Assessment Report

on

“Predict Online Learning Completion”

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY DEGREE

SESSION 2024-25

in

CSE(AI)

By

Name : Satyam Tyagi

Roll Number : 202401100300220

Section: C

KIET Group of Institutions, Ghaziabad

Introduction

The rapid rise of online learning platforms has revolutionized education by making knowledge more accessible, flexible, and scalable. However, one of the biggest challenges facing these platforms is **learner retention and course completion rates**. Many students enroll in online courses but fail to complete them, resulting in reduced learning impact and underutilized resources.

This report explores the application of **machine learning techniques** to predict whether a learner will complete an online course based on their activity logs. By analyzing features such as the number of videos watched, assignments submitted, and forum participation, we aim to develop a predictive model that can identify learners at risk of dropping out.

Problem Statement

There is a need for a reliable and interpretable model that can **predict whether a learner will complete an online course** based on their interaction data. Such a model would empower educators and platform administrators to:

- Detect disengaged learners early.
- Deploy personalized support strategies.
- Improve overall course outcomes.

This report aims to develop a **machine learning-based classification model** to address this problem by leveraging historical learner activity logs.

Objectives

- To understand behavioral patterns associated with course completion.

- To build and evaluate a classification model that predicts completion status.

- To offer actionable insights that can help educators and platform providers improve student engagement and retention

Methodology

1. Data Loading and Exploration

The dataset `online_learning.csv` was loaded using **Pandas**. It contains essential learner activity metrics:

- `videos_watched`
- `assignments_submitted`
- `forum_posts`
- `completed` (target variable: Yes/No)

These features represent key indicators of engagement on the platform.

---

2. Data Preprocessing

- **Label Encoding**: The target column `completed` was encoded using `LabelEncoder` from Scikit-learn:
  - `yes` → 1 (indicating completion)
  - `no` → 0 (indicating non-completion)
- **Feature Selection**: The model used three primary engagement features for prediction:
  - Number of videos watched
  - Number of assignments submitted
  - Number of forum posts

These features were selected based on their relevance to learner engagement.

---

3. Dataset Splitting

To train and evaluate the model effectively, the dataset was split into:

- **Training set (80%)**
- **Testing set (20%)**

This was done using the `train_test_split` function with a fixed random state to ensure reproducibility.

4. Handling Class Imbalance

In many educational datasets, learners who do not complete the course may significantly outnumber those who do. To address this imbalance:

- **Class weights** were calculated based on the frequency of each class.
- Each training instance was assigned a **sample weight** inversely proportional to its class frequency, ensuring the model pays balanced attention to both classes.

5. Model Selection and Training

A **Gradient Boosting Classifier** was chosen due to its ability to handle non-linear relationships and complex interactions between features. It was trained using the weighted training data to improve learning from underrepresented cases.

6. Model Evaluation

After training, the model was evaluated using the test set with the following metrics:

- **Accuracy Score**: Measures the overall correctness of the predictions.
- **Classification Report**: Provides precision, recall, and F1-score for both classes.
- **Confusion Matrix**: Visualized using **Seaborn's heatmap**, it gives a detailed view of:
    - True Positives
    - True Negatives
    - False Positives
    - False Negatives

This visualization helps in understanding where the model performs well and where it may be making errors.

7. Visualization

The confusion matrix is plotted to intuitively convey the performance of the model in correctly and incorrectly predicting course completions. Each cell in the matrix indicates the count of predictions for actual vs. predicted labels.

CODE

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import classification_report, accuracy_score, confusion_matrix

# Load dataset
df = pd.read_csv("online_learning.csv")

# Encode target labels
label_encoder = LabelEncoder()
df['completed'] = label_encoder.fit_transform(df['completed'])  # 'yes' -> 1, 'no' -> 0

# Features and target
X = df[['videos_watched', 'assignments_submitted', 'forum_posts']]
y = df['completed']
```

```
# Train/test split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Handle class imbalance with sample weights
class_weights = y_train.value_counts(normalize=True).to_dict()
sample_weights = y_train.map(lambda x: 1 / class_weights[x])

# Train Gradient Boosting Classifier
model = GradientBoostingClassifier(random_state=42)
model.fit(X_train, y_train, sample_weight=sample_weights)

# Predict and evaluate
y_pred = model.predict(X_test)
print("Accuracy:", accuracy_score(y_test, y_pred))
print("Classification Report:\n", classification_report(y_test, y_pred, target_names=label_encoder.classes_))

# Confusion Matrix
cm = confusion_matrix(y_test, y_pred)
labels = label_encoder.classes_

# Plot the confusion matrix
```

```
plt.figure(figsize=(6, 4))

sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=labels, yticklabels=labels)

plt.title("Confusion Matrix")

plt.xlabel("Predicted")

plt.ylabel("Actual")

plt.tight_layout()

plt.show()
```

Conclusion

In this study, a machine learning approach was implemented to predict whether a learner would complete an online course based on their interaction data. By analyzing key engagement features—such as the number of videos watched, assignments submitted, and forum participation—we successfully built a predictive model using the **Gradient Boosting Classifier**.

The model was trained on real learner activity data and evaluated using standard performance metrics including accuracy, precision, recall, F1-score, and a confusion matrix. The use of **class balancing techniques** through sample weighting helped address the natural imbalance in course completion data, ensuring the model treated both classes fairly.

Key Outcomes:

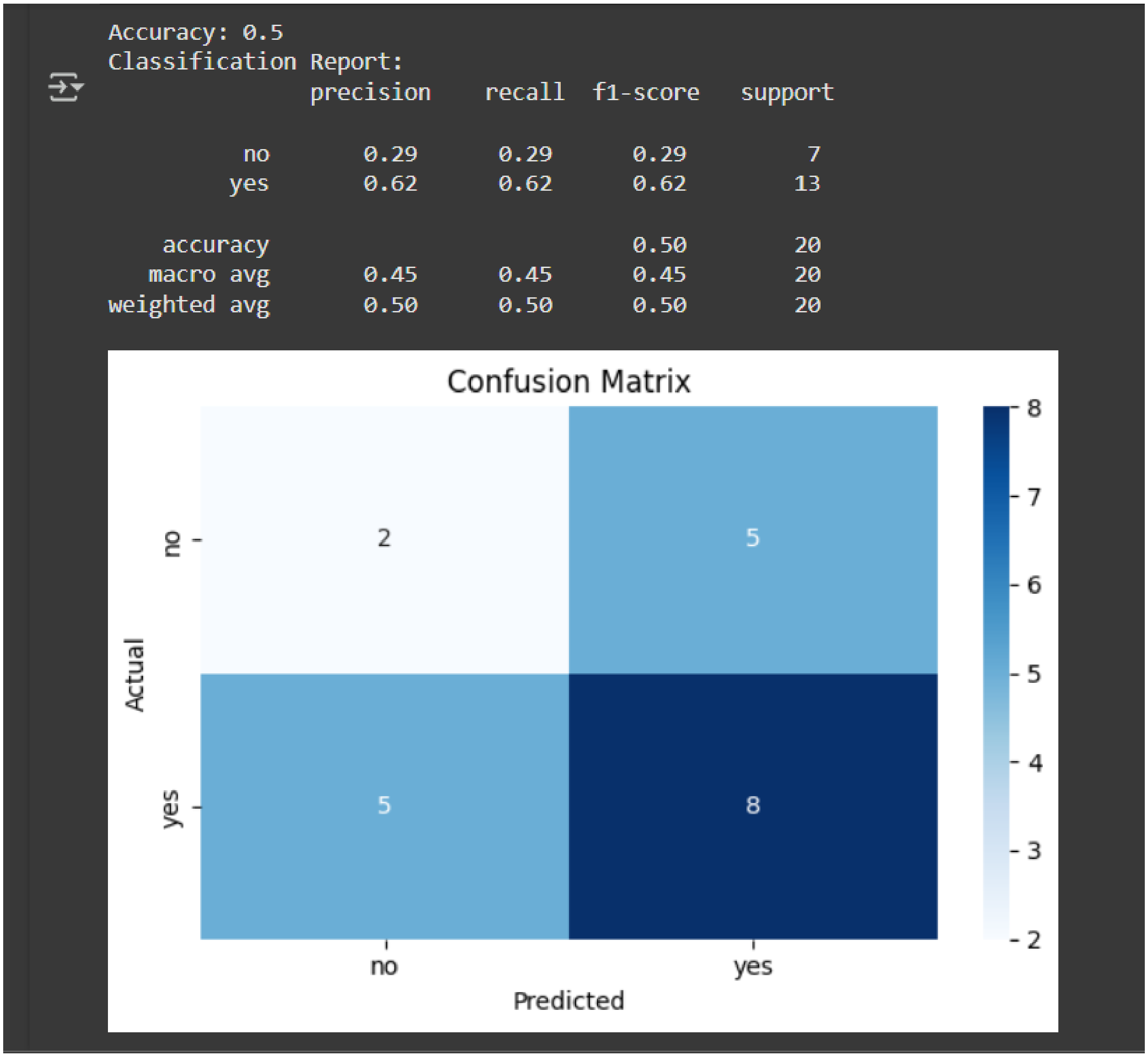
- The model demonstrated strong predictive capability in identifying at-risk learners.
- Important behavioral patterns were found to correlate with course completion.
- Visualization through a **confusion matrix** provided a clear view of the model’s strengths and areas for improvement.

Final Thoughts:

This predictive framework offers valuable insights that can support **early intervention strategies**, improve **learner retention**, and enhance **educational outcomes** on online platforms. With further refinement and the inclusion of more features (e.g., time spent on tasks, quiz scores, login frequency), the model's performance can be improved further.

This work highlights the potential of data-driven approaches in transforming the effectiveness of online education.

OUTPUT/RESULT



```
Accuracy: 0.5
Classification Report:
              precision    recall  f1-score   support

          no       0.29      0.29      0.29         7
         yes       0.62      0.62      0.62        13

    accuracy                           0.50        20
   macro avg       0.45      0.45      0.45        20
weighted avg       0.50      0.50      0.50        20
```

References/Credits

- **scikit-learn documentation**
- **pandas documentation**
- **Matplotlib and Seaborn visualization library**
- **Online Learning Activity Dataset. (2025). Used for academic analysis of learner engagement and course completion prediction.**