班级·学号 计卓220A班202208070127 姓名 杨征西 实验日期 2025/3/28 任课教师 项炬

|  |  |  |
| --- | --- | --- |
| **实验名称** | 实验1 机器学习模型评估方法实践 | **验证型** |
| **实验目的及要求：**  1.掌握留出法、交叉验证法、自助法等数据集拆分方法；  2.掌握错误率、准确率、精确度、召回率、F1指标、真阳性率、假阳性率等指标的计算方法；  3.能够计算并绘制Precision-Recall（PR）曲线，并计算曲线下面积；  4.能够计算并绘制ROC曲线，并计算曲线下面积；  5.了解调用机器学习算法实现算法性能评估及预测的基本流程。 | | |
| **实验内容：** | | |
| 注：建议采用python、matlab  【实验项目1：数据集拆分方法】   1. “留出法” 拆分数据集的实现与测试分析； 2. “交叉验证法”拆分数据集的实现与测试分析； 3. “自助法”拆分数据集的实现与测试分析。   注：自行随机生成数据集（特征+标签），或调用scikit-learn内置数据集。  【实验项目2：评估指标计算方法】   1. “错误率”、“准确率”、“精确度” 、“召回率” 、“F1”、“真阳性率” 、“假阳性率”等指标的计算函数实现与测试分析； 2. Precision-Recall（PR）曲线与曲线下面积的计算函数实现与测试分析； 3. ROC曲线与计算曲线下面积的计算函数实现与测试分析。   注：可自行生成预测分值列表和标签列表用于测试代码，或调用scikit-learn内置数据集和预测方法。  【实验项目3：评估流程实践】   1. 尝试采用“交叉验证”或者其它方式，评估机器学习算法的泛化性能，了解评估流程（应多次拆分数据集，训练、测试模型，计算评估指标的平均值、方差等）。     参考教材资料  《机器学习》，周志华，清华大学出版社，2016年  sklearn模型选择和评估 https://www.scikitlearn.com.cn/0.21.3/30/  实验考核材料  （1）实验报告：按实验内容及任务要求，记录各项任务的关键步骤和截屏；  （2）代码脚本：保存完整的执行脚本。  将所有材料打包，提交到作业系统。 | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **操作步骤：**  【实验项目1：数据集拆分方法**】**  **1.实现“留出法”，并观察、分析数据集拆分结果。**   1. 原理阐述。阐述“留出法”的原理。   把全部数据按照一定比例分成两部分，一部分用来训练模型，叫训练集；另一部分用来测试训练好的模型效果，叫测试集，像 70% 的数据作为训练集，30% 作为测试集。在分之前，一般要把数据打乱，让每个数据都有平等机会被分到训练集或测试集里。之后用训练集训练模型，再用测试集去测试这个模型，看看模型在没见过的数据上表现怎么样，这就能大概知道模型的真实能力。为了让评估结果更靠谱，通常会多划分几次数据集，每次划分后都训练和测试模型，最后综合多次结果来评价模型。   1. 方法实现  * （自己）实现“留出法”拆分数据集的函数（非调用）。 * 对输入、输出参数等做出恰当注释（报告中直接提供代码截图即可）。      1. 数据集准备。至少包含两类样本，随机生成不同规模（例如，样本数=10、50、100）的数据集，或者导入python或matlab数据集，并给出各类样本的数量及比例（表1）。  |  |  |  | | --- | --- | --- | | 表1：数据集样本分布情况（附件材料需提供相应的数据集）。 | | | | 数据集  名称 | 样本数量/比例 | | | 类别1（0） | 类别2（1） | | 数据集1 | 503 | 497 | | 数据集2 | 700 | 300 | | 数据集3 | 902 | 98 |  1. 样本分布分析。在给定数据规模下（例如N=100），设定训练集、测试集分别占90%和10%时（或其他比例），利用其数据集拆分函数，多次（3-5次）拆分数据集，通过表或图的方式（自拟），观察样本数量及比例的波动，包括：  * 观察所得到的训练集、测试集中各自“样本数量”，及训练集与测试集“样本比例”的变化（表或图展示）,并解释其原因；     图表 1训练集和测试集样本数量及比例变化  样本数量与比例不变：由于设置了test\_size = 0.1，所以每次拆分数据集时，测试集样本数量会固定为总样本数量的 10%，训练集样本数量为总样本数量的 90%。在样本总数不变的情况下，这两个数量应该是固定的，不会有波动。   * 观察“训练集”中包含的不同类别的样本数量及比例的变化（表或图展示）,并解释其原因；     图表 2训练集不同类别样本数量和比例变化  每次调用hold\_out\_split函数时，np.random.shuffle(indices)会对数据的索引进行随机打乱。这就导致每次划分训练集和测试集时，数据的分布是随机的，不同类别的样本被分配到训练集的数量和比例也会不同。由于随机打乱的存在，不同类别的样本在每次拆分时可能以不同的比例进入训练集。例如，在某次拆分中，类别 0 的样本可能较多地被分配到训练集，而在另一次拆分中，类别 1 的样本可能更多地进入训练集。这种随机性使得训练集中不同类别的样本数量和比例产生波动   * 观察“测试集”中包含的不同类别的样本数量及比例的变化（表或图展示）,并解释其原因。     图表 3测试集不同类别样本数量和比例变化  理由同训练集。  思考题：如果数据集的样本规模更小（例如N=10），结果如何？结合上述样本数量、样本比例变化情况，分析变化幅度的影响因素，如何减小波动？在极端条件下，可能出现（测试集或训练集中）某些类别缺失的问题。哪些情况下容易出现类别缺失问题，如何缓解这类问题。   1. 当数据集的样本规模更小时（如 N = 10），训练集和测试集的样本数量及比例波动情况会更加明显。 2. 样本规模；不同类别的样本数量；拆分比例等都会影响变化幅度，可以采取增加样本规模、分层抽样、多次拆分取平均等方法来减小波动。 3. 对于类别缺失的问题，可以采取数据增强、调整拆分策略、合并类别的方法来缓解类别缺失的情况。   **2.实现“交叉验证法”，并观察、分析数据集拆分结果。**   1. 原理阐述。 阐述“交叉验证法”的原理。   交叉验证法是一种用于评估机器学习模型性能的技术，其原理是将原始数据集分成多个子集，通过在不同子集上进行训练和验证，来更全面、客观地评估模型的性能，避免因数据集划分方式不同而导致的评估偏差   1. 方法实现。 （自己）实现“交叉验证法”拆分数据集的函数（非调用），对输入输出参数等做出注释（报告中直接提供代码截图即可）。      1. 数据集准备。同上。 2. 样本波动分析  * 利用其函数将数据集（多次）拆分成3或者5份（子集合），通过表或图的方式（自拟），观察每个子集合中，不同类别的样本数量及比例的变化（图、表自拟）；     图表 4数据集1不同类别样本数量及比例    图表 5数据集2不同类别样本数量及比例    图表 6数据集3不同类别样本数量及比例   * 分析变化的影响因素，如何减小这种变化，如何避免子集合中某些类别缺失的情况？  1. 确保数据集足够大：如果数据集本身规模较小，那么在拆分时更容易出现某些类别缺失的情况。增加数据集的规模可以降低这种风险。 2. 检查和调整拆分结果：在拆分完成后，检查每个子集合中各类别样本的数量，如果发现某个类别在某个子集合中缺失，可以适当调整拆分方式，例如从其他子集合中移动一些该类别的样本到缺失的子集合中。 3. 使用更合理的拆分算法：除了简单的随机拆分，还可以使用一些更复杂的拆分算法，如分 层k 折交叉验证，它可以在保证每个折（子集合）中各类别比例与原始数据集相似的同时，避免类别缺失的情况。   **3.实现“自助法”，并观察、分析数据集拆分结果。**   1. 原理阐述。阐述“自助法”的原理。   “自助法”（Bootstrap Sampling）是一种有放回的抽样方法，常用于在样本量有限的情况下评估模型的性能。其基本原理如下：假设原始数据集有 N 个样本，自助法通过有放回地从原始数据集中抽取 N 个样本（每次抽取一个样本后，将其放回数据集，以便下一次仍有机会被抽到），组成一个新的数据集，称为自助样本集（即训练集）   1. 方法实现。（自己）实现“自助法”拆分数据集的函数（非调用），对输入输出参数等做出注释（报告中直接提供代码截图即可）。      1. 数据集准备。同上。 2. 样本波动分析。利用其函数（多次）产生训练集和测试集，通过表或图的方式（自拟）展示统计结果：  * 观察、分析测试集样本数占总样本量的比例及变化情况（图、表自拟）；     图表 7测试集样本数占比变化   * 观察、分析测试集中不同类别的样本数量及比例的变化情况（图、表自拟）。     图表 8测试集中不同类别的样本比例变化    图表 9测试集中不同类别的样本数量变化  【实验项目2：评估指标计算方法】   * 1. 原理阐述   阐述“错误率”、“准确率”、“精确度” 、“召回率” 、“F1”、“真阳性率” 、“假阳性率”等指标，以及ROC曲线和Precision-Recall（PR）曲线的定义。   1. 错误率：分类错误的样本数占总样本数的比例。 2. 准确率：分类正确的样本数占总样本数的比例。 3. 精确度：在所有被预测为正类的样本中，实际为正类的比例。 4. 召回率：在所有实际为正类的样本中，被正确预测为正类的比例。 5. F1：综合考虑精确度和召回率的指标，是精确度和召回率的调和平均值。 6. 真阳性率：同召回率，即实际为正类的样本中被正确预测为正类的比例。 7. 假阳性率：在所有实际为负类的样本中，被错误预测为正类的比例。 8. ROC 曲线：以假阳性率为横轴、真阳性率为纵轴绘制的曲线，用于展示分类器在不同阈值下的性能。 9. Precision - Recall（PR）曲线：以召回率为横轴、精确度为纵轴绘制的曲线，用于展示分类器在不同阈值下精确度和召回率的权衡。    1. 方法实现  * 实现“错误率”、“准确率”、“精确度” 、“召回率” 、“F1”、“真阳性率” 、“假阳性率”等指标的计算函数（可将多个指标的计算集成在一个函数中，直接提供代码）。      * 实现ROC曲线及曲线下面积（AUPROC）的计算函数。      * 实现Precision-Recall（PR）曲线及曲线下面积（AUPRC）的计算函数。      * 1. 方法测试   利用自己实现的指标计算函数和（python或matlab等）系统自带的函数，在给定两个标签向量（真实的标签向量和预测的标签向量）情况下，计算“错误率”、“准确率”、“精确度” 、“召回率” 、“F1”、“真阳性率” 、“假阳性率”、AUPROC、AUPRC等指标，ROC曲线与Precision-Recall（PR）曲线，并展示、分析结果。    图表 10数据集1的ROC和PR曲线    图表 11数据集2的ROC和PR曲线    图表 12数据集3的ROC和PR曲线   |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | |  | | | | | | | | | | | 测试 | 错误率 | 准确率 | 精确度 | 召回率 | F1 | 真阳性率 | 假阳性率 | AUPROC | AUPRC | | 数据集1自定义函数 | 0.399 | 0.601 | 0.578526 | 0.726358 | 0.644068 | 0.726358 | 0.522863 | 0.668292 | 0.699265 | | 数据集1系统自带函数 | 0.399 | 0.601 | 0.578526 | 0.726358 | 0.644068 | 0.726358 | 0.522863 | 0.668292 | 0.701277 | | 数据集2自定义函数 | 0.338 | 0.662 | 0.448649 | 0.553333 | 0.495522 | 0.553333 | 0.291429 | 0.742990 | 0.617718 | | 数据集2系统自带函数 | 0.338 | 0.662 | 0.448649 | 0.553333 | 0.495522 | 0.553333 | 0.291429 | 0.742990 | 0.621051 | | 数据集3自定义函数 | 0.143 | 0.857 | 0.340426 | 0.489796 | 0.401674 | 0.489796 | 0.103104 | 0.836735 | 0.554672 | | 数据集3系统自带函数 | 0.143 | 0.857 | 0.340426 | 0.489796 | 0.401674 | 0.489796 | 0.103104 | 0.836735 | 0.564876 |   注：可自行生成预测分值列表和标签列表用于测试代码，或调用scikit-learn内置数据集和预测方法。  【实验项目3：评估流程实践】  1.阐述采用“交叉验证”或者其它方法，评估机器学习算法泛化性能的一般流程。  交叉验证是一种常用的评估机器学习算法泛化性能的方法，其核心思想是将原始数据集进行多次划分，在不同的子集上进行训练和测试，从而更全面地评估模型的性能。以 k 折交叉验证为例，其一般流程如下：   1. **数据准备**：收集和整理用于训练和评估模型的数据集。该数据集应包含特征矩阵（如样本的各种属性值）和对应的标签向量（如分类问题中的类别标签）。 2. **选择k值**：确定将数据集划分为k个子集的数量。k值的选择通常根据数据集大小和计算资源来决定，常见的取值有5、10等。较大的k值能更充分利用数据，但计算成本也更高；较小的k 值计算速度快，但评估结果的稳定性可能较差。 3. **划分数据集**：将原始数据集随机且均匀地划分为k个互不相交的子集，每个子集的大小尽量相等。 4. **循环训练和测试**：进行k轮训练和测试。在每一轮中，选择其中一个子集作为测试集，其余k-1个子集合并作为训练集。使用训练集训练模型，然后用测试集评估模型性能，记录下相应的性能指标（如准确率、召回率、F1值等）。 5. **计算平均性能指标**：完成k轮循环后，计算k次评估得到的性能指标的平均值和标准差。平均值用于评估模型的整体性能，标准差则反映了性能指标的波动程度，标准差越小，说明模型的性能越稳定。 6. **结果分析**：根据计算得到的平均性能指标和标准差，评估模型的泛化能力。如果平均性能指标较高且标准差较小，说明模型在不同数据集上表现较为稳定，泛化能力较强；反之，则可能需要调整模型、增加数据量或尝试其他算法。   2.以某机器学习算法为例，实现评估流程代码。可以某模拟或真实数据集为例，结合性能评估指标函数，给出该机器学习算法的预测性能指标（曲线），图表皆可。  以线性回归预测加利福尼亚房价结果为例，给出代码    其线性回归预测结果    其线性回归残差图 | | | | |
| **评阅成绩及评语** | **评 语** | | | |
| **操作正确，操作步骤书写清楚、工整**  **操作较正确，操作步骤书写较清楚**  **操作基本正确，操作步骤书写基本清楚**  **操作部分正确，操作步骤书写部分清楚**  **操作错误多，操作步骤书写不清楚** | | | |
| **评阅成绩** | **优 良 中 及格 不及格** | | |
| **评阅教师** |  | **评阅日期** | **年　　月　　日** |