Project 2: Extract, Transform, and Load Data
Mason DeJesus and Henry White
July 16, 2024

Introduction

This project utilizes a dataset consisting of crowdfunding campaigns and their related information, such as goals, number of backers, outcome, and category/subcategory. The purpose of this project was to extract the data from two original .xlsx files into four tables and load them into a relational database utilizing PostgreSQL. Before loading, the data needed to be appropriately transformed, recognizing that the source file contained multiple columns combined into one. To demonstrate the SQL database worked properly, data was queried from it and visualizations were constructed.

Database Design Considerations

Data would ultimately be stored across four tables; to ensure the tables could be joined, certain columns were identified as primary and foreign keys. The entity relationship diagram shown below indicates how the primary keys and foreign keys across tables were connected.

Each table contains a “last_updated” column which, during the load phase, takes the current datetime and applies it to the rows of the table. This is best practice for the benefit of any future users of the data seeking history on previous updates.

Extract/Transform/Load Code

In extracting and transforming the data, the strategy involved using the unique categories and subcategories from the campaigns excel file to create two tables, append those tables with unique ids for each category, load the contacts and campaigns in their own dataframe, and then transform the data to match the columns outlined in the ERD.

In the campaign dataframe section, challenges arose when working with source data’s “launched_at” and “deadline” columns. These were originally written as unix values and needed to be written to datetimes. Our conversion initially appended new columns to the dataframe instead of overwriting the original, which caused significant issues when attempting to load the data. Investigation into this issue however yielded a solution which was easily implementable.

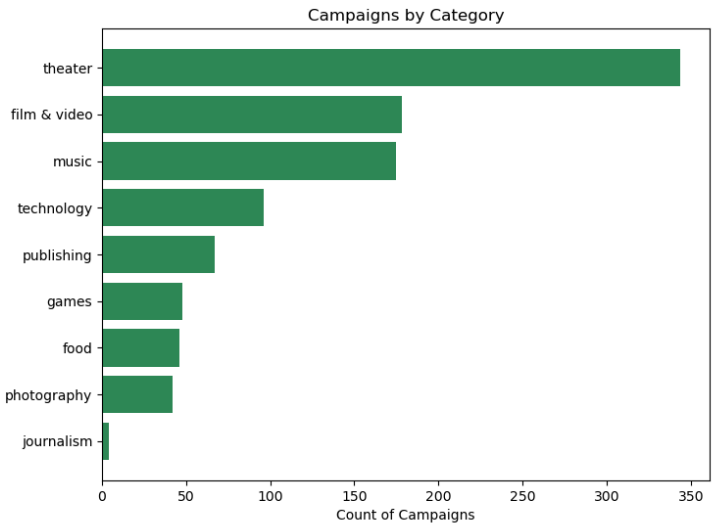
Multiple attempts were needed to load the data into PostgreSQL via python, but the issue arose from a mismatch in the names of columns between our datagrames and the SQL tables. This was resolved relatively quickly, through multiple iterations.

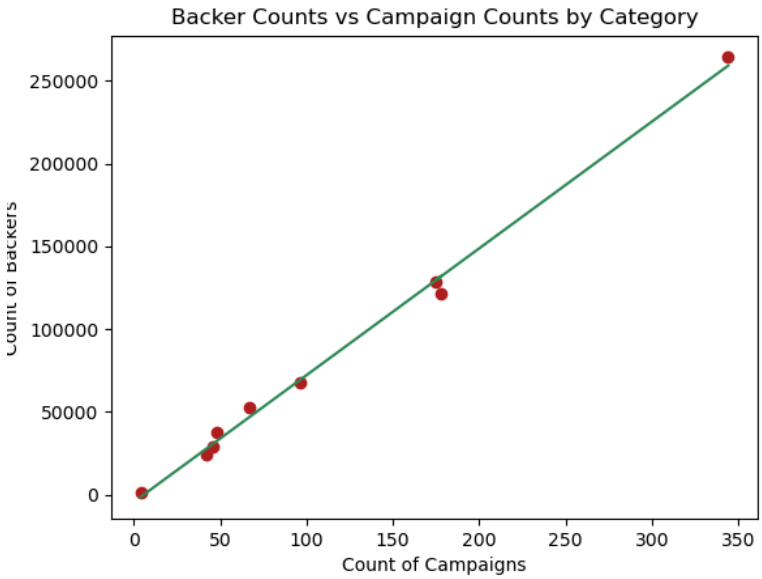
Analysis

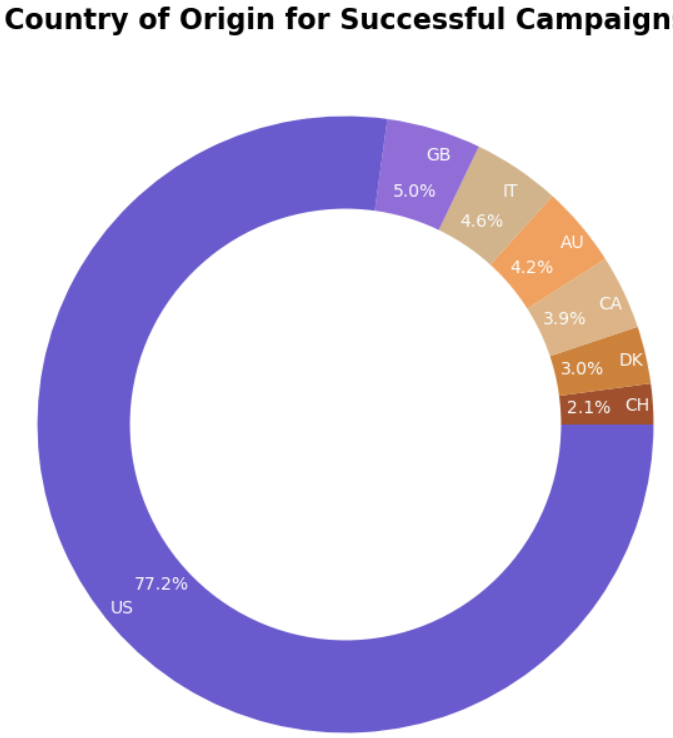
Three different queries were made of the database around the following questions:

1. Which categories had the most campaigns?
2. Is there a correlation between number of campaigns and total number of backers for each category (i.e. is the average number of backers per campaign consistent)?
3. Which countries are the origin of the most successful campaigns?

The questions allowed for the creation of the following visualizations:







Full derivation of these visualizations may be seen in the “Queries” notebook.

The data visualizations demonstrated the following:

1. Theater, film and video, and music were the categories with the most number of campaigns.
2. There is a very strong correlation between number of campaigns and total number of backers for each category, and the average number of backers is 766 per campaign.
3. The vast majority of successful campaigns originate in the United States (US) with the United Kingdom (GB) coming in a distant second at 5%.

Bias/Limitations

This analysis is limited by the sole source of data with not much disclosed about it's collection or surrounding context. There is no easy way to determine if this data is complete or representative of all crowdfunding campaigns. The data is current through 2022, so while not outdated, there are historical factors such as the COVID pandemic that could cause this data to not be representative of future iterations of this project.

Through the nature of the project, there was a specified format for how this data was to be transformed and stored, a potential limitation. A relational database was created; should future relevant data be acquired that is not flat or of a different structure, a separate noSQL database would need to be constructed.

Conclusions

The process of this project led to learning through a very structured path; this demonstrated application of knowledge around the ETL process and relational databases, but also limitations in creativity. However, this can be foundational to future projects or applications of the competencies practiced. Despite setbacks during the project, the desired end result was achieved, and a demonstration of ERD development, data transformation in python, querying with SQL, and the use of data visualizations was present in this project.