可解释机器学习

黑盒模型制作指南。

克里斯托夫莫纳尔

2019年7月16日

前言

机器学习对于改进产品、过程和研究有很大的潜力。但是计算机通常不能解释他们的预测，这是机器学习的一个障碍。这本书是关于使机器学习模型和他们的决定可解释。

在探索了可解释性的概念之后，您将学习简单的、可解释的模型，如决策树、决策规则和线性回归。后面几章重点介绍了解释黑箱模型的一般模型不可知论方法，如特征重要性和积累的局部效应，以及用沙普利值和石灰解释个别预测。

对各种解释方法进行了深入的解释和批判性的讨论。他们怎么在引擎盖下工作？他们的优点和缺点是什么？如何解释它们的输出？本书将使您能够选择和正确应用最适合您的机器学习项目的解释方法。

这本书的重点是表格式数据（也称为关系数据或结构化数据）的机器学习模型，而不是计算机视觉和自然语言处理任务。对于机器学习实践者、数据科学家、统计学家以及任何其他对机器学习模型的解释感兴趣的人，推荐阅读本书。

您可以购买PDF和电子书版本（epub、mobi）

您可以购买打印版本。

关于我：我叫克里斯托夫莫纳尔，我是一个统计学家和机器学习者。我的目标是使机器学习具有可解释性。如果您有兴趣提高机器学习模型的可解释性，请随时联系我！

Mail：

网站：

在Twitter上关注我！

封面

创意共享许可

这本书是根据授权的。

第一章引言

这本书向你解释了如何使（监督）机器学习模型可解释。本章包含一些数学公式，但即使没有公式，您也应该能够理解这些方法背后的思想。这本书不适合那些试图从头学习机器的人。如果你对机器学习不熟悉，有很多书和其他资源可以学习基础知识。我建议黑斯迪、蒂比西拉尼和弗里德曼（2009年）的《统计学习要素》一书以及在线学习平台coursera.com从机器学习开始。这本书和这门课程都是免费的！

机器学习模型解释的新方法以惊人的速度发布。要跟上出版的每一件事都是疯狂的，根本不可能。这就是为什么你在这本书中找不到最新颖和最奇特的方法，而是建立了机器学习可解释性的方法和基本概念。这些基础知识使您能够理解机器学习模型。将基本概念内化也使你能够更好地理解和评估你开始阅读这本书后5分钟内发表的关于可解释性的任何新论文（我可能夸大了出版率）。

这本书开始于一些（反乌托邦），不需要理解这本书，但希望将娱乐和使你思考。然后这本书探讨的概念。我们将讨论可解释性何时重要以及有哪些不同类型的解释。书中使用的术语可以在中查找。大多数解释的模型和方法都是用真实的数据例子来表示的，这些例子描述了使机器学习可解释的一种方法，即使用线性模型或决策树。另一种选择是使用它可以应用于任何受监控的机器学习模型。模型不可知论方法一章论述了偏相关图和置换特征重要性等方法。模型不可知方法通过改变机器学习模型的输入和测量预测输出的变化来工作。本章将讨论返回数据实例作为解释的模型不可知方法。所有模型不可知方法都可以根据它们是解释所有数据实例的全局模型行为还是单个预测来进一步区分。以下方法解释了模型的整体行为：、、和。为了解释我们的个别预测，（和密切相关的：.一些方法可以用来解释全局模型行为和单个预测的两个方面：和。

这本书以乐观的观点结尾，对可能出现的情况持乐观态度。

你可以从头到尾阅读这本书，也可以直接跳到你感兴趣的方法上。

我希望你会喜欢这本书！

1。弗里德曼、杰罗姆、特雷弗·黑斯迪和罗伯特·提比西拉尼。“统计学习要素”。www.web.stanford.edu/~hassie/elemstatlearn/（2009年）。

1.1故事时间

我们将从一些短篇小说开始。每个故事都是对可解释机器学习的夸张要求。如果你赶时间，你可以跳过故事。如果你想得到娱乐和（去）动力，继续读下去！

该格式的灵感来自杰克克拉克在他的科技故事。如果你喜欢这种故事，或者你对人工智能感兴趣，我建议你注册。

闪电从不打两次

## 2030年：瑞士医学实验室

“这绝对不是最糟糕的死法！“汤姆总结了一下，试图在悲剧中找到一些积极的东西。他从静脉输液柱上取下了泵。

“他死的原因是错误的，”莉娜补充说。

“当然是用错了吗啡泵！为我们创造更多的工作！“汤姆一边抱怨一边拧下泵的后板。拆下所有的螺丝后，他把盘子举起放在一边。他把电缆插入诊断端口。

“你不只是抱怨有工作，是吗？“莉娜嘲笑他。“当然不是。从未！”他用讽刺的语气喊道。

他启动了水泵的计算机。

莉娜把电缆的另一端插入她的平板电脑。“好吧，诊断正在运行，”她宣布。“我真的很好奇出了什么问题。”

“它确实把我们的无名氏射入了涅盘。高浓度的吗啡。人。我是说…这是第一次，对吧？通常情况下，一个坏泵会释放出太少的甜食或者什么都没有。但是，你知道，从来没有像那疯狂的射击那样，”汤姆解释说。

“我知道。你不必说服我……嘿，看看那个。”莉娜举起她的平板电脑。“你看到这里的山峰了吗？这就是止痛药混合物的功效。看！这一行显示参考水平。

这个可怜的家伙的血液系统中混合了止痛药，可以杀死他17次以上。用我们的泵注射。在这里……”她侧击，“在这里你可以看到病人死亡的时刻。”

“那么，知道发生了什么事吗，老板？“汤姆问他的上司。

“嗯……传感器看起来不错。心率、氧含量、葡萄糖等数据按预期收集。血氧数据中有些值缺失，但这并不罕见。看这里。传感器还检测到病人心率减慢，以及吗啡衍生物和其他止痛药导致的皮质醇水平极低。”她继续浏览诊断报告。

汤姆被屏幕迷住了。这是他第一次调查真正的设备故障。

“好吧，这是我们的第一块拼图。系统未能向医院的通信通道发送警告。警告已触发，但在协议级别被拒绝。可能是我们的错，也可能是医院的错。请将日志发送给IT团队，”莉娜告诉汤姆。

汤姆点了点头，眼睛仍然盯着屏幕。

莉娜继续说：“真奇怪。警告还应导致泵关闭。但显然没有做到。那一定是个虫子。质量团队错过了一些东西。真的很糟糕。可能与协议问题有关。”

“所以，泵的紧急系统不知怎么地坏了，但是为什么泵会充满香蕉，给约翰·多伊注射那么多止痛药呢？“汤姆想知道。

“好问题。你是对的。方案紧急故障除外，泵不应该使用那么多的药物。考虑到皮质醇和其他警告信号的水平较低，该算法应该提前停止。

“也许是运气不好，比如百万分之一的运气，比如被闪电击中？“汤姆问她。“不，汤姆。如果你已经阅读了我发给你的文件，你就会知道这个泵首先是在动物实验中训练的，然后是在人类身上训练的，以学习根据感觉输入来注射完美数量的止痛药。泵的算法可能不透明且复杂，但不是随机的。这意味着在同样的情况下，泵的工作原理会完全相同。我们的病人会再次死亡。感官输入的组合或不期望的交互作用一定触发了泵的错误行为。这就是为什么我们必须深入挖掘，找出这里发生的事情，”莉娜解释说。

“我明白了……”，汤姆回答，陷入沉思。“病人不是很快就会死吗？因为癌症还是什么？“

莉娜边看分析报告边点头。

汤姆站起来走到窗前。他向外看，眼睛盯着远处的一个点。

“也许机器帮了他一个忙，让他摆脱痛苦。不再受苦。也许它只是做了正确的事。就像闪电，但是，你知道，一个好的。我的意思是喜欢彩票，但不是随机的。但出于某种原因。如果我是水泵，我也会这样做的。”她最后抬起头看着他。

他一直在看外面的东西。

两人都沉默了一会儿。

莉娜又低下头继续分析。“不，汤姆。这是一只虫子……只是一只该死的虫子。“信任倒下。”

## 2050年：新加坡的地铁站

她冲到璧山地铁站。她已经在工作了。新的神经架构的测试现在应该已经完成了。她领导重新设计了政府的“个人实体纳税亲和力预测系统”，该系统预测一个人是否会向税务局隐瞒资金。她的团队想出了一个优雅的工程。如果成功，该系统不仅可以为税务局服务，还可以输入其他系统，如反恐警报系统和商业登记处。有一天，政府甚至可以将这些预测纳入公民信任评分。公民信任评分评估一个人的可信度。这个估计会影响到你日常生活的每一部分，比如贷款或者你要等多久才能拿到新护照。当她走下扶梯时，她想象着她的团队系统与公民信任评分系统的整合会是怎样的。

她经常在不降低行走速度的情况下用手擦拭RFID阅读器。她的思想被占据了，但是感官期望和现实的不一致，在她的大脑中敲响了警钟。

太晚了。

她先是鼻子撞上地铁入口大门，然后屁股先倒在地上。门本来应该开的，…但没有。她目瞪口呆地站起来，看着门旁边的屏风。“请再试一次，”屏幕上一个友好的笑脸建议道。一个人路过，不理她，把他的手擦在读者身上。门开了，他进去了。门又关上了。她擦了擦鼻子。它疼，但至少没有流血。她试图开门，但又被拒绝了。这很奇怪。也许她的公共交通账户没有足够的代币。她看了看智能手表查看帐户余额。

“登录被拒绝。请联系您的公民咨询局！“她的手表通知了她。

恶心的感觉像拳头打在她的肚子上。她怀疑发生了什么事。为了证实她的理论，她启动了移动游戏“狙击手协会”，一个自我射击。应用程序又自动关闭了，这证实了她的理论。她头晕了，又坐在地板上。

只有一个可能的解释：她的公民信任分数下降了。基本上。小幅度的下降意味着一些小的不便，比如没有头等舱的航班，或者需要等待更长的时间才能获得官方文件。信任度低的分数是罕见的，这意味着你被列为对社会的威胁。对付这些人的一个措施是让他们远离公共场所，比如地铁。政府限制公民信任分数低的受试者的金融交易。他们也开始积极监控你在社交媒体上的行为，甚至限制某些内容，比如暴力游戏。你的公民信任分数越低，就越难成倍增加。得分很低的人通常不会康复。

她想不出任何理由为什么她的分数会下降。分数基于机器学习。公民信任评分系统就像一台运转良好的发动机，运转着社会。信任评分系统的绩效始终受到密切监控。自本世纪初以来，机器学习已经好多了。它已经变得如此有效，以至于信任评分系统做出的决定不再有争议。一个可靠的系统。

她绝望地笑了。可靠的系统。只要。系统很少出故障。但失败了。她一定是那些特殊情况中的一个；制度上的错误；从现在起，她就成了被抛弃的人。没有人敢质疑这个制度。它太融入了政府，融入了社会，这是不可质疑的。在仅存的少数几个民主国家，禁止开展反民主运动，不是因为这些国家在本质上是恶意的，而是因为它们会破坏现有的制度。同样的逻辑也适用于现在更常见的算法。由于对现状的危险，禁止对算法进行批评。

算法信任是社会秩序的结构。为了共同的利益，罕见的虚假信任评分被默认。成百上千的其他预测系统和数据库输入了分数，使得不可能知道是什么导致了她的分数下降。她觉得自己身上和身下都有一个巨大的黑洞。她惊恐地看着空虚。

她的税收亲和力系统最终被纳入公民信托评分系统，但她一直不知道。

费米的回形针

## 612年（火星定居后）：火星博物馆

“历史很无聊，”Xola低声对她的朋友说。Xola，一个蓝头发的女孩，正懒洋洋地追着一架投影仪，她的左手在房间里嗡嗡作响。“历史很重要，”老师看着女孩们，用一种沮丧的声音说。Xola脸红了。她没想到老师会偷听她说话。

“Xola，你刚学了什么？”老师问她。“古人用尽了地球上所有的资源，然后死了？“她仔细地问。“不，他们让气候变热，不是人，而是电脑和机器。“这是地球，而不是地球，”另一个叫林·塞拉的女孩点头表示同意。带着一丝骄傲，老师微笑着点了点头。“你们都是对的。你知道为什么会这样吗？因为人们目光短浅，贪婪？“Xola问。“人们无法停止他们的机器！”林脱口而出。

“再说一次，你们都是对的，”老师决定，“但这比这复杂得多。当时大多数人都不知道发生了什么。有些人看到了巨大的变化，但无法逆转。这一时期最著名的作品是一首匿名作者的诗。它最能捕捉当时发生的事情。仔细听！“

老师开始写这首诗。十几架小型无人机在孩子们面前重新定位，开始将视频直接投射到他们的眼睛里。它显示一个穿着西装的人站在森林里，只剩下树桩。他开始说话：

机器计算；机器预测。

我们继续前进，因为我们是它的一部分。

我们追求一个经过训练的最佳状态。

最优解是一维的、局部的、无约束的。

硅和肉，追求指数性。

成长是我们的心态。当所有的奖励都被收集起来，副作用被忽略；当所有的硬币都被开采出来，自然已经落后；

我们会有麻烦的，

毕竟，指数增长是一个泡沫。

公地悲剧正在上演，

爆炸，

在我们眼前。

冰冷的计算和贪婪，让地球充满热量。一切都在死亡，

我们是顺从的。

就像蒙着眼睛的马一样，我们在自己创造的种族中赛跑，向着文明的伟大过滤器前进。

所以我们坚持不懈地前进。

因为我们是机器的一部分。

拥抱熵。

“黑暗的记忆，”老师说，打破了房间里的沉默。“它将上载到您的库中。你的家庭作业要记到下个星期。”Xola叹了口气。她设法抓住了一只小无人机。无人机从中央处理器和引擎发出的声音很温暖。Xola喜欢它温暖她的手。

1.2什么是机器学习？

机器学习是计算机用来根据数据做出和改进预测或行为的一套方法。

例如，为了预测房屋的价值，计算机将从过去的房屋销售中学习模式。这本书集中在监督机器学习上，它涵盖了所有的预测问题，其中我们有一个数据集，我们已经知道感兴趣的结果（如过去的房价），并希望学习预测新数据的结果。例如，从监督学习中排除的是群集任务（=无监督学习），在这些任务中，我们没有感兴趣的特定结果，但希望找到数据点的群集。此外，诸如强化学习之类的事情也被排除在外，在这种情况下，代理通过在一个环境中（如玩俄罗斯方块的计算机）的行为来学习优化某种奖励。监督学习的目标是学习一个预测模型，将数据特征（如房屋大小、位置、楼层类型等）映射到输出（如房价）。如果输出是分类的，则任务称为分类，如果输出是数值的，则称为回归。机器学习算法通过估计参数（如权重）或学习结构（如树）来学习模型。该算法由一个最小化的分数或损失函数指导。在房屋价值示例中，机器将估计房屋价格和预测价格之间的差额最小化。然后，可以使用经过充分训练的机器学习模型来预测新实例。

房价估算、产品推荐、路标检测、信用违约预测和欺诈检测：所有这些例子都有共同点，可以通过机器学习来解决。任务不同，但方法相同：

第一步：数据采集。越多越好。数据必须包含要预测的结果以及要从中进行预测的其他信息。对于街道标志检测器（“图像中有街道标志吗？”“），您将收集街道图像并标记街道标志是否可见。对于信用违约预测者，您需要实际贷款的过去数据、客户是否拖欠贷款的信息以及有助于您做出预测的数据，例如收入、过去信用违约等。对于自动房屋价值评估程序，您可以从过去的房屋销售中收集数据和有关房地产的信息，如大小、位置等。

第二步：将这些信息输入机器学习算法，生成信号检测器模型、信用评级模型或房屋价值估计器。

步骤3：将模型与新数据一起使用。将模型集成到产品或流程中，例如自动驾驶汽车、信贷申请流程或房地产市场网站。

机器在许多任务上都超过人类，例如下棋（或最近的下棋）或预测天气。即使机器和人一样好，或者在任务上有点差，在速度、再现性和缩放方面仍然有很大的优势。一个一旦实现的机器学习模型可以比人类更快地完成一项任务，可靠地提供一致的结果，并且可以无限地复制。在另一台机器上复制机器学习模型既快又便宜。训练一个人完成一项任务可能需要几十年（尤其是当他们年轻时），而且代价很高。使用机器学习的一个主要缺点是，机器解决的数据和任务的见解隐藏在越来越复杂的模型中。你需要数以百万计的数字来描述一个深层的神经网络，而没有办法完全理解这个模型。其他模型，如随机森林，由数百个决策树组成，它们“投票”预测。为了理解这个决定是如何做出的，你必须查看数百棵树中每一棵树的投票和结构。不管你有多聪明或者你的工作记忆有多好，这都是行不通的。即使每个模型都可以被解释，但性能最好的模型通常是无法解释的多个模型（也称为集合）的混合。如果只关注性能，您将自动获得越来越多不透明的模型。看一看：胜出的模型大多是模型的集合，或是非常复杂的模型，如增强树或深层神经网络。

1.3术语

为避免歧义造成混淆，本书中使用的术语定义如下：

算法是机器为达到特定目标而遵循的一组规则。可以将算法视为定义输入、输出以及从输入到输出所需的所有步骤的配方。烹饪食谱是一种算法，其中成分是输入，熟食是输出，准备和烹饪步骤是算法指令。

机器学习是一套方法，允许计算机从数据中学习，以做出和改进预测（例如癌症、每周销售、信用违约）。机器学习是从“正常编程”到“间接编程”的一种范式转换，所有指令都必须明确地提供给计算机，而“间接编程”是通过提供数据来实现的。

学习者或机器学习算法是用来从数据学习机器学习模型的程序。另一个名字是“诱导器”（例如“树诱导器”）。

机器学习模型是将输入映射到预测的学习程序。这可以是线性模型或神经网络的一组权重。“模型”这个不太具体的词的其他名称是“预测器”，或者——取决于任务——“分类器”或者“回归模型”。在公式中，经过训练的机器学习模型称为\（\hat f \）或\（\hat f（x））。

图1.1：学习者从标记的培训数据学习模型。该模型用于预测。

黑盒模型是一个不揭示其内部机制的系统。在机器学习中，“黑匣子”描述了通过查看参数（例如神经网络）无法理解的模型。黑盒的对立面有时被称为白盒，在本书中被称为可解释性，将机器学习模型视为黑盒，即使它们不是。

可解释机器学习是指使机器学习系统的行为和预测对人类可理解的方法和模型。

数据集是一个表，其中包含机器从中学习的数据。数据集包含要预测的功能和目标。当用于诱导模型时，数据集称为训练数据。

实例是数据集中的一行。“instance”的其他名称是：（data）point，example，observation。实例由功能值\（x^（i）\）和目标结果（如果已知）组成。

这些特征是用于预测或分类的输入。功能是数据集中的列。在整本书中，人们都认为特征是可以解释的，这意味着很容易理解它们的含义，比如某一天的温度或一个人的身高。特征的可解释性是一个很大的假设。但是，如果很难理解输入特性，那么更难理解模型的功能。对于单个实例，具有所有特性的矩阵称为x和\（x^（i）\）。所有实例的单个功能的向量是\（x\_j），而功能j和实例i的值是\（x^（i）\_j）。

目标是机器学会预测的信息。在数学公式中，对于单个实例，目标通常称为y或（y\_i）。

机器学习任务是一个具有特征和目标的数据集的组合。根据目标的类型，任务可以是分类、回归、生存分析、聚类或异常值检测。

预测是机器学习模型根据给定的特性“猜测”目标值应该是什么。在本书中，模型预测用\（\hat f（x^（i））或\（\hat y）表示。

1。“算法定义”。（2017年）。

第二章解释能力

对可解释性没有数学定义。Miller（2017）对（非数学）的定义是：可解释性是指人们能够理解决策原因的程度。另一个问题是：可解释性是指一个人能够始终如一地预测模型结果的程度机器学习模型的可解释性越高，就越容易理解为什么做出某些决定或预测。如果一个模型的决策比另一个模型的决策更容易理解，那么它比另一个模型更容易解释。我将同时使用可解释和可解释的术语。像Miller（2017）一样，我认为区分术语的可解释性/可解释性和解释性是有意义的。我将使用“解释”来解释个别预测。去了解我们人类认为是一个很好的解释。

* 1. 米勒，蒂姆。“人工智能的解释：社会科学的洞察”，arxiv再版：1706.07269。（2017年）。
  2. Kim、Be、Rajiv Khanna和Oluwasanmi O.Koyejo。“例子不够，要学会批判！对可解释性的批评〉，《神经信息处理系统的进展》（2016年）。

2.1可解释性的重要性

如果一个机器学习模型运行良好，为什么我们不信任这个模型而忽略它为什么做出某个决定呢？“问题在于，单一指标（如分类精度）是对大多数现实任务的不完整描述。”（Doshi Velez和Kim 2017

让我们更深入地探讨解释性如此重要的原因。当涉及到预测建模时，你必须权衡一下：你只是想知道什么是预测的吗？例如，客户流失的概率，或者某些药物对患者的疗效。或者你想知道为什么会做出预测，并且可能会因为预测性能的下降而付出解释性的代价吗？在某些情况下，您不关心做出决定的原因，只需知道测试数据集的预测性能是好的就足够了。但在其他情况下，了解“为什么”可以帮助您了解更多关于问题、数据以及模型可能失败的原因的信息。有些模型可能不需要解释，因为它们在低风险环境中使用，这意味着错误不会产生严重后果（例如电影推荐系统），或者该方法已经被广泛研究和评估（例如光学字符识别）。对可解释性的需求来自问题形式化的不完整性（Doshi Velez和Kim 2017），这意味着对于某些问题或任务，不足以得到预测（什么）。模型还必须解释它是如何得出预测的（原因），因为正确的预测只部分地解决了您最初的问题。以下原因推动了对解释和解释的需求（Doshi Velez和Kim 2017和Miller 2017）。

人类的好奇心和学习能力：人类有一个环境的心理模型，当一些意想不到的事情发生时，这个模型就会被更新。通过查找意外事件的解释来执行此更新。例如，一个人突然感到不适，他问：“为什么我感到如此不适？“。他知道每次吃那些红色浆果他都会生病。他更新了他的精神模型，并决定浆果导致了疾病，因此应该避免。当研究中使用不透明的机器学习模型时，如果模型只给出没有解释的预测，科学发现仍然完全隐藏。为了便于学习并满足好奇心，为什么某些预测或行为是由机器产生的，解释能力和解释是至关重要的。当然，人类不需要对所发生的一切进行解释。对于大多数人来说，他们不理解计算机是如何工作的是可以的。意外事件使我们好奇。例如：为什么我的计算机意外关闭？

与学习密切相关的是人类在世界上寻找意义的愿望。我们希望协调我们知识结构要素之间的矛盾或不一致。“为什么我的狗会咬我，尽管它以前从来没有这样做过？“一个人可能会问。对狗过去行为的了解与新做的、令人不快的咬伤经历之间存在着矛盾。兽医的解释调和了狗主人的矛盾：“狗处于压力和咬伤之下。”机器的决定对人的生活影响越大，机器解释它的行为就越重要。如果机器学习模型拒绝贷款申请，这对于申请者来说可能是完全意外的。他们只能用某种解释来调和期望和现实之间的这种不一致。这些解释实际上并不需要完全解释情况，但应该解决一个主要原因。另一个例子是算法产品推荐。就我个人而言，我总是想，为什么某些产品或电影在算法上被推荐给我。通常情况很清楚：网络上的广告跟随着我，因为我最近买了一台洗衣机，我知道在接下来的几天里，我会被洗衣机的广告跟随着。是的，如果我的购物车里已经有一顶冬帽，建议戴手套是有意义的。算法推荐这部电影，因为喜欢我喜欢的其他电影的用户也喜欢推荐的电影。越来越多的互联网公司在他们的建议中增加了解释。亚马逊产品推荐就是一个很好的例子，它基于经常购买的产品组合：

图2.1：从亚马逊购买油漆时的推荐产品。

在许多科学学科中，从定性方法到定量方法都发生了变化。

（例如社会学、心理学），以及机器学习（生物学、基因组学）。科学的目标是获取知识，但是许多问题都是通过大数据集和黑盒机器学习模型来解决的。模型本身成为知识的来源，而不是数据。可解释性使得提取模型捕获的额外知识成为可能。

机器学习模型承担需要安全措施和测试的现实任务。想象一下，一辆自动驾驶的汽车根据深度学习系统自动检测骑自行车的人。您希望100%地确定系统所学的抽象是无错误的，因为在骑自行车的人身上运行是非常糟糕的。一个解释可能会揭示，最重要的学习功能是识别自行车的两个轮子，这个解释可以帮助您思考边缘情况，例如自行车的侧面袋部分覆盖了车轮。

默认情况下，机器学习模型从训练数据中提取偏差。这可以把你的机器学习模式变成歧视受保护群体的种族主义者。可解释性是检测机器学习模型偏差的一种有用的调试工具。你培训过的自动批准或拒绝信用申请的机器学习模式可能会歧视少数人。你的主要目标是只向最终会偿还贷款的人发放贷款。在这种情况下，问题公式的不完整性在于，你不仅想要最小化贷款违约，而且也有义务不根据某些人口统计数据加以歧视。这是问题公式（以低风险和合规的方式发放贷款）中的一个附加约束，而机器学习模型优化所针对的损失函数没有涵盖这一约束。

将机器和算法整合到日常生活中的过程需要可解释性，以增加社会接受度。人们把信仰、欲望、意图等归因于物体。在一个著名的实验中，海德和西梅尔（1944年）向参与者展示了一些形状的视频，其中一个圆圈打开一扇“门”进入一个“房间”（只是一个长方形）。参与者描述了形状的动作，就像他们描述了人的行为一样，为形状分配了意图，甚至情感和个性特征。机器人就是一个很好的例子，就像我的吸尘器，我把它命名为“狗”。如果Doge被卡住了，我想：“Doge想继续打扫，但因为它卡住了而向我求助。”后来，当Doge打扫完并搜索家庭基地重新充电时，我想：“Doge有重新充电的愿望，并打算找到家庭基地。”我还将人格特征归为：“狗有点笨，但很可爱。”这是我的想法，特别是当我发现狗在尽职地打扫房子的时候撞倒了一棵植物。解释其预测的机器或算法会得到更多的认可。另见，它认为解释是一个社会过程。

解释用于管理社会互动。通过创造某事物的共同意义，解释者影响着接受解释者的行为、情感和信仰。机器要与我们互动，可能需要塑造我们的情感和信仰。机器必须“说服”我们，这样它们才能达到预期的目标。我不会完全接受我的机器人真空吸尘器，如果它没有在某种程度上解释它的行为。真空吸尘器创造了一个共同的意义，例如，“事故”（如被困在浴室地毯上……再次），解释说它被卡住了，而不是简单地停止工作而不发表评论。有趣的是，解释机器（创建信任）的目标和接收者的目标（理解预测或行为）之间可能存在偏差。也许对Doge被卡住的原因的完整解释可能是电池电量非常低，其中一个轮子工作不正常，还有一个bug让机器人一次又一次地走到同一个地方，即使有障碍物。这些原因（还有其他一些）导致机器人被卡住，但它只解释了有什么东西挡在路上，这足以让我相信它的行为，并得到事故的共同含义。顺便说一下，多格又被困在浴室里了。我们每次都要把地毯拆下来，然后让狗吸尘。

图2.2：我们的真空吸尘器Doge卡住了。作为事故的解释，多格告诉我们，它需要在一个平坦的表面上。

机器学习模型只有在可以解释时才能进行调试和审核。即使在低风险环境中，例如电影推荐，在研究和开发阶段以及部署之后，解释能力也是很有价值的。后来，当一个模型用于一个产品时，事情可能会出错。对错误预测的解释有助于理解错误的原因。它为如何修复系统提供了一个方向。考虑一个Husky与Wolf分类器的例子，它将一些Huskie误认为是狼。使用可解释的机器学习方法，您会发现错误分类是由于图像上的雪造成的。分类器学会了使用snow作为一个特征来将图像分类为“wolf”，这对于在训练数据集中分离狼和huskie可能是有意义的，但在实际应用中却没有意义。

如果您能够确保机器学习模型能够解释决策，您还可以更容易地检查以下特征（Doshi Velez和Kim 2017）：

* 公平性：确保预测是公正的，不含蓄或明确地歧视受保护的群体。一个可解释的模型可以告诉你为什么它决定了一个人不应该得到贷款，而且对于一个人来说，判断这个决定是否基于学习人口统计学（如种族）的偏见会变得更容易。
* 隐私：确保保护数据中的敏感信息。
* 可靠性或稳健性：确保输入的小变化不会导致预测的大变化。
* 因果关系：检查是否只找到因果关系。
* 信任：与黑板相比，人们更容易信任解释其决策的系统。

当我们不需要解释的时候。

以下场景说明了当我们不需要甚至不需要机器学习模型的可解释性时。

如果模型没有显著影响，则不需要解释性。想象一下有人

迈克正在做一个机器学习方面的项目，根据Facebook的数据预测他的朋友们下一个假期会去哪里。迈克只是喜欢用有根据的猜测让他的朋友们吃惊，他们会去哪里度假。如果模型是错误的，那就没有真正的问题了（最坏的情况是，迈克有点尴尬），如果迈克不能解释模型的输出，那就没有问题了。在这种情况下，没有可解释性是完全可以的。如果迈克开始围绕这些度假目的地的预测建立业务，情况将会改变。如果模型是错误的，企业可能会赔钱，或者模型可能会因为种族偏见而对某些人更糟。一旦模型产生重大影响，无论是金融还是社会，可解释性就变得相关。

当问题研究得很好时，不需要解释性。对一些应用进行了充分的研究，以便对模型有足够的实践经验，并随着时间的推移解决了模型的问题。一个很好的例子是光学字符识别的机器学习模型，它处理信封中的图像并提取地址。这些系统有多年的使用经验，很明显它们是有效的。此外，我们对获取有关手头任务的更多信息并不真正感兴趣。

可解释性可能使人或程序能够操纵系统。欺骗系统的用户的问题是由模型的创建者和用户的目标不匹配造成的。信用评分是这样一个系统，因为银行希望确保贷款只发放给可能归还贷款的申请人，而申请人的目标是获得贷款，即使银行不想给他们一个。这两个目标之间的不匹配导致了鼓励申请者对系统进行博弈，以增加他们获得贷款的机会。如果申请人知道拥有两张以上的信用卡会对他的分数产生负面影响，他只需返回第三张信用卡来提高分数，并在贷款获得批准后组织新的信用卡。虽然他的分数有所提高，但实际偿还贷款的可能性保持不变。只有当输入是因果特性的代理时，系统才能进行博弈，但实际上不会导致结果。尽可能避免使用代理功能，因为它们使模型具有可玩性。例如，谷歌开发了一个名为“谷歌流感趋势”的系统来预测流感爆发。该系统将谷歌搜索与流感爆发相关联，但其表现不佳。搜索查询的分布发生了变化，谷歌流感趋势错过了许多流感爆发。谷歌搜索不会引起流感。

当人们寻找“发烧”这样的症状时，这仅仅是与实际流感爆发的相关性。理想情况下，模型只使用因果特性，因为它们不可玩。

1. Doshi Velez，终曲，曾是Kim。“对可解释机器学习的严格科学研究”，编号：ML:1-13。（2017年）。
2. 海德、弗里茨和玛丽安·西梅尔。“表观行为的实验研究”，《美国心理学杂志》57（2）。JSTOR:243–59。（1944年）。

2.2解释方法分类

机器学习的可解释性方法可以根据不同的标准进行分类。

内在的还是事后的？该标准通过限制机器学习模型（固有）的复杂性或应用培训后分析模型的方法（事后分析）来区分是否实现了可解释性。内在可解释性是指由于结构简单而被认为是可解释的机器学习模型，如短决策树或稀疏线性模型。事后解释性是指模型训练后解释方法的应用。例如，排列特征重要性是一种事后解释方法。事后处理方法也可以应用于本质上可解释的模型。例如，可以计算决策树的排列特征重要性。本书各章的组织由和之间的区别决定。

解释方法的结果可以根据不同的解释方法的结果进行粗略的区分。

* 特征汇总统计：许多解释方法为每个特征提供汇总统计。有些方法返回每个特征一个数字，例如特征重要性，或者更复杂的结果，例如成对特征交互强度，它由

每个功能对的编号。

* 功能摘要可视化：大多数功能摘要统计信息也可以可视化。有些功能摘要实际上只有在可视化的情况下才有意义，而表是错误的选择。特征的部分依赖就是这样一种情况。部分相关图是显示特征和平均预测结果的曲线。表示部分依赖关系的最佳方法是实际绘制曲线，而不是打印坐标。
* 模型内部（例如学习的权重）：对本质上可解释的模型的解释属于这一类。例如，线性模型中的权重或决策树的学习树结构（用于分割的特征和阈值）。例如，在线性模型中，模型内部和特征汇总统计之间的线条模糊，因为权重同时是模型内部和特征汇总统计。输出模型内部的另一种方法是在卷积神经网络中学习特征检测器的可视化。输出模型内部的可解释性方法是通过定义特定于模型的方法（见下一个标准）。
* 数据点：此类别包括返回数据点（已存在或新创建）以使模型可解释的所有方法。一种方法叫做反事实解释。为了解释对数据实例的预测，该方法通过改变预测结果以相关方式改变的某些特征（例如预测类中的翻转），找到类似的数据点。另一个例子是识别预测类的原型。为了有用，输出新数据点的解释方法要求可以解释数据点本身。这对图像和文本很有效，但对于具有数百个特性的表格数据不太有用。
* 本质上可解释模型：解释黑盒模型的一个解决方案是用可解释模型（全局或局部）对其进行近似。可解释模型本身是通过查看内部模型参数或特性摘要统计信息来解释的。

具体型号还是不可知型号？特定于模型的解释工具仅限于特定的模型类。线性模型中回归权重的解释是特定于模型的解释，因为根据定义，本质上可解释模型的解释总是特定于模型的。仅用于解释神经网络的工具是特定于模型的。模型不可知工具可用于任何机器学习模型，并在模型经过培训（事后）后应用。这些不可知论方法通常通过分析特征输入和输出对来工作。根据定义，这些方法不能访问模型内部，如权重或结构信息。

本地还是全球？解释方法是否解释单个预测或整个模型行为？还是范围介于两者之间？在下一节中阅读关于范围标准的更多信息。

2.3解释范围

算法训练产生预测的模型。每个步骤都可以根据透明度或可解释性进行评估。

* + 1. 算法透明度

算法是如何创建模型的？

算法透明度是指算法如何从数据中学习模型，以及它可以学习到什么样的关系。如果使用卷积神经网络对图像进行分类，可以解释该算法在最底层学习边缘检测器和滤波器。这是对算法如何工作的理解，但不适用于最终学习的特定模型，也不适用于单个预测的生成方式。算法的透明性只需要对算法的了解，而不需要对数据或学习模型的了解。这本书的重点是模型的可解释性，而不是算法的透明度。对线性模型的最小二乘法等算法进行了深入的研究和理解。他们的特点是透明度高。深度学习方法（通过具有数百万权重的网络推动梯度）不太容易理解，内部工作是正在进行的研究的重点。它们被认为是不透明的。

* + 1. 全球整体模型的可解释性

经过训练的模型如何进行预测？

如果你能一次理解整个模型，你可以将模型描述为可解释的（Lipton201）。要解释全局模型输出，您需要经过培训的模型、算法知识和数据。这种水平的可解释性是基于对模型特征和每个学习组件（如权重、其他参数和结构）的整体看法来理解模型是如何做出决策的。哪些特性很重要，它们之间会发生什么样的交互？全局模型的可解释性有助于理解基于特性的目标结果的分布。全球模型的可解释性在实践中很难实现。任何超过几个参数或权重的模型都不可能适合普通人的短期记忆。我认为你不能真正想象一个具有5个特征的线性模型，因为这意味着要在5维空间中从思想上绘制估计的超平面。任何超过3维的特征空间都是人类无法想象的。通常，当人们试图理解一个模型时，他们只考虑其中的一部分，例如线性模型中的权重。

* + 1. 模块化水平上的全局模型可解释性

模型的某些部分如何影响预测？

一个具有数百个特性的朴素的Bayes模型对我和您来说太大了，无法保存在我们的工作记忆中。即使我们能够记住所有的权重，我们也无法快速预测新的数据点。此外，您需要在头脑中对所有特性进行联合分布，以估计每个特性的重要性以及特性如何平均影响预测。一项不可能的任务。但是你很容易理解一个重量。虽然全局模型的可解释性通常是无法达到的，但至少有机会在模块级别上理解某些模型。并非所有模型都可以在参数级别上解释。对于线性模型，可解释部分是权重，对于树来说，是分割（选定的特征加上截止点）和叶节点预测。例如，线性模型看起来似乎可以在模块化级别上完美地解释，但单个权重的解释与所有其他权重是相互关联的。对单个权重的解释总是伴随着脚注，即其他输入特性保持相同的值，这在许多实际应用中并非如此。一个预测房屋价值的线性模型，考虑到房屋的大小和房间的数量，可以对房间特征有负权重。这种情况可能发生，因为已经存在高度相关的房屋尺寸特征。在一个人们更喜欢大房间的市场中，房间少的房子比房间多的房子更值钱，如果两个房间的大小相同的话。权重仅在模型中其他特征的上下文中有意义。但是，线性模型中的权重仍然可以比深层神经网络中的权重更好地解释。

2.3.4单一预测的局部可解释性

为什么模型会对一个实例做出某种预测？

您可以放大一个实例，检查模型为这个输入预测了什么，并解释原因。如果你观察一个单独的预测，那么这个原本复杂的模型的行为可能表现得更为愉快。在局部，预测可能只依赖于线性或单调的某些特征，而不是对它们有复杂的依赖性。例如，房子的价值可能与它的大小成非线性关系。但是，如果您只查看一个特定的100平方米的房子，那么对于该数据子集，您的模型预测可能与大小成线性关系。你可以通过模拟当你增加或减少10平方米的尺寸时，预测的价格是如何变化的来发现这一点。因此，本地解释比全局解释更准确。这本书提出的方法，可以使个人的预测更容易解释。

2.3.5一组预测的局部可解释性

为什么模型对一组实例进行特定的预测？

多个实例的模型预测可以用全局模型解释方法（模块化级别）或单个实例的解释来解释。全局方法可以通过获取实例组、将其视为完整的数据集以及使用包含此子集的全局方法来应用。可以对每个实例使用单独的解释方法，然后为整个组列出或聚合。

1。模型可解释性的神话〉，《arxiv预印arxiv:1606.03490》（2016）。

2.4可解释性评价

对于机器学习中的可解释性没有真正的共识。如何测量也不清楚。但有一些初步的研究，并试图制定一些评估方法，如下面的章节所述。

Doshi Velez和Kim（2017）提出了三个评价可解释性的主要水平：

应用级评估（实际任务）：将解释放入产品中，由最终用户进行测试。想象一下有一个机器学习组件的骨折检测软件，它可以定位和标记X光片中的骨折。在应用层，放射科医生将直接测试骨折检测软件来评估模型。这需要一个良好的实验装置和对如何评估质量的理解。一个很好的基准总是一个人在解释同一个决定时有多好。

人级评估（简单任务）是一种简化的应用程序级评估。不同的是，这些实验不是由领域专家进行的，而是由外行进行的。这使得实验更便宜（特别是如果领域专家是放射科医生的话），并且更容易找到更多的测试人员。一个例子是向用户展示不同的解释，用户会选择最好的解释。

功能级别评估（代理任务）不需要人工。当所使用的模型类已经由其他人在人类级别的评估中进行了评估时，这是最有效的。例如，可能知道最终用户了解决策树。在这种情况下，解释质量的代理可能是树的深度。树越短，解释性得分越高。增加约束条件是有意义的，即与较大的树相比，树的预测性能保持良好且不会降低太多。

下一章着重于对功能级别上的单个预测的解释进行评估。我们将在评估中考虑解释的相关性质是什么？

2.5解释性质

我们要解释机器学习模型的预测。为了实现这一点，我们依赖于某种解释方法，这是一种生成解释的算法。解释通常以一种人性化的方式将实例的特征值与其模型预测联系起来。其他类型的解释包括一组数据实例（例如，对于k-最近邻模型）。例如，我们可以使用支持向量机预测癌症风险，并使用生成决策树作为解释的来解释预测。或者我们可以用线性回归模型代替支持向量机。线性回归模型已经配备了解释方法（权重解释）。

我们更详细地了解了解释方法和解释的性质（Robnik Sikonja和Bohanec，2018年）。这些属性可用于判断解释方法或解释的好坏。对于所有这些属性来说，如何正确地测量它们还不清楚，因此其中一个挑战是如何正式确定它们的计算方式。

解释方法的性质

* 表达力是该方法能够产生的解释的“语言”或结构。一种解释方法可以生成if-then规则、决策树、加权和、自然语言或其他东西。
* 半透明性描述了解释方法依赖于查看机器学习模型（如其参数）的程度。例如，依赖于内在可解释模型（如线性回归模型（特定于模型））的解释方法是高度透明的。方法仅依赖于操作输入和观察预测，其半透明性为零。根据具体情况，可能需要不同程度的半透明。高半透明性的优点是该方法可以依靠更多的信息来生成解释。低透明度的优点是解释方法更易于移植。
* 可移植性描述了机器学习模型的范围，可以使用解释方法。低半透明的方法具有较高的可移植性，因为它们将机器学习模型视为黑盒。代理模型可能是具有最高可移植性的解释方法。仅适用于复发性神经网络的方法具有低可移植性。
* 算法复杂性描述了生成解释的方法的计算复杂性。当计算时间成为生成解释的瓶颈时，必须考虑此属性。

个别解释的性质

* 准确度：一个解释如何预测看不见的数据？如果将解释用于预测而不是机器学习模型，那么高精度尤为重要。如果机器学习模型的精度也很低，并且目标是解释黑盒模型的作用，那么低精度就很好了。在这种情况下，只有忠诚才是重要的。
* 保真度：这个解释与黑匣子模型的预测有多接近？高保真度是解释的重要属性之一，因为低保真度的解释对解释机器学习模型是无用的。准确度和逼真度密切相关。如果黑盒模型具有较高的精度和解释的高保真度，说明也具有较高的精度。一些解释只提供局部保真度，这意味着解释仅与数据子集（例如

）甚至只针对单个数据实例（例如）。

* 一致性：在接受过相同任务培训的模型和产生类似预测的模型之间，解释有多大不同？例如，我在同一个任务上训练一个支持向量机和一个线性回归模型，两者都产生非常相似的预测。我用自己选择的方法计算解释，并分析解释有多不同。如果解释非常相似，说明是高度一致的。我发现这个属性有点棘手，因为这两个模型可以使用不同的特性，但得到相似的预测（也叫。在这种情况下，高一致性是不可取的，因为解释必须非常不同。如果模型确实依赖于相似的关系，则需要高一致性。
* 稳定性：对类似情况的解释有多相似？虽然一致性比较模型之间的解释，但稳定性比较固定模型的类似实例之间的解释。高稳定性意味着实例特征的细微变化不会实质性改变解释（除非这些细微变化也会强烈改变预测）。缺乏稳定性可能是解释方法方差较大的结果。换句话说，解释方法受到要解释的实例的特征值的微小变化的强烈影响。缺乏稳定性也可能是由解释方法的不确定成分引起的，如数据采样步骤，如用途。高稳定性总是可取的。
* 可理解性：人类对解释的理解程度如何？这看起来像是许多房产中的一个，但它是房间里的大象。很难定义和衡量，但要正确处理非常重要。许多人都同意，可理解性取决于观众。测量可理解性的想法包括测量解释的大小（线性模型中具有非零权重的特征的数量，决策规则的数量，…）或测试人们如何从解释中预测机器学习模型的行为。还应考虑解释中使用的特征的可理解性。特征的复杂转换可能比原始特征更难理解。
* 确定性：解释是否反映了机器学习模型的确定性？许多机器学习模型只给出预测，而没有对模型的声明，即预测是正确的。如果模型预测一个病人患癌症的概率为4%，那么另一个具有不同特征值的病人接受癌症的概率是否与4%一样确定？一个包含模型确定性的解释是非常有用的。
* 重要程度：解释在多大程度上反映了解释的特征或部分的重要性？例如，如果生成决策规则作为对单个预测的解释，那么是否清楚该规则的哪些条件最重要？
* 新颖性：解释是否反映了待解释的数据实例来自远离培训数据分布的“新”区域？在这种情况下，模型可能不准确，解释可能毫无用处。新颖性的概念与确定性的概念有关。新颖性越高，由于缺乏数据，模型的确定性就越低。
* 代表性：一个解释包括多少个实例？解释可以覆盖整个模型（例如线性回归模型中的权重解释），也可以只表示单个预测（例如）。

1。Robnik Sikonja、Marko和Marko Bohanec。“基于扰动的预测模型解释”，人类和机器学习。施普林格，查姆。159-175页。（2018年）。

2.6人性化解释

让我们更深入地挖掘并发现我们人类所认为的“好的”解释以及对可解释机器学习的影响。人文科学研究可以帮助我们找到答案。Miller（2017）对有关解释的出版物进行了大量调查，本章以他的总结为基础。

在这一章中，我想让你相信以下几点：作为一个事件的解释，人类更喜欢简短的解释（只有1或2个原因），将当前的情况与事件不会发生的情况进行对比。特别是异常原因提供了很好的解释。解释是解释者与被解释者（接受解释者）之间的社会互动，因此社会语境对解释的实际内容有很大的影响。

当你需要一个特定的预测或行为的所有因素的解释时，你不需要一个人类友好的解释，而是一个完全的因果归因。如果法律要求您指定所有影响功能或调试机器学习模型，则可能需要因果属性。在这种情况下，忽略以下几点。在所有其他情况下，当外行或时间很少的人是解释的接受者时，下面的部分对您来说应该很有趣。

2.6.1什么是解释？

解释是为什么问题的答案（Miller，2017年）。

* 为什么治疗不起作用？
* 为什么我的贷款被拒绝了？
* 为什么外星人还没有联系我们？

前两个问题可以用“日常”的解释来回答，而第三个问题则来自“更普遍的科学现象和哲学问题”。我们关注“日常”类型的解释，因为这些解释与可解释机器学习相关。以“如何”开头的问题通常可以改为“为什么”问题：“我的贷款是如何被拒绝的？“可以变成”为什么我的贷款被拒绝了？“。

在下文中，“解释”一词是指解释的社会和认知过程，也指这些过程的产物。解释者可以是人或机器。

2.6.2什么是好的解释？

本节进一步浓缩了米勒关于“好”解释的总结，并为可解释机器学习添加了具体含义。

解释具有对比性（Lipton，1990年）。人类通常不会问为什么会做出某种预测，但为什么会做出这种预测而不是另一种预测。我们倾向于在反事实的情况下思考，即，“如果输入x不同，预测会是怎样的？”“。对于房价预测，业主可能会感兴趣，为什么预测价格比他们预期的低价格高。如果我的贷款申请被拒绝，我不想听到所有的因素，通常是赞成或反对拒绝。我对申请中需要更改才能获得贷款的因素感兴趣。我想知道我的申请和我的申请将被接受的版本之间的对比。认识到对比解释的重要性是可解释机器学习的一个重要发现。从大多数可解释的模型中，您可以提取一个解释，它隐式地将实例的预测与人工数据实例的预测或实例的平均值进行对比。医生可能会问：“为什么这种药对我的病人不起作用？“。他们可能需要一种解释，将他们的患者与药物作用的患者以及与无反应患者相似的患者进行对比。对比解释比完整解释更容易理解。医生对药物为什么不起作用的问题的一个完整解释可能包括：病人已经有10年的疾病，11个基因过度表达，病人身体很快将药物分解成无效的化学物质，…一个对比的解释可能是更简单：与有反应的患者相比，无反应的患者具有一定的基因组合，使药物的疗效降低。最好的解释是强调感兴趣的对象和参考对象之间最大的差异。这对于可解释机器学习意味着什么：人类不希望对预测有一个完整的解释，而是希望将不同之处与另一个实例的预测（可以是人工的）进行比较。创建对比解释依赖于应用程序，因为它需要一个参考点来进行比较。这可能取决于要解释的数据点，也取决于接收解释的用户。一个房价预测网站的用户可能想得到一个与他们自己的房子或网站上的另一个房子或附近的一个普通房子的房价预测对比的解释。自动创建对比解释的解决方案还可能涉及在数据中查找原型或原型。

选择解释。人们不希望对事件的实际原因和完整原因进行解释。我们习惯于从各种可能的原因中选择一个或两个原因作为解释。作为证据，打开电视新闻：“股票价格的下跌被归咎于由于最新软件更新的问题而对该公司产品越来越强烈的反弹。”

“朱巴萨和他的球队因为防守薄弱而输掉了比赛：他们给对手太多的空间来发挥他们的战略。”

“对既定机构和政府越来越不信任是降低选民投票率的主要因素。”

一个事件可以由各种原因解释的事实被称为拉森效应。

Rashomon是一部日本电影，讲述了一个武士死亡的另类、矛盾的故事（解释）。对于机器学习模型，如果能从不同的特性做出一个好的预测是有利的。将多个模型与不同的特性（不同的解释）结合在一起的集成方法通常表现良好，因为对这些“故事”进行平均可以使预测更加可靠和准确。但这也意味着有不止一个选择性的解释为什么做出了某种预测。

它对可解释机器学习意味着什么：做一个非常简短的解释，只给出1到3个原因，即使世界更复杂。这个做得很好。

解释是社会性的。它们是解释者和解释接受者之间对话或互动的一部分。社会语境决定了解释的内容和性质。如果我想向技术人员解释为什么数字加密货币价值如此之高，我会这样说：“分散的、分布式的、基于区块链的账本，不能由一个中央实体控制，与那些想确保财富安全的人产生共鸣，这就解释了但对我的祖母，我会说：“看，奶奶：加密货币有点像电脑黄金。人们喜欢并为黄金付出很多，年轻人喜欢并为电脑黄金付出很多。”

它对可解释机器学习意味着什么：注意机器学习应用程序的社会环境和目标受众。正确使用机器学习模型的社会部分完全取决于您的具体应用。找人文学科的专家（如心理学家和社会学家）来帮助你。

解释的重点是异常。人们更关注异常原因来解释事件（Kahnemann和Tversky，1981年）。这些都是发生概率很小但仍然发生的原因。消除这些异常原因将大大改变结果（反事实解释）。人类认为这些“异常”原因是很好的解释。Trumbelj和Kononenko（2011）的一个例子是：假设我们有教师和学生之间的测试情况数据集。学生们参加一门课程，并在成功演讲后直接通过该课程。老师可以选择另外问学生问题来测试他们的知识。不能回答这些问题的学生将不及格。学生可以有不同程度的准备，这转化为正确回答老师问题的不同概率（如果他们决定测试学生）。我们要预测一个学生是否会通过这门课，并解释我们的预测。如果老师没有提出任何额外的问题，通过的概率是100%，否则通过的概率取决于学生的准备水平和正确回答问题的概率。

情景1：教师通常会向学生提出额外的问题（例如，100次中有95次）。一个没有学习的学生（10%的机会通过问题部分）不是幸运的学生，他得到了额外的问题，但他没有正确回答。学生为什么不及格？我想说不学习是学生的错。

情景2：教师很少问其他问题（例如，100次中有2次）。对于一个没有为这些问题学习过的学生来说，我们预测通过课程的可能性很高，因为问题不太可能。当然，其中一个学生没有准备好这些问题，这给了他10%的机会通过这些问题。他很不走运，老师又问了一些学生不能回答的问题，结果他没能通过这门课。失败的原因是什么？我认为现在更好的解释是“因为老师测试了学生”。老师不太可能考试，所以老师表现异常。

这对于可解释机器学习意味着什么：如果一个预测的输入特征在任何意义上都是异常的（比如分类特征的一个罕见类别），并且该特征影响了预测，那么应该将其包括在解释中，即使其他“正常”特征具有对预测的影响与异常预测相同。在我们的房价预测例子中，一个不正常的特征可能是一个相当昂贵的房子有两个阳台。即使某些归因方法发现，这两个阳台的价格差异与上述平均住房面积、良好的邻里环境或最近的装修一样大，但“两个阳台”的异常特征可能是解释为什么房子如此昂贵的最好解释。

解释是真实的。事实证明，好的解释是真实的（即在其他情况下）。但令人不安的是，这并不是“好”解释的最重要因素。例如，选择性似乎比真实性更重要。仅选择一个或两个可能原因的解释很少涵盖相关原因的完整列表。选择性忽略了部分事实。例如，只有一个或两个因素导致了股市崩溃，这是不正确的，但事实是，有数百万个原因影响了数百万人的行为，最终导致了股市崩溃。

它对可解释机器学习意味着什么：解释应该尽可能真实地预测事件，在机器学习中有时被称为忠诚。所以如果我们说第二个阳台增加了一套房子的价格，那么这也应该适用于其他的房子（或者至少是类似的房子）。对人类来说，解释的忠实性不如它的选择性、对比度和社会方面重要。

好的解释与解释者先前的信仰是一致的。人类往往忽视与他们先前信仰不一致的信息。这种效应被称为确认偏差（Nickerson，1998年）。这种偏袒并不能免除解释的责任。人们往往会贬低或忽视与他们的信仰不一致的解释。这套信仰因人而异，但也有基于群体的先验信仰，如政治世界观。

它对可解释机器学习意味着什么：良好的解释与先前的信念是一致的。这很难整合到机器学习中，可能会大大降低预测性能。我们先前认为房屋大小对预测价格的影响是，房屋越大，价格越高。假设一个模型也显示了房屋大小对一些房屋的预测价格的负面影响。模型之所以了解这一点，是因为它提高了预测性能（由于一些复杂的交互作用），但这种行为与我们先前的信念强烈矛盾。您可以强制执行单调性约束（一个特征只能影响一个方向的预测），或者使用类似具有此属性的线性模型的东西。

好的解释是一般的和可能的。一个可以解释许多事件的原因是非常普遍的，可以被认为是一个很好的解释。请注意，这与异常原因能够做出良好解释的说法相矛盾。正如我所见，不正常的原因胜过一般的原因。根据定义，在给定的情况下，异常原因是罕见的。在没有异常事件的情况下，一般的解释被认为是一个很好的解释。还请记住，人们往往会误判联合事件的概率。（乔是图书管理员。他更可能是一个害羞的人还是一个喜欢读书的害羞的人？一个很好的例子是“房子之所以昂贵是因为它很大”，这是一个非常普遍的，很好的解释为什么房子昂贵或便宜。

对于可解释的机器学习意味着什么：通用性可以很容易地通过特性的支持来衡量，即解释应用到的实例数除以实例总数。

1. 彼得·利普顿。“对比解释”，《皇家哲学研究所补编》27（1990）：247-266。
2. 卡尼曼、丹尼尔和阿莫斯·特沃斯基。“模拟启发式”，斯坦福大学CA心理学系。（1981年）。
3. \_Trumbelj、Erik和Igor Kononenko。“一种可视化和解释黑盒回归模型的一般方法”，在国际自适应和自然计算算法会议上，21-30。斯普林格。（2011年）。
4. 确认偏差：在许多方面普遍存在的现象〉，《普通心理学评论》2（2）。教育出版基金会：175。（1998年）。

第三章数据集

在这本书中，所有的模型和技术都被应用于在线免费提供的真实数据集。我们将为不同的任务使用不同的数据集：分类、回归和文本分类。

3.1自行车租赁（回归）

此数据集包含华盛顿自行车租赁公司租赁自行车的每日计数，以及天气和季节信息。这些数据是由首都比克夏公开提供的。Fanaee-T和Gama（2013）增加了天气数据和季节信息。我们的目标是根据天气和一天的不同来预测租多少辆自行车。数据可以从下载。

新特性被添加到数据集中，并不是所有的原始特性都被用于本书中的示例。以下是使用的功能列表：

* 包括临时和注册用户在内的自行车数量。计数用作回归任务的目标。
* 春天、夏天、秋天或冬天。
* 指示当天是否为假日。
* 2011年或2012年。
* 自2011年1月1日（数据集中的第一天）起的天数。引入此功能是为了考虑随时间变化的趋势。
* 指示当天是工作日还是周末。
* 那天的天气情况。什么之中的一个：

晴朗，少云，部分多云，多云

雾+云，雾+碎云，雾+少云，雾

小雪、小雨+雷雨+散云、小雨+散云

大雨+冰盘+雷雨+薄雾、雪+薄雾

* 温度（摄氏度）。
* 相对湿度百分比（0到100）。
* 风速，单位：km/h。

对于本书中的例子，数据已经过了一些处理。您可以在本书中找到处理脚本和

1。Fanaee-t、Hadi和Joao Gama。“集成探测器与背景知识相结合的事件标记”，《人工智能进展》。施普林格柏林海德堡，1-15。（2013年）。

3.2 YouTube垃圾邮件评论（文本分类）

作为文本分类的一个例子，我们使用了来自5个不同YouTube视频的1956条评论。值得庆幸的是，在一篇关于垃圾邮件分类的文章中使用此数据集的作者制作了这些数据（Alberto、Lochter和Almeida（2015年）。

这些评论是在2015年上半年通过YouTube API从YouTube上观看最多的十个视频中的五个收集的。所有5个都是音乐视频。其中之一是韩国艺术家psy的“江南风格”。其他的艺术家是凯蒂·佩里、lmfao、eminem和夏奇拉。

签出一些评论。这些评论被手动标记为垃圾邮件或合法。垃圾邮件用“1”编码，合法评论用“0”编码。

内容类

嗯，不管怎样，看看这个你[管]频道：Kobyoshi02 1

嘿，伙计们，看看我的新频道，我们的第一个视频，这是我们，猴子！！！！我是

1只穿白衬衫的猴子，请留言并订阅！！！！！

为了测试，我必须说murdev.com 1我在我的频道上摇着我性感的屁股享受^^1手表？v=vtarggvgtwq检查。1

嘿，看看我的新网站！！这个网站是关于儿童的。Kidsmediausa。COM 1

订阅我的频道1我一打开它就静音。我一打开它就只想查看视图…0

你应该看看我的频道有没有搞笑的视频！！1和u应该检查我的频道，告诉我下一步该怎么做！一

你也可以去YouTube看看评论部分。但请不要被YouTube地狱抓到，最后看的是猴子从海滩上的游客那里偷喝鸡尾酒的视频。自2015年以来，谷歌垃圾邮件探测器也可能发生了很大变化。

.

如果您想玩弄这些数据，可以在书的Github存储库中找到随和的一些便利功能。

1。阿尔贝托，T\_lio C，约翰内斯诉洛克特，蒂亚戈·阿尔梅达。“TubeSpam：评论YouTube上的垃圾邮件过滤”，在机器学习和应用（ICMLA）中，IEEE第14届国际会议，138-43。IEEE。（2015年）。

3.3宫颈癌危险因素（分类）

宫颈癌数据集包含预测女性是否会患宫颈癌的指标和危险因素。这些特征包括人口统计学数据（如年龄）、生活方式和病史。数据可从下载，并由Fernandes、Cardoso和Fernandes（2017）描述。

本书示例中使用的数据功能子集包括：

* 年龄（年）
* 性伴侣数量
* 第一次性交（岁）
* 怀孕次数
* 吸烟是还是不
* 吸烟（以年计）
* 激素避孕药是或否
* 激素避孕药（以年计）
* 宫内节育器是或否（宫内节育器）
* 使用宫内节育器（IUD）的年数
* 患者是否患有性传播疾病（STD）是或否
* 性病诊断次数
* 第一次性病诊断后的时间
* 上次性病诊断以来的时间
* 活检结果为“健康”或“癌症”。目标结果。

活检是诊断宫颈癌的金标准。对于本书中的例子，活检结果被用作目标。每列的缺失值都是由模式（最频繁的值）输入的，这可能是一个坏的解决方案，因为真正的答案可能与缺失值的概率相关。可能存在偏见，因为这些问题是非常私人的。但这并不是一本关于缺失数据插补的书，因此模式插补必须足以作为例子。

要使用此数据集复制本书的示例，请在本书的Github存储库中找到和。

1。费尔南德斯、凯尔文、杰米·卡多索和杰西卡·费尔南德斯。“应用于宫颈癌筛查的部分可观察性转移学习”，伊比利亚模式识别和图像分析会议，243-50。斯普林格。（2017年）。

第四章解释模型

实现可解释性的最简单方法是只使用创建可解释模型的算法子集。线性回归、逻辑回归和决策树是常用的可解释模型。

在下面的章节中，我们将讨论这些模型。不详细，只有基础知识，因为已经有大量的书、视频、教程、论文和更多可用的材料。我们将关注如何解释模型。这本书讨论得很详细。它还列出了。

除k-最近邻法外，本书中解释的所有可解释模型都可以在模块级别上解释。下表概述了可解释的模型类型及其属性。如果特征和目标之间的关联是线性的，那么模型就是线性的。具有单调性约束的模型确保特征和目标结果之间的关系在整个特征范围内始终朝着相同的方向：特征值的增加要么总是导致目标结果的增加，要么总是导致目标结果的减少。单调性对于模型的解释是有用的，因为它使理解关系变得更容易。一些模型可以自动包含功能之间的交互，以预测目标结果。通过手动创建交互功能，可以将交互包括在任何类型的模型中。交互可以提高预测性能，但太多或太复杂的交互可能会损害可解释性。有些模型只处理回归，有些只处理分类，还有一些模型两者都处理。

从这个表中，您可以为您的任务选择一个合适的可解释模型，回归（regr）或分类（class）：

算法LinearMonotoneInteractionOnTask

|  |  |  |  |
| --- | --- | --- | --- |
| 线性回归是的 | 是的 | 不 | 雷格 |
| 逻辑回归否 | 是的 | 不 | 班 |
| 决策树否 | 一些 | 是的 | 类，regr |
| 规则匹配是 | 不 | 是的 | 类，regr |
| 天真的拜斯不 | 是的 | 不 | 班 |
| K-最近的邻居 | 不 | 不 | 类，regr |

4.1线性回归

线性回归模型将目标预测为特征输入的加权和。所学关系的线性使解释变得容易。长期以来，统计学家、计算机科学家和其他解决定量问题的人都使用线性回归模型。

线性模型可用于模拟回归目标y对某些特征x的依赖性。所学关系是线性的，可以针对单个实例i编写如下：

\[Y=\β0+β1 x 1+ldots+\βp x p+epsilon\]

实例的预测结果是其p特征的加权和。betas（\（\ beta j \）表示学习的特征权重或系数。总和中的第一个权重（\（\β\u 0\）称为截距，不与特征相乘。epsilon（\（\ epsilon\）是我们仍然犯的错误，即预测与实际结果之间的差异。假设这些误差服从高斯分布，这意味着我们在正负方向上都会产生误差，并且会产生许多小的误差和很少的大的误差。

可以使用各种方法来估计最佳权重。通常使用普通最小二乘法来找到最小化实际结果和估计结果之间平方差的权重：

\[\hat \boldSymbol \beta=\arg\！\最小\贝塔\贝塔\贝塔\贝塔\贝塔贝塔贝塔

我们不会详细讨论如何找到最佳权重，但如果您感兴趣，可以阅读《统计学习要素》（Friedman、Hastie和Tibshirani 2009）一书的第3.2章或线性回归模型上的其他在线资源。

线性回归模型的最大优点是线性：它使估计过程简单，最重要的是，这些线性方程在模块化水平（即权重）上具有易于理解的解释。这是线性模型和所有类似模型在医学、社会学、心理学等许多定量研究领域如此广泛的主要原因之一。例如，在医学领域，不仅要预测患者的临床结果，而且要量化药物的影响，同时以可解释的方式考虑性别、年龄和其他特征。

估计的权重带有置信区间。置信区间是权重估计的一个范围，它以一定的置信度覆盖“真实”权重。例如，权重为2的95%置信区间可能在1到3之间。这个区间的解释是：如果我们用新的抽样数据重复估计100次，置信区间将包括100个案例中95个的真实权重，假设线性回归模型是正确的数据模型。

模型是否为“正确”模型取决于数据中的关系是否满足某些假设，即线性、正态性、同方差性、独立性、固定特征和不存在多重共线性。

线性度

线性回归模型迫使预测成为特征的线性组合，这是其最大的强度和最大的限制。线性导致可解释模型。线性效应易于量化和描述。它们是添加剂，因此很容易分离效果。如果怀疑特征交互或特征与目标值的非线性关联，可以添加交互项或使用回归样条线。

正态性

假设给定特征的目标结果服从正态分布。如果违反此假设，则特征权重的估计置信区间无效。

同方差（常方差）

假设误差项的方差在整个特征空间内是恒定的。假设你想根据居住面积（平方米）来预测房子的价值。您估计一个线性模型，假设不管房子大小，预测响应周围的误差具有相同的方差。这种假设在现实中经常被违背。在房屋实例中，对于较大的房屋，预测价格周围的误差项方差可能更高，因为价格更高，价格波动的空间也更大。假设线性回归模型中的平均误差（预测价格和实际价格之间的差异）为50000欧元。如果假设同方差，则假设平均误差为50000，对于成本为100万的房屋和成本仅为40000的房屋。这是不合理的，因为这意味着我们可以预期负房价。

独立性

假设每个实例独立于任何其他实例。如果您执行重复测量，例如每个患者进行多次血液测试，则数据点不是独立的。对于依赖数据，您需要特殊的线性回归模型，如混合效应模型或GEE。如果使用“正态”线性回归模型，可能会从模型中得出错误的结论。

固定功能

输入特性被认为是“固定的”。固定意味着它们被视为“给定常数”，而不是统计变量。这意味着它们没有测量误差。这是一个相当不切实际的假设。然而，如果没有这个假设，您将不得不适应非常复杂的测量误差模型，这些模型解释了输入特性的测量误差。通常你不想这样做。

不存在多重共线性

您不需要强相关的特性，因为这会扰乱对权重的估计。在两个特征强相关的情况下，由于特征效果是可加的，因此估计权重变得有问题，并且无法确定哪一个相关特征将影响归为属性。

4.1.1解释

线性回归模型中权重的解释取决于相应特征的类型。

* 数值特征：增加一个单位的数值特征会改变其权重的估计结果。数字特征的一个例子是房屋的大小。
* 二进制特性：为每个实例获取两个可能值之一的特性。例如，“房子有花园”。其中一个值被视为引用类别（在某些用0编码的编程语言中），例如“no garden”。将特征从引用类别更改为其他类别会根据特征的权重更改估计结果。
* 具有多个类别的分类特征：具有固定数量的可能值的特征。

例如，“地板类型”功能，可能有“地毯”、“强化木地板”和“拼花地板”等类别。处理多个类别的解决方案是一个热编码，这意味着每个类别都有自己的二进制列。对于具有L类别的分类功能，您只需要L-1列，因为L-th列将具有冗余信息（例如，当列1到L-1的值都为0时，例如，我们知道此实例的分类功能具有L类别）。然后对每个类别的解释与对二进制特征的解释相同。有些语言，例如R，允许您以各种方式对分类特征进行编码，例如

* 截距\（\beta\u 0\）：截距是“常量特征”的特征权重，对于所有实例都是1。大多数软件包会自动添加这个“1”特性来估计截获。解释是：例如，所有数字特征值

在零和分类特征值的参考类别，模型预测是截距权重。截距的解释通常不相关，因为所有特征值都为零的实例通常没有意义。只有当特性标准化（平均值为零，标准偏差为一）时，解释才有意义。然后，截距反映了一个实例的预测结果，其中所有特征都处于其平均值。

使用以下文本模板可以自动解释线性回归模型中的特征。

## 数字特征的解释

当所有其他特征值保持不变时，特征值（x\_k\_）增加一个单位，Y的预测值增加一个单位。

## 分类特征的解释

当所有其他特性保持不变时，将特性（x\_k\_）从引用类别更改为其他类别会增加对y的预测（beta\_k\_）。

解释线性模型的另一个重要度量是r平方度量。rsquared告诉你目标结果的总方差中有多少是由模型解释的。r平方越高，模型对数据的解释就越好。计算平方的公式为：

\[R^2=1-SSE/SST\]

SSE是误差项的平方和：

\[sse=\sum\_i=1 ^n（y^（i）-\hat y（i））^2\]

sst是数据方差的平方和：

\[sst=\sum\_i=1 ^n（y^（i）-\bar y）^2\]

SSE告诉你在拟合线性模型后还有多少方差，这是通过预测值和实际目标值之间的平方差来测量的。SST是目标结果的总方差。r平方告诉你有多少方差可以用线性模型来解释。对于模型根本不解释数据的模型，r平方的范围为0；对于解释数据中所有方差的模型，r平方的范围为1。

有一个陷阱，因为r平方随模型中的特性数量的增加而增加，即使它们根本不包含任何关于目标值的信息。因此，最好使用调整后的r平方，这就解释了模型中使用的特征数量。其计算如下：

\[\bar r ^2=r^2-（1-r^2）\frac p n-p-1 \]，其中p是功能的数量，n是实例的数量。

解释一个R平方很低（调整后）的模型是没有意义的，因为这样的模型基本上不能解释大部分的方差。对权重的任何解释都没有意义。

## 功能重要性

一个特征在线性回归模型中的重要性可以用它的t统计量的绝对值来衡量。t统计量是以标准误差为尺度的估计权重。

\[T \那\贝塔=\frac \那\贝塔\贝塔

让我们检查一下这个公式告诉我们：特征的重要性随着权重的增加而增加。这是有道理的。估计权重的方差越大（=我们对正确值的确定度越低），特性就越不重要。这也是有道理的。

4.1.2示例

在这个例子中，我们使用线性回归模型来预测给定天气和日历信息的特定日期。为了便于解释，我们检查了估计的回归权重。这些特征包括数值特征和分类特征。对于每个特征，该表显示估计的权重、估计的标准误差（SE）和t统计量的绝对值（t）。

## 重量SE t|

（截距）2399.4 238.310.1季夏季899.3 122.37.4季秋季138.2 161.70.9季冬季425.6 110.8 3.8假日-686.1 203.33.4工作日工作日124.9 73.3 1.7天气预报-379.4 87.6 4.3天气预报/雪/风暴-1901.5223.68.5

|  |  |  |  |
| --- | --- | --- | --- |
| 临时雇员 | 110.7条 | 7.0条 | 15.7条 |
| 嗡嗡声 | -第17.4条 | 第3.2条 | 第5.5条 |
| 风速 | -第42.5条 | 6.9条 | 6.2条 |
| 自2011年以来的天数 | 第4.9条 | 0.2条 | 28.5条 |

数值特征（温度）的解释：当所有其他特征保持不变时，温度升高1摄氏度，自行车的预测数量增加110.7辆。

分类特征（“天气状况”）的解释：与良好天气相比，当下雨、下雪或暴风雨时，自行车的估计数量减少了-1901.5——再次假设所有其他特征不变。当天气有雾时，由于所有其他特征保持不变，自行车的预测数量比好天气低-379.4。

所有的解释总是伴随着脚注，“所有其他特征保持不变”。这是因为线性回归模型的性质。预测目标是加权特征的线性组合。估计的线性方程是特征/目标空间中的超平面（单个特征的简单直线）。权重指定超平面在每个方向上的坡度（渐变）。好的一面是，可加性将单个特征效果的解释与所有其他特征隔离开来。这是可能的，因为方程式中的所有特征效果（=权重乘以特征值）都与加号组合在一起。在事物的坏的一面，解释忽略了特征的联合分布。增加一个特性，但不改变另一个特性，可能导致不现实或至少不太可能的数据点。例如，如果不增加房子的大小，增加房间的数量可能是不现实的。

4.1.3目视解释

各种各样的可视化使线性回归模型容易和快速地为人类掌握。

## 4.1.3.1重量图

权重表的信息（权重和方差估计）可以在权重图中可视化。下图显示了前一个线性回归模型的结果。

图4.1：权重显示为点，95%置信区间显示为线。

重量图显示，下雨/下雪/暴风雨天气对预测的自行车数量有很大的负面影响。工作日特征的权重接近于零，95%的区间中包含零，这意味着该影响在统计上不显著。一些置信区间很短，估计值接近于零，但特征效应具有统计学意义。温度就是这样的一个候选者。权重图的问题在于特征是在不同的尺度上测量的。而对于天气来说，估计的重量反映了好天气和下雨/暴风雨/下雪天气之间的差异，因为温度只反映了1摄氏度的增加。在拟合线性模型之前，可以通过缩放特征（一个的零平均值和标准偏差），使估计的权重更具可比性。

## 4.1.3.2效果图

当线性回归模型的权重与实际特征值相乘时，可以更有意义地进行分析。重量取决于特征的比例，如果你有一个测量人的身高的特征，并且你从米转换到厘米，那么重量会有所不同。权重会改变，但数据中的实际效果不会改变。了解您的特性在数据中的分布也是很重要的，因为如果您的方差非常小，这意味着几乎所有的实例都有类似的贡献。效果图可以帮助您了解权重和特征组合对数据预测的贡献程度。首先计算效果，即每个特征的权重乘以实例的特征值：

\[\text效果（i）=w j x j（i）]

使用箱线图可以可视化效果。箱线图中的框包含一半数据的效果范围（25%到75%的效果分位数）。框中的垂直线是中间值效应，即50%的实例对预测的影响较小，另一半则较高。水平线延伸到\（\pm1.5\text iqr/\sqrt n \），iqr是四分位数之间的范围（75%分位数减去25%分位数）。这些点是离群值。与每个类别都有自己的行的权重图相比，分类特征效果可以总结为一个单独的箱线图。

图4.2：特征效果图显示了每个特征数据的效果分布（=特征值乘以特征权重）。

对预期租用自行车数量的最大贡献来自于温度和天数特征，这反映了自行车租赁的趋势。温度在很大程度上有助于预测。日趋势特征从零变为大的正贡献，因为数据集中的第一天（2011年1月1日）具有非常小的趋势效应，并且该特征的估计权重为正（4.93）。这意味着效果会随着每天的增加而增加，并且在数据集中的最后一天（2012年12月31日）效果最高。请注意，对于负权重的效果，具有正效果的实例是具有负特征值的实例。例如，风速负效应大的天气就是风速大的天气。

4.1.4解释个别预测

一个实例的每个特性对预测有多大贡献？这可以通过计算这个实例的效果来回答。对实例特定效果的解释仅与每个特征的效果分布相比较才有意义。我们要从自行车数据集中解释第6个实例的线性模型的预测。实例具有以下功能值。

|  |  |
| --- | --- |
| 功能 | 价值 |
| 季节 | 春天 |
| 年 | 2011年 |
| mnth公司 | 简 |
| 假日 | 没有假期 |
| 平日 | 清华大学 |
| 工作日 | 工作日 |
| 气象卫星 | 很好。 |
| 临时雇员 | 1.604356个 |
| 嗡嗡声 | 51.8261个 |
| 风速 | 6.000868年 |

## 特征值

CNT 1606号

自2015年以来的天数

为了获得这个实例的特征效果，我们必须将它的特征值乘以线性回归模型中相应的权重。对于“工作日”功能的值“工作日”，效果为124.9。对于1.6摄氏度的温度，影响是177.6。我们将这些单独的效果作为交叉点添加到效果图中，这显示了数据中效果的分布。这使我们能够将个体效应与数据中的效应分布进行比较。

图4.3：一个实例的效果图显示了效果分布，并突出了感兴趣实例的效果。

如果我们对训练数据实例的预测进行平均，我们得到的平均值是4504。相比之下，第六个实例的预测很小，因为预测的自行车租金只有1571辆。效果图揭示了原因。箱线图显示数据集所有实例的效果分布，交叉显示第6个实例的效果。第6个实例具有低温效应，因为在这一天，温度为2度，与大多数其他日期相比较低（记住温度特征的权重为正）。此外，与其他数据实例相比，趋势特征“自2011年以来的天数”的影响较小，因为该实例来自2011年初（5天），并且趋势特征也具有正权重。

4.1.5分类特征编码

分类特征的编码有几种方法，选择影响权重的解释。

线性回归模型的标准是处理编码，这在大多数情况下是足够的。使用不同的编码归根结底就是从具有分类特征的单个列创建不同的（设计）矩阵。本节介绍了三种不同的编码，但还有更多。使用的示例有六个实例和三个类别的分类功能。对于前两个实例，该特性采用类别A；对于实例3和4，采用类别B；对于后两个实例，采用类别C。

## 治疗编码

在治疗编码中，每个类别的权重是对应类别和参考类别之间预测的估计差异。线性模型的截距是参考类别的平均值（当所有其他特征保持不变时）。设计矩阵的第一列是截距，它总是1。第二列表示实例一是否在B类中，第三列表示实例一是否在C类中，A类不需要列，因为这样的话，线性方程会被过度指定，并且找不到权重的唯一解。只需知道一个实例既不属于B类也不属于C类就足够了。

功能矩阵：\[\begin pmatrix 1&0&0\\1&0&0\\1&1&0\\1&1&0\\1&0&1\\1&0&1\\end pmatrix \]效果编码

每个类别的权重是从相应类别到总体平均值的估计Y差（假定所有其他特征为零或参考类别）。第一列用于估计截距。与截距相关联的权重\（\beta 0 \）表示总体平均值，第二列的权重\（\beta 1 \）表示总体平均值与B类之间的差异。B类的总影响为\（\beta 0+β1 \）。C类的解释是等效的。对于参考类别A，总体效应与总体平均值和总体效应之间存在差异。

功能矩阵：

## 虚拟编码

每个类别的\（\beta\）是每个类别的估计平均值y（假定所有其他特征值为零或参考类别）。注意这里省略了截距，这样就可以找到线性模型权重的唯一解。

功能矩阵：

如果你想深入了解分类特征的不同编码，结帐和

4.1.6线性模型是否有很好的解释？

从构成良好解释的属性来看，线性模型并不能创建最佳解释。它们是对比的，但是引用实例是一个数据点，其中所有数字特征都为零，分类特征位于它们的引用类别中。这通常是一个人工的、毫无意义的实例，不太可能出现在您的数据或现实中。有一个例外：如果所有的数字特征都是平均中心（特征减去特征的平均值），并且所有的分类特征都是效果编码的，那么参考实例就是所有特征都接受平均特征值的数据点。这也可能是一个不存在的数据点，但至少可能更有可能或更有意义。在这种情况下，权重乘以特征值（特征效果）解释了与“平均实例”相比，对预测结果的贡献。一个好的解释的另一个方面是选择性，这可以在线性模型中通过使用较少的特征或训练稀疏的线性模型来实现。但默认情况下，线性模型不会创建选择性解释。线性模型创造真实的解释，只要线性方程是一个适当的模型，关系到特征和结果。非线性和相互作用越多，线性模型越不准确，解释越不真实。线性使得解释更为一般和简单。我认为，模型的线性本质是人们使用线性模型解释关系的主要因素。

4.1.7稀疏线性模型

我选择的线性模型的例子看起来都很漂亮和整洁，不是吗？但实际上，您可能不只是拥有少数几个特性，而是拥有成百上千个特性。你的线性回归模型呢？解释能力下降。您甚至可能会发现自己处于这样一种情况，即特性比实例多，并且根本不适合标准线性模型。好消息是，有很多方法可以将稀疏性（即很少的特征）引入线性模型。

## 4.1.7.1套索

套索法是一种将稀疏性引入线性回归模型的自动简便方法。lasso代表“最小绝对收缩和选择操作符”，当应用于线性回归模型时，执行特征选择和所选特征权重的正则化。让我们考虑权重优化的最小化问题：

\[Min \BoldSymbol \Beta \ Left（\Frac 1 n \ Sum i=1 ^n（y（i）-x t \BoldSymbol \Beta）^2 \ Right）\]Lasso为这个优化问题添加了一个术语。

\[最小\粗体符号\β\左（\frac（i）-x（i）

特征向量的l1范数术语\（\ boldSymbol \ beta u 1\导致对大权重的惩罚。由于使用了l1范数，许多权重的估计值为0，而其他权重的估计值则缩水。参数lambda（\（\lambda\）控制正则化效果的强度，通常通过交叉验证进行调整。尤其是当lambda很大时，许多权重变为0。特征权重可以可视化为惩罚项lambda的函数。每个特征权重由下图中的曲线表示。

图4.4：随着权重惩罚的增加，越来越少的功能接受非零权重估计。这些曲线也称为正则化路径。图上的数字是非零权重的数目。

我们应该为lambda选择什么值？如果您将惩罚项视为一个优化参数，那么您可以找到lambda，它通过交叉验证将模型错误最小化。还可以将lambda视为控制模型可解释性的参数。惩罚越大，模型中出现的特征就越少（因为它们的权重为零），并且模型的解释效果越好。

## 套索示例

我们将用套索预测自行车的租金。我们预先设置了模型中需要的特性数量。让我们先将数字设置为2个功能：

## 重量

季节性春季0.00季节性夏季0.00季节性秋季0.00季节性冬季0.00节假日0.00工作日0.00天气性天气性0.00天气性降雨/雪/风暴0.00

|  |  |
| --- | --- |
| 临时雇员 | 第52.33条 |
| 嗡嗡声 | 0.00美元 |
| 风速 | 0.00美元 |
| 自2011年以来的天数 | 2.15条 |

前两个在套索路径中具有非零权重的特征是温度（“temp”）和时间趋势（“自2011年以来的天数”）。

现在，让我们选择5个功能：

## 重量

|  |  |
| --- | --- |
| 季节性春天 | -389.99元 |
| 季节性夏季 | 0.00美元 |
| 季节性下降 | 0.00美元 |
| 季节性冬季 | 0.00美元 |
| 假日 | 0.00美元 |
| 工作日工作日 | 0.00美元 |
| 天气状况 | 0.00美元 |

天气雨/雪/风暴-862.27

|  |  |
| --- | --- |
| 临时雇员 | 85.58美元 |
| 嗡嗡声 | -3.04年 |
| 风速 | 0.00美元 |
| 自2011年以来的天数 | 3.82条 |

请注意，“临时”和“自2011年以来的天数”的权重不同于具有两个功能的模型。原因是，通过减少lambda，即使是已经“在”模型中的特性也会受到较少的惩罚，并可能获得更大的绝对权重。套索权重的解释与线性回归模型中权重的解释相对应。您只需要注意功能是否标准化，因为这会影响权重。在这个例子中，这些特性被软件标准化了，但是权重被自动转换回来，以便我们匹配原始的特性比例。

## 线性模型中稀疏性的其他方法

广泛的方法可以用来减少线性模型中的特征数量。

预处理方法：

* 手动选择功能：您可以始终使用专家知识来选择或放弃某些功能。最大的缺点是它不能被自动化，您需要与理解数据的人取得联系。
* 单变量选择：一个例子是相关系数。您只考虑超过特征和目标之间相关阈值的特征。缺点是它只单独考虑特性。在线性模型解释了其他一些特征之前，某些特征可能不会显示相关性。那些你会错过的单变量选择方法。

逐步方法：

* 正向选择：用一个特征拟合线性模型。对每个功能都这样做。选择最有效的模型（例如，最高r平方）。现在，对于其余的特性，通过将每个特性添加到当前的最佳模型中来适应模型的不同版本。选择表现最好的一个。继续，直到达到某个条件，例如模型中的最大特征数。
* 反向选择：类似于正向选择。但是，不要添加功能，从包含所有功能的模型开始，并尝试删除哪些功能以获得最高的性能提高。重复此操作，直到达到某个停止标准。

我建议使用lasso，因为它可以自动化，同时考虑所有功能，并且可以通过lambda进行控制。它也适用于分类。

4.1.8优势

将预测建模为一个加权和，使预测的生成变得透明。使用lasso，我们可以确保使用的功能数量仍然很小。

许多人使用线性回归模型。这意味着在许多地方，预测建模和推理都被接受。有很高水平的集体经验和专业知识，包括线性回归模型和软件实现的教材。线性回归可以在R、Python、Java、朱丽亚、斯卡拉、JavaScript、……中找到。

从数学上讲，估计权重很简单，而且您可以保证找到最佳权重（假设数据满足线性回归模型的所有假设）。

与权重一起，你可以得到置信区间、测试和可靠的统计理论。线性回归模型也有许多扩展（请参见。

4.1.9缺点

线性回归模型只能表示线性关系，即输入特征的加权和。每一个非线性或相互作用都必须是手工制作的，并明确地作为输入特征提供给模型。

对于预测性能来说，线性模型通常也不是那么好，因为可以学习的关系是如此有限，而且通常过于简单化了复杂的现实。

一个权重的解释可能是非指导性的，因为它依赖于所有其他特性。在线性模型中，与结果Y和另一个特征高度正相关的特征可能会得到负权重，因为在另一个相关特征下，它与高维空间中的Y呈负相关。完全相关的特征使得我们甚至不可能找到线性方程的唯一解。一个例子：你有一个模型来预测一个房子的价值，并且有一些特征，比如房间的数量和房子的大小。房子的大小和房间的数量是高度相关的：房子越大，拥有的房间就越多。如果将这两个特性都纳入线性模型中，可能会发生这样的情况：房屋的大小是更好的预测因素，并且会得到很大的正权重。房间的数量最终可能会得到一个负的权重，因为考虑到一个房子的大小相同，增加房间的数量可能会使它的价值降低，或者当相关性太强时，线性方程变得不稳定。

1。弗里德曼、杰罗姆、特雷弗·黑斯迪和罗伯特·提比西拉尼。“统计学习要素”。www.web.stanford.edu/~hassie/elemstatlearn/（2009年）。

4.2逻辑回归

逻辑回归模型的概率分类问题有两个可能的结果。它是分类问题线性回归模型的扩展。

4.2.1线性回归分类有什么问题？

线性回归模型能很好地进行回归，但分类失败。为什么？对于两个类，可以将其中一个类标记为0，另一个标记为1，然后使用线性回归。从技术上讲，它是可行的，大多数线性模型程序都会为您提供权重。但这种方法存在一些问题：

线性模型不输出概率，但它将类视为数字（0和1），并适合最佳超平面（对于单个特征，它是一条直线），以最小化点和超平面之间的距离。所以它只是在点之间插入，你不能把它解释为概率。

一个线性模型也会外推并给出低于零和高于一的值。这是一个很好的迹象，表明可能有一种更智能的分类方法。

由于预测的结果不是概率，而是点之间的线性插值，因此没有一个有意义的阈值可以用来区分一个类和另一个类。这一问题已得到很好的说明。

线性模型不能扩展到具有多个类的分类问题。您必须开始用2、3等标记下一个类。类可能没有任何有意义的顺序，但是线性模型会在特性和类预测之间的关系上强制一个奇怪的结构。具有正权重的特征值越高，它对具有更高数字的类的预测所起的作用就越大，即使恰好获得相似数字的类与其他类的距离不近。

图4.5：线性模型根据肿瘤的大小将其分类为恶性（1）或良性（0）。这些线条显示了线性模型的预测。对于左边的数据，我们可以使用0.5作为分类阈值。在引入更多的恶性肿瘤病例后，回归线偏移和0.5的阈值不再区分这两类。点稍微抖动以减少过度绘制。

4.2.2理论

分类的解决方案是逻辑回归。而不是拟合直线或超平面，逻辑回归模型使用逻辑函数挤压0到1之间的线性方程的输出。后勤功能定义如下：

\[\text logistic（\eta）=\frac 1 1+exp（-\eta）\]如下：

图4.6：后勤功能。它输出介于0和1之间的数字。在输入0时，输出0.5。

从线性回归到逻辑回归的步骤有点简单。在线性回归模型中，我们用一个线性方程来模拟结果和特征之间的关系：

\[\hat y（i）=\beta 0+beta 1 x ^（i）1+ldots+\beta p x ^（i）p \]

对于分类，我们倾向于0到1之间的概率，因此我们将方程的右边包装成逻辑函数。这将强制输出仅假定0和1之间的值。

\[P（Y^（I）=1）=\frac 1 1+exp（-（\beta 0+beta 1 x（I）1+ldots+\beta P x（I）p）\]

让我们再次回顾一下肿瘤大小的例子。但我们使用的不是线性回归模型，而是逻辑回归模型：

图4.7：logistic回归模型根据肿瘤大小找到良恶性之间的正确判定界限。该行是逻辑函数的移位和压缩，以适应数据。

使用逻辑回归进行分类更好，在这两种情况下，我们都可以使用0.5作为阈值。附加点的包含并不真正影响估计曲线。

4.2.3解释

逻辑回归中权重的解释不同于线性回归中权重的解释，因为逻辑回归的结果是0到1之间的概率。权重不再线性地影响概率。加权和由逻辑函数转换为概率。因此，我们需要为解释重新建立方程，以便只有线性项在公式的右侧。

\[对数\左（\frac p（y=1）1-p（y=1）\右）=log \左（\frac p（y=1）p（y=0）\右）=\beta 0＋\ beta 1 x 1＋\ ldots+\beta p x p \]

我们将log（）函数中的术语称为“odds”（事件概率除以无事件概率），用对数表示，称为log odds。

该公式表明，逻辑回归模型是对数概率的线性模型。伟大的！听起来没什么帮助！只要稍微改变一下这些术语，就可以知道当一个特性（x\_j）被1个单位改变时，预测是如何变化的。为此，我们首先可以将exp（）函数应用于公式的两边：

\[\frac p（y=1）1-p（y=1）=odds=exp\left（\beta 0+beta 1 x 1+ldots+\beta p x p right）\]

然后我们比较当我们将一个特性值增加1时会发生什么。但是，我们不看差异，而是看两个预测的比率：

\[\frac odds x j+1 odds=\frac exp\ left（\beta 0+beta 1 1+ldots+\beta j（x j+1）+\ldots+

\βββββββββββββββββββxββββββxββ

\右）\]

我们应用以下规则：\[\frac exp（a）exp（b）=exp（a-b）\]并删除许多术语：

\[\frac odds x j+1 odds=exp \ left（\beta j（x j+1）-\beta j x j right）=exp \ left（\beta j \ right）\]

最后，我们得到了一个简单的特性权重exp（）。一个功能单位的改变会使优势比（乘法）改变一个系数。我们也可以这样解释：一个单位的改变（x\_j）会增加相应权重值的对数优势比。大多数人解释比值比是因为人们知道思考某个事物的对数（）对大脑来说很困难。解释赔率已经需要一些习惯。例如，如果概率为2，则表示y=1的概率是y=0的两倍。如果权重（=logods ratio）为0.7，则将相应的特征增加一个单位，将优势乘以exp（0.7）（约2），优势变为4。但通常你不处理赔率，只把权重解释为赔率。因为为了实际计算概率，您需要为每个特性设置一个值，这只有在您想查看数据集的一个特定实例时才有意义。

这些是对具有不同特征类型的逻辑回归模型的解释：

* 数字特征：如果将特征值（x\_j）增加一个单位，则估计的零散值将增加一个系数（exp（\beta\_j））。
* 二进制分类特征：特征的两个值之一是引用类别（在某些语言中，是用0编码的）。将特性\（x j\_ \）从引用类别更改为其他类别会将估计的概率更改为\（\exp（\beta j）系数）。
* 具有两个以上类别的分类功能：处理多个类别的一个解决方案是一个热编码，这意味着每个类别都有自己的列。对于具有L类别的分类功能，您只需要L-1列，否则它将被过度包装。第l类是参考类。您可以使用线性回归中使用的任何其他编码。每个类别的解释就相当于二进制特征的解释。
* 截距\（\beta 0 \）：当所有数值特征为零且分类特征位于参考类别之外时，估计的概率为\（\exp（\beta 0））。截距权重的解释通常不相关。

4.2.4示例

我们使用逻辑回归模型，根据一些风险因素进行预测。下表显示了估计权重、相关的优势比和估计的标准误差。

表4.1：在宫颈癌数据集上拟合逻辑回归模型的结果。所示为模型中使用的特征、它们的估计权重和相应的优势比，以及估计权重的标准误差。

## 加权赔率。误差

截距2.91 18.36 0.32

激素避孕药Y/N0.12 1.12 0.30

吸烟Y/N-0.26 0.77 0.37

怀孕次数-0.04 0.96 0.10

诊断的性病数量-0.82 0.44 0.33

宫内节育器Y/N-0.62 0.54 0.40

数字特征的解释（“诊断的性病数量”）：当所有其他特征保持不变时，诊断的性病（性传播疾病）数量的增加改变（减少）癌症与非癌症的几率增加0.44倍。记住，相关性并不意味着因果关系。这里没有获得性传播疾病的建议。

分类特征的解释（“激素避孕药Y/N”）：对于使用激素避孕药的妇女来说，与没有激素避孕药的妇女相比，癌症与没有癌症的几率高出1.12倍，因为所有其他特征保持不变。

就像线性模型一样，解释总是伴随着“所有其他特征保持不变”这一条款。

4.2.5优缺点

许多优点和缺点也适用于逻辑回归模型。

逻辑回归已被许多不同的人广泛使用，但它难以克服其限制性表达（例如，必须手动添加交互），其他模型可能具有更好的预测性能。

逻辑回归模型的另一个缺点是解释更困难，因为权重的解释是乘法而不是加法。

逻辑回归可能受到完全分离的影响。如果有一个特性能够将两个类完全分开，那么逻辑回归模型就不能再被训练了。这是因为该特性的权重不会收敛，因为最佳权重是无限的。这真的有点不幸，因为这样的功能真的很有用。但是如果你有一个简单的规则将两个类分开，你就不需要机器学习了。完全分离问题可以通过引入权的惩罚或定义权的先验概率分布来解决。

从好的方面来说，逻辑回归模型不仅是一个分类模型，而且给你概率。与只能提供最终分类的模型相比，这是一个很大的优势。知道一个实例对于一个类来说有99%的概率，而对于51%的类来说有很大的不同。

逻辑回归也可以从二元分类扩展到多类分类。然后它被称为多项式回归。

4.2.6软件

我在所有的例子中都使用了r中的glm函数。您可以在任何可用于执行数据分析的编程语言中找到逻辑回归，如Python、Java、StATA、Matlab、…

4.3 GLM、GAM等

最大的优点也是最大的缺点是预测被建模为特征的加权和。此外，线性模型还有许多其他假设。坏消息是（好吧，不是真的新闻）所有这些假设在现实中经常被违背：给定特征的结果可能具有非高斯分布，特征可能相互作用，特征和结果之间的关系可能是非线性的。好消息是，统计界已经开发了各种修改，将线性回归模型从一个简单的刀片转变为一把瑞士刀。

这一章绝对不是你扩展线性模型的明确指南。相反，它是一个扩展的概述，如广义线性模型（glms）和广义加法模型（gams），并给您一些直觉。阅读之后，您应该对如何扩展线性模型有一个坚实的概述。如果您想先了解更多关于线性回归模型的信息，我建议您阅读，如果您还没有阅读的话。

让我们记住线性回归模型的公式：

\[Y=\β0+β1 x 1+ldots+\βp x p+epsilon\]

线性回归模型假设一个实例的结果y可以用它的p特征的加权和表示，其中一个单独的误差（epsilon）遵循高斯分布。通过将数据强制到公式的这个corset中，我们获得了许多模型的可解释性。特征效应是加性的，意味着没有相互作用，并且关系是线性的，意味着一个单位的特征增加可以直接转化为预测结果的增加/减少。线性模型允许我们将特征和预期结果之间的关系压缩成一个单一的数字，即估计的权重。

但对于许多现实世界的预测问题来说，简单的加权和太过严格。在本章中，我们将学习经典线性回归模型的三个问题以及如何解决它们。可能违反的假设还有很多问题，但我们将重点关注下图所示的三个问题：

图4.8：线性模型的三个假设（左侧）：给定特征的结果的高斯分布、可加性（=无相互作用）和线性关系。现实通常不符合这些假设（右侧）：结果可能具有非高斯分布，特征可能相互作用，关系可能是非线性的。

所有这些问题都有解决方案：

问题：给定特征的目标结果y不遵循高斯分布。例如：假设我想预测我在给定的一天骑自行车的时间。作为特征，我有一天的类型，天气等等。如果我使用线性模型，它可以预测负分钟，因为它假设高斯分布，不会在0分钟停止。如果我想用线性模型预测概率，我可以得到负的或大于1的概率。

解决方案：

问题：功能相互作用。

一般来说，小雨对我骑车的欲望有轻微的负面影响。但是在夏天，在交通高峰期，我欢迎下雨，因为那时所有晴天的自行车手都呆在家里，我自己有自行车道！这是一种时间和天气之间的相互作用，不能由纯加法模型捕获。

解决方案：。

问题：特征和y之间的真实关系不是线性的。

例如：在0到25摄氏度之间，温度对我骑车欲望的影响可能是线性的，这意味着从0到1摄氏度的增加导致骑车欲望的增加与从20到21摄氏度的增加相同。但在较高的温度下，我骑车的动机会减弱甚至降低——我不喜欢在太热的时候骑车。解决：

本章介绍了这三个问题的解决方案。线性模型的许多进一步扩展被省略了。如果我试图在这里涵盖所有的内容，这一章很快就会变成一本书中的一本书，关于一个已经被许多其他书所涵盖的主题。但是既然您已经在这里了，我已经为线性模型扩展做了一个小问题和解决方案概述，您可以在中找到。解决方案的名称旨在作为搜索的起点。

4.3.1非高斯结果-GLMS

线性回归模型假定，给定输入特征的结果遵循高斯分布。这一假设排除了许多情况：结果也可以是一个类别（癌症对健康）、一个计数（儿童数量）、事件发生的时间（机器故障的时间）或具有一些非常高价值（家庭收入）的非常扭曲的结果。线性回归模型可以扩展为所有这些类型的结果的模型。这个扩展称为广义线性模型，简称GLMS。在本章中，我将对通用框架和该框架中的特定模型使用glm这个名称。任何GLM的核心概念都是：保留特征的加权和，但允许非高斯结果分布，并通过可能的非线性函数连接该分布的预期平均值和加权和。例如，logistic回归模型假设结果为伯努利分布，并使用logistic函数连接预期平均值和加权和。

GLM使用链接函数g将特征的加权和与假定分布的平均值进行数学链接，根据结果类型可以灵活选择。

\[g（e\_y（y\_x））=\beta\_+\beta\_x\_1+ldots \beta\_p x\_p \]

GLMS由三个部分组成：连接函数G、加权和（x^t\beta）（有时称为线性预测因子）和定义指数族的概率分布。

指数族是一组分布，可以用相同的（参数化的）公式编写，其中包括指数、分布的平均值和方差以及一些其他参数。我不会深入研究数学细节，因为这是一个非常大的宇宙，我不想进入它自己。维基百科有一个整洁的。可以为您的GLM选择此列表中的任何分发。根据要预测的结果类型，选择合适的分布。结果是否算数（例如，家庭中的孩子人数）？那么泊松分布可能是一个很好的选择。结果是否总是积极的（例如两个事件之间的时间）？那么指数分布可能是一个很好的选择。

让我们把经典线性模型看作是GLM的一个特例。经典线性模型中高斯分布的连接函数是一个简单的恒等函数。高斯分布由均值和方差参数进行参数化。平均值描述了我们平均期望的值，而方差则描述了该平均值周围值的变化程度。在线性模型中，连接函数将特征的加权和连接到高斯分布的平均值。

在GLM框架下，这个概念推广到任何分布（指数族）和任意链接函数。如果y是某物的计数，例如某一天某人喝的咖啡数量，我们可以用一个带有泊松分布和自然对数的glm作为链接函数来建模：

\[ln（e\_y（y\_x）=x ^ t \β\]

逻辑回归模型也是一个假定伯努利分布并使用逻辑函数作为链接函数的GLM。逻辑回归中使用的二项分布的平均值是y为1的概率。

\[X ^ T \β=ln \左（\frac E Y（Y X）1-E Y（Y X）\右）=ln \左（\frac P（Y=1 X）1-P（Y=1 X）\右）\]

如果我们把这个方程解成p（y=1）在一边，我们得到逻辑回归公式：

\[P（Y=1）=\frac 1 1+exp（-x ^ t \ beta）\]

指数族的每一个分布都有一个标准的链接函数，可以从分布中数学地推导出来。GLM框架使选择链接函数独立于分布成为可能。如何选择正确的链接功能？没有完美的食谱。你要考虑到关于你的目标分布的知识，同时也要考虑到理论上的考虑，以及模型与你的实际数据的匹配程度。对于某些分布，规范链接函数可能导致对该分布无效的值。在指数分布的情况下，正则连接函数是负的逆函数，这可能导致负的预测超出指数分布的范围。因为您可以选择任何链接函数，所以简单的解决方案是选择另一个尊重分布域的函数。

## 示例

我模拟了一个关于咖啡饮用行为的数据集，以强调对GLMS的需求。假设你收集了关于你每天喝咖啡行为的数据。如果你不喜欢咖啡，就假装是茶。除了杯子的数量，你还记录了你目前的压力水平，从1到10，你前一天晚上睡得有多好，从1到10，以及你那天是否必须工作。其目的是预测咖啡的数量，考虑到功能压力、睡眠和工作。我模拟了200天的数据。压力和睡眠在1到10之间被均匀地吸引，工作是/否被50/50的机会吸引（多么美好的生活！）.然后，每天从泊松分布中提取咖啡的数量，将强度（也是泊松分布的预期值）建模为睡眠、压力和工作特征的函数。你可以猜到这个故事的结局：“嘿，让我们用一个线性模型来建模这个数据……哦，它不起作用……让我们用泊松分布来尝试一个GLM……惊喜！现在它开始工作了！“。我希望对你来说我没有把这个故事搞得太糟。

让我们看看目标变量的分布，即给定日期的咖啡数量：

图4.9：200天每日咖啡数量的模拟分布。

在200天中的76天你根本没有喝咖啡，在最极端的一天你喝了7杯。让我们天真地使用一个线性模型来预测咖啡的数量，使用睡眠水平、压力水平和工作是/否作为特征。当我们错误地假设高斯分布时，会发生什么错误？错误的假设可能会使估计无效，尤其是权重的置信区间。更明显的问题是预测与真实结果的“允许”域不匹配，如下图所示。

图4.10：咖啡的预测数量取决于压力、睡眠和工作。线性模型预测负值。

线性模型没有意义，因为它预测咖啡的负数量。这个问题可以用广义线性模型来解决。我们可以改变连接函数和假定的分布。一种可能是保持高斯分布，并使用一个总是导致正预测的链接函数，例如对数链接（逆函数是exp函数）而不是恒等函数。更好的是：我们选择一个与数据生成过程相对应的分布和一个适当的链接函数。由于结果是一个计数，泊松分布是一个自然选择，同时对数作为链接函数。在这种情况下，数据甚至是由泊松分布生成的，因此泊松GLM是最佳选择。拟合的泊松GLM导致预测值的以下分布：

图4.11：咖啡的预测数量取决于压力、睡眠和工作。具有泊松假设和日志链接的GLM是此数据集的适当模型。

没有负数量的咖啡，现在看起来好多了。

## GLM权重的解释

假设分布和链接函数决定了如何解释估计的特征权重。在Coffee Count示例中，我使用了一个带有泊松分布和日志链接的GLM，这意味着特性和预期结果之间存在以下关系。

\[ln（e（\text咖啡费\\123\125;，\text \125\ \ \\125\\125\\睡眠 ；+\贝塔\文本工作x \文本工作\]

为了解释权重，我们颠倒了链接函数，这样我们就可以解释特征对预期结果的影响，而不是对预期结果的对数的影响。

\[e（\text咖啡费\\text，\text睡眠，\text工作）=exp（\beta+\beta \125\123; \\\\\ +\贝塔\文本工作x \文本工作）]

由于所有权重都在指数函数中，因此效果解释不是加性的，而是乘性的，因为exp（a+b）是exp（a）乘以exp（b）。解释的最后一个要素是玩具例子的实际重量。下表列出了估计的权重和exp（权重）以及95%的置信区间：

## 权重exp（权重）[2.5%，97.5%]

（截距）-0.16 0.85[0.54，1.32]应力0.12 1.12[1.07，1.18]睡眠-0.15 0.86[0.82，0.90]工作是0.80 2.23[1.72，2.93]

压力水平提高一点，咖啡的预期数量乘以系数1.12。

将睡眠质量提高一点，咖啡的预期数量乘以系数0.86。预计一个工作日的咖啡数量平均是休息日咖啡数量的2.23倍。总之，压力越大，睡眠越少，工作越多，咖啡消耗越多。

在本节中，您学习了一些关于当目标不遵循高斯分布时有用的广义线性模型的知识。接下来，我们将研究如何将两个特征之间的交互集成到线性回归模型中。

4.3.2相互作用

线性回归模型假设一个特征的影响是相同的，而不考虑其他特征的值（=无交互作用）。但数据中经常有交互作用。为了预测租房情况，温度和工作日之间可能存在相互作用。也许，当人们不得不工作时，温度对租用自行车的数量影响不大，因为不管发生什么，人们都会骑着租用的自行车去工作。在休息日，许多人为了享乐而骑车，但只有在天气足够暖和的时候。当涉及到自行车租赁时，你可能期望温度和工作日之间的相互作用。

我们如何才能得到包含交互作用的线性模型？在拟合线性模型之前，请在特征矩阵中添加一列，表示特征之间的交互，并像往常一样拟合模型。该解决方案在某种程度上很优雅，因为它不需要对线性模型进行任何更改，只需要数据中的其他列。在工作日和温度示例中，我们将添加一个新特性，该特性不包含工作日，否则它具有温度特性的值，假设工作日是参考类别。假设我们的数据如下：

## 工作温度

# Y 25

N 12

N 30

Y 5

线性模型使用的数据矩阵看起来略有不同。下表显示了如果我们不指定任何交互，为模型准备的数据是什么样子的。通常，这种转换是由任何统计软件自动执行的。

## 截取工作时间

1 1 1

1 0 0

1 0 30

# 1 1 1

第一列是截距项。第二列对分类功能进行编码，引用类别为0，其他类别为1。第三列包含温度。

如果我们希望线性模型考虑温度和工作日特征之间的相互作用，我们必须为相互作用添加一列：

## 截取工作临时温度

1 1 25 25个

# 1 0 12 0

1 0 30 0

1 1 5 5 5

新的列“worky.temp”捕获了功能工作日（Work）和温度（Temp）之间的交互。对于一个实例，如果工作特性处于引用类别（“n”表示无工作日），则此新特性列为零，否则它假定实例温度特性的值。使用这种编码方式，线性模型可以学习两种类型的日的温度的不同线性效应。这是两个功能之间的交互作用。在没有交互项的情况下，分类特征和数值特征的组合效应可以用一条垂直移动的线来描述。如果我们将相互作用包括在内，我们就允许数值特征（斜率）的影响在每个类别中具有不同的值。

两个分类特征的交互作用也有类似的效果。我们创建了表示类别组合的附加功能。以下是一些包含工作日（Work）和分类天气特征（WTHR）的人工数据：

## 工作时间

Y 2 N 0

N 1

Y 2

接下来，我们将包括交互术语：

## 截水工程，1号、2号、1号、2号

1 1 0 1 0 1

1 0 0 0 0 0 0

1 0 1 0 0 0

1 1 0 1 0 1

第一列用于估计截距。第二列是编码的工作特性。第三列和第四列用于天气功能，这需要两列，因为您需要两个权重来捕获三个类别的效果，其中一个类别是参考类别。其余的列捕获交互。对于这两个功能的每个类别（引用类别除外），如果两个功能都有一个特定的类别，我们将创建一个新的功能列，即1，否则为0。

对于两个数值特征，交互柱更容易构造：我们简单地将两个数值特征相乘。

有一些方法可以自动检测和添加交互术语。其中一个可以在中找到。规则拟合算法首先挖掘交互项，然后估计包含交互的线性回归模型。

## 例子

让我们回到我们已经在中建模的。这一次，我们还考虑了温度和工作日特征之间的相互作用。这将导致以下估计的权重和置信区间。

## 标准重量误差2.5%97.5%

|  |  |  |
| --- | --- | --- |
| （截距） | 2185.8 250.2条 | 1694.6 2677.1条 |
| 季节性夏季 | 893.8 121.8条 | 654.7 1132.9条 |
| 季节性下降 | 137.1 161.0条 | -179.0 453.2条 |
| 季节性冬季 | 426.5 110.3条 | 209.9 643.2条 |
| 假日 | -674.4 202.5条 | -1071.9-276.9页 |
| 工作日工作日 | 451.9 141.7条 | 173.7 730.1条 |
| 天气状况 | -382.1 87.2条 | -553.3-211.0 |
| 天气状况/ | -1898.2222.7年 | -2335.4-1461.0页 |
| 临时雇员 | 125.4 8.9条 | 108.0 142.9页 |
| 嗡嗡声 | -17.5 3.2 | -23.7至11.3 |
| 风速 | -42.1 6.9条 | -55.5至28.6 |
| 自2011年以来的天数 | 4.9 0.2条 | 4.6第5.3条 |
| 工作日工作日：临时-21.8 8.1 | | -37.7-5.9条 |

额外的相互作用效应为负（-21.8），与零显著不同，如95%置信区间所示，不包括零。顺便说一句，数据不是IID，因为彼此接近的日子并不相互独立。置信区间可能会产生误导，就拿它来说吧。交互项改变了相关特征权重的解释。如果是工作日，温度会有负面影响吗？答案是否定的，即使表格建议未经培训的用户使用。我们不能孤立地解释“工作日工作日：温度”交互权重，因为这样的解释是：“在保持所有其他特征值不变的情况下，增加工作日温度的交互作用会减少预测的自行车数量。”但是，在牵引效应只会增加温度的主要影响。假设这是一个工作日，我们想知道如果今天的温度升高1度会发生什么。然后我们需要将“temp”和“workingdayworking day:temp”的权重相加，以确定估计值增加了多少。

从视觉上更容易理解交互。通过引入分类特征和数值特征之间的相互作用项，我们得到了温度的两个斜率，而不是一个。人们不需要工作的天数（“无工作日”）的温度斜率可直接从表（125.4）中读取。人们必须工作的天数的温度斜率（“工作日”）是两个温度权重的总和（125.4-21.8=103.6）。“非工作日”线在温度=0时的截距由线性模型（2185.8）的截距项确定。“工作日”线在温度=0时的截距由截距项+工作日的影响（2185.8+451.9=2637.7）确定。

图4.12：温度和工作日对线性模型预测自行车数量的影响（包括相互作用）。实际上，我们得到了两个温度的斜率，一个对应于工作日特征的每个类别。

4.3.3非线性效应-GAMS

世界不是线性的。线性模型中的线性意味着，无论一个实例在某个特定特征中具有什么值，将该值增加一个单位对预测结果总是有相同的影响。假设温度在10摄氏度时升高1度，对租用自行车的数量产生的影响是否与温度在40摄氏度时升高相同？直观地说，人们期望把温度从10摄氏度提高到11摄氏度会对自行车租赁产生积极影响，从40摄氏度提高到41摄氏度会产生消极影响，正如你在书中的许多例子中看到的那样。温度特性对租用自行车的数量有线性的、积极的影响，但在某些时候它会变平，甚至在高温下也会产生负面影响。线性模型并不关心，它会尽职地找到最佳的线性平面（通过最小化欧几里得距离）。

可以使用以下方法之一对非线性关系建模：

* 特征的简单转换（例如对数）
* 功能分类
* 广义加法模型

在详细介绍每种方法之前，让我们先从一个示例开始，说明这三种方法。我使用并训练了一个仅具有温度特性的线性模型来预测出租自行车的数量。下图显示了估算的斜率：标准线性模型、转换温度的线性模型（对数）、将温度视为分类特征并使用回归样条（GAM）的线性模型。

图4.13：仅使用温度特征预测租用自行车的数量。线性模型（左上角）不适合数据。一种解决方案是用对数（右上）、分类（左下）来转换特征，这通常是一个错误的决定，或者使用可自动拟合平滑温度曲线（右下）的广义加法模型。

## 特征转换

通常将特征的对数用作转换。使用对数表示，每10倍的温度升高对自行车数量有相同的线性影响，因此从1摄氏度到10摄氏度的变化与从0.1到1（听起来错误）的变化具有相同的效果。特征变换的其他例子有平方根、平方函数和指数函数。使用特征转换意味着用特征的函数（如对数）替换数据中该特征的列，并像往常一样拟合线性模型。一些统计程序还允许您在线性模型调用中指定转换。当您转换功能时，您可以发挥创造力。特征的解释会根据所选转换而变化。如果使用对数转换，线性模型中的解释将变为：“如果特征的对数增加1，则预测将增加相应的权重。”当使用具有非同一性函数的链接函数的GLM时，则解释将得到更复杂的是，因为您必须将这两种转换合并到解释中（除非它们相互抵消，如log和exp，否则解释会更容易）。

## 功能分类

实现非线性效果的另一种可能性是离散化特征，将其转化为分类特征。例如，您可以将温度特性切割为20个间隔，级别为[-10，-5），

[-5，0），…等等。当使用分类温度而不是连续温度时，线性模型将估计阶跃函数，因为每个级别都有自己的估计。这种方法的问题在于，它需要更多的数据，更容易过拟合，而且不清楚如何有意义地离散特征（等距间隔或分位数）？有多少个间隔？）只有在有很强的理由的情况下，我才会使用离散化。例如，使模型与另一项研究相比较。

## 广义加法模型

为什么不“简单”地允许（广义）线性模型学习非线性关系呢？这就是游戏背后的动机。GAM放宽了这种限制，即关系必须是一个简单的加权和，而假定结果可以由每个特征的任意函数的和建模。在数学上，gam中的关系如下：

\[g（e\_y（y\_x））=\beta\_+f\_1（x\_1）+f\_2（x\_2）+\ldots+f\_p（x\_p）\]

该公式与GLM公式类似，不同之处在于线性项\（\beta\_j\_x\_j）被更灵活的函数\（f\_j（x\_j））取代。gam的核心仍然是特征效果的总和，但是您可以选择允许某些特征和输出之间存在非线性关系。线性效应也包含在框架中，因为对于要线性处理的特性，您可以将它们的形式限制为（x\_j（x\_j））仅采用（x\_j\_β\_j\）。

最大的问题是如何学习非线性函数。答案称为“样条函数”或“样条函数”。样条函数是可以组合起来以近似任意函数的函数。有点像堆放乐高积木来建造更复杂的东西。定义这些样条函数的方法有很多种，令人困惑。如果您有兴趣了解更多关于定义样条曲线的所有方法，我希望您在旅途中好运。我不想在这里讨论细节，我只想建立一个直觉。对于理解样条曲线，我个人最大的帮助是可视化单个样条函数，并研究如何修改数据矩阵。例如，为了用样条曲线对温度进行建模，我们从数据中删除了温度特性，并将其替换为4列，每列表示一个样条曲线函数。通常你会有更多的样条函数，我只是为了说明的目的减少了数量。这些新样条线特征的每个实例的值取决于实例的温度值。加上所有的线性效应，GAM还可以估计这些样条权重。gams还为权重引入了一个惩罚术语，以使它们接近零。这有效地降低了花键的灵活性，并减少了过拟合。然后通过交叉验证来调整通常用于控制曲线灵活性的平滑度参数。忽略惩罚项，样条曲线非线性建模是一种奇特的特征工程。

在我们仅使用温度预测GAM自行车数量的示例中，模型特征矩阵如下所示：

## （截距）s（温度）.1s（温度）.2s（温度）.3s（温度）.4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1个 | 0.93分 | -0.14分 | 0.21分 | -0.83分 |
| 1个 | 0.83分 | -0.27分 | 0.27分 | -0.72分 |
| 1个 | 1.32条 | 0.71分 | -0.39分 | -1.63条 |
| 1个 | 1.32条 | 0.70分 | -0.38分 | -1.61条 |
| 1个 | 1.29条 | 0.58分 | -0.26分 | -1.47条 |
| 1个 | 1.32条 | 0.68分 | -0.36分 | -1.59条 |

每一行表示数据中的单个实例（一天）。每个样条线列包含特定温度值下样条线函数的值。下图显示了这些样条函数的外观：

图4.14：为了平滑地模拟温度效应，我们使用4个样条函数。每个温度值映射到（此处）4个样条曲线值。如果实例的温度为30°C，则第一个样条曲线特征的值为-1，第二个为0.7，第三个为-0.8，第四个为1.7。

GAM为每个温度样条线功能指定权重：

重量（截距）4504.35 s（温度）1-989.34 s（温度）2 740.08 s（温度）3 2309.84 s（温度）4 558.27

而实际曲线是由用估计权重加权的样条函数之和得到的，如下所示：

图4.15：用于预测租用自行车数量的温度的gam特征效应（温度是唯一的特征）。

平滑效果的解释需要对拟合曲线进行目视检查。样条曲线通常以均值预测为中心，因此曲线上的一个点与均值预测是不同的。例如，在0摄氏度时，预测的自行车数量比平均预测值低3000辆。

4.3.4优势

线性模型的所有这些扩展本身就是一个宇宙。无论您面对的线性模型问题是什么，您可能会找到一个扩展来修复它。

大多数方法已经使用了几十年。例如，GAM已经有将近30年的历史了。许多来自工业界的研究人员和实践者对线性模型非常有经验，并且这些方法在许多社区被认为是建模的现状。

除了进行预测之外，您还可以使用模型进行推理，得出关于数据的结论——假设模型假设没有违反。你可以得到权重的置信区间，显著性检验，预测区间等等。

统计软件通常有很好的接口来适应GLMS、GAMS和更特殊的线性模型。

许多机器学习模型的不透明性源于1）缺乏稀疏性，这意味着使用了许多特性；2）以非线性方式处理的特性，这意味着您需要一个以上的权重来描述效果；3）特性之间交互的建模锿。假设线性模型具有很高的可解释性，但通常是在不真实的情况下，本章所述的扩展提供了一种很好的方法，以实现平稳过渡到更灵活的模型，同时保留一些可解释性。

4.3.5缺点

作为优势，我说过线性模型生活在自己的宇宙中。扩展简单线性模型的方法太多了，不仅仅是初学者。事实上，有多个平行的宇宙，因为许多研究者和实践者的社区都有他们自己的名字来命名那些或多或少做相同事情的方法，这可能非常令人困惑。

线性模型的大多数修改都会使模型的解释性变差。非同一性函数的任何链接函数（在GLM中）都会使解释复杂化；交互也会使解释复杂化；非线性特征效应要么不太直观（如对数转换），要么不能再由单个数字（如样条函数）来概括。

GLMS、GAMS等依赖于有关数据生成过程的假设。如果违反了这些规定，则对权重的解释将不再有效。

在许多情况下，随机森林或梯度树增强等基于树的集合的性能比最复杂的线性模型要好。这部分是我自己的经验，也部分是来自kaggle.com等平台上的获奖模型的观察结果。

4.3.6软件

本章中的所有示例都是使用R语言创建的。对于gam，使用了gam包，但还有许多其他包。R有大量的包来扩展线性回归模型。R是任何其他分析语言都无法超越的，它是线性回归模型扩展的每一个可想象的扩展的所在。您将发现在python中实现gams（例如，但这些实现还不够成熟）。

4.3.7进一步扩展

如前所述，这里列出了您可能遇到的线性模型问题，以及此问题的解决方案名称，您可以将其复制并粘贴到您最喜爱的搜索引擎中。

我的数据违反了独立和相同分布（IID）的假设。

例如，对同一患者重复测量。

搜索混合模型或广义估计方程。

我的模型有异方差错误。

例如，在预测房屋价值时，昂贵房屋的模型误差通常较高，这违背了线性模型的同方差性。寻找稳健回归。

我有强烈影响我的模型的异常值。寻找稳健回归。

我想预测事件发生前的时间。

事件到事件的数据通常带有经过审查的度量，这意味着在某些情况下没有足够的时间来观察事件。例如，一家公司想预测其制冰机的故障，但只有两年的数据。一些机器两年后仍然完好无损，但可能会在以后失效。

搜索参数生存模型，Cox回归，生存分析。

我要预测的结果是一个类别。

如果结果有两个类别，请使用A，它为类别建模概率。

如果你有更多的类别，搜索多项式回归。逻辑回归和多项式回归都是GLMS。

我想预测有序的类别。

例如，学校成绩。

搜索比例优势模型。

我的结果是一个计数（就像一个家庭中孩子的数量）。

寻找泊松回归。

泊松模型也是一个GLM。您可能还遇到一个问题，即0的计数值非常频繁。

寻找零膨胀泊松回归，跨栏模型。

我不确定模型中需要包含哪些特征才能得出正确的因果结论。例如，我想知道药物对血压的影响。这种药物对某些血量有直接影响，这种血量会影响结果。我应该将血量纳入回归模型吗？

寻找因果推理，调解分析。

我缺少数据。

寻找多重插补。

我想把先前的知识整合到我的模型中。寻找贝叶斯推理。

我最近有点情绪低落。

搜索“亚马逊Alexa Gone Wild！！！！从头到尾的完整版本”。

4.4决策树

线性回归和逻辑回归模型在特征与结果之间的关系为非线性或特征相互作用的情况下失败。是时候照亮决策树了！基于树的模型根据特征中的某些截止值多次分割数据。通过拆分，可以创建数据集的不同子集，每个实例都属于一个子集。最后的子集称为终端或叶节点，中间的子集称为内部节点或拆分节点。为了预测每个叶节点的结果，使用该节点中训练数据的平均结果。树木可用于分类和回归。

有多种算法可以生长一棵树。它们在树的可能结构（例如每个节点的拆分数量）、如何查找拆分的标准、何时停止拆分以及如何估计叶节点内的简单模型等方面存在差异。分类和回归树（cart）算法可能是最流行的树归纳算法。我们将关注Cart，但对于大多数其他树类型来说，解释是类似的。我推荐《统计学习要素》（Friedman、Hastie和Tibshirani，2009年）一书来更详细地介绍Cart。

图4.16：带有人工数据的决策树。特性x1的值大于3的实例将在节点5中结束。所有其他实例都分配给节点3或节点4，这取决于功能x2的值是否超过1。

下面的公式描述了结果Y和特征X之间的关系。

\[\hat y=\hat f（x）=\sum m=1 ^mc i x\in r \]

每个实例正好属于一个叶节点（=subset\（r\_m\）。\（i x）是标识函数，如果\（x\）在子集\（r\_m\）中，如果一个实例属于叶节点（r\_l），则预测结果为\（\hat y=c\_l\），其中\（c\_l\）是叶节点（r\_l\）中所有训练实例的平均值。

但是子集是从哪里来的呢？这很简单：Cart采用一个特性，并确定回归任务的哪个临界点将y的方差最小化，或者分类任务的y类分布的基尼指数最小化。方差告诉我们节点中的y值围绕其平均值分布的程度。基尼指数告诉我们一个节点是如何“不纯”的，例如，如果所有的类都有相同的频率，那么这个节点就是不纯的，如果只有一个类存在，那么它就是最大的纯的。当节点中的数据点具有非常相似的y值时，方差和基尼指数最小化。因此，最佳截止点使两个结果子集在目标结果方面尽可能不同。对于分类特征，该算法尝试通过尝试不同类别分组来创建子集。在确定每个特征的最佳截止值后，算法选择将导致方差或基尼指数最佳分割的特征，并将此分割添加到树中。该算法继续此搜索并在两个新节点中递归拆分，直到达到停止条件。可能的条件是：拆分前必须在节点中的最小实例数，或必须在终端节点中的最小实例数。

4.4.1解释

解释很简单：从根节点开始，转到下一个节点，边缘告诉您要查看的子集。一旦到达叶节点，该节点将告诉您预测的结果。所有边缘通过“和”连接。

模板：如果特征x比阈值c小/大，那么预测结果就是该节点中实例y的平均值。

## 功能重要性

在决策树中，一个特征的总体重要性可以用以下方法计算：通过使用该特征的所有拆分，并测量它相对于父节点减少了多少方差或基尼指数。所有进口货物之和按比例计算为100。这意味着每个重要性都可以被解释为在整个模型重要性中所占的份额。

## 树木分解

通过将决策路径分解为每个特征的一个组件，可以解释决策树的单个预测。我们可以通过树跟踪决策，并通过在每个决策节点上添加的贡献来解释预测。

决策树中的根节点是我们的起点。如果我们使用根节点进行预测，它将预测训练数据结果的平均值。根据路径中的下一个节点，在下一个拆分中，我们要么减去这个和，要么添加一个项。为了得到最终的预测，我们必须遵循要解释的数据实例的路径，并不断地添加到公式中。

\[\hat f（x）=\bar y+\sum d=1 ^d\text split.contrib（d，x）=\bar y+sum j=1 ^p\text feat.contrib（j，x）\]

单个实例的预测是目标结果的平均值加上在根节点和实例结束的终端节点之间发生的D拆分的所有贡献的总和。不过，我们对分割贡献不感兴趣，而是对特性贡献感兴趣。一个特性可能用于多个拆分，或者根本不用于。我们可以为每个P特性添加贡献，并获得每个特性对预测贡献多少的解释。

4.4.2示例

让我们再看看。我们想用决策树来预测某一天租用自行车的数量。所学的树如下所示：

图4.17：自行车租赁数据上的回归树。树的最大允许深度设置为2。已为分割选择趋势特征（自2011年以来的天数）和温度（温度）。箱线图显示了终端节点中自行车数量的分布。

第一个分割和第二个分割中的一个是使用趋势功能执行的，该功能统计数据收集开始后的天数，并涵盖自行车租赁服务随着时间的推移变得越来越流行的趋势。在第105天之前的几天，自行车的预测数量大约是1800辆，在第106天和第430天之间，大约是3900辆。对于第430天之后的几天，预测值为4600（如果温度低于12度）或6600（如果温度高于12度）。

特性的重要性告诉我们特性在多大程度上有助于提高所有节点的纯度。这里使用了方差，因为预测自行车租赁是一个回归任务。

可视化树表明温度和时间趋势都被用于分割，但没有量化哪个特征更重要。特征重要性测度表明时间趋势远比温度重要。

图4.18：通过平均提高节点纯度来衡量特征的重要性。

4.4.3优势

树结构非常适合捕获数据中特征之间的交互。

这些数据以不同的组结束，这些组通常比多维超平面上的点更容易理解，就像线性回归一样。可以说，解释相当简单。

树结构也有一个自然的可视化，其节点和边缘。

树可以产生良好的解释，如中所定义。树结构自动邀请将单个实例的预测值视为反事实：“如果某个特征大于/小于分割点，则预测值将是y1而不是y2。树的解释具有对比性，因为您总是可以将实例的预测与相关的“假设”场景（由树定义）进行比较，这些场景只是树的其他叶节点。如果这棵树很短，像一到三个深的裂口，那么所得到的解释是有选择性的。深度为三的树最多需要三个特征和拆分点来为单个实例的预测创建解释。预测的真实性取决于树的预测性能。对于短树的解释非常简单和一般，因为对于每一个拆分，实例都属于一个或另一个叶子，并且二进制决策很容易理解。

不需要转换功能。在线性模型中，有时需要取特征的对数。决策树同样适用于特征的任何单调变换。

4.4.4缺点

树不能处理线性关系。输入特征和结果之间的任何线性关系都必须通过分割来近似，从而创建一个阶跃函数。这是不有效的。

这与缺乏平滑度是紧密相连的。输入特性的微小变化会对预测结果产生很大影响，这通常是不可取的。想象一棵树预测房屋的价值，而这棵树使用房屋的大小作为分割特征之一。分裂发生在100.5平方米。想象一下，使用你的决策树模型的房价估计器的用户：他们测量他们的房子，得出房子有99平方米的结论，将其输入价格计算器，得到20万欧元的预测。用户注意到他们忘记了测量一个2平方米的小储藏室。储藏室有一道倾斜的墙，所以他们不确定是否能计算出所有的面积，或者只计算其中的一半。所以他们决定尝试100.0和101.0平方米。结果：价格计算器输出了20万欧元和20.5万欧元，这是相当不合理的，因为从99平方米到100没有变化。

树木也相当不稳定。训练数据集中的一些更改可以创建完全不同的树。这是因为每个拆分都依赖于父拆分。如果选择其他特征作为第一个拆分特征，则整个树结构将发生更改。如果结构如此容易更改，它不会在模型中创建信心。

决策树很容易理解——只要它们很短。终端节点的数量随着深度的增加而迅速增加。终端节点越多，树越深，就越难理解树的决策规则。深度为1表示2个终端节点。深度2表示最多4个节点。深度3表示最多8个节点。树中终端节点的最大数量是深度的2倍。

4.4.5软件

对于本章中的示例，我使用了实现cart（分类和回归树）的rpart r包。Cart是用许多编程语言实现的，包括有争议的，Cart是一个相当古老的，有些过时的算法，还有一些有趣的新算法来拟合树。您可以在关键字“递归分区”下找到决策树的一些R包的概述。

1。弗里德曼、杰罗姆、特雷弗·黑斯迪和罗伯特·提比西拉尼。“统计学习要素”。www.web.stanford.edu/~hassie/elemstatlearn/（2009年）。

4.5决策规则

决策规则是一个简单的if-then语句，由条件（也称为先行条件）和预测组成。例如：如果今天下雨，如果是四月（条件），那么明天下雨（预测）。可以使用单个决策规则或多个规则的组合进行预测。

决策规则遵循一个一般的结构：如果条件满足，那么做一个特定的预测。决策规则可能是最容易解释的预测模型。它们的if-then结构在语义上类似于自然语言和我们的思维方式，前提是条件是由可理解的特征构建的，条件的长度很短（少数特征=与and组合的值对），并且没有太多规则。在编程中，编写if-then规则是很自然的。机器学习中的新功能是通过算法学习决策规则。

想象一下，使用一种算法来学习预测房屋价值的决策规则（低、中或高）。这个模型学习到的一个决策规则可能是：如果一个房子超过100平方米，并且有一个花园，那么它的价值就很高。更正式一点：如果尺寸大于100且花园=1，则值=高。

让我们打破决策规则：

* 尺寸>100是if部分的第一个条件。
* garden=1是if部分的第二个条件。
* 这两个条件通过“和”连接以创建新条件。这两个规则都必须适用。
* 预测结果（然后部分）值=高。

决策规则在条件中至少使用一个feature=value语句，对于“and”可以添加的数量没有上限。一个例外是没有显式ifpart的默认规则，当没有其他规则应用时该规则也适用，但稍后将对此进行详细说明。

决策规则的有用性通常概括为两个数字：支持和准确性。

规则的支持或覆盖率：应用规则条件的实例的百分比

被称为支持。例如，规则大小=大，位置=好，然后值=高，用于预测房屋价值。假设1000间房子中有100间大而位置好，那么规则的支持率是10%。预测（然后是部分）对支护计算不重要。

规则的准确性或可信度：规则的准确性是衡量规则在预测规则条件适用的实例的正确类时的准确性的指标。例如：假设100个房屋中，规则大小=大，位置=好，则值=高，85个房屋中的值=高，14个房屋中的值=中，1个房屋中的值=低，则规则的准确度为85%。

通常在精度和支持之间有一个权衡：通过在条件中添加更多的特性，我们可以获得更高的精度，但会失去支持。

为了创建一个好的分类器来预测房子的价值，你可能不仅需要学习一条规则，而且可能需要学习10或20条规则。然后事情会变得更复杂，你会遇到以下问题之一：

* 规则可以重叠：如果我想预测一栋房子的价值和两个或两个以上的规则，它们会给我矛盾的预测呢？
* 没有适用的规则：如果我想预测一所房子的价值，但没有适用的规则怎么办？

组合多个规则有两种主要策略：决策列表（有序）和决策集（无序）。这两种策略都暗示了对重叠规则问题的不同解决方案。

决策列表向决策规则引入一个顺序。例如，如果第一条规则的条件为真，则使用第一条规则的预测。如果没有，我们将转到下一个规则，检查它是否适用，依此类推。决策列表通过只返回应用列表中第一个规则的预测来解决重叠规则的问题。

决策集类似于规则的民主，只是有些规则可能具有更高的投票权。在一套规则中，这些规则要么相互排斥，要么有解决冲突的策略，例如多数投票，这些策略可以通过个别规则的准确性或其他质量措施进行加权。当一些规则适用时，解释性可能会受到影响。

决策列表和决策集都可能面临没有规则适用于实例的问题。这可以通过引入默认规则来解决。默认规则是在没有其他规则适用时应用的规则。默认规则的预测通常是数据点中最频繁的一类，而其他规则不包括这些数据点。如果一组规则或一系列规则覆盖了整个特性空间，我们称之为详尽的。通过添加默认规则，集合或列表将自动变得详尽。

从数据中学习规则的方法有很多，本书远没有涵盖所有这些方法。这一章向你展示了其中三个。选择这些算法是为了涵盖广泛的学习规则的一般思想，因此这三种算法代表了非常不同的方法。

1. ONER从一个功能学习规则。ONER的特点是其简单性、可解释性和作为基准的使用。
2. 顺序覆盖是一个迭代学习规则并删除新规则所覆盖的数据点的一般过程。这个过程被许多规则学习算法使用。
3. 贝叶斯规则列表使用贝叶斯统计将预先挖掘的频繁模式组合成决策列表。使用预挖掘模式是许多规则学习算法常用的方法。

让我们从最简单的方法开始：使用单个最佳特性来学习规则。

4.5.1从单个功能（ONER）学习规则

Holte（1993）提出的ONER算法是最简单的规则归纳算法之一。从所有特性中，ONER选择一个包含有关感兴趣结果的最多信息的特性，并从该特性创建决策规则。

尽管命名为oner，代表“一条规则”，但是算法生成了不止一条规则：它实际上是所选最佳特征的每个唯一特征值的一条规则。一个更好的名字应该是一个特征。

算法简单快速：

1. 通过选择适当的间隔来离散连续特征。
2. 对于每个功能：

在特性值和（分类）结果之间创建交叉表。

对于功能的每个值，创建一个规则，预测具有此特定功能值（可以从交叉表中读取）的实例的最频繁类。计算功能规则的总错误。

1. 选择总误差最小的特征。

ONER始终覆盖数据集的所有实例，因为它使用所选功能的所有级别。缺失值可以作为附加特征值处理，也可以预先输入。

ONER模型是一个只有一个拆分的决策树。拆分不一定像购物车中那样是二进制的，但取决于唯一功能值的数量。

让我们来看一个例子，ONER是如何选择最佳特性的。下表显示了有关房屋的人工数据集，其中包含有关房屋价值、位置、大小以及是否允许养宠物的信息。我们有兴趣学习一个简单的模型来预测房子的价值。

|  |  |
| --- | --- |
| 位置大小宠物 | 价值 |
| 好的小的是的 | 高的 |
| 好大不 | 高的 |
| 好大不 | 高的 |
| 不好的介质 | 中等的 |

好的中草药好的小草药

|  |  |  |
| --- | --- | --- |
| 坏的 | 中号是的 | 中等的 |
| 坏的 | 小是的 | 低的 |
| 坏的 | 中号是的 | 低的 |
| 坏的 | 小号不 | 低的 |

ONER在每个功能和结果之间创建交叉表：

## 值=低值=中值=高

位置=坏3 2 0位置=好0 2 3

## 值=低值=中值=高

大小=大0 2大小=中1 3 0大小=小2 1

## 值=低值=中值=高

宠物=否1 2宠物=仅2类0宠物=是2 1 1

对于每一个特性，我们一行一行地遍历表：每个特性值都是规则的if部分；对于具有这个特性值的实例，最常见的类是预测，然后是规则的一部分。例如，大小特征与级别小，中，大三个结果规则。对于每个特性，我们计算生成的规则的总错误率，即错误的总和。位置功能具有可能的值bad和good。不良地段房屋最常见的价值是低值，当我们用低值作为预测时，我们会犯两个错误，因为两个房屋的价值是中等的。好地段的房屋预测值很高，我们又犯了两个错误，因为两个房屋的价值中等。我们使用位置特征的误差是4/10，对于尺寸特征是3/10，对于宠物特征是4/10。“大小”功能生成误差最小的规则，并将用于最终的ONER模型：

如果大小=小，则值=小

如果大小=中等，则值=中等

如果大小=大，则值=高

ONER喜欢具有许多可能级别的功能，因为这些功能可以更容易地覆盖目标。设想一个数据集，它只包含噪声，不包含信号，这意味着所有特性都具有随机值，并且对目标没有预测值。有些功能的级别比其他功能高。具有更多级别的功能现在可以更容易地过拟合。对于数据中的每个实例，具有单独级别的功能将完美地预测整个培训数据集。一种解决方案是将数据拆分为训练集和验证集，学习训练数据的规则，并评估在验证集上选择功能时的总错误。

领带是另一个问题，即当两个功能导致相同的总错误时。ONER通过使用具有最小误差的第一个特征或具有卡方检验的最小p值的特征来解决关系。

## 例子

让我们用真实的数据来尝试一下ONER。我们使用来测试ONER算法。将所有连续输入特征离散为5个分位数。将创建以下规则：

## 年龄预测

（12.9,27.2]健康

（27.2,41.4]健康

（41.4,55.6]健康

（55.6,69.8]健康

（69.8,84.1]健康

ONER选择年龄特征作为最佳预测特征。因为癌症是罕见的，所以对于每一个统治阶级来说，预测的标签总是健康的，这是相当没有帮助的。在这种不平衡的情况下使用标签预测是没有意义的。“年龄”间隔与癌症/健康之间的交叉表以及患癌症的妇女的百分比更能提供信息：

## #癌症健康型（癌症）

|  |  |  |
| --- | --- | --- |
| 年龄=（12.9,27.2]26 | 477个 | 0.05分 |
| 年龄=（27.2,41.4）25 | 290个 | 0.08分 |
| 年龄=（41.4,55.6）4 | 31岁 | 0.11分 |
| 年龄=（55.6,69.8）0 | 1个 | 0.00美元 |
| 年龄=（69.8,84.1]0 | 4 | 0.00美元 |

但是在您开始解释任何东西之前：因为每个特性和每个值的预测都是健康的，所以所有特性的总错误率都是相同的。默认情况下，通过使用错误率最低的功能中的第一个功能（这里，所有功能都有55/858）来解决总错误中的关系，这恰好是年龄功能。

ONER不支持回归任务。但是我们可以通过将连续结果分割成区间，将回归任务转化为分类任务。我们使用这个技巧通过将自行车的数量减少到四个四分位数（0-25%、25-50%、50-75%和75-100%）来预测ONER的数量。下表显示了安装ONER模型后所选的功能：

MNTHPrediction一月[223152]二月[223152]三月[223152]

4月（31524548）

5月（59568714）6月（45485956）

七月（59568714）八月（59568714）九月（59568714）

OKT（59568714）11月（31524548）

德兹[223152]

所选功能是月份。月功能有（惊喜！）12个功能级别，比大多数其他功能都要高。因此存在过度安装的危险。更乐观的一点是：月份特性可以处理季节性趋势（例如冬天租的自行车更少），而且预测似乎是明智的。

现在，我们从简单的ONER算法转移到一个更复杂的过程，使用由几个特征组成的条件更复杂的规则：顺序覆盖。

4.5.2顺序覆盖

顺序覆盖是一个通用过程，它反复学习单个规则以创建一个决策列表（或集合），该决策列表（或集合）按规则覆盖整个数据集规则。许多规则学习算法是顺序覆盖算法的变体。本章介绍了主要的配方和使用开膛手，一个变种的顺序覆盖算法为例。

这个想法很简单：首先，找到一个适用于某些数据点的好规则。删除规则所涵盖的所有数据点。当条件适用时，不管这些点是否正确分类，都会覆盖数据点。重复规则学习和删除包含的点和剩余的点，直到不再留下点或满足另一个停止条件。结果是一个决策列表。这种重复规则学习和删除覆盖数据点的方法称为“分离和征服”。

假设我们已经有了一个算法，可以创建一个覆盖部分数据的规则。两个类（一个正类，一个负类）的顺序覆盖算法的工作原理如下：

* 从一个空的规则列表（rlist）开始。
* 学习规则R。
* 当规则列表低于某个质量阈值（或正面示例未被覆盖）时：

将规则R添加到rlist。

删除规则R涵盖的所有数据点。

学习其他有关剩余数据的规则。

* 返回决策列表。

图4.19：覆盖算法的工作原理是用单个规则依次覆盖特征空间，并删除那些规则已经覆盖的数据点。为了实现可视化，特性x1和x2是连续的，但是大多数规则学习算法都需要分类特性。

例如：我们有一个任务和数据集来预测房屋的大小、位置以及是否允许养宠物。我们学习了第一条规则，结果是：如果大小=大，位置=好，那么值=高。然后我们从数据集中删除所有位于良好位置的大房子。利用剩下的数据，我们学习下一个规则。可能：如果位置=好，则值=中。请注意，这条规则是在数据上学习的，在好的位置没有大房子，只留下中小型房子在好的位置。

对于多类设置，必须修改方法。首先，这些课程的顺序是随着流行率的增加。顺序覆盖算法从最小公共类开始，为其学习规则，删除所有覆盖的实例，然后转到第二个最小公共类，依此类推。当前班级始终被视为阳性班级，所有患病率较高的班级都被合并为阴性班级。最后一个类是默认规则。这在分类中也被称为一对一策略。

我们如何学习一条规则？ONER算法在这里是无用的，因为它总是覆盖整个特征空间。但是还有很多其他的可能性。一种可能是通过梁搜索从决策树中学习单个规则：

* 学习决策树（使用购物车或其他树学习算法）。
* 从根节点开始，递归地选择最纯的节点（例如，具有较低的错误分类率）。
* 终端节点的大多数类用作规则预测；通向该节点的路径用作规则条件。

下图说明了树中的波束搜索：

图4.20：通过搜索决策树的路径来学习规则。发展决策树来预测感兴趣的目标。我们从根节点开始，贪婪地迭代地遵循本地生成最纯子集（例如，最高精度）的路径，并将所有拆分值添加到规则条件中。结果是：如果位置=好，大小=大，那么值=高。

学习单个规则是一个搜索问题，其中搜索空间是所有可能规则的空间。搜索的目标是根据某些条件找到最佳规则。有许多不同的搜索策略：爬山搜索，波束搜索，穷尽搜索，最佳优先搜索，有序搜索，随机搜索，自上而下搜索，自下而上搜索，…

Cohen（1995）的Ripper（重复增量修剪以减少误差）是顺序覆盖算法的一个变种。Ripper有点复杂，它使用后处理阶段（规则修剪）来优化决策列表（或集合）。裂土器可以在有序或无序模式下运行，并生成决策列表或决策集。

## 示例

我们将使用ripper作为例子。

裂土器算法在的分类任务中找不到任何规则。

当我们在回归任务中使用ripper来预测一些规则时，我们发现了。由于裂土器只能用于分类，自行车计数必须转化为分类结果。我通过把自行车的数量减少到四分之一来达到这个目的。例如（45485956）是覆盖4548和5956之间预测自行车数量的时间间隔。下表显示了已学习规则的决策列表。

## 规则

（自2011年以来的天数>=438）和（温度>=17）以及（温度<=27）和（hum<=67）=>cnt=（59568714]

（自2011年以来的天数>=443）和（温度>=12）以及（天气状况=良好）和（湿度>=59）=>cnt=（59568714]

（自2011年以来的天数>=441）和（风速<=10）以及（温度==13）=>cnt=（59568714]

（temp>=12）和（hum<=68）以及（自\_2011年以来的天数>=551）=>cnt=（59568714]

（自2011年以来的天数>=100）和（自2011年以来的天数<=434）以及（hum<=72）和（工作日=

工作日）=>CNT=（31524548]

（自2011年以来的天数>=106）和（自2011年以来的天数<=323）=>cnt=（31524548]=>cnt=[223152]

解释很简单：如果条件适用，我们可以预测右侧的自行车数量间隔。最后一个规则是当其他规则都不适用于实例时应用的默认规则。要预测新实例，请从列表顶部开始，检查规则是否适用。当一个条件匹配时，规则的右侧就是这个实例的预测。默认规则确保始终存在预测。

4.5.3贝叶斯规则列表

在本节中，我将向您展示学习决策列表的另一种方法，它遵循以下粗略的方法：

1. 从数据中预挖掘频繁的模式，这些模式可用作决策规则的条件。
2. 从一系列预先挖掘的规则中学习决策列表。

使用此配方的特定方法称为贝叶斯规则列表（Letham等人，2015年）或简称为BRL。BRL使用贝叶斯统计从频繁模式中学习决策列表，这是FP树算法（Borgelt 2005）的前提，但让我们从BRL的第一步慢慢开始。

## 频繁模式的预挖掘

频繁模式是特征值的频繁（共）出现。作为预处理步骤

BRL算法，我们使用特征（在这个步骤中我们不需要目标结果）并从中提取频繁发生的模式。模式可以是单个特征值（如大小=中等）或特征值（如大小=中等）和位置=差）的组合。

模式的频率是通过其在数据集中的支持来测量的：

\[支持（x j=a）=\frac 1 n \ sum i=1 ^ni（x ^（i）j=a）\]

其中a是功能值，n是数据集中的数据点数量，i是如果实例i的功能（x\_j）具有级别a，则返回1的indicator函数，否则为0。在房屋价值数据集中，如果20%的房屋没有阳台，80%的房屋有一个或多个阳台，那么对模式阳台的支持为20%。还可以测量功能值组合的支持，例如阳台=0和宠物=allowed。

有许多算法可以找到这种频繁的模式，例如Apriori或FP增长。您使用的模式并不重要，只有找到模式的速度不同，但得到的模式总是相同的。

我将给你一个大致的概念，先验算法如何工作，以找到频繁的模式。实际上，Apriori算法由两部分组成，第一部分查找频繁的模式，第二部分从中构建关联规则。对于BRL算法，我们只对Apriori第一部分中生成的频繁模式感兴趣。

在第一步中，Apriori算法从所有支持大于用户定义的最小支持的特性值开始。如果用户说最小支持应该是10%，只有5%的房子的大小=大，我们会删除这个功能值，只保留大小=中，大小=小作为模式。这并不意味着房屋从数据中删除，它只是意味着“大小=大”不会作为频繁模式返回。先验算法基于具有单个特征值的频繁模式，迭代地寻找越来越高阶的特征值组合。模式是通过将feature=value语句与逻辑and（例如，size=medium和location=bad）相结合来构造的。将删除支持低于最小支持的生成模式。最后我们有了所有常见的模式。频繁模式的任何子集都会再次频繁出现，这称为Apriori属性。直观地说，它是有意义的：通过从模式中删除一个条件，简化的模式只能覆盖更多或相同数量的数据点，但不能少于。例如，如果20%的房屋大小=中等，位置=良好，那么只有大小=中等的房屋的支撑为20%或更大。Apriori属性用于减少要检查的模式的数量。只有在频繁模式的情况下，我们必须检查高阶模式。

现在我们已经完成了贝叶斯规则列表算法的预挖掘条件。但在我们继续进行BRL的第二步之前，我想提示一下基于预先挖掘的模式的规则学习的另一种方法。其他方法建议将感兴趣的结果包含到频繁的模式挖掘过程中，并执行构建ifthen规则的Apriori算法的第二部分。由于该算法是无监督的，因此该部分还包含我们不感兴趣的特征值。但是我们可以通过规则过滤，这些规则在当时的部分中只具有感兴趣的结果。这些规则已经形成了一个决策集，但也可以对规则进行排列、删减、删除或重新组合。

然而，在BRL方法中，我们使用频繁的模式，学习然后的部分以及如何使用贝叶斯统计将模式安排到决策列表中。

## 学习贝叶斯规则列表

BRL算法的目标是通过选择预先挖掘的条件来学习一个准确的决策列表，同时对规则较少、条件较短的列表进行优先级排序。BRL通过定义决策列表的分布来实现这一目标，其中包含条件长度（最好是较短的规则）和规则数量（最好是较短的列表）的优先分布。

列表的后验概率分布使我们能够在假设短的情况下，以及列表与数据的匹配程度下，判断决策列表的可能性有多大。我们的目标是找到最大化后验概率的列表。由于无法直接从列表的分布中找到准确的最佳列表，BRL建议采用以下方法：

1. 生成从先验分布中随机抽取的初始决策列表。
2. 通过添加、切换或删除规则来迭代修改列表，确保生成的列表允许列表的后分布。
3. 根据后验分布，从概率最高的抽样列表中选择决策列表。

让我们更仔细地研究一下算法：该算法从预先挖掘特征值模式开始，使用fp增长算法。BRL对目标的分布和定义目标分布的参数的分布做了许多假设。（那是贝叶斯统计。）如果你不熟悉贝叶斯统计，不要太在意下面的解释。重要的是要知道贝叶斯方法是一种结合现有知识或需求（所谓的先验分布）的方法，同时也适合于数据。在决策列表的情况下，贝叶斯方法是有意义的，因为先前的假设用短规则推动决策列表变短。

目标是从后验分布中抽取决策列表D：

\[\Underrace P（D X，Y，A、\alpha、\lambda、\eta）Posteriori \Propto\Underrace P（Y X，D、\alpha）

\_可能性\cdot\underbrace p（d a、\lambda、\eta）先验\]

其中，d是决策列表，x是特征，y是目标，a是预先挖掘的条件集，决策列表的先前预期长度，a是规则中先前预期的条件数，a是正类和负类的先前伪计数，其中最好固定在（1,1）。

\[P（D X，Y，A、\alpha、\lambda、\eta）\]

根据观察数据和先验假设，量化决策列表的可能性。这与给定决策列表和数据的结果Y的可能性成正比，乘以给定先前假设和预先挖掘条件的列表概率。

\[P（Y\_X，D，alpha）\]

给定决策列表和数据，是观察到的y的可能性。brl假设y由Dirichlet多项式分布生成。决策列表D解释数据越多，可能性越高。\[P（D A、\lambda、\eta）\]

是决策列表的优先分发。它将列表中规则数的截断泊松分布（参数\（\lambda\）与规则条件中特征值数的截断泊松分布（参数\（\eta\）相乘。

如果一个决策列表能够很好地解释结果Y，并且根据先前的假设，它具有很高的后验概率。

贝叶斯统计中的估计总是有点棘手，因为我们通常不能直接计算出正确的答案，但是我们必须绘制候选者，对他们进行评估，并使用马尔可夫链蒙特卡罗方法更新后验估计。对于决策列表，这更为棘手，因为我们必须从决策列表的分布中提取数据。BRL的作者建议首先绘制一个初始决策列表，然后对其进行迭代修改，从列表的后分布（决策列表的马尔可夫链）中生成决策列表的样本。结果可能依赖于初始决策列表，因此建议重复此过程以确保列表的多样性。软件实现中的默认值是10次。以下配方告诉我们如何绘制初始决策列表：

* 带FP增长的矿前模式。
* 从截断的泊松分布中采样列表长度参数m。
* 对于默认规则：对目标值的Dirichlet多项式分布参数\（\theta\_0\）进行采样（即在没有其他应用时应用的规则）。
* 对于决策列表规则j=1，…，m，do：

对规则J的规则长度参数L（条件数）进行采样。

从预采条件中取样长度为（L\_j）的条件。

对该部分的Dirichlet多项式分布参数进行抽样（即给定规则的目标结果的分布）

* 对于数据集中的每个观测：

从首先应用的决策列表中查找规则（从上到下）。

根据适用规则建议的概率分布（二项式）得出预测结果。

下一步是从这个初始样本开始生成许多新的列表，从决策列表的后分布中获取许多样本。

新的决策列表从初始列表开始取样，然后随机将规则移动到列表中的其他位置，或者从预先挖掘的条件向当前决策列表添加规则，或者从决策列表中删除规则。随机选择要切换、添加或删除的规则。在每个步骤中，该算法评估决策列表的后验概率（准确性和短性的混合）。大都会黑斯廷斯算法确保我们对具有高后验概率的决策列表进行抽样。这个过程为我们提供了许多来自决策列表分发的样本。BRL算法选择后验概率最高的样本的决策列表。

## 示例

这就是理论，现在让我们看看BRL方法的实际应用。示例使用了Yang等人（2017）提出的一种更快的BRL变种，称为可伸缩贝叶斯规则列表（SBRL）。我们使用SBRL算法来预测i首先必须离散化所有输入特征，以便SBRL算法工作。为此，我将基于分位数值频率的连续特征组合起来。我们得到以下规则：

## 规则

如果stds=1（规则[259]），则正概率=0.16049383

否则，如果荷尔蒙避孕药..年=[0,10）（规则[82]），则正概率=0.04685408，否则（默认规则），则正概率=0.2777778

注意，我们得到了合理的规则，因为当时的预测部分不是类结果，而是癌症的预测概率。

这些条件是从使用fp增长算法预先挖掘的模式中选择的。下表显示了SBRL算法在构建决策列表时可以选择的条件池。在我作为用户允许的条件下，功能值的最大数目是2。以下是十种图案的示例：

预采条件怀孕次数=[3.67,7.33）

Iud=0，stds=1

number.of.sexual.partners=[1,10），stds..time.since.last.diagnosis=[1,8]

第一，性交=10,17.3，性病=0

Smokes=1，Iud..年。=[0,6.33）

激素。避孕药。年。=[10,20），性病。数量。诊断=[0,1）

年龄=[13,36.7）

激素。避孕药=1，性病。数量。诊断=[0,1）

number.of.sexual.partners=[1,10），stds..number.=[0,1.33）

stds..number.=[1.33,2.67），stds..time.since.first.diagnosis=[1,8）

接下来，我们将SBRL算法应用到。只有当预测自行车数量的回归问题转化为二元分类任务时，这才有效。我随意创建了一个分类任务，创建了一个标签，如果一天的自行车数量超过4000辆，则为1，否则为0。

以下列表由SBRL学习：

## 规则

如果yr=2011，temp=[-5.22,7.35）（规则[718]），则正概率=0.01041667，否则，如果yr=2012，temp=[7.35,19.9）（规则[823]），则正概率=0.88125000，否则，如果yr=2012，temp=[19.9,32.5]（规则[816]），则正概率=0.99253731，否则，如果季节=春季（规则[351]），则POSI如果年=2011，温度=[7.35,19.9）（规则[730]），则主动概率=0.06410256；否则，则主动概率=0.79746835。

让我们预测一下，2012年，在17摄氏度的温度下，自行车的数量将超过4000辆。第一条规则不适用，因为它只适用于2011年的几天。第二条规则适用，因为这一天是在2012年，17度是在时间间隔[7.35,19.9]。我们对这种可能性的预测是超过4000辆自行车的出租率是88%。4.5.4优势

本节一般讨论if-then规则的好处。

如果-那么规则很容易解释。它们可能是可解释模型中最容易解释的。此声明仅适用于规则数量较少、规则条件较短（最多3个）以及规则组织在决策列表或不重叠的决策集中的情况。

决策规则可以像决策树那样具有表现力，同时更紧凑。决策树通常也会遭受复制的子树，也就是说，当左子节点和右子节点中的拆分具有相同的结构时。

使用if-then规则的预测很快，因为只需要检查几个二进制语句就可以确定哪些规则适用。

决策规则对于输入特征的单调变换具有很强的鲁棒性，因为条件中只有阈值会发生变化。它们对于异常值也很健壮，因为它只在条件适用或不适用时才重要。

if-then规则通常生成稀疏模型，这意味着不包含许多特性。它们仅为模型选择相关功能。例如，默认情况下，线性模型为每个输入特征指定权重。不相关的特性可以简单地被if-then规则忽略。

像来自ONER这样的简单规则可以用作更复杂算法的基线。

4.5.5缺点

本节一般讨论if-then规则的缺点。

关于if-then规则的研究和文献主要集中在分类上，几乎完全忽略了回归。虽然你可以将一个连续的目标划分为间隔，并将其转化为分类问题，但你总是会丢失信息。一般来说，如果方法可以同时用于回归和分类，则更具吸引力。

这些特征通常也必须是分类的。这意味着如果要使用数字功能，必须对它们进行分类。有很多方法可以将一个连续的特性分割成间隔，但这并不是一件微不足道的事情，并且会带来许多没有明确答案的问题。该功能应划分为多少个间隔？分割标准是什么：固定的间隔长度、分位数或其他什么？对连续特性进行分类是一个经常被忽视的问题，人们只使用下一个最佳方法（就像我在示例中所做的那样）。

许多旧的规则学习算法容易过度拟合。这里给出的算法都至少有一些保护措施来防止过拟合：ONER是有限的，因为它只能使用一个特性（只有当特性具有太多的级别或有许多特性时才有问题，这等同于多个测试问题），ripper进行修剪和间隔ESIA规则列表对决策列表施加了优先分布。

在描述特征和输出之间的线性关系时，决策规则是不好的。这是他们与决策树共享的问题。决策树和规则只能产生阶跃预测函数，其中预测的变化总是离散的阶跃，而不是平滑的曲线。这与输入必须是分类的问题有关。在决策树中，它们通过拆分来隐式分类。

4.5.6软件和备选方案

在中实现了ONER，本书中的示例使用了ONER。OneR也可以在Java、R和Python中实现。裂土器也在WEKA中实现。对于这些示例，我在sbrl中使用了jrip的r实现，它是可用的as（我在示例中使用的），in或as。

我甚至不会列出学习决策规则集和列表的所有备选方案，而是指出一些总结工作。我推荐Fuernkranz等人（2012年）的《规则学习的基础》一书，这是一本关于学习规则的广泛著作，适用于那些想深入研究该主题的人。它为思考学习规则提供了一个整体框架，并提出了许多规则学习算法。我还建议检查哪个工具开膛手，M5规则，ONER，零件等等。if-then规则可以用在线性模型中，如本书中关于

* 1. 《机器学习》11.1（1993）：63-90。
  2. 快速有效规则归纳〉，《机器学习程序》（1995年）。115-123页。
  3. Letham，Benjamin等人“使用规则和贝叶斯分析的可解释分类器：建立更好的中风预测模型。”《应用统计学年鉴》9.3（2015）：1350-1371。
  4. 《FP增长算法的一种实现》，《第一届会议录》

开放源代码数据挖掘国际研讨会频繁模式挖掘

实施-OSDM'05，1–5.（2005年）。

* 1. 杨、洪宇、辛西娅·鲁丁和玛戈·塞尔茨。第34届国际机器学习会议论文集70。JMLR。组织，2017年。
  2. F\_rnkranz、Johannes、Dragan Gamberger和Nada Lavra\_。“规则学习的基础”，斯普林格科学与商业媒体（2012年）。

4.6规则拟合

Friedman和Popescu（2008）的Rulefit算法学习稀疏线性模型，其中包括以决策规则形式自动检测到的交互效应。

线性回归模型不考虑特征之间的相互作用。拥有一个像线性模型一样简单和可解释的模型，同时又集成了特征交互，这难道不方便吗？rulefit填补了这个空白。rulefit学习具有原始特征的稀疏线性模型，以及许多新的特征，这些特征是决策规则。这些新特性捕获原始特性之间的交互。RuleFit自动从决策树生成这些功能。通过将拆分的决策组合成规则，可以将树中的每条路径转换为决策规则。节点预测将被丢弃，决策规则中仅使用拆分：

图4.21:4规则可以从具有3个终端节点的树中生成。

这些决策树来自哪里？这些树被训练来预测感兴趣的结果。这样可以确保分割对预测任务有意义。任何生成大量树的算法都可以用于规则拟合，例如随机林。每棵树被分解成决策规则，作为稀疏线性回归模型（lasso）中的附加特征。

Rulefit论文使用波士顿住房数据来说明这一点：目的是预测波士顿社区的房屋价值中位数。Rulefit生成的规则之一是：如果房间数>6.64，一氧化氮浓度<0.67，则1或0。

Rulefit还提供了一个特征重要性度量，有助于识别对预测很重要的线性项和规则。根据回归模型的权重计算特征重要性。重要性度量可以针对原始特性进行聚合（这些特性以“原始”形式使用，并且可能在许多决策规则中使用）。

Rulefit还引入了部分相关图，通过改变特征来显示预测的平均变化。偏相关图是一种模型不可知论方法，可用于任何模型，并在

4.6.1解释和示例

由于rulefit最终估计了一个线性模型，所以解释与“normal”相同。唯一的区别是该模型具有从决策规则派生的新特性。决策规则是二进制特性：值1表示满足规则的所有条件，否则值为0。对于规则拟合中的线性项，其解释与线性回归模型中的解释相同：如果特征增加一个单位，则预测的结果将根据相应的特征权重变化。

在这个例子中，我们使用rulefit来预测给定日期的数量。该表显示了rulefit生成的五个规则，以及它们的套索权重和重要性。计算将在本章后面解释。

## 说明权重重要性

|  |  |
| --- | --- |
| 自2011年起第111天及天气状况（良好、雾状）793 | 303个 |
| 37.25<=hum<=90-20 | 272个 |
| 温度>13&自2011年以来的天数>554 676 | 239个 |
| 4<=风速<=24-41 | 202 |
| 自2011年以来的天数>428天，温度>5 366天 | 179个 |

最重要的规则是：“自2011年11月11日起的天数和天气状况”（“好”，“雾”），相应的重量为793。其解释是：如果自2011年以来的天数大于111天且天气状况稳定（“良好”，“雾状”），那么当所有其他特征值保持不变时，预测的自行车数量将增加793辆。总共有278个规则是根据最初的8个特性创建的。相当多！但多亏了套索，278只中只有58只的体重与0不一样。

计算全局特征重要性表明，温度和时间趋势是最重要的特征：

图4.22：预测自行车数量的规则拟合模型的特征重要性度量。预测的最重要特征是温度和时间趋势。

特征重要性度量包括原始特征项的重要性和特征出现的所有决策规则。

## 解释模板

这种解释类似于线性模型：如果一个单元的特征（x）发生变化，则预测结果会发生变化，前提是所有其他特征保持不变。决策规则的权重解释是一种特殊情况：如果一个决策规则的所有条件都适用，那么预测的结果会改变（在线性模型中规则的已知权重）。

对于分类（使用逻辑回归而不是线性回归）：如果决策规则的所有条件都适用（r\_k），则事件与无事件的概率会变化一个系数。

4.6.2理论

让我们深入研究rulefit算法的技术细节。Rulefit由两个组件组成：第一个组件从决策树创建“规则”，第二个组件使用原始特性和新规则作为输入来适应线性模型（因此称为“Rulefit”）。

## 步骤1:规则生成

规则是什么样子的？算法生成的规则具有简单的形式。例如：如果x2<3和x5<7，则为1，否则为0。这些规则是通过分解决策树来构建的：到树中节点的任何路径都可以转换为决策规则。规则中使用的树被拟合来预测目标结果。因此，对分割和结果规则进行了优化，以预测您感兴趣的结果。您只需将导致某个节点的二进制决策与“and”链接起来，voil\_，您就有了一个规则。人们希望能制定出许多多样而有意义的规则。梯度增强用于通过回归或分类Y来适应一组决策树，使用您的原始特征X。每个结果树被转换成多个规则。不仅可以增强树，而且可以使用任何树集成算法来生成树以进行规则拟合。树系综可以用以下一般公式来描述：

\[f（x）=a\_+\和m=1 ^m a f m（x）\]

m是树的数目，而（f\_m（x）是m-树的预测函数。（α）是权重。袋装套装、随机森林、Adaboost和Mart生产树套装，可用于规则拟合。

我们从合奏的所有树中创造出规则。每个规则（r\_m）的形式如下：

\[r\_m（x）=\prod j \ in \ text t u m i（x\_j \ in s jm）\]

其中\（\text t m）是m-th树中使用的一组功能，i是当功能\（x \_j\）位于j-th功能的指定值子集（由树拆分指定）中时为1的指标函数，否则为0。对于数字特性，是特性值范围内的间隔。间隔看起来像两种情况之一：

\[X\_S\_JM，\TEXT LOWER\_

\xYJ

该特征的进一步分割可能导致更复杂的间隔。对于分类特征，子集S包含特征的某些特定类别。

自行车租赁数据集的组成示例：

\[R 17（x）=i（x 123;\text <15）CDOT （x）=i（x \\\125\<20）\_]

如果满足所有三个条件，则此规则返回1，否则返回0。rulefit从树中提取所有可能的规则，而不仅仅是从叶节点中。因此，另一个将要创建的规则是：

\[R 18（X）=I（X \文本温度<15）\CDOT I（X \文本天气\在\ \文本良好，\文本多云

总的来说，由m树和（t\_m）终端节点组成的集合所创建的规则数为：

\[k=\总和m=1 ^m2（t m-1）\]

Rulefit作者引入的一个技巧是学习具有随机深度的树，以便生成许多具有不同长度的不同规则。注意，我们丢弃每个节点中的预测值，只保留将我们引导到节点的条件，然后从中创建规则。决策规则的权重在Rulefit的步骤2中进行。

另一种查看步骤1的方法：rulefit从原始特性生成一组新特性。

这些特性是二进制的，可以表示原始特性的非常复杂的交互。选择规则最大化预测任务。规则是从协变量矩阵X自动生成的。您可以简单地将规则视为基于原始特性的新特性。

## 第二步：稀疏线性模型

在步骤1中有很多规则。由于第一步只能被视为特征转换，因此您仍然没有完成模型的拟合。此外，您还希望减少规则的数量。除了规则之外，原始数据集中的所有“原始”特性也将用于稀疏线性模型。每一条规则和每一个原始特征都成为线性模型中的一个特征，并得到一个权重估计。添加原始特征是因为树在表示y和x之间的简单线性关系时失败。在训练稀疏线性模型之前，我们先对原始特征进行Winsorize，以便它们对异常值更为强大：\[l\_j^\*（x\_j）=min（\delta\_j^+，max（\delta\_j^-，x\_j））\]

其中\（\ delta ^-\）和\（\ delta ^+\）是功能\（x j\）的数据分布的\（\ delta\）分位数。选择0.05代表\（\ delta\）意味着在5%最小值或5%最大值中的任何特征值（x\ j\）将分别设置为5%或95%的分位数。根据经验，您可以选择\（\delta\）=0.025。此外，必须对线性项进行归一化，使其具有与典型决策规则相同的优先重要性：

\[L\_j（x\_j）=0.4\cdot l ^\*\_j（x\_j）/std（l ^\*\_j（x\_j））\]

（0.4）是规则的平均标准偏差，统一支持分布为（s\_k \ sim u（0,1））。

我们结合这两种特征生成新的特征矩阵，并用lasso训练稀疏线性模型，其结构如下：

\[\hat f（x）=\hat \β

其中\（\hat \ alpha \）是规则功能的估计权重向量，而\（\hat \ beta \）是原始功能的权重向量。由于rulefit使用lasso，因此loss函数将获得额外的约束，强制某些权重获得零估计：

\[(\{\hat{\alpha}\}\_1^K,\{\hat{\beta}\}\_0^p)=argmin\_{\{\hat{\alpha}\}\_1^K,\{\hat{\beta}\}\_0^p}\sum\_{i=1} ^n{}L(y^{(i)},f(x^{(i)}))+\lambda\cdot\left(\sum\_{k=1}^K|\alpha\_k|+\sum\_{j=1}^p|b\_j|\right)\]

结果是一个线性模型，对所有原始特征和规则都有线性影响。与线性模型的解释相同，唯一的区别是一些特征现在是二进制规则。

## 步骤3（可选）：功能重要性

对于原始特征的线性项，使用标准化预测因子测量特征重要性：

\[I\_j=\Hat \Beta \U j \Cdot Std（L\_j（x\_j））\]

其中，βj是来自lasso模型的权重，std（l\_j（x\_j））是线性项相对于数据的标准偏差。

对于决策规则术语，重要性的计算公式如下：

\[我\那\阿尔法\cdot \ sqrt s u k（1-s \]

其中\（\hat \ alpha u k \）是决策规则的关联lasso权重，而\（s k \）是数据中功能的支持，这是决策规则适用的数据点的百分比（其中\（r \_k（x）=0\）：

\[S K=\frac 1 N \ sum I=1 ^N R U K（X ^（I））]

一个特征作为一个线性项出现，也可能出现在许多决策规则中。我们如何衡量功能的总体重要性？一个特征的重要性（j（x）可以为每个单独的预测进行测量：

\[j\_j（x）=i\_l（x）+和x\_j \ in r\_k i\_k（x）/m\_k \]

其中，（i\_l）是线性项的重要性，以及（i\_k）决策规则的重要性，其中（x\_j）出现，并且（m\_k）是构成规则的特征数（r\_k）。从所有实例中添加功能重要性可以使我们获得全局功能重要性：

\[J（x）=\sum i=1 ^n j\_j（x^（i））]

可以选择实例的子集并计算此组的功能重要性。

4.6.3优势

rulefit自动将特征交互添加到线性模型中。因此，它解决了线性模型的问题，需要手动添加交互项，这对非线性关系的建模有一定帮助。

Rulefit可以处理分类和回归任务。

创建的规则易于解释，因为它们是二元决策规则。规则是否适用于实例。只有规则中的条件数量不太多，才能保证良好的解释性。我觉得有1到3个条件的规则是合理的。这意味着树群中树的最大深度为3。

即使模型中有许多规则，它们也不适用于每个实例。对于个别情况，只有少数规则适用（=具有非零权重）。这提高了本地的可解释性。

Rulefit提出了一系列有用的诊断工具。这些工具是模型不可知论的，所以您可以在本书的模型不可知论部分找到它们。

4.6.4缺点

有时rulefit会创建许多规则，在lasso模型中获得非零权重。随着模型中特征数量的增加，解释性降低。一个有希望的解决方案是强制特征效果是单调的，这意味着特征的增加必须导致预测的增加。

一个奇闻轶事的缺点：论文声称规则拟合的良好性能——通常接近于随机森林的预测性能！–但在我亲自尝试的少数案例中，表现令人失望。只需针对您的问题尝试一下，看看它是如何工作的。

rulefit过程的最终产物是一个具有额外的花哨特性（决策规则）的线性模型。但由于它是一个线性模型，因此权重解释仍然是非确定性的。它和通常的线性回归模型一样带有相同的“脚注”：“……假设所有的特性都是固定的。”当你有重叠的规则时，它会变得更加复杂。例如，自行车预测的一个决策规则（特征）可能是：“temp>10”，另一个规则可能是“temp>15&weather='good'”。如果天气好，温度在15度以上，温度自动高于10度。在第二条规则适用的情况下，第一条规则也适用。对第二条规则的估计重量的解释是：“假设所有其他特征保持不变，当天气好，温度超过15度时，预测的自行车数量增加了”（β2）。但是，现在很明显，“所有其他特征都是固定的”是有问题的，因为如果规则2适用，规则1也适用，并且解释是无意义的。

4.6.5软件和备选方案

Rulefit算法由Fokkema和Christoffersen（2017）在R中实现，您可以找到

一个非常相似的框架是一个python模块，它也从集合中提取规则。它在学习最终规则的方式上有所不同：首先，skope规则根据回忆和精度阈值删除性能低下的规则。然后，根据逻辑项（变量+大/小运算符）的多样性和规则的性能（F1分数）执行选择，删除重复的和类似的规则。最后一步并不依赖于使用lasso，而是只考虑现成的f1分数和构成规则的逻辑术语。

* + 1. Friedman、Jerome H和Bogdan E Popescu。“通过规则集合进行预测性学习。”应用统计学的名称。Jstor，916–54。（2008年）。
    2. Fokkema、Marjolein和Benjamin Christoffersen。“预：预测规则集合”。（2017年）。

4.7其他可解释模型

可解释模型的列表不断增长，而且规模未知。它包括简单的模型，如线性模型、决策树和朴素贝叶斯，但也包括更复杂的模型，这些模型结合或修改了不可解释的机器学习模型，使它们更易于解释。尤其是后一类模型的出版物目前正以高频率出版，很难跟上发展的步伐。这本书在这一章只取笑朴素的贝叶斯分类器和k-最近的邻居。

4.7.1朴素贝叶斯分类器

朴素贝叶斯分类器使用条件概率的贝叶斯定理。对于每个特征，它根据特征值计算类的概率。naive bayes分类器独立地计算每个特征的类概率，这相当于特征独立性的强（naive）假设。Naive Bayes是一个条件概率模型，它对类（c\_k）的概率建模如下：

\[P（C\_K\_X）=\frac 1 Z P（C\_K）\prod I=1 ^N P（X\_I C\_U K）\]

术语z是一个比例参数，它确保所有类的概率之和为1（否则它们将不是概率）。类的条件概率是类概率乘以给定类的每个特征的概率，用z归一化。这个公式可以用贝叶斯定理导出。

由于独立性假设，朴素贝叶斯是一个可解释的模型。它可以在模块级别上解释。对于每个特征来说，它对某个类预测的贡献是非常明显的，因为我们可以解释条件概率。

4.7.2 k-最近的邻居

k-最近邻法可用于回归和分类，并利用数据点的最近邻进行预测。对于分类，k-最近邻方法为实例的最近邻分配最常见的类。对于回归，它取邻居结果的平均值。棘手的部分是找到正确的k并决定如何测量实例之间的距离，这最终定义了邻域。

K-最近邻模型不同于本书提出的其他可解释模型，因为它是一种基于实例的学习算法。如何解释k-最近邻？

首先，没有要学习的参数，所以在模块级别上没有可解释性。

此外，由于模型本身是局部的，没有显式学习的全局权重或结构，因此缺乏全局模型的可解释性。也许在地方一级可以解释？要解释预测，您总是可以检索用于预测的k个邻居。模型是否可解释完全取决于您能否“解释”数据集中的单个实例。如果一个实例包含成百上千个特性，那么它就不可解释了，我会争辩。但是，如果您没有一些特性或将实例简化为最重要的特性的方法，那么展示K最近的邻居可以为您提供很好的解释。

第五章模型不可知论方法

将解释与机器学习模型分离（=模型不可知解释方法）具有一些优势（Ribeiro、Singh和Guestrin，2016年）。模型不可知解释方法相对于模型特定解释方法的最大优势在于其灵活性。当解释方法可以应用于任何模型时，机器学习开发人员可以自由使用他们喜欢的任何机器学习模型。任何建立在机器学习模型解释之上的东西，例如图形或用户界面，也会独立于底层的机器学习模型。通常，不只是一个模型，而是对许多类型的机器学习模型进行评估以解决一个任务，并且在比较模型的可解释性时，更容易使用模型不可知论解释，因为相同的方法可用于任何类型的模型。

模型不可知论解释方法的一个替代方法是仅使用，与其他机器学习模型相比，这种方法通常有一个很大的缺点，即预测性能会丢失，并且您只能使用一种模型。另一种选择是使用特定于模型的解释方法。这样做的缺点是它也将您绑定到一个模型类型，并且很难切换到其他类型。

模型不可知论解释系统的理想方面是（Ribeiro、Singh和Guestrin，2016年）：

* 模型灵活性：解释方法可以与任何机器学习模型一起使用，如随机森林和深层神经网络。
* 解释的灵活性：你不局限于某种形式的解释。在某些情况下，使用线性公式可能很有用，而在其他情况下，使用具有重要特征的图形可能很有用。
* 表示灵活性：解释系统应该能够使用不同的特征表示作为被解释的模型。对于使用抽象词嵌入向量的文本分类器，最好使用单个词的存在进行解释。

大局

让我们从高层次上看一下模型不可知论的可解释性。我们通过收集数据来捕捉世界，并通过学习使用机器学习模型来预测（任务的）数据来进一步抽象世界。可解释性只是帮助人们理解的另一层。

图5.1：可解释机器学习的大画面。现实世界在以解释的形式到达人类之前经历了许多层次。

最底层是世界。这可以说是自然本身，就像人体的生物学和它对药物的反应一样，但也可以说是更抽象的东西，比如房地产市场。世界层包含所有可以观察到的和感兴趣的东西。最终，我们想了解这个世界并与之互动。

第二层是数据层。我们必须将世界数字化，使之能被计算机处理，也能存储信息。数据层包含来自图像、文本、表格数据等的任何内容。

通过对基于数据层的机器学习模型进行拟合，得到了黑盒模型层。机器学习算法利用来自现实世界的数据进行学习，以做出预测或找到结构。

黑盒模型层之上是可解释方法层，它帮助我们处理机器学习模型的不透明度。特定诊断最重要的特征是什么？为什么金融交易被归类为欺诈？

最后一层被人占据。看！这一个向你挥手，因为你正在阅读这本书，并帮助提供更好的黑盒模型的解释！人类最终是解释的消费者。

这种多层抽象也有助于理解统计学家和机器学习实践者在方法上的差异。统计学家处理数据层，例如规划临床试验或设计调查。他们跳过黑盒模型层，直接进入可解释性方法层。机器学习专家也处理数据层，例如收集皮肤癌图像的标记样本或爬行维基百科。然后他们训练一个黑匣子机器学习模型。跳过可解释性方法层，人类直接处理黑箱模型预测。可解释的机器学习融合了统计学家和机器学习专家的工作，这很好。

当然，这张图并不能捕捉到一切：数据可能来自模拟。黑匣子模型还输出预测，这些预测甚至可能无法到达人类，但只提供其他机器，等等。但总的来说，理解可解释性如何成为机器学习模型之上的新层是一个有用的抽象。

1。Ribeiro、Marco Tulio、Sameer Singh和Carlos Guestrin。“机器学习的模型不可知论可解释性”，ICML机器学习中的人类可解释性研讨会。（2016年）。

5.1部分相关图（PDP）

偏相关图（简称pdp或pd图）显示了一个或两个特征对机器学习模型预测结果的边际效应（J.H.Friedman 2001）。部分相关图可以显示目标和特征之间的关系是线性的、单调的还是更复杂的。例如，当应用于线性回归模型时，偏相关图总是显示线性关系。

回归的偏相关函数定义为：

\[\Hat F X（X S）=E X C \ Left[\Hat F（X S，X C \ Right]=\Int\Hat F（X S，X C）D\MathBB P（X C）]

（x\_s）是绘制部分依赖函数的特征，而（x\_c）是机器学习模型中使用的其他特征。通常情况下，集合S中只有一个或两个特征。S中的特征是那些我们想知道对预测的影响的特征。特征向量（x\_s）和（x\_c）组合构成了整个特征空间x。部分依赖性通过将机器学习模型输出边缘化在集合c中特征的分布上起作用，以便函数显示我们感兴趣的集合s中特征之间的关系。以及预测的结果。通过对其他特性的边缘化，我们得到了一个仅依赖于S中特性的函数，与包含的其他特性的交互。

通过计算训练数据中的平均值，即蒙特卡罗方法，估计部分函数（即：

\[\hat f x（x s）=\frac 1 n \ sum i=1 f（x（i）c）部分函数告诉我们特征的给定值对预测的平均边际影响是什么。在这个公式中，（x^（i）c）是数据集中我们不感兴趣的特性的实际特性值，n是数据集中的实例数。PDP的一个假设是，C中的特征与S中的特征不相关。如果违反此假设，则为部分相关图计算的平均值将包括非常不可能或甚至不可能的数据点（见缺点）。

对于机器学习模型输出概率的分类，部分依赖图显示给定不同特征值的特定类在S中的概率。处理多个类的简单方法是为每个类绘制一条线或图。

部分相关图是一种全局方法：该方法考虑了所有实例，并给出了特征与预测结果的全局关系的说明。

分类特征

到目前为止，我们只考虑了数值特征。对于分类特征，部分依赖性很容易计算。对于每个类别，我们通过强制所有数据实例具有相同的类别来获得PDP估计。例如，如果我们查看自行车租赁数据集并对季节的部分依赖关系图感兴趣，我们会得到4个数字，每个季节一个。为了计算“夏季”的值，我们将所有数据实例的季节替换为“夏季”，并对预测进行平均。

5.1.1示例

实际上，特征集通常只包含一个或最多两个特征，因为一个特征生成二维图，两个特征生成三维图。除此之外的一切都很棘手。即使是2D纸或显示器上的3D也已经很有挑战性了。

让我们回到回归示例，在该示例中我们预测首先我们拟合了机器学习模型，然后分析了部分依赖关系。在这种情况下，我们拟合了一个随机森林来预测自行车的数量，并使用偏相关图来可视化模型所了解的关系。天气特征对预测自行车数量的影响如下图所示。

图5.2：自行车数量预测模型和温度、湿度和风速的PDPS。最大的差别可以在温度上看到。温度越高，租的自行车越多。这种趋势上升到20摄氏度，然后在30摄氏度时变平并略有下降。X轴上的标记表示数据分布。

对于温暖但不太热的天气，该模型预测平均有大量租用自行车。当湿度超过60%时，潜在的骑自行车的人越来越不愿意租自行车。此外，风越大，人们喜欢骑车的次数就越少，这是有道理的。有趣的是，当风速从25公里/小时增加到35公里/小时时，自行车租赁量的预测值不会下降，但培训数据不多，因此机器学习模型可能无法对该范围进行有意义的预测。至少凭直觉，我预计自行车的数量会随着风速的增加而减少，特别是当风速很高的时候。

为了说明具有分类特征的部分依赖关系图，我们研究了季节特征对自行车租赁预测的影响。

图5.3：自行车数量预测模型和季节的PDP。出乎意料的是，所有的季节都表现出同样的效果，只有在春季，该模型预测自行车租赁量会减少。

我们还计算了的部分依赖性。这一次，我们根据风险因素对一个女人是否会患上宫颈癌进行了随机森林预测。我们计算并可视化随机森林癌症概率对不同特征的部分依赖性：

图5.4：基于年龄和使用激素避孕药的年龄的癌症概率的pdps。对于年龄而言，pdp显示40岁之前的概率很低，40岁之后的概率增加。激素避孕药使用年限越长，预测的癌症风险就越高，尤其是10年后。对于这两个特征，没有多少具有大值的数据点可用，因此这些区域的PD估计不太可靠。

我们还可以同时看到两个特性的部分依赖性：

图5.5：癌症概率的pdp以及年龄和怀孕次数的相互作用。图中显示45岁时癌症发病率增加。对于25岁以下的女性，与0个或2个以上怀孕的女性相比，1个或2个怀孕的女性预测癌症风险更低。但在得出结论时要小心：这可能只是一种相关性，而不是因果关系！

5.1.2优势

部分相关图的计算是直观的：如果我们强制所有数据点假设该特征值，则特定特征值的部分相关函数表示平均预测。根据我的经验，外行人通常很快就能理解PDP的概念。

如果计算PDP的特征与其他特征不相关，则PDP完美地表示该特征如何影响平均预测。在不相关的情况下，解释是清楚的：偏相关图显示了当j-th特性改变时，数据集中的平均预测是如何变化的。当特性相关时，它就更复杂了，参见缺点。

部分依赖图易于实现。

部分相关图的计算有一个因果解释。我们介入了

描述并测量预测中的变化。在此过程中，我们分析了特征与预测之间的因果关系。这种关系是模型的因果关系——因为我们明确地将结果建模为特征的函数——但不一定是真实世界的结果！

5.1.3缺点

偏相关函数中的实际最大特征数是2。这不是PDP的错误，而是二维表示（纸张或屏幕）的错误，也是我们无法想象超过3个维度的错误。

某些局部放电图不显示特征分布。忽略分布可能会产生误导，因为您可能会高估几乎没有数据的区域。这个问题很容易通过显示RUG（X轴上数据点的指示器）或柱状图来解决。

独立性假设是PD图的最大问题。假设计算部分依赖性的特征与其他特征不相关。例如，假设你想预测一个人走路的速度，考虑到这个人的体重和身高。对于其中一个特征的部分依赖性，例如高度，我们假设其他特征（重量）与高度不相关，这显然是一个错误的假设。为了计算某一高度（例如200 cm）的PDP，我们对重量的边际分布进行平均，其中可能包括50 kg以下的重量，这对于2米长的人来说是不现实的。换言之：当特征相关时，我们在特征分布区域中创建新的数据点，其中实际概率非常低（例如，不太可能有人身高2米，但体重小于50千克）。解决这个问题的一个方法是使用条件而不是边际分布的短ALE图。

异质效应可能被隐藏，因为PD图只显示平均边际效应。假设对于一个特征，一半的数据点与预测呈正相关——特征值越大，预测越大——另一半的数据点与预测呈负相关——特征值越小，预测越大。pd曲线可以是一条水平线，因为数据集的两部分的效果可以相互抵消。然后得出结论，该特性对预测没有影响。通过绘制而不是聚合线，我们可以发现异构效果。

5.1.4软件和备选方案

有许多R包可以实现PDP。我在示例中使用了IML包，但也有PDP或DALEX。在python中，你可以使用skater。

本书中介绍的PDP替代方案有：

1. 贪婪函数近似：梯度增强机〉，《统计年鉴》（2001）：1189-1232。
2. 赵、清远和特雷弗·黑斯迪。“黑匣子模型的因果解释”，《商业与经济统计杂志》即将出版。（2017年）。

5.2个人条件期望（ICE）

单个条件期望（ICE）图为每个实例显示一行，显示当某个特性更改时实例的预测如何更改。

特征平均效果的部分依赖图是一种全局方法，因为它不关注特定实例，而是关注总体平均值。对于单个数据实例，等价于PDP的方法称为单个条件期望（ICE）图（Goldstein等人2017。冰图分别显示了预测对每个实例的特征的依赖性，与部分依赖图中的一条线相比，每个实例产生一条线。pdp是冰层的平均线。一条线（和一个实例）的值可以通过保持所有其他功能相同来计算，通过将该功能的值替换为网格中的值来创建该实例的变体，并使用这些新创建的实例的黑盒模型进行预测。结果是一组具有来自网格的特征值和相应预测的点。

观察个人期望而不是部分依赖有什么意义？部分依赖图可以掩盖由交互创建的异构关系。PDP可以向您展示特征和预测之间的平均关系。只有当计算PDP的功能与其他功能之间的交互较弱时，这才有效。在相互作用的情况下，冰图将提供更多的洞察。

一个更正式的定义：在冰区中，对于每一个实例，在（x 123;（i），x c ^；（（i）；）125\ \123;（i）\（x（i）125\\\\\\125\\（（125;（i）i）\\\），而\（x^（i）c）保持不变。

5.2.1示例

让我们回到，看看每个实例的预测如何与特性“年龄”相关联。我们将分析一个随机森林，该森林预测一个给定危险因素的妇女患癌症的可能性。在我们已经看到的癌症概率在50岁左右增加，但这对数据集中的每个女人都是真的吗？冰图显示，对于大多数女性来说，年龄效应遵循50岁时的平均增长模式，但也有一些例外：对于少数年轻时具有高预测概率的女性，预测的癌症概率不会随年龄变化太大。

图5.6：按年龄划分的宫颈癌概率冰点图。每一行代表一个女人。对于大多数女性来说，随着年龄的增长，预测癌症的可能性也会增加。对于一些癌症预测概率在0.4以上的女性来说，在更高的年龄段预测变化不大。

下一个图显示了的冰层图。基础预测模型是一个随机森林。

图5.7：根据天气条件预测自行车租赁的冰层图。在偏相关图中也可以观察到同样的效果。

所有曲线似乎都遵循相同的过程，因此没有明显的相互作用。这意味着，PDP已经很好地总结了显示的特征和预测的自行车数量之间的关系。

## 5.2.1.1中心冰层图

冰层图有一个问题：有时很难判断冰层曲线在个体之间是否不同，因为它们从不同的预测开始。一个简单的解决方案是将曲线集中在特征的某个点上，只显示预测到该点的差异。生成的图称为中心冰图（C-ICE）。将曲线锚定在特征的下端是一个不错的选择。新曲线定义为：

\[\Hat（i）=\Hat（i）-\MathBF 1 \Hat F（x，x（i）c）]

其中\（\mathbf 1 \）是具有适当数量尺寸（通常为一个或两个）的1的矢量，其中\（\hat f \）是拟合模型，x是锚定点。a

## 5.2.1.2示例

例如，以宫颈癌冰点图作为年龄，并以观察到的最年轻年龄为中心：

图5.8：根据年龄预测癌症概率的中心冰层图。行在14岁时固定为0。与14岁相比，大多数女性的预测值在45岁之前保持不变，直到预测的概率增加。

中心冰图使比较单个实例的曲线更容易。如果我们不希望看到预测值的绝对变化，而是预测值与特征范围的固定点之间的差异，那么这一点很有用。

让我们来看看自行车租赁预测的中心冰区：

图5.9：根据天气条件预测的自行车数量的中心冰层图。这些线显示了预测值与各自特征值在观测最小值时的预测值之间的差异。

## 5.2.1.3衍生冰图

另一种使从视觉上更容易发现异质性的方法是观察预测函数相对于特征的个别导数。生成的图称为衍生冰图（dice）。函数（或曲线）的导数告诉您是否发生了变化以及它们发生的方向。利用导数冰层图，很容易发现特征值的范围，在该范围内，黑匣子预测（至少某些）实例会发生变化。如果分析的特征（x\_s）和其他特征（x\_c）之间没有相互作用，那么预测函数可以表示为：

\[\hat f（x）=\hat f（x s，x c）=g（x s）+h（x c），\quad\text with \ quad\frac \ delta\hat（x）\ delta x=g'（x s）]

如果没有相互作用，个别偏导数在所有情况下都应该是相同的。如果它们不同，那是由于相互作用，并且在D-冰图中可见。除了显示预测函数导数相对于S中特征的个别曲线外，显示导数的标准差有助于突出S中特征中估计导数中具有异质性的区域。导数冰层图需要很长的时间来计算，这是相当不切实际的。

5.2.2优势

个体条件期望曲线比部分依赖图更直观。如果我们改变感兴趣的特性，一行代表一个实例的预测。

与部分依赖图不同，冰曲线可以揭示异质关系。

5.2.3缺点

冰曲线只能有意义地显示一个特征，因为两个特征需要绘制多个覆盖曲面，并且在绘图中看不到任何内容。

冰曲线与PDPS有着相同的问题：如果感兴趣的特征与其他特征相关，那么根据联合特征分布，直线上的某些点可能是无效的数据点。

如果绘制了许多冰的曲线，绘图可能会变得过于拥挤，您将看不到任何东西。解决方案：要么在线条中添加一些透明度，要么只绘制线条的示例。

在冰区，要看到平均值可能不容易。这有一个简单的解决方案：将单个条件期望曲线与部分依赖图结合起来。

5.2.4软件和备选方案

冰图在R包IML（用于这些示例）、冰箱和PDP中实现。另一个和冰非常相似的R包是康迪斯。

1. Goldstein、Alex等。“窥视黑盒子：用个体条件期望图可视化统计学习”，《计算和图形统计杂志》24.1（2015）：44-65。
2. Goldstein、Alex等。“包装‘冰箱’”（2017年）。

5.3累积局部效应（ALE）图

累积局部效应描述特征如何平均影响机器学习模型的预测。ALE图是部分依赖图（PDP）的一种快速、无偏的替代方法。

我建议首先阅读，因为它们更容易理解，而且两种方法都有相同的目标：都描述了一个特性如何平均地影响预测。在下面的部分中，我想让您相信，当特性相关时，部分依赖关系图有一个严重的问题。

5.3.1动机和直觉

如果机器学习模型的特征是相关的，那么部分依赖图就不可信。计算与其他特征强相关的特征的部分相关图涉及对实际中不太可能出现的人工数据实例的平均预测。这会对估计的特征效果产生很大的偏差。想象一下，计算机器学习模型的部分依赖图，该模型根据房间数量和居住面积大小预测房屋的价值。我们对生活区对预测值的影响感兴趣。作为提醒，部分依赖图的方法是：1）选择功能。2）定义网格。3）每个网格值：a）用网格值替换特征，b）平均预测。4）绘制曲线。为了计算PDP的第一个网格值（比如30米），我们将两个实例的居住面积都替换为30米，即使是有10个房间的房屋。在我看来，这房子很不寻常。2部分依赖图将这些不切实际的房屋包含在特征效果估计中，并假装一切都很好。下图说明了两个相关特性，以及偏相关绘图方法如何平均不太可能的实例的预测。

图5.10：强相关特征x1和x2。为了计算x1在0.75时的特征效果，pdp将所有实例的x1替换为0.75，错误地假设x2在x1时的分布=

0.75与x2（垂直线）的边缘分布相同。这导致不太可能的x1和x2组合（例如，x1=0.75时x2=0.2），PDP使用这些组合计算平均效应。

我们能做些什么来得到一个关于特征相关性的特征效果估计呢？我们可以对特性的条件分布取平均值，也就是说，在网格值为x1时，我们用类似的x1值对实例的预测取平均值。使用条件分布计算特征效果的解决方案称为边际图或M图（混淆名称，因为它们基于条件分布，而不是边际分布）。等等，我有没有答应过你谈一下艾尔的阴谋？M图不是我们正在寻找的解决方案。为什么M图不能解决我们的问题？如果我们将所有房屋的预测平均为30米左右，我们估计了居住面积和房间数量的2个综合效应，因为它们之间存在相关性。假设居住面积对房屋的预测价值没有影响，只有房间的数量有影响。M图仍然显示，由于房间数量随着居住面积的增加而增加，因此居住面积的大小会增加预测值。下面的图表显示了两个相关的特性MPLOT是如何工作的。

图5.11：强相关特征x1和x2。条件分布上的M图平均值。这里x2在x1=0.75时的条件分布。局部预测的平均值会导致混合两种特征的影响。

M图避免对不可能的数据实例进行平均预测，但它们混合了特征的影响和所有相关特征的影响。ALE图通过计算（也基于特征的条件分布）预测差异而不是平均值来解决这个问题。对于30 m处的居住面积的影响，ALE方法使用了所有约30 m的房屋，得到假设这些房屋为31 m的2个模型预测减去假设它们为2 29 m的预测。这给了我们生活区的纯粹效果，而不是混合2个相关特征的效果。使用差异会阻碍其他功能的效果。下图提供了如何计算ALE图的直观性。

图5.12：与x2相关的特性x1的ALE计算。首先，我们将特征划分为间隔（垂直线）。对于区间中的数据实例（点），我们用区间的上下限（水平线）代替特征，计算预测的差异。这些差异随后被累积并集中，形成ALE曲线。

要总结每种类型的图（pdp、m、ale）如何计算某个特征在某个网格值v下的效果：

部分依赖图：“让我展示一下当每个数据实例的该特性值为V时，模型平均预测的内容。我忽略值v是否对所有数据实例都有意义。”

M-plots：“让我向您展示该模型对数据实例的平均预测，这些数据实例的值接近该特性的v。影响可能是由于该功能，也可能是由于相关功能。”

ale plots：“让我演示一下，对于该窗口中的数据实例，模型预测是如何在围绕v的特性的一个小“窗口”中发生变化的。”

5.3.2理论

pd、m和ale图在数学上有什么不同？这三种方法的共同点是，它们将复杂的预测函数f简化为仅依赖一（或两）个特征的函数。这三种方法都通过平均其他特征的影响来降低函数，但它们在计算预测的平均值或预测的差异以及在边际分布或条件分布上是否进行平均值方面存在差异。

部分依赖图平均预测超过边际分布。

\开始 123; 123; 123; X X X 123; C x\_c \结束对齐\*\]

这是预测函数f的值，在特征值\（x\_s\）处，对\（x\_c\）中的所有特征取平均值。平均是指计算集合C中特征的边际期望E，它是通过概率分布加权的预测的积分。听起来很奇怪，但是要计算边缘分布上的预期值，我们只需获取所有数据实例，强制它们对集合S中的特性具有特定的网格值，并平均此操作数据集的预测值。这个过程确保我们对特征的边缘分布进行平均。

M图对条件分布的预测进行平均。

\[\begin align\*\hat f x s，m（x s）&=e x c x u s \ left[\hat f（x s，x c）x s=x s \ right]\

\&=\int\_x\_c \hat\_f（x\_s，x\_c）\mathbb\_p（x\_c\_x\_s）d\_x\_c\end\_align\*\]

与PDP相比唯一的变化是，我们根据感兴趣的特征的每个网格值来平均预测，而不是假设每个网格值的边际分布。在实践中，这意味着我们必须定义一个邻域，例如计算30米对预测房屋价值的影响，我们可以将所有房屋的预测平均为28到32米2。

ALE图平均预测中的变化，并在网格上进行累积（稍后的计算将详细介绍）。

\[\begin align\*\hat f x s，ale（x s）=&\int z 0,1 x e x u c x u s \ left[\hat f s（x s，x c）|

x\_s=z\_s \right]dz\_s-\text常数\\\ \\\ \ x U s \\ 123; 123; f \美国，x U c \\123; \123; \123; 123; \\123; \123 d z\_s-\text常量\结束对齐\*\]

该公式揭示了M图的三个差异。首先，我们平均预测的变化，而不是预测本身。这种变化被定义为梯度（但在随后的实际计算中，被一个区间内预测的差异所取代）。

\[\hat f ^s（x\_s，x\_c）=\frac \ delta \hat f（x\_s，x\_c）\ delta x\_u s \]

第二个差异是z上的附加积分，我们在集合s中积累了特征范围内的局部梯度，这给了我们特征对预测的影响。对于实际的计算，z被一个区间网格代替，在这个网格中我们计算预测的变化。ALE方法不直接求预测的平均值，而是计算特征S的预测差异，并将特征S上的导数整合在一起以估计效果。嗯，听起来很愚蠢。派生和集成通常会相互抵消，比如先减去，然后再加上相同的数字。为什么这里有意义？导数（或区间差）隔离了感兴趣特征的影响，并阻止了相关特征的影响。

ALE图与M图的第三个区别是我们从结果中减去一个常数。此步骤将ALE图居中，以便对数据的平均影响为零。

还有一个问题：并不是所有的模型都有一个梯度，例如随机森林没有梯度。但是正如您将看到的，实际的计算没有梯度，而是使用间隔。让我们更深入地探讨ALE图的估计。

5.3.3估算

首先，我将描述如何估计ALE图的单个数值特征，随后是两个数值特征和单个分类特征。为了估计局部效应，我们将特征划分为多个区间，并计算预测的差异。此过程近似于渐变，也适用于没有渐变的模型。

首先，我们估计非中心效应：

\[\hat \tilde f j，ale（x）=\sum k=1 ^ k j（x）\frac 1 n j（k）\ sum i:x（i）\ in n u j（k）

\左[F（Z K，J，X ^（I）\集减）-F（Z K-1，J，X ^（I）\集减J）\右]\]

让我们把这个公式分解，从右边开始。这个名称累积的局部效果很好地反映了这个公式的所有单个组成部分。ALE方法的核心是计算预测中的差异，因此我们用网格值Z替换感兴趣的特征。预测中的差异是特征在特定间隔内对单个实例的影响。右边的总和将所有实例的影响相加，并将其作为邻域（n\_j（k））显示在公式中。我们将此和除以此间隔中的实例数，得到此间隔预测的平均差。间隔中的平均值包含在名称ale中的术语local中。左和符号意味着我们在所有间隔中积累平均效果。例如，位于第三个间隔中的特征值的（未确定的）ALE是第一个间隔、第二个间隔和第三个间隔的效果之和。ALE中累积的单词反映了这一点。

这个效果居中，所以平均效果为零。

\ \123;（x）-\ frc \ \ \\123; \\ 123;（i）j）]

与数据的平均预测值相比，ALE值可以解释为某一特定值下特征的主要影响。例如，当j-th特征值为3时，ale估计值为-2（x\_j=3），则预测值比平均预测值低2。

特征分布的分位数用作定义间隔的网格。使用分位数可以确保在每个间隔中有相同数量的数据实例。分位数的缺点是间隔的长度可能非常不同。如果感兴趣的特性是非常偏斜的，这可能会导致一些奇怪的ALE图，例如许多低值和只有几个非常高的值。

## 两个特征相互作用的ALE图

ALE图还可以显示两个特征的交互作用。计算原理与单个特征相同，但我们使用矩形单元而不是间隔，因为我们必须将效果累积为二维。除了调整总体平均效果外，我们还调整了这两个特征的主要效果。这意味着两个特征的ALE估计二阶效应，而不包括特征的主要效应。换句话说，两个特征的ALE只显示了这两个特征的附加交互效果。我把2dale图的公式留给你，因为它们很长，读起来很不愉快。如果你对这个计算感兴趣，我可以参考这篇论文，公式（13）–（16）。我将依靠可视化来发展二阶ALE计算的直觉。

图5.13:2d-ALE的计算。我们在这两个功能上放置一个网格。在每个网格单元中，我们计算内所有实例的二阶差。我们首先用单元格角的值替换x1和x2的值。如果a、b、c和d代表“角点”——操作实例的预测（如图中所示），则二阶差为（d-c）-（b-a）。每个单元的平均2ndorder差在网格上累积并居中。

在前面的图中，由于相关性，许多单元格是空的。在ALE图中，可以使用灰显或暗显框来可视化。或者，您可以用最近的非空单元的ALE估计替换空单元缺少的ALE估计。

由于两个特征的ALE估计仅显示了特征的二阶效应，因此解释需要特别注意。二阶效应是在考虑了特征的主要影响后，特征的附加交互效应。假设两个特征不相互作用，但每个特征对预测结果都有线性影响。在每个特征的一维ALE图中，我们可以看到一条直线作为估计的ALE曲线。但是当我们绘制二维ALE估计时，它们应该接近于零，因为二阶效应只是交互作用的附加效应。ALE图和PD图在这方面有所不同：PDP始终显示总效果，ALE图显示一阶或二阶效果。这些是不依赖于基础数学的设计决策。你可以减去部分依赖图中的低阶效应得到纯的主效应或二阶效应，或者，你可以通过避免减去低阶效应得到总ALE图的估计值。

累积的局部效应也可以针对任意高阶（三个或多个特征的相互作用）进行计算，但如中所述，只有至多两个特征是有意义的，因为更高的相互作用无法可视化，甚至无法有意义地解释。

## 分类特征的ALE

根据定义，累积局部效果方法需要特征值具有顺序，因为该方法在特定方向上累积效果。分类特征没有任何自然顺序。为了计算分类特征的ALE图，我们必须以某种方式创建或找到一个顺序。分类的顺序影响着累积局部效应的计算和解释。

一种解决方案是根据类别的相似性根据其他特征对其进行排序。两个类别之间的距离是每个要素距离的总和。特征距离比较了这两类中的累积分布，也称为Kolmogorovsmirnov距离（对于数值特征）或相对频率表（对于分类特征）。一旦我们有了所有类别之间的距离，我们就使用多维尺度将距离矩阵减少到一维距离度量。这给了我们一个基于相似性的分类顺序。

为了更清楚地说明这一点，这里有一个例子：假设我们有两个分类特征“季节”和“天气”，以及一个数字特征“温度”。对于第一个分类特征（季节），我们要计算销售业绩。该功能分为“春”、“夏”、“秋”、“冬”。我们开始计算“春季”和“夏季”之间的距离。距离是特征温度和天气之间距离的总和。对于温度，我们以所有季节“春季”的实例为例，计算经验累积分布函数，并对季节“夏季”的实例做同样的计算，并用Kolmogorov-Smirnov统计量测量它们的距离。对于天气特征，我们计算所有“春季”实例的每种天气类型的概率，对“夏季”实例进行相同的计算，并求出概率分布中的绝对距离。如果“春季”和“夏季”的温度和天气差异很大，则总类别距离较大。我们对其他季节对重复该过程，并通过多维缩放将生成的距离矩阵减少到单个维度。

5.3.4示例

让我们看看ALE计划的实施。我构建了一个部分依赖图失效的场景。该场景由一个预测模型和两个强相关特征组成。预测模型主要是一个线性回归模型，但是在这两个我们从未观察到的特征的组合中做了一些奇怪的事情。

图5.14：两个特征和预测结果。模型预测两个特征（阴影背景）的总和，但如果x1大于0.7且x2小于0.3，则模型始终预测2。该区域远离数据分布（点云），不影响模型的性能，也不影响模型的解释。

这是一个现实的，相关的场景吗？训练模型时，学习算法将现有训练数据实例的损失降至最低。奇怪的事情可能发生在培训数据的分布之外，因为模型不会因为在这些领域做奇怪的事情而受到惩罚。离开数据分布被称为外推法，也可以用来愚弄机器学习模型，如中所述。在我们的小示例中，可以看到与ALE图相比，部分依赖图的行为。

图5.15：使用pdp（上排）和ale（下排）计算的特征效果比较。PDP估计值受数据分布之外模型的奇怪行为（图中的陡跳）的影响。ALE图正确识别了机器学习模型在特征和预测之间存在线性关系，忽略了没有数据的区域。

但看到我们的模型在x1>0.7和x2<0.3时表现得很奇怪，这不是很有趣吗？是的，也不是。因为这些数据实例在物理上可能是不可能的，或者至少是极不可能的，所以研究这些实例通常是不相关的。但是，如果您怀疑您的测试分布可能略有不同，并且某些实例实际上在这个范围内，那么在计算特性效果时包含这个区域将是很有趣的。但是，必须有意识地决定是否包括我们尚未观察到数据的区域，并且它不应该是选择方法（如PDP）的副作用。如果您怀疑模型稍后将用于不同分布的数据，我建议使用ALE图并模拟您期望的数据分布。

转向一个真实的数据集，让我们根据天气和白天来预测，并检查ALE图是否真的像承诺的那样工作。我们训练一个回归树来预测一天内租用自行车的数量，并使用ALE图来分析温度、相对湿度和风速对预测的影响。让我们看看ALE的情节是怎么说的：

图5.16：基于温度、湿度和风速的自行车预测模型的ALE图。温度对预测有很大影响。平均预测值随着温度的升高而升高，但再次下降到25摄氏度以上。湿度有负面影响：当湿度超过60%时，相对湿度越高，预测值越低。风速对预测影响不大。

让我们来看看温度、湿度、风速和所有其他特征之间的关系。由于数据也包含分类特征，所以我们不能只使用皮尔逊相关系数，这只在两个特征都是数值的情况下才有效。相反，我训练一个线性模型来预测温度，例如，基于作为输入的其他特征之一。然后我测量线性模型中的另一个特征解释了多少方差，并取平方根。如果另一个特征是数值的，则结果等于标准皮尔逊相关系数的绝对值。但是这种基于模型的“方差解释”方法（也称为方差分析，它代表方差分析）即使其他特征是分类的也可以工作。“方差解释”的测量值总是在0（无关联）和1（温度可以从另一个特征完美预测）之间。我们计算了温度、湿度和风速随其他特征的变化。解释的方差（相关性）越高，PD图的问题（潜在）就越多。下图显示了天气特征与其他特征的关联程度。

图5.17：当我们训练一个线性模型时，温度、湿度和风速与所有特征之间的相关性的强度，根据解释的方差量进行测量。

例如温度预测和季节特征。对于温度，我们观察到——不足为奇——与季节和月份有很高的相关性。湿度与天气状况有关。

这种相关性分析表明，我们可能会遇到部分依赖图的问题，特别是温度特征。好吧，你自己看看：

图5.18：温度、湿度和风速的PDPS。与ALE地块相比，

在高温或高湿度条件下，预测的自行车数量减少较小。PDP使用所有数据实例来计算高温的影响，即使它们是，例如，“冬季”季节的实例。ALE图更可靠。

下一步，让我们看看ALE图对于分类特征的作用。月份是一个分类特征，我们要分析它对自行车预测数量的影响。可以说，月份已经有了一定的顺序（1月到12月），但是让我们试着看看如果我们先按相似性重新排序类别，然后计算影响会发生什么。根据其他特征（如温度或是否为假日）按每月天数的相似性排列月份。

图5.19：分类特征月的ALE图。月份是根据它们之间的相似性来排序的，基于按月分布的其他特性。我们观察到，1月、3月和4月，尤其是12月和11月，对预计租用自行车数量的影响比其他月份要小。

由于许多特征与天气有关，月份顺序强烈反映了月份之间的天气是多么相似。所有较冷的月份都在左侧（2月至4月），较温暖的月份则在右侧（10月至8月）。请记住，相似性计算中也包含了非天气特征，例如，假日的相对频率与计算月份之间相似性的温度具有相同的权重。

接下来，我们考虑了湿度和温度对自行车预测数量的二阶效应。记住，二阶效应是两个特征的附加交互效应，不包括主要效应。这意味着，例如，在二阶ALE图中，您将看不到高湿度导致预测自行车数量平均较低的主要影响。

图5.20：湿度和温度对预测租用自行车数量的二阶效应的ALE图。当已经考虑到主要影响时，较浅的阴影表示高于平均值，较深的阴影表示低于平均值的预测。该图揭示了温度和湿度之间的相互作用：湿热天气增加了预测。在寒冷和潮湿的天气下，预测的自行车数量会受到额外的负面影响。

请记住，湿度和温度的两个主要影响都表明，预测的自行车数量在非常炎热和潮湿的天气会减少。在湿热天气中，温度和湿度的综合效应不是主要效应的总和，而是大于总和。为了强调纯二阶效应（刚才看到的二维ALE图）和总效应之间的区别，让我们来看看偏相关图。PDP显示了综合效应，综合了平均预测、两个主要效应和二阶效应（交互作用）。

图5.21：温度和湿度对预测自行车数量的总影响的PDP。与仅显示交互的二维ALE图相比，该图结合了每个特征的主要效果及其交互效果。

如果你只对交互感兴趣，你应该考虑二阶效应，因为总效应将主要效应混合到图中。但是，如果您想知道这些特性的组合效果，您应该看看总效果（PDP显示）。例如，如果您想知道30摄氏度和80%湿度下自行车的预期数量，可以直接从2dpd中读取。如果您想从ALE图中读取相同的内容，您需要查看三个图：温度、湿度和温度+湿度的ALE图，还需要了解总体平均预测。在两个特征没有交互作用的场景中，这两个特征的总效果图可能会产生误导，因为它可能显示出复杂的景观，暗示了一些交互作用，但它只是两个主要影响的产物。二阶效应会立即表明没有相互作用。

现在的自行车够多了，我们来看看分类任务。我们训练一个随机森林来预测基于风险因素的概率。我们将两个特征的累积局部效应可视化：

图5.22：年龄和年龄与激素避孕药对宫颈癌预测概率影响的ALE图。对于年龄特征，ALE图显示，到40岁时，预测的癌症概率平均较低，之后会增加。激素避孕药使用年限与8年后预测的癌症风险较高相关。

接下来，我们来看看怀孕次数和年龄之间的相互作用。

图5.23：妊娠次数和年龄的二级效应ALE图。对情节的解释有点不确定，显示了什么似乎是过度适合。例如，该图显示了18-20岁和3个以上怀孕（癌症概率增加5个百分点）时的奇怪模型行为。在这组年龄和怀孕次数的数据中，女性并不多（实际数据显示为点），因此在培训过程中，该模型不会因女性犯错误而受到严重惩罚。

5.3.5优势

ALE图是无偏的，这意味着当特征相关时它们仍然有效。部分依赖图在这种情况下失败，因为它们被边缘化了不太可能或甚至是物理上不可能的特征值组合。

ALE图的计算速度比PDP快，并且使用O（N）进行缩放，因为可能的最大间隔数是每个实例具有一个间隔的实例数。PDP需要n倍的网格点估计数。对于20个网格点，PDP需要比最坏情况下ALE图多20倍的预测，其中使用的间隔与实例相同。

ALE图的解释是清楚的：在给定值的条件下，改变特征对预测的相对影响可以从ALE图中读出。ALE图的中心为零。这使得他们的解释很好，因为ALE曲线的每一点的值与平均预测值是不同的。二维ALE图只显示交互：如果两个功能不交互，则图中不显示任何内容。

总之，在大多数情况下，我更喜欢ALE图而不是PDP，因为特性通常在某种程度上是相关的。

5.3.6缺点

ALE情节可能会变得有点不稳定（许多小的起伏），间隔时间很长。在这种情况下，减少间隔数会使估计更稳定，但也会使预测模型变得更平滑，并隐藏一些真正的复杂性。没有完美的解决方案来设置间隔的数量。如果数字太小，ALE图可能不太准确。如果数字太高，曲线可能会变得不稳定。

与PDP不同，ALE图不伴有冰曲线。对于PDP来说，ICE曲线是很好的，因为它们可以揭示特征效果的异质性，这意味着一个特征的效果对于数据的子集是不同的。对于ALE图，只能按间隔检查实例之间的效果是否不同，但每个间隔都有不同的实例，因此它与ICE曲线不同。

二阶ALE估计在整个特征空间中具有不同的稳定性，这是不以任何方式可视化的。其原因是，对一个单元中局部效果的每个估计都使用不同数量的数据实例。因此，所有的估计都有不同的准确性（但它们仍然是最好的估计）。这个问题存在于一个不太严重的主效应ALE图版本中。由于使用分位数作为网格，所有间隔中的实例数都是相同的，但在某些区域会有许多短间隔，ALE曲线将由更多的估计组成。但对于可以占整个曲线很大一部分的长间隔，实例相对较少。例如，这发生在宫颈癌预测ALE图中。

二阶效应图解释起来有点烦人，因为你总是要记住主要效应。将热图解读为这两个特征的总效应是很有诱惑力的，但它只是交互作用的附加效应。纯粹的二阶效应对于发现和探索相互作用是很有趣的，但是对于解释这种效应是什么样子，我认为把主要效应整合到情节中更有意义。

与部分依赖图相比，ALE图的实现更为复杂，也更不直观。

即使ALE图在相关特征的情况下没有偏差，但当特征强相关时，解释仍然困难。因为如果它们之间有很强的相关性，那么分析同时改变两个特性而不是孤立地改变这两个特性的效果是有意义的。这种缺点并不是针对ALE图，而是一个具有强相关特征的一般问题。

如果特征不相关，计算时间也不成问题，则PDP稍好，因为它们更容易理解，并且可以与冰曲线一起绘制。

缺点列表已经变得相当长了，但是不要被我使用的词的数量愚弄：作为经验法则：使用ALE而不是PDP。

5.3.7实施和备选方案

我有没有提到这一点，并且是另一种选择？=）

据我所知，ALE绘图目前仅在R中实现，一次由发明家自己实现，一次由

1。可视化黑盒监督学习模型中预测变量的影响〉，《ARXIV预印本：1612.08468》（2016）。

5.4特征交互

当特征在预测模型中相互作用时，由于一个特征的影响取决于另一个特征的值，因此预测不能表示为特征效果的总和。亚里士多德的谓词“整体大于各部分之和”适用于相互作用的存在。

5.4.1特征交互？

如果机器学习模型基于两个特征进行预测，我们可以将预测分解为四个术语：常量项、第一个特征项、第二个特征项和两个特征之间的交互项。

两个特征之间的相互作用是在考虑到个别特征的影响后，通过改变特征发生的预测变化。

例如，模型使用房屋大小（大或小）和位置（好或坏）作为特征来预测房屋的价值，从而得出四个可能的预测：

位置大小预测

好大300000好小200000坏大250000坏小150000

我们将模型预测分解为以下几个部分：一个常量项（150000）、大小特征的影响（+100000，如果大；+0，如果小）和位置的影响（+50000，如果好；+0，如果坏）。这种分解完全解释了模型预测。没有交互作用，因为模型预测是大小和位置的单一特征效应的总和。当你把一个小房子变大的时候，不管位置如何，预测总是增加10万。另外，不管大小，一个好位置和一个坏位置的预测差异是50000。

现在我们来看一个交互示例：

位置大小预测

好大400000好小200000坏大250000坏小150000

我们将预测表分解为以下几部分：一个常量项（150000）、大小特征的影响（+100000表示大，+0表示小），以及位置的影响（+50000表示好，+0表示坏）。对于这张表，我们需要一个额外的交互术语：如果房子很大并且位置好，则+100000。这是大小和位置之间的相互作用，因为在这种情况下，大房子和小房子的预测差异取决于位置。

估计交互强度的一种方法是测量预测的变化在多大程度上取决于特征的交互作用。这项测量被称为H统计，由Friedman和Popescu（2008）介绍。

5.4.2理论：弗里德曼H统计量

我们将处理两种情况：首先，一种双向交互度量，它告诉我们模型中的两个特征是否相互作用，以及扩展到什么程度；其次，一种全面交互度量，它告诉我们模型中的特征是否与所有其他FEA相互作用，以及扩展到什么程度。好的。理论上，可以测量任意数量特征之间的任意交互，但这两种情况是最有趣的。

如果两个特征不相互作用，我们可以分解如下（假设部分相关函数以零为中心）：

\[pd\_j k（x\_j，x\_k）=pd\_j（x\_j）+pd\_k（x\_k）\]

其中（pd\_j k（x\_j，x\_k）是两个功能的双向部分依赖函数，以及\（pd\_j（x\_j）和\（pd\_k（x\_k））单个功能的部分依赖函数。

同样，如果一个特征与其他特征没有交互作用，我们可以将预测函数\（\hat f（x））表示为部分依赖函数的和，其中第一个和仅依赖于j，第二个和依赖于除j以外的所有其他特征：

\[\Hat f（x）=Pd（x j）+Pd-j（x-j）\]

其中\（pd-j（x-j））是依赖于除jth特性之外的所有特性的部分依赖函数。

这种分解表示部分依赖（或完全预测）函数，没有相互作用（特征J和K之间，或分别是J和所有其他特征）。在下一步中，我们测量了观察到的偏相关函数和没有相互作用的分解函数之间的差异。我们计算部分依赖（测量两个特征之间的相互作用）或整个函数（测量特征与所有其他特征之间的相互作用）输出的方差。交互作用（观察到的和没有交互作用的pd之间的差异）解释的方差量用作交互强度统计。如果完全没有交互作用，则统计值为0；如果用偏相关函数的和解释了\（pd jk \）或\（\hat f \）的所有方差，则统计值为1。两个特征之间的交互统计量为1意味着每个单独的pd函数都是常数，对预测的影响只来自于交互作用。

从数学上讲，弗里德曼和波佩斯库提出的特征j和k之间相互作用的h-统计量是：

\[H^2\_{jk}=\sum\_{i=1}^n\left[PD\_{jk}(x\_{j}^{(i)},x\_k^{(i)})-PD\_j(x\_j^{(i)})-PD\_k(x\_{k}^{(i)}) \right]^2/\sum\_{i=1}^n{PD}^2\_{jk}(x\_j^{(i)},x\_k^{(i)})\]

同样适用于测量特征J是否与任何其他特征相互作用的情况，即测量特征J是否与任何其他特征相互作用，即[H^2 123; i=1 ^n\left[\hat（x（i）（i）123;（i））-pd \123;-j（（（u i=1 ^n\hat f ^2（x ^（i））\]

H统计量的评估成本很高，因为它在所有数据点上迭代，并且在每个点上，必须评估部分依赖性，而这反过来又对所有N个数据点进行评估。在最坏的情况下，我们需要2n调用机器学习模型预测函数来计算双向H-2统计（j vs.k）和3n计算总H统计（j vs.all）。为了加快计算速度，我们可以从n个数据点进行2次采样。这有增加偏相关估计方差的缺点，使得H统计量不稳定。因此，如果您使用采样来减少计算负担，请确保采样足够的数据点。

弗里德曼和波佩斯库还提出了一个检验统计来评估H统计与零是否有显著差异。无效假设是没有相互作用。要在零假设下生成交互统计，必须能够调整模型，使其在特征j和k或所有其他项之间没有交互。这不可能适用于所有类型的模型。因此，这个测试是特定于模型的，而不是不可知模型的，因此这里没有介绍。

如果预测是概率的话，交互强度统计也可以应用于分类设置中。

5.4.3示例

让我们看看在实践中交互的特性是什么样的！我们测量了支持向量机中的特征交互强度，它根据天气和日历特征预测特征的数量。下图显示了特征交互H统计：

图5.24：预测自行车租赁的支持向量机的每个特征与所有其他特征的交互强度（H统计）。总体而言，特征之间的交互作用非常弱（低于每个特征解释的方差的10%）。

在下一个例子中，我们计算分类问题的交互统计。我们分析了随机森林特征之间的相互作用，并给出了一些风险因素。

图5.25:a的每个特征与所有其他特征的交互强度（H统计）

随机森林预测宫颈癌的概率。激素避孕药的使用年限与其他特征的相对相互作用效应最高，其次是怀孕次数。

在查看每个特征与所有其他特征的特征交互之后，我们可以选择其中一个特征，并深入了解所选特征与其他特征之间的所有双向交互。

图5.26：怀孕次数与其他特征之间的双向互动强度（H统计）。怀孕次数和年龄之间有很强的相互作用。

5.4.4优势

相互作用的H统计量通过偏相关分解有一个基本理论。

H统计量有一个有意义的解释：交互作用被定义为由交互作用解释的方差份额。

由于统计数据是无量纲的，并且总是在0和1之间，所以它在特性上甚至在模型上都是可比较的。

统计数据检测各种交互，不管它们的具体形式如何。

利用H统计量，还可以分析任意更高的交互作用，如3个或更多特征之间的交互强度。

5.4.5缺点

你会注意到的第一件事是：交互H统计需要很长的时间来计算，因为它的计算成本很高。

计算包括估计边际分布。如果我们不使用所有的数据点，这些估计值也有一定的差异。这意味着，当我们对点进行采样时，各个运行的估计值也不同，结果可能不稳定。我建议重复H统计计算几次，看看您是否有足够的数据来获得稳定的结果。

不清楚交互作用是否显著大于0。我们需要进行一个统计测试，但是这个测试在模型不可知的版本中还没有（现在）可用。

在检验问题上，当H统计量足够大时，我们很难认为相互作用“强”。

H统计量告诉我们相互作用的强度，但它并不能告诉我们相互作用的样子。这就是目的。一个有意义的工作流是测量交互强度，然后为您感兴趣的交互创建二维部分依赖关系图。

如果输入是像素，则不能有意义地使用H统计。因此，该方法对图像分类器没有实际意义。

交互统计是在我们可以独立洗牌特征的假设下工作的。如果这些特征关联性很强，那么就违反了这个假设，我们集成了在现实中不太可能出现的特征组合。这和偏相关图的问题是一样的。一般来说，你不能说这会导致高估或低估。

有时结果是奇怪的，对于小的模拟不能产生预期的结果。但这更像是一个奇闻轶事。

5.4.6实施

对于本书中的示例，我使用了R包IML，它在上可用，而开发版本在上也有其他的实现，它们集中在特定的模型上：R包实现和H统计。R包实现了梯度增强模型和H统计。

5.4.7备选方案

H统计并不是衡量相互作用的唯一方法：

Hooker（2004）提出的变量交互网络（VIN）是一种将预测函数分解为主要效应和特征交互的方法。然后将功能之间的交互视为一个网络。不幸的是，目前还没有可用的软件。

Greenwell等人（2018）基于部分依赖性的特征交互测量了两个特征之间的交互。这种方法测量一个特征在另一个特征的不同固定点上的重要性（定义为部分依赖函数的方差）。如果差异很大，那么这些特性就相互作用，如果是零，它们就不相互作用。包上有相应的R包VIP，还包括部分依赖图和特征重要性。

1. Friedman、Jerome H和Bogdan E Popescu。“通过规则集合进行预测性学习。”应用统计学的名称。Jstor，916–54。（2008年）。
2. 贾尔斯·胡克。“发现黑盒函数中的加性结构”，《Tenthacm-Sigkdd知识发现和数据挖掘国际会议论文集》。（2004年）。
3. Greenwell、Brandon M、Bradley C.Boehmke和Andrew J.McCarthy。“简单有效的基于模型的可变重要性度量”，arxiv预印arxiv:1805.04755（2018）。

[↩](https://christophm.github.io/interpretable-ml-book/interaction.html#fnref34)

5.5特征重要性

特征的重要性在于，在对特征值进行排序后，模型的预测误差增大，从而破坏了特征与真实结果之间的关系。

5.5.1理论

这个概念非常简单：我们通过计算模型在排列特征后预测误差的增加来衡量特征的重要性。如果改变一个特征的值会增加模型的误差，那么这个特征是“重要的”，因为在这种情况下，模型依赖于该特征来进行预测。如果改变一个特征的值使模型误差保持不变，那么它就“不重要”，因为在这种情况下，模型忽略了预测的特征。Breiman（2001年）对随机森林引入排列特征重要性测量。基于这一观点，Fisher、Rudin和Dominici（2018）提出了一个功能重要性的模型不可知论版本，并将其称为模型依赖。他们还介绍了关于特征重要性的更先进的思想，例如考虑到许多预测模型可能很好地预测数据的（特定于模型的）版本。他们的论文值得一读。

基于Fisher、Rudin和Dominici（2018）的排列特征重要性算法：

输入：训练模型F，特征矩阵X，目标向量Y，误差测量L（Y，F）。

1. 估计原始模型误差eorig=l（y，f（x））（例如均方误差）
2. 对于每个特征j=1，…，p do：

通过在数据X中排列特征j生成特征矩阵xperm。这打破了特征j与真实结果y之间的关联。

根据排列数据的预测，估计误差eperm=l（y，f（xperm））。

计算排列特征重要性fi=ej排列/eorig。或者，可以使用差异：fi=ej perm-eorig

1. 按降序FI对功能进行排序。

Fisher、Rudin和Dominici（2018）在他们的论文中建议将数据集分成两半，并交换两半的特征j的值，而不是排列特征j。如果你考虑的话，这与排列特征j完全相同。如果需要更准确的估计，可以通过将每个实例与其他实例（除了自身）的特征j值配对来估计排列特征j的错误。这为您提供了一个大小为n（n-1）的数据集来估计排列错误，并且需要大量的计算时间。我只能推荐使用n（n-1）-方法，如果你真的想得到非常准确的估计。

5.5.2我应该计算培训或测试数据的重要性吗？

我没有明确的答案。

回答有关培训或测试数据的问题涉及到什么特性重要的基本问题。理解基于训练的特性重要性与基于测试数据的特性重要性之间的区别的最佳方法是一个“极端”的例子。我训练了一个支持向量机来预测给定50个随机特征（200个实例）的连续随机目标结果。“随机”是指目标结果独立于50个特征。这就像根据最新的彩票号码预测明天的温度。如果模型“学习”了任何关系，那么它就太合适了。事实上，SVM确实在训练数据上做了过多的调整。训练数据的平均绝对误差（简称：MAE）为0.29，测试数据的平均绝对误差（简称：MAE）为0.82，这也是预测平均结果为0（MAE为0.78）的最佳可能模型的误差。换句话说，SVM模型是垃圾。对于这个过度安装的SVM的50个特性，您期望的特性重要性值是多少？零，因为没有任何功能有助于改进未公开测试数据的性能？或者，不管所学的关系是否归纳为未看到的数据，导入是否反映了模型对每个特性的依赖程度？让我们来看看培训和测试数据的特征重要性的分布是如何不同的。

图5.27：按数据类型划分的特征重要性值分布。对一个具有50个随机特征和200个实例的回归数据集进行了SVM训练。支持向量机对数据进行了超拟合：基于训练数据的特征重要性显示出许多重要的特征。根据未公开的测试数据计算，特征重要性接近1的比率（=不重要）。

我不清楚这两个结果中哪一个更可取。因此，我将尝试为这两个版本做一个案例，让你自己决定。

## 测试数据案例

这是一个简单的例子：基于训练数据的模型错误估计是垃圾->功能重要性依赖于模型错误估计->基于训练数据的功能重要性是垃圾。实际上，这是你在机器学习中学到的第一件事：如果你在模型训练的同一个数据上测量模型误差（或性能），那么测量通常过于乐观，这意味着模型的工作效果似乎比实际情况要好得多。由于排列特征的重要性依赖于模型误差的测量，所以我们应该使用看不见的测试数据。基于训练数据的特征重要性使得我们错误地认为特征对于预测是重要的，而事实上，模型只是过拟合，特征根本不重要。

## 培训数据案例

使用培训数据的论据有些难以表述，但与使用测试数据的论据一样具有说服力。我们再看看我们的垃圾SVM。根据训练数据，最重要的功能是x42。让我们看一下特性X42的部分依赖关系图。部分相关图显示了模型输出如何根据特征的变化而变化，而不依赖于泛化误差。不管PDP是用训练数据还是测试数据计算的。

图5.28:x42功能的PDP，根据训练数据，根据功能重要性，它是最重要的功能。该图显示了SVM如何依赖此功能进行预测。

该图清楚地表明，支持向量机已经学会了依赖特征X42进行预测，但根据基于测试数据的特征重要性（1.04），这并不重要。根据训练数据，重要性为1.21，反映出模型已经学会使用这一特性。基于训练数据的特征重要性告诉我们哪些特征对模型很重要，因为在这个意义上，模型的预测依赖于这些特征。

作为使用培训数据的案例的一部分，我想介绍一个反对测试数据的论点。在实践中，您希望使用所有的数据来训练您的模型，最终获得最佳可能的模型。这意味着没有留下未使用的测试数据来计算特性的重要性。当您想要估计模型的泛化错误时，也会遇到同样的问题。如果将（嵌套的）交叉验证用于特性重要性估计，那么您会遇到这样的问题：特性重要性不是在具有所有数据的最终模型上计算的，而是在具有可能表现不同的数据子集的模型上计算的。

最后，您需要决定您是想知道模型在多大程度上依赖于每个特性来进行预测（->训练数据），还是该特性在多大程度上有助于模型在未知数据上的性能（->测试数据）。据我所知，目前还没有关于培训与测试数据问题的研究。它需要比我的“垃圾SVM”示例更彻底的检查。我们需要更多的研究和使用这些工具的经验来获得更好的理解。

接下来，我们将看一些例子。我基于训练数据的重要性计算，因为我必须选择一个训练数据，并且使用训练数据需要少几行代码。

5.5.3示例和解释

我展示了分类和回归的例子。

## 宫颈癌（分类）

我们拟合一个随机森林模型进行预测。我们测量误差增加1-AUC（1减去ROC曲线下的面积）。与模型误差增加1倍（无变化）相关的特征对于预测宫颈癌并不重要。

图5.29：随机森林预测宫颈癌各特征的重要性。最重要的特征是年龄。置换年龄导致1-AUC增加5.17倍。

最重要的特征是年龄与排列后误差增加5.17有关。

## 自行车共享（回归）

我们拟合了一个支持向量机模型来预测给定的天气条件和日历信息。作为误差测量，我们使用平均绝对误差。

图5.30：使用支持向量机预测自行车计数时每个功能的重要性。最重要的是临时工，最不重要的是假期。

5.5.4优势

很好的解释：特征的重要性是当特征的信息被破坏时，模型误差的增加。

功能重要性提供了对模型行为的高度压缩的全局洞察。

使用误差比而不是误差差的一个积极方面是，在不同的问题上，特征重要性度量是可比较的。

重要性度量自动考虑与其他功能的所有交互。通过排列功能，您还可以破坏与其他功能的交互效果。这意味着排列特征的重要性同时考虑了主特征效应和交互效应对模型性能的影响。这也是一个缺点，因为两个特性之间交互的重要性包含在两个特性的重要性度量中。这意味着特性导入不等于总性能下降，但总和更大。只有在特征之间没有交互作用的情况下，如线性模型中，重要的东西才会大致相加。

排列特征重要性不需要重新训练模型。其他一些方法建议删除特征，重新训练模型，然后比较模型错误。由于机器学习模型的再培训需要很长时间，“只有”排列一个特性可以节省很多时间。对具有特征子集的模型进行再培训的重要性方法乍一看似乎很直观，但是具有减少数据的模型对于特征重要性来说毫无意义。我们对固定模型的特征重要性感兴趣。使用缩减的数据集重新训练将创建一个

不同于我们感兴趣的型号。假设您训练一个稀疏线性模型（使用lasso），该模型具有固定数量的非零权重特征。数据集有100个特性，您可以将非零权重的数量设置为5。您将分析其中一个具有非零权重的特性的重要性。删除该功能并重新导入模型。模型性能保持不变，因为另一个同样好的特性得到非零权重，您的结论是该特性并不重要。另一个例子：模型是一个决策树，我们分析了选择作为第一个分割的特征的重要性。删除该功能并重新导入模型。由于选择了另一个特性作为第一个拆分，所以整棵树可能非常不同，这意味着我们比较（可能）完全不同的树的错误率，以确定该特性对其中一棵树的重要性。

5.5.5缺点

不清楚您是否应该使用培训或测试数据来计算特性的重要性。

排列特征重要性与模型的误差有关。这并不是天生的坏，但在某些情况下不是你需要的。在某些情况下，您可能更愿意知道一个特性的模型输出变化有多大，而不考虑它对性能的意义。例如，您希望了解当有人操纵特性时，模型的输出有多健壮。在这种情况下，您将不感兴趣的是，当一个特性被排列时，模型性能会下降多少，但是每个特性解释了模型的输出方差有多少。当模型很好地概括（即，它不过度拟合）时，模型方差（由特征解释）和特征重要性有很强的相关性。

你需要获得真正的结果。如果有人只向您提供模型和未标记的数据，而不是真正的结果，那么您就无法计算排列特征的重要性。

排列特征的重要性取决于对特征的梳理，这增加了测量的随机性。当排列重复时，结果可能大不相同。重复排列和对重要度量值求平均值超过重复次数会使度量值稳定，但会增加计算时间。

如果特征是相关的，排列特征的重要性可能会受到不现实的数据实例的影响。问题与：当两个或多个特征相关时，特征排列产生不太可能的数据实例。当它们是正相关的（比如一个人的身高和体重）并且我改变了其中一个特征时，我创建了新的实例，这些实例不太可能，甚至在身体上是不可能的（例如2米重30公斤的人），但是我使用这些新实例来衡量重要性。换言之，对于相关特征的排列特征重要性，我们考虑了当我们将特征与现实中永远不会观察到的值交换时，模型性能会降低多少。检查特征是否强相关，如果是，请注意特征重要性的解释。

另一个棘手的问题是：添加相关特征可以通过在两个特征之间拆分重要性来降低相关特征的重要性。让我给你举一个我所说的“分裂”特征重要性的例子：我们想预测下雨的可能性，并将前一天早上8点的温度作为特征，以及其他不相关的特征。我训练了一个随机森林，结果发现温度是最重要的特征，一切都很好，第二天晚上我睡得很好。现在想象另一个场景，在其中我将上午9:00的温度作为一个与上午8:00的温度强相关的特征。如果我已经知道上午8点的温度，那么上午9点的温度不会给我提供更多的信息。但是拥有更多的功能总是很好的，对吧？我训练了一个具有两个温度特征和不相关特征的随机森林。一些树木在随机的森林中会在上午8点的温度下生长，另一些则在上午9点的温度下生长，另一些则是同时生长，另一些则没有。两个温度特征在一起比以前的单个温度特征更重要，但不是在重要特征列表的顶部，每个温度现在处于中间。通过引入一个相关的特性，我将最重要的特性从重要性阶梯的顶端踢到了平庸。一方面这很好，因为它只是反映了底层机器学习模型的行为，这里是随机森林。上午8点的温度变得不那么重要了，因为模型现在也可以依赖于上午9点的测量。另一方面，它使特征重要性的解释变得相当困难。假设您想要检查这些特性的测量误差。检查费用昂贵，您决定只检查最重要的3个功能。在第一种情况下，您将检查温度，在第二种情况下，您将不包括任何温度特性，因为它们现在具有相同的重要性。尽管重要性值在模型行为级别上可能是有意义的，但是如果您有相关的特性，则会令人困惑。

5.5.6软件和备选方案

例子中使用了IML R包。dalex r包和python skater模块还实现了模型不可知排列特性的重要性。

一种称为自适应特征重要性算法的算法，用于为重要性提供p值。

1. 《随机森林》，机器学习45（1）。斯普林格：5-32（2001）。
2. Fisher、Aaron、Cynthia Rudin和Francesca Dominici。“从‘Rashomon’的角度来看，模型类依赖：任何机器学习模型类的可变重要性度量。”（2018年）。

5.6全球代理

全局代理模型是一个可解释的模型，它经过训练以近似黑箱模型的预测。我们可以通过解释代理模型得出关于黑盒模型的结论。通过使用更多的机器学习来解决机器学习的可解释性！

5.6.1理论

代理模型也用于工程：如果感兴趣的结果昂贵、耗时或难以测量（例如，因为它来自复杂的计算机模拟），则可以使用廉价、快速的结果代理模型。工程中使用的替代模型和可解释机器学习中使用的替代模型的区别在于，基础模型是机器学习模型（而不是模拟），并且替代模型必须是可解释的。（可解释的）替代模型的目的是尽可能准确地近似基础模型的预测，同时可解释。替代模型的概念可以在不同的名称下找到：近似模型、元模型、响应面模型、仿真器……

关于这个理论：实际上没有太多的理论需要理解代理模型。我们希望在g可解释的约束下，尽可能接近我们的黑箱预测函数f和代替品模型预测函数g。对于函数g，可以使用任何可解释的模型，例如来自的模型。

例如线性模型：

\[g（x）=\beta+\beta x+\ldots+\beta x p\]或决策树：

\[g（x）=\sum m=1 ^mc i \ x \ in r m \

训练代理模型是一种模型不可知方法，因为它不需要任何关于黑盒模型内部工作的信息，只需要访问数据和预测函数。如果将底层机器学习模型替换为另一个模型，您仍然可以使用代理方法。黑盒模型类型和代理模型类型的选择是分离的。

执行以下步骤以获取代理模型：

1. 选择一个数据集X。它可以是用于培训黑盒模型的同一个数据集，也可以是来自同一分发版的新数据集。甚至可以根据应用程序选择数据的子集或点网格。
2. 对于选定的数据集X，获取黑盒模型的预测。
3. 选择可解释的模型类型（线性模型、决策树…）。
4. 在数据集X及其预测上训练可解释模型。
5. 祝贺你！现在您有了一个代理模型。
6. 测量代理模型复制黑盒模型预测的效果。
7. 解释代理模型。

您可能会发现代理模型的方法有一些额外的步骤或稍有不同，但一般的想法通常如本文所述。

测量代理复制黑盒模型的一种方法是R平方测量：

\[r^2=1-\frac sse sst=1-\frac \ sum i=1 ^n（\hat y（i）-\hat（i））^2 sum i=1 ^n（\hat y

^（i）-\bar \hat y）^2 \]

其中\（\hat y（i）\）是对代理模型第i个实例的预测，其中\（\hat（i）\）黑箱模型的预测，以及\（\bar \ hat y）黑箱模型预测的平均值。SSE代表平方和误差，SST代表平方和总误差。R平方测度可以解释为代理模型捕获的方差百分比。如果r平方接近1（=低SSE），那么可解释模型非常接近黑箱模型的行为。如果可解释模型非常接近，您可能希望用可解释模型替换复杂模型。如果r平方接近0（=高SSE），则可解释模型无法解释黑盒模型。

注意，我们没有讨论底层黑盒模型的模型性能，

也就是说，它在预测实际结果时表现得好坏。黑盒模型的性能在训练代理模型中不起作用。代理模型的解释仍然有效，因为它声明了模型而不是真实世界。但是，当然，如果黑盒模型不好，对替代模型的解释就变得无关紧要了，因为黑盒模型本身就是无关紧要的。

我们还可以基于原始数据的子集构建一个代理模型，或者重新计算实例。通过这种方式，我们改变了代理模型输入的分布，从而改变了解释的焦点（然后它不再是真正的全局性的）。如果我们用特定的数据实例对数据进行局部加权（实例越接近所选实例，它们的权重越高），我们就得到一个局部代理模型，它可以解释实例的个别预测。阅读5.6.2示例中有关本地模型的更多信息

为了演示代理模型，我们考虑回归和分类示例。

首先，我们训练一个支持向量机来预测给定的天气和日历信息。由于支持向量机的解释性不强，因此我们用购物车决策树训练一个代理作为可解释模型来近似支持向量机的行为。

图5.31：代理树的终端节点，它近似于自行车租赁数据集上训练的支持向量机的预测。节点中的分布表明，当温度高于13摄氏度时，以及当2年内的某一天晚些时候（临界点为435天），代理树预测租用自行车的数量会更高。

代理模型的r平方（方差解释）为0.77，这意味着它很好地近似了底层的黑箱行为，但并不完美。如果合适的话，我们可以扔掉支持向量机，用树代替。

在我们的第二个例子中，我们预测随机森林的概率。再次，我们用原始数据集训练决策树，但将随机森林的预测作为结果，而不是数据中的真实类（健康与癌症）。

图5.32：一个代理树的终端节点，它近似于一个随机森林对宫颈癌数据集的预测。节点中的计数显示节点中黑盒模型分类的频率。

代理模型的r平方（方差解释）为0.2，这意味着它不能很好地近似随机森林，在得出复杂模型的结论时，我们不应该过度解释树。

5.6.3优势

代理模型方法是灵活的：可以使用来自的任何模型。这也意味着您不仅可以交换可解释模型，还可以交换底层的黑盒模型。假设您创建了一些复杂的模型，并将其解释给公司中的不同团队。一个团队熟悉线性模型，另一个团队可以理解决策树。您可以为原始黑盒模型训练两个代理模型（线性模型和决策树），并提供两种解释。如果你找到一个性能更好的黑盒模型，你不需要改变你的解释方法，因为你可以使用同一类的代理模型。

我认为这种方法是非常直观和直接的。这意味着它很容易实现，也很容易向不熟悉数据科学或机器学习的人解释。

使用r平方度量，我们可以很容易地度量我们的代理模型在近似黑箱预测方面有多好。

5.6.4缺点

您必须知道，您得出的结论是关于模型的，而不是关于数据的，因为代理模型从未看到真正的结果。

目前还不清楚R平方的最佳截止点是什么，以便确信代理模型与黑盒模型足够接近。解释了80%的差异？50%？99%？

我们可以测量代理模型与黑盒模型之间的距离。让我们假设我们不是很接近，而是足够接近。对于数据集的一个子集，可解释模型可能非常接近，但对于另一个子集，可解释模型可能存在很大差异。在这种情况下，对简单模型的解释对所有数据点都不一样好。

作为代理选择的可解释模型有其所有的优点和缺点。

有些人认为，一般来说，没有本质上可解释的模型（甚至包括线性模型和决策树），甚至有一个可解释性的错觉也是危险的。如果你同意这个观点，那么这个方法当然不适合你。

5.6.5软件

我使用IML R包作为例子。如果您可以训练机器学习模型，那么您应该能够自己实现代理模型。简单地训练一个可解释的模型来预测黑箱模型的预测。

5.7局部替代物（石灰）

局部代理模型是可解释的模型，用于解释黑盒机器学习模型的个别预测。局部可解释模型不可知论解释（LIME）是作者提出的一种局部替代模型的具体实现。代理模型被训练为近似底层黑盒模型的预测。Lime没有培训全球替代模型，而是专注于培训本地替代模型来解释个人预测。

这个想法很直观。首先，忘记培训数据，想象一下你只有黑匣子模型，在那里你可以输入数据点并得到模型的预测。你可以随时探测盒子。您的目标是了解为什么机器学习模型

一定的预测。Lime测试当您将数据的变化输入机器学习模型时，预测会发生什么。Lime生成一个新的数据集，该数据集由排列的样本和黑箱模型的相应预测组成。在这个新的数据集上，Lime然后训练一个可解释的模型，该模型通过采样实例与相关实例的接近程度来加权。可解释模型可以是例如或中的任何内容。学习模型应该是机器学习模型局部预测的一个很好的近似值，但它不一定是一个好的全局近似值。这种精度也称为局部保真度。

在数学上，具有可解释性约束的局部替代模型可以表示为：

\[\text解释（x）=\arg\min g \ in g l（f，g，\pi x）+\omega（g）\]

举例来说，x的解释模型是g模型（例如线性回归模型），该模型最小化损失l（例如均方误差），该模型测量解释与原始模型f（例如xgboost模型）的预测的接近程度，而模型复杂性（ω（g））保持在较低水平（例如。g.喜欢较少的功能）。G是一系列可能的解释，例如所有可能的线性回归模型。邻近度度量\（\pi\_x\）定义了我们在解释时考虑到的实例x周围的邻居有多大。实际上，石灰只能优化损失部分。用户必须确定复杂性，例如通过选择线性回归模型可能使用的最大特征数。

培训本地代理模型的方法：

* 选择您感兴趣的实例，对其黑盒预测进行解释。
* 扰乱你的数据集，得到这些新点的黑匣子预测。
* 根据新样品与相关实例的接近程度对其进行称重。
* 在数据集上训练加权的、可解释的模型。
* 通过解释局部模型来解释预测。

例如，在和中的当前实现中，可以选择线性回归作为可解释的代理模型。事先，您必须选择k，即您希望在可解释模型中具有的功能的数量。K越低，解释模型就越容易。较高的k值可能产生高保真度的模型。有几种方法可以训练具有K特征的模型。一个好的选择是一个具有高正则化参数\（\lambda\）的lasso模型生成一个没有任何特性的模型。通过用一个接一个的缓慢下降的\（\lambda\）重新训练lasso模型，这些特性得到了不同于零的重量估计。如果有K

功能在模型中，您已达到所需的功能数量。其他策略是向前或向后选择特性。这意味着您要么从完整的模型（=包含所有特性）开始，要么从只有截距的模型开始，然后测试添加或删除哪个特性会带来最大的改进，直到达到具有k特性的模型为止。

如何获得数据的变化？这取决于数据类型，可以是文本、图像或表格数据。对于文本和图像，解决方案是打开或关闭单个单词或超级像素。在表格数据的情况下，Lime通过单独干扰每个特征，从正态分布中提取特征的平均值和标准偏差来创建新的样本。

5.7.1表格数据用石灰

表格数据是以表格形式出现的数据，每一行代表一个实例，每一列代表一个特性。石灰样品不是在感兴趣的情况下采集的，而是从培训数据的质量中心采集的，这是有问题的。但是，它增加了一些样本点预测结果与感兴趣的数据点不同的可能性，并且莱姆至少可以学到一些解释。

最好直观地解释采样和本地模型培训的工作原理：

图5.33：表格数据的石灰算法。a）随机森林预测，给出特征x1和x2。预测类：1（深色）或0（浅色）。b）兴趣实例（大点）和从正态分布（小点）中采样的数据。c）为相关实例附近的点指定更高的权重。d）网格的符号显示了从加权样本中本地学习的模型的分类。白线表示决策边界（P（class=1）=0.5）。

像往常一样，魔鬼在细节上。在一个点周围定义一个有意义的邻域是困难的。Lime目前使用指数平滑内核来定义邻域。平滑内核是一个函数，它接受两个数据实例并返回一个邻近度量。内核宽度决定了邻域的大小：一个较小的内核宽度意味着一个实例必须非常接近于影响本地模型，一个较大的内核宽度意味着距离较远的实例也会影响模型。如果你看一下，你会发现它使用了指数平滑内核（在标准化数据上），内核宽度是训练数据列数的平方根的0.75倍。它看起来像是一行无辜的代码，但它就像一头大象坐在你的客厅里，旁边是你从祖父母那里得到的好瓷器。最大的问题是我们没有找到最佳内核或宽度的好方法。而0.75是从哪里来的呢？在某些情况下，您可以通过更改内核宽度来轻松地转换您的解释，如下图所示：

图5.34：实例x=1.6的预测说明。黑匣子模型的预测取决于一个单一的特征，显示为一条粗线，数据的分布显示为地毯。计算了三种不同核宽的局部代理模型。所得到的线性回归模型取决于内核宽度：对于x=1.6，特征是否有负的、正的或没有影响？

该示例仅显示一个功能。在高维特征空间中，情况会变得更糟。距离测量是否应平等对待所有特征也很不清楚。功能x1的距离单位是否与功能x2的距离单位相同？距离测量是非常任意的，不同维度的距离（又称特征）可能根本无法比较。

## 5.7.1.1示例

让我们看一个具体的例子。我们回到并将预测问题转化为一个分类：在考虑到自行车租赁随着时间的推移变得越来越流行的趋势后，我们想知道某一天租赁的自行车数量是否会高于或低于趋势线。你也可以将“高于”解释为高于自行车平均数，但要根据趋势进行调整。

首先，我们在分类任务中训练一个有100棵树的随机森林。根据天气和日历信息，租赁自行车的数量将在哪一天高于无趋势平均值？

创建的解释具有两个功能。针对具有不同预测类的两个实例训练的稀疏局部线性模型的结果：

图5.35：自行车租赁数据集两个实例的石灰说明。气温升高和天气状况良好对预测有积极影响。X轴显示特征效果：权重乘以实际特征值。

从图中可以清楚地看出，分类特征比数字特征更容易解释。一种解决方案是将数字特征分类到容器中。

5.7.2文本用石灰

文本的石灰与表格数据的石灰不同。数据变化的产生方式不同：

从原始文本开始，通过从原始文本中随机删除单词来创建新文本。数据集用每个字的二进制特性表示。如果包含相应的单词，则功能为1；如果已删除，则功能为0。

## 5.7.2.1示例

在本例中，我们将其分类为垃圾邮件或普通邮件。

黑盒模型是一种基于文档词矩阵训练的深层决策树。每个注释是一个文档（=一行），每个列是给定单词的出现次数。短决策树很容易理解，但在这种情况下，树很深。此外，代替这棵树，可能有一个反复出现的神经网络或支持向量机训练的字嵌入（抽象向量）。让我们看看这个数据集的两条注释和相应的类（1表示垃圾邮件，0表示普通注释）：

## 内容类

帕西是个好人0

173圣诞歌曲请访问我的频道！；1）

下一步是创建本地模型中使用的数据集的一些变体。例如，其中一条评论的一些变化：

## 为圣诞节而来的探视频道！；）探针重量

21 0 1 1 0 0 1 0.17 0.57

30 1 1 1 1 0 1 0.17 0.71

41 0 0 1 1 1 0.99 0.71

51 0 1 1 1 1 0.99 0.86

60 1 1 1 0 0 1 0.17 0.57

每列对应句子中的一个词。每行都是一个变体，1表示单词是此变体的一部分，0表示单词已被删除。其中一个变奏曲的对应句子是“圣诞歌拜访我的；）”。“prob”列显示每个句子变化的垃圾邮件预测概率。“权重”列显示变化与原句的接近程度，计算为1减去被删除的词的比例，例如，如果7个词中有1个被删除，接近程度为1-1/7=0.86。

下面是两个句子（一个垃圾邮件，一个没有垃圾邮件），它们的估计局部权重由Lime算法找到：

## caseblabel\_probfeature feature\_weight

1 0.1701170好0.000000

1 0.1701170 A 0.000000

1. 0.1701170为0.000000
2. 0.9939024频道！6.180747年

0.000000为2 0.9939024

2 0.9939024；）0.000000

“频道”一词表示垃圾邮件的可能性很高。对于非垃圾邮件注释，没有估计非零权重，因为无论删除哪个单词，预测类都保持不变。

5.7.3图像用石灰

这一部分是由维伦娜·霍恩施密德写的。

图像的石灰与表格数据和文本的石灰不同。直观地说，干扰单个像素没有多大意义，因为许多多个像素对一个类有贡献。随机改变单个像素可能不会改变太多的预测。因此，通过将图像分割为“超级像素”并关闭或打开超级像素，可以创建图像的变化。超级像素是具有相似颜色的互连像素，可以通过将每个像素替换为用户定义的颜色（如灰色）来关闭超级像素。用户还可以指定在每个排列中关闭超像素的概率。

## 5.7.3.1示例

由于图像解释的计算速度相当慢，因此包含了一个预先计算的示例，我们还将使用它来显示该方法的输出。解释可以直接显示在图像样本上。因为每个图像可以有几个预测标签（按概率排序），所以我们可以解释前n个标签。在下面的图片中，前三个预测是电吉他、原声吉他和拉布拉多。

图5.36：谷歌的初始神经网络对图像分类前三类的石灰解释。示例取自石灰纸（Ribeiro等人，2016）。

第一种情况的预测和解释是非常合理的。电吉他的第一个预测当然是错误的，但解释表明神经网络的行为仍然是合理的，因为识别出的图像部分表明这可能是电吉他。

5.7.4优势

即使您替换了底层机器学习模型，您仍然可以使用相同的本地可解释模型进行解释。假设观察解释的人最了解决策树。因为您使用本地代理模型，所以您可以使用决策树作为解释，而不必实际使用决策树来进行预测。例如，您可以使用SVM。如果结果证明XGBoost模型工作得更好，那么您可以替换SVM，并将其用作决策树来解释预测。

本地代理模型受益于培训和解释可解释模型的文献和经验。

当使用套索或短树时，产生的解释很短（=选择性），可能具有对比性。因此，他们做到了。这就是为什么我在接受解释的人是外行或时间很少的人的应用程序中看到了更多的石灰。对于完整的属性来说，这是不够的，所以我在法规遵从性场景中看不到石灰，在这些场景中，法律可能要求您完全解释一个预测。此外，对于调试机器学习模型，有所有的原因而不是少数是有用的。

Lime是少数几种适用于表格数据、文本和图像的方法之一。

保真度度量（可解释模型与黑箱预测的近似程度）使我们很好地了解了可解释模型在解释感兴趣的数据实例附近的黑箱预测方面的可靠性。

石灰是用python（and）和r（and）实现的，并且非常容易使用。

使用本地代理模型创建的解释可以使用原始模型以外的其他功能。与其他方法相比，这可能是一个很大的优势，尤其是在原始特性无法进行解释的情况下。文本分类器可以依靠抽象的单词嵌入作为特征，但解释可以基于句子中单词的存在或不存在。回归模型可以依赖某些属性的不可解释转换，但可以使用原始属性创建解释。

5.7.5缺点

在使用表格数据时，正确定义邻域是一个很大的、尚未解决的问题。在我看来，这是石灰最大的问题，也是我建议只小心使用石灰的原因。对于每个应用程序，您必须尝试不同的内核设置，并亲自查看解释是否有意义。不幸的是，这是我能给出的找到好的内核宽度的最佳建议。

在当前石灰的实施中，可以改进取样。数据点从高斯分布中取样，忽略特征之间的相关性。这可能导致不太可能的数据点，然后可以用来学习本地解释模型。

解释模型的复杂性必须提前定义。这只是一个小小的抱怨，因为最终用户必须定义忠诚度和稀疏度之间的折衷。

另一个真正大的问题是解释的不稳定性。作者在一篇文章中指出，在模拟环境中，两个非常接近的点的解释有很大的不同。而且，根据我的经验，如果你重复取样过程，那么得出的解释可能会不同。不稳定意味着很难相信解释，你应该非常挑剔。

结论：以石灰为具体实施对象的局部替代模型具有很好的应用前景。但该方法仍处于开发阶段，需要解决许多问题才能安全应用。

1. Ribeiro、Marco Tulio、Sameer Singh和Carlos Guestrin。“我为什么要相信你？：解释任何分类器的预测〉，《第22届ACM Sigkdd知识发现和数据挖掘国际会议论文集》。ACM（2016年）。
2. Alvarez Melis、David和Tommi S.Jaakkola。“关于解释方法的稳健性”，ARXIV预印ARXIV:1806.08049（2018）。

5.8夏普利值

一个预测可以通过假设实例的每个特征值都是一个“玩家”来解释，在一个预测是支出的游戏中。沙普利值——来自联合博弈理论的一种方法——告诉我们如何在特征之间公平地分配“支出”。

5.8.1总体思路

假设以下情况：

你已经训练了一个机器学习模型来预测公寓价格。对于某个公寓，它预测30万欧元，你需要解释一下这个预测。公寓面积为50米，2楼为IS2，附近有公园，禁止养猫：

图5.37：一套5000万平方米的公寓，附近有公园和禁猫区，预计价格为2 300000欧元。我们的目标是解释这些特征值如何对预测做出贡献。

所有公寓的平均预测值为31万欧元。与平均预测相比，每个特征值对预测的贡献有多大？

对于线性回归模型，答案很简单。每个特征的效果是特征的权重乘以特征值。这仅仅是因为模型的线性度。对于更复杂的模型，我们需要一个不同的解决方案。例如，建议本地模型来估计效果。另一种解决方案来自合作博弈理论：沙普利值，由沙普利（1953）创造，是一种根据玩家对总支出的贡献分配支出给玩家的方法。玩家在联盟中合作，并从这种合作中获得一定的收益。

玩家？游戏？支出？机器学习预测和可解释性有什么联系？“游戏”是数据集单个实例的预测任务。“增益”是这个实例的实际预测减去所有实例的平均预测。“玩家”是实例的特征值，它们协同工作以获得收益（=预测某个值）。在我们的公寓示例中，功能值“公园附近”、“禁止猫”、“50区”和“2楼”共同实现了300000欧元的预测。我们的目标是解释实际预测（300000欧元）和平均预测（310000欧元）之间的差异：差异-10000欧元。

答案可能是：附近的公园捐款3万欧元；50号的捐款1万欧元；2楼的捐款0欧元；禁止猫捐款5万欧元。捐款总计10000欧元，最终预测减去平均预测房价。

我们如何计算一个特性的夏普利值？

沙普利值是特征值在所有可能的联合体中的平均边际贡献。现在全部清除？

在下图中，我们评估了当添加到附近公园和50号大小的联盟中时，禁止猫功能值的贡献。我们通过从数据中随机抽取另一套公寓，并将其值用于楼层特征，模拟出只有附近的公园、Catbaned和50号才是一个联合体。二楼的价值被随机抽取的一楼所取代，然后我们预测这个组合的公寓价格（310000欧元）。在第二步中，我们将禁止加入联盟的猫从随机抽取的公寓中删除，将其替换为允许/禁止使用的猫功能的随机值。在这个例子中，它被加泰罗尼亚化了，但它可能又被禁止了。我们预测附近公园联盟和50号（32万欧元）的公寓价格。禁止使用猫的捐款为31万欧元-32万欧元=10万欧元。这个估计值取决于作为cat和floor特征值的“捐赠者”的随机抽取公寓的值。如果我们重复这个抽样步骤并平均贡献，我们将得到更好的估计。

图5.38：一个重复的样本来估计当加入附近公园和50区联盟时禁止猫对预测的贡献。

我们对所有可能的联盟重复这个计算。沙普利值是所有可能联盟的边际贡献的平均值。计算时间随特征个数呈指数增长。保持计算时间可控的一个解决方案是只计算可能联合的几个样本的贡献。

下图显示了确定禁用的cat的shapley值所需的所有特征值组合。第一行显示没有任何特征值的联合。第二行、第三行和第四行显示了不同的联盟，联盟规模不断扩大，以“”分隔。总之，以下联盟是可能的：

* 没有功能值
* 停在附近
* 尺寸-50
* 二楼
* 附近停车+50码
* 附近停车+二楼
* 尺寸-50+二楼
* 将车停在+50+二楼附近。

对于每一个联合体，我们计算出有或无特征值cat被禁止的预测公寓价格，并利用差异得到边际贡献。沙普利值是边际贡献的（加权）平均值。我们用公寓数据集中的随机特征值替换不在组合中的特征值，以从机器学习模型中获得预测。

图5.39：计算cat禁止特征值的精确shapley值所需的所有8个联合。

如果我们估计所有特征值的沙普利值，我们得到预测值在特征值之间的完整分布（减去平均值）。

5.8.2示例和解释

特征值j的shapley值的解释是：与数据集的平均预测相比，第j个特征的值对这个特定实例的预测有贡献。

沙普利值适用于分类（如果我们处理概率）和回归。

我们使用shapley值分析随机森林模型预测的预测：

图5.40：宫颈癌数据集中女性的夏普利值。预测值为0.53，这名妇女的癌症概率比平均预测值0.03高0.51。诊断出的性病数量增加的概率最大。贡献之和产生实际和平均预测之间的差异（0.51）。

我们还训练了一个随机森林，根据天气和日历信息，预测一天租用自行车的数量。为特定日期的随机森林预测创建的解释：

图5.41:285天的沙普利值。预计有2475辆出租自行车，这一天比平均预计的4516辆低-2041辆。天气状况和湿度的影响最大，负向贡献最大。这一天的温度有一个积极的贡献。沙普利值之和产生实际和平均预测的差异（-2041）。

注意正确地解释夏普利值：夏普利值是一个特征值对不同组合预测的平均贡献。当我们从模型中删除特征时，shapley值并不是预测的差异。

5.8.3详细的夏普利值

本节将更深入地探讨对好奇读者的夏普利值的定义和计算。如果您对技术细节不感兴趣，请跳过本节，直接转到“优缺点”。

我们对每个特征如何影响数据点的预测感兴趣。在线性模型中，很容易计算个体效应。以下是一个数据实例的线性模型预测的样子：

\[\hat f（x）=\beta+\beta 1 x 1+ldots+\beta p x p \]

其中x是要计算其贡献的实例。每个\（x\_j\）都是一个特征值，j=1，…，p.（\beta\_j\）是与特征j相对应的权重。

第j个特征对预测的贡献（phi-j）为123;f（x）（x）的预测，其贡献为（f j j x j-e（\βj 123; 123; j j 123; j j \125；）\]

其中\（e（\beta jx j）是特征j的平均效应估计值。贡献是特征效应减去平均效应的差额。好极了！现在我们知道每个特性对预测的贡献有多大。如果我们在一个实例中合计所有特性贡献，结果如下：

\[\begin align\*\ sum j=1 ^ p \ phi j（\hat f）=&\sum j=1 ^p（\beta j x j-e（\beta j x j））。\

\（\beta U 0+\sum j=1 \125;^p\beta j x\_j）-（\beta U 0+\sum j=1 \^ 123; 123; 123; 123; 123; 123; 123; \123; j \）e（\beta 123; \\\\]

这是数据点x的预测值减去平均预测值。功能贡献可以是负的。

我们能为任何型号的汽车做同样的工作吗？将它作为一个模型不可知论工具是很好的。由于在其他模型类型中我们通常没有类似的权重，所以我们需要一个不同的解决方案。

帮助来自意想不到的地方：合作博弈论。沙普利值是一种计算任何机器学习模型单个预测的特征贡献的解决方案。

## 5.8.3.1夏普利值

沙普利值通过S中玩家的值函数val定义。

特征值的沙普利值是其对支出的贡献，对所有可能的特征值组合进行加权和求和：

\[\phi\_j（val）=\sum\_s\ substeq\ x\_1，\ldots，x\_p \setminus\ x\_j\ \frac s！\左（P-S-1\右）！{p！}

\左（Val\左（S\杯\ X\ J\右）-Val（S\右）\]

其中s是模型中使用的特征的子集，x是要解释的实例的特征值的向量，p是特征的数量。\（val\_x（s）是对集合S中的特征值的预测，这些特征值在集合S中未包含的特征上被边缘化：

\[val x（s）=\int \hat f（x 1，\ldots，x p）d \mathbb p x \notin s-e x（\hat（f（x））\]

实际上，您对不包含S的每个功能执行多个集成。具体示例：机器学习模型使用4个功能x1、x2、x3和x4，我们评估由功能值x1和x3组成的联盟S的预测：

\[val\_{x}(S)=val\_{x}(\{x\_{1},x\_{3}\})=\int\_{\mathbb{R}}\int\_{\mathbb{R}}\hat{f} (x\_{1},X\_{2},x\_{3},X\_{4})d\mathbb{P}\_{X\_2X\_4}-E\_X(\hat{f}(X))\] This looks similar to the feature contributions in the linear model!

不要被“值”这个词的许多用法混淆：特征值是一个特征和实例的数值或分类值；沙普利值是对预测的特征贡献；价值函数是玩家联盟的支付函数（特征值）。

沙普利值是唯一满足属性效率、对称性、虚拟性和可加性的属性方法，可以看作是公平支出的定义。

## 效率

特征贡献必须加上x和平均值的预测差异。\[\sum\nolimits\_j=1 ^p\phi\_j=\hat f（x）-e\_x（\hat f（x））\]

## 对称性

如果两个特征值j和k对所有可能的联合贡献相等，那么它们的贡献应该相同。如果

\[VAL（S\ CUP\ X\ U J\）=VAL（S\ CUP\ X\ U K\）\]所有

\[s\ substeq \ x 1，\ldots，x p \ setminus \ x j，x k \]然后

\[\phi\_j=\phi\_k\_ \]

## 笨蛋

不改变预测值的特征j（无论添加到哪个特征值组合中）的夏普利值应为0。如果

\[VAL（S \杯\ X \ U J \）=VAL（S）所有

\[s\ substeq \ x 1、\ldots、x p \]然后

\[\phi\_j=0\]

## 可加性

对于一个组合付款val+val的游戏，沙普利值如下：+\[\phi\_j+\phi\_j^+\]

假设您训练了一个随机森林，这意味着预测是许多决策树的平均值。可加性属性保证对于特征值，可以分别计算每棵树的夏普利值，对其进行平均，并获得随机林的特征值的夏普利值。

## 5.8.3.2直觉

理解shapley值的直观方法如下所示：特征值按随机顺序输入房间。房间中的所有功能值都参与游戏（=有助于预测）。特征值的shapley值是当特征值加入时，房间中已经存在的联盟收到的预测的平均变化。

## 5.8.3.3估计沙普利值

所有可能的特征值组合（集合）都必须使用或不使用第j个特征来计算精确的沙普利值。对于多个特征，随着更多特征的添加，可能的联合数呈指数级增加，这个问题的确切解决方案会变得有问题。Strumbelj等人（2014）提出蒙特卡罗采样近似值：

\[\Hat \Phi+J

其中，x的预测是x的预测，但随机数个特征值被随机数据点z的特征值替换为随机数个特征值，除了特征j的各自值之外，x矢量与（x ^ f \\（x ^（x ^ \ ^ m \）也是从样本X中提取，每一个新实例都是一种“弗兰肯斯坦怪物”，由两个实例组合而成。

单特征值的近似夏普利估计：

* 输出：第j个特征值的夏普利值
* 要求：迭代次数m，兴趣实例x，特征索引j，数据矩阵x，机器学习模型f
* 对于所有m=1，…，m：

从数据矩阵X中绘制随机实例Z

选择特征值的随机排列o

订单实例x:\（x\_o=（x（1），\ldots，x（j），\ldots，x（p））\）

订单实例Z:\（Z\_o=（Z（1），\ldots，Z（j），\ldots，Z（p））\）

构建两个新实例

具有特征J:\（X+J=（X（1），\ldots，X（J-1），X（J），Z（J+1），\ldots，Z（P））

无特征J:\（X-J=（X（1），\ldots，X（J-1），Z（J），Z（J+1），\ldots，Z（P））

计算边际贡献：（\phi\_j^ m=\hat f（x+j）-\hat f（x-j））

* 计算夏普利值的平均值为：

首先，选择感兴趣的实例x、特征j和迭代次数m。对于每个迭代，从数据中选择一个随机实例z，并生成特征的随机顺序。通过组合来自感兴趣的实例x和示例z的值，创建了两个新实例。第一个实例\（x+j \）是感兴趣的实例，但功能j的值之前（包括值）的所有值都由示例z的功能值替换。第二个实例\（x-j \）是相似的，但所有的值都按之前的顺序排列，但不包括用样本z中特征j的值替换的特征j。从黑匣子中预测的差异计算出来：

\[\phi\_j^ m=\hat f（x^m+j）-\hat f（x^m-j）\]

所有这些差异均为平均值，结果如下：

\[\phi\_j（x）=\frac 1 m \ sum m=1 ^m\phi\_j^ m \]

通过x的概率分布隐式地对样本进行平均。

必须对每个特征重复此过程，才能获得所有的夏普利值。

5.8.4优势

预测和平均预测之间的差异在实例的特征值（shapley值的效率属性）之间是公平分布的。此属性将shapley值与其他方法（如）区分开来。石灰不能保证预测在各特征之间是公平分布的。沙普利值可能是提供完整解释的唯一方法。在法律要求可解释性的情况下（如欧盟的“解释权”），沙普利值可能是唯一合法的方法，因为它基于一个坚实的理论，并公平地分配影响。我不是律师，所以这只反映了我对要求的直觉。

shapley值允许进行对比解释。您可以将预测与整个数据集的平均预测进行比较，而不是将其与子集或甚至单个数据点进行比较。这种对比也是当地模特如莱姆所没有的。

沙普利值是唯一具有坚实理论的解释方法。公理——效率、对称性、哑性、可加性——给出了一个合理的解释基础。像Lime这样的方法假定机器学习模型的局部线性行为，但是没有理论解释为什么这会起作用。

把预测解释为特征值所玩的游戏，真是让人吃惊。

5.8.5缺点

沙普利值需要大量的计算时间。在99.9%的实际问题中，只有近似解是可行的。精确计算沙普利值的计算成本很高，因为有两种可能的特征值组合，并且必须通过绘制随机实例来模拟AK特征的“缺失”，这会增加沙普利值估计的方差。对联合体的指数数进行抽样，限制迭代次数m，减少m可以减少计算时间，但增加了沙普利值的方差。对于迭代次数没有很好的经验法则，m.m应该足够大以精确估计沙普利值，但是足够小以在合理的时间内完成计算。应该可以根据切尔诺夫边界选择m，但我没有看到任何关于机器学习预测的沙普利值的论文。

沙普利值可能会被误解。特征值的沙普利值不是从模型训练中去除特征后预测值的差值。沙普利值的解释是：给定当前的一组特征值，特征值对实际预测和平均预测之间的差异的贡献就是估计的沙普利值。

如果您寻求稀疏解释（包含很少特性的解释），则shapley值是错误的解释方法。使用shapley值方法创建的解释始终使用所有功能。人类喜欢有选择性的解释，比如石灰。石灰可能是解释外行人必须处理的更好的选择。另一个解决方案是由Lundberg和Lee（2016年）提出的，它基于沙普利值，但也可以提供很少的功能解释。

shapley值为每个特征返回一个简单的值，但没有像lime这样的预测模型。这意味着它不能用于对投入变化预测的变化做出声明，例如：“如果我一年多赚300欧元，我的信用评分将增加5分。”

另一个缺点是，如果要计算新数据实例的shapley值，则需要访问数据。访问预测函数是不够的，因为您需要用随机抽取的数据实例中的值替换感兴趣实例的部分。只有当您可以创建看起来像真实数据实例但不是来自培训数据的实际实例的数据实例时，才能避免这种情况。

与许多其他基于排列的解释方法一样，沙普利值方法在特征相关时会遇到不现实的数据实例。为了模拟联合中缺少特征值，我们将特征边缘化。这是通过从特征的边缘分布中抽取值来实现的。只要这些功能是独立的，就可以了。当特性是依赖的，那么我们可能会对这个实例没有意义的特性值进行采样。但是我们将使用这些来计算特性的夏普利值。据我所知，对于沙普利的价值观，没有研究这意味着什么，也没有关于如何解决它的建议。一种解决方案可能是将相关特征排列在一起，并为它们获得一个相互的沙普利值。或者，可能需要调整采样过程来考虑特征的依赖性。

5.8.6软件和备选方案

沙普利值在IML R包中实现。

shap是shapley值的另一种形式，在python包shap中实现。shap将shapley值方法转化为一个优化问题，并使用一个特殊的核函数来度量数据实例的接近性。shap的结果是稀疏的（许多shapley值估计为零），这是与经典shapley值最大的区别。

另一种方法叫做分解，它在分解R包中实现。分解还显示了每个特征对预测的贡献，但是逐步计算它们。让我们重用这个游戏类比：我们从一个空团队开始，添加对预测贡献最大的特性值，然后迭代直到添加所有特性值。每个特性值贡献多少取决于“团队”中已经存在的各自特性值，这是细分方法的最大缺点。它比沙普利值法更快，对于没有交互作用的模型，结果是相同的。

1. 劳埃德S.沙普利，“n人博弈的价值”，《博弈论的贡献》2.28（1953）：307-317。
2. \_Trumbelj、Erik和Igor Kononenko。“用特征贡献解释预测模型和个人预测”，《知识与信息系统》41.3（2014）：647-665。
3. 伦德伯格，斯科特和苏在李。“模型预测解释方法的意外统一”，ARXIV预印ARXIV:1611.07478（2016）。
4. Staniak、Mateusz和Przemyslaw Biecek。“使用实时和分解包的模型预测说明”，ARXIV预印ARXIV:1804.01955（2018）。

第6章基于实例的解释

基于示例的解释方法选择数据集的特定实例来解释机器学习模型的行为或解释基础数据分布。

基于实例的解释大多是模型不可知论，因为它们使任何机器学习模型更易于解释。与模型不可知方法的区别在于，基于示例的方法通过选择数据集的实例来解释模型，而不是通过创建特性摘要（如或）来解释模型。基于实例的解释只有在我们能够以人性化的方式表示数据的实例时才有意义。这对图像很有效，因为我们可以直接查看它们。一般来说，如果实例的特征值包含更多的上下文，这意味着数据具有像图像或文本那样的结构，那么基于示例的方法可以很好地工作。以有意义的方式表示表格数据更具挑战性，因为一个实例可以包含数百或数千个（不太结构化）特性。列出所有用于描述实例的特性值通常不有用。如果只有一小部分特性，或者我们有一种方法来总结一个实例，那么它会很好地工作。

基于实例的解释有助于人类构建机器学习模型的心理模型和机器学习模型所训练的数据。它尤其有助于理解复杂的数据分布。但我所说的基于实例的解释是什么意思呢？我们经常在工作和日常生活中使用它们。让我们从一些例子开始。

医生看到病人有不寻常的咳嗽和轻微的发烧。病人的症状使她想起了几年前的另一个病人，症状相似。她怀疑她现在的病人可能患有同一种疾病，并采集血样来检测这种特定的疾病。

一位数据科学家正在为他的一位客户进行一项新的项目：分析导致键盘生产机器故障的风险因素。数据科学家记得他所做的一个类似的项目，并重用旧项目中的部分代码，因为他认为客户需要相同的分析。

一只小猫坐在一座燃烧的无人居住的房子的窗台上。消防队已经到了，其中一名消防队员想了想，他是否可以冒险进入大楼去救小猫。作为一名消防员，他还记得类似的情况：燃烧缓慢一段时间的旧木屋经常不稳定，最终倒塌。因为这个案子的相似性，他决定不进去，因为房子倒塌的风险太大了。幸运的是，这只小猫跳出窗户，安全着陆，没有人在火灾中受伤。幸福的结局。

这些故事用例子或类比来说明我们人类是如何思考的。基于实例的解释的蓝图是：事物B与事物A和a引起的y相似，所以我预测b也会引起y。隐含地，一些机器学习方法是基于工作实例的。根据预测目标的重要特征中数据点的相似性将数据划分为节点。决策树通过查找相似的实例（=在同一终端节点中）并返回这些实例结果的平均值作为预测值来获取新数据实例的预测。k-最近邻（knn）方法与基于实例的预测明确工作。对于一个新的实例，knn模型定位k-最近的邻居（例如k=3个最近的实例），并返回这些邻居结果的平均值作为预测。knn的预测可以通过返回k邻居来解释，同样，只有当我们有一个很好的方法来表示一个实例时，k邻居才有意义。

本部分各章包括以下基于实例的解释方法：

* + 告诉我们一个实例必须如何改变才能显著改变其预测。通过创建反事实的实例，我们了解了模型是如何做出预测的，并且可以解释个别的预测。
  + 是用来愚弄机器学习模型的反事实。重点是推翻预测，而不是解释它。
  + 是从数据中选择的代表性实例，而批评是那些原型不能很好地表示的实例。
  + 是对预测模型参数或预测本身影响最大的训练数据点。识别和分析有影响的实例有助于发现数据问题，调试模型并更好地了解模型的行为。
  + ：基于实例的（可解释的）机器学习模型。
  1. Aamodt、Agnar和Enric广场。“基于案例的推理：基础问题、方法变化和系统方法。”人工智能通信7.1（1994）：39-59。
  2. Kim、Be、Rajiv Khanna和Oluwasanmi O.Koyejo。“例子不够，要学会批判！对可解释性的批评〉，《神经信息处理系统的进展》（2016年）。

6.1反事实解释

一个反事实的解释描述了一种因果关系的形式：“如果x没有发生，y就不会发生”。例如：“如果我不喝这杯热咖啡，我就不会把舌头烧伤。”事件Y是我烧了舌头；因为X是我喝了一杯热咖啡。反事实思维需要想象一个与观察到的事实相矛盾的假想现实（例如，一个我没有喝热咖啡的世界），因此得名“反事实”。在反事实方面思考的能力使我们人类比其他动物更聪明。

在可解释的机器学习中，反事实的解释可以用来解释个别情况的预测。“事件”是一个实例的预测结果，“原因”是输入到模型中的该实例的特定特征值，“原因”是一个特定的预测。以图形的形式显示，输入和预测之间的关系非常简单：特征值导致预测。

图6.1：机器学习模型输入与预测之间的因果关系，当模型仅被视为一个黑匣子时。输入导致预测（不一定反映数据的真正因果关系）。

即使在现实中，输入和要预测的结果之间的关系可能不是因果关系，我们也可以将模型的输入视为预测的原因。

在这个简单的图中，很容易看出我们如何模拟机器学习模型预测的反事实：我们只需在进行预测之前更改实例的特征值，然后分析预测是如何变化的。我们对预测以相关方式发生变化的情景感兴趣，例如预测类的翻转（例如，接受或拒绝信贷申请）或预测达到某一阈值（例如，癌症的概率达到10%）。对预测的反事实解释描述了将预测更改为预定义输出的特征值的最小更改。

反事实解释方法是模型不可知论，因为它只适用于模型的输入和输出。这种方法也会让人感到宾至如归。

解释可以表示为特征值差异的总结（“更改特征A和B以更改预测”）。但是反事实的解释本身就是一个新的例子，所以它活在这一章中（“从实例x开始，改变a和b以获得反事实的实例”）。与反事实不同，不必是训练数据中的实际实例，而是可以是特征值的新组合。

在讨论如何创建反事实之前，我想讨论一些反事实的用例，以及一个好的反事实解释是怎样的。

在第一个例子中，Peter申请贷款，并被（机器学习驱动的）银行软件拒绝。他想知道为什么他的申请被拒绝，以及如何提高他获得贷款的机会。“为什么”的问题可以被表述为一个反事实：将预测从拒绝变为批准的特征（收入、信用卡数量、年龄等）的最小变化是什么？一个可能的答案是：如果彼得每年多赚1万欧元，他就会得到贷款。或者，如果彼得的信用卡更少，而且5年前没有拖欠贷款，他就会得到贷款。彼得永远不会知道拒绝的原因，因为银行对透明度没有兴趣，但那是另一回事。

在我们的第二个例子中，我们要解释一个模型，它用反事实的解释来预测一个连续的结果。安娜想租出她的公寓，但她不知道要多少钱，所以她决定训练一个机器学习模型来预测租金。当然，因为安娜是一个数据科学家，所以她就是这样解决她的问题的。在输入了所有关于尺寸、位置、是否允许养宠物等细节后，模特告诉她可以收取900欧元的费用。她期望1000欧元或更多，但她相信她的模型，并决定发挥公寓的功能价值，看看她如何提高公寓的价值。她发现这套公寓如果再大1500万欧元，可以以1000多欧元的价格出租。有趣，但非2个可操作的知识，因为她不能扩大她的公寓。最后，通过只调整她控制下的功能值（内置厨房是/否、允许宠物是/否、地板类型等），她发现如果她允许宠物并安装隔热性能更好的窗户，她可以收取1000欧元。安娜凭直觉用反事实来改变结果。

反事实是，因为它们与当前实例形成对比，而且它们是选择性的，这意味着它们通常只关注少量的特性更改。但是，反事实也会受到“拉森效应”的影响。《罗生门》是一部日本电影，其中一个武士的谋杀案是由不同的人讲述的。每一个故事都同样很好地解释了结果，但这些故事相互矛盾。反事实也可能发生同样的情况，因为通常有多种不同的反事实解释。每一个反事实都讲述了一个不同的“故事”，讲述了某个结果是如何达到的。一个反事实可以说改变了特征A，另一个反事实可以说留下了相同的，但改变了特征B，即A

矛盾。这个多个事实的问题可以通过报告所有的反事实的解释来解决，也可以通过有一个标准来评估反事实并选择最好的一个。

说到标准，我们如何定义一个好的反事实解释？首先，反事实解释的用户在一个实例的预测中定义了一个相关的变化（=替代现实），因此一个明显的首要要求是反事实实例尽可能地生成预先定义的预测。不可能总是完全匹配预定义的输出。在一个包含两个类、一个稀有类和一个频繁类的分类环境中，模型总是可以将一个实例分类为频繁类。更改特性值，使预测标签从普通类翻转到罕见类可能是不可能的。因此，我们希望放宽这样一个要求，即反事实的预测结果必须与定义的结果完全对应。在分类示例中，我们可以寻找一个反事实，其中稀有类的预测概率增加到10%，而不是当前的2%。那么问题是，特征的最小变化是什么，使得预测概率从2%变为10%（或接近10%）。另一个质量标准是，关于特征值，反事实应该尽可能类似于实例。这需要在两个实例之间测量距离。反事实不仅要接近原案，而且要尽可能少地改变特征。这可以通过选择适当的距离测量来实现，如曼哈顿距离。最后一个要求是反事实实例应该具有可能的特征值。如果一套公寓的大小为负数或房间数设置为200，那么就没有必要对租金示例做出反事实的解释。根据数据的联合分布，如10间房20米的公寓不应被视为反事实的解释，当反事实可能出现时，情况会更好。

6.1.1产生反事实解释

一种简单而幼稚的产生反事实解释的方法是通过试错法进行搜索。这种方法包括随机更改感兴趣的实例的特征值，并在预测所需的输出时停止。比如安娜试图找到一套公寓的版本，她可以收取更多的租金。但是有比尝试和错误更好的方法。首先，我们定义了一个损失函数，它将利益实例、反事实和期望（反事实）结果作为输入。损失衡量反事实的预测结果与预先确定的结果之间的距离，以及反事实与利益实例之间的距离。我们可以使用优化算法直接优化损失，也可以通过搜索实例来优化损失，如“增长球体”方法（请参见）。

在本节中，我将介绍Wachter等人（2017）建议的方法，他们建议将以下损失降至最低。

\[L（x，x^ \ prime，y^ \ prime，lambda）=\lambda\cdot（\hat f（x^ \ prime）-y^ \ prime）^2+d（x，x^ \ prime）\]

第一项是模型预测反事实x'和期望结果y'之间的二次距离，用户必须提前定义。第二个术语是要解释的实例x和反事实x'之间的距离d’，但稍后会详细介绍。参数\（\lambda\）平衡预测中的距离（第一项）和特征值中的距离（第二项）。对给定的\（\lambda\）求解损失，并返回反事实x'。较高的值\（\lambda\）意味着我们更喜欢接近期望结果y的反事实，较低的值意味着我们更喜欢特征值中与x非常相似的反事实x。如果\（\lambda\）非常大，无论预测距离x多远，都将选择预测最接近y'的实例。最终，用户必须决定如何平衡反事实预测与所需结果匹配的要求。t反事实类似于x。该方法的作者建议不要为\（\lambda\）选择一个值，而是选择一个公差\（\epsilon\）来决定反事实实例的预测距离y'的距离。这个约束可以写为：

\[\hat f（x ^ \prime）-y ^ \prime \leq \epsilon \]

为了最小化这个损失函数，可以使用任何合适的优化算法，例如Nelder Mead。如果您可以访问机器学习模型的渐变，那么您可以使用基于渐变的方法，如Adam。要解释的实例x、所需的输出y'和公差参数\（\epsilon\）必须提前设置。对于x'和（局部）最佳反事实x'的损失函数最小化，同时增加\（\lambda\）直到找到足够接近的解（=在公差参数内）。

\[\arg\min\_x^ \ prime \ max \ lambda l（x，x^ \ prime，y^ \ prime，lambda）\]

用于测量实例x和反事实x'之间距离的函数d是曼哈顿距离加权特征，具有反向中值绝对偏差（MAD）。

\[d（x，x^ \ prime）=\sum j=1 ^p \ frac x j-x ^ \ prime mad j \]

总距离是所有P向特征距离的总和，即实例x和反事实x'之间特征值的绝对差异。特征距离通过特征j在数据集上的中值绝对偏差的倒数缩放，定义为：

\[mad\_j=\text中位数1，ldots，n（x i，j-\text中位数1，ldots，n（x l，j）]]

向量的中位数是向量值的一半大，另一半小的值。MAD是特征方差的等价物，但我们不使用平均值作为中心，而使用平方距离求和，而是使用中间值作为中心，并在绝对距离求和。与欧氏距离相比，距离函数具有引入稀疏性的优点。这意味着当较少的特征不同时，两个点之间的距离更近。对于离群值来说，它更强大。使用MAD进行缩放是必要的，以使所有功能达到相同的比例——无论您是以平方米还是平方英尺为单位测量公寓的大小都不重要。

制作反事实的方法很简单：

1. 选择要解释的实例x、所需的结果y、公差（epsilon）和（low）初始值（lambda）。
2. 抽样一个随机的实例作为初始反事实。
3. 以初始抽样反事实为出发点，对损失进行优化。
4. 而\（\hat f（x ^ \ prime）-y ^ \ prime>\epsilon\）：

增加\（\lambda\）。

以当前反事实为起点优化损失。退回减少损失的反事实。

1. 重复步骤2-4并返回反事实列表或最小化损失的列表。

6.1.2示例

这两个例子都来自Wachter等人（2017年）的研究。

在第一个例子中，作者训练了一个三层完全连接的神经网络，根据法学院入学前的平均成绩（GPA）、种族和法学院入学考试成绩，预测法学院学生的第一年平均成绩。目标是为每个回答以下问题的学生找到反事实的解释：输入特性需要如何改变，才能得到0的预测分数？因为分数以前已经被标准化了，所以分数为0的学生和学生的平均分一样好。负分表示低于平均分的结果，正分表示高于平均分的结果。

下表显示了所学的反事实：

## ScoregpalSatracegpa x'Lsat x'race x'

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0.17 3.1条 | 39.0分 | 第3.1条 | 34.0条 | 0 |
| 0.54 3.7条 | 48.0分 | 第3.7条 | 第32.4条 | 0 |
| -0.77 3.3条 | 28.0 1 | 第3.3条 | 33.5条 | 0 |
| -0.83 2.4条 | 28.5一 | 第2.4条 | 35.8条 | 0 |
| -0.57 2.7条 | 18.3 0 | 2.7条 | 第34.9条 | 0 |

第一列包含预测得分，接下来的3列包含原始特征值，最后3列包含导致得分接近0的反事实特征值。前两行是预测高于平均水平的学生，其他三行低于平均水平。前两行的反事实描述了如何改变学生的特征以降低预测分数，另外三个案例则描述了如何改变才能将分数提高到平均水平。提高分数的反事实总是将种族从黑色（用1编码）变为白色（用0编码），这显示了模型的种族偏见。在反事实中，GPA没有改变，但是LSAT改变了。

第二个例子显示了预测糖尿病风险的反事实解释。一个三层的完全连接的神经网络被训练来预测糖尿病的风险，这取决于年龄，体重指数，怀孕次数等等。反事实回答了这个问题：哪些特征值必须改变才能将糖尿病的风险评分提高或降低到0.5？发现以下反事实：

* 人1：如果你的2小时血清胰岛素水平为154.3，你将得到0.51分。
* 人2：如果你的2小时血清胰岛素水平为169.5，你将得到0.51分。
* 人3：如果你的血糖浓度是158.3，你的2小时血清胰岛素水平是160.5，你将得到0.51分。

6.1.3优势

反事实解释的解释非常清楚。如果根据反事实更改实例的特征值，则预测将更改为预定义的预测。没有额外的假设，背景中也没有魔力。这也意味着它不像其他方法那样危险，因为我们还不清楚可以在多大程度上推断出解释的局部模型。

反事实方法创建了一个新的实例，但是我们也可以通过报告哪些特征值发生了变化来总结反事实。这为我们提供了两个报告结果的选项。您可以报告反事实实例，也可以突出显示感兴趣的实例和反事实实例之间已更改的功能。

反事实方法不需要访问数据或模型。它只需要访问模型的预测函数，例如，它也可以通过Web API工作。对于由第三方审计的公司或在不披露模型或数据的情况下为用户提供解释的公司来说，这是很有吸引力的。由于商业秘密或数据保护原因，公司有兴趣保护模型和数据。反事实解释在解释模型预测和保护模型所有者利益之间提供了平衡。

该方法也适用于不使用机器学习的系统。我们可以为任何接收输入和返回输出的系统创建反事实。预测公寓租金的系统也可能包括手写规则，而反事实的解释仍然有效。

反事实解释方法相对容易实现，因为它本质上是一个损失函数，可以用标准优化器库进行优化。必须考虑一些额外的细节，例如将特征值限制在有意义的范围内（例如，只有正的公寓尺寸）。

6.1.4缺点

对于每一个例子，你通常会发现多个反事实的解释（拉森蒙效应）。这是不方便的——大多数人喜欢简单的解释，而不是现实世界的复杂性。这也是一个实际的挑战。例如，我们生成了23个反事实的解释。我们都报告了吗？只有最好的？如果它们都是相对“好”的，但是非常不同呢？对于每个项目，必须重新回答这些问题。有多个反事实的解释也是有利的，因为这样人类就可以选择与他们以前的知识相对应的解释。

对于给定的公差（epsilon），不能保证找到了反事实的实例。这不一定是方法的错误，而是取决于数据。

该方法不能很好地处理多层次的分类特征。该方法的作者建议对分类特征的每个特征值组合分别运行该方法，但是如果您有多个具有多个值的分类特征，这将导致组合爆炸。例如，6个具有10个唯一级别的分类功能意味着100万次运行。Martens等人（2014年）提出了仅针对分类特征的解决方案。一个好的解决方案是使用优化器来解决连续和离散输入混合的问题。

反事实方法缺乏通用的软件实现。方法只有在实现时才有用。幸运的是，它应该很容易实现，希望我可以很快删除这条语句。

6.1.5软件和备选方案

不幸的是，目前没有可用于反事实解释的软件。

Martens等人（2014）提出了一种非常类似的方法来解释文件分类。在他们的工作中，他们重点解释为什么一个文档被或没有被归类为一个特定的类。与本章介绍的方法不同的是，Martens等人（2014年）将重点放在文本分类器上，文本分类器将单词出现作为输入。

另一种搜索反事实的方法是Laugel等人（2017）的生长球体算法。该方法首先在兴趣点周围绘制一个球体，在该球体内采样点，检查其中一个采样点是否产生所需的预测、收缩或扩展SPH。因此，在找到（稀疏的）反事实并最终返回之前。他们在论文中没有使用“反事实”这个词，但方法非常相似。它们还定义了一个有利于反事实的损失函数，并且尽可能少地改变特征值。

他们不直接优化函数，而是建议使用球体进行上述搜索。

图6.2:Laugel等人关于生长球体和选择稀疏反事实的说明。

Al（2017年）。

Ribeiro等人（2018）的锚与反事实相反。锚回答问题：哪些特征足以锚定预测，即更改其他特征不能更改预测？一旦我们找到了作为预测锚定的特性，我们将不再通过改变锚定中未使用的特性来发现反事实的实例。

图6.3:Ribeiro等人（2018年）的锚示例。

1. 瓦赫特、桑德拉、布伦特·米特斯塔特和克里斯·拉塞尔。“不打开黑匣子的反事实解释：自动决策和GDPR。”（2017年）。
2. 马丁斯、大卫和福斯特教务长。“解释数据驱动文档分类。”（2014年）。

[↩](https://christophm.github.io/interpretable-ml-book/counterfactual.html#fnref46)

1. Laugel，Thibault等人“机械学习中基于比较的可解释性的逆向分类”，ARXIV预印ARXIV:1712.08443（2017）。
2. Ribeiro、Marco Tulio、Sameer Singh和Carlos Guestrin。“锚：高精度模型不可知论解释”，AAAI人工智能会议（2018）。

6.2对抗性示例

一个敌对的例子是一个小的，故意的特征干扰，导致机器学习模型作出错误的预测的例子。我建议先阅读这一章，因为它们的概念非常相似。对抗性的例子是反事实的例子，目的是欺骗模型，而不是解释它。

为什么我们对敌对的例子感兴趣？他们不只是好奇的机器学习模型的副产品没有实际意义吗？答案是明确的“不”。对抗性示例使机器学习模型容易受到攻击，如以下场景中所示。

一辆自动驾驶的汽车撞到另一辆车上，因为它忽略了停车标志。有人在标志上放了一张图片，它看起来像一个停车标志，对人类来说有点脏，但设计得像一个停车禁止标志，用于汽车的标志识别软件。

垃圾邮件检测器无法将电子邮件分类为垃圾邮件。垃圾邮件被设计成类似于普通邮件，但目的是欺骗收件人。

一台机器学习型扫描仪在机场扫描手提箱中的武器。为了避免被发现，开发了一把刀，使系统认为它是一把伞。

让我们来看一些创建对抗性示例的方法。

6.2.1方法和示例

有许多技巧可以创建对抗性的例子。大多数方法都建议在将预测转移到期望的（对抗的）结果的同时，将对抗性实例和要操纵的实例之间的距离最小化。有些方法需要访问模型的梯度，当然这些方法只适用于基于梯度的模型，如神经网络，其他方法只需要访问预测函数，这使得这些方法模型不可知。本节中的方法主要集中在基于深度神经网络的图像分类器上，在这方面做了大量的研究，对敌方图像的可视化非常有教育意义。图像的对抗性示例是带有有意扰动像素的图像，目的是在应用期间欺骗模型。这些例子令人印象深刻地展示了深层神经网络如何容易被对人类无害的图像欺骗。如果你还没有看到这些例子，你可能会感到惊讶，因为对人类观察者来说，预测的变化是不可理解的。相反的例子就像是光学错觉，但对机器来说是如此。

## 我的狗出毛病了

Szegedy等人（2013年）在他们的工作“神经网络的有趣特性”中使用了基于梯度的优化方法来寻找深层神经网络的对抗性示例。

图6.4:Szegedy等人（2013年）对Alexnet的敌对示例。左列中的所有图像都已正确分类。中间一列显示添加到图像中的（放大的）错误，以生成右一列中的图像，所有分类（错误）为“鸵鸟”。

这些相反的例子是通过最小化与r有关的以下函数而产生的：

\[损失（\hat f（x+r），l）+c\cdot r \]

在这个公式中，x是一个图像（表示为像素向量），r是对像素的更改以创建一个对抗图像（x+r生成一个新图像），l是所需的结果类，参数c用于平衡图像之间的距离和预测之间的距离。第一项是对抗性实例的预测结果与期望的L类之间的距离，第二项是测量对抗性实例与原始图像之间的距离。该公式几乎与要生成的损失函数相同。R还有其他约束，因此像素值保持在0和1之间。作者建议用盒约束的L-BFGS来解决这个优化问题，这是一种适用于梯度的优化算法。

## 扰动熊猫：快速梯度标记法

Goodfellow等人（2014年发明了用于生成对抗图像的快速梯度符号方法。

梯度符号法利用底层模型的梯度来寻找对立的例子。

原始图像X是通过在每个像素上加上或减去一个小的错误来操作的。我们加或减\（\epsilon\）取决于像素的渐变符号是正的还是负的。在梯度方向上增加误差意味着图像被有意修改，从而导致模型分类失败。

图6.5:Goodfellow等人（2014年）在神经网络中使熊猫看起来像长臂猿。通过将小扰动（中间图像）添加到原始熊猫像素（左图像）中，作者创建了一个敌对的例子，它被归类为长臂猿（右图像），但在人类看来像熊猫。

下面的公式描述了快速梯度符号方法的核心内容：

其中\（\bigstriangledown\_x j\）是模型损失函数相对于原始输入像素向量x的梯度，y是x的真标签向量，而\（\theta\）是模型参数向量。从梯度向量（与输入像素的向量一样长）我们只需要符号：如果像素强度增加损失（模型产生的误差），梯度符号为正（+1），如果像素强度减少损失，则为负（-1）。当神经网络线性处理输入像素强度和类分数之间的关系时，就会出现此漏洞。特别是，倾向于线性化的神经网络结构，如LSTMS、MAXOUT网络、带有RELU激活单元的网络或其他线性机器学习算法（如逻辑回归）易受梯度符号法的影响。攻击是通过外推法进行的。输入像素强度和类分数之间的线性关系导致易受异常值的影响，即通过将像素值移动到数据分布之外的区域，可以欺骗模型。我期望这些对抗性的例子对于给定的神经网络体系结构是非常具体的。但事实证明，你可以重复使用敌对的例子来欺骗在同一个任务上训练了不同体系结构的网络。

Goodfellow等人（2014）建议在培训数据中添加对抗性示例，以学习健壮的模型。

## 水母……不，等等。浴缸：1像素攻击

Goodfellow及其同事（2014年）提出的方法需要改变很多像素，即使只是稍微改变一点。但是如果你只能改变一个像素呢？你能欺骗一个机器学习模型吗？Su等人（2019年）表明，通过改变单个像素实际上可以欺骗图像分类器。

图6.6：通过故意改变一个像素（用圆圈标记），在ImageNet上训练的神经网络被欺骗，从而预测错误的类而不是原始类。Su等人（2019）的研究。

类似于反事实，1像素攻击寻找一个修改过的示例x'接近原始图像x，但将预测改为对抗结果。然而，亲密度的定义不同：只有一个像素可能会改变。1像素攻击使用差分进化来找出要更改的像素以及如何更改。物种的生物进化松散地激发了差异进化。一个被称为候选解决方案的群体一代一代地重组，直到找到解决方案。每个候选解决方案编码一个像素修改，并由五个元素的向量表示：x和y坐标以及红、绿和蓝（rgb）值。例如，搜索从400个候选解决方案（=像素修改建议）开始，并使用以下公式从父代创建新一代候选解决方案（子代）：

\[X I（G+1）=X R1（G）+F\CDOT（X R2（G）+X R3（G））]

其中每个（x\_i）是候选解的元素（x坐标、y坐标、红色、绿色或蓝色），g是当前生成的，f是缩放参数（设置为0.5），r1、r2和r3是不同的随机数。每个新的子候选解决方案依次是一个像素，其中包含五个位置和颜色属性，每个属性都是三个随机父像素的混合。

如果一个候选解决方案是一个对立的示例，意味着它被分类为不正确的类，或者达到了用户指定的最大迭代次数，则停止创建子级。

## 一切都是一台烤面包机：敌对的补丁

我最喜欢的方法之一就是把敌对的例子带到现实中。Brown等人（2017）设计了一个可打印标签，可以贴在物体旁边，使其看起来像图像分类器的烤面包机。干得漂亮！

图6.7：一个贴纸，它使在ImageNet上训练的VGG16分类器将香蕉的图像分类为烤面包机。Brown等人的工作（2017年）。

这种方法不同于迄今为止针对对抗性实例所提出的方法，因为消除了对抗性图像必须非常接近原始图像的限制。相反，该方法将图像的一部分完全替换为可以呈现任何形状的补丁。补丁的图像在不同的背景图像上进行了优化，补丁在图像上的位置不同，有时会移动，有时会变大或变小并旋转，因此补丁可以在许多情况下工作。最后，该优化后的图像可以在野外打印并用于欺骗图像分类器。

## 千万不要把3D打印的乌龟带到枪战中——即使你的电脑认为这是个好主意：强大的对抗性例子

下一种方法是在烤面包机上添加另一个维度：Athalye等人（2017）3印刷了一只乌龟，从几乎所有可能的角度来看，它看起来就像一把步枪。是的，你读对了。对人类来说，一个看起来像乌龟的物体在计算机上看起来就像一把步枪！

图6.8：一只3D打印乌龟，被TensorFlow的标准预先训练识别为步枪。

inceptionv3分类器。Athalye等人的工作（2017年）

作者发现了一种在3D中为二维分类器创建对抗性示例的方法，这种分类器在转换过程中具有对抗性，例如旋转乌龟、放大等所有可能性。其他方法，如快速梯度法，在图像旋转或视角变化时不再有效。Athalye等人（2017）提出了期望过度转换（EOT）算法，这是一种生成对抗性示例的方法，甚至在图像转换时也能工作。EOT背后的主要思想是通过许多可能的转换来优化敌对的例子。在给定可能变换的选定分布的情况下，EOT不会将敌对示例和原始图像之间的距离最小化，而是将两个示例之间的预期距离保持在某个阈值以下。转换下的预期距离可写为：

\[\mathbb e t\sim t[d（t（x^\prime），t（x））]\

其中x是原始图像，t（x）是转换后的图像（例如旋转），x'是敌对的例子，t（x'）是转换后的版本。除了处理转换的分布之外，eot方法还遵循熟悉的模式，将搜索敌对示例作为优化问题进行框架化。我们试图找到一个对抗性的例子x'来最大化所选类（y\_t）的概率（例如“来复枪”）在可能的转换分布t：

\[\arg\max\_x ^ \prime \mathbb e t \sim t[日志p（y\_t t（x ^ \prime））]\]

在限制条件下，敌对示例x'和原始图像x之间所有可能转换的预期距离保持在某个阈值以下：

\[\mathbb e t\sim t[d（t（x^\prime），t（x））]<\epsilon\quad\text and \quad x \in[0,1]^d\]

我认为我们应该关注这种方法的可能性。其他方法是基于数字图像的处理。然而，这些3D打印的、强大的对抗性示例可以插入任何真实场景中，并欺骗计算机对对象进行错误分类。让我们把它转过来：如果有人制造了一把看起来像乌龟的步枪怎么办？

## 蒙蔽的对手：黑匣子攻击

想象一下下面的场景：我让您通过Web API访问我的优秀图像分类器。您可以从模型中获得预测，但是您没有访问模型参数的权限。从您的沙发方便，您可以发送数据和我的服务答案与相应的分类。大多数的对抗性攻击并不能在这种情况下工作，因为它们需要访问底层深层神经网络的梯度来找到对抗性的例子。Papernot和同事（2017年）表明，在没有内部模型信息和无法访问培训数据的情况下，可以创建具有对抗性的示例。这种（几乎）零知识攻击称为黑箱攻击。

工作原理：

1. 从一些与训练数据来自同一域的图像开始，例如，如果要攻击的分类器是数字分类器，则使用数字图像。需要了解该领域的知识，但不需要访问培训数据。
2. 从黑盒中获取当前图像集的预测。
3. 在当前图像集（例如神经网络）上训练代理模型。
4. 使用启发式方法创建一组新的合成图像，该方法检查当前图像集，在哪个方向上操作像素以使模型输出具有更大的方差。
5. 对预定义的epoch数重复步骤2到4。
6. 使用快速梯度法（或相似法）为代理模型创建对抗性示例。
7. 用敌对的例子攻击原始模型。

代理模型的目的是近似黑箱模型的决策边界，但不一定达到相同的精度。

作者通过攻击在各种云机器学习服务上训练的图像分类器来测试这种方法。这些服务在用户上传的图像和标签上训练图像分类器。软件会自动训练模型（有时使用用户未知的算法），并部署它。然后，分类器对上传的图像进行预测，但无法检查或下载模型本身。作者能够为不同的提供者找到对抗性的例子，多达84%的对抗性的例子被错误分类。

如果要欺骗的黑盒模型不是神经网络，这种方法甚至可以工作。这包括没有梯度的机器学习模型，例如决策树。

6.2.2网络安全观

机器学习处理已知的未知：从已知的分布预测未知的数据点。对攻击的防御处理未知的未知：从未知的敌方输入分布中可靠地预测未知的数据点。随着机器学习被集成到越来越多的系统中，例如自动车辆或医疗设备，它们也成为攻击的入口点。即使机器学习模型在测试数据集上的预测是100%正确的，也可以发现相反的例子来欺骗模型。针对网络攻击的机器学习模型防御是网络安全领域的一个新的组成部分。

Biggio等人（2018年）对本节所依据的对抗性机器学习的十年研究进行了很好的回顾。网络安全是一场军备竞赛，在这场竞赛中，攻击者和防御者一次又一次地相互竞争。

网络安全有三条黄金法则：1）了解你的对手2）积极主动3）保护你自己。

不同的应用程序有不同的对手。试图通过电子邮件骗取他人钱财的人是用户和电子邮件服务提供商的对手代理。提供者想要保护他们的用户，这样他们就可以继续使用他们的邮件程序，攻击者想让人们给他们钱。了解你的对手意味着了解他们的目标。假设您不知道这些垃圾邮件发送者存在，并且电子邮件服务的唯一滥用是发送盗版音乐副本，那么防御将有所不同（例如，扫描附件以获取受版权保护的材料，而不是分析文本中的垃圾邮件指标）。

积极主动意味着积极测试和识别系统的弱点。当你积极地试图用敌对的例子欺骗模型，然后进行防御时，你是主动的。使用解释方法来了解哪些特征是重要的，以及特征如何影响预测，也是了解机器学习模型弱点的一个积极步骤。作为数据科学家，你是否相信你的模型在这个危险的世界里，而从来没有超越过对测试数据集的预测能力？您是否分析了模型在不同场景中的行为，确定了最重要的输入，检查了一些示例的预测解释？你有没有试着找到对抗性的输入？机器学习模型的可解释性在网络安全中起着重要作用。与主动性相反，被动性意味着等待系统被攻击，然后才了解问题并安装一些防御措施。

我们如何保护我们的机器学习系统不受敌对例子的影响？一种主动的方法是对分类器进行反复的再培训，其中包含对抗性示例，也称为对抗性训练。其他方法是基于博弈论的，例如学习特征的不变变换或鲁棒优化（正则化）。另一种方法是使用多个分类器，而不是一个分类器，让他们投票预测（合集），但这并不能保证工作，因为他们都可能遭受类似的敌对例子。另一个也不太好用的方法是梯度屏蔽，它通过使用最近邻分类器而不是原始模型来构造一个没有有用梯度的模型。

我们可以通过攻击者对系统的了解程度来区分攻击类型。攻击者可能具有完善的知识（白盒攻击），这意味着他们了解模型的类型、参数和训练数据等所有信息；攻击者可能具有部分知识（灰盒攻击），这意味着他们可能只知道模型的特征表示和类型。已使用的模型，但无法访问培训数据或参数；攻击者可能没有任何知识（黑盒攻击），这意味着他们只能以黑盒方式查询模型，但无法访问培训数据或模型参数信息。根据信息的级别，攻击者可以使用不同的技术攻击模型。正如我们在示例中看到的，即使在黑盒情况下，也可以创建敌对的示例，因此隐藏有关数据和模型的信息不足以抵御攻击。

考虑到攻击者和防御者之间的猫捉老鼠游戏的性质，我们将在这一领域看到许多发展和创新。想想不断演变的许多不同类型的垃圾邮件。提出了针对机器学习模型的新攻击方法，并针对这些新攻击提出了新的防御措施。更强大的攻击是为了躲避最新的防御等，无限期。通过这一章，我希望能够使您对对抗性例子的问题更加敏感，并且只有通过积极地研究机器学习模型，我们才能发现并弥补弱点。

1. Szegedy，Christian等人“神经网络的有趣特性”，arxiv预印arxiv:1312.6199（2013）。
2. Goodfellow、Ian J、Jonathon Shlens和Christian Szegedy。“解释和利用个人示例”，arxiv预印arxiv:1412.6572（2014）。
3. 苏、加威、达尼洛·瓦斯卡塞洛斯·瓦加斯和樱井。“愚弄深度神经网络的单像素攻击”，《IEEE进化计算汇刊》（2019年）。
4. Brown，Tom B.等人“对抗补丁。”arxiv预印arxiv:1712.09665（2017）。
5. 阿塔利、安尼和伊利亚·萨茨基弗。“合成强大的对抗性示例”，arxiv预印arxiv:1707.07397（2017）。
6. Papernot，Nicolas等人“针对机器学习的实际黑盒攻击”，2017年ACM亚洲计算机与通信安全会议进程。ACM（2017年）。
7. Biggio、Battista和Fabio Roli。“野性模式：对抗性机器学习兴起十年后”，《模式识别》84（2018）：317-331。

6.3原型和批评

原型是代表所有数据的数据实例。批评是一个数据实例，它不能很好地由一组原型表示。批评的目的是与原型一起提供见解，特别是对于原型不能很好地表示的数据点。原型和批评可以独立于机器学习模型来描述数据，但它们也可以用于创建可解释模型或使黑盒模型可解释。

在本章中，我使用表达式“数据点”来指代单个实例，以强调对实例也是坐标系中的一个点的解释，其中每个特征都是一个维度。下图显示了一个模拟的数据分布，其中一些实例被选为原型，一些实例被选为批评。小点是数据，大点是原型，大点是批评。选择原型（手动）以覆盖数据分布的中心，批评是没有原型的集群中的点。原型和批评总是来自数据的实际实例。

图6.9：具有两个特性x1和x2的数据分布的原型和批评。

我手动选择了原型，它的缩放效果不好，可能导致效果不佳。有许多方法可以在数据中找到原型。其中之一是k-medoid，一种与k-means算法相关的聚类算法。任何将实际数据点作为聚类中心返回的聚类算法都有资格选择原型。但这些方法中的大多数只能找到原型，但没有批评。本章介绍了Kim等人（2016）提出的MMD批评家，一种将原型和批评结合在一个框架中的方法。

MMD批评家比较了数据的分布和所选原型的分布。这是理解MMD批评家方法的核心概念。MMD批评家选择的原型可以最小化两个分布之间的差异。高密度区域的数据点是很好的原型，尤其是从不同的“数据集群”中选择点时。从原型无法很好解释的区域中选取数据点作为批评。

让我们深入研究这个理论。

6.3.1理论

可以简要总结高层次的MMD批评程序：

1. 选择要查找的原型和批评的数量。
2. 用贪婪的搜索寻找原型。选择原型是为了使原型的分布接近数据分布。
3. 贪婪地寻找批评。当原型的分布不同于数据的分布时，选择点作为批评。

我们需要一些成分来找到一个数据集的原型和批评与MMD批评家。作为最基本的成分，我们需要一个核函数来估计数据密度。内核是一个函数，它根据两个数据点的接近程度来衡量它们的权重。基于密度估计，我们需要一个度量来告诉我们两个分布有多不同，这样我们就可以确定我们选择的原型的分布是否接近数据分布。这可以通过测量最大平均偏差（mmd）来解决。另外，基于内核函数，我们需要见证函数来告诉我们在一个特定的数据点上两个分布有多不同。利用见证函数，我们可以选择批评，即原型和数据分布偏离的数据点，见证函数具有较大的绝对值。最后一个要素是对好的原型和批评的搜索策略，它通过一个简单的贪婪搜索来解决。

让我们从最大平均差（mmd）开始，它测量两个分布之间的差异。选择原型可以创建原型的密度分布。我们要评估原型分布是否与数据分布不同。我们用核密度函数来估计两者。最大平均差度量两个分布之间的差异，即根据两个分布，期望值之间差异的函数空间上的上确界。全部清除？就我个人而言，当我看到数据是如何计算的时，我更理解这些概念。下面的公式显示了如何计算平方mmd测量值（mmd2）：

\[mmd^2=\frac 1 m^2 \ sum i，j=1 k（z i，z j）-\frac 2 m n \ sum i，j=1 m，n k（z i，x j）+\frac 1

\_n^2 \总和i，j=1 ^n k（x\_i，x\_j）]

k是一个核心函数，用于测量两点的相似性，但稍后将详细介绍。m是原型z的数量，n是原始数据集中的数据点x的数量。原型Z是数据点X的选择。每个点都是多维的，也就是说它可以有多个特性。MMD批评家的目标是最小化MMD2。mmd2越接近于零，原型的分布越符合数据。将mmd2降为零的关键是中间的术语，它计算原型和所有其他数据点之间的平均接近度（乘以2）。如果这个术语加上第一个术语（原型彼此之间的平均接近度）加上最后一个术语（数据彼此之间的平均接近度），那么原型就完美地解释了数据。尝试一下如果使用所有n个数据点作为原型，公式会发生什么情况。

下图说明了mmd2测量值。第一个图显示了具有两个特征的数据点，其中数据密度的估计以阴影背景显示。每个其他绘图显示不同的原型选择，以及绘图标题中的mmd2度量。原型是大点，它们的分布显示为轮廓线。选择最能覆盖这些场景中数据的原型（左下角）具有最小的差异值。

图6.10：具有两个特征和不同原型选择的数据集的平方最大平均差异度量（mmd2）。

内核的选择是径向基函数kernel:\[k（x，x^\prime）=exp\left（\gamma x-x^\prime ^2\right）\]

其中x-x'是两点之间的欧几里得距离，而\（\gamma\）是一个缩放参数。2.当两点之间的距离无穷远时，内核值随两点之间的距离而减小，范围在0和1之间：当两点之间无限远时为零；当两点相等时为一。

我们将mmd2度量、内核和贪婪搜索结合在一个查找原型的算法中：

* 从一个空的原型列表开始。
* 当原型数量低于所选数量m时：

对于数据集中的每个点，检查将该点添加到原型列表时MMD2减少了多少。将最小化mmd2的数据点添加到列表中。

* 返回原型列表。

找到批评的剩余要素是见证函数，它告诉我们在一个特定点上两个密度估计值有多少不同。可通过以下方式估算：

\[见证人（x）=\frac 1 n \ sum i=1 ^nk（x，x i）-\frac 1 m \ sum j=1 ^mk（x，z j）\]

对于两个数据集（具有相同的特性），见证函数为您提供了一种评估方法，在这种方法中，经验分布X点更适合。为了找到批评，我们从消极和积极两个方面寻找证人功能的极端价值。见证函数中的第一项是X点和原型之间的平均邻近度，第二项分别是X点和数据之间的平均邻近度。如果X点的见证函数接近于零，则数据和原型的密度函数紧密相连，这意味着原型的分布类似于X点的数据分布。X点的正见证函数意味着原型的分布Tion高估了数据分布（例如，如果我们选择了一个原型，但附近只有很少的数据点）；X点的负见证函数意味着原型分布低估了数据分布（例如，如果X附近有很多数据点，但我们有e未选择附近的任何原型）。

为了给你更多的直觉，让我们先用最低的mmd2重用图中的原型，并显示几个手动选择的点的见证函数。下图中的标签显示了标记为正方形的各个点的见证函数值。只有中间的一点具有很高的绝对价值，因此是一个很好的批评候选人。

图6.11：不同点见证功能的评估。

见证函数允许我们显式地搜索原型不能很好地表示的数据实例。批评是证人功能中绝对值较高的点。和原型一样，批评也是通过贪婪的搜索发现的。但是，我们并没有减少整个MMD2，而是在寻找能够最大化成本函数的点，其中包括见证函数和正则化项。优化函数中的附加项加强了点的多样性，这是为了使点来自不同的簇。

第二步与如何找到原型无关。我也可以选择一些原型并使用这里描述的过程来学习批评。或者原型可以来自任何聚类过程，比如k-medoid。

这就是《货币市场部评论理论》的重要组成部分。还有一个问题：MMD批评家如何被用于可解释的机器学习？

MMD批评家可以通过三种方式添加可解释性：帮助更好地理解数据分布；通过构建可解释模型；通过使黑盒模型可解释。

如果您将MMD批评家应用于您的数据以查找原型和批评，它将提高您对数据的理解，特别是如果您具有复杂的边缘案例数据分布。但有了MMD批评家，你可以实现更多！

例如，您可以创建一个可解释的预测模型：所谓的“最近的原型模型”。预测函数定义为：

\[\hat f（x）=argmax i \ in s k（x，x i）\]

这意味着我们从最接近新数据点的一组原型中选择原型I，从这个意义上说，它产生了内核函数的最高值。原型本身作为预测的解释返回。此过程有三个调整参数：内核类型、内核缩放参数和原型数量。所有参数都可以在交叉验证循环中进行优化。这种方法不使用批评。

作为第三种选择，我们可以使用MMD批评家，通过检查原型和批评以及它们的模型预测，使任何机器学习模型都可以全球解释。程序如下：

1. 与MMD评论家一起寻找原型和评论。
2. 像往常一样训练机器学习模型。
3. 用机器学习模型预测原型和批评的结果。
4. 分析预测：在哪种情况下算法是错误的？现在，您有一些很好地表示数据的示例，帮助您发现机器学习模型的弱点。

这有什么帮助？还记得谷歌的图像分类器识别出黑人是大猩猩吗？也许他们应该在部署图像识别模型之前使用这里描述的过程。仅仅检查模型的性能是不够的，因为如果它是99%正确的，这个问题可能仍然在1%之内。标签也可能是错误的！检查所有的训练数据，并做一个健全的检查，如果预测有问题，可能已经揭示了问题，但将是不可行的。但是，选择——比如说几千个——原型和批评是可行的，并且可能揭示了数据的一个问题：它可能表明缺乏皮肤黝黑的人的图像，这表明数据集的多样性存在问题。或者它可以展示一个或多个以深色皮肤为原型的人的图像，或者（可能）是对臭名昭著的“大猩猩”分类的批评。我不保证MMD批评家一定会截获这些错误，但这是一个良好的理智检查。

6.3.2示例

我从MMD评论家的论文中举了一些例子。这两个应用程序都基于图像数据集。每幅图像都用2048维的图像嵌入来表示。图像嵌入是一个带有数字的向量，它捕获图像的抽象属性。嵌入向量通常是从神经网络中提取出来的，神经网络经过训练可以解决图像识别任务，在这种情况下，图像网络的挑战。使用这些嵌入向量计算图像之间的核距离。

第一个数据集包含来自ImageNet数据集的不同狗品种。MMD批评家应用于两个犬类的数据。在狗的左边，原型通常显示狗的脸，而批评是没有狗的脸或不同颜色（如黑白）的图像。在右侧，原型包含了狗的户外图像。批评包括穿着服装的狗和其他不寻常的案例。

图6.12：来自ImageNet数据集的两种狗品种的原型和批评。

MMD评论家的另一个例子使用手写数字数据集。

查看实际的原型和批评，您可能会注意到每个数字的图像数量是不同的。这是因为在整个数据集中搜索了固定数量的原型和批评，而不是每个类使用固定数量。正如预期的那样，原型显示了不同的数字写入方式。批评包括线条粗细异常的例子，也包括无法识别的数字。

图6.13：手写数字数据集的原型和批评。

6.3.3优势

在一项用户研究中，MMD批评家的作者给参与者提供了图片，他们必须在视觉上与两组图片中的一组相匹配，每一组图片代表两个类别中的一个（例如两个狗品种）。当一组学生展示原型和批评时，他们表现得最好，而不是一个班级的随机图片。

你可以自由选择原型和批评的数量。

MMD批评家处理数据的密度估计。这适用于任何类型的数据和任何类型的机器学习模型。

该算法易于实现。

MMD批评家在提高可解释性方面非常灵活。它可以用来理解复杂的数据分布。它可以用来建立一个可解释的机器学习模型。也可以为黑匣子机器学习模型的决策提供依据。

发现批评与原型的选择过程无关。但是根据MMD评论家选择原型是有意义的，因为原型和批评都是使用相同的方法来比较原型和数据密度创建的。

6.3.4缺点

你必须选择原型和批评的数量。尽管这样做很好，但也是一个缺点。我们实际需要多少原型和批评？越多越好？越少越好？一种解决方案是通过测量人类观察图像的时间来选择原型和批评的数量，这取决于特定的应用程序。只有在使用MMD批评家构建分类器时，我们才能直接对其进行优化。一种解决方案可以是显示X轴上原型数量和Y轴上mmd2测量值的屏幕图。我们将选择mmd2曲线变平的原型数量。

其他参数是内核和内核缩放参数的选择。我们和原型和批评的数量有同样的问题：我们如何选择内核及其缩放参数？同样，当我们使用mmd critical作为最近的原型分类器时，我们可以调整内核参数。然而，对于MMD批评家的无监督使用案例，还不清楚。（也许我在这里有点苛刻，因为所有无监督的方法都有这个问题。）

它将所有特性作为输入，忽略了一些特性可能与预测感兴趣的结果无关的事实。一种解决方案是只使用相关功能，例如图像嵌入而不是原始像素。只要我们有一种方法将原始实例投影到只包含相关信息的表示上，这就可以工作。

有一些代码是可用的，但是它还没有被很好地打包和文档化的软件实现。

6.3.5规范和备选方案

可以在以下位置找到MMD批评家的实现：。

找到原型最简单的替代方法是Kaufman等人（1987年）。

1. Kim、Be、Rajiv Khanna和Oluwasanmi O.Koyejo。“例子不够，要学会批判！对可解释性的批评〉，《神经信息处理系统的进展》（2016年）。
2. 考夫曼、伦纳德和彼得·卢梭。“通过类药物聚集”。北荷兰（1987）。

6.4影响因素

机器学习模型最终是训练数据的产物，删除其中一个训练实例可能会影响生成的模型。当从训练数据中删除训练实例会显著改变模型的参数或预测时，我们称之为“有影响的”。通过识别有影响的培训实例，我们可以“调试”机器学习模型，更好地解释它们的行为和预测。

本章向您展示两种识别有影响的实例的方法，即删除诊断和影响函数。这两种方法都基于稳健的统计数据，该统计数据提供的统计方法受异常值或违反模型假设的影响较小。稳健统计还提供了测量数据的稳健估计（如平均估计或预测模型的权重）的方法。

想象一下，你想估计一下你所在城市的人们的平均收入，然后随机问十个街上的人他们挣多少钱。除了你的样本可能真的很糟糕之外，你的平均收入估计有多少会受到一个人的影响？为了回答这个问题，您可以通过省略单个答案来重新计算平均值，或者通过“影响函数”来数学推导平均值的影响方式。采用删除法，我们将平均值重新计算十次，每次忽略一个损益表，并测量平均估计值的变化量。一个巨大的变化意味着一个例子是非常有影响力的。第二种方法是用一个无穷小的权重来增加一个人的权重，这相当于统计或模型的一阶导数的计算。这种方法也称为“无穷小方法”或“影响函数”。答案是，顺便说一句，你的平均估计可能会受到一个答案的强烈影响，因为平均值与单个值成线性比例。一个更有力的选择是中位数（一半人赚得更多，另一半人赚得更少），因为即使你的样本中收入最高的人赚得更多十倍，结果的中位数也不会改变。

删除诊断和影响功能也可应用于机器学习模型的参数或预测，以更好地了解其行为或解释个别预测。在我们研究这两种寻找有影响的实例的方法之前，我们将研究离群值实例和有影响的实例之间的区别。

离群值

离群值是一个远离数据集中其他实例的实例。“远”是指距离，例如欧几里得距离，到所有其他实例的距离非常大。在一组新生儿中，体重6公斤的新生儿被认为是异常值。在一个以支票账户为主的银行账户数据集中，一个专门的贷款账户（大额负余额，很少有交易）将被视为异常值。下图显示了1维度分布的异常值。

图6.14：特征X遵循高斯分布，X=8处有一个异常值。

离群值可以是有趣的数据点（例如当异常值影响模型时，它也是一个有影响的实例。有影响的实例

一个有影响的实例是一个数据实例，它的删除对训练模型有很大的影响。

当模型被从训练数据中删除的特定实例重新训练时，模型参数或预测的更改越多，该实例的影响就越大。一个实例对训练模型是否有影响也取决于它对目标Y的值。下图显示了一个对线性回归模型有影响的实例。

图6.15：具有一个特征的线性模型。培训一次完整的数据，一次没有影响的实例。删除影响的实例会显著改变拟合的坡度（重量/系数）。

为什么有影响的实例有助于理解模型？

对解释性有影响的实例背后的关键思想是跟踪模型参数和预测，使其回到最初的位置：培训数据。学习者，即生成机器学习模型的算法，是一个函数，它获取由特征X和目标Y组成的训练数据，并生成机器学习模型。例如，决策树的学习者是一种选择分割特征和分割值的算法。神经网络的学习者使用反向传播来找到最佳权重。

图6.16：学习者从培训数据（功能和目标）中学习模型。该模型对新数据进行预测。

我们会询问如果在培训过程中从培训数据中删除实例，模型参数或预测将如何更改。这与其他可解释性方法形成了对比，这些方法分析了当我们操纵要预测的实例（如或）的特征时，预测是如何变化的。对于有影响的实例，我们不将模型视为固定的，而是将其视为训练数据的函数。有影响的实例帮助我们回答有关全球模型行为和个人预测的问题。模型参数或预测的最有影响的实例是什么？对于一个特定的预测，哪些是最有影响的实例？有影响的实例告诉我们模型可能会出现问题的实例，应该检查哪些培训实例是否存在错误，并给出模型的健壮性印象。如果单个实例对模型预测和参数有很大的影响，我们可能不信任模型。至少这会使我们进一步调查。

我们如何才能找到有影响力的实例？我们有两种测量影响的方法：第一种方法是从训练数据中删除实例，在减少的训练数据集上重新训练模型，并观察模型参数或预测的差异（单个或整个数据集）。第二个选项是通过根据模型参数的梯度近似参数更改来增加数据实例的权重。删除方法更容易理解和激发上权方法，因此我们从前者开始。

6.4.1删除诊断

统计学家已经在有影响的实例领域做了大量的研究，特别是对于（广义）线性回归模型。当你搜索“有影响的观察结果”时，第一个搜索结果是关于dfbeta和cook距离等度量的。dfbeta测量删除实例对模型参数的影响。Cook的距离（Cook，197）测量删除实例对模型预测的影响。对于这两个度量，我们必须重复地重新培训模型，每次都忽略单个实例。将模型的所有实例的参数或预测与模型的参数或预测进行比较，其中一个实例已从训练数据中删除。

DFBETA定义为：

\[dfbeta i=\beta-\beta ^（-i）\]

其中\（\beta\）是模型在所有数据实例上进行训练时的权重向量，而\（\beta^（-i）\）是模型在没有实例i的情况下进行训练时的权重向量。我会说非常直观。DFBETA仅适用于具有权重参数的模型，如逻辑回归或神经网络，但不适用于决策树、树集合、某些支持向量机等模型。

Cook距离是为线性回归模型而发明的，并且存在广义线性回归模型的近似值。Cook对训练实例的距离定义为当第i个实例从模型训练中移除时，预测结果的平方差之和。

\[d i=\frac \ sum j=1 ^n（\hat y（-i））^2 p\cdot mse]

其中，分子是有第i个实例和没有第i个实例的模型预测之间的平方差，在数据集上求和。分母是特征数p乘以均方误差。无论删除哪个实例，所有实例的分母都相同。库克的距离告诉我们，当我们从训练中移除第i个实例时，线性模型的预测输出会发生多大的变化。

我们能把库克的距离和dfbeta用于任何机器学习模型吗？dfbeta需要模型参数，因此此度量仅适用于参数化模型。库克的距离不需要任何模型参数。有趣的是，Cook的距离通常不在线性模型和广义线性模型的范围之外，但考虑到特定实例删除前后模型预测之间的差异的想法是非常普遍的。库克距离定义的一个问题是MSE，这对所有类型的预测模型（如分类）都没有意义。

对模型预测影响的最简单影响测量可写如下：

\[\text影响^（-i）=\frac

这个表达式基本上是库克距离的分子，用绝对差代替平方差相加。这是我做的一个选择，因为它对后面的例子有意义。删除诊断度量的一般形式包括选择一个度量（如预测结果），并计算在所有实例上训练的模型和删除实例时度量的差异。

我们可以很容易地将影响分解，为实例j的预测解释第i次培训实例的影响是什么：

\[\文本影响（-i）=\左\右（-i）\右]

这也适用于模型参数的差异或损失的差异。在下面的示例中，我们将使用这些简单的影响度量。

## 删除诊断示例

在下面的例子中，我们训练一个支持向量机来预测给定的风险因素，并测量哪些训练实例对整体和特定的预测最有影响。由于癌症预测是一个分类问题，因此我们将影响作为癌症预测概率的差异来衡量。当实例从模型训练中移除时，如果预测的概率在数据集中平均显著增加或减少，则实例具有影响。对所有858个培训实例的影响测量需要对所有数据进行一次模型培训，并在每次删除其中一个实例的情况下对模型进行858次重新培训（=培训数据的大小）。

最有影响的实例的影响度量约为0.01。0.01的影响意味着如果我们去掉第540个实例，预测的概率平均变化1个百分点。考虑到癌症的平均预测概率为6.4%，这是相当可观的。所有可能删除的影响度量的平均值为0.2个百分点。现在我们知道哪些数据实例对模型最有影响。这对于调试数据已经很有用了。是否存在问题实例？是否存在测量误差？有影响的实例是第一个应该检查错误的实例，因为其中的每个错误都强烈影响模型预测。

除了模型调试之外，我们可以学习一些东西来更好地理解模型吗？仅仅打印出前10个最有影响力的实例并不是很有用，因为它只是一个具有许多特性的实例表。所有将实例作为输出返回的方法只有在我们有好的方法来表示它们的情况下才有意义。但是，当我们问：有影响力的实例和没有影响力的实例有什么区别时，我们可以更好地理解哪些实例是有影响力的？我们可以将这个问题转化为回归问题，并将实例的影响建模为其特征值的函数。我们可以从本章中自由选择任何型号。对于这个例子，我选择了一个决策树（下图），它显示来自35岁及以上妇女的数据对支持向量机最有影响。在数据集中的所有女性中，858人中有153人年龄大于35岁。在这一章中，我们已经看到，40岁以后，癌症的预测概率急剧增加，而且年龄也是最重要的特征之一。影响分析告诉我们，当预测癌症的年龄更高时，模型变得越来越不稳定。这本身就是有价值的信息。这意味着这些实例中的错误会对模型产生很大的影响。

图6.17：模拟实例影响与其特征之间关系的决策树。树的最大深度设置为2。

这第一个影响分析揭示了最具影响力的总体情况。现在我们选择其中一个实例，即第7个实例，我们想通过找到最有影响的训练数据实例来解释预测。这就像一个反事实的问题：如果我们从培训过程中忽略了实例I，那么实例7的预测会发生什么变化？我们对所有实例重复此删除。然后，我们选择在训练中忽略实例7时，导致预测最大变化的训练实例，并用它们来解释该实例的模型预测。我选择解释这个预测，例如7，因为它是癌症预测概率最高的实例（7.35%），我认为这是一个有趣的案例，需要更深入地分析。我们可以返回前10个最有影响力的实例来预测作为表格打印的第7个实例。不是很有用，因为我们看不到太多。再次，通过分析影响实例和非影响实例的特点，找出影响实例和非影响实例之间的区别是更有意义的。我们使用一个经过训练的决策树来预测给定特征的影响，但实际上我们滥用它只是为了找到一个结构，而不是实际预测某些东西。下面的决策树显示了哪种训练实例对预测第7个实例最有影响。

图6.18：解释哪些实例对预测第7个实例最有影响的决策树。吸烟18.5年或更长时间的妇女的数据对第7例的预测有很大影响，绝对预测的平均变化为癌症概率的11.7个百分点。

吸烟或已吸烟18.5年或更长时间的妇女的数据实例对第7例的预测有很大影响。第7例背后的女人吸烟了34年。数据显示，12名女性（1.40%）吸烟18.5年或更长时间。在收集其中一名妇女的吸烟年限时所犯的任何错误都将对第7例的预测结果产生巨大影响。

当我们删除实例663时，预测会发生最极端的变化。该患者据称吸烟22年，与决策树的结果一致。如果删除实例663，第7个实例的预测概率从7.35%变为66.60%。

如果我们仔细看看最有影响的实例的特性，我们可以看到另一个可能的问题。数据显示，这名妇女28岁，已经吸烟22年了。要么是非常极端的情况，她6岁就开始抽烟，要么就是数据错误。我倾向于相信后者。在这种情况下，我们必须质疑数据的准确性。

这些示例显示了识别对调试模型有影响的实例是多么有用。该方法的一个问题是，模型需要针对每个训练实例进行重新训练。整个再培训过程可能非常缓慢，因为如果您有数千个培训实例，则必须对模型进行数千次再培训。假设模型需要一天的训练时间，并且您有1000个训练实例，那么计算有影响的实例（没有并行化）将需要近3年的时间。没人有时间做这个。在本章的其余部分中，我将向您展示一种不需要重新培训模型的方法。

6.4.2影响功能

你：我想知道一个训练实例对特定预测的影响。

研究：您可以删除培训实例，重新培训模型，并测量预测的差异。

你：太好了！但是你有没有一个方法可以让我不用再培训？这需要很多时间。研究：你有一个损失函数的模型，它的参数是两倍可微的吗？

你：我训练了一个神经网络来处理物流损失。所以是的。

研究：然后用影响函数来近似实例对模型参数和预测的影响。影响函数是衡量模型参数或预测对训练实例的依赖程度的指标。方法不是删除实例，而是通过一个非常小的步骤对丢失的实例进行加权。该方法包括利用梯度和黑森矩阵近似当前模型参数的损失。损失增加权重类似于删除实例。你：太好了，这就是我要找的！

Koh和Liang（2017年建议使用影响函数（一种稳健统计方法）来测量实例如何影响模型参数或预测。与删除诊断一样，影响函数跟踪模型参数和预测返回到负责的培训实例。然而，该方法并没有删除训练实例，而是近似于当实例在经验风险（训练数据损失之和）中被加权时，模型的变化程度。

影响函数的方法要求访问与模型参数相关的损失梯度，这只适用于机器学习模型的一个子集。逻辑回归、神经网络和支持向量机是合格的，像随机森林这样的基于树的方法是不合格的。影响函数有助于理解模型行为、调试模型和检测数据集中的错误。

以下部分解释了影响函数背后的直觉和数学。

## 影响函数背后的数学

影响函数背后的关键思想是通过无限小的步骤（epsilon）增加训练实例的损失，从而产生新的模型参数：

\[\hat \theta \epsilon，z=\arg \min \theta \i n \theta（1-\epsilon \frac n \sum i=1 l（z i，theta）

+\ epsilon l（z，theta）\]

其中\（\theta\）是模型参数向量，而\（\hat \ theta \ epsilon，z \）是用非常小的数字\（\epsilon\）对z进行加权后的参数向量。l是模型训练的损失函数，z是训练数据，z是我们想要增加权重来模拟其移除的训练实例。这个公式背后的直觉是：如果我们将训练数据中的某个特定实例（z\_i）稍微增加一点（\（\ epsilon\）并相应地降低其他数据实例的权重，损失会有多大变化？参数向量是什么样子来优化这个新的组合损失？参数的影响函数，即上权训练实例Z对参数的影响，可计算如下。

\[I \ Theta \\ \ Epsilon，Z \123\ \ \\123; 123; \θl（z，that \θ）\]

最后一个表达式\（\nabla \theta l（z，that \theta）是相对于上权训练实例参数的损失梯度。梯度是训练实例丢失的变化率。它告诉我们，当我们稍微改变模型参数时，损失会有多大的变化。梯度向量中的正项意味着相应模型参数的微小增加会增加损失，负项意味着参数的增加会减少损失。第一部分\（h^-1 \ that \ theta）是逆Hessian矩阵（损失相对于模型参数的二阶导数）。黑森矩阵是梯度变化率，或表示为损耗，它是损耗变化率的变化率。可以使用下列公式进行估算：

更非正式的是：黑森矩阵记录了损失在某一点上的弯曲程度。Hessian是一个矩阵，而不仅仅是一个向量，因为它描述了损失的曲率，曲率取决于我们观察的方向。如果你有许多参数，那么海森矩阵的实际计算是耗时的。Koh和Liang提出了一些有效计算的技巧，这超出了本章的范围。如上述公式所述，更新模型参数，相当于在估计模型参数周围形成二次展开后采取一个牛顿步骤。

这个影响函数公式背后的直觉是什么？公式来自于围绕参数形成二次展开式。这意味着我们实际上不知道，或者它太复杂了，无法计算实例z的损失在被删除/加权时究竟会有多大的变化。我们利用当前模型参数设置下的陡度（=梯度）和曲率（=黑森矩阵）的信息对函数进行局部近似。通过这种损失近似，我们可以计算出如果我们对实例z进行加权，新参数将大致是什么样子：

\[\Hat \Theta-Z \约\Hat \Theta-\Frac 1 n i \ Text up，参数（Z）]

近似参数向量基本上是原始参数减去z的损失梯度（因为我们想减少损失），用曲率（=乘以逆Hessian矩阵）缩放，用1除以n缩放，因为这是单个训练实例的权重。

下图显示了向上加权的工作原理。x轴显示了\（\theta\）参数的值，y轴显示了加权实例z对应的损失值。这里的模型参数是一维的，用于演示，但实际上通常是高维的。我们只移动1到n的方向上，以改善损失，例如z。我们不知道当我们删除z时损失会如何变化，但是使用损失的一阶和二阶导数，我们围绕当前模型参数创建这个二次近似，并假设真正的损失是如何表现的。

图6.19：通过在当前模型参数周围形成损耗的二次扩展，并将1/N移动到具有上加权实例z（y轴）的损耗改善最多的方向，更新模型参数（x轴）。如果我们删除Z并在减少的数据上对模型进行训练，则损失中实例Z的上浮近似于参数变化。

我们实际上不需要计算新的参数，但是我们可以使用影响函数来测量z对参数的影响。

当我们增加训练实例z的权重时，预测是如何变化的？我们既可以计算新参数，然后使用新参数化模型进行预测，也可以直接计算实例z对预测的影响，因为我们可以使用链规则计算影响：

\[\begin align\*i up，loss（z，z test）&=\ left.\frac d l（z test，\hat \theta \epsilon，z）d \epsilon

\右\ epsilon=0 \ \左.\nabla \ theta l（z测试，\t hat \ theta）^t\frac d\that \ theta \ epsilon，z

；）\结束对齐\*\]

这个方程的第一行意味着我们测量训练实例对某个预测（z\_test \）的影响，当我们增加实例z的权重并获得新参数时，作为测试实例损失的变化。对于方程的第二行，我们应用了导数链规则，得到了测试实例损失的导数，它与参数乘以z对参数的影响。在第三行，我们将表达式替换为参数的影响函数。第三行中的第一个术语是测试实例相对于模型参数的梯度。

有一个公式是伟大的，科学和准确的方式显示事物。但我认为对公式的含义有一些直觉是非常重要的。（i\_ text up，loss）的公式指出，训练实例z对实例预测的影响函数为“当我们对inst进行权重计算时，实例对模型参数变化的反应程度”乘以“参数变化的程度”。安斯Z“。阅读公式的另一种方法是：影响与训练和测试损失的梯度大小成正比。训练损失梯度越大，对参数的影响越大，对测试预测的影响越大。测试预测的梯度越高，对测试实例的影响就越大。整个结构也可以看作是训练和测试实例之间相似性的度量（由模型学习）。

这就是理论和直觉。下一节将解释如何应用影响函数。

## 影响函数的应用

影响函数有许多应用，其中一些已经在本章中介绍。

## 了解模型行为

不同的机器学习模型有不同的预测方法。即使两个模型具有相同的性能，它们根据特性进行预测的方式也可能非常不同，因此在不同的场景中失败。通过识别有影响的实例来了解模型的特定弱点，有助于在您的头脑中形成机器学习模型行为的“心理模型”。下图显示了一个例子，其中训练了支持向量机（SVM）和神经网络来区分狗和鱼的图像。对于这两个模型来说，最具影响力的鱼类典型形象的例子是非常不同的。对于SVM来说，如果实例在颜色上相似，那么它们是有影响的。对于神经网络来说，如果实例在概念上相似，那么它们是有影响的。对于神经网络来说，甚至一张狗的图像也是最有影响的图像之一，这表明它学习了概念，而不是颜色空间中的欧几里得距离。

图6.20：狗还是鱼？对于与测试图像颜色相似的支持向量机预测（中行）图像，影响最大。对于神经网络预测（底排），不同环境下的鱼类影响最大，但也有狗的图像（右上角）。Koh和Liang的作品（2017年）。

## 处理域不匹配/调试模型错误

处理域不匹配与更好地理解模型行为密切相关。域不匹配意味着训练数据和测试数据的分布是不同的，这可能导致模型在测试数据上的性能不佳。影响函数可以识别导致错误的训练实例。假设您为接受过手术的患者培训了一个预测模型。所有这些病人都来自同一家医院。现在，您在另一家医院使用该模型，发现它对许多患者都不起作用。当然，你假设两个医院的病人不同，如果你看他们的数据，你会发现他们在很多方面都不同。但是什么是“破坏”模型的特性或实例呢？在这里，有影响力的例子也是回答这个问题的好方法。你可以选取一个新的病人，模型对他们做出了错误的预测，找到并分析最有影响的例子。例如，这可能表明第二家医院平均有老年患者，从培训数据来看，最有影响的例子是第一家医院的少数老年患者，而该模型仅仅缺少学习预测该亚组的数据。结论是，为了在第二家医院更好地工作，该模型需要对更多年龄较大的患者进行培训。

## 修复培训数据

如果您对可以检查正确性的培训实例数量有限制，那么如何进行有效的选择呢？最好的方法是选择最有影响力的实例，因为根据定义，它们对模型的影响最大。即使您有一个具有明显不正确值的实例，如果该实例不具有影响，并且您只需要预测模型的数据，那么最好检查具有影响的实例。例如，您培训一个模型来预测患者是应该住院还是应该提前出院。您真的想确保模型是健壮的，并做出正确的预测，因为错误地释放患者可能会产生不良后果。患者记录可能非常混乱，因此您对数据的质量没有完全的信心。但是检查病人信息和纠正它是非常耗时的，因为一旦你报告了需要检查的病人，医院实际上需要派人更仔细地查看所选病人的记录，这些记录可能是手写的，并放在一些拱门中。伊夫。检查患者的数据可能需要一个小时或更长时间。考虑到这些成本，只检查一些重要的数据实例是有意义的。最好的方法是选择对预测模型影响较大的患者。Koh和Liang（2017）表明，这种类型的选择比随机选择或损失最大或分类错误的选择效果要好得多。

6.4.3识别有影响的实例的优势

删除诊断和影响函数的方法与A look at important instances中提出的基于特征扰动的方法有很大的不同，它强调了训练数据在学习过程中的作用。这使得影响函数和删除诊断成为机器学习模型的最佳调试工具之一。在本书中介绍的技术中，它们是唯一直接帮助识别应该检查错误的实例的技术。

删除诊断是模型不可知论，这意味着该方法可以应用于任何模型。基于导数的影响函数也可以应用于广泛的模型。

我们可以使用这些方法来比较不同的机器学习模型，更好地了解它们的不同行为，而不仅仅是比较预测性能。

在本章中我们没有讨论过这个主题，但是通过衍生工具的影响函数也可以用来创建对抗训练数据。这些实例的操作方式使得当模型在这些被操作的实例上进行培训时，模型无法正确地预测某些测试实例。与中的方法不同的是，攻击发生在训练期间，也称为中毒攻击。如果您感兴趣，请阅读Koh和Liang（2017）的论文。

对于缺失诊断和影响函数，我们考虑了预测中的差异，对于影响函数，考虑了损失的增加。但是，实际上，这种方法可以推广到任何形式的问题：“当我们删除或增加实例z的权重时，会发生什么？“，在这里你可以用你的欲望模型的任何功能来填充”…“。您可以分析培训实例对模型的整体损失有多大的影响。您可以分析培训实例对功能重要性的影响程度。可以分析培训实例对培训A时为第一个拆分选择的功能有多大影响。

6.4.4识别有影响的实例的缺点

删除诊断计算起来非常昂贵，因为它们需要重新培训。但历史表明计算机资源在不断增加。20年前的一个计算在资源方面是不可想象的，可以很容易地用你的智能手机进行。您可以在笔记本电脑上以秒/分钟的速度训练具有数千个训练实例和数百个参数的模型。因此，假设删除诊断在10年内即使对大型神经网络也能毫无问题地工作，这并不是一个大的飞跃。

影响函数是删除诊断的一个很好的替代方法，但只适用于参数可微的模型，如神经网络。它们不适用于基于树的方法，如随机森林、增强树或决策树。即使你有带参数和损失函数的模型，损失也可能是不可微的。但对于最后一个问题，有一个诀窍：当基础模型使用铰链损失而不是某些可微分损失时，使用可微分损失代替计算影响。对于影响函数，损失被问题损失的平滑版本所取代，但是模型仍然可以用非光滑损失进行训练。

影响函数只是近似的，因为该方法围绕参数形成二次展开。近似值可能是错误的，当移除实例时，实例的影响实际上更高或更低。Koh和Liang（2017）举例说明，通过影响函数计算的影响接近于删除实例后实际重新约束模型时获得的影响度量。但不能保证近似值总是如此接近。

我们称之为有影响力或无影响力的实例的影响度量没有明确的界限。根据影响对实例进行分类是很有用的，但最好的方法不仅是对实例进行分类，而且实际上是区分有影响的实例和无影响的实例。例如，如果您为一个测试实例确定10个最有影响力的培训实例，其中一些可能不具有影响力，因为例如，只有前3个才真正具有影响力。

影响措施只考虑单个实例的删除，不考虑同时删除多个实例。较大的数据实例组可能具有一些交互作用，这些交互作用对模型训练和预测有很大影响。但问题在于组合数学：从数据中删除单个实例有n种可能性。有n次（n-1）可能从训练数据中删除两个实例。有n次（n-1）次（n-2）的可能性删除三个…我想你可以看到这是去哪里，有太多的组合。

6.4.5软件和备选方案

删除诊断很容易实现。看看我在本章中写的例子。

对于线性模型和广义线性模型，许多影响度量如Cook距离在统计包的R中实现。

Koh和Liang从他们的论文中发布了影响函数的python代码。

太好了！不幸的是，它是本文的“唯一”代码，而不是维护和文档化的Python模块。代码集中在TensorFlow库上，因此您不能使用它直接用于使用其他框架（如sci kit learn）的黑盒模型。

Keita Kurita写了一篇帮助我更好地理解Koh和Liang的论文。博客文章对黑匣子模型的影响函数背后的数学进行了更深入的探讨，并讨论了一些有效实现该方法的数学“技巧”。

1. 库克，R.丹尼斯。“线性回归中影响观察的检测”，《技术计量学》19.1（1977）：15-18。
2. Koh、庞伟和Percy Liang。“通过影响函数了解黑盒预测”，arxiv预印arxiv:1703.04730（2017）。

第七章神经网络解释

本章目前仅在Web版本中可用。接下来是电子书和印刷品。

接下来的章节重点介绍神经网络的解释方法。这些方法将神经网络学习到的特征和概念可视化，解释个别的预测并简化神经网络。

深度学习非常成功，尤其是在涉及图像和文本的任务中，如图像分类和语言翻译。深度神经网络的成功故事始于2012年，当时图像网图像分类挑战是通过深度学习方法赢得的。从那时起，我们见证了寒武纪深层神经网络体系结构的爆炸，随着越来越多的权重参数向深层网络发展。

为了用神经网络进行预测，输入的数据通过多层乘法和非线性变换传递。根据神经网络的结构，单个预测可能涉及数百万个数学运算。我们人类不可能遵循从数据输入到预测的精确映射。为了理解神经网络的预测，我们必须考虑数百万个以复杂方式相互作用的权重。为了解释神经网络的行为和预测，我们需要具体的解释方法。本章假设您熟悉深度学习，包括卷积神经网络。

我们当然可以使用如或的方法，但有两个原因可以解释为什么要考虑专门为神经网络开发的解释方法：首先，神经网络学习隐藏层中的特征和概念，我们需要特殊的工具来发现它们。第二，梯度可以用来实现比模型不可知论方法（从外部看模型）更有效的解释方法。另外，本书中的大多数其他方法都是为了解释表格数据的模型。图像和文本数据需要不同的方法。

下一章包括以下主题：

* ：神经网络学习了哪些功能？从与功能可视化密切相关的：我们如何操作输入以获得错误的分类？
* （进行中）：神经网络学习了哪些更抽象的概念？
* （进行中）：每个输入是如何促进特定预测的？
* （进行中）：我们如何用更简单的模型解释神经网络？

1。奥尔加·鲁萨科夫斯基、贾邓、郝苏、乔纳森·克劳斯、桑吉夫·萨蒂什、肖恩·马，

黄志恒，卡尔帕蒂，科斯拉，伯恩斯坦，伯格和李

菲菲。（\*=贡献均等）ImageNet大规模视觉识别挑战。IJCV，

二千零一十五

7.1功能可视化

本章目前仅在Web版本中可用。接下来是电子书和印刷品。

卷积神经网络从原始图像像素学习抽象特征。特征可视化方法的目的是通过生成最大限度地激活训练卷积神经网络单元的图像来实现这些特征的可视化。

深层神经网络的最大优点之一是它们能自动学习隐藏层中的高级特征。这部分地减少了对特性工程的需求。假设您希望使用支持向量机构建图像分类器。原始像素矩阵不是培训SVM的最佳输入，因此您可以创建基于颜色、频域、边缘检测器等的新功能。利用卷积神经网络，将图像以原始形式（像素）送入网络，进行多次变换，首先通过多个卷积层，然后通过完全连接的层，并转化为分类或预测。在训练过程中，卷积神经网络在其层次上学习新的、越来越复杂的特征。

图7.1：通过在ImageNet数据上训练的卷积神经网络（起始v1）学习的特征。这些特征包括从低卷积层（左）中的简单特征到高卷积层（右）中的更抽象特征。来自Olah等人的图2017年（CCBY 4.0）。

* 第一个卷积层学习边缘和简单纹理等特征。
* 后来卷积层学习更复杂的纹理和图案等特征。
* 最后一个卷积层学习对象或对象部分等特征。
* 完全连接的层学习将激活从高级功能连接到要预测的单个类。

凉爽的。但我们如何才能真正得到这些幻觉图像呢？

使学习的特征显式化的方法称为特征可视化。神经网络的一个单元的特征可视化是通过找到最大限度地激活该单元的输入来实现的。“单位”是指单个神经元、整个特征图、整个（卷积）层或分类中的最终类概率（或推荐的相应的前Softmax神经元）。单个神经元是网络的原子单位，因此我们可以通过为每个神经元创建特征可视化来获取最多的信息。但有一个问题：神经网络通常包含数百万个神经元，观察每个神经元的特征可视化需要很长时间。通道（有时称为激活映射）作为单元是功能可视化的一个很好的选择。我们可以更进一步，想象一个完整的卷积层。层作为一个单元被用于谷歌的deepdream，它反复地将一个层的可视化特性添加到原始图像中，从而产生一个梦幻般的输入版本。

图7.2：可以对不同的单元进行功能可视化。a）卷积神经元，b）

卷积通道，c）卷积层，d）神经元，e）隐层，f）类概率神经元（或对应的前Softmax神经元）

7.1.1优化功能可视化

在数学术语中，特征可视化是一个优化问题。我们假设神经网络的权重是固定的，这意味着网络是经过训练的。我们正在寻找一个新的图像，它可以最大化一个单元的（平均）激活，这里是一个神经元：

\[img^\*=\arg\max img h n，x，y，z（img）\]

函数（h）是神经元的激活，img是网络的输入（图像），x和y描述神经元的空间位置，n指定层，z是通道索引。对于n层中整个通道z的平均激活，我们最大化：

\[img^\*=\arg\max img \ sum x，y h n，x，y，z（img）\]

在这个公式中，Z通道中的所有神经元都具有相同的权重。或者，你也可以最大化随机方向，这意味着神经元将被不同的参数所乘，包括负方向。通过这种方式，我们研究神经元如何在通道内相互作用。除了最大化激活，您还可以最小化激活（对应于最大化负方向）。有趣的是，当你把负方向最大化时，你会得到同一个单元非常不同的特性：

图7.3：混合层4d pre relu中起始v1神经元484的正（左）和负（右）激活。当神经元被轮子最大限度地激活时，似乎有眼睛的东西会产生负的激活。代码：

我们可以用不同的方法来解决这个优化问题。首先，我们为什么要生成新的图像？我们可以简单地搜索我们的训练图像，并选择那些最大化激活的图像。这是一种有效的方法，但是使用训练数据有一个问题，即图像上的元素可以相互关联，我们看不到神经网络真正在寻找什么。如果某个通道的高激活率图像显示狗和网球，我们不知道神经网络是看狗还是看网球，或者两者都看。

另一种方法是从随机噪声开始生成新的图像。为了获得有意义的可视化效果，通常对图像有限制，例如只允许进行小的更改。为了减少特征可视化中的噪声，可以在优化步骤之前对图像应用抖动、旋转或缩放。其他正则化选项包括频率惩罚（例如减少相邻像素的方差）或生成具有已知优先级的图像，例如生成对抗网络（gans）或去噪自动编码器。

图7.4：从随机图像到最大化激活的迭代优化。Olah等人2017年（CC-BY 4.0）

7.1.2与对抗性示例的连接

在特征可视化和两种技术之间有一个联系，即最大限度地激活神经网络单元。对于对抗性的例子，我们寻找对抗性（错误）类神经元的最大激活。一个区别是我们从图像开始：对于对抗性的例子，它是我们想要为其生成对抗性图像的图像。对于特征可视化，根据方法，它是随机噪声。

7.1.3文本和表格数据

本文主要研究卷积神经网络在图像识别中的特征可视化。从技术上讲，没有什么可以阻止您找到最大限度地激活表格数据完全连接的神经网络的神经元或文本数据的循环神经网络的输入。您可能不再称它为功能可视化，因为“功能”将是表格数据输入或文本。对于信用违约预测，输入可能是先前信用的数量、移动合同的数量、地址和几十个其他功能。一个神经元的习得特征将是几十个特征的某种组合。对于复发性神经网络而言，更好地可视化网络学习的内容：Karpathy等人（2015年）表明，复发性神经网络确实具有学习可解释特征的神经元。他们训练了一个角色级别的模型，该模型从前面的角色预测序列中的下一个角色。一旦出现一个开放性支撑“（发生时，其中一个神经元被高度激活，并在匹配的闭合支架时被解除激活”）。其他的神经元在一条线的末端激发。一些神经元在URL中激活。与CNN的特征可视化不同的是，这些例子是通过优化发现的，但是通过研究训练数据中的神经元激活。

7.1.4优势

* 特征可视化使人们对神经网络的工作有了独特的认识，特别是对于图像识别。鉴于神经网络的复杂性和不透明性，特征可视化是分析和描述神经网络的重要步骤。通过特征可视化，我们了解到神经网络首先学习简单的边缘和纹理检测器以及更高层次的抽象部分和对象。
* 特征可视化是一个很好的工具，可以以一种非技术性的方式交流神经网络的工作方式。
* 特征可视化可以与特征属性方法相结合，解释哪些像素对分类很重要。这两种方法的结合可以解释单个分类，以及与分类相关的学习特性的局部可视化。见
* 最后，功能可视化可以制作出很好的桌面壁纸和T恤。

7.1.5缺点

* 许多特征可视化图像是完全无法解释的，但包含一些抽象的特征，我们没有文字或心理概念。功能可视化和培训数据的显示会有所帮助。这些图像可能还不能揭示神经网络的反应，只能显示“也许图像中一定有黄色的东西”。
* 有太多的单元需要查看，即使“仅”可视化通道激活。对于初始版本1，已经有超过5000个来自9个卷积层的信道。如果您还想显示负激活加上一些训练数据中最大或最小激活频道的图像（比如4个正、4个负图像），那么您必须已经显示超过50000个图像。你甚至还没有开始研究随机的方向或者为激活频道创建一组不同的多个图像。
* 可解释性错觉？这些图像可以传达一种错觉，即我们理解神经网络在做什么。但是我们真的了解神经网络里发生了什么吗？即使我们看到成百上千的特征可视化，我们也无法理解神经网络。神经网络仍然是不可理解的：神经元以复杂的方式相互作用，正激活和负激活是不相关的，多个神经元可能学习非常相似的特征，对于许多特征，我们没有同等的人类概念。我们不能陷入相信我们完全理解神经网络的陷阱，因为我们相信我们看到第7层的349神经元被雏菊激活。

7.1.6软件和其他材料

功能可视化有一个开源实现，叫做你可以通过使用LucidGithub页面上提供的笔记本链接在你的浏览器中方便地尝试它。不需要额外的软件。

如果你想更深入地研究功能可视化，可以看看district.pub在线期刊，尤其是Olah等人的功能可视化文章。从中我使用了许多图像，以及关于可解释性的构建块。

1. Nguyen，Anh等人“通过DeepGenerator网络合成神经网络中神经元的首选输入。”神经信息处理系统的进展。2016。
2. Nguyen，Anh等人“即插即用生成网络：潜在空间中图像的条件迭代生成”，《IEEE计算机视觉和模式识别会议论文集》。2017。
3. Karpathy、Andrej、Justin Johnson和Li Fei Fei。“可视化和理解递归网络”，arxiv预印arxiv:1506.02078（2015）。
4. Olah等人，“特征可视化”，蒸馏，2017年。
5. Olah，et al.，“可解释性的基石”，蒸馏，2018年。

第八章水晶球

可解释机器学习的未来是什么？这一章是一个推测性的心理练习和一个主观的猜测如何解释机器学习将发展。我以相当悲观的态度打开了这本书，并希望以一种更加乐观的态度结束这本书。

我的“预测”基于三个前提：

1. 数字化：任何（有趣的）信息都将数字化。想想电子现金和在线交易。想想电子书、音乐和视频。想想关于我们的环境、人类行为、工业生产过程等的所有感官数据。所有东西数字化的驱动因素包括：廉价的计算机/传感器/存储、规模效应（赢家全力以赴）、新的商业模式、模块化的价值链、成本压力等等。
2. 自动化：当一个任务可以被自动化，并且自动化的成本低于一段时间内执行该任务的成本时，该任务将被自动化。甚至在引进计算机之前，我们已经有了一定程度的自动化。例如，织机自动编织或蒸汽机自动马力。但是计算机和数字化将自动化带到了下一个层次。简单地说，您可以编程forloops、编写Excel宏、自动化电子邮件响应等等，这就显示了个人可以自动化的程度。自动售票机可以自动购买火车票（不再需要出纳），洗衣机可以自动洗衣，定期订单可以自动付款等。自动化任务可以腾出时间和金钱，因此有一个巨大的经济和个人动机来实现自动化。我们目前正在观察语言翻译、驾驶的自动化，甚至在很小程度上观察科学发现。
3. 错误的指定：我们不能用它的所有约束完美地指定一个目标。想象一个瓶子里的精灵，它总是按照字面意思表达你的愿望：“我想成为世界上最富有的人！“->你成为最富有的人，但作为副作用，你持有的货币因通货膨胀而崩溃。

“我想快乐地度过余生！“->在接下来的5分钟里，你感到非常高兴，然后精灵杀死了你。

“我祝愿世界和平！“->精灵杀死所有人。

我们错误地指定目标，要么是因为我们不知道所有的约束，要么是因为我们无法度量它们。让我们把公司看作是不完善目标规范的一个例子。公司的简单目标是为股东赚钱。但是，这个规范并没有捕捉到真正的目标和我们真正努力追求的所有约束：例如，我们不欣赏一家公司为了赚钱而杀人，毒害河流，或者仅仅打印自己的钱。我们发明了法律、法规、制裁、合规程序、工会等来修补不完善的目标规范。另一个你可以亲身体验的例子是一个游戏，你玩一台机器，目的是生产尽可能多的回形针。警告：它会上瘾。我不想把它搞得太糟，但让我们说事情很快就失控了。在机器学习中，目标规范中的缺陷来自不完善的数据抽象（偏态总体、测量误差等）、不受约束的损失函数、对约束缺乏了解、训练数据和应用数据之间分布的转移等等。

数字化正在推动自动化。不完善的目标规范与自动化冲突。我声称，这种冲突部分是通过解释方法来调解的。

我们预测的舞台已经准备好了，水晶球已经准备好了，现在我们来看看这个领域的发展方向！

8.1机器学习的未来

没有机器学习就没有可解释的机器学习。因此，在讨论可解释性之前，我们必须猜测机器学习的方向。

机器学习（或“人工智能”）与许多承诺和期望有关。但是，让我们从一个不那么乐观的观察开始：虽然科学开发了许多奇特的机器学习工具，但根据我的经验，很难将它们集成到现有的过程和产品中。不是因为这不可能，而是因为公司和机构需要时间来赶上。在当前人工智能炒作的淘金热中，公司开设了“人工智能实验室”、“机器学习单元”和“数据科学家”、“机器学习专家”、“人工智能工程师”等等，但现实是，在我的经验中，相当令人沮丧。通常情况下，公司甚至没有所需格式的数据，数据科学家们会等待数月。有时公司对人工智能和数据科学有如此高的期望，因为媒体使得数据科学家永远无法实现它们。通常没有人知道如何将数据科学家整合到现有的结构和许多其他问题中。这导致了我的第一个预言。

机器学习会缓慢而稳定地成长。

数字化正在推进，自动化的诱惑也在不断地拉动。即使机器学习采用的道路是缓慢和坚如磐石的，机器学习也在不断地从科学转向商业过程、产品和现实世界的应用。

我认为我们需要更好地向非专家解释什么类型的问题可以被表述为机器学习问题。我认识许多高薪的数据科学家，他们通过报表和SQL查询来执行Excel计算或经典的商业智能，而不是应用机器学习。但一些公司已经成功地使用了机器学习，而大型互联网公司则处于最前沿。我们需要找到更好的方法将机器学习整合到流程和产品中，培训人员并开发易于使用的机器学习工具。我相信机器学习将变得更加容易使用：我们已经看到机器学习变得越来越容易使用，例如通过云服务（“机器学习作为一种服务”），只需在周围添加一些流行词。一旦机器学习成熟了——这个刚学走路的孩子已经迈出了第一步——我的下一个预测是：

机器学习会带来很多东西。

根据“任何可以自动化的东西都是自动化的”的原则，我得出结论，只要可能，任务将被制定为预测问题，并通过机器学习来解决。机器学习是自动化的一种形式，或者至少可以是其中的一部分。目前由人类执行的许多任务都被机器学习所取代。以下是一些任务示例，其中机器学习用于自动化部分任务：

* 文件的整理/决策/完成（如保险公司、立法部门或咨询公司）
* 数据驱动的决策，如信贷应用程序
* 药物发现
* 装配线的质量控制
* 自动驾驶汽车
* 疾病诊断
* 翻译。在这本书中，我使用了一个名为（由深度神经网络支持的）的翻译服务，通过将句子从英语翻译成德语，然后再翻译成英语来改进我的句子。
* …

机器学习的突破不仅是通过更好的计算机/更多的数据/更好的软件实现的，还包括：

可解释性工具促进机器学习的采用。

基于机器学习模型的目标永远不能被完美地指定的前提，可以解释的机器学习是必要的，以弥补错误指定和实际目标之间的差距。在许多领域和部门，可解释性将成为采用机器学习的催化剂。一些轶事证据：我和许多人交谈过，他们不使用机器学习，因为他们不能向其他人解释模型。我相信，可解释性将解决这个问题，使机器学习对需要某种透明度的组织和人员具有吸引力。除了对问题的错误描述，许多行业还需要可解释性，无论是出于法律原因，还是出于风险规避，或是为了深入了解基础任务。

机器学习使建模过程自动化，使人类远离数据和底层任务：这增加了实验设计、培训分布选择、采样、数据编码、特征工程等问题的风险。解释工具使识别这些问题更加容易。

8.2可解释性的未来

让我们来看看机器学习可解释性可能的未来。

重点是模型不可知论解释工具。

当它与底层机器学习模型分离时，自动化可解释性就容易得多。模型不可知解释的优点在于其模块性。我们可以很容易地替换底层的机器学习模型。我们可以很容易地取代解释方法。基于这些原因，模型不可知论方法将有更好的扩展性。这就是为什么我相信模型不可知论方法将在长期内变得更占主导地位。但本质上可解释的方法也会有一席之地。

机器学习将是自动化的，有了它，就可以解释。

一个已经明显的趋势是模型培训的自动化。这包括自动化工程和特性选择、自动超参数优化、不同模型的比较以及模型的集成或叠加。结果是最佳预测模型。当我们使用模型不可知论解释方法时，我们可以自动将它们应用于从自动化机器学习过程中出现的任何模型。在某种程度上，我们也可以自动化第二步：自动计算特征重要性、绘制部分依赖关系、训练代理模型等等。没有人阻止您自动计算所有这些模型解释。实际的解释仍然需要人。想象一下：你上传一个数据集，指定预测目标，然后按下一个按钮，最佳的预测模型就会得到训练，程序会给出模型的所有解释。已经有了第一批产品，我认为对于许多应用程序来说，使用这些自动化机器学习服务就足够了。今天，任何人都可以在不了解HTML、CSS和JavaScript的情况下创建网站，但周围仍然有许多Web开发人员。同样，我相信每个人都可以在不知道如何编程的情况下训练机器学习模型，而且仍然需要机器学习专家。

我们不分析数据，我们分析模型。

原始数据本身总是无用的。（我故意夸大。事实上，你需要对数据有一个深刻的理解来进行有意义的分析。）我不关心数据；我关心数据中包含的知识。可解释机器学习是从数据中提取知识的一种很好的方法。您可以广泛地探测模型，模型自动识别特征是否和如何与预测相关（许多模型都有内置的特征选择），模型可以自动检测关系的表示方式，如果经过正确的训练，最终模型是一个非常很好地接近现实。

许多分析工具已经基于数据模型（因为它们基于分布假设）：

* 像学生t检验一样的简单假设检验。
* 调整混杂因素的假设检验（通常是GLMS）
* 方差分析（方差分析）
* 相关系数（标准化线性回归系数与

皮尔逊相关系数）

* …

我在这里告诉你的并不是什么新鲜事。那么，为什么要从分析基于假设的透明模型转向分析无假设的黑盒模型呢？因为做出所有这些假设都是有问题的：它们通常是错误的（除非你相信世界上大多数地方都遵循高斯分布），难以检查，非常不灵活，很难自动化。在许多领域，基于假设的模型对未接触测试数据的预测性能通常比黑盒机器学习模型差。这只适用于大数据集，因为具有良好假设的可解释模型在小数据集上的性能通常比黑盒模型好。黑盒机器学习方法需要大量的数据才能很好地工作。随着所有事物的数字化，我们将拥有更大的数据集，因此机器学习的方法变得更具吸引力。我们不做假设，我们尽可能接近现实（同时避免过度拟合训练数据）。我认为我们应该开发统计学中的所有工具来回答问题（假设测试、相关度量、交互度量、可视化工具、置信区间、p值、预测区间、概率分布），并将它们改写为黑盒模型。在某种程度上，这已经发生了：

* 让我们采用一个经典的线性模型：标准化回归系数已经是一个重要的度量。有了，我们就有了一个可以与任何模型一起工作的工具。
* 在线性模型中，系数测量单个特征对预测结果的影响。这个的广义版本是。
* 测试A或B是否更好：为此，我们也可以使用偏相关函数。我们还没有（据我所知）对任意黑盒模型进行统计测试。

数据科学家将自动进行处理。

我相信，对于许多分析和预测任务，数据科学家最终会自动完成他们的工作。为了实现这一点，任务必须定义得很好，并且周围必须有一些流程和例程。如今，这些程序和过程已经不复存在，但数据科学家和同事们正在研究它们。随着机器学习成为许多行业和机构不可或缺的一部分，许多任务将自动化。

机器人和程序会自我解释。

我们需要更多直观的机器和程序接口，以充分利用机器学习。一些例子：一辆自动驾驶的汽车，它报告为什么突然停止（“70%的可能性，一个孩子会过马路”）；一个信用违约程序，解释了为什么信用申请被拒绝（“申请人有太多的信用卡，并受雇于一个不稳定的工作）。“）；一个机器人手臂，解释了为什么它将物品从传送带移到垃圾箱中（“物品底部有一种热”）。

可解释性可以促进机器智能研究。

我可以想象，通过对程序和机器如何解释自己进行更多的研究，我们可以提高对智能的理解，并更好地创建智能机器。

最后，所有这些预测都是推测，我们必须看看未来到底会带来什么。形成自己的观点，继续学习！

第九章贡献

感谢您阅读我的关于可解释机器学习的书。这本书正在不断地发展。它将随着时间的推移而改进，并将添加更多章节。非常类似于软件的开发方式。

这本书的所有文本和代码都是开源的。在Github页面上，您可以建议进行修复，如果发现错误或丢失某些内容。

如果你想提供更多的帮助，问题页面也是解决问题的最佳地方，也是一个为这本书做出贡献的好方法。如果您对更大的贡献感兴趣，请给我发一条信息，告诉我您的具体想法：

第十章引用本书

如果你发现这本书对你的博客文章、研究文章或产品有用，如果你能引用这本书，我将不胜感激。你可以这样引用这本书：

克里斯托夫·莫纳尔。”可解释的机器学习。《黑盒模型制作指南》，2019年。https://christophm.github.io/可解释的ml book/。

或使用以下bibtex条目：

@书号：Molnar2019，

标题=可解释机器学习，作者=Christoph Molnar，

注=\url https://christophm.github.io/interpretable mlbook/，

年份=2019，

subtitle=一个使黑匣子模型可解释的指南

我一直对解释方法在工业和研究中的使用位置和方式感到好奇。如果你用这本书作参考，那就太好了，如果你给我写一行，告诉我为什么。这当然是可选的，只是为了满足我自己的好奇心和激发有趣的交流。我的邮件是。

第十一章确认

写这本书很有趣。但这也是一个很大的工作，我很高兴我得到的支持。

我最大的感谢是凯特琳，她在工作时间和精力上都做得最辛苦：她从头到尾校对了这本书，发现了许多拼写错误和前后不一致的地方，我永远也找不到。我非常感谢她的支持。

非常感谢Verena Haunschmid撰写了这一部分。她在数据科学领域工作，我建议在Twitter上关注她：我也要感谢Github上的所有人！

此外，我还要感谢所有制作插图的人：封面是我朋友设计的。中的图形是由创建的，使用的图标来自于中关于由Verena Haunschmid设计的在中创建的图形。我要感谢所有允许我使用他们研究文章中的图片的研究人员。

至少在三个方面，我出版这本书的方式是非常规的。首先，它既可以作为网站，也可以作为电子书/pdf提供。我用来创建这本书的软件叫做Bookdown，由他编写，他创建了许多R包，可以很容易地将R代码和文本结合起来。谢谢！其次，我在平台上自行出版这本书，而不是与传统出版商合作。第三，我把这本书作为正在进行中的书出版，这极大地帮助了我获得反馈，并在这一过程中将其货币化。多亏了Leanpub使这成为可能，公平处理版税。

我也要感谢你，亲爱的读者，阅读这本没有大出版商的书。

感谢巴伐利亚州科学和艺术部在巴伐利亚数字化中心框架内资助我的可解释机器学习研究。

工具书类

“算法定义”。（2017年）。

Aamodt、Agnar和Enric广场。“基于案例的推理：基础问题、方法变化和系统方法。”人工智能通信7.1（1994）：39-59。

阿尔贝托，T\_lio C，约翰内斯诉洛克特，蒂亚戈·阿尔梅达。“TubeSpam:评论垃圾邮件过滤

YouTube，“在机器学习和应用（ICMLA）中，IEEE第14届国际会议，138-43。IEEE。（2015年）。

Alvarez Melis、David和Tommi S.Jaakkola。“关于解释方法的稳健性”，ARXIV预印ARXIV:1806.08049（2018）。

可视化黑盒监督学习模型中预测变量的影响〉，《ARXIV预印本：1612.08468》（2016）。

阿塔利、安尼和伊利亚·萨茨基弗。“合成强大的对抗性示例”，arxiv预印arxiv:1707.07397（2017）。

Biggio、Battista和Fabio Roli。“野生模式：对抗机器学习兴起十年后”，《模式识别》84（2018）：317-331。

《随机森林》，机器学习45（1）。斯普林格：5-32（2001）。Brown，Tom B.等人“对抗补丁。”arxiv预印arxiv:1712.09665（2017）。

快速有效规则归纳〉，《机器学习程序》（1995年）。115-123页。

库克，R.丹尼斯。“线性回归中影响观察的检测”，《技术计量学》19.1（1977）：15-18。

Doshi Velez，终曲，曾是Kim。“向可解释机器学习的严谨科学迈进”，ML:1-13。（2017年）。

Fanaee-t、Hadi和Joao Gama。“集成探测器与背景知识相结合的事件标记”，《人工智能进展》。施普林格柏林海德堡，1-15。（2013年）。

费尔南德斯、凯尔文、杰米·卡多索和杰西卡·费尔南德斯。“应用于宫颈癌筛查的部分可观察性转移学习”，伊比利亚模式识别和图像分析会议，243-50。斯普林格。（2017年）。

Fisher、Aaron、Cynthia Rudin和Francesca Dominici。“从‘拉森蒙’的角度来看，模型类依赖：任何机器学习模型类的可变重要性度量。”（2018年）。

Fokkema、Marjolein和Benjamin Christoffersen。“预：预测规则集合”。（2017年）。

Friedman、Jerome H和Bogdan E Popescu。“通过规则集合进行预测性学习”，《应用统计学年鉴》。Jstor，916–54。（2008年）。

贪婪函数近似：梯度增强机〉，《统计年鉴》（2001）：1189-1232。

弗里德曼、杰罗姆、特雷弗·黑斯迪和罗伯特·提比西拉尼。“统计学习要素”。www.web.stanford.edu/~hassie/elemstatlearn/（2009年）。

F\_rnkranz、Johannes、Dragan Gamberger和Nada Lavra\_。“规则学习的基础”，斯普林格科学与商业媒体（2012年）。

Goldstein、Alex等。“包装‘冰箱’”（2017年）。

Goldstein、Alex等。“窥视黑盒子：用个体条件期望图可视化统计学习”，《计算和图形统计杂志》24.1（2015）：44-65。

Goodfellow、Ian J、Jonathon Shlens和Christian Szegedy。“解释和利用敌对的例子”，arxiv预印arxiv:1412.6572（2014）。

Greenwell、Brandon M、Bradley C.Boehmke和Andrew J.McCarthy。“一个简单有效的基于模型的变量重要性度量”，arxiv预印arxiv:1805.04755（2018）。

海德、弗里茨和玛丽安·西梅尔。“表观行为的实验研究”，《美国心理学杂志》57（2）。JSTOR:243–59。（1944年）。

《机器学习》11.1（1993）：63-90，〈非常简单的分类规则在最常用的数据集上表现良好〉。

贾尔斯·胡克。“发现黑盒函数中的加性结构”，第十届ACM Sigkdd知识发现和数据挖掘国际会议论文集。（2004年）。

卡尼曼、丹尼尔和阿莫斯·特沃斯基。“模拟启发式”，斯坦福大学CA心理学系。（1981年）。

考夫曼、伦纳德和彼得·卢梭。“通过类药物聚集”。北荷兰（1987）。

Kim、Be、Rajiv Khanna和Oluwasanmi O.Koyejo。“榜样不够，要学会批评！对可解释性的批评〉，《神经信息处理系统的进展》（2016年）。

Kim，Be等人“特征属性之外的可解释性：概念激活向量（TCAV）的定量测试。”ARXIV预印ARXIV:1711.11279（2017）。

Koh、庞伟和Percy Liang。“通过影响函数了解黑盒预测”，arxiv预印arxiv:1703.04730（2017）。

Laugel，Thibault等人“机器学习中基于比较的可解释性的逆向分类”，ARXIV预印ARXIV:1712.08443（2017）。

Letham，Benjamin等人“使用规则和贝叶斯分析的可解释分类器：建立更好的中风预测模型。”《应用统计学年鉴》9.3（2015）：1350-1371。

彼得·利普顿。“对比解释”，《皇家哲学研究所补编》27（1990）：247-266。

模型可解释性的神话〉，《arxiv预印arxiv:1606.03490》（2016）。

伦德伯格，斯科特和苏在李。“模型预测解释方法的意外统一”，ARXIV预印ARXIV:1611.07478（2016）。

马丁斯、大卫和福斯特教务长。“解释数据驱动文档分类。”（2014年）。

米勒，蒂姆。“人工智能的解释：社会科学的见解”，arxiv预印本：1706.07269。（2017年）。

Nguyen，Anh等人“即插即用生成网络：潜在空间图像的条件迭代生成”，《IEEE计算机视觉和模式识别会议论文集》。2017年。

Nguyen，Anh等人“通过深生成器网络合成神经网络中神经元的首选输入。”神经信息处理系统的进展。2016年。

确认偏差：在许多方面普遍存在的现象〉，《普通心理学评论》2（2）。教育出版基金会：175.（1998年）。

Olga Russakovsky、Jia Deng、Hao Su、Jonathan Krause、Sanjeev Satheesh、Sean Ma、Zhiheng Huang、Andrej Karpathy、Aditya Khosla、Michael Bernstein、Alexander C.Berg和Li Fei Fei。（\*=贡献均等）ImageNet大规模视觉识别挑战。IJCV，2015年

Papernot，Nicolas等人“针对机器学习的实际黑盒攻击”，《2017年亚洲计算机与通信安全会议论文集》。ACM（2017年）。

Ribeiro、Marco Tulio、Sameer Singh和Carlos Guestrin。“锚：高精度模型不可知论解释”，AAAI人工智能会议（2018）。

Ribeiro、Marco Tulio、Sameer Singh和Carlos Guestrin。“机器学习的模型不可知论可解释性”，ICML机器学习中的人类可解释性研讨会。（2016年）。

Ribeiro、Marco Tulio、Sameer Singh和Carlos Guestrin。“我为什么要相信你？：解释任何分类器的预测〉，《第22届ACM Sigkdd知识发现和数据挖掘国际会议论文集》。ACM（2016年）。

劳埃德S.沙普利，“n人博弈的价值”，《博弈论的贡献》2.28（1953）：307-317。

Staniak、Mateusz和Przemyslaw Biecek。“使用实时和故障包的模型预测说明”，ARXIV预印ARXIV:1804.01955（2018）。

苏、加威、达尼洛·瓦斯卡塞洛斯·瓦加斯和樱井。“愚弄深度神经网络的单像素攻击”，《IEEE进化计算汇刊》（2019年）。

Szegedy，Christian等人“神经网络的有趣特性”，arxiv预印arxiv:1312.6199（2013）。

瓦赫特、桑德拉、布伦特·米特斯塔特和克里斯·拉塞尔。“不打开黑匣子的反事实解释：自动决策和GDPR。”（2017）。

杨、洪宇、辛西娅·鲁丁和玛戈·塞尔茨。“可扩展贝叶斯规则列表”，《第34届国际机器学习会议论文集》第70卷。jmlr.组织，2017年。

赵、清远和特雷弗·黑斯迪。“黑匣子模型的因果解释”，《商业与经济统计杂志》即将出版。（2017年）。

\_Trumbelj、Erik和Igor Kononenko。“一种可视化和解释黑盒回归模型的一般方法”，在国际自适应和自然计算算法会议上，21-30。斯普林格。（2011年）。

\_Trumbelj、Erik和Igor Kononenko。“用特征贡献解释预测模型和个人预测”，知识和信息系统41.3（2014）：647-665。

用于示例的R包

基础。R核心团队（2017年）。R：统计计算的语言和环境。R统计计算基金会，奥地利维也纳。统一资源定位地址

数据表。Matt Dowle和Arun Srinivasan（2019年）。data.table:data.frame的扩展。R包版本1.12.2。

普里尔。Hadley Wickham、Romain Fran\_ois、Lionel Henry和Kirill M\_ller（2019年）。数据处理语法。R包版本0.8.3。

**ggplot2**. Hadley Wickham, Winston Chang, Lionel Henry, Thomas Lin Pedersen, Kohske Takahashi, Claus Wilke, Kara Woo and Hiroaki Yutani (2019). ggplot2: Create Elegant Data Visualisations Using the Grammar of Graphics. R package version 3.2.0. <https://CRAN.R-project.org/package=ggplot2>

**iml**. Christoph Molnar (2019). iml: Interpretable Machine Learning. R package version 0.9.0. [https:// CRAN.R-project.org/package=iml](https://CRAN.R-project.org/package=iml)

**knitr**. Yihui Xie (2019). knitr: A General-Purpose Package for Dynamic Report Generation in R. R package version 1.23. <https://CRAN.R-project.org/package=knitr>

**libcoin**. Torsten Hothorn (2019). libcoin: Linear Test Statistics for Permutation Inference. R package version 1.0-4. <https://CRAN.R-project.org/package=libcoin>

**memoise**. Hadley Wickham, Jim Hester, Kirill Müller and Daniel Cook (2017). memoise:

Memoisation of Functions. R package version 1.1.0. <https://CRAN.R-project.org/package=memoise>

**mlr**. Bernd Bischl, Michel Lang, Lars Kotthoff, Julia Schiffner, Jakob Richter, Zachary Jones, Giuseppe Casalicchio, Mason Gallo and Patrick Schratz (2019). mlr: Machine Learning in R. R package version 2.14.0. <https://CRAN.R-project.org/package=mlr>

**mvtnorm**. Alan Genz, Frank Bretz, Tetsuhisa Miwa, Xuefei Mi and Torsten Hothorn (2019). mvtnorm: Multivariate Normal and t Distributions. R package version 1.0-11. [https://CRAN.R-project.org/ package=mvtnorm](https://CRAN.R-project.org/package=mvtnorm)

**NLP**. Kurt Hornik (2018). NLP: Natural Language Processing Infrastructure. R package version

0.2-0. <https://CRAN.R-project.org/package=NLP>

**ParamHelpers**. Bernd Bischl, Michel Lang, Jakob Richter, Jakob Bossek, Daniel Horn and Pascal

Kerschke (2019). ParamHelpers: Helpers for Parameters in Black-Box Optimization, Tuning and Machine Learning. R package version 1.12. <https://CRAN.R-project.org/package=ParamHelpers>

**partykit**. Torsten Hothorn and Achim Zeileis (2019). partykit: A Toolkit for Recursive Partytioning. R package version 1.2-4. <https://CRAN.R-project.org/package=partykit>

**pre**. Marjolein Fokkema and Benjamin Christoffersen (2019). pre: Prediction Rule Ensembles. R package version 0.7.1. <https://CRAN.R-project.org/package=pre>

**readr**. Hadley Wickham, Jim Hester and Romain Francois (2018). readr: Read Rectangular Text Data. R package version 1.3.1. <https://CRAN.R-project.org/package=readr>

**rpart**. Terry Therneau and Beth Atkinson (2019). rpart: Recursive Partitioning and Regression Trees. R package version 4.1-15. <https://CRAN.R-project.org/package=rpart>

**tidyr**. Hadley Wickham and Lionel Henry (2019). tidyr: Easily Tidy Data with ‘spread()’ and ‘gather()’ Functions. R package version 0.8.3. <https://CRAN.R-project.org/package=tidyr>

**tm**. Ingo Feinerer and Kurt Hornik (2018). tm: Text Mining Package. R package version 0.7-6. [https:// CRAN.R-project.org/package=tm](https://CRAN.R-project.org/package=tm)

**viridis**. Simon Garnier (2018). viridis: Default Color Maps from ‘matplotlib’. R package version 0.5.1. <https://CRAN.R-project.org/package=viridis>

**viridisLite**. Simon Garnier (2018). viridisLite: Default Color Maps from ‘matplotlib’ (Lite Version). R

package version 0.3.0. <https://CRAN.R-project.org/package=viridisLite>