Introduction

The video game industry produces thousands of titles annually, and users rely heavily on game ratings to decide what to play. Accurate rating predictions can enhance recommendation systems and inform both consumers and developers. This project aimed to build a machine learning model that predicts a game's rating based on various attributes gathered from online sources.

Data Collection

To construct our dataset, we scraped data from two major video game-related websites:

- **Website A** (e.g., Metacritic): Provided user and critic scores, platform, genre, release date.
- **Website B** (e.g., Steam or IGDB): Provided user reviews, developer/publisher information, game tags, pricing, and more.

We used Python libraries such as **requests**, **BeautifulSoup**, and **Selenium** to extract and automate the scraping process

Data Cleaning and Wrangling

The raw data from both sources had inconsistencies and missing values. The following steps were taken:

- **Handling missing values**: Dropped irrelevant rows, and imputed values for common fields like price or release year.
- **Unifying formats**: Standardized genres, platforms, and date formats.
- **Merging data**: Combined datasets using common keys like game title and release year, handling fuzzy matches when necessary.
- **Feature engineering**: Created new features like review sentiment score, average word length in descriptions, release decade, etc.

Exploratory Data Analysis (EDA)

We conducted several visualizations to understand the data:

- **Histograms** showing the distribution of game ratings.
- **Boxplots** comparing ratings across genres and platforms.
- **Correlation matrix** to analyze feature relationships.
- **Time trends** showing rating evolution by year or by platform.

These and lots of more insights guided feature selection for the model.

Model Building

Target Variable

The model predicts the **rating score** (e.g., from 0 to 100 or 1 to 10), treated as a regression problem.

Model Pipeline

- **Preprocessing**: Numerical features were scaled; categorical features were one-hot encoded or embedded.
- **Split**: Dataset was split into training (80%) and testing (20%) sets.
- **Models used**:
    - Linear Regression (baseline)
    - Random Forest Regressor
    - XGBoost Regressor
    - Boosting

Evaluation Metrics

We evaluated performance using:

- Mean Absolute Error (MAE)
- Root Mean Squared Error (RMSE)
- R^2 Score

The **Random Forest model** performed the best with an R^2: 0.7553659843749155 and the lowest RMSE.

Results and Interpretation

The model was able to predict game ratings with a high degree of accuracy. Key factors influencing ratings included:

- Review sentiment
- Genre
- Developer reputation
- Release year
- Platform

Games with high user sentiment and from well-known publishers generally scored higher.

Conclusion

This project demonstrated that machine learning can be effectively used to predict video game ratings using publicly available data. The process involved scraping, cleaning, exploring, and modeling data. The final model has potential applications in recommendation engines and market analytics for game developers.

Future Work

- Improve data accuracy with more robust entity matching.
- Include more subjective features such as trailer analysis or in-game screenshots (via computer vision).
- Extend model to predict user engagement or revenue.