COMP 7150 — Data Science Project Report

Hicks, Eric
elhicks@memphis.edu

Kelly, Craig
cnkelly@memphis.edu

December 2, 2016

Abstract

A raw data source was mining for data, which was cleaned and filtered into a usable dataset. This dataset was searched for interesting details based upon the authors' preconceptions. They were found.

1 Introduction

In this paper, the path and results of the authors' dissection of the Steam platform will be detailed. The reason Steam was chosen for this project, was due to its large acceptance in the PC space. Steam is a company that sells PC, Mac, and Linux games for download over the Internet. With weekly sales, daily releases, a managed library, and no computer limit on games, it is the largest digital distributor of games and the most widely used. Additionally, Steam stores metrics on over a 100 million users on nearly 10,000 games, making it a perfect example of a feature rich dataset. However, as Steam has no saved repository of data publicly available, one was created for this project, hereby called the Steam dataset. [7]

2 Steam Dataset

This dataset and analysis project was inspired by the SteamDB project [1]. SteamDB does not provide data for download, and does not provide an API. SteamDB uses the SteamKit2 library [2] for acessing the Steam API [3]. This does not appear to unusual; for instance the site Rhekua [9] makes no data available either.

Although, several external sites keep track of various parts of Steam, none of them gathers the data already available by web crawling Steam. Of the few sites that record data not saved by Steam such as Rhekua [9], none of them release snapshots of their gathered data. This lead the authors to manually mine data from the main Steam site.

TODO: steamspy (needs update of first paragraph)

TODO: some kind of graph of data process?

TODO: appendix with field listing? or just a link to the repo?

2.1 Acquisition

SteamKit2 is designed for the entire API made available by Steam **??**, much of which requires a developer/partner agreement with Steam **??**. The authors chose to use SteamKit2 as a starting point, but to write their own, slim data acquisition library.

TODO: describe process with steam store

TODO: multiple full pulls from steam and merge with steamspy

2.2 Cleaning and Formatting

TODO: reduced to 78 fields TODO: explain field removal

2.3 Metacritic

While many of the data dimensions are self-explanatory, Metacritic might require some explanation. Metacritic provides a rating service for video games, movies, television shows, and music [4]. The Metacritic score is a proprietary weighted average of various professional critics and publications, scaled from 1 to 100 [5]. It is one of the most frequently used metrics for video game quality [6].

2.4 Caveats

The process used could not capture all data dimensions used by Steam. Some are not recorded and others can only be accessed by an administrator account. The number of owners for each game could not be acquired, but has been estimated using an outside

site [10]. Recommendations were collected as a total of reviews and as two separate values for positive and negative reviews. The sentiment of the reviews that Steam displays was also not able to be captured. Steam doesn't store historical price data, so sale history and price trends can not be analyzed from the data. These limitations did not stop the authors from gaining interesting insight from the data.

3 Predictions

As both authors were familiar with Steam before the start of the project, they made some predictions of what would be found in the data:

TODO: more predictions

1. SteamOS games available on Steam would be a perfect match to Linux games available.
2. The most recommended and the highest rated genre is action.
3. That Metacritic scores are an inverse bell curve when sorted by recommendation, i.e. lower and higher scoring games would have more recommendations that games with a middle score.
4. That free games get more reviews per game and have lower ratings than paid games.

4 Testing

Both authors ran different models on the data independent of each other to cover the most ground. Every feature pair was plotted as a scatter plot to gain a basic understanding of feature relations alongside reading through the raw data. From here other options were tried.

5 Results

The authors found results corresponding to all predictions. Some other interesting results were found as a result of exploratory analysis.

TOOD: craig's analysis

TODO: reference for metacritic

5.1 SteamOS and Linux

As a sanity check, the authors insure that a simple prediction was verifiable by the dataset. By definition, all Steam games available on the Linux platform must also be on the SteamOS platform because SteamOS is only way to run a Steam game on Linux. The data verified this "prediction".

TODO: figure for SteamOS and Linux

5.2 Free vs Non-Free Games

Free games do get more recommendations than paid games (see figure 5.2). However, they are also lower rated lower than paid games (see figure 5.2). In fact, "free to play" as a genre received the most feedback via recommendations (see figure 5.3), but had the lowest rating scores (see figure 5.3). As a result, we conclude that free games tend to have negative reviews; in fact, the plurality of reviews for free games might well be negative. (TODO: last sentence in "Future Work".)

Figure 1: Count of recommendations from users on Free vs Non-Free games

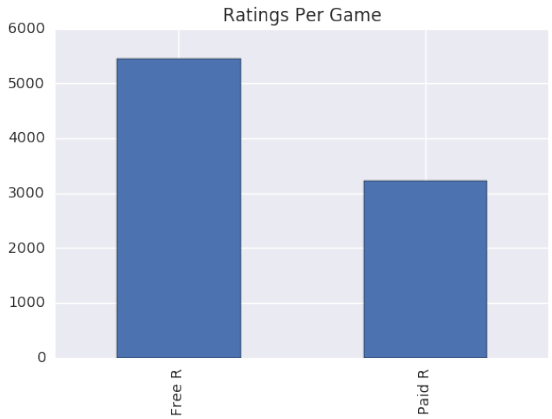


Figure 2: Metacritic mean score on Free vs Non-Free games

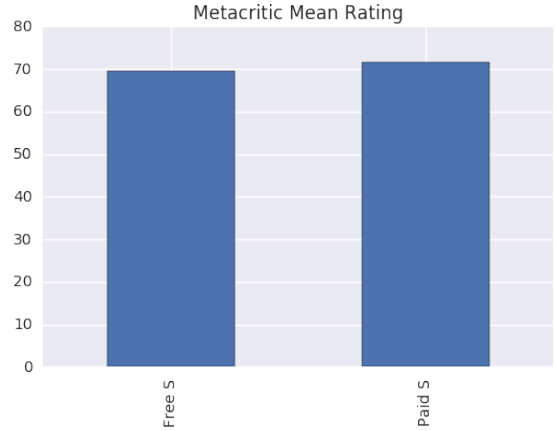
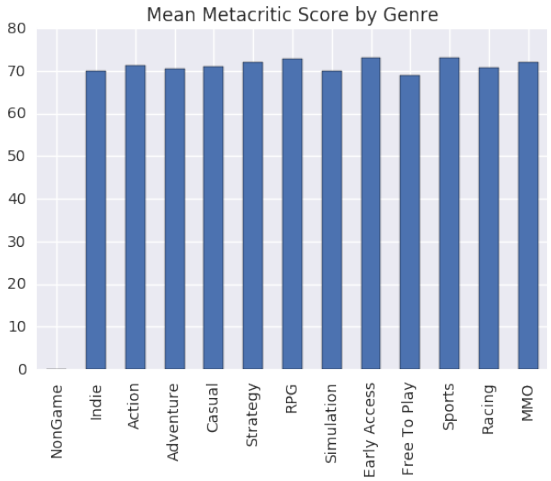


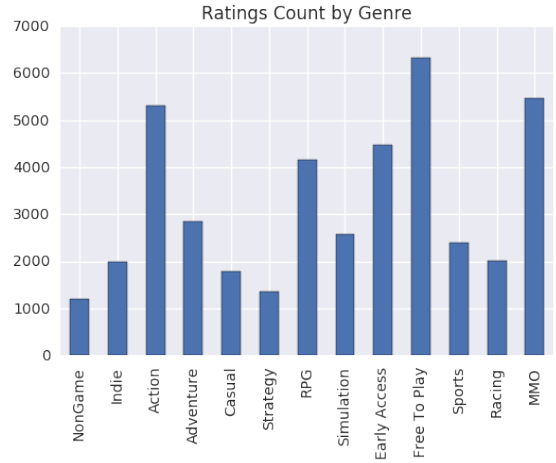
Figure 4: Metacritic mean score by genre



5.3 Genre

The most recommended genre was free to play and not action. The least recommended genre was non-game software (see figure 5.3). The highest scoring genre was sports instead of action. The lowest scoring genre was free to play (see figure 5.3).

Figure 3: Count of recommendations from users by genre



5.4 Recommendations, Ratings, and Price

Although not part of the original prediction, some relationships (or lack thereof) were noticed. Pricing compared to Metacritic scores is mostly uniform. The gaps for certain prices are almost entirely due to Steam's pricing structure. Pricing compared to user recommendations is also nearly uniform. There is a small increase in recommendations for cheaper games, but it is not significant. Please see figures 5.4, 5.4, and 5.4.

Figure 5: Relationship of Count of User Recommendations to Metacritic Score

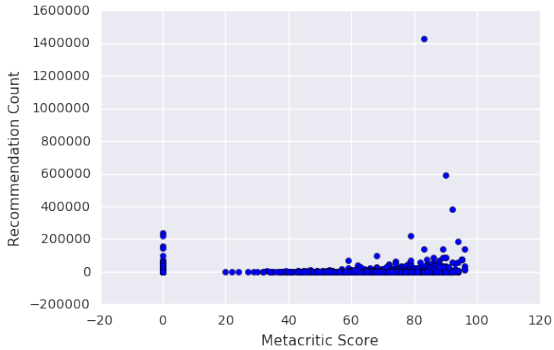


Figure 6: Relationship of Metacritic Score to Initial Price

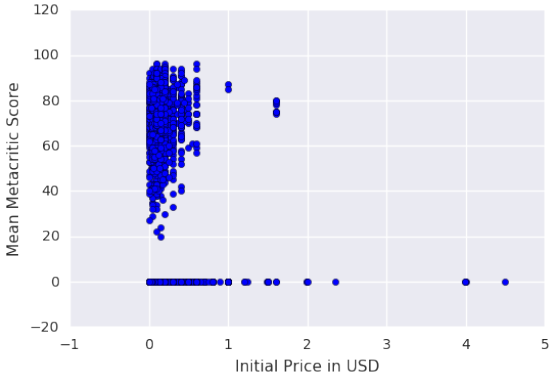
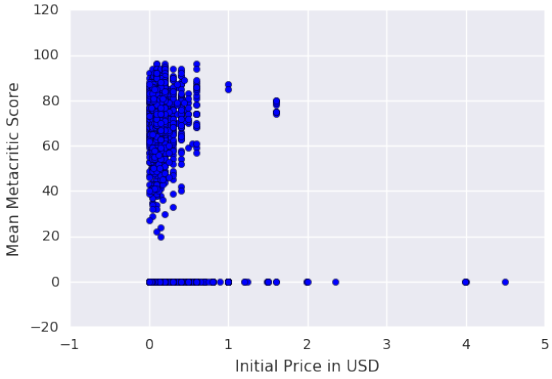


Figure 7: Relationship of Initial Price to Count of User Recommendations



6 Future Work

Given a larger timespan to collect data and more sophisticated web crawlers, more data come be pulled from Steam. This would allow for most, if not all of the caveats mentioned before to be removed. With historical data, price trends for games of different genres could be predicted along with the time frames for price drops. A divided recommendation statistic could be used to determine if certain genres get more positive reviews than others, as well as show a correlation (or lack thereof) between Metacritic scores and positive recommendations.

7 Conclusion

After finding a suitable platform to mine data from, the authors acquired raw data from Steam. This data was cleaned, formatted, and processed to find several interesting results. While all of the authors' predictions were not proven in testing, the ones that were wrong proved to be the most surprising.

TODO: more text

References

[1] .

[2] .

[3] .

[4] .

[5] .

[6] .

[7] Valve Corporation. *Steam*. URL: `http://store.steampowered.com/`.

[8] Valve Corporation. *Steam Community : Steam Web API Documentation*. URL: `https://steamcommunity.com/dev`.

[9] Dillon DeLoss. *Rhekua*. URL: `http://steamsales.rhekua.com/`.

[10] Sergey Galyonkin. *Steam Spy*. URL: `http://steamspy.com/`.