J. Ian Stewart
Professor Jason Zietz
INFO 2201
6 December 2022

Final Project Write-up

Project Objective

Steam and Epic Games are two of the most popular game services in the world, but how do they compare?

Proposal vs Project

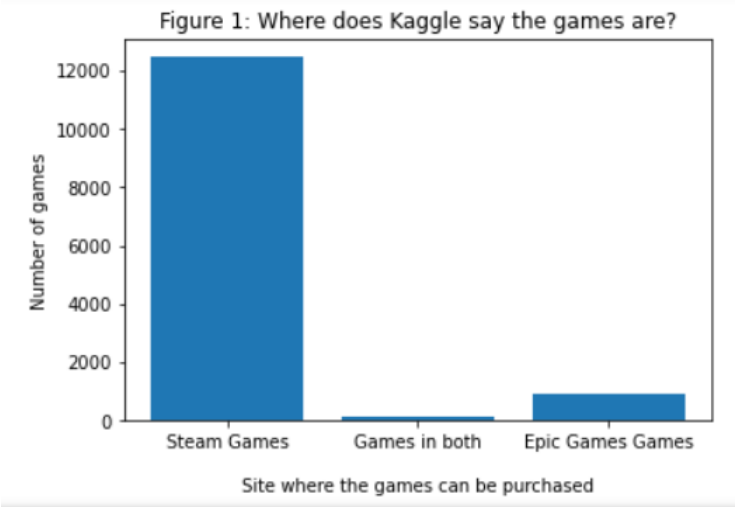
Originally, I wanted to compare Steam, Epic Games, and Xbox's sales or my personal statistics among the three services. I ended up not doing this because no APIs held data on Steam, Epic Games, or Xbox's sales for specific games. Additionally, the Epic Games API I was originally intending to use did not have any way to get my personal statistics. Because I could not do the project I set out to do in my project proposal, I changed my project. I ended up asking different questions. Between just Steam and Epic Games, does one service have better deals if you wait for sales? Does one service have better deals for better games? Is there a genre that is more expensive on Epic Games, on Steam? These are the questions that I wanted my data to answer.

Getting the Games

To accomplish these tasks I got two CSVs from Kaggle: one for Epic Games games and one for Steam games. Unfortunately, these datasets were not perfect. The Epic Games CSV was limited to 899 games, the price was in units of Indian Rupees, and the data set was last updated 7 months ago. I decided to continue to use this dataset because it was the most up-to-date dataset I could find on Kaggle for Epic Games. The Steam dataset was much bigger than the Epic Games dataset, mostly because Steam just has so many more games, and only had an initial price, rather than a current price like the Epic Games dataset. I decided to use this dataset because it was very recently updated, only a month ago, and it was a very complete dataset.

Where are the Games?

With these two datasets, I first read all of the game names into two lists, one for Steam, and one for Epic Games. From here I got a list of games the two datasets had in common by comparing the game titles in both lists. I converted these lists to sets to remove any non-unique values and plotted the number of Steam games, the number of games both services have, and the number of Epic Games games. Figure 1 shows this plot and visually how many more games the Steam dataset had compared to Epic Games, and how few they have in common, only 113 games.



Data Aggregation

With my newly acquired list of games that both companies had, I used the IsThereAnyDeal.com API to get the historical lowest price on Steam and Epic Games. This API uses a certain format for searching games called plain where the title is made to be all lowercase and the spaces are removed. To use the API, I converted all of the game titles that the two services had in common to their plain equivalent. Unfortunately, the endpoint for the historical lowest price only allows for five games to be searched for at a time. I solved this by going through my list five games at a time and putting all of the JSON response objects into a list to later go through. This endpoint works by searching for up to five games and selecting which stores to find the lowest prices on, which were Steam and Epic Games. An issue with this, however, is that the shops and their lowest prices come back in random order. This was a problem because I wanted to make a data frame where each row had the game name, its lowest price on Steam, and its lowest price on Epic Games. To make this data frame, I created three lists: the remaining games that returned data from the API, their lowest price on Steam, and their lowest price on Epic Games. These lists are organized so that the first row of all three lists describes one row of the data frame. To solve the issue of random shop orders, I made a temporary dictionary for each game in each JSON object, where the key was the shop name and the value was the lowest price for that shop. This was much simpler than using an if statement to check which dictionary was for Epic Games and which one was for Steam.

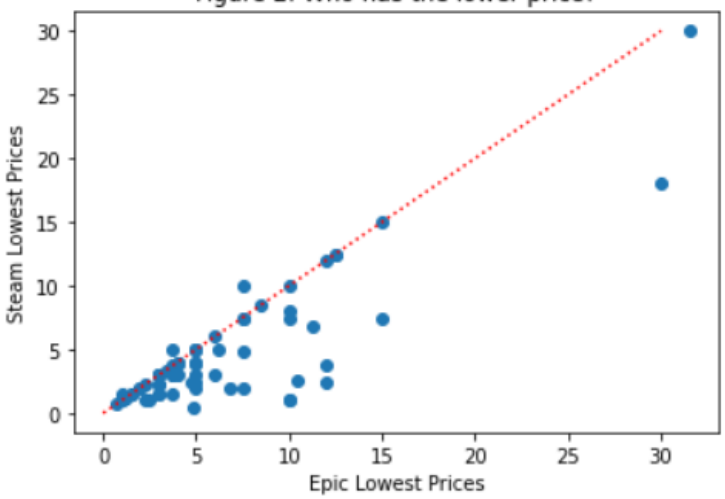
With my newly created data frame, I added two more columns: one for the actual price difference, and one where the decimal was cut off to allow for better data analysis. Unfortunately this data frame now only had 67 games from the 113 earlier, so I decided to add some more data in the form of the Metacritic score and Metacritic's user score. I did this by making my own CSV where the first column was the game name, the second column was the Metacritic score, and the last column was the user score on Metacritic and manually looked up each game's PC version. If

there wasn't any data, I made the data -1 to easily identify it. I then made a new data frame and combined both data frames. With my full data frame, I started my data analysis.

Which service has the lower price?

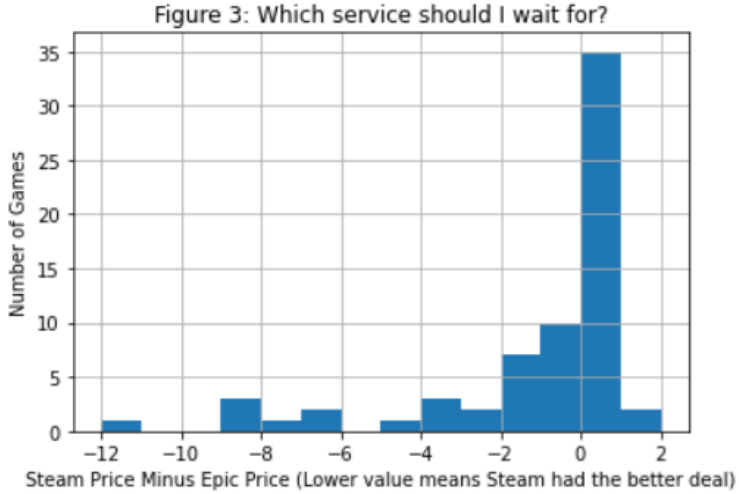
I started by first making a scatter plot to compare Steam's lowest price to Epic Games' lowest price. Figure 2 shows the produced scatter plot with a line indicating both services had the same lowest price. If a point is below this line, then Steam had the better deal compared to Epic Games. There are a lot of data points below this line, meaning that, at least for this dataset, it is better to wait for deals on Steam compared to Epic Games.

Figure 2: Who has the lower price?



To visualize this further, I made a bar chart of the difference in Steam price and Epic Price without the decimal, which is why I cut off the decimal point. Figure 3 shows that when there was a price difference, more often than not, Steam had a lower price.

Figure 3: Which service should I wait for?



Are there better deals for better games?

Next, I was curious if Steam made better deals for better games. I did this by comparing the price difference to the Metacritic and user scores. I made two graphs: one with unclean data and one with clean data. When I say “unclean” I am talking about the -1s that were added if there was no score for that game. Figure 4 shows the unclean data, where the blue dots are the price difference vs Metacritic score and the orange dots are the price difference vs user score. The dots on the left are games where the Metacritic or user score was unavailable. The one game where the user data was not present is much further to the left compared to the missing Metacritic scores because the Metacritic was out of 100 and the user score was out of 10, so I multiplied the user score by 10 to get both on the same scale.

Figure 4: Unclean: Are there better deals on better games?

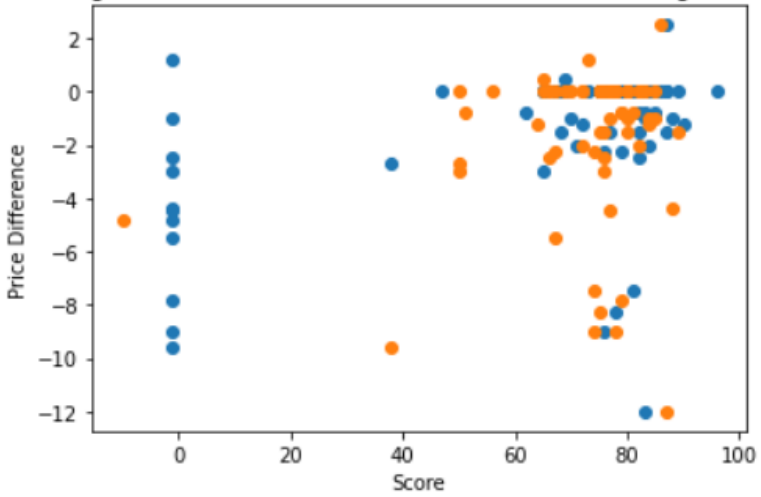
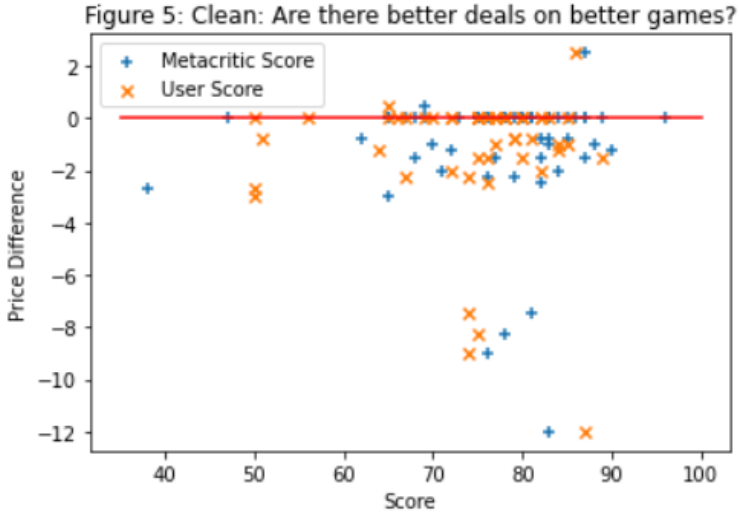


Figure 5 shows the clean data with a nice legend and line signifying a price difference of 0. I used a different version of subplots compared to what we did in class to add the legend to the graph (the source is commented in the code). This figure shows that there is no correlation between how highly a game is rated and how low Steam will go compared to Epic Games.

Figure 5: Clean: Are there better deals on better games?



Is there a more expensive genre?

Finally, I was curious to see if there was a genre that was more expensive compared to others. The Steam dataset had an interesting way of recording the genres where there were columns called GenreIsAction, for example, and it would be only true and false values. I decided to use the same system for Epic Games to compare the same genres for both Steam and Epic Games (excluding massively multiplayer from Epic Games as it does not have that genre). I first had to clean the Epic Games data by getting rid of any games that did not have a price or were free. I ran into a small issue where I would get a warning that I was editing a copy of the data frame and not the actual data frame. I solved it with a stack overflow search (the source is commented in the code). I then made eight new columns to hold the eight genres. For each genre, I searched whether or not the "genres of games" column had the genre in it. I then would populate the row with 1 if the genre was present and 0 if it was not. By filling the columns with 1s and 0s, I can multiply each genre series by the price series to get a list with prices, where all of the numbers that are not 0 are a game with that genre. I then removed all of the 0s to just get the games with that genre. One caveat of this method is that free games are not accounted for as there is no difference between NULL and 0 (this is why I removed the free games). From here I plotted a bar graph where each bar is a genre and the height is the average price of a game in that genre. Figure 5 shows Epic Games' results. Casual games are much cheaper compared to other genres and action, strategy, and RPG games are $2-4 more on average.

Figure 6: On Epic Games, is a genre more expensive than others?

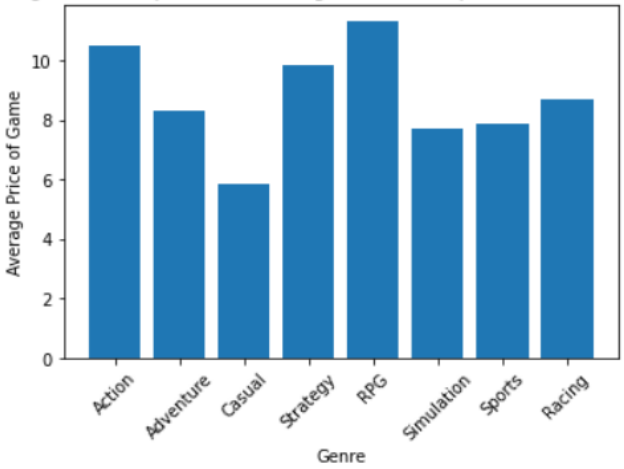
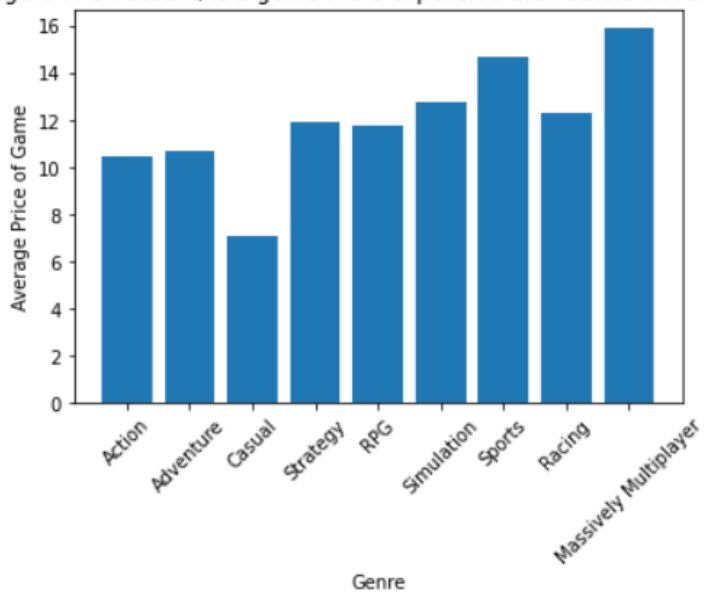


Figure 7 shows Steam's results. Casual games are also cheaper compared to other genres, just like Epic Games. Interestingly, however, RPG and action games are not more expensive compared to other genres, but massively multiplayer and sports games are $2-4 more expensive.

Figure 7: On Steam, is a genre more expensive than others on release?



Conclusions

In conclusion, Steam is better to wait for deals compared to Epic Games because they tend to go lower. Unfortunately, there is no correlation between a game being better and Steam having a lower price compared to Epic Games. I would have loved to buy better games for cheaper on Steam. I found it incredibly interesting that Steam and Epic Games had different genres that were more expensive, but was not surprised to find that casual games were cheap among both services. If I were to continue to work on this project, I would filter out more categories from the Epic Games database, rather than just the genres the Steam database had. I would also try to find a Steam database with a more complete genre list to run the same analysis. Finally, my next question would be: are better games more expensive?