Title - Real-Time Anomaly Detection and Forecasting in Streaming Platform Engagement using Time Series

Github - https://github.com/Baddala-Govardhan/CSCI485_Spring25_Govardhan_Baddala/tree/main/Finial%20Project

Full overview of the project:

This project focuses on building a dynamic, real-time dashboard to monitor and forecast user engagement for a streaming platform using historical data from netflix_users.csv. The dataset contains 25,000 records with 8 key variables including age, country, subscription type, watch hours, favorite genre, and last login. We engineered additional features like age groups, churn rate, and login counts. The dashboard is not static — it simulates daily updates, and with every refresh, new calculations are performed for forecasting, churn, and anomalies. Using this base data, we forecasted 14 future days of user logins using both ARIMA (for trend-based patterns) and LSTM (for complex, non-linear behavior). The dashboard displays not only historical trends but also real-time predictions and anomalies in user activity.

The LSTM model used in this project was tuned with the following parameters: WINDOW_SIZE = 14, hidden_size = 25, epochs = 25, and learning_rate = 0.01. Forecasted values from both ARIMA and LSTM models are clearly visualized in the dashboard. We also implemented Isolation Forest to detect anomalies in daily login counts. In short, this dashboard uses pre-defined historical data as a base but extends it with predictive modeling and daily simulation, offering a fully dynamic and insightful tool for streaming analytics and decision-making.

Step 1: Loading and Preparing the Data

- First, we load the dataset from a file called netflix_users.csv using pandas.
- It contains 25,000 users, with information like:
    - Name, Age, Country
    - Subscription type
    - How many hours they watched
    - Favorite genre
    - Last time they logged in

- Then, we convert the Last_Login column into date format, so we can work with it as time series data.
- Finally, we check the structure of the dataset using df.info():
    - There are no missing values
    - All columns have the correct data types
    - Most importantly, Last_Login is now ready for time-based analysis

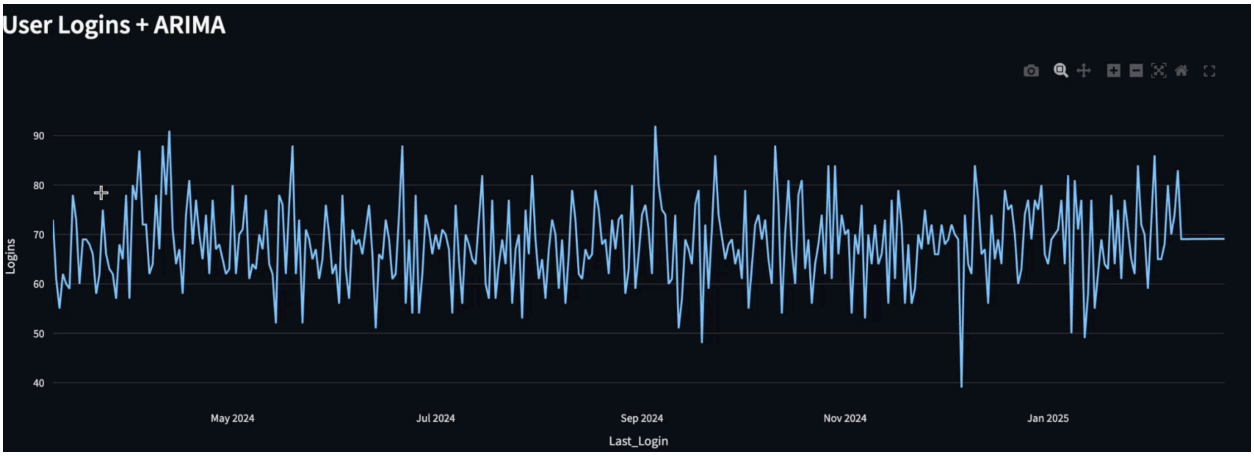
Step 2: Feature Engineering and Churn Simulation

After loading the original dataset, the next step was to create new features that are essential for analysis, forecasting, and visualization.

We started by grouping users into age categories (e.g., <18, 18–24, 25–34, etc.) using their Age values. This helped us analyze engagement trends across age segments. Then, since the dataset didn't include real churn data, we simulated churn rates based on how many days had passed since the user's last login. The idea was: the longer a user has been inactive, the more likely they are to churn. We added some random variation (noise) and seasonal patterns (using a sine function) to make the churn rate more realistic. We also added a churn_flag column, which marks users as high risk if their churn rate goes above 15%.

This step was important because it gave us a target variable (churn) to analyze and forecast, and provided grouped features like Age_Group that made the dashboard more detailed and insightful.

Step 3: ARIMA Forecast Plot – User Logins



This plot shows how user logins over time are forecasted using the ARIMA model (AutoRegressive Integrated Moving Average).

Plot Shows:

X-axis: Time (from March 2024 to early 2025) based on Last_Login

Y-axis: Number of user logins each day

The blue line represents the actual login counts, including past data and predicted values (towards the end from march)

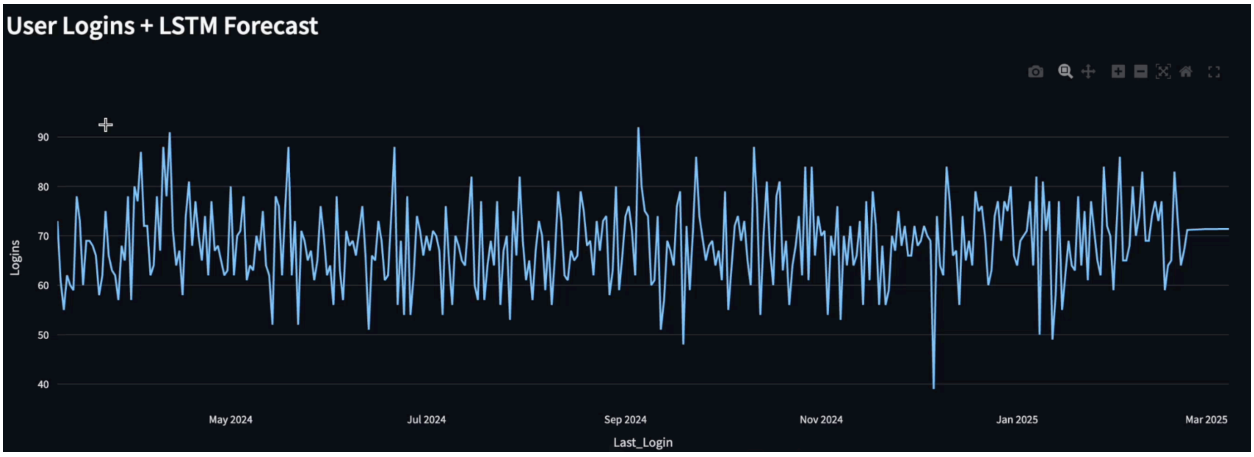
Interpretation:

- You can see regular ups and downs, showing natural daily variations in login activity.
- Around early 2025, the line becomes slightly flatter this is where ARIMA starts forecasting.
- The model tries to predict the next 14 days of login activity based on historical trends

Limitation:

- While ARIMA captures the general pattern, it doesn't handle sudden spikes or drops very well.
- That's why the end of the curve may appear smoother or less reactive than the real data.

Step 4: LSTM Forecast Plot – User Logins



This plot shows the LSTM model's forecast for user login behavior over time, based on past activity.

Plot Shows:

- X-axis: Dates from March 2024 to March 2025 (Last_Login)
- Y-axis: Number of logins per day
- The blue line includes both actual user logins and predicted values at the end (from the LSTM model

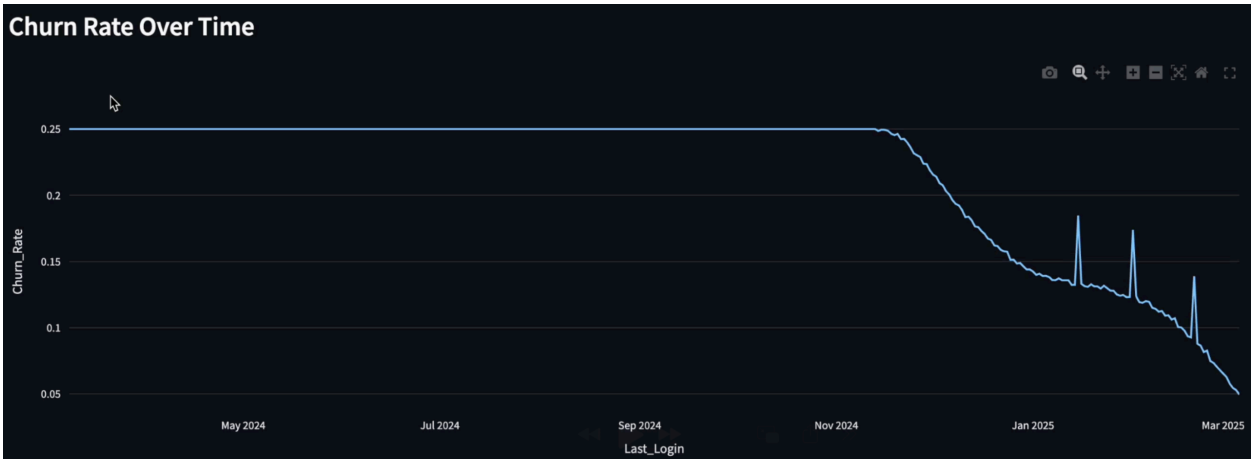
Interpretation:

- The pattern is more flexible and reactive than ARIMA
- It captures small ups and downs more smoothly showing that the model can follow non-linear patterns
- The end of the line (flat portion) is the LSTM's 14-day forecast

LSTM Works Here:

- LSTM stands for Long Short-Term Memory a type of deep learning model that learns from sequences
- It's trained on historical login data and uses a rolling window of previous days (past 14 days) to predict future values
- Unlike ARIMA, it doesn't assume stationarity and can handle seasonality and noise

Step 5: Churn Rate Over Time



This chart shows how user churn rate changed over time in the dataset, simulating real user behavior on a streaming platform.

Plot Shows:

- X-axis: Time (Last_Login) from May 2024 to March 2025
- Y-axis: Churn rate values (from 0.05 to 0.25)
- Line: Represents the average churn rate of users on each day

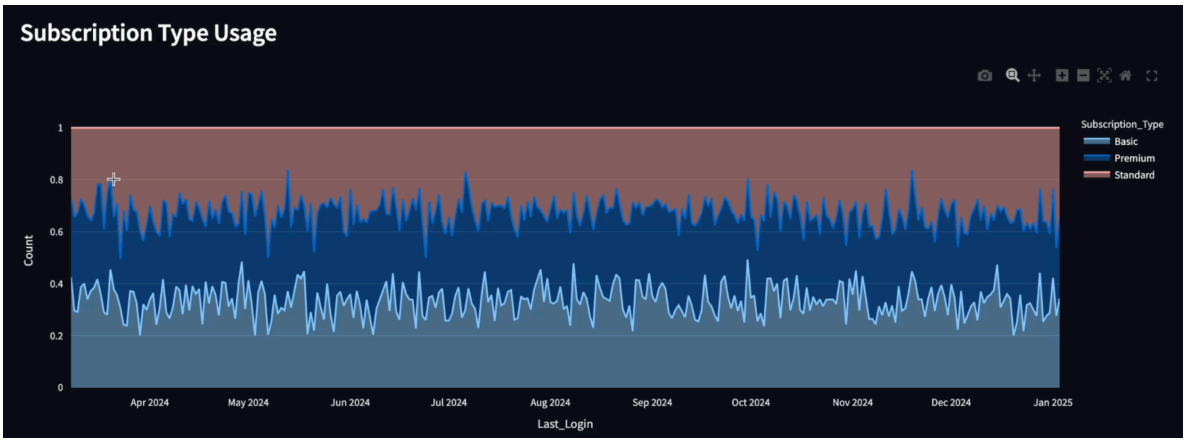
Interpretation:

- In the beginning (May to November 2024), churn rate was constant and high at 0.25
- Starting around December 2024, churn rate began to gradually decline
- There are a few sudden spikes, which were intentionally added to simulate random anomalies
- By March 2025, churn rate dropped below 0.1, indicating stronger retention

Why This Happens:

- Churn rate is simulated based on how long it's been since a user logged in
- Over time, as the dashboard moves day by day, fewer users are at risk of churning (recently active users dominate the view)
- The random spikes reflect artificial disruptions like service outages or bad user experience moments

Step 6: Subscription Type Usage



This area chart shows how users with different subscription plans (Basic, Premium, and Standard) are distributed over time based on their last login date.

Plot Shows:

- X-axis: Time (Last_Login) — from March 2024 to early 2025
- Y-axis: Fraction of users (normalized count) logging in on each day
- The stacked colored areas represent:
    - Light Blue: Basic Plan
    - Dark Blue: Premium Plan
    - Pink: Standard Plan

Interpretation:

- Premium users (dark blue) seem to have the most consistent engagement throughout the timeline.
- Standard plan users (pink) make up the largest share overall, especially in early 2024.
- Basic plan usage is the lowest, but still fluctuates consistently.

Why This is Useful:

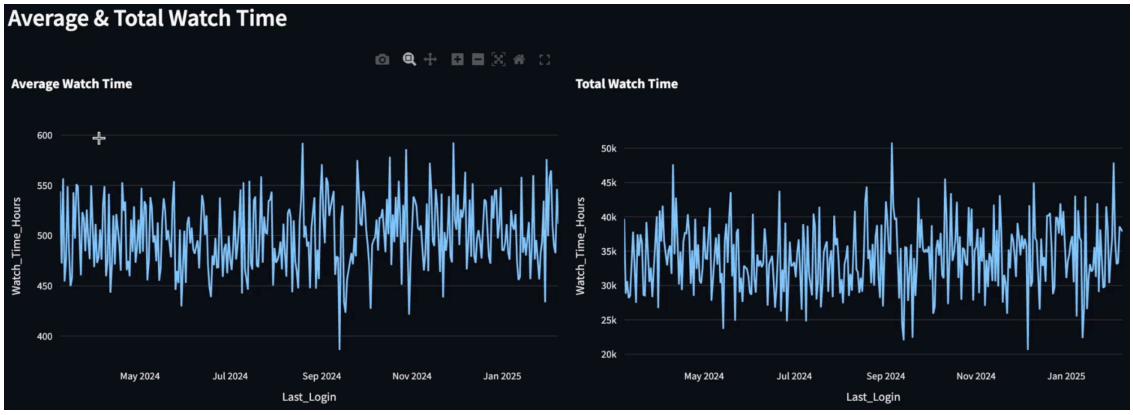
- This helps understand which subscription tiers are most active.
- The business team can use this to:
    - Target underperforming segments with offers
    - Evaluate if premium users show better retention
- Combined with churn plots, it helps see which plan users are leaving or staying more.

Key Takeaways:

- Premium users are more steady and loyal
- Standard plan dominates early usage but fluctuates
- Supports decisions on pricing, feature rollouts, or loyalty rewards

Step 7: Average & Total Watch Time

This dual-panel visualization provides insights into how much content users are watching on the streaming platform both on a per-user basis and across the whole platform.



Charts Show:

Left Plot: Average Watch Time

- Y-axis: Average hours watched per user per day
- X-axis: Last_Login (March 2024 to March 2025)
- The plot shows daily fluctuations around the 500–550 hour range.

Right Plot: Total Watch Time

- Y-axis: Total watch hours from all users combined each day
- Shows daily totals ranging from 20k to nearly 50k hours
- Spikes represent heavy usage days; drops may indicate low user activity or anomalies

Interpretation:

- Average watch time remains fairly stable over the year, showing consistent user engagement
- Total watch time is more volatile because it depends on how many users log in on that day
- Sudden peaks or dips in total time could relate to holidays, content drops, or outages

Why This Insight Matters:

- Tells us whether users are still watching a lot per session (Avg) even if fewer are logging in
- Helps correlate with churn or anomalies (e.g., drop in total watch but avg stays high = fewer but loyal users)
- Useful for content planning, peak traffic prediction, and system scaling

Step 8: Genre Popularity Over Time

This stacked area chart visualizes how popular each content genre was among users, based on their Favorite_Genre, over time.

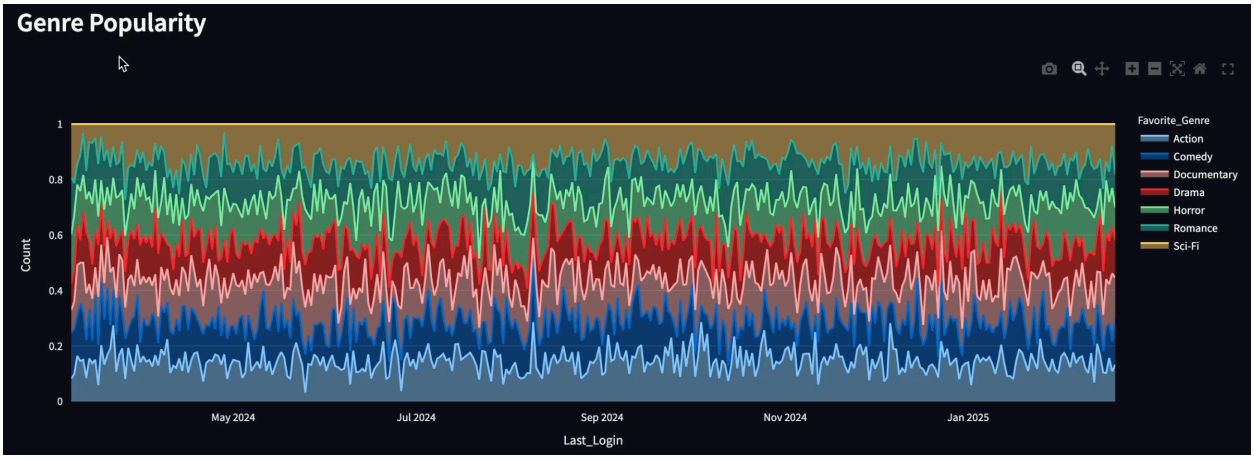


Chart Shows:

- X-axis: Time (Last_Login) from early 2024 to March 2025
- Y-axis: Proportional count of users (normalized to 1 per day)
- Each colored area represents a genre:
    - Action (bottom)
    - Comedy
    - Documentary
    - Drama
    - Horror
    - Romance
    - Sci-Fi (top)

Interpretation:

- Romance, Sci-Fi, and Horror seem to have consistently high popularity, occupying larger areas at the top
- Comedy and Action have more modest but steady presence
- Some genre shifts can be seen over time slight dips and rises may reflect content releases or user trends

Insight:

- This helps streaming platforms understand what content keeps users engaged
- If a high churn period coincides with a drop in a popular genre, content planning teams can respond accordingly
- It also shows stable genre preferences, useful for recommendation algorithms

Step 9: Age Group Distribution

This bar chart shows how users are spread across different age groups in the dataset.

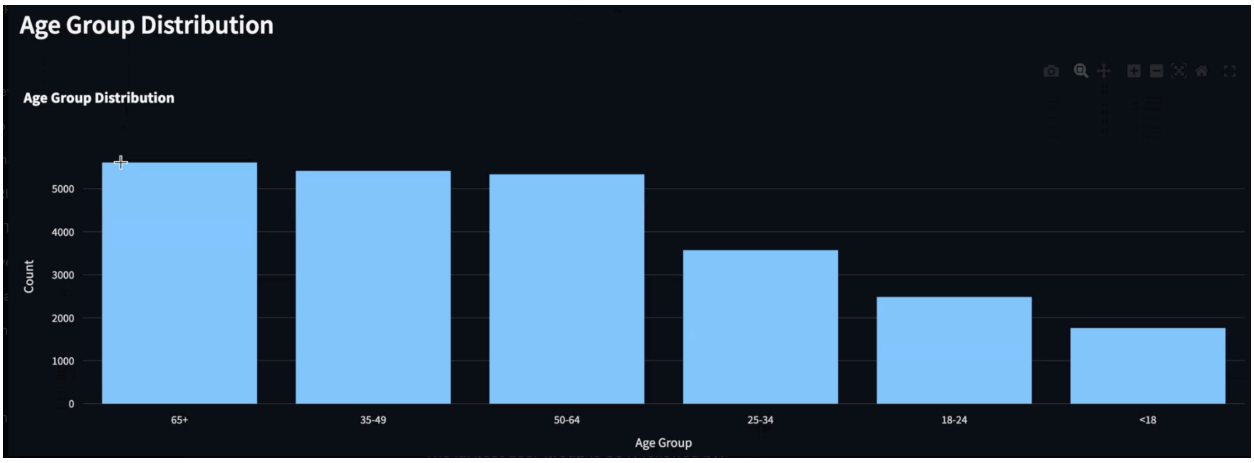


Chart Shows:

- X-axis: Age groups:
  <18, 18–24, 25–34, 35–49, 50–64, and 65+
- Y-axis: Number of users in each group
- Bars represent how many users from each age range have logged in over time

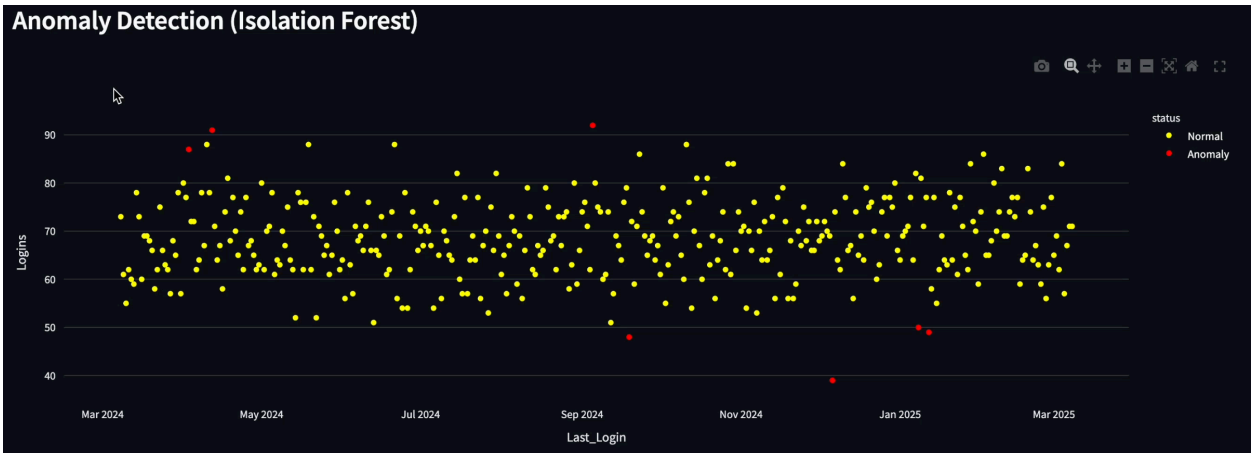
Interpretation:

- The largest user group is 65+, followed by:
  - 35–49
  - 50–64
- The youngest groups (<18 and 18–24) have the lowest counts

Insights:

- Helps understand which age segments dominate platform usage
- Important for:
    - Targeted marketing (e.g., offer discounts for seniors or youth)
    - Content planning (genres preferred by each age group)
- Can be used alongside churn metrics to see which age groups are at higher risk

Step 10: Anomaly Detection Using Isolation Forest



This scatter plot shows which days had unusual login activity, detected using the Isolation Forest algorithm.

Chart Shows:

- X-axis: Time (Last_Login) from March 2024 to March 2025
- Y-axis: Number of user logins each day
- Yellow Dots: Normal login activity
- Red Dots: Detected anomalies (either unusually high or low logins)

Interpretation:

- Most days have login counts between 60–80, and are marked as normal
- Red points represent outliers days with very low or very high login counts
- Anomalies are spread throughout the year, with a cluster visible around early 2025

Why Isolation Forest?

- It’s an unsupervised machine learning algorithm
- It works by isolating data points that behave differently from the rest
- Great for time series where we don’t have labeled anomalies

Insight Is Valuable:

- Helps detect unusual user behavior early
    - Could be caused by platform bugs, content drops, or external events
- Supports business decisions like:
    - Platform performance
    - Triggering alerts for sudden drops
    - Understanding peak load days for scaling