Play Next in Steam

Game Recommendation System

STEAM®

Erdos Data Science Bootcamp Project
Spring 2023

Team Gamers' Gaming Habits

Maria Tiongco, Joshua Schroeder, Aycan Katitas

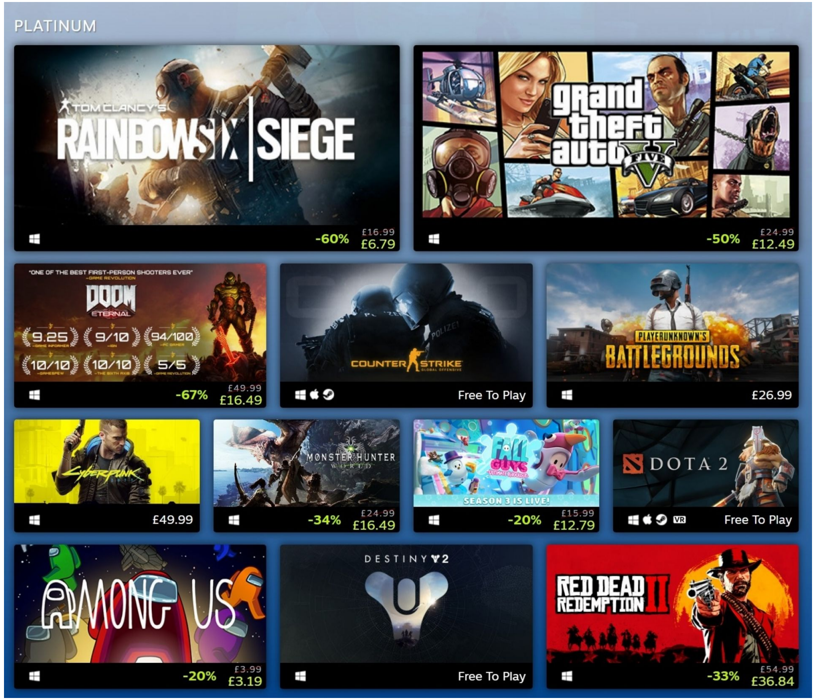
The Problem - What Should the Gamers Play Next?

- Steam is an online gaming platform with more than 120 million monthly active users and over 50,000 games.
- With more than 10,000 new games added in 2022, it is difficult for gamers to pick which game to play next.
- Solution: Recommender algorithm that predicts the user’s playtime in an unseen game.
- How does it benefit:
    - Gamers:
        - Get more precise information on their next potential game
        - More transparency since our algorithm predicts game play times.
        - Can divide current game cost by predicted playtime to get game cost/hour→helps with purchase decision
    - Steam App:
        - Provides users personalized game recommendations.
        - Provide cool statistics for users to discover what type of games they like.

Overview

Goal

Build a game recommendation system that tells users which games to play based on predicted play time using the user's past gaming behavior and similar behavior by other users.



Data

8,691 users

Top 986 games

649,731 playtime records > 0 minutes

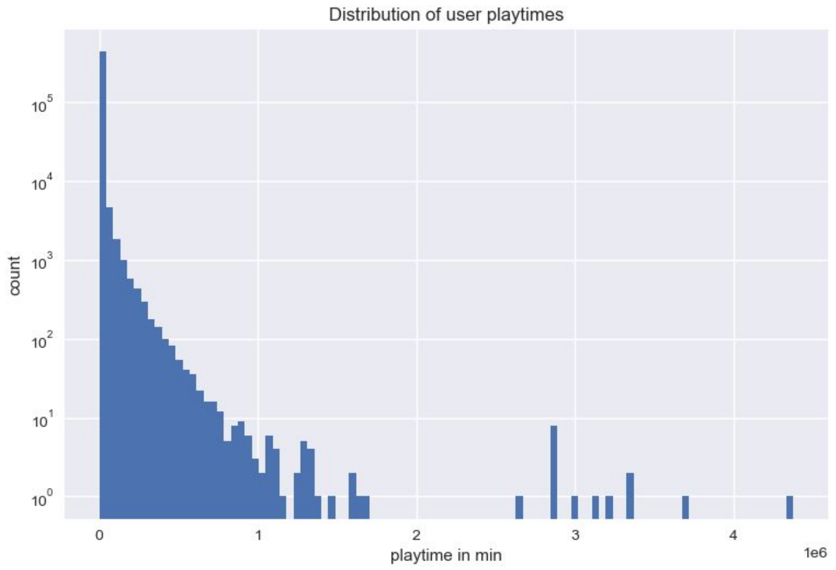
Data collected from Steam Web and SteamSpy API.

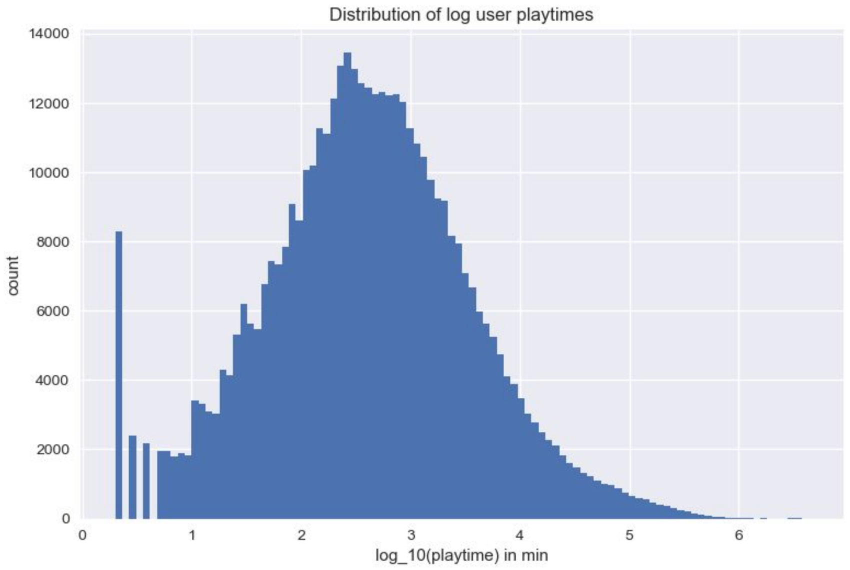
Data Collection Process

- Obtained Steam Web API Key.
- Collected information on owned games from 8,691 public steam users:
  - Generated 17-digit random steam user ids - ~2,000 users collected.
  - Pulled friend list of users to collect ~6,000 additional users from the original users’ friend lists.
- Retrieved top ~1,000 games owned on Steam app.

Play time: The total number of minutes played on record, since Steam started tracking total playtime in 2009.

Log-scaling the data

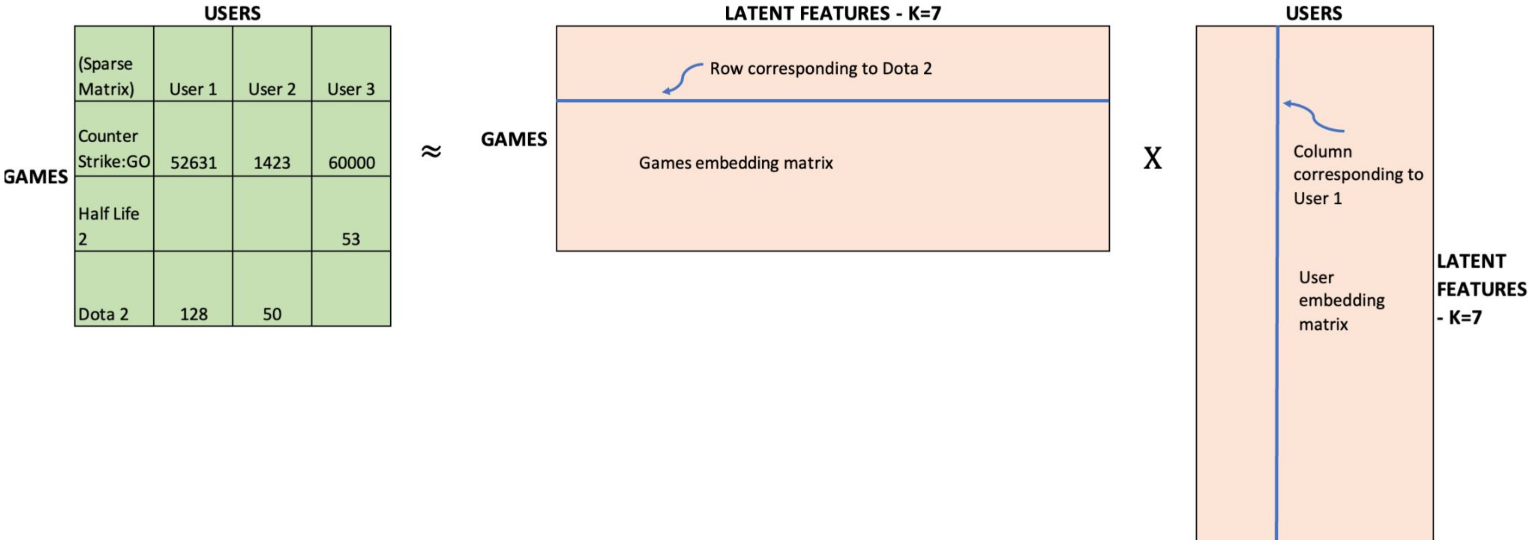




User playtime seems to follow a log-normal distribution, and the outliers prevent convergence unless the playtime is log-scaled; hence we predicted the log of total number of minutes played. (Right is log-scaled, left is not)

Matrix Factorization Algorithm

- To fill in the sparse matrix of Steam user played game time, we use **matrix factorization.**
  - k latent features - hidden features that determine the user's' preferences for each game. Optimized using a validation set, k=7.
  - Employed l2 regularization to prevent overfitting.



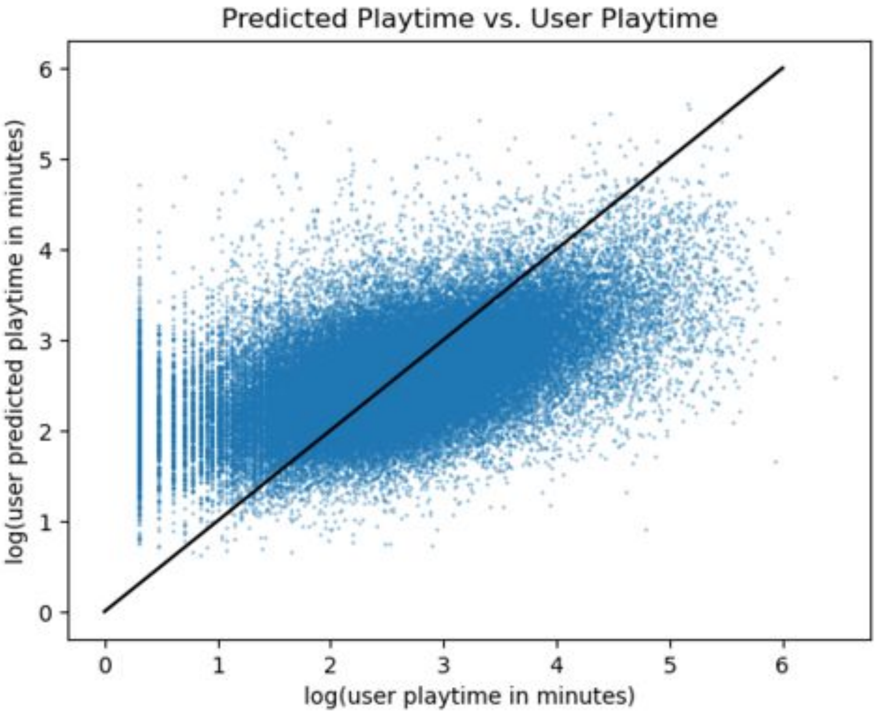
Matrix Factorization Algorithm

Steps:

1) Split non-missing observations into training (85%) and validation set (15%).
2) Initialize the user and game matrices randomly.
3) Use Stochastic Gradient Descent to update weights.
4) Use loss on validation set to optimize number of latent features k, l2 loss, and number of epochs to train.
5) Predict playtime for users’ unplayed games via fitting new user profile weights using gradient descent (but leaving the game weights unchanged).

Results

Validation set mean squared error = 0.65



game_name	prediction (hours)
Counter-Strike: Global Offensive	304.4
Football Manager 2022	199.5
EA SPORTS FIFA 23	123.9
Football Manager 2019	121.4
Cookie Clicker	119.4
Grand Theft Auto V	104.5
Football Manager 2021	102.5
EA SPORTS FIFA 21	90.0
Tom Clancy's Rainbow Six Siege	84.2
PUBG: BATTLEGROUNDS	83.5

Conclusion and Future Directions

- Past behavior and similar user profiles are good predictors of gamer habits.
- Future iterations of the model should:
  - Gather more data to improve the model performance.
  - Account for players who owned but never played the game.
  - Work on getting better predictions for the outliers.
  - Discover the latent features to improve interpretability.