机器学习-点集分类

韩琳
hanlin3309@163.com

Abstract

该报告使用Decision Trees, AdaBoost + DecisionTrees，SVM方法对月亮形状数据集进行分类，并且对于SVM方法尝试了三种核函数，最终Decision Trees, AdaBoost + DecisionTrees，SVM (rbf Kernel)得到较准确的分类结果，SVM (linear Kernel)、SVM (poly Kernel)分类结果不佳，体现了Decision Trees，SVM (rbf Kernel)两种方法在处理非线性数据时具有较好的效果

Introduction

二分类是模式识别的基础任务，广泛用于医疗诊断、欺诈检测等。Decision Trees, AdaBoost + DecisionTrees，SVM等算法是常见的二分类解决算法，决策树易于解释，AdaBoost提升分类性能，非线性核函数的SVM擅长处理高维数据，三者各有优势。

Methodology

1.Decision Tree

决策树是一种流行且功能强大的工具，用于机器学习和数据挖掘，用于分类和回归任务。它是一个树状结构模型，其中内部节点表示对属性的测试，分支表示这些测试的结果，叶节点表示决策结果或类标签。从根到叶的路径表示分类规则或回归路径。

```
# 决策树分类器
dt_clf = DecisionTreeClassifier(random_state=42)
dt_clf.fit(X_train, y_train)
dt_pred = dt_clf.predict(X_test)
print("Decision Tree Performance:")
print(classification_report(y_test, dt_pred))
print("Accuracy:", accuracy_score(y_test, dt_pred))
# 绘制决策树结果
plot_predictions(X_test, y_test, dt_pred, 'Decision Tree Predictions')
```

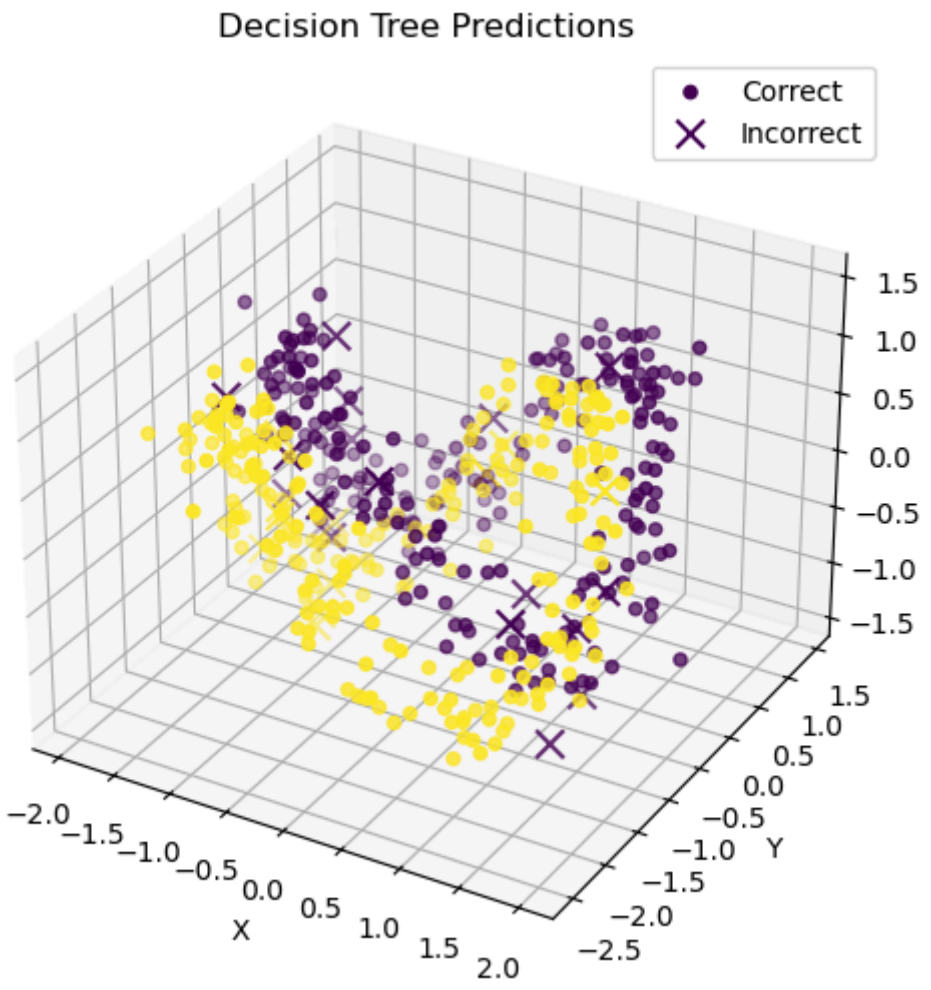
本实验调用**sklearn**库中的 **DecisionTreeClassifier**方法，选取random_state=42的随机等价划分保证结果的可重复性并且控制随即结果，训练结果在测试集上的Accuracy达到0.94

```
Decision Tree Performance:
              precision    recall  f1-score   support

         0.0       0.95      0.92      0.94       250
         1.0       0.93      0.96      0.94       250

    accuracy                           0.94       500
   macro avg       0.94      0.94      0.94       500
weighted avg       0.94      0.94      0.94       500

Accuracy: 0.94
```



2.Decision Trees+AdaBoost

AdaBoost 背后的核心原则是将一系列弱学习器(即，仅比随机猜测稍好一点的模型，例如小决策树)拟合反复修改的数据版本。然后，通过加权多数票(或总和)将所有这些预测组合在一起，以生成最终预测。

```
# AdaBoost + 决策树分类器
ab_clf =
AdaBoostClassifier(estimator=DecisionTreeClassifier(random_state=40),
n_estimators=50, random_state=42)
ab_clf.fit(X_train, y_train)
ab_pred = ab_clf.predict(X_test)
print("\nAdaBoost + Decision Tree Performance:")
print(classification_report(y_test, ab_pred))
print("Accuracy:", accuracy_score(y_test, ab_pred))
plot_predictions(X_test, y_test, ab_pred, 'AdaBoost Predictions')
```

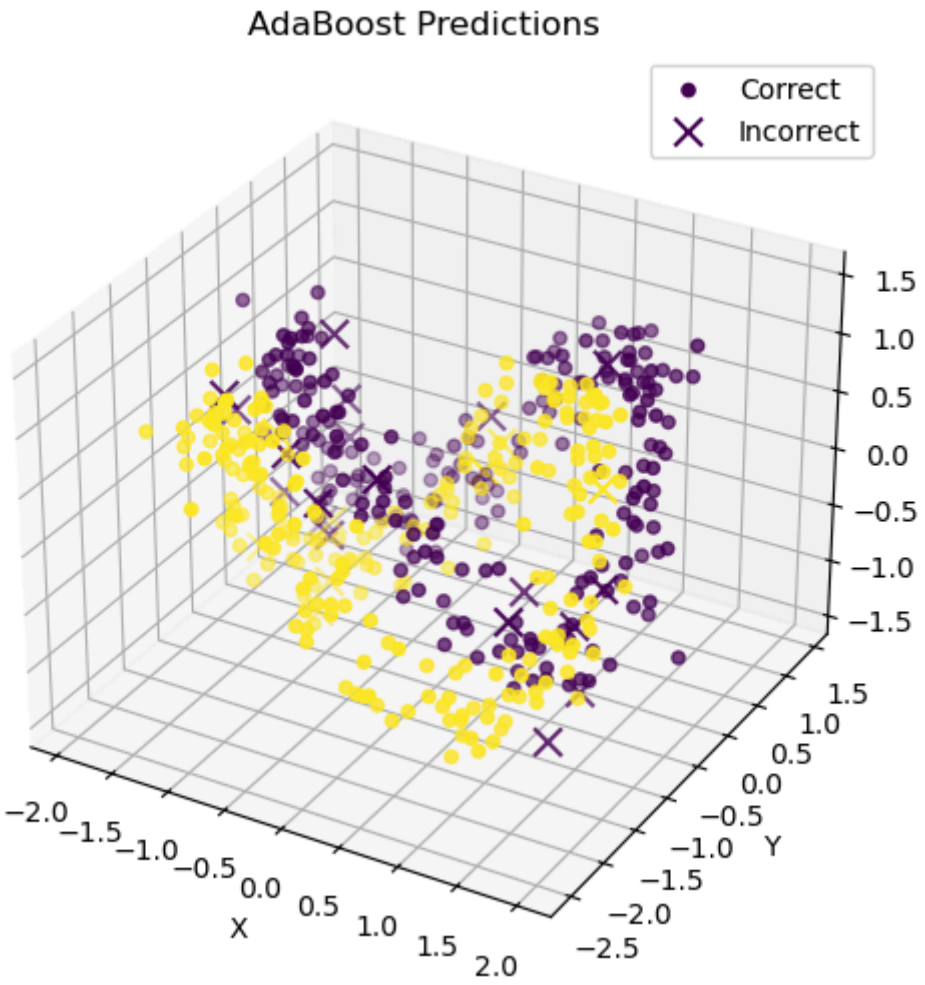
本实验调用**sklearn**库中的 **AdaBoostClassifier**方法，选取random_state=40的随机等价划分子决策树，选用选取random_state=42划分Adaboost本身，保证结果的可重复性并且控制随即结果，n_estimators取50表示使用50个决策树进行集成，训练结果在测试集上的Accuracy达到0.944，准确率略微高于Decision Trees,效果提升不明显

```
AdaBoost + Decision Tree Performance:
              precision    recall  f1-score   support

         0.0       0.97      0.92      0.94       250
         1.0       0.92      0.97      0.95       250

    accuracy                           0.94       500
   macro avg       0.95      0.94      0.94       500
weighted avg       0.95      0.94      0.94       500

Accuracy: 0.944
```



3.SVM

支持向量机（**Support Vector Machine, SVM**） 是一种经典的有监督机器学习算法，主要用于分类和回归任务。其核心思想是通过在特征空间中寻找一个 最大化间隔的超平面，将不同类别的数据点分开，从而实现对新样本的高效分类。

```
# SVM 分类器（线性核）
svm_linear = SVC(kernel='linear', random_state=42)
svm_linear.fit(X_train, y_train)
svm_linear_pred = svm_linear.predict(X_test)
print("\nSVM (Linear Kernel) Performance:")
print(classification_report(y_test, svm_linear_pred))
print("Accuracy:", accuracy_score(y_test, svm_linear_pred))
plot_predictions(X_test, y_test, svm_linear_pred, 'SVM (linear Kernel)
Predictions')

# SVM 分类器（多项式核）
#多个degree值的多项式核SVM分类器
svm_poly = SVC(kernel='poly', degree=3, random_state=42)
svm_poly.fit(X_train, y_train)
svm_poly_pred = svm_poly.predict(X_test)
print("\nSVM (Polynomial Kernel) Performance:")
print(classification_report(y_test, svm_poly_pred))
print("Accuracy:", accuracy_score(y_test, svm_poly_pred))
```

```
plot_predictions(X_test, y_test, svm_poly_pred, 'SVM (poly Kernel)
Predictions')

# SVM 分类器（RBF核）
svm_rbf = SVC(kernel='rbf', random_state=42)
svm_rbf.fit(X_train, y_train)
svm_rbf_pred = svm_rbf.predict(X_test)
print("\nSVM (RBF Kernel) Performance:")
print(classification_report(y_test, svm_rbf_pred))
print("Accuracy:", accuracy_score(y_test, svm_rbf_pred))
plot_predictions(X_test, y_test, svm_poly_pred, 'SVM (rbf Kernel)
Predictions')
```

本实验调用**sklearn**库中的 **SVM**方法，选取random_state=42,并且使用linear、poly、rbf三种核函数，其中poly核函数调用了3次多项式进行拟合，得到三个核函数的预测结果，其中rbf核函数准确率最高，poly核函数其次，linear核函数方法准确率最低

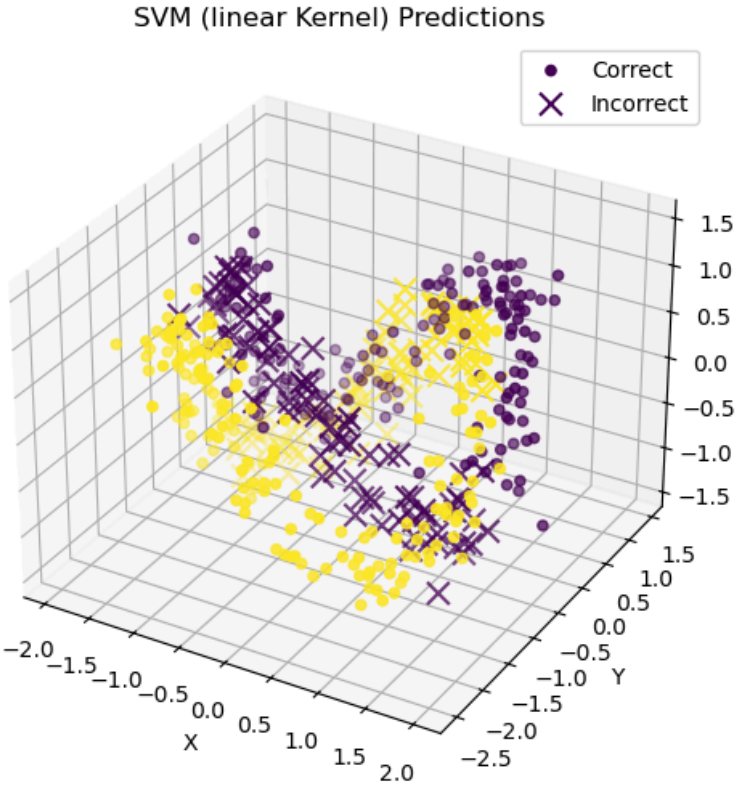
- **linear kernel**

```
SVM (Linear Kernel) Performance:
              precision    recall  f1-score   support

         0.0       0.67      0.66      0.66       250
         1.0       0.66      0.67      0.67       250

    accuracy                           0.67       500
   macro avg       0.67      0.67      0.67       500
weighted avg       0.67      0.67      0.67       500

Accuracy: 0.666
```



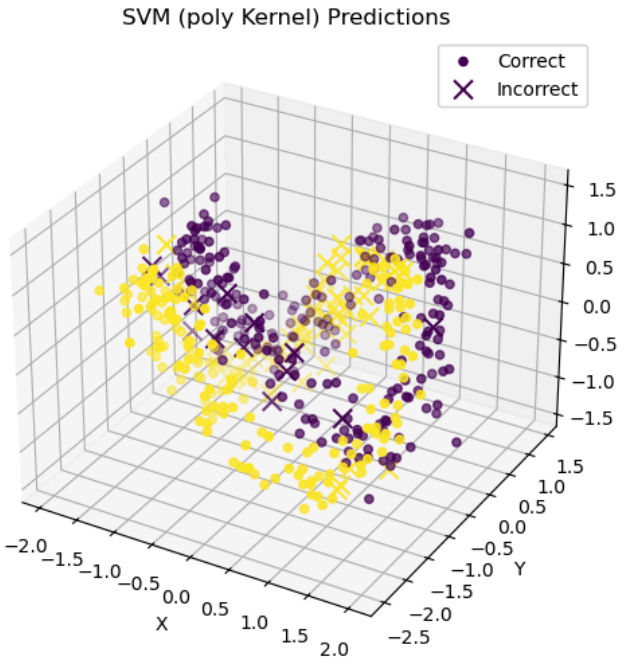
- **poly kernel**

```
SVM (Polynomial Kernel) Performance:
              precision    recall  f1-score   support

         0.0       0.79      0.95      0.86       250
         1.0       0.93      0.74      0.83       250

    accuracy                           0.85       500
   macro avg       0.86      0.85      0.84       500
weighted avg       0.86      0.85      0.84       500

Accuracy: 0.846
```



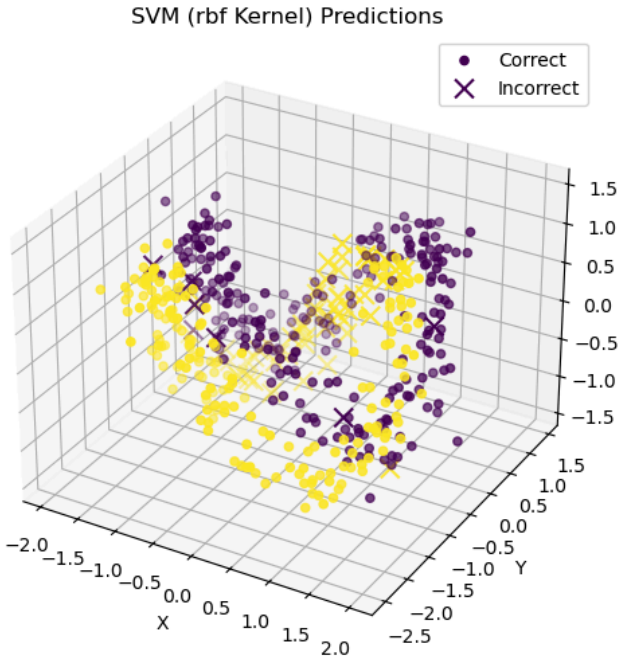
- **rbf kernel**

```
SVM (RBF Kernel) Performance:
              precision    recall  f1-score   support

         0.0       0.98      0.96      0.97       250
         1.0       0.96      0.98      0.97       250

    accuracy                           0.97       500
   macro avg       0.97      0.97      0.97       500
weighted avg       0.97      0.97      0.97       500

Accuracy: 0.972
```



Conclusions

本实验使用的几种分类方法中，Decision Trees, AdaBoost + DecisionTrees，RBF SVM分类准确率达到了0.94以上，其中RBF SVM方法分类效果最好，这体现了**高斯核SVM**在处理非线性数据二分类时的优势，能够在超平面高维度将数据准确划分，**多项式核方法以及线性核方法**准确率不佳，体现了这两种核函数在处理简单非线性数据时的局限性，启示我使用SVM方法时要根据数据的先验特征选取适合的核函数方法核参数，**决策树**算法在处理非线性问题时也同样体现出较好的性能，**AdaBoost** 通过集成弱决策树，在非线性拟合和泛化能力上优于单决策树。若数据接近线性，线性核 SVM 足够高效；若存在复杂非线性，高斯核 SVM 和集成方法（如 AdaBoost）更优。

方法	非线性拟合能力	泛化能力	参数复杂度	噪声鲁棒性	决策边界特性	适用数据类型
高斯核 SVM	最强（无限维）	强（间隔最大化）	低（2 参数）	强	连续平滑曲面	复杂非线性数据
AdaBoost 决策树	较强（集成分裂）	较强（降低方差）	中（基学习器参数）	中等	集成阶梯边界	中等非线性数据
决策树	中等（单棵分裂）	弱（易过拟合）	低（剪枝参数）	弱	单棵阶梯边界	简单非线性数据
多项式核 SVM	有限（固定多项式）	弱（高次过拟合）	高（3 参数）	弱	固定多项式曲面	明确多项式关系数据
线性核 SVM	无（仅线性）	中等（依赖 C）	最低（1 参数）	中等	线性超平面	严格线性可分数据