决策树预测负债数据集

眭永熙[1)]

[1)](南京大学 计算机科学与技术系, 南京市中国 210023)

摘 要 本文旨在探索如何使用决策树算法预测负债数据集，以帮助金融公司评估借款人的信用能力和做出明智的借贷决策。本文将进行全面的数据分析和预处理，以提高预测模型的性能。通过利用机器学习技术，我们的目标是自动化评估过程，并根据申请人提供的详细信息提供可靠的工具来预测贷款结果。本文使用的模型在测试集上准确率达到了 0.73，f1-score 达到了 0.82。

关键词 负债数据集；决策树算法；机器学习；数据分析；分类模型

中图法分类号 TP391 DOI 号 10.11897/SP.J.1016.01.2023.00001

Using Decision Tree To Predict Loan Dataset

SUI Yong-Xi[1)]

[1)](Department of Computer Science, Nanjing University, Nanjing 210023, China)

Abstract This paper aims to explore how to use decision tree algorithm to predict the Loan Data dataset, in order to help financial companies evaluate borrowers' creditworthiness and make wise lending decisions. This paper will conduct comprehensive data analysis and preprocessing to improve the performance of the prediction model. By utilizing machine learning techniques, our goal is to automate the evaluation process and provide reliable tools for predicting loan outcomes based on detailed information provided by applicants. Our model has achieved an accuracy of 0.73 and a f1-score of 0.82 in testing dataset.

Key words Loan Data dataset; Decision Tree; Machine Learning; Data Analysis; Classification Model

1 引言

金融行业在很大程度上依赖贷款服务来满足个人的多样化财务需求。在申请贷款时，用户需要提交贷款申请，然后金融公司根据用户的申请评估其贷款资格。自动化贷款资格审核流程已成为金融机构的重要目标，它可以提高运营效率并提升效益。

本文将探索负债数据集（Loan Data dataset），数据集如表 1。该数据集提供了关于贷款申请过程的有价值的信息，包括借款人的人口统计信息、贷款细节和贷款状态。本文的目标是使用决策树算法实现一个分类模型，用于确定贷款是否能够还清。

了解贷款偿还状况对金融公司评估借款人的信用能力和做出明智的借贷决策至关重要。通过利用机器学习技术，我们的目标是自动化这个评估过程，并根据申请人提供的详细信息提供可靠的工具来预测贷款结果。

收稿日期：2023-05-22；修改日期：2023-05-22 眭永熙，学号：522022330058. E-mail: 522022330058@smail.nju.edu.cn.
第 1 作者手机号码：……，E-mail: 522022330058@smail.nju.edu.cn

为了实现这一目标，我们将进行全面的数据分析，从探索和理解数据集的结构和特征开始。我们将进行数据预处理，处理缺失值、异常值和其他数据不一致性。我们将采用特征工程技术来创建新的有意义的特征，这些特征有可能提高我们分类模型的性能。

接下来，我们将使用处理后的数据集训练决策树模型。决策树是直观且可解释的模型，可以处理数值和分类特征。它们提供了决策过程的透明表示，更容易理解影响贷款偿还结果的因素。

为了评估模型的性能，我们将数据集分为训练集和测试集。我们将根据准确率、精确率、召回率和 F1 分数等各种性能指标评估训练后的决策树模型。选择表现最佳的模型作为我们最终的分类模型。

表 1 Loan Data 数据集

Unnamed: 0.1	Unnamed: 0	loan_status	Principal	terms	effective_date	due_date	age	education	Gender
0	0	PAIDOFF	1000	30	9/8/2016	10/7/2016	45	High School or Below	male
2	2	PAIDOFF	1000	30	9/8/2016	10/7/2016	33	Bechalor	female
3	3	PAIDOFF	1000	15	9/8/2016	9/22/2016	27	college	male
4	4	PAIDOFF	1000	30	9/9/2016	10/8/2016	28	college	female
6	6	PAIDOFF	1000	30	9/9/2016	10/8/2016	29	college	male

2 数据分析

2.1 数据集特征

负债数据集（Loan Data）一共有 346 条数据，每条数据包括以下几个特征：

1. 贷款状态（Loan_status）：该特征描述了贷款的当前状态，可能的取值包括已偿还（Paid Off）、催收中（In Collection）。这是我们要分类的目标标签。

2. 基本贷款金额（Principal）：这个特征表示贷款的基本本金金额，即最初的贷款金额。

3. 还款周期（Terms）：该特征描述了贷款的还款周期，可以是每周（7 天）、每两周或每月。

4. 贷款起效日期（Effective_date）：该特征表示贷款生效的日期和时间，即贷款开始生效的时间点。

5. 到期日期（Due_date）：由于是一次性还款安排，每笔贷款都有一个单一的到期日期，该特征描述了贷款的到期日期。

6. 年龄（Age）、教育程度（Education）和性别（Gender）：这些特征提供了借款人的基本人口统计信息。年龄表示借款人的年龄，教育程度表示借款人的受教育程度，而性别表示借款人的性别。

数据集的标签分布如图 1所示，其中已偿还（Paid Off）标签有 260 个，占比约为 75%；而催收中（In Collection）标签有 86 个，占比约为 25%。

2.2 连续值特征分析

负债数据集中有两项连续值特征，分别是基本贷款金额（Principal）和年龄（Age）。我们根据不同

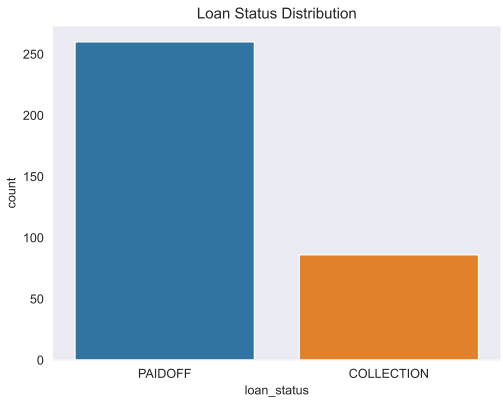

图 1 数据集标签分布

标签下这两个特征的数据分布，绘制对应的箱图，并分析这两个特征与标签的相关性。

如图 2.(a) 所示，在已偿还标签下，基本贷款金额分布在 1000 到 800 之间；而在催收中标签下，基本贷款金额集中在 1000。这个数据分布说明贷款金额越高，越有可能没有偿还，呈现出一定的负相关性。

如图 2.(b) 所示，处于已偿还标签的客户平均年龄相比处于催收中的客户平均年龄略高，但两者没有特别明显的差距，因此该特征与标签有微弱的正相关性。

2.3 离散值特征分析

负债数据集有三项离散值特征，分别是还款周期（Terms）、教育程度（Education）和性别（Gender）。我们根据不同取值下，两种标签所占的百分比，绘制对应的柱状图，并分析三个特征与标签的相关性。如图 3.(a) 所示，在四种教育程度中，本科、专业学校、高中或以下的标签分布和总体的标签分布一致，既已偿还为 75%、催收中为 25%；而在硕士以上的教育程度，已偿还和催收中各为 50%。这说明硕士以上学历面临更大的负载风险，这可能和他们支出较多、需要的借款金额较大，但是其偿还能力

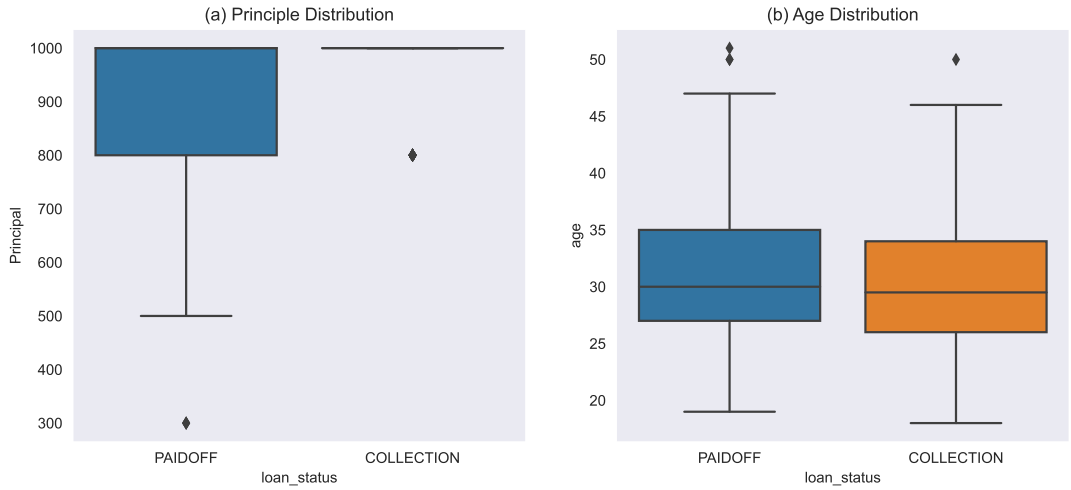

图 2　基本贷款金额（Principal）和年龄（Age）在不同贷款状态（Loan_status）下的分布

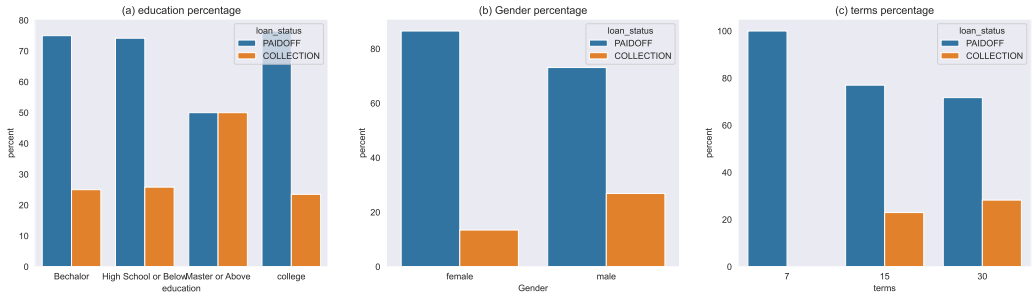

图 3　贷款状态（Loan_status）在还款周期（Terms）、教育程度（Education）和性别（Gender）中所占的百分比

受限于工资水平，因此更容易处于催收中的状态。

如图 3.(b) 所示，女性的已偿还占比为 80% 以上，而男性的已偿还占比为 70% 左右，女性的偿还率明显高于男性，这也许和两种性别所承担的经济压力有关。

如图 3.(c) 所示，还款周期为 7 天的短期贷款的偿还率为 100%，这种特殊的分布使我们注意到决策树模型（Decision Tree）对于该数据集的天然优势：当数据项的还款周期为 7 天时，有接近 100% 的概率为已偿还。此外，还款周期越高，已偿还的占比就越低，这体现出了一定的负相关性。

2.4　选择模型

根据上两节的分析，我们不难看出该数据在不同的特征值取值下有明显的数据倾斜。特别是对于还款周期（Terms），当还款周期为 7 天时，贷款状态有接近 100% 的概率是已偿还（Paid Off）。针对这种数据特征，有两种模型比较适合，分别是决策树模型（Decision Tree）和朴素贝叶斯网络（Bayesian Network）。有以下几点原因：

1. 决策树模型的适应性：决策树模型在处理不平衡数据集方面表现良好。它可以通过特征选择和分裂来捕捉数据集中的关键特征和模式，并根据这些特征进行分类。决策树模型可以根据数据的不平衡性自动调整节点分裂的条件，从而有效处理数据倾斜问题[1]。

2. 朴素贝叶斯网络的假设独立性：朴素贝叶斯网络假设特征之间是条件独立的。在数据倾斜的情况下，朴素贝叶斯网络可以通过学习各个特征的条件概率分布来做出分类。即使某些特征在给定类别下具有极高的概率，朴素贝叶斯网络仍然可以从其他特征中获得一些信息来进行分类[2]。

3. 对于还款周期为 7 天时贷款状态的数据倾斜情况，如果使用其他模型（如逻辑回归或支持向量机），可能会受到倾斜数据的影响，导致模型对其他还款周期下的贷款状态分类不准确。而决策树模型和朴素贝叶斯网络能够灵活地对

待不平衡数据，并通过特征选择、分裂和概率计算等方式准确分类不同还款周期下的贷款状态。

在本文中，我们选择决策树模型来分类贷款状态，这是考虑到当还款周期为 7 天时，可以直接分类为已偿还，这天然适合使用决策树模型。

3 数据预处理

3.1 编码

为了使用决策树模型，我们需要将数据集中的字符串类型的特征值编码为整数。对于性别（Gender），我们使用二进制编码：将男性（male）编码为 0，将女性（female）编码为 1。对于教育程度（education），我们使用顺序编码：将本科（Bechalor）编码为 0，将高中或以下（High School or Below）编码为 1，将硕士以上（Master or Above）编码为 2，将专业学校（college）编码为 3。

此外，还需要对数据集的标签进行编码，我们将催收中（COLLECTION）标签编码为 0，将已偿还（PAIDOFF）标签编码为 1。

3.2 特征工程

数据集中的贷款起效日期（Effective_date）和到期日期（Due_date）是一个日期值，我们把日期值按照年、月、日拆成三项。由于所有的贷款都发生在 2016 年，因此我们删去了年份列。

3.3 相关性分析

将所有的特征值都转换为数字之后，我们就可以对数据集不同特征之间做相关性分析。根据 Pearson 相关系数，计算得出相关性矩阵，如图 4所示。可以看出，贷款起效月份相关性为 0，这是因为贷款起效月份只有一个值，因此可以去掉这一列。

3.4 正则化

如表 2所示，是预处理之后的数据，其中每一个特征值都是数字，我们对每一列进行正则化处理。

3.5 数据集划分

本文采用 8:2 的比例随机采样，将正则化之后的数据集划分为训练集和测试集，训练集用于训练决策树模型，测试集用于评估模型的效果。

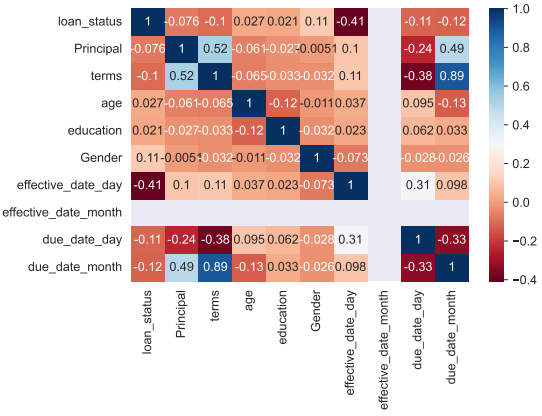

图 4 相关性矩阵热点图

4 决策树算法

决策树是一种基于树形结构的机器学习模型，常用于解决分类和回归问题。它通过对输入数据进行逐步的分割和判定，形成一个树状的决策流程，从而做出分类或预测。

决策树模型由三个主要部分组成：根节点、内部节点和叶节点。根节点代表整个数据集，内部节点表示特征的判定条件，而叶节点则代表最终的预测结果或分类标签。

决策树的构建过程基于特征选择和节点分裂的原则[3]：

特征选择：决策树根据特征选择准则来确定每个节点上最重要的特征。常用的特征选择方法包括信息增益（Information Gain）、基尼系数（Gini Index）和方差减少（Variance Reduction）等。

节点分裂：在决策树的构建过程中，每个节点都会根据选定的特征进行分裂。这个过程将数据集分成更具统一性的子集。分裂的准则可以是特征的阈值比较、离散特征的取值判定等。

递归构建：决策树是通过递归地构建子树来完成的。在每个节点上，选择最佳特征进行分裂，并继续在子节点上重复这个过程，直到满足终止条件，例如达到最大深度、节点包含的样本数小于某个阈值或节点的纯度达到一定程度。

剪枝：为了避免过拟合，决策树还可以进行剪枝操作。剪枝是通过修剪决策树的一些分支或叶节点来减少模型复杂度，提高泛化能力。

算法伪代码见algorithm 1。本文使用 python 库 scikit-learn 作为算法实现[4]。

我们用训练完成的决策树模型对测试集做分类，并使用 f1-score 和准确率作为评估指标，对模

表 2 预处理后的 Loan Data

loan_status	Principal	terms	age	education	Gender	effective_day	effective_month	due_day	due_month
1	1000	30	45	1.0	0	3	9	4	10
1	1000	30	33	0.0	1	3	9	4	10
1	1000	15	27	3.0	0	3	9	3	9
1	1000	30	28	3.0	1	4	9	5	10
1	1000	30	29	3.0	0	4	9	5	10

型做了评估。决策树模型的 f1-score 为 0.82，准确率为 0.73。

5 总结

本文对负债数据集（Loan Data）做了细致的数据分析，并根据分析结果说明了选择使用决策树模型的原因。接着我们针对决策树模型对数据做了预处理，并分析了不同特征和分类标签的相关性。最后我们在数据集上训练了决策树模型，并在测试集做了评估，模型的 f1-score 为 0.82，准确率为 0.73。

致 谢 致谢内容。

参考文献

[1] QUINLAN J R. Induction of decision trees[J]. Machine learning, 1986, 1:81-106.

[2] FRIEDMAN N, GEIGER D, GOLDSZMIDT M. Bayesian network classifiers[J]. Machine learning, 1997, 29:131-163.

[3] QUINLAN J R. Simplifying decision trees[J]. International journal of man-machine studies, 1987, 27(3):221-234.

[4] PEDREGOSA F, VAROQUAUX G, GRAMFORT A, et al. Scikit-learn: Machine learning in python[J]. the Journal of machine Learning research, 2011, 12:2825-2830.

附录 A

SUI Yong-Xi, Graduate Student in Nanjing University, Student ID:522022330058.

Algorithm 1: 决策树算法

```
Procedure BuildDecisionTree(数据, 特征, 目标)
    if 数据中的样本属于同一类别 then
        return 类别作为叶子节点
    if 特征为空或数据中的样本在特征上的取值相同 then
        return 数据中样本数最多的类别作为叶子节点
    最佳分裂特征 ← SelectBestSplitFeature(数据, 特征, 目标);
    创建以最佳分裂特征为分裂特征的根节点;
    根据最佳分裂特征的取值将数据分为子集 sub_data;
    for each 子集 sub_data do
        if sub_data 为空 then
            创建以数据中样本数最多的类别为类别的叶子节点;
            将叶子节点添加为根节点的子节点;
        else
            BuildDecisionTree(sub_data, 特征 - {最佳分裂特征}, 目标);
            将返回的子树添加为根节点的子节点;
    return 根节点;
Procedure SelectBestSplitFeature(数据, 特征, 目标)
    最佳分裂特征 ← None;
    最佳信息增益 ← −∞;
    for each 特征 in 特征 do
        计算特征上的信息增益增益;
        if 增益 > 最佳信息增益 then
            更新最佳信息增益为增益;
            更新最佳分裂特征为特征;
    return 最佳分裂特征;
Procedure PredictSample(样本, 树)
    if 树是叶子节点 then
        return 叶子节点的类别;
    获取树节点的分裂特征特征;
    找到与特征取值相对应的子树 sub_tree;
    PredictSample(样本, sub_tree);
```