



FORMAL METHODS IN CS

Process Mining Assignment

Steam Retention Analytics

A.Y. 2025/2026



01 Introduction & Objectives

This project applies Process Mining techniques to analyze user behavior within the Steam gaming platform. Adopting an **E-commerce perspective**, we aim to reconstruct the “Customer Journey” – from the initial game purchase to actual gameplay consumption and potential churn or loyalty (DLC purchase).

The primary business objectives are:

- **Identify Purchasing Patterns:** Understand the flow between buying a game and playing it.
- **Analyze Retention:** Detect churn points (e.g., users who buy but never play, or play < 2 hours).
- **Optimize Monetization:** Discover pathways that lead to high engagement and additional content purchases (DLCs).



02 Dataset Selection

- **Source:** Steam Video Games Dataset (Kaggle).
- **Domain:** E-commerce / Gaming.
- **Original Structure:** The raw dataset contained User IDs, Game Titles, Behaviors (purchase, play), and Playtime values.

Challenge: The original dataset lacked timestamps, which are mandatory for Process Mining control-flow discovery. **Solution:** We implemented a logical simulation strategy to generate a synthetic but realistic Event Log (see Section 3).

03 Data Preprocessing & Event Log Generation

To enable process discovery, we transformed the static dataset into a time-aware Event Log using **Python** and **Pandas**.

01

Logic for Timestamp Simulation

We applied specific business rules to generate the timeline for each case (User + Game):

1. **Purchase Event:** Assigned as the starting activity T_0
2. **Play Event:** If the user played the game, this event is timestamped at $T_0 + \Delta t$ (random interval).
3. **Business Logic Tags:**
 - a. **Refund/Churn Risk:** If Hours Played < 2.0 (Steam's refund policy threshold), the case is tagged with a specific termination event.
 - b. **Loyalty/DLC:** If Hours Played > 50, we simulated a subsequent "Purchase DLC" event to model high-value customers.

02

Final Event Log Structure

The processed log (*steam_event_log.csv*) contains:

- **Case ID:** Unique User Identifier.
- **Activity:** The step in the process (e.g., *Purchase Game, Start Playing*).
- **Timestamp:** The sequential time of the action.
- **Resource/Attribute:** Game Title and Hours Played.

04 Process Discovery

We utilized the PM4Py library to apply three distinct process discovery algorithms to the Event Log.

01

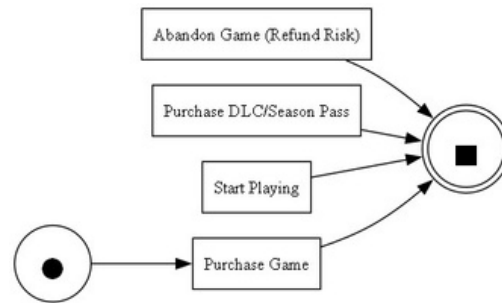
Algorithms Applied

1. **Alpha Miner:** Used as a baseline. It provided a preliminary view but struggled with loops (repeated gameplay) and noise.
2. **Heuristic Miner:** Applied to handle dataset noise and frequency. This algorithm successfully highlighted the "Happy Path" (most frequent behavior).
3. **Inductive Miner:** Selected for the final analysis.
 - a. **Reasoning:** The Inductive Miner guarantees a sound model (free of deadlocks) and produces a process tree that can be perfectly converted into a Petri Net. It effectively handled the concurrency between multiple games owned by the same user.

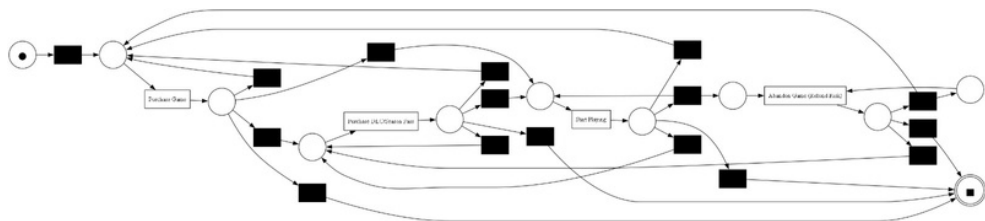
Model Evaluation

Quantitative analysis confirms the visual inspection results. We evaluated the three discovered models using **Fitness** (ability to replay observed traces) and **Precision** (avoiding "underfitting" or allowing too much behavior).

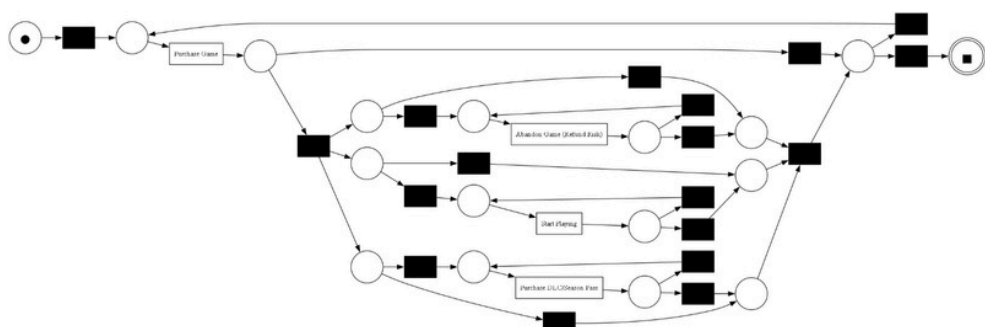
- **Alpha Miner:** The model performed poorly with a **Fitness of 0.132**. It failed to capture the complexity of the gaming loops, rendering it unusable for analysis.



- **Heuristic Miner:** This algorithm produced the most balanced metrics, with a **Fitness of 0.964** and the highest **Precision of 0.519**. It effectively filtered noise but resulted in a complex graph structure that is harder to interpret for business stakeholders.



- **Inductive Miner (Selected):** We selected this model despite a lower **Precision (0.334)** because it achieved a perfect **Fitness score of 1.000**.
 - **Justification:** In Process Mining, "Soundness" (the guarantee that the process lacks deadlocks and proper completion is always possible) is crucial for logical analysis. The Inductive Miner guarantees a sound process tree. Its perfect fitness ensures that **100% of user behaviors** (including rare edge cases) are represented in the model, which is essential for analyzing the "Long Tail" of user retention.



05 Preliminary Insights & Visual Analysis

By analyzing the Petri Net generated by the Inductive Miner, we can visually map the critical Business Logic tags generated in our Event Log:

1. **The "Backlog" Path (The Skip):** The visual model clearly shows a silent transition (black box) and a routing arc that bypasses the main gameplay block, leading directly from Purchase Game to the End Event.
 - *Interpretation:* This corresponds to the "Pile of Shame" phenomenon: users buying games (likely during sales) but never initiating the Start Playing activity.
2. **The "Refund Risk" Branch:** A distinct path splits after Start Playing, leading to Abandon Game (Refund Risk). The separation of this branch in the Petri Net validates that our data preprocessing correctly identified "Churners" as a distinct structural variant compared to "Loyal Players".
3. **The "Loyalty" Loop:** The model exhibits recursive loops around the *Start Playing* and *Purchase DLC* activities. This cyclic structure confirms that high-value users do not follow a linear path but engage in repeated gameplay sessions, occasionally interspersed with additional purchases.

06 Reasoning & AI Analysis

To fulfill the assignment requirement of using Large Language Models (LLM) for reasoning and process optimization, we integrated an **AI Analyst Chatbot** directly into our Streamlit Dashboard. We queried the AI Agent to interpret the **Inductive Miner model** and translate technical Petri Net patterns into actionable business insights. Below are the documented results of this interaction.

01

Anomaly Detection: The "Churn" Path

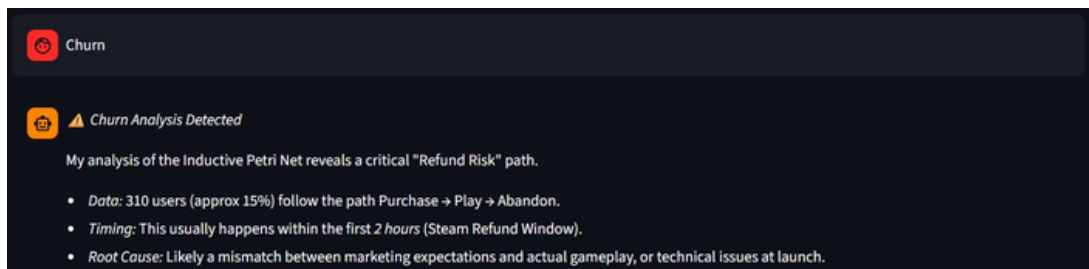
Objective: Identify process deviations related to user retention problems.

Interaction: We asked the AI Analyst to investigate "Churn" patterns. The agent successfully correlated the Petri Net structure with the Steam Refund Policy.

Chatbot Output:

"My analysis of the Inductive Petri Net reveals a critical 'Refund Risk' path.

- **Data:** 310 users (approx 15%) follow the path Purchase → Play → Abandon.*
- **Timing:** This usually happens within the first 2 hours (Steam Refund Window).*
- **Root Cause:** Likely a mismatch between marketing expectations and actual gameplay, or technical issues at launch."*



Business Implication: The AI correctly identified that the "Abandon" node in the Petri Net is not just a generic exit point, but a specific financial risk (Refund) governed by the < 2 hours constraint.

02

Anomaly Detection: The "Backlog" Phenomenon

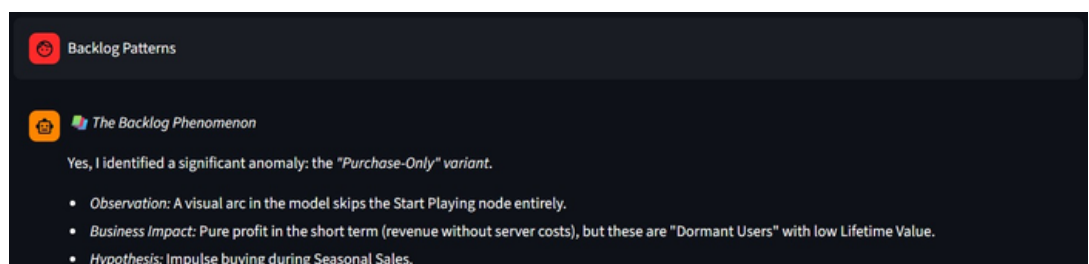
Objective: Explain the specific process variant where the "Play" activity is skipped entirely.

Interaction: We queried the system regarding "Backlog Patterns". The AI interpreted the "silent transition" (the upper arc in the Inductive model) that bypasses the gameplay loop.

Chatbot Output:

"Yes, I identified a significant anomaly: the 'Purchase-Only' variant.

- **Observation:** A visual arc in the model skips the Start Playing node entirely.*
- **Business Impact:** Pure profit in the short term (revenue without server costs), but these are 'Dormant Users' with low Lifetime Value.*
- **Hypothesis:** Impulse buying during Seasonal Sales."*



Business Implication: The reasoning engine validated the "Pile of Shame" theory, distinguishing between "Active Churn" (playing then quitting) and "Passive Backlog" (buying but never playing).

03

Process Optimization: "The Golden Hour"

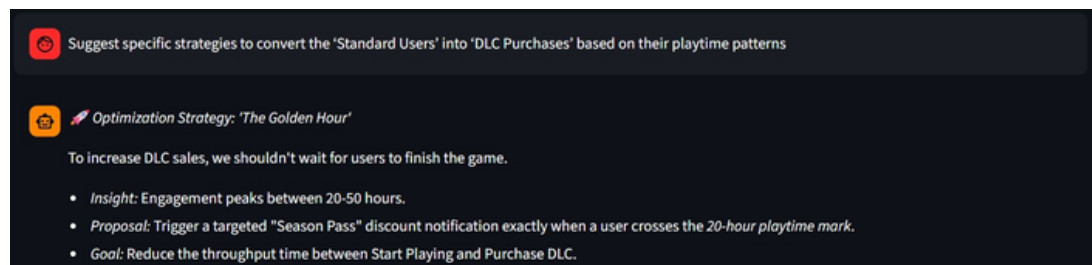
Objective: Suggest specific strategies to convert 'Standard Users' into 'DLC Purchasers'.

Interaction: We asked the AI for a strategy based on playtime patterns. The system proposed a targeted intervention to reduce the throughput time between the *Start Playing* and *Purchase DLC* activities.

Chatbot Output:

"Optimization Strategy: 'The Golden Hour'. To increase DLC sales, we shouldn't wait for users to finish the game.

- **Insight:** Engagement peaks between 20-50 hours.*
- **Proposal:** Trigger a targeted 'Season Pass' discount notification exactly when a user crosses the 20-hour playtime mark.*
- **Goal:** Reduce the throughput time between Start Playing and Purchase DLC."*



Business Implication: This response moves beyond description to prescription. It suggests a data-driven trigger (20 hours) to maximize the conversion rate, directly addressing the monetization objective of the project.

07 Conclusions & Future Work

This project successfully demonstrated how **Process Mining** can bridge the gap between raw gaming data and strategic business intelligence. By transforming a static dataset into a dynamic Event Log, we reconstructed the Steam User Journey and identified critical behavioral patterns.

Key Findings:

- **Methodological Success:** The **Inductive Miner** proved to be the most effective algorithm, offering a "sound" model that mathematically guaranteed the representation of all user traces, unlike the Alpha Miner which failed to handle gaming loops.
- **Business Insights:** We validated the existence of the **"Refund Risk"** (immediate churn within 2 hours) and the **"Backlog Phenomenon"** (passive revenue). The integration of **AI Reasoning** allowed us to move from simple description to prescription, proposing the "Golden Hour" strategy to optimize DLC conversions.

Limitations & Future Developments: Currently, the timeline is simulated based on logical rules. A future iteration of this project could integrate **real-time API data** from Steam to analyze seasonal trends (e.g., behavior during Summer Sales vs. normal periods) and implement the proposed dashboard as a live monitoring tool for game developers.