DataNarrative 1

Ushasree Thumma
Roll No. :22110272
Computer Science and Engineering

***Abstract*—This report is about analyzing a dataset using python programming including libraries like matplotlib, pandas.**

Keywords—Python, Libraries, Functions, Pandas, Matplotlib, Metadata, ISBN, Registration group element

I. OVERVIEW OF THE DATASET

This dataset is for recommending books. In this dataset, we have:

- six million ratings for ten thousand most popular books.
- Books marked to be read by the users
- Book metadata
- Tags, genres

II. SCIENTIFIC QUESTIONS OR HYPOTHESIS

1. In which country maximum no. of books were published?
2. Which book received the maximum average rating and who wrote it? When it was published?
3. Find out the book's details which received more 5-star ratings.
4. In which year more no. of books were published?
5. The majority of people recommend which books? Generally, people recommend books which have a 5-star rating.
6. Which user have read more books?

III. DETAILS OF LIBRARIES AND FUNCTIONS

1.*Pandas* [1] *Library:*
It is a library in python which is used for working with data sets. It can analyze and give a conclusion for large data also. It can do any operations on the dataset. It can also be used for visualizing the large dataset.

2.*Matplotlib* [2] *Library:*
It is also a library in python but used for plotting the data. We can plot beautifully by conveying various details regarding the graph and can choose colours and markers for plotting. It is the best way to visualize the data.

3. *Function* .loc[]:
"To access a group of rows and columns by labels" *
4. *Function* .iloc[]:
It gives the "purely-location based indexing for selection by position" *
5. *Function* .value_counts():
It gives the frequency for data in descending order
6. *Function* .unique():
It prints only unique values

IV.ANSWERS TO THE QUESTIONS (WITH APPROPPRIATE ILLUSTRATIONS)

1.
Input:
Here is the code:

```
## Counting how many books were published in different countries
eng,french,german,japan,russian,china=0,0,0,0,0,0
for i in books["isbn13"]:
  i=str(i)
  if i=="nan"or i=="":
    pass
  elif i[3]=="0" or i[3]=="1":
    eng+=1
  elif i[3]=="2":
    french+=1
  elif i[3]=="3":
    german+=1
  elif i[3]=="4":
    japan+=1
  elif i[3]=="5":
    russian+=1
  elif i[3]=="7":
    china+=1

y=[eng,french,german,japan,russian,china]

## Plotting pie graph for the no. of books published in different countries
plt.pie(y,shadow=True,colors=["#FFFF00","r","orange","g","c","y"])
plt.legend(["English-speaking countries","French-speaking countries",
            "German-speaking countries","Japan","Russian-speaking countries",
            "People's Republic of China"],
           borderpad=2,bbox_to_anchor=(0.35,1.0),title="Books were published in :")
plt.show()
```

In this code, I am using isbn13 column from books.csv file. According to International ISBN Agency, the allocated registration groups are 0-5,7 if the ISBN registration group element [3] is a single prefix element. It is the second element in 13-digit ISBN and indicates the country or language area where a book was published.

TABLE I

Registration Group element	**Country/language area**
0 or 1	English-speaking countries
2	French-speaking countries
3	German-speaking countries
4	Japan
5	Russian-speaking countries
7	People's Republic of China

So, I have counted how many books (if it's isbn13 has single prefix registration group element) were published in country/language area by checking which registration group element it has and stored them into corresponding variables. Finally, I have plotted pie graph to visualize my findings.

Output:

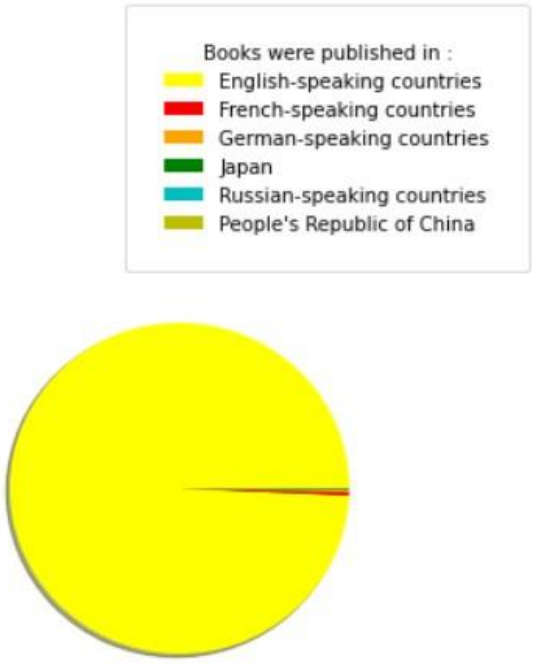

Fig. 1. No. of books published in different countries

Conclusion:
It is clear that most of the books were published in English-speaking countries.

2.
Input:
Here is the code:

```
c=books.loc[books["average_rating"]==max(books["average_rating"]),["authors","original_title","original_publication_year"]]
print(c)
```

I am finding position of the data which satisfies the condition that average rating of that book is equal to the maximum of the average ratings by .loc[] . I am also collecting the corresponding author, name of the book and publication year.

Answer:
I got this result
Author: Bill Watterson
Book: The Complete Calvin and Hobbes
Publication year: 2005

3.
Input:

```
##Details of the book for which readers gave more 5-star ratings
a=books.loc[books["ratings_5"]==max(books["ratings_5"]),["book_id","authors","original_title","average_rating","goodreads_book_id","best_book_id"]].iloc[0]
print(a)
```

In this question also, I am posing a condition like in the above problem. But here I am finding the position of the data which has maximum 5-star ratings by .loc[] and collecting the other details corresponding to that.

Answer:
Book Name: Harry Potter and the Philosopher's Stone
Authors: J.K. Rowling, Mary GrandPré
Book_id:2
goodreads_book_id:3
best_book_id:3
average_rating:4.44

4.
Input:

```
## Calculating how many books published in each year
pub_year_books_num=books["original_publication_year"].value_counts()
pub_year_books_num
```

To solve this question, I am using "original_publication_year" column in books. Here, the elements in that column maps to different books. Finding frequencies of those years will give the no. of books published in that particular year. So, I am using .value_counts() for that column. This will give output having unique values of year and their corresponding frequencies and in decreasing order so that the first element displayed has more frequency. Therefore, the first element will be our answer.
Answer:
In the year 2012, more no. of books were published.
5.
Input:

```
a=rating.loc[rating["rating"]==5,"book_id"].unique()
print(a)
```

If we need to buy any book, then people say that purchase a book which has 5-star rating (positive feedback). So, we should find which book has 5-star rating. Same as above one, by posing the required condition, I have collected the corresponding book ids. But still as users are different, they might have same opinion on the book and gave the same rating. So, I am just collecting unique book ids by using .unique().
Answer:
Array of that book ids was printed.
[258 260 9296 ... 7467 9548 9392]
6.
Input:

```
## Counting no.of users who have given more ratings
a=rating["user_id"].value_counts()
print(a)
```

Users who have given ratings must have read the books. By counting how many times they have given ratings, I will know how many books they have read. So. I've used .value_counts () to count how many times each user has given rating for different books. As discussed in earlier, output will be in descending order and the first element has more frequency.
Answer:
I've got the two user ids 12874, 30944. Both of them have read 200 different books.

V.SUMMARY OF THE OBSERVATIONS

1. By observing the output of the 6th question, I came to know that most of the users are enthusiastic readers.
2. The book "Harry Potter and the Philosopher's Stone" has the maximum 5-star ratings among all the books but it's average rating is not maximum.
3. The book "The Complete Calvin and Hobbes" has maximum average rating but it has low 5 star-rating.

ACKNOWLEDGMENT

I would like to express my sincere gratitude to my ES114 instructor, Dr.R.Shanmuga for giving me opportunity to analyse this dataset. I am extremely grateful for knowledge he offered me.

References

[1] "Pandas Documentation — Pandas 1.0.1 Documentation." n.d. Pandas.pydata.org. https://pandas.pydata.org/docs/.

[2] Matplotlib. 2012. "Matplotlib: Python Plotting — Matplotlib 3.1.1 Documentation." Matplotlib.org. 2012. https://matplotlib.org/.

[3] I. S. "List of ISBN Registration Groups." Wikipedia. Wikimedia Foundation, January 7, 2023. https://en.wikipedia.org/wiki/List_of_ISBN_registration_groups.