于感知器的训练步骤，创造了一个良好的线性分类器。

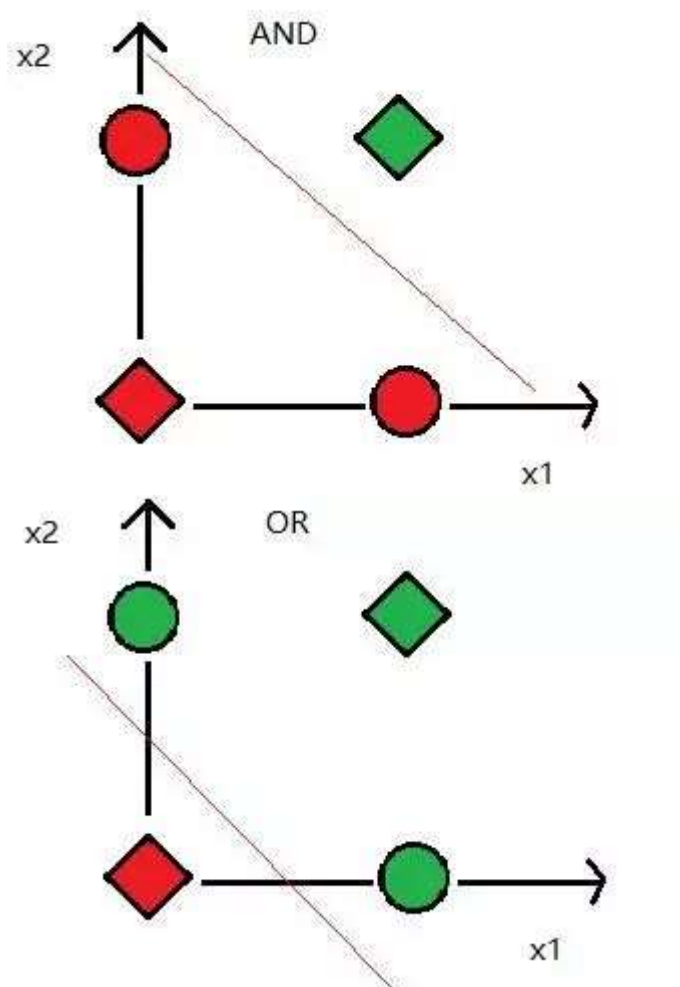
1967年，The nearest neighbor algorithm

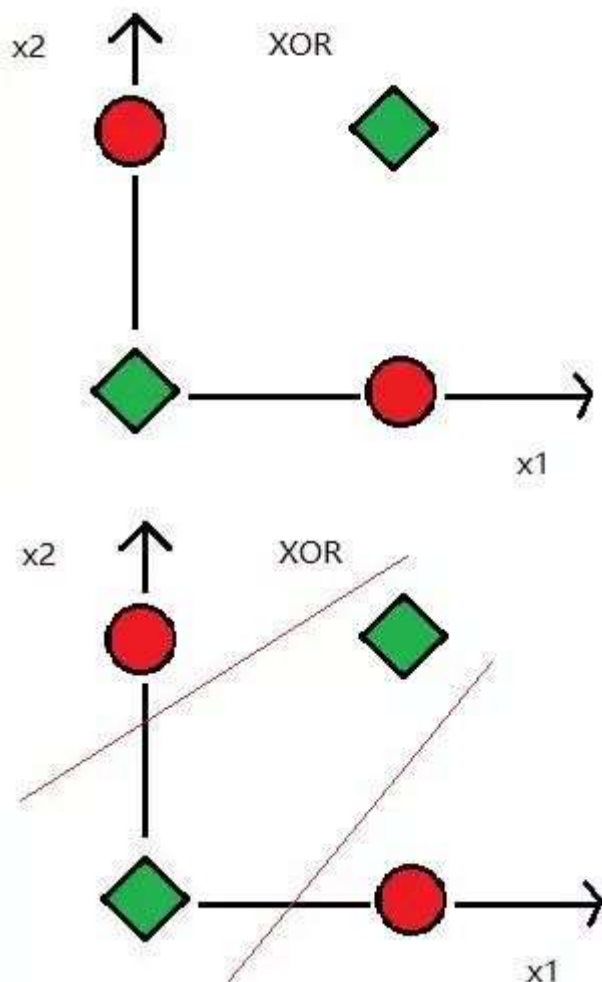


图五 kNN算法 (图片来源: 维基百科)

1967年, 最近邻算法 (The nearest neighbor algorithm) 出现, 使计算机可以进行简单的模式识别。kNN算法的核心思想是如果一个样本在特征空间中的 k 个最相邻的样本中的大多数属于某一个类别, 则该样本也属于这个类别, 并具有这个类别上样本的特性。这就是所谓的“少数听从多数”原则。

1969, Minsky, XOR problem





图六 XOR问题，数据线性不可分

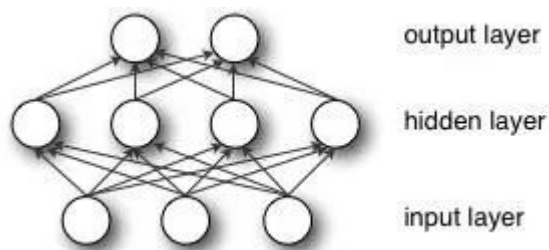
1969年马文·明斯基提出了著名的XOR问题,指出感知机在线性不可分的数据分布上是失效的。此后神经网络的研究者进入了寒冬,直到1980年才再一次复苏。

1.2 停滞不前的瓶颈时期

从60年代中到70年代末,机器学习的发展步伐几乎处于停滞状态。无论是理论研究还是计算机硬件限制,使得整个人工智能领域的发展都遇到了很大的瓶颈。虽然这个时期温斯顿(Winston)的结构学习系统和海斯·罗思(Hayes Roth)等的基于逻辑的归纳学习系统取得较大的进展,但只能学习单一概念,而且未能投入实际应用。而神经网络学习机因理论缺陷也未能达到预期效果而转入低潮。

1.3 希望之光重新点亮

1981, Werbos, Multi-Layer Perceptron (MLP)

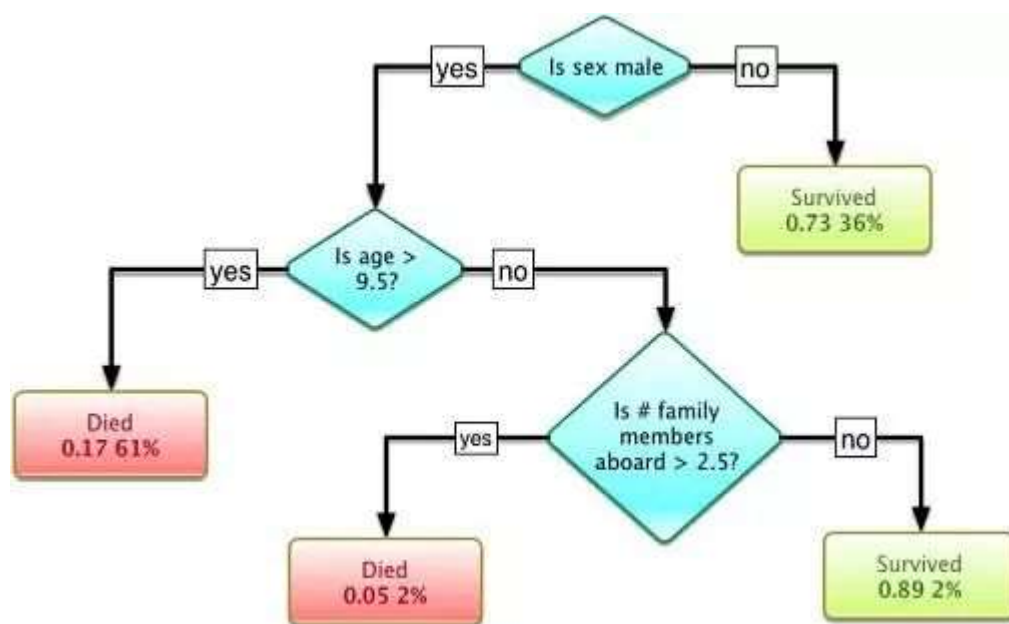


图七 多层感知机（或者人工神经网络）

伟博斯在1981年的神经网络反向传播(BP)算法中具体提出多层感知机模型。虽然BP算法早在1970年就已经以“自动微分的反向模型 (reverse mode of automatic differentiation)”为名提出来了，但直到此时才真正发挥效用，并且直到今天BP算法仍然是神经网络架构的关键因素。有了这些新思想，神经网络的研究又加快了。

在1985-1986年，神经网络研究人员（鲁梅尔哈特，辛顿，威廉姆斯-赫，尼尔森）相继提出了使用BP算法训练的多参数线性规划（MLP）的理念，成为后来深度学习的基石。

1986, Quinlan, Decision Trees



图八 决策树（图片来源：维基百科）

在另一个谱系中，昆兰于1986年提出了一种非常出名的机器学习算法，我们称之为“决策树”，更具体的说是ID3算法。这是另一个主流机器学习算法的突破点。此外ID3算法也被发布成为了一款软件，它能以简单的规划和明确的推论找到更多的现实案例，而这一点正好和神经网络黑箱模型相反。

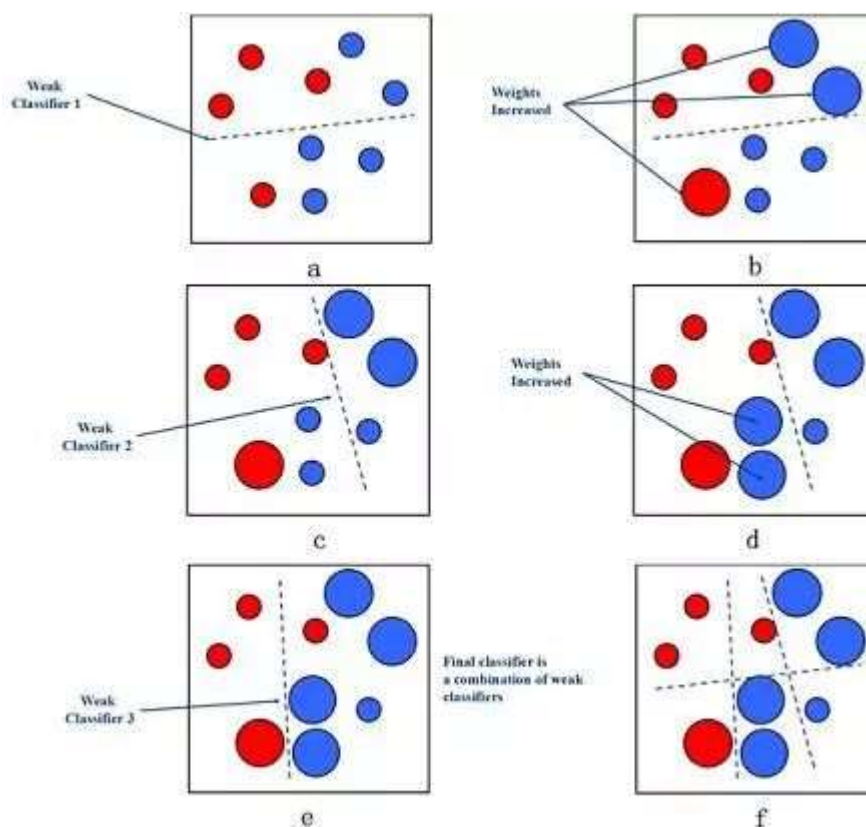
决策树是一个预测模型，他代表的是对象属性与对象值之间的一种映射关系。树中每个节点表示某个对象，而每个分叉路径则代表的某个可能的属性值，而每个叶结点则对应从根节点到该

叶节点所经历的路径所表示的对象的值。决策树仅有单一输出，若欲有复数输出，可以建立独立的决策树以处理不同输出。数据挖掘中决策树是一种经常要用到的技术，可以用于分析数据，同样也可以用来作预测。

在ID3算法提出来以后，研究社区已经探索了许多不同的选择或改进（如ID4、回归树、CART算法等），这些算法仍然活跃在机器学习领域中。

1.4 现代机器学习的成型时期

1990, Schapire, Boosting



图九 Boosting算法（图片来源：百度百科）

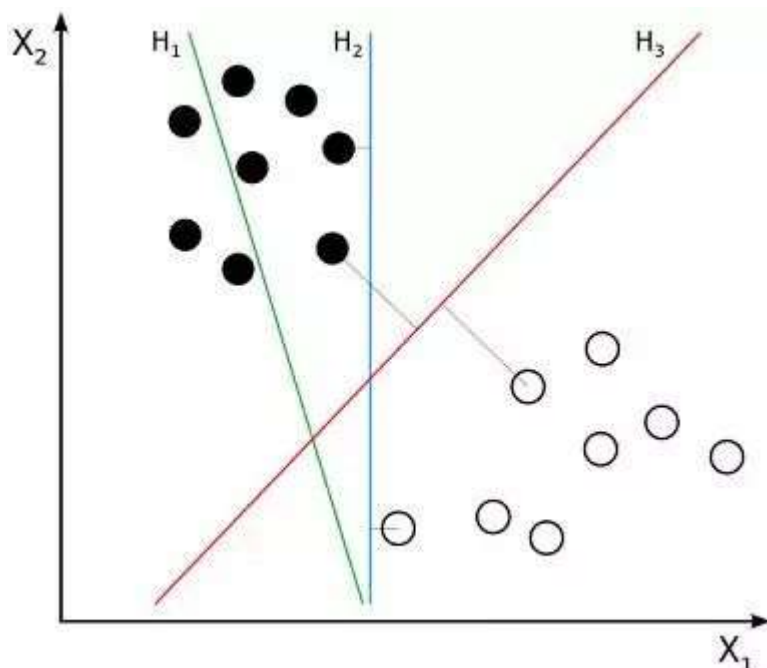
1990年, Schapire最先构造出一种多项式级的算法，这就是最初的Boosting算法。一年后, Freund提出了一种效率更高的Boosting算法。但是, 这两种算法存在共同的实践上的缺陷，那就是都要求事先知道弱学习算法学习正确的下限。

1995年， Freund和Schapire改进了Boosting算法，提出了 AdaBoost (Adaptive Boosting) 算法, 该算法效率和Freund于1991年提出的 Boosting算法几乎相同，但不需要任何关于弱学习器的先验知识，因而更容易应用到实际问题当中。

Boosting方法是一种用来提高弱分类算法准确度的方法, 这种方法通过构造一个预测函数系列, 然后以一定的方式将他们组合成一个预测函数。它是一种框架算法, 主要是通过通过对样本集的操作

作获得样本子集,然后用弱分类算法在样本子集上训练生成一系列的基分类器。

1995, Vapnik and Cortes, Support Vector Machines (SVM)



图十 支持向量机 (图片来源: 维基百科)

支持向量机的出现是机器学习领域的另一大重要突破, 该算法具有非常强大的理论地位和实证结果。那一段时间机器学习研究也分为NN和SVM两派。然而, 在2000年左右提出了带核函数的支持向量机后, SVM在许多以前由NN占据的任务中获得了更好的效果。此外, SVM相对于NN还能利用所有关于凸优化、泛化边际理论和核函数的深厚知识。因此SVM可以从不同的学科中大力推动理论和实践的改进。

而神经网络遭受到又一个质疑, 通过Hochreiter等人1991年和Hochreiter等人在2001年的研究表明在应用BP算法学习时, NN神经元饱和后会出现梯度损失 (gradient loss) 的情况。简单地说, 在一定数量的epochs训练后, NN会产生过拟合现象, 因此这一时期NN与SVM相比处于劣势。

2001, Breiman, Random Forests(RF)

决策树模型由布雷曼博士在2001年提出, 它是通过集成学习的思想将多棵树集成的一种算法, 它的基本单元是决策树, 而它的本质属于机器学习的一大分支——集成学习 (Ensemble Learning) 方法。随机森林的名称中有两个关键词, 一个是“随机”, 一个就是“森林”。“森林”我们很好理解, 一棵叫做树, 那么成百上千棵就可以叫做森林了, 这样的比喻还是很贴切的, 其实这也是随机森林的主要思想——集成思想的体现。

其实从直观角度来解释, 每棵决策树都是一个分类器 (假设现在针对的是分类问题), 那么对

于一个输入样本，N棵树会有N个分类结果。而**随机森林集成了所有的分类投票结果，将投票次数最多的类别指定为最终的输出，这就是一种最简单的 Bagging 思想。**

1.5 爆发时期

2006, Hinton, Deep Learning

在机器学习发展分为两个部分，浅层学习（Shallow Learning）和深度学习（Deep Learning）。浅层学习起源上世纪20年代人工神经网络的反向传播算法的发明，使得基于统计的机器学习算法大行其道，虽然这时候的人工神经网络算法也被称为多层感知机，但由于多层网络训练困难，通常都是只有一层隐含层的浅层模型。

神经网络研究领域领军者Hinton在2006年提出了神经网络Deep Learning算法，使神经网络的能力大大提高，向支持向量机发出挑战。2006年，机器学习领域的泰斗Hinton和他的学生Salakhutdinov在顶尖学术刊物《Science》上发表了一篇文章，开启了深度学习在学术界和工业界的浪潮。

这篇文章有两个主要的讯息：1) 很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，在这篇文章中，逐层初始化是通过无监督学习实现的。

2015年，为纪念人工智能概念提出60周年，LeCun、Bengio和Hinton推出了深度学习的联合综述。

深度学习可以让那些拥有多个处理层的计算模型来学习具有多层次抽象的数据的表示。这些方法在许多方面都带来了显著的改善，包括最先进的语音识别、视觉对象识别、对象检测和许多其它领域，例如药物发现和基因组学等。深度学习能够发现大数据中的复杂结构。它是利用BP算法来完成这个发现过程的。BP算法能够指导机器如何从前一层获取误差而改变本层的内部参数，这些内部参数可以用于计算表示。深度卷积网络在处理图像、视频、语音和音频方面带来了突破，而递归网络在处理序列数据，比如文本和语音方面表现出了闪亮的一面。

当前统计学习领域最热门方法主要有deep learning和SVM（support vector machine），它们是统计学习的代表方法。可以认为神经网络与支持向量机都源自于感知机。

神经网络与支持向量机一直处于“竞争”关系。**SVM应用核函数的展开定理，无需知道非线性映射的显式表达式；由于是在高维特征空间中建立线性学习机，所以与线性模型相比，不但几乎不增加计算的复杂性，而且在某种程度上避免了“维数灾难”。**而早先的神经网络算法比较容易过训练，大量的经验参数需要设置；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。

神经网络模型貌似能够实现更加艰难的任务，如目标识别、语音识别、自然语言处理等。但

是，应该注意的是，这绝对不意味着其他机器学习方法的终结。尽管深度学习的成功案例迅速增长，但是对这些模型的训练成本是相当高的，调整外部参数也是很麻烦。同时，SVM的简单性促使其仍然最为广泛使用的机器学习方式。

1.6 启示与未来的发展

人工智能机器学习是诞生于20世纪中叶的一门年轻的学科，它对人类的生产、生活方式产生了重大的影响，也引发了激烈的哲学争论。但总的来说，机器学习的发展与其他一般事物的发展并无太大区别，同样可以用哲学的发展的眼光来看待。

机器学习的发展并不是一帆风顺的，也经历了螺旋式上升的过程，成就与坎坷并存。其中大量的研究学者的成果才有了今天人工智能的空前繁荣，是量变到质变的过程，也是内因和外因的共同结果。

机器学习的发展诠释了多学科交叉的重要性和必要性。然而这种交叉不是简单地彼此知道几个名词或概念就可以的，是需要真正的融会贯通：

- 统计学家弗莱德曼早期从事物理学研究，他是优化算法大师，而且他的编程能力同样令人赞叹；
- 乔丹教授既是一流的计算机学家，又是一流的统计学家，而他的博士专业为心理学，他能够承担起建立统计机器学习的重任；
- 辛顿教授是世界最著名的认知心理学家和计算机科学家。虽然他很早就成就斐然，在学术界声名鹊起，但他依然始终活跃在一线，自己写代码。他提出的许多想法简单、可行又非常有效，被称为伟大的思想家。正是由于他的睿智和身体力行，深度学习技术迎来了革命性的突破。
- ...

深度学习的成功不是源自脑科学或认知科学的进展，而是因为大数据的驱动和计算能力的极大提升。可以说机器学习是由学术界、工业界、创业界（或竞赛界）等合力造就的。学术界是引擎，工业界是驱动，创业界是活力和未来。学术界和工业界应该有各自的职责和分工。学术界的职责在于建立和发展机器学习学科，培养机器学习领域的专门人才；而大项目、大工程更应该由市场来驱动，由工业界来实施和完成。

对于机器学习的发展前途，中科院数学与系统科学研究院陆汝铃老师在为南京大学周志华老师的《机器学习》一书作序时提出了六大问题，我觉得这些问题也许正是影响机器学习未来发展方向的基本问题，因此我摘录其中五个在此（有两个问题属于同一个主题，合并之）：

- 问题一：在人工智能发展早期，机器学习的技术内涵几乎全部是符号学习，可是从二十世纪九十年代开始，统计机器学习有一匹黑马横空出世，迅速压倒并取代了符号学习的地位。人们可能会问，符号学习是否被彻底忽略了？他还能成为机器学习的研究对象吗？它是否能继续在统计学习的阴影里苟延残喘？

第一种观点：退出历史舞台——没有人抱有这种想法。

第二种观点：统计学习和符号学习结合起来——王珏教授认为，现在机器学习已经到了一个转折点，统计学习要想进入一个更高级的形式，就应该和知识相结合，否则就会停留于现状而止步不前。

第三种观点：符号学习还有翻身之日——Chandrasekaran教授认为机器学习并不会回到“河西”，而是随着技术的进步逐渐转向基本的认知科学。

- 问题二：统计机器学习的算法都是基于样本数据独立同分布的假设，但自然界现象千变万化，哪里有那么多独立同分布？那么“独立同分布”条件对于机器学习来说是必需的吗？独立同分布的不存在一定是不可逾越的障碍吗？

迁移学习也许会给问题的解决带来一丝曙光？

- 问题三：近年来出现了一些新的动向，比如深度学习。但他们真的代表机器学习新的方向吗？

包括周志华老师在内的一些学者认为深度学习掀起的热潮大过它本身的贡献，在理论和技术上并没有太多的创新，只不过硬件技术的革命使得人们能采用原来复杂度很高的算法，从而得到更精细的结果。

- 问题四：机器学习研究出现以来，我们看到的主要是从符号方法到统计方法的演变，用到的数学主要是概率统计。但是今天数学之大，就像大海，难道只有统计方法适合于在机器学习方面的应用？

目前流行学习已经“有点意思了”，但数学理论的介入程度远远不够，有待更多数学家参与，开辟新的模式、理论和方法。

- 问题五：大数据时代的出现，有没有给机器学习带来本质性的影响？

大数据时代给机器学习带来了前所未有的机遇，但是同样的统计、采样方法相较以前有什么本质不同吗？又从量变过渡到质变吗？数理统计方法有没有发生质的变化？大数据时代正在呼吁什么样的机器学习方法？哪些方法又是大数据研究的驱动而产生的呢？

2、机器学习基本概念

权威定义：

Arthur samuel: 在不直接针对问题进行编程的情况下，赋予计算机学习能力的一个研究领域。

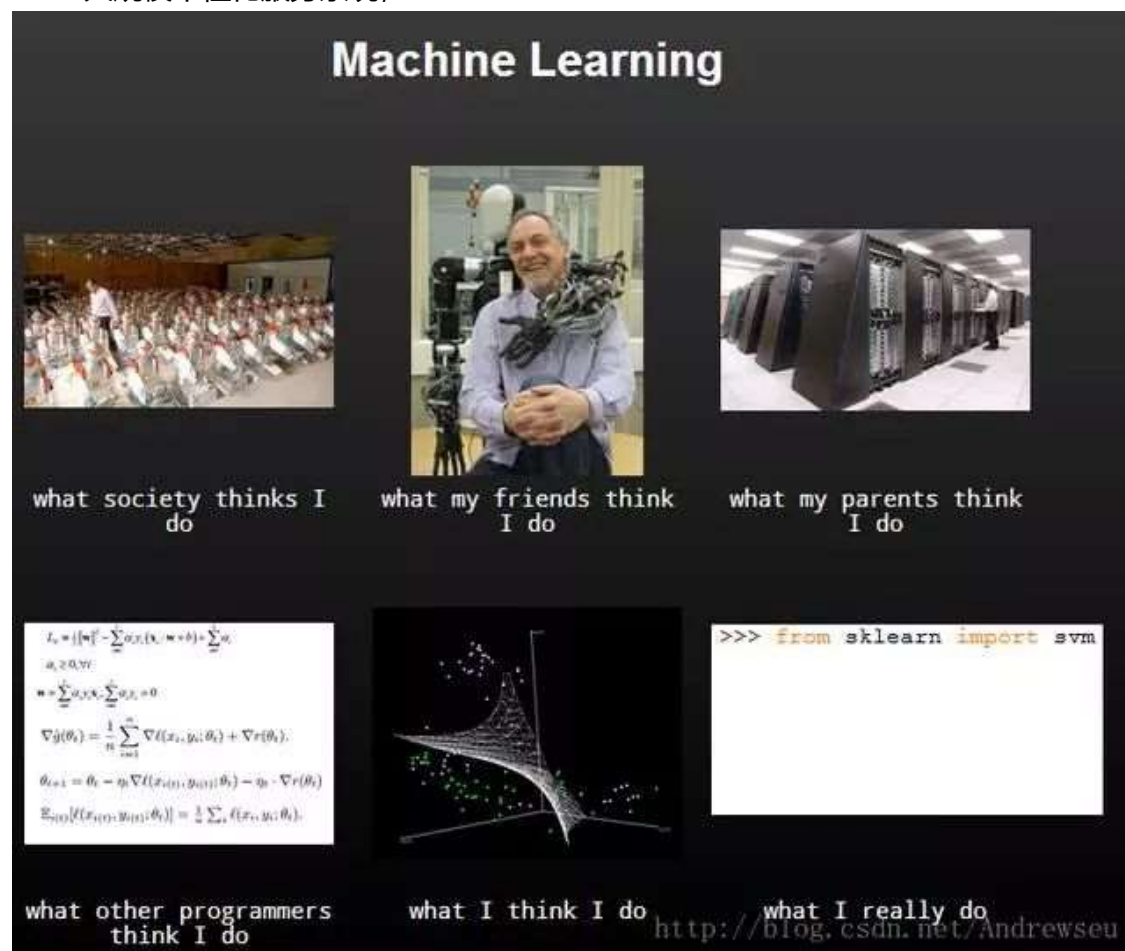
Tom Mitchell: 对于某类任务 T 和性能度量 P ，如果计算机程序在 T 上以 P 衡量的性能随着经验 E 而自我完善，那么就称这个计算机程序从经验 E 学习。

其实随着学习的深入，慢慢会发现机器学习越来越难定义，因为涉及到的领域很广，应用也很广，现在基本成为计算机相关专业的标配，但是在实际的操作过程中，又慢慢会发现其实机器学习也是很简单的一件事，我们最的大部分事情其实就是两件事情，一个是分类，一个是回归。比如房价的预测、股价的预测等是回归问题，情感判别、信用卡是否发放等则是属于分类。现实的情况一般是给我们一堆数据，我们根据专业知识和一些经验提取最能表达数据的

特征，然后我们再用算法去建模，等有未知数据过来的时候我们就能够预测到这个是属于哪个类别或者说预测到是一个什么值以便作出下一步的决策。比如说人脸识别系统，目的是作为一个验证系统，可能是一个权限管理，如果是系统中的人则有权限否则没有权限，首先给到我们的数据是一堆人脸的照片，第一步要做的事情是对数据进行预处理，然后是提取人脸特征，最后选择算法比如说SVM或者RF等等，算法的最终选择设计到评价标准，这个后面具体讲，这样我们就建立了一个人脸识别的模型，当系统输入一张人脸，我们就能够知道他是不是在系统之中。机器学习的整个流程不过就这几步，最后不过就是参数寻优，包括现在如火如荼的机器学习。

当我们判断是否要使机器学习时，可以看看是不是以下的场景

1. 人类不能手动编程；
2. 人类不能很好的定义这个问题的解决方案是什么；
3. 人类不能做到的需要极度快速决策的系统；
4. 大规模个性化服务系统；

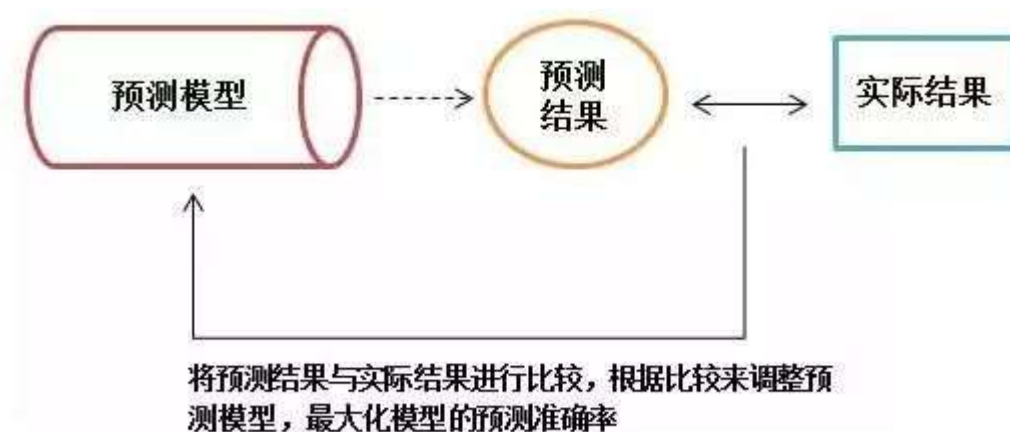


3、机器学习分类

3.1、监督式学习 Supervised Learning

在监督式学习下，每组训练数据都有一个标识值或结果值，如客户流失对应1，不流失对应0。在建立预测模型的时候，监督式学习建立一个学习过程，将预测的结果与训练数据的实际结果

进行比较，不断的调整预测模型，直到模型的预测结果达到一个预期的准确率。



分类 Classification

1. K最近邻 K-Nearest Neighbor (KNN)
2. 朴素贝叶斯 Naive Bayes
3. 决策树 Decision Tree: C4.5、分类回归树 Classification And Regression Tree (CART)
4. 支持向量机器 Support Vector Machine (SVM)

回归 Regression

1. 线性回归 linear regression
2. 局部加权回归 Locally weighted regression
3. 逻辑回归 logistic Regression
4. 逐步回归 stepwise regression
5. 多元自适应回归样条法 multivariate adaptive regression splines
6. 局部散点平滑估计 Locally estimated scatter plot smoothing (LOESS)
7. 岭回归 Ridge Regression
8. Least Absolute Shrinkage and Selection Operator (LASSO)
9. 弹性网络 Elastic Net
10. 多项式回归 Polynomial Regression

排序 Rank

1. 单文档分类 Pointwise: McRank
2. 文档对方法 (Pairwise) : Ranking SVM、RankNet、Frank、RankBoost
3. 文档列表方法 (Listwise) : AdaRank、SoftRank、LambdaMART

匹配学习

1. 人工神经网络: 感知神经网络 Perception Neural Network、反向传递 Back Propagation、Hopfield网络、自组织映射 Self-Organizing Map (SOM)、学习矢量

量化 Learning Vector Quantization (LVQ)

3.2 半监督学习

在半监督学习方式下，训练数据有部分被标识，部分没有被标识，这种模型首先需要学习数据的内在结构，以便合理的组织数据来进行预测。算法上，包括一些对常用监督式学习算法的延伸，这些算法首先试图对未标识数据进行建模，在此基础上再对标识的数据进行预测。如深度学习：

深度学习 Deep Learning

深度学习是 监督学习的匹配学习中人工神经网络延伸出来发展出来的。

1. 受限波尔兹曼机 Restricted Boltzmann Machine (RBM)
2. 深度信念网络 Deep Belief Networks (DBN)
3. 卷积网络 Convolutional Network
4. 栈式自编码 Stacked Auto-encoders

3.3 无监督学习 Unsupervised Learning

在非监督式学习中，数据并不被特别标识，学习模型是为了推断出数据的一些内在结构。

聚类 Cluster

1. K均值 k-means
2. 最大期望算法 Expectation Maximization (EM)
3. 降维方法 Dimensionality Reduction：主成分分析Principal Component Analysis (PCA)、偏最小二乘回归 Partial Least Squares Regression (PLS)、Sammon映射 Sammon Mapping、多维尺度分析 Multidimensional Scaling (MDS)、投影寻踪 Projection Pursuit、RD

关联规则 Association Rule

1. Apriori
2. Eclat

3.4 增强学习 Reinforcement Learning

在之前的讨论中，我们总是给定一个样本 x ，然后给或者不给标识值或结果值（给了就是监督式学习，不给就是无监督式学习）。之后对样本进行拟合、分类、聚类或者降维等操作。然而对于很多序列决策或者控制问题，很难有这么规则的样本。比如，四足机器人的控制问题，刚开始都不知道应该让其动那条腿，在移动过程中，也不知道怎么让机器人自动找到合适的前进方向。

增强学习要解决的是这样的问题：一个能感知环境的自治agent，怎样通过学习选择能达到其

目标的最优动作。这个很具有普遍性的问题应用于学习控制移动机器人，在工厂中学习最优操作工序以及学习棋类对弈等。当agent在其环境中做出每个动作时，施教者会提供奖励或惩罚信息，以表示结果状态的正确与否。例如，在训练agent进行棋类对弈时，施教者可在游戏胜利时给出正回报，而在游戏失败时给出负回报，其他时候为零回报。agent的任务就是从这个非直接的，有延迟的回报中学习，以便后续的动作产生最大的累积效应。

1. Q-Learning
2. 时间差学习 Temporal difference learning

3.5 其他

集成算法

集成算法用一些相对较弱的学习模型独立地就同样的样本进行训练，然后把结果整合起来进行整体预测。

1. Boosting
2. Bootstrapped Aggregation (Bagging)
3. AdaBoost
4. 堆叠泛化 Stacked Generalization
5. 梯度推进机 Gradient Boosting Machine (GBM)
6. 随机森林 Random Forest

参考：

<http://www.studyai.com/article/ad75a319>

<http://www.ppvke.com/Blog/archives/22887>

<http://blog.csdn.net/andrewseu/article/details/53488664>