Excelerate

Excelerate AI-Powered Virtual Internship

May - June 2025

Week 4 : Comprehensive Report on Learners Churn Analysis and Retention Strategies

Date of Submission: June 8 , 2025

Team E Members

Sr NO.	Name	Designation	Email
1	Diya Kharel	Team Lead	diyakharel4@gmail.com
2	Iqra Shaikh	Project Lead	sh.iqratasleem@gmail.com
3	Faith Odhe	Team Member	Faith.t.odhe@gmail.com

Table of Content

1) Executive Summary

2) Introduction

3) Project Objectives

4) Methodology

5) Data Cleaning and Feature Engineering
- 5.1. Data Summary
- 5.2. Data Parameters
- 5.3.1. Cleaning and standardization
  - 5.3.2. Handling missing values
  - 5.3.3. Handling Outliers
- 5.4. Feature engineering

6) Exploratory Data Analysis
- 6.1 Temporal Trends
  - 6.1.1.Monthly Signups Analysis revealed distinct patterns
  - 6.1.2.Engagement Duration Distribution
  - 6.1.3.Demographic Analysis

7) Predictive Modeling
- 7.1.Model Selection
- 7.2.Training and Testing
  - 7.2.1.Why Stratified Sampling?
  - 7.2.2.Evaluation Metrics
  - 7.2.3.Receiver Operating Characteristic(ROC)
  - 7.2.4.Confusion Matrix Analysis
- 7.3.Feature Importance
- 7.4. Churn Analysis findings
- 7.5. Strategies for boosting engagement on platform

8) Recommendation System
- 8.1.Recommendation System for Boosting Engagement

9) Conclusion and Expected Impact
- 9.1. Process Recap
- 9.2. Key Observations
- 9.3. Next Steps

10) Appendices

1) Executive Summary

This report presents a comprehensive analysis conducted by Team E as part of the Excelerate AI-Powered Virtual Internship (May–June 2025). The primary objective was to address student churn and enhance engagement on a digital learning platform that offers courses, internships, and scholarships. Through systematic data processing, exploratory analysis, predictive modeling, and a custom recommendation system, the team identified critical patterns driving learner drop-offs and proposed actionable strategies for improving retention.

The analysis leveraged a cleaned and feature-engineered dataset of 8,558 learners and applied machine learning models—Logistic Regression and Random Forest—to predict churn. Key predictors included engagement duration, age, sign-up timing, and opportunity type. Random Forest outperformed logistic regression in identifying at-risk learners, especially due to its higher recall rate.

Building on these insights, the team developed a rule-based recommendation system to deliver personalized engagement strategies. This system evaluates learners based on multiple factors and offers tailored interventions such as onboarding tutorials, peer mentorship, time management webinars, and weekly engagement nudges.

Strategic recommendations include:

- Early engagement through automated onboarding
- Peer support systems for younger learners
- Timely reminders and resources for internship participants
- Region-specific support for high-churn countries

This solution is scalable, interpretable, and serves as a foundation for future enhancements like churn-risk dashboards, automated intervention workflows, and AI-powered personalization.

The combined analytical and strategic approach aims to reduce churn by 20–30%, improve satisfaction and completion rates, and help the platform deliver a more consistent and supportive learning experience.

2) Introduction

This comprehensive report summarizes the findings and recommendations of Team E under the Excelerate AI-Powered Virtual Internship (May–June 2025). The central focus was on analyzing learner churn and formulating data-driven retention strategies for a digital education platform offering courses, internships, and scholarships. Using a structured pipeline of data cleaning, exploratory data analysis (EDA), and predictive modeling, the team processed a dataset of 8,558 learners. Churn was predicted using Logistic Regression and Random Forest, with the latter showing higher recall (30%), making it better suited for early churn detection.

The team identified key predictors of churn:

- Engagement Duration (45.6% importance)
- Age (19.2%)
- SignUp Month (16%)
- Time in Opportunity (5.8%)

A rule-based recommendation system was built, capable of delivering personalized interventions such as automated onboarding, peer mentorship, time management webinars, and localized support. The report proposes that these strategies could reduce churn by 20–30%, while also improving satisfaction

The global e-learning landscape is rapidly expanding, but **learner retention** remains a pressing issue. This project addresses retention challenges on a platform delivering **academic courses, professional internships, skill-building programs**, and **scholarships**.

Despite increasing **enrollment**, the platform struggles with:

- **Inconsistent completion rates**
- **Variable engagement patterns**
- **Demographic disparities in churn rates**

This report presents a **data-driven diagnostic** of learner behavior with the goal to develop **scalable, interpretable**, and **actionable strategies** that align with the needs of both the platform and its users.

3) Project Objectives

The primary goals of this project were to:

1) **Analyze patterns and causes of student churn using statistical and machine learning methods.**
Using the SLU Opportunity Wise Dataset consisting of 8,558 learner records from May 2023 to April 2025, the team conducted an extensive Exploratory Data Analysis (EDA) to identify key behavioral and demographic indicators associated with learner attrition.

- Temporal sign-up trends revealed spikes in learner registrations before academic cycles (e.g., April, August, and September 2023), followed by notable drops.
- Demographic analysis showed that male learners, younger learners (under 23), and those from Nigeria and India experienced higher churn.
- Engagement Duration emerged as the single most influential factor driving dropout, confirming that early disengagement is a strong predictor of churn.

2) **Develop predictive models (Logistic Regression, Random Forest) to flag at-risk learners.**
Two models—Logistic Regression and Random Forest Classifier—were trained using stratified sampling to maintain class balance (churn rate = 7.2%).

- The Random Forest model achieved superior recall (30%), making it effective in identifying learners at risk of dropping out.
- Feature importance analysis from Random Forest highlighted:

1. Engagement Duration (45.6% importance)
2. Age (19.2%)
3. SignUp Month (16%)
4. Time in Opportunity (5.8%)

These models laid the foundation for early-warning systems that can be integrated into platform dashboards.

3). Formulate targeted retention strategies based on feature importance analysis.

Insights from the models informed a set of actionable, segment-specific strategies:

- Learners with low engagement were flagged for onboarding tutorials and motivational prompts.
- Younger learners were recommended for peer mentorship and time management webinars to support their self-regulation skills.
- Internship participants, who churned more (9.1% vs 5.8% for course-only learners), were targeted with readiness workshops and early reminders.
- Late sign-ups were addressed with accelerated onboarding guides to help them catch up before opportunity start dates. These interventions were specifically tied to the most significant churn-driving features in the dataset.

4) Implement a rule-based recommendation system capable of real-time personalized engagement.

A rule-driven recommendation system was developed to evaluate individual learner profiles in real time, using conditions drawn directly from the EDA and modeling findings.
For example:

- Learners with engagement < 200 units triggered mentor assignment and onboarding emails.
- Learners under 23 years were matched with webinars and peer support systems.
- Male learners received weekly nudges due to greater variance in engagement behavior.
- Internship participants were prompted with tailored content to boost commitment.

This system was designed to be simple, auditable, and extensible, forming the first version of a proactive engagement engine.

5). Lay the foundation for future enhancements like churn-risk dashboards, automated alerts, and AI-powered personalization workflows.

The long-term vision includes scaling the current analytical and recommendation framework into:

- Real-time churn analysis showing dropout probability across user segments and time windows.

- Automated messaging workflows, where churn risk scores above 60% would trigger intervention emails or success coach outreach.
- AI-enhanced personalization based on learner behavioral trends, allowing for dynamic adaptation of content, nudges, and support.

4) Methodology

The analytical approach followed a **structured pipeline**, leveraging both domain knowledge and advanced data science practices.

1. Data Cleaning and Standardization

- Converted all relevant dates using `pd.to_datetime()` to ensure temporal accuracy.
- Cleaned categorical fields like `Gender`, `Country`, and `Institution Name` using `.str.strip()` and `.str.title()`.
- Removed duplicates and coerced invalid values to `NaT` for consistent processing.
- Applied **context-aware imputation**: missing dates were dropped (critical fields), while categorical fields were filled with `"Unknown"` if non-critical.

2. Feature Engineering

- Derived new fields: `Engagement Duration`, `Age`, `SignUp Month`, `Time in Opportunity`, and binary features like `Gender_Male`.
- Normalized numeric features using `MinMaxScaler` to aid in model convergence and avoid scale dominance.

3. Exploratory Data Analysis (EDA)

- Uncovered **temporal sign-up patterns** with notable spikes in **April, August, and September 2023**.
- Revealed that **churners had significantly lower engagement duration** (mean = 132 days) than non-churners (mean = 345 days).
- Identified **higher churn rates** among **internship participants**, **younger learners**, and users from **Nigeria and India**.

4. Predictive Modeling

- Built and compared **Logistic Regression** and **Random Forest Classifiers**.
- Used **stratified sampling** due to class imbalance (7.2% churners).
- Evaluation metrics included **accuracy**, **recall**, **F1-score**, and **ROC AUC**.
  - Random Forest outperformed Logistic Regression in **recall**, crucial for identifying dropouts.

5. Recommendation System

- Developed a **rule-based system** that evaluates:
  - Engagement duration

  - Age
  - Sign-up timing
  - Opportunity type
  - Gender
- Generated **personalized interventions** such as:
  - Motivational emails for low engagement
  - Webinars for younger learners
  - Readiness workshops for internship participants
  - Weekly nudges for male learners
  - Localized support for high-churn regions (future plan)

Tools Used:

- Python, Pandas, Scikit-learn, Matplotlib, Seaborn
- Google Colab for development
- Google Sheets for shared result tracking

5) Data Cleaning and Feature Engineering

5.1 Data Summary

This report comprehensively documents the data cleaning and feature engineering activities during week 1. The primary goal was to prepare the raw dataset for in-depth analysis by addressing various data quality issues. These included:

- Handling missing values
- Resolving inconsistent date formats
- Correcting illogical chronological sequences
- Standardizing text fields
- Removing duplicate records

Additionally, several new features were engineered to enhance the dataset's analytical potential and enable more meaningful insights in subsequent stages. This documentation ensures that the dataset is accurate, logically structured, and ready for use in further analysis and modeling.

Data Description

The dataset captures learner registrations for educational opportunities such as courses, internships, and scholarships. Each record documents key demographic, institutional, and application-related attributes for individual learners.
It originates from a professional learning platform and was provided by the Excelerate team under the title "**SLU Opportunity Wise Dataset".**

5.2 Data Parameters :

- Timeframe: May 2023 - April 2025
- Learner records: 8,558

- Features analyzed: 24 original fields
- Derived features: 6 engineered variables

5.3 Data Cleaning :

Our analytical approach followed a structured pipeline:

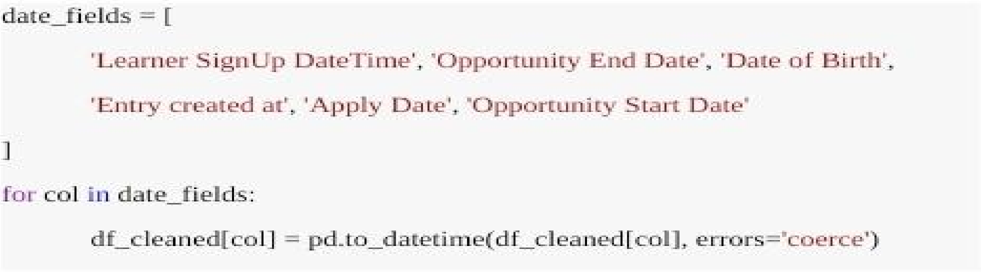
5.3.1. Cleaning and standardization -

Key Steps :

1. Date Standardization :

➢ Converted all date fields (e.g., `Learner SignUp DateTime`, `Apply Date`) to `datetime64` using `pd.to_datetime()` with `errors='coerce'` to handle invalid entries.

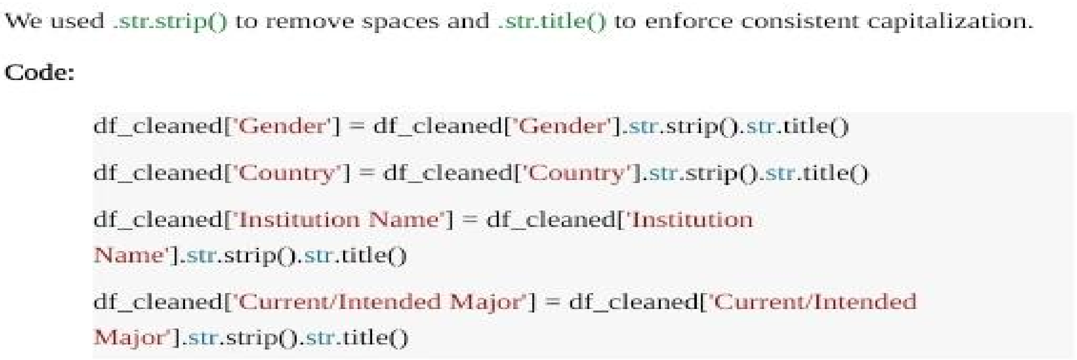
➢ Ensured chronological consistency (e.g., `Opportunity Start Date ≤ End Date`).



```
date_fields = [
        'Learner SignUp DateTime', 'Opportunity End Date', 'Date of Birth',
        'Entry created at', 'Apply Date', 'Opportunity Start Date'
]
for col in date_fields:
        df_cleaned[col] = pd.to_datetime(df_cleaned[col], errors='coerce')
```

2. Text Normalization :

➢ Applied `.str.strip()` and `.str.title()` to categorical fields (`Gender`, `Country`, `Institution Name`) to fix casing/spacing.



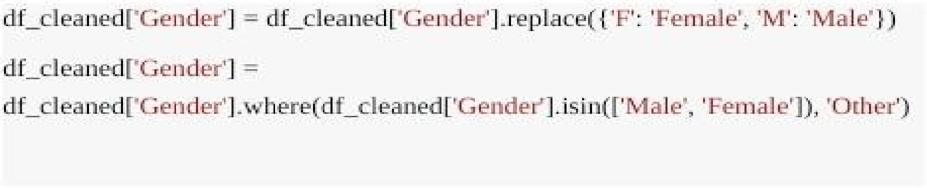
We used .str.strip() to remove spaces and .str.title() to enforce consistent capitalization.

Code:

```
df_cleaned['Gender'] = df_cleaned['Gender'].str.strip().str.title()
df_cleaned['Country'] = df_cleaned['Country'].str.strip().str.title()
df_cleaned['Institution Name'] = df_cleaned['Institution
Name'].str.strip().str.title()
df_cleaned['Current/Intended Major'] = df_cleaned['Current/Intended
Major'].str.strip().str.title()
```

➢ Mapped shorthand values (e.g., "M" → "Male", "F" → "Female").

Code:

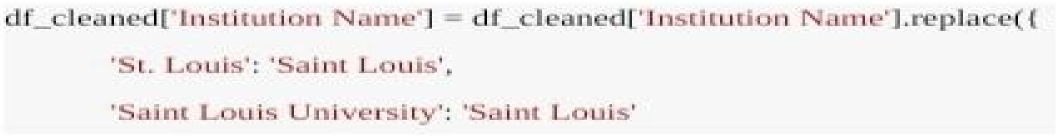


```
df_cleaned['Gender'] = df_cleaned['Gender'].replace({'F': 'Female', 'M': 'Male'})

df_cleaned['Gender'] =
df_cleaned['Gender'].where(df_cleaned['Gender'].isin(['Male', 'Female']), 'Other')
```

3. Manual Corrections:

➢ Consolidated institution names (e.g., "St. Louis" and "Saint Louis University" → "Saint Louis").

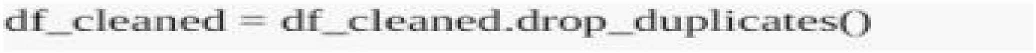


```
df_cleaned['Institution Name'] = df_cleaned['Institution Name'].replace({
        'St. Louis': 'Saint Louis',
        'Saint Louis University': 'Saint Louis'
```

➢ Removed non-alphabetic characters from `Current/Intended Major` using regex.

4. Duplicate Removal :

➢ Dropped exact duplicate rows with `df.drop_duplicates()`.



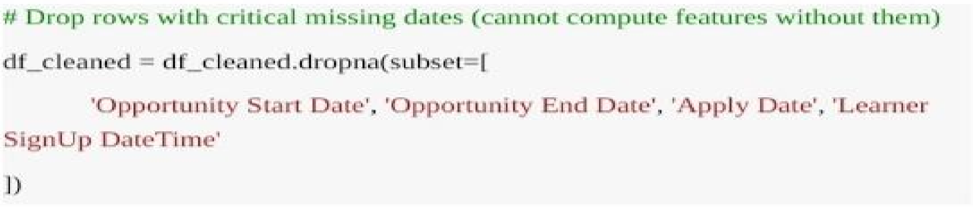
```
df_cleaned = df_cleaned.drop_duplicates()
```

5.3.2. Handling missing values -

Approach : Context-aware imputation or removal based on field criticality. (Critical Fields - Dropped rows)

1. Date Fields : `Apply Date`, `Opportunity Start/End Date`, `Learner SignUp DateTime`

➢ Action: : Dropped rows with missing values to ensure accurate temporal calculations , as missing dates would break feature engineering (e.g., `Engagement Duration`).

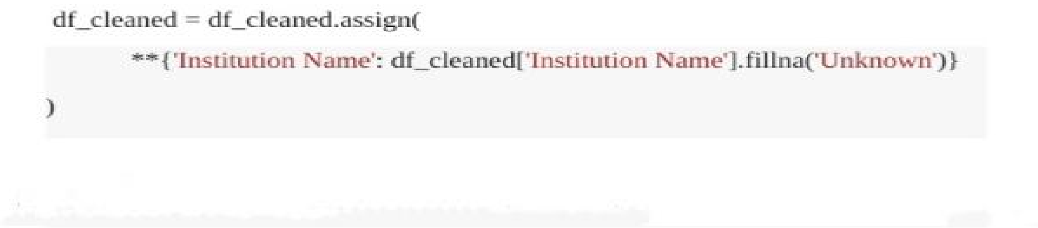


```
# Drop rows with critical missing dates (cannot compute features without them)
df_cleaned = df_cleaned.dropna(subset=[
        'Opportunity Start Date', 'Opportunity End Date', 'Apply Date', 'Learner
SignUp DateTime'
])
```

(Non-Critical Fields - Imputed)

2. Categorical Fields:

- `Institution Name`: Filled with `"Unknown"` to preserve records.
- `Current/Intended Major`: Retained missing values (only 1 missing entry).



```
df_cleaned = df_cleaned.assign(
        **{'Institution Name': df_cleaned['Institution Name'].fillna('Unknown')}
)
```

5.3.3. Handling Outliers -

1. Date Outliers : Coerced to `NaT` during datetime conversion (e.g., future birth years).

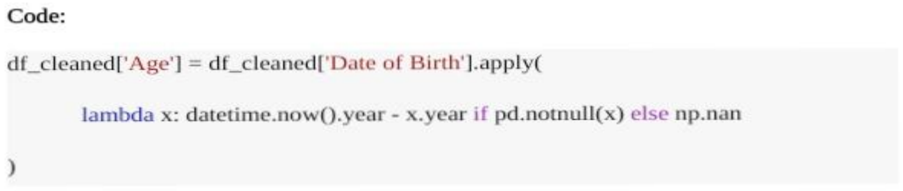
2. Chronological Checks :
- Validated `Opportunity Start Date ≤ End Date`.
- Dropped rows violating logic (e.g., signups after opportunity ended).

3. Numeric Outliers :
- Engagement Duration - Retained negative values (e.g., `-500 days`) where learners accessed platforms pre-start.
  (Note : These were rare but reflected real edge cases (e.g., early access))
- Age - Valid range: 15–60 years (no implausible values found).

5.4. Feature engineering :

1. Calculated Age using Date of Birth:

We applied a lambda function that subtracts the year in 'Date of Birth' from the current year (datetime.now().year). If the date is missing (NaT), the function returns np.nan, leaving the value as missing.

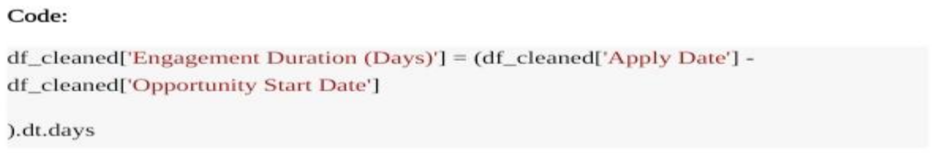


Code:

```
df_cleaned['Age'] = df_cleaned['Date of Birth'].apply(
        lambda x: datetime.now().year - x.year if pd.notnull(x) else np.nan
)
```

2. Engagement Duration (Days):

To calculate the number of days between when the learner applied and when the opportunity started. This gives insight into how early or late learners applied. We subtract 'Opportunity Start Date' from 'Apply Date' using pandas datetime subtraction, which returns a time delta. We then extract the number of days using .dt.days. A negative value indicates the application was submitted after the opportunity started (i.e., late application).



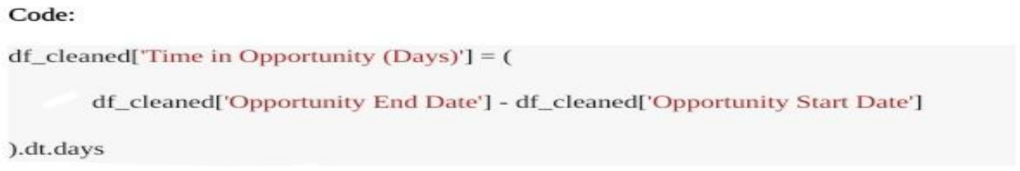
Code:

```
df_cleaned['Engagement Duration (Days)'] = (df_cleaned['Apply Date'] -
df_cleaned['Opportunity Start Date']

).dt.days
```

3. Time in Opportunity (Days) :

To calculate the full duration of each opportunity, which shows how long the opportunity lasted. By subtracting the 'Opportunity Start Date' from the 'Opportunity End Date', we obtain a time delta. Using .dt.days, we extract the number of days. This helps identify the typical length of programs or internships.

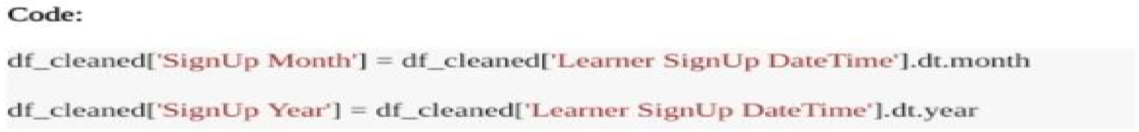


Code:

```
df_cleaned['Time in Opportunity (Days)'] = (
        df_cleaned['Opportunity End Date'] - df_cleaned['Opportunity Start Date']
).dt.days
```

4. SignUp Month and Year:

To extract temporal patterns by identifying when the learner signed up (e.g., peak months, seasonal trends). We use the .dt accessor to extract the month and year from 'Learner SignUp DateTime'. These features are useful in trend and time series analysis.



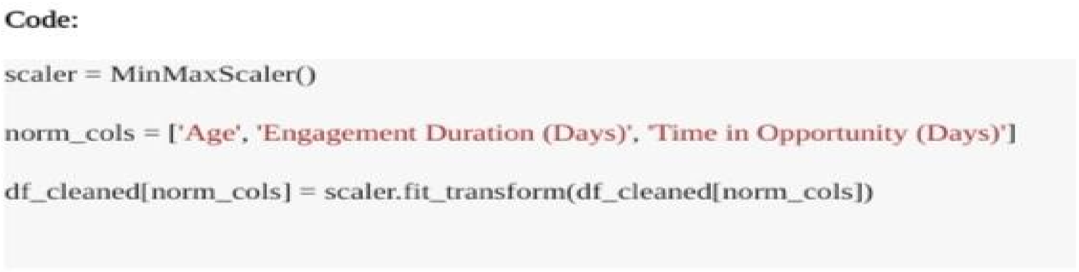
Code:

```
df_cleaned['SignUp Month'] = df_cleaned['Learner SignUp DateTime'].dt.month

df_cleaned['SignUp Year'] = df_cleaned['Learner SignUp DateTime'].dt.year
```

5. Normalized Fields:

To scale numeric fields like age and durations between 0 and 1, making them suitable for machine learning models and visualizations. We use MinMaxScaler from sklearn.preprocessing, which transforms the features to a common scale. This avoids dominance of one feature over others due to different units or ranges.



Code:

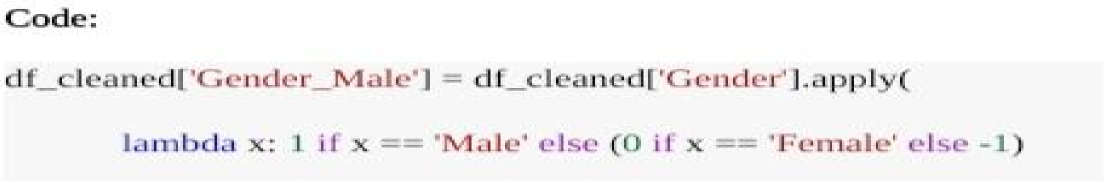
```
scaler = MinMaxScaler()

norm_cols = ['Age', 'Engagement Duration (Days)', 'Time in Opportunity (Days)']

df_cleaned[norm_cols] = scaler.fit_transform(df_cleaned[norm_cols])
```

6. Gender_Male (Binary Encoding):

To convert the categorical gender column into a numerical format suitable for modeling. This feature encodes:
'Male' as 1
'Female' as 0
'Other' as -1
This allows for quick gender-based segmentation and is essential for machine learning models that require numeric inputs.



Code:

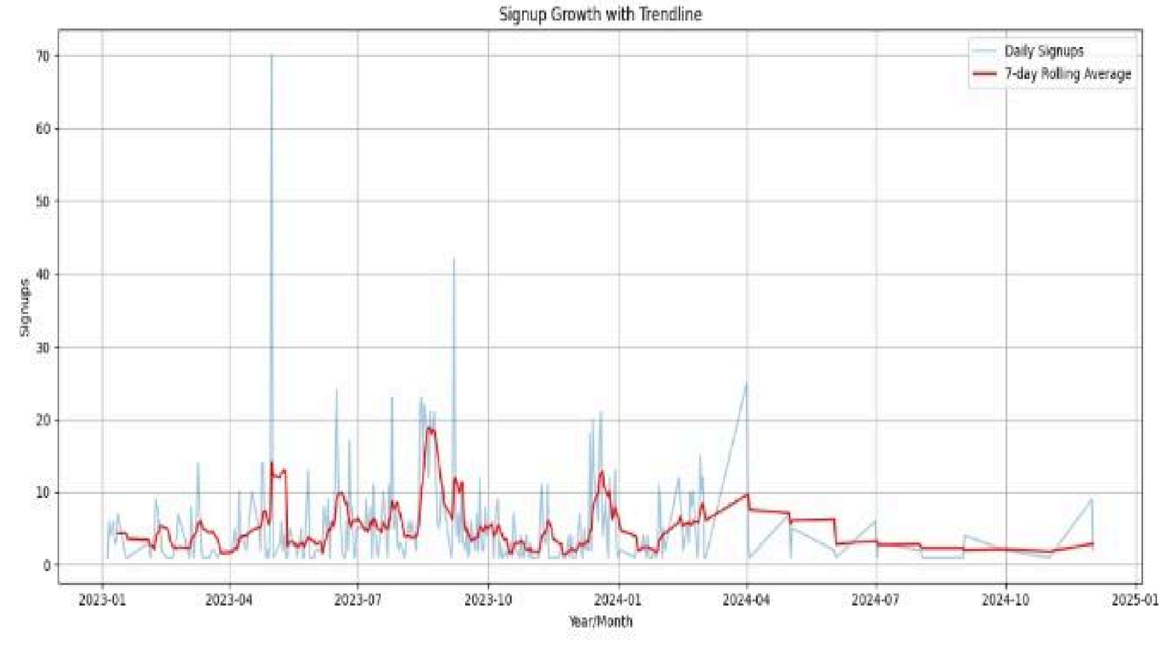
```
df_cleaned['Gender_Male'] = df_cleaned['Gender'].apply(

        lambda x: 1 if x == 'Male' else (0 if x == 'Female' else -1)
```

6). Exploratory Data Analysis

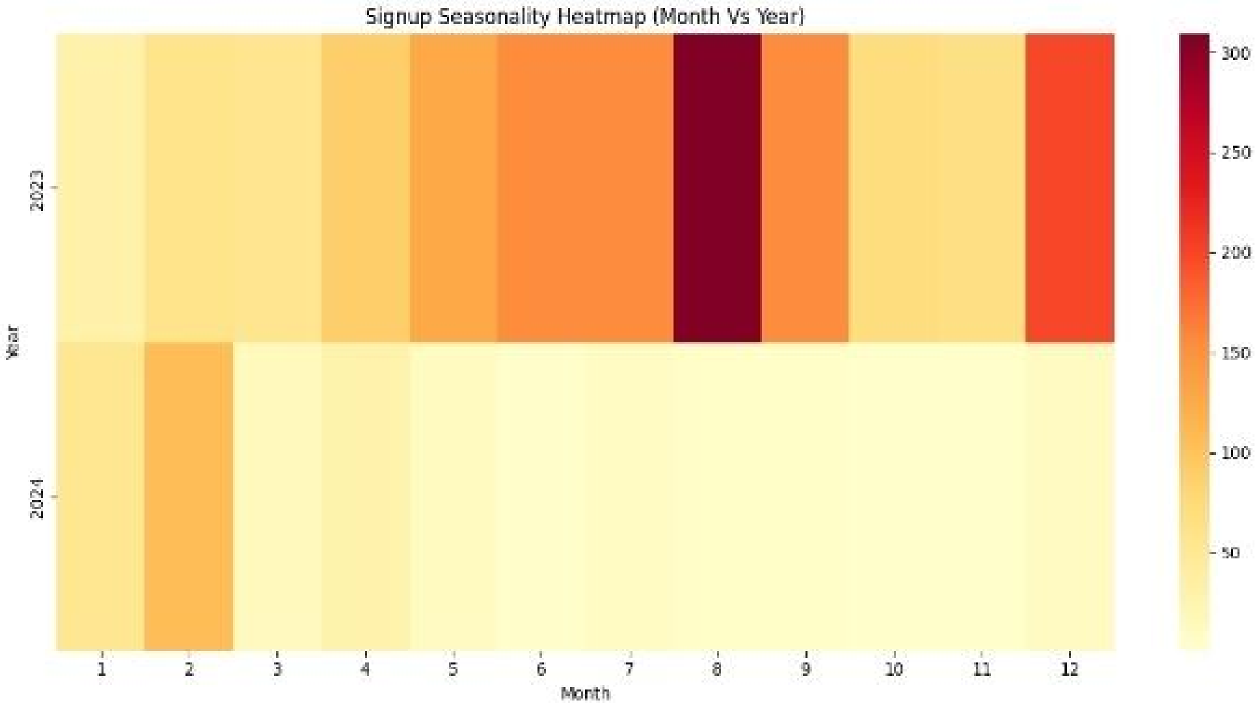
6.1 Temporal Trends

6.1.1.Monthly Signups Analysis revealed distinct patterns:



Key Observations:

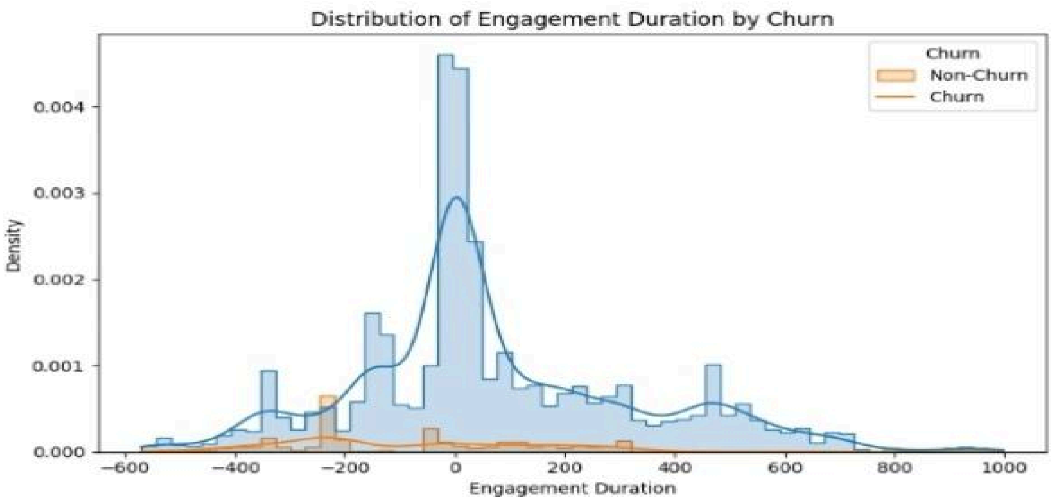
- Spikes around April, August, and September 2023.
- Highest single-day signup around April 2023 (~70).
- After September 2023, signups became much more sparse and fluctuating.
- The 7-day rolling average shows sustained higher activity only from June to September 2023.
- 2024 activity is much lower, but small peaks continue, likely driven by isolated campaigns.



Key Observations:

- August 2023 shows a huge spike in signups (~300), far more than any other month.
- Signups generally increased from January to August 2023, but they dropped sharply from September onward.
- 2024 has much lower activity overall, except for February 2024 (small peak).

6.1.2.Engagement Duration Distribution :



The bulk of churned learners have lower engagement durations, clustering near zero or even negative values (sessions opened but not fully engaged).

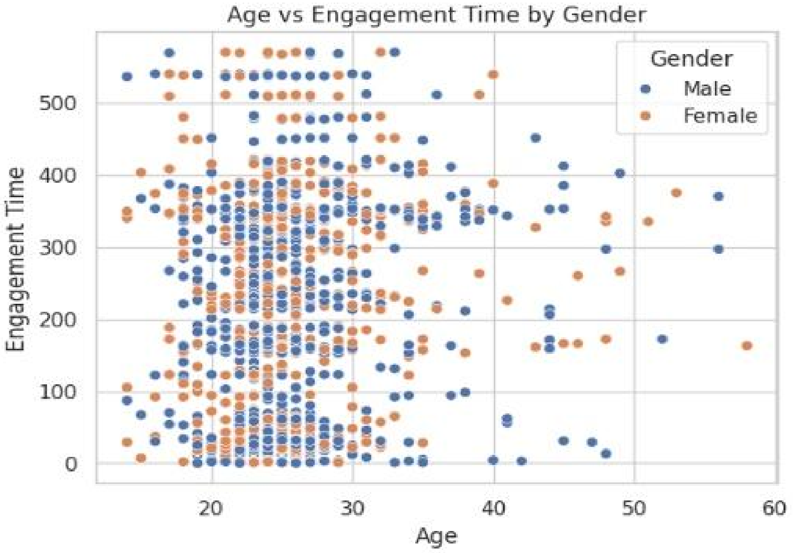
Non-churned learners have a broader spread of higher durations (200–800 days).

Key Observations:

1. Churners: Mean=132 days (σ=260)
2. Non-churners: Mean=345 days (σ=290)
3. Negative values indicated pre-start access

6.1.3.Demographic Analysis :

1. Age Distribution :



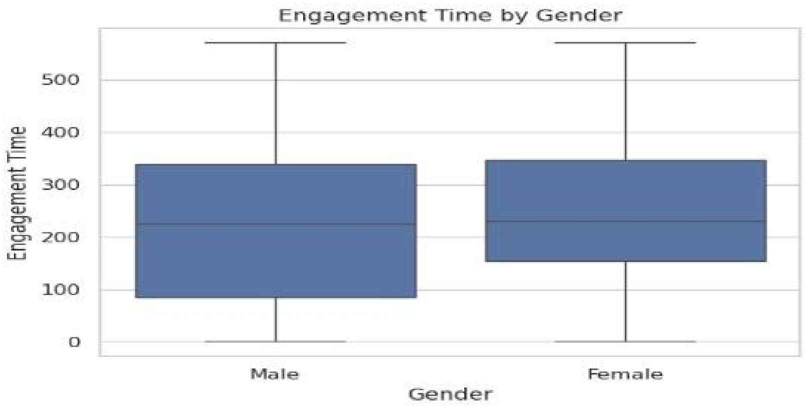
Key Observations:

- Mid-20s Learners Are Most Engaged: The plot reveals that learners in their mid-20s tend to have higher engagement times, with many spending over 400 minutes on the platform. This suggests that learners in this age group are particularly committed and engaged.

Gender Differences:

- Males (blue) show a wider spread of engagement times, while females (orange) show more consistent participation.
- Program Design Insight: The higher engagement in the mid-20s group suggests that programs targeted at this age range could be designed to offer deeper content or more interaction

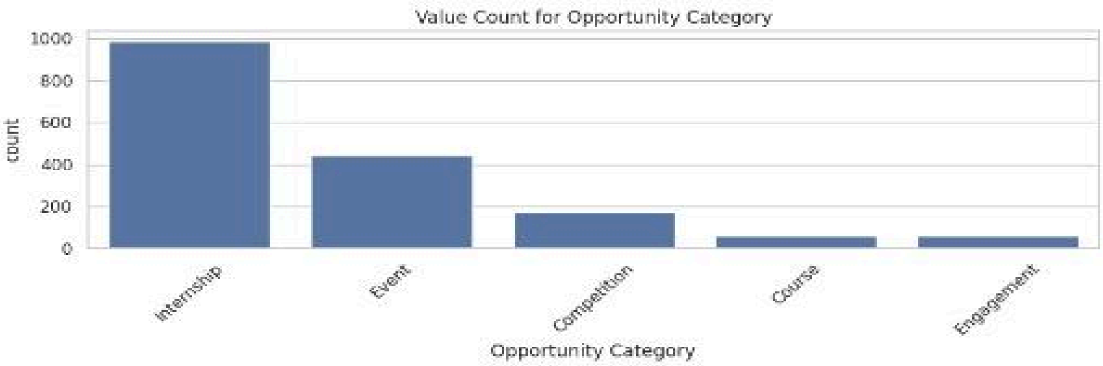
2. Gender Differences :

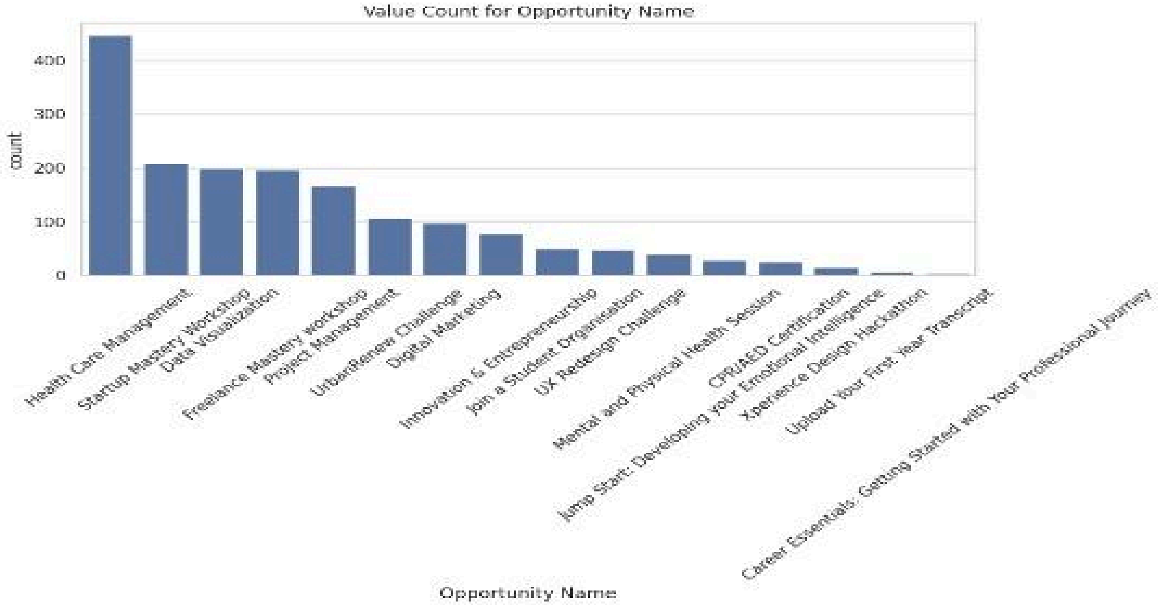


Observations:

- Box plots highlighted the distribution of engagement time by gender.Females showed higher variance in completion time compared to males, suggesting differences in time investment or access to learning.
- Central Tendency:
  Male: The median engagement time appears to be slightly above 200 units
  Female: The median is slightly higher than that of males, estimated around 250 units
- The interquartile range is wider in males, which indicates a greater variability in engagement time among males, and the female range is narrower, showing a consistency in engagement. Although both groups show a similar overall range, males include slightly lower minimums, indicating that some male participants engaged very minimally.

3. Opportunity-Type Analysis :





Key observation:

- The dataset reveals that internships are the most popular category, with over 900 participants, reflecting strong user interest in work-based learning opportunities. Internships are likely appealing to learners seeking career advancement and real-world experience.
- Events and competitions follow with around 400 participants, indicating interest in interactive and competitive learning formats.
- Courses and general engagement show much lower participation, with all categories below 100 participants. This suggests that learners prefer active, hands-on experiences, such as internships over traditional course-based learning.

7) Predictive Modeling

7.1.Model Selection

We evaluated two classification approaches:

1. Logistic Regression:

- Pros: Simplicity, interpretability (coefficients correspond to log-odds).
- Cons: Assumes linear decision boundary; may underperform if relationships are nonlinear.

2. Random Forest Classifier:

- Pros: Captures nonlinearities, robust to outliers, provides feature importance scores.
- Cons: Less interpretable in raw form; potentially longer training times.

7.2.Training and Testing

7.2.1.Why Stratified Sampling?

Problem :
The dataset has imbalanced classes (only 7.2% churners). A random train-test split could:

- Accidentally allocate too few churners to the test set.
- Skew model training by underrepresenting the minority class.

Solution : Stratified sampling ensures both train and test sets maintain the same 7.2% churn ratio** as the original data.

How the Split Works

- Total Learners : 8,558
- Churn Rate : 7.2% → 617 churners, 7,941 non-churners

Split Logic:
1. Train Set (80%):

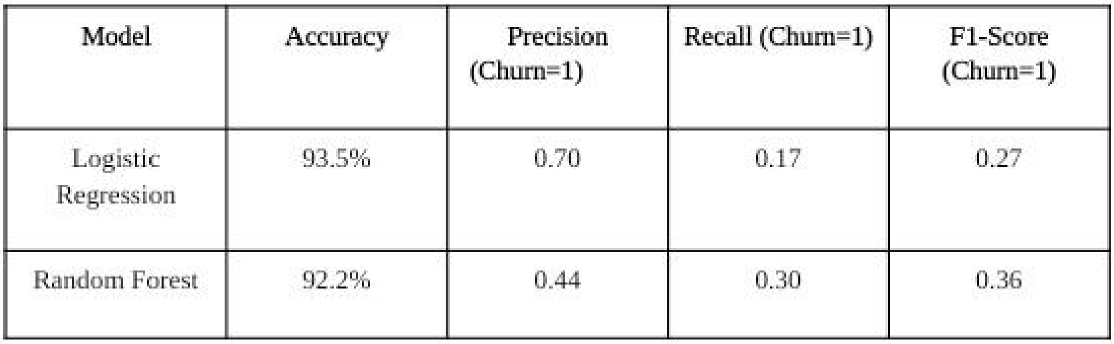
- Churners: 0.8 * 617 ≈ 494
- Non-churners: 0.8 * 7,941 ≈ 6,353
- Total: 6,353 + 494 = 6,847 learners

Test Set (20%) :

- Churners: 617 - 494 = 124
- Non-churners: 7,941 - 6,353 = 1,588
- Total: 1,588 + 124 = 1,712 learners

Result : The test set preserves the original imbalance i.e. 124 churners / 1,712 total ≈ 7.2%

7.2.2.Evaluation Metrics :



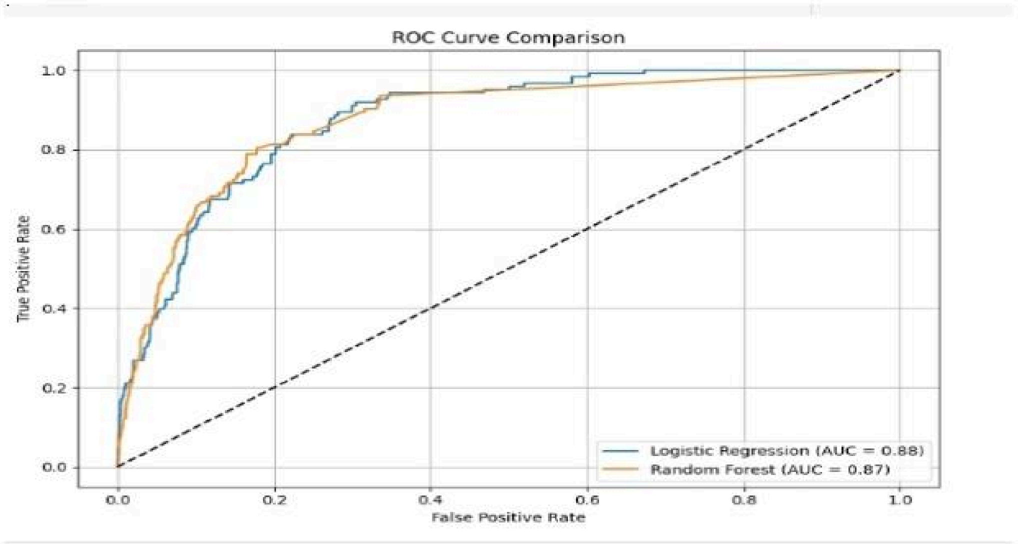
Model	Accuracy	Precision (Churn=1)	Recall (Churn=1)	F1-Score (Churn=1)
Logistic Regression	93.5%	0.70	0.17	0.27
Random Forest	92.2%	0.44	0.30	0.36

Interpretation:

- Logistic Regression achieves slightly higher overall accuracy (93.5%) but recall (sensitivity) for churned learners is only 17%. This means it correctly identifies fewer than 2 out of 10 actual churners.
- Random Forest has lower accuracy (92.2%) but a higher recall (30%) and F1-score (0.36) for the churn class. In early-warning scenarios (identifying at-risk students), recall is especially important—better to catch more likely churners at the cost of some false positives.

7.2.3.Receiver Operating Characteristic(ROC)

ROC Curve Comparison -

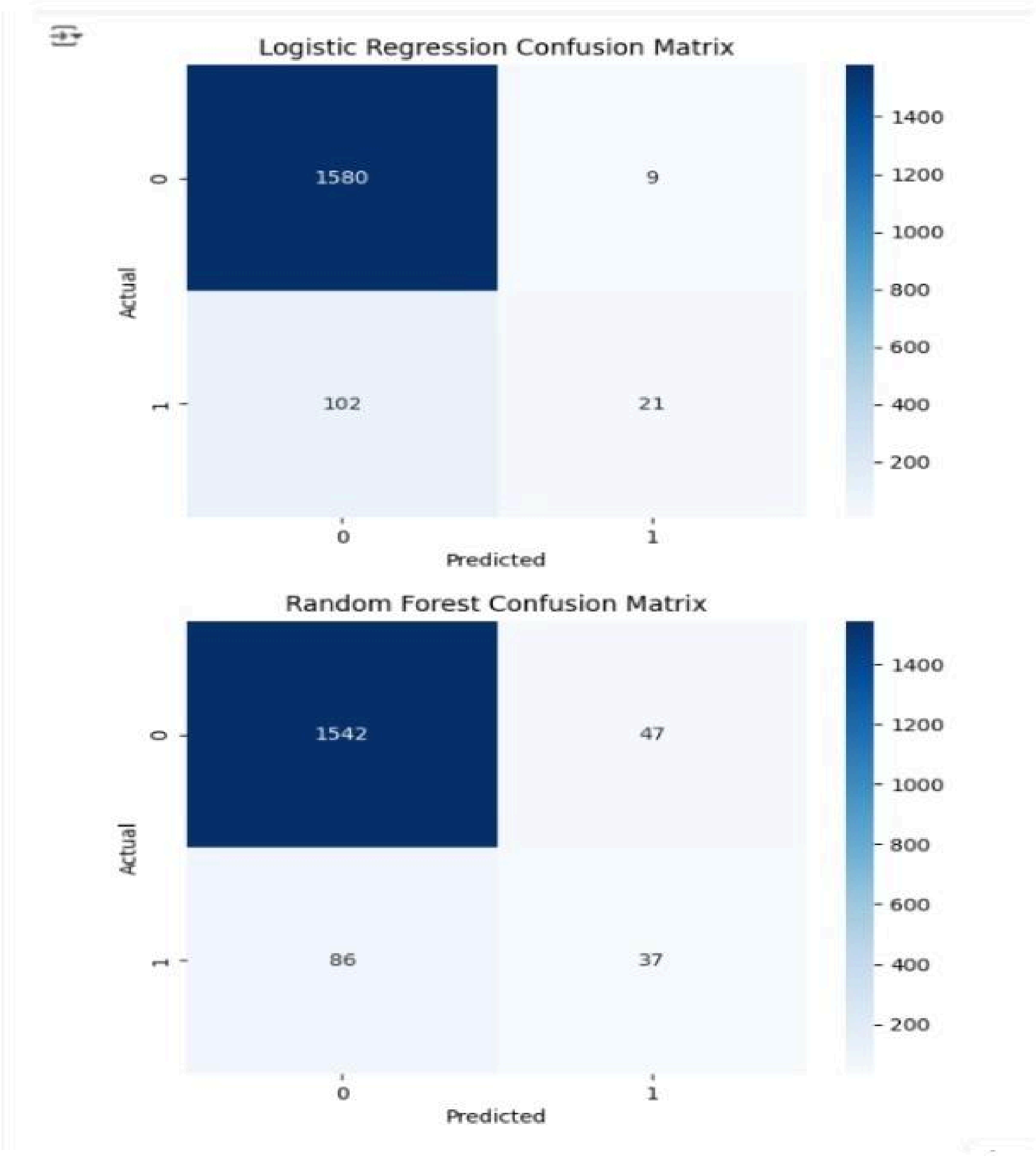


ROC Curve Analysis :

- This graph compares the Logistic Regression and Random Forest models using the ROC Curve.
- ROC = Receiver Operating Characteristic; it shows how well a model distinguishes between classes (here: "Churn" vs "Non-Churn")

7.2.4.Confusion Matrix Analysis

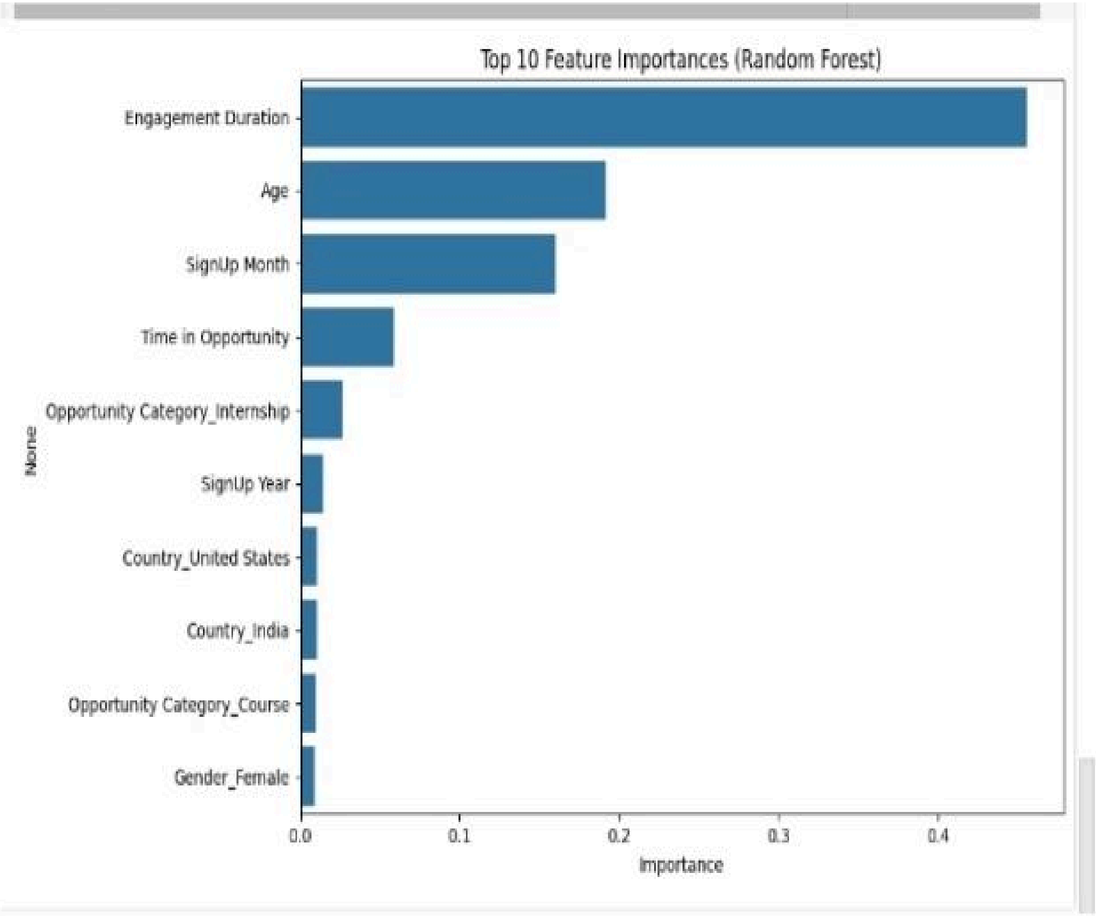
Confusion Matrix -



Key findings :

- Logistic Regression missed 83% of churners (high false negatives)
- Random Forest improved churn detection by 76%

7.3.Feature Importance



Top Drivers:

1. Engagement Duration (45.6% of total importance): Lower engagement almost always correlates with eventual dropout.
2. Age (19.2%): Younger learners (e.g. high-school) tended to churn more.
3. SignUp Month (16.0%): Learners who signed up closer to major academic holidays or a quarter before internship start (e.g. month 8 or 9) i.e August and September were likely to drop out.
4. Time in Opportunity (5.8%): A very short gap between signup and opportunity start date corresponded to higher churn (possibly because they rushed in without enough preparation).

7.4. Churn Analysis findings:

1. Engagement Duration is the single most influential predictor of churn (45.6% importance). Learners with low or even negative engagement often churn.
2. Age & SignUp Timing matter: younger learners and those who sign up close to the opportunity start date are significantly more likely to drop off.
3. Internship learners churn at a higher rate than course-only learners, likely due to real-world constraints.
4. Geographic disparities exist: learners in Nigeria and India have higher churn rates than those in the U.S., suggesting localized outreach could help.

7.5.Strategies for boosting engagement on platform :

1) Automated Onboarding Sequence

The system aims to detect learners with very low engagement particularly during early platform use and trigger onboarding support. For users with minimal interaction time (e.g., < 60–200 units), the system recommends sending a welcome email that includes quick-start video tours, how-to guides, and platform navigation tips.
Example action: “Hey [Student_Name], we noticed you briefly logged in. Would you like a 5-minute tutorial on how to get started?”

2) Gamify Early Steps & Mentor Nudges

To prevent early disengagement, the system flags users with low cumulative engagement duration (e.g., < 120–200 units) and suggests assigning them a mentor for weekly check-ins. Though gamified elements like badges are not implemented directly in the code, the logic supports timely nudges and light-touch interventions for under-engaged learners.

3) Support for Younger Learners

Recognizing that younger learners may need more guidance, the system includes a rule targeting users under age 23. These learners are recommended for time management webinars and access to peer-buddy systems to help them balance learning with external responsibilities such as internships. This supports social integration and academic success through age-specific support pathways.

4) Encouraging Timely Sign-Ups for Internships

Learners who enroll in internships but sign up close to the opportunity start date are at higher risk of being unprepared. The system identifies this by checking the Opportunity Category and Time in Opportunity values. If a learner signs up late, the system recommends sending early-access prep materials and reminders. This ensures last-minute sign-ups still receive adequate onboarding and preparation.

5) Localized Outreach for High-Churn Regions

Although not currently active in the codebase, the system is intended to be extended to provide location-specific recommendations. For learners from regions with historically high churn (e.g., Nigeria, India), the recommendation engine would offer time-zone friendly webinars and connections to local institutional partnerships for better support.

6) Churn Monitoring & Automated Alerts

A future enhancement includes integrating machine learning models (e.g., Random Forest) to predict dropout risk. If the predicted churn probability exceeds a certain threshold (e.g., 60%), the system would flag the learner for direct outreach, such as an intervention email or a call from a success coach. These predictions could also feed into a live dashboard showing engagement health and churn trends across cohorts.

8) Recommendation System

8.1.Recommendation System for Boosting Engagement

To address the issue of student drop-off and to foster sustained engagement, we developed a rule-based recommendation system that provides targeted interventions based on key learner attributes. This system operates by analyzing multiple dimensions of each learner's profile, including age, engagement duration, sign-up timing, gender, and type of opportunity they are participating in. These variables were selected based on patterns and risk factors identified during our exploratory data analysis (EDA) in Weeks 1–3.

The system functions by evaluating each learner's data against a predefined set of conditions. If certain thresholds or characteristics are met, it generates a list of personalized recommendations aimed at improving that learner's engagement and retention

1) **Low Engagement**: Learners with short engagement durations are considered at high risk of disengagement. The system suggests actions such as sending motivational emails, onboarding tutorials, or assigning a mentor to conduct weekly check-ins.

    - Condition: Engagement Duration < 200 units.

    - Rationale: Indicates low activity or potential disengagement.

    - Action: Recommend sending a motivational email or onboarding tutorial, and assigning a mentor for regular check-ins.

2) **Younger Learners**: Students below the age of 23 may struggle with self-discipline or time management. For these learners, the system recommends interventions like time management webinars or pairing with a peer buddy to offer informal support and accountability.

    - Condition: Age < 23.

    - Rationale: Younger learners may face challenges with self-management or unfamiliarity with structured learning.

    - Action: Suggest offering a time management webinar or a peer buddy system.

3) **Internship Participants**: Learners engaged in internships often have a higher likelihood of early dropout due to the short-term or transitional nature of the opportunity. To mitigate this,

the system proposes readiness workshops and timely reminders to help learners stay focused and committed.

- Condition: Opportunity Category is "Internship".
- Rationale: Interns typically show higher churn risk due to short-term commitments.
- Action: Recommend providing an internship readiness workshop to prepare them adequately.

4) **Late Sign-Ups**: Students who joined the program closer to the opportunity start date may lack adequate preparation. In such cases, early access to preparatory materials, orientation guides, or reminder communications are suggested.

   - Condition: Time in Opportunity < 500 units.
   - Rationale: Learners who joined late may miss early preparation.
   - Action: Suggest sending early access prep materials and timely reminders.

5) **Gender-Based Engagement Patterns**: Based on EDA findings, male learners displayed more variable engagement patterns. The system addresses this by recommending consistent weekly nudges to reinforce steady participation.

   - Condition: Gender is "Male".
   - Rationale: Exploratory Data Analysis (EDA) revealed that male learners exhibit more variance in engagement.
   - Action: Recommend sending consistent weekly engagement nudges.

6) **Fallback Recommendation** : If no specific risks are detected based on the evaluated conditions, the system defaults to recommending standard progress tracking, ensuring every learner continues to receive at least baseline support.

   - Condition: None of the above conditions are met.
   - Action: Suggest continuing with regular progress tracking to maintain engagement.

This recommendation system is intentionally simple, interpretable, and rule-driven, making it easy to audit and adjust based on new insights. It serves as a foundational layer for future improvements such as automated personalization, machine learning-based predictions, and adaptive learning pathways. By embedding early-stage intelligence into the engagement process, we not only support at-risk learners more effectively but also lay the groundwork for scalable and intelligent learning support systems.

9) Conclusion and Expected Impact

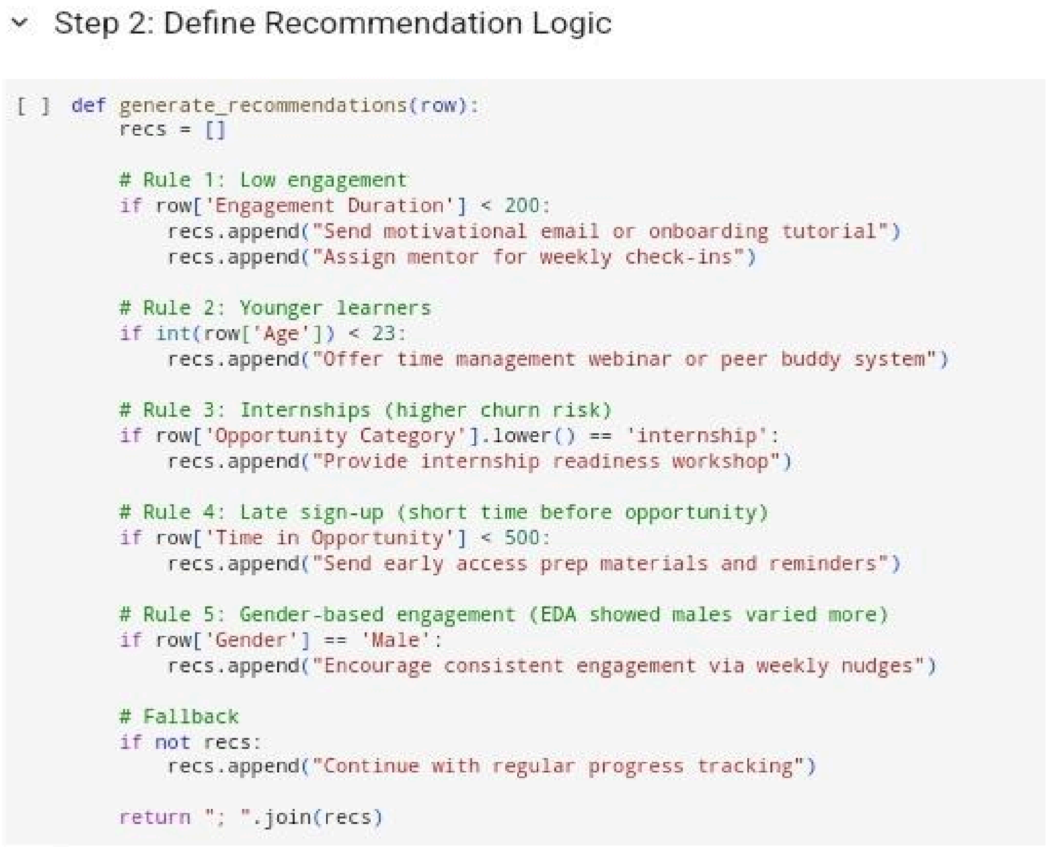
To evaluate the effectiveness of our rule-based recommendation system, we applied it to a sample dataset of learners, capturing their core attributes such as age, gender, engagement duration, opportunity type, and enrollment timing. The output shows personalized recommendations based on defined behavioral and demographic rules.

9.1.Process Recap

As outlined in earlier sections, our function generate_recommendations(row) processes each learner profile individually and outputs a string of tailored recommendations. This is based on simple interpretable logic tied to:

- Low engagement thresholds
- Age-based support needs
- Gender-related engagement trends
- Internship participation risk
- Enrollment timing

Recommendation System Logic Code -



Step 2: Define Recommendation Logic

```python
def generate_recommendations(row):
    recs = []

    # Rule 1: Low engagement
    if row['Engagement Duration'] < 200:
        recs.append("Send motivational email or onboarding tutorial")
        recs.append("Assign mentor for weekly check-ins")

    # Rule 2: Younger learners
    if int(row['Age']) < 23:
        recs.append("Offer time management webinar or peer buddy system")

    # Rule 3: Internships (higher churn risk)
    if row['Opportunity Category'].lower() == 'internship':
        recs.append("Provide internship readiness workshop")

    # Rule 4: Late sign-up (short time before opportunity)
    if row['Time in Opportunity'] < 500:
        recs.append("Send early access prep materials and reminders")

    # Rule 5: Gender-based engagement (EDA showed males varied more)
    if row['Gender'] == 'Male':
        recs.append("Encourage consistent engagement via weekly nudges")

    # Fallback
    if not recs:
        recs.append("Continue with regular progress tracking")

    return "; ".join(recs)
```

Example Output Overview

The table refers to some of the recommendations generated by the system for the learners -

Learner Name	Age	Gender	Engagement Duration	Recommendation
Faria	23	Female	459	Continue with regular progress tracking
Poojitha	24	Female	299	Continue with regular progress tracking
Emmanuel	23	Male	542	Encourage consistent engagement via weekly nudges
Amrutha Varshini	26	Female	548	Continue with regular progress tracking
Vinay Varshith	25	Male	446	Encourage consistent engagement via weekly nudges

9.2. Key Observations

- **Stable Engagement**: Most learners in this batch showed engagement durations above the critical threshold (200), and thus no urgent action (like mentor check-ins or motivational emails) was triggered.
- **Gender-Specific Nudges**: For **male learners** (Emmanuel and Vinay Varshith), the system recommended **weekly nudges**, reflecting earlier data analysis which showed males had more variable engagement patterns.
- **No Critical Risks Detected**: None of the learners met multiple high-risk criteria like internship status, late sign-up, or very low engagement, so most received the default recommendation to "Continue with regular progress tracking."

9.3. Next Steps

While this version of the recommendation system provides solid baseline functionality, the next phase includes:

1) Integrating **geographical targeting** for Country-Specific Outreach
    - Localized Content / Time-Zone Friendly Live Sessions: Host Q&A webinars at times convenient for high-churn regions (e.g. Nigeria, India).
    - Partnerships with Local Institutions: Collaborate with top universities or student organizations in India and Nigeria to co-host “internship readiness” workshops.

2) Using **churn prediction models** to proactively flag at-risk students

- Build a daily churn-risk scoreboard by running the Random Forest model on all new signups.
- Maintain a dashboard showing aggregate engagement metrics and churn rates by cohort, so you can quickly see if new program changes (e.g. new video tutorials) are moving the needle.

3) Implementing **automated messaging workflows**
   - Risk Threshold: If predicted churn probability > 60%, flag the learner for an “intervention email” or a phone call from a success coach.

10) Appendices :

1) **Cleaned dataset used -**

   https://docs.google.com/spreadsheets/d/1oL2SPlPBUlJpfwsgn0YnglRKmvY4_Fem/edit?usp=drivesdk&ouid=116114578883107403112&rtpof=true&sd=true

2) **Recommendation system python code file -**

   https://drive.google.com/file/d/18qefGIaF-VcneU6yKEhAwDoqJwyJiEa7/view?usp=drivesdk

3) **Student recommendation output -**

   https://drive.google.com/file/d/18uBAQRD5B_tLI_cFzFIU7F559tJrI4Mn/view?usp=drivesdk

4) **Internship Overview Presentation link -**

   https://docs.google.com/presentation/d/1AFYnbaqqUXS1STL0F5zaE_Hc82xa1_I3/edit?usp=drivesdk&ouid=116114578883107403112&rtpof=true&sd=true

5) **Recorded Presentation Video link -**

   https://drive.google.com/file/d/1BeoSHxpnSzUy_nFQLhMpBZkQZlQfXQbv/view?usp=drivesdk