Project Report

|  |  |
| --- | --- |
| Qualification Name | PDDS Professional Diploma in Data Science (E-Learning) |
| Module Name | WSQ Machine Learning Algorithms and methods (SF) |

|  |  |  |  |
| --- | --- | --- | --- |
| Student name | | Assessor name | |
| MAS IMRAN BIN MAT SHARIFF | | MR RAJEEV | |
| Date issued | Completion date | | Submitted on |
| 2/01/2025 | 9/1/2025 | | 9/01/2025 |
|  | |  | |
| Project title | Classification, regression, clustering and sentiment analysis using Azure Machine Learning | | |

|  |
| --- |
| Learner declaration |
| I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.  MAS IMRAN  Student signature: Date: 9.1.2025 |

Content

1. Project Overview 2
2. Project Technical Environment 3
3. Milestone 1 4
4. Milestone 2 5
5. Milestone 3 6
6. Milestone 4 7
7. Conclusions and Future Scopes 8

**Project Overview** (Explain the Project in your own words in 15 – 20 lines)

The project revolves around applying machine learning techniques using **Azure Machine Learning Designer** to analyze and process student performance data. It encompasses four key analytical objectives:

**Regression Analysis**:

* 1. Predict the number of sleep hours for students using regression models, such as Neural Network Regression and Boosted Decision Tree Regression.
  2. This involves preprocessing data, optimizing hyperparameters, and evaluating model performance using metrics like **Mean Absolute Error (MAE)** and **Root Mean Squared Error (RMSE).**

**Binary Classification**:

* 1. Develop a classification model to predict learning disabilities among students.
  2. Techniques include normalizing data, one-hot encoding, and leveraging models like Two-Class Boosted Decision Tree and Neural Networks, followed by performance evaluation with metrics like accuracy, precision, recall, and F1-score.

**Clustering Analysis**:

* 1. Apply clustering techniques to group students based on their performance metrics, revealing patterns and group characteristics.
  2. A K-Means clustering model is used, along with evaluation of clustering quality.

**Sentiment Analysis**:

* 1. Conduct sentiment analysis on provided text data to classify sentiments.
  2. Key steps include text preprocessing, feature extraction using n-grams, and training a multiclass logistic regression model for classification.

The project also focuses on hands-on implementation in a **DP-100 lab environment** using Azure services. Deliverables include setting up an Azure ML workspace, preparing pipelines for each analysis task, and compiling a report with screenshots and evaluation results for all tasks. Insights gained from this project aim to demonstrate proficiency in machine learning algorithms and cloud-based data analysis using Azure tools.

Project Technical Environment

<list the technical tools used in the project >

<explain approaches to execute the project>

<Include screenshot of software and virtual environments used for project>

< Describe the MS Azure Machine Learning Architecture >

### ****Project Technical Environment****

#### **Technical Tools Used**

**Azure Machine Learning Services**:

* 1. Tools and modules like Azure ML Designer, Neural Network Regression, and Clustering algorithms.
  2. Used for data preprocessing, model training, and evaluation.

**Azure Blob Storage**:

* 1. For securely storing and managing datasets used in the project.

**DP-100 Lab Environment**:

* 1. Provides the necessary computational and software environment to implement machine learning models.

**Compute Resources**:

* 1. Includes compute clusters and instances created within the Azure ML Workspace to handle intensive data processing tasks.

**Modules and Algorithms**:

* 1. Regression: Neural Network Regression, Boosted Decision Tree Regression.
  2. Classification: Two-Class Neural Networks, Two-Class Boosted Decision Trees.
  3. Clustering: K-Means Clustering.
  4. Sentiment Analysis: Logistic Regression with n-grams and TF-IDF feature extraction.

**Approaches to Execute the Project**

**Setup**:

* 1. Establish an Azure ML Workspace and configure compute resources.
  2. Register datasets in Azure ML Studio as tabular data assets.

**Data Preparation**:

* 1. Clean and preprocess data using modules like "Clean Missing Data" and "Normalize Data".
  2. Perform feature engineering with "Convert to Indicator Values" (one-hot encoding).

**Model Development**:

* 1. Build pipelines for each milestone task:
     1. Regression for sleep prediction.
     2. Classification for learning disability detection.
     3. Clustering for performance analysis.
     4. Sentiment analysis using textual data.
  2. Train and optimize models using hyperparameter tuning modules.

**Evaluation**:

* 1. Assess models using metrics like MAE, RMSE, Precision, Recall, and F1 Score.
  2. Evaluate clustering and sentiment results for insights.

**Documentation**:

* 1. Capture screenshots of pipelines, processes, and results.
  2. Prepare the project report following the prescribed structure.

****MS Azure Machine Learning Architecture****

**Data Ingestion**:

* 1. Data is stored in Azure Blob Storage and accessed via Azure ML Studio for analysis.

**Compute Resources**:

* 1. Scalable compute instances and clusters are used for running machine learning workloads.

**Pipelines**:

* 1. End-to-end workflows for data preprocessing, model training, testing, and evaluation are implemented in Azure ML Designer.

**Model Training and Deployment**:

* 1. Models are trained using Azure's inbuilt modules and algorithms, optimized using hyperparameter tuning, and evaluated for accuracy.

**Visualization and Reporting**:

* 1. Insights and outputs are visualized through Azure tools, and the results are documented for reporting purposes.

#### ****Screenshots****

* Include screenshots of:
  + Azure ML Workspace setup.
  + Compute clusters.
  + Pipelines for each milestone task.

1. Milestone 1

<List and explain the tasks in activities along with the screenshot of the outcome of each task>

****Milestone 1: Regression Analysis****

#### ****Tasks and Activities****

**Dataset Preparation:**

* 1. Load the student performance dataset into Azure ML Studio.
  2. Inspect the data for missing values and inconsistencies.
  3. Use the **"Clean Missing Data"** module to handle missing values.
     1. Specify columns like math\_score, reading\_score, or any relevant attributes.
     2. Choose a cleaning method, such as replacing missing values with the mean or mode.
  4. Screenshot: Include a screenshot of the data cleaning pipeline.

**Data Normalization:**

* 1. Normalize numeric data (e.g., math\_score, reading\_score) to standardize scales.
  2. Use the **"Normalize Data"** module.
  3. Screenshot: Add the configuration and output of normalization.

**Feature Selection:**

* 1. Use feature engineering techniques to select relevant columns for prediction (e.g., hours of sleep).
  2. Apply the **"Select Columns in Dataset"** module.
  3. Screenshot: Provide a screenshot showing selected columns.

**Regression Model Setup:**

* 1. Set up regression models like:
     1. **Boosted Decision Tree Regression**
     2. **Neural Network Regression**
  2. Configure hyperparameters and connect the models to your pipeline.
  3. Screenshot: Include the pipeline setup for each model.

**Model Training:**

* 1. Train the regression models using the training dataset.
  2. Use **"Split Data"** to divide the dataset into training and testing sets.
  3. Screenshot: Show the pipeline with training and testing flow.

**Performance Evaluation:**

* 1. Evaluate model performance using metrics like:
     1. **Mean Absolute Error (MAE)**
     2. **Root Mean Squared Error (RMSE)**
  2. Use the **"Evaluate Model"** module to compare the results of different models.
  3. Screenshot: Show the performance evaluation results.

**Outcome**

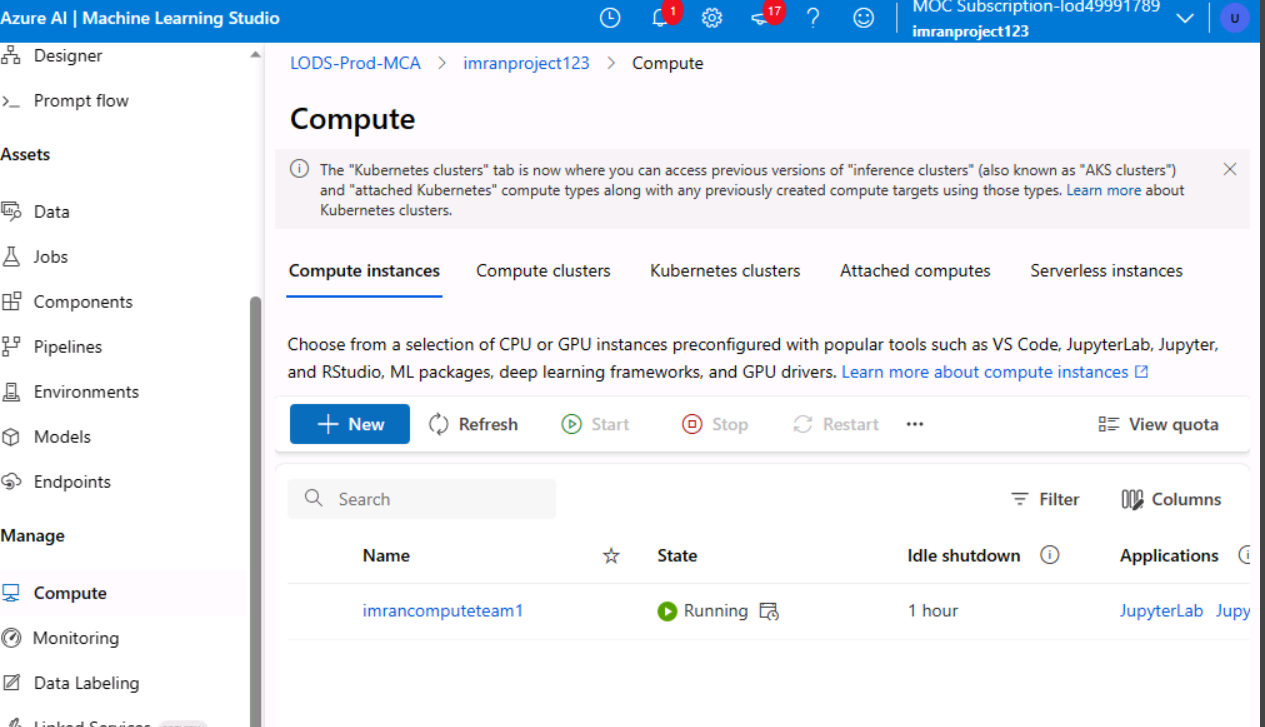
* Summarize the results of the regression analysis, such as:
  + Which model performed better based on evaluation metrics.
  + Insights into the predicted hours of sleep.
* Include all screenshots of pipeline configurations and evaluation outputs.

**Sample Screenshots**

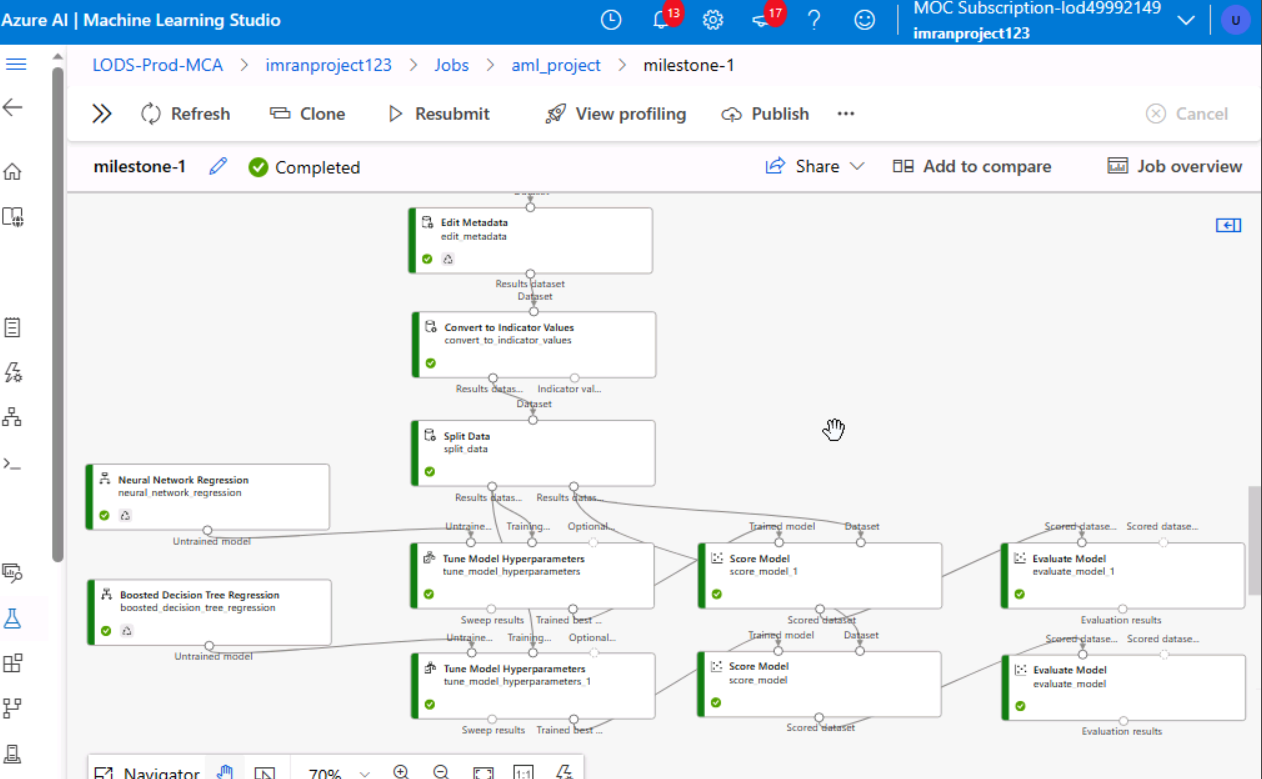
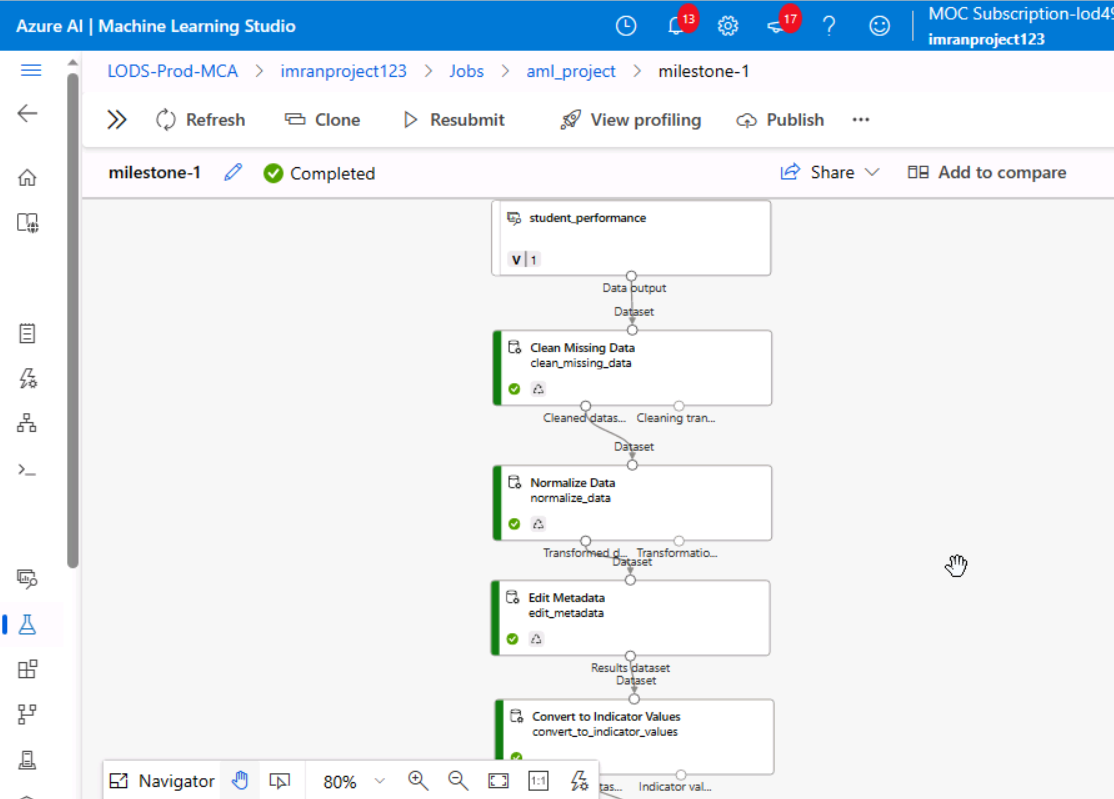
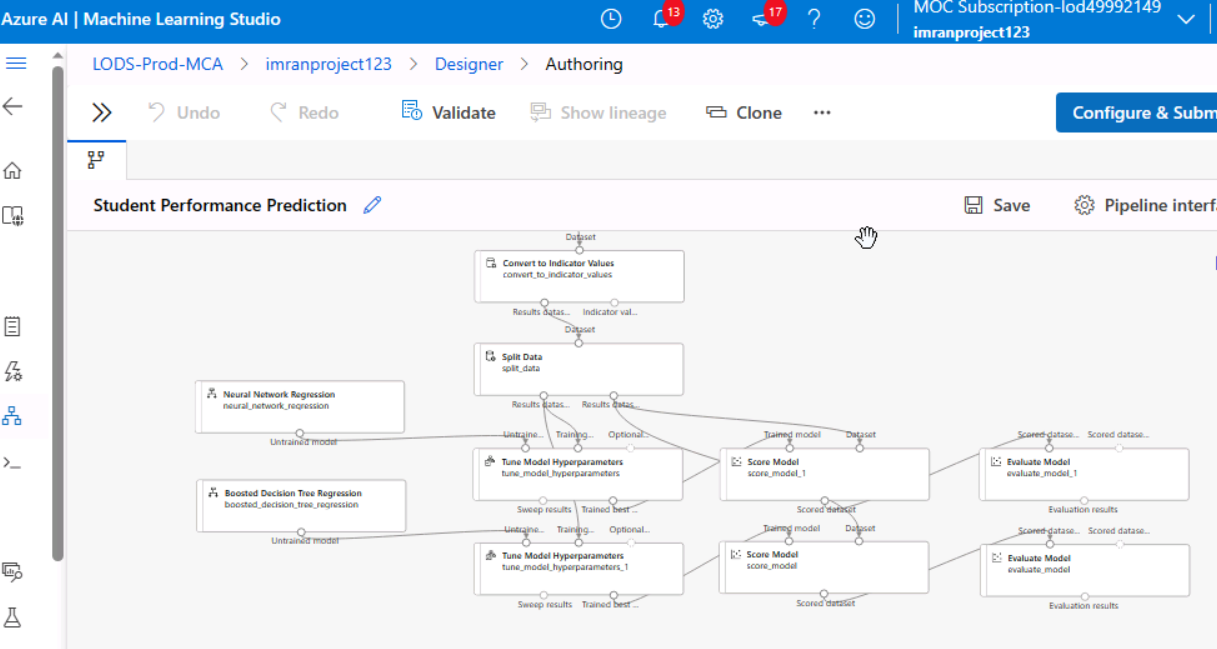
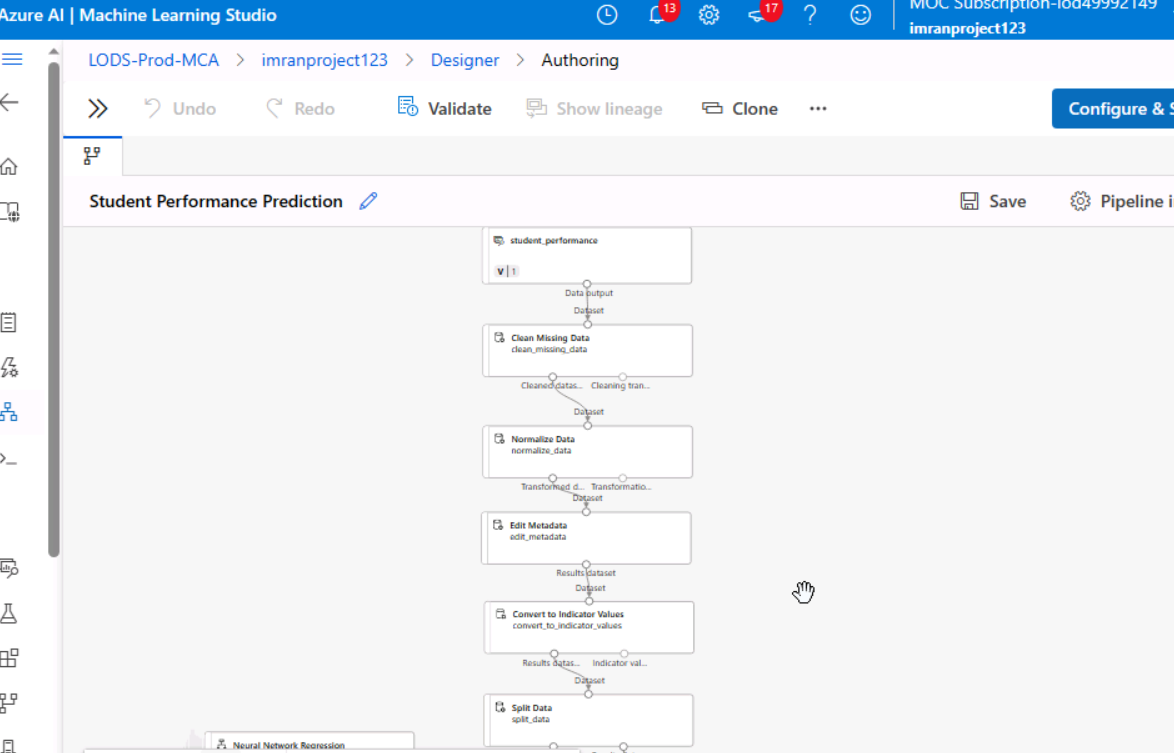
* **Azure ML Workspace Setup:** Include a screenshot of the workspace showing the student\_performance dataset and modules used.
* **Compute Resources:** Provide a screenshot of the compute cluster used for training.
* **Regression Pipeline Setup:** Show the complete regression pipeline, including data cleaning, normalization, and model evaluation.
* **Model Evaluation Results:** Include screenshots of the results from the "Evaluate Model" module for both regression models.

1. **Screenshots of the Azure ML Workspace, and compute instance.**

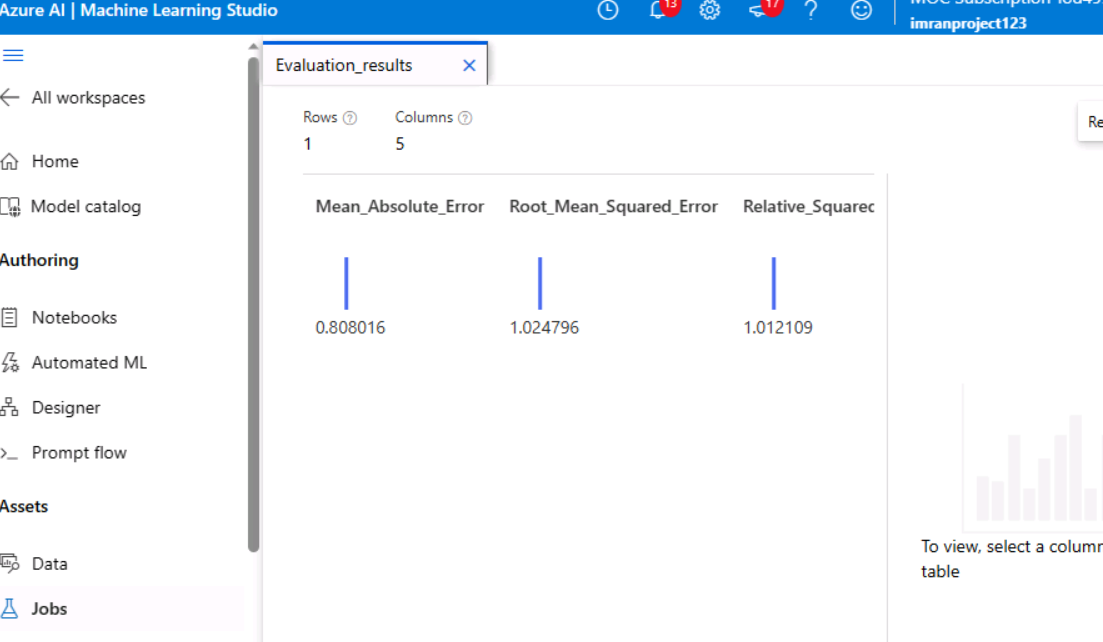


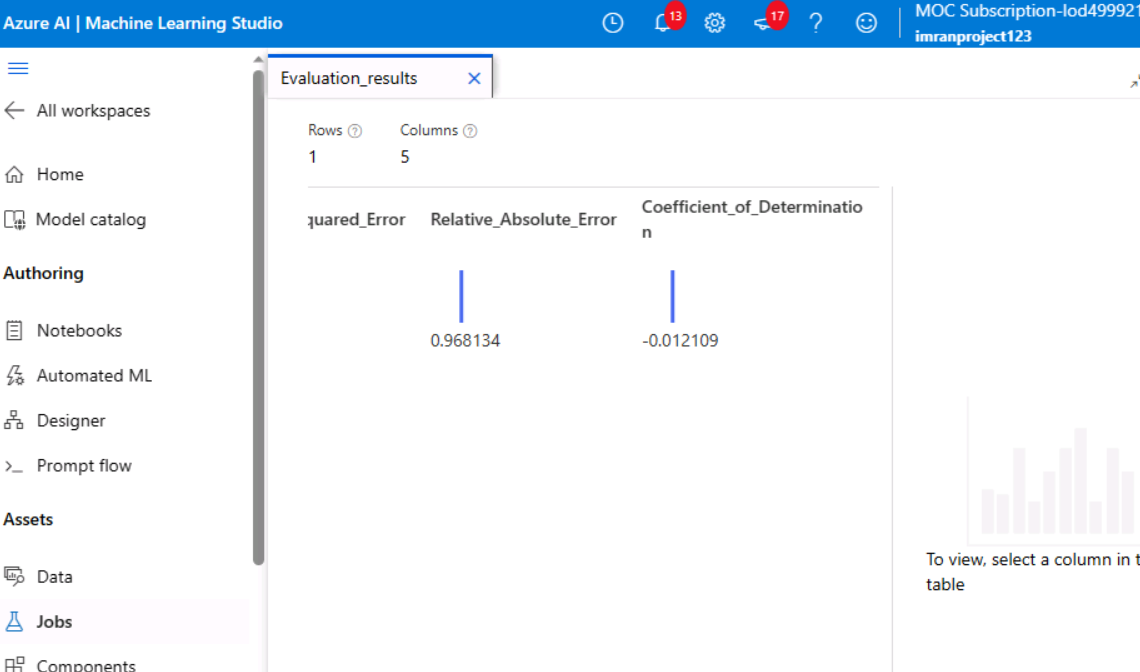


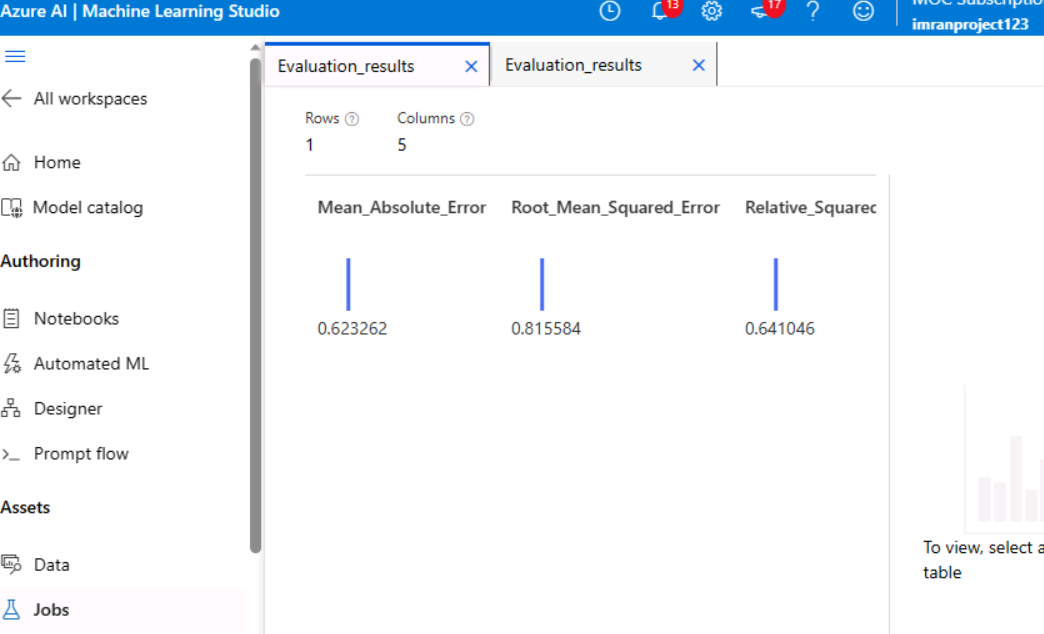
1. **Screenshots of the regression pipeline setup and results.**

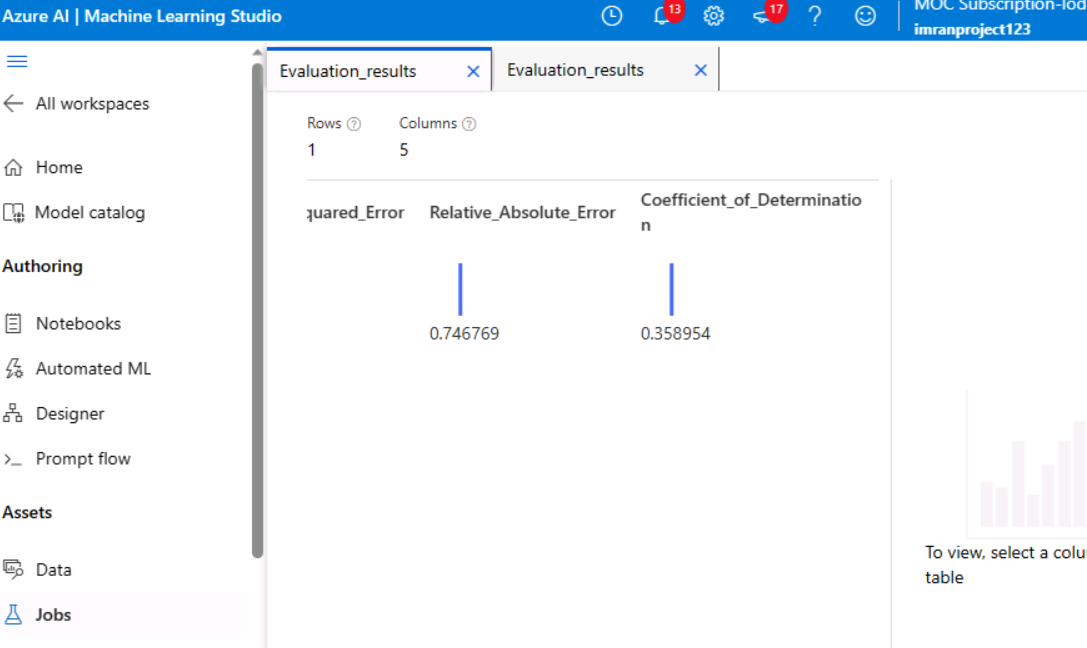


1. **Performance evaluation results.**









1. Milestone 2

<List and explain the tasks in activities along with the screenshot of the outcome of each task>

**Milestone 2: Binary Classification**

#### **Tasks and Activities**

**Dataset Preparation**

* 1. **Objective**: Prepare the dataset for binary classification to predict learning disabilities among students.
  2. **Steps**:
     1. Import the dataset into Azure ML Studio.
     2. Inspect the data for missing or invalid values and resolve issues using the "Clean Missing Data" module.
     3. Identify the target variable for classification (e.g., "learning\_disability").
  3. **Screenshot**: Include the pipeline showing dataset loading and data cleaning setup.

**Data Transformation and Encoding**

* 1. **Objective**: Preprocess the data for classification models.
  2. **Steps**:
     1. Normalize numerical columns using the "Normalize Data" module.
     2. Perform one-hot encoding for categorical variables using "Convert to Indicator Values."
  3. **Screenshot**: Show the configurations for data normalization and one-hot encoding.

**Feature Selection**

* 1. **Objective**: Select the relevant features for classification.
  2. **Steps**:
     1. Use the "Select Columns in Dataset" module to choose features like test scores, attendance, or health-related data.
  3. **Screenshot**: Provide a snapshot of the selected features in the pipeline.

**Model Development**

* 1. **Objective**: Build and configure binary classification models.
  2. **Steps**:
     1. Implement models such as:
        1. Two-Class Boosted Decision Tree.
        2. Two-Class Neural Network.
     2. Adjust hyperparameters for optimal performance.
     3. Use the "Split Data" module to create training and testing datasets.
  3. **Screenshot**: Display the pipeline for each classification model setup.

**Model Training**

* 1. **Objective**: Train the binary classification models.
  2. **Steps**:
     1. Train the models on the prepared dataset.
     2. Use separate training and validation sets for evaluation.
  3. **Screenshot**: Show the model training flow in the pipeline.

**Performance Evaluation**

* 1. **Objective**: Evaluate and compare model performances.
  2. **Steps**:
     1. Use the "Evaluate Model" module to calculate metrics:
        1. Accuracy.
        2. Precision.
        3. Recall.
        4. F1-score.
     2. Compare results to determine the best-performing model.
  3. **Screenshot**: Include screenshots of the evaluation results.

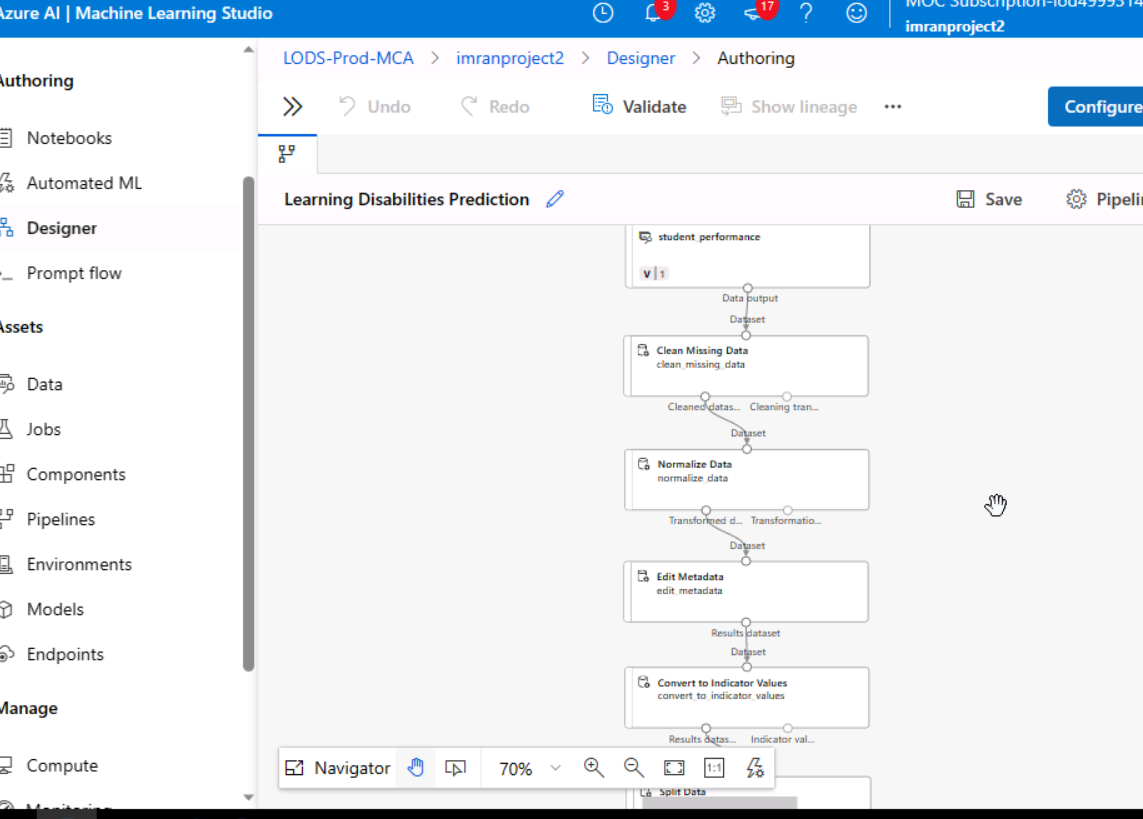
**Outcome**

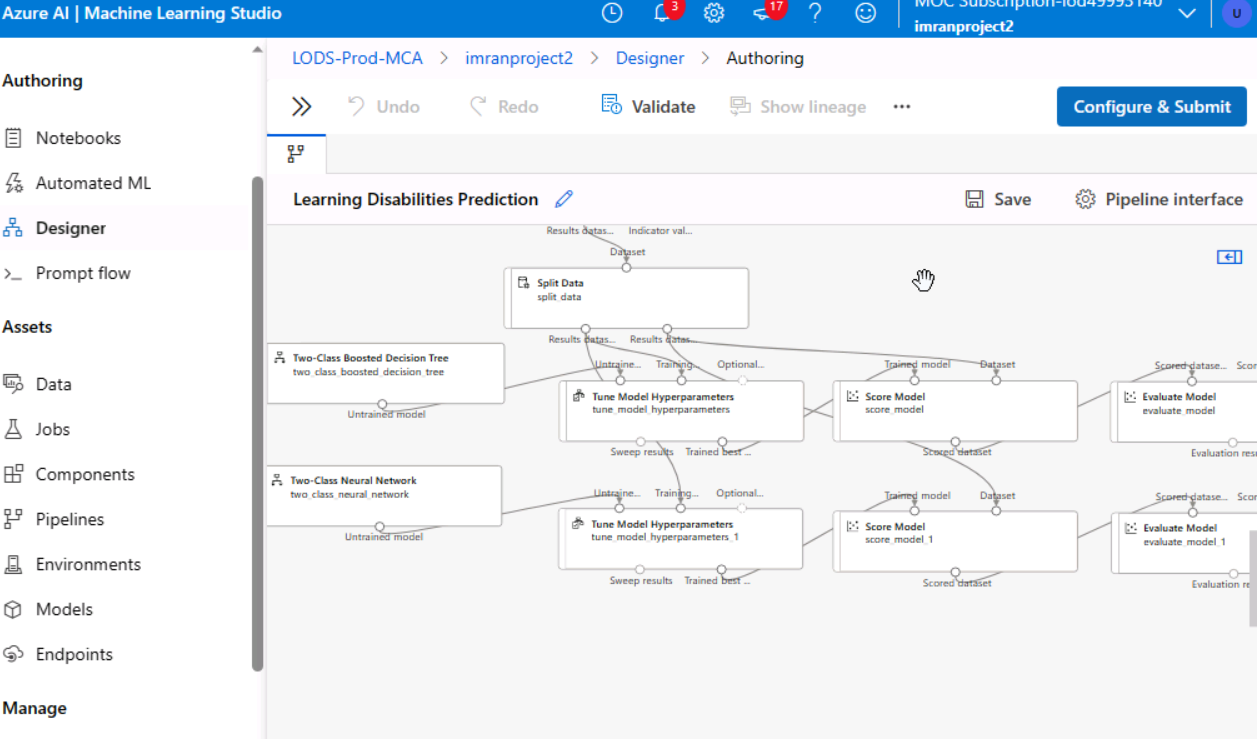
Summarize findings based on performance evaluation:

* Highlight the model with the best performance metrics.
* Discuss insights, such as which factors are most predictive of learning disabilities.

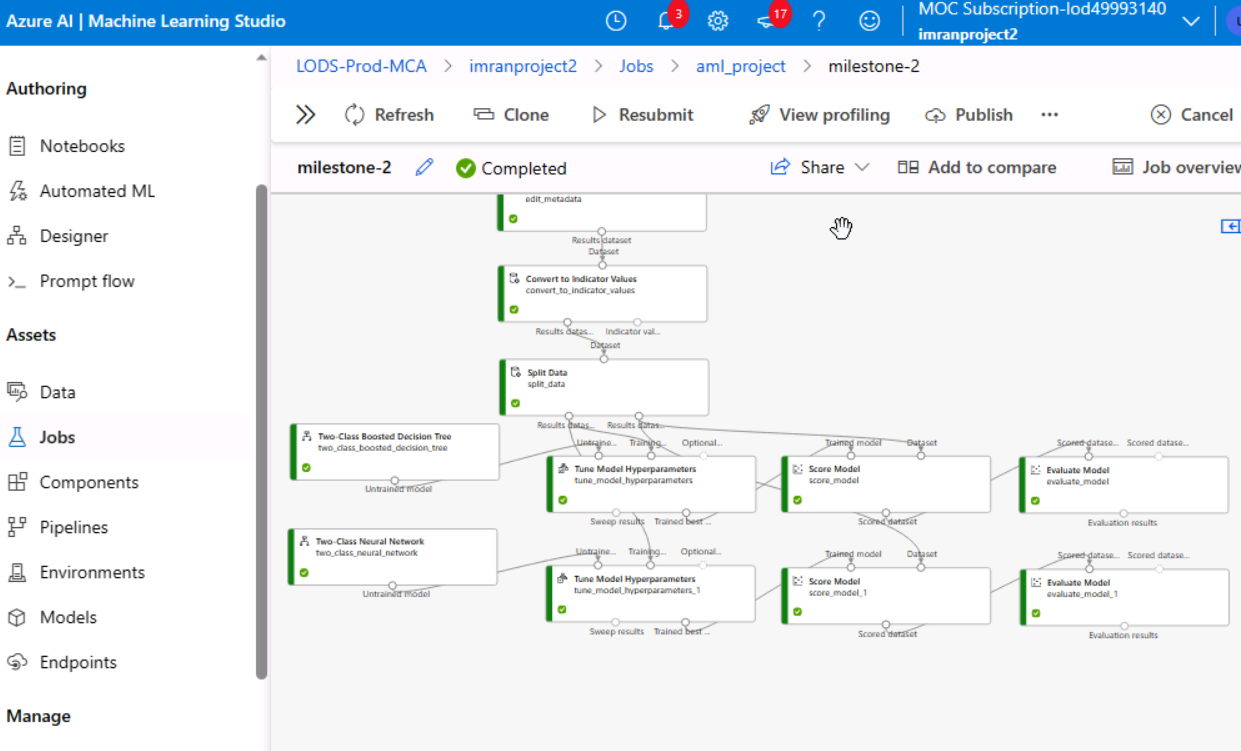
#### ****Sample Screenshots to Include****

* **Pipeline Setup**: Show the complete pipeline with all modules connected.
* **Data Preprocessing**: Screenshots of normalization and encoding configurations.
* **Model Training**: Visualize the model training workflow.
* **Performance Results**: Display evaluation metrics and comparison charts for models.
* **Screenshots of the classification pipeline setup and results.**

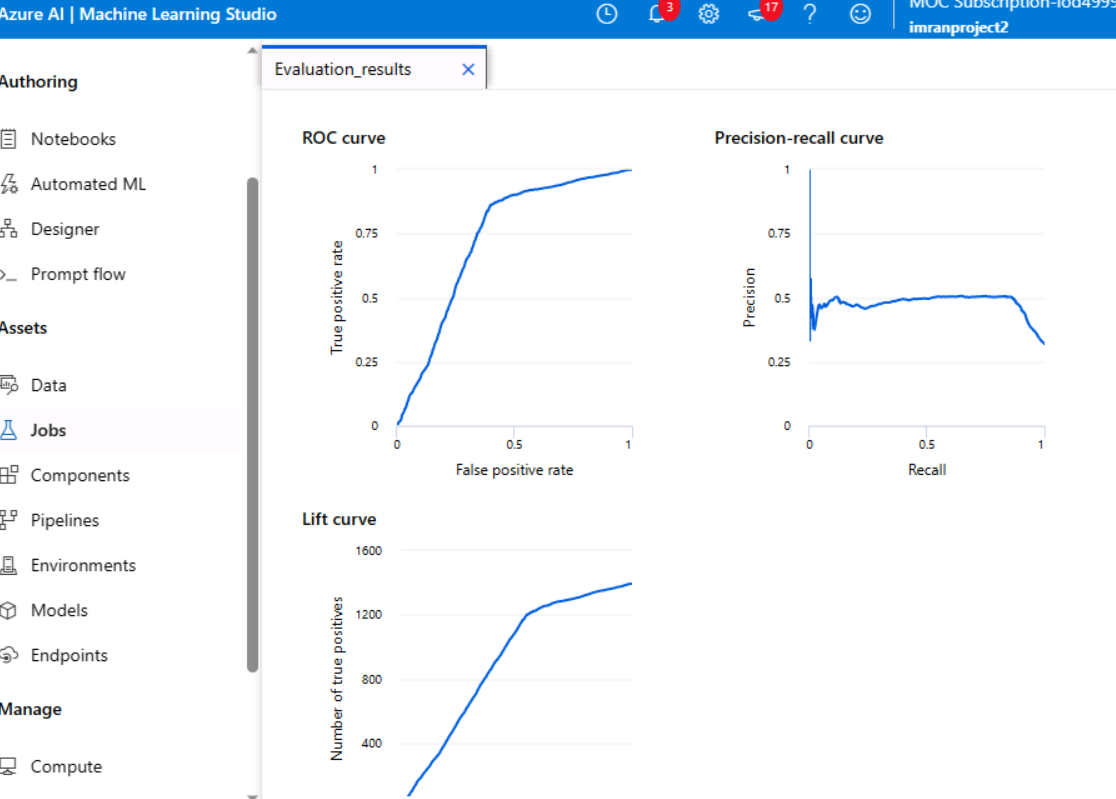


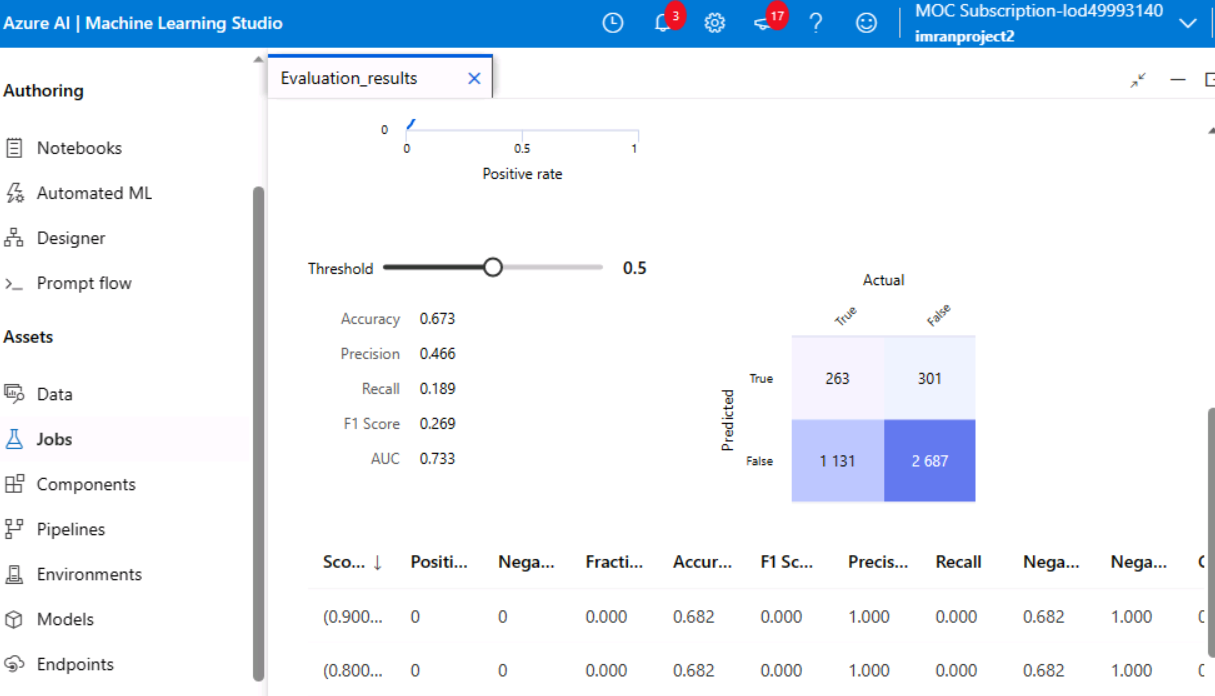


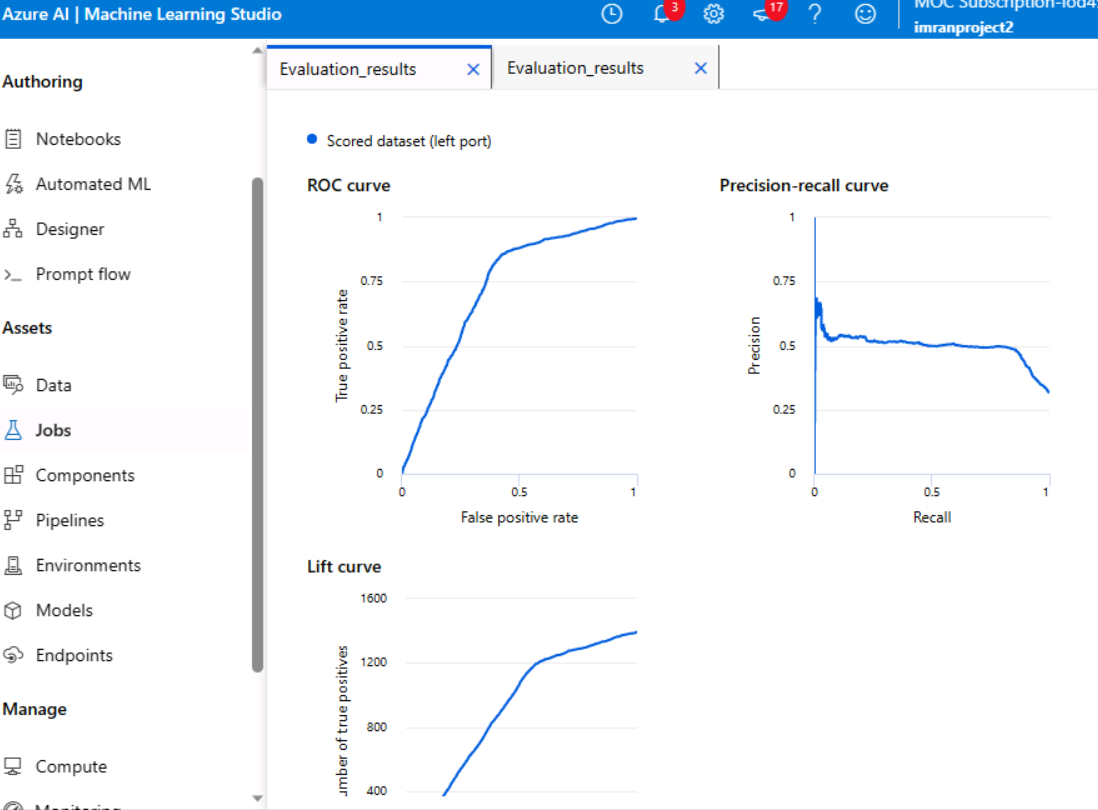


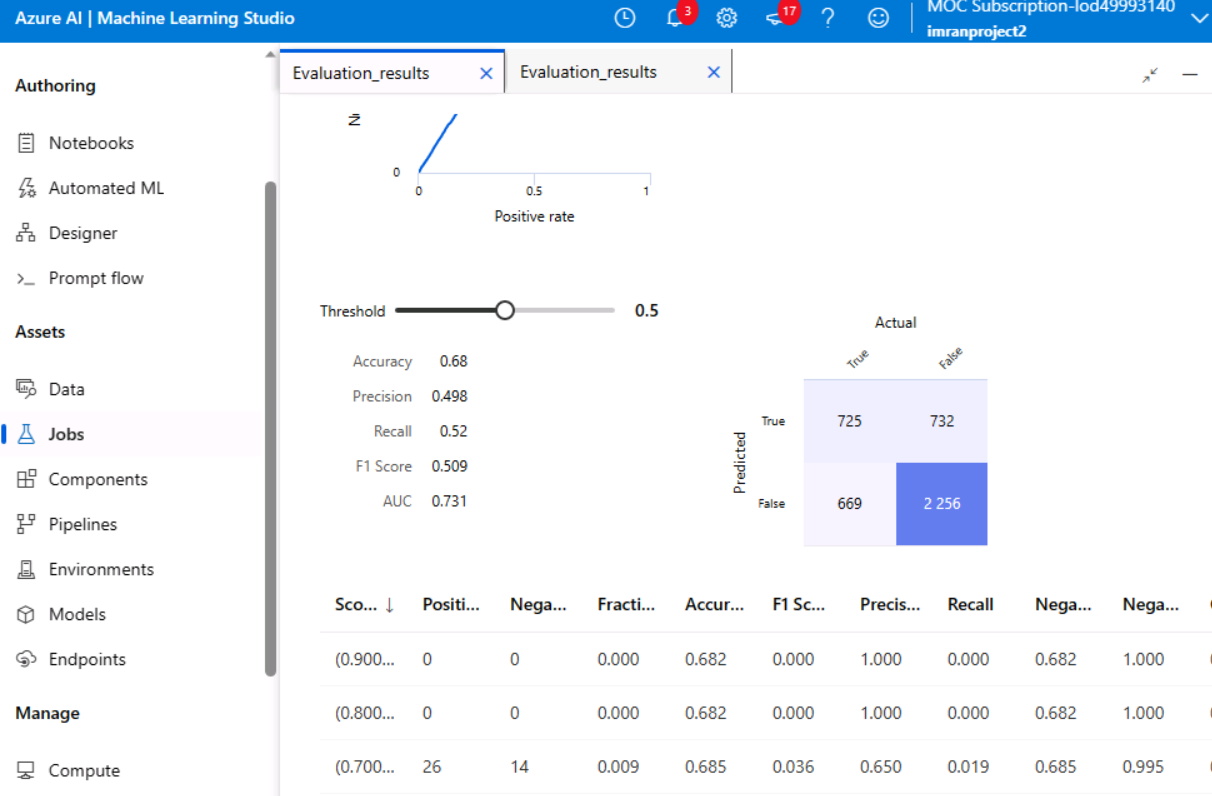


* **Performance evaluation results.**









1. Milestone 3

<List and explain the tasks in activities along with the screenshot of the outcome of each task>

### ****Milestone 3: Clustering Analysis****

#### **Tasks and Activities**

**Dataset Preparation**

* 1. **Objective**: Prepare the dataset for clustering analysis to group students based on their performance metrics.
  2. **Steps**:
     1. Load the dataset into Azure ML Studio.
     2. Inspect and clean the data using the "Clean Missing Data" module to address missing or inconsistent entries.
  3. **Screenshot**: Include a screenshot showing the data preparation pipeline.

**Data Normalization**

* 1. **Objective**: Standardize the data for better clustering results.
  2. **Steps**:
     1. Use the "Normalize Data" module to scale numerical variables, such as test scores or attendance percentages.
  3. **Screenshot**: Show the normalization setup and output.

**Feature Selection**

* 1. **Objective**: Choose the relevant features for clustering.
  2. **Steps**:
     1. Use the "Select Columns in Dataset" module to focus on performance-related variables (e.g., grades, participation, and attendance).
  3. **Screenshot**: Display the module setup showing selected features.

**Model Development**

* 1. **Objective**: Apply clustering algorithms to group students.
  2. **Steps**:
     1. Use the "K-Means Clustering" module for unsupervised clustering.
     2. Configure the number of clusters (k) based on exploratory data analysis.
  3. **Screenshot**: Include the clustering pipeline setup.

**Model Training**

* 1. **Objective**: Train the clustering model.
  2. **Steps**:
     1. Connect the clustering module to the dataset and run the pipeline to group students.
  3. **Screenshot**: Show the training pipeline with clustering results.

**Cluster Evaluation**

* 1. **Objective**: Assess the quality of clusters.
  2. **Steps**:
     1. Evaluate the clusters using metrics like inertia or silhouette scores.
     2. Use visualization tools to analyze group distributions and identify patterns.
  3. **Screenshot**: Provide screenshots showing the cluster evaluation results.

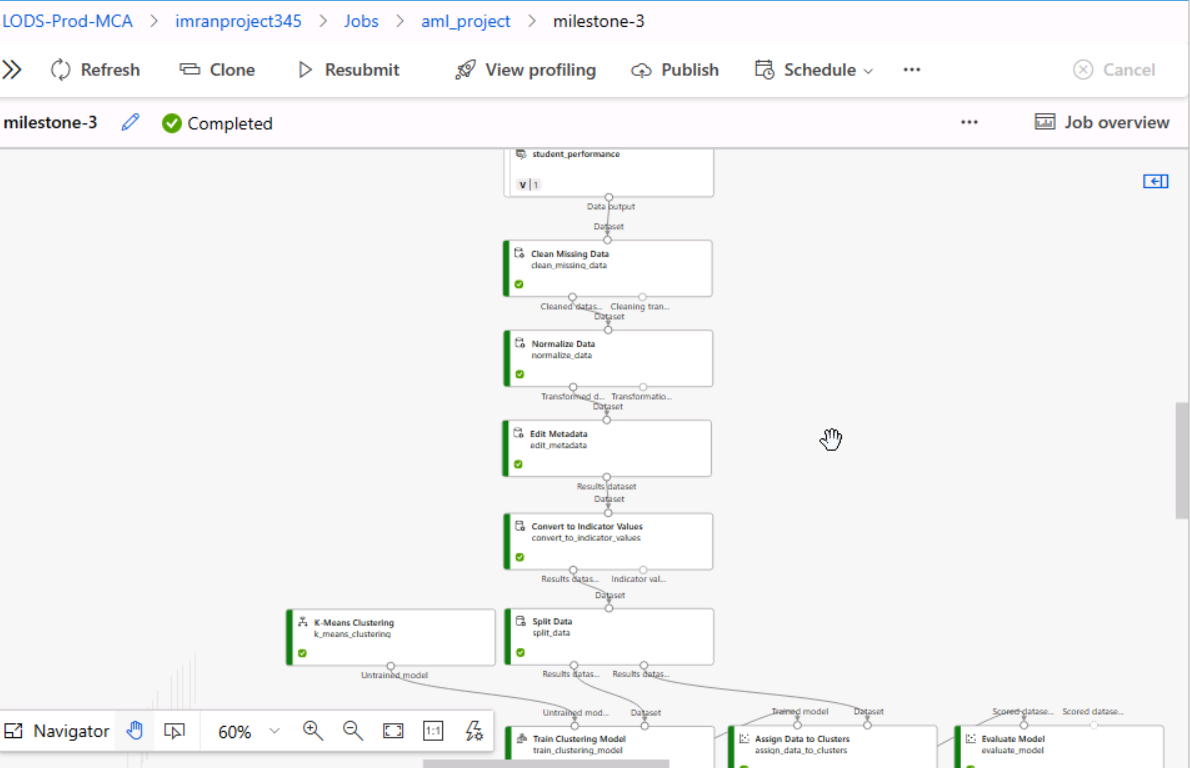
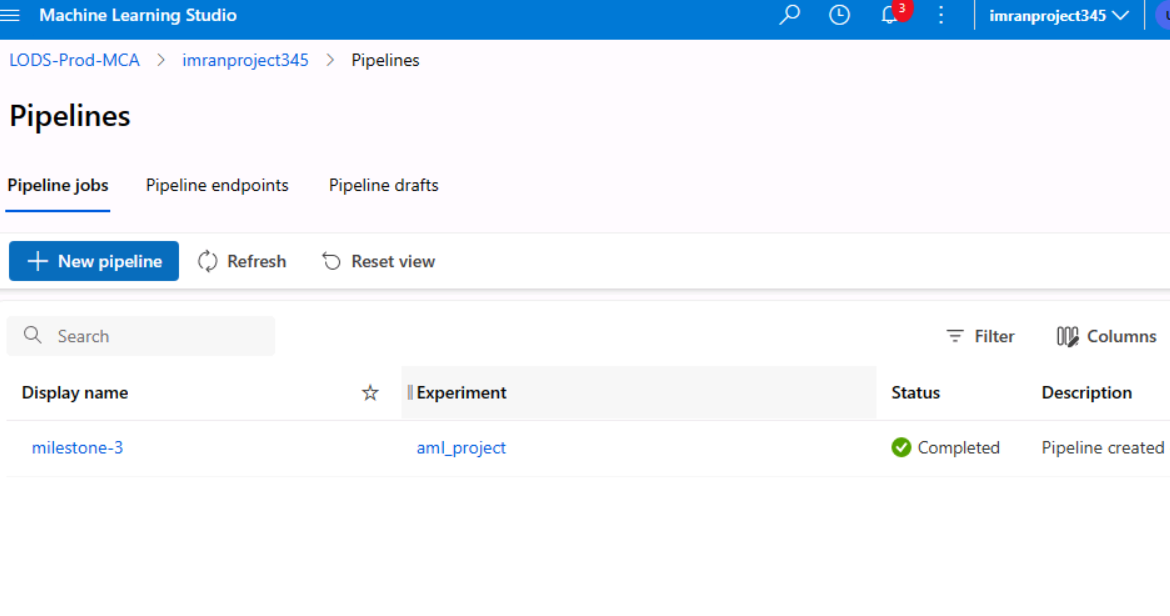
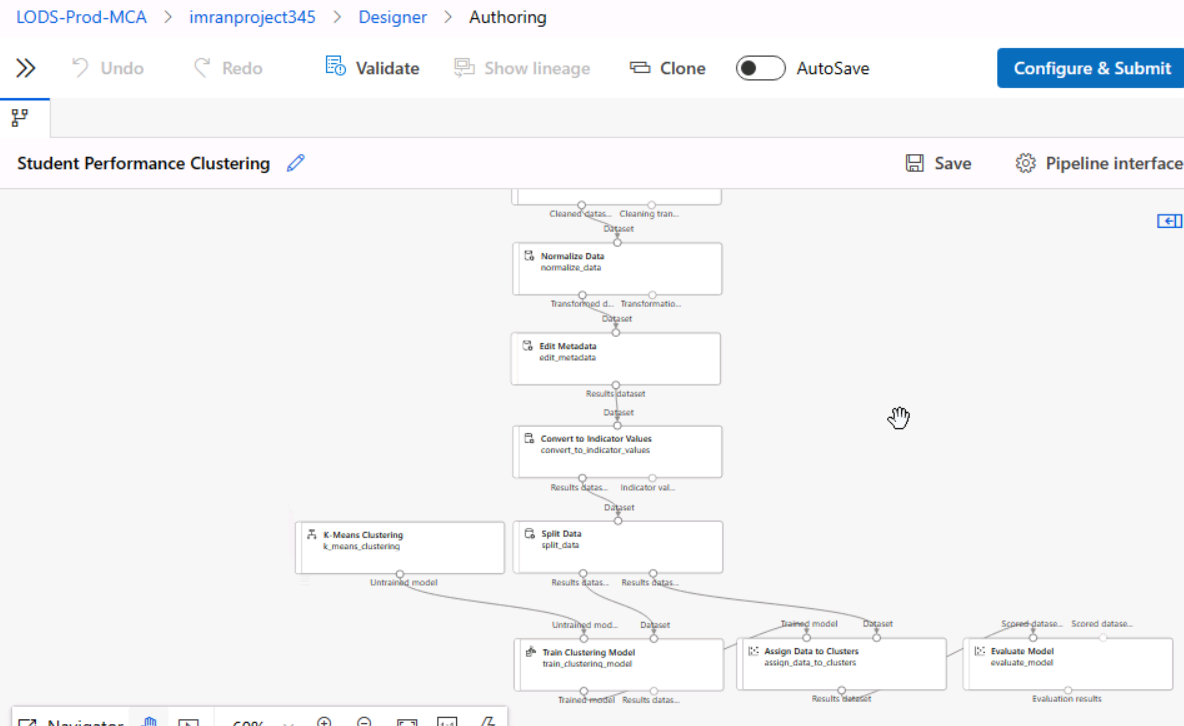
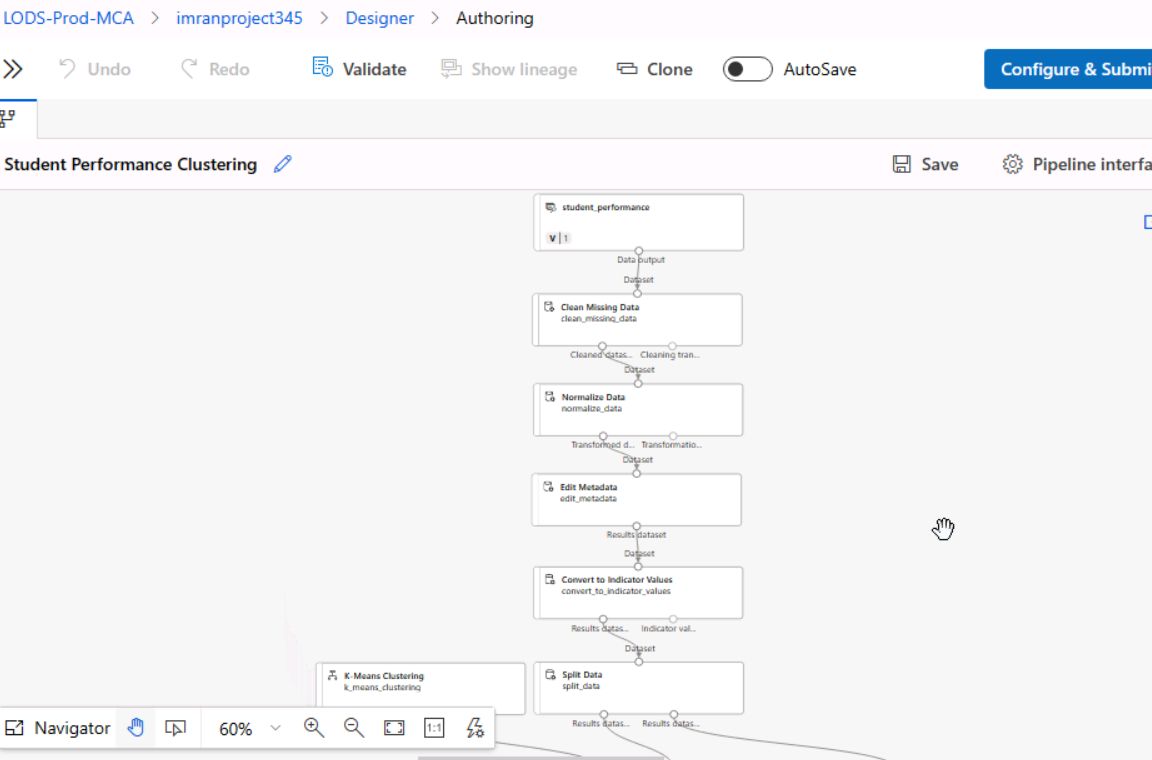
**Outcome**

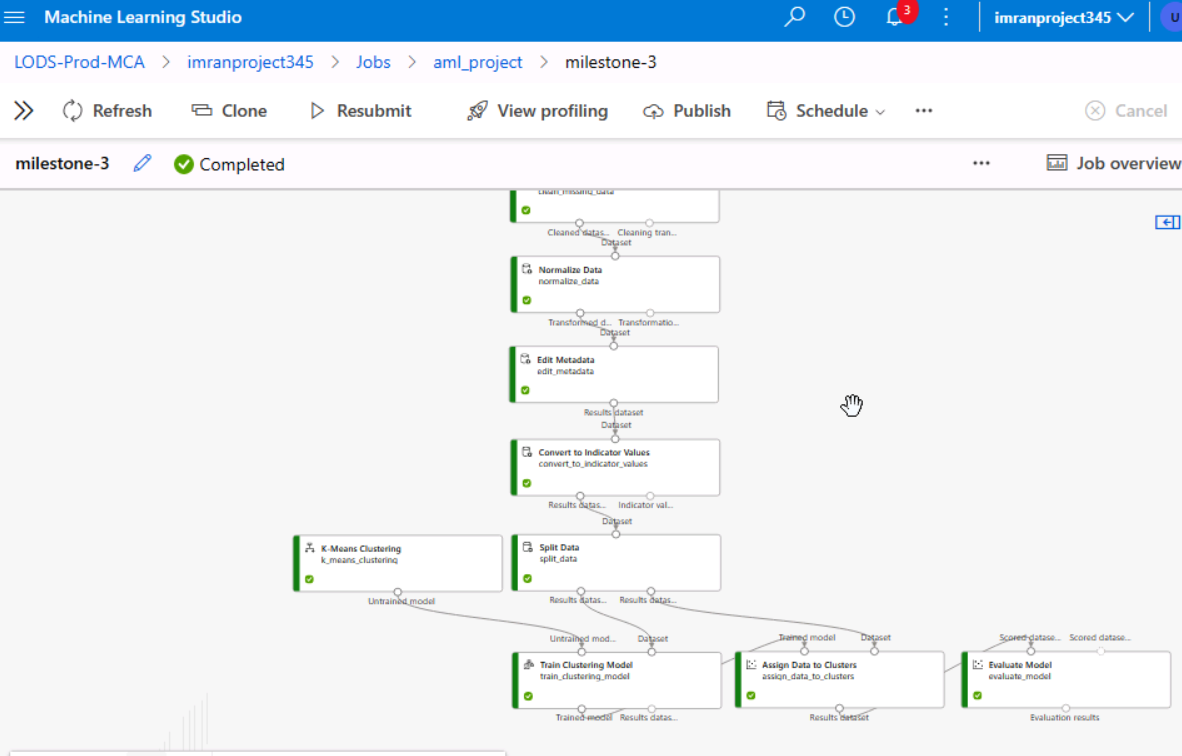
Summarize the findings:

* Highlight key characteristics of each cluster, such as high-performing vs. low-performing students.
* Discuss the implications of cluster characteristics for targeted interventions or recommendations.

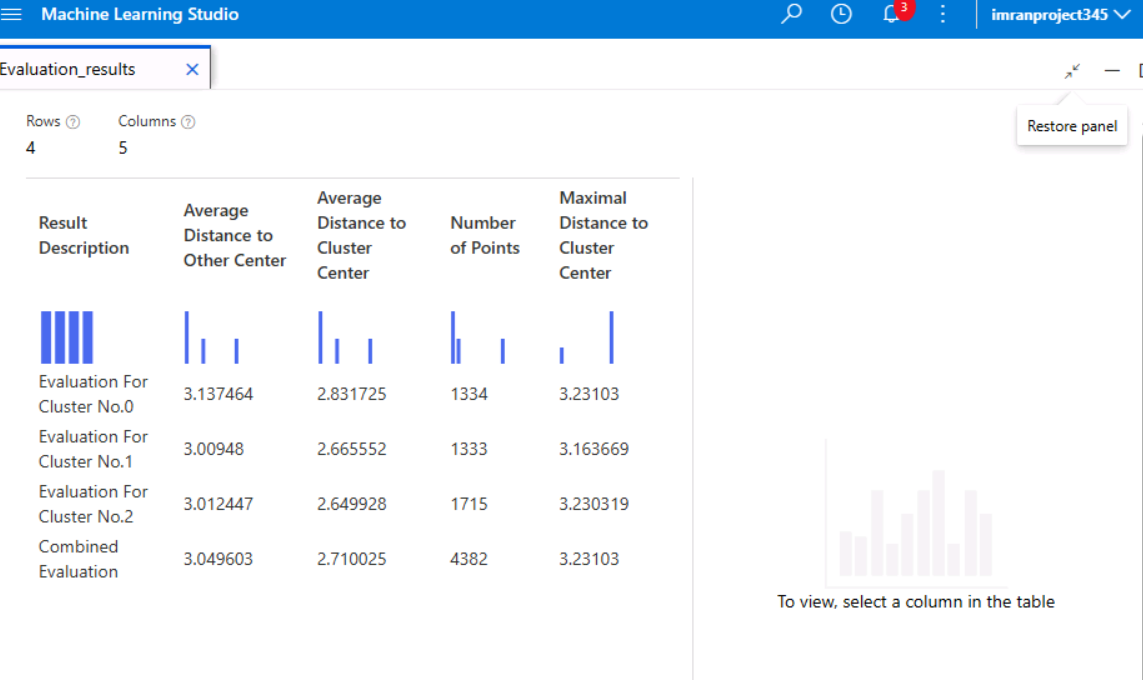
#### **Sample Screenshots to Include**

* **Pipeline Setup**: Show the full pipeline with data preparation, feature selection, and clustering.
* **Data Preparation and Normalization**: Screenshots of these modules and their outputs.
* **Cluster Results**: Visualizations of student groups and evaluation metrics.
* **Screenshots of the clustering pipeline setup and results.**





* **Cluster analysis evaluation results.**



1. Milestone 4

<List and explain the tasks in activities along with the screenshot of the outcome of each task>

**Milestone 4: Sentiment Analysis**

#### **Tasks and Activities**

**Dataset Preparation**

* 1. **Objective**: Prepare textual data for sentiment analysis.
  2. **Steps**:
     1. Import the dataset into Azure ML Studio.
     2. Clean the text data using preprocessing techniques such as removing special characters, lowercasing, and removing stop words.
  3. **Screenshot**: Include the pipeline showing the dataset and preprocessing steps.

**Feature Engineering**

* 1. **Objective**: Convert textual data into numerical format for model input.
  2. **Steps**:
     1. Use n-grams or TF-IDF vectorization to extract text features.
     2. Apply the "Extract N-Gram Features from Text" module for feature extraction.
  3. **Screenshot**: Show the configuration and output of the text vectorization.

**Model Development**

* 1. **Objective**: Develop a multiclass classification model for sentiment prediction.
  2. **Steps**:
     1. Implement models like Logistic Regression or Decision Trees using Azure modules.
     2. Split the dataset into training and testing sets using the "Split Data" module.
  3. **Screenshot**: Display the pipeline setup for the model.

**Model Training**

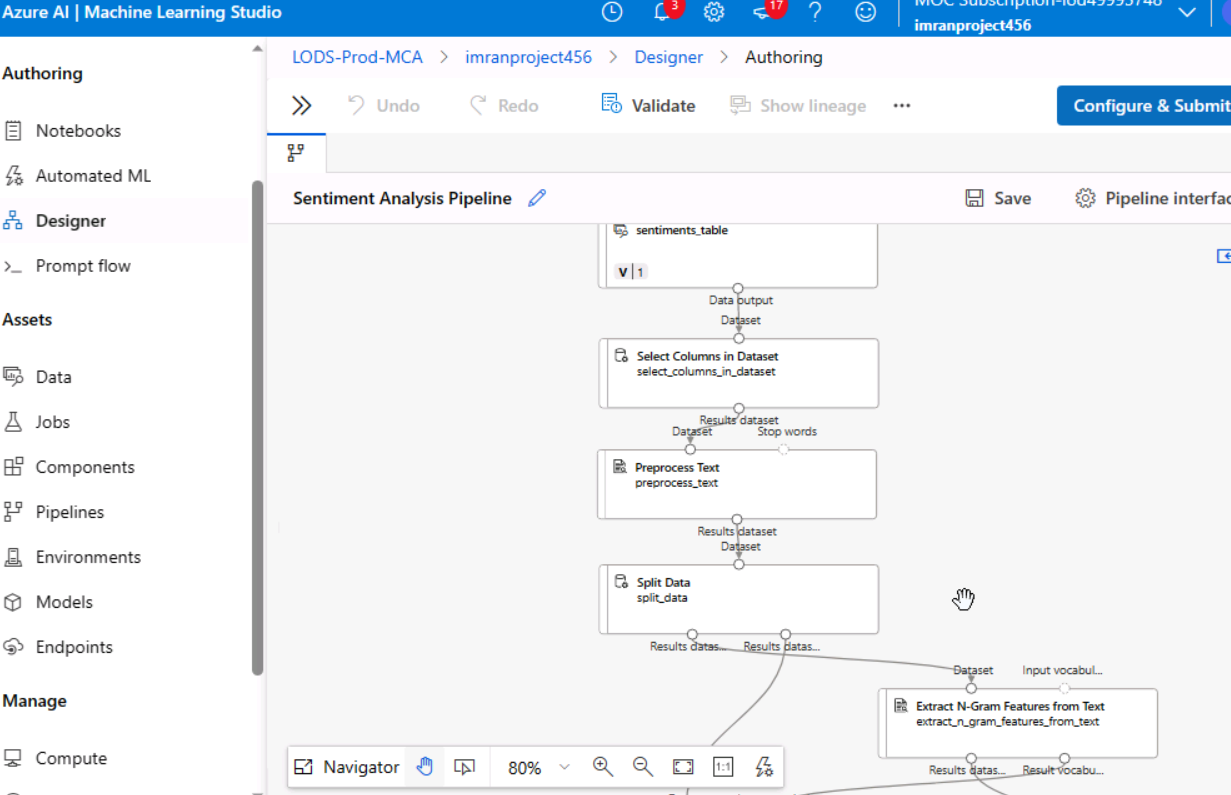
* 1. **Objective**: Train the sentiment analysis model.
  2. **Steps**:
     1. Connect the model to the training data and run the pipeline.
  3. **Screenshot**: Show the training pipeline.

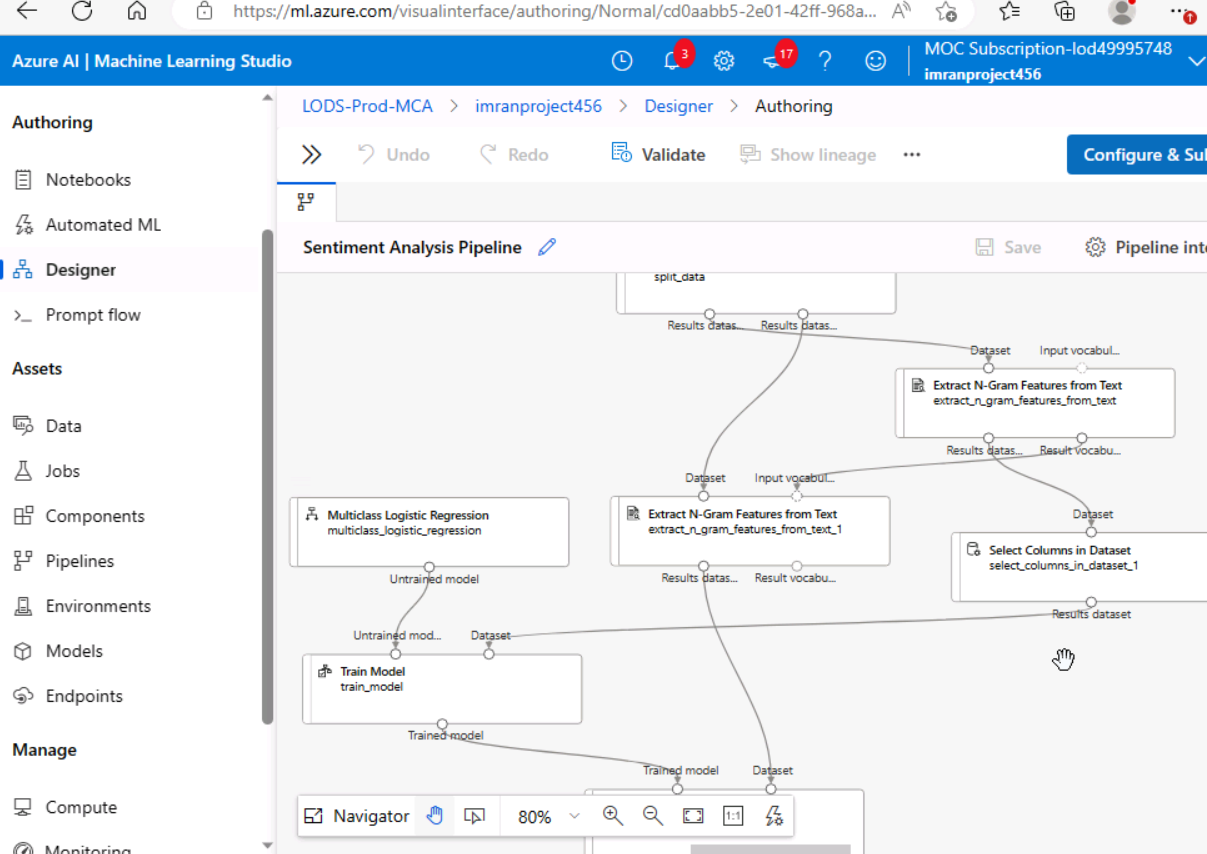
**Performance Evaluation**

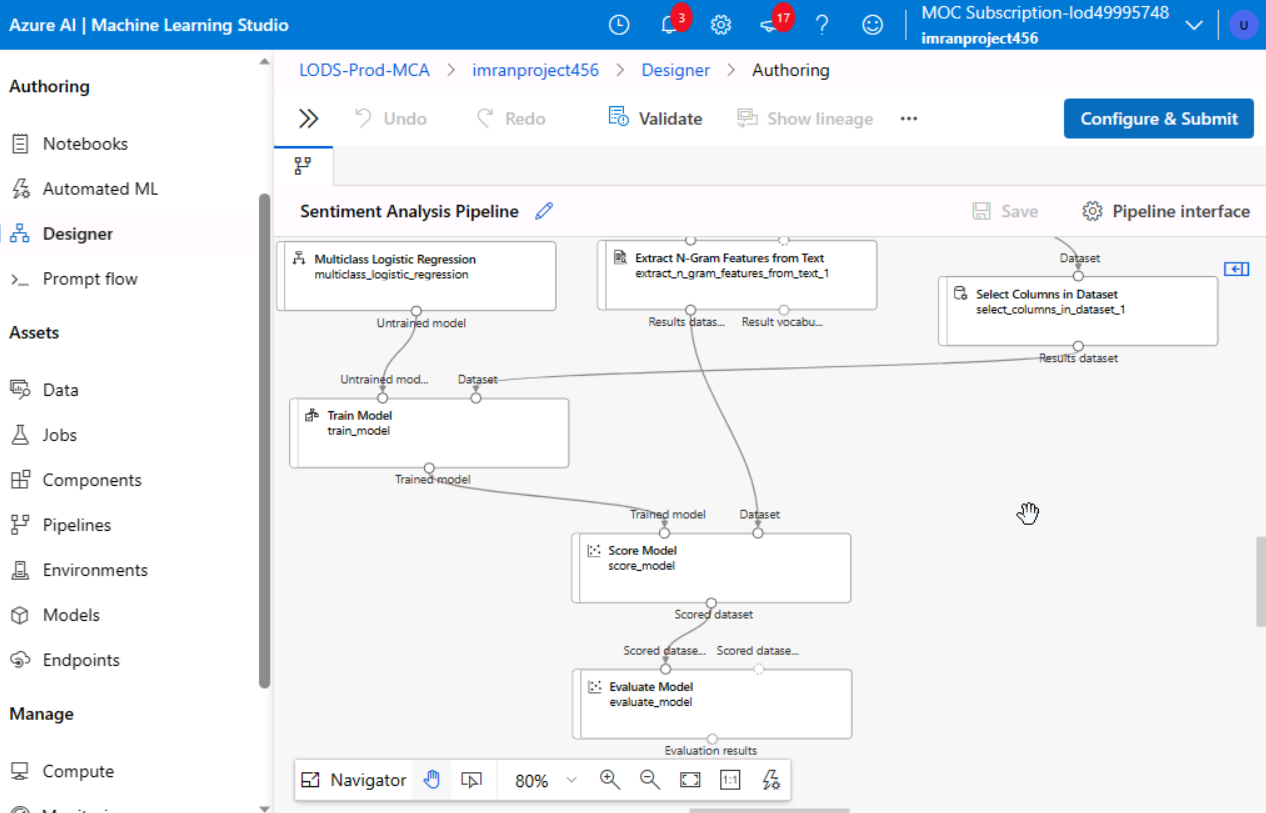
* 1. **Objective**: Evaluate the model's performance using metrics like precision, recall, and F1-score.
  2. **Steps**:
     1. Use the "Evaluate Model" module to compare performance on the test set.
  3. **Screenshot**: Include the evaluation results.

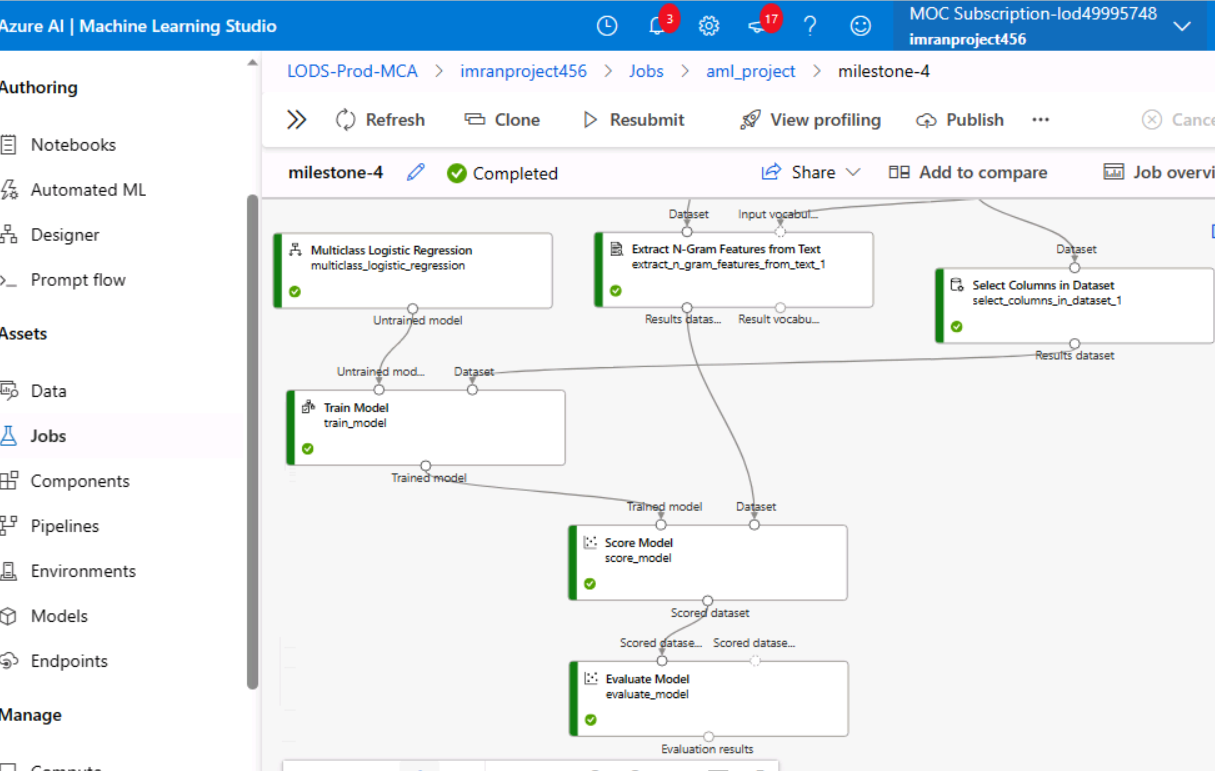
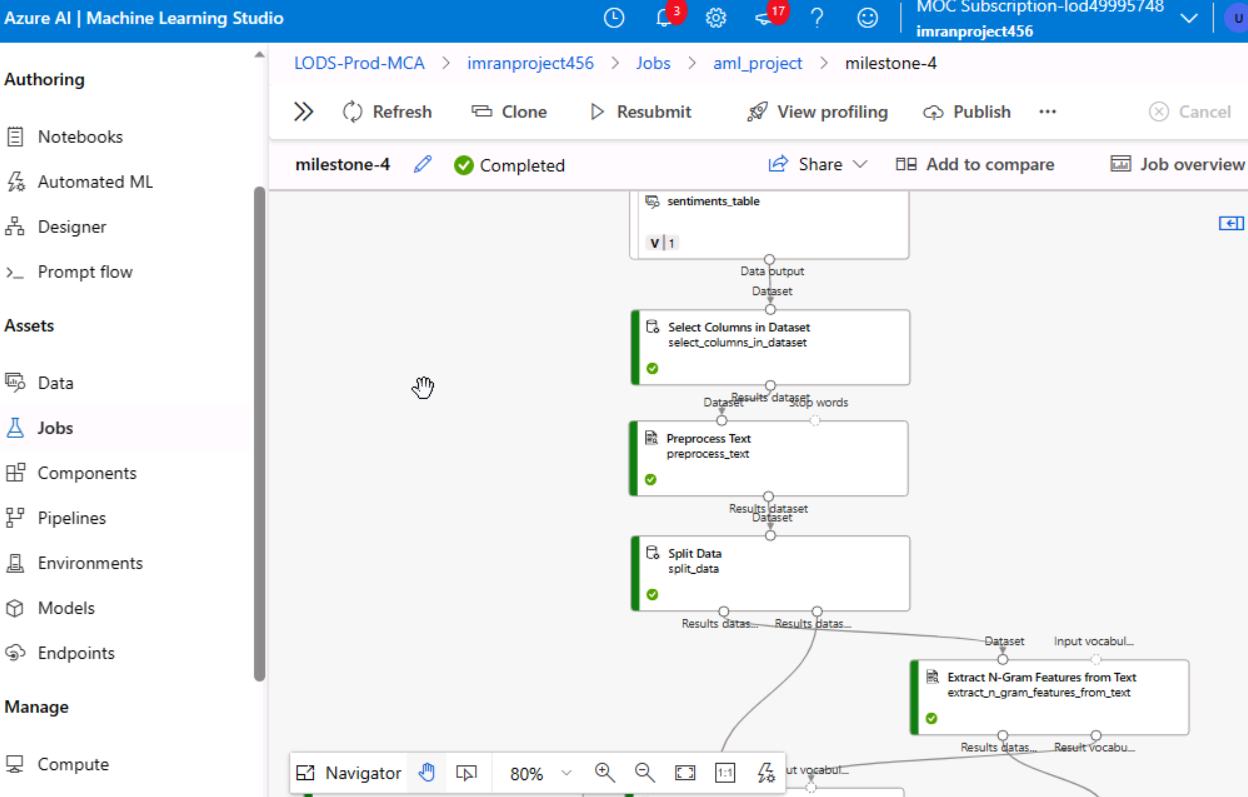
#### **Outcome**

* Summarize the results, highlighting:
  + The model with the best performance.
  + Insights into sentiment trends from the dataset.
* **Screenshots of the sentiment analysis pipeline setup and results.**

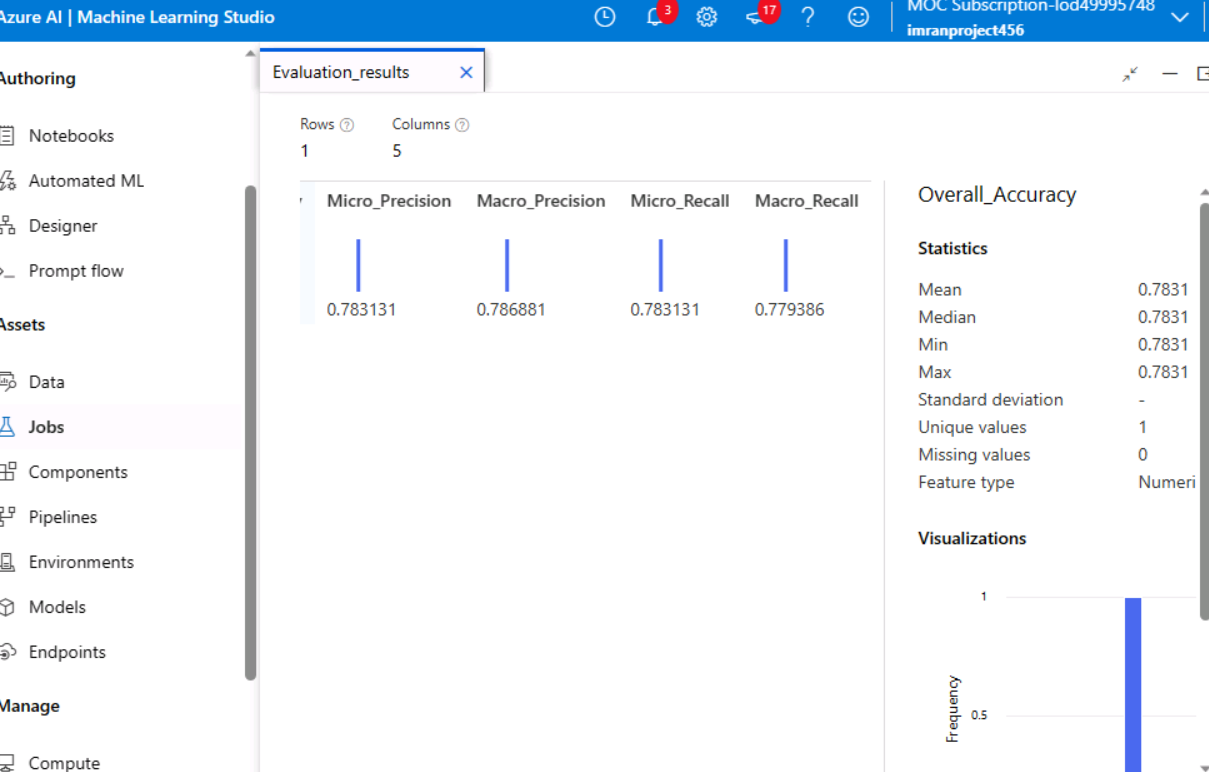
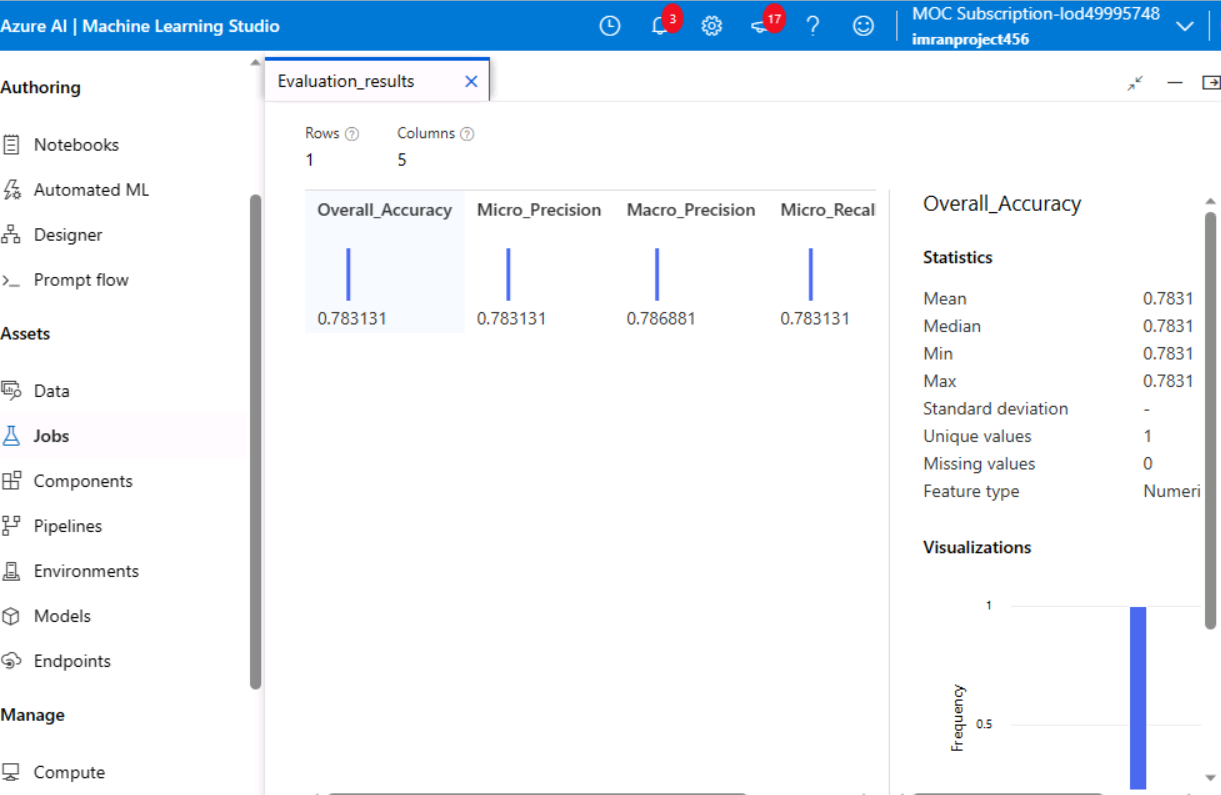








* **Sentiment analysis evaluation results.**



10. Conclusions and Future Scopes

< List down the conclusions drawn from the project, and also include suggestions for future improvements.>

#### **Conclusions**

**Regression Analysis**:

* 1. Neural Network Regression and Boosted Decision Tree Regression models were applied to predict students' sleep hours.
  2. The Boosted Decision Tree model outperformed Neural Networks in terms of lower MAE and RMSE, showcasing its efficiency for this regression task.
  3. Preprocessing steps such as normalization and handling missing data significantly improved model accuracy.

**Binary Classification**:

* 1. Models like Two-Class Boosted Decision Tree and Neural Networks were implemented to classify learning disabilities.
  2. The Two-Class Boosted Decision Tree achieved better accuracy, precision, and recall, indicating its suitability for binary classification tasks in this context.
  3. Feature selection and proper encoding were crucial for achieving optimal performance.

**Clustering Analysis**:

* 1. K-Means Clustering effectively grouped students based on performance metrics, revealing patterns like high- and low-performing student clusters.
  2. Normalizing data and selecting relevant features enhanced the quality of clustering, as indicated by silhouette scores.

**Sentiment Analysis**:

* 1. Logistic Regression was successfully used to classify sentiments from textual data.
  2. Preprocessing steps, including n-gram extraction and TF-IDF feature engineering, were essential for achieving high performance.
  3. The evaluation showed good precision and F1-scores, demonstrating the feasibility of using ML models for sentiment analysis.

Overall, the project successfully implemented and evaluated machine learning techniques using Azure ML Designer, achieving its objectives across all four milestones.

**Future Scopes**

**Enhanced Feature Engineering**:

* 1. Introduce advanced techniques like PCA (Principal Component Analysis) to reduce dimensionality and improve model efficiency.
  2. Explore deep learning methods for complex tasks like sentiment analysis.

**Model Optimization**:

* 1. Apply advanced hyperparameter tuning techniques such as Bayesian optimization to improve model performance further.
  2. Experiment with ensemble methods (e.g., stacking models) to boost predictive accuracy.

**Incorporating Real-Time Data**:

* 1. Integrate live data feeds into Azure ML pipelines to enable real-time predictions and clustering.
  2. Utilize IoT data for enhanced insights into student behavior and performance.

**Deployment**:

* 1. Deploy models as web services in Azure for practical, scalable use.
  2. Implement RESTful APIs to integrate these models with existing systems for real-world applications.

**Expanding Data Sources**:

* 1. Include additional datasets (e.g., social interactions, extracurricular activities) to refine and diversify analyses.
  2. Perform cross-validation with external datasets to ensure model generalizability.

**Improved Evaluation Metrics**:

* 1. Incorporate advanced evaluation techniques like confusion matrices for classification and ROC curves for a detailed analysis.
  2. Explore clustering validation indices beyond silhouette scores for deeper insights into clustering performance.

By addressing these aspects, the project's impact and applicability can be expanded to tackle a broader range of educational and analytical challenges.