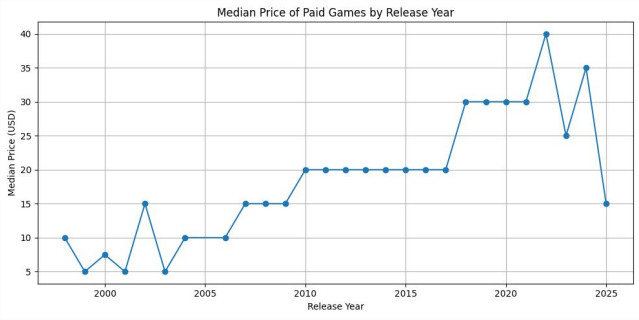
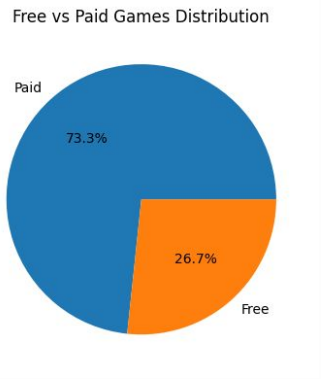
Predicting Steam Game Prices

Stats 418 Yanze Guo

Data Collection & EDA





Data Source

- Scraped Top 1,000 Steam games
- First used SteamSpy API for popularity rankings
- Then queried Steam Store API for:
  - Price in USD
  - Release year
  - Review counts & rating
  - Genres & multiplayer flag
- Output: ~1,000 games × 8 features

Methodology

Cleaning

- All free-to-play games are removed, leaving ≈ 700 paid entries.
- Records missing the year, review stats or price are dropped to keep data clean.
- The genres string is split into dummy variables and only genres that appear in ≥ 30 games are kept to avoid sparse noise.

Feature set:

- Numeric — release year, positive-review ratio, total reviews
- Binary — multiplayer flag plus the genre dummies

Model:

- The target is log(price + 1), which normalises the heavy-tailed price distribution.
- Trained a Ridge regression on 80 % of the data and evaluated on the remaining 20 %.

Result

- On the held-out test set the model achieves $R^2 \approx 0.50$
- The trained pipeline and genre encoder are saved to model/paid_price_model.pkl for use by the API.

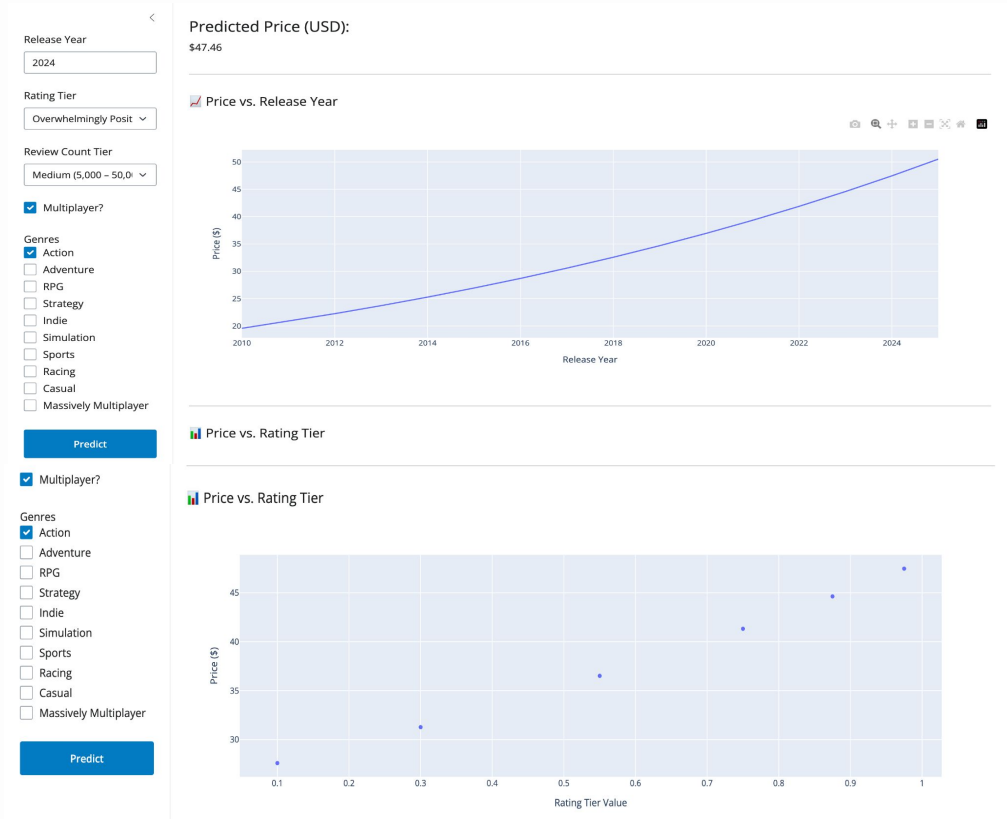
Deployment & Live Demo

Flask API

- Endpoint /predict returns price in JSON
- Docker to Google Cloud Run

Shiny for Python UI

- Sidebar lets users pick year, reviews, rating, genres, multiplayer
- Calls the API with requests
- Separate Docker to Cloud Run, env var API_URL links to API



Limitations

- No discount history or regional pricing
- Does not considered free to play games
- Game pricing is still partly driven by publisher reputation, promotional strategy, and market conditions—factors our current feature set cannot fully capture.