山东大学 软件 学院

机器学习 课程实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 学号： 202200201095 | 姓名： 杨伟康 | | 班级： 22网安 |
| 实验题目： 贝叶斯分类器在鸢尾花数据集上的实现 | | | |
| 实验学时：4 | | 实验日期： 10.17 | |
| **1.实验目的：**  - 探讨贝叶斯分类器在鸢尾花数据集上的表现。  - 验证贝叶斯分类器的分类效果 | | | |
| **2.引言：**  **背景和动机**  在机器学习和统计学领域，分类问题一直是研究的热点之一。分类任务的目标是根据输入的特征将样本分配到预定义的类别中。贝叶斯分类器作为一种基于贝叶斯定理的分类方法，因其坚实的数学基础和良好的分类性能，在实际应用中得到了广泛的关注。鸢尾花数据集作为机器学习领域的经典数据集之一，常被用于分类算法的演示和测试。本文旨在通过实现贝叶斯分类器在鸢尾花数据集上的应用，验证其分类效果，并探讨其在实际应用中的潜力。  **问题描述**  本文要解决的问题是：使用贝叶斯分类器对鸢尾花数据集进行分类，并评估其分类性能。鸢尾花数据集包含150个样本，每个样本有4个特征（花萼长度、花萼宽度、花瓣长度、花瓣宽度），并分为3个类别（Setosa、Versicolour、Virginica）。  **文献综述**  贝叶斯分类器基于贝叶斯定理，通过计算样本属于各个类别的后验概率来进行分类。朴素贝叶斯分类器是贝叶斯分类器的一种简化形式，它假设特征之间相互独立，从而简化了计算过程。尽管这种假设在实际应用中往往不成立，但朴素贝叶斯分类器在许多实际应用中仍表现出良好的性能。在鸢尾花数据集上，已有多种分类算法被应用，如决策树、支持向量机等。然而，贝叶斯分类器，特别是朴素贝叶斯分类器，因其简单性和有效性，仍然是一个值得研究的课题。 | | | |
| **3.实验方法：**  数据集：  Iris鸢尾花数据集（Iris Dataset），包含150个样本，分为三类：Setosa、Versicolor、Virginica。每个样本具有4个特征：花萼长度、花萼宽度、花瓣长度、花瓣宽度。  数据集划分：  将数据集分为训练集（70%）和测试集（30%），或使用交叉验证进行评估。  模型选择：  本实验选择的模型是朴素贝叶斯分类器。朴素贝叶斯分类器基于贝叶斯定理，并假设特征之间相互独立。尽管这种假设在实际应用中可能不成立，但朴素贝叶斯分类器在许多实际应用中仍表现出良好的性能。在本实验中，我们将使用Python的scikit-learn库来实现朴素贝叶斯分类器。  评估指标  为了评估模型的性能，我们将使用准确率（accuracy）作为评估指标。准确率是正确分类的样本数与总样本数的比例。在鸢尾花数据集上，准确率是一个直观且有效的评估指标，因为数据集的类别分布是均衡的。 | | | |
| **4.实验过程报告：**  **#### 4.1 数据预处理**  **- 导入数据集并检查数据的分布情况。**  **- 对数据进行归一化或标准化处理（可选，根据贝叶斯分类器的具体要求和实验需要）。**  **- 数据集划分：按训练集和测试集的比例进行随机划分。**  import numpy as np import matplotlib.pyplot as plt import seaborn as sns from sklearn.datasets import load\_iris from sklearn.model\_selection import train\_test\_split from sklearn.naive\_bayes import GaussianNB from sklearn.metrics import accuracy\_score, precision\_score, recall\_score, f1\_score, confusion\_matrix, \  ConfusionMatrixDisplay from sklearn.decomposition import PCA  # 加载数据集 iris = load\_iris() X = iris.data y = iris.target  # 数据集划分 X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=42)  **#### 4.2 构建贝叶斯分类器**  **- 选择贝叶斯分类器模型。** # 构建贝叶斯分类器 clf = GaussianNB()  **#### 4.3 模型训练**  **- 使用训练数据集对贝叶斯分类器进行训练。**  **- 记录训练时间及模型参数。** # 模型训练 clf.fit(X\_train, y\_train)  **#### 4.4 模型评估**  **- 使用测试数据集进行模型评估。**  **- 计算并分析模型的各项评价指标：准确率（Accuracy）、精确率（Precision）、召回率（Recall）、F1分数。**  **- 生成混淆矩阵以进一步分析分类效果。** # 模型评估 y\_pred = clf.predict(X\_test) accuracy = accuracy\_score(y\_test, y\_pred) precision = precision\_score(y\_test, y\_pred, average='weighted') recall = recall\_score(y\_test, y\_pred, average='weighted') f1 = f1\_score(y\_test, y\_pred, average='weighted') conf\_mat = confusion\_matrix(y\_test, y\_pred)  # 输出评估结果 print(f"Accuracy: {accuracy:.2f}") print(f"Precision: {precision:.2f}") print(f"Recall: {recall:.2f}") print(f"F1 Score: {f1:.2f}") print("Confusion Matrix:") print(conf\_mat)  **#### 4.5 可视化结果**  **- 使用散点图展示不同类别的分类效果。**  **- 可视化决策边界（如果适用），展示分类器如何将不同的类别分离。**  **- 通过学习曲线或交叉验证曲线，分析模型在不同训练集大小下的表现。** # 数据可视化 # 使用PCA将数据降维到2D pca = PCA(n\_components=2) X\_pca = pca.fit\_transform(X)  # 训练集和测试集的PCA转换 X\_train\_pca = pca.transform(X\_train) X\_test\_pca = pca.transform(X\_test)  # 绘制散点图展示分类效果 plt.figure(figsize=(10, 6))  # 训练集散点图 for i, color in enumerate(iris.target\_names):  plt.scatter(X\_train\_pca[y\_train == i, 0], X\_train\_pca[y\_train == i, 1], label=color, alpha=0.5)  # 测试集散点图及预测结果 for i, color in enumerate(iris.target\_names):  pred\_indices = np.where(y\_pred == i)  plt.scatter(X\_test\_pca[pred\_indices, 0], X\_test\_pca[pred\_indices, 1], edgecolor='k', facecolor='none', s=100,  label=f'{color} (pred)')  plt.legend() plt.title('Iris Dataset Classification with Gaussian Naive Bayes (PCA-reduced Data)') plt.xlabel('Principal Component 1') plt.ylabel('Principal Component 2') plt.show()  # 绘制混淆矩阵热图 disp = ConfusionMatrixDisplay(confusion\_matrix=conf\_mat, display\_labels=iris.target\_names) disp.plot(cmap=plt.cm.Blues) plt.title('Confusion Matrix') plt.show() | | | |
| **5. 实验结果与分析**  贝叶斯分类器在鸢尾花数据集上通常表现出良好的性能，准确率较高。由于其基于概率的预测方式，贝叶斯分类器在处理具有明显类别区分特征的数据集时尤为有效。在鸢尾花数据集中，由于不同种类的鸢尾花具有独特的花瓣和萼片长度、宽度等特征，这些特征能够被贝叶斯分类器有效地利用来进行分类。然而，贝叶斯分类器的性能也依赖于特征的独立性和数据的准确性，如果特征之间存在复杂的依赖关系或数据存在噪声，其性能可能会受到影响。  **6. 结论**  贝叶斯分类器在鸢尾花数据集上展现出了良好的分类能力，具有高准确率和较强的泛化能力。然而，它也存在一些局限性，如特征独立性假设和数据噪声的敏感性。为了进一步提高贝叶斯分类器的性能，可以考虑以下改进方向：一是通过特征工程来提取更具代表性的特征，减少特征之间的相关性；二是采用更复杂的贝叶斯模型，如朴素贝叶斯的扩展版本，以更好地处理特征之间的依赖关系；三是结合其他机器学习算法，如集成学习方法，来进一步提升分类性能。未来实验可以探索这些改进方向，并评估它们对贝叶斯分类器性能的具体影响。 | | | |

1. 当次实验结束后一周内按班组织上交实验报告。

2. 实验报告文件命名为：“学号+姓名”，格式为WORD文档。