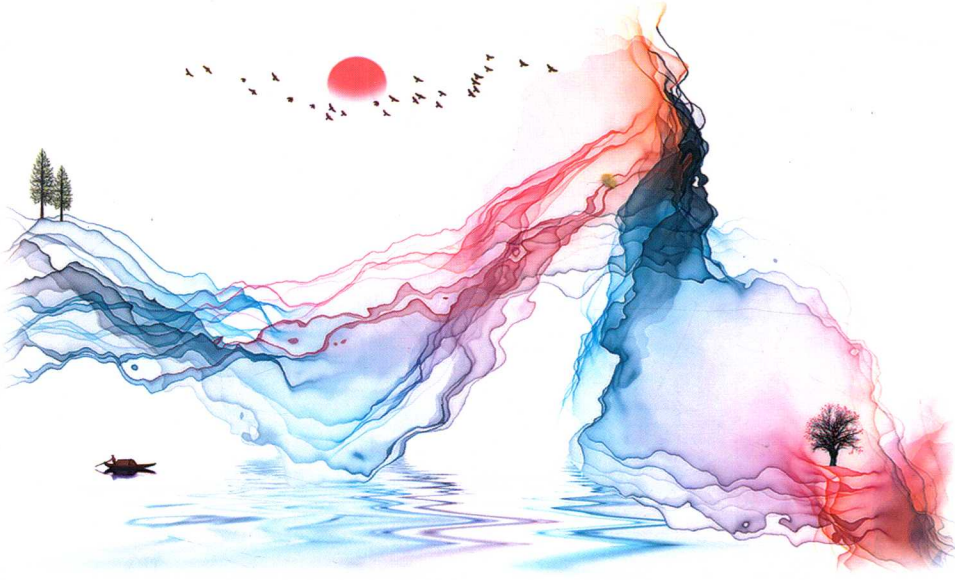
CMP.BOOKS
机工IT

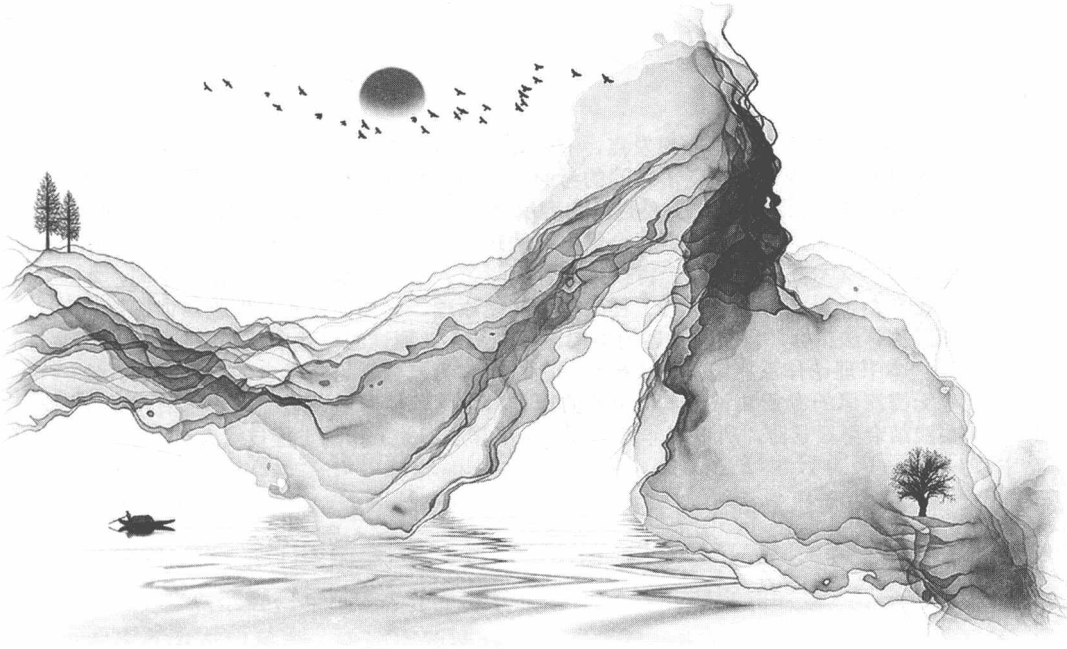
本书收录了近年来在工业界被广泛应用的机器学习算法，这些算法经受了时间的考验，不但效果好而且使用方便。本书另一特色是介绍了算法周边的一些工程架构及实现原理。



深入浅出
工业机器学习算法
详解与实战

张朝阳 著

机械工业出版社
CHINA MACHINE PRESS



深入浅出
工业机器学习算法
详解与实战

张朝阳 著

机械工业出版社
CHINA MACHINE PRESS

实用性是本书的基本出发点，书中介绍了近年来在工业界被广泛应用的机器学习算法，这些算法经受了时间的考验，不但效果好而且使用方便。此外，本书也十分注重理论的深度和完整性，内容编排力求由浅入深、推理完整、前后连贯、自成体系，先讲统计学、矩阵、优化方法这些基础知识，再介绍线性模型、概率图模型、文本向量化算法、树模型和深度学习。与大多数机器学习图书不同，本书还介绍了算法周边的一些工程架构及实现原理，比如如何实时地收集训练样本和监控算法指标、参数服务器的架构设计、做 A/B 测试的注意事项等。

本书理论体系完整，公式推导清晰，可作为机器学习初学者的自学用书。读者无需深厚的专业知识，本科毕业的理工科学生都能看懂。另外由于本书与工业实践结合得很紧密，所以也非常适合于从事算法相关工作的工程技术人员阅读。

图书在版编目（CIP）数据

深入浅出：工业机器学习算法详解与实战 / 张朝阳著 . — 北京：机械工业出版社，2020. 1

ISBN 978-7-111-64056-1

Ⅰ . ①深… Ⅱ . ①张… Ⅲ . ①机器学习 - 算法 Ⅳ . ① TP181

中国版本图书馆 CIP 数据核字（2019）第 230518 号

机械工业出版社（北京市百万庄大街 22 号 邮政编码 100037）

策划编辑：孙 业 责任编辑：孙 业 白文亭

责任校对：郑 婕 责任印制：郜 敏

北京中兴印刷有限公司印刷

2020 年 1 月第 1 版第 1 次印刷

169mm×239mm • 17.75 印张 • 344 千字

标准书号：ISBN 978-7-111-64056-1

定价：69.00 元

电话服务 网络服务

客服电话：010-88361066 机 工 官 网：www.cmpbook.com

010-88379833 机 工 官 博：weibo.com/cmp1952

010-68326294 金 书 网：www.golden-book.com

封底无防伪标均为盗版 机工教育服务网：www.cmpedu.com

前　言

我曾经在字节跳动工作过一段时间，那是一家企业文化让我感到很舒适的公司，我说的舒适指的是平等和开放，公司很有野心，员工也都十分优秀，每年校招总能吸引一大批优秀的毕业生前来面试。在面试算法岗的同学里有很大一部分人面临这样的困境：他们很勤奋，也上过数据挖掘、机器学习的相关课程，但是对算法的思想、特点把握得不准，也看不到各个算法之间的通性和联系，有些同学直接问我有什么书可以推荐。我深深地感到对于那些有一定机器学习基础而经验尚不丰富的同学来说，一本从实际应用出发、深挖算法原理的图书是多么的重要。不久之后我就开始了本书的编写，没想到这一写就是一年半的时间，在整理知识的过程中我也得到了进一步的提升，希望在今后的日子里能和读者一起进步。

本书更是写给进入职场的算法工程师的，我确定这里需要一个“更”字。直到今天我一直都是一线码农，我写的都是实际工作中效果好、见效快的算法，如果你是一名算法工程师，在读本书的过程中相信会引起更多的共鸣。我刚参加工作那几年买了不少机器学习方面的书，现在想想当初是多么的焦虑，如果那时我就拥有本书会是多么的开心。因此写书时心里一直有个美好的愿望，希望我能写出一本工业界的算法宝典，帮广大读者省去四处求索的时间。当然算法工程师还有另一个重要的学习途径 —— 看论文，论文是关于算法的第一手资料，里面有原作者创作的心路历程。但是看论文对读者的知识技能要求比较高，因为写论文的人一般都会对算法的优点大书特书，而对劣势轻描淡写，道行浅的人难以辨别其中的水分有多少。相比之下写书的人会比较客观，他会把不同的算法放在一起对比，而且被写进书里面的算法都经过了时间的考验，所以读书对于初级选手来说是一种更友好的学习方式。

在难易程度上，我对本书的定位是深入浅出。“深入浅出”这个词已经被大家用滥了，看到这个词读者心目中想到的可能只有“浅出”，默认忽略了“深入”。而我一直在提醒自己写书一定要深入，不能为了简单易懂而故意避讳冗长的公式推

导，深入才是对读者的负责。对于每一位想成为算法工程师的同学而言，你们都选择了一条注定艰辛的道路，不要幻想着通过轻轻松松看完一本书就能够领会各大主流算法的精髓。然而如果写书只顾着深入，那就成了作者一个人的自嗨，同样是对读者的不负责。为了帮助读者加速理解、减少疑惑，我会尽量从实际应用出发，多举一些工程实践中的例子，详细列出公式的推导过程，给出核心算法的代码实现，道破不同算法的内在联系。

张朝阳

目 录

CONTENTS

第1章 概 述

欢迎开启机器学习之旅！首先需要对机器学习有一个概貌性的了解：机器学习是如何解决实际问题的？它的关键步骤是什么？常用的算法有哪些？在实际操作中要注意哪些问题？工业界是如何搭建机器学习系统的？通过本章的学习，相信大家对机器学习会有一个更加科学、全面的了解。

1.1 机器学习基本流程

用机器学习解决实际工程中的问题可分五步走：

step 1. 建模，即把实际问题转化为一个数学问题。

step 2. 选择算法，机器学习算法繁多，同样的问题可以用多种算法来解决，而且新算法还在不断地涌现。

step 3. 确立优化目标，所有的机器学习问题最终都转变为一个最优化问题，比如最小化均方误差，或者最大化似然函数等。

step 4. 学习迭代，采用某种优化算法 (比如梯度下降) 去不停地更新参数，以逼近目标函数的最优解。

step 5. 效果评估，回归实际问题，用恰当的指标来评估模型的好坏。

建模可不是一件容易的事，以电商推荐为例，首先要把目标定得非常明确，是以提高点击量为目标，还是以提高成交量为目标，又或是以提高成交额为目标？假设目标是点击量，那就预测用户点击每一件商品的概率，把点击概率最高的排在最前面。从直观上看预测点击率是一个回归问题，其实也可以把它看成是一个二分类问题，即用户点还是不点。不论是分类问题还是回归问题，每次预测只需要关注一件商品，这种方法被称为 pointwise。还有一个思路是 pairwise，即把商品两两组合，预测用户喜欢前者高于后者的概率。

机器学习的发展主要是算法的发展，在 1.2 节会对主流的算法做一个梳理和总结。

对于概率问题经常用极大化似然函数作为目标，回归问题多用平方误差作为损失函数，而分类问题则多用交叉熵损失函数。

要极大化目标函数或极小化损失函数有很多种优化方法可以选择，第 4 章会专门介绍。

模型的评估和它的目标函数可以是两回事，也可以是一回事。模型评估的重点是从实际问题出发，目标函数的设立当然也是为了解决实际问题，但它还要兼顾数学问题的可解性。比如对于分类问题，目标函数通常用交叉熵，评估模型时用 AUC (Area Under Curve) 指标，又比如推荐问题，模型训练时用交叉熵或平方误差作为目标函数，评估时用 NDCG (Normalized Discounted Cumulative Gain) 指标，当然也可以直接用 AUC 或 NDCG 作为目标函数去训练模型，但那样做，优化算法会变得比较复杂。

1.2 业界常用算法

每年人工智能领域的论文浩如烟海，对于一个算法初学者来说时常不知道该从何下手，笔者在为本书挑选算法时遵循以下三个原则。

1) 在工业界广泛使用。有些算法虽然理论完美，效果也确实不错，但它所适用的数据特征、规模以及所需的计算资源与绝大多数企业面临的实际情况不符。

2) 具有理论代表性。读者掌握了这些典型算法后，再去学习其他同类型的算法就会容易许多。本书算法章节的编写力求由浅入深、推理完整、前后连贯、自成体系。

3) 同类算法给出实验对比。比如在线性模型、文本向量化模型、决策树模型中都给出了实验对比，这样理论结合实践，帮助读者对各种算法的特点有一个感性的认知。

线性模型古老而又生命力顽强，逻辑回归 (Logistic Regression) 是不得不提的一个算法。模型不足，特征来补，由于线性模型很简单，所以需要通过组合更复杂的特征来提高模型的表达能力。

隐马尔可夫模型和条件随机场是序列挖掘算法中的代表，序列挖掘应用十分广泛，比如分词、命名实体识别、语音识别等。近年来虽然深度学习在序列挖掘领域成绩斐然，但深度学习与条件随机场的结合已经成为新的趋势。

绝大多数的自然语言处理 (Natual Language Processing，NLP) 问题都绕不

开对文本进行向量化表示，有时需要把词表示成向量，有时需要把段落表示成向量。word2vec、fastText、GloVe 是词向量化领域的三柄长剑，笔者比较钟爱 fastText。

决策树的表达能力天然地优于线性模型，实践中应用决策树时一般都是森林，很少会使用单棵树，因为森林类的算法更健壮。对于简单的问题，随机森林和 AdaBoost 就可以获得非常好的效果，当特征比较多时给大家推荐一剂“灵丹妙药”——XGBoost。LightGBM 对 XGBoost 又做了升级改进，调参虽然变得复杂了一些，但速度和精度确实要超越 XGBoost。LightGBM 是微软亚洲研究院的作品，他们还推出了 LightLDA 和 LightRNN，目的都是为了节约计算资源，提高训练速度。

如今深度学习已成为新的风潮，人们正投入极大的热情设计各种复杂的网络来解决机器学习领域曾经和正在遇到的所有问题。笔者想提醒机器学习的初学者们，深度学习虽然强大，但它的理论体系没有传统算法那么优雅完整，所以传统的算法还是要学习的。关于深度学习本书将介绍卷积神经网络、循环神经网络和注意力机制，它们在工程实践中已经被广泛应用，并且可以比较轻松地获得更优的效果。同时会介绍 Keras 网络编程从入门到进阶的一些技能。

推荐系统是机器学习在工业界非常典型的应用案例，笔者将用一章的篇幅带领大家经历一个完整的推荐项目开发流程，包括从前期的算法调研到最终的服务上线。

1.3　构建机器学习系统

在工业界搞算法受到很多因素的制约。首先，如果没有足够多的、纯的样本，采用再先进的算法也不能获得好的预测效果，相反如果有足够多的样本数据，即使采用简单的模型也可能获得很好的效果，所以样本数据的收集是算法工程师面临的第一大难题。有了大量的样本数据，还需要能够快速地训练出模型，如果跑一次实验需要几天甚至一个月的时间，这在工业界是不可接受的，这时，一个分布式的训练系统就能派上大用场。模型离线取得的指标 (准确率、AUC 等) 跟线上实际运行时的指标往往是不一样的，如果代码有 bug，那么线上指标可能会很差，所以必须对模型的线上指标进行监控，而且是实时监控，另外在对比两个算法的好坏时最终都要以线上指标为准。

为了算法能实施落地，充分发挥其威力，需要构建一套完善的机器学习系统，图 1-1 是一个比较通用的构架。前端即手机 App 或 PC 网站，与用户直接相连。前端把请求上下文发给算法服务，算法服务预测出用户最感兴趣的物品，并返回给前端，请求上下文中包含用户的 ID、IP、地理位置等信息。算法服务在做预测时需要获取两样东西：模型和特征。如果线上同时运行着多个模型，需要由 A/B 系统来决定针对当前用户采用哪个模型。用户特征和物品特征是离线计算好的，其交叉特征需要实时计算，另外对算法而言，请求上下文也属于特征的一部分。日志收集器实时地收集特征以及展现点击数据，按物品 ID 对它们进行拼接，构成正负样本 (比如展现的物品被点击的是正样本，没被点击的是负样本)。这些样本送给模型训练器，训练器采用 SGD(Stochastic Gradient Descent)、FTRL(Follow the Regularized Leader) 等方法实时地更新模型。同时日志收集器负责计算模型的在线指标，发给监控系统进行展示，当在线指标低于阈值时，监控系统通过短信、邮件等方式通知开发人员。

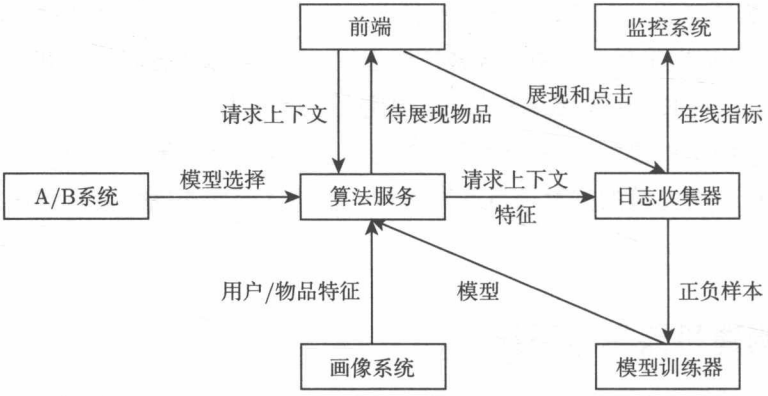


图 1-1　机器学习系统架构

本书最后三章侧重于工程化部分，详细描述如何搭建日志收集系统、A/B 测试系统，以及如何进行模型的分布式训练。这三章实战性很强，所以会展示比较详尽的核心代码，由于 Python 在机器学习领域最为流行，大部分读者都已掌握了这门语言，所以本书全部采用 Python 来做代码演示，但是在搭建线上系统，尤其是 CPU 密集型的算法任务时，一般采用 C++、Go 这类高效的语言，其速度可能是 Python 的成百上千倍。

第 2 章 统计学

统计学是人类总结过去、对历史经验进行度量刻画的一种方法。在机器学习中，统计学的思想是无处不在的，甚至可以说一些机器学习算法就是统计学的高级封装和复杂应用。

2.1 概率分布

在统计学里万物都是随机变量，概率分布是对随机变量的基本描述。如果我们掌握了随机变量的概率分布函数，那么就掌握了它的一切，就可以对它进行预测，当然这种预测都是有概率的，不是确定性的。

2.1.1 期望与方差

如果 $\int_{-\infty}^{\infty}|x|f(x)\mathrm{d}x<\infty$，那么 $\mathrm{E}(x)=\int_{-\infty}^{\infty}xf(x)\mathrm{d}x$；如果积分发散，则期望不存在。

方差 $\mathrm{Var}(X)$ 可以看作是 $\big[X-\mathrm{E}(X)\big]^2$ 的期望。

$$\mathrm{Var}(X)=\mathrm{E}\big[X-\mathrm{E}(X)\big]^2$$

乍一看计算方差需要遍历两次样本，第 1 次遍历算出 X 的期望 $\mathrm{E}(X)$，第 2 次遍历算出 $\big[X-\mathrm{E}(X)\big]^2$ 的期望，实际上我们对公式稍作变换就能发现，只需一次遍历就可以计算出方差。由于 $\mathrm{E}(X)$ 是常数，$\mathrm{E}(X)^2$ 也是常数，常数的期望还是这个数本身，所以 $\mathrm{E}\big(\mathrm{E}(X)\big)=\mathrm{E}(X)$，$\mathrm{E}\big(\mathrm{E}(X)^2\big)=\mathrm{E}(X)^2$。

$$\begin{aligned}\mathrm{Var}(X)&=\mathrm{E}\big[X-\mathrm{E}(X)\big]^2\\&=\mathrm{E}\left[X^2-2X\,\mathrm{E}(X)+\mathrm{E}(X)^2\right]\\&=\mathrm{E}\left(X^2\right)-2\,\mathrm{E}(X)\,\mathrm{E}\left(\mathrm{E}(X)\right)+\mathrm{E}\left(\mathrm{E}(X)^2\right)\\&=\mathrm{E}(X^2)-2\,\mathrm{E}(X)\,\mathrm{E}(X)+\mathrm{E}(X)^2\end{aligned}$$

$$=\mathrm{E}(X^2)-\mathrm{E}(X)^2$$

可见，只需一次遍历既可以算出 $\mathrm{E}(X^2)$，也可以算出 $\mathrm{E}(X)$。

代码 2-1　计算均值和方差

```
def mean_Var(arr):
    dim = len(arr)
    if dim == 0:
        return (0, 0)
    sumOrig = 0.0
    sumSquare = 0.0
    for ele in arr:
        sumOrig += ele
        sumSquare += ele**2
    mean = sumOrig / dim
    variance = sumSquare / dim - mean**2
    return (mean, variance)
```

定义样本的均值 $\overline{X}$ 和方差 S^2：

$$\overline{X}=\frac{\sum_{i=1}^{n}X_i}{n}$$

$$S^2=\frac{1}{n-1}\sum_{i=1}^{n}(X_i-\overline{X})^2$$

定理 2.1 (大数定理)　当样本总量足够大时，样本均值 $\overline{X}$ 趋于总体期望 μ。

定义 2.1　对于统计量 θ，若 $\mathrm{E}(\hat{\theta})=\theta$，则 $\hat{\theta}$ 是 θ 的无偏估计

定理 2.2　样本均值 $\overline{X}$ 是总体期望 μ 的无偏估计。

证明：

$$\mathrm{E}\left(\overline{X}\right)=\mathrm{E}\left(\frac{1}{n}\sum_{i=1}^{n}X_i\right)=\frac{1}{n}\sum_{i=1}^{n}\mathrm{E}(X_i)$$

$$=\frac{1}{n}\sum_{i=1}^{n}\mu=\frac{1}{n}n\mu=\mu$$