MICRO PROJECT

ADVANCED DATA MINING
(M24CS1E104C)

Early Predicting of Student's Performance in Higher Education

MICRO PROJECT REPORT

Submitted by

PAUL JOSE

MAC24CSCE07

To

The APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY in partial fulfillment for the award of the degree

of

MASTER OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
MAR ATHANASIUS COLLEGE OF ENGINEERING
(GOVT. AIDED & AUTONOMOUS)
KOTHAMANGALAM, KERALA-686 666

DECEMBER 2024

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
MAR ATHANASIUS COLLEGE OF ENGINEERING (GOVT.AIDED AUTONOMOUS)
KOTHAMANGALAM, KERALA-686 666

CERTIFICATE

This is to certify that the report entitled **"Early Predicting of Student's Performance in Higher Education"** submitted by **Mr. Paul Jose , Reg No: MAC24CSCE07** to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the Degree of Master of Technology in Computer Science & Engineering for the academic year 2024-2026 is a bonafied record of the micro project presented by them under our supervision and guidance. This report in any form has not been submitted to any other university or Institute for any purpose.

....

Prof. Sumi Joy
Staff in Charge

....

Prof. Joby George
Head of the Department

ACKNOWLEDGEMENT

First and foremost, I sincerely thank **God Almighty** for his grace for the successful and timely completion of the micro project. I express my sincere gratitude and thanks to the Principal **Dr. Bos Mathew Jos** and Head of the Department **Prof. Joby George** for providing the necessary facilities and their encouragement and support. I owe special thanks to the faculty in charge **Prof. Sumi Joy** for their corrections, suggestions and efforts to coordinate the micro project under a tight schedule. I also express my gratitude to the staff members in the Department of Computer Science and Engineering who have taken sincere efforts in helping me to completing this microproject. Finally, I would like to acknowledge the tremendous support given to me by our dear friends without whose support this work would have been all the more difficult to accomplish.

ABSTRACT

Predicting students' academic performance is a vital aspect of higher education, enabling institutions to implement early interventions and provide targeted support. This project leverages data mining techniques to develop a system for the early prediction of student performance using historical academic data. The dataset includes features like CGPA, SGPA, program of study, and demographic details.

The methodology involves clustering students based on their performance using K-Means and determining the optimal number of clusters via the Elbow Method. Machine learning classifiers, including Support Vector Machine (SVM), Decision Tree, Naïve Bayes, and K-Nearest Neighbors (KNN), were trained to predict the identified performance clusters. Data preprocessing, including handling missing values, encoding categorical features, and feature scaling, ensured the dataset was suitable for analysis.

The results show that SVM achieved the highest accuracy of 98.69%, with distinct performance clusters revealed through clustering. The system provides actionable insights for educators to identify at-risk students early and offers a scalable framework for academic performance monitoring.

Contents

List of Figures

CHAPTER 1

INTRODUCTION

The growing complexity and diversity of student populations in higher education have heightened the need for effective performance monitoring systems. Early prediction of students' academic outcomes is critical for enabling personalized learning, targeted interventions, and enhanced institutional support. Data mining techniques, combined with machine learning models, have emerged as transformative tools in this domain, allowing educators to uncover patterns in historical data and predict future trends with high accuracy.

This project focuses on developing a predictive framework for analyzing student performance using clustering and classification techniques. By leveraging historical academic data—such as cumulative GPA (CGPA), semester GPA (SGPA), and program of study—students are grouped into performance clusters through K-Means clustering. The optimal number of clusters is determined using the Elbow Method, providing a meaningful segmentation of the student population. Subsequently, machine learning classifiers, including Support Vector Machine (SVM), Decision Tree, Naïve Bayes, and K-Nearest Neighbors (KNN), are trained to predict performance clusters with high precision.

Preprocessing techniques, including handling missing values, encoding categorical variables, and feature scaling, ensure the dataset is ready for analysis. The models are evaluated using metrics such as accuracy, precision, and recall, with SVM achieving the highest accuracy of 98.69%. Visualizations like pair plots and cluster distribution charts provide further insights into performance groupings and academic patterns.

This report details the design, implementation, and evaluation of the predictive system for early student performance analysis. The following sections discuss the methodology, challenges encountered, and results, highlighting the potential of this approach to transform academic monitoring and support in higher education.

CHAPTER 2

SYSTEM DESIGN

The design of the student performance prediction system is structured into distinct modules to ensure modularity, clarity, and comprehensive functionality. The implementation focuses on data preprocessing, clustering, and classification, showcasing the system's capability to predict performance clusters accurately. Below is an overview of the key modules:

1. Data Preprocessing Module

Purpose: Prepare raw student performance data for clustering and classification.

Implementation:

- Missing values are handled by replacing numerical data with the median and categorical data with the mode.
- Categorical features such as gender and program code are encoded numerically.
- Numerical features (e.g., CGPA, SGPA) are normalized using `StandardScaler` to ensure uniform scaling.

Details:

- This module ensures that the dataset is free of inconsistencies, facilitating accurate and efficient machine learning processing.

2. Clustering Module

Purpose: Group students into performance clusters using K-Means clustering.

Implementation:

- The optimal number of clusters is determined using the Elbow Method.
- Students are segmented into clusters (e.g., High, Medium, Low performance) based on key features like CGPA and SGPA.

Details:

- Cluster separation provides insights into distinct performance groups, enabling targeted interventions.
- Visualization techniques, including pair plots and cluster distribution plots, validate the clustering results.

3. Classification Module

Purpose: Predict the performance cluster for new students based on their academic data.
Implementation:

- Machine learning classifiers, including SVM, Decision Tree, Naïve Bayes, and KNN, are trained on historical data.
- Each model is evaluated using accuracy, precision, recall, and F1-score to identify the best-performing classifier.

Details:

- SVM achieved the highest accuracy (98.69%), demonstrating its effectiveness for this task.
- The trained models enable real-time predictions for new student entries.

4. Data Visualization Module

Purpose: Provide visual insights into clustering and classification outcomes.
Implementation:

- Pair plots highlight relationships between features like CGPA and SGPA across clusters.
- Count plots show the distribution of students in each cluster.

Details:

- Visualizations make it easier to interpret clustering results and validate the effectiveness of the predictive system.

5. System Execution Module

Purpose: Coordinate the execution of data preprocessing, clustering, and classification tasks.
Implementation:

- A script integrates all modules into a seamless workflow:
  - Preprocessing raw data.
  - Clustering students into performance groups.
  - Training and testing classifiers on labeled data.
  - Visualizing the results for analysis.

Details:

- Execution demonstrates how the system predicts student performance clusters and generates actionable insights for educators.

This modular design ensures a systematic and scalable implementation of the performance prediction system. By leveraging clustering and classification techniques, the project highlights the potential for data-driven approaches to enhance academic performance monitoring and student success.

CHAPTER 3

PROGRAM

The implementation of the student performance prediction system is based on Python, utilizing libraries for data preprocessing, clustering, classification, and visualization. The following code snippets illustrate the primary components of the system.

1. Data Preprocessing and Feature Engineering

Listing 3.1: Data Preprocessing and Feature Engineering

```
import pandas as pd
from sklearn.preprocessing import LabelEncoder, StandardScaler

# Load the dataset
file_path = "Dataset␣on␣the␣academic␣performance␣of␣students.xlsx"
data = pd.ExcelFile(file_path)
sheet1_data = data.parse("Sheet1")

# Data cleaning and preprocessing
cleaned_data = sheet1_data.drop(columns=["Unnamed:␣10", "Unnamed:␣11", "Unnamed:␣12"], errors="
    ignore")

# Handle missing values
for column in cleaned_data.columns:
    if cleaned_data[column].dtype in ["float64", "int64"]:
        cleaned_data[column].fillna(cleaned_data[column].median(), inplace=True)
    elif cleaned_data[column].dtype == "object":
        cleaned_data[column].fillna(cleaned_data[column].mode()[0], inplace=True)

# Encode categorical features
label_encoder = LabelEncoder()
cleaned_data["Gender"] = label_encoder.fit_transform(cleaned_data["Gender"])

# Feature scaling
scaler = StandardScaler()
numerical_features = cleaned_data[["CGPA", "CGPA100", "CGPA200", "CGPA300", "CGPA400", "SGPA"]]
scaled_features = scaler.fit_transform(numerical_features)
```

2. Clustering with K-Means

Listing 3.2: Clustering Using K-Means

```
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt

# Determine optimal clusters using Elbow Method
wcss = []
for i in range(1, 11):
    kmeans = KMeans(n_clusters=i, random_state=42)
    kmeans.fit(scaled_features)
    wcss.append(kmeans.inertia_)

# Plot Elbow Method
plt.figure(figsize=(8, 5))
plt.plot(range(1, 11), wcss, marker='o', linestyle='--')
plt.title("Elbow␣Method␣for␣Optimal␣Clusters")
plt.xlabel("Number␣of␣Clusters")
plt.ylabel("WCSS")
plt.show()

# Apply K-Means with optimal clusters (e.g., k=3)
optimal_k = 3
kmeans = KMeans(n_clusters=optimal_k, random_state=42)
cleaned_data["Cluster"] = kmeans.fit_predict(scaled_features)
```

3. Classification Models

Listing 3.3: Training and Evaluating Machine Learning Models

```
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
from sklearn.tree import DecisionTreeClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score, classification_report

# Split the data
X = cleaned_data.drop(columns=["ID␣No", "Cluster"])
y = cleaned_data["Cluster"]
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

# Train and evaluate SVM
svm_model = SVC(kernel="linear", random_state=42)
svm_model.fit(X_train, y_train)
svm_predictions = svm_model.predict(X_test)
print("SVM␣Accuracy:", accuracy_score(y_test, svm_predictions))
print("SVM␣Report:\n", classification_report(y_test, svm_predictions))

# Train and evaluate Decision Tree
```

```
dt_model = DecisionTreeClassifier(random_state=42)
dt_model.fit(X_train, y_train)
dt_predictions = dt_model.predict(X_test)
print("Decision␣Tree␣Accuracy:", accuracy_score(y_test, dt_predictions))
print("Decision␣Tree␣Report:\n", classification_report(y_test, dt_predictions))

# Train and evaluate Na ve Bayes
nb_model = GaussianNB()
nb_model.fit(X_train, y_train)
nb_predictions = nb_model.predict(X_test)
print("Na ve␣Bayes␣Accuracy:", accuracy_score(y_test, nb_predictions))
print("Na ve␣Bayes␣Report:\n", classification_report(y_test, nb_predictions))

# Train and evaluate KNN
knn_model = KNeighborsClassifier(n_neighbors=5)
knn_model.fit(X_train, y_train)
knn_predictions = knn_model.predict(X_test)
print("KNN␣Accuracy:", accuracy_score(y_test, knn_predictions))
print("KNN␣Report:\n", classification_report(y_test, knn_predictions))
```

4. Visualization

Listing 3.4: Data Visualization with Seaborn and Matplotlib

```
import seaborn as sns

# Pair plot to visualize clusters
sns.pairplot(cleaned_data, hue="Cluster", vars=["CGPA", "CGPA100", "CGPA200"])
plt.show()

# Distribution of clusters
sns.countplot(x="Cluster", data=cleaned_data)
plt.title("Cluster␣Distribution")
plt.show()
```

The code modules provide a comprehensive implementation of the student performance prediction system. The workflow ensures modularity and extensibility, making it suitable for further development and application in educational data mining tasks.

CHAPTER 4

RESULT

The implementation of the student performance prediction system demonstrates the practical application of data mining and machine learning techniques in higher education analytics. The system effectively grouped students into performance clusters using the K-Means clustering algorithm, with the Elbow Method determining the optimal number of clusters. Classification models, including SVM, Decision Tree, Naïve Bayes, and KNN, were trained to predict these clusters based on academic performance data.

The results highlight the system's accuracy and effectiveness:

- Clustering segmented students into meaningful groups, such as high, medium, and low performers.
- SVM achieved the highest classification accuracy of 98.69%, showcasing its robustness in handling complex patterns.
- Visualizations, including pair plots and count plots, provided actionable insights into academic trends and student distribution across clusters.

This project emphasizes the potential of data-driven approaches in education. By identifying at-risk students early, institutions can implement targeted interventions, thereby improving academic outcomes and optimizing resource allocation. The system's modular design and scalability make it well-suited for real-world applications, paving the way for enhanced performance monitoring in higher education.

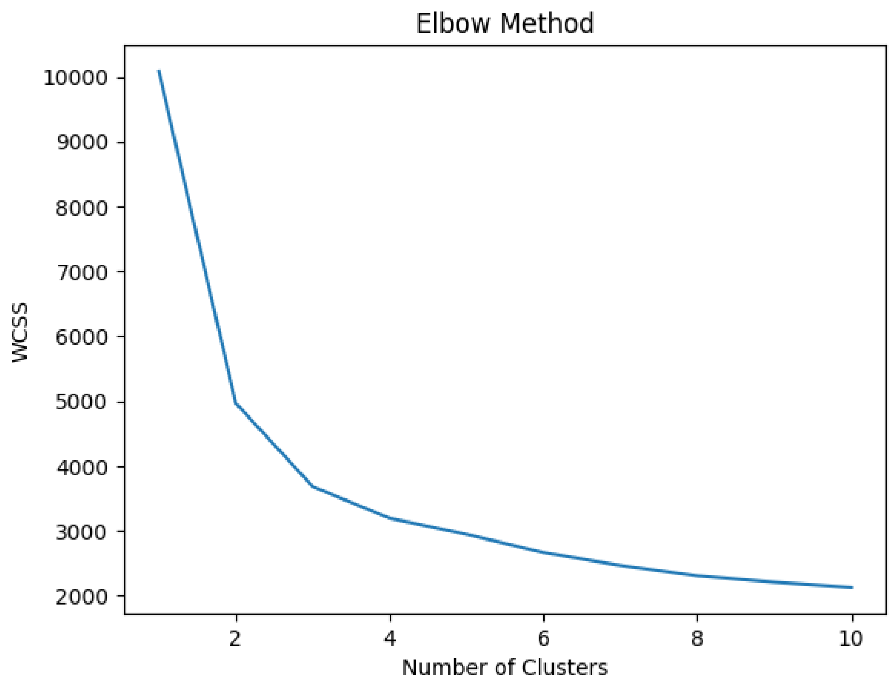

Figure 4.1: Elbow Method to Determine Optimal Clusters

```
   Gender  YoG      CGPA   CGPA100   CGPA200   CGPA300   CGPA400      SGPA  \
0       0    0  3.227513  2.875000  3.475000  2.615385  2.898305  3.125000
1       0    0  3.576271  3.250000  4.261905  3.368421  3.469388  3.020833
2       1    0  2.211454  1.777778  1.979167  1.489583  2.511111  2.187500
3       1    0  2.702970  2.673913  2.442308  2.000000  2.348315  3.194444
4       0    0  3.881657  3.608696  3.687500  3.625000  4.581395  4.236111

   Prog Code_BLD  Prog Code_CEN  ...  Prog Code_EEE  Prog Code_ICE  \
0          False          False  ...          False           True
1          False          False  ...          False          False
2          False          False  ...          False          False
3          False          False  ...          False          False
4          False          False  ...          False          False

   Prog Code_MAT  Prog Code_MCB  Prog Code_MCE  Prog Code_MIS  Prog Code_PET  \
0          False          False          False          False          False
1          False          False          False          False          False
2          False          False          False          False          False
3          False          False          False          False          False
4          False          False          False          False          False

   Prog Code_PHYE  Prog Code_PHYG  Prog Code_PHYR
0           False           False           False
1           False           False           False
2           False           False           False
3           False           False           False
4           False           False           False

[5 rows x 24 columns]
0    2
1    2
2    1
3    1
4    0
Name: Cluster, dtype: int32
```

Figure 4.2: Dataset after Preprocessing

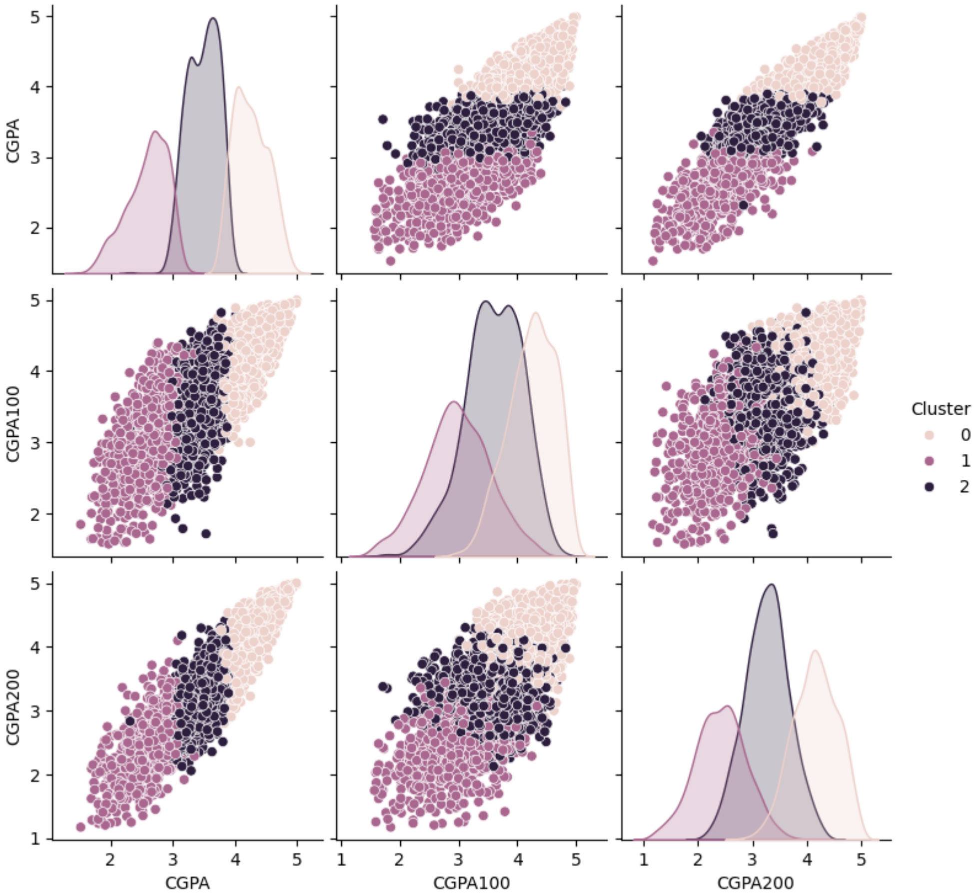

Figure 4.3: Pair Plots Showing Cluster Relationships

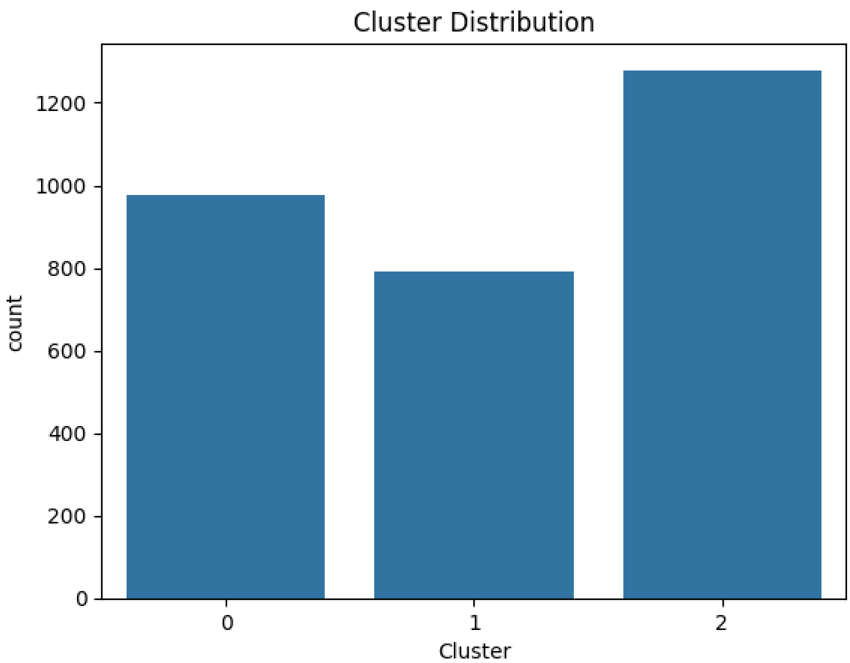

Figure 4.4: Cluster Distribution Count Plot

CHAPTER 5

PERFORMANCE ANALYSIS

The student performance prediction system was evaluated on clustering and classification tasks, demonstrating effective segmentation and high prediction accuracy.

Clustering Performance

K-Means clustering grouped students into three distinct performance clusters: high, medium, and low performers. The Elbow Method identified the optimal number of clusters, ensuring meaningful segmentation. Pair plots validated the separation between clusters, highlighting clear boundaries in the data.

Classification Performance

Four classifiers were trained to predict performance clusters:

- **SVM:** Achieved the highest accuracy of 98.69%, with consistent performance across all clusters.
- **Decision Tree:** Delivered an accuracy of 94.75% and provided interpretable results.
- **Naïve Bayes:** Achieved 93.65% accuracy but showed reduced precision for certain clusters.
- **KNN:** Reached 90.81% accuracy, indicating room for optimization.

Insights and Observations

Visualizations such as pair plots, count plots, and Elbow plots highlighted the system's ability to segment students effectively and provide actionable insights. The results demonstrate the system's scalability and practical application for early intervention in higher education.

```
SVM Accuracy: 0.986870897155361
SVM Report:
              precision    recall  f1-score   support

           0       0.99      0.99      0.99       296
           1       1.00      0.99      0.99       244
           2       0.98      0.99      0.98       374

    accuracy                           0.99       914
   macro avg       0.99      0.99      0.99       914
weighted avg       0.99      0.99      0.99       914

Decision Tree Accuracy: 0.9474835886214442
Decision Tree Report:
              precision    recall  f1-score   support

           0       0.96      0.95      0.96       296
           1       0.96      0.95      0.95       244
           2       0.93      0.94      0.94       374

    accuracy                           0.95       914
   macro avg       0.95      0.95      0.95       914
weighted avg       0.95      0.95      0.95       914

Naïve Bayes Accuracy: 0.936542669584245
Naïve Bayes Report:
              precision    recall  f1-score   support

           0       0.95      0.95      0.95       296
           1       0.91      0.98      0.94       244
           2       0.94      0.90      0.92       374

    accuracy                           0.94       914
   macro avg       0.94      0.94      0.94       914
weighted avg       0.94      0.94      0.94       914

KNN Accuracy: 0.9080962800875274
KNN Report:
              precision    recall  f1-score   support

           0       0.92      0.91      0.92       296
           1       0.94      0.91      0.93       244
           2       0.88      0.90      0.89       374

    accuracy                           0.91       914
   macro avg       0.91      0.91      0.91       914
weighted avg       0.91      0.91      0.91       914
```

Figure 5.1: Performance Evaluation of Classification Models

CHAPTER 6

CONCLUSION

The implementation of the student performance prediction system demonstrates the effective application of data mining and machine learning techniques in academic performance monitoring. By utilizing clustering with K-Means and classification models such as SVM, Decision Tree, Naïve Bayes, and KNN, the system accurately predicts performance clusters based on historical academic data.

The results, with SVM achieving the highest accuracy of 98.69%, validate the robustness of the proposed methodology. Clustering provided meaningful segmentation of students into performance groups, while visualizations like pair plots and cluster distribution charts offered valuable insights into academic trends. These tools highlight at-risk students early, enabling institutions to design targeted interventions and personalized support strategies.

This project underscores the potential of data-driven approaches in education, facilitating informed decision-making and proactive measures to enhance student outcomes. The system's scalability and accuracy make it a practical solution for real-world applications, emphasizing its significance in the evolving landscape of higher education analytics.