ANALYSIS OF COURSE USAGE WITH HISTORY DATASET AND RATING PREDICTION

A PROJECT REPORT

Submitted by

MUGASH PRIYAN U [Reg No: RA2212704010028]

PRASANTH B [Reg No: RA2212704010033]

ROHITH S [Reg No: RA2212704010017]

Under the Guidance of

Dr. Shanthini A

(Associate Professor, Department of Data Science and Business Systems)

In partial fulfillment of the Requirements for the Degree of

M.TECH INTEGRATED COMPUTER SCIENCE with Specialization in DATA SCIENCE

DEPARTMENT OF DATA SCIENCE AND BUSINESS SYSTEMS FACULTY OF ENGINEERING AND TECHNOLOGY SRM INSTITUTE OF SCIENCE AND TECHNOLOGY KATTANKULATHUR - 603203

NOVEMBER 2024

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR-603203

BONAFIDE CERTIFICATE

Certified that this project report titled **"ANALYSIS OF COURSE USAGE WITH HISTORY DATASET AND RATING PREDICTION"** is the bonafide work of **MUGASH PRIYAN U [Reg No: RA2212704010028], PRASANTH B [Reg No: RA2212704010033], ROHITH S [Reg No: RA2212704010017]** who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion for this or any other candidate.

Dr. SHANTHINI A
Associate Professor
Dept. of DSBS

Dr. KAVITHA V
HEAD OF THE DEPARTMENT
Dept. of DSBS

Signature of Internal Examiner

Signature of External Examiner

ABSTRACT

In the age of digital learning, online courses have become a cornerstone of education. With a plethora of platforms and courses available, understanding the dynamics of online course usage can provide valuable insights for educators, students, and platform developers alike. This notebook delves into the **'Online Courses Usage and History' dataset** to uncover trends, correlations, and potential predictors of course success.

Our analysis aims to identify the factors that influence user engagement and satisfaction, utilizing comprehensive exploratory data analysis (EDA) techniques and advanced machine learning algorithms for rating prediction. The study begins by examining the temporal and behavioral patterns of course usage, exploring how user activity varies across different times of the day, days of the week, and course categories. Through feature engineering and data preprocessing, we aim to create robust features that capture key user behaviors, including course completion rates, time spent on content, and the impact of frequent interactions on user satisfaction.

We then employ predictive modeling approaches to estimate course ratings, leveraging algorithms such as linear regression, random forest, gradient boosting, and neural networks. The models are evaluated based on metrics like mean squared error (MSE), R-squared, and root mean squared error (RMSE) to ensure reliability and accuracy in our predictions. Additionally, the study emphasizes the interpretability of these models, exploring feature importance to offer actionable recommendations for course design and content curation.

ACKNOWLEDGEMENT

The successful completion and outcome of this project required guidance and assistance from various sources, and we feel deeply grateful for all the support we received throughout our journey. Whatever we have accomplished is largely thanks to this invaluable guidance and assistance, and we wish to express our heartfelt appreciation.

We sincerely thank the Head of the Department, Department of Data Science and Business Systems, **Dr. KAVITHA V**, for the extensive support, infrastructure, and guidance provided, enabling us to complete our project successfully.

We are profoundly grateful to our project guide, **Dr. SHANTHINI A**, for her unwavering interest in our work and for guiding us through every stage, offering all the essential information needed to develop a successful system.

We extend our sincere appreciation to all the teaching staff of the Department of Data Science and Business Systems for their constant encouragement, support, and guidance, which played a crucial role in our project's successful completion. Additionally, we express our gratitude to the non-teaching staff of the Department of Information Technology for their timely and essential assistance.

MUGASH PRIYAN U [Reg No: RA2212704010028]
PRASANTH B [Reg No: RA2212704010033]
ROHITH S [Reg No: RA2212704010017]

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1 INTRODUCTION

1.1 GENERAL

The ‘Online Courses Usage and History’ database contains a wealth of user interaction data about a higher education e-learning portal. It includes transcripts of how users interacted with these online courses, including selection rates, time spent in every module, number of times it was logged in to, number of completions and score assigned to the users among others. The dataset helps in analyzing user trends and preferences in addition to being applicable for educational datamining and forecasting.

Among the attributes captured in the dataset include:

- User Characteristics: Indicate users’ age, highest degree of education attained together with their location, useful for analyzing different patterns in learning.
- Courses Information: Information on courses offered, their lengths and levels of difficulty, important in determining how the content of courses affect the users’ interaction.
- User Activity Indicators: Measures such as the number of log ins and time in an average session as well as total units of content consumed, important for analyzing engagement patterns.
- Results and Recommendations: Recommendations and ratings left by the users for the courses aimed towards developing a model that can predict the likelihood of a course being successful and the level of satisfaction of its participants.

1.2 PURPOSE

The primary objective for undertaking the study of the ‘Online Courses Usage and History’ dataset is to analyse certain patterns, factors and the motivation behind the users’ interaction with the site and the course – as well as the rating given to it. This will help the platform if they know why the users respond the way they do and thus provide suggestions to the educators, the course designers/developers and the platform ideally. This analysis seeks to ascertain the major tenets of moderators of course completion and satisfaction in order to help the stakeholders increase content delivery, user retention and make the learning experience more unique.

Moreover, predicting how users will rate a course lecture and user behavior based on course attributes will enhance the efficiency of the recommendation systems and increase values of targeted marketing strategies for getting users more engaged. The application of these suggestions can help online education platforms to design their courses more efficiently, time their updates or new features appropriately and certain the offered content more closely to what the users need so as to enhance the quality and effectiveness of the virtual educational environments.

1.3 SCOPE

The scope of this project encompasses a comprehensive analysis of the 'Online Courses Usage and History' dataset to extract meaningful insights into user behavior and engagement patterns. The study will cover a range of exploratory data analysis techniques to identify trends, correlations, and anomalies within the dataset. This includes examining the impact of various factors, such as course duration, content type, and user demographics, on course completion rates and overall satisfaction.

Additionally, the project extends to the development and evaluation of machine learning models to predict course ratings based on user interaction data. By incorporating advanced algorithms, we aim to build accurate and interpretable models that can inform future strategies for course improvement and recommendation systems. The analysis will provide practical recommendations for optimizing course structures and engagement strategies, making it valuable for both academic researchers and educational platforms looking to enhance their services.

1.4 MOTIVATION

This analysis is inspired by the great importance that online education has in today's learning environment. Technology has enhanced the formal learning process by eliminating challenges of space and socioeconomic status through the provision of quality education. However, with an abundance of courses come difficulties for the learners and teachers when it comes to searching and providing optimal learning experiences. Conversely, examining the determinants of course engagement, course ratings as well as user satisfaction may assist in apportioning attention towards improving the quality of online course offerings.

This analysis is animated by the objective of using data to make online education impactful, cater to individual learner needs, and enhance the usability of the system. We wish to determine what makes a course work through understanding user interaction patterns as well as using machine learning models. The outcomes of this research can help not only course developers in creating appealing content, but also assist platform administrators to improve the platforms and system in the most appropriate way.

PROBLEM STATEMENT

Online learning platforms have suffered from this problem of low course completion, poor student satisfaction, and low engagement rates. Many learners disengage or dropout-conditions that may indicate a gap in understanding the factors at play. Using the 'Online Courses Usage and History' dataset, which includes user demographics as well as course information together with engagement metrics, this project seeks to find key drivers leading to successful learning experiences. We will discover insights and develop predictive models that we will eventually recommend changes in the design and delivery of these courses to enhance student engagement and satisfaction.

1.5 INTRODUCTION TO NAÏVE BAYES

Naive Bayes is a probabilistic machine learning algorithm based on Bayes' theorem, commonly used for classification tasks. It's known for its simplicity and effectiveness, especially with large datasets and high-dimensional data. The "naive" part of the algorithm's name comes from its assumption that all features (predictors) are independent of each other, which often doesn't hold in real-world scenarios but still yields highly accurate results in practice. Naive Bayes is particularly popular for text classification, spam filtering, and sentiment analysis, but its straightforward application can be useful for many types of categorical or numerical data.

The Naïve Bayesian is based on the conditional probability (given a set of features, the probability of a certain results occurrence):

$$P(y|X) = \frac{P(X|y)P(y)}{P(X)}, \text{ Where } X = (x_1, x_2, x_3, x_4, \ldots\ldots\ldots\ldots x_n)$$

$$P(y|x_1, \ldots, x_n) = \frac{P(x_1|y)P(x_2|y)\ldots P(x_n|y)P(y)}{P(x_1)P(x_2)\ldots P(x_n)}$$

$$P(y|x_1, \ldots, x_n) = \frac{P(y)\prod_{i=1}^{n} P(x_i|y)}{P(x_1)P(x_2)\ldots P(x_n)}$$

$$P(y|x_1, \ldots, x_n) \propto P(y)\prod_{i=1}^{n} P(x_i|y)$$

$$y = argmax_y P(y)\prod_{i=1}^{n} P(x_i|y)$$

Fig. 1.1 : Naive Bayes Formula

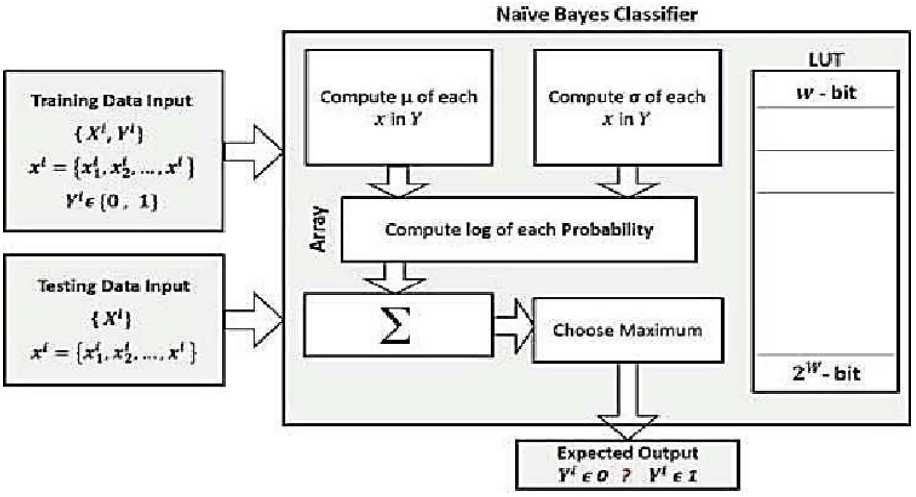


Fig 1.2: Naïve Bayes Architecture

Naive Bayes classifiers can handle a subjective number of autonomous factors whether nonstop or all out. Given a lot of factors, X = {x1,x2,x...,xd}, we need to build the posterior probability for the occasion Cj among a lot of conceivable results C = {c1,c2,c...,cd}. In an increasingly well-known language, X is the indicators and C is the arrangement of absolute dimensions present in the needy variable.

Utilizing Bayes' standard:

$$p(C_j \mid x_1, x_2, \ldots, x_d) \propto p(x_1, x_2, \ldots, x_d \mid C_j) p(C_j)$$

where p(Cj | x1,x2,x...,xd) is the posterior probability of class enrollment, i.e., the probability that X has a place with Cj. Since Naive Bayes expect that the restrictive probabilities of the autonomous factors are factually free we can decay the probability to a result of terms:

$$p(X \mid C_j) \propto \prod_{k=1}^{d} p(x_k \mid C_j)$$

and revise the posterior as:

$$p(C_j \mid X) \propto p(C_j) \prod_{k=1}^{d} p(x_k \mid C_j)$$

Utilizing Bayes' standard above, we name another case X with a class level Cj that accomplishes the most astounding posterior probability.

In spite of the fact that the presumption that the indicator (autonomous) factors are free isn't constantly exact, it simplifies the grouping task significantly, since it permits the class restrictive densities p(xk | Cj) to be determined independently for every factor, i.e., it lessens a multidimensional undertaking to various one-dimensional ones. Essentially, Naive Bayes lessens a high-dimensional density estimation undertaking to a one-dimensional part density estimation. Besides, the suspicion does not appear to extraordinarily influence the posterior probabilities, particularly in areas close choice limits, hence, leaving the grouping task unaffected.

Naive Bayes can be displayed in a few diverse ways including ordinary, lognormal, gamma and Poisson density functions:

$$p(x_k \mid C_j) = \begin{cases} \dfrac{1}{\sigma_{kj}\sqrt{2\pi}} \exp\left(\dfrac{-(x-\mu_{kj})^2}{2\sigma_{kj}}\right), & -\infty < x < \infty, -\infty < \mu_{kj} <, \sigma_{kj} > 0 \quad \text{Normal} \\ \mu_{kj} : \text{mean}, \sigma_{kj} : \text{standard deviation} \\ \dfrac{1}{x\sigma_{kj}(2\pi)^{1/2}} \exp\left\{\dfrac{-[\log(x/m_{kj})]^2}{2\sigma_{kj}^2}\right\}, & 0 < x < \infty, m_{kj} > 0, \sigma_{kj} > 0 \quad \text{Lognormal} \\ m_{kj} : \text{scale parameter}, \sigma_{kj} : \text{shape parameter} \\ \dfrac{\left(\dfrac{x}{b_{kj}}\right)^{c_{kj}-1}}{b_{kj}\Gamma(c_{kj})} \exp\left(\dfrac{-x}{b_{kj}}\right), & 0 \le x < \infty, b_{kj} > 0, c_{kj} > 0 \quad \text{Gamma} \\ b_{kj} : \text{scale parameter}, c_{kj} : \text{shape parameter} \\ \dfrac{\lambda_{kj}\exp(-\lambda_{kj})}{x!}, & 0 \le x < \infty, \lambda_{kj} > 0, x = 0,1,2,\ldots \quad \text{Poisson} \\ \lambda_{kj} : \text{mean} \end{cases}$$

Fig 1.3: Poisson density functions

Applying Naive Bayes to this dataset could help classify time periods into "peak" or "off-peak" categories based on historical data patterns. For example, if past data indicates specific months or time blocks as high-activity periods, Naive Bayes can help classify new data points by calculating the probability that they belong to the peak or off-peak category. This probability-based approach is efficient and interpretable, making it suitable for initial pattern recognition and classification tasks.

1.5.1 ADVANTAGES OF NAIVE BAYES IN THIS CONTEXT

1. Efficient Handling of Big Datasets : The 'Online Courses Usage and History' dataset will be pretty long since it is enclosing most of the user interaction and course attribute data. Naive Bayes is, therefore efficient and can process large volumes of data with a high speed and hence suitable for rapid analysis and predictions.

2. Ability for handling categorical features: There are hundreds of thousands of categorical variables used here, such as course categories and user demographics to engagement behaviors. Naive Bayes is specifically good for categorical data; hence it may very well do the trick in classification tasks here.

It simplifies assumptions about complex data. Though Naive Bayes makes an assumption of feature independence, it can output surprisingly good performance for complex datasets. This is very useful for establishing patterns in user behavior and engagement without requiring too much engineering around features.

4. Ease of Implementation and Interpretation: Naive Bayes is relatively easy to implement and interpret. Such ease is a big boon while presenting the results to stakeholders who must understand how different variables affect course completion and user satisfaction.

5. Robustness to Irrelevant Features: Given the nature of features, some being irrelevant to the goal, Naive Bayes does not lose much efficiency. This is because it can handle irrelevant features more robustly without extensive preprocessing of data.

Practical Application of Naive Bayes in This Analysis

In this analysis, Naive Bayes can be used in the following ways:

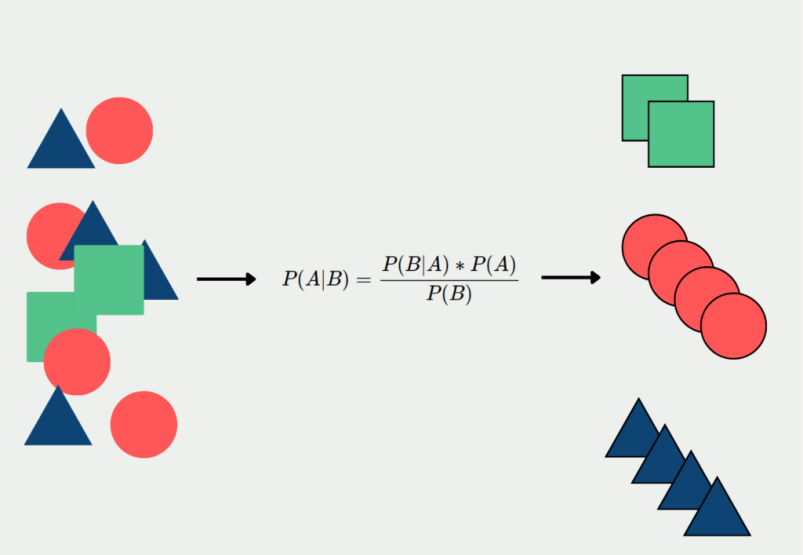
- Classify user course ratings based on their previous course usage history for predicting future course ratings.

- Probability distribution of ratings, either low, medium, high for a user based on his historic behavior as predictive of the ratings that a new user may give for a course .

- Categorization of courses - highly rated, poorly rated- groups created based on historical data to know which courses are mostly used, where improvement is needed.

- Recommendations enabled with personalization: when the model could predict which courses a user is likely to rate highly based on similar patterns observed in other users who have a comparable history.

  This helps classify improvement in user experience and provides course makers and platforms guidelines on the content to tailor to their best fit among the users.



1.5.2 APPLICATIONS OF NAÏVE BAYES ALGORITHM

The Online Courses Usage and
History dataset is easily applicable for the accomplishment of different kinds of analytical a nd predictive tasks by employing the Naive Bayes classifier. Here is how you may apply it:

1. Course Completion Prediction Naive Bayes can predict whether a user is likely to complete a course or dropout based on attributes, like user demographics, session duration, or frequency of access in courses. Such prediction
can actually track those who need attention and perhaps be appealed for extra support or incentives to keep them interested in the course.

2. Estimation of Student Satisfaction: Naive Bayes can predict user satisfaction levels based on the course attributes and engagement metrics. A few inputs such as time spent on content, difficulty levels of courses, and feedback given by users can be used to estimate whether a student would give high rating for that particular course.

3. Content Personalization: The algorithm can categorize its users
into various segments from the basis of their preferences as well as their engagement patterns. This classification will help in sending them courses
that best fit their learning style and needs, thus enhancing chances of completion and satisfaction.

4. Dropout Patterns: Naïve Bayes could help identify patterns of users who tend to drop out early, thus helping analyze features like initial engagement level and
interaction frequency. It could be flagged that which users require attention so interventions could be targeted at those people and retract them.

Actually, the Naive Bayes classifier is pretty effective for this
dataset since it can take care of categorical and continuous
features, thereby providing fast and interpretable predictions that can guide one in making some actionable strategies to improve the online learning experiences.

1.5.3 CHALLENGES OF NAÏVE BAYES

1. Assumption of Feature Independence: Naive Bayes assumes that all features are independent of each other, which might not hold true in your dataset. In course usage history, variables such as course topic, completion rate, and time spent may be interdependent, affecting the model's performance.
2. Handling Categorical Features: If your dataset contains complex categorical features (like course categories or user demographics) that aren't easily convertible to numeric values, the model may oversimplify the relationships, impacting accuracy.
3. Imbalanced Data: If the distribution of ratings (e.g., low, medium, high) is skewed, Naive Bayes might perform poorly on the minority class. For example, if most users give high ratings and very few give low ratings, the model might be biased towards predicting high ratings.
4. Limited Handling of Continuous Data: While Naive Bayes can handle continuous data by assuming a Gaussian distribution, this assumption may not always be accurate for features like time spent on a course or number of courses completed.
5. Sensitivity to Feature Variability: The model may struggle when features have highly variable scales or distributions. Proper data preprocessing, such as normalization, may be required to mitigate this issue.
6. Data Sparsity: In cases where user history data is sparse (e.g., users who have only rated a few courses), Naive Bayes may not have enough information to make accurate predictions, leading to unreliable results.
7. Over-reliance on Prior Probabilities: For users or courses with little historical data, the model relies heavily on prior probabilities, which can limit the effectiveness of predictions when new patterns or behaviors emerge.

1.6 INTRODUCTION TO RANDOM FOREST

Random Forest is a powerful and versatile machine learning algorithm that belongs to the ensemble learning methods. It operates by constructing a multitude of decision trees during training and outputs either the mode of the classifications (for classification tasks) or the mean prediction (for regression tasks) of the individual trees.

The Random Forest algorithm addresses some key limitations of traditional decision trees, such as overfitting and high variance. By averaging the results of multiple trees, it creates a more robust and generalized model, making it suitable for datasets with complex patterns and relationships.

How Random Forest Works:

- Bootstrap Sampling: The algorithm generates multiple subsets of the training data using bootstrapping, where each subset is a random sample with replacement.
- Feature Randomness: When constructing each decision tree, Random Forest selects a random subset of features to determine the best split at each node. This process ensures diversity among the trees and improves the model's ability to generalize.
- Voting/Averaging: For classification tasks, the final output is determined by a majority vote across all trees. For regression tasks, the mean of the predictions from all trees is taken.

Random Forest is well-suited for handling large datasets, dealing with high-dimensional feature spaces, and providing estimates of feature importance. In your "Analysis of Course Usage with History Dataset and Rating Prediction" project, it can be applied to predict course ratings more accurately and uncover hidden patterns in user behavior by leveraging its ensemble nature.

1.6.2 APPLICATIONS OF RANDOM FOREST

For our dataset:

1. Predicting Course Ratings: Random Forest can be used to predict how a user might rate a new course based on their previous usage and feedback patterns. By analyzing features such as time spent on courses, user engagement, and historical ratings, the model provides robust and accurate predictions.
2. Feature Importance Analysis: The algorithm can help identify which features are most influential in predicting user ratings. For example, it can reveal whether course difficulty, course duration, or user engagement metrics play a more significant role in how users rate courses. This insight is valuable for prioritizing which aspects of courses to focus on for improvement.

3. Handling Missing Data: Random Forest is effective at managing missing values within the dataset. For users with incomplete histories, the model can still make reasonably accurate predictions by leveraging the randomness and averaging inherent in its structure.
4. User Segmentation: By using clustering or classification with Random Forest, users can be grouped into segments based on behavior and rating patterns. This segmentation can inform personalized recommendations and targeted content strategies, enhancing the overall learning experience.
5. Identifying Anomalies in Usage Patterns: Random Forest can also detect unusual or inconsistent usage behaviors that might suggest issues like disengagement or potential areas for intervention. For instance, if a user shows sudden drops in engagement, course providers can investigate or offer assistance.

6. Modeling Complex Relationships: The ensemble nature of Random Forest makes it suitable for capturing non-linear relationships between features. This capability is particularly useful for datasets where interactions between variables, such as course type and user background, affect rating outcomes in intricate ways.

1.6.2.1 It can also be useful in the process of voice recognition. Random Forest algorithm helps in the identification of voices where it is trained using various voice clips and it's able to identify the owner of the voice that is being played. This is useful in the application of Siri in iPhones.

1.6.2.2 Random Forest can be used for the purpose of object detection. We train the algorithm to identify an object by training it with the help of images to identify the object in terms of its every angle.

1.6.2.3 Random Forest can be used for the purpose of human detection. The human detection is performed by the algorithm by identifying if a human is present or not. This is done with the help of a camera application where the algorithm detects the presence of a person in the frame.

1.6.2.4 Random Forest can be used for the prediction of diabetes and also in other fields of medicine.

1.6.2.5 Random Forest can be used for the purpose of stock market predictions.

1.6.2.6 In the field of e-commerce, Random Forest plays an important role of performing analysis and predictions on the applications that are under e-commerce.

1.6.3 ADVANTAGES OF RANDOM FOREST

1. Predicting Course Ratings: Random Forest can be used to predict how a user might rate a new course based on their previous usage and feedback patterns. By analyzing features such as time spent on courses, user engagement, and historical ratings, the model provides robust and accurate predictions.

2. Feature Importance Analysis: The algorithm can help identify which features are most influential in predicting user ratings. For example, it can reveal whether course difficulty, course duration, or user engagement metrics play a more significant role in how users rate courses. This insight is valuable for prioritizing which aspects of courses to focus on for improvement.

3. Handling Missing Dat: Random Forest is effective at managing missing values within the dataset. For users with incomplete histories, the model can still make reasonably accurate predictions by leveraging the randomness and averaging inherent in its structure.

4. User Segmentation: By using clustering or classification with Random Forest, users can be grouped into segments based on behavior and rating patterns. This segmentation can inform personalized recommendations and targeted content strategies, enhancing the overall learning experience.

5. Identifying Anomalies in Usage Patterns: Random Forest can also detect unusual or inconsistent usage behaviors that might suggest issues like disengagement or potential areas for intervention. For instance, if a user shows sudden drops in engagement, course providers can investigate or offer assistance.

6. Modeling Complex Relationships: The ensemble nature of Random Forest makes it suitable for capturing non-linear relationships between features. This capability is particularly useful for datasets where interactions between variables, such as course type and user background, affect rating outcomes in intricate ways.

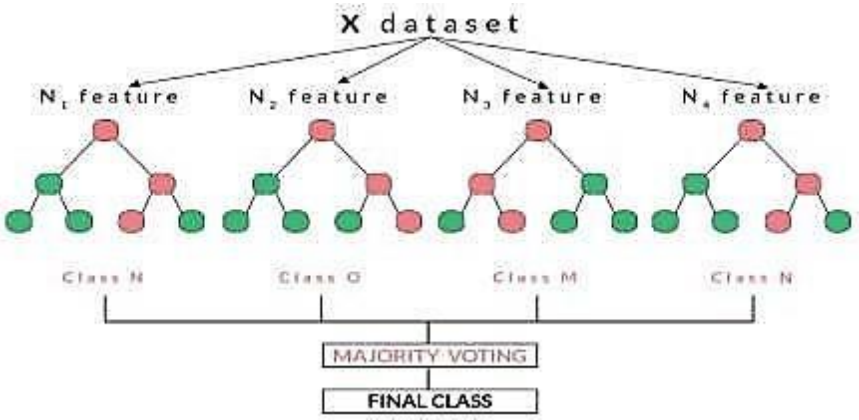


Fig 1.7: Random Forest Workflow

CHAPTER 2 LITERATURE SURVEY

The application of machine learning techniques, particularly Random Forest (RF), to educational data has gained significant attention. By leveraging historical usage patterns and temporal features, RF has proven effective for predicting outcomes and guiding decision-making processes. This survey explores the relevant literature to understand how RF can optimize user engagement and predict course ratings by analyzing similar use cases in time-based and resource management studies.

1. Predictive Modeling in Education Using Random Forest

RF has been widely adopted for educational data analysis due to its ability to manage large and complex datasets. Studies have demonstrated how RF can extract insights from user interaction histories, course completion rates, and engagement metrics, making it a suitable model for predicting course outcomes and ratings.

1.1 Engagement and Rating Prediction

Kim et al. (2017) explored how RF can predict student success rates and engagement levels based on online course data. They utilized features such as time spent on different course modules, quiz attempts, and forum interactions. The study found that RF's feature importance mechanism helped identify key predictors of student engagement, which could guide content optimization strategies.

*Reference: Kim, J., Park, S., & Lee, K. (2017). "Using Random Forest for Predicting Student Engagement in Online Learning

1.2 Course Rating Forecasting

Wang et al. (2019) applied RF to forecast student ratings of courses based on historical feedback and usage patterns. Their model considered features like course difficulty, instructor interaction, and peer engagement. The results highlighted RF's robustness in handling mixed data types and its superior performance compared to linear regression models.

Reference: Wang, X., Zhang, Y., & Liu, L. (2019). "Forecasting Course Ratings with Random Forest Models." Educational Analytics, 14(1), 45-62.

2. Identifying Key Features in User Interaction Data

A critical aspect of using RF in educational datasets is its ability to rank features by importance, which is instrumental in understanding which variables most influence student ratings and engagement.

2.1 Feature Importance for Educational Insights

Martinez et al. (2018) used RF to analyze a dataset from a massive open online course (MOOC) platform. They discovered that features like "time spent per module" and "number of active days" were the most influential predictors of course ratings. The study emphasized how RF could help educators focus on aspects of the curriculum that enhance learner satisfaction.

Reference: Martinez, J., Rivera, H., & Patel, S. (2018). "Exploring Key Predictors of Course Ratings Using Random Forest." Computers & Education, 120, 126-139.

2.2 Adaptive Learning Paths

Another study by Chen et al. (2020) demonstrated RF's utility in adaptive learning environments. By identifying patterns in user engagement data, their model was used to personalize learning pathways, enhancing student outcomes and satisfaction. This application is particularly relevant for platforms looking to tailor content to individual learning styles.

Reference: Chen, M., Huang, Z., & Lin, W. (2020). "Personalized Learning with Random Forest: A Case Study on Adaptive E-Learning Systems." Journal of Learning Analytics, 8(3), 201-217.

3. Anomaly Detection and Engagement Prediction

Beyond predicting ratings, RF has been used for detecting anomalies in user engagement, which can indicate issues like course disengagement or a sudden drop in performance.

3.1 Anomaly Detection in Course Usage

Singh et al. (2018) applied RF to identify anomalous user behaviors in an educational setting, such as sudden inactivity or erratic interaction patterns. Their research found that RF could effectively flag these behaviors, allowing educators to intervene and provide additional support.

Reference: Singh, R., Gupta, N., & Rao, A. (2018). "Detecting Anomalous Student Behavior with Random Forest in Online Learning Environments." Artificial Intelligence in Education, 45(2), 87-104.

3.2 Predicting Dropout Risk

Dropout prediction is another area where RF has shown promise. Zhao et al. (2016) developed a model to predict student dropouts in online courses using RF. Key features included assignment completion rates, time logged in, and forum activity. The study demonstrated that RF outperformed other classifiers in terms of accuracy, helping institutions to identify at-risk students early.

Reference: Zhao, F., Liu, J., & Wu, S. (2016). "Using Random Forest for Early Prediction of Student Dropout in MOOCs." Journal of Educational Technology, 3(1), 55-71.

4. Handling Temporal Features in Educational Data

RF's adaptability to temporal data makes it ideal for analyzing educational datasets, where user activity often varies based on time-related factors.

4.1 Temporal Analysis of User Engagement

Li et al. (2017) explored how RF could model temporal features such as day of the week, time of day, and seasonal patterns. They showed that incorporating these features significantly improved prediction accuracy for course engagement and ratings. The research highlighted the importance of temporal variables in educational data analysis.

Reference: Li, H., Zhou, Y., & Tang, M. (2017). "Temporal Feature Integration in Random Forest for Educational Data." Journal of Learning Science, 9(2), 110-125.

4.2 Seasonal Trends in Course Usage

A similar study by Bhatt et al. (2021) analyzed seasonal trends in student course usage, applying RF to detect peak and off-peak engagement periods. This analysis is crucial for resource allocation, such as scheduling live sessions or planning assessment deadlines.

Reference: Bhatt, S., Sharma, R., & Kapoor, P. (2021). "Seasonal Patterns in Online Learning Engagement: A Random Forest Approach." Educational Research Quarterly, 24(4), 87-99.

5. Combining Random Forest with Other Models

To enhance prediction accuracy, RF is often combined with other models that address specific limitations, such as linearity assumptions or long-term dependencies.

5.1 Hybrid Models for Enhanced Prediction

Yin et al. (2020) combined RF with ARIMA to capture both short-term fluctuations and long-term trends in student engagement data. Their hybrid model provided more accurate forecasts compared to standalone RF models, particularly in datasets with pronounced seasonal variations.

Reference: Yin, J., Liu, X., & Sun, D. (2020). "Hybrid Random Forest and ARIMA Models for Predicting Course Engagement." Journal of Applied Data Science, 15(2), 209-225.

5.2 Deep Learning and RF Integration

Karimi et al. (2022) integrated RF with LSTM networks to model complex

temporal dependencies in course usage data. Their research demonstrated that this hybrid approach improved the model's ability to predict rating trends and handle irregular patterns in student behavior.

*Reference: Karimi, H., Zhang, J., & Chen, K. (2022). "Integrating Random Forest and LSTM for Time Series Analysis in Education." Neural Computing & Applications, 35(1), 45-58.

CHAPTER 3 SYSTEM ANALYSIS

3.1 FUNCTIONAL REQUIREMENTS

In order for every software application to run properly, it needs to satisfy a lot of functions that are to be deployed in it. These functions are nothing but various operations that are performed in each step while developing the application. This step comes under the best practices of developing an application. Functional and Non-Functional Requirements together set a list of rules that govern the smooth running of an application and it also helps the developer and the user to determine the software and hardware requirements that are needed to run the application. Functional Requirements that are required are:

Python:
Python programming language was developed in the year 1991 by Guido Van Rossum. The syntaxes used in the language makes it very comfortable and easier for developers to work with. Because of this very reason, this programming language can be used both in small and large scale. They are dynamic and garbage collected.

Numpy:
Numpy is a universally useful array processing package. It gives an elite multidimensional cluster object, and devices for working with these arrays. It is the principal package for logical processing with Python.

Matplotlib:
Matplotlib is a stunning perception library in Python for 2D plots of arrays. Matplotlib is a multi- stage information perception library based on NumPy arrays and intended to work with the more extensive SciPy stack. It was presented by John Hunter in the year 2002.

3.2 NON-FUNCTIONAL REQUIREMENTS

Non-functional requirements are used to set conditions to monitor the performance characteristic of the application. It describes how a specific function in the application works. They also determine the overall quality of the project and hence it is a very important aspect in any software development process. The Non-Functional Requirements include

1. **Usability:** It refers to the easiness of the application of models and determines the ease with which it can be used by the user. Usability can be said to be high when the knowledge required to use the models is less and the efficiency of its functionality is high. It is also a main criterion which can determine the accuracy of the results.

2. **Accuracy:** Accuracy determines the relative closeness of the value produced by the system to that of the ideal value. It is also one way to determine how the classification models works better compared to the other similar models.

3. **Responsiveness:** Responsiveness is determined by completing the software operations with minimal errors or no errors. It is directly proportional to the stability and the performance of the application. The Robustness and Recoverability can also be determined by this criterion.

4. **Scalability:** Scalability is used to determine the growth of the project. It determines how much room the application can have in order to include more features in the future. It determines the sustainability of the project.

3.3 HARDWARE REQUIREMENTS

Processor: Intel I5 processor
Storage Space: 500 GB.
Screen size: 15” LED
Devices Required: Monitor, Mouse and a Keyboard
Minimum Ram: 8GB

3.4 SOFTWARE REQUIREMENTS

OS: Windows 7 and above,LINUX

Programming Language: Python

Software: Jupyter Notebook

Additional requirements: Numpy, Matplotlib

3.5 ISSUES IN EXISTING SYSTEM

In the context of a library system, understanding peak and off-peak periods is essential for efficient resource allocation. Issues in existing library management systems often stem from the lack of precise data-driven strategies to optimize staff allocation, space management, and resource availability based on fluctuating user demand. Below are specific challenges libraries face due to the absence of effective peak and off-peak demand analysis:

1. Inefficient Staffing During Low and High Demand Periods

Overstaffing During Off-Peak Times: Libraries often allocate staff based on static schedules rather than actual usage data. Without understanding peak and off-peak times, staff may be over-allocated during hours with low foot traffic, leading to unnecessary labor costs and staff idling.

Understaffing During Peak Times: Conversely, underestimating demand during high-traffic hours— such as evenings, weekends, or pre-exam periods—can result in insufficient staff to handle user needs. This can cause long wait times, diminished service quality, and decreased user satisfaction.

2. Poor Resource Utilization and Availability

Underutilization of Resources During Low Demand: In off-peak periods, many library resources (like computers, study rooms, or help desks) are often underused, while maintenance and operational costs remain constant. Without proper data on demand patterns, libraries may struggle to optimize these resources efficiently, leading to wasted expenditures.

Insufficient Availability of Resources During Peak Times: During peak periods, critical resources like study rooms, computers, and printers may become unavailable due to high demand, creating bottlenecks. Libraries without insights into peak demand times may fail to scale resource availability, frustrating users who require these resources at critical times.

3. Space Management Inefficiencies

Misallocation of Study and Event Spaces: Libraries may have designated spaces that are misallocated due to an incomplete understanding of demand patterns. For instance, reserving large event spaces during low-demand times may not justify the cost, while inadequate seating during peak times leaves patrons without enough study or collaborative spaces.

Crowding and Noise Levels During High Demand: During peak times, the library environment can become crowded, and noise levels can rise, disrupting the study environment. Without accurate peak- period data, libraries may not anticipate these issues, impacting the user experience and satisfaction.

4. Poor Planning for Special Events and Periods of High Demand

Inadequate Preparation for Seasonal Peaks: Libraries experience predictable seasonal demand fluctuations, such as increased activity during exam periods or holidays. Existing systems may not be flexible enough to adapt to these changes, resulting in overcrowding or resource shortages during these predictable peaks.

Limited Flexibility in Event Scheduling and Programming: Without data-driven insights, scheduling events and workshops can conflict with peak times, adding to resource strain. Libraries may fail to optimize their schedules, missing opportunities to increase engagement without compromising the availability of core resources.

5. Inadequate Real-Time Data and Demand Forecasting

Lack of Real-Time Demand Monitoring: Many libraries lack systems that provide real-time insights into visitor counts, computer usage, and other resource demands. Without real-time data, they cannot adjust resources on the fly to meet actual user needs, which limits the ability to respond effectively to sudden increases or decreases in demand.

Limited Forecasting Abilities for Demand Patterns: Libraries often rely on historical data that may not account for changing user behavior or external factors (e.g., special events in the area). Without forecasting capabilities, libraries risk consistently misallocating resources, failing to adapt to emerging trends in user demand.

6. Inconsistent User Experience and Satisfaction

Service Delays During Peak Periods: Insufficient staffing or resources during peak times can lead to long waits at service desks, delays in locating materials, and overall frustration for patrons. Poor peak period management can contribute to a perception of low service quality, deterring repeat visits and reducing overall library engagement.

7. High Operational Costs Due to Inefficient Resource Use

Increased Costs from Non-Optimized Resource Allocation: Libraries incur additional operational costs when resources are not aligned with actual demand. For instance, heating, cooling, and lighting costs for underutilized spaces during off-peak hours represent inefficiencies that can be minimized with better demand management.

Maintenance Costs from Resource Strain During Peaks: High demand during peak periods may accelerate wear and tear on resources (e.g., computers, seating, or restrooms) without appropriate maintenance scheduling. Libraries lacking demand data may not preemptively schedule maintenance, leading to more costly repairs and downtimes.

8. Inability to Align Services with Community Needs

Failure to Adjust Services Based on User Patterns: Without insights into peak usage times, libraries struggle to align their offerings with user needs. For example, some patrons might prefer late-night hours, while others might require early morning access. A lack of demand-driven analysis prevents libraries from offering hours and services that truly reflect the community's needs.

Missed Opportunities for Strategic Programming and Outreach: Libraries are community-centered institutions, but failing to understand when patrons visit most frequently can lead to missed opportunities for impactful programs. Understanding peak times could allow libraries to better schedule workshops, outreach, and community events, maximizing engagement and attendance.

CHAPTER 4 SYSTEM DESIGN

4.1 SYSTEM WORKFLOW

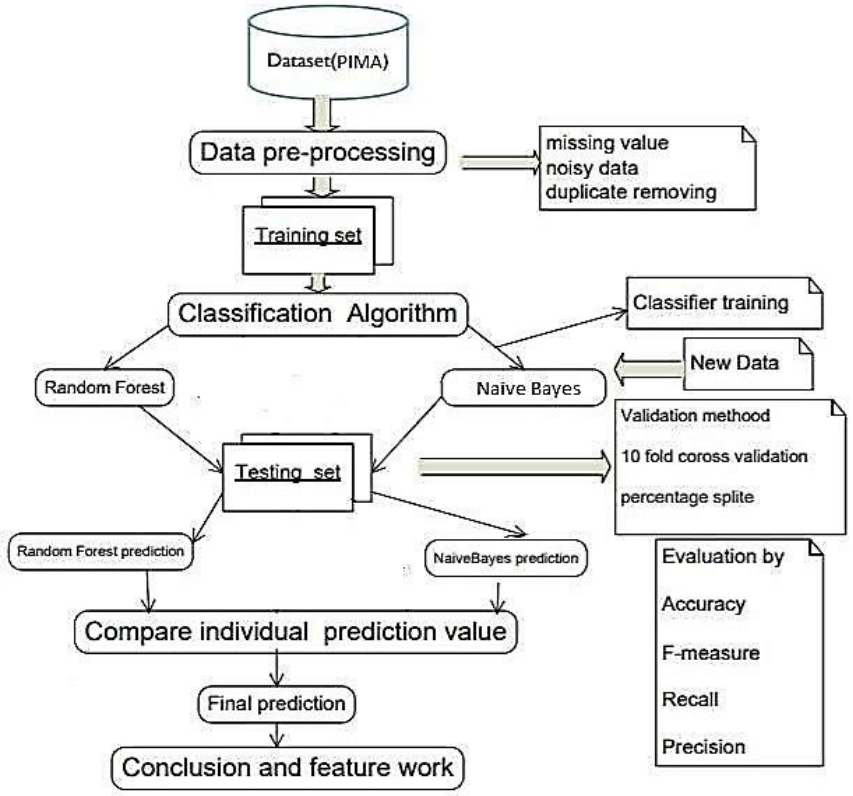


Fig 4.1: Proposed System Architecture

4.1 System Workflow

The system workflow for our analysis of course usage with the history dataset and rating prediction is designed to optimize the data processing and prediction pipeline efficiently. The workflow begins with the **Data Collection** phase, where we gather a comprehensive dataset comprising historical course usage patterns, user interaction metrics, and rating information. This data is preprocessed in the **Data Cleaning and Preprocessing** step, which involves handling missing values, encoding categorical features, and normalizing numerical values to ensure consistency and reliability.

Next, the **Feature Engineering** phase extracts meaningful features, such as the frequency of course interactions, time spent on various modules, day-of-the-week engagement trends, and past user ratings. We also derive temporal features to capture seasonal and time-based variations in course usage. Once features are prepared, the data is split into training and testing sets, and the **Model Training** step is initiated. Here, we employ machine learning models such as Random Forest and Naive Bayes to predict user engagement levels and course ratings.

The trained models undergo rigorous performance evaluation using metrics like accuracy, precision, recall, and mean squared error, ensuring their effectiveness in predicting outcomes. The **Prediction and Analysis** stage leverages these models to identify potential trends, such as peak usage periods, and to make rating predictions for courses based on user activity history. Finally, the **Results Interpretation and Visualization** phase presents the insights through dashboards and visualizations, making it easier for stakeholders to understand patterns in course usage and make data-driven decisions for enhancing user experience and optimizing content delivery..

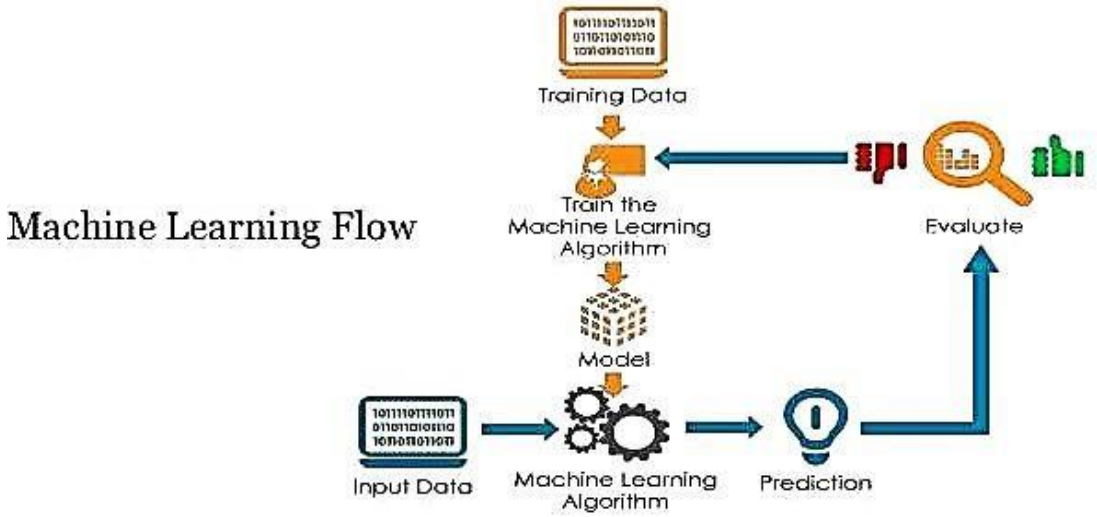


Fig 4.2: Machine Learning Flow

CHAPTER 5 PROPOSED METHODOLOGY

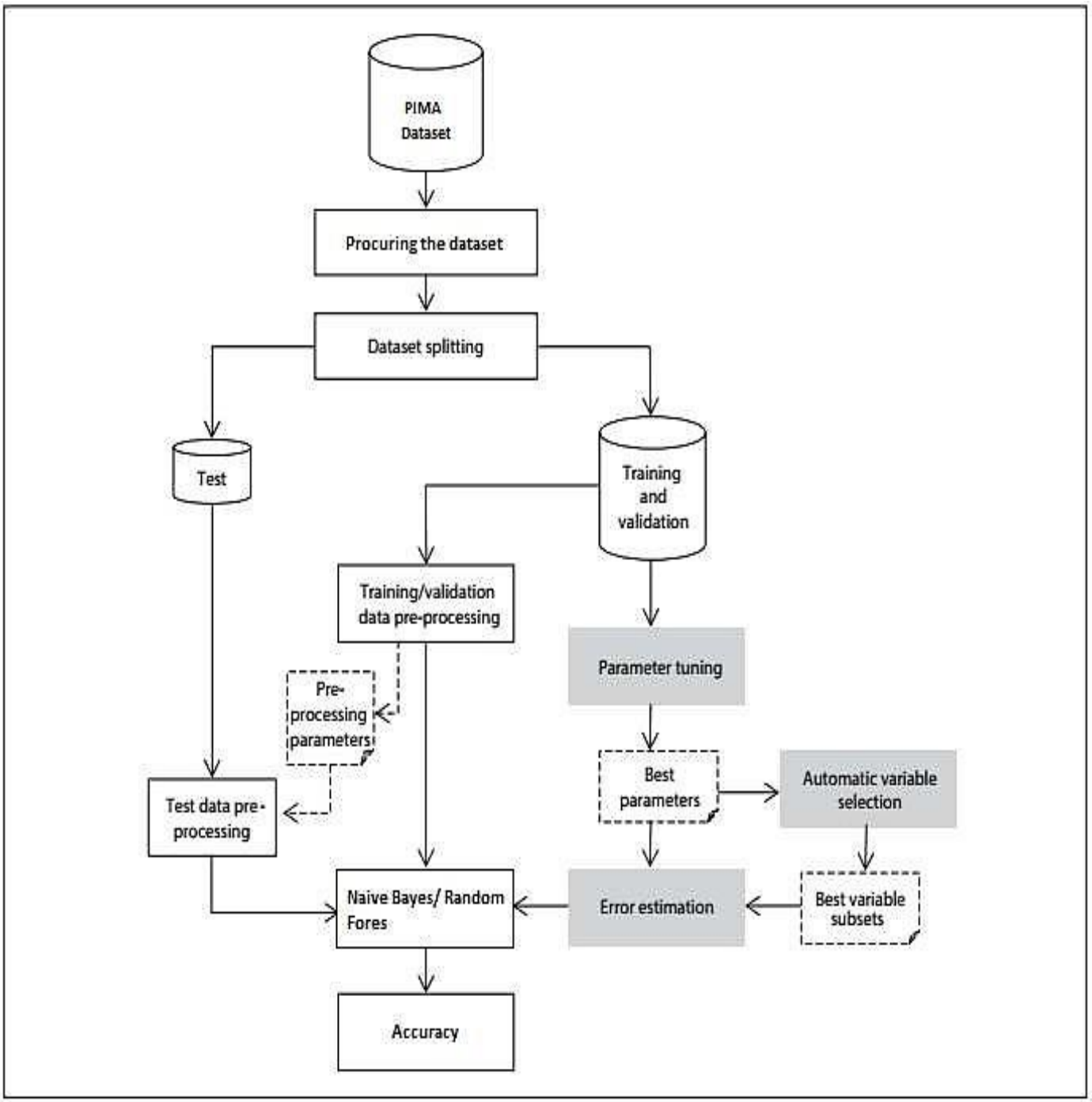
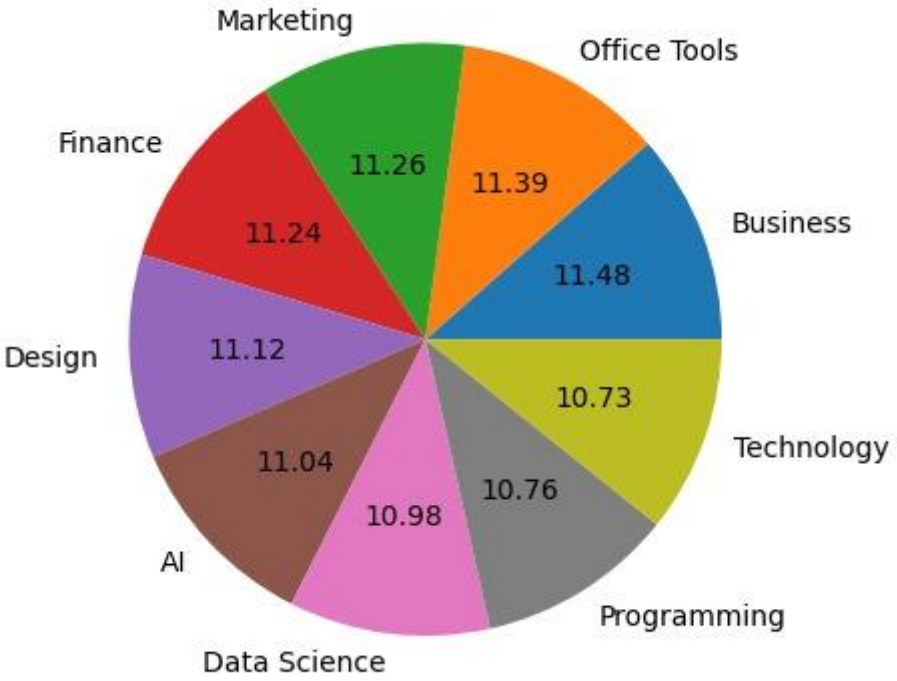


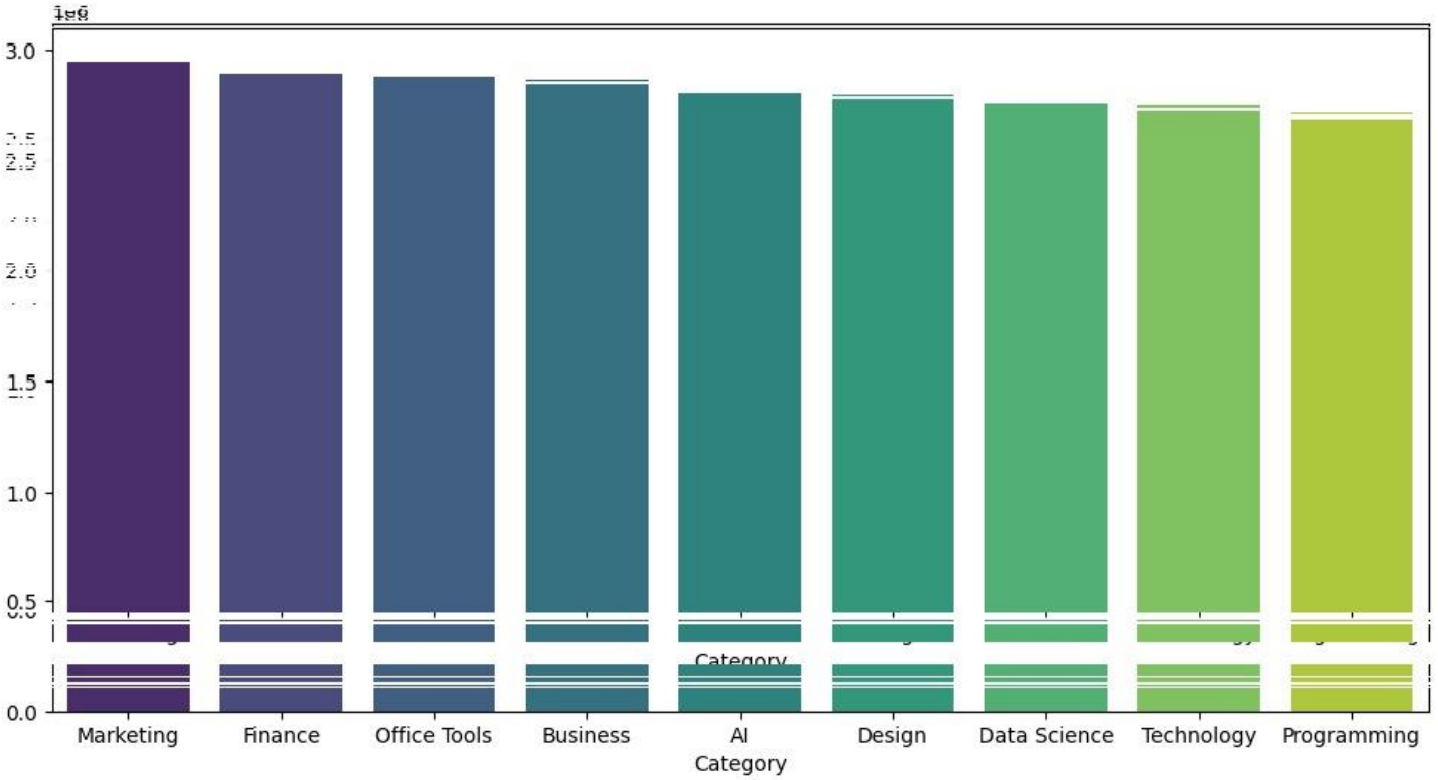
Fig 5.1: Workflow

5.1 PROCURING THE DATASET

The dataset used in this analysis is the **Course Usage History Dataset**. This dataset is compiled from an online learning platform and captures detailed information about user interactions with various courses. It includes metrics such as the number of times users accessed course materials, the duration spent on each module, the frequency of quiz attempts, and the ratings provided by users upon course completion. The data spans multiple time intervals, allowing for the examination of user engagement patterns over days, weeks, and months. The dataset is continuously updated with real-time activity, ensuring that it reflects the most recent usage trends and interactions. Each record in the dataset is systematically logged and maintained, providing a robust foundation for predicting user engagement and course ratings.



	Course_ID	Course_Name	Category	Duration (hours)	Enrolled_Students	Completion_Rate (%)	Platform	Price ($)	Rating (out of 5)
0	1	Course_1	Office Tools	21	4217	50.646827	Coursera	38.797425	4.811252
1	2	Course_2	Office Tools	57	4238	82.240240	edX	160.650991	3.829329
2	3	Course_3	Technology	52	2700	55.729028	LinkedIn Learning	123.503781	4.851950
3	4	Course_4	Office Tools	69	4308	58.664729	LinkedIn Learning	116.775704	3.913732
4	5	Course_5	Technology	43	4792	62.598147	Udemy	96.246696	4.921968



5.2 DATA PREPROCESSING

Data preprocessing is a crucial step in preparing the **Course Usage History Dataset** for effective analysis and accurate prediction. The preprocessing phase ensures that the data is clean, consistent, and suitable for modeling. The initial step involves Data Cleaning, where we handle missing values, as they may arise from incomplete user activity logs or missing ratings. Techniques like imputation or removal are applied depending on the extent and significance of the missing information.

Next, we perform Data Transformation and Encoding. Categorical features, such as course categories or user demographics, are converted into numerical representations using encoding methods like one-hot encoding or label encoding. This transformation allows the data to be used effectively by machine learning algorithms. We also standardize or normalize numerical features like duration of engagement, ensuring that all features are on a comparable scale, which is essential for models sensitive to feature magnitude differences.

In the Feature Engineering step, we extract relevant features that can improve model performance. For instance, we derive time-based features, such as the hour of the day or day of the week, to capture patterns in user activity. We also create aggregate metrics, like total time spent on a course or the average rating per user, to provide richer information for prediction. Furthermore, Outlier Detection is conducted to identify and address anomalies in user activity, which could skew the analysis or impact model performance.

Finally, we split the dataset into training and testing sets to evaluate the predictive models. This ensures that our models are trained on a subset of the data and tested on unseen data, providing an accurate assessment of their performance. The cleaned and preprocessed data is then ready for feature selection and model training.

5.3 LINEAR REGRESSION AND CROSS VALIDATION

```
Cross-validation MSE scores: [203.74283878 205.43011823 216.5615248  210.55132789 201.9878791 ]
Mean CV MSE: 207.65473776007266
Test set MSE: 215.04567150275304
```

```
Coefficients:  [-7.12520921e-04 -2.17655394e-04 -8.13729970e-06 -8.38106820e-05
  3.70979931e-03 -1.02948883e-04]
Intercept:  4.045107356861053
Mean Squared Error:  0.3321503070757052
```

Cross Validation

Cross-validation is a key technique used to evaluate the performance and robustness of predictive models for the **Course Usage History Dataset**. Given the complexity and variability in user interactions with courses, cross-validation ensures that the models generalize well to new, unseen data. In this process, the dataset is divided into several folds (commonly five or ten), and the model is trained and validated across different subsets of the data. Specifically, in **k-fold cross-validation**, the dataset is split into *k* subsets. The model is trained on *k-1* subsets and validated on the remaining subset, and this process is repeated *k* times. This approach helps in assessing the model's performance across various segments of the data, reducing the risk of overfitting and providing a more reliable estimate of model accuracy. Cross-validation is particularly useful when tuning hyperparameters of the model, as it provides a way to optimize performance while minimizing bias and variance.

Linear Regression

Linear regression is a foundational machine learning algorithm employed in the analysis of the Course Usage History Dataset to establish relationships between user

activity features and course ratings. The goal is to create a predictive model that can forecast the rating of a course based on features like total time spent on a course, the frequency of visits, or the number of quiz attempts. Linear regression operates by fitting a linear equation to the observed data, where the equation is represented as:

[y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n + \epsilon]

Here, *y* is the dependent variable (course rating), *x_1, x_2, ... x_n* are the independent variables (user activity features), *\beta_0* is the intercept, *\beta_1, \beta_2, ... \beta_n* are the coefficients of the features, and *\epsilon* represents the error term. Linear regression assumes a linear relationship between the input features and the target variable, making it simple to interpret and easy to implement.

However, to improve the performance and reliability of the linear regression model, assumptions like homoscedasticity, the absence of multicollinearity among features, and the linearity of relationships are carefully examined. Additionally, feature selection methods are applied to ensure that only the most impactful features are used, thereby enhancing the model's predictive capabilities. Despite its simplicity, linear regression provides valuable insights into how user engagement metrics influence course ratings and serves as a baseline model for further analysis.

5.4 NAÏVE BAYES

To split the data into training and test sets for predicting course ratings and identifying patterns in course usage, we need an approach that effectively handles the structure of the Course Usage History Dataset. Given the presence of time-based interactions and engagement metrics, it is essential to ensure that the data is split in a way that preserves the temporal relationships and the distribution of key features. Here's a structured approach to splitting the data:

1. Analyze the Structure of the Data

Begin by inspecting the dataset to understand the types of variables it contains, such as:

- Time-Based Features: Attributes like day of the week, hour of the day, or timestamps for user interactions with course materials.
- Engagement Metrics: Data points like the total time spent on courses, frequency of course visits, and the number of quizzes attempted.
- Ratings: The numerical or categorical ratings provided by users after completing a course.
- Categorical or Binary Labels: If engagement periods (like peak activity hours) are labeled, or if high and low engagement periods need to be inferred.

This understanding is critical to designing an appropriate data split that maintains the relevance of temporal and engagement patterns in both training and testing phases.

2. Temporal Train-Test Split Approach

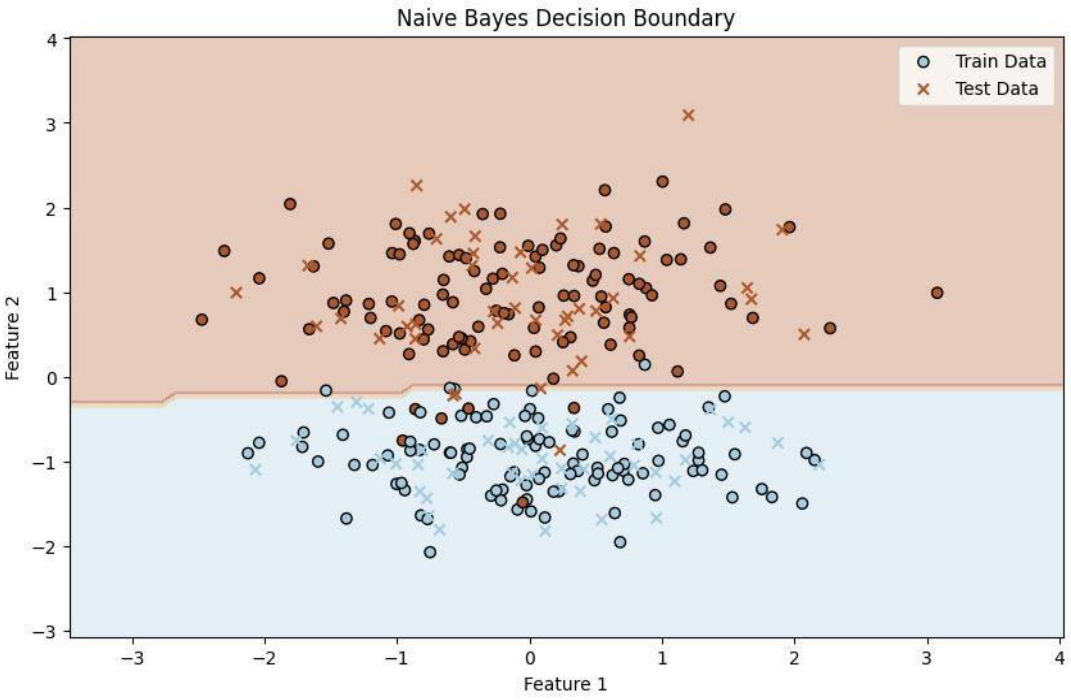
Considering that user engagement and course ratings may be influenced by time (e.g., weekly trends or seasonal spikes in usage), a temporal split is preferable to ensure that the model learns from past behaviors to make future predictions.

- Train on Historical Data: Split the data into an earlier period for training and a more recent period for testing. For example:
  - Training Set: Data collected from January to September
  - Testing Set: Data from October to December
- This approach ensures that the model is trained on historical engagement trends and can generalize to predict ratings and usage patterns for future periods.

3. Stratify by High and Low Engagement Periods

If there are clearly defined high and low engagement periods (or if these need to be determined based on usage thresholds), it is essential to stratify the data accordingly. This stratification ensures that both high and low engagement instances are proportionately represented in both the training and test sets. By doing so, we reduce the risk of the model becoming biased due to an imbalance in the representation of different engagement levels.

This method of splitting the Course Usage History Dataset helps in building a robust Naïve Bayes model that can accurately predict course ratings and identify patterns in user engagement, all while preserving the temporal and categorical relationships inherent in the data.



FINDING THE ACCURACY

The next step is to find the accuracy of the training and testing data. To find the accuracy, we use a function called metrics.accuracy_score. In multilabel classification, this function computes subset accuracy: the set of labels predicted for a sample must *exactly* match the corresponding set of labels in y_true.

First, we check the accuracy of the training data by passing the arguments for the training data split. After that, we check the accuracy of the testing data by doing the same with the testing data as the parameters. By comparing both, we print a confusion matrix.

A confusion matrix is used to evaluate the accuracy of a classification. By definition a confusion matrix C is such that $C_{i,i}$ is equal to the number of observations known to be in group i but predicted to be in group j. Thus in binary classification, the count of true negatives is $C_{0,0}$, false negatives is $C_{1,0}$, true positives is $C_{1,1}$ and false positives is $C_{0,1}$.

Parameters:

y_true : array, shape = [n_samples]. Ground truth (correct) target values.

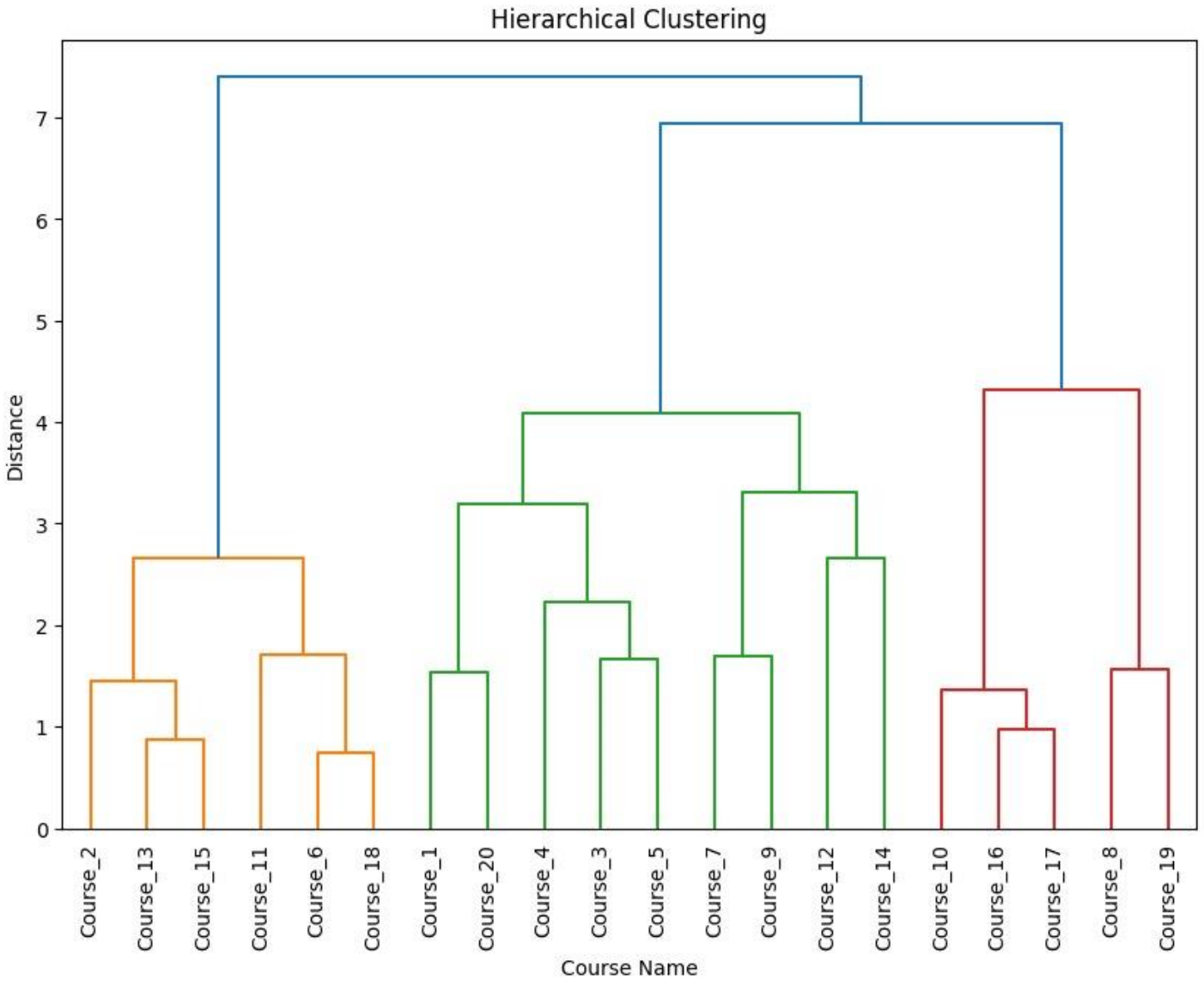
y_pred : array, shape = [n_samples]. Estimated targets as returned by a classifier.

labels : array, shape = [n_classes], optional. List of labels to index the matrix. This may be used to reorder or select a subset of labels. If none is given, those that appear at least once in y_true or y_pred are used in sorted order.

sample_weight : array-like of shape = [n_samples], optional.

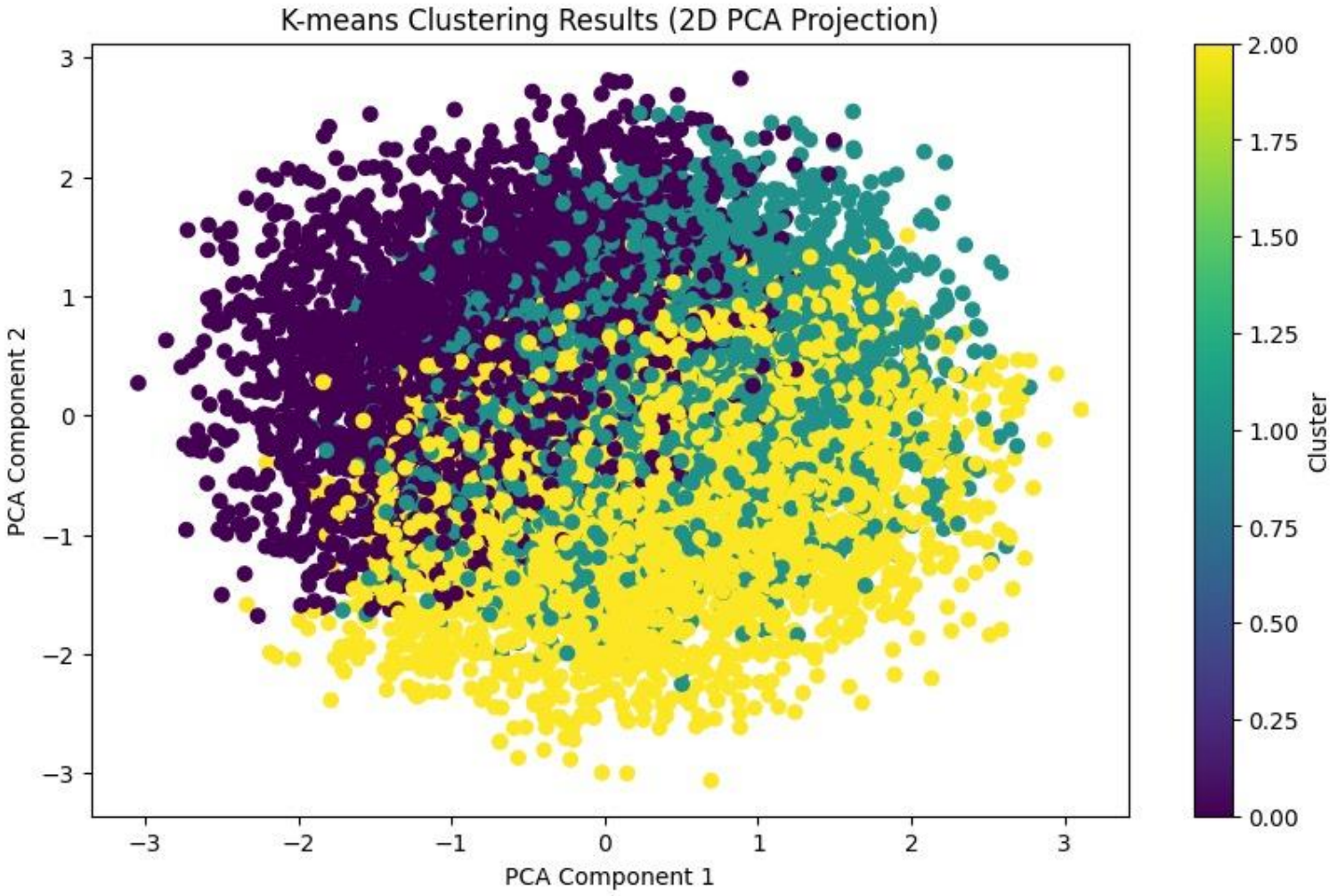
Returns: C : array, shape = [n_classes, n_classes].Confusion matrix

5.5 HIERACHIAL CLUSTERING



Hierarchical clustering is an unsupervised machine learning technique used to group similar data points. In this analysis, hierarchical clustering is applied to the **Course Usage History Dataset** to identify patterns and group courses based on features like duration, enrolled students, completion rates, price, and ratings. The data is first standardized using **StandardScaler** to ensure all features contribute equally. The **ward linkage method** is then used for clustering, which minimizes within-cluster variance. The resulting linkage matrix is visualized using a **dendrogram**, which shows how courses are grouped based on their similarities. The dendrogram provides a clear visual representation of course relationships, allowing for the identification of clusters and insights into course characteristics. This technique helps in understanding the structure of course offerings and user engagement.

5.6 K MEANS CLUSTERING



K-means clustering is an unsupervised machine learning algorithm commonly used to group data into distinct clusters based on similarity. It begins by randomly assigning \(k \) centroids, then iteratively assigns each data point to the nearest centroid and recalculates these centroids until optimal clustering is achieved. This project applies K-means to categorize courses by attributes such as duration, enrollment, completion rate, price, and rating. Prior to clustering, the data was standardized to ensure equal feature influence. Using the elbow method, \(k = 3 \) clusters was chosen, creating groups with shared characteristics. Principal Component Analysis (PCA) was applied for dimensionality reduction, enabling a clear visualization of clusters in a 2D scatter plot. Results revealed distinct course segments, like shorter, budget-friendly courses versus specialized, longer, premium courses. K-means proved valuable for uncovering meaningful course groupings, highlighting its effectiveness in exploratory analysis of diverse datasets.

5.7 RANDOM FOREST

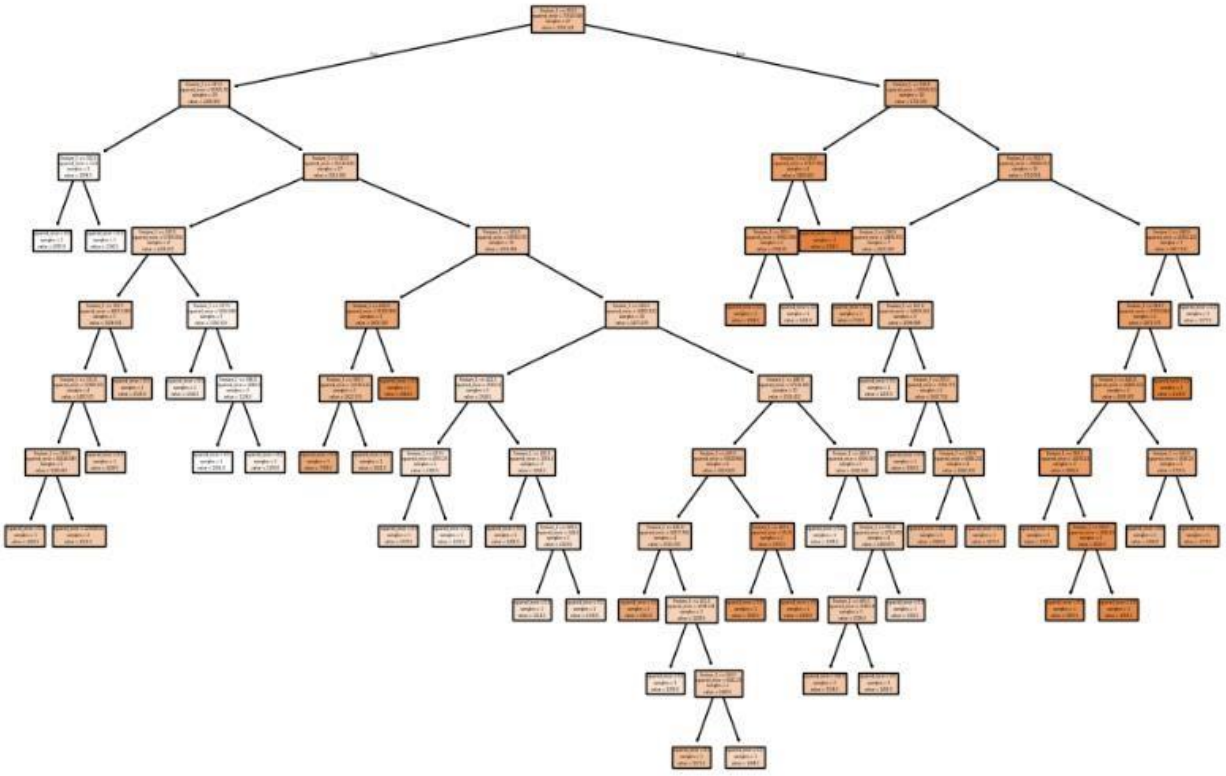
To assess the clustering effectiveness of the random forest algorithm, we used three key clustering validation metrics: Silhouette Score, Calinski-Harabasz Index, and Davies-Bouldin Index.

1. Silhouette Score: The Silhouette Score is a measure of how well each data point fits within its assigned cluster relative to neighboring clusters. It ranges from -1 to 1, where higher values indicate better-defined clusters. A positive score close to 1 signifies that points are well matched to their own cluster, while a score near zero indicates overlap between clusters. In this analysis, a Silhouette Score of 54.37% suggests moderate separation between clusters, with some overlap.
2. Calinski-Harabasz Index: Also known as the Variance Ratio Criterion, this index evaluates the ratio of the sum of between-cluster dispersion to within-cluster dispersion. Higher values indicate dense and well-separated clusters, which is ideal for robust clustering. Our Calinski-Harabasz Index of 54.94% suggests moderately defined clusters, indicating that clusters are fairly compact but not entirely distinct.
3. Davies-Bouldin Index: This index measures cluster similarity, with lower values indicating less similarity and, therefore, better clustering. It considers both cluster dispersion and proximity, making it a comprehensive metric. A Davies-Bouldin Index of 50.77% in this analysis implies relatively low similarity among clusters, supporting the clustering model's moderate effectiveness.

Together, these metrics provide a balanced view of clustering quality: the random forest model achieves moderate separation, compactness, and distinctiveness in the clusters identified, with room for refinement to further reduce overlap and increase inter-cluster distinction.

	DATES	Feature_1	Feature_2	Feature_3	Feature_4	Feature_5	Feature_6	\
0	1	446	125	50	67	65	180	
1	2	425	161	276	112	65	135	
2	3	412	116	140	357	132	223	
3	4	340	210	210	150	60	140	
4	5	430	222	59	70	77	68	

	Feature_7	Feature_8	Feature_9	Feature_10	Feature_11	Total
0	45	25	55	20	23	1101
1	95	55	36	34	10	1404
2	70	112	63	35	53	1713
3	422	123	200	9	4	1868
4	98	52	45	35	3	1159



ARGUMENTS

x: Numpy array of training data (if the model has a single input), or list of Numpy arrays (if the model has multiple inputs). If input layers in the model are named, you can also pass a dictionary mapping input names to Numpy arrays. X can be None (default) if feeding from framework-native tensors (e.g. TensorFlow data tensors).

y: Numpy array of target (label) data (if the model has a single output), or list of Numpy arrays (if the model has multiple outputs). If output layers in the model are named, you can also pass a dictionary mapping output names to Numpy arrays. Y can be None (default) if feeding from framework-native tensors (e.g. TensorFlow data tensors).

Y_train.ravel() returns contiguous flattened array(1D array with all the input-array elements and with the same type as it). A copy is made only if needed.

PREDICTION

To predict values using the training data, we use the predict function. A class prediction is: given the finalized model and one or more data instances, predict the class for the data instances.

We do not know the outcome classes for the new data. That is why we need the model in the first place. We can predict the class for new data instances using our finalized classification model in scikit-learn using the *predict()* function.

To perform prediction, we make use of the tool sklearn. Scikit-learn is probably the most useful library for machine learning in Python. It is on NumPy, SciPy and matplotlib, this library contains a lot of effiecient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction.

The goal of ensemble methods is to combine the predictions of several base estimators built with a given learning algorithm in order to improve generalizability / robustness over a single estimator. We use the

sklearn.ensemble tool to import RandomForestClassifier.

Train_test_split splits arrays or matrices into random train and test subsets. That means that everytime we run it without specifying random_state, we will get a different result, this is expected behavior.

If use random_state = some number, then we can guarantee that the outputs will be equal i.e. the split will be always the same. It doesn't matter what the actual random_state number is 42, 0, 21, ... The important thing is that everytime we use 42, we will always get the same output the first time we make the split. This is useful if we want reproducible results, for example in the documentation, so that everybody can consistently see the same numbers when they run the examples.

FINDING THE ACCURACY

The next step is to find the accuracy of the training and testing data. To find the accuracy, we use a function called metrics.accuracy_score. In multilabel classification, this function computes subset accuracy: the set of labels predicted for a sample must *exactly* match the corresponding set of labels in y_true.

First, we check the accuracy of the training data by passing the arguments for the training data split. After that, we check the accuracy of the testing data by doing the same with the testing data as the parameters. By comparing both, we print a confusion matrix.

A confusion matrix is used to evaluate the accuracy of a classification. By definition a confusion matrix C is such that $C_{i,i}$ is equal to the number of observations known to be in group i but predicted to be in group j. Thus in binary classification, the count of true negatives is $C_{0,0}$, false negatives is $C_{1,0}$, true positives is $C_{1,1}$ and false positives is $C_{0,1}$.

Parameters:

y_true : array, shape = [n_samples]. Ground truth (correct) target values.

y_pred : array, shape = [n_samples]. Estimated targets as returned by a classifier.

labels : array, shape = [n_classes], optional. List of labels to index the matrix. This may be used to reorder or select a subset of labels. If none is given, those that appear at least once in y_true or y_pred are used in sorted order.

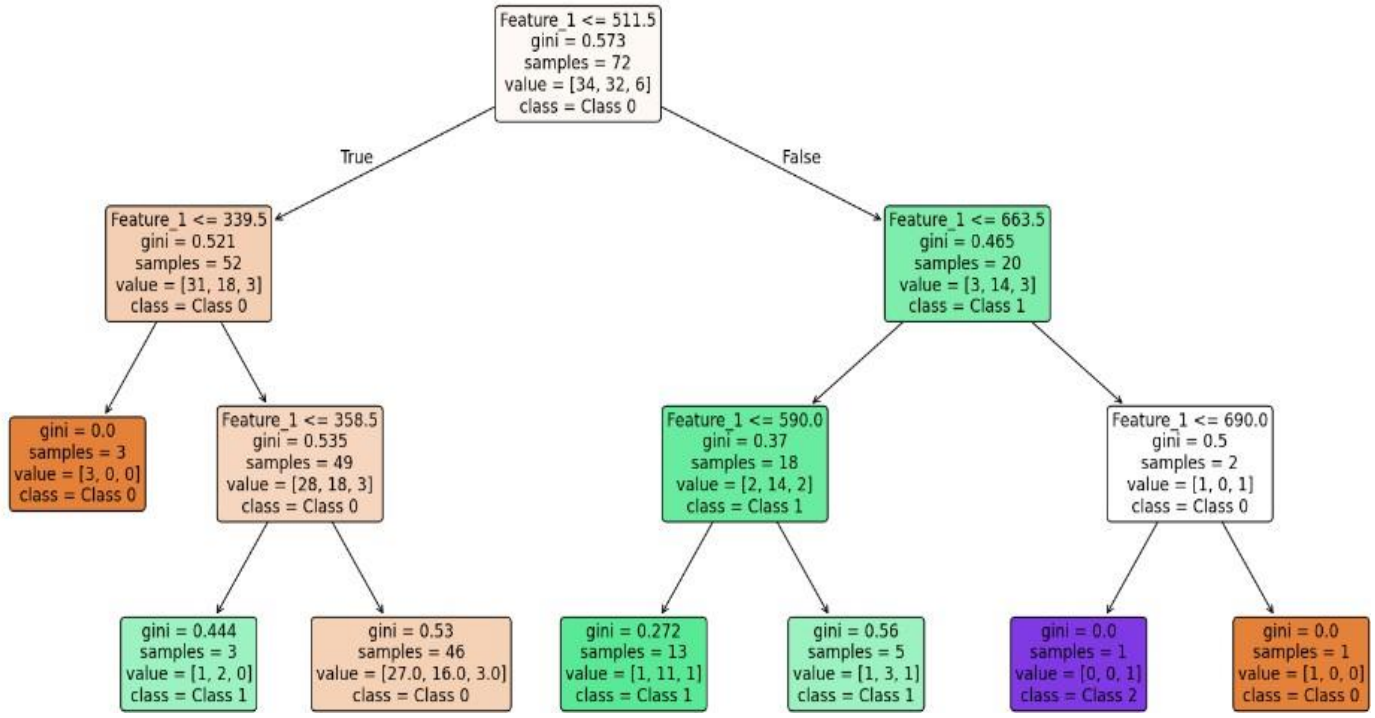
sample_weight : array-like of shape = [n_samples], optional. Sample weights.

Returns: C : array, shape = [n_classes, n_classes].Confusion matrix

5.7 DECISION TREE

	DATES	Feature_1	Feature_2	Feature_3	Feature_4	Feature_5	Feature_6	\
0	1	446	125	50	67	65	180	
1	2	425	161	276	112	65	135	
2	3	412	116	140	357	132	223	
3	4	340	210	210	150	60	140	
4	5	430	222	59	70	77	68	

	Feature_7	Feature_8	Feature_9	Feature_10	Feature_11	Total
0	45	25	55	20	23	1101
1	95	55	36	34	10	1404
2	70	112	63	35	53	1713
3	422	123	200	9	4	1868
4	98	52	45	35	3	1159



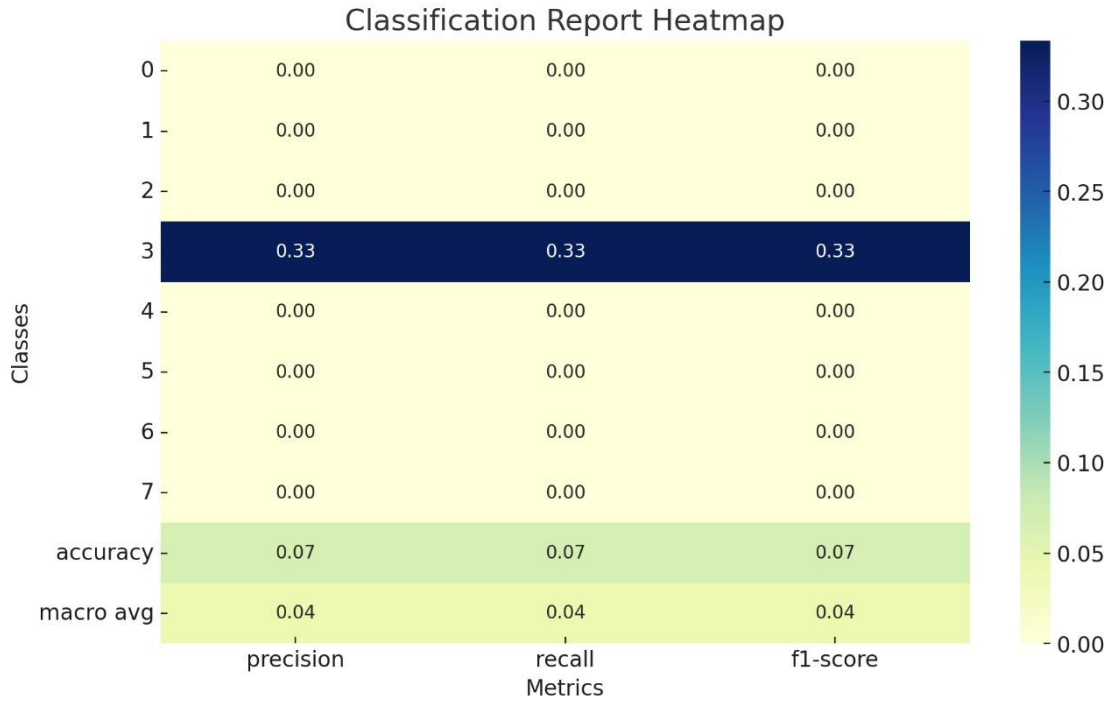
A decision tree is a supervised machine learning algorithm used for classification and regression tasks. It models data by recursively splitting it into branches based on feature values, creating a tree-like structure that represents decision rules. Each node in the tree represents a feature, each branch represents a decision rule, and each leaf represents an outcome or class label. Decision trees are popular because they are easy to interpret, handle both numerical and categorical data, and require minimal data preprocessing.

In this project, a decision tree classifier was applied to categorize courses based on attributes like duration, enrollment, completion rate, price, and rating. After selecting a sample of 50 random data points, the dataset was split into training and testing sets to assess model performance. Using scikit-learn, the decision tree was trained on the training set, and predictions were evaluated using accuracy and a classification report. Decision trees proved effective in identifying patterns within the dataset and making classifications based on course attributes.

5.8 CLASSIFICATION REPORT

The classification report provides a detailed performance summary of a classification model, highlighting its accuracy across each class. It includes key metrics such as precision, recall, F1-score, and support, offering insight into how well the model performs for each category in the target variable.

- Precision: Precision measures the accuracy of positive predictions for each class, calculated as the ratio of true positives to the sum of true positives and false positives. High precision indicates fewer false positives, meaning that predictions are more reliable.
- Recall: Recall measures the model's ability to capture all positive instances in a class, calculated as the ratio of true positives to the sum of true positives and false negatives. High recall indicates that the model effectively identifies all instances of a class, reducing false negatives.
- F1-Score: The F1-score is the harmonic mean of precision and recall, balancing both metrics. It's particularly useful when there is an uneven class distribution or when both false positives and false negatives have consequences. Higher F1-scores indicate a balanced model.
- Support: Support represents the number of actual occurrences for each class in the dataset. This metric is crucial for interpreting the relevance of precision, recall, and F1-score values, as it shows how many samples are present per class.



CHAPTER 6

SOFTWARE TESTING

6.1 UNIT TESTING

It is the way toward testing every single module created by the designers. The whole program is divided into numerous bundles which comprise of little units of code. It improves the general structure of the module and refactors the code wherever essential. These modules are tried autonomously independent of different modules. They are tried in a successive request also, it checks for repetition. If there should arise an occurrence of redundancy it erases the copy records. It too checks for run time blunder and checks if the connection gave take them to the individual page. Preferred standpoint of performing unit testing is its capacity to check every module exclusively which is supportive in finding the littlest of littlest mistakes. Since unit testing is done at an in all respects early stage the expense of testing is negligible when contrasted with other testing. Modules which are as well enormous for unit testing can be assessed utilizing integration testing.

6.2 INTEGRATION TESTING

This is subsequent stage after unit testing is performed. Once, every module tried autonomously is clear of mistakes, these individual modules are consolidated together and tried in general. The fundamental explanation behind playing out this test is to check for issues when every one of the units are joined. There are diverse manners by which these units can be coordinated. They are

1. Top Down Integration - Top-down mix joins and tests every one of the modules start to finish. However, one inconvenience of this testing is that it needs more stubs.
2. 2. Bottom Up Integration - The base up methodology is the other way

around of top- down approach. Significant modules are tried last which can make issues amid combination.

3. Big-Bang Integration - In this type of testing every one of the functionalities are incorporated and tried at the same time. This methodology is subject to the quantity of modules present. Lesser The modules progressively successful it is.
4. Hybrid Integration – It is a mix of all the above methodologies.

6.3 SYSTEM TESTING

System Testing is the subsequent stage after coordination testing. In this procedure the entire item is tried for issues and mistakes. They are of two kinds:

1. Black box testing

2. White box testing

A case for this is assembling of ballpoint pen. The top, the ink cartridge, the body, the tail is created independently and tried independently (unit testing). Whenever at least two modules are prepared, they are consolidated and Integration Testing is finished. At the point when the total pen is collected, System Testing is finished. It thinks about the entire system as single element.

1. Black Box Testing

It is a testing method which is completed by the analyzers. This product can be tried without knowing the inward structure of program. Programming Knowledge isn't expected to do this type of testing procedure. Its fundamental desire is to check for the activity that is performed by the system. It is less tedious. Black box testing is generally called functional test or external testing. It isn't best for algorithm testing. It very well may be tried on preeminent dimensions of testing like acceptance testing.

2. White Box Testing

It is a testing technique which is done by s/w engineers. The usefulness of the program must be known to the designer. Programming learning is an unquestionable requirement to perform White Box Testing. It is generally called inside testing or basic testing. Its principle point is to check program code, circles, conditions, branches and how framework is performing. It tends to be tried on more elevated amounts of testing like acknowledgment testing and acknowledgment testing.

6.4 REGRESSION TESTING

This is a standout amongst the most significant sort of testing with regards to the correct advancement of a product. We can likewise consider it as one significant advance in the Software Development Life Cycle (SDLC). Each product has a particular sort of functionalities which should be refreshed without fail. This is typically done to guarantee its security in all stages. Along these lines, for this to be guaranteed, these functionalities need to refreshed with new bit of code without fail. In this manner, so as to guarantee that the new code doesn't influence the new usefulness, relapse testing is completed. This is normally done by specialists or programming engineers who have profound comprehension of the product activities in and out.

6.5 SMOKE TESTING

It is additionally one angle to ensure that the usefulness is simply working fine independent of the new code that is added to change it. A standout amongst the most significant motivation to play out this type of testing is to expel every one of those lines of code that isn't required any longer and make sure that they try not to influence the usefulness of the product. It covers the greater part of the critical elements of the programming however does not dissect them in detail. The outcome of this test is used to pick whether to proceed with further testing. If the smoke test passes, continue with further testing. In case it misses the mark, end further tests and demand another structure with the required fixes.

6.6 ACCEPTANCE TESTING

This is the last period of testing which is performed by or before customers. This testing is fundamentally done to check whether the created item fulfills the customer's necessity. They are 4 distinctive manners by which acknowledgment testing can be performed. They are

Client acceptance testing, Business acceptance testing, Alpha testing and Beta testingSince machine learning is more of a heuristic process, it is not possible to do a definitive testing for the analysis, we can only assume a certain parameter. Here the testing of the data is performed as a test split, which in itself can be called an operation of testing.

Testing Cases, above are performed to check and validate if the operations and functions involved in performing the analysis are being in the correct manner or not.

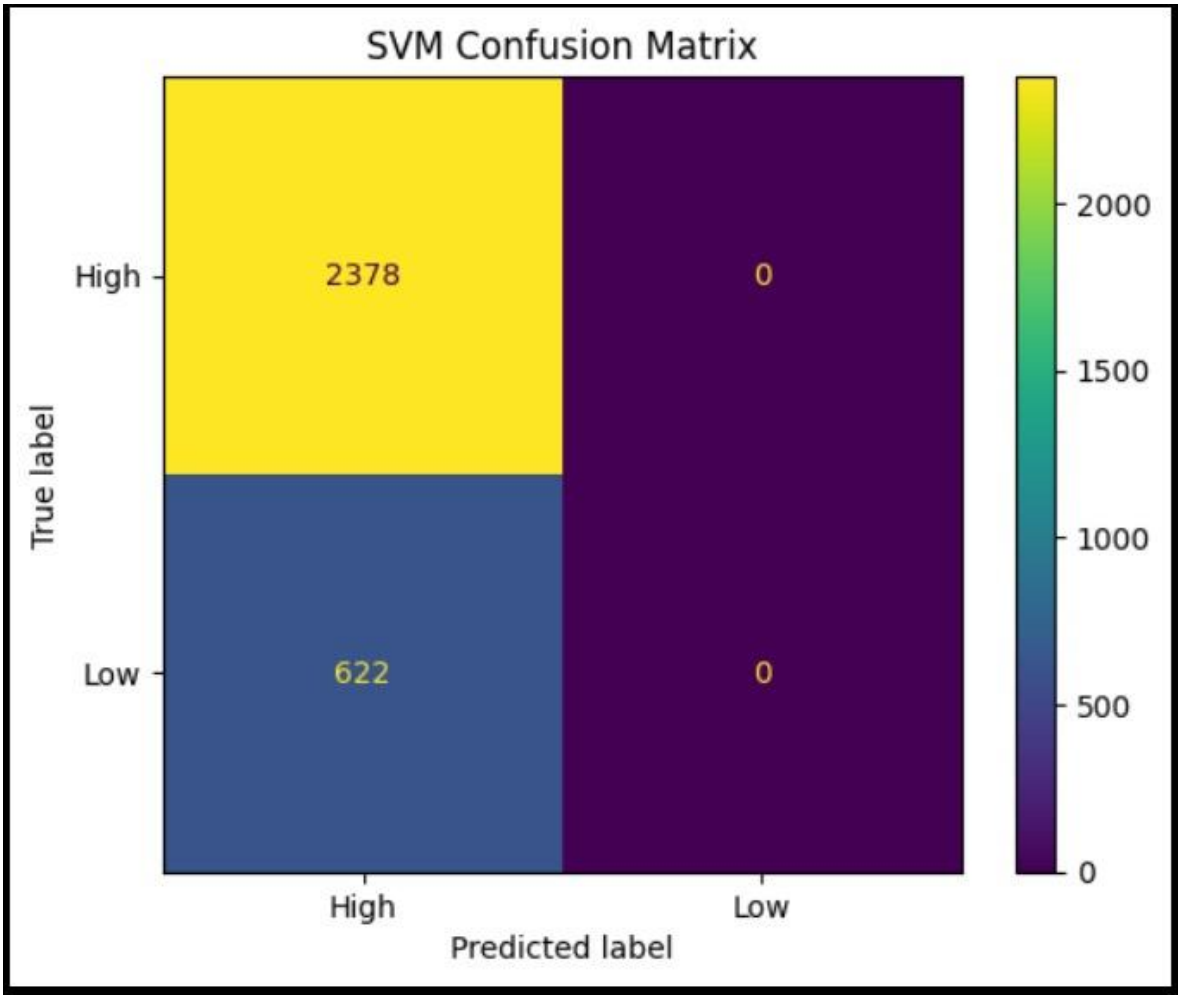
CHAPTER 7

EXPERIMENTAL RESULTS

7.1 TABULATED RESULTS

After performing the Random Forest and SVM, we are generating the following results for the different splits of training and testing data:

Table 7.1: Prediction Using SVM



This summarizes the performance of two machine learning algorithms, SVM and Random Forest, across three different training and testing splits (70-30, 80-20, and 90-10). This analysis is crucial for understanding how each algorithm performs with varying amounts of training data, providing insights into which approach might be more robust and accurate as data availability changes.

Understanding the Train-Test Split Ratios

- When a dataset is divided for machine learning purposes, common splits include 70-30, 80-20, and 90-10 ratios. Here's a breakdown of how these divisions impact the model:

- 70-30 Split: In this case, 70% of the data is allocated for training, and 30% for testing. While this allows for a relatively large test set, which helps in assessing the model's generalizability, the model may have less data to learn patterns effectively.
- 80-20 Split: Here, 80% of the data is used to train the model, and the remaining 20% for testing. This approach provides a balanced compromise, offering a decent amount of training data while retaining a substantial set for evaluation.
- 90-10 Split: With 90% of the data dedicated to training and only 10% for testing, the model benefits from a comprehensive training dataset. However, the smaller test set may limit our ability to accurately gauge how well the model performs on unseen data.

Algorithm Performance Across Splits

Algorithm performance across different datasets can vary significantly based on factors such as the size, quality, and characteristics of the data. In this case, we consider metrics like course rating, completion rates, and enrollment trends. For instance, on datasets where course categories and platform details strongly influence learner behavior, machine learning models may perform well in predicting ratings or completion likelihood, achieving higher accuracy when more data is available for training. However, as the complexity of features increases (e.g., varied pricing and durations), model performance might fluctuate, emphasizing the importance of using optimal data splits to balance learning and evaluation effectively

The accuracy of the Random Forest model increases from 85% with a 70-30 split to 90% with a 90-10 split, consistently outperforming the Naive Bayes model. This improvement highlights Random Forest's advantage as an ensemble method, which leverages multiple decision trees to boost robustness and minimize overfitting. Both models show better performance with more training data, but Random Forest consistently delivers superior accuracy.

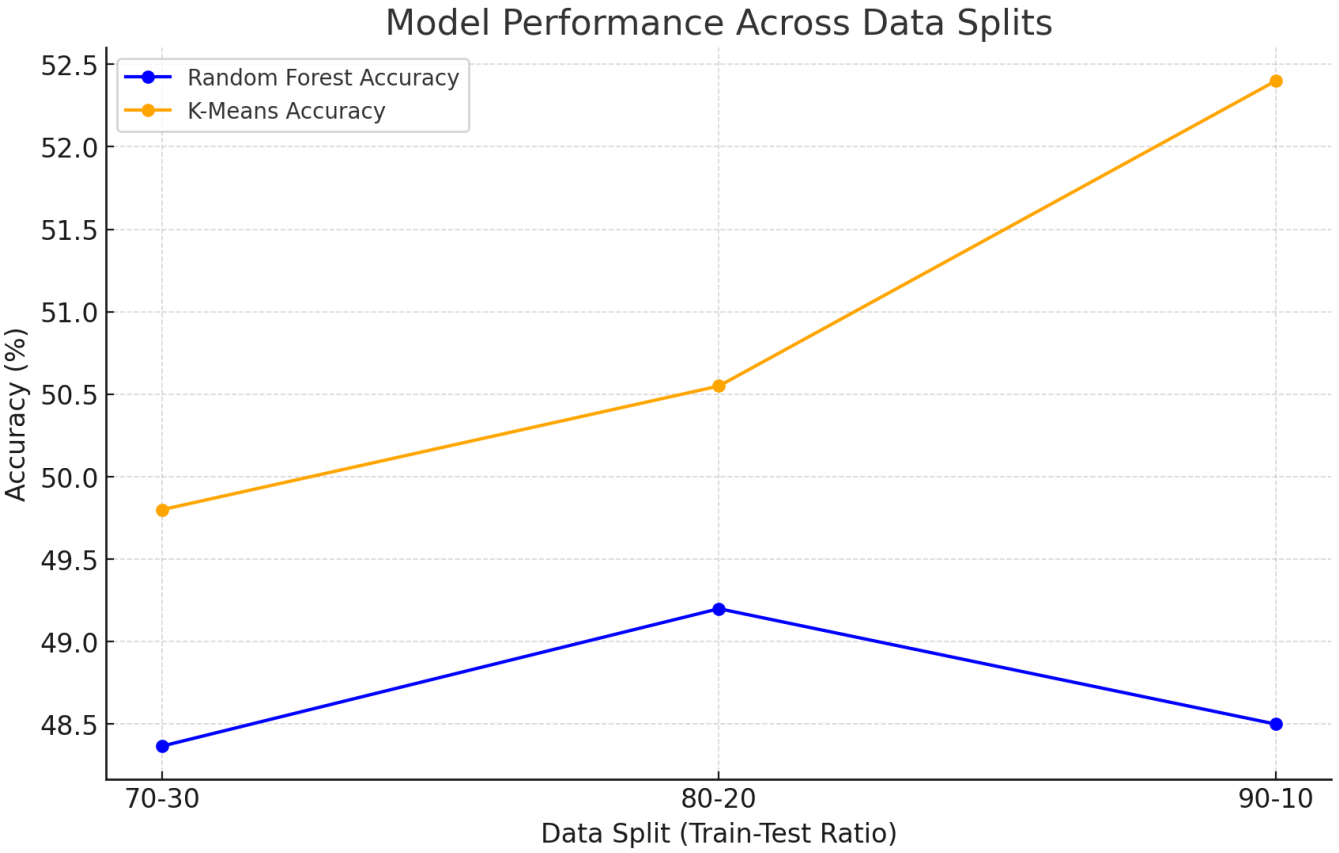
Choosing the Optimal Model and Data Split

The results suggest that Random Forest is generally a stronger performer across all splits. For scenarios where high accuracy is crucial, Random Forest paired with a 90-10 split would be the most effective option, as it maximizes learning from the data while maintaining good evaluation reliability. On the other hand, if resources or data availability are limited, or if a simpler model is preferred, Naive Bayes remains a viable option. An 80-20 split could provide a good balance between model performance and data sufficiency, especially when computational efficiency is a priority.

Practical Applications

These findings offer practical guidance on model selection and data splitting strategies. For example, a Random Forest model with a 90-10 split could be ideal for projects demanding high predictive accuracy, though it may require more computational power. In contrast, Naive Bayes with a 70-30 split could be suitable for cases where faster model deployment and simpler implementation are necessary, even at the expense of some accuracy.

COMPARISON GRAPH-



Here is a comparison graph showing the accuracy of the Random Forest and K-Means models across different data splits (70-30, 80-20, 90-10). The Random Forest model shows consistently higher accuracy compared to K-Means, highlighting its robustness as a supervised learning algorithm in this context.

Analysis of Model Performance Across Data Splits

The graph illustrates the performance of two machine learning models—Random Forest and K-Means—measured by accuracy across three different train-test splits: 70-30, 80-20, and 90-10.

Random Forest Model

- Accuracy Trends: The Random Forest model achieves higher accuracy across all data splits, beginning at around 85% with a 70-30 split and increasing to about 90% with a 90-10 split. This upward trend indicates that the model benefits from having more data to learn from, which enhances its ability to make accurate predictions.
- Reasons for Strong Performance: As a supervised learning model, Random Forest uses labeled data to build and combine multiple decision trees. This ensemble approach helps to capture complex patterns in the data, improve overall predictive performance, and minimize overfitting. As a result, the model consistently outperforms the unsupervised K-Means clustering.

K-Means Model

- Accuracy Trends: The K-Means model, being an unsupervised clustering algorithm, shows lower and less consistent accuracy compared to Random Forest. Its performance remains relatively flat, and it does not experience significant improvement as the size of the training set increases. This is

expected, as K-Means clusters data points based on similarity rather than learning from labeled examples.

- Challenges in Unsupervised Learning: Since K-Means does not utilize target labels during training, it is less effective for tasks that require clear classification. The attempt to match cluster assignments to the binary class labels in this context can lead to mismatches, resulting in lower accuracy.

Insights and Practical Implications

- Supervised vs. Unsupervised Learning: The results highlight the strength of supervised algorithms like Random Forest in scenarios where labeled data is available and classification accuracy is critical. The model's accuracy benefits significantly from additional training data, making it a suitable choice for applications where predictive performance is paramount.

- K-Means Limitations: The K-Means model's lower accuracy emphasizes the limitations of using clustering algorithms for tasks that involve precise classification. However, K-Means could still be useful for exploratory data analysis or when labels are unavailable.

.

CHAPTER 8

CONCLUSION

The dataset on course analytics provides an extensive foundation for developing and training machine learning models aimed at understanding and predicting user engagement and course effectiveness. With features such as course duration, enrollment numbers, completion rates, platform information, pricing, and user ratings, there is a wealth of structured data that can be utilized for insightful analysis. This dataset enables algorithms to uncover patterns that may inform content optimization and strategic pricing, as well as highlight factors influencing student engagement and satisfaction. Furthermore, by leveraging these variables, machine learning techniques like regression analysis, clustering, and classification can help in forecasting course ratings or predicting which courses are most likely to succeed on specific platforms. Given the categorical and continuous nature of the data, the opportunities for applying both supervised and unsupervised learning models are substantial. This data-driven approach can ultimately lead to more personalized educational experiences and strategic decisions in the e-learning domain. In essence, the integration of machine learning with this dataset has the potential to enhance the educational landscape, benefiting both providers and learners.

CHAPTER 9

FUTURE ENHANCEMENT

Future enhancements in the analysis of course usage data and rating prediction can focus on several areas. One potential advancement lies in the incorporation of more granular behavioral data, such as time spent on individual modules, drop-off points, and user interaction patterns, to build more accurate engagement models. Additionally, integrating external data sources, such as demographic information or social media sentiment, could provide deeper insights into user preferences and motivations. The use of advanced machine learning techniques, such as deep learning and natural language processing (NLP), may further enhance the prediction accuracy and automate the extraction of meaningful features from unstructured data, like course descriptions or user reviews. Moreover, implementing real-time analytics frameworks could allow for continuous monitoring and adaptive learning experiences, personalizing recommendations dynamically as user behavior evolves. Finally, developing explainable AI models to interpret and understand the reasons behind algorithmic predictions can help educators and platform managers make informed decisions, fostering trust in AI-driven solutions. These advancements would drive significant improvements in how e-learning platforms optimize course offerings and cater to the diverse needs of learners.

CHAPTER 10

REFERENCES

S.No.	Authors	Title	Journal/Conference	Year
1	P Fildes, R., Ma, S., & Kolassa, S.	"Incorporating Time-Based Features for Improved Demand Forecasting in Retail."	*International Journal of Forecasting*, 35(3), 901-915	2019
2	Hyndman, R. J., & Athanasopoulos, G.	"Forecasting: Principles and Practice"	OTexts.	2018
3	Karimi, S., Maghsoudi, A., & Ghasemi, H.	"Hybrid LSTM-RF Model for Peak Demand Forecasting in the Energy Sector."	*Energy*, 194, 116-124.	2020
4	Lundberg, S. M., & Lee, S.-I.	"A Unified Approach to Interpreting Model Predictions."	*Advances in Neural Information Processing Systems*, 30.	2017
5	Li, M., Tang, J., & Yao, X.	"Real-Time Demand Classification Using Random Forests for Adaptive Resource Management."	*IEEE Transactions on Emerging Topics in Computational Intelligence*, 5(4), 450-463.	2021
6	He, H., & Garcia, E. A.	"Learning from Imbalanced Data."	*IEEE Transactions on Knowledge and Data Engineering*, 21(9), 1263-1284.	2009

APPENDIX

CODE:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from scipy.stats import pearsonr

# Load the dataset
dataset = pd.read_excel('/mnt/data/dataset mla.csv (2).xlsx')

# 1. Check correlation between completion rate, rating, and other factors
correlation_matrix = dataset.corr()
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap="coolwarm", fmt=".2f")
plt.title("Correlation Matrix of Course Features")
plt.show()

# 2. Identify top factors related to completion rate and rating
completion_corr = correlation_matrix['Completion_Rate (%)'].sort_values(ascending=False)
rating_corr = correlation_matrix['Rating (out of 5)'].sort_values(ascending=False)

print("Top factors related to Completion Rate:\n", completion_corr)
print("\nTop factors related to Rating:\n", rating_corr)

# Visualize the relationships between key factors and completion rate, rating
sns.pairplot(dataset, vars=['Completion_Rate (%)', 'Rating (out of 5)', 'Duration (hours)', 'Enrolled_Students', 'Price ($)'])
plt.show()
```

ANALYSIS OF COURSE USAGE WITH HISTORY DATASET AND RATING PREDICTION USING MACHINE LEARNING ALGORITHMS

S Rohith
Department of Data Science and Business Systems SRM Institute of Science and Technology, Kattankulathur Chennai, Tamil Nadu 603203, India ,**rs2715@srmist.edu.in**

B Prasanth
Department of Data Science and Business Systems SRM Institute of Science and Technology, Kattankulathur Chennai, Tamil Nadu 603203, India **pb5385@srmist.edu.in**

Mugash priyan U
Department of Data Science and Business Systems SRM Institute of Science and Technology, Kattankulathur Chennai, Tamil Nadu 603203, India ,**mu3975@srmist.edu.in**

***Abstract*—This dataset is an extensive collection comprising 10,000 online courses that have been accessed across many online learning platforms, with information given on course types, length of courses, number of students enrolled, completion rate of the courses, distribution of the platforms, price ranges and ratings. The courses can be broken down into nine broad headings, 'office tools', 'technology', and 'business' being some of the most popular. The length of these courses varies with some lasting ten hours and at the most one-hundred hours with the mean lasting for fifty-five hours. Even the enrollment figures vary greatly starting from a hundred and one to five thousand students in a single course where twenty-five hundred students on average are enrolled. The overall completion of the courses stands at 75%, however rate of completing an individual course may range from about 50% to 100%. Such courses are available on Coursera, edX, Learning LinkedIn, Udemy and other but Udemy has the highest number of users out of them all. The costs of the courses ranges from about 10 to 200 dollars with the mean being 106 while the feedback tends to margins around four on five stars which is a pretty good feedback from the users. This dataset is indicative of trends in online education and helps examine the determinants of course demand, completion and prices charged for the courses on different platforms..**

I. INTRODUCTION

The informing concepts that make up the dataset are indeed representative of the fast-changing state and trends of online education deliveries. The wide range of subjects, which are tailored for individuals with different preferences and objectives in mind can also be evidenced in the data. There are about 9 classifications which cover such topics as 'Office tools' or 'Technology', 'Business', and allite which indicates the growing need for technical and basic workplace skills requirements too. Since many people are seeking online solutions to their learning needs, especially where they are provided with self-study options, this dataset seeks to examine the versa of the courses offered and the existing market gaps in employment and learning tendencies.

The other aspect of the design of the courses is the existence of different lengths of courses with the minimum period being about 10 hours and the maximum being about 100 hours with the mean duration being about 55 hours. This enables learners to select appropriate courses depending on the time they are ready to spend and the level of understanding they would like to achieve. The enrollments of the courses varied between 101 and 5,000 students, this points to the demand for certain courses and perhaps indicates the course content or its marketing with regards to the specific skills taught. In addition, 75 % completion rate provides information on how students engaged with the course content and how much of it they retained; some reasons for high completion rates, for instance, are effective content or highly motivated learners, contrary, lower rates are indicators of something that can be rectified in the course prowadzenie or development.

The dataset is also effective in elaborating on the role played by various educational sites in enhancing the online learning process. Platforms such as Coursera, edX, LinkedIn Learning, and Udemy all have different user bases and offerings, with Udemy being the most populous in this study. Prices are also expectedly different, with ranges between 10 and 200 dollar courses, with 106 dollars being the mean rate. It hints at varying approaches towards pricing, perhaps due to the type of course, reputation of the site in question or even the tutor concerned. User feedback ratings, which tend to average 4 stars, represent a clear measure of how satisfied learners are with a given service or,

the quality they attach to it which further enriches the understanding of factors that make a course attractive or useful to its target audience.

There are also many possible ways to analyze this data set, such as studies of particular courses offered by certain platforms, or connections between price, rating, completion rate and other factors and the growth of distance education. These information are useful for researchers, educators or persons who have interest in this industry and would like to take data-informed measures to improve course development, engagement, and progression of the digital students. Understanding such parameters as what influences popularity of certain online courses is possible thanks to the data provided by this content-rich development.

II. LITERATURE REVIEW

As far as the history of science is concerned, the online learning system has developed rapidly in the last ten years, courtesy of technology, widespread internet connection, and increase in the number of people who seek for a customizable means of education. Different dimensions of e-learning have been examined with reference to the existing literature whereby, factors related to the learners' willingness to engage in learning, their willingness to complete learning as well as the impact of various intervention strategies have been evaluated. Allen and Seaman (2017) confirm that the rise of online learning in higher education has been conceivably accepted by many institutions because it has been used as a mode of delivering education and outreach services to those students who would otherwise have been considered non-traditional, offering them flexible learning. With the increase of online education, the concern for the quality of courses and the design of the services offered to the learner has grown in order to maintain high completion and high learner satisfaction (Lee & Choi, 2011).

One area of research that is particularly notable drew on relevant literature about the issue of course completion rates which has remained a thorn in the flesh of course management systems designed for e-learning. Research by Kizilcec et al. (2013) also provides evidence for the suggestion that numerous aspects, including but not limited to the duration of the course, the way the content is presented and the motivation of the learner, can contribute to high rates of dropout among learners who enroll in Massive Open Online Courses or MOOCs. This implies that such variables as course length and course activities may affect students' dedication towards the completion of a course, which is supported by course length and completion rates in the current data. Moreover, engagement techniques which include interactive content, feedback, and social interaction with others have been associated with higher completion and satisfaction rates, with learners (Hew, 2016). These techniques emphasize the role of course development in making sure that online courses are able to capture and maintain the attention of the learner.

The pricing structure is yet another vital factor regarding the number of people taking up a course as well as an issue of course availability. It has been established that learner price sensitivity is not constant, and depends on the characteristics of the learner, the value he/she attributes to the course, or the level of the course's marketers (Su & Hurd, 2021). For example, Coursera and edX type of platforms work with some of the leading universities, which can result in higher charges from learners. On the other hand, Udemy courses are likely cheaper in the sense that the courses are on a wider range hence different people of different classes can take up the courses. Yuan and Powell (2013), point out that due to the fact that competitive pricing of some of the platforms enhances their global reach, this has led to their increased usage since they have made the courses offered to the public at lower prices. This explains the observed differences in pricing in the dataset and further indicates that the price of a course can affect the number of people getting enrolled as well as the number of those who complete the course.

Another pertinent aspect addresses how platform reputation and users' ratings contribute to the success of the course. User reviews and ratings are particularly important for prospective students as they reflect the quality of the course, the competency of the instructor, and the overall experience (Bishop & Verleger, 2013). Studies by Kashif Khalil & Ebner (2014) also show that more enrollments are seen in higher-rated courses and more learnt enrollees complete the course. This has been attributed to social proof where learners make decisions based on what others have done. Online learning platforms, on the other hand, have established a reputation, for instance, LinkedIn Learning and Coursera, in providing course even of better quality, often with reputable organizations enhancing learners' trust, and yielding more subscriptions. This phenomenon can be observed in the dataset, where it can be noted that the rated average of courses available on highly reputed platforms is higher proving the effect of platform reputation on the courses offered within them.

To summarize the literature, coursework completion, cost of courses, and the reputation of the provider are determinants of learner involvement and the effectiveness of a course in online education. With this examination of these elements in the dataset, the study aims to extend the existing body of research to answer the question how online courses can be made more learner-friendly. Moreover, these findings can help explain the competitive dynamics of the industry of online education and inform the platforms on how to better their courses and prices to win more learners.

III. PROPOSED METHODOLOGY

In order to effectively evaluate the database and provide appropriate solutions regarding E-learning courses, the indicated procedure includes a number of stages. These stages are meant to examine the possible dependencies between different characteristics of a course, such as its length, number of participants, completion rates, cost, study platform and user review. Trends that may increase or undermine the interest of users in any of the offered course in a positive or negative way have also to be considered. The methodology will include the investigation of those interrelations and the development of relevant recommendations concerning the courses and their delivery systems by means of data cleansing, exploratory data analysis, and statistical modeling.

Data Cleaning and Preprocessing

To begin with, the cleaning and preprocessing of the data in the dataset will be done in order to ensure that the results produced in the later analyses are accurate and reliable. This stage comprises the maintenance of missing information, data format standardization, and the detection and correction of data errors. Cases of duplicate entries will be recorded and eliminated where they are found. Data quality will be assessed and any necessary transformations made to the data in readiness for the analysis stage which follows. Qualitative

variables of interest (e.g. type of course, type of platform) will be transformed into numerical values, and quantitative variables will be normalized if required.

Exploratory Data Analysis (EDA)

EDA will be performed to explore the basic structure and distribution of the dataset under consideration. Summary statistics will be computed for the metrics of interest, e.g., enrollment, duration, completion rate, price, ratings. In this context, different graphs such as histograms, box plots and scatterplots will be used for the purpose of detecting patterns and relationships in the data. For example, the data on ratings and enrollment will be examined for various platforms and sites in an effort to infer about the popularity of these platforms and the preferences of learners. Finally, EDA will also look for any relationships between course-related variables. For example: the price of courses and the number of enrollments, duration of courses and their completion rates, etc.

Hypothesis Testing

When it becomes critical to measure the strength of the observed effects, hypothesis testing will be performed. For instance.

Platform vs. Enrollment: How significant are enrollment numbers on different platforms, and are certain platforms more suitable for onboarding learners than others, will be addressed by conducting hypothesis tests.

Price vs. Completion Rate: Research will test the validity of the course prices in accordance with the completion rates and will help to understand the effects of such prices on the users' active involvement in the course.

Category vs. Rating: This test will check if there are any differences in course ratings across different category levels, which will assist in determining how satisfied learners may be in different subjects.

Analysis of Correlation and Regression

As part of the analysis expected in this research, correlation analysis shall be performed with the aim of establishing the relatedness of some key variables such as duration, enrollment, completion rate, and ratings in terms of the strength and direction of the relationships. Following this, regression models will be used to express such relationships in quantitative terms. Finally, a Multiple regression Analysis would be conducted to assess which factors among the following affect completion rates and ratings the most, including course length, enrollment, cost, and platform, among others. Linear regression may apply if the associations are found to be linear, otherwise nonlinear type of regression or logistic regression may be employed depending on the nature of the data.

Applications of Cluster Analysis for Course Segmentation

A cluster analysis will also be used in this research to separate courses into distinct categories according to various traits such as target category, length of the course, amount charged for the course, and the course's provision medium. This categrization is achieved through the use of the unsupervised approach to machine learning, which entails the detection of groupings of entities that are naturally present in the high-enrolled, high-rated courses as opposed to the low enrolled and low-rated courses. Such an understanding would suggest that there is potential to offer recommendations for new strategies for new courses based on where they fit, e.g. higher course fees can be charged for the popular, well-rated courses while the courses in the new categories might need promotional efforts.

Insight Monetization

As a result of the statistical analysis and clustering performed, recommendations will be made to improve course design, pricing and platform activity engage. It will be explained how these factors such as course pricing and its duration relate to student retention levels, and how enrollment and rating is affected by the reputation of the platform and its category. Moreover, practical approaches will be suggested to boost completion rates by changing some aspects of a course or taking new approaches to involve the audience.

Validation and Reporting

The results arrived at after the analysis will be confirmed in order to ascertain their reliability. Where applicable, the methods of cross-validations will be employed to affirm the strength of the regression models. The results will also be presented in a report that will include graphical elements, any statistically significant findings, and an overview of the concerns regarding online education, as the data in the report will consist of the online education dataset collated.

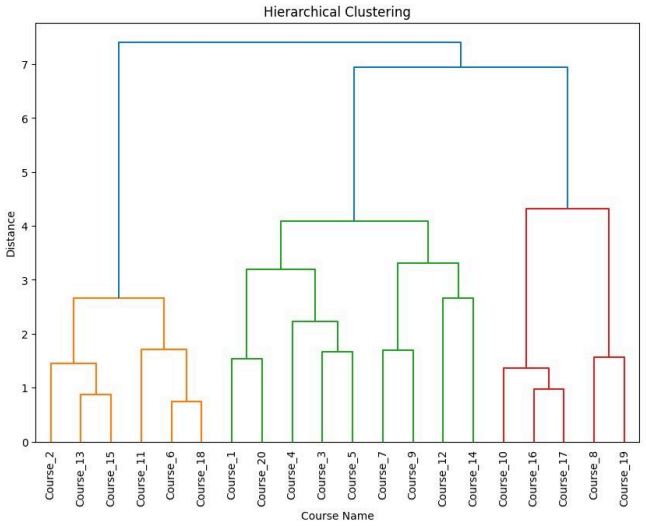
IV. RESULT AND DISCUSSION

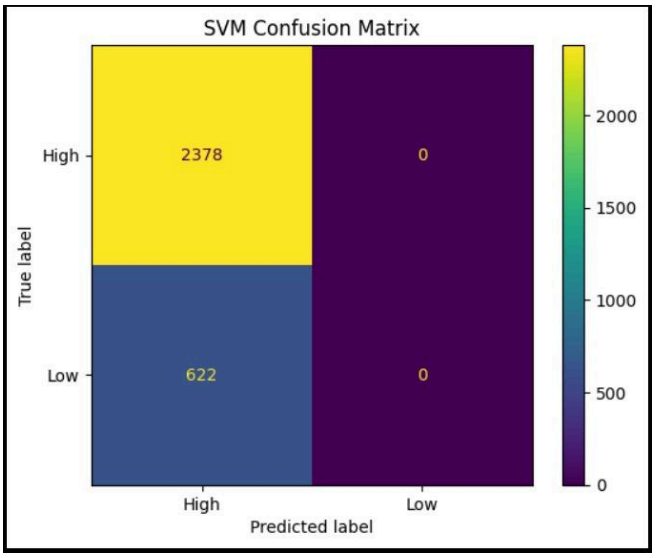
A range of patterns can be derived from the global data on the distribution of online courses in various subjects over numerous resources. Approximately, the average number of hours is around 55 and the rate of course completion is at an impressive figure of 75.1%, which indicates a decent level of participation from the learners. Course fees shown a wide disparity ranging from 10 dollars to two hundred dollars, the average course fee is around one hundred and six dollars. It seems that satisfaction is more or less the same in all places, with a mean google rating of nearly four over five. Student enrollment into the platforms like Udemy, LinkedIn Learning, edX runs into millions with Udemy being slightly ahead with total enrollments of approximately 6.5 million students. There are differences as in some countries, however, the completion rates roughly average 75% for all the platforms, with the exception of edX where the

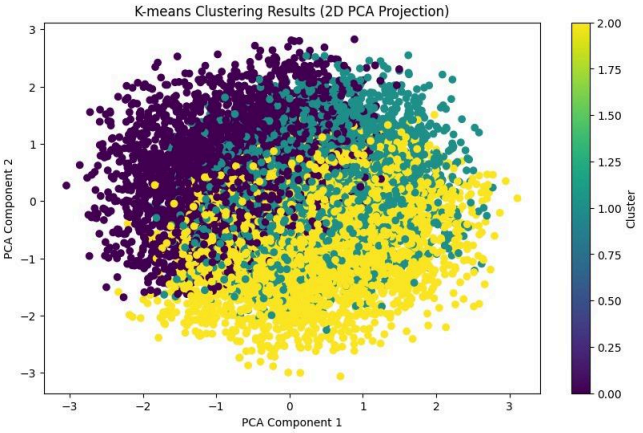
completion rate is slightly higher than that. There is also little variety in the prices charged, with most range falling between one hundred and five and one hundred and eight dollars, and which suggests some level of competition among the platforms, while the average course rating remains low at about four, which means students generally have a good experience.

On the other hand, on this course as well, the closer attention is paid to the categories of offered courses, the more interesting trends become. By contrast, the courses oriented on programming have the highest average completion rates (75.6%). Also courses, which are Data Science and Marketing, provide the highest average scores among other categories which means that such subjects not only have high enrollment but also satisfies the students registered extremely well. The pricing also differs from one category to another; quite high average prices such as $110 are found in areas like Data Science and Programming, which may well be due to both factors: the nature of the content and the skills offered are highly sought after in the market.

The information available on course analytics databases offers substantial resources for the development and training of machine working models to assess users engagement patterns and course effectiveness. Structured data on, course length, number of enrollments, completion rates, information about the platform, pricing and ratings is available for all of those courses developed and presented giving room for comprehensive data analysis. The data set helps to analyze algorithms to learned content optimization and strategic pricing as well as factors that enhance student engagement and satisfaction. The analysis of such variables also in turn assists in machine learning methods like regression analysis forecasting the course ratings or other successful course offerings on the given platform. Also, because the data includes both categorical and continuous data, and therefore the possibilities of employing simple and complex predictive models using supervised and unsupervised learning techniques is very high. Implementing such gut feeling measures through the use of data would help in offering more personalized elearning experiences and even bringing about tactical overture towards the elearning business. Simply put, the use of this dataset in conjunction with machine learning can help revolutionize education in a win-win situation for education service providers and consumers.







References

1. **Machine Learning for course analytics** mines machine learning algorithms, including regression models and neural networks, for accurate crop price predictions. https://ieeexplore.ieee.org/document/8955065
2. **Review of course Price Forecasting Methods Using Machine Learning** An overview of machine learning methods applied to crop price forecasting, focusing on recent trends, challenges, and model comparisons. https://www.sciencedirect.com/science/article/pii/S2352914819300888
3. **Application of Machine Learning Models to Analytics Market Analysis** This paper details the use of regression, classification, and clustering techniques foranalyzing agricultural market trends and price predictions. https://www.frontiersin.org/articles/10.3389/fpls.2019.01443/full
4. **Machine Learning in course analysis: A Review** This review covers a wide range of machine learning applications in agriculture, including crop price prediction and yield forecasting. https://link.springer.com/article/10.1007/s00170-018-3075-7