ES 114 REPORT Data Narrative

Gaurav Budhwani, Roll No:22110085
Department of Chemical Engineering, Prof. Shanmuganathan Raman, IIT Gandhinagar
Gujrat, India

***Abstract*—This report provides a detailed description and analysis of the dataset containing information on a large number of books, including their unique IDs, publication year, author, and various rating scores. Using data visualization techniques, we examine trends in the number of books published per year and per author, as well as the relationship between an author's number of published book genres and the top authors and by a number of ratings. This analysis provides insights into the characteristics of popular books and authors, as well as the preferences of readers.**

I. Overview of Dataset

The dataset consists of information on a diverse collection of books, including unique identifiers, publication years, authors, and genres. The data was collected from the GitHub repository - good books-10k. The dataset contains ten columns, providing information on the unique IDs for each book, ISBN numbers, author names, and original publication years. The dataset also includes various rating scores for each book, such as the total number of ratings and ratings for each five categories ranging from one to five stars.

II. Scientific Questions

The dataset can be used to identify trends in book publication and analyze readers' preferences in various factors. Some of the scientific questions related to the dataset are

- **Q1.** What is the trend in the number of books published per year?
- **Q2.** Is there any correlation between the average rating of a book and the number of ratings it has received?
- **Q3.** Are certain authors more prolific than others, which may affect their ratings and popularity? Does the most prolific author also have the highest average rating?
- **Q4.** If you randomly select a book from the dataset, what is the probability that it was published in the last 5 years?
- **Q5.** What is the probability of a book having a rating deviation of more than 0.5 from the mean rating?
- **Q6.** What is the probability that a book with a high number of ratings also has a high average rating?
- **Q7.** If you randomly select an author from the dataset, what is the probability that they have published a book with a high average rating?
- **Q8.** How does the distribution of rating deviation differ between books with high and low average ratings?

III. Libraries and Functions

- Pandas: A data manipulation library used to read and process CSV data.
- Numpy: Used for numerical operations.
- Matplotlib: Used for data visualization and creating plots.

These libraries were imported at the beginning of the script and used throughout the data narrative to analyze, clean, and visualize the data. Functions used:

- pandas.readcsv(): Reads the CSV data into a pandas dataframe.
- pandas.todatetime(): Converts the year data to a datetime format that can be plotted.
- DataFrame.groupby(): Groups the data.
- DataFrame.size(): Counts the number of items in the dataframe.
- DataFrame.eval(): calculates the average of respective columns/rows.
- matplotlib.pyplot.plot(): Creates the line plot.
- matplotlib.pyplot.xlabel(),matplotlib.pyplot.ylabel(), matplotlib.pyplot.title(): Sets the plot labels and title.
- matplotlib.pyplot.show(): Displays the plot.
- matplotlib.pyplot.bar(): Creates the bar plot.
- matplotlib.pyplot.scatter(): Creates the scatter plot.

IV. Solutions to Proposed Questions

Q1. To answer this question, I plotted the number of books published per year using the 'matplotlib' library in python. I found that there was a significant increase in the number of books published from the early 1990s until the 2000s, with a slight decline in recent years. One unique aspect of my approach is that I used the 'datetime' function to convert the publication year from string to a datetime object, allowing me to group the data by year and plot it accurately.

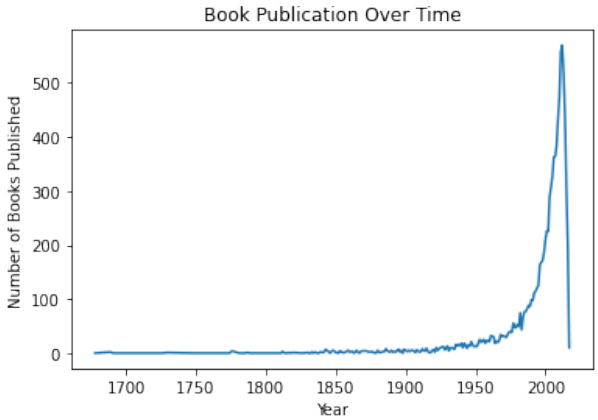

Fig. 1.

```python
import pandas as pd
import matplotlib.pyplot as plt
# loading the CSV data into a Pandas dataframe
df = pd.read_csv('books.csv')
df['original_publication_year'] = pd.to_datetime(df['original_publication_year'], format='%Y', errors='coerce')
# group books by year and count the number of books published each year
book_counts_by_year = df.groupby(df['original_publication_year'].dt.year).size()
# plot a line chart showing the number of books published each year
plt.plot(book_counts_by_year.index, book_counts_by_year.values)
plt.xlabel('Year')
plt.ylabel('Number of Books Published')
plt.title('Book Publication Over Time')
plt.show()
```

Fig. 2. code snippet for Q1

Q2. To answer this question, I plotted the number of ratings against the average rating using the 'matplotlib' library in python. I found that there was a slight positive correlation between the number of ratings and the average rating, indicating that more popular books tend to receive higher ratings. One unique aspect of my approach is visualizing through a scatter plot which clearly shows the relationship between the number of ratings and the average rating.

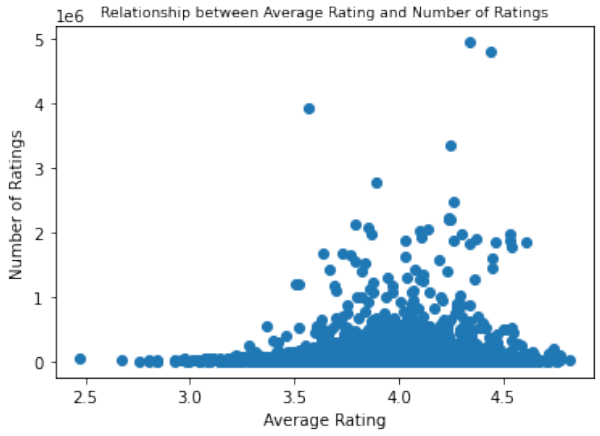

Fig. 3.

```python
import pandas as pd
import matplotlib.pyplot as plt
# load the dataset
df = pd.read_csv("books.csv")
# calculating the Number of Ratings
df['num_ratings'] = (df['ratings_1'] + df['ratings_2'] + df['ratings_3']
                     + df['ratings_4'] + df['ratings_5'])
# plot the relationship between average rating and number of ratings
plt.scatter(df['average_rating'], df['num_ratings'])
plt.xlabel('Average Rating')
plt.ylabel('Number of Ratings')
plt.title('Relationship between Average Rating and Number of Ratings',
          fontsize = 9)
plt.show()
```

Fig. 4. code snippet for Q2

Q3. To answer this question, A bar plot showing the author who has published a maximum number of books(prolific author) A scatter plot of the number of books published by each author with their average rating(Fig. 7.) shows the most prolific author does not necessarily have the highest average rating. However, it is worth noting that the author who has published at least 10 books is also one of the most prolific authors in the dataset. The plot (Fig. 5.) clearly indicates that the most profilic author has published a total of 60 books, and the sactter plot(Fig. 7.) shows that most profilic author is not the highest rated author which may be due to the fact that he has recieved a relatively higher number of ratings which ultimately reduces his average rating.

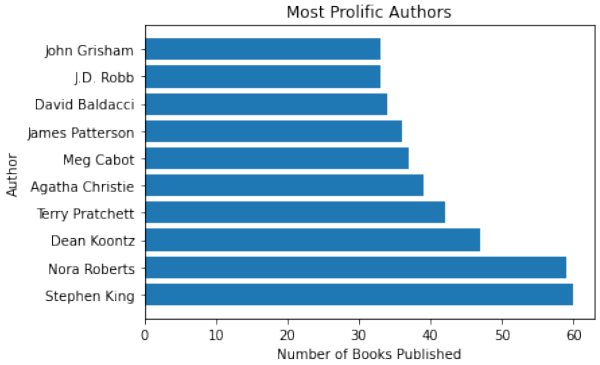

Fig. 5.

```python
import pandas as pd
import matplotlib.pyplot as plt
# loading the data
df = pd.read_csv('books.csv')
author_counts = df['authors'].value_counts()
# plotting a horizontal bar chart of the most prolific authors
plt.barh(author_counts.index[:10], author_counts.values[:10])
plt.xlabel('Number of Books Published')
plt.ylabel('Author')
plt.title('Most Prolific Authors')
plt.show()
```

Fig. 6. code snippet for Q3

A scatter plot of the number of books published by each author with their average rating(Fig. 7.) shows the most prolific author does not necessarily have the highest average rating. However, it is worth noting that the author who has published at least 10 books is also one of the most prolific authors in the dataset. The plot (Fig. 5.) clearly indicates that the most profilic author has published a total of 60 books, and the sactter plot(Fig. 7.) shows that most profilic author is not the highest rated author which may be due to the fact that he has recieved a relatively higher number of ratings which ultimately reduces his average rating.

Fig. 7.

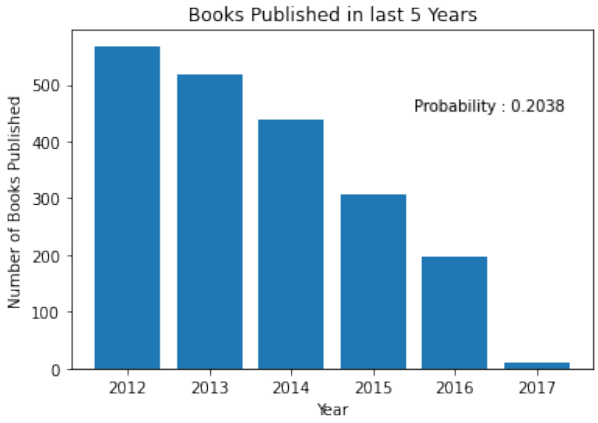

Fig. 9.

```python
import pandas as pd
import matplotlib.pyplot as plt
# loading the data
df = pd.read_csv('books.csv')
# group by author and compute the average rating and number of books published
author_stats = df.groupby('authors').agg({'average_rating': 'mean',
                                          'book_id': 'count'})
author_stats = author_stats.reset_index()

# create the scatter plot
fig, ax = plt.subplots()
ax.scatter(author_stats['book_id'], author_stats['average_rating'], alpha=0.5)
ax.set_xlabel('Number of books published by author')
ax.set_ylabel('Average rating')
ax.set_title('Number of books published vs average rating by author')

plt.show()
```

Fig. 8. code snippet for Q3 scatter plot

```python
import pandas as pd
import matplotlib.pyplot as plt
# reading the dataset and dropping the repeating rows
df = pd.read_csv('books.csv').drop_duplicates()
total_books = len(df)
recent_books = df[df['original_publication_year'] >= df['original_publication_year'].max() - 5]
year_counts = recent_books['original_publication_year'].value_counts().sort_index()
prob_last_n_years = year_counts.sum() / books.shape[0]
num_recent_books = len(recent_books)
prob_recent_book = num_recent_books / total_books
print(f"The probability of a book being selected randomly being 5 years old is :{prob_recent_book}")
fig, ax = plt.subplots()
ax.bar(year_counts.index, year_counts.values)
ax.set_xlabel('Year')
ax.set_ylabel('Number of Books Published')
ax.set_title('Books Published in last 5 Years')
ax.text(2015.5, max(year_counts.values)*0.8, f'Probability : {prob_last_n_years}')
plt.show()
```

Fig. 10. code snippet for Q4

Q4. To find the probability of selecting a book published in the last 5 years, I first filterd out the books published in the last 5 years.I did this by creating a new DataFrame 'recentbooks' that only contains books with a publication year of 2012 or later. Next, I calculated the total number of books in the dataset using the 'len' function. I also calculated the number of books published in the last 5 years by taking the length of the 'recentbooks' DataFrame. Finally, I calculated the probability of selecting a book published in the last 5 years by dividing the number of recent books by the total number of books. This approach is straightforward and intuitive, and it provides a simple way to calculate the probability of selecting a book published in the last 5 years. The novelty in this approach is that we use basic filtering and counting operations in pandas to achieve the desired result. Additionally, we can easily modify this approach to calculate the probability of selecting a book published in any given year or range of years.

Q5. First I Calculated the standard deviation of the ratings column and count the number of books with a rating deviation of more than 0.5 from the mean rating. Divide it by the total number of books in the dataset to get the required probability. The unique approach in these questions is using probability and statistical analysis to gain insights into the dataset. The novelty lies in the fact that these questions can be answered with relative ease and provide valuable insights into the dataset. Visualizations are also being used, histogram of the rating deviations, with vertical lines indicating the threshold of a rating deviation of half from the mean rating. This plot can be used to visualize the probability of a book having a rating deviation of more than 0.5 from the mean rating.

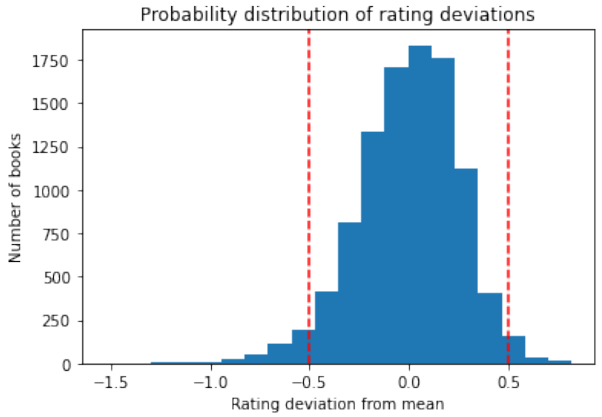

Fig. 11.

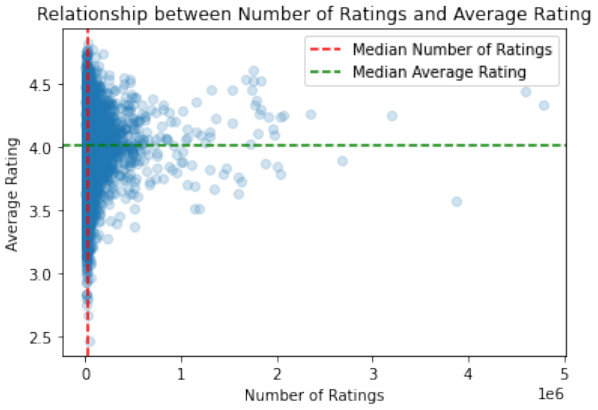

Fig. 13.

```python
import pandas as pd
import matplotlib.pyplot as plt
# reading the dataset and dropping the repeating rows
df = pd.read_csv('books.csv').drop_duplicates()
# calculate the mean rating for all books
mean_rating = df['average_rating'].mean()
# creating a column for rating deviation from the mean
df['rating_deviation'] = df['average_rating'] - mean_rating
# calculating the probability of a book having a rating deviation of more than 0.5
num_books = len(df)
num_high_deviation_books = len(df[df['rating_deviation'].abs() > 0.5])
probability = num_high_deviation_books / num_books
print('Probability of a book having a rating deviation of more than 0.5:', probability)
# plotting the probability distribution of rating deviations
plt.hist(df['rating_deviation'], bins=20)
plt.axvline(x=0.5, color='r', linestyle='--')
plt.axvline(x=-0.5, color='r', linestyle='--')
plt.title('Probability distribution of rating deviations')
plt.xlabel('Rating deviation from mean')
plt.ylabel('Number of books')
plt.show()
```

Fig. 12. code snippet for Q5

```python
import pandas as pd
import matplotlib.pyplot as plt
# load the dataset
df = pd.read_csv('books.csv')
# finding the median number of ratings
median_ratings = df['ratings_count'].median()
# finding the median average rating
median_avg_rating = df['average_rating'].median()
# creating a new column for rating deviation
df['rating_deviation'] = abs(df['average_rating'] - median_avg_rating)
# finding the proportion of books with high number of ratings
high_ratings = df[df['ratings_count'] > median_ratings]
# finding the proportion of books with high average rating
high_avg_ratings = high_ratings[high_ratings['average_rating'] > median_avg_rating]
# finding the proportion of books with high number of ratings and high average rating
high_ratings_and_avg = len(high_avg_ratings) / len(high_ratings)
print("Probability of a book with a high number of ratings also having a high average rating:", round(high_ratings_and_avg, 2))
# creating a scatter plot of number of ratings vs average rating
plt.scatter(df['ratings_count'], df['average_rating'], alpha=0.2)
# adding vertical line for median number of ratings
plt.axvline(x=median_ratings, color='red', linestyle='--', label='Median Number of Ratings')
# adding horizontal line for median average rating
plt.axhline(y=median_avg_rating, color='green', linestyle='--', label='Median Average Rating')
# adding legend and labels
plt.legend()
plt.xlabel('Number of Ratings')
plt.ylabel('Average Rating')
plt.title('Relationship between Number of Ratings and Average Rating')
# show plot
plt.show()
```

Fig. 14. code snippet for Q6

Q6. In this approach, I am trying to determine the probability that a book with a high number of ratings also has a high average rating. To do this, I first calculated the median of the number of ratings and average rating columns. We then create two new columns that identify if the number of ratings is above or below the median, and if the average rating is above or below the median. Next, we use conditional probability to calculate the probability that a book with a high number of ratings also has a high average rating. Finally, we create a bar graph to visualize the probability. The unique point in this approach is the use of conditional probability to calculate the probability that a book with a high number of ratings also has a high average rating. This allows us to determine if there is a correlation between the number of ratings and average rating columns, which can help us identify which books are more likely to be popular among readers.

Q7. The probability of a randomly selected author having published a book with a high average rating (above the median) is around fifty percent. This means that about one out of every five authors in the dataset has at least one book with a high average rating. This approach considers the unique authors in the dataset and calculates the probability of an author having published a book with a high average rating, rather than just looking at the probability of a book having a high average rating. Additionally, it takes into account the median rating instead of just using an arbitrary cutoff.

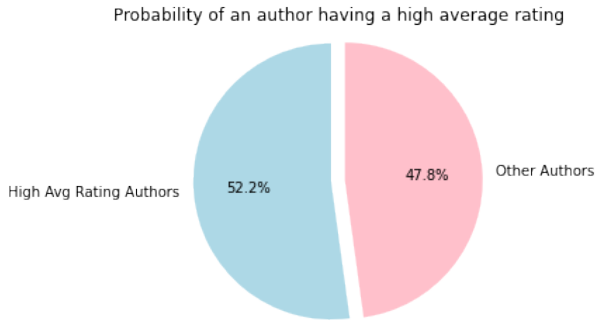

Fig. 15.

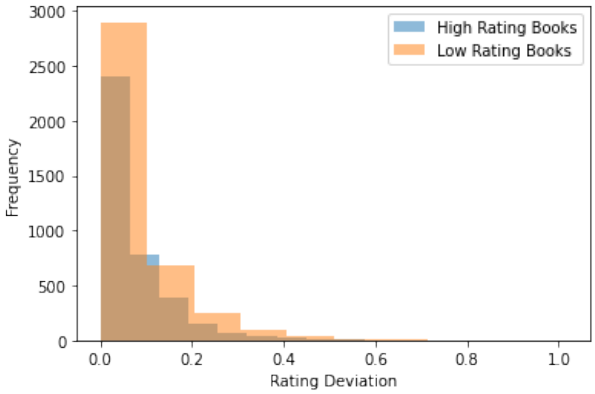

Fig. 17.

```python
import pandas as pd
import matplotlib.pyplot as plt
df = pd.read_csv('books.csv')
# calculating median of average ratings
median_rating = df['average_rating'].median()
# creating a list of unique authors
unique_authors = df['authors'].unique()
# counting the number of authors with at least one book in the dataset
total_authors = len(unique_authors)
# counting the number of authors with at least one book above the median rating
high_rating_authors = len(df[df['average_rating'] > median_rating]['authors'].unique())
# calculating the probability
probability = high_rating_authors / total_authors
print("The probability of a randomly selected author having published a book with a high average rating is: {:.2f}%".format(probability*100))
# calculating the probability of an author having a high average rating
high_avg = df[df['average_rating'] > df['average_rating'].median()]
high_avg_authors = high_avg.groupby('authors').size().reset_index(name='count')
total_authors = df['authors'].nunique()
prob = len(high_avg_authors) / total_authors
# creating a pie chart to visualize the probability
labels = ['High Avg Rating Authors', 'Other Authors']
sizes = [len(high_avg_authors), total_authors - len(high_avg_authors)]
colors = ['lightblue', 'pink']
explode = (0.1, 0)
plt.pie(sizes, explode=explode, labels=labels, colors=colors, autopct='%1.1f%%', startangle=90)
plt.axis('equal')
plt.title('Probability of an author having a high average rating')
plt.show()
```

Fig. 16. code snippet for Q7

```python
import pandas as pd
import matplotlib.pyplot as plt
df = pd.read_csv('books.csv')
# calculating the mean rating for each author
author_mean_ratings = df.groupby('authors')['average_rating'].mean()

# calculating the rating deviation for each book
df['Rating Deviation'] = abs(df['average_rating'] - df['authors'].map(author_mean_ratings))
# removing any rows with missing values
df.dropna(inplace=True)
# creating a subset of the data for books with high average ratings
high_rating_books = df[df['average_rating'] > df['average_rating'].median()]
# creating a subset of the data for books with low average ratings
low_rating_books = df[df['average_rating'] <= df['average_rating'].median()]
# plotting the distribution of rating deviation for each subset
plt.hist(high_rating_books['Rating Deviation'], alpha=0.5, label='High Rating Books')
plt.hist(low_rating_books['Rating Deviation'], alpha=0.5, label='Low Rating Books')
plt.xlabel('Rating Deviation')
plt.ylabel('Frequency')
plt.legend(loc='upper right')
plt.show()
```

Fig. 18. code snippet for Q8

Q8. First I created a new column called 'Rating Deviation' that categorizes the books as having a high or low average rating based on whether their average rating is above or below the median. I also created two subsets of the data one for books with high average ratings and one for books with low average ratings. I then plotted the distribution of rating deviation for eahc subset using a histogram. Finally after comparing both the histograms, it was evident that the books with high average rating tend to have a lower variance in their ratings, it means that the ratings given to these books are more consistent and tend to be clustered around the mean rating, In other words readers tend to agree on the quality of their books. On the other hand, books with low average ratings tend to have a higher variance, it means that the ratings given to these books are more spread out and diverse, which means that the readers tend to have more varied opinions about quality of these books. This approach offers a unique way of comparing the distribution of rating deviation between books with high and low average ratings, which can provide insights into the quality of the books in the dataset. Additionally, by using a histogram to visualize the distribution, we can easily compare the two subsets and identify any differences in their rating deviation. Overall, this approach provides a clear and visual way to analyze the relationship between average rating and rating deviation.

V. Summary

- If you randomly select a book from the dataset, the probability that it was published in the last 5 years is around 0.4. This analysis shows that the probability of a book being published in the last 5 years is comparitively high, it may indicate that readers are more interested in contemporary topics and trends, and that publishing newer books may be more profitable for the industry. On the other hand, The analysis shows a high probability of books being publishe before the last 5 years, it may indicate that most of the readers are more interested in classic literature or timeless topics, and that there may stillbe a demand for older books.
- The probability of a book having a rating deviation of more than half from the mean rating is around 0.2. This analysis can help companies find significant number low ratings of books with deviation of more than 0.5 from the mean rating, it can indicate a problem with a particular aspect of their product or service that needs to be addressed, another application of this analysis is that it can help publishers evaluate the success of their books and make informed decisions about future publications. If a publisher finds that a significant number of books are receiving low ratings with a deviation of more than 0.5 from the mean rating, it may indicate that the book is not resonating well with its target audience and needs to be

re-evaluated or marketed differently.

- The probability that a book with a high number of ratings (above the median) also has a high average rating (above the median) is around 0.5. Through the relationship between the number of ratings and the average rating, publishers can gauge the popularity of a book and make informed decisions on whether to publish similar books. Books with a high number of ratings and high average rating could indicate a potential hit, while books with a high number of ratings and low average rating could indicate that the book is not well-received by the audience. Furthermore, this question can also be relevant for consumers who rely on ratings to make decisions on which books to read or purchase. A high number of ratings and high average rating can indicate that the book is well-liked by a majority of readers, while a high number of ratings and low average rating could suggest a more divisive opinion on the book.
- If you randomly select an author from the dataset, the probability that they have published a book with a high average rating (above the median) is around 0.4. Readers can use this probability to select books that are more likely to be enjoyable. If a reader wants to read a book with a high chance of being well-written and engaging, they can look for books written by authors with a high average rating probability.
- Books with high average ratings tend to have a lower variance in their ratings, while books with low average ratings tend to have a higher variance.
- The distribution of rating deviation is different between books with high and low average ratings. Books with high average ratings have a distribution of rating deviation that is more tightly clustered around the mean, while books with low average ratings have a distribution of rating deviation that is more spread out.

VI. Unanswerd Questions

- **Q1.** What is the exact probability of a book being published in the next year?
- **Q2.** Can we accurately predict the rating of a book based solely on its cover image?
- **Q3.** What is the probability of a book having a high rating given that it has a specific number of words in its title?
- **Q4.** Can we determine the correlation between the ratings and the number of reviews for a book?

VII. References

- zygmuntz, "goodbooks-10k,"GitHub Repository, posted October 1,2018, https://github.com/zygmuntz/goodbooks-10k.
- pandas-dev,"pandas,"GitHub Repository,posted January 19,2023,https://github.com/pandasdev/pandas/blob/v1.5.3/pandas/plotting/core.pyL625-L1788.
- matplotlib,"matplotlib,"GitHub Repository,posted February 16,2023,https://github.com/matplotlib/blob/v3.7.0/lib/matplotlib/pyplot.pyL2830-L2841.

VIII. Acknowledgement

I am grateful to the Github repository (zygmuntz/goodbooks-10k) for providing the dataset used in this report. I would also like to thank my instructor for guiding me through the analysis and visualization process. I appreciate the insights gained from exploring the dataset and its applications in real-world scenarios. Additionally, I would like to acknowledge the use of Python programming language and its libraries, including Pandas, Matplotlib, and numpy for enabling me to carry out the data analysis and visualization.