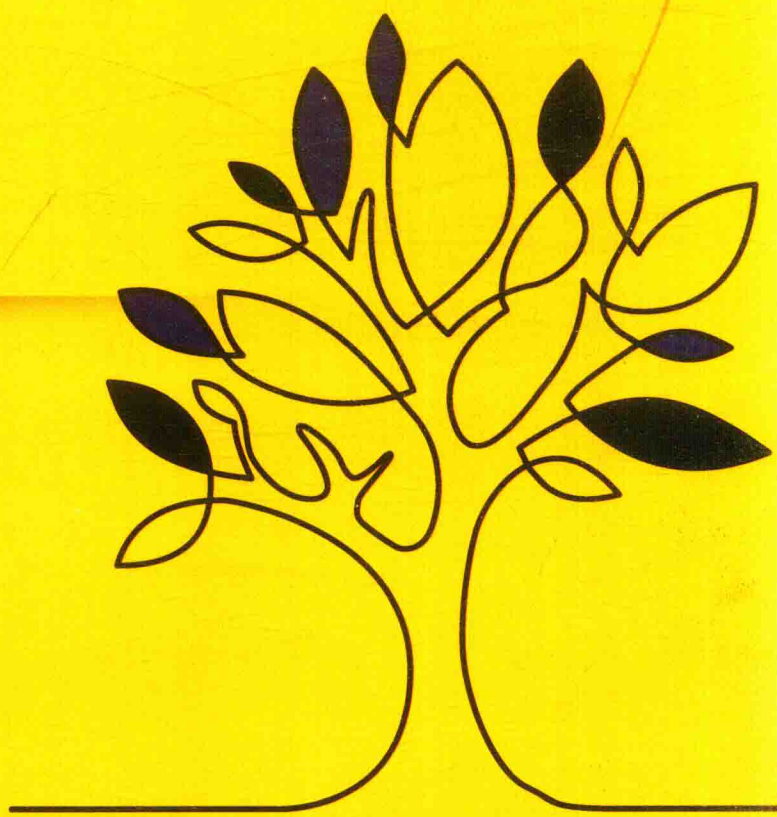
人工智能 教育丛书

Deep Learning

深度学习

主编 刘玉良 戴凤智 张 全



西安电子科技大学出版社
http://www.xduph.com

XDUP 580200

内容简介

深度学习理论由Hinton等人于2006年提出，其概念源于对人工神经网络的研究。深度学习技术通过组合数据的低层特征形成更加抽象的高层属性，以发现数据的分布式特征表示。

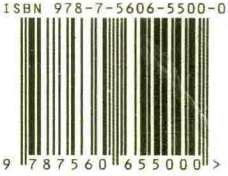
本书主要阐述基于深度学习理论的一些模型和算法。全书共分为8章，主要内容包括绪论、TensorFlow和Keras简介、简单神经网络、图像类数据处理、序列类数据处理、深度学习模型优化、数据和模型的处理与调试、现代深度学习模型概述。附录给出了基于深度学习的视频目标跟踪研究进展综述和Q-Learning算法的参考代码。为便于学习和参考，各章均包含丰富的思考题。

本书主要面向工科院校人工智能、模式识别、数据挖掘和深度学习等专业的本科生，也可供相关专业的研究生和工程技术人员参考。

人工智能 教育丛书

现代神经网络教程
简明人工智能
模式识别
计算智能导论
量子计算智能
智能算法导论
机器学习
多目标进化学习
人工智能创新实验教程
人工智能、类脑计算与图像解译前沿
深度神经网络FPGA开发技术
遥感影像深度学习智能解译与识别
智能影像与大数据挖掘
● 深度学习

ISBN 978-7-5606-5500-0



定价：52.00元

封面设计：李尘工作室

人工智能 教育丛书

深度学习

主　编　刘玉良　戴凤智　张　全
副主编　魏宝昌　李　杰
参　编　袁亚圣　章朱明　尹　迪　侯　伟
　　叶忠用　金　霞　陈　莹

西安电子科技大学出版社
http://www.xduph.com

内 容 简 介

深度学习理论由 Hinton 等人于 2006 年提出，其概念源于对人工神经网络的研究。深度学习技术通过组合数据的低层特征形成更加抽象的高层属性，以发现数据的分布式特征表示。

本书主要阐述基于深度学习理论的一些模型和算法。全书共分为 8 章，主要内容包括绪论、TensorFlow 和 Keras 简介、简单神经网络、图像类数据处理、序列类数据处理、深度学习模型优化、数据和模型的处理与调试、现代深度学习模型概述。附录给出了基于深度学习的视频目标跟踪研究进展综述和 Q-Learning 算法的参考代码。为便于学习和参考，各章均包含丰富的思考题。

本书主要面向工科院校人工智能、模式识别、数据挖掘和深度学习等专业的本科生，也可供相关专业的研究生和工程技术人员参考。

图书在版编目(CIP)数据

深度学习/刘玉良，戴凤智，张全主编. —西安：西安电子科技大学出版社，2020.1
ISBN 978-7-5606-5500-0

Ⅰ. ① 深… Ⅱ. ① 刘… ② 戴… ③ 张… Ⅲ. ① 机器学习 Ⅳ. ① TP181

中国版本图书馆 CIP 数据核字(2019)第 251907 号

策划编辑　刘玉芳
责任编辑　许青青
出版发行　西安电子科技大学出版社(西安市太白南路 2 号)
电　　话　(029)88242885　88201467　　邮　　编　710071
网　　址　www.xduph.com　　电子邮箱　xdupfxb001@163.com
经　　销　新华书店
印刷单位　陕西天意印务有限责任公司
版　　次　2020 年 1 月第 1 版　2020 年 1 月第 1 次印刷
开　　本　787 毫米×960 毫米　1/16　印张　20
字　　数　411千字
印　　数　1～3000 册
定　　价　52.00 元
ISBN 978-7-5606-5500-0/TP

XDUP 5802001-1

序

深度学习是机器学习的最新分支之一，也是人工智能算法领域的最重要组成部分。近年来，深度学习无论在基础理论还是实际应用上均取得了重大进展，已发展成为信息科学领域解决实际问题的重要方法。目前，深度学习已在人工智能的不同领域得到了成功应用，尤其在我们熟知的语音与图像识别、自然语言处理、空间运动再现、航空航天控制等方面更是表现出色，受到了学术界与工业界的广泛关注。

作为一本深度学习方面的通识类本科教材，本书内容的选择重点放在了实际应用上，而没有追求详细的理论推导和深度分析。书中涉及的多个应用示例既有深度学习中传统的案例，也有经过作者反复验证后的凝练升华。读者只需按照书中介绍的步骤进行操作，就可以重现相关的实验结果。一旦读者对深度学习有了操作上的感性认识，再去加深理论学习并推而广之将变得更加容易可行。事实上，本书正是集作者多年教学经验所成，其初稿已在多个大学团队使用，教学效果很好。本书初稿先后获得了 2017 年中国轻工业联合会优秀教材二等奖和 2018 年高等教育天津市级教学成果二等奖。

本书作者之一戴凤智博士曾师从日本人工生命与机器人(AROB)之父、大分大学杉坂政典教授，我也有幸在杉坂政典教授创办的 AROB 国际会议上与作者多次相遇交流，深感他在深度学习与人工智能以及其他相关研究热点方向上走在了教学和科研的前沿。本书的出版无疑是国内人工智能教学研究方面的一件喜事，我非常乐意把这本优秀的教材推荐给大家，期望广大读者能从中受益。

北京航空航天大学
贾英民
2019 年 9 月

前　言

2015 年 10 月，以深度学习技术为基础的人工智能围棋程序 AlphaGo 连续五局击败欧洲围棋冠军樊辉；2016 年 3 月，AlphaGo 以 4∶1 战胜排名第四的韩国职业棋手李世石；2017 年 5 月，AlphaGo 又以 3∶0 战胜排名第一的中国职业棋手柯洁。自此国内迎来深度学习的研究热潮。2019 年图灵奖授予了深度学习研究领域的 Hinton 教授等三人。

深度学习理论是由 Hinton 等人于 2006 年提出的，其概念源于人工神经网络的研究，它通过组合低层特征来形成更加抽象的高层表示属性类别或特征，以发现数据的分布式特征表示。

深度学习在短短数年间迅猛发展，颠覆了语音识别、图像分类、文本理解等领域的算法设计思路，逐渐形成了一种从训练数据出发，经过一个端到端的模型，然后输出得到最终结果的新模式。这一创新不仅使整个系统变得更加简洁，而且准确度也能够通过综合调整深度神经网络中各个层的特征信息来不断提升。深度学习凭借大数据时代与电子技术的助力实现了迅速发展。深度学习网络对各种海量数据(不管是标注数据、弱标注数据还是仅仅数据本身)都可以加以利用并完全自动地学习深层的知识表达，这种知识其本质就是原始数据的高度浓缩与概括。

得益于深度学习强大的特征提取能力及其在计算机视觉、语音识别、大数据等领域取得的巨大成功，深度学习已经进入人们的视野，将其应用于实际工程方面具有良好的发展前景，并已成为当前领域的热点研究方向。

本书主要阐述基于深度学习理论的一些模型和算法，在对其进行详细介绍的同时，吸纳了国内外许多具有代表性的最新研究成果。全书取材新颖，内容丰富，注重理论与实际的结合，提供了将深度学习应用到实际项目中所需要的知识。

全书内容共分为 8 章：第 1 章绪论，介绍深度学习的思想、定义和研究现状以及未来的发展趋势，由刘玉良、侯伟编写；第 2 章 TensorFlow 和 Keras 简介，主要介绍 TensorFlow 以及 Keras 框架，并通过实际操作进行演示说明，由戴凤智、魏宝昌编写；第 3 章简单神经网络，从人脑学习的角度引入人工神经网络的概念，讲述基本理论并通过 Keras 予

以实现，由张全、袁亚圣编写；第4章图像类数据处理，以卷积神经网络为依托，介绍相关的深度技术，并将其应用于图像类的数据处理之中，由叶忠用、尹迪编写；第5章序列类数据处理，主要介绍了一系列序列类数据的处理方法并通过Keras进行实现，由金霞、陈莹编写；第6章深度学习模型优化，系统地介绍了几种常见的深度学习模型的优化方法，同时讨论影响深度学习模型优化的因素，由李杰、章朱明编写。第7章数据和模型的处理与调试，介绍了如何根据具体应用挑选一个合适的算法以及对系统的性能进行评价，并根据实验反馈的性能改进机器学习系统，由张全、袁亚圣编写；第8章现代深度学习模型概述，介绍了几种常见的深度学习模型，包括玻尔兹曼机、自编码器、深度信念网络等，为将来的模型设计提供思路，由张全、尹迪编写。魏宝昌完成了本书附录部分的编写工作。全书由刘玉良、戴凤智和张全最终整理并成稿。

本书在编写过程中得到了中国人工智能学会智能空天系统专业委员会主任、北京航空航天大学贾英民教授的鼓励和支持，他亲自为本书作序。作为后辈学者，本书作者非常感谢贾英民教授的提携。本书的出版是天津科技大学电子信息与自动化学院集体努力的结果。其中，刘玉良老师及其研究生团队成员张全、李杰、章朱明、侯伟，以及戴凤智老师及其研究生团队成员魏宝昌、叶忠用、金霞、袁亚圣、尹迪、陈莹均参与了本书的编写与校正工作。同时感谢西安电子科技大学出版社领导和参与此书编辑出版的工作人员的大力协助，倘若没有他们的热情支持，本书难以如此迅速地和大家见面。此外，如果没有各位学者以公开出版物、课程和代码的方式分享他们的成果，本书可能也不会存在，这里对于他们的奉献表示衷心感谢。

由于编者水平有限，书中不妥之处在所难免，恳请读者批评指正。

编　者

2019年9月8日

目　录

第1章 绪 论

1.1 引 言

或许我们已经不记得在刚出生时睁开眼睛那一刻对这个世界的好奇，或许我们已经忘记在尝试了解这个世界时对一切都是那么新鲜的感觉。但是当我们刚刚记事时，可能有过这样的记忆：家人曾经将一个我们没见过的圆圆或扁圆的东西放在我们眼前，告诉我们它叫橘子。尝了尝之后，我们便记住了这种有着酸酸甜甜味道、圆圆或扁圆形状的橘黄色或者绿色的水果。

后来我们又遇到了橙子，它长得确实和橘子很像，所以我们总是认错它。但是我们通过观察，发现它的果蒂的形状和切开后的样子与橘子不一样。此时，我们发现自己又认识了一种新的水果，而且似乎比当初认识橘子的时候更省力。

再后来，随着时光流逝，我们上学了，耳边总能听到爸爸妈妈的唠叨："你要多多看书，勤奋复习，这样才能取得好成绩。"就这样，我们便知道了学习时只要下足了功夫，弄清了概念，做好了作业，自然就会取得不错的成绩……

以上我们回忆了许多往昔的时光。作为开场，我们将大致介绍一下什么是机器学习(Machine Learning)。

不难发现，在众多回忆中，学习过程似乎占据了我们成长的绝大部分，我们通常通过学习某些经验，从而获得识别及判断某些事物的能力。

例如，为什么当我们观察到橘黄色的、圆圆的、具有酸酸甜甜味道的物体时首先会想到它是一个橘子？这是因为在我们以前的经历中已经学习到了足够的经验，当我们观察到具有上述特征的物体后便能识别物体的种类。

为什么我们知道只要下足了功夫，弄清了概念，做好了作业，自然就会取得不错的成绩？这也是因为在实际生活当中，我们发现在努力以后可以获得更好的成绩，而松懈往往会导致成绩下滑，此时就要改变学习态度了。

在生活中我们往往也会认错具有相近特征的物体，但是在了解它们的区别以后，基本就可以根据已经了解的事物特征快速地识别与区分相似的事物(例如前面提到的橘子与橙子的区别)。由此可见，我们可以做出准确有效判断的前提是积累大量的相关经验，通过这些经验获得事物的特征后就可以对其进行判断分类了。

我们可以理解，人类是通过学习的方法来完成经验的积累并利用获得的经验完成判断的。那么计算机能够具有这样的能力吗？创造具有智能的机器一直是人们梦寐以求的理想。

机器学习就是帮助我们完成这个梦想的一门学科，它致力于研究如何通过计算的手段，采用学习样本的方式来自动改善系统的性能。在计算机系统中，经验通常以数据的形式存在。因此，机器学习就是指通过使计算机自动学习隐含在数据中的事物特征来使计算机拥有智能。

这种提取隐含特征的方法首先需要利用计算机在原始数据上建立模型(Model)，也就是我们所说的学习算法(Learning Algorithm)。有了学习算法以后，可以把数据(水果的外部形状、颜色、味道、切开的形状等)提供给它。这样就可以根据已经生成的模型在面对新情况(一个水果)时，给我们提供一个相应的判断(比如说它是一个橘子)。如果说计算机科学是研究"算法"的，那么与此对应地可以说机器学习是研究"学习算法"的。

在这里，我们所说的"模型"泛指从原始数据中学习到的"经验"以及分类方法。

1.2 基本术语

要进行机器学习，正如前面所述，我们需要数据。假如我们收集到了一些关于水果的数据：(外部形状＝圆形；颜色＝橘黄色；味道＝酸甜；切开的形状＝一瓣一瓣的)，(外部形状＝椭圆形；颜色＝橘黄色；味道＝酸；切开的形状＝一瓣一瓣的)…… 每一对括号就是一条记录，也就是一个**样本**(Sample)。许多样本的集合就形成了一个**数据集**(Dataset)。

我们也知道描述一个事物应该是多方面的，如外部形状、颜色、味道等，那么这些能够反映物体特性的事项被称为**特征**(Feature)。而对它们的具体描述，如圆形、橘黄色、酸甜等被称为**属性**(Attribute)。我们可以想象，应该从多方面去描述一个物体，有效特征越多，就可以越详细地描述事物，那么这种特征的种类个数便称为**维数**(Dimensionality)。

也正如描述三维空间那样，我们可以用三维数据去描述一个物体所处的位置。对应上面描述的橘子，我们使用了四维数据来描述它(外部形状、颜色、味道、切开的形状)。那么每个橘子都可以在这样的一个四维空间中找到描述它的唯一位置。我们将这个空间称为**样本空间**(Sample Space)。

由于每个橘子(样本)在这样的空间中都有着自己的坐标位置，因此我们把每一个样本的全部特征统称为一个**特征向量**(Feature Vector)。我们可以通过得到的特征向量并根据经验来进行判断。也就是说，我们可以根据观察来判断这个东西究竟是不是橘子(类型)。这样的类型我们称之为**标签**(Label)，通常用标签来表示每一个样本的所属类型。具体示例如表 1.1 所示。

表 1.1 中第一行第 2～5 列外部形状、颜色、味道、切开的形状为特征，从第二行开始每一个编号后为一个样本，每一个样本针对不同特征的具体描述为该样本的属性。每一个

样本构成一个特征向量，例如编号1的样本的特征向量为(圆形，橘黄色，酸甜，一瓣一瓣)。最后一列类型数据为标签。

表1.1 水果判断数据集

编号	外部形状	颜色	味道	剥开外皮后的形状	类型
1	圆形	橘黄色	酸甜	一瓣一瓣	橘子
2	长条状	黄色	甜	长条形	香蕉
3	圆形	橘黄色	酸	不分瓣	橙子

我们可以将一个数据集抽象为数学形式 D，该数据集由许多样本 x_i 构成，即 $D=\{x_1, x_2, \cdots, x_i\}$。每一个样本 x_i 由众多属性 a_j 构成，即 $x_i=\{a_1, a_2, \cdots, a_j\}$，这个样本空间是一个 j 维空间。

当获得一系列水果的数据后，我们就可以根据这些数据总结经验了，这一过程就是家人把一个东西放在我们面前帮助我们认识它的过程。所以把从数据中获得模型的过程称为**学习**(Learning)或者**训练**(Training)。这个过程要通过某个具体的学习算法来完成。

在训练过程中所使用的数据集称为**训练数据**(Training Data)，训练数据中的每一个样本称为**训练样本**(Training Sample)，训练样本组成的集合称为**训练集**(Training Set)。学习算法的目的就是找到在数据中存在的某种潜在的规律，因此称之为**假设**(Hypothesis)。也就是说，学习算法用于拟合或者逼近这种潜在的规律。

当学习算法学习到某种经验以后会对输入的数据做出一个判断，这个判断称为**实际输出**(Actual Output)。我们希望它能做出正确的判断，那么按我们的希望做出的正确判断称为**目标输出**(Target Output)。

综合以上，可以将学习算法 f 利用数据 x 给出实际输出 O 的过程抽象为数学表达式，即 $O=f(x)$，这样的一个过程也被称为**预测**(Prediction)。

那么如何评价实际输出与目标输出的差距呢？我们需要通过度量两者之间的偏差来量化学习模型的效果，此时就需要**损失函数**(Loss Function)，也称为**代价函数**(Cost Function)。

这个损失函数度量的就是现在的实际状态与希望状态之间的“距离”。当损失函数大的时候就意味着我们要改变自己的学习方法了。损失函数应具有如下性质：

(1) 函数值非负。

(2) 实际输出与目标输出越接近，损失函数值越小，反之则越大。

常见的损失函数有二次代价函数、交叉熵代价函数等，其表达式分别如下：

$$\text{Loss}=\frac{1}{2n}\sum_x \| f(x)-t \|^2 \tag{1.1}$$

$$\text{Loss}=-\frac{1}{n}\sum_x [t\ln f(x)+(1-t)\ln(1-f(x))] \tag{1.2}$$

其中，n 是训练数据的总数，求和是在所有的训练输入 x 上进行的，$f(x)$ 是对应的目标输出。

通过学习算法找到隐藏在数据中的规律后，需要利用一定的手段给出结果。如果我们想要预测的是一个个独立的物体，如橘子、香蕉等，那么此时的学习任务称为**分类**(Classification)。如果想要预测的是连续值，如 35% 的概率是橘子，63% 的概率是橙子，2% 的概率是香蕉，那么此类学习任务被称为**回归**(Regression)。若仅仅涉及两类事物的分类，则称为**二分类**(Binary-Classification)(如仅区分橘子和香蕉)；若区分多种类别，则称为**多分类**(Multi-Classification)(如分类橘子、橙子和香蕉等)。

总体来说，预测任务就是通过对训练集 $\{(x_1, t_1), (x_2, t_2), \cdots, (x_j, t_j)\}$ 进行学习，建立一个从输入空间到标签空间的映射 f。

我们知道，学习并不是一蹴而就的事情，它需要反复，需要不断地复习。正如当初父母教育我们的时候所说的话一样——你要多多看书，勤奋复习，这样才能取得好成绩，机器学习亦是如此。我们把计算机反复学习的过程称为**迭代**(Iteration)，反复学习的次数称为**迭代次数**(Epoch)。

在获得模型以后，工作并没有全部结束。如果我们仅仅通过训练集的性能来评价模型，显然是不科学的，因为我们希望获得的模型不只适用于训练集的规律。所以我们还要将学习到的模型应用于一个与训练集完全独立的数据集来测试它的性能，这一过程称为**测试**(Testing)。用于测试的数据集称为**测试集**(Testing Set)。

很多文献还提及了**验证集**(Validation Set)。我们有时会根据模型在验证集上的表现对模型的众多参数进行调优。验证集是与训练集相互独立的数据集合，在优化模型参数的过程中有时会使用到验证集。这也是验证集与测试集的区别，因为测试集专门用来测试模型性能，不会直接参与对模型的优化工作。

在训练和测试的过程中可能会出现如下两种情况：

(1) 模型在训练集上表现得很好而在测试集上表现得很糟。这种情况我们称之为**过拟合**(Over Fitting)。

(2) 在测试集上的表现比在训练集上的表现要好，这种情况称为**欠拟合**(Under Fitting)。

总的来说，过拟合发生在过度学习了训练集的特征而导致对除训练集以外的其他数据不能很好识别的情况；而欠拟合描述的是没有充分学习训练集的特征而导致整体性能不佳的情况。

如上所述，机器学习的过程实际上是一个从输入空间到标签空间的映射，也就是一个函数关系，我们用图 1.1 和图 1.2 来描述。

从图 1.1 中可以看出，学习算法(曲线)完全拟合了特征(图中的各个点)，对训练集给出了一个十分准确的拟合。从图 1.2 中可以看出，虽然学习算法(直线)没有准确地拟合所有的特征点，但是也对训练集给出了一个较为合理的拟合。

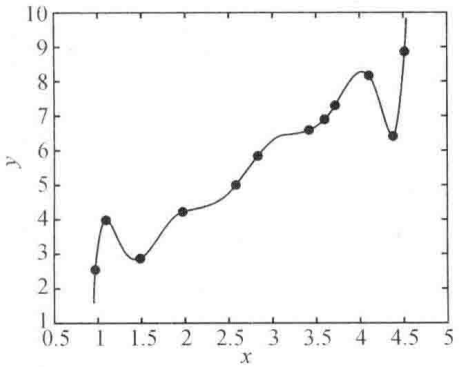


图 1.1　曲线完好地拟合了特征点

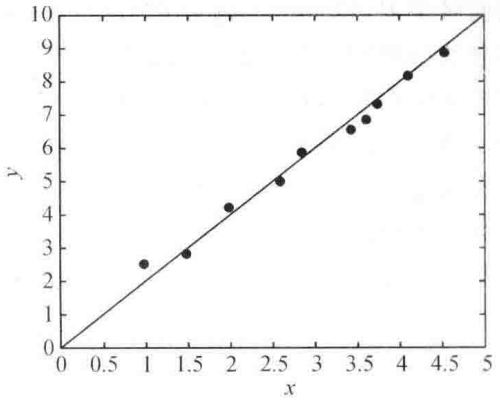


图 1.2　直线对特征点的拟合情况

虽然在没有应用背景的情况下我们很难直接说这两种拟合方法哪个更好，但是从中可以看出：

(1) 高阶多项式(如比较复杂的曲线)可以完全描述真实情况下的训练集特征。但是由于它过度地拟合了训练数据的特征，因此有可能导致它不适用于训练集以外的其他数据。

(2) 低阶多项式曲线并没有严格地描述训练集的所有特征，所以它可能拥有更为强大的抗干扰能力。但是由于模型未完全学习训练集的所有特征，因此也可能存在着欠拟合现象。

我们必须记住：模型既不能过于复杂，也不能过于简单。建立模型的目的就是以适当的精度去挖掘数据中隐藏的特征和联系，这里存在一个度。经过训练的模型如果对具有同一规律的学习集以外的数据也能给出合适的输出，就称该模型具有**泛化能力**(Generalization Ability)，也称为模型的**鲁棒性**(Robustness)。我们人类在泛化上表现得很好。例如，给一个儿童几幅大象的图片，他就能快速地学会认识其他大象。当然，他偶尔也会搞错，很可能将一头犀牛误认为大象，但是一般来说，识别错误的概率会很小。这是因为我们有个系统，即人的大脑，它拥有超强的学习能力，在受到少量图像的训练后，大脑系统能够学会在其他图像上进行推广。

我们再回忆一下前面提到的认识橘子和橙子的过程。当我们记住了橘子的特征以后再去认识橙子的时候，是不是感觉到轻松很多？人类可以将以前学到的知识应用于解决新的问题，能够更快地解决问题或取得更好的效果。这样的学习方法称为**迁移学习**(Transfer Learning)。迁移学习的最大优点就是可以从以前的任务当中学习知识或经验，并应用于新的任务中。这种学习模式可以节约大量时间成本与计算资源。

前面提到的儿童在家人监督下的学习，实际上是对有标签的数据的学习。但是在实际生活当中我们不可能保证所获得的数据全部具有正确的标签，那么我们就要进行没有标签的学习。顾名思义，前一种对于有标签数据的学习就如同有一个老师在一直监督我们一样，我们称之为监督学习(Supervised Learning)或有教师学习；而对没有标签的数据的学习称

为无监督学习(Unsupervised Learning)或无教师学习。

分类与回归是有监督学习的代表，而无监督学习的代表则是聚类(Clustering)。聚类就是训练集中的样本自发地分成若干组，每组称为一个簇(Cluster)，这些自动形成的簇可能对应一些潜在的概念划分。这样的学习过程有助于我们了解数据内在的规律，能为更深入分析数据建立基础。

通常我们假设样本空间中的全部样本都服从一个未知的分布 D (Distribution)，我们获得的每一个样本都是从这样的一个训练集中采样获得的，因此这些样本都是独立同分布的(Independent and Identically Distributed)。所以训练样本越多，我们能够得到关于 D 的信息越多，这样也就越有可能获得具有更强泛化能力的模型。

1.3 监督学习算法

简单而言，监督学习算法就是给定一组输入 x 和输出 y 的训练集，通过学习来获得输入和输出两者之间的映射关系。

1.3.1 支持向量机

支持向量机(Support Vector Machine，SVM)是监督学习中最有影响力的方法之一。不同于逻辑回归，SVM 输出的仅仅是样本的类别，而不是概率。也不同于传统的线性回归，SVM 的重要创新就是核函数(Kernel Function)。这些概念将在后面进行较为详细的论述。

我们知道，分类学习最基本的思路就是在训练集 $D = \{(x_1, y_1), (x_2, y_2), \cdots, (x_m, y_m)\}$ (以二维平面为例)中划分不同种类样本的超平面。但是这样划分的超平面可能有很多，如图 1.3 所示，如何去寻找最优超平面呢？

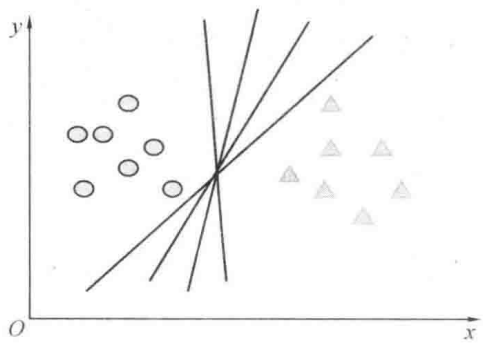


图 1.3 分类超平面不唯一

从直觉上讲，我们似乎应该从两类样本的正中间来分开它们，因为这样最鲜明而且距离两类样本的边界都最远，不容易发生分类错误。也就是说，这样划分超平面对训练样本局部扰动的抗干扰性能最好，鲁棒性也最高。

那么如何将分类超平面抽象成数学模型呢？我们在二维空间中可以用一个线性方程来表示分类超平面(一条直线)，即

$$\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{x} + \boldsymbol{b} = \boldsymbol{0} \tag{1.3}$$

其中，$\boldsymbol{\omega}$ 为法向量，对应线性方程中的斜率 $1/k$，它决定了超平面的方向；$\boldsymbol{b}$ 为位移量，决定了超平面与坐标原点的距离。