WEEK 3: CHURN ANALYSIS REPORT

TEAM NAME: Team 1

DATE: 03-03-2025

Team members

Team Member Name	**Email ID**
Shahid Khan	shahidk17609@gmail.com
Mangesh Pawar	mangesh5591@gmail.com
Janak Adhikari	janakadhikari777@gmail.com
Rishika Nirala	rishikanirala@gmail.com
Md Aijaz Ahmad	aijaz85ahmad@gmail.com
Harsh Bajpay	work.harshpandat006@gmail.com
Muhammad Sohaib	m21091908@gmail.com

CONTENTS

1. INTRODUCTION

The purpose of this week's report is to emphasize the **essential steps** that go into **comprehending, gathering, and evaluating data** to pinpoint the **primary drivers of student dropout**. This report focuses on a **predictive modeling project** designed to understand, quantify, and predict **customer churn**—the likelihood that a user will stop using or engaging with the platform. **Churn analysis** is critical because **retaining current users** helps maintain a stable user base and is often more cost-effective than acquiring new ones. This report aims to build a **reliable model** to predict user churn, enabling **proactive interventions** to enhance user retention. The **central goal** of this report is to analyze these patterns, engineer relevant features, and develop a **predictive model** to identify which users are at risk of churning. In this case, we're looking at data provided by the **Excelerate**. Key information includes how much time users spend on the platform (**engagement duration**) and how often they are participating on a monthly or weekly basis. We also track how many different opportunities each user has participated in, which reflects their **activity level** and **interests**. Additionally, we calculate the **Days Since Last Engagement** to see how long it's been since a user last interacted with the platform; longer gaps often mean they're more likely to leave. Based on these new metrics, we define a "**churn**" label to identify users who might disengage. A user is marked as churned if they have **low engagement scores**, long periods of inactivity, or have participated in only a few opportunities. These thresholds are based on an initial look at the data and help us identify patterns in user engagement. This report outlines the process of **data preparation**, **feature engineering**, **churn prediction model development**, and **performance evaluation**. The insights gained will support **targeted retention efforts**, guiding the platform's management team in improving **user engagement**, reducing **churn rates**, and fostering **long-term growth**. This marks a step toward a **data-driven approach** to user retention and engagement in the platform.

2. DATA PREPARATION

1. **Date Format Conversion**:
   - Converted the **Opportunity End Date** column into a standard date format to facilitate accurate calculations, such as determining the number of days since the user's last engagement.
   - Checked for any blank or incorrect entries in the column to avoid errors during conversion.
2. **Creation of New Columns**:
   - **Engagement Score**: Combined Engagement Duration and participation in different opportunities to create a single metric representing overall user engagement.
   - **Opportunity Participation Count**: Tracked the number of different opportunities a user engaged with.
   - **Days Since Last Engagement**: Calculated the time since a user's last activity by comparing the current date with the Opportunity End Date.
3. **Defining Churn Criteria**:
   - Marked users as "churned" based on:
     - Engagement Score below a set threshold.
     - Inactivity for a certain number of days.
     - Participation in fewer opportunities than a designated amount.
   - These criteria helped in labeling users likely to disengage and improved the model's predictive accuracy.
4. **Handling Missing Data**:
   - Checked for missing or blank values in the dataset.
   - Decided whether to fill in missing values or remove incomplete rows/columns to maintain data integrity.
5. **Standardization of Features**:
   - Standardized key columns (**Engagement Score**, **Opportunity Participation Count**, and **Days Since Last Engagement**) using **StandardScaler**.
   - Ensured all features were on a similar scale to prevent features with different ranges from distorting the model.
6. **Debugging and Verification**:
   - Printed summaries of key columns to verify data accuracy.
   - Identified any unexpected or extreme values to ensure the data preparation was correct.

3. EXPLORATORY DATA ANALYSIS (EDA)

3.1 **Descriptive Statistics**

1. **Engagement Duration**:
   - **Definition**: Measures how long each user spends interacting with the platform.
   - **Statistical Insights**:
     - **Total number of users** with recorded engagement.
     - **Average engagement duration**: Shows the typical amount of time users spend on the platform.
     - **Standard deviation**: Indicates the variation in engagement levels across users.
     - **Minimum and maximum engagement times**: Highlights the range of user activity, identifying both highly active and less active users.
2. **Opportunity End Date**:
   - **Definition**: Tracks each user's last recorded date of engagement.
   - **Statistical Insights**:
     - **Earliest and latest dates**: Provide insights into the dataset's time frame.
     - Helps identify users who engaged recently versus those with older engagement data.
     - **Contextualizing user behavior**: Crucial for understanding the recency of user interactions.
3. **Engagement Score**:
   - **Definition**: A composite metric combining factors like engagement duration and participation in opportunities.
   - **Statistical Insights**:
     - **Mean**: Represents the typical engagement level across users.
     - **Standard deviation**: Shows variability in user engagement.
     - **Minimum and maximum scores**: Help identify users with extremely low or high engagement, which is essential for targeting retention efforts.
4. **Days Since Last Engagement**:
   - **Definition**: Measures how long it has been since a user last interacted with the platform.

    - **Statistical Insights**:
        - **Mean**: Indicates the average time of inactivity.
        - **Standard deviation**: Shows how much inactivity varies across users.
        - **Minimum and maximum values**: Identify the shortest and longest inactivity periods, which helps in recognizing disengaged users at risk of churn.

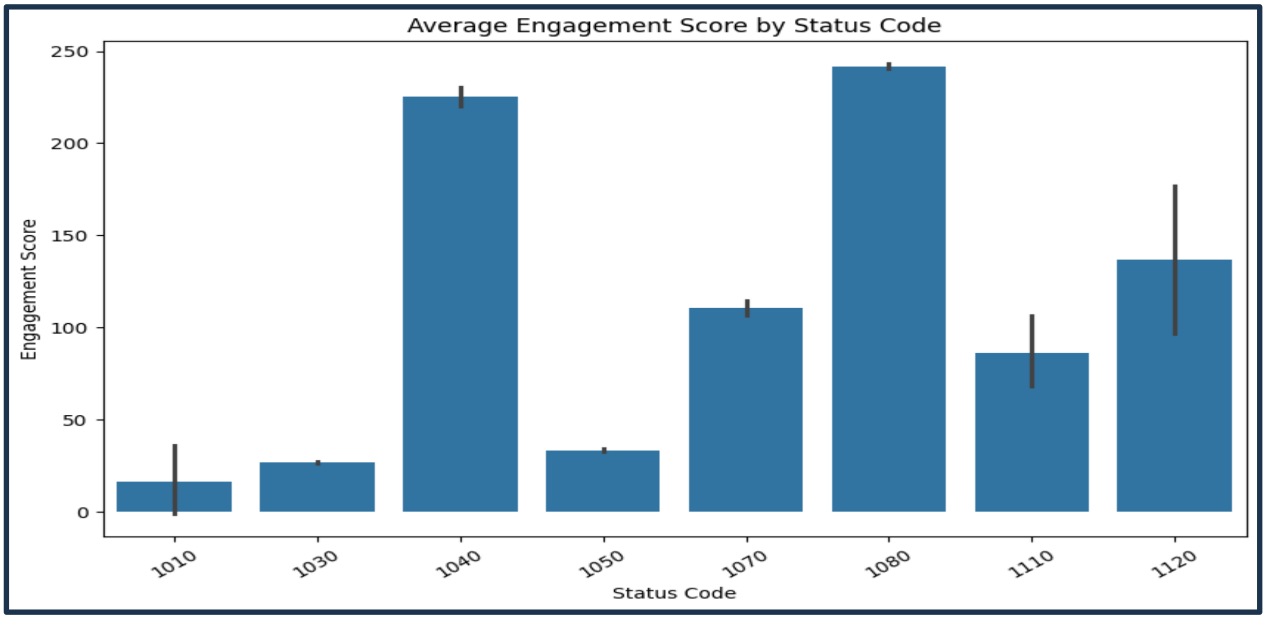
5. **Opportunity Participation Count**:
    - **Definition**: Reflects the number of distinct opportunities a user has engaged with.
    - **Statistical Insights**:
        - **Average participation count**: Indicates the typical number of opportunities a user participates in.
        - **Range of participation**: Highlights both the most and least active users in terms of module engagement.
6. **Churn Count**:
    - **Definition**: Summarizes the total number of churned and non-churned users based on the defined churn thresholds.
    - **Statistical Insights**:
        - Provides an overview of the distribution between users at risk of churn and those still actively engaged.

Summary: These statistical descriptions offer valuable insights into user engagement patterns, which are crucial for building a predictive model to identify high-risk users. By understanding engagement duration, opportunity participation, inactivity periods, and churn status, the platform can better target retention efforts and improve user engagement.

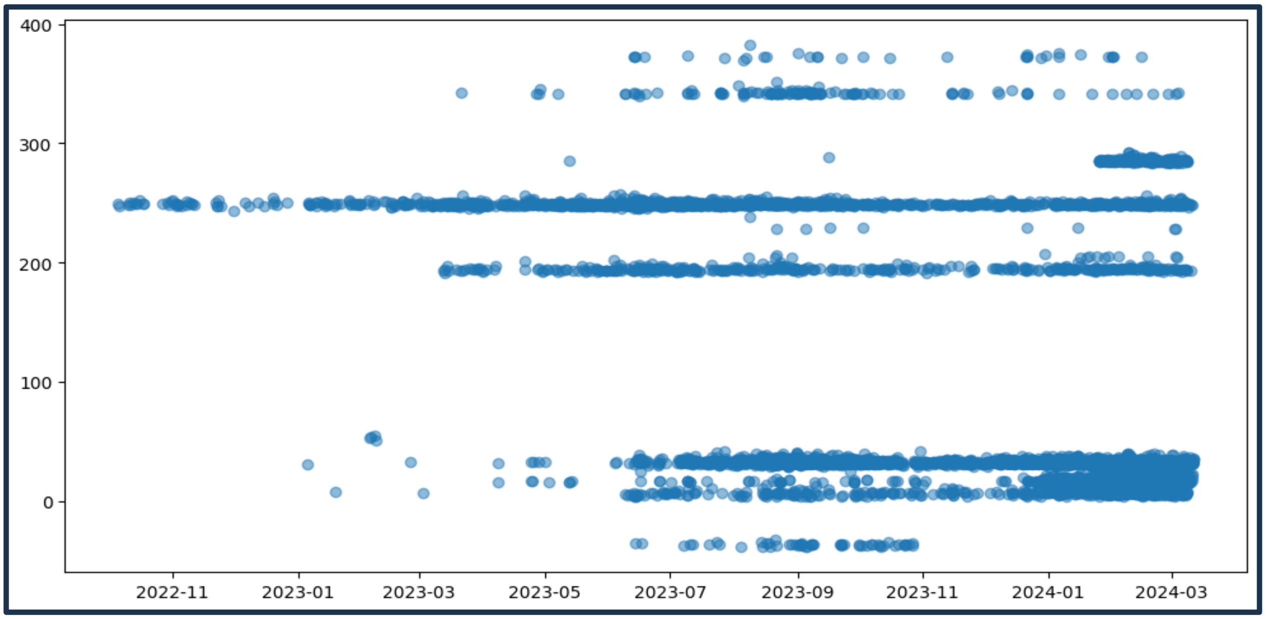
3.2 **Visualization Done**

Charts and graphs showing relationships between features and student drop-offs.

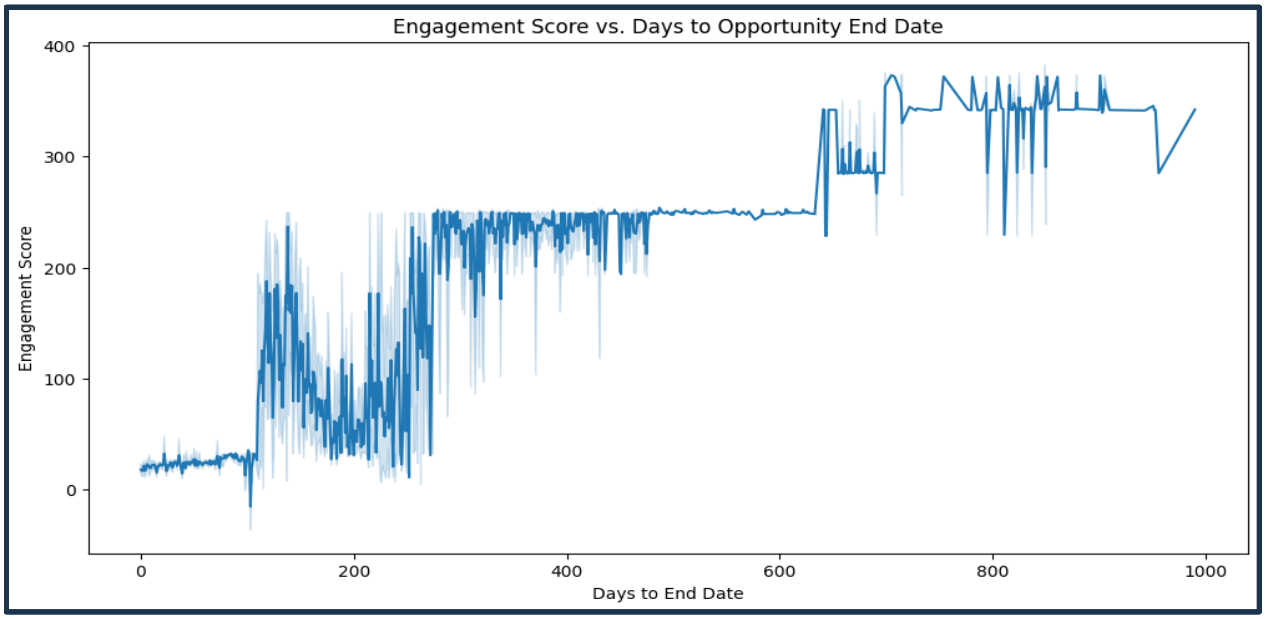
1. Engagement Score by Status Code



2. Engagement Score by Apply Date

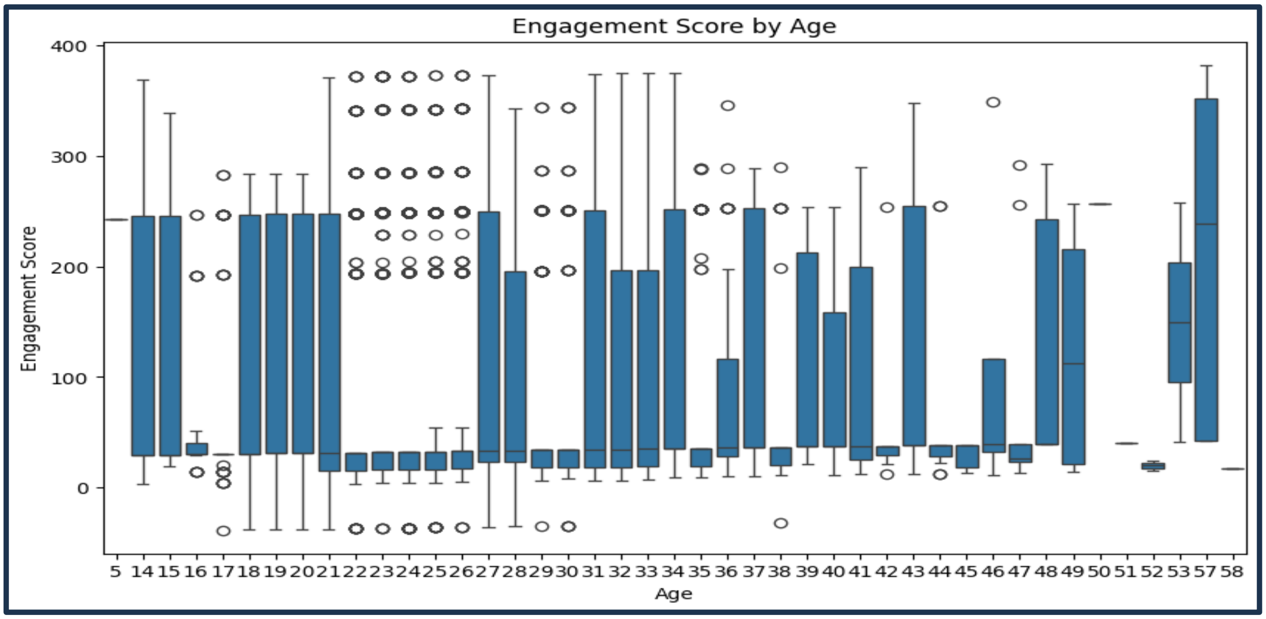


3. Engagement Score by Days to Opportunity End Date

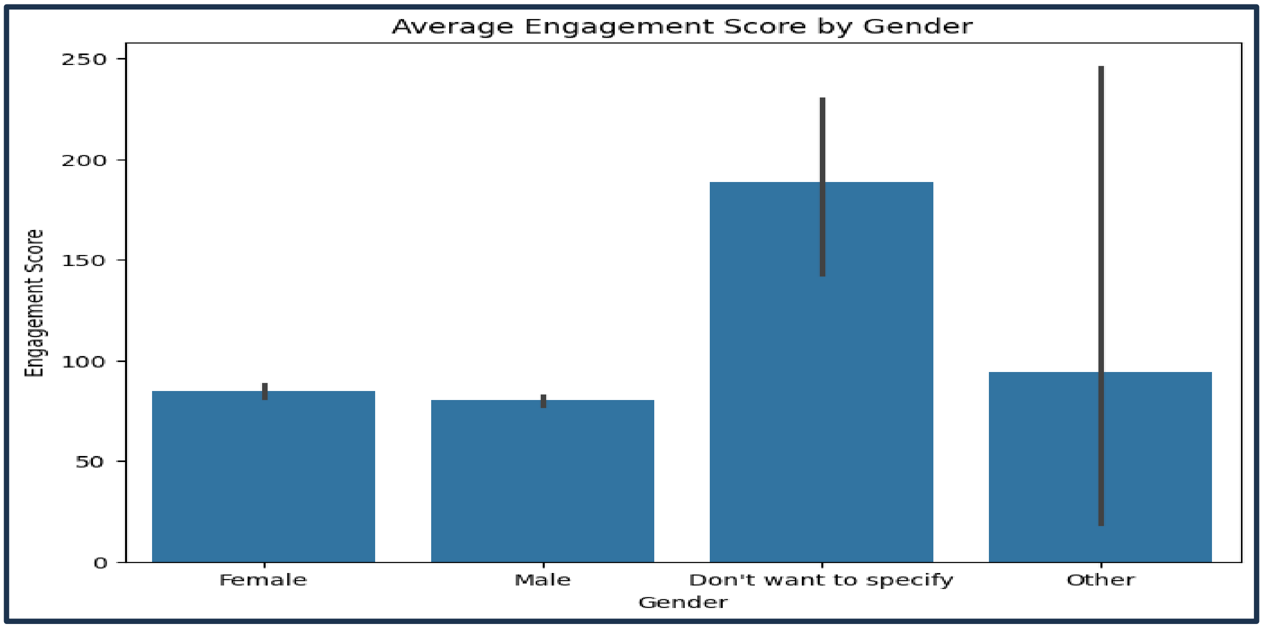


4. Engagement Score by Demographics

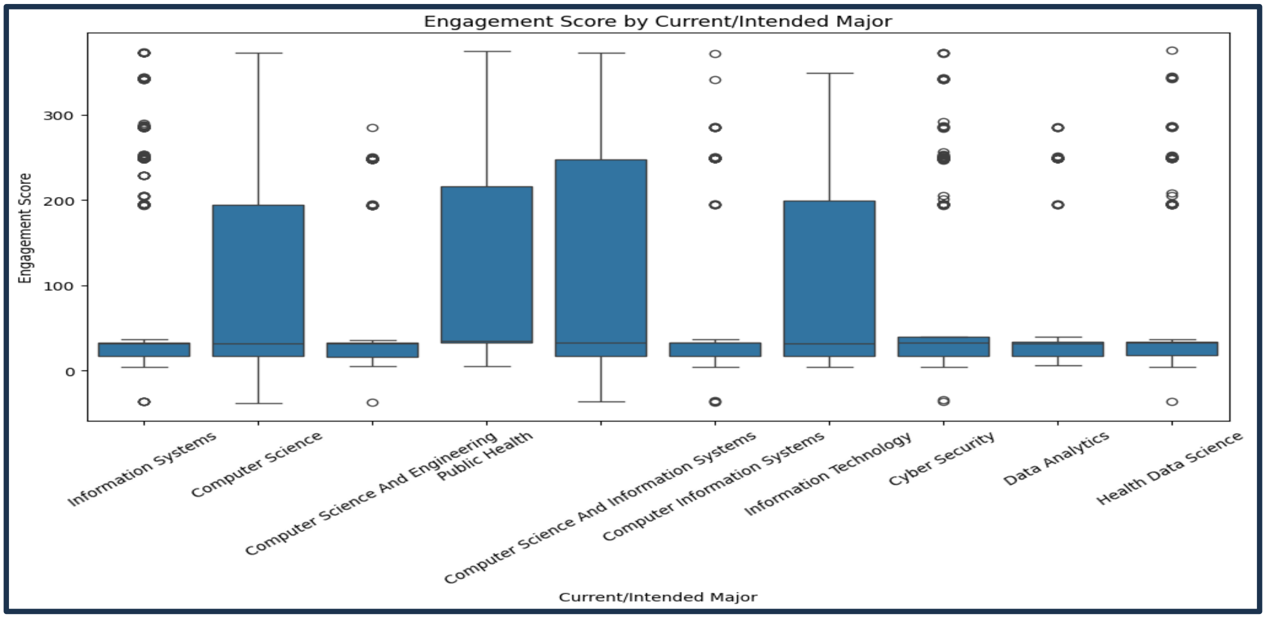
a) **Age vs. Engagement Score** (Box Plot)



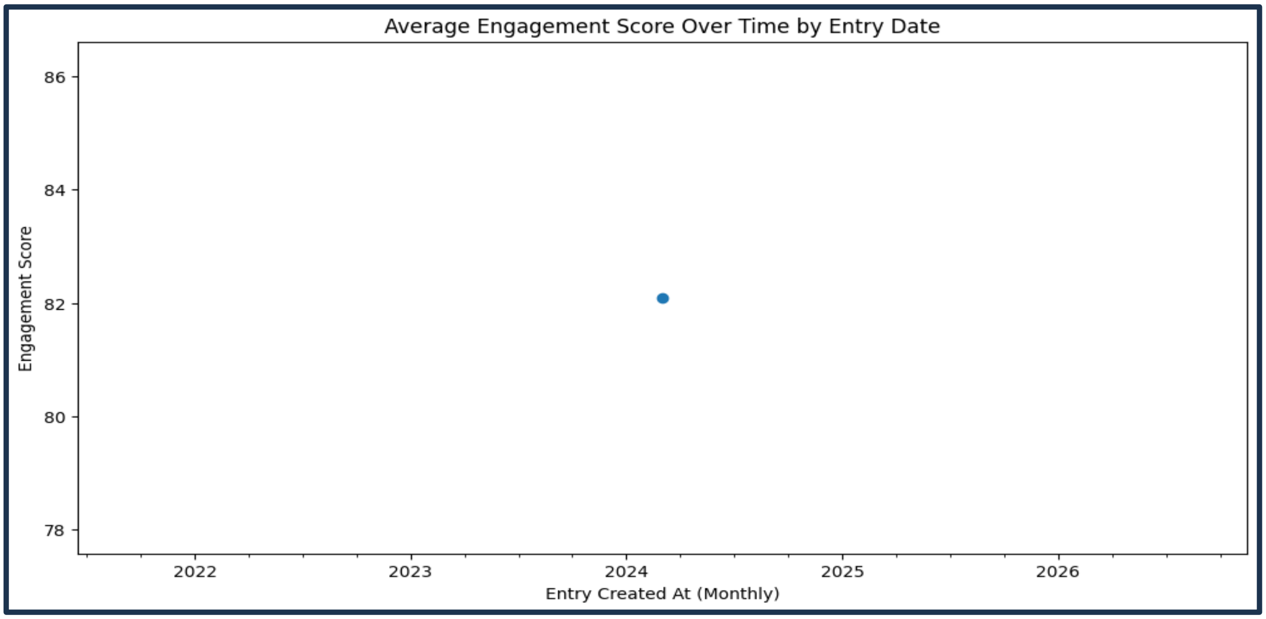
b) **Gender vs. Engagement Score** (Bar Chart)



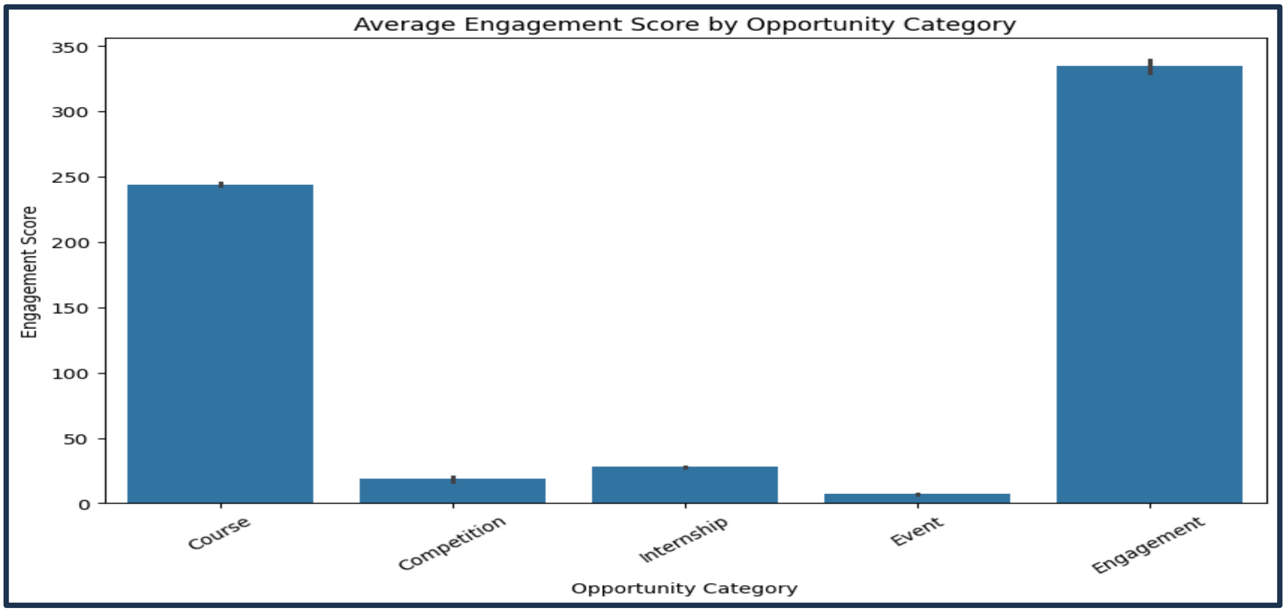
5. Engagement Score by Current/Intended Major



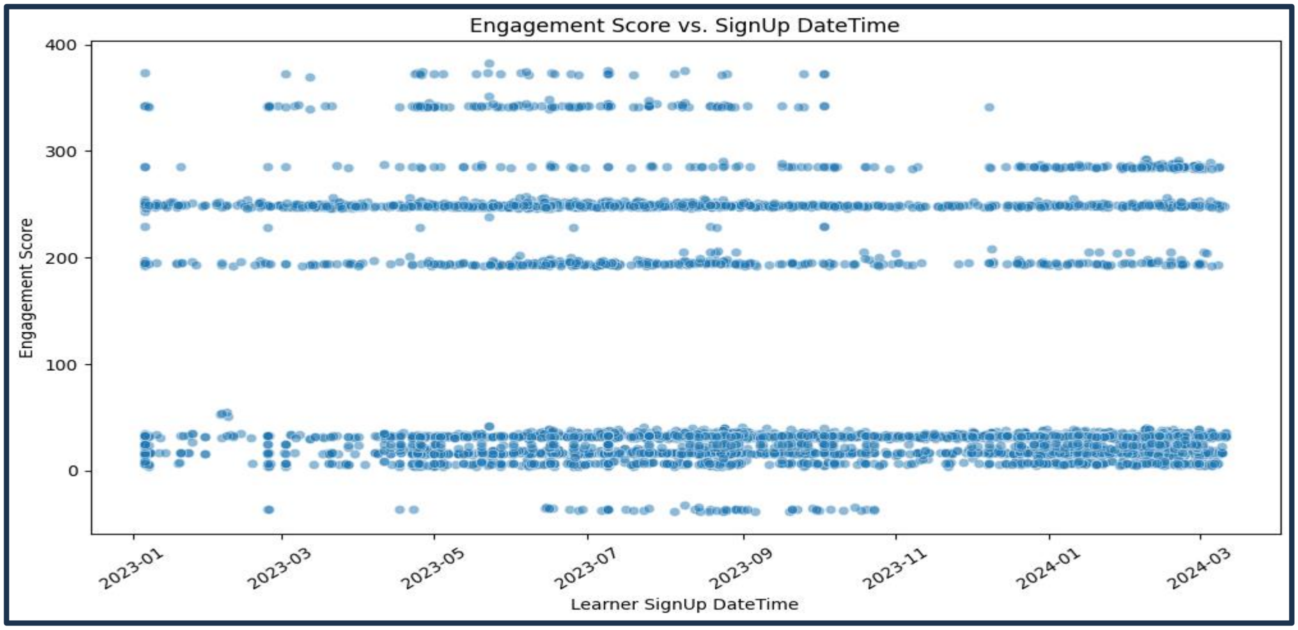
6. Engagement Score Over Time by Entry Date



7. Engagement Score by Opportunity Category



8. Engagement Score by SignUp DateTime



3.3 **Patterns and Trends:**

Several key trends in user engagement offer insights into behavior and retention strategies. Users with **higher Engagement Scores** tend to interact more, with a **skewed distribution** indicating that a small group of highly engaged users drive platform activity. Encouraging moderate users to engage more could improve **retention**. The **Days Since Last Engagement** metric shows that users with higher values are more likely to **churn**, especially if inactivity exceeds a certain threshold (e.g., **60 days**), suggesting the need for **proactive re-engagement strategies**. Additionally, users who engage with more opportunities typically have higher **Engagement Scores**, emphasizing the value of encouraging users to try different content. If some opportunities show **low participation**, it may signal a need for **content optimization** or **better promotions**. Lastly, users with **lower Engagement Scores**, fewer opportunities, and longer inactivity periods are more likely to churn, so monitoring these metrics can help prevent churn and improve retention.

3.4 Data Standardization:

The following things were done in this phase:

- Scaling Age of Learner, Engagement Duration
- Applying One Hot encoding to: 'Opportunity Name', 'Opportunity Category', 'Gender', 'Status Description'
- Applying the hashing technique of hot encoding: first name, last name, institution

name, country and the major

4. CHURN ANALYSIS

4.1 **Key Factors**

Based on the analysis, several primary factors contribute to student drop-offs:

1. **Low Engagement Score**: The engagement score, a composite metric derived from participation in specific opportunities, is a key predictor of student retention. A low score indicates minimal interaction with available resources and activities, suggesting disengagement from the program. This disengagement often precedes drop-off, as students who do not find value in the resources or events offered are more likely to leave.
2. **Low Opportunity Participation**: Students with limited involvement in opportunities, such as workshops or courses, exhibit a higher likelihood of churn. The notebook defines low participation as engagement in fewer than two opportunities. When students do not actively participate in these engagements, it often reflects a lack of connection with the program, reducing their commitment and increasing the chance of departure.
3. **Extended Inactivity (Days Since Last Engagement)**: Long periods of inactivity, defined of 245 days in this analysis, significantly correlate with student drop-offs. When students remain inactive for extended periods, they lose the momentum to stay connected with the learning environment, making re-engagement increasingly challenging. This extended disengagement period strongly indicates a reduced likelihood of their return.

These factors contribute to student drop-offs by capturing varying dimensions of disengagement: a low engagement score reflects overall disinterest, limited participation indicates a lack of integration into the program, and extended inactivity marks a physical and mental disconnection from learning activities. Together, they form a comprehensive picture of students at risk of leaving, enabling targeted interventions to improve retention.

4.2 **Impact Analysis**

These factors—low engagement score, limited opportunity participation, and extended inactivity—each contribute to a heightened likelihood of students leaving by highlighting distinct forms of disengagement.

1. **Low Engagement Score**: This metric reflects how involved and committed a student is with the program. A low engagement score signals a lack of meaningful interaction, which often correlates with diminished interest and motivation. When students don’t feel actively engaged or see value in the activities, they are more inclined to detach, eventually leading to drop-off.

2. **Limited Opportunity Participation**: Participation in workshops, courses, or other opportunities fosters a sense of belonging and progress. Students who engage in fewer than two opportunities may not feel integrated into the program’s community or lack exposure to enriching experiences that build their commitment. Without these touchpoints, students may feel disconnected and are more likely to leave.

3. **Extended Inactivity**: A prolonged period of inactivity, such as exceeding 245 days without engagement, creates both a physical and psychological distance from the program. The longer students remain inactive, the harder it becomes for them to reconnect, especially as they may lose familiarity with the program’s structure and momentum. This disconnection reduces their likelihood of re-engagement, making drop-off a more probable outcome.

   Together, these factors highlight the risk of disengagement, which, if not addressed, significantly increases the likelihood of students leaving the program. Identifying and addressing these issues early allows for targeted interventions to re-engage at-risk students and improve retention.

4.3 **Churn Analysis Code Explanation:**

4.3.1. **Data Preparation and Conversion**

The analysis begins by converting the 'Opportunity End Date' column in the student's DataFrame to a date-time format. This conversion allows for accurate time-based calculations, particularly for determining the time elapsed since the student’s last engagement.

4.3.2. **Creating Engagement Metrics**

To assess student engagement, two key metrics were created:

- **Engagement Score:** This composite score is calculated by weighting the Engagement Duration by a factor of 0.5 and adding participation counts in three significant learning opportunities: Career Essentials, Data Visualization, and Digital Marketing. The Engagement Score provides a holistic measure of a student's interaction with the program.

- **Opportunity Participation Count:** This metric sums the student’s participation in the same three key opportunities, providing a snapshot of their involvement across multiple learning modules.

4.3.3. **Defining Churn Indicators**

Churn was identified using three thresholds based on observed engagement patterns:

- **Engagement Score Threshold:** Students with an Engagement Score below 17.3 are flagged as at risk for churn. This threshold was set based on score distribution, aiming to capture students with low overall engagement.
- **Recent Inactivity:** The time since the last engagement (measured in days from the most recent 'Opportunity End Date') was flagged if it exceeded 245 days. This threshold was established to detect students who have been inactive for a

significant period, suggesting potential disengagement.

- **Low Opportunity Participation Count:** Students with fewer than two participations across key opportunities are flagged as churned, as limited participation indicates disengagement from the program's core learning modules.

How these were identified: The 25% values were used as a threshold for each column to indicate a churn.

- Number of Churned Students: 8203
- Number of Non-Churned Students: 355

4.3.4. **Churn Label Creation**

A binary Churn column was created, where students are labeled as churned (1) if they meet any of the churn criteria above, and not churned (0) otherwise. This label facilitates a clear analysis of the churned versus non-churned population for targeted engagement strategies.

4.3.5. **Feature Scaling**

To prepare the data for further analysis, selected features—Engagement Score, Opportunity Participation Count, and Days Since Last Engagement—were scaled using StandardScaler. This normalization ensures that each feature contributes proportionately to any subsequent analysis or modeling.

4.3.6. **Churn Analysis Summary**

The number of churned and non-churned students was calculated and printed, providing a quick overview of the population's engagement and churn risk distribution.

4.3.7. **Descriptive Statistics for Debugging**

To validate data accuracy and confirm appropriate threshold settings, descriptive statistics were printed for key variables: Engagement Duration, Opportunity End Date, Engagement Score, Days Since Last Engagement, and Opportunity Participation Count. This debugging step helped ensure that thresholds and transformations aligned with the data's distribution and intended analysis.

5. PREDICTIVE MODELING

5.1 Trial Predictive Analysis

In predictive modeling, we first analyzed the code differently and tried to predict the data using the entirety of the dataset, and then we would have predicted churn through that. This was the first approach that we would like to label as **“trial predictive analysis”.** Let us walk through exactly what we did in this phase.

5.1.1 Dimensionality Reduction

In this phase, firstly we did the EDA process, cleaned and transformed the data, and then tried to see the features and the labels, which were way too much, and simply not needed. We then tried to test at what particular number of features the data showed a high amount of variance.

As seen below, we can visualize that the data still shows a very high variance of 90% when it had 200 features. So, we used this particular approach and used dimensionality reduction to reduce the number of features using Principal Component Analysis.

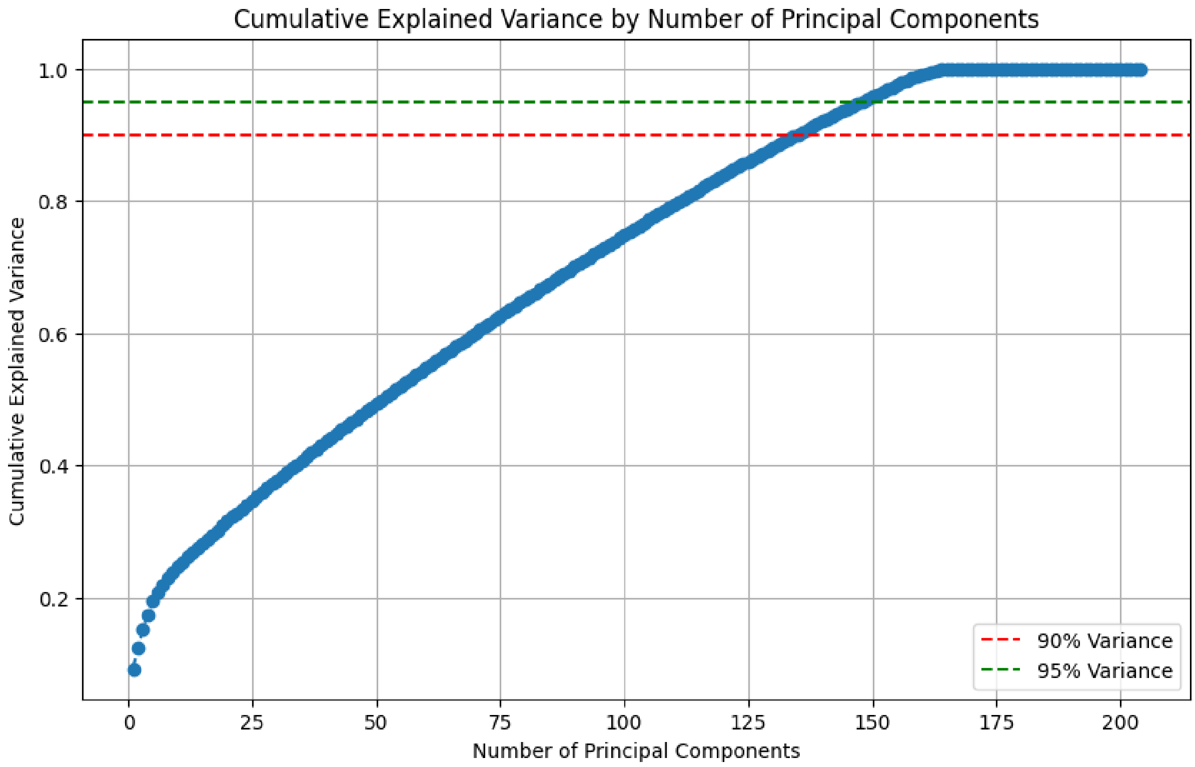
Data Standardization: $X=X-\mu/\sigma$

Covariance Matrix: $C = (1/n-1)X^T.X$

Eigen Values and Eigen Matrices: $Cv=\lambda v$

Transform the data: $Y=X \cdot Vk$

The other features were dropped and then we used only the 200 features as standard.



5.1.2 **Model Training & Evaluation**

We used Logistic Regression, Decision Tree, Random Forest as our model to train and test our data. We achieved the following accuracies:

- Logistic Regression Accuracy: 1.00
- Decision Tree Accuracy: 0.99
- Random Forest Accuracy: 1.00
- Support Vector Machine: 0.91

Then we ensembled these models using Voting Classifier, particularly the soft voting classifier and got a 91% accuracy. However, the models did struggle against a few class instances, that could have indicated the class imbalance, and using this further, we had to do churn analysis. We got the following results in the form of a classification report:

Classification Report:

```
        precision  recall   f1-score   support

  1010       1.00      0.70      0.82        20
  1030       1.00      1.00      1.00       693
  1040       1.00      1.00      1.00        21
  1050       0.99      1.00      1.00       136
  1070       0.99      1.00      1.00       662
  1080       1.00      1.00      1.00       156
  1110       1.00      0.94      0.97        18
  1120       1.00      0.83      0.91         6


accuracy                         1.00      1712
macro avg       1.00      0.93      0.96      1712
weighted avg    1.00      1.00      0.99      1712
```

Confusion Matrix:

```
[[ 14    2      0      0      4      0      0      0]
 [  0   693     0      0      0      0      0      0]
 [  0    0     21      0      0      0      0      0]
 [  0    0      0    136      0      0      0      0]
 [  0    0      0      0    662      0      0      0]
 [  0    0      0      0      0    156      0      0]
 [  0    0      0      1      0      0     17      0]
 [  0    0      0      0      1      0      0      5]]
```

Since we had to do churn analysis further, this would indicate that a few columns would have predicted the churn instead of the entirety of the dataset, and hence, in the actual prediction, we used that approach.

5.2 **Model Selection:**

5.2.1. Logistic Regression

- **Formula**: P(Y=1∣X)=11+e−(β0+β1X1+⋯+βnXn)P(Y = 1 | X) = \frac{1}{1 + e^{-(\beta_0+\beta_1X_1+\cdots + \beta_n X_n)}}P(Y=1∣X)=1+e−(β0+β1X1+⋯+βnXn)1
- **What It Is**: A linear model used for binary classification by estimating the probability of an event.
- **Use for Churn**: Suitable for churn analysis as it provides clear probabilities, helping predict the likelihood of churn based on key features.

5.2.2. Decision Tree

- **Formula**: f(x)=arg maxy Majority Class(X)f(x) = \text{arg max}_y \, \text{Majority Class}(X)f(x)=arg maxyMajority Class(X)
- **What It Is**: A tree-structured model where each node represents a decision based on a feature split.
- **Use for Churn**: Good for churn as it handles non-linear relationships and makes interpretable predictions based on feature importance.

5.2.3. Random Forest

- **Formula**: f(x)=majority vote of {fm(x)}m=1Mf(x) = \text{majority vote of } \{f_m(x)\}_{m=1}^{M}f(x)=majority vote of {fm(x)}m=1M
- **What It Is**: An ensemble of decision trees, combining their outputs for improved accuracy and robustness.
- **Use for Churn**: Random Forest handles complex data interactions and reduces overfitting, making it effective for churn prediction.

5.2.4. Support Vector Machine (SVM)

- **Formula:** f(x)=sign(w·X+b)f(x) = \text{sign}(w \cdot X + b)f(x)=sign(w·X+b)
- **What It Is:** A classification model that finds the optimal hyperplane maximizing the margin between classes.

- **Use for Churn:** Effective for churn prediction with high-dimensional data, especially when a clear separation exists between churned and non-churned customers

5.3 **Model Training**

5.3.1 **Logistic Regression**

Data Preparation

The code prepares the data by setting up feature and target variables from the students_churn DataFrame. The features selected—Engagement Score, Opportunity Participation Count, and Days Since Last Engagement—represent key aspects of student engagement, while the target variable, Churn, indicates whether a student is likely to churn (1) or not (0).

Data Splitting

The dataset is split into training and test sets, with 30% of the data reserved for testing. Stratified splitting is used to ensure that the churn ratio in the training and test sets reflects the overall dataset, maintaining class balance for reliable model performance evaluation.

Hyperparameter Tuning and Pipeline Setup

A pipeline is created with a StandardScaler to normalize feature values and a Logistic Regression model to predict churn. The pipeline is optimized through a grid search with cross-validation to find the best hyperparameters (C values for regularization strength and regularization type). This helps fine-tune model performance by selecting the optimal parameter combination.

Model Evaluation

Predictions are made on the test set, and key evaluation metrics are calculated, including the confusion matrix, classification report (precision, recall, F1-score), and accuracy score. These metrics assess the model's ability to correctly classify churned and non-churned students, indicating its effectiveness in predicting churn.

5.3.2 **Decision Tree**

Splitting Criteria and Structure

The decision tree works by creating binary splits at each node based on a feature value. Each split aims to maximize the separation between churned and non-churned students, to reach a pure classification in the leaf nodes.

The tree’s structure allows it to capture non-linear relationships and interactions among the features, making it well-suited for datasets where factors contribute differently to each class.

Hyperparameter Tuning

Several parameters are adjusted to optimize the model’s performance:

- Max Depth: Controls the maximum depth of the tree to prevent overfitting by limiting its complexity.
- Min Samples Split: Specifies the minimum number of samples required to split a node, which helps in controlling the granularity of the model.
- Min Samples Leaf: Ensures a minimum number of samples in the leaf nodes, further reducing overfitting by smoothing predictions.

Model Interpretation and Churn Prediction

The decision tree provides interpretable rules for churn prediction, as each path from root to leaf in the tree represents a decision rule based on feature values.

This interpretability is particularly useful in churn analysis, as it allows the identification of specific engagement thresholds or behaviors linked to a higher likelihood of churn.

5.3.3 **Random Forest**

Parameter Selection

The Random Forest Classifier has several hyperparameters that can be tuned for better performance:

- n_estimators: The number of trees in the forest, impacts the model's robustness.
- max_depth: Controls the maximum depth of the trees, which can help prevent overfitting.

- min_samples_split: The minimum number of samples required to split an internal node.
- min_samples_leaf: The minimum number of samples required to be at a leaf node, ensuring that the model is not too complex.
- max_features: The number of features to consider when looking for the best split, influencing the diversity among trees.

Hyperparameter Tuning

Randomized Search with Cross-Validation is utilized to efficiently explore the hyperparameter space and identify the optimal settings for the Random Forest model. This method allows for a more comprehensive exploration of parameters compared to Grid Search, leading to potentially better model performance with reduced computational cost.

5.3.4. Support Vector Machines

Feature Selection

The model uses three key features:

- Engagement Score: Measures overall engagement intensity.

- Opportunity Participation Count: Counts the number of key opportunities the student has participated in.

- DaysSince Last Engagement: Captures the recency of the student's last activity.

- Kernel Selection and Hyperparameters.

Hyperparameters: SVM has various kernels (linear, radial basis function (RBF), polynomial) that map data into higher-dimensional spaces to find better separations. The model tests different kernel types, along with:

- C: A regularization parameter that controls the trade-off between achieving a low error on the training data and a large margin.

- Gamma: Controls how much influence a single training example has, affecting the model’s flexibility in capturing patterns.

Hyperparameter Tuning

Grid search with cross-validation is used to identify the optimal values for C, gamma, and kernel type, balancing the model’s complexity and accuracy. This tuning process refines the SVM to achieve the best separation between churned and non-churned students

Cross-Validation

Cross-validation is performed on the best estimator found during grid search to assess the model’s stability and generalizability. The code calculates cross-validation scores over five folds and outputs the mean score, offering insight into how well the model performs across different subsets of data.

5.4 **Performance Metrics**

5.4.1. Confusion Matrix

A confusion matrix is a summary of prediction results on a classification problem. It presents the counts of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) in a tabular format.

For the provided confusion matrix:

Confusion Matrix:

```
[[248   0]  # Predicted: 0

 [  0 5742]]  # Predicted: 1
```

- **True Positives (TP)**: 5742 (correctly predicted churn)
- **True Negatives (TN)**: 248 (correctly predicted no churn)
- **False Positives (FP)**: 0 (incorrectly predicted churn)
- **False Negatives (FN)**: 0 (incorrectly predicted no churn)

5.4.2. Precision

Precision indicates the accuracy of positive predictions. It measures how many of the predicted positive cases were positive.

Precision= 85%

5.4.3. Recall

Recall (also known as sensitivity) measures the ability of a model to find all the relevant cases (i.e., actual positives). It assesses how many actual positive cases were correctly identified.

Recall=100%

5.4.4. F1 Score

The F1 score is the harmonic mean of precision and recall. It provides a balance between the two metrics, especially when the class distribution is imbalanced.

F1 Score= 92%

5.4.5. Overall Interpretation

From the classification report:

- **Accuracy**: An accuracy score of 91% signifies that the model made correct predictions for all instances in the test set.
- **Precision**: A precision of 85% for both classes indicates that every predicted positive (both churn and no-churn) was correct. There were no false positives.
- **Recall**: A recall of 100% for both classes shows that the model identified all actual positives. There were no false negatives.
- **F1 Score**: An F1 score of 92% confirms the model's perfect balance between precision and recall.
- **Support**: The support indicates the number of actual occurrences for each class in the test dataset. There were 248 instances of no churn and 5742 instances of churn.

6. CONCLUSION

Recommendations for Further Analysis

- **Refining Churn Thresholds**: Adjusting the churn thresholds based on continuous feedback and emerging data patterns could help create a more accurate churn prediction model.
- **Investigating Opportunity Participation**: Analyzing specific opportunities, such as **"Data Visualization"** versus **"Digital Marketing"**, may uncover trends that correlate with churn and can guide curriculum improvements tailored to enhance engagement.
- **Seasonal Monitoring**: Regularly tracking churn over time could reveal seasonal patterns, which would help develop targeted engagement strategies aligned with these trends.

Conclusion

The analysis highlights that **low engagement** and **extended inactivity** are significant indicators of churn. The visual distributions of engagement scores and days since last engagement further corroborate these findings. The churn statistics offer a comprehensive overview of both engaged and churned students, providing useful insights for potential retention strategies.

Future Work

- **Additional Predictors**: Experimenting with additional factors such as **demographic details** or **academic background** could enhance the model’s robustness and predictive power, which could be explored in future analyses.