TURING 图灵原创

美团技术丛书

美团算法团队◎著

美团机器学习实践

- 美团AI+O2O智慧结晶
- 机器学习算法落地实践
- 涵盖搜索、推荐、风控、计算广告、图像处理领域

中国工信出版集团

人民邮电出版社
POSTS & TELECOM PRESS

美团算法团队由数百名优秀算法工程师组成，负责构建美团这个生活服务互联网大平台的“大脑”，涵盖搜索、推荐、广告、风控、机器学习、计算机视觉、语音、自然语言处理、智能调度、机器人和无人配送等多个技术方向，在帮助美团数亿活跃用户改善用户体验的同时，也帮助餐饮、酒店、婚庆、丽人、亲子等200多个行业的数百万商户提升运营效率。我们致力于通过算法和人工智能技术，帮大家吃得更好，活得更好。

更多详情请关注微信公众号：meituantech。

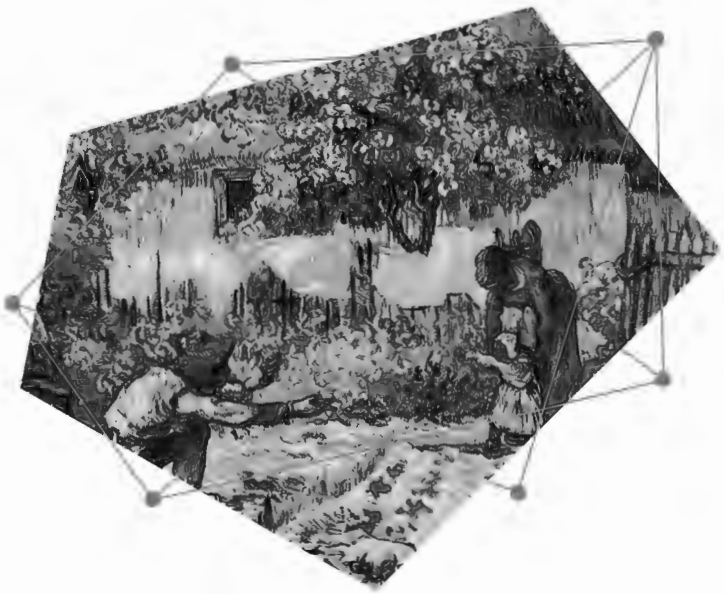
数字版权声明

TURING 图灵原创

美团技术丛书

美团机器学习实践

美团算法团队◎著



人民邮电出版社

北京

图书在版编目（CIP）数据

美团机器学习实践 / 美团算法团队著. -- 北京 : 人民邮电出版社, 2018.8
（图灵原创）
ISBN 978-7-115-48463-5

Ⅰ. ①美… Ⅱ. ①美… Ⅲ. ①机器学习－应用－网络营销 Ⅳ. ①F713.365.2

中国版本图书馆CIP数据核字(2018)第086804号

内 容 提 要

人工智能技术正以一种前所未有的速度深刻地改变着我们的生活，引导了第四次工业革命。美团作为国内 O2O 领域领先的服务平台，结合自身的业务场景和数据，积极进行了人工智能领域的应用探索：在美团的搜索、推荐、计算广告、风控和图像处理等领域，相关的人工智能技术得到广泛的应用。本书包括通用流程、数据挖掘、搜索和推荐、计算广告、深度学习以及算法工程 6 大部分内容，全面介绍了美团在多个重要方面对机器学习的应用。

本书非常适合有一定机器学习基础的工程技术人员和在校大学生学习和阅读。通过本书，有经验的算法工程师可以了解美团在这方面的做法，在校大学生可以学习机器学习算法如何在具体的业务场景中落地。

◆ 著　　美团算法团队
责任编辑　陈兴璐
责任印制　周昇亮
◆ 人民邮电出版社出版发行　　北京市丰台区成寿寺路11号
邮编　100164　　电子邮件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
北京　　印刷
◆ 开本：800×1000　1/16
印张：20
字数：450千字　　2018年 8 月第 1 版
印数：1－4 000册　　2018年 8 月北京第 1 次印刷

定价：79.00元

读者服务热线：(010)51095186转600　印装质量热线：(010)81055316
反盗版热线：(010)81055315
广告经营许可证：京东工商广登字 20170147 号

序

被邀请作为美团工程师的代表为本书写篇简单的序言，我深感荣幸。

本书是一本关于机器学习和数据挖掘在真实的业务场景如何落地、如何发挥作用的书。它是美团的算法工程师们利用工作之余的时间，集体创作完成的。作者来自美团公司的各个部门，既包括负责用户画像、文本图像理解这样偏基础的研发部门，也包括广告、搜索以及推荐这样的产品研发团队。本书的写作内容和过程也充分体现了美团工程师团队的气质——踏实务实，同时又热爱学习和分享。

机器学习以及数据挖掘相关技术如今在美团公司内的几乎所有环节和场景都有应用，从直接关系到用户体验的搜索和推荐，再到提高配送人员效率的智能配送和调度算法，以及商家端的广告系统和智能选址等服务，甚至App的bug分类，这些你能想到或者不能想到的机器学习乃至人工智能相关技术都发挥了巨大的价值。当然，我们做的这些工作还远远不够，越是在O2O领域持续工作，我们越能感受到人工智能技术可能给这个行业带来的巨大改变和广阔前景。机器学习、运筹调度、IOT、AR、语音以及视觉感知等所有这些方向，都还有巨大的技术挑战和应用空间等着我们去突破，从而让人们“吃得更好，活得更好”。

和传统的机器学习相关的理论教科书相比，本书侧重于这些理论如何在真实的业务场景落地，所使用的都是美团公司内的真实案例。这也是我们编写本书的初衷。我们注意到在这个人工智能技术成为中国的国家战略的时代，有很多卓越的国内外学者贡献了大量的机器学习和人工智能的理论书籍，但作为第四代工业革命浪潮的代表技术，其在各行各业内的具体应用案例和工程实践也同样重要，而这方面的书籍是相对较少的。希望我们在这本书中的分享能够起到抛砖引玉的作用，同时也能在这方面给广大读者带来一定的收获。

最后，也非常欢迎对本书有任何建议或者意见的读者，联系美团技术团队。机器学习以及人工智能技术，一方面理论还在飞速发展，另一方面新的应用也层出不穷。作为这方面从业者的我们，尤其希望和大家产生交流和碰撞。我们深信，交流和碰撞是促使我们进步的动力。

张锦懋

美团首席科学家

前　言

人工智能技术正以前所未有的速度深刻地改变着我们的生活，引导了第四次工业革命。在这次技术革命中，为了抢占人工智能发展的战略机遇，构筑我国在人工智能的领先优势，国务院制定了新一代人工智能发展规划，体现了我国政府对人工智能的高度重视。未来这个领域将迎来重大的发展机遇，同时也面临着巨大的挑战，这就对每一位人工智能领域的从业人员和有志于在这个领域发展的科技人员的技术水平和专业领域知识提出了更高的要求。

机器学习是人工智能领域最重要的方向之一，它分为三个主要的研究领域：监督学习、非监督学习和强化学习。监督学习可以细分为分类和回归，它需要有样本标注，样本的质量和规模决定了模型的复杂度和效果，这也是为什么人工智能需要大数据作为支撑的重要原因。监督学习是目前应用最广泛的一种机器学习方法，比如我们常见的广告点击率预估、商品推荐、搜索排序等。非监督学习可以细分为聚类、降维等方向，它可以发掘在大量未标注数据中的规律。强化学习是智能系统从环境到行为映射的学习，以使奖励函数值最大，被认为是最接近人类的学习行为，在工业控制、机器人行为决策等领域得到广泛的应用。

近年来深度学习的提出和普及，使得一些在传统的机器学习领域解决不好的问题得到极大的改善，比如图像识别ImageNet使分类的错误率已经缩小到原来的1/10，并超过了人类的识别准确率。深度学习是目前人工智能领域发展最为活跃的领域。大量的模型和理论不断地涌现，比如媒体常报道的机器画画就是GAN模型的应用。还有所说的机器作诗、机器写新闻，也是基于深度学习的RNN模型。深度学习已经完全统治了图像和语音识别的机器学习领域，并且在自然语言处理领域也在不断发掘新的应用。深度学习和强化学习相结合极大地影响了强化学习领域，采用深度网络来改造强化学习中的函数值拟合，取得了非常不错的效果，比如大名鼎鼎的Alpha Go和Alpha Zero的本质都是深度强化学习的应用。深度学习领域现在还在迅速发展之中。反向传播是深度学习的根基之一，有几十年的使用历史。但是最近深度学习之父Hinton呼吁对反向传播保持怀疑态度，并提出了新的Capsule网络。传统的神经网络中，每一个神经元输入和输出都是标量，而Capsule网络中是一个或一组向量，每一层之间通过迭代路由协议机制激活更高层的Capsule。这有可能成为深度学习领域的重大变革。

美团作为国内O2O领域领先的服务平台，结合自身的业务场景和数据，积极进行了人工智能领域的应用探索。在美团的搜索、推荐、计算广告、风控、图像处理等领域，相关的人工智能技术得到广泛的应用，并取得了不错的效果。我们组建了算法技术通道，并制定了相关的课程体系

和分享机制。经过多年的努力，美团在人工智能和O2O的结合上，积累了丰富的经验。写作本书的目的之一就是与业界分享这些经验，共同推进AI + O2O的发展。

本书分为6大部分，全面介绍了美团在多个重要方面对机器学习的应用。

- 第一部分是通用流程，包括第1~4章。这里讲述了机器学习解决实际问题的通用流程：如何分析问题，如何进行特征工程、常见模型的比较和选择，以及如何进行效果评测；最后还介绍了在各类机器学习竞赛中常用的模型融合技巧。
- 第二部分是数据挖掘，包括第5~7章。用户画像在业务上有着重要的作用，是个性化推荐排序的基础。曾经出现网上流传的百度内部截图、搜狗上市新闻为什么没有推荐给CEO的情况，解决这类问题的关键在于用户画像技术。这里详细介绍了美团在这方面的实践。实体链接是知识图谱和POI数据建设的重要基础，评论挖掘是UGC内容挖掘的常见应用，这里也介绍了我们关于UGC内容挖掘的做法。
- 第三部分是搜索和推荐，包括第8~10章。不同于全网网页搜索、垂直搜索和商品搜索，O2O领域的搜索排序有着自身的特点，面临的挑战也存在差异。本部分分享了关于搜索排序中常见的查询分析、用户意图识别、机器学习排序的做法和实践。推荐在O2O场景下有着非常关键的作用，最后对推荐部分也作了介绍。
- 第四部分是计算广告，包括第11章和第12章。计算广告是互联网目前主流的盈利模式之一，这里从广告设计的机制特点、定向方式、用户偏好、损失建模等方面，详细地介绍我们在这个领域的实践。
- 第五部分是深度学习，包括第13~15章。这里介绍了美团在计算机视觉和自然处理领域的深度学习实践。深度学习在业务上的应用非常多，限于篇幅，我们主要分享了在图像分类、OCR识别、图像质量优化、情感分析、机器学习排序方面的应用。
- 第六部分是算法工程，包括第16章和第17章。机器学习算法要在实际应用中更好地落地，相关的工程也非常重要。这里我们主要介绍了在大规模机器学习、特征的生产和监控、模型线上效果实验和评测等方面的工作。

本书并不是一本机器学习的理论教材，它的内容非常广泛，主要侧重工业界的业务实践。本书非常适合有一定机器学习基础的工程技术人员和在校大学生学习和阅读。通过阅读本书，有经验的算法工程师可以了解美团在这方面的做法，在校大学生可以学习机器学习算法如何在具体的业务场景中落地。

本书内容涉及美团多个事业群的工作，得到了美团技术委员会、技术学院和算法通道的大力支持。非常感谢参与本书编写和校对的算法工程师们，你们平时的工作已非常繁忙，正是因为你们利用自己的休息时间辛勤地参与本书的编写和校对，无私地分享自己的经验和智慧，本书才得以完成。

本书由陈华清统一规划、整理、主持编写。参与本书写作的作者还有易根良、陈振、石晓巍、聂鹏宇、曲思聪、袁博、朱日兵、仙云森、周翔、唐金川、刘铭、曹浩、戚亦平、魏晓明、蒋前

程、付晴川、雷军、李彪、燕鹏、顾昊和王磊。本书从开始规划、斟酌内容、反复修改，到最终定稿，历时一年的时间。在此对参与写作的所有作者们表示诚挚的敬意和感谢。

陈华清

2018年5月

目　录

第一部分　通用流程

第二部分 数据挖掘

第三部分 搜索和推荐

第六部分 算法工程

第一部分

通用流程

- 第 1 章　问题建模
- 第 2 章　特征工程
- 第 3 章　常用模型
- 第 4 章　模型融合

第 1 章 问题建模 1

随着大数据时代的到来，机器学习成为解决问题的一种关键工具。不管是在工业界还是在学术界，机器学习都是炙手可热的方向。但是学术界和工业界对机器学习的研究各有侧重。学术界侧重于对机器学习理论的研究，工业界侧重于如何用机器学习来解决实际问题。我们结合机器学习的实践经验，详细介绍机器学习解决问题的整个流程。图1-1即为我们概括的机器学习解决问题的通用流程，通用流程主要分为4大部分。

- **问题建模**。解决一个机器学习问题都是从问题建模开始。首先需要收集问题的资料，深入理解问题，然后将问题抽象成机器可预测的问题。在这个过程中要明确业务指标和模型预测目标，根据预测目标选择适当的评估指标用于模型评估。接着从原始数据中选择最相关的样本子集用于模型训练，并对样本子集划分训练集和测试集，应用交叉验证的方法对模型进行选择和评估。
- **特征工程**。完成问题建模、对数据进行筛选和清洗之后的步骤，就是对数据抽取特征，即特征工程。特征工程是一项很重要但又很困难的任务，不仅需要我们对模型和算法有深入的理解，还需要我们有很扎实的专业领域知识。工业界大多数成功应用机器学习的问题，都是在特征工程方面做得很好。虽然不同模型和不同问题都会导致特征工程差异很大，但仍有很多特征工程的技巧可以通用。
- **模型选择**。我们进行特征工程是为了将特征输入给模型，让模型从数据中学习规律。但是模型有很多，不同的模型有很大差别，使用场景不同，能够处理的特征也有很大差异。当我们经过特征工程得到一份高质量的特征之后，还需要考虑哪个模型能够更准确地从数据中学习到相应规律。从众多模型中选择最佳的模型也需要对模型有很深入的理解。
- **模型融合**。如上所言，不同模型会有很大差别，能够从数据中学到的规律也会不同。我们可以采用模型融合的方法，充分利用不同模型的差异，以进一步优化目标。

后面的章节会详细介绍特征工程、模型选择和模型融合。本章主要介绍问题建模，包括评价指标、样本选择和交叉验证等内容。

从机器学习的发展现状来看，很多机器学习从业者在处理问题时是直接进行特征工程和模型选择，而忽略了问题建模。问题建模是十分重要的一个环节，必不可少。评价指标很多，我们应

该选择一个能跟业务指标波动一致的评估指标，这样通过观察评估指标就能判断模型效果，可以大大提高模型迭代效率。否则评估指标都没有参考意义。有一个好的评估指标未必足够，还需要选择一种好的交叉验证方法，比如对只有100条样本的测试集评估准确率，模型A准确率100%，模型B准确率95%，因为测试集太小，得到的准确率不能充分代表模型的好坏，我们无法确定模型A一定比模型B好。同样我们的原始数据不可避免地会有异常数据，比如系统异常导致日志记录错误，将异常数据和低质量数据用于模型训练势必会导致模型效果变差，通过样本选择提高数据质量能起到事半功倍的效果。

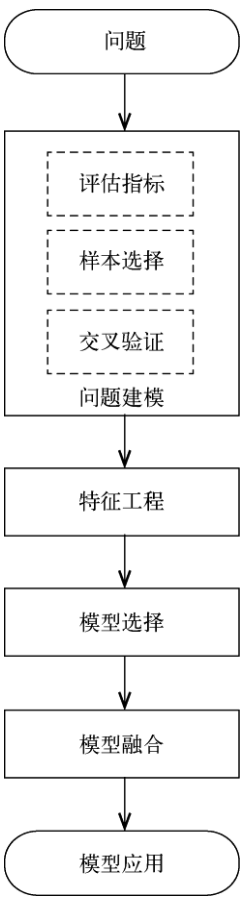


图1-1 机器学习通用流程图

1.1 评估指标

评估指标用于反映模型效果。在预测问题中，要评估模型的效果，就需要将模型预测结果$f(X)$和真实标注Y进行比较，评估指标定义为$f(X)$和Y的函数。

$$\text{score} = \text{metric}(f(X), Y)$$

模型的好坏是相对的，在对比不同的模型效果时，使用不同评估指标往往会导致不同的结论。

通常，线下使用的是机器学习评估指标，线上使用的是业务指标。如果线下指标和线上指标不同，则可能会出现线下指标变好而线上指标变差的现象。为此，在一个新问题的开始阶段，都会进行多轮模型迭代，来探索与线上业务指标一致的线下指标，尽可能使线下指标的变化趋势跟线上指标一致。没有一个跟线上一致的线下指标，意味着线下指标没有参考价值，想判断此次试验是否有效，只能上线实验。而上线实验成本远高于离线实验成本，通常需要在线实验较长时间并对效果进行可信度检验（如t-test）之后才能得出结论，这必然会导致模型迭代进度变慢。

评估指标根据任务类型分类，可分为分类指标、回归指标、聚类指标和排序指标等。下面将介绍一些常用的评估指标以及它们适用场景。

1.1.1　分类指标

1. 精确率和召回率

精确率和召回率多用于二分类问题，可结合混淆矩阵介绍，如表1-1所示。

表1-1　混淆矩阵

	预测结果		
真实结果		正（P）	负（N）
	正（P）	TP	FN
	负（N）	FP	TN

其中，TP（真正，True Positive）表示真实结果为正例，预测结果也是正例；FP（假正，False Positive）表示真实结果为负例，预测结果却是正例；TN（真负，True Negative）表示真实结果为正例，预测结果却是负例；FN（假负，False Negative）表示真实结果为负例，预测结果也是负例。显然，TP + FP + FN + TN = 样本总数。

精确率P和召回率R的定义为：

$$\text{精确率}（P）= \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{召回率}（R）= \frac{\text{TP}}{\text{TP} + \text{FN}}$$

理想情况下，精确率和召回率两者都越高越好。然而事实上这两者在某些情况下是矛盾的：精确率高时，召回率低；而精确率低时，召回率高。比如在搜索网页时，如果只返回最相关的那一个网页，那精确率就是100%，而召回率就很低；如果返回全部网页，那召回率为100%，而精确率就很低。因此在不同场合需要根据实际需求判断哪个指标更重要。

我们以召回率R为横轴、以精确率P为纵轴能够画出P-R曲线，如图1-2所示。P-R曲线越靠近右上角性能越好，曲线下的面积叫AP分数（Average Precision Score，平均精确率分数）。对比不同模型的AP分数，能在一定程度上反映模型的精确率和召回率都高的比例。但这个值计算不方便，人们设计了一些综合考虑精确率和召回率的指标。

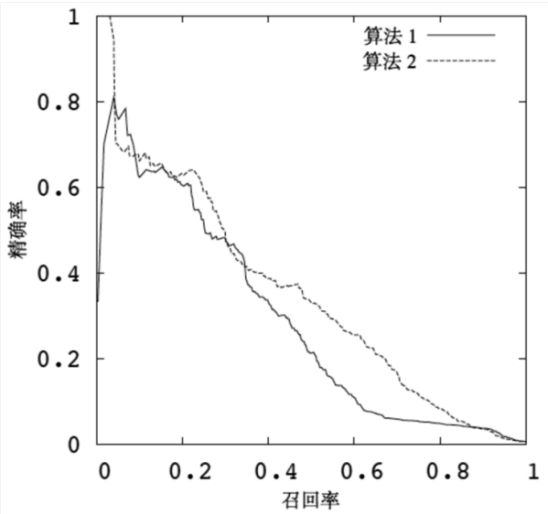


图1-2 P-R曲线

F_1值就是这样一个常用的指标。F_1值是精确率和召回率的调和平均值：

$$\frac{2}{F_1}=\frac{1}{P}+\frac{1}{R}$$

F值可泛化为对精确率和召回率赋不同权重进行加权调和：

$$F_\alpha=\frac{(1+\alpha^2)\cdot P\cdot R}{\alpha^2\cdot P+R}$$

此外，准确率和错误率也是常用的评估指标。

$$\text{准确率（accuracy）}=\frac{\text{TP}+\text{TN}}{\text{TP}+\text{FP}+\text{FN}+\text{TN}}$$

$$\text{错误率（error rate）}=\frac{\text{FP}+\text{FN}}{\text{TP}+\text{FP}+\text{FN}+\text{TN}}$$

精确率和准确率是比较容易混淆的两个评估指标，两者是有区别的。精确率是一个二分类指标，而准确率能应用于多分类，其计算公式为：

$$\text{准确率（accuracy）}=\frac{1}{n}\sum_{i=1}^{n}I(f(x_i)=y_i)$$

2. ROC与AUC

在众多的机器学习模型中，很多模型输出是预测概率。而使用精确率、召回率这类指标进行模型评估时，还需要对预测概率设分类阈值，比如预测概率大于阈值为正例，反之为负例。这使得模型多了一个超参数，并且这个超参数会影响模型的泛化能力。

接收者操作特征（Receiver Operating Characteristic，ROC）曲线不需要设定这样的阈值。ROC曲线纵坐标是真正率，横坐标是假正率，如图1-3所示。其对应的计算公式为：

$$\text{真正率（TPR）} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{假正率（FPR）} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

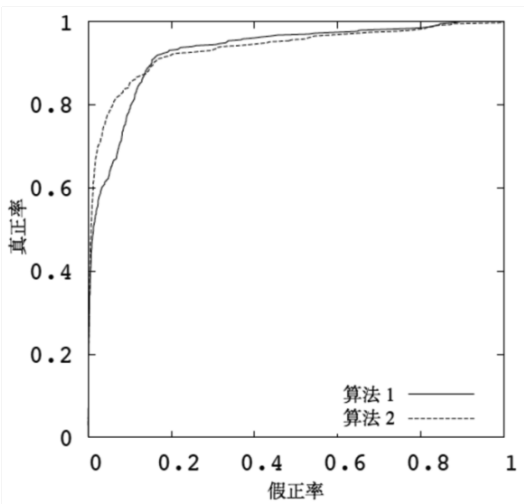


图1-3　ROC曲线

ROC曲线与P-R曲线有些类似。ROC曲线越靠近左上角性能越好。左上角坐标为(0,1)，即FPR=0，TPR=1，根据FPR和TPR公式可以得知，此时FN=0，FP=0，模型对所有样本分类正确。绘制ROC曲线很简单，首先对所有样本按预测概率排序，以每条样本的预测概率为阈值，计算对应的FPR和TPR，然后用线段连接。当数据量少时，绘制的ROC曲线不平滑；当数据量大时，绘制的ROC曲线会趋于平滑。

AUC（Area Under Roc Curve）即ROC曲线下的面积，取值越大说明模型越可能将正样本排在负样本前面。AUC还有一些统计特性：AUC等于随机挑选一个正样本（P）和负样本（N）时，分类器将正样本排前面的概率；AUC和Wilcoxon Test of Ranks等价；AUC还和基尼（Gini）系数有联系，满足等式Gini + 1 = 2 · AUC。

AUC的计算方法有多种，从物理意义角度理解，AUC计算的是ROC曲线下的面积：

$$\mathrm{AUC}=\sum_{i\in(P+N)}\frac{(\mathrm{TPR}_i+\mathrm{TPR}_{i-1})\cdot(\mathrm{FPR}_i-\mathrm{FPR}_{i-1})}{2}$$

从概率意义角度理解，AUC考虑的是样本的排序质量，它与排序误差有密切关系，可得到计算公式：

$$\mathrm{AUC}=\frac{\sum_{i\in P}\mathrm{rank}_i-\frac{|P|\cdot(|P|+1)}{2}}{|P|\cdot|N|}$$

其中，rank为样本排序位置从1开始，$|P|$为正样本数，$|N|$为负样本数。

AUC计算主要与排序有关，所以它对排序敏感，而对预测分数没那么敏感。

3. 对数损失

对数损失（Logistic Loss，logloss）是对预测概率的似然估计，其标准形式为：

$$\mathrm{logloss}=-\log P(Y|X)$$

对数损失最小化本质上是利用样本中的已知分布，求解导致这种分布的最佳模型参数，使这种分布出现概率最大。

对数损失对应的二分类的计算公式为：

$$\mathrm{logloss}=-\frac{1}{N}\sum_{i=1}^{N}(y\cdot\log p_i+(1-y)\cdot\log p_i)$$

其中，$y\in\{0,1\}$，p_i为第i条样本预测为1的概率。

对数损失在多分类问题中也使用广泛，其计算公式为：

$$\mathrm{logloss}=-\frac{1}{N}\cdot\frac{1}{C}\cdot\sum_{i=1}^{N}\sum_{j=1}^{C}y_{ij}\cdot\log p_{ij}$$

其中，N为样本数，C为类别数，$y_{ij}=1$表示第i条样本的类别为j，p_{ij}为第i条样本类别j的概率。

logloss衡量的是预测概率分布和真实概率分布的差异性，取值越小越好。与AUC不同，logloss对预测概率敏感。

1.1.2 回归指标

1. 平均绝对误差

平均绝对误差（Mean Absolute Error，MAE），也叫L_1范数损失（L_1-norm Loss），其公式为：

$$\mathrm{MAE}=\frac{1}{N}\cdot\sum_{i=1}^{N}|y_i-p_i|$$

其中，N为样本数，y_i为第i条样本的真实值，p_i为第i条样本的预测值。MAE是绝对误差的平均值，因为预测误差有正有负，绝对值可以避免正负抵消。MAE能很好地刻画预测值与真实值的偏差。模型使用MAE作为损失函数则是对数据分布的中值进行拟合。某些模型（如XGBoost）必须要求损失函数有二阶导数，所以不能直接优化MAE。

加权平均绝对误差（Weighted Mean Absolute Error，WMAE）是基于MAE的变种评估指标，对每条样本考虑不同的权重，比如考虑时间因素，离当前时间越久的样本权重越低。其计算公式为：

$$\mathrm{WMAE}=\frac{1}{N}\cdot\sum_{i=1}^{N}w_i|y_i-p_i|$$

其中，w_i是第i条样本的权重。

2. 平均绝对百分误差

平均绝对百分误差（Mean Absolute Percentage Error，MAPE）的公式为：

$$\mathrm{MAPE}=\frac{100}{N}\cdot\sum_{i=1}^{N}\left|\frac{y_i-p_i}{y_i}\right|,\ y_i\neq 0$$

MAPE通过计算绝对误差百分比来表示预测效果，其取值越小越好。如果MAPE = 10，这表示预测平均偏离真实值10%。MAPE计算与量纲无关，因此在特定场景下不同问题具有一定可比性。MAPE的缺点也比较明显，在$y_i=0$处无定义，并且如果y_i接近0可能导致MAPE大于100%。而且，MAPE对负值误差的惩罚大于正值误差。基于这些缺点也有一些改进的评价指标，如MASE、sMAPE、MDA。

3. 均方根误差

均方根误差（Root Mean Squared Error，RMSE）的公式为：

$$\mathrm{RMSE}=\sqrt{\frac{1}{N}\cdot\sum_{i=1}^{N}(y_i-p_i)^2}$$

RMSE代表的是预测值和真实值差值的样本标准差。和MAE比，RMSE对大误差样本有更大的惩罚；但它也对离群点敏感，其健壮性不如MAE。模型使用RMSE作为损失函数则是对数据分布的平均值进行拟合。

基于均方根误差也有一个常用的变种评估指标叫均方根对数误差（Root Mean Squared Logarithmic Error，RMSLE），其公式为：

$$\text{RMSLE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\log(y_i+1)-\log(p_i+1))^2}$$

RMSLE对预测值偏小的样本惩罚比对预测值偏大的样本惩罚更大，比如一个酒店消费均价是200元，预测成150元的惩罚会比预测成250元的大。如果评估指标选用RMSLE，没办法直接优化RMSLE但是能直接优化RMSE的模型，通常会先对预测目标进行对数变换$y_{\text{new}} = \log(y+1)$，最后预测值再还原$p = \exp p_{\text{new}} - 1$。

1.1.3 排序指标

1. 平均准确率均值

平均准确率均值（Mean Average Precision，MAP）的公式分两部分计算，先计算一次排序的平均准确率，再计算总体的平均准确率。常用的MAP指标会限定评估排在前面的文档质量。

$$\text{AP@}K = \frac{\sum_{k=1}^{\min(M,K)} P(k)\cdot \text{rel}(k)}{\min(M,K)}$$

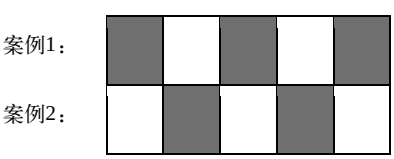
$$P(i) = \frac{\text{前}i\text{个结果中相关文档数量}}{i}$$

其中，AP@K表示计算前K个结果的平均准确率；M表示每次排序的文档总数，可能一次返回文档数不足K个；$P(k)$表示前k个结果的准确率；$\text{rel}(k)$表示第k个结果是否是相关文档，相关取值为1，不相关取值为0。

$$\text{MAP@}K = \sum_{q=1}^{Q}\frac{\text{AP}_q\text{@}K}{Q}$$

其中，Q为查询的数量，$\text{AP}_q\text{@}K$为第q次查询的AP@K结果。

下面举个例子说明，其中，黑色代表相关，白色代表不相关。



案例1有$P(1)=\frac{1}{1}$，$P(3)=\frac{2}{3}$，$P(5)=\frac{3}{5}$；可以计算AP@5 $=\frac{1}{3}(1+\frac{2}{3}+\frac{3}{5})\approx 0.76$。

案例2有$P(2)=\frac{1}{2}$，$P(4)=\frac{2}{4}$；可以计算AP@5 $=\frac{1}{2}\left(\frac{1}{2}+\frac{2}{4}\right)=0.5$。

那么进一步计算MAP@5 $=\frac{1}{2}(0.76+0.5)=0.63$。

2. NDCG

NDCG（Normalized Discounted Cumulative Gain，归一化贴现累计收益）是常用的一个衡量排序质量的指标，其公式为：

$$\mathrm{DCG@}K = \sum_{k=1}^{K} \frac{2^{\mathrm{rel}_k} - 1}{\log_2(k+1)}$$

$$\mathrm{IDCG@}K = \sum_{k=1}^{|\mathrm{REL}|} \frac{2^{\mathrm{rel}_k} - 1}{\log_2(k+1)}$$

$$\mathrm{NDCG@}K = \frac{\mathrm{DCG@}K}{\mathrm{IDCG}_K}$$

其中，NDCG@K表示计算前K个结果的NDCG；rel_k表示第k个位置的相关性得分；IDCG@K是前K个排序返回结果集能得到的最佳排序结果，用于归一化DCG@K；|REL|为结果集按相关性排序后的相关性得分列表。

相对于MAP指标，描述相关性只用0/1二值描述，NDCG相关性度量则可分更多等级。比如网页排序中常用的5个等级使评分更丰富。但是相关性描述是一个超参数，需要人为定义。此外，NDCG还考虑了位置偏置，使不同位置权重不同。

下面用一个例子来帮助理解——计算NDCG@4，其中预定义$\mathrm{rel} = \{0,1,2\}$，取值越大说明越相关，如表1-2所示。

表1-2　示例：计算NDCG@4

文档最佳排序	doc1	doc2	doc3	doc4
最佳排序相应文档相关性rel_k	2	1	1	0
模型排序	doc3	doc2	doc1	doc4

可计算$\mathrm{IDCG@4} = 3 + \frac{1}{\log_2 3} + \frac{1}{2} + 0 \approx 4.13$，$\mathrm{DCG@4} = 1 + \frac{1}{\log_2 3} + \frac{1}{2} + 0 \approx 2.13$，因此可计算出$\mathrm{NDCG@}K \approx 0.52$。

1.2　样本选择

样本选择是数据预处理中非常重要的一个环节，主要是从海量数据中识别和选择相关性高的数据作为机器学习模型输入。样本选择的目的是从完整训练集T中选择一个子集$S \subset T$，子集S不再包含冗余样本，如图1-4所示。最理想的样本选择结果是，选择了最少量的训练集S，而模型效果依然不会变差，即满足$P(\mathrm{Algo}_S) = P(\mathrm{Algo}_T)$，其中，$P$表示模型评估函数，Algo表示机器学习模型。做样本选择主要有以下三点好处。

- 当数据量过大时，程序有时候会耗费大量计算资源和计算时间，有时候甚至不能正常运行。减小数据量能够缩减模型的运算时间，使某些因为数据量过大无法应用机器学习模型的问题变得可能。
- 全部的数据集包含丰富的信息，但是一个具体的问题，通常只需要选取一部分问题相关的信息，相关性太低的数据对解决特定问题可能没有帮助。
- 数据中几乎不可避免地会有噪声数据，这些噪声可能是系统原因导致数据有错误、重复等。通过去除噪声能提高训练集的数据质量，改善模型效果。

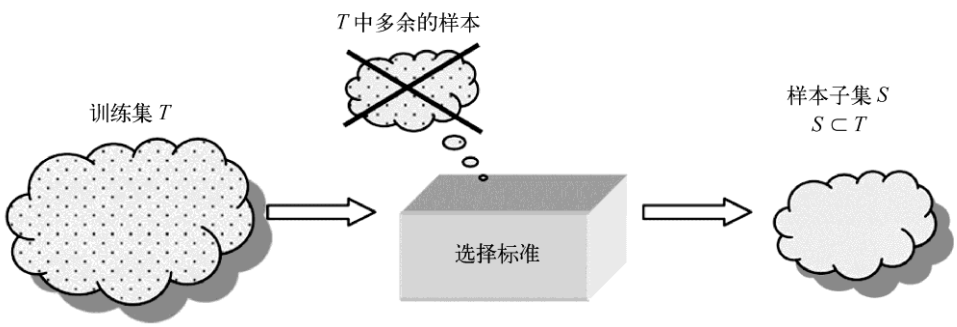


图1-4 样本选择流程图

样本选择有很多方法，数据去噪、采样是相对简单有效的方法，也有广泛的应用。当然还有很多方法不依赖采样，而是通过搜索整个数据集或利用算法来实现样本选择的，这类方法可总结为原型选择（Prototype Selection，PS）和训练集选择（Training Set Selection，TSS）。

1.2.1 数据去噪

数据中含有噪声数据几乎是不可避免的问题。噪声的存在会导致数据质量变低，影响模型的效果，但通过在训练集中引入噪声数据也能起到提升模型健壮性的作用。因此，包含噪声数据的问题是非常复杂的，特别是当选择的模型对噪声敏感的时候，问题会更严重。要进行去噪操作，对噪声进行识别是十分关键的一个步骤。识别出了噪声之后，可以采取直接过滤或者修改噪声数据等多种做法。

噪声在监督学习问题中影响明显，会改变特征和标注之间的关系，影响到特征提取；并且有噪声训练集和无噪声训练集得到的模型也会有差异。为此，提高模型健壮性，会使得模型对噪声数据不那么敏感。当需要处理噪声数据的时候，通常会权衡模型的健壮性和模型的效果。

噪声数据可能是特征值不对，比如特征值缺失、超出特征值域范围等；也可能是标注不对，比如二分类正样本标注成负样本。数据去噪很多是检测和去除训练数据中标注带噪声的实例，去除这样的噪声数据对实验结论是有帮助的；而去除特征带噪声的数据在很多地方表明效果反而变差，由此可见噪声特征带有的一定信息能够用于构建模型，比如特征缺失时，可以认为没有特征

也是一个特征，这也能描述一定的信息。

针对误标注实例有很多成功的处理方案，最常见的有集成过滤法（Ensemble Filter，EF）、交叉验证委员会过滤法（Cross-Validated Committees Filter，CVCF）和迭代分割过滤法（Iterative-Partitioning Filter，IPF）这三种方法，这些方法都是基于融合或者投票的思想进行数据过滤的。

除了这些过滤方法外，其实还会考虑就业务的本身性质做一些数据过滤工作，比如清洗爬虫数据和不具代表性样本等。再如过滤掉无效曝光数据，根据用户最后一次点击行为的位置，过滤掉最后一次点击之后的展示，可以认为用户没有看到，也可以保留最后一次点击之后的少数几个曝光，如图1-5所示。

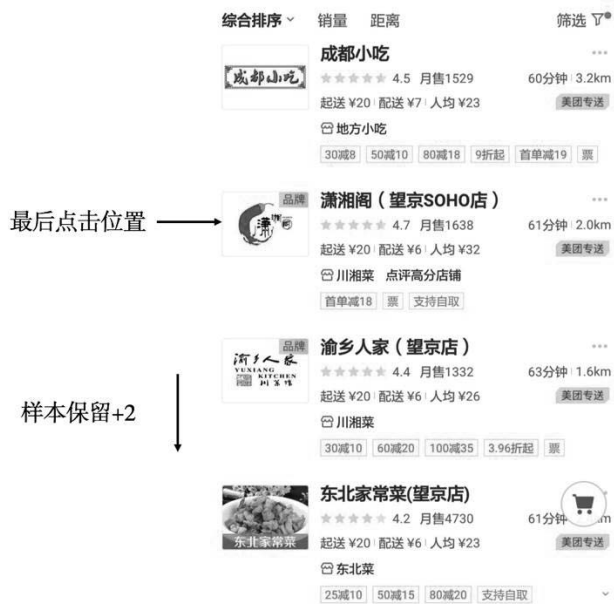


图1-5　样本过滤逻辑示例

1.2.2　采样

采样是一个完善的统计技术，从整体选择一部分来进行推论。采样能够克服高维特征以及大量数据导致的问题，有助于降低成本，缩短时间甚至提升效果，在不平衡分类问题中还能帮助平衡样本比例。进行采样时最关心采样方法和采样比例。

通常，考虑样本对总体的估计值不超出边际误差的情况下是能够计算出样本量的。如下面的概率不等式：

$$P(|e - e_0| \geqslant \epsilon) \leqslant \delta$$

对于给定的ϵ和δ，能够计算出采样大小n。其中，e代表样本的估计，通常是样本大小n的函数；e_0代表真实的样本；ϵ是置信度；$1-\delta$是置信区间。然而，e_0一般都是未知的，通常会先从样本中采样一个小的有m条样本的数据集，对e_0进行估计，之后再计算对应的n值。如果$n \geqslant m$，然后再从余下的样本集中选取额外的$n-m$条样本；如果$n \leqslant m$，那么就将m条样本作为采样结果。

一个好的样本子集应该具有无偏性（Unbiasedness）和很小的样本方差（Sampling Variance）。其中无偏性指的是对样本的期望等于全体样本期望，即$E(e)=e_0$。样本方差是衡量样本估计值和真实值的偏差，即$\mathrm{Var}(e)=E[e-E(e)]^2$，小方差能保证估计值不会产生太大偏差。

现有的采样方法有很多，下文简单介绍5种采样方法。

- **无放回简单随机抽样**（Simple Random Sample Without Replacement）。它从含N条样本的数据集T中采样s（$s \leqslant N$）条样本，每条样本被采到的概率相等且都为$\frac{1}{N}$。
- **有放回简单抽样**（Simple Random Sample With Replacement）。它和无放回简单随机抽样类似，不同的是每次从数据集T中抽取一条样本后，还将这条样本放回到数据集T中，因此每条样本可能多次被选中。
- **平衡采样**（Balanced Sample）。它根据目标变量进行采样，依据预定义的比例对样本进行重新组合，在不平衡分类问题中有十分成功的应用。不平衡分类问题指分类任务中不同类别的数据量差异巨大，通常会对小数据量的类别进行上采样，或者对大数据量的类别进行下采样。比如一份二分类样本有100条正样本、10 000条负样本，采样目标是使正负样本比例为1∶10，那么上采样就是对正样本复制10遍，负采样就是对负样本随机删除部分样本留下1000条；ADASYN和SMOTE算法是上采样里两个比较常用的方法。
- **整群采样**（Cluster Sample）。它先将数据集T中的数据分组成G个互斥的簇，然后再从G个簇中简单随机采样s（$s \leqslant G$）个簇作为样本集，这个方法是分两个阶段完成采样的。
- **分层采样**（Stratified Sample）。数据集T划分成不同的层，然后在每层内部进行简单随机抽样，最后汇总成样本集合S。该方法也常用于不平衡分类问题中，和平衡采样非常相关。该方法分别对每个类别进行采样，能使每个类别在样本集S中的分布和数据集T中的分布更为一致。比如对平衡采样中的二分类数据进行分层采样，目的是采样90%数据，分层采样以采样率0.9分别对正负样本采样，能保证正负比例还是1∶100；如果对全部10 100条样本采样90%，可能出现正样本10条、负样本9080条的情况。

1.2.3 原型选择和训练集选择

原型选择是基于实例的方法，在样本选择过程中不需要训练模型，而是选取相似度或距离度量指标来找到分类精度和数据量最佳的训练集，多数采用KNN算法。训练集选择则是构建预测模型来进行样本选择的方法的统称，比如决策树、ANN和SVM等算法。原型选择和训练集选择两大类别的样本选择方法有很多，然而没有一种方法能够通用。

原型选择有很多分类标准，根据从数据集T中选择样本集S的方向可以分为以下5类。

- **增量法**。开始时令$S=\emptyset$，然后逐条遍历数据集T中的每条样本，如果满足条件则加入S中。
- **递减法**。和增量法相反，开始时令$S=T$，然后逐条查找待过滤的样本从S中删除。
- **批量法**。和递减法类似，批量法先判断一批数据的每条数据是否应该删除，然后再将这批数据中全部满足删除条件的样本一起删除。
- **混合法**。预先选定一部分样本$S\neq\emptyset$，然后迭代地增加或删除满足对应条件的样本。
- **固定法**。是混合法的一个子方法，但最终选择的样本数是固定的。

原型选择也可以类似特征选择，根据选择样本的策略进行分类。

- **包装器**。根据模型的目标函数，一般是模型预测结果来进行样本选择。
- **过滤器**。样本的选择标准不基于模型。

还可以根据选择的样本，原型选择相关算法可分为如下三类。

- Condensation。保留决策边界处样本。
- Edition。删除边界的异常点，使得边界更平滑。
- Hybrid。尝试找到最小的样本集S，能够保持甚至提升测试集上的泛化精度。

大量原型选择算法都可以根据上述分类标准进行划分。在此就不针对具体的算法展开介绍了。

1.3　交叉验证

在离线环节，需要对模型进行评估，根据评估指标选择最佳模型。用于模型训练的数据集叫训练集，用于评估模型的数据叫测试集；训练集上的误差称为训练误差或经验误差，测试集上的误差称为测试误差。测试样本是用于测试模型对新样本的学习能力，所以在假设测试数据和真实数据是独立同分布的前提下，测试误差可以作为泛化误差的近似。模型对新样本的学习能力十分重要，我们希望模型对已有样本进行学习，尽可能将样本中潜在的普遍规律学到。如果模型在训练集上效果极好，但是在测试集上效果很差，这说明模型将训练集中的一些规律当作普遍规律，于是就过拟合了。测试集可以帮助防止过拟合，还能够帮助指导模型调参。通常而言，训练集和测试集互斥，训练集越多，得到的模型效果越好；测试集越多，得到的结论越可信。我们将划分训练集和测试集的方法统称为交叉验证。交叉验证有很多方法，不同方法适用不同场景。下面介绍几种常用的交叉验证方法。

1.3.1　留出法

留出法（Hold-Out）是将数据集$\mathcal{D}=\{(x_1,y_1),\cdots,(x_n,y_n)\}$随机划分成两份互斥的数据集，一份作为训练集$\mathcal{D}_{tr}$，一份作为测试集$\mathcal{D}_{te}$，在$\mathcal{D}_{tr}$上训练模型，然后用$\mathcal{D}_{te}$评估模型效果。本质上，留出法并非一种交叉验证方法，因为数据并没有交叉。

留出法只需将数据划分成两部分，简单好实现，如图1-6所示。但这种方法的缺点也比较明显，它不能充分利用数据训练模型，并且训练集和测试集的划分严重影响最终结果。$\mathcal{D}_{te}$的数据量越大，$\mathcal{D}_{tr}$就越小，得到的模型很可能和全量数据$\mathcal{D}$得到的模型产生大的偏差；$\mathcal{D}_{tr}$的数据量越大，$\mathcal{D}_{te}$就越小，得到的结论可信度变低。通常的做法是，2/3数据作为训练集，1/3数据作为测试集。

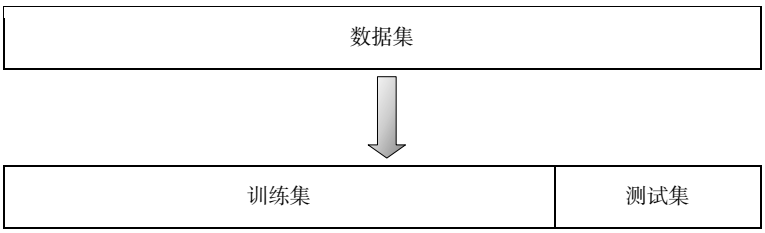


图1-6 留出法

除了划分测试集数据量对结论有影响外，划分哪些样本作为测试集也会影响实验结论，因为这将导致数据分布发生变化。比如二分类问题有1500条正样本和1500条负样本，将1/3数据作为测试集，应该使得测试集正负样本均在50条左右；如果测试集由50条正样本和950条负样本组成，实验结论将因为样本分布差异悬殊而有很大偏差。因此，考虑到单次留出法得到的结论往往不靠谱，我们会进行多次留出法实验，每次随机划分，最终将多次得到的实验结论进行平均。

实际工作中有一种普遍的应用场景广泛使用留出法：数据有明显的时间序列因素，即线上数据的时间都在离线数据集之后，这种情况下应该根据时间对离线数据集划分训练集和测试集，使测试集时间分布在训练集时间之后。

比如，在2017年6月初需要训练模型，可以采用2017年1月到2017年4月的数据作为训练集，2017年5月的数据作为测试集。

1.3.2 K 折交叉验证

K折交叉验证（K-fold Cross Validation）将数据集$\mathcal{D}$划分成K份互斥数据集$\mathcal{D}_k$，满足$\mathcal{D} = D_1 \cup \cdots \cup D_K$，一般是平均分配使每份数据量接近并且数据分布尽可能一致。每次用一份数据测试，其余$K-1$份数据训练，需要迭代K轮得到K个模型；最后再将K份测试结果汇总到一起评估一个离线指标。

$$\text{cv\_score} = \frac{1}{K}\sum_{k=1}^{K} L(P_k, Y_k)$$

K折交叉验证的稳定性与K取值有很大关系。K值太小实验稳定性依然偏低，K值太大又可能导致实验成本高，K最常用的取值是5和10，如图1-7所示。K折交叉验证能够更好地避免过拟合和欠拟合，得到的结论也更有说服力。

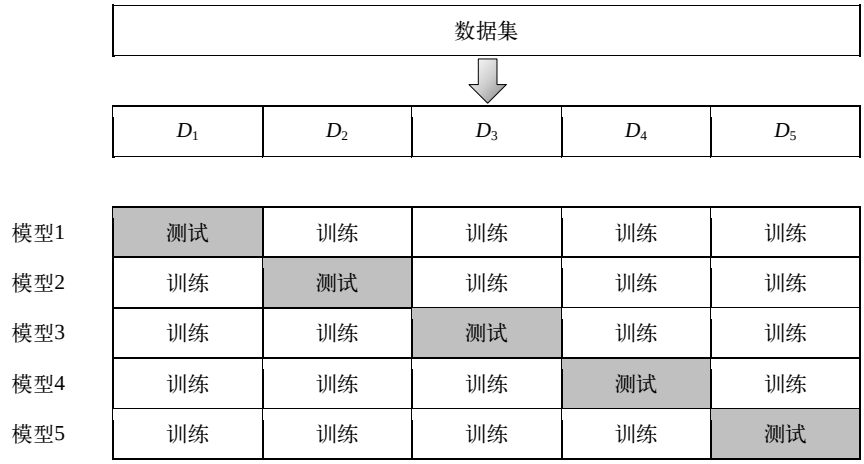


图1-7　5折交叉验证

相比留出法，K折交叉验证更为复杂，需要训练K个模型，但是数据利用率更高。$K = 2$时，K折交叉验证和留出法仍有差异，留出法相当于用$\mathcal{D}_1$训练$\mathcal{D}_2$测试得到测试结果cv_score_1，而2折交叉验证还会用$\mathcal{D}_2$训练$\mathcal{D}_1$测试得到测试结果cv_score_2再取两次结果的平均值。另外，K折交叉验证也可能因为单次K份数据划分导致数据分布发生变化而引入偏差，因此也经常会进行多次K折交叉验证后求平均。比如进行10次5折交叉验证，这10次划分5折交叉验证得到的数据会不同。

假定数据集$\mathcal{D}$中有N条数据，当K折交叉验证的$K = N$时，就是留一法（Leave-One-Out，LOO），即每一条样本当测试集，其余数据作训练。LOO策略的优缺点都很明显。训练N个模型，每个模型都基本用到了全部的数据，得到的模型和全部数据$\mathcal{D}$得到的模型更相似，并且不再受随机样本划分方式的影响，因为划分方式只有一种了。但是当样本量N很大时，计算成本非常高，计算甚至不可行，而且每个模型只有一条测试数据，不能有效帮助每个模型调参达到最优。但是在数据稀疏时，LOO很适用。

基于K折交叉验证还变种出一个在类不均衡情况下常用的方法，叫作分层K折交叉验证（Stratified K-Fold）。该方法对每个类别进行K折划分，使每份数据中各类别的数据分布与完整数据集分布更一致。比如二分类数据进行5折交叉验证，类别1有30条数据，类别2有300条数据，在划分成5折时，如果随机划分成5份，这可能导致5份数据中的类别1数据量差别很大，导致每份数据训练出来的模型对类别1的分类效果差异很大，影响整体效果。如果通过分层5折交叉验证，即分别对2个类别划分，使每份数据有6条类别1样本，60条类别2样本，每份数据分布都和整体数据分布一致，得到的模型也就更可信。

1.3.3　自助法

自助法（Bootstrapping）以自主采样（Bootstrap Sampling）[Efron and Tibshirani, 1993]为基础，使用有放回的重复采样的方式进行训练集、测试集构建。比如为了构建n条样本的训练集，每次

从数据集$\mathcal{D}$采样一条放入训练集，然后又放回重新采样，重复n次得到n条样本的训练集，然后将没出现过的样本作为测试集。从操作过程可知，一些样本在训练集中重复出现，而另一些样本在训练集中从未出现。我们可以计算样本从未在训练集中出现的概率。在每次采样的时候，每条样本没被采到的概率$P_0 = 1 - \frac{1}{n}$，经过n次采样还没被采到的概率为$(1 - \frac{1}{n})^n$，取极限可得

$$\lim_{n\to\infty}\left(1-\frac{1}{n}\right)^n = \mathrm{e}^{-1} \approx 0.368$$

这意味着当数据量很大的时候，约有36.8%的样本不会出现在训练集中。显然训练集有n条样本，测试集有约0.368n条样本。留出法和K折交叉验证法在训练模型时用的数据都只是整个数据集$\mathcal{D}$的一个子集，得到的模型会因为训练集大小不一致产生一定的偏差。而自助法能够更好地解决这个问题。但自助法改变了初始数据集的分布，会引入估计偏差，所以在数据量足够时，一般采用留出法和交叉验证法。而在数据量较小，并且难以有效区分训练集和测试集时，自助法很有用。

参考文献

[1] Davis J, Goadrich M. The relationship between precision-recall and ROC curves. Proceedings of the 23rd international conference on machine learning. ACM, 2006.

[2] Fawcett T. An introduction to ROC analysis. Pattern recognition letters. 2006.

[3] Hanley J A, McNeil B J. The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology, 1982, 143: 29-36.

[4] Hand D J, Till R J. A simple generalization of the area under the ROC curve to multiple class classification problems. Mach. Learning, 2001, 45 (2): 171-186.

[5] Olvera-López J A, Carrasco-Ochoa J A, Martínez-Trinidad J F, et al. A review of instance selection methods[J]. Artificial intelligence review, 2010, 34(2): 133-143.

[6] García S, Luengo J, Herrera F. Data preprocessing in data mining[M]. Springer, 2015.

[7] Instance selection and construction for data mining[M]. Springer Science & Business Media, 2013.

[8] 周志华. 机器学习[M]. 北京：清华大学出版社, 2016.

第 2 章 特征工程

在机器学习应用中，特征工程扮演着重要的角色，可以说特征工程是机器学习应用的基础。在机器学习界流传着这样一句话：“数据和特征决定了机器学习算法的上限，而模型和算法只是不断逼近这个上限而已。”在机器学习应用中，特征工程介于“数据”和“模型”之间，特征工程是使用数据的专业领域知识创建能够使机器学习算法工作的特征的过程。美国计算机科学家Peter Norvig有两句经典名言:“基于大量数据的简单模型胜于基于少量数据的复杂模型。”以及“更多的数据胜于聪明的算法，而好的数据胜于多的数据。”因此，特征工程的前提便是收集足够多的数据，其次则是从大量数据中提取关键信息并表示为模型所需要的形式。合适的特征可以让模型预测更加容易，机器学习应用更有可能成功。

纵观Kaggle、KDD等国内外大大小小的比赛以及工业界的应用，它们其实并没有用到很复杂的模型和算法，大多数成功都是在特征工程这个环节做了出色的工作。吴恩达曾说过：“特征工程不仅操作困难、耗时，而且需要专业领域知识。应用机器学习基本上就是特征工程。”相信大多数人都会同意。在机器学习应用中，我们大多数时间都在进行特征工程和数据清洗，而算法和模型的优化仅仅占了一小部分。遗憾的是，目前大多数书籍中并没有提到特征工程，对于特征工程的介绍更多则是特征选择的方法。这是因为，好的特征工程不仅需要我们对模型和算法有深入的理解，更需要较强的专业领域知识。特征工程不仅跟模型相关，而且跟实际问题是强相关的。针对不同问题，特征工程所用的方法可能相差很大，很难总结出一套比较通用的方法。尽管如此，但仍然有很多特征工程的技巧在不同问题中都适用。在本章，我们将介绍特征工程中通用的方法和技巧，以及常用特征选择方法。

2.1 特征提取

从数学的角度讲，特征工程就是将原始数据空间变换到新的特征空间，或者说是换一种数据的表达方式，在新的特征空间中，模型能够更好地学习数据中的规律。因此，特征抽取就是对原始数据进行变换的过程。大多数模型和算法都要求输入是维度相同的实向量，因此特征工程首先需要将原始数据转化为实向量。原始数据有很多类型，比如数值类型、离散类型，还有文本、图像以及视频等。将原始数据转化为实向量后，对应的特征空间并不一定是最佳的特征空间。为了

让模型更好地学习到数据中隐藏的规律，我们可能还需要对特征做进一步的变换。将原始数据空间变换为模型输入向量空间的过程便是特征工程所要做的事情。事实上，如果特征工程足够复杂，即使是最简单的模型，也能表现出非常好的效果。然而，复杂的模型在一定程度上减少了特征工程需要做的工作。因此，特征工程和模型二者此消彼长。例如，对于线性模型，我们需要将类别变量进行独热编码等处理，但对于复杂一些的模型如树模型，可以直接处理类别变量。对于更高级的深度神经网络，模型可以自动进行特征表示。

那么特征工程具体怎么操作呢？首先，我们需要知道手上有哪些可以使用的数据，可以获取哪些数据以及这些数据的获取成本。我们拥有的数据越多、越有价值，机器学习应用成功的可能性往往就越高。但有些数据的获取成本非常昂贵，而且从数据量和覆盖面两方面来讲，数据都很有限，所以我们面临的挑战更多在于从现有数据中挖掘出对机器学习模型有用的特征。特征的挖掘一般跟专业领域知识强相关，特征工程可以说是业务逻辑的一种数据层面的表示。所以特征工程的第一步是理解业务数据和业务逻辑。特征提取也可以看作用特征描述业务逻辑的过程，特征提取的目标是对业务进行精确、全面的描述。同时，生产的特征最终用于模型预测，因此我们需要理解模型和算法，清楚模型需要什么样的输入才能有较精确的预测结果。此外，为了进一步提升模型效果，我们还需要对特征进行特殊的数学变换。

根据机器学习算法所要学习的目标和业务逻辑，我们需要考虑数据中有哪些可能相关的要素。例如在美团酒店搜索排序中，酒店的销量、价格、用户的消费水平等是强相关的因素，用户的年龄、位置可能是弱相关的因素，用户的ID是完全无关的因素。在确定了哪些因素可能与预测目标相关后，我们需要将此信息表示为数值类型，即为特征抽取的过程。例如在美团酒店搜索排序中，描述酒店的特征有酒店的位置、上线时间、星级等，描述用户的特征有用户的注册时间、VIP等级、地理位置等，这些特征都是静态的，可以从数据库中抽取。除此之外，用户在App上的浏览、交易等行为记录中包含了大量的信息，特征抽取则主要是从这些信息抽取出相关因素，用数值变量进行表示。常用的统计特征有计数特征，如浏览次数、下单次数等；比率特征，如点击率、转化率等；统计量特征，如价格均值、标准差、分位数、偏度、峰度等。

2.1.1 探索性数据分析

当你手上有一份数据，但对这份数据完全陌生且没有足够的专业背景知识时，你可能感觉无从下手。如果不做任何数据分析和预处理，直接将数据喂给模型，得到的效果一般都不会太好。这是因为喂给模型的数据不具有好的特征。如何发现好的特征呢？当然专业领域知识很重要，但是在没有足够专业领域知识的情况下，通过探索性数据分析往往能够发现效果不错的特征。

在统计学里，探索性数据分析（Exploratory Data Analysis，EDA）是采用各种技术（大部分为可视化技术）在尽量少的先验假设条件下，探索数据内部结构和规律的一种数据分析方法或理念。特别是当我们对数据中的信息没有足够的先验知识，不知道该用什么方法进行分析时，先对数据进行探索性分析，发现数据的模式和特点，就能够灵活地选择和调整合适的模型。EDA由美

国著名统计学家约翰·图基（John W. Tukey）在20世纪60年代提出。EDA的目的是尽可能地洞察数据集、发现数据的内部结构、提取重要的特征、检测异常值、检验基本假设、建立初步的模型。EDA的特点是从数据本身出发，不拘泥于传统的统计方法，强调数据可视化。EDA工具有很多，但EDA更多是方法论而不是特定的技术。EDA技术通常可分为两类。一类是可视化技术，如箱形图、直方图、多变量图、链图、帕累托图、散点图、茎叶图、平行坐标、让步比、多维尺度分析、目标投影追踪、主成分分析、多线性主成分分析、降维、非线性降维等；另一类是定量技术，如样本均值、方差、分位数、峰度、偏度等。

2.1.2　数值特征

数值类型的数据具有实际测量意义，例如人的身高、体重、血压等，或者是计数，例如一个网站被浏览多少次、一种产品被购买多少次等（统计学家也称数值类型的数据为定量数据）。数值类型的数据可以分为离散型和连续型。离散型数据表示的量是可数的，其可以是有限个值，也可以是无限个值。例如，100次硬币投掷中正面朝上的个数取值为0到100，但是获得100次正面朝上所需要的投掷次数取值为0到正无穷。连续型数据表示测量得到的量，其取值是不可数的，可以用实数轴上的区间表示，为了便于记录，通常指只保留部分有效数字，例如人的体重的取值可以是70.41千克，或者是70.414 863千克。

机器学习模型可以直接将数值类型的数据格式作为输入，但这并不意味着没有必要进行特征工程。好的特征不仅能表示出数据中隐藏的关键信息，而且还与模型的假设一致。通常情况下，对数值类型的数据进行适当的数值变换能带来不错的效果提升。对于数值特征，我们主要考虑的因素是它的大小和分布。对于那些目标变量为输入特征的光滑函数的模型，如线性回归、逻辑回归等，其对输入特征的大小很敏感。因此，当使用光滑函数建模时，有必要对输入进行归一化。而对于那些基于树的模型，例如随机森林、梯度提升树等，其对输入特征的大小不敏感，输入不需要进行归一化。但是，对于树模型，如果特征取值无限大也会有问题。如果模型对输入特征和目标变量有一些隐式或者显式的假设，则数据的分布对模型很重要。例如，线性回归训练通常使用平方损失函数，其等价于假设预测误差服从高斯分布。因此，如果输出变量分布在不同尺度时，这个假设不再成立。在这种情况下，我们有必要对目标变量进行变换使其满足假设。严格地说，这种方法应该称为“目标”工程，而不是特征工程。除了对特征进行变换以满足模型的假设，我们也可以对特征进行交叉组合。特征交叉提升了模型的表达能力，让线性模型具有非线性模型的性质，而树模型天然有这种性质。下面我们详细介绍8种常见的数值特征的处理方法。

- **截断**。对于连续型数值特征，有时候太多的精度可能只是噪声。因此，可以在保留重要信息的前提下对特征进行截断，截断后的特征也可以看作是类别特征。另外，至于长尾的数据，可以先进行对数转换，然后进行截断。
- **二值化**。数值特征的一种常用类型是计数特征，如网站每天的访问量、餐厅的评论数、用户对一首歌的播放次数等。在大数据时代，计数可以非常快地累加。处理计数特征时，首先要考虑的是，保留为原始计数还是转换为二值变量来标识是否存在或者进行分桶操作。

- **分桶**。在购物网站上，每件商品都会显示用户的评论次数。假设我们的任务是利用逻辑回归模型来预测用户对某件商品的购买概率。商品的评论次数可能是一个有用的特征，因为评论次数跟商品的热度有很强的相关性，那么我们应该直接使用原始的评论次数作为特征还是需要进行预处理？如果商品的评论次数跨越不同的数量级，则它不是一个好的特征，例如对于逻辑回归模型，一个特征对应一个系数，从而模型往往只对比较大的特征值敏感。对于这种情况，通常的解决方法是进行分桶。分桶是将数值变量分到一个桶里并分配一个桶编号，常见的分桶方法有固定宽度的分桶。对于固定宽度的分桶，每个桶的值域是固定，如果每个桶的大小一样，它也称为均匀分桶，例如将人的年龄分为0~9岁、10~19岁等。除此之外，桶的宽度也可以自定义。如果数值跨越不同数量级，可以根据10（或者其他任何适当的常数）的幂来分桶，如0~9、10~99、100~999、1000~9999等，这种方法和对数变换非常相关。另一种分桶方式是分位数分桶，虽然固定宽度的分桶易于实现，但如果数值变量的取值存在很大间隔时，有些桶里没有数据，可以基于数据的分布进行分桶，也即分位数分桶，其对应的桶里面数据一样多。除此之外，也可以使用模型找到最佳分桶，例如利用聚类的方式将特征分为多个类别。分桶操作也可以看作是对数值变量的离散化，因此分桶后也可以将分桶操作当成类别变量进行处理。
- **缩放**。缩放即将数值变量缩放到一个确定的范围。常见的缩放有：标准化缩放（也称为Z缩放），即将数值变量的均值变为0，方差变为1，对于那些目标变量为输入特征的光滑函数的模型，如线性回归、逻辑回归等，其对输入特征的大小很敏感，对特征进行标准化比较有效；最大最小值缩放及最大绝对值缩放；基于某种范数的归一化，如使用L_1范数、L_2范数将数值向量的范数变为1；平方根缩放或对数缩放，对数缩放对于处理长尾分且取值为正数的数值变量非常有效，它将大端长尾压缩为短尾，并将小端进行延伸，平方根或者对数变换是幂变换的特例，在统计学中都称为方差稳定的变换，其中Box-Cox变换是简化的幂变换，Box-Cox 转换仅对取值为正数的数值变量起作用；对于有异常点的数据，可以使用更加健壮的缩放，与一般的标准化基于标准差进行缩放不同的是，健壮的缩放使用中位数而不是均值，基于分位数而不是方差。
- **缺失值处理**。实际问题中经常会遇到特征缺失的情形，但是大多数模型并不能处理特征缺失的情况，缺失特征的处理方式会影响模型效果。对于特征缺失，我们有两类处理方法。第一种是补一个值，例如最简单的方法是补一个均值；对于包含异常值的变量，更加健壮一些的方法则是补一个中位数；除此之外还可以使用模型预测缺失值。另外一种则是直接忽略，即将缺失作为一种信息进行编码喂给模型让其进行学习。现在有一些模型可以直接处理缺失值，例如XGBoost模型可以处理缺失特征。
- **特征交叉**。特征交叉可以表示数值特征之间的相互作用，例如可以对两个数值变量进行加、减、乘、除等操作，可以通过特征选择方法（如统计检验或者模型的特征重要性）来选择有用的交叉组合。有些特征交叉组合，虽然没有直观的解释，但有可能对于模型效果有很大的提升。除了构造交叉特征外，有些模型可以自动进行特征交叉组合，例如FM和FFM模型等。特征交叉可以在线性模型中引入非线性性质，提升模型的表达能力。

- **非线性编码**。线性模型往往很难学习到数据中的非线性关系，除了采用特征交叉的方式之外，也可以通过非线性编码来提升线性模型的效果。例如使用多项式核，高斯核等，但选择合适的核函数并不容易。另外一种方法是将随机森林模型的叶节点进行编码喂给线性模型，这样线性模型的特征包含了复杂的非线性信息。还有基因算法以及局部线性嵌入、谱嵌入、t-SNE等。
- **行统计量**。除了对原始数值变量进行处理之外，直接对行向量进行统计也可以作为一类特征，如统计行向量中空值的个数、0的个数、正值或负值的个数，以及均值、方差、最小值、最大值、偏度、峰度等。

以上内容是常见的数值特征的预处理方法。具体采取哪一种处理方式不仅依赖于业务和数据本身，还依赖于所选取的模型，因此首先要理解数据和业务逻辑以及模型的特点，才能更好地进行特征工程。

2.1.3　类别特征

类别数据表示的量可以是人的性别、婚姻状况、家乡或者他们喜欢的电影类型等。类别数据的取值可以是数值类型（例如“1”代表男性，“0”代表女性），但是数值没有任何数学意义，它们不能进行数学运算。类别数据的另一个名称是定性数据。类别特征不仅可以由原始数据中直接提取，也可以通过将数值特征离散化得到。下面我们介绍几种常见的类别变量的处理方法。

- **自然数编码**。类别特征，一般首先要转换为数值类型才能喂给模型。最简单的编码方式是自然数编码，即给每一个类别分配一个编号，对类别编号进行洗牌，训练多个模型进行融合可以进一步提升模型效果。
- **独热编码**。通常，直接将类别特征的自然数编码特征喂给模型，效果可能比较差，尤其是线性模型。这是因为，对于类别特征的自然数编码，取值大小没有物理含义，直接喂给线性模型没有任何意义。我们常用的一种做法是对类别特征进行独热编码，这样每个特征取值对应一维特征，独热编码得到稀疏的特征矩阵。
- **分层编码**。对于邮政编码或者身份证号等类别特征，可以取不同位数进行分层，然后按层次进行自然数编码，这类编码一般需要专业领域知识。
- **散列编码**。对于有些取值特别多的类别特征，使用独热编码得到的特征矩阵非常稀疏，因此在进行独热编码之前可以先对类别进行散列编码，这样可以避免特征矩阵过于稀疏。实际应用中我们可以重复多次选取不同的散列函数，利用融合的方式来提升模型效果。散列方法可能会导致特征取值冲突，这种冲突通常会削弱模型的效果。自然数编码和分层编码可以看作散列编码的特例。
- **计数编码**。计数编码是将类别特征用其对应的计数来代替，这对线性和非线性模型都有效。这种方法对异常值比较敏感，特征取值也可能冲突。
- **计数排名编码**。它利用计数的排名对类别特征进行编码，其对线性和非线性模型都有效，而且对异常点不敏感，类别特征取值不会冲突。

- **目标编码**。它基于目标变量对类别特征进行编码。对于基数（类别变量所有可能不同取值的个数）很大的离散特征，例如IP地址、网站域名、城市名、家庭地址、街道、产品编号等，上述预处理方法效果往往不太好。因为对于自然数编码方法，简单模型容易欠拟合，而复杂模型容易过拟合；对于独热编码方法，得到的特征矩阵太稀疏。对于高基数类别变量，一种有效方式则是基于目标变量对类别特征进行编码，即有监督的编码方法，其适用于分类和回归问题。例如对于分类问题，采用交叉验证的方式，即将样本划分为5份，针对其中每一份数据，计算离散特征每个取值在另外4 份数据中每个类别的比例。为了避免过拟合，也可以采用嵌套的交叉验证划分方法。回归问题同样采用交叉验证的方式计算目标变量均值对类别变量编码。在实际问题中，我们往往利用历史数据来预测未来结果。因此我们一般基于时间信息来划分训练集和验证集，利用相同时间窗口大小的历史数据来对类别特征进行编码。例如，在广告点击率预测问题中，我们计算广告主ID在过去固定一段时间内的点击率，对广告主ID进行目标编码。目标编码方法对于基数较低的离散变量通常很有效，但对于基数特别高的离散变量，可能会有过拟合的风险。因为很多类别特征的取值样本个数太少，不具有统计意义。对于这种情况，我们通常采用贝叶斯方法，即对统计特征进行贝叶斯平滑，如拉普拉斯平滑或者先验概率和后验概率加权平均的方式。
- **类别特征之间交叉组合**。除了前面提到的数值特征之间的交叉组合外，类别特征之间的交叉组合也是很重要的特征。两个类别特征进行笛卡儿积操作可以产生新的类别特征，这种操作适用于两个类别特征的基数较小的情况。两个类别特征的笛卡儿积取值个数是两个类别特征取值个数的乘积，如果两个类别特征的基数很大时，交叉后的特征基数太大，效果可能并不好。除了两个类别特征的交叉，多个类别特征也可以交叉组合，根据实际需要可以进行二阶及二阶以上的交叉组合，最后通过特征选择方法选取重要的组合方式。除了上面提到的交叉组合外，另一种特征组合方式是基于统计的组合。例如针对城市ID和商品ID两个类别特征，我们可以计算某个城市有多少不同的商品ID以及当前ID出现次数的分布，从而得到新的数值特征，或者计算某个城市出现次数最多的商品ID，从而得到一个新的类别特征。对于多个类别特征也可以进行同样的操作。例如针对年龄、性别、产品ID三个类别特征，可以计算某个年龄段不同性别的人购买过多少产品或者对当前产品ID购买次数的分布等。在实际应用中，类别特征之间的组合方式千变万化，这类特征一般从业务逻辑的角度出发进行构造。相比类别特征之间的笛卡儿积操作，基于分组统计的特征组合方式计算更加复杂，而且一般强依赖专业领域知识，因此需要对业务逻辑有较好的理解。
- **类别特征和数值特征之间交叉组合**。除了数值特征之间的组合以及类别特征之间的组合之外，类别特征和数值特征之间也可以进行组合。这类特征通常是在类别特征某个类别中计算数值特征的一些统计量。例如针对用户ID，统计过去一段时间内在网站上的浏览次数、购买次数，以及购买价格的统计量，如均值、中位数、标准差、最大值和最小值等；针对产品，统计用户对产品的评分、评价次数、购买次数、浏览次数等。再比如，

统计产品在某个区域的销量、产品的价格，或者当前产品的价格跟产品所在区域内的平均价格的差价等。可以看出，这类特征也强依赖专业领域知识。上面的这种组合方式也可以看作是利用数值特征对类别特征进行编码，与前面提到的基于目标变量对类别变量进行编码的方法不同的是，这里不需要划分训练集进行计算。

数值特征和类别特征是机器学习应用中最常见的两类特征。上面我们提到了关于这两类特征的一些常用的特征预处理技巧，基于这些技巧可以构造大量特征。但我们无法构造所有可能的特征表达形式，一方面要考虑模型的使用成本，另一方面也要考虑特征的构造成本。当然，我们可以通过特征选择选取重要的特征，但特征选择成本也很高。因此，在实际应用中我们选择性地构造特征。对于不同类别的特征采取哪一种或者哪几种方法，则依赖于我们对业务和数据本身的理解以及对模型的理解。通过对数据内部结构和规律的探索性分析，可以找到跟目标变量相关的信息，进而根据模型需要的输入形式利用预处理技术对这些信息进行编码，即构造特征。

2.1.4　时间特征

在实际应用中，时间往往是一个非常重要的因素，例如用户在购物网站上的浏览、购买、收藏的时间，产品在购物网站上的上线时间，顾客在银行的存款和借款时间、还款时间等。时间变量通常以日期（如2017/05/07 12:36:49）、时间戳（如1494391009）等形式表示。时间变量可以直接作为类别变量处理，类别特征的处理方式对于时间特征同样适用。但时间变量还包含其他更加丰富的信息。时间变量常用的表达方式有年、月、日、时、分、秒、星期几，以及一年过了多少天、一天过了多少分钟、季度、是否闰年、是否季度初、是否季度末、是否月初、是否月末、是否周末，还有是否营业时间、是否节假日等。除了对单个时间变量的预处理之外，根据具体业务对两个时间变量之间进行组合也能提取重要的特征。例如可以计算产品上线到现在经过了多长时间，顾客上次借款距离现在的时间间隔，两个时间间隔之间是否包含节假日或其他特殊日期等。

除了上面提到的基于时间本身的特征之外，时间变量更重要的是时间序列相关的特征。时间序列不仅包含一维时间变量，还有一维其他变量，如股票价格、天气温度、降雨量、订单量等。时间序列分析的主要目的是基于历史数据来预测未来信息。对于时间序列，我们关心的是长期的变动趋势、周期性的变动（如季节性变动）以及不规则的变动。对于时间序列信息，当前时间点之前的信息通常很重要，例如滞后特征（也称为lag特征）使用非常广泛。滞后特征是时间序列预测问题转化为监督学习问题的一种经典方法。若我们的问题是利用历史数据预测未来，则对于t时刻，可以将$t-1$、$t-2$和$t-3$时刻的值作为特征使用。若我们的问题可以考虑未来信息，则$t+1$、$t+2$和$t+3$时刻的值也可以作为特征使用。另一种有效方式是滑动窗口统计特征，例如计算前n个值的均值（回归问题），或者前n个值中每个类别的分布（分类问题）。时间窗口的选取可以有多种方式，上面提到的滞后特征是滑动窗口统计的一种特例，对应时间窗口宽度是1。另一种常用的窗口设置包含所有历史数据，称为扩展窗口统计。

2.1.5 空间特征

基于空间位置变量也是一类非常重要的信息，例如经纬度。对于经纬度，除了将其作为数值变量使用之外，还有其他更加有效的使用方式。例如可以对经纬度做散列，从而对空间区域进行分块，得到一个类别特征，也可以通过坐标拾取系统获得当前位置的行政区ID、街道ID、城市ID等类别特征，进而利用类别特征的处理方式进行特征预处理。还可以计算两个位置之间的距离，如用户到超市或者电影院、餐厅的距离。距离的计算方式也有很多种，例如可以计算欧氏距离、球面距离以及曼哈顿距离，也可以是真实的街道距离。

上面提到的时间变量和空间变量只是两种比较典型的变量，通过对这两种变量进行预处理进而转换为多个数值变量和类别变量。实际应用中还有很多类似的变量，从单个变量出发就可以构造很多特征，这类特征的构造主要依赖对数据本身的理解。

2.1.6 文本特征

自然语言要处理的对象是文本信息。对于文本特征，类别特征的处理方法同样适用，基于深度学习的自动特征工程效果变得越来越好。但是好的特征仍然具有竞争力。文本特征往往产生特别稀疏的特征矩阵。我们可以从以下几个方面对文本特征进行预处理：将字符转化为小写、分词、去除无用字符、提取词根、拼写纠错、词干提取、标点符号编码、文档特征、实体插入和提取、Word2Vec、文本相似性、去除停止词、去除稀有词、TF-IDF、LDA、LSA等。

- **语料构建**。构建一个由文档或短语组成的矩阵。矩阵的每一行为文档，可以理解为对产品的描述，每一列为单词。通常，文档的个数与样本个数一致。
- **文本清洗**。如果数据通过网页抓取，首先剔除文本中的HTML标记；停止词只用于语句的构建，但不包含任何真实的信息，因此需要剔除；为了避免文本中的大小写差异，整个文本通常需要转换为小写形式；统一编码；去除标点符号；去除数字；去除空格；还原为词根。但是在某些情况下，文本不一定需要进行清洗，这取决于具体的应用场景。例如考虑某编辑员对某物品的描述，如果我们关心的对象是物品，则需要去除噪声，保留关键信息，但如果我们关心的对象是编辑员，则噪声信息一定程度上反映了此编辑员的水平。
- **分词**。
  - **词性标注**。词语通常有三类重要的词性：名词、动词和形容词。名词特指人、动物、概念或事物，动词表达动作，形容词描述了名词的属性。词性标注的目标是为文本中的每个词标注一个合适的词性，词性标注可以帮助我们了解语言的内在结构。
  - **词形还原和词干提取**。词形还原即把任何形式的语言词汇还原为一般形式（能表达完整语义）。词干提取是抽取词的词干和词根形式（不一定能够表达完整语义）。两者都能够有效归并词形。
  - **文本统计特征**。文本统计特征是最简单的文本特征，它不需要考虑词序信息，包括计

算文本的长度、单词个数、数字个数、字母个数、大小写单词个数、大小写字母个数、标点符号个数、特殊字符个数等，数字占比、字母占比、特殊字符占比等，以及名词个数、动词个数等。

- N-Gram模型。在自然语言处理中，N-Gram模型将文本转换为连续序列，序列的每一项包含n个元素（可以是单词或者字母等元素），例如“the dog smelled like a skunk”，得到3-Gram（the dog smelled，dog smelled like，smelled like a，like a skunk）。这种想法是为了将一个或者两个甚至多个单词同时出现的信息喂给模型。为了更好地保留词序信息，构建更有效的语言模型，我们希望在N-Gram模型中选用更大的n。但是当n很大时，数据会很稀疏。3-Gram是常用的选择。统计语言模型一般都是基于N-Gram的统计估计条件概率，基于神经网络的语言模型也是对N-Gram进行建模。

- Skip-Gram模型。

- 词集模型。机器学习模型不能直接处理文本，因此我们需要将文本（或者N-Gram序列）转化为实数或者实向量。在词集模型中，向量的每个分量的取值为0和1，代表单词是否在文档中出现。向量空间模型没有考虑词序信息。
- 词袋模型。在词集模型中，向量的取值不考虑单词出现的次数，这会损失很多信息。词袋模型中，向量的每个分量的取值为单词在文档中的词频，为了避免向量的维度太大，通常会过滤掉在文档集合中词频很小的单词。
- TF-IDF。TF（Term Frequency，词频）、IDF（Inverse Document Frequency，逆文档频率），用来评估单词对于文件集或语料库中的其中一份文件的重要程度。单词或短语的重要性随着它在文件中出现的次数成正比增加，同时随着它在语料库中出现的频率成反比下降。假设词汇表有N个词，文档d对应的向量表示为$\boldsymbol{v}_d = [w_{1,d}, w_{2,d}, \cdots, w_{N,d}]^{\mathrm{T}}$，其中$w_{t,d} = tf_{t,d} \cdot \log\frac{|D|}{|\{d' \in D | t \in d'\}|}$，$tf_{t,d}$为单词$t$在文档$d$中的词频（局部参数），$\log\frac{|D|}{|\{d' \in D | t \in d'\}|}$为逆文档频率（全局参数），$|D|$为总文档数，$|\{d' \in D | t \in d'\}|$为包含单词$t$的文档数。TF-IDF的主要思想是：如果某个词或短语在一篇文章中出现的频率TF高，并且在其他文章中很少出现，则认为此词或短语具有很好的类别区分能力，适合用来分类。TF-IDF模型是经典的向量空间模型（Vector Space Model，VSM），我们可以基于文档的向量表示计算文档之间的相似度，但不能很好地表示特别长的文档，而且这种向量表示也没有考虑词序信息。基于TF-IDF和词袋模型得到的表示文本的向量往往维度非常大，因此实际应用中一般需要降维处理。

- 余弦相似度。在信息检索中，我们往往需要计算检索词q和文档d之间的相关性。例如将检索词和文档都表示为向量，计算两个向量d和q之间的余弦相似度。

$$\cos\theta = \frac{d \cdot q}{\|d\|\|q\|} = \frac{\sum_{i=1}^{n} d_i q_i}{\sum_{i=1}^{n} d_i^2 \sum_{i=1}^{n} q_i^2}$$

- **Jaccard相似度**。另外一种常用的相似度是Jaccard相似度，它为两个文档中相交的单词个数除以两个文档出现单词的总和：

$$J(d_1, d_2) = \frac{|d_1 \cap d_2|}{|d_1 \cup d_2|}$$

  我们还可以定义Jaccard距离：

$$d_J(d_1, d_2) = 1 - J(d_1, d_2) = \frac{|d_1 \cup d_2| - |d_1 \cap d_2|}{|d_1 \cup d_2|}$$

- **Levenshtein（编辑距离）**。编辑距离是指两个字符串由一个转成另外一个所需要的最少编辑操作（如插入、删除、替换）次数，它也是衡量两个字符串相似度的指标。在自然语言处理中，单词一般作为基本的处理单元。
- **隐性语义分析**。隐性语义分析是把高维的向量空间模型表示的文档映射到低维的潜在语义空间中。隐性语义分析采用将文档或词矩阵进行奇异值分解（Singular Value Decomposition，SVD）的方法。由于奇异值分解的方法本身是对文档特征进行排序，我们可以通过限制奇异值的个数对数据进行降噪和降维。一般而言，文档和文档或者文档和查询之间的相似性在简化的潜在语义空间的表达更为可靠。
- **Word2Vec**。Word2Vec是最常用的一种单词嵌入，即将单词所在的空间（高维空间）映射到一个低维的向量空间中，这样每个单词对应一个向量，通过计算向量之间的余弦相似度就可以得到某个单词的同义词。传统的单词表示，如独热编码，仅仅是将词转化为数字表示，不包含任何语义信息。而单词嵌入包含了单词的语义信息，这类表示称为分布式表示。

2.2 特征选择

与特征提取是从原始数据中构造新的特征不同，特征选择是从这些特征集合中选出一个子集。特征选择对于机器学习应用来说非常重要。特征选择也称为属性选择或变量选择，是指为了构建模型而选择相关特征子集的过程。特征选择的目的有如下三个。

- **简化模型，使模型更易于研究人员和用户理解**。可解释性不仅让我们对模型效果的稳定性有更多的把握，而且也能为业务运营等工作提供指引和决策支持。
- **改善性能**。特征选择的另一个作用是节省存储和计算开销。
- **改善通用性、降低过拟合风险**。特征的增多会大大增加模型的搜索空间，大多数模型所需要的训练样本数目随着特征数量的增加而显著增加，特征的增加虽然能更好地拟合训练数据，但也可能增加方差。

好的特征选择不仅能够提升模型的性能，更能帮助我们理解数据的特点和底层结果，这对进一步改善模型和算法都有着重要作用。使用特征选择的前提是：训练数据中包含许多冗余或者无关的特征，移除这些特征并不会导致丢失信息。冗余和无关是两个概念。如果一个特征本身有用，但这个特征与另外一个有用的特征强相关，则这个特征可能就变得冗余。特征选择常用于特征很

多但样本相对较少的情况。

特征选择一般包括产生过程、评价函数、停止准则、验证过程。为了进行特征选择，我们首先需要产生特征或特征子集候选集合，其次需要衡量特征或特征子集的重要性或者好坏程度，因此需要量化特征变量和目标变量之间的联系以及特征之间的相互联系。为了避免过拟合，我们一般采用交叉验证的方式来评估特征的好坏；为了减少计算复杂度，我们可能还需要设定一个阈值，当评价函数值达到阈值后搜索停止；最后，我们需要再验证数据集上验证选出来的特征子集的有效性。

2.2.1　过滤方法

使用过滤方法进行特征选择不需要依赖任何机器学习算法，如图2-1所示。过滤方法一般分为单变量和多变量两类。单变量过滤方法不需要考虑特征之间的相互关系，而多变量过滤方法考虑了特征变量之间的相互关系。常用的单变量过滤方法是基于特征变量和目标变量之间的相关性或互信息。单变量过滤方法按照特征变量和目标变量之间的相关性对特征进行排序，过滤掉最不相关的特征变量。这类方法的优点是计算效率高、不易过拟合。由于单变量过滤方法只考虑单特征变量和目标变量的相关性，过滤方法可能选出冗余的特征，所以单变量过滤方法主要用于预处理。常用多变量过滤方法有基于相关性和一致性的特征选择。

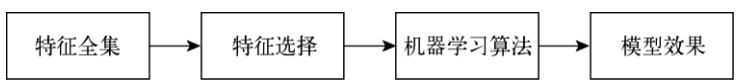


图2-1　特征选择中的过滤方法

下面详细介绍几种常用的过滤方法。

- **覆盖率**。它计算每个特征的覆盖率（特征在训练集中出现的比例）。若特征的覆盖率很小，例如我们有10 000个样本，某个特征只出现了5次，则此覆盖率对模型的预测作用不大，覆盖率很小的特征可以剔除。
- **皮尔森相关系数**。皮尔森相关系数用于度量两个变量X和Y之间的线性相关性，两个变量之间的皮尔森相关系数为两个变量之间的协方差和标准差的商：

$$\rho_{X,Y}=\frac{\operatorname{cov}(X,Y)}{\sigma_X\sigma_Y}=\frac{\mathrm{E}[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X\sigma_Y}$$

  样本上的相关系数为：

$$r=\frac{\sum_{i=1}^{n}\left(X_i-\overline{X}\right)\left(Y_i-\overline{Y}\right)}{\sqrt{\sum_{i=1}^{n}\left(X_i-\overline{X}\right)^2}\sqrt{\sum_{i=1}^{n}\left(Y_i-\overline{Y}\right)^2}}$$

- **Fisher得分**。对于分类问题，好的特征应该是在同一个类别中的取值比较相似，而在不同类别之间的取值差异比较大。因此，特征i的重要性可以用Fisher得分S_i表示：

$$S_i = \frac{\sum_{j=1}^{K} n_j (\mu_{ij} - \mu_i)^2}{\sum_{j=1}^{K} n_j \rho_{ij}^2}$$

其中，μ_{ij}和ρ_{ij}分别是特征i在类别j中均值和方差，μ_i为特征i的均值，n_j为类别j中的样本数。Fisher得分越高，特征在不同类别之间的差异性越大、在同一类别中的差异性越小，则特征越重要。

- **假设检验**。假设特征变量和目标变量之间相互独立，将其作为H0假设，选择适当检验方法计算统计量，然后根据统计量确定P值做出统计推断。例如对于特征变量为类别变量而目标变量为连续数值变量的情况，可以使用方差分析（Analysis of Variance，ANOVA），对于特征变量和目标变量都为连续数值变量的情况，可以使用皮尔森卡方检验。卡方统计量如下：

$$\chi^2 = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{(O_{i,j} - E_{i,j})^2}{{E_{i,j}}^2} = N \sum_{i,j} p_i p_j \left(\frac{(O_{i,j}/N) - p_i p_j}{p_i p_j} \right)^2$$

其中，O_i为类型为i的观测样本的个数，N为总样本数。卡方统计量取值越大，特征相关性越高。

- **互信息**。在概率论和信息论中，互信息（或Kullback-Leibler散度、相对熵）用来度量两个变量之间的相关性。互信息越大则表明两个变量相关性越高，互信息为0时，两个变量相互独立。对于两个离散随机变量X和Y，互信息计算公式如下：

$$I(X;Y) = \sum_{y \in Y} \sum_{x \in X} p(x,y) \log \left(\frac{p(x,y)}{p(x)p(y)} \right) = D_{KL}(p(x,y) \parallel p(x)p(y))$$

其中，$p(x)$和$p(y)$为X和Y的边际概率分布函数，$p(x,y)$为X和Y的联合概率分布函数。直观上，互信息度量两个随机变量之间共享的信息，也可以表示为由于X的引入而使Y的不确定度减少的量，这时候互信息与信息增益相同。

- **最小冗余最大相关性**（Minimum Redundancy Maximum Relevance，mRMR）。由于单变量过滤方法只考虑了单特征变量和目标变量之间的相关性，因此选择的特征子集可能过于冗余。mRMR方法在进行特征选择的时候考虑到了特征之间的冗余性，具体做法是对跟已选择特征的相关性较高的冗余特征进行惩罚。mRMR方法可以使用多种相关性的度量指标，例如互信息、相关系数以及其他距离或者相似度分数。假如选择互信息作为特征变量和目标变量之间相关性的度量指标，特征集合S和目标变量c之间的相关性可以定义为，特征集合中所有单个特征变量f_i和目标变量c的互信息值$I(f_i;c)$的平均值：

$$D(S,c) = \frac{1}{|S|} \sum_{f_i \in S} I(f_i;c)$$

S中所有特征的冗余性为所有特征变量之间的互信息$I(f_i;f_j)$的平均值：

$$R(S)=\frac{1}{|S|^2}\sum_{f_i,f_j\in S} I(f_i;f_j)$$

则mRMR准则定义为：

$$\text{mRMR}=\max_{S}[D(S,c)-R(S)]$$

通过求解上述优化问题就可以得到特征子集。在一些特定的情形下，mRMR算法可能对特征的重要性估计不足，它没有考虑到特征之间的组合可能与目标变量比较相关。如果单个特征的分类能力都比较弱，但进行组合后分类能力很强，这时mRMR方法效果一般比较差（例如目标变量由特征变量进行XOR运算得到）。mRMR是一种典型的进行特征选择的增量贪心策略：某个特征一旦被选择了，在后续的步骤不会删除。mRMR可以改写为全局的二次规划的优化问题（即特征集合为特征全集的情况）：

$$\text{QPFS}=\min_{x}[\alpha\boldsymbol{x}^{\mathrm{T}}\boldsymbol{H}\boldsymbol{x}-\boldsymbol{x}^{\mathrm{T}}\boldsymbol{F}]\ \text{s.t.}\sum_{i=1}^{n}x_i=1,\ x_i\geqslant 0$$

$\boldsymbol{F}$为特征变量和目标变量相关性向量，$\boldsymbol{H}$为度量特征变量之间的冗余性的矩阵。QPFS可以通过二次规划求解。QPFS偏向于选择熵比较小的特征，这是因为特征自身的冗余性$I(f_i;f_j)$。

另外一种全局的基于互信息的方法是基于条件相关性的：

$$\text{SPEC}_{\text{CMI}}=\max_{x}[\boldsymbol{x}^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{x}]\ \text{s.t.}\ \|\boldsymbol{x}\|=1,\ x_i\geqslant 0$$

其中，$Q_{ii}=I(f_i;c)$，$Q_{ij}=I(f_i;c|f_j)$，$i\neq j$。SPEC_{CMI}方法的优点是可以通过求解矩阵$\boldsymbol{Q}$的主特征向量来求解，而且可以处理二阶的特征组合。

- **相关特征选择**。相关特征选择（Correlation Feature Selection，CFS）基于以下一个假设来评估特征集合的重要性：好的特征集合包含跟目标变量非常相关的特征，但这些特征之间彼此不相关。对于包含k个特征的集合，CFS准则定义如下：

$$\text{CFS}=\max_{S_k}\left[\frac{r_{cf_1}+r_{cf_2}+\cdots+r_{cf_k}}{\sqrt{k+2(r_{f_1f_2}+\cdots+r_{f_if_j}+\cdots+r_{f_kf_1})}}\right]$$

其中，r_{cf_i}和$r_{f_if_j}$是特征变量和目标变量之间的相关性以及特征变量之间的相关性，这里的相关性不一定是皮尔森相关系数或斯皮尔曼相关系数。

过滤方法其实是更广泛的结构学习的一种特例。特征选择旨在找到跟具体的目标变量相关的特征集合，结构学习需要找到所有变量之间的相互联系，结构学习通常将这些联系表示为一个图。最常见的结构学习算法假设数据由一个贝叶斯网络生成，这时结构为一个有向图模型。特征选择中过滤方法的最优解是目标变量节点的马尔可夫毯，在贝叶斯网络中，每一个节点有且仅有马尔可夫毯。

2.2.2 封装方法

由于过滤方法与具体的机器学习算法相互独立，因此过滤方法没有考虑选择的特征集合在具体机器学习算法上的效果。与过滤方法不同，封装方法直接使用机器学习算法评估特征子集的效果，它可以检测出两个或者多个特征之间的交互关系，而且选择的特征子集让模型的效果达到最优，如图2-2所示。封装方法是特征子集搜索和评估指标相结合的方法，前者提供候选的新特征子集，后者则基于新特征子集训练一个模型，并用验证集进行评估，为每一组特征子集进行打分。最简单的方法则是在每一个特征子集上训练并评估模型，从而找出最优的特征子集。

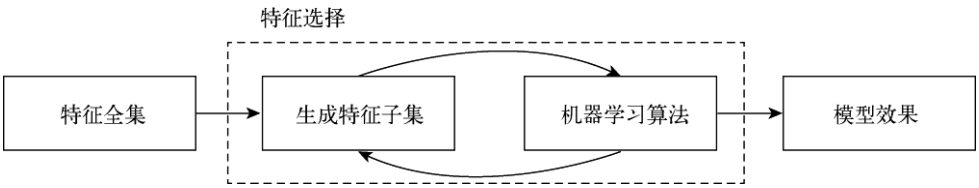


图2-2 特征选择中的封装方法

封装方法需要对每一组特征子集训练一个模型，所以计算量很大。封装方法的缺点是：样本不够充分的情况下容易过拟合；特征变量较多时计算复杂度太高。下面详细介绍几种常用的特征子集搜索算法。

- **完全搜索**。完全搜索分为穷举和非穷举两类。广度优先搜索的时间复杂度太高，它不实用；分支定界搜索在穷举搜索的基础上加入了分支限界，若断定某些分支不可能搜索出比当前找到的最优解更优的解，则可以剪掉这些分支；定向搜索首先选择N个得分最高的特征作为特征子集，将其加入一个限制最大长度的优先队列，每次从队列中取出得分最高的子集，然后穷举向该子集加入一个特征后产生的所有特征集，将这些特征集加入队列；最优优先搜索与定向搜索类似，唯一的不同是不限制优先队列的长度。
- **启发式搜索**。序列向前选择，特征子集从空集开始，每次只加入一个特征，这是一种贪心算法；序列向后选择则相反，特征子集从全集开始，每次删除一个特征；双向搜索同时使用序列向前选择和向后选择，当两者搜索到相同的特征子集时停止。对于增L去R选择算法，若算法从空集开始，每轮先添加L个特征，再删除R个特征；若算法由全集开始，则每轮先删除R个特征，再添加L个特征。序列浮动选择每次选择添加和删除的特征个数不是固定的。
- **随机搜索**。执行序列向前或者向后选择的时候，此算法随机选择特征子集。

2.2.3 嵌入方法

过滤方法与机器学习算法相互独立，而且不需要交叉验证，计算效率比较高。但是嵌入方法没有考虑机器学习算法的特点。封装方法使用预先定义的机器学习算法来评估特征子集的质量，需要很多次训练模型，计算效率很低。嵌入方法则将特征选择嵌入到模型的构建过程中，具有封

装方法与机器学习算法相结合的优点，而且具有过滤方法计算效率高的优点，如图2-3所示。嵌入方法是实际应用中最常见的方法，弥补了前面两种方法的不足。

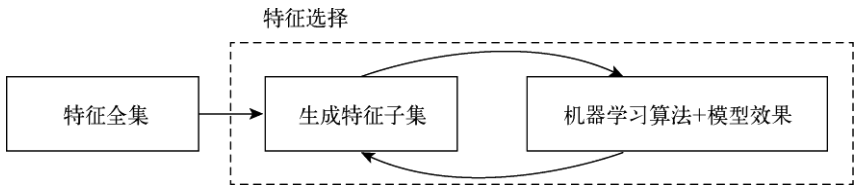


图2-3　特征选择中的嵌入方法

嵌入方法最经典的例子是LASSO（Least Absolute Shrinkage and Selection Operator）方法。

$$\min_{\beta\in\mathbb{R}^p}\left\{\frac{1}{N}||y-X\boldsymbol{\beta}||_2^2+\lambda||\boldsymbol{\beta}||_1\right\}$$

在LASSO方法之前，大家都采用岭回归，通过对回归系数进行衰减来防止过拟合，但是岭回归不能进行特征选择，对模型的可解释性没有帮助。LASSO方法类似岭回归，它通过对回归系数添加L_1惩罚项来防止过拟合，可以让特定的回归系数变为0，从而可以选择一个不包含那些系数的更简单的模型。实际应用中，λ越大，回归系数越稀疏，λ一般采用交叉验证的方式来确定。除了对最简单的线性回归系数添加L_1惩罚项之外，任何广义线性模型如逻辑回归、FM/FFM以及神经网络模型，都可以添加L_1惩罚项。除了简单的LASSO算法，嵌入方法还有结构化LASSO算法。常见的如Group LASSO算法，它对特征集合分组，对每一组采用类似LASSO的方法进行选择。

$$\min_{\beta\in\mathbb{R}^p}\left\{\frac{1}{N}||y-X\boldsymbol{\beta}||_2^2+\lambda\sum_{j=1}^{J}||\boldsymbol{\beta}_j||_{K_j}\right\},||\boldsymbol{\beta}_j||_{K_j}=\sqrt{\boldsymbol{\beta}_j{}^{\mathrm{T}}K_j\boldsymbol{\beta}_j}$$

另外一类嵌入方法是基于树模型的特征选择方法。在决策树中，深度较浅的节点一般对应的特征分类能力更强（可以将更多的样本区分开）。对于基于决策树的算法，如随机森林，重要的特征更有可能出现在深度较浅的节点，而且出现的次数可能越多。因此，可以基于树模型中特征出现次数等指标对特征进行重要性排序。

2.2.4　小结

表2-1对前面提到的三种常用的特征选择方法进行了比较。

表2-1 常用特征选择方法比较

特征选择方法		优 点	缺 点	举 例
过滤方法	单变量	速度快 可扩展 跟机器学习模型独立	忽略特征之间的关系 忽略了特征和模型之间的关系	卡方检验 信息增益 相关系数
	多变量	考虑了特征之间的相关性 跟机器学习模型独立 计算复杂度优于封装方法	计算速度和可扩展性低于单变量的方法 忽略了特征和模型之间的关系	基于相关性的特征选择（CFS） MBF FCBF
封装方法	确定性算法	简单 跟机器学习模型相关 考虑特征之间的相互作用 计算密集程度低于随机算法	容易过拟合 相比随机算法容易卡在局部最优子集（贪心搜索） 依赖机器学习模型	序列向前特征选择（SFS） 序列向后特征删减（SBE） 增q删r
	随机算法	不容易达到局部极小点 跟机器学习模型相关 考虑特征之间的相互作用	计算密集型 依赖机器学习模型 相比确定系算法过拟合的风险较高	模拟退火 随机爬山 基因算法
嵌入方法	与模型相关 计算复杂度优于封装方法 考虑特征之间的相互作用		依赖机器学习模型	决策树、随机森林、梯度提升数 SVM LASSO

2.2.5 工具介绍

针对特征选择，目前已经有很多开源的工具包可以使用。针对过滤方法，若数据量较小，可以使用Sklearn里面的feature_selection模块；若数据量较大，可以使用Spark MLlib。针对嵌入方法，一般机器学习包的线性模型都支持L_1正则，如Spark MLlib和Sklearn等。除此之外，在实际应用中比较常用的特征选择方法还有基于树模型的算法包，如Sklearn中的随机森林以及目前在工业界广泛使用的XGBoost，它们都支持根据不同指标（如增益或者分裂次数等）对特征进行排序。针对XGBoost模型，Xgbfi提供了多种指标对特征以及特征组合进行排序。

参考文献

[1] Ng A. Machine learning and AI via brain simulations. 2013.

[2] Tukey J W. Exploratory data analysis. 1977, 2.

[3] Tukey J W. We need both exploratory and confirmatory. The american statistician, 1980, 34(1): 23-25.

[4] Chen T Q , Guestrin C. XGBoost: a scalable tree boosting system. Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. ACM, 2016.

[5] Juan Y C, Zhuang Y, Chin W S, et al. Field-aware factorization machines for CTR prediction. Proceedings of the 10th ACM conference on recommender systems. ACM, 2016.

[6] He X R, Pan J F, Jin O, et al. Practical lessons from predicting clicks on ads at facebook. Proceedings of the eighth international workshop on data mining for online advertising. ACM, 2014.

[7] Micci-Barreca D. A preprocessing scheme for high-cardinality categorical attributes in classification and prediction problems. ACM SIGKDD explorations newsletter, 2001, 3(1): 27-32.

[8] Leskovec J, Rajaraman A, Ullman J D. Mining of massive datasets. Cambridge university press, 2014.

[9] Mikolov T, Chen K, Corrado T, et al. Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781, 2013.

[10] Peng H C, Long F H, Ding C. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. IEEE transactions on pattern analysis and machine intelligence, 2005, 27(8): 1226-1238.

[11] Hall M A. Correlation-based feature selection for machine learning. 1999.

[12] Tibshirani R. Regression shrinkage and selection via the lasso. Journal of the royal statistical society. Series B (Methodological) ,1996: 267-288.

[13] Guyon I, Gunn S, Nikravesh M, et al. Feature extraction: foundations and applications. Springer, 2008, 207.

[14] Zheng A. Mastering feature engineering: principles and techniques for data scientists. 2016.

机器学习技术发展迅猛，不再是书本上陌生的概念，已经在方方面面影响着人们的生活。美团是全球领先的互联网+生活服务平台，技术正在这里帮助人们吃得更好、活得更好。本书全面、真实地向读者展示了机器学习在生活服务多种场景中的成功实践。

刘彭程
美团技术委员会执行主席

机器学习技术领域很广，可以应用在不同场景；美团的业务刚好也是多种多样的，其中应用了很多不同的机器学习技术。本书把这些理论和实践结合起来，可以帮助读者更好地学习、理解和应用机器学习技术。

夏华夏
美团科学家、副总裁

和传统的机器学习相关的理论教科书相比，本书侧重于这些理论如何在真实的业务场景落地，所使用的都是美团公司内的真实案例。……希望我们在这本书中的分享能够起到抛砖引玉的作用，同时也能在这方面给广大读者带来一定的收获。

张锦懋
美团科学家

图灵社区：iTuring.cn
反馈/投稿/推荐邮箱：contact@turingbook.com
读者热线：(010) 51095186-600
分类建议 计算机/人工智能
人民邮电出版社网址：www.ptpress.com.cn

ISBN 978-7-115-48463-5

ISBN 978-7-115-48463-5
定价：79.00元