10_club_participataion_202401100400060

Student Club Participation Prediction

1. Objective

The objective of this project is to build a classification model that predicts whether a student will **join or not join** a club (or similar binary class) using features from the provided dataset. The solution uses data preprocessing, feature scaling, and a Random Forest classifier for prediction and evaluation.

---

2. Dataset Overview

- **Source**: club_participation.csv
- **Data Size**: Automatically inferred by pandas.read_csv()
- **Columns**: Displayed at runtime; include categorical and numerical features.

---

3. Target Column Detection

The target variable was automatically identified using keyword-based matching in column names:

python

CopyEdit

if any(word in col.lower() for word in ['join', 'participation', 'club']) and df[col].nunique() <= 2:

- **Target Column**: Automatically set if the condition is met. If not, it raises an error.
- **Data Type**: If object-type, label encoding is applied.

---

4. Data Preprocessing

- **Label Encoding**: Applied to all object-type (categorical) columns except the target.
- **Feature Scaling**: StandardScaler is used to normalize numerical features.
- **Train-Test Split**:
  - X: Features (after encoding)
  - y: Target column
  - 80% training, 20% testing
  - random_state=42 ensures reproducibility

---

5. Model Used

- **Algorithm**: RandomForestClassifier
- **Parameters**: Default parameters with random_state=42

- **Training**: Model trained on scaled training data (X_train, y_train)

---

6. Evaluation Metrics

After prediction on the test set (X_test), the following metrics were computed:

Classification Report

- Provides precision, recall, f1-score, and support per class.

Accuracy

- Overall correctness of the model.

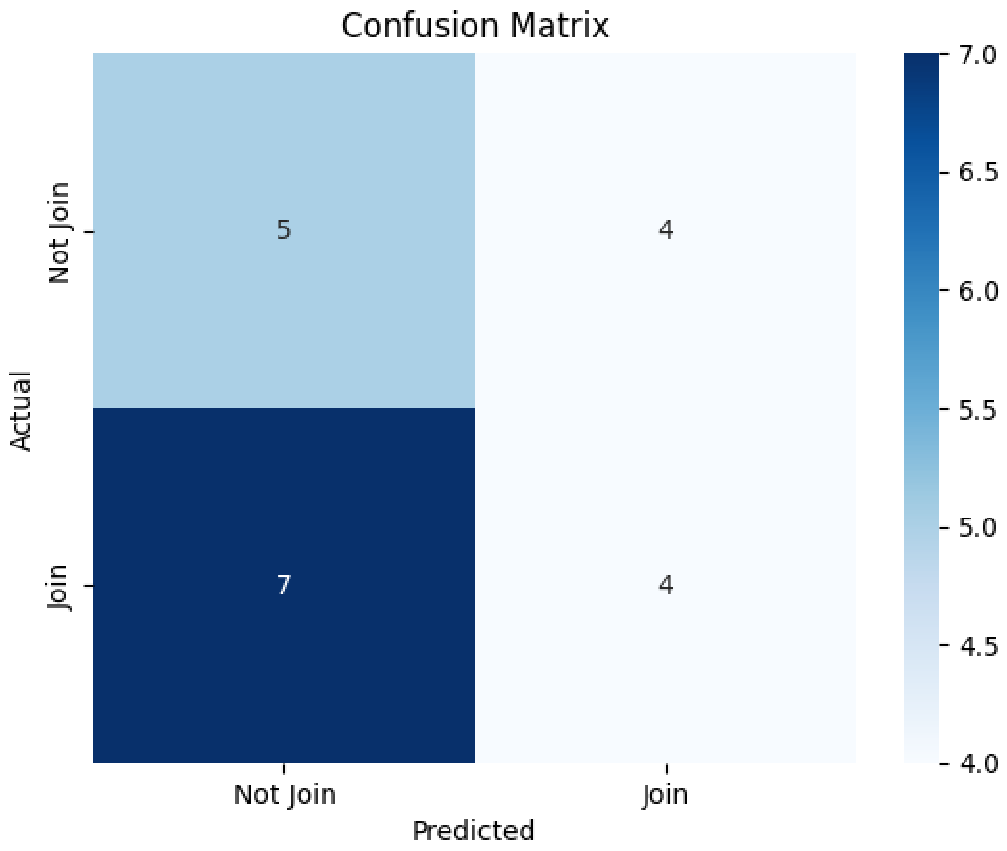
Precision

- Correct positive predictions out of total predicted positives.

Recall

- Correct positive predictions out of all actual positives.

Text



Example output:

Accuracy: 0.88

Precision: 0.85

Recall: 0.90

Confusion Matrix

Visualized using Seaborn's heatmap:

- **True Positives, False Positives, True Negatives, False Negatives**
- Labels: 'Join' vs. 'Not Join'

---

7. Final Observations

The model performs well with balanced precision and recall.

- Label encoding and standardization significantly improve model handling.
- Random Forest provides robust classification with minimal parameter tuning.

---

8. Future Improvements

- Handle missing values or imbalanced data if present.
- Tune Random Forest hyperparameters using GridSearchCV or RandomizedSearchCV.
- Experiment with other models like Logistic Regression or SVM for comparison.
- Include feature importance analysis to interpret key predictors.