第一章 绪论

1.1选题背景与研究意义

近几年以来，以大数据、云计算、搜索引擎为代表的互联网技术飞速发展，给人类社会生活的各个层面带来了巨变，也对传统的金融领域的相关业务和一些商业模式进行了变革，快速发展的互联网技术和金融相关业务的紧密结合，催生出了互联网金融服务。当前，以P2P借贷、消费金融、小额贷款等为代表的互联网金融商业模式蓬勃发展。技术的进步促进了金融业务的延伸，解决了很多个人和中小型企业的资金需求，有助于消费升级和经济的发展。2015年，国务院发布了《国务院关于积极推进“互联网+”行动的指导意见》，普惠金融成为文件中重点支持发展的领域。因此近几年互联网金融服务的发展呈现出了勃勃生机，互联网金融平台数、借款投资人数、成交额逐年增长。

金融业务的核心在于风险的管控，互联网金融业务也不例外。风险的经营管理工作对于互联网金融服务的健康快速生长至关重要。由于前期互联网金融业务的野蛮生长，平台“跑路”、用户违约的现象频频出现，引发了人们对网络金融市场的担忧，制约了行业的发展。互联网金融业务的开展面临着诸多的风险，比如操作风险、网络风险、信用风险等等，其中信用风险相对于传统金融业务来说更加突出。信用风险可以分为平台的信用风险和借款人的信用风险。平台的信用风险受到相关政府机构的监管，借款人风险是信用风险的主要来源，因为线上认证方式相对于传统商业银行的线下审核来说很难保证信息的有效性。信息的不对等使得借款平台在交易完成后很难对借款人形成有效监督，所以如何在交易完成前对借款人进行风险评估，对于稳定平台的正常运营、保证借出人利益、促进行业发展具有重要意义。

我国征信体系尚未发展完善，中国人民银行的征信体系尚未向互联网金融机构开放。新巴塞尔协议提出了传统商业银行的三种信用风险评估方法：内部违约经验、外部评级映射以及统计违约模型。传统的商业银行虽然有丰富的客户数据积累和完善的经营管理经验，但是传统商业银行的风控方法不适用于大数据时代的互联网金融服务。一方面，传统的信用评估方法多依赖于规则引擎和专家判断，但互联网金融服务的快速迅速，交易笔数大，依靠人力来完成所有的信用风险评估决策已经变成一件不可能的事情。另一方面，央行的征信数据有一定缺陷。其一央行征信系统中有大量个人没有信贷记录，其二，虽然央行的征信数据规模已经较大、覆盖范围已经较广，但是随着信息技术的快速发展，个人会在社交网络、电商网购、三方运营商等范围内产生大量的有价值的数据，央行并没有将其纳入，所以数据的时效性上也存在一定的缺陷。

基于上面的分析，利用大数据进行个人信用风险评估是解决互联网金融风险管控问题的一条途径。互联网的普及让很多人参与到了信息世界当中，个人在网购、社交等方面产生了大量数据，同时互联网金融服务平台可以获取用户的注册信息、平台登录修改信息、历史借贷信息等，将这些多维度数据进行整合分析，这样可以将分散在不同平台上的有价值的信息融合成更加全面的信息，深度挖掘这些信息，建立信用风险评估模型，对个人信用风险进行评估，这样就解决了传统风控方法对互联网金融服务不适用的现状，弥补数据上的缺陷，成为互联网金融机构进行风险管控的一种强有力的方法。

通过大数据进行个人信用风险评估的本质是通过挖掘多维度数据背后隐藏着的有价值信息，并将其用于建立区分能力强、稳定性好、评估准确的个人信用风险评估模型，而模型的建立需要算法的支持。大数据环境下，用户行为的广泛分布导致数据的稀疏性较强，数据的广泛覆盖导致数据维度较多，在丰富数据全面性的同时也引入了一些区分能力较弱的单变量，这给传统的在业务逻辑的架构下、利用专家经验或统计分析方法建模的方法框架带来了困难。而最近几年机器学习算法的快速迭代为解决这种困难提供了技术上的支持。

机器学习的研究目的是利用计算机强大的计算能力来模拟人脑的学习过程，赋予计算机以智能。可以把机器学习算法看成一种预测技术，它可以从历史数据中学习相关规律，并对未知的数据做出预测。机器学习的实质是在函数空间中搜索出既能较好拟合原始数据又能对未知数据有较好泛化能力的函数。信用风险评估实质上也是一个分类的问题，利用用户多维度的信息和历史活动数据，学习出一个最佳的函数，具有很好的泛化能力对于新的用户进行预测。由于机器学习算法具有很强的处理非线性数据的能力，所以机器学习算法能够很好地应用于个人信用风险评估的场景。

综上所述，研究利用机器学习算法解决大数据时代背景下个人信用评估的问题具有非常强的现实意义。第一，有助于促进各征信机构创新个人信用风险评估方法，将风险评估模型与业务结合。第二，有助于提高互联网金融服务机构的资产质量，提高风险管理的水平，更好地防范系统性风险。第三，有助于促进传统金融机构风险评估观念的转变，利用大数据思维看待贷前准入和贷后管理的各个环节。第四，有助于促进互联网金融这种商业模式的发展，使普惠金融政策落地，让更多人享受到金融服务，促进消费升级。

1.2 国内外研究现状

个人信用风险评估的发展大概可以分为三个阶段。第一阶段主要是定型分析阶段。在20世纪30年代初，信用风险评估开始受到一些金融机构的重视，一直到20世纪70年代，信用风险评估的方法还停留在人为判断，主要依靠信贷委员会的通过各种信息的人工判断，得到最终的评估结果。第二个阶段主要是定量统计分析阶段，1941年左右，David Durand[1]是第一个利用判别分析技术进行风险评估的人，通过统计学的相关方法，来评判一个相关客户的好与坏，信用风险评估由此进入了一个新的研究阶段，完成了从定性分析到定量分析的转变。之后，研究者们提出了很多定量的风险模型。20世纪70年代以来，随着信用卡业务的普及，定量分析的信用评估模型被广泛的应用，相比于主观判断，定量模型使不良贷款率大幅下降[2],给金融机构带来了丰厚的利润。第三个阶段是定量智能分析阶段：20世纪90年代以来，以逻辑回归、神经网络为代表的一些人工智能算法被用来构建信用风险评估模型。传统统计学的建模方法关于线性和分布等方面的一些前提假设在一定程度上限制了统计学建模方法的应用范围，而机器学习的一些算法具有强大的处理非线性数据的能力，突破了传统统计方法，在模型效果方面得到了一定的提升。

在国外，Li和Hand将Logistic回归与其他方法相比，用于风险评估。[3]支持向量机(SVM)是上世纪90年代末提出来的一种用于分类的机器学习算法，Hens 和 Tiwari (2012)

[4]，采用支持向量机作为其进行信用风险评估的主要技术。判别分析(DA)由Fisher在1986年提出，之后Akkoc在2012年将其与其他信用风险评估技术进行了比较[5]。Giudici（2001）[6]提出了利用条件贝叶斯独立图来提取信用评分应用中变量关联结构间的深层信息。随后贝叶斯网络也被用在信用评估的其他方面。决策树（Breiman 等人，1984[7]）是一种利用历史数据构建组织成树状结构的所谓决策规则的分类方法，Kao 等人（2012）[8]提出了基于 CART 的信用评分模型，并尝试与贝叶斯网络模型进行融合。神经网络（Ripley，1996）[9]是一种处理非线性问题的系统，神经网络是机器学习中一个庞大的分支，很多学者尝试了使用不同的人工神经网络来解决信用评估的问题: 概率神经网络（Pang，2005[10]，）部分逻辑回归人工神经网络（Lisboa 等人，2009[11]），突触再塑性神经网络（MarcanoCedeno 等人，2011[12]）和混合神经网络（Chuang 和 Huang，2011[13]）。Kambal等（2013）基于银行数据，利用遗传算法进行变量筛选并人工神经网络（ANN） 的方法进行信用评价。

国内传统的金融机构利用机器学习算法进行信用风险评估模型的研究起步较晚，虽然国内的传统商业银行已经建立以较为完善的风险管理机制，但由于实际情况的要求，中小商业银行的信用风险评估还停留在定性分析的阶段，少部分银行采用了一些定量统计分析建模的方法。但随着消费金融等业务的发展、数据维度的丰富，越来越多的金融机构开始重视利用机器学习算法构建量化模型进行风控，这方面的研究最近几年也开始丰富起来。陈建（2005 年）研究通过人工智能相关技术对信用卡交易进行评分[14]；严华、胡孟梁等人(2006 年)在使用贝叶斯分类算法对信用卡风控历史数据进行了挖掘，对参与信用卡交易的客户进行分类[15]。Ling 等人（2012）采用支持向量机作为其信用评分的主要技术[16]。伍保华(2010年)则将 BP 神经网络技术应用于信用卡评分的相关研究中[17]。

1.3 本文研究内容与结构安排

1.3.1 本文的研究内容

本文的研究内容主要是在大数据背景之下，以互联网金融机构的真实脱敏数据为基础，探究处理高维度、稀疏、不平衡数据的处理方法，使用传统统计分析和机器学习方法分别建模，比较传统模型和机器学习模型之间的效果。实现并改进基于随机森林和神经网络的深度网络结构的一些前沿算法，并将其用于个人信用风险评估，提升模型的效果。最后对多个单模型进行模型的融合，研究stacking在模型效果提升上的作用。

1.3..2 本论文的结构安排

第一章，绪论，介绍了本文的研究背景以及研究意义，指出互联网、大数据蓬勃发展的时代背景下，研究个人信用风险评估模型对于互联网金融的重要意义。简要介绍了信用风险评估的国内外研究现状和研究方法。

第二章，介绍统计分析建模数据处理方法及相关技术，并利用Python实现建模。介绍信用风险模型的常用评价指标，观测传统统计模型在数据集上的表现。

第三章，介绍机器学习领域中数据处理的方法及一些经典算法，并利用这些算法进行建模，分别观测单模型在数据集上的指标。

第四章，介绍多粒度扫描级联森林模型，并改进模型算法使之更适用于金融行业的结构化数据，用python实现并验证算法的改进效果。

第五章，介绍模型融合相关理论，并利用stacking技术将单模型及改进的模型进行融合，在各个指标上观测模型融合的提升效果。

第六章，总结与展望，总结本文的相关研究成果，分析还存在的问题与不足，展望模型的后续相关研究。

第二章 基于统计分析方法的建模研究

个人信用风险主要指交易对手未能履行约定契约中的义务而造成经济损失的风险,即受信人不能履行还本付息的责任而使授信人的预期收益与实际收益发生偏离的可能性，它是金融风险的主要类型。在信贷场景中，经常在准入环节对借款人未来一段时间风险进行评估，预测违约、逾期等行为的概率。进行风险评估之前，通常要设定一个观察期与表现期，在观察期内收集样本，以及带时间变量的相关记录，在表现期内根据客户的表现来进行好坏样本的定义。这种评估模型由于是利用历史数据进行建模，所以需要不断更新。模型的开发主要分为以下几个阶段：数据的准备与处理、特征的构建、模型的训练、模型的评估等。

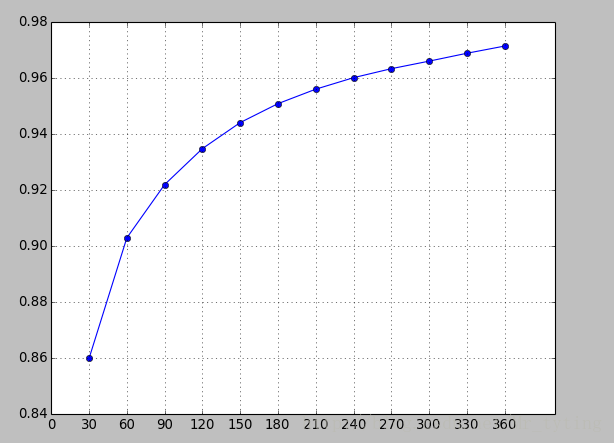
2.1 数据准备与处理

本次采用的数据是某互联网金融机构的真实数据，主要包括三个部分，第一个部分是用户的基本数据，每个字段集都有多维度的信息，其中idx是每个用户的unique key，User\_Info是用户基本特征字段集，WeblogInfo是用户的网络行为字段集，Education\_Info是用户的学历学籍字段集，ThirdParty\_Info\_PeriodN代表第三方数据时间字段集，SocialNetwork代表社交网络字段集，ListingInfo代表借款成交的时间，Target则代表了在观察期内用户是否违约。第二部分是用户在机构平台上登录信息，ListingInfo代表着成交时间，LogInfo1代表的是用户操作代码，LogInfo2代表操作类别，LogInfo3代表着登录时间。第三个部分是用户在机构平台上修改个人信息的一些记录，其中ListingInfo1代表着成交时间，UserupdateInfo1是用户修改的内容，包括联系方式、婚姻等信息。UserupdateInfo2代表着用户修改上述信息的时间。

对于第一部分数据，有一定的缺失值，对于类别型变量，当变量的缺失超过50%，则删除此变量；当缺失率在30%~50%时，则把缺失本身当成是一种类别；当缺失率低于30%的时候在剩余变量值中进行随机抽样进行填补。对于连续型变量，缺失率不高时可以根据数据的近似分布进行插补。

对于第二部分和第三部分数据，都是和时间序列有关的数据，需要经过处理才能使用，这里采用时间切片的方法来提取特征。时间切片是指两个时刻间的跨度，时间切片的选取有一定的要求，要保证数据的覆盖度与相关性，通过数据分析统计可见，由图1所示

，



选取借款前180天的数据在操作数上可以覆盖到95%的用户，180天之后覆盖度增长较慢，所以我们基于180天之内的窗口来做特征衍生的工作。对于我们第二部分和第三部分数据，首先进行异常数据检测，统计数据中是否出现在迭代后出现的用户记录，并将其删除。检测完毕后，按照统计出的符合要求的时间切边进行变量的衍生，分别统计出借款前5、15、30、90、150、180天登录操作次数、不同操作次数、平均每天不同操作类型的操作次数、更新信息的次数、更新信息的种类、平均每种信息类型的更新次数以及是否修改手机号码等重要信息等等特征。

2.2 特征工程构建

2.2.1 特征的分箱

分箱方法是一种简单常用的预处理方法，通过考察相邻数据来确定最终值。所谓“分箱”，实际上就是按照属性值划分的子区间，如果一个属性值处于某个子区间范围内，就称把该属性值放进这个子区间所代表的“箱子”内。将连续变量离散化是进行分箱的目的。分箱主要具有以下优点：

（1）离散特征的增加和减少都很容易，易于模型的快速迭代

（2）稀疏向量内积乘法运算速度快，计算结果方便存储，容易扩展

（3）离散化后的特征对异常数据有很强的鲁棒性

（4）离散化后可以进行特征交叉，由M+N个变量变为M\*N个变量，进一步引入非线性，提升表达能力

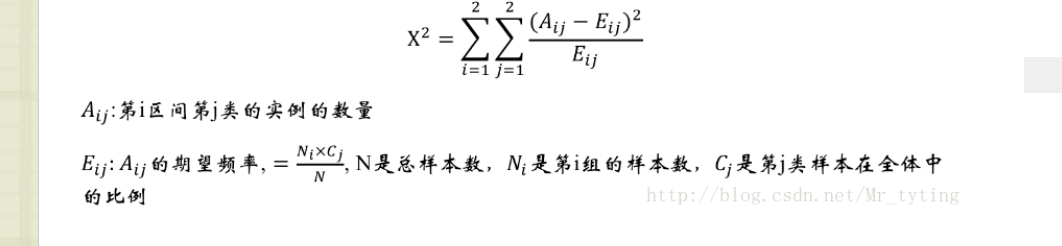
（5）单变量离散化为N个后，每个变量有单独的权重，相当于为模型引入了非线性，能够提升模型表达能力

特征分箱有多重方法，可以根据对目标变量的需求划分为有监督分箱和无监督分箱，有监督分箱主要可以分为Best-KS和ChiMerge两种方法，无监督分箱主要可以分为等频、等距和聚类等几种方法。这里我们主要采用有监督的卡方分箱法(ChiMerge)，这是一种自底向上、基于合并的数据离散化方法，它依赖于卡方检验:具有最小卡方值的相邻区间合并在一起,直到满足确定的停止准则。它的思想是对于精确的离散化，相对类频率在一个区间内应当完全一致。因此,如果两个相邻的区间具有非常类似的类分布，则这两个区间可以合并；否则，它们应当保持分开。而低卡方值表明它们具有相似的类分布。

分箱的基本步骤如下：

（1）预先设定一个卡方的阈值，然后进行初始化，将需要离散的特征进行排序，每个实例划分成一个区间。

（2）计算每一对相邻区间的卡方值，并将卡方最小的一对区间进行合并。

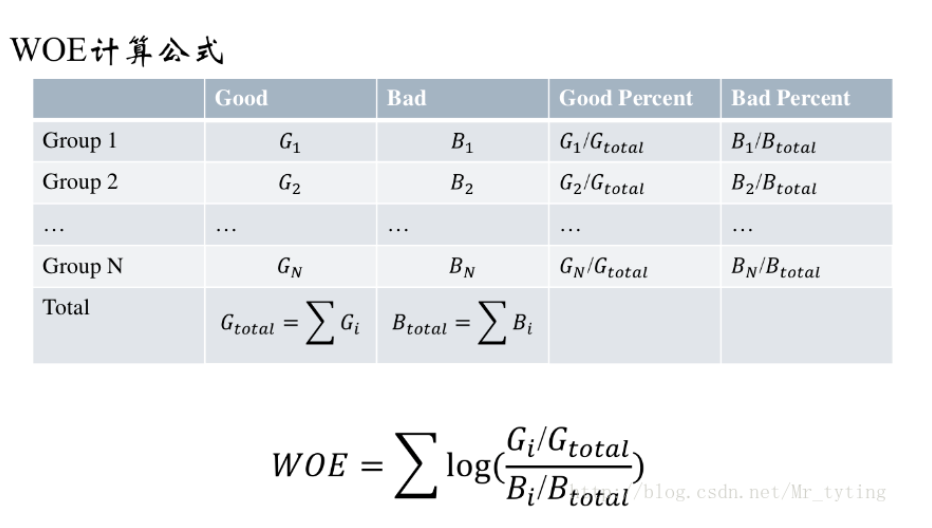


关于阈值的选择，类别和属性独立时,有90%的可能性,计算得到的卡方值会小于4.6。大于阈值4.6的卡方值就说明属性和类不是相互独立的，不能合并。如果阈值选的大,区间合并就会进行很多次,离散后的区间数量少、区间大。ChiMerge算法推荐使用0.90、0.95、0.99置信度,最大区间数取10到15之间.有时也可以不考虑卡方阈值,此时可以考虑最小区间数或者最大区间数。指定区间数量的上限和下限,最多几个区间,最少几个区间。

在此数据集上，对连续型变量使用ChiMerge的方法进行分箱，如果数据中包含有特殊值，比如-1，则单独列为一箱；对类别型的变量，如果变量类别数较少，原则上可以不用分箱，当类别较多时可以用类别的bad rate代替原来的值，转换成连续型变量之后再采用上述的方法进行分箱计算，对本数据集用5作为是否转换的阈值。完成分箱后，需要检查分箱后每箱bad rate的单调性，如果不满足，则将相近的两箱合并，直到变量的bad rate单调（有时可以放宽到U型）。当某个箱的bad rate为0，则需要和最小的bad rate的箱体进行合并。所有工作完成后，需要对每个变量的分箱情况进行检查，当某一箱体的数据量占据总数据量的85%以上时，对此变量进行删除。

2.2.2 WOE编码

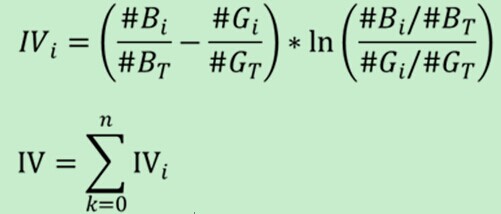
对数据进行分箱处理过后，需要进行WOE编码才能放入模型。WOE(weight of evidence), 证据权重，一种有监督的编码方式,将预测类别的集中度的属性作为编码的数值。WOE编码具有许多优势，因为WOE编码可以将特征的值规范到一定的尺度，从经验上讲，WOE的绝对值波动范围大约在0.1~3之间，并且经过WOE编码之后，就具有一定的业务意义。WOE的计算公式如下：



WOE编码之后，自变量其实具备了某种标准化的性质，也就是说，自变量内部的各个取值之间都可以直接进行比较（WOE之间的比较），而不同自变量之间的各种取值也可以通过WOE进行直接的比较。进一步地，可以研究自变量内部WOE值的变异（波动）情况，结合模型拟合出的系数，构造出各个自变量的贡献率及相对重要性。一般地，系数越大，woe的方差越大，则自变量的贡献率越大（类似于某种方差贡献率）。

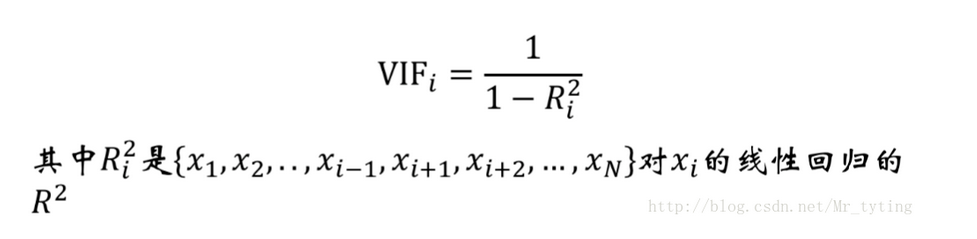
2.2.3 IV值及VIF值

IV的全称是Information Value，即信息价值或信息量。IV值是一个在多维度变量中挑选变量的方法。挑选变量有多方面的因素需要考虑，比如变量间的相关性、变量的简单性（变量易生成和试用），变量的健壮性以及变量在业务上的可解释性等等，但其中最重要的和最直接的衡量标准是变量的预测能力。IV就是这样一种指标，用来衡量每自变量的预测能力，并根据这些量化指标的大小，来确定何种变量进入模型。IV值的计算公式如下：



高的IV值表示该特征和目标变量的关联度较高，但过高的IV也存在潜在风险（比如需要检查是否是时候变量），通常认为0.02以上的IV值具有可预测性。

VIF（Variance Inflation Factor），即方差膨胀引子，是指解释变量之间存在多重共线性时的方差与不存在多重共线性时的方差之比。VIF的计算公式如下：



通常当VIF大于10时，多重共线性较为严重，需要逐一剔除变量，通常是按照剔除IV值小的变量的规则进行。通常情况下，计算VIF这一步不是必须的，在进行单变量处理以后，放进逻辑回归模型进行训练预测，如果效果非常不好时，才需要做多变量分析，消除多重共线性。

2.3 模型的评估

在建模场景里，当模型建立以后，需要评估模型的效果以决定模型是否可用。根据不同的模型类型和模型场景，模型也有不同的度量评估方法。预测模型分为回归模型与分类模型，用于评价这两种模型的度量是不同的。在个人信用风险评估的场景中，我们面对的是一个分类的问题。在分类问题中有两种算法，一是类别输出型，比如支持向量机、KNN等，输出的是最终类别0或1,。二是概率输出型，如逻辑回归、随机森林等，是以概率作为输出的，需要通过一个阈值，将输出的概率转换为输出的类别。下面介绍分类问题常用的评估指标：

2.3.1 混淆矩阵

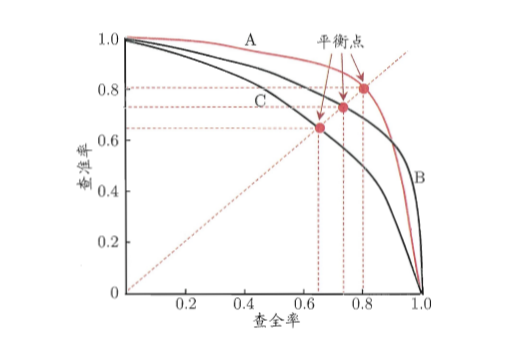
考虑一个二分问题，即将实例分成正类（positive）或负类（negative）。对一个二分问题来说，会出现四种情况。如果一个实例是正类并且也被 预测成正类，即为真正类（True positive）,如果实例是负类被预测成正类，称之为假正类（False positive）。相应地，如果实例是负类被预测成负类，称之为真负类（True negative）,正类被预测成负类则为假负类（false negative）。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | 预测 |  |  | |
|  | 1 | 0 | 合计 |
| 实际 | 1 | True Positive（TP） | False Negative（FN） | Actual Positive(TP+FN) |
|  | 0 | False Positive（FP) | True Negative(TN) | Actual Negative(FP+TN) |
| 合计 |  | Predicted Positive(TP+FP) | Predicted Negative(FN+TN) | TP+FP+FN+TN |

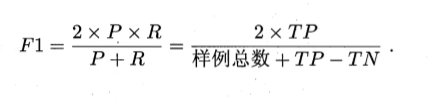
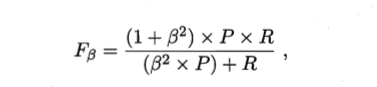
从上述列表可以引入两个新的变量，其一是真正类率(true positive rate ,TPR), 计算公式为TPR=TP/ (TP+ FN)，刻画的是分类器所识别出的 正实例占所有正实例的比例。另外一个是假正类率(false positive rate, FPR),计算公式为FPR= FP / (FP + TN)，计算的是分类器错认为正类的负实例占所有负实例的比例。还有一个真负类率（True Negative Rate，TNR），也称为specificity,计算公式为TNR=TN/ (FP+ TN) = 1-FPR。除此之外，还可以定义出查准率(precision)的公式是P=TP/(TP+FP),它计算的是所有"正确被检索的item(TP)"占所有"实际被检索到的(TP+FP)"的比例.;查全率(recall)的公式是R=TP/(TP+FN),它计算的是所有"正确被检索的item(TP)"占所有"应该检索到的item(TP+FN)"的比例;准确率（Accuracy）的计算公式是ACC=(TP + TN) / (P+N)，它计算的是预测对的正负样本数之和占总样本数的比例。

2.3.2 PR曲线与F1度量

在很多的情况下，可以根据学习器的预测结果对样例的预测结果进行排序，将排序靠前的样本当做最为”可能”的正样本，而排序较为靠后的则是学习器认为“最不可能的”正样本，按照这种规则将样本进行排序，则每次可以计算出当前的查准率与查全率，以查准率为纵轴、查全率为横轴画一张图，然后就可以得到查准率-查全率曲线，可以简称为“P-R曲线”，“P-R曲线”的示意图如下图所示：



P-R图直观地显示出了学习器在样本总体上的查全率、查准率，在进行学习器的评估比较时，若一个学习器的P-R曲线被另一个学习器的P-R曲线完全包含，则可以断定被包含的学习器的性能不如后者，如上图中曲线A完全包含了曲线C，则学习器A的性能优于学习器C。但有时也会出现P-R曲线出现交叉的情况，比如曲线A与曲线B，很难直接一般性地判定学习器A与学习器B的性能优劣。这时可以把P-R曲线下的面积作为一个比较合理的判据，它在一定程度上表征了学习器在查准率和查全率两个指标上都取得较高值的比例，但这个值不太容易估算，因此需要设计一些综合考虑查准率和查全率的度量。

“平衡点”（Break-Event Point，简称BEP）就是这样一个度量，它是查全率等于查准率时的取值，如上图中学习器C的BEP是0.64，而基于BEP的比较，可以认为学习器A优于B。此外还可以用F1度量来综合考虑查准率和查全率，F1度量的计算公式为然而在不同场景中，查准率和查全率的重要程度是不同的，比如在商品推荐系统中，希望尽可能地少对用户造成困扰，此时查准率较为重要；在我们的个人信用风险评估场景中，我们希望尽可能少的漏掉可能违约的用户，所以查全率更为重要。基于此，我们可以根据场景需要给予查准率和查全率不同的权重，由此给出F1度量的一般形式: 

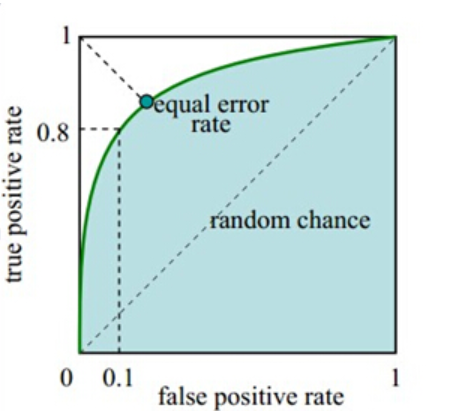
其中B>0度量了查全率对查准率的相对重要性。B>1时查全率具有更大影响；B<1时查准率有更大的影响。

2.3.3 ROC曲线与AUC值

对于概率输出型模型，会为测试样本输出一个概率预测，然后将这个预测与一个分类阈值(threshold)进行比较，若大于这个设定的阈值，则分为正类，否则则将其分为反类。比如神经网络在一般情况下会对每个测试样本输出一个[0.0,1.0]之间的实值，然后将这个值与0.5进行比较，大于0.5则判定为正例，否则为反例。这个概率预测的好坏程度直接就决定了学习器的泛化能力。在实际评判中，可以根据概率的预测结果，对测试样本进行排序，将最有可能为正例的样本排在前面，将最不可能是正例的样本排在后面，这样分类过程就相当于在这个排序当中以某个值当做“截断点”，将样本分为两个部分，前一部分判作正例，另一部分判作反例。

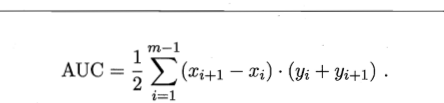
在不同的任务当中，我们可以根据不同的任务需求来选择不同的截断点，例如在更重视查准率的的情况下，我们可以选择排序中较为靠前的点作为截断点；若更加重视查全率，则可以选择较为靠后的点作为截断点。因此，排序本身质量的好坏，体现了综合考虑学习器在不同的任务下“期望泛化性能”的好坏，或者一般情况下泛化性能的好坏。因此，ROC曲线则是从这个角度出发来研究学习器泛化性能强有力的工具。

ROC的全称是“受试者工作特征”(Receiver Operating Characteristic)曲线，它源于二战中的敌机雷达信号分析技术，之后开始被用于一些心理学、医学检测当中，此后被引入了机器学习领域。与PR曲线类似，根据学习器的测试结果进行排序，每次计算出混淆矩阵中的两个量，分别以它们为横纵坐标、纵坐标作图，就得到了ROC曲线。与P-R曲线使用查准率与查全率作为横纵坐标不同，ROC曲线的纵轴是真正类率(TPR)，横轴是假正类率(FPR)，如下图所示

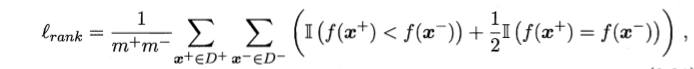


显然，对角线对应于“随机猜测”模型，而左上角点(0,1)则对应于将所有正例排在所有反例之前的理想模型。利用ROC曲线进行模型效果比较时，与P-R曲线类似，若一个学习器的曲线把另一个学习器的曲线完全包住，则可以断言前者的性能要优于后者；若两个模型的ROC曲线发生交叉，则不能下一般性的结论。此时若需要比较两个学习器的效果，则可以利用AUC进行比较(Area Under ROC Curve)。

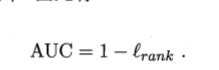
由定义可知，AUC可以通过对ROC曲线下各部分的面积求和得到，假定ROC曲线的坐标为的点按序连接而形成的，则AUC可以估算为：



形式化地看，AUC考虑的是样本预测的排序质量，因此它和排序的误差有着非常密切的联系，给定m+个正例和m-个反例，令D+和D-分别表示正、反例集合，则排序的损失loss可以定义为



即考虑每一对正、反例，若正例的预测值小于反例，则记一个惩罚，若相等，则记0.5个罚分，容易看出lrank对应的是ROC曲线之上的面积：若一个正例在ROC曲线上对应标记点的坐标为(x,y)，则x恰是排序在其之前的反例所占的比例，即假正例率，因此有



2.3.4 KS曲线与AR值

K-S曲线图（Kolmogorov-Smirnov chart）是用来评估分类模型表现的图。更准确的来说，K-S是用来度量正样例与负样例分类区分程度的。K-S曲线的画法和ROC曲线异曲同工，当我们用模型作为评估，然后取不同的概率值作为分类的阈值，计算出不同的假正类率(FPR)和真正类率(TPR),接下来分别用TPR的值、FPR的值作为纵坐标的值，画出的两条曲线即为K-S曲线。

从K-S曲线能够衍生出KS值，其中KS=max（TPR-FPR），若模型能够把所有样本严格按照正样例和负样例分成两组，则模型的K-S值为100，如果模型是随机区分正样例与负样例，则模型的K-S值为0.所以分类模型的K-S值都在0到100之间，值越大，模型从阴性数据中区分阳性数据的能力越强。

K-S曲线图能够直观地找出模型中差异最大的分数段，适合用于个人信用风险评估的场景，但同时，KS值只能反映出区分度最大的概率区间，不能反映所有概率区间的效果，所以评价模型时需要多指标综合考虑。

AR(Accuracy Ratio)是衡量模型预测能力的指标，需要一个完整的表现周期，主要表征模型区分负样例的能力。同样的，以总样例的累积比例作为X轴，以负样例占总样例的累积比例作为Y轴，AR值等于模型在随机模型之上的面积除以理想模型在随机模型之上所占的比例。AR值越高，则模型的区分效果就越好。

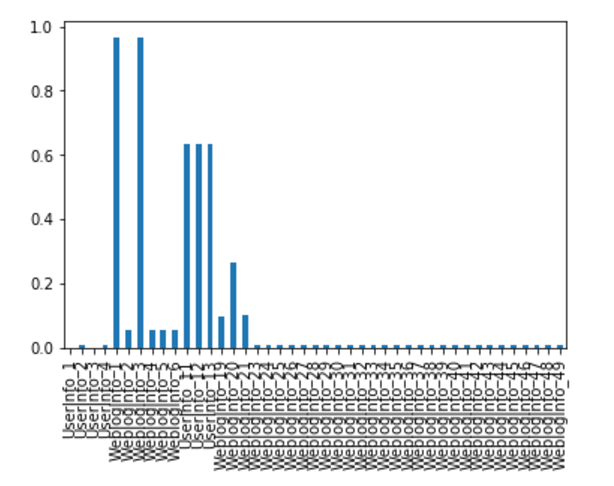
第三章 基于经典机器学习算法的建模研究

通过上一章的实验分析可以看出，传统的统计建模方法在多维度、高稀疏、单变量相关性较弱的互联网金融机构数据上的表现并不是很好，因此我们需要探索新的数据分析及建模算法，以适用于新的数据环境。机器学习算法的快速发展为我们解决这类问题提出了新的思路。这一章将介绍常用的机器学习算法以及建模流程，并用经典的机器学习算法进行建模，与传统统计建模的模型效果进行比较。

3.1 数据的准备与处理

3.1.1 缺失值的处理

对于数据的第一部分，首先统计只有一个值的字段并将其去除，因为这种变量没有区分能力。然后统计各变量的缺失情况，如下图所示：



从图中可以看出有些变量的缺失率已经到了90%以上，然后将其剔除。从样本维度考虑缺失问题，将缺失特征维度大于30的变量进行剔除。对于类别型变量，当缺失率相对较高，将缺失单独作为一种类别；当缺失率不是特别高时，用众数填充，保持数据的分布。对于连续型变量，统计数值型变量的方差，将方差过小、区分能力较弱的变量以及缺失率过高的变量进行剔除；当缺失率不是特别高，当数据近似服从正态分布，则用均值进行填充，当数据是长尾数据时，利用中位数进行填补以保持数据的排序特性。

3.1.2 One-Hot编码

One-Hot编码，即独热编码，又称一位有效编码，其方法是用N位状态寄存器来对N个状态进行编码，每个状态都由独立的寄存器位来表示，并且在任意时刻只有其中一位有效。假设某个属性的取值为非数值的离散集合[离散值1，离散值2，…，离散值m]，则针对该属性的编码为一个m元的元组：

(V1,V2,…,VM),vi属于{0,1}，i=1,2,…,m

且(V1,V2,…,VM)有且仅有一个为1，其余的分量均为0。

比如对于性别属性[男，女]，则男性可以编码为(0,1),女性可以编码为(1,0)。对于国籍属性[中国，美国，英国]，则中国可以编码为(1,0,0)，美国可以编码为(0,1,0)，英国可以编码为(0,0,1)，变量之间的编码还可以进行拼接。由此可以看出One-Hot编码具有以下优点：

1.能够处理非数值型属性

2.在一定程度上扩充了特征

3.编码后的属性是稀疏的，存在大量的零分量元素，有利于计算。

3.2 特征工程的构建

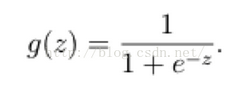
第一部分数据中有四列关于地理位置信息的特征，在统计建模中经常会忽略这些信息，但是在大数据的数据环境中，可以充分挖掘各种信息，因此我们可以对地理信息进行处理。机器学习中，对于类别型数据经常采用One-Hot编码的技术，但是如果直接对地理信息采用one-hot编码，则会产生几百维极为稀疏的矩阵，产生维度灾难。因此从以下几个方面提取信息。一.对各省的违约率进行统计，将UserInfo\_7字段进行映射。经统计，四川、山东、吉林等7个省违约率较高，其他的省份分为一类，映射完之后再进行One-Hot编码。二.将城市映射为城市等级。对城市进行按类型合并，比如将北京、上海、深圳、广州归为一类城市，赋值为1。同理，将二线城市合并为2，三线城市合并为3。三.引入经纬度进行划分。以上对地理位置信息的处理都是基于类别类型，除此之外，我们可以收集各个城市的经纬度，这样就可以把类别型变量转换为连续型变量。比如北京市，可以用(39.92,116.46)替换，得到北纬和东经两个数值型特征。

第二部分数据，对于带有时间序列的数据，仍然按照时间切片的方法进行处理，与2.1节中类似，在此不再赘述。

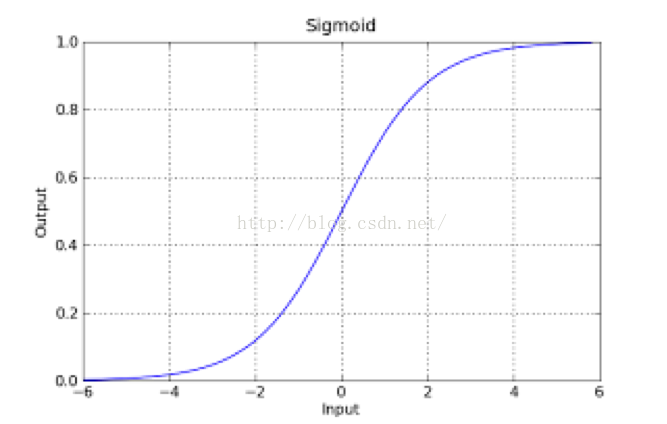
3.3 模型的建立

3.3.1逻辑回归

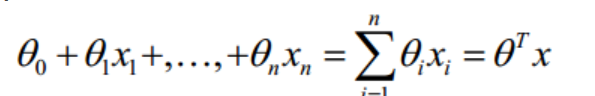
回归分析用来描述自变量x和因变量Y之间的关系，或者说自变量X对因变量Y的影响程度，并对因变量Y进行预测。其中因变量是我们希望获得的结果，自变量是影响结果的潜在因素，自变量可以有一个，也可以有多个。一个自变量的叫做一元回归分析，超过一个自变量的叫做多元回归分析。回归分析经常用在解决需要预测的变量是连续型变量的情况下，比如根据房屋的条件进行房屋价格的预测。而当我们需要解决分类问题时，就会用到逻辑回归(Logistic Regression)。它的核心思想是，如果线性回归的结果输出是一个连续值，而值的范围是无法限定的，那我们需要寻找一个办法把这个结果值映射为可以帮助我们判断的结果。而如果输出结果是 (0,1) 的一个概率值，这个问题就可以得以解决。由此我们很容易想到利用sigmoid函数进行映射，其公式为：



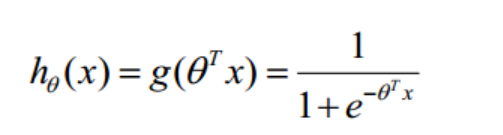
其函数图像为：

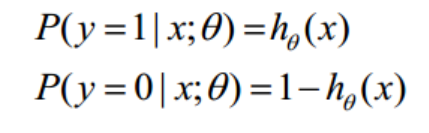


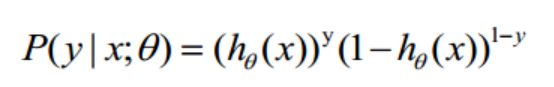
从函数图上可以看出，函数y=g(z)在z=0的时候取值为1/2，而随着z逐渐变小，函数值趋于0，z逐渐变大的同时函数值逐渐趋于1，而这正是一个概率的范围。所以我们定义线性回归的预测函数为Y=WTX，那么逻辑回归的输出Y= g(WTX)，其中y=g(z)函数正是上述sigmoid函数

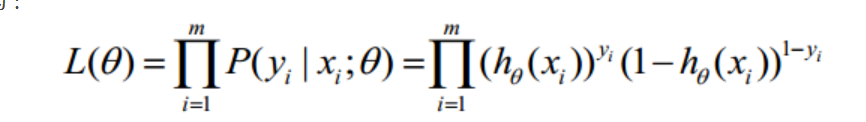
对于分类边界为线性边界的情况，边界的形式如下

于是构造预测函数为：

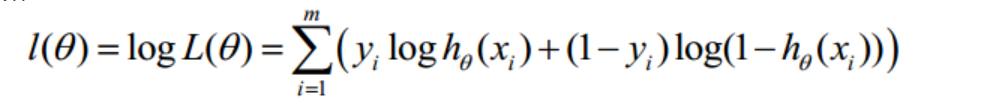


函数的值有特殊的含义，它表示结果取1的概率，因此对于输入x分类结果为类别1和类别0的概率分别为：（1）

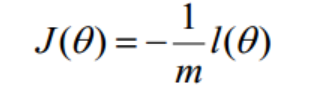
因为我们需要求解参数，所以我们基于最大似然估计来推导损失函数，假设样本之间服从二项分布，则(1)可以综合写成

取似然函数为

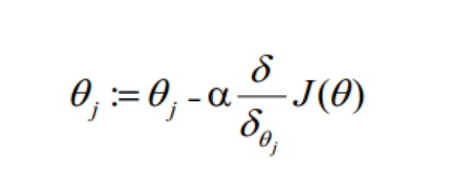
取对数似然为

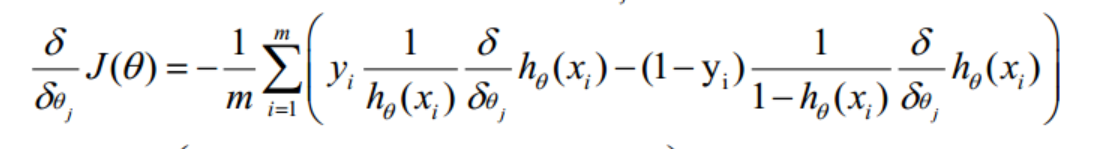


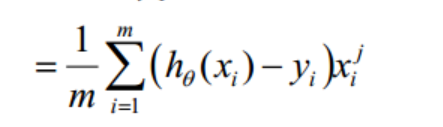
最大似然估计就是求使http://img.my.csdn.net/uploads/201407/16/1405496927_9629.png取最大值时的θ，其实这里可以使用梯度上升法求解，求得的θ就是要求的最佳参数。为了利用梯度下降法求解最佳参数，我们乘以一个负系数，最终的损失函数为：



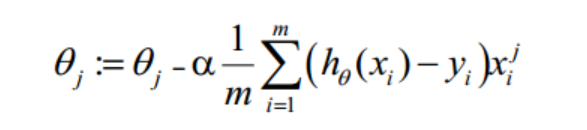
利用梯度下降法求解参数的更新过程：



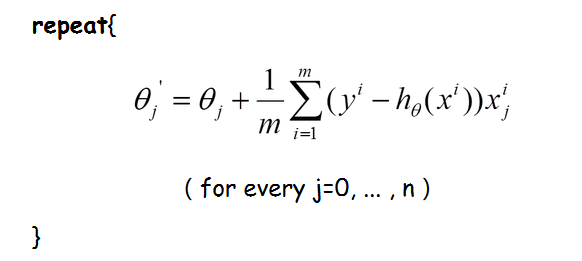
其中



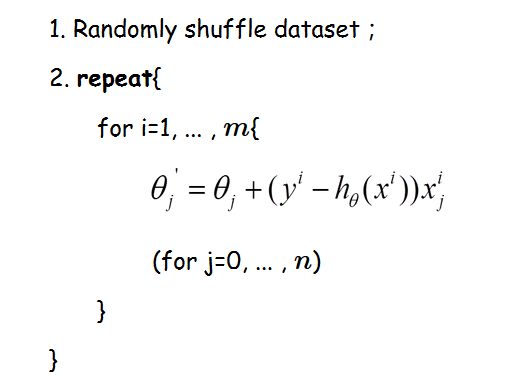
所以0000的更新过程为：



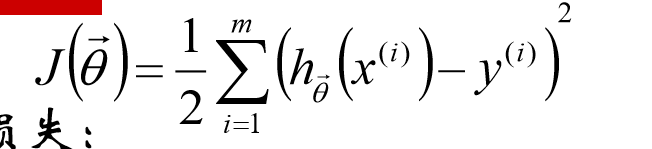
使用梯度下降法进行训练时，通常有三种方法批量梯度下降法BGD、随机梯度下降法SGD、小批量梯度下降法MBGD。批量梯度下降法（Batch Gradient Descent，简称BGD）是梯度下降法最原始的形式，它的具体思路是在更新每一参数时都使用所有的样本来进行更新，其伪代码形式如下：

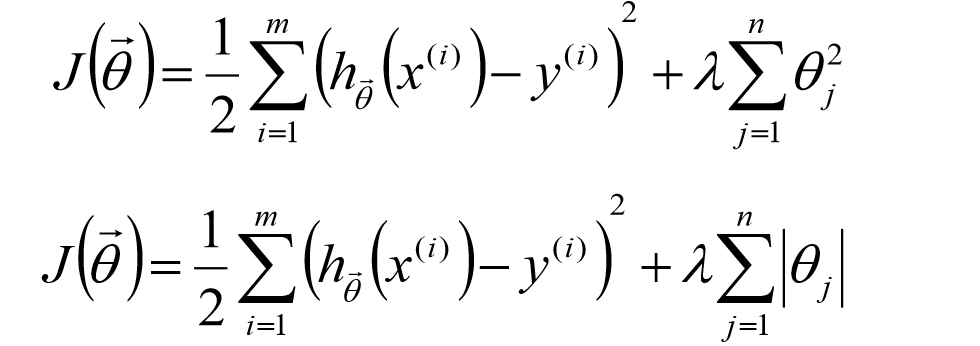


从上面公式可以注意到，它得到的是一个全局最优解，但是每迭代一步，都要用到训练集所有的数据，如果样本数目m很大，迭代速度也会较慢。

由于批量梯度下降法在更新每一个参数时，都需要所有的训练样本，所以训练过程会随着样本数量的加大而变得异常的缓慢。随机梯度下降法（Stochastic Gradient Descent，简称SGD）正是为了解决批量梯度下降法这一弊端而提出的。其伪代码是

随机梯度下降是通过每个样本来迭代更新一次，如果样本量很大的情况（例如几十万），那么可能只用其中几万条或者几千条的样本，就已经将theta迭代到最优解了。但是SGD的问题是噪音较多，使得 SGD并不是每次迭代都向着整体最优化方向。上述两种方法，各自均有优缺点，于是可以在两种方法中寻找一个折衷，即，算法的训练过程比较快，而且也要保证最终参数训练的准确率，产生了小批量梯度下降法（Mini-batch Gradient Descent，简称MBGD）。MBGD综合了上述两种方法的特点，每次使用一定数量的训练样本进行迭代。

逻辑回归通常会面临过拟合的问题，对于线性回归或逻辑回归的损失函数构成的模型，可能会有些权重很大，有些权重很小，导致过拟合（就是过分拟合了训练数据），使得模型的复杂度提高，泛化能力较差（对未知数据的预测能力）。通常会有两种解决方法，第一种是利用特征选择的一些算法选择一些特征，第二种方法即是使用正则化。则化是结构风险最小化策略的实现，是在经验风险上加一个正则化项或惩罚项。正则化项一般是模型复杂度的单调递增函数，模型越复杂，正则化项就越大。比如对于线性回归问题其目标函数为

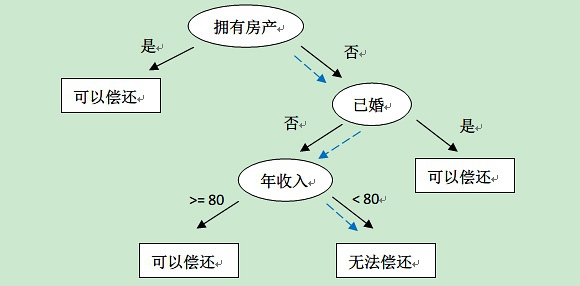
可以在目标函数中加入参数的一阶范数分或者二阶平方和，别称为L1正则和L2正则。其形式分别为

其中L1正则更倾向于使解稀疏化。在逻辑回归的损失函数中同样可以加入L1或L2正则来避免过拟合，而可以通过控制系数来控制对拟合数据惩罚的大小。

对于多分类问题，可以看成多个二分类问题，利用SoftMax回归进行解决，我们研究场景中对应的是一个二分类问题，在此不再赘述。

3.3.2决策树与随机森林

决策树(Decision Tree)是功能强大且广受欢迎的分类和预测方法，它是一种有监督的学习算法，以树状图为基础，其输出结果为一系列简单实用的规则，故得名决策树。决策树是一个预测模型；他代表的是对象属性与对象值之间的一种映射关系。树中每个节点表示某个对象，而每个分叉路径则代表的某个可能的属性值，而每个叶结点则对应从根节点到该叶节点所经历的路径所表示的对象的值。决策树仅有单一输出，若欲有复数输出，可以建立独立的决策树以处理不同输出。决策树的示意图如下图：



决策树的学习主要分为以下三个阶段：

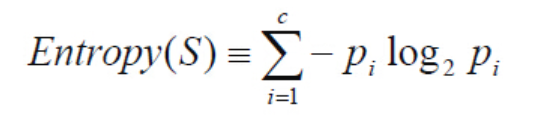
（1）特征选择：从训练数据的特征中选择一个特征作为当前节点的分裂标准（特征选择的标准不同产生了不同的特征决策树算法）。

（2）决策树生成：根据所选特征评估标准，从上至下递归地生成子节点，直到数据集不可分则停止决策树停止声场。

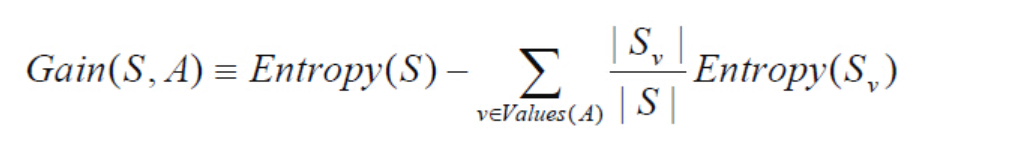
（3）剪枝：决策树容易过拟合，需要剪枝来缩小树的结构和规模（包括预剪枝和后剪枝）。

特征选择就是选取有较强分类能力的特征。分类能力通过信息增益或者信息增益比来刻画。特征选择的标准是找出局部最优的特征作为判断进行切分，取决于切分后节点数据集合中类别的有序程度(纯度)，划分后的分区数据越纯，切分规则越合适。衡量节点数据集合的纯度有：熵、基尼系数和方差。属性选择度量算法有很多，一般使用自顶向下递归分治法，并采用不回溯的贪心策略。不纯度的选取有多种方法，每种方法也就形成了不同的决策树方法，比如ID3算法使用信息增益作为不纯度；C4.5算法使用信息增益率作为不纯度；CART算法使用基尼系数作为不纯度。

ID3算法是由Ross Quinlan提出的决策树的一种算法实现，以信息论为基础，以信息熵和信息增益为衡量标准，从而实现对数据的归纳分类。ID3算法的核心思想是以信息增益度量属性选择，选择分裂后信息增益最大的属性进行分裂。信息熵（entropy）是用来衡量一个随机变量出现的期望值。如果信息的不确定性越大，熵的值也就越大，出现的各种情况也就越多。其公式是：

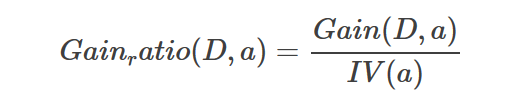


其中，S为所有事件集合，p为发生概率，c为特征总数。注意：熵是以2进制位的个数来度量编码长度的，因此熵的最大值是log2C。信息增益（information gain）是指信息划分前后的熵的变化，也就是说由于使用这个属性分割样例而导致的期望熵降低。也就是说，信息增益就是原有信息熵与属性划分后信息熵（需要对划分后的信息熵取期望值）的差值，具体计算法如下：

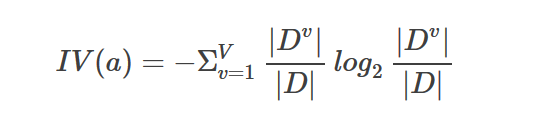


其中，第二项为属性A对S划分的期望信息。

ID3算法存在一个问题，就是偏向于多值属性，例如，如果存在唯一标识属性ID，则ID3会选择它作为分裂属性，这样虽然使得划分充分纯净，但这种划分对分类几乎毫无用处。ID3的后继算法C4.5使用增益率（gain ratio）的信息增益扩充，试图克服这个偏倚。C4.5决策树算法不直接使用信息增益，而是使用信息增益率来选择最优划分属性。



其中



*IV*(*a*)称为属性a的固有值，属性a的可能取值数目越多，则*IV*(*a*)的值通常会越大。增益率准则对取值数目较少的属性有所偏好。

还有另一种算法，CART分类树(Classification and Regression Tree)使用基尼指数来选择属性的划分，通过基尼值来度量数据集的纯度。假设有K个分类，样本点属于第k类的概率为pk=P（）#华书P39 公式

对每个特征A以及它可能的每个值a，计算Gini(D,A)，在所有的特征A以及所有的切分点a中，基尼指数最小的A和a就是最优特征和最优切分点。根据最优特征和最优切分点将训练集D切分成两个子节点。对两个子节点递归调用上面的步骤，直到满足停止条件为止。通常所用的停止条件为下列三个之一：一是节点中样本数小于预定值，二是样本集的基尼指数小于预定值，三是没有更多的特征可用。

决策树生成算法递归地产生决策树，知道不能继续下去为止。这样产生的树往往对训练数据分类十分准确，但泛化能力较差，即很容易出现过拟合的现象。过拟合的原因在于学习时过多地考虑如何提高对训练数据集的正确分类，从而构建出过于复杂的决策树。解决这个问题的办法是考虑决策树的复杂度，对已经生成的决策树进行简化。

以CART剪枝为例，CART剪枝算法由两步组成：首先从生成算法产生的决策树T0底端开始不断剪枝，直到T0的根结点，形成一个子树序列{T0，T1，…,TN}，然后通过交叉验证法在独立的验证数据集上对子树序列进行测试，从中选择最优子树。具体地，在剪枝过程中，计算子树的损失函数：

CaT=C(T)+a|T|,

其中，C（T）为对训练数据的预测误差，|T|为子树的叶结点个数，a》=0为参数。从整体树T0开始剪枝，对T0的任意内部节点t。以t为单结点树的损失函数是：

Ca(Tt)=C(Tt)+a|Tt|

当a为某一个值时，有a=C（t）-CTt（）/T-1，此时Tt与t有相同的损失函数值，而t的结点更少，因此t比Tt更可取，对Tt进行剪枝，对不同的结点进行此操作，在剪枝后会得到子树序列T0，T1，。。。，Tn，对于这个子树序列在独立的验证数据集进行交叉验证，选择最优的子树Ta，剪枝的操作完成。

尽管有剪枝等等方法，一棵树的生成肯定还是不如多棵树，因此就有了随机森林，解决决策树泛化能力弱的缺点。而同一批数据，用同样的算法只能产生一棵树，这时Bagging策略可以帮助我们产生不同的数据集，即从样本集（假设样本集N个数据点）中重采样选出Nb个样本（有放回的采样，样本数据点个数仍然不变为N），在所有样本上，对这n个样本建立分类器（ID3\C4.5\CART\），重复以上两步m次，获得m个分类器，最后根据这m个分类器的投票结果，决定数据属于哪一类。

随机森林在bagging的基础上更进一步：

1.样本的随机：从样本集中用Bootstrap随机选取n个样本

2.特征的随机：从所有属性中随机选取K个属性(通常情况下，推荐值k=log2 d)，选择最佳分割属性作为节点建立决策树。

3.重复以上两步m次，即建立了m棵决策树。

4.这m个CART形成随机森林，通过投票表决结果，决定数据属于的类别。

随机森林简单、容易实现、计算开销小，在很多任务中展现出强大的性能，随机森林的基学习器的多样性来自于样本扰动与自属性扰动，从而可以使最终集成的泛化性能可以通过个体学习器之间的差异度的增加而进一步的提升。

随机森林的收敛性与Bagging相似。随机森林的起始性能往往相对较差，特别是在集成中只包含一个基学习器，然而随着个体学习器数目的增加，随机森林通常会收敛到更低的泛化误差。

3.3.3 梯度提升决策树

集成学习的方法除了以随机森林为代表的Bagging方法，另一种则是Boosting方法，在Boosting算法中，个体学习器之间存在着强依赖关系，必须穿行生成，而梯度提升决策树GBDT (Gradient Boosting Decison Tree)则是Boosting算法的一种典型代表。

提升方法实际采用加法模型(即基函数的线性组合)与前向分布算法，以决策树为基函数的提升方法称为提升树(boosting tree)。对分类问题决策树是二叉分类树，对回归问题决策树是二叉回归树。提升模型可以表示为决策树的加法模型：

P147李航

首先确定初始提升树f=0，第m步的模型是:

8.24

其中，fm-1X为当前模型，通过经验风险极小化确定下一棵决策树的参数0M,

8.25

由于树的线性组合可以很好地拟合训练数据，即使数据中输入与输出之间的关系很复杂也是如此。不同的问题提升树学习算法，主要区别在于使用的损失函数不同，比如回归问题采用的是平方误差损失函数。已知一个数据集T，若回归树将输入空间划分为J个互不相交的区域，并在每个区域上确定输出的常量Cj，则数可以表示为

李航 P147 8.26

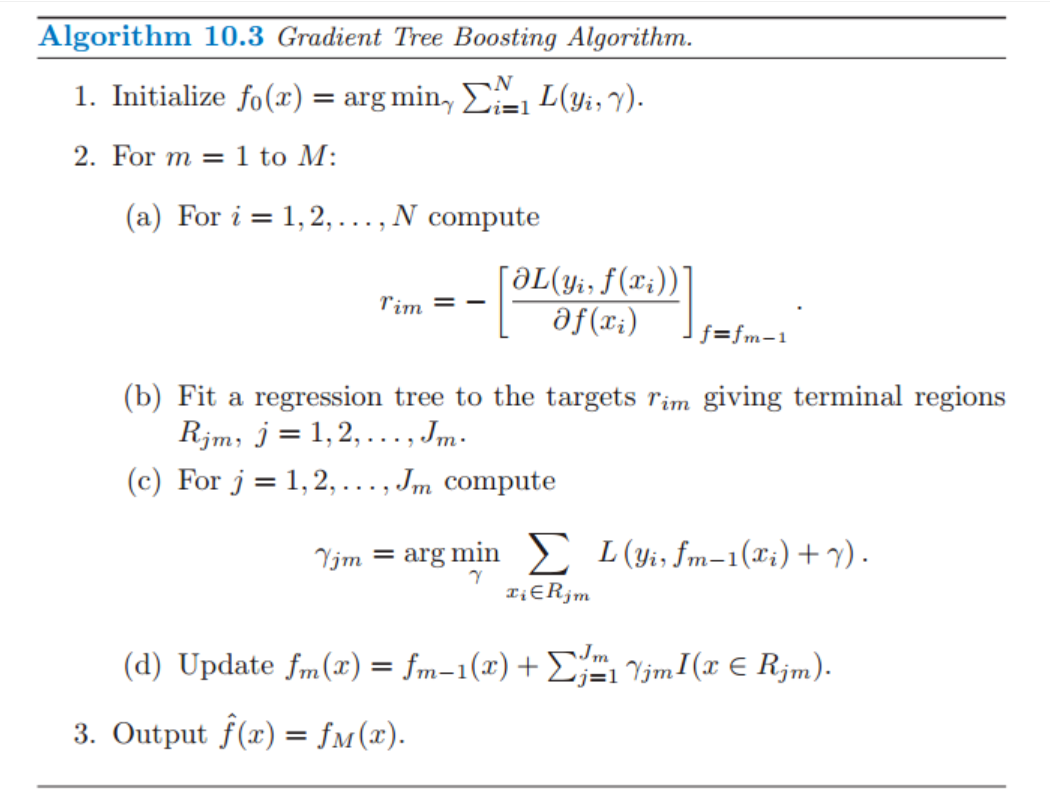
当我们进行提升时，使用以下前向分步算法

李航P148 8.27

这里 8.27是当前模型拟合数据的残差，所以，对回归问题的提升树算法来说，只需简单地拟合当前模型的残差就可以了。

当损失函数是平方损失和指数损失函数时，每一步的优化是很简单的，但是对于一般的损失函数而言，往往每一步的优化不是特别简单，针对这一问题产生了梯度提升算法，利用了最速下降法的近似方法，其关键是利用损失函数的负梯度在当前模型的值：

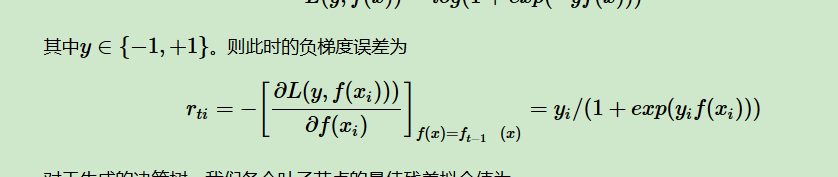
李航P151

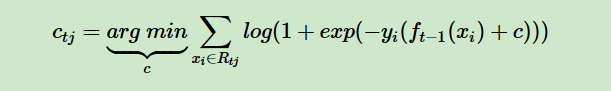
作为残差的近似值，拟合一个回归树。GBDT整个算法如下：

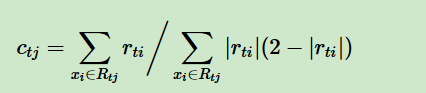
GBDT的分类算法在思想上和GBDT的回归算法没有区别，但是由于样本输出不是连续的值，而是离散的类别，导致我们无法直接从输出类别去拟合类别输出的误差。为了解决这个问题，主要有两个方法，一个是用指数损失函数，此时GBDT退化为Adaboost算法。另一种方法是用类似于逻辑回归的对数似然损失函数的方法。也就是说，我们用的是类别的预测概率值和真实概率值的差来拟合损失。

对应于我们要解决的二分类问题，如果用类似于逻辑回归的对数似然损失函数，则损失函数为：





对于生成的决策树，我们各个叶子节点的最佳残差拟合值为　由于上式比较难优化，我们一般使用近似值代替：



　除了负梯度计算和叶子节点的最佳残差拟合的线性搜索，二元GBDT分类和GBDT回归算法过程相同。

GBDT有很多优点，可以灵活处理各种类型的数据，包括连续值和离散值，此外可以在相对较少的调参时间下获得较好的准确率，所以GBDT被认为是统计学习中最有效的方法之一。

3.3.4 人工神经网络

受生物学的启发，人工神经网络是由一系列简单的单元互相紧密联系构成的，每个单元有一定数量的实数输入和唯一实数输出。神经网络的一个重要用途就是接受和处理传感器产生的复杂输入并进行自适应性的学习。人工神经网络算法模拟生物神经网络，是一种模式匹配算法，通常用于解决分类和回归问题。

神经网络中最基本的成分是神经元，又叫感知器，一个神经元的基本结构如图3-1所示。

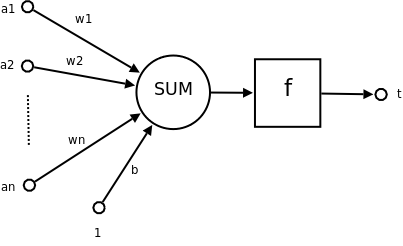


图3-1中的圆圈就表示一个神经元，为输入向量的各个分量，为各个输入向量对应的权值，b为偏置项，f为激活函数，t为神经元的输出，各个变量间的数学关系为：

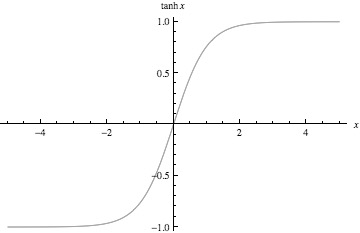
通过式3-1可见，输出变量t为各输入分量与权重矢量之间的内积，之后经过激活函数的作用。在神经网络中，激活函数的作用是能够给神经网络加入一些非线性的变换，使得神经能够处理更为一些复杂的非线性问题。激活函数主要有以下几个特点：

(1)非线性。激活函数为线性函数时，少量层数的神经网络就可以逼近大部分的函数。

(2) 可微性。因最优化算法需要求导数，因此激活函数需要可微。

(3) 单调性。当激活函数是单调的时候，单层网络能够保证是凸函数。

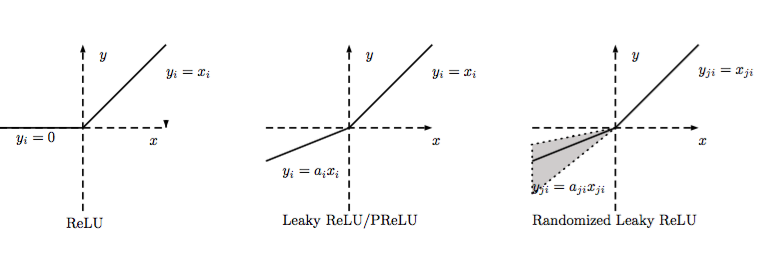
常用的激活有tanh、relu等。tanh的公式为：

其图像是：

从图上可以看出，tanh为0均值，可以产生负梯度。

relu激活函数的公式为：

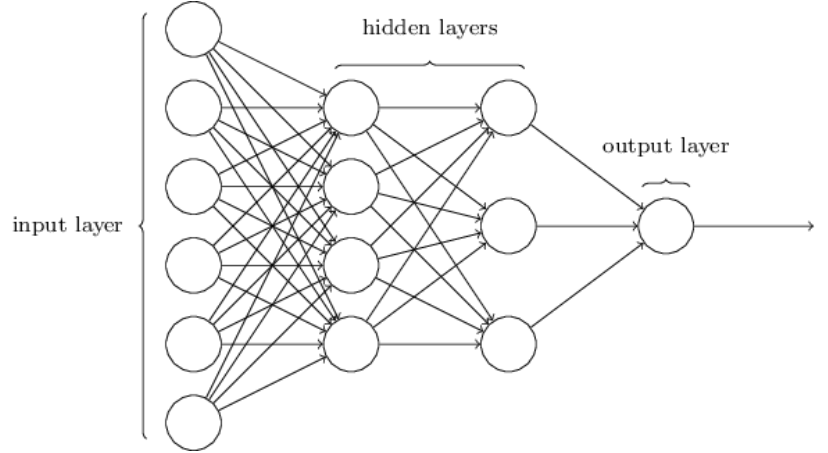
图像为下图中左边所示：



relu及其变化示意图

relu在优化的过程中能够很快地收敛，且relu函数缓解了梯度消失的问题。

以神经元为基本单元就可以构成一个神经网络。一个神经网络由三个基本部分组成：输入层、隐藏层、输出层。神经网络每一层由多个神经元组成，前层神经元的输出作为后层神经元的输入，由图所示



输入层由很多个神经元组成，接受大量的信息输入，比如每个样本的各个维度的信息。隐藏层的层数可以根据任务的复杂度来确定，当任务较为复杂时，可以设置多层隐藏层，使得神经网络具有很强的非线性学习能力，从而能拟合很复杂的函数。输出层通常与任务要求的输出相一致，例如在个人信用风险评估场景中，是一个二分类任务，类别数即为输出层的神经元的个数。

神经网络虽然具有很强的非线性拟合能力，但很多的神经元产生了很多的权重参数，因此需要大量的训练样本来更新权重参数，而训练的过程则需要用到反向传播算法。反向传导算法的核心是对神经网络中损失函数关于权重和偏置项的偏微分表达，它不仅是一个快速学习的算法，而且可以给出权值和偏置项的改变是如何作用于整个网络的。假设在一个神经网络中，第层的第k个神经元到第 层中第j个神经元的权值为，第 层中第j个神经元的偏执项为，第l层第j个神经元的激活值为，第 层第j个神经元的加权输入和为，则根据公式3-1可以得到

|  |  |  |
| --- | --- | --- |
|  |  | (3-6) |

假设损失函数C可写为单个训练样本的损失函数和的平均值，即

|  |  |  |
| --- | --- | --- |
|  |  | (3-7) |

其中，损失函数C是神经网络输出值的方程。

根据以上条件可得到以下反向传导算法的计算步骤：

（1）输入x：设置输入层的激活值为；

（2）前馈计算：对于计算和；

（3）计算输出残差：

|  |  |  |
| --- | --- | --- |
|  |  | (3-8) |

（4）反向传导计算各层神经元的残差：

|  |  |  |
| --- | --- | --- |
|  |  | (3-9) |

其中。

（5）计算损失函数C关于w和b的梯度：

|  |  |  |
| --- | --- | --- |
|  |  | (3-10) |

3.4 模型结果评估及分析

第四章 基于深度模型的建模研究

4.1 gcForest

4.2改进的gcForest算法

4.2.1 多样性理论

4.2.2 XGBoost

4.3 Wide&DeepLearning

4.4模型结果评估及分析

第五章 模型融合技术在建模中的应用研究

5.1 Stacking理论介绍

5.2 模型stacking实现

5.3 模型结果评估及分析

第六章 总结与展望

参考文献

[1]Durand D. Risk Elements in Consumer Instalment Financing. National Bureau of Economic Research,

1941.

[2]Myers J.H. & Forgy E.W. The development of numerical credit evaluation systems. Journal of the American Statistical Association, 1963, 58(303): 799-806.

[3] Li, H., Hand, D. Direct versus indirect credit scoring classifications[J]. Journal of the Operational Research Society., 2002. 53 (6), 647–654.

[4] Hens, A., Tiwari, M. Computational time reduction for credit scoring: An integrated approach based on support vector machine and stratified sampling method[J]. Expert Systems with Applications,2012. 39 (8), 6774–6781.

[5] Akkoc, S. An empirical comparison of conventional techniques, neural networks and the three stage hybrid adaptive neuro fuzzy inference system (anfis) model for credit scoring analysis: The case of turkish credit card data[J]. European Journal of Operational Research., 2012. 222 (1), 168–178.

[6] Giudici, P. Bayesian data mining, with application to benchmarking and credit scoring[J].

Applied Stochastic Models in Business and Industry ., 2001. 17 (1), 69–81.

[7] Breiman, L., Friedman, J. H., Olshen, R. A., Stone, C. J. Classification and regression trees[M].wadsworth & brooks. Monterey, CA. 1984.

[8] Kao, L.J., Chiu, C.C., Chiu, F.Y. A bayesian latent variable model with classification and

regression tree approach for behavior and credit scoring[J]. Knowledge-Based Systems.,2012.36,245–252.

[9] Ripley, B. D. Pattern Recognition and Neural Networks[M]. Cambridge University Press. 1996.

[10] Pang, S.-L.Study on credit scoring model and forecasting based on probabilistic neural

network[J]. Xitong Gongcheng Lilun yu Shijian/System Engineering Theory and Practice., 2005. 25(5), 43–48.

[11] Lisboa, P., Etchells, T., Jarman, I., Arsene, C., Aung, M., Eleuteri, A., Taktak, A., Am- brogi, F.,Boracchi, P., Biganzoli, E.Partial logistic artificial neural network for competing risks regularized with automatic relevance determination[J]. IEEE Transactions on Neural Networks., 2009. 20 (9),1403–1416.

[12] Marcano-Cedeno, A., Marin-De-La-Barcena, A., Jimenez-Trillo, J., Pinuela, J., Andina, D.

Artificial metaplasticity neural network applied to credit scoring[J]. International Journal of Neural Systems ., 201121 (4), 311–317.

[13] Chuang, C.-L., Huang, S.-T.A hybrid neural network approach for credit scoring. Expert

Systems[J] ., 2011. 28 (2), 185–196.

[14] 陈建，智能性交易欺诈风险评分模型的开发与应用[J]，中国信用卡，2006(6)，3-16.

[15] 严华，胡孟英，蔡瑞英，防止信用卡欺诈的系统设计[J]，微计算机信息，2006(12)，63-65.

[16] Ling, Y., Cao, Q., Zhang, H. Credit scoring using multi-kernel support vector ma- chine and chaos particle swarm optimization[J]. International Journal of Computational Intelligence and Applications., 2012.11 (3), 12500198:1–12500198:13.

[17] 伍保华，基于神经网络的信用卡反欺诈系统研究[D]，武汉，武汉理工大学

计算机科学与技术，2010，7-18.