

智能之门

神经网络和深度学习入门

(基于PYTHON的实现)

STEP 1 基本概念

第 1 章

概论

1.1 人工智能发展简史

1.2 人工智能的定义

1.3 范式的演化

在概论与基本概念中，首先对人工智能的发展简史、定义、以及科学范式的演化进行介绍，并列举了一些有趣的实例，让大家对人工智能的世界观方法论形成一个基本的认识。

1.1 人工智能发展简史

➤ 从计算机科学出现之时，科学家就开始探讨计算机是否能有“智能”。

1950年，英国科学家艾伦图灵发表了论文讨论创造出具有真正智能的机器的可能性，并提出了著名的图灵测试：如果一台机器能够与人类展开对话而不能被辨别出其机器身份，那么称这台机器具有智能。现在活跃于电脑，手机，还有各种硬件上的“智能助手”在各自的功能领域，通常被大众认为挺有智能的。

但是那些“智能助手”，聊天机器人真的有智能么？

1.1 人工智能发展简史

➤ 1980年美国哲学家约翰·瑟尔（JOHN SEARLE）提出了有趣的中文房间问题。

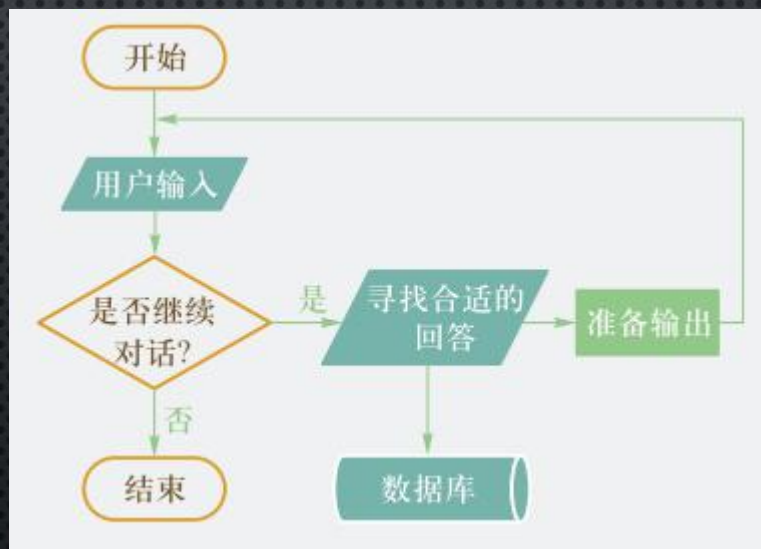
一个对中文一窍不通，只会英语的人被关在一个封闭房间中。房间里有一本英文手册，说明该如何处理收到的汉语信息。房间外的人向房间内输入中文问题，房间内的人便按照手册的说明，查找到合适的指示，将相应的中文字符组合成对问题的解答，并将答案输出。

房间外面的人看到自己递进去的中文输入能得到回答，很可能就会认为房间内的人有智能，懂中文，就像现在的聊天机器人那样，那么这是“智能”么？



1.1 人工智能发展简史

如果可以写一段程序，根据一些规则和已有的数据，和用户进行某种程度的“智能”对话，程序接到用户的输入后，如果不是结束会话的指令，就在数据库中寻找合适的回答，然后根据情况准备输出，如此循环往复。

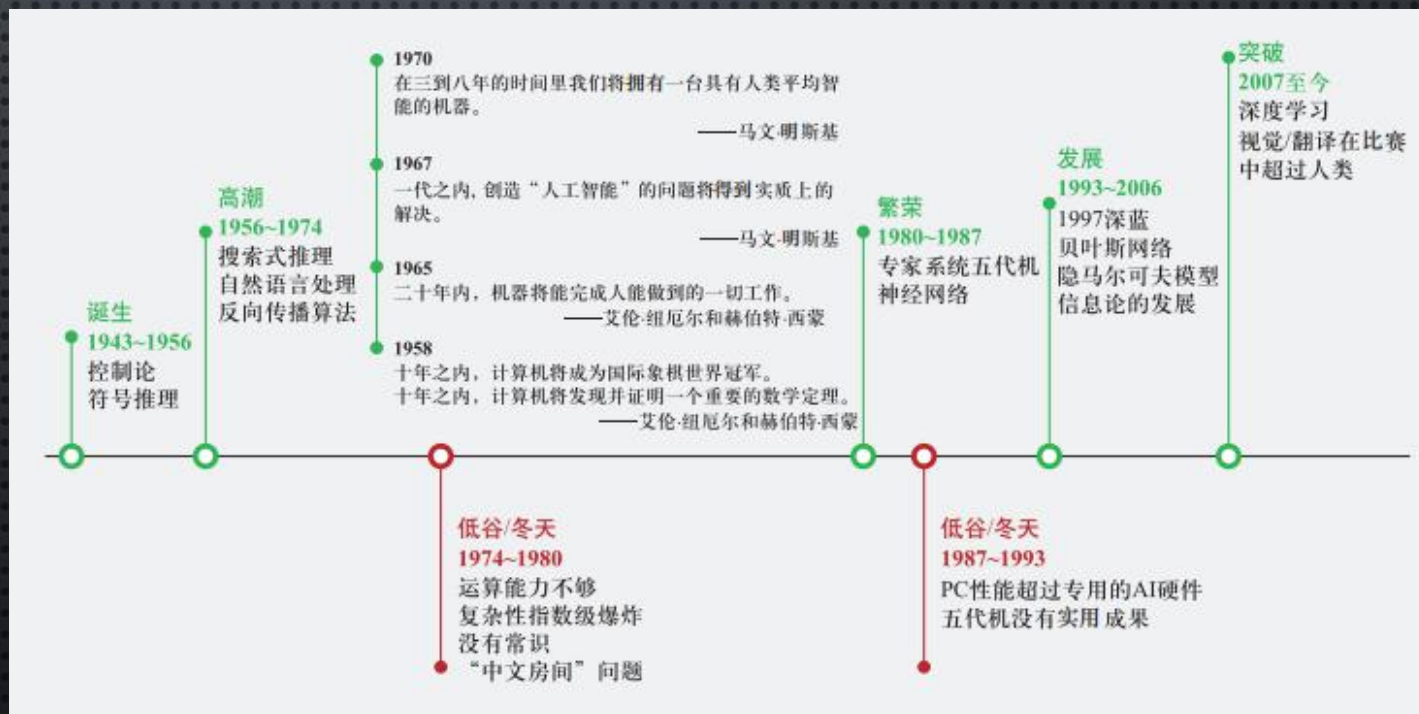


1.1 人工智能发展简史

从1956年的达特茅斯会议开始，人工智能作为一个专门的研究领域出现，经历了超过半个世纪的起伏，终于在2007年前后，迎来了又一次大发展。

➤ 人工智能的“起伏”发展

- 研究（包括技术）取得了一定的进展。
- 研究的进展让人们看到了人工智能的潜力，并对此产生过高的期望。（如1958-1970年间）
- 上述过高的期望让产业界开始开发各种应用，并积极宣传。
- 当这些应用未能全部满足期望时，人工智能行业进入低谷，直到下一波研究和技术取得突破性进展。（如2007年后）



1.1 人工智能发展简史

● QUESTION

- ◆ 现在全国各地高校新建立了七十多个AI学院和专业，这么快的速度的确和巨大的期望值有关。我们不妨问自己，如果AI的发展在几年后又会出现寒冬，那时候我们培养的AI专业的学生刚刚毕业，他们应该怎么应对？还能通过自己的“人工智能”专业找到如意的工作么？那时候需要什么样的技能呢？
- ◆ 在中文房间问题中，房间中的人懂中文么？我们可以把房间内的人等同于电脑的CPU，房间内的说明书相当于程序，房间外的人等同于用户。那么，我们看到用户和电脑在用中文交流，就像用户和中文聊天机器人程序交互一样。这个电脑真的懂中文么？它有智能么？请阐述你同意或反对的理由。

1.2 人工智能的定义

目前，人工智能是一个非常庞杂的概念。从人们对人工智能的期待、技术特点和应用领域的角度，都有很多不同的概念。

➤ 第一个层面，人们对人工智能的期待可以分为：

- 智能地把某件特定的事情做好，在某个领域增强人类的智慧，这种方式又叫智能增强。例如搜索引擎、自动翻译、智能助手等，帮助人类完成某种特定任务。这也叫“弱人工智能”或“狭义人工智能”。
- 像人类一样能认知、思考与判断，却模拟人类的智能，这是人工智能学科一开始就希望达到的目标。这样的智能也叫“通用人工智能”或“强人工智能”。

➤ 第二个层面，从技术的特点来看：

- 如果能让运行程序的计算机来学习并自动掌握某些规律，某种程度上可以谈是实现了狭义的人工智能，这种方法即机器学习。

1.2 人工智能的定义

如果程序解某种任务 T 的效果 P 随着经验 E 的增加而得到了提高，那么这个程序就能从经验 E 中学到了关于任务 T 的知识，并让效果 P 得到提高。

➤ 机器学习的具体过程：

- 选择一个模型结构（例如逻辑回归、决策树等），构建上文中的程序。
- 将训练数据（包含输入和输出）输入模型中，不断学习，得到经验 E 。
- 通过不断执行任务 T 并衡量效果 P ，让 P 不断提高，直到达到一个满意的值。

1.2 人工智能的定义

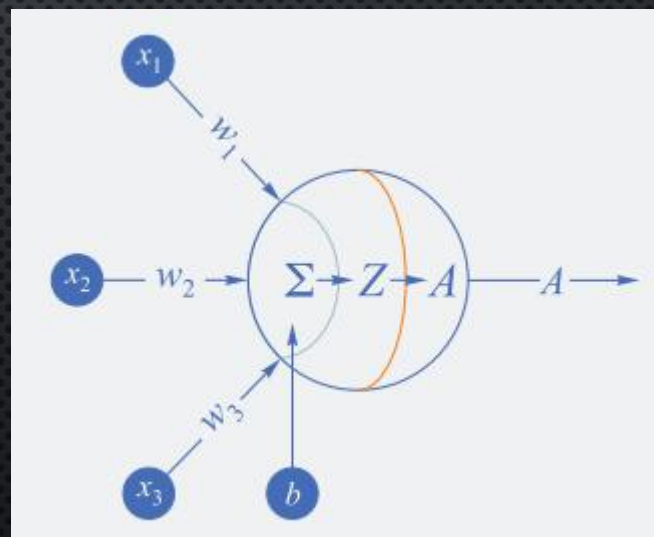
➤ 机器学习的各种方法如何从经验中学习？大致可以分为下面三种类型。

- 监督学习：通过标注的数据来学习，例如，程序通过学习标注了正确答案的手写数字的图像数据，它就能认识其他的手写数字。
- 无监督学习：通过没有标注的数据来学习。这种算法可以发现数据中自然形成的共同特性（聚类），可以用来发现不同数据之间的联系，例如，买了商品A的顾客往往也购买了商品B。
- 强化学习：我们可以让程序选择和它的环境互动（例如玩一个游戏），环境给程序的反馈是一些“奖励”（例如游戏中获得高分），程序要学习到一个模型，能在这种环境中得到高的分数，不仅是当前局面要得到高分，而且最终的结果也要是高分才行。

1.2 人工智能的定义

➤ 神经网络

- 神经网络模型是一个重要的方法，它的原型在1943年就出现了，在生物神经网络中，每个神经元与其他神经元相连，当它兴奋时，就会像相邻的神经元发送化学物质，从而改变这些神经元内的电位；如果某神经元的电位超过了一个阈值，那么它就会被激活（兴奋），向其他神经元发送化学物质。把许多这样的神经元按照一定的层次结构连接起来，我们就构建了一个神经网络。下图是 M-P 神经元模型的示意图。
- 随着数据的丰富和机器计算力的增强，人们不断增加神经网络的层数，相邻层次之间的输入输出由非线性函数来控制，这就产生了深度神经网络。
- 随着人们不断的调整网络结构，DNN 也演变出许多不同的网络拓扑结构，例如卷积神经网络，循环神经网络，长短期记忆，生成对抗网络，迁移学习等，且这些模型还在不断演化中。



1.2 人工智能的定义

➤ 第三个层面，从应用的角度来看，狭义人工智能在各个领域都取得了很大的成果：

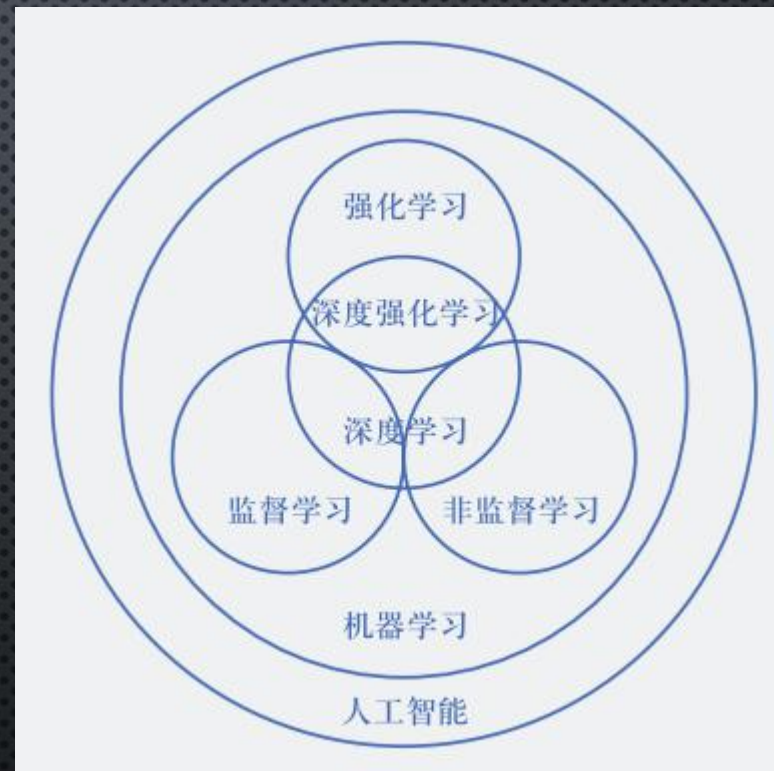
- 一种是标杆式的任务。2015年，AI取得了超过人类的成果。在翻译领域（微软的中英翻译）、阅读理解（SQUAD 比赛）、下围棋（2016）、德州扑克（2019）和麻将（2019）领域中，我们也看到了AI取得了达到或超过人类最高水平的成绩。
- 另一种，是AI技术和各种其他技术结合，解决政府，企业，个人用户的需求。在政府方面，把所有计算，数据，云端和物联网终端的设备联系起来，搭建一个能支持智能决定的系统，现代社会的城市管理，金融，医疗，物流和交通管理等等都运行在这样的系统上。

微软亚洲研究院与某大型船运公司在深度强化学习领域展开了研究合作。船运公司经营着数百条货轮，在上百个港口搬运集装箱，其中装有不同种类、不同优先级和不同目的地的货物，经常会出现某港口有大量空集装箱，而另外的港口却急需集装箱的情况。如何最大化利用货轮的运输能力，避免过多转运空的货柜集装箱？双方合作开发了一套基于大数据和强化学习的全新解决方案，能够在货船抵达港口时就预估出需要装载或者卸载的空集装箱的最优数量。这个方案大幅度降低了港口货柜短缺的情况，提高了效率。

1.2 人工智能的定义

本系列的内容其实只是庞大AI系统中几个微小的部分，右图显示了弱人工智能领域中机器学习部分的内容。

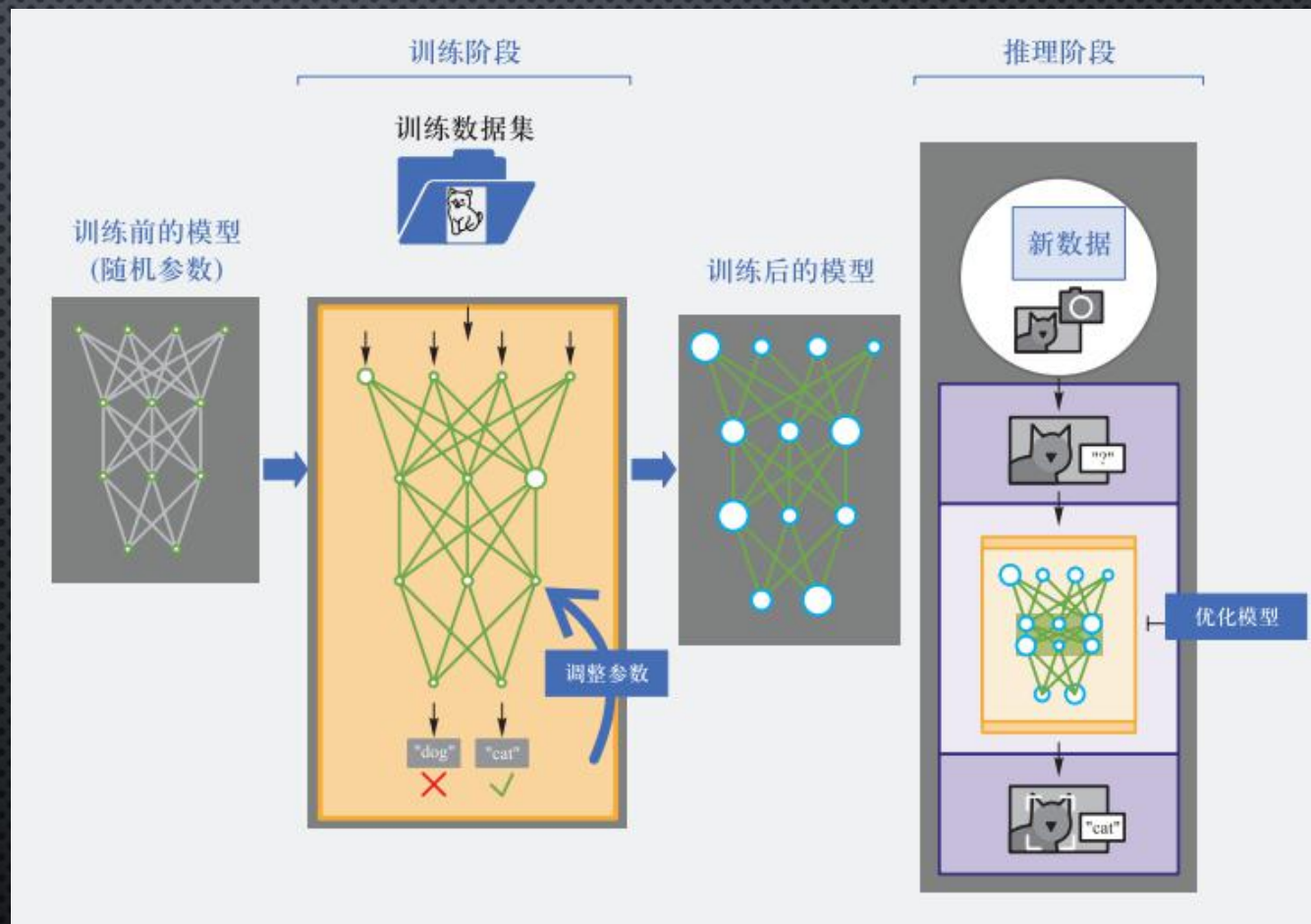
从这个角度说，“智能之门”正是想说明这只是进入人工智能领域的一个小门而已，读者可以自行学习和探索其他门道。



1.2 人工智能的定义

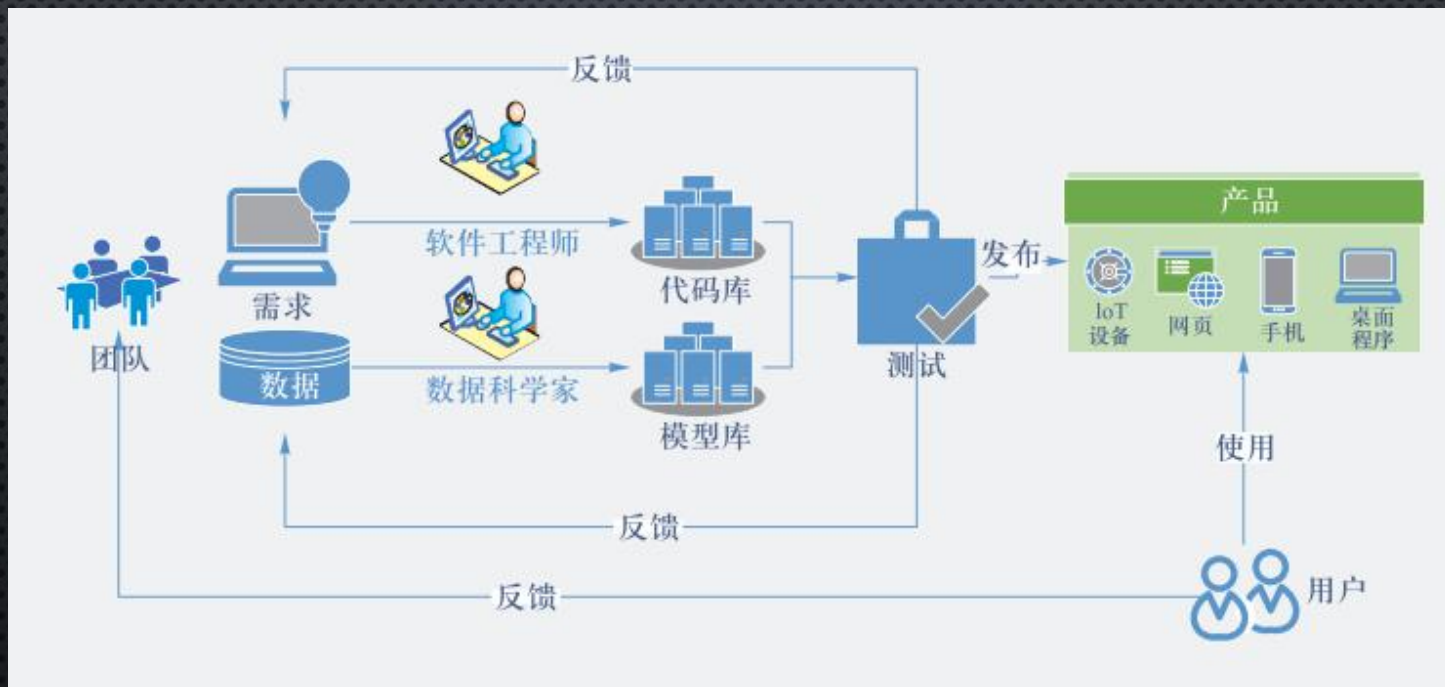
- 一个典型的机器学习模型是怎么得来的，又是怎么在应用中使用的？

以一个有监督学习的例子来说明，要做一个“看照片识别猫”的小应用，且希望该应用判断“照片中是否有猫”的错误率不高于5%，具体模型的生成与应用流程如右图所示。



1.2 人工智能的定义

在现代软件开发流程中，程序的开发，和AI模型的开发的生命周期应该如何协作呢？软件工程师和数据科学家并肩工作，一个完善代码库，另一个完善模型库，最后的产品通过各种途径（网页/桌面程序/手机/IoT设备）交到用户手中。下图展示了这个协作的过程。







1.3 范式的演化

从古至今，人类一直在试图了解客观规律，找到事物变化的相互关系。了解客观规律需要方法论，探索世界、寻找规律的方法论在历史上也发生了几次革命性的改变，即科学范式的演化。

Science Paradigms

- Thousand years ago:
science was **empirical**
describing natural phenomena
- Last few hundred years:
theoretical branch
using models, generalizations
- Last few decades:
a **computational** branch
simulating complex phenomena
- Today: **data exploration** (eScience)
unify theory, experiment, and simulation
 - Data captured by instruments or generated by simulator
 - Processed by software
 - Information/knowledge stored in computer
 - Scientist analyzes database/files using data management and statistics



$$\left(\frac{\dot{a}}{a}\right)^2 = \frac{4\pi G\rho}{3} - K\frac{c^2}{a^2}$$

1.3 范式的演化

➤ 第一阶段：经验

- 从几千年前到几百年前，人们通过观察自然现象来归纳总结一些规律。

孔子东游，见两小儿辩日，问其故。

一儿曰：“我以日始出时去人近，而日中时远也。”

一儿以日初出远，而日中时近也。

一儿曰：“日初出大如车盖，及日中则如盘盂，此不为远者小而近者大乎？”

一儿曰：“日初出沧沧凉凉，及其日中如探汤，此不为近者热而远者凉乎？”

孔子不能决也。

两小儿笑曰：“孰为汝多知乎？”

1.3 范式的演化

➤ 第二阶段：理论

- 这一阶段，科学家们开始明确定义，速度是什么，质量是什么，化学元素是什么（不再是五行和燃素）……也开始构建各种模型，在模型中尽量撇除次要和无关因素，例如我们在中学的物理实验中，要假设“斜面足够光滑，无摩擦力”，“空气阻力可以忽略不计”，等等。在这个理论推导阶段，以伽利略、牛顿为代表的科学家，开启了现代科学之门。伽利略在比萨斜塔做的试验推翻了两千多年来人们想当然的“定律”。

➤ 第三阶段：计算仿真

- 从二十世纪中期开始，利用电子计算机对科学实验进行模拟仿真的模式得到迅速普及，人们可以对复杂现象通过模拟仿真，推演更复杂的现象，典型案例如模拟核试验、天气预报等。这样计算机仿真越来越多地取代实验，逐渐成为科研的常规方法。科学家先定义问题，确认假设，再利用数据进行分析和验证。

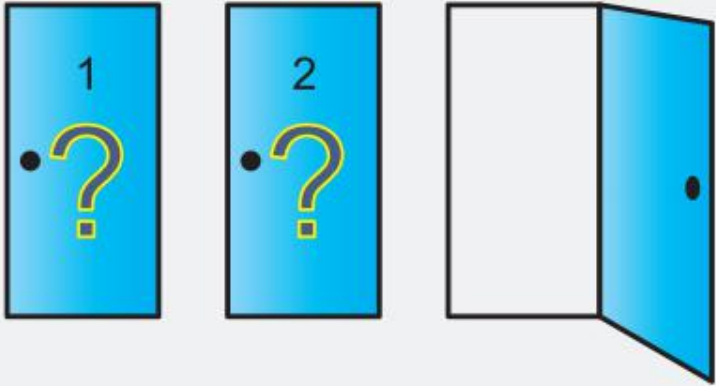
1.3 范式的演化

➤ 第四阶段：数据探索

- 最后我们到了“数据探索”阶段。在这个阶段，科学家收集数据，分析数据，探索新的规律。在深度学习的浪潮中出现的许多结果就是基于海量数据学习得来的。有些数据并不是从现实世界中收集而来，而是由计算机程序自己生成，例如，在ALPHAGO算法训练的过程中，它和自己对弈了数百万局，这个数量大大超过了所有记录下来的职业选手棋谱的数量。

1.3 范式的演化

顾客参加一个抽奖活动，三扇关闭的门后面只有一个有奖品，顾客选择一个门之后，主持人会打开一个没有奖品的门，并给顾客一次改变选择的机会。



因果关系 - 行动改变了选择的概率么?

- 三个门后面有两个门是空的，只有一个有奖品。
- 要求：选择一个门，争取获得奖品
- 你选中一个门，还没有打开。
- 然后主持人打开另外一个空的门
- 此时，你换另外一个门会得到更大的获胜概率么？

1.3 范式的演化

➤ 经验归纳

- 很多人会是从生活经验出发，感觉自己会中计，因此决定“我换了就上当了，我不换”等。当然，也可用类比推理的办法，对100扇门的情况进行分析。

➤ 理论推导

- 设 B, C 为改变和不改变选择的情况下中奖的概率，通过条件概率可计算得

$$P(B) = \frac{2}{3}, \quad P(C) = \frac{1}{3}$$

➤ 数据模拟

- 用程序做数据模拟的方法，得到随机模拟的概率关系，如右图。

```
import numpy as np
import random
def DoorAndPrizeSim(switch, loopNum):
    win = 0
    total = 0

    for loop in range(loopNum):
        prize = random.randint(0,2)
        initialChoice = random.randint(0,2)
        doors = [0,1,2]          #set all value to 0,1,2
        doors.remove(prize)       #remote the door with prize

        if (initialChoice in doors):
            doors.remove(initialChoice)

        #randomly open a door that doesn't have the prize, and it's not the chosen
        n = len(doors)
        r = random.randint(0,n-1)

        openDoor = doors[r]

        if (switch):
            secondChoice = 3 - openDoor - initialChoice
        else:
            secondChoice = initialChoice

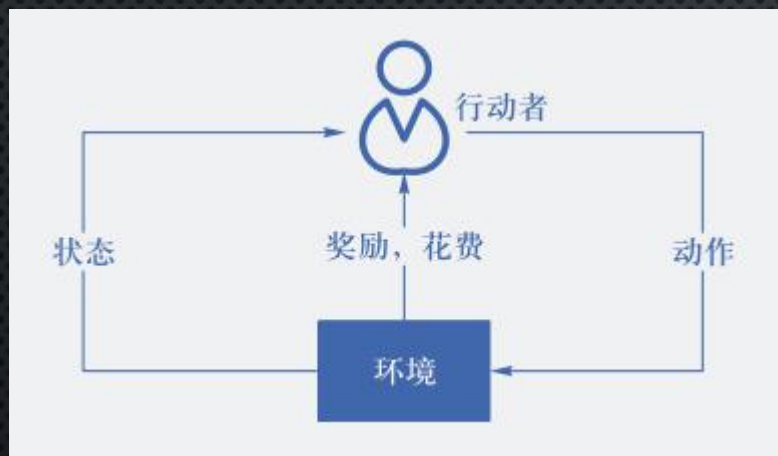
        #see if we have a winning door, which means the prize == secondChoice
        total +=1
        if (secondChoice == prize):
            win += 1

    return (win/total)
```


1.3 范式的演化

➤ 数据探索

- 假设顾客为左图的行动者，他身处环境中，有一定的状态，他为了达到一定的目的（总的奖励），不断地采取一系列的动作去尝试与环境进行交互，这些交互会给他带来奖励，同时改变他的状态，他可以交互中根据反馈不断地调整策略，试图了解到状态、动作和总的奖励关系。强化学习可以通过表格来跟踪和调整这些关系（例如 Q-LEARNING 方法，框架见右图），或者通过神经网络来达到同样的目的。



// 维护一个表格，命名为 Q，存储 Agent 所在的“状态”和“收益”

1. 起始状态，表格 Q 所有内容都清零
2. 在每一个选择的机会时候
3. 查找目前状态中可能收益最大的动作
4. 执行动作，得到收益，进入下一个状态
5. 按照规则，更新表格 Q 中的收益，规则是
6. $\text{新收益} = \text{原来的收益} + (\text{新收益} + \text{未来可能获得的最大收益}) * \text{折扣率}$

1.3 范式的演化

深度学习还有很多更强有力的学习方法，将在下面的章节中一步一步地介绍如何构建各种神经网络，完成各种层次的数据探索任务，以及它们在实践中的应用。

● QUESTION

◆ 计算机的计算能力越来越强，仿真的能力也随之提高。那么，是否存在计算机不能仿真的情况？例如，超级计算机能否运用各种模型和数据，计算出第二天的彩票中奖号码？

THE END

谢谢！