机器学习实践实验报告

蒋雨初[1)]

[1)](东南大学人工智能学院, 南京市 210000)

摘 要 本研究针对机器学习专题实践课程的贷款数据集进行了分类任务。首先，对给定的贷款数据集进行了详细的数据分析和清洗工作。考虑到数据特性，选定了支持向量机（SVM）、逻辑回归、CatBoost 方法作为主要的算法进行测试。实验的目标是预测贷款是否会被偿还，这是一个典型的二分类问题。为了评估各个模型的效果，本研究采用了 macro F1-score 作为性能评估标准，并得到了最佳的算法设置。

关键词 机器学习，二分类

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

Machine learning practice report

Yuchu Jiang[1)]

[1)](Department of Artificial Intelligence, Southern East University, Nanjing 210000)

Abstract This study conducted a classification task on the loan data set of the machine learning special topic practical course. First, detailed data analysis and cleaning work was performed on the given loan data set. Considering the data characteristics, support vector machine (SVM), logistic regression, and CatBoost methods were selected as the main algorithms for testing. The goal of the experiment is to predict whether the loan will be repaid, which is a typical two-classification problem. In order to evaluate the effect of each model, this study used macro F1-score as the performance evaluation criteria, and obtained the optimal algorithm settings.

Key words Machine Learning; Binary Classification

1 实验设置

1.1 实验任务

大多数金融公司都提供贷款服务。当用户向金融公司申请贷款时，用户首先需要提交贷款申请，然后金融公司验证用户的贷款资格，并根据用户的申请决定是否向用户提供贷款服务。金融公司希望根据用户填写在线申请表时提供的详细信息来自动化贷款资格审查过程。这是一个二分类问题，需要训练一个模型来预测贷款是否会被偿还。

- 问题理解与分析：对本实验问题的理解。
- 算法动机与背景介绍：介绍选择此算法的动机，并提供算法的背景知识。
- 算法的技术细节：包括算法的伪代码在内的全面技术细节。
- 性能描述与分析：对算法的性能结果进行描述和分析。
- 结论与讨论（可选）：总结研究并讨论未来的研究方向或潜在的改进措施。

1.2 实验平台

jupyter notebook

python=3.11

2 问题理解与数据分析

2.1 问题理解

这是一个二分类任务, 训练集有 346 个样本, 测试集有 54 个测试样本，这是一个较小数据集。同时有 7 个特征（principals, terms, effective date, due date, age, education, gender）和一个标签（loan_status）。

对于这个数据集, 可以考虑支持向量机（SVM）、逻辑回归、Boosting 方法，因为这样的传统机器学习方法往往可以在小的、特征不多的简单数据集上获得不错的效果。

收稿日期：2023-11-13；修改日期：2023-11-13 蒋雨初，男，2001 年生，本科，学生，主要研究领域为目标识别、零样本学习. E-mail: 865710157@qq.com.
第 1 作者手机号码：15687150255，E-mail: 865710157@qq.com

2.2 探索性数据分析

这个数据集是关于过去的贷款信息。train.csv 数据集包含了 346 名客户的详细信息，这些客户的贷款已经还清或者违约。在移除两个无关的列之后，它包括 7 个字段，其中有 3 个数值特征、2 个离散特征和 2 个日期特征，如图 1所示:

先把标题名称统一为首字母大写，再把日期特征做差合并为新的数值特征（Num of days）以便后续处理。最终的数据集（部分）如图 2所示:

2.3 数据可视化

在这一部分，我对 train.csv 中的数据进行了可视化分析，以探究数据各个特征的性质。

离散特征与标签

如图 3所示，观测到原始数据集中样本是极不平衡的，有显著的男性样本多于女性样本、PAIDEOFF 样本多于 COLLECTIONS 样本的特点。此外，受教育程度为 Master or Above 的样本极少。

数值特征

如图 4所示，观测到数据中存异常点，并且 Principal 的分布非常不均衡。

2.4 数据预处理

根据第 2.3 节的结果，可以观察到本实验中的数据集存在以下特点:

- 类别不平衡。对此，可以考虑使用过采样技术例如合成少数过采样技术（SMOTE)[1] 及其变体。也可以考虑使用类别加权法，为少样本类提供更大的分类错误惩罚。
- 存在离群值。对此，可以考虑根据四分位数剔除离群值。

因此，在预处理阶段，除了将离散特征转为数值特征和归一化之外，还要考虑使用 SMOTE 算法合成少数样本与使用四分位数剔除离群值。

离散特征的 One-hot Encoding

由于离散特征无法直接参与训练，因此需要离散特征转为数值特征。对于性别，可以直接把 male 变为 0，female 变为 1。对于受教育程度，使用 One-hot Encoding 进行编码。由于受教育程度为 Master or Above 的样本极少，因此在编码的过程中，可以直接剔除该值。

归一化

归一化是指将数值特征缩放到一个特定的范围（通常是 0 到 1 或-1 到 1），以便在不同尺度上的特征之间建立平衡。

不同特征可能有不同的尺度和单位，归一化确保这些特征在模型中以相同的尺度考虑，避免某些特征由于尺度较大而对模型的影响过大。特别地，某些机器学习算法（如支持向量机、K-近邻）在处理归一化数据时表现更好。

离群值剔除

使用四分位数来识别和移除离群值，即那些远离数据集中心趋势的值。

离群值可能会扭曲模型的训练，导致不准确的结果，尤其是线性回归模型，对离群值格外敏感。剔除它们可以提高模型的准确性和稳定性。

少样本合成

特地选用 SMOTE-NC（合成少数过采样技术-针对类别数据的扩展）用于解决分类数据集的不平衡问题，通过合成新的少数类样本来平衡类别分布。SMOTE-NC 是针对类别数据的特化[1]，能帮助模型更好地学习少数类别的特征，减少模型对多数类的偏见，提高对少数类的预测准确度。

3 相关工作

3.1 SMOTE-NC

SMOTE 是一种综合采样人工合成数据算法,用于解决数据类别不平衡问题 (Imbalanced class problem)，以 Over-sampling 少数类和 Under-sampling 多数类结合的方式来合成数据。算法流程如算法**??**所示。

SMOTE 负责接受要采样的类数据集 X,返回一个经过 SMOTE 采样后的数据集，大小为 $(N/100) * T$，函数有三个参数，分别是 T: 需要处理的数据集 X 的样本数量; N: 采样比例，一般为 100, 200, 300 等整百数，对应即 1 倍，2 倍，3 倍； K: 为采样的最近邻数量，论文中默认为 5。SMOTE 代码思想非常简单，扫描每一个样本点，计算每一个样本点的 K 个最近邻，将每一个最近邻样本点的索引记录在`nnarray`中，之后传入`Populate(N, i, nnarray)`中即完成一个样本点的采样。

`lstPopulate`则负责根据`nnarray`中的索引去随机生成 N 个与观测样本`i`相似的样本。该函数会计算随机邻近点`nn`与观测样本点`i`的每一个特征之间的差距`dif`，将其差距乘上一个 [0, 1] 随机因子`gap`，再将`dif*gap`的值加上观测点`i`即完成了一个特征的合成。

Field	Description
Loan_status	Whether a loan is paid off on in collection
Principal	Basic principal loan amount at the
Terms	Origination terms which can be weekly (7 days), biweekly, and monthly payoff schedule
Effective_date	When the loan got originated and took effects
Due_date	Since it's one-time payoff schedule, each loan has one single due date
Age	Age of applicant
Education	Education of applicant
Gender	The gender of applicant

图 1　数据集各个字段的解释。其中 Loan_status 是标签，Principal, Terms 和 Age 是数值属性，Effective_data 和 Due_data 是日期属性，Education 和 Gender 是离散属性。

	Loan_status	Principal	Terms	Age	Education	Gender	Num of days
0	PAIDOFF	1000	30	45	High School or Below	male	29
1	PAIDOFF	1000	30	33	Bechalor	female	29
2	PAIDOFF	1000	15	27	college	male	14
3	PAIDOFF	1000	30	28	college	female	29
4	PAIDOFF	1000	30	29	college	male	29

图 2　经过简单处理后的前 5 行数据集展示。

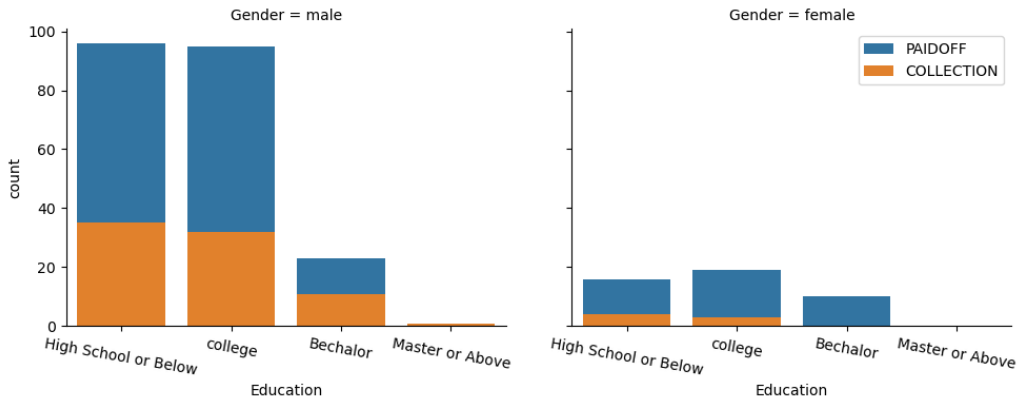

图 3　使用堆叠条形图可视化不同性别下受教育程度与贷款状态的关系。

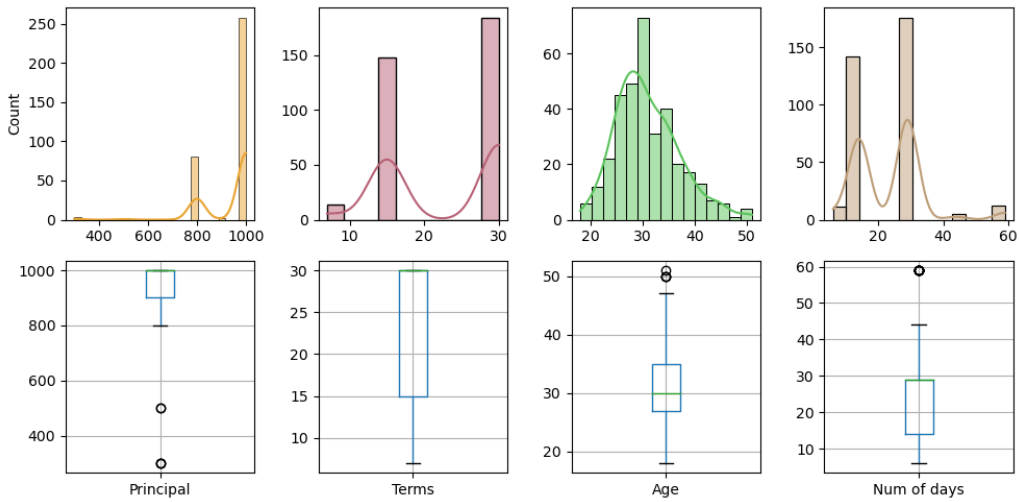

图 4　分别使用条形图和箱型图可视化 4 个数值特征。

SMOTE-NC 是针对离散特征的特化，具体来说有以下处理：

1. 中值计算：计算少数类的所有连续特征的标准差的中值。如果样本与其潜在最近邻之间的标称特征不同，则该中值将包含在欧几里德距离计算中。我们使用中位数来惩罚标称特征的差异，其惩罚量与连续特征值的典型差异相关。

2. 最近邻计算：使用连续特征空间计算正在识别的 k 近邻的特征向量（少数类样本）与其他特征向量（少数类样本）之间的欧氏距离。对于所考虑的特征向量与其潜在最近邻之间的每个不同的标称特征，在欧几里德距离计算中包括先前计算的标准差的中值。

3. 填充合成样本：新的合成少数类样本的连续特征是使用与前面描述的 SMOTE 相同的方法创建的。标称特征被赋予出现在大多数 k-最近邻中的值。

3.2　逻辑回归

逻辑回归是一种广泛使用的统计方法，用于预测一个依赖变量的二分类结果[2]。它是线性回归的一个特例，但用于分类问题而非回归问题。逻辑回归通过使用逻辑函数（通常是 sigmoid 函数）将线性回归的输出映射到 0 和 1 之间，用于估计概率。这使得它非常适合于二分类问题。其基本形式如下：

$$P(Y=1)=\frac{1}{1+e^{-(\beta_0+\beta_1 X)}}$$

其中 $P(Y=1)$ 是因变量为类别 1 的概率。X 是自变量。β_0 和 β_1 是模型参数。

逻辑回归适用较小的数据集，同时要求自变量与对数几率是线性关系，此外，还不能有严重的多重共线性。逻辑回归算法的伪代码如算法2所示。

3.3　多重共线性

多重共线性（Multicollinearity）是回归分析中的一个现象，指的是在多变量回归模型中，预测变量（自变量）之间存在高度相关关系。在理想情况下，一个回归分析模型中的预测变量应相互独立，但现实数据中往往存在一定程度的相关性。当多重共线性显著时，它会导致回归模型的系数估计不稳定，标准误巨大，从而影响模型的解释能力和预测准确性。

为了评估数据集是否存在多重共线性问题，通常可以展示相关系数热力图或使用方差膨胀因子来定量探究。

方差膨胀因子（Variance Inflation Factor，VIF）是一种量化多重共线性程度的指标[3]。VIF 计算的基本原理是对模型中的每一个独立变量进行回归分析，以该变量为因变量，其他所有变量为自变量。VIF 值是这种回归分析中得到的 R 平方值的一个函数，具体的计算公式是：

$$\mathrm{VIF}_i=\frac{1}{1-R_i^2}$$

其中，R_i^2 是变量 i 作为因变量进行回归分析时的确定系数（R-squared）。

VIF 的值范围从 1 以上，VIF 为 1 意味着该变

算法 1 SMOTE-NC [T, N, k]

输入： Number of minority class samples T; Amount of SMOTE N; Number of nearest neighbors k

输出： (N/100) * T synthetic minority class samples

IF $N < 100$ **THEN**
  Randomize the T minority class samples
  $T \leftarrow (N/100) \times T$
  $N \leftarrow 100$
END IF
$N \leftarrow \lfloor N/100 \rfloor$ *(The amount of SMOTE is assumed to be in integral multiples of 100.)*
k = Number of nearest neighbors
numattrs = Number of attributes
Sample[][]: array for original minority class samples
newindex $\leftarrow 0$ *(keeps a count of the number of synthetic samples generated)*
Synthetic[][]: array for synthetic samples
FOR $i = 1$ to T **DO**
  Compute k nearest neighbors for i, and save the indices in the *nnarray*
  Call Populate(N, i, *nnarray*)
END FOR
Populate$N$, i, *nnarray*
WHILE $N \neq 0$ **DO**
  Choose a random number between 1 and k, call it *nn* (select one of the k nearest neighbors of i)
  FOR *attr* = 1 to *numattrs* **DO**
    Compute: $dif \leftarrow Sample[nnarray[nn]][attr] - Sample[i][attr]$
    Compute: $gap \leftarrow$ random number between 0 and 1
    $Synthetic[newindex][attr] \leftarrow Sample[i][attr] + gap \times dif$
  END FOR
  $newindex \leftarrow newindex + 1$
  $N \leftarrow N - 1$
END WHILE

算法 2 Logistic Regression

输入： Training dataset $D = \{(\mathbf{x}_i, y_i)\}$, learning rate η, regularization parameter λ

输出： Logistic Regression Model parameters $\mathbf{w}$

Initialize the weight vector $\mathbf{w}$ with zeros or small random values
WHILE convergence criterion is not met **DO**
  FOR each $(\mathbf{x}_i, y_i)$ in D **DO**
    Compute prediction $\hat{y}_i = \sigma(\mathbf{w}^\top \mathbf{x}_i)$
    where $\sigma(z) = \frac{1}{1+e^{-z}}$ is the logistic function
    Update $\mathbf{w} \leftarrow \mathbf{w} - \eta \cdot (\hat{y}_i - y_i)\mathbf{x}_i - \eta\lambda\mathbf{w}$
  END FOR
END WHILE
RETURN $\mathbf{w}$

量与其他变量之间没有相关性，即不存在多重共线性。VIF 值越大，表明共线性问题越严重。通常使用的判断标准是：

- VIF 值小于 10，表明多重共线性是中等的，不会对回归模型造成太大问题。
- VIF 值在 10 到 100 之间，表明存在较高程度的多重共线性，需要进一步分析。
- VIF 值大于 100，表明存在极端的多重共线性，通常需要重新审视模型的变量，或者使用降维技术如主成分分析（PCA）来减少变量之间的相关性。

对于实验中所使用的数据集，进行预处理之后计算出的各个特征的 VIF 值如表 1所示，可注意到由 Education 的 one-hot 编码生成的特征共线性程度较高，然而由于本实验特征本来就少，因此也难以使用降维技术，所以这会导致逻辑回归算法性能较差。

3.4 支持向量机

支持向量机（SVM）是一种监督学习算法，主要用于分类任务[4]。其基本原理是在特征空间中寻找一个超平面，以此来区分不同的类别。这个超平面被选定为最大化两个类别数据点之间的边界（称为间隔）。SVM 通过引入核技巧，使得算法能够在高维空间中有效工作，即使数据在原始空间中不是线性可分的。

SVM 最适合中小规模数据集，尤其是在特征维度较高的情况下表现良好。其优势在于处理那些

Features	VIF
High School or Below	81.84
college	81.18
Bachelor	38.71
Num of days	4.42
Terms	4.31
Principal	1.43
Age	1.05
Gender	1.02

表 1 依次经过离群值移除、one-hot 编码、归一化之后数据集中各个特征的 VIF 值。

在原始空间中不是线性可分的数据集，通过使用不同的核函数（如线性核、多项式核、径向基函数核等），SVM 能够在更高维的空间中找到决策边界。然而，对于大规模数据集，SVM 的训练时间可能会比较长。SVM 的伪代码如算法3所示。

算法 3 Support Vector Machine Algorithm

输入： Training set $\{(x_1, y_1), ..., (x_n, y_n)\}$, where $x_i \in \mathbb{R}^d$ and $y_i \in \{-1, 1\}$

输出： Optimal hyperplane that separates the classes

Initialize weights w (vector in $\mathbb{R}^d$) and bias b (scalar)

η: Learning rate (a small positive number)

λ: Regularization parameter (controls trade-off between maximizing the margin and minimizing the classification error)

REPEAT

  FOR each (x_i, y_i) in training set **DO**

    IF $y_i(w \cdot x_i + b) < 1$ **THEN**

      // Misclassified or within the margin

      Update $w \leftarrow w + \eta(y_i x_i - 2\lambda w)$

      Update $b \leftarrow b + \eta y_i$

    ELSE

      // Correctly classified and outside the margin

      Update $w \leftarrow w - \eta 2\lambda w$

    END IF

  END FOR

UNTIL convergence or maximum iterations reached

3.5 CatBoost

CatBoost(Categorical Boosting)是一种基于梯度提升决策树(Gradient Boosting Decision Tree, GBDT)的机器学习算法[5]，由 Yandex 团队开发。它是为了解决各种类型的数据中的预测任务而设计的，特别是具有大量类别特征的数据集。CatBoost 的核心特点之一是它对类别特征的处理方式——它使用了一种特殊的统计编码技术来处理类别特征，无需人为地进行独热编码。这种处理方式允许 CatBoost 直接使用类别特征，进而提升模型在处理此类数据时的性能。与 XGBoost、LightGBM 相比，CatBoost 的创新点有：

- 嵌入了自动将类别型特征处理为数值型特征的创新算法。首先对 categorical features 做一些统计，计算某个类别特征（category）出现的频率，之后加上超参数，生成新的数值型特征（numerical features）。
- Catboost 还使用了组合类别特征，可以利用到特征之间的联系，这极大的丰富了特征维度。
- 采用排序提升的方法对抗训练集中的噪声点，从而避免梯度估计的偏差，进而解决预测偏移的问题。
- 采用了完全对称树作为基模型。

3.5.1 预测偏移

当在训练样本上训练的预测模型与测试样本的分布之间存在变化时，就会发生预测变化。这种情况经常发生，因为预测模型是使用所有训练样本进行训练的，但并非所有训练样本都完全代表测试样本。当使用所有训练样本训练预测模型时，可能会导致模型有偏差，因为模型在学习阶段已经看到了标签。在梯度提升中，当前估计 F^t 基于我们之前在每次迭代中构建的先前模型 F^{t-1}。为了更好地证明预测偏移，它本质上意味着 $F^{t-1}(x_k)|x_k$（其中 x_k 来自训练样本）从 $F^{t-1}(x)|x$（其中 x 来自测试样本）偏移。这是由目标泄漏引起的，其中估计模型的分布与测试样本的分布不同。因此，该模型存在偏差，在某些情况下可能导致过度拟合。为了解决这个问题，CatBoost 引入了有序提升算法。

在有序提升中，每一步提升都会获得一个新的训练数据集。这意味着模型经过训练，使得我们之前获得的模型应用于这组新的训练样本。这保证了我们之前获得的模型没有看到新训练集中的标签。因此，每一步提升的训练模型都没有偏差。为了更好地说明这一点，假设 M_t 是第 $t+1^{\text{st}}$ 次迭代中的当前模型，仅使用前 t 个样本学习，则残差

$r^{t+1} = y_i - M_t$ 不变，因为模型是在没有新训练集中的观测值的情况下进行训练的。训练样本的随机排列 σ 的生成支持了这一点。为了说明这个想法，请参见图 5。算法伪代码见4。不幸的是，由于需要训练 n 个不同的模型，该算法在大多数实际任务中并不可行，这使得复杂性和内存需求增加了 n 倍。因此，在 CatBoost 中对这个问题做了一些改进，见算法5和6的 Ordered 模式。

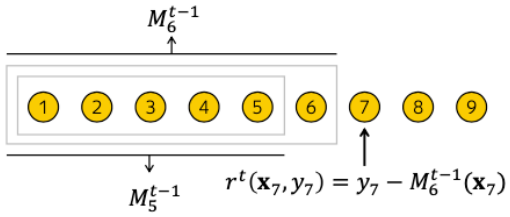

图 5 Ordered boosting principle, ordered by σ

算法 4 Orderd boosting

输入： $\{(x_k, y_k)\}_{k=1}^n$; I

$\sigma \leftarrow$ random permutation of $[1, n]$;
$M_0 \leftarrow 0$ for $i = 1$ to n;
FOR $t \leftarrow 1$ to I **DO**
  FOR $i \leftarrow 1$ to n **DO**
    $r_i \leftarrow y_i - M_{\sigma(i)-1}(x_i)$;
  END FOR
  FOR $i \leftarrow 1$ to n **DO**
    $\Delta M \leftarrow \text{LearnModel}\left((x_j, r_j) : \sigma(j) \leqslant i\right)$;
    $M_i \leftarrow M_i + \Delta M$;
  END FOR
END FOR
RETURN M_n

3.5.2 算法过程

CatBoost 算法通过初始化模型为零预测并在每次迭代中构建决策树来逐步优化模型。它可以在两种模式下运行：普通模式和有序模式。在有序模式下，它采用随机排列和分层的方法来更新模型预测，以此减少计算负担并避免过拟合。在每次迭代中，它计算当前模型的梯度，根据这些梯度更新树结构，并调整模型预测。最终，模型是所有树预测的加权和，其中权重由学习率和节点的梯度确定。这个过程旨在提升模型的准确性，同时处理类别特征和防止过拟合。算法流程见5。其中关键步骤是`BuildTree`，用于构建单个决策树，流程见6。

4 实验

4.1 评价指标

正如第 2.3 节中的讨论，数据集是类别不平衡，因此使用准确率、F1 分数作为主要评价指标并不合理，还应该考虑平衡准确分数（balanced accuracy score）[?] 或 AUC。

4.1.1 macro F1 分数

F1 分数（F1 Score）是分类任务中的一个性能评估指标，尤其是在数据集的类分布不平衡时，它提供了比单纯的准确率（Accuracy）更好的性能度量。F1 分数是精确率（Precision）和召回率（Recall）的调和平均值，它们的定义如下：

- 精确率（Precision）：在所有预测为正的样本中，实际为正的样本所占的比例。
- 召回率（Recall）：在所有实际为正的样本中，被正确预测为正的样本所占的比例。

F1 分数的计算公式是：

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

F1 分数的取值范围是 0 到 1，F1 分数越高，模型的性能越好。它试图在精确率和召回率之间达到平衡，这是因为单纯提高精确率（例如，只预测我们最有信心的几个样本）可能会降低召回率，反之亦然。因此，F1 分数对于那些需要在精确率和召回率之间权衡的场景非常有用。

Macro F1 分数是每一个类别的 F1 分数的平均值，对于类别不平衡的数据集特别有用，能够更好地评估模型在少数类上的性能。计算公式是

$$\text{macro F1} = \frac{\sum_i F1_i}{N}$$

4.1.2 平衡准确分数

平衡准确分数（Balanced Accuracy Score）是分类性能的一个评估指标，特别适用于处理类别不平衡问题。平衡准确分数是每个类别的召回率的平均值。这意味着它不会因为数据中有更多的一个类别而偏向于该类别，而是给所有类别同等的权重。当类分布不均匀时，平衡准确分数是一个比传统的准确率更有用的性能指标。

平衡准确分数的计算方法是：

算法 5 CatBoost

输入： $\{(x_i, y_i)\}_{i=1}^{n}$, I, α, L, s, Mode

$\sigma_r \leftarrow$ random permutation of $[1, n]$ for $r = 0..s$
$M_0(i) \leftarrow 0$ for $i = 1..n$
IF Mode = Plain **THEN**
  FOR $r \leftarrow 1$ to s **DO**
    $M_r(i) \leftarrow 0$ for $i : \sigma_r(i) \leqslant 2^j + 1$
  END FOR
END IF
IF Mode = Ordered **THEN**
  FOR $j \leftarrow 1$ to $\log_2 n$ **DO**
    FOR $r \leftarrow 1$ to s **DO**
      $M_{r,j}(i) \leftarrow 0$ for $i = 1..2^j + 1$
    END FOR
  END FOR
END IF
FOR $t \leftarrow 1$ to I **DO**
  $T_t, \{M_r^t\} \leftarrow \text{BuildTree}(\{M_r^{t-1}\}, \{(x_i, y_i)\}_{i=1}^{n}, \alpha, L, \{\sigma_i\}_{i=1}^{s}, \text{Mode})$
  $\text{leaf}_0(i) \leftarrow \text{GetLeaf}(x_i, T_t, \sigma_0)$ for $i = 1..n$
  $\text{grad}_0 \leftarrow \text{CalcGradient}(L, M_0, y)$
  FOR each leaf j in T_t **DO**
    $b_j^t \leftarrow \text{avg}(\text{grad}_0(i))$ for $i : \text{leaf}_0(i) = j$
    $M_0(i) \leftarrow M_0(i) + \alpha b_j^t$ for $i : \text{leaf}_0(i) = j$
  END FOR
END FOR
RETURN $F(x) = \sum_{t=1}^{I} \sum_j \alpha b_j^t \mathbb{1}_{\{\text{GetLeaf}(x, T_t, \text{ApplyMode}) = j\}}$

$$\text{Balanced Accuracy} = \frac{1}{N} \sum_{i=1}^{N} \frac{TP_i}{TP_i + FN_i}$$

其中，N 是类别的数量，TP_i 是对于第 i 类的真正例（true positives），FN_i 是对于第 i 类的假负例（false negatives）。

4.2 算法比较

本次实验中我考察了三种算法，分别是逻辑回归、支持向量机和 CatBoost，并分别测试了一般设置 (直接使用分类器在预处理后的数据上训练和预测)、使用 SMOTE-NC 合成数据和类别加权方法（即使用`class_weight = 'balanced'`得到的分类器）下的性能，结果如表 2所示。由于类别不平衡，可见在准确率上一般设置获得了最好效果，然而在 macro F1 分数和平衡准确分数上，使用了 SMOTE-NC 合成样本的支持向量机表现最优。此外，在类别平衡的设置下，使用合成了样本之后的训练集训练，性能大多都有所提升。

	准确率 @100	macro F1 分数 @100	平衡准确分数 @100
逻辑回归	**77**/58/51	43/56/49	50/59/59
支持向量机	**77**/68/52	43/**68**/51	50/**69**/60
CatBoost	**77**/66/60	43//66/57	50/68/65

表 2 实验结果。每一格数据的含义是一般设置/SMOTE-NC/类别加权平衡。

5 结论与讨论

本实验实现了三种算法，并以不同方式对数据的特征进行了特殊处理，完成了二分类任务。

具体而言，针对数据集中的日期特征做了额外处理，还移除了一些不重要的特征；针对数据集较小的性质选择了传统的机器学习算法；针对数据集

算法 6 Function BuildTree

输入： $M, \{(x_i, y_i)\}_{i=1}^{n}, \alpha, L, \{\sigma_i\}_{i=1}^{s}, \text{Mode}$

grad $\leftarrow$ CalcGradient(L, M, y);
$r \leftarrow$ random($1, s$);
IF Mode = Plain **THEN**
  $G \leftarrow (\text{grad}_i \text{ for } i = 1..n)$;
END IF
IF Mode = Ordered **THEN**
  $G \leftarrow (\text{grad}_i, \log_2(\sigma_i) - 1) \text{ for } i = 1..n)$;
END IF
$T \leftarrow$ empty tree;
FOR each step of top-down procedure **DO**
  FOR each candidate split c do **DO**
    $T_c \leftarrow$ add split to T;
    $\text{leaf}_i \leftarrow \text{GetLeaf}(x_i, T_c, \sigma) \text{ for } i = 1..n$;
    IF Mode = Plain **THEN**
      $\Delta_i \leftarrow \text{avg}(\text{grad}_p \text{ for } p : \text{leaf}_p = \text{leaf}_i) \text{ for } i = 1..n$;
    END IF
    IF Mode = Ordered **THEN**
      $\Delta_i \leftarrow \text{avg}(\text{grad}_p, \log_2(\sigma_p) - 1) \text{ for } p : \text{leaf}_p = \text{leaf}_i, \sigma_p < \sigma_i \text{ for } i = 1..n$;
    END IF
    $\text{loss}(T_c) \leftarrow \cos(\Delta, G)$;
  END FOR
  $T \leftarrow \arg\min_T \text{loss}(T_c)$;
END FOR
FOR $r' \leftarrow 1$ to s **DO**
  $\text{leaf}_{r'} \leftarrow \text{GetLeaf}(x_i, T, \sigma_{r'}) \text{ for } i = 1..n$;
  IF Mode = Plain **THEN**
    $M_{r'}(i) \leftarrow M_{r'}(i) - \alpha\text{avg}(\text{grad}_{r'}) \text{ for } p : \text{leaf}_{r'}(p) = \text{leaf}_{r'}(i) \text{ for } i = 1..n$;
  END IF
  IF Mode = Ordered **THEN**
    FOR $j \leftarrow 1$ to $\log_2 n$ **DO**
      $M_{r',j}(i) \leftarrow M_{r',j}(i) - \alpha\text{avg}(\text{grad}_{r',j}(p) \text{ for } p : \text{leaf}_{r'}(p) = \text{leaf}_{r'}(i), \sigma_{r'}(p) \leqslant 2^j) \text{ for } i = 1..n$;
    END FOR
  END IF
END FOR
RETURN T, M

类别不平衡的特点使用了代价敏感学习方法。事实上，关于含有类别特征的数据集，还可以使用更新的 SMOTE-ENC[6] 进行小类别样本合成。

由于原始数据集较小，所以在使用了代价敏感学习方法之后性能并没有十分出色。从一般常识来说，小数据集不适合深度学习方法。但近年来提出的 TabTransformer[7] 可能会有不错的性能，值得后续的尝试。

最终 SVM 在使用了 SMOTE-NC 进行少样本数据合成后的数据集上训练取得了最佳效果。

参考文献

[1] CHAWLA N V, BOWYER K W, HALL L O, et al. Smote: synthetic minority over-sampling technique[J]. Journal of artificial intelligence research, 2002, 16:321-357.

[2] CHUNG M K. Introduction to logistic regression[J]. arXiv preprint arXiv:2008.13567, 2020.

[3] GÓMEZ R S, et al. Centered and non-centered variance inflation factor [J]. arXiv preprint arXiv:1905.12293, 2019.

[4] HEARST M, DUMAIS S, OSUNA E, et al. Support vector machines [J/OL]. IEEE Intelligent Systems and their Applications, 1998, 13(4): 18-28. DOI: 10.1109/5254.708428.

[5] PROKHORENKOVA L, GUSEV G, VOROBEV A, et al. Catboost: unbiased boosting with categorical features[J]. Advances in neural information processing systems, 2018, 31.

[6] MUKHERJEE M, KHUSHI M. Smote-enc: A novel smote-based method to generate synthetic data for nominal and continuous features [J]. Applied System Innovation, 2021, 4(1):18.

[7] HUANG X, KHETAN A, CVITKOVIC M, et al. Tabtransformer: Tabular data modeling using contextual embeddings[J]. arXiv preprint arXiv:2012.06678, 2020.