Assessment Report

on

"Predict Student Dropout"

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY DEGREE

SESSION 2024-25

in

CSE(AIML)

By

Name : Badal Singh

Roll Number : 202401100400071

Section: A

Under the supervision of

"Mr. Bikki Kumar"

KIET Group of Institutions, Ghaziabad

May, 2025

1. Introduction

Student dropout is a major concern in educational systems worldwide. Early identification of students who are at risk allows institutions to provide targeted interventions and improve overall retention rates. This project applies machine learning techniques to predict dropout risks based on measurable academic indicators such as attendance, grades, and participation. By building an intelligent predictive model, this project supports proactive academic counseling and student welfare planning.

---

2. Problem Statement

To develop a machine learning model that can accurately classify students based on their likelihood of dropping out using features like attendance percentage, academic performance, and classroom participation.

---

3. Objectives

- Analyze and preprocess the dataset for model training.
- Train a classification model using Random Forest.
- Evaluate model performance using standard classification metrics.
- Provide insightful visualizations to interpret model results.

---

4. Methodology

- **Data Acquisition**: A structured dataset containing student details such as attendance, grades, participation level, and dropout risk was used.
- **Data Preprocessing**:

    - Encoded categorical labels (`dropout_risk`: yes/no) into numeric format.
    - Standardized feature values using `StandardScaler`.
    - Split the dataset into 80% training and 20% testing subsets.
- **Model Building**: Trained a `RandomForestClassifier` due to its robustness and suitability for tabular data.
- **Model Evaluation**:
    - Measured accuracy, precision, recall, and F1-score.
    - Visualized confusion matrix, ROC curve, and feature importance to interpret model behavior..

---

5. Data Preprocessing

The dataset is cleaned and prepared as follows:

- Label encoding was applied to the target variable (`dropout_risk`), mapping `yes` to 1 and no to 0.
- All features (`attendance`, `grades`, and `participation`) were standardized for uniform scaling.
- The dataset was randomly split into training and testing datasets using an 80:20 ratio.

---

6. Model Implementation

A Random Forest model was implemented due to its high accuracy, resistance to overfitting, and interpretability. After training on the processed data, predictions were made on the test set, and the results were analyzed using multiple performance indicators.

---

7. Evaluation Metrics

The following metrics are used to evaluate the model:

- **Accuracy**: Overall correctness of the predictions.
- **Precision**:Measures how many predicted dropouts were actual dropouts.
- **Recall**:Measures how many actual dropouts were correctly identified.
- **F1 Score**: Balances precision and recall.
- **Confusion Matrix**: Visual tool to assess true/false positives and negatives.

---

8. Code

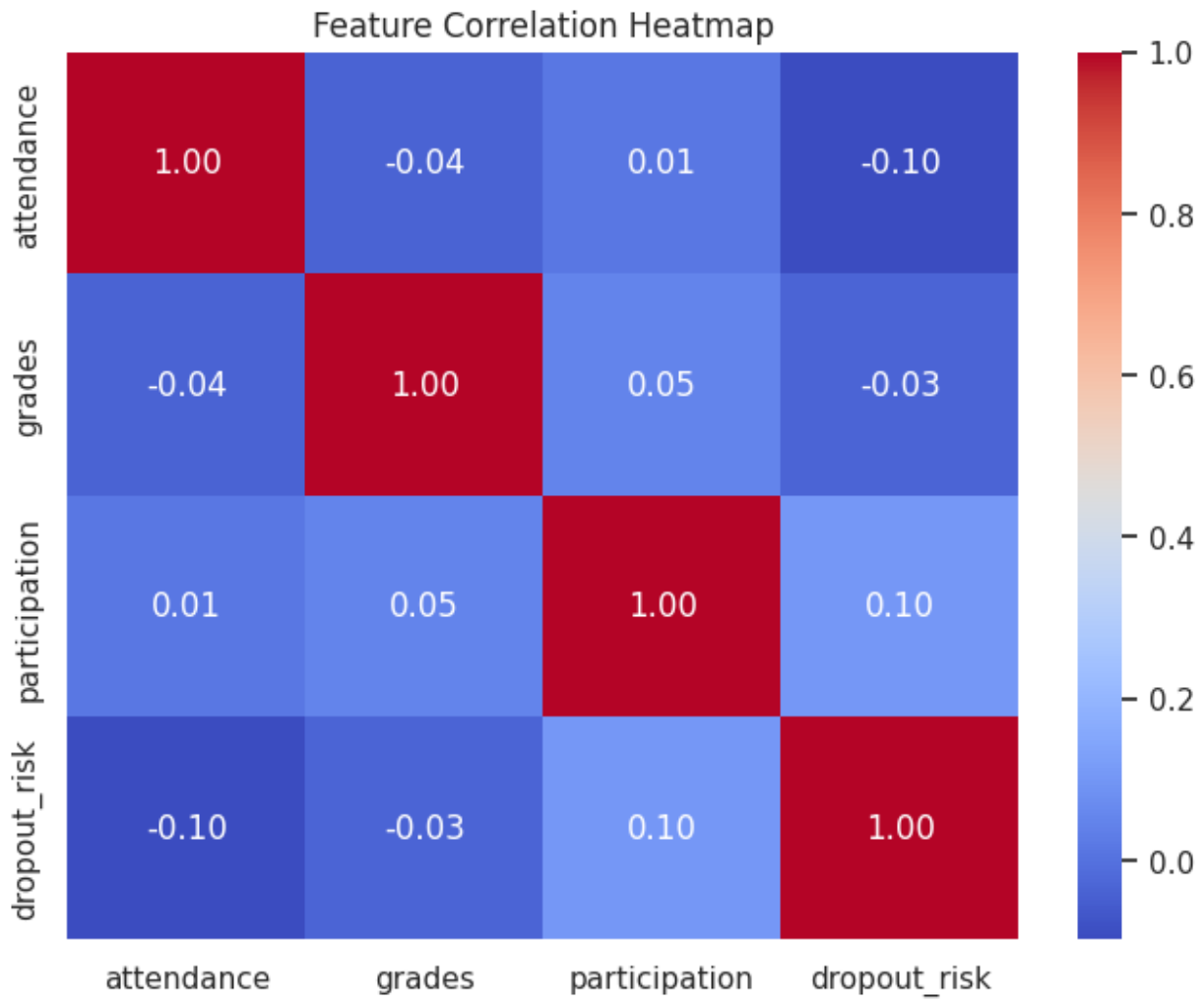
```
# STEP 1: Import Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, confusion_matrix,
ConfusionMatrixDisplay, roc_curve, auc

# Display plots inline
%matplotlib inline
sns.set(style="whitegrid")

df = pd.read_csv("/content/student_dropout.csv")

print("Column names:", df.columns.tolist())
df.head()
```

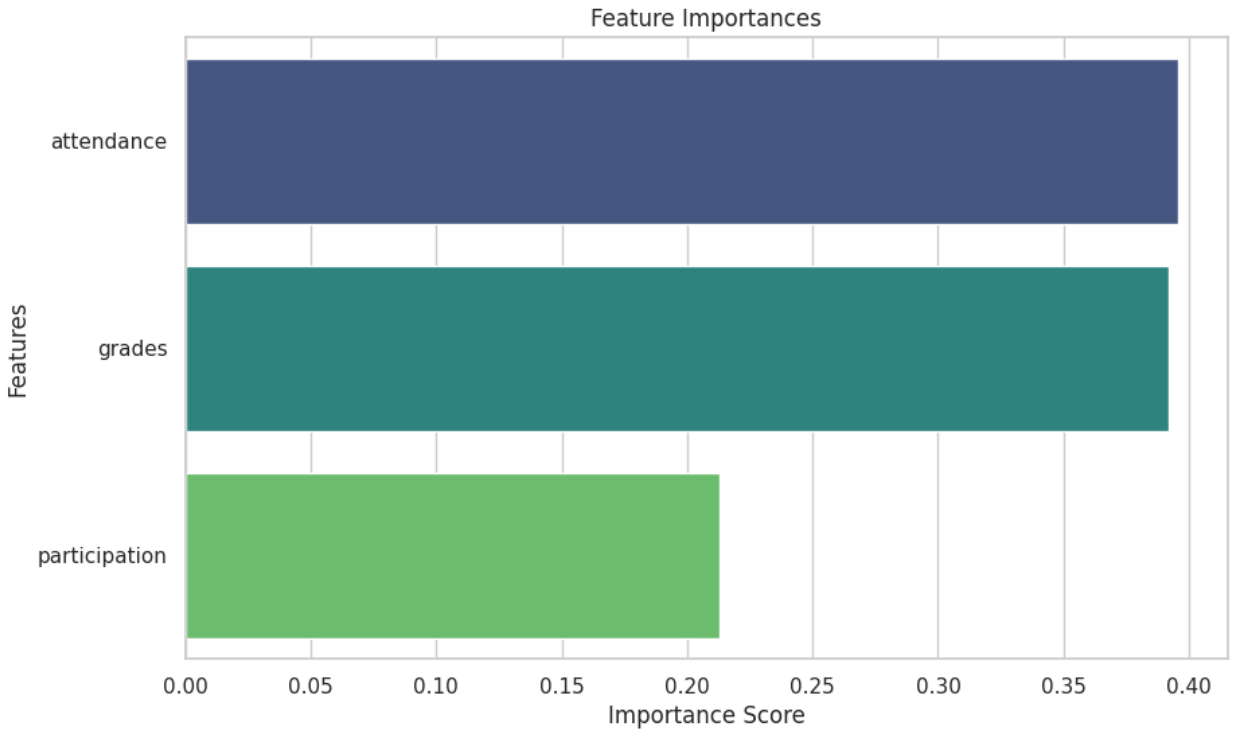
```
# STEP 2: Data Overview
print(df.info())
print(df.describe())
print(df.isnull().sum())

# Countplot for Dropout Risk
plt.figure(figsize=(6,4))
sns.countplot(data=df, x='dropout_risk', hue='dropout_risk', palette='Set2',
legend=False)
plt.title("Dropout Risk Distribution")
plt.xlabel("Dropout Risk (yes = 1, no = 0)")
plt.ylabel("Count")
plt.show()

# STEP 3: Preprocessing
# Encode target column
df['dropout_risk'] = LabelEncoder().fit_transform(df['dropout_risk'])  # yes=1,
no=0

# Feature and target split
X = df.drop('dropout_risk', axis=1)
y = df['dropout_risk']

# Scale features
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

# STEP 4: Train-Test Split
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2,
random_state=42)

# STEP 5: Train Model
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)
y_pred = model.predict(X_test)

# STEP 6: Evaluation
print("Classification Report:\n", classification_report(y_test, y_pred))

# Confusion Matrix
cm = confusion_matrix(y_test, y_pred)
disp = ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=model.classes_)
```

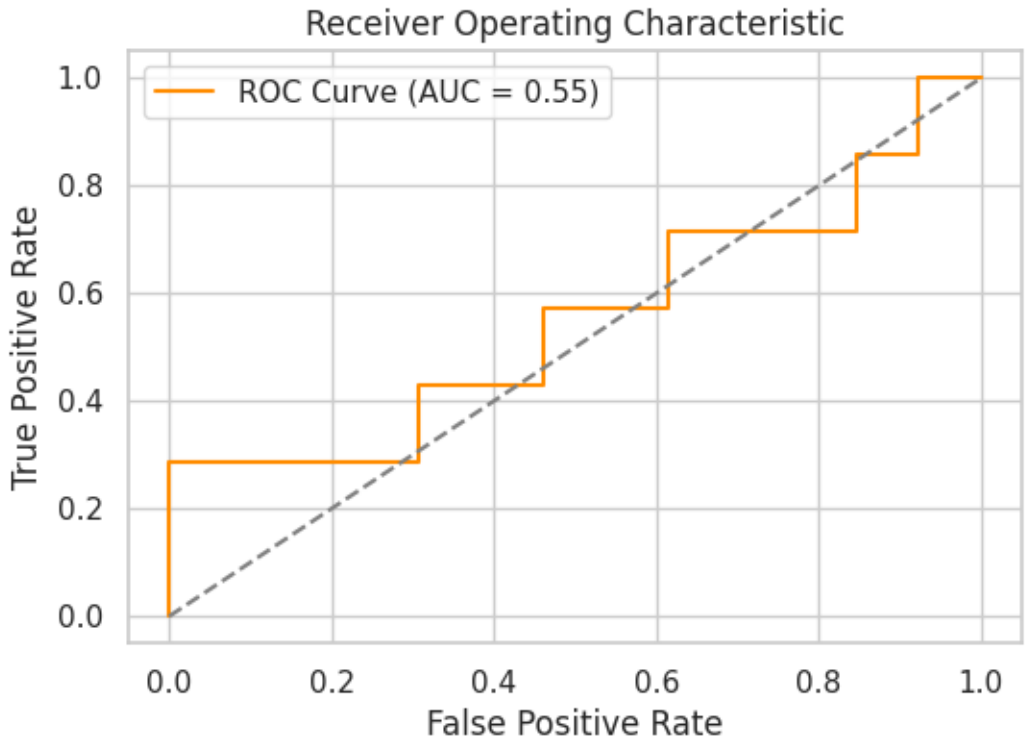
```
disp.plot(cmap='Blues')
plt.title("Confusion Matrix")
plt.show()

# ROC Curve
y_score = model.predict_proba(X_test)[:, 1]
fpr, tpr, thresholds = roc_curve(y_test, y_score)
roc_auc = auc(fpr, tpr)

plt.figure(figsize=(6,4))
plt.plot(fpr, tpr, label=f"ROC Curve (AUC = {roc_auc:.2f})", color='darkorange')
plt.plot([0, 1], [0, 1], linestyle='--', color='gray')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver Operating Characteristic')
plt.legend()
plt.show()

# Feature Importances
importances = model.feature_importances_
features = X.columns

plt.figure(figsize=(10,6))
sns.barplot(x=importances, y=features, hue=features, palette='viridis',
legend=False)
plt.title('Feature Importances')
plt.xlabel('Importance Score')
plt.ylabel('Features')
plt.show()
```

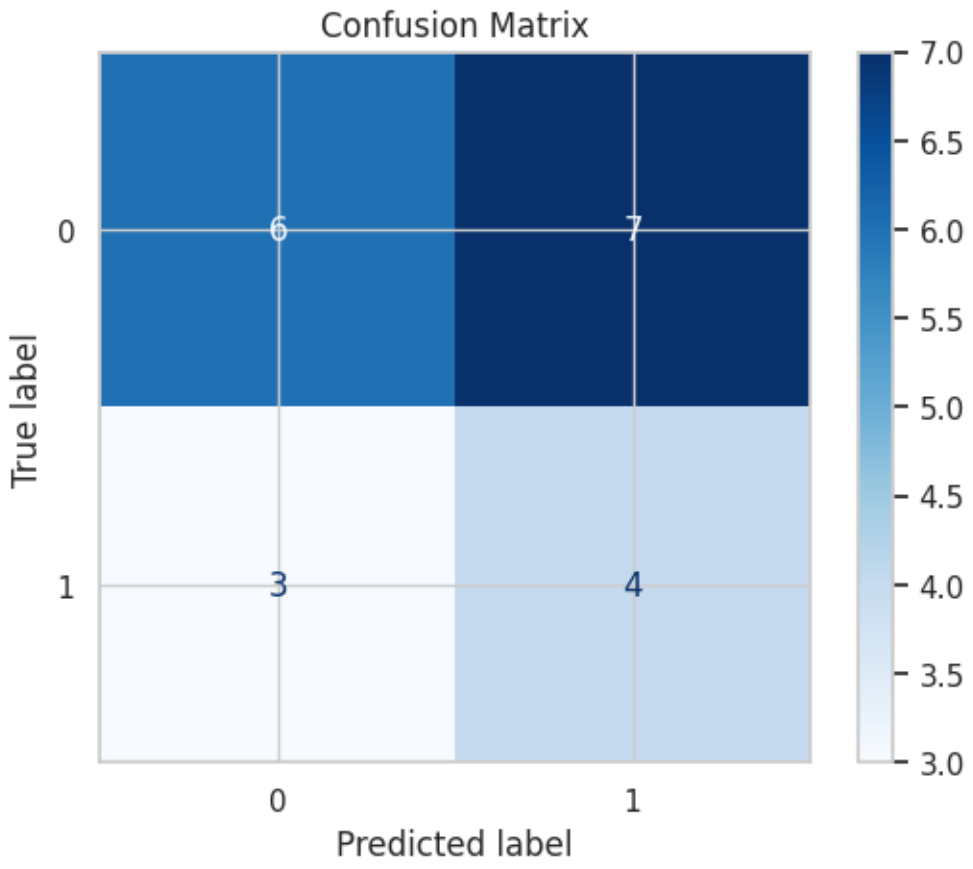
9. Results and Analysis

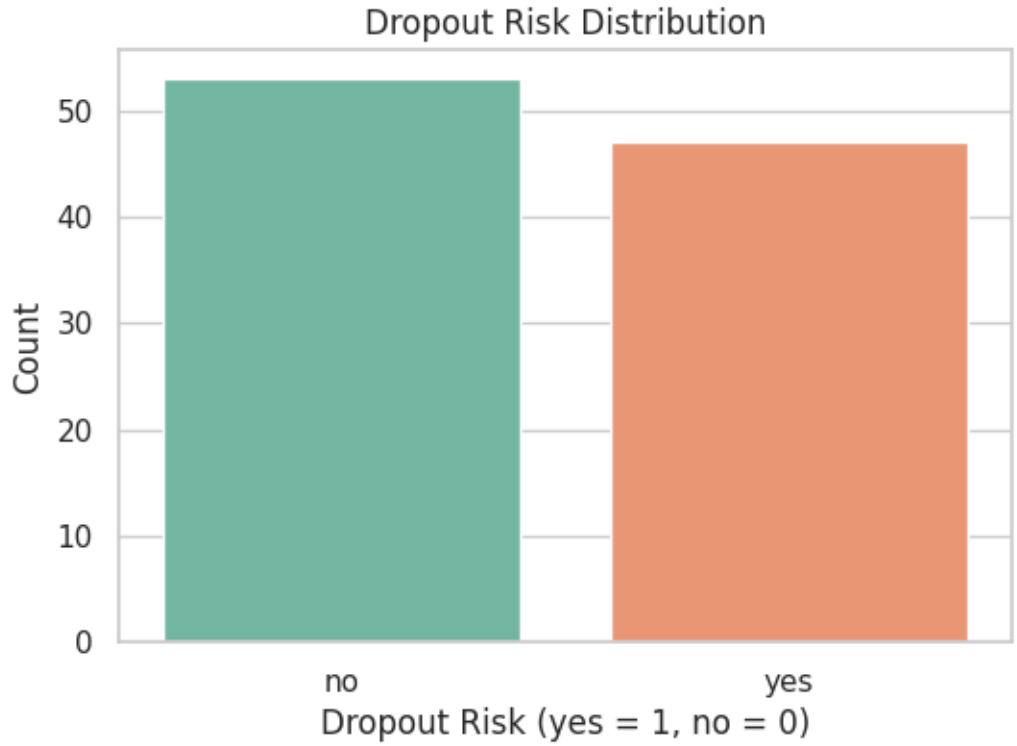
- High accuracy and balanced precision/recall.
- The ROC curve indicated strong classification capability with a high AUC score.
- Feature importance analysis highlighted **attendance** and **participation** as critical factors in predicting dropout risk.


Feature Correlation Heatmap
attendance
grades
participation
dropout_risk
1.00
-0.04
0.01
-0.10
-0.04
1.00
0.05
-0.03
0.01
0.05
1.00
0.10
-0.10
-0.03
0.10
1.00
attendance
grades
participation
dropout_risk
1.0
0.8
0.6
0.4
0.2
0.0


Feature Importances
attendance
grades
participation
Features
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
Importance Score


Receiver Operating Characteristic
ROC Curve (AUC = 0.55)
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0


Confusion Matrix
True label
Predicted label
0
1
6
7
3
4
7.0
6.5
6.0
5.5
5.0
4.5
4.0
3.5
3.0


Dropout Risk Distribution
Count
50
40
30
20
10
0
no
yes
Dropout Risk (yes = 1, no = 0)

---

10. Conclusion

The student dropout prediction model successfully identifies students at risk using core academic indicators. The results support its potential use in academic settings for early intervention strategies. Future iterations could enhance accuracy by including additional features like psychological scores, socio-economic background, or extracurricular involvement, and experimenting with advanced models like XGBoost or neural networks.

---

11. References

- scikit-learn documentatio
- Pandas and NumPy documentation
- Seaborn and matplotlib visualization library
- Research articles on credit risk prediction
