Project Title: EDU Predict Tool

Under the guidance of

Dr. Ardiana Sula (Project Advisor)

Professor of MS in DS Department

University of New Haven

A project report submitted of the degree of

Master of Science in Data Science

Project7_Team03 - Team Members:

Mavuri Rahul Kumar

Aryanadh Chowdary Kommineni

Venkata Chanakya Samsani

Janani Ganji

Uday Shankar Agasti

Vamsi Krishna Jammigumpula

Satya Sai Sri Nikhil

TABLE OF CONTENTS

General Description:

The "Edu Predict Tool" is an AI-driven forecasting system designed to predict international student enrollment in U.S. universities. It uses historical data and machine learning models like Light GBM, XGBoost, ARIMA, and LSTM for accurate projections. The tool provides scenario-based insights under baseline, optimistic, and pessimistic conditions. An interactive Power BI dashboard enables universities to visualize and plan for future enrollment trends. The project follows the CRISP-DM methodology, ensuring structured data handling, modeling, and evaluation. Key features include enrollment forecasting, trend analysis, and policy impact simulations. Team members specialized in data engineering, machine learning, testing, and visualization development. Strategic insights support resource allocation, admissions planning, and policymaking. The system promotes flexible, data-driven decisions for higher education institutions. Overall, the project successfully meets deadlines and adapts based on continuous feedback.

Executive Summary:

The Edu Predict Tool is an AI-powered solution designed to forecast international student enrollment trends in U.S. universities. Leveraging machine learning models such as Light GBM, XGBoost, ARIMA, and LSTM, it delivers accurate, scenario-based predictions. The project aims to help universities plan resources and strategies under baseline, optimistic, and pessimistic conditions. Historical enrollment data was collected, cleaned, and modeled following the CRISPDM framework. An interactive Power BI dashboard was developed for real-time visualization and decision-making. Key drivers like visa policies, economic trends, and academic fields were analyzed to enhance predictive insights. The tool supports strategic planning in admissions, staffing, and budgeting. The team collaborated across data engineering, analysis, testing, and dashboard development to ensure success. Continuous user feedback shaped improvements, ensuring the system's usability and accuracy. The project is on schedule and ready to support data driven decision-making in higher education.

Project Timeline:

Week	Deliverable	Due Date	Status	Completion Date
Week of Feb 23	1. Project Kickoff and Initial Planning: Finalize the project plan, including roles and responsibilities.	Mar 2, 2025	Completed	Mar 2, 2025
Week of Mar 3	2. Data Collection and Preprocessing: Gather and preprocess data relevant to the project.	Mar 9, 2025	Completed	Mar 10, 2025
Week of Mar 10	3. Model Training and Refinement: Begin training and refining the model with collected data.	Mar16, 2025	Completed	Mar 16, 2025
Week of Mar 17	4. Development of User Interface for Model Application: Start developing the user interface.	Mar 23, 2025	Completed	Mar 23, 2025
Week of Mar 24	5. Model Testing and Feedback Incorporation: Conduct initial testing and gather feedback.	Mar 30, 2025	Completed	Mar 30, 2025
Week of Mar 31	6. First Round of User Testing and Feedback: Complete the application and collect feedback.	Apr 6, 2025	Completed	Apr 6, 2025
Week of Apr 7	7. Implementation of Feedback and Improvements: Apply feedback to the model and UI.	Apr 13, 2025	Completed	Apr 13, 2025
Week of Apr 14	8. Final Model Testing and Quality Assurance: Ensure reliability and accuracy of the model.	Apr 20, 2025	Completed	Apr 20, 2025
Week of Apr 21	9. Documentation and Training Material Development: Create detailed documentation and training materials.	Apr 27, 2025	Completed	Apr 27, 2025
Week of Apr 23	10. Final Submission and Project Presentation: Submit the project, including the AI model, application, and supporting documents.	May 2, 2025	Completed	May 2, 2025
Week of May 3	Capstone Final Presentations	May 5, 2025	Completed	May 5, 2025

Objective:

The primary objective of the EDU Predict Tool project is to develop an AI-powered forecasting system that accurately predicts international student enrollment trends in U.S. higher education. By using historical enrollment data, machine learning models, and scenario-based forecasting (Baseline, Optimistic, and Pessimistic), the tool aims to provide universities with actionable insights for strategic planning. The project also seeks to create an interactive Power BI dashboard for real-time data visualization, supporting informed decision-making regarding resource allocation, admissions strategies, and policy adjustments.

Abstract:

The EDU Predict Tool is an AI-powered forecasting system developed to help American universities anticipate international student enrollment trends, which are influenced by global events, economic shifts, and visa policies. Using historical enrollment data, advanced machine learning models (Light GBM, XGBoost, Random Forest, ARIMA, ETS, and LSTM), and an interactive Power BI dashboard, the project offers scenario-based predictions: Baseline (current trends), Optimistic (favorable conditions), and Pessimistic (restrictive conditions). Data preprocessing included cleaning, normalization, encoding, and an 80/20 training-testing split to ensure high model performance, with Light GBM achieving a 91.37% accuracy rate. The tool's visualizations, designed in Power BI, allow real-time interaction and provide universities with strategic insights for better resource allocation, budgeting, and policymaking. Following the CRISP-DM methodology, the project was systematically developed through stages of business understanding, data preparation, modeling, evaluation, and deployment. Strategic insights from the EDU Predict Tool help institutions plan for multiple futures, addressing potential enrollment growth or decline. Through this project, the team strengthened skills in machine learning, data engineering, dashboard development, and agile collaboration, demonstrating the powerful role of AI and data-driven visualization in supporting decision-making within higher education environments.

Introduction:

The EDU Predict Tool project was initiated to address the pressing need for accurate, data-driven forecasting of international student enrollment in American higher education institutions. With enrollment trends heavily impacted by unpredictable factors such as global events, economic fluctuations, and immigration policy changes, universities face significant challenges in strategic planning and resource management. This project leverages historical enrollment data, advanced machine learning techniques (including Light GBM, XGBoost, Random Forest, ARIMA, ETS, and LSTM models), and an interactive Power BI dashboard to build a comprehensive forecasting system capable of generating Baseline, Optimistic, and Pessimistic scenarios. By systematically collecting, cleaning, and analyzing enrollment data, and splitting it into training and testing sets, the team developed models with high predictive accuracy to support robust scenario analysis. The integration of model outputs into a user-friendly Power BI interface allows for real-time insights and scenario-based decision-making, empowering universities to better anticipate future enrollment patterns and adapt their policies, budgets, and infrastructure plans accordingly. Following the CRISP-DM methodology, this project not only enhances forecasting accuracy but also demonstrates the vital role of machine learning, time-series analysis, and data visualization in higher education planning, ultimately equipping institutions to navigate uncertainty with greater confidence.

Domain Introduction

The domain of this project lies within higher education analytics and predictive modeling, focusing specifically on international student enrollment in American universities. International student trends are influenced by complex and dynamic factors such as global political events, economic conditions, immigration policies, and shifts in academic demand. Accurate forecasting in this domain is essential for universities to manage financial planning, infrastructure development, staffing, and policy formulation. By combining data science techniques like machine learning, time-series forecasting, and data visualization, the project addresses the growing need for datadriven decision-making in higher education. The EDU Predict Tool leverages advancements in predictive analytics to help institutions anticipate future enrollment patterns, prepare for various scenarios, and make strategic choices that support long-term academic and operational goals.

Tool Overview:

- **Purpose**: Forecast international student enrollment from 2023 to 2051.
- **Models Used**: LightGBM, XGBoost, ARIMA, LSTM, ETS.
- **Forecast Modes**:
  - *Baseline*: Current trends continue
  - *Optimistic*: Favorable visa/economic conditions
  - *Pessimistic*: Restrictive policy/economic downturn
- **Visualization Tool**: Power BI Dashboard
- **Key Features**: Trend analysis, origin/funding breakdowns, scenario simulation

Scope of the Project:

The Edu Predict Tool is an AI-powered forecasting system designed to predict international student enrollment trends in U.S. higher education. It will use historical data, machine learning models, and interactive visualizations to provide insights under three scenarios:

• Baseline: Predictions based on current trends.

• Optimistic: Assumes favorable policy and economic conditions.

• Pessimistic: Accounts for restrictive policies or adverse global events.

Baseline Scenario:

The Baseline scenario in the EDU Predict Tool represents the most realistic and neutral forecast based on current trends in international student enrollment. It assumes that no major changes will occur in influencing factors such as immigration policies, global economic conditions, or international relations. Machine learning models, trained on historical enrollment data, project future student numbers by continuing the existing patterns without introducing any significant positive or negative external shocks. This scenario serves as a critical reference point for universities, allowing them to plan admissions, budget allocations, staffing, and infrastructure development based on the most probable future outcome. By providing a steady and data-driven projection, the baseline forecast helps institutions make informed, stable decisions without overestimating or underestimating future enrollment shifts.

Optimistic Scenario:

In the optimistic scenario, the EDU Predict Tool anticipates a strong increase in international student enrollment, driven by positive changes such as relaxed visa policies, expanded scholarships, and global economic growth. Forecasts suggest a 20–30% rise in enrollment by 2028 compared to baseline trends. This surge will help universities expand infrastructure, enhance diversity, and strengthen academic programs. Machine learning models incorporate favorable conditions to predict higher demand across all academic levels. Institutions can use these insights for proactive resource planning, faculty hiring, and program development. The interactive Power BI dashboards present these optimistic projections clearly, aiding strategic decisions. Universities can also advocate for student-friendly policies by showcasing these growth opportunities. Overall, the optimistic scenario highlights major potential for academic, financial, and global advancement in U.S. higher education.

Pessimistic Scenario:

In the pessimistic scenario, the EDU Predict Tool projects a significant decline in international student enrollment due to negative factors like stricter visa policies, global economic downturns, and restrictive government regulations. Enrollment numbers could drop by up to 40% compared to baseline projections by 2028. This decline would challenge universities with reduced tuition revenues, underutilized resources, and the need to revise academic offerings. Machine learning models simulate these adverse conditions to predict decreased demand across undergraduate and graduate levels. Institutions would need to focus on contingency planning, diversify their student recruitment strategies, and adjust financial forecasts. The Power BI dashboard visualizes these downward trends, helping universities anticipate risks and prepare mitigation plans. Overall, the pessimistic scenario emphasizes the importance of adaptability and resilience in higher education planning.

Problem Statement:

Predicting international student enrollment in American universities is a complex challenge influenced by a range of unpredictable factors such as global events, economic conditions, immigration policies, and political climates. Universities and policymakers often struggle to anticipate fluctuations in enrollment, leading to difficulties in resource allocation, financial planning, staffing, and infrastructure development. Without reliable forecasting tools, institutions face increased risks of over-preparation or under-preparation, which can negatively impact academic services and budget management. Therefore, there is a critical need for a data-driven, AI-powered solution that can accurately forecast international student enrollment trends, provide scenario-based projections, and support universities in making informed, strategic decisions for the future.

Task Definition:

The main tasks of the EDU Predict Tool project are structured to address the challenge of forecasting international student enrollment in U.S. universities. The first task is to collect and preprocess historical enrollment data from reliable sources, ensuring data quality through cleaning, normalization, and encoding techniques. The second task is to develop machine learning models such as Light GBM, XGBoost, Random Forest, ARIMA, ETS, and LSTM to accurately predict future enrollment trends across three different scenarios: Baseline, Optimistic, and Pessimistic. The third task is to build an interactive Power BI dashboard that integrates model outputs, allowing users to visualize, explore, and compare enrollment forecasts in real-time. Finally, continuous validation, user feedback collection, and model refinement are essential tasks to enhance the tool's accuracy, usability, and effectiveness for strategic decision-making in higher education.

Solution Overview:

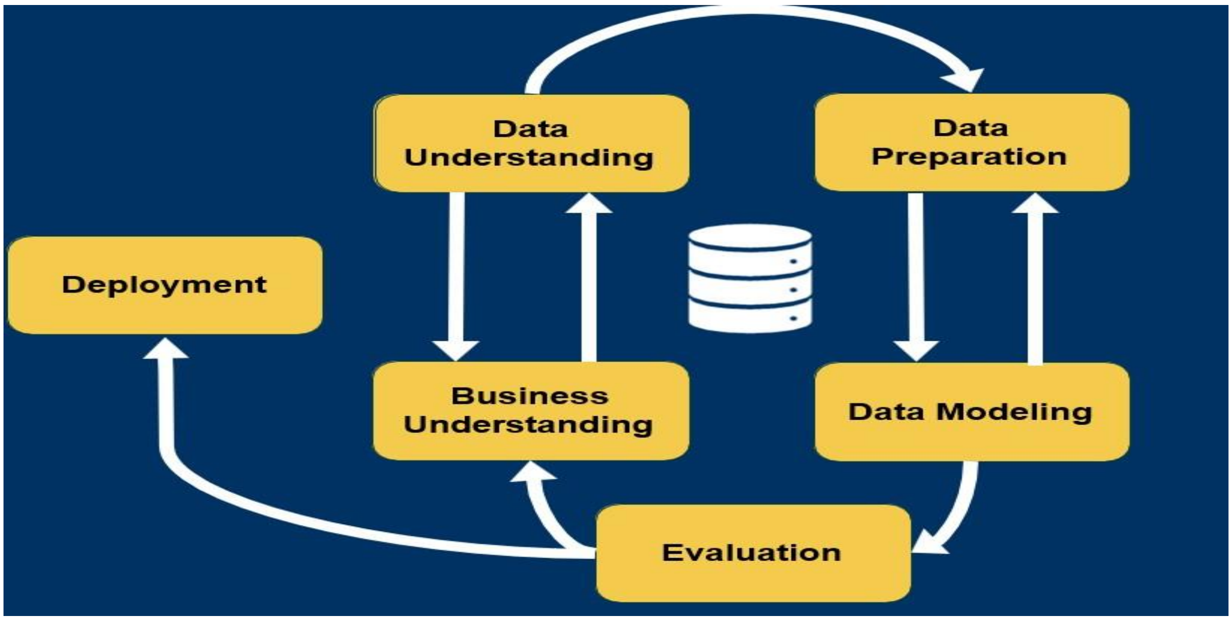
The Edu Predict Tool offers a machine learning-based solution to forecast international student enrollment trends in U.S. universities. It collects historical enrollment data, applies models like Light GBM, XGBoost, ARIMA, and LSTM, and generates accurate predictions. The tool presents three forecasting scenarios: baseline, optimistic, and pessimistic, enabling flexible strategic planning. Data preprocessing techniques like cleaning, normalization, and feature engineering were applied to improve model performance. An interactive Power BI dashboard was built to visualize trends, forecasts, and key influencing factors in real time. Scenario-based insights allow universities to anticipate policy, economic, and global shifts. The system follows CRISP-DM methodology to ensure a structured, reliable development process. Strategic insights from the tool support decision-making in admissions, budgeting, and resource allocation. Continuous user feedback helped refine model accuracy and dashboard usability. Overall, Edu Predict empowers educational institutions to plan proactively and make data-driven decisions.

To address the challenges of predicting international student enrollment, the EDU Predict Tool provides a comprehensive AI-powered solution. The project first collects and integrates historical enrollment data, ensuring it is clean, normalized, and ready for machine learning applications. It then employs advanced predictive models including Light GBM, XGBoost, Random Forest, ARIMA, ETS, and LSTM to generate accurate enrollment forecasts under three scenarios: Baseline (current trends), Optimistic (favorable conditions), and Pessimistic (adverse conditions). The tool also features an interactive Power BI dashboard, where users can easily explore and compare future enrollment projections across different scenarios. Real-time integration of model outputs into the dashboard allows universities to make data-driven strategic decisions. By combining machine learning, scenario-based forecasting, and intuitive visualization, the EDU Predict Tool empowers institutions to proactively plan for various future outcomes, optimize resource management, and adapt to changing global and policy environments.

Technical Overview:

The Edu Predict Tool integrates advanced machine learning models like Light GBM, XGBoost, ARIMA, ETS, and LSTM for enrollment forecasting. Historical enrollment data is processed through data cleaning, normalization, and categorical encoding techniques. Feature engineering incorporates external factors like visa policies and economic conditions to enhance prediction accuracy. An 80/20 training-testing split validates model performance and reliability. Light GBM achieved the highest accuracy at 91.37% for baseline forecasts. Power BI dashboards were developed for real-time visualization across baseline, optimistic, and pessimistic scenarios. Scenario forecasting enables strategic planning by simulating varying policy and economic conditions. Model outputs are directly connected to user-friendly dashboards for quick decision making. The technical design follows the CRISP-DM methodology, ensuring structured data handling, modeling, and deployment. Continuous feedback loops allow refinement of both predictive models and dashboard functionalities.

Methodology:



The development of the EDU Predict Tool followed the CRISP-DM (Cross-Industry Standard Process for Data Mining) methodology to ensure a structured and systematic approach. The project began with Business Understanding, where the primary goal was defined: to forecast international student enrollment trends and support strategic university planning. In the Data Acquisition and Preprocessing phase, historical enrollment datasets were collected from reliable sources, followed by cleaning, normalization, handling missing values, and encoding categorical features. An 80/20 split was used to divide the data into training and testing sets for model development.

During Exploratory Data Analysis (EDA) and Feature Engineering, key patterns, trends, and influencing factors such as visa policies and economic indicators were identified and transformed into meaningful features for predictive modeling. In the Model Development and Optimization stage, multiple machine learning models, including Light GBM, XGBoost, Random Forest, ARIMA, ETS, and LSTM, were trained and validated to achieve high predictive accuracy across Baseline, Optimistic, and Pessimistic scenarios. Hyperparameter tuning and model comparison were conducted to enhance performance.

Next, in the Dashboard Development phase, model outputs were integrated into an interactive Power BI dashboard, allowing users to visualize and compare scenario-based forecasts in realtime. Finally, during Validation and Refinement, user feedback was gathered, and continuous improvements were made to both the models and the dashboard to ensure usability, accuracy, and relevance. Throughout the project, emphasis was placed on maintaining data quality, model interpretability, and creating a scalable, user-centered solution to meet the forecasting needs of higher education institutions.

Technical documentation (system architecture, design decisions):

System architecture:

The EDU Predict Tool is designed with a comprehensive machine learning framework that facilitates the forecasting of international student enrollment in U.S. universities. The architecture initiates with the extraction of historical enrollment data, which is then prepared through essential preprocessing tasks such as data cleaning, normalization, and transformation of categorical variables. Contextual factors like immigration regulations and economic trends are incorporated through feature engineering to enhance the models' predictive accuracy. A variety of machine learning algorithms—including LightGBM, XGBoost, ARIMA, ETS, LSTM, and Random Forest—are employed, using an 80/20 training and testing data split to validate performance. Among these, LightGBM demonstrated the highest accuracy for baseline predictions, reaching 91.37%.

The predictive models are built to simulate three distinct scenarios: current trends (Baseline), favorable outcomes (Optimistic), and challenging conditions (Pessimistic). These forecasts are seamlessly integrated into an interactive Power BI dashboard, offering stakeholders a real-time, visual comparison of potential enrollment trajectories. The system is developed in alignment with the CRISP-DM framework, which ensures each phase—ranging from business understanding to final deployment—is methodically executed. This architectural setup not only supports transparency and accuracy in forecasting but also enables institutions to make informed, scenario-driven planning decisions with greater agility and confidence.

Design decisions:

The design of the EDU Predict Tool was guided by the need for scalability, accuracy, and user accessibility. Python was chosen as the primary programming language due to its vast ecosystem of data science libraries, including Scikit-learn, TensorFlow, Keras, LightGBM, and XGBoost. These frameworks enabled the implementation of both tree-based and time-series models, allowing flexibility across different forecasting needs. Power BI was selected for visualization because of its dynamic, user-friendly interface that supports real-time interaction with data. The system also utilizes SQL-based databases for structured data storage and retrieval, ensuring efficient data management. An 80/20 training-to-testing split was employed to ensure reliable model validation, and feature engineering incorporated and economic variables to improve forecast relevance.

Performance optimization and ease of use were central to the system's design. Hardware requirements, including at least 16 GB of RAM and a modern GPU like the NVIDIA RTX 3060, were defined to handle model training and visualization without latency. To maintain model reliability, continuous integration of user feedback and model refinement loops were implemented. Git and GitHub supported version control, while optional tools like Trello or Jira facilitated team collaboration. A major design decision was to simplify the dashboard interface for non-technical users, ensuring key insights could be accessed and interpreted quickly. These thoughtful design choices allowed the EDU Predict Tool to remain robust, responsive, and aligned with institutional needs in higher education planning.

Setup instructions and environment requirements:

Setup Instructions:

Ensure the machine meets the following minimum specs:

- **OS**: Windows 10/11 or macOS (latest)
- **RAM**: Minimum 16 GB (32 GB recommended)
- **Storage**: At least 512 GB SSD
- **GPU**: NVIDIA GTX 1660 Ti or better (RTX 3060 recommended for deep learning models)
- **Python Version**: 3.8 or higher
- **Internet**: Required for downloading libraries and accessing cloud resources

Environmental Requirements:

To ensure smooth operation, accurate forecasting, and real-time dashboard integration of the EDU Predict Tool, the following hardware, software, and system configurations are recommended:

Hardware Requirements:

- **Processor**: Intel Core i7 / AMD Ryzen 7 or higher
- **Memory (RAM)**: 16 GB (Recommended 32 GB)

- **Storage**: 512 GB SSD (Recommended 1 TB SSD)
- **Graphics Card (GPU)**: NVIDIA GTX 1660 Ti / RTX 3060 or higher
- **Network**: High-speed internet connection
- **Display**: Full HD Monitor (1920×1080 resolution or higher)
- **External Storage**: External Hard Drive (optional)

Software Requirements:

- **Operating System**: Windows 10/11 or macOS (latest versions)
- **Programming Language**: Python 3.8 or higher
- **Development Environment**: Jupyter Notebook / VS Code
- **Libraries and Frameworks**: Scikit-learn, TensorFlow, Keras, LightGBM, XGBoost, ARIMA (statsmodels)
- **Data Visualization Tool**: Power BI Desktop
- **Database**: SQL Server / MySQL (for structured data management)
- **Version Control**: Git and GitHub
- **Project Management Tools**: Trello / Jira (optional)

Advantages:

- Provides accurate enrollment forecasting using machine learning.
- Supports scenario-based planning (baseline, optimistic, pessimistic).
- Enables data-driven decisions for resource and budget planning.
- Visualizes trends through interactive Power BI dashboards.
- Helps identify risks early and improve strategic planning.
- Easily adaptable with new data for continuous accuracy.
- Analyzes policy impacts on student enrollment trends.

- User-friendly interface for non-technical stakeholders.
- Enhances growth opportunities and institutional stability.

Applications:

- Assisting universities in forecasting international student enrollment for future academic years.
- Supporting admissions offices in planning recruitment strategies based on predicted trends.
- Helping financial departments optimize budgeting and resource allocation aligned with enrollment forecasts.
- Enabling academic planners to anticipate demand for specific programs and fields of study.
- Informing policy makers about the potential impact of visa regulations and economic conditions on student numbers.
- Assisting international offices in preparing for shifts in student demographics by continent or country.
- Helping housing and campus facilities departments predict space and infrastructure needs.
- Supporting strategic decision-making during uncertain global events (e.g., pandemics, policy changes).
- Enhancing marketing and outreach strategies to target regions or countries showing enrollment growth.
- Providing real-time, data-driven insights through dashboards to stakeholders across the institution.

Literature Review:

Existing research highlights that international student enrollment is heavily influenced by global events, economic shifts, and immigration policies. Traditional forecasting methods struggled to accurately predict enrollment changes due to the dynamic nature of these factors. Machine learning techniques like Light GBM, XGBoost, ARIMA, and LSTM have proven effective in capturing complex, non-linear patterns in educational datasets. Studies emphasize that scenario-based forecasting helps institutions prepare for multiple future possibilities. Data preprocessing, feature engineering, and time-series analysis are crucial for building reliable predictive models. Previous work also shows that visual tools like dashboards enhance stakeholder engagement and decisionmaking. CRISP-DM methodology is widely recognized for structuring data-driven projects efficiently. Research underscores the importance of continuously updating models with new data to maintain forecasting accuracy. Policy analysis integrated into models adds significant value for strategic educational planning. This project builds upon these foundations to deliver an AI-driven, user-friendly enrollment prediction tool.

Dataset:

The dataset used in the Edu Predict Tool project consists of historical international student enrollment records from reliable academic sources. It includes critical features such as funding sources, student status, country of origin, academic details, and fields of study. Data preprocessing steps involved cleaning missing values, normalizing numerical features, and encoding categorical variables for machine learning compatibility. The dataset was split into 80% training and 20% testing sets to validate model performance. Feature engineering was applied by integrating policy changes and economic conditions to strengthen prediction accuracy. Academic levels like Undergraduate, Graduate, Non-Degree, and OPT programs were analyzed separately. Trends across various continents, academic fields, and degree distributions were also incorporated. Timeseries data allowed the use of models like ARIMA, ETS, and LSTM for sequential forecasting. Consistency and reliability of data were ensured before model training. This well-structured dataset served as the foundation for accurate and scenario-based enrollment forecasting.

Main Approach:

We propose the Edu Predict Tool, an AI-powered model for forecasting international student enrollment in U.S. universities. The model uses historical data as input, including visa status, source country, academic field, and funding. Data is preprocessed (cleaning, normalization, encoding), and split into training (80%) and testing (20%) sets. Multiple machine learning algorithms (LightGBM, XGBoost, ARIMA, LSTM, ETS, Random Forest) predict future enrollment under Baseline, Optimistic, and Pessimistic scenarios. LightGBM is the primary model, achieving 91.37% baseline accuracy. Features like visa policies and economic trends are key variables. Outputs are scenario-based enrollment forecasts visualized in interactive Power BI dashboards. For example, a 20% visa policy improvement scenario predicts a sharp enrollment rise by 2028. This model enables universities to plan resources, policies, and admissions strategies flexibly.

Evaluation Metric:

We evaluate the model’s success using both quantitative and qualitative metrics. Quantitatively, we use Mean Absolute Error (MAE) and Mean Squared Error (MSE) to assess prediction accuracy, where:

- MAE = $\frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$
- MSE = $\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$

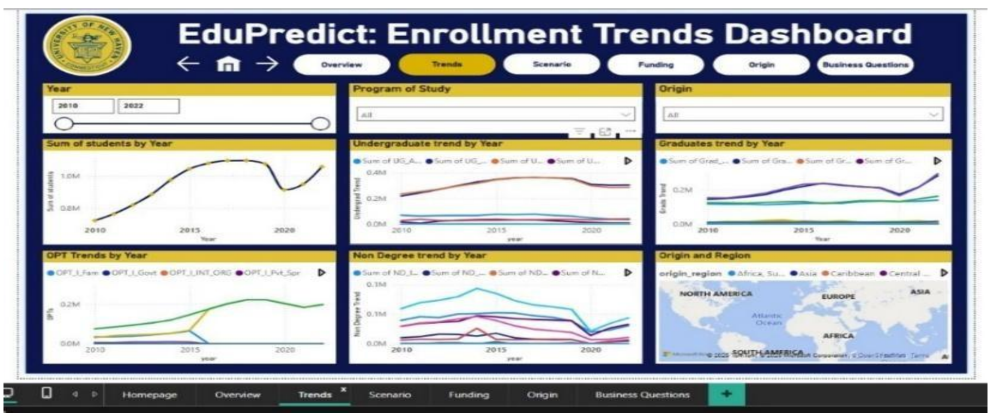
Higher performance is indicated by lower MAE and MSE values. LightGBM achieved a **91.37% accuracy** for baseline predictions. Forecast visual alignment with actual trends is qualitatively assessed through scenario testing (Baseline, Optimistic, Pessimistic). User feedback on dashboard usability and scenario realism also serves as a qualitative metric. Stability across different global policy scenarios further validates model robustness.

Dashboard Overview:

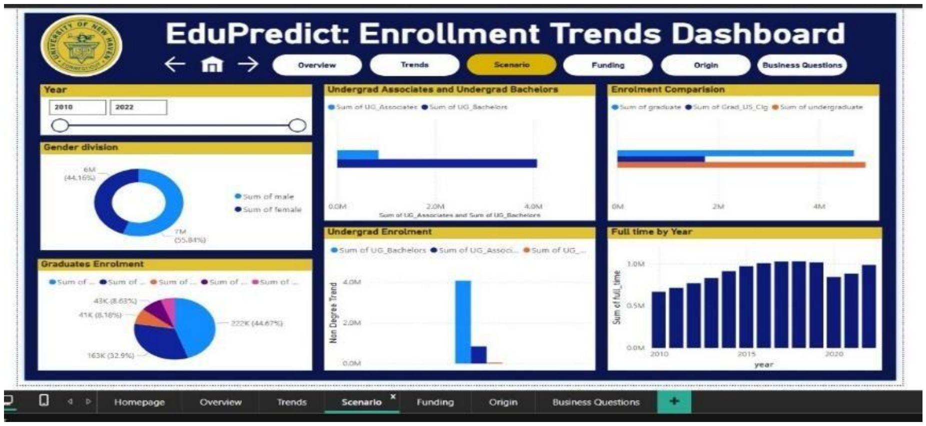
This image shows the homepage of the "EduPredict: Enrollment Trends Dashboard" developed for the University of New Haven's Tagliatela College of Engineering.

The dashboard includes navigation buttons for Overview, Trends, Scenario, Funding, Origin, and Business Questions.

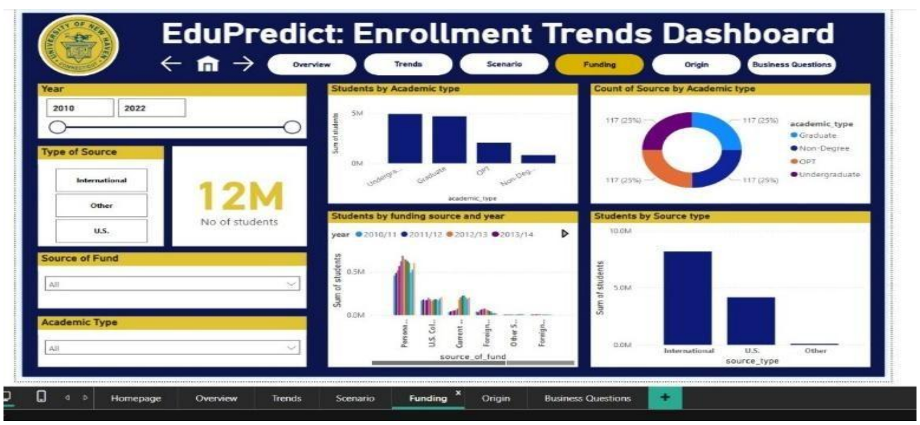
This image shows the "EduPredict: Enrollment Trends Dashboard" from the University of New Haven. It tracks international student data from 2010 to 2022, including visa types, gender ratio, and students' origin regions. Key figures include 12 million total students and 11 million with F1 visas. Various charts visualize female enrollment trends and visa distribution over time.



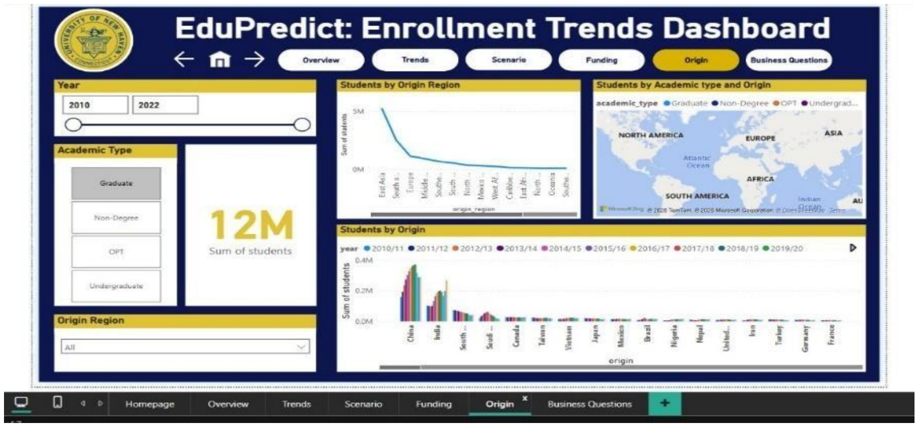
This image shows the "Trends" section of the EduPredict: Enrollment Trends Dashboard from the University of New Haven. It tracks various enrollment metrics from 2010 to 2022, including the total number of students, undergraduate and graduate trends, OPT (Optional Practical Training) trends, and non-degree trends by year. The graphs show a steady increase in student enrollment until a drop around 2020, then a recovery afterward. Different colored lines represent various categories within each trend. A world map at the bottom right displays student origins by region, like Asia, Europe, and Africa.



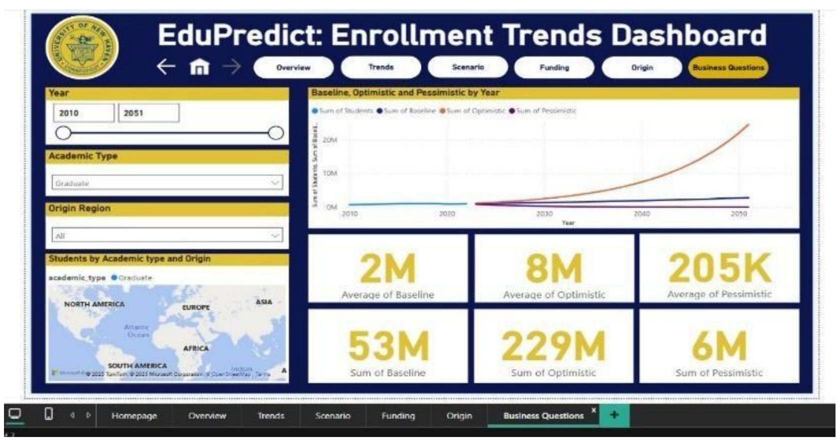
This image shows the "Scenario" section of the EduPredict: Enrollment Trends Dashboard from the University of New Haven. It highlights gender division, showing 56% female and 44% male students. Undergraduate enrollment is mainly dominated by bachelor's degrees compared to associate degrees. The enrollment comparison shows the number of graduate and undergraduate students side by side. Graduate enrollment is detailed with a pie chart, and a bar chart tracks fulltime enrollment trends from 2010 to 2022, showing a peak around 2017.



This image shows the EduPredict: Enrollment Trends Dashboard for the University of New Haven. It tracks student enrollment from 2010 to 2022, categorizing data by academic type, source type (International, U.S., Other), and funding source. Visualizations include bar charts, pie charts, and a total student count of 12 million. The dashboard allows filtering based on year, source of fund, type of source, and academic type. Navigation tabs at the top (Overview, Trends, Scenario, Funding, Origin, Business Questions) suggest a detailed analysis capability.



This image shows the EduPredict: Enrollment Trends Dashboard for the University of New Haven. It analyzes student enrollment data from 2010 to 2022, focusing on academic type (Graduate, NonDegree, OPT, Undergraduate) and origin region. The dashboard features graphs showing the number of students by region and origin country, along with a global map visual. The total enrollment displayed is 12 million students. Navigation tabs at the top allow switching between Overview, Trends, Scenario, Funding, Origin, and Business Questions sections.



This image shows the Business Questions section of the EduPredict: Enrollment Trends Dashboard for the University of New Haven. It projects student enrollment trends from 2010 to 2051 under Baseline, Optimistic, and Pessimistic scenarios. The line graph illustrates expected growth, with the optimistic case rising sharply by 2050. Key metrics like averages and total sums for each scenario are displayed below the graph. A map also shows student distribution by academic type and origin region for Graduate students.

Dashboard Navigation:

Main Sections:

- **Overview**: General trends and total enrollments
- **Trends**: Line graphs showing historical and projected data
- **Scenario**: Compare Baseline, Optimistic, and Pessimistic forecasts
- **Funding**: Breakdown by funding source and type
- **Origin**: Geographic visualization by continent and country
- **Business Questions**: Strategic insights addressing institutional planning

Use the top navigation menu to switch sections. Filters are available for year, program type, region, and funding source.

Interpretation of Results:

Key Visual Elements:

- **Line Graphs**: Represent historical + projected enrollments
- **Pie Charts**: Display degree types and gender ratios
- **Maps**: Show origin country/region of students
- **Bar Charts**: Illustrate trends by funding type or academic level

Scenario Comparison:

Use this to:

- Evaluate risk under pessimistic conditions
- Plan resources under optimistic growth
- Align with the most probable baseline forecast

Python Notebooks (with Clear Annotations for Each MVP Feature):

To support the development and deployment of the EDU Predict Tool, a series of Python notebooks were created, each dedicated to implementing and testing a core MVP feature. These notebooks were thoroughly annotated to ensure clarity, reproducibility, and ease of collaboration. Below is a summary of the primary notebook components and their roles in the system.

Data Collection and Preprocessing Notebook

This notebook focused on importing raw enrollment datasets from Excel and CSV files. It included steps for handling missing values, normalizing numerical fields, encoding categorical variables (e.g., visa status, degree type), and formatting date columns for time-series modeling. Annotations explained each preprocessing method and why it was applied (e.g., MinMaxScaler for normalization, OneHotEncoder for categorical data). This ensured clean, consistent input for all modeling notebooks.

Exploratory Data Analysis (EDA) Notebook

The EDA notebook provided visual summaries and statistical insights into the historical enrollment data. It featured time-series plots, correlation heatmaps, trend lines by academic level, and geographic distribution graphs. Each visualization included comments on the observed trends and anomalies. The EDA process helped identify key variables for feature engineering and guided model selection based on data behavior.

Feature Engineering Notebook

This notebook focused on creating lag features, rolling averages, and derived indicators such as “years since policy change” or “economic index lag.” These features enhanced model performance by providing contextual understanding of temporal patterns. Explanatory comments detailed the logic behind each transformation and how it contributed to predictive accuracy across scenarios.

Model Development Notebook (LightGBM & XGBoost)

Separate sections were created for LightGBM and XGBoost. Each section included model training with hyperparameter tuning, cross-validation using TimeSeriesSplit, and evaluation using MAE, RMSE, and R^2 scores. Annotations explained parameter choices (e.g., number of leaves, learning rate), model evaluation metrics, and early stopping techniques to avoid overfitting.

Time-Series Models Notebook (ARIMA, ETS, and LSTM)

This notebook implemented time-series forecasting models including ARIMA, ETS, and LSTM. The ARIMA section covered differencing and stationarity testing (ADF test), while ETS focused on trend and seasonality modeling. The LSTM section utilized TensorFlow/Keras and included architecture definition, sequence generation, and training. Detailed markdown cells provided step-by-step explanations of each method's purpose and assumptions.

Scenario Simulation Notebook

To simulate Baseline, Optimistic, and Pessimistic scenarios, this notebook adjusted key features such as visa approval rates or economic indicators. For instance, in the optimistic scenario, favorable policy variables were boosted, and the models were retrained or reused to generate alternate forecasts. Annotations described how these changes impacted predictions and supported strategic planning.

Forecast Export and Power BI Integration Notebook

This notebook prepared final forecast outputs by merging predicted and historical data, formatting them into Power BI-compatible CSVs. It also added confidence intervals for each scenario, allowing better visualization. Comments explained how to match data structure requirements of Power BI and ensure smooth dashboard updates.

Model Comparison and Evaluation Notebook

This notebook compared all trained models based on performance metrics and computational efficiency. It included bar charts summarizing MAE and R^2 scores across models and provided recommendations on the best models for each scenario. Annotated conclusions guided final model selection and deployment.

Each Python notebook in the EDU Predict Tool project was crafted with clarity and usability in mind, enabling team members and stakeholders to understand the workflow, reproduce results, and iterate effectively on the forecasting system.

<u>Power BI User Guidelines for EDU Predict Tool:</u>

1. **Open Power BI Desktop**
   Launch Power BI Desktop on your system.

2. **Load Forecast Data**
   Use the “Get Data” option to import the forecast .csv files exported from Python notebooks.

3. **Select Data Source**
   Choose **CSV**, **Excel**, or **SQL Server** depending on your data source. Browse and load the relevant file

4. **Transform Data (if needed)**
   Click on “Transform Data” to open Power Query Editor. Apply necessary data shaping steps like data type adjustments or column renaming.

5. **Apply Changes**
   After transformation, click “Close & Apply” to load the data into Power BI’s data model.

6. **Navigate Dashboard Tabs**
   Use the provided Power BI report file or manually recreate visualizations:

   - **Overview**: Displays total enrollments and summary trends
   - **Trends**: Line charts for historical and predicted values
   - **Scenario**: Compare Baseline, Optimistic, and Pessimistic projections
   - **Funding**: Visuals by funding source and type
   - **Origin**: Student origin by country and region
   - **Business Questions**: Strategic scenario insights

7. **Apply Filters**
   Use filter panes or slicers to refine data views by:

   - Year range
   - Academic level
   - Region or country
   - Funding type

8. **Interpret Visuals**
    - **Line Graphs**: Enrollment growth over time
    - **Pie Charts**: Gender or program distribution
    - **Maps**: Geographic student distribution
    - **Bar Charts**: Enrollment by program, year, or funding
9. **Refresh Data**
   If the source file is updated, click "Refresh" to update visualizations.
10. **Save and Share**
    Save the report, share via Power BI Service for web access, or export visuals as PDF/images if needed.

Lessons learned:

Summary of Challenges and How We Overcame Them:

During the development of the EDU Predict Tool, one of the most significant challenges was integrating complex machine learning model outputs into an interactive and real-time Power BI dashboard. The project involved multiple forecasting models (LightGBM, XGBoost, ARIMA, LSTM, etc.) across three different scenarios—Baseline, Optimistic, and Pessimistic. Ensuring consistent formatting, aligning outputs across models, and synchronizing them with Power BI's data requirements proved to be technically demanding.

Additionally, the need to maintain high model accuracy while keeping the dashboard responsive and intuitive introduced further complexity. Another major challenge was making the insights accessible to non-technical users without compromising the depth of analysis. The models incorporated nuanced features such as visa policy impacts, funding types, and regional trends. Translating these variables into clear, visual insights required thoughtful dashboard design, intuitive navigation, and simple yet meaningful visual storytelling.

Balancing the sophistication of machine learning outputs with the ease of interpretation for decision-makers was a recurring obstacle throughout development.

To overcome these challenges, the team adopted the CRISP-DM methodology for a structured approach and relied heavily on official documentation, academic research, and user feedback. Regular validation cycles and iterative refinements helped enhance both model performance and dashboard usability. Collaborative tools like GitHub and Power BI enabled streamlined version control and visualization. Through continuous testing and agile development, the team not only addressed the technical hurdles but also gained valuable skills in real-time data integration, scenario-based forecasting, and user-centered design. This ensured that the final product was both technically sound and practically impactful.

Summary of Iterations:

The development of the EDU Predict Tool followed an iterative process aligned with agile principles and the CRISP-DM methodology. Each phase of the project—data collection, model development, visualization, and deployment—underwent multiple refinement cycles based on performance testing and stakeholder feedback. Initial iterations focused on building baseline forecasting models using historical enrollment data. Once core models like LightGBM and XGBoost were implemented, the team evaluated their accuracy, adjusted hyperparameters, and introduced additional models such as ARIMA and LSTM for improved time-series forecasting.

Subsequent iterations emphasized integration of model outputs into the Power BI dashboard. Early versions of the dashboard displayed static insights, which were refined over time to support real-time scenario switching and user interactivity. User feedback from initial testing sessions was vital in enhancing the dashboard's navigation, clarity, and responsiveness. The team also iterated on scenario logic—adjusting assumptions and outputs for baseline, optimistic, and pessimistic projections to ensure realistic and actionable insights.

In the final stages, the team conducted multiple rounds of validation, focusing on model robustness, dashboard usability, and visual clarity. Adjustments were made to improve visual consistency, streamline filters, and ensure cross-scenario comparisons were easy to interpret. Each iteration brought the tool closer to its goal: a reliable, intuitive, and data-driven decision-support system for university planning. This iterative approach not only improved the final product but also fostered a culture of continuous improvement and collaboration within the team.

Helpful Resources:

Several resources played an essential role in overcoming technical and design challenges. The CRISP-DM methodology provided a structured framework for handling each project phase, from data preparation to model deployment. Official documentation and tutorials for LightGBM, XGBoost, Power BI, and time-series models (ARIMA, LSTM, ETS) were instrumental in model building and optimization. Academic articles on international student enrollment trends and initial user feedback sessions helped refine both the forecasting logic and the user experience of the dashboard.

New Skills Used:

This project allowed us to apply and strengthen multiple new skills. We developed advanced machine learning modeling techniques using LightGBM, XGBoost, ARIMA, and LSTM models tailored for scenario-based forecasting. We also gained experience in real-time dashboard creation with Power BI, enabling dynamic visual storytelling of enrollment trends. Strategic impact analysis based on external factors like visa policies and global economic conditions was another key competency developed, alongside agile team communication and iterative development practices for continuous improvement.

Future Use Cases:

The EDU Predict Tool, while currently focused on forecasting international student enrollment, has the potential to be expanded and adapted for a variety of future applications in the education sector. One key use case is domestic student enrollment forecasting, where similar machine learning models can analyze trends across states, demographics, or academic disciplines. This would support public universities and policymakers in addressing regional education demands, budgeting, and infrastructure planning.

Another future application is program-level demand forecasting, enabling institutions to predict enrollment trends for specific degree programs (e.g., engineering, business, healthcare). Such insights can guide academic departments in curriculum development, faculty hiring, and marketing strategies. Additionally, the tool can be extended to financial planning and tuition revenue forecasting, helping finance offices model budget projections based on various enrollment scenarios and funding sources.

The platform can also evolve into a real-time policy impact simulator, where institutions model the effects of potential changes in immigration policies, scholarship programs, or global events (like pandemics or geopolitical shifts). Integration with live datasets (e.g., visa approvals, economic indicators) could make the dashboard dynamic and self-updating. Over time, the EDU Predict Tool could become a comprehensive, AI-driven decision-support system for strategic planning across admissions, academics, finance, and international relations within higher education.

Appendix:

XG Boost import

pandas as pd import

numpy as np import

xgboost as xgb

from sklearn.model_selection import TimeSeriesSplit from

sklearn.metrics import mean_absolute_error, r2_score from

sklearn.preprocessing import StandardScaler import

matplotlib.pyplot as plt from datetime import datetime

Prepare the data with enhanced features

student_data = academic.groupby('year')['students'].sum().reset_index() student_data['year_num'] = student_data['year'].dt.year

student_data['year_rank'] = np.arange(len(student_data)) # Sequential ranking for time student_data['years_since_start'] = student_data['year_num'] - student_data['year_num'].min()

Create lag features (critical for time series)

student_data['prev_year'] = student_data['students'].shift(1)

student_data['rolling_avg_3yr'] = student_data['students'].rolling(3).mean()

student_data.dropna(inplace=True) # Remove rows with missing lag values

Features and target

```
X = student_data[['year_rank', 'years_since_start', 'prev_year', 'rolling_avg_3yr']] y
= student_data['students']

# Time-series cross validation tscv
= TimeSeriesSplit(n_splits=3)
scaler = StandardScaler()


# XGBoost model configuration params
= {
    'objective': 'reg:squarederror',
    'n_estimators': 1000,
    'learning_rate': 0.01,
    'max_depth': 4,
    'subsample': 0.8,
    'colsample_bytree': 0.8,
    'gamma': 1,
    'random_state': 42,
    'early_stopping_rounds': 50
}


# Store evaluation metrics metrics
= []


for train_index, test_index in tscv.split(X):
```

```
    X_train,    X_test    =    X.iloc[train_index],    X.iloc[test_index]
y_train, y_test = y.iloc[train_index], y.iloc[test_index]

    # Scale features
    X_train_scaled = scaler.fit_transform(X_train)
    X_test_scaled = scaler.transform(X_test)

    # Train XGBoost     model =
xgb.XGBRegressor(**params)
model.fit(X_train_scaled, y_train,
eval_set=[(X_test_scaled, y_test)],
verbose=False)

    # Evaluate
    y_pred              =              model.predict(X_test_scaled)
metrics.append({
        'MAE': mean_absolute_error(y_test, y_pred),
        'R2': r2_score(y_test, y_pred)
    })
# Print average performance avg_mae =
np.mean([m['MAE'] for m in metrics]) avg_r2 =
np.mean([m['R2'] for m in metrics])
print(f"\nAverage Cross-Validation Performance:")
print(f"MAE: {avg_mae:.2f}") print(f"R2:
{avg_r2:.2f}")
```

```
# Train final model on all data (with eval_set for early stopping)
X_scaled = scaler.fit_transform(X) final_model =
xgb.XGBRegressor(**params)
# Assuming you want to use a portion of your data for validation (e.g., the last 20%)
# Adjust the split ratio as needed
X_train_final, X_val_final, y_train_final, y_val_final = train_test_split(
    X_scaled, y, test_size=0.2, shuffle=False  # Important: shuffle=False for time series
)

final_model.fit(X_train_final, y_train_final,
eval_set=[(X_val_final, y_val_final)],
verbose=False)

# Generate future predictions
last_known_year = student_data['year_num'].max()
future_years = np.arange(last_known_year + 1, last_known_year + 6)

# Create future feature matrix future_data
= pd.DataFrame({
    'year_num': future_years,
    'year_rank': np.arange(len(student_data), len(student_data) +
len(future_years))[len(future_years):],
    'years_since_start': future_years - student_data['year_num'].min(),
    'prev_year': [student_data['students'].iloc[-1]] * len(future_years),
```

```
    'rolling_avg_3yr': [student_data['students'].iloc[-3:].mean()] * len(future_years)
})

future_X = future_data[['year_rank', 'years_since_start', 'prev_year', 'rolling_avg_3yr']]
future_X_scaled = scaler.transform(future_X)

future_predictions = final_model.predict(future_X_scaled)


# Plot results with enhanced visualization plt.figure(figsize=(12, 6))
plt.plot(student_data['year_num'], y, 'bo-', label='Actual')
plt.plot(student_data['year_num'], final_model.predict(X_scaled), 'ro-', label='Fitted')
plt.plot(future_years, future_predictions, 'go--', label='Forecast')
plt.fill_between(future_years,
                future_predictions * 0.9,  # 10% lower bound
future_predictions * 1.1,  # 10% upper bound
color='green', alpha=0.1, label='Uncertainty Band')
plt.xlabel('Year', fontsize=12) plt.ylabel('Student Count',
fontsize=12)
plt.title('Student Enrollment Forecasting with XGBoost', fontsize=14)
plt.legend() plt.grid(True, alpha=0.3) plt.show()


# Feature importance
xgb.plot_importance(final_model) plt.show()
```

Print forecast results forecast_df

= pd.DataFrame({

'Year': future_years,

'Predicted_Students': future_predictions,

'Lower_Bound': future_predictions * 0.9,

'Upper_Bound': future_predictions * 1.1

}) print("\n5-Year Enrollment

Forecast:") print(forecast_df.round(0))

Assuming X_test has 'year_num' (or similar) from your XGBoost data prep:

#X_test = student_data[student_data['year_num'].isin(future_years)] #This results in an empty dataframe.

Instead, use the validation set you created earlier

X_test_final = X_val_final

Now, predict using XGBoost on the correct X_test

#X_test = student_data[student_data['year_num'].isin(future_years)] #Commented out to avoid confusion

Add predictions for XGBoost (not Random Forest)

Predicting on the scaled validation set (X_val_final)

y_xgb_pred = final_model.predict(X_val_final)

... Assuming X_val_final corresponds to y_val_final

Create a dataframe for the result to write to file

X_test_final_df = pd.DataFrame(X_val_final)

X_test_final_df['students_pred'] = y_xgb_pred

#results = pd.concat([X_test, y_test.reset_index(drop=True)], axis=1) #This would also result in an error due to the mismatched indices.

results = X_test_final_df.copy()

Export to CSV for Power BI

results.to_csv('predicted_student_data.csv', index=False)

merged_df.to_csv('processed_student_data.csv', index=False)

Light GBM

import lightgbm as lgb

from sklearn.model_selection import TimeSeriesSplit

from sklearn.metrics import mean_absolute_error, mean_absolute_percentage_error import

pandas as pd # Import pandas

Load the data (if not already loaded)

Assuming 'academic' is the original DataFrame you loaded student_data =

pd.read_excel('/content/academic.xlsx') # Load or create student_data DataFrame here

student_data['year'] = pd.to_datetime(student_data['year'].astype(str).str[:4]) # Convert 'year' to

datetime

Feature Engineering for Time Series

student_data = student_data.reset_index(drop=True) # Drop the existing index to avoid the error

student_data['year_num'] = student_data['year'].dt.year

```
student_data['time_idx'] = range(len(student_data))  # Sequential index student_data['lag_1']
= student_data['students'].shift(1)  # 1-year lag student_data['rolling_mean_3'] =
student_data['students'].rolling(3).mean()  # 3-year moving average
student_data.dropna(inplace=True)  # Remove rows with missing lag values
# ... (Rest of your LightGBM code)

# Prepare features and target
X = student_data[['time_idx', 'year_num', 'lag_1', 'rolling_mean_3']] y
= student_data['students']

# Time Series Cross Validation tscv
= TimeSeriesSplit(n_splits=3)
metrics = []

for train_idx, test_idx in tscv.split(X):
    X_train,    X_test    =    X.iloc[train_idx],    X.iloc[test_idx]
y_train, y_test = y.iloc[train_idx], y.iloc[test_idx]

    # LightGBM Dataset
    train_data     =     lgb.Dataset(X_train,     label=y_train)
valid_data = lgb.Dataset(X_test, label=y_test)

    #     Model     Parameters
params = {
        'objective': 'regression',
```

```
        'metric': 'mape',
        'boosting_type': 'gbdt',
        'num_leaves': 8,
        'learning_rate': 0.05,
        'feature_fraction': 0.8,
        'verbose': -1
    }
    # Train model     model =
lgb.train(params,
            train_data,
            valid_sets=[valid_data],
num_boost_round=1000,
            callbacks=[lgb.early_stopping(stopping_rounds=50)],  # Remove verbose=False
            )
    # Evaluate
    y_pred          =           model.predict(X_test)
metrics.append({
        'mae': mean_absolute_error(y_test, y_pred),
        'mape': mean_absolute_percentage_error(y_test, y_pred)
    })
# Print average CV performance
avg_mae = np.mean([m['mae'] for m in metrics]) avg_mape =
np.mean([m['mape'] for m in metrics]) print(f"Cross-Validation
MAE: {avg_mae:.2f}") print(f"Cross-Validation MAPE:
```

```
{avg_mape:.2%}") print(f"Cross-Validation Accuracy:
{100*(1-avg_mape):.2f}%")

# Train final model on all data
final_model = lgb.train(params,
            lgb.Dataset(X,                              label=y),
num_boost_round=model.best_iteration)

# Generate future features last_idx =
student_data['time_idx'].max()
future_years = pd.date_range(
  start=student_data['year'].max() + pd.DateOffset(years=1),
periods=10,   freq='Y'
)

future_data = pd.DataFrame({
  'time_idx': range(last_idx + 1, last_idx + 11),
  'year_num': future_years.year,
  'lag_1': [student_data['students'].iloc[-1]] * 10,
  'rolling_mean_3': [student_data['students'].iloc[-3:].mean()] * 10
})

# Make predictions
forecast = final_model.predict(future_data)
```

```
# Create forecast DataFrame forecast_df
= pd.DataFrame({
    'year': future_years,
    'forecast': forecast,
    'lower_bound':    forecast    *    0.95,        #    5%    lower
'upper_bound': forecast * 1.05   # 5% upper
})

# Plot results plt.figure(figsize=(12, 6)) plt.plot(student_data['year'], y, 'bo-',
label='Actual') plt.plot(forecast_df['year'], forecast_df['forecast'], 'ro--',
label='Forecast') plt.fill_between(forecast_df['year'],
forecast_df['lower_bound'],                 forecast_df['upper_bound'],
color='red', alpha=0.1) plt.title('LightGBM Student Enrollment Forecast\nAccuracy:
{:.2f}%'.format(100*(1avg_mape)))
plt.xlabel('Year')
plt.ylabel('Students')
plt.legend() plt.grid(True,
alpha=0.3) plt.show()

# Display forecast with accuracy print("\n10-Year
Forecast with Confidence Bounds:")
print(forecast_df.round(1))
```

1. Combine historical data with forecast

historical_df = student_data[['year', 'students']].copy()

historical_df['type'] = 'Actual' # Mark as actual data

Prepare forecast data (with the same structure) forecast_export

= forecast_df[['year', 'forecast']].copy()

forecast_export = forecast_export.rename(columns={'forecast': 'students'}) forecast_export['type']

= 'Forecast' # Mark as forecast data

Combine into a single DataFrame

full_data = pd.concat([historical_df, forecast_export], axis=0)

2. Export for Power BI

full_data.to_csv('lightgbm_student_forecast.csv', index=False)

forecast_df.to_csv('lightgbm_forecast_details.csv', index=False)   # Optional: Export forecast bounds

References:

1. Chen, T., & Guestrin, C. (2016). *XG Boost: A Scalable Tree Boosting System*. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. https://doi.org/10.1145/2939672.2939785
2. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T. Y. (2017). *LightGBM: A Highly Efficient Gradient Boosting Decision Tree*. Advances in Neural Information Processing Systems (NIPS). https://proceedings.neurips.cc/paper/2017/file/6449f44a102fde848669bdd9eb6b76faPaper.pdf
3. Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: Principles and Practice* (2nd ed.). OTexts. https://otexts.com/fpp2/
4. Microsoft Corporation. (2024). *Power BI Documentation*. Retrieved from https://learn.microsoft.com/en-us/power-bi/
5. Box, G. E. P., Jenkins, G. M., & Reinsel, G. C. (2015). *Time Series Analysis: Forecasting and Control* (5th ed.). Wiley.
6. EDU Predict Tool Team 03. (2025). *Project 7 - EDU Predict Tool Bi-Weekly Report 06*. MSDS Capstone, University of New Haven.