DS210 Final Project

This project aims to analyze a dataset of video games to explore the relationship between various factors (such as developers, game genres, platforms, final ratings, wishlist counts, and review counts) and game sales (represented by plays, or the number of players). By cleaning, merging data, and implementing statistical analysis and data visualization in Rust, we seek to answer the following questions:

1. What is the relationship between the final rating (final_rating) of a game and its sales (plays)?
2. How do wishlist counts (wishlists) impact sales?
3. Are review counts (reviews) highly correlated with sales?
4. Which developers, genres, and platforms have games with the highest average sales?
5. Can we gain deeper insights into the gaming market's characteristics from correlations and categorical statistics?

Data Description and Processing Steps

Data Source and Size

The dataset contains over 195,000 records of games, ensuring it is large enough (thousands to tens of thousands of rows). The data was initially scattered across multiple CSV files (e.g., games.csv, developers.csv, genres.csv, platforms.csv, scores.csv). We cleaned and merged the data using Python, producing a single dataset, games_combined_cleaned.csv. (Since the cleaned and merged dataset is already provided, the Python scripts for cleaning are not included.)

Cleaning Steps

1. Remove unnecessary columns: Columns such as description were discarded, leaving only those helpful for the analysis.
2. Combine many-to-many relationships: Fields like developers, genres, and platforms were merged into comma-separated strings to facilitate later statistical analysis.
3. Aggregate ratings: Scores from scores.csv were converted into a single weighted average score (final_rating).
4. Handle missing and invalid values:
   Set final_rating=0.0 for games without ratings.
   Strictly cleaned empty and non-numeric values to ensure compatibility with Rust's strict type system.

Each record represents a single game, with the following fields:

final_rating: The final weighted rating of the game (0.0 indicates no ratings).
plays: The number of players (sales indicator).
wishlists: The number of wishlists the game is on.
reviews: The number of reviews the game has received.
developer: A comma-separated list of developers.

genre: A comma-separated list of genres.
platform: A comma-separated list of platforms.
Additional fields like id, name, and date are included for potential future analyses but are not actively used in this project.

Key Code Files

1. analysis.rs:

Defines the CombinedGame struct and the following key functions:

1. pearson_correlation(x, y): Calculates Pearson correlation coefficient to evaluate the linear relationship between two numerical variables.
2. analyze_categorical(field_name, games, top_n): Performs grouped statistics on categorical variables (e.g., developers, genres, platforms) and outputs the top N by average plays.
3. median(vals): Helper function to calculate the median.
4. plot_scatter(...): Uses the plotters crate to create scatter plots showing the relationship between two numerical variables.

2. main.rs:

Entry point of the program:

1. Reads and deserializes games_combined_cleaned.csv into a vector of CombinedGame.
2. Filters out games with final_rating=0.0, focusing on those with actual ratings.
3. Computes correlations between plays and final_rating, wishlists, and reviews using pearson_correlation.
4. Outputs top 10 statistics for developer, genre, and platform using analyze_categorical.
5. Generates scatter plots for final_rating vs plays and wishlists vs plays using plot_scatter.

Output

Example program output:

```
Loaded 195191 combined game records.
---------- Correlations with plays (only games with final_rating>0)
------------------------------
Correlation(plays, final_rating): 0.0982
Correlation(plays, wishlists): 0.6819
Correlation(plays, reviews): 0.9217
--------------------------------------------------------------------------------------------------------------
---------- Top 10 developer by average plays (final_rating>0) ------------------------------
1: Tencent Holdings -> mean: 33000.00, median: 33000.00
2: ConcernedApe -> mean: 27561.50, median: 27561.50
...
--------------------------------------------------------------------------------------------------------------
```

Analysis

1. Correlations:
    (1) final_rating and plays: Correlation is only 0.0982, indicating a weak linear relationship. However, scatter plots (final_rating_vs_plays.png) show that most high-sales games tend to have higher ratings, suggesting quality games often achieve better sales.
    (2) wishlists and plays: A medium positive correlation (0.6819) suggests that games with more wishlists generally perform better in terms of sales.
    (3) reviews and plays: A very high positive correlation (0.9217) indicates that highly reviewed games are often highly played, possibly due to higher attention and discussions.
2. Categorical Analysis:
    (1) Developers: Top developers such as Tencent Holdings and ConcernedApe dominate in terms of average game sales, showing their strong market influence.
    (2) Genres: Top genres like Brawler, MOBA, and Turn-Based Strategy show a bias towards certain niche but popular categories.
    (3) Platforms: Top platforms like OnLive Game System and Google Stadia reflect the influence of specific platforms on game popularity.
3. Scatter Plots:
    (1) final_rating_vs_plays.png: Shows that high ratings do not guarantee high sales but are often associated with games that achieve high sales.
    (2) wishlists_vs_plays.png: Displays a clearer positive trend, where games with higher wishlist counts tend to have higher sales.

Conclusion

This project demonstrates the end-to-end process of analyzing game data using Rust, from data cleaning and integration to statistical analysis and visualization. Key takeaways include:

Strongest Predictor: Reviews (reviews) have the highest correlation with sales.

Moderate Predictor: Wishlists (wishlists) show a moderate positive impact on sales.

Weakest Predictor: Final ratings (final_rating) alone are not strong sales indicators, though they are often associated with high-sales games.

Market Structure: The results reveal how a few developers, genres, and platforms significantly dominate the market, with many outliers driving average sales higher.

Future improvements could include advanced modeling (e.g., regression, clustering), deeper market segmentation, or further cleaning to address potential data biases.