ETL Project Report

Analysts: Sarah Zachrich, Madison Chamberlain, Kevin Schram, Robin Bun, Andrew Marino

Objective: Finding data from Steam (a cloud-base gaming library), we will create tables/collections that are stored in a database. Once the datasets have been identified, we will perform ETL (Extract, Transform, Load) on the data to clean it up and make any necessary joins, filtering, etc.

We chose SteamDB (database) and SteamSpy as our two main data sources. This data was in html format and we used basic scrapes to obtain various data from both sites. We later saved all data to .csv files. We used Jupyter Notebook to clean up the data as specified below. We also had to merge our two datasets. Because we needed to do joins and want to be able to query our data for various items we chose to use a relational database (SQL). Below are some ideas on what to use this data for and some of the challenges we had to overcome.

ETL Process:

Extract:

- We extracted the data by web scraping the Top Rated Games from SteamDB
    - ■ From SteamDB we scraped :
        - Game name
        - Game appid
        - Number of Positive Ratings
        - Number of Negative Ratings
        - The Steam Rating
        - The Percentage of Positive Rating

- We used the appIDs of the games we scraped from the SteamDB to scrape the website SteamSpy
    - ■ From SteamSpy we scraped :
        - Developer(s)
        - Publisher(s)
        - Genre(s)
        - Owners
        - Metascores

Transform

- Cleaning the data
    - ○ Drop the ‘#” column
    - ○ Had to remove some appIDs that did not return any data
    - ○ Replaced ‘n/a’ in the MetaScores with 0's

    - Replaced japanese and russian names with english text
    - Convert Metascore format to decimal current format (89% to .89)
- Merge data sources on game name

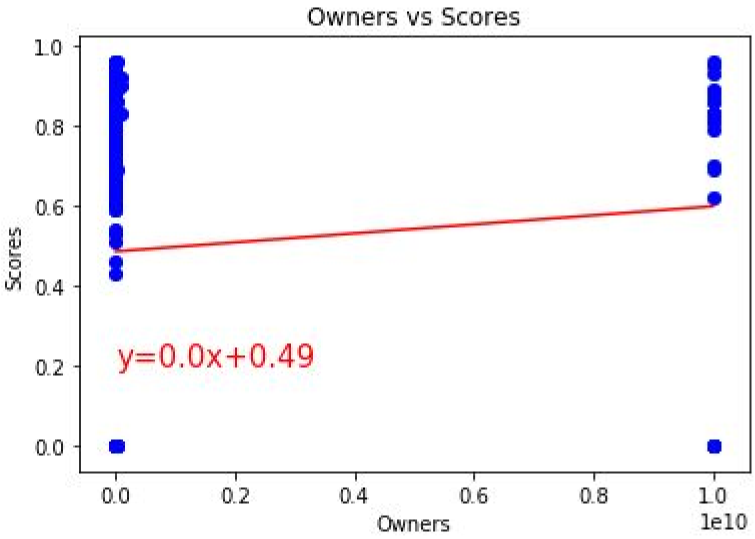
Load

- Create Schema / Data structure for Table/tables
- Load data into relational DB / SQL DB

<u>How we will use the data:</u>

Create a scatterplot of owners vs metascore to see if higher rated games have more data

```
The r-squared value is 0.002274121261447703
```



Other questions to be explored with further data and time:

- Pull data on achievement completion percentage per game to see which games have highest completion rate
- Relation of genre and metascore to explore game popularity
- Player demographics vs game genre (ex. Male vs. Female game preference)
- Price of game vs number of owners also how price changes over time
- Pulling further data from steam API to see what else we can explore

<u>Challenges Encountered:</u>

- Some games titles have special characters & non-English names - This created issues with merging and querying and in part was due to the encoding
- Null rows/values we had to ‘fix’ - either removed or added 0 values
- Some values were ranges which was messing up our queries so we had to change to a single value to be able to use
- Our original scrape was pulling non-genre data as genre. Some games also had multiple genres. We had to fix this to pull correct data and include all genres per game.