Documentation - exercise 3
dataset : Book Dataset

Professor : Dr. Kheradpisheh
Teacher Assistant : MohammadReza Khanmohammadi

By: Katayoun Kobraei

1. Introduction to Dataset

The Book price dataset contains key details that capture different aspects of a book's identity and how it is received. Here's a brief rundown of the main features:

- Title: The unique identifier for the book.
- Author: The person or people who wrote the book, a significant factor in how the book is perceived.
- Edition: Information about the specific edition of the book, impacting its rarity and desirability.
- Reviews: User feedback that gives us a sense of how well-received and popular a book is.
- Ratings: Overall scores given by readers, contributing to the perceived quality of the book.
- Synopsis: A brief summary of the book's content, offering insights into its themes and narrative.
- Genre: The category the book falls into, influencing its intended audience and market.
- BookCategory: The broader category under which the book is classified, giving additional context to its nature.
- Price: The variable we aim to predict, representing the market price of the book.

By making use of these features, we are encouraged to apply advanced data science techniques to discover hidden connections, patterns, and insights. These findings can ultimately improve the accuracy of predictions for book prices that is the actual task.

2. Abstract

The main purpose of this assignment is to gain a deep understanding of how to work with raw data and transform it into meaningful features that will improve model accuracy beyond the norm. This process involves identifying key features, dealing with missing data, and understanding how scaling and standardization affect model performance. We will explore categorical variables, which will enhance and enrich our data preprocessing journey. This will leads to a deep understanding of feature engineering and preprocessing the data.

3. Overview of the dataset

Train and test data's shape:
In the first place we concat test data and train data. Because we wanted all changes in columns (like encoding, feature extraction) to happen for both of them. But we separated these two dataset using a column in concat dataset named 'Set' to have access to them separately. The value of training data would be 'train' and the value of test data would be 'test'. Final size of the concat dataset was 6236 rows × 11 columns.

Initial features:
We check columns to be familiar with all data features. Then we print some examples of these columns:

Title:

```
1                        Guru Dutt: A Tragedy in Three Acts
```

Author:

```
1           Arun Khopkar
```

Edition:

```
1         Paperback,- 7 Nov 2012
```

We noticed that this feature is not just the released date and it contains more meaningful data.

Reviews:

```
1       3.9 out of 5 stars
```

Ratings:

```
1        14 customer reviews
2         6 customer reviews
3        13 customer reviews
4          1 customer review
```

We saw that actually reviews and rationing should be something numerical but they are saved as a string so we need to extract the actual number out of these.

Describe target value:

	Price
count	5699.000000
mean	554.857428
std	674.363427
min	25.000000
25%	249.000000
50%	373.000000
75%	599.000000
max	14100.000000

To check type of each column:

```
RangeIndex: 6236 entries, 0 to 6235
Data columns (total 11 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Title          6236 non-null   object
 1   Author         6236 non-null   object
 2   Edition        6236 non-null   object
 3   Reviews        6236 non-null   object
 4   Ratings        6236 non-null   object
 5   Synopsis       6236 non-null   object
 6   Genre          6236 non-null   object
 7   BookCategory   6236 non-null   object
 8   Price          5699 non-null   float64
 9   Set            6236 non-null   object
 10  Unnamed: 0     537 non-null    float64
dtypes: float64(2), object(9)
```

4. Feature Engineering & Transformation

Authors Analysis:
Some books had multiple authors. If we check these rows we see multiple authors come with these punctuation. We printed to see how many books had multiple authors. We saw just 4 books had 5 authors. 19 books had 4 authors, 58 books had 3 authors and 238 books had 2 authors. So we needed to convert all of these values to a unique format. We deleted all punctuation and separated all authors to a list.

	Title	Author	Num_Authors
11	Blockchain Revolution: How the Technology Behi...	Don Tapscott, Alex Tapscott	2
16	My First Book of London	Charlotte Guillain, Roland Dry	2
19	Introducing Data Science: Big Data, Machine Le...	Davy Cielen, Arno D.B. Meysman, Mohamed Ali	3
29	Memories, Dreams, Reflections (Vintage)	C. G. Jung, Aniela Jaffe, Clara Winston, Richa...	4
31	The Archer Files: The Complete Short Stories o...	Ross Macdonald, Tom Nolan	2
71	A Doll's House and Other Plays (Penguin Classics)	Henrik Ibsen, Deborah Dawkin, Erik Skuggevik, ...	4
77	Art: A World History	Elke Linda Buchholz, Susanne Kaeppele, Karolin...	5
80	Batman Eternal Vol. 2 (The New 52) (Batman Ete...	Scott Snyder, Tim Seeley, Jason Fabok	3
89	The Tatas: How a Family Built a Business and a...	Girish Kuber, Vikrant Pande	2

Ratings Analysis:
When we checked the column we saw it should actually be a numerical column but its values are saved in string. So we extract these counts of rating out of each value instead of that string value.

```
0        8
1       14
2        6
3       13
4        1
        ..
6231     2
6232     9
6233     3
6234     4
6235     2
Name: Ratings, Length: 6236, dtype: int64
```

Edition Analysis:

```
0       Paperback,- 10 Mar 2016
1        Paperback,- 7 Nov 2012
2       Paperback,- 25 Feb 1982
3        Paperback,- 5 Oct 2017
4       Hardcover,- 10 Oct 2006
              ...
6231     Paperback,- 8 Aug 2018
6232    Paperback,- 21 Nov 2016
6233     Paperback,- 8 Jun 2006
6234    Paperback,- 15 Jan 2015
6235    Paperback,- 21 Dec 2016
```

If we check edition correctly we see it contains 4 meaningful values:

1. type of print: which could be 'Paperback', 'Hardcover', 'Mass Market Paperback', 'Sheet music', 'Flexibound', 'Plastic Comb', 'Loose Leaf', 'Tankobon Softcover', 'Perfect Paperback', 'Board book', 'Cards', 'Spiral-bound','Product Bundle', 'Library Binding' and 'Leather Bound' values. We know that each of these value can affect the price of the book so we make a column out of it.
2. type of edition: which could have Import', 'Deckle Edge', 'Box set', 'International Edition', 'Unabridged', 'Special Edition', 'Student Edition', 'Illustrated', 'Abridged, 'DVD,NTSC', 'Bargain Price', 'Large Print', 'Audiobook', 'Print', 'Facsimile', 'Box set', 'Facsimile', 'Deluxe Edition', 'Kindle eBook', 'EveryBook' and 'ADPCM' values. We made a separate column for these values named 'Edition_Type' too.

```
NaN                                5451
Import                              614
Illustrated                          46
Unabridged                           18
Special Edition                      18
Student Edition                      13
Box set                              11
International Edition                10
Abridged                              8
Deckle Edge                           7
Large Print                           6
Illustrated,Import                    5
Abridged,Audiobook,Box set            5
Print                                 3
Audiobook                             3
Large Print,Import                    2
Facsimile                             2
Bargain Price                         1
DVD,NTSC                              1
Import,Facsimile                      1
Abridged,Import                       1
Student Edition,Special Edition       1
Audiobook,Unabridged                  1
Abridged,Audiobook,Large Print        1
Deluxe Edition                        1
Kindle eBook                          1
Facsimile,Import                      1
Illustrated,Large Print,Audiobook     1
EveryBook                             1
Illustrated,Large Print               1
ADPCM                                 1
```

3. Month: we extract the month that the book was released. It was not clean so we needed to specify valid names and convert invalid values to ‘NaN’.

```
Oct    639
Sep    543
May    537
Jan    514
Jun    501
Nov    487
Apr    469
Jul    457
Mar    455
Aug    446
Feb    410
Dec    408
       341
ort      9
set      5
ed,      4
rt,      3
int      2
TSC      1
on,      1
ile      1
ion      1
ged      1
ook      1
```

→

```
Oct    639
Sep    543
May    537
Jan    514
Jun    501
Nov    487
Apr    469
Jul    457
Mar    455
Aug    446
Feb    410
Dec    408
NaN    370
```

4. Year: we did the same thing like month for year values and converted all year values to integers.

```
array(['2016', '2012', '1982', '2017', '2006', '2009', '2018', '2015',
       '2013', '1999', '2002', '2011', '1991', '2014', '1989', '2000',
       '2005', '2019', '2008', '2004', '2010', '2007', '2001', '1969',
       '1993', '1992', '2003', '1996', 'port', '1997', '1995', 'NTSC',
       '1987', '1986', '1990', '1988', '1981', '1976', '1994', '1998',
       '1977', '1974', '1983', '1971', '1985', '1978', 'mile', ' set',
       'tion', '1964', '1984', '1980', 'dged', '1979', 'rint', '1960',
       '1970', '1975', '1905', '1900', 'book', '1961', '1925', '1973'],
      dtype=object)
```

Reviews Analysis:

Review was the same as the ratings column. It was a numerical column but it was saved as categorical. So I extract numbers in each row and put it as an actual value of review.

```
0       4.0
1       3.9
2       4.8
3       4.1
4       5.0
       ...
6231    5.0
6232    3.3
6233    3.8
6234    3.5
6235    3.9
```

Title-Synopsis Analysis:

First we make a specific column named ‘titles_synopses’ to have both title and summary of the book together. Title and summary are the most identifying information that we have. We decided to translate this information to topics to extract features out of it. We extract 24 topics (24 columns) that each value in the ‘titles_synopses’ column could have from 0 to 100 percent.
The preprocessing workflow addresses the 'Title' and 'Synopsis' columns, containing textual information from various books. A challenge arises due to non-English text, originating from an Indian website. To ensure linguistic consistency, non-English content is translated to English using the Googletrans library. The decision to merge 'Title' and 'Synopsis' aims for a comprehensive analysis. The text undergoes systematic preprocessing, including HTML code removal, contractions expansion, punctuation removal, stop words elimination, and lemmatization. Each step refines and standardizes the text, aligning it with the requirements of subsequent analytical or modeling tasks, especially in natural language processing.

	Topic 0	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9	...	Topi
0	0.005425	0.005425	0.005425	0.005425	0.005425	0.019888	0.005425	0.005425	0.005425	0.005425	...	0.02
1	0.005153	0.005153	0.005153	0.005153	0.005153	0.005153	0.005153	0.005153	0.005153	0.005153	...	0.00
2	0.003773	0.003773	0.003773	0.003773	0.003773	0.003773	0.003773	0.003773	0.003773	0.039609	...	0.20
3	0.006727	0.006727	0.006727	0.006727	0.006727	0.006727	0.006727	0.006727	0.006727	0.006727	...	0.11
4	0.005459	0.005459	0.005459	0.005459	0.064523	0.005459	0.005459	0.005459	0.005459	0.005459	...	0.05
...	...	...	...	...	...	...	...	...	...	...	...	
5694	0.003914	0.003914	0.003914	0.003914	0.015534	0.089028	0.003914	0.003914	0.003914	0.003914	...	0.00
5695	0.004571	0.004571	0.119704	0.004571	0.094247	0.004571	0.004571	0.004571	0.004571	0.004571	...	0.00
5696	0.005314	0.005314	0.005314	0.005314	0.182416	0.005314	0.005314	0.005314	0.005314	0.005314	...	0.00
5697	0.006383	0.006383	0.006383	0.006383	0.006383	0.006383	0.006383	0.006383	0.006383	0.006383	...	0.00
5698	0.005417	0.005417	0.005417	0.005417	0.005417	0.005417	0.005417	0.005417	0.005417	0.005417	...	0.00

Popularity column:

We decided to make a new column named popularity. It is actually a weighted value of ratings. We multiply ratings and reviews to get this column.

Season Column:

We made another column named season. We decide to have the effect of season as a feature because we know it has a huge impact on pisces in markets. But the month itself did not show this impact.

5. Filling null values

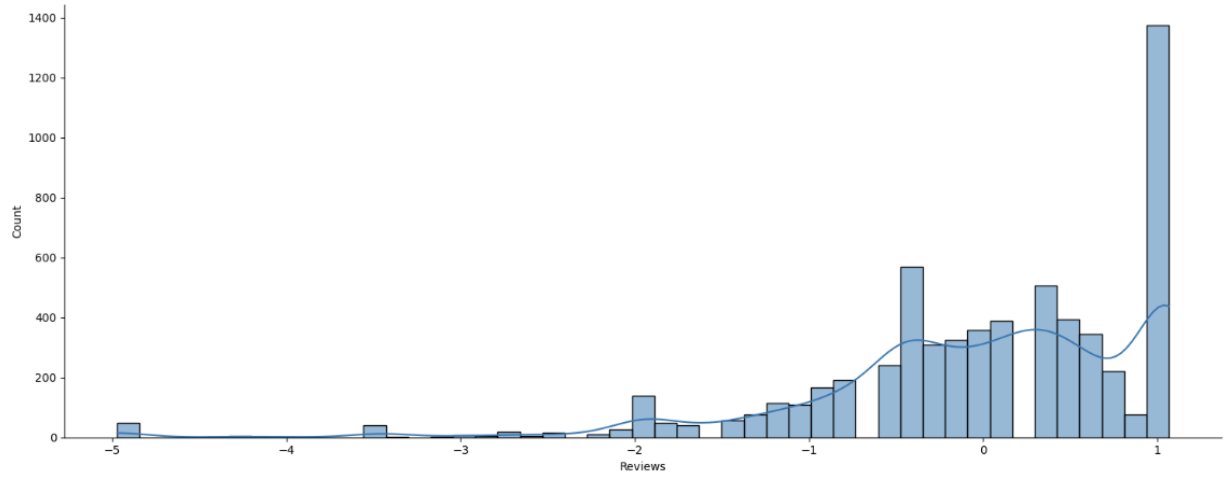
We decide to fill null values with the mode value of each column. The mode represents the most frequently occurring value within a given column, and selecting it as the filling criterion is based on the premise that it signifies the most probable value for that particular column. This decision is rooted in the idea that by substituting null values with the mode, we are infusing the dataset with the most common and likely values, contributing to the overall completeness and reliability of the data.

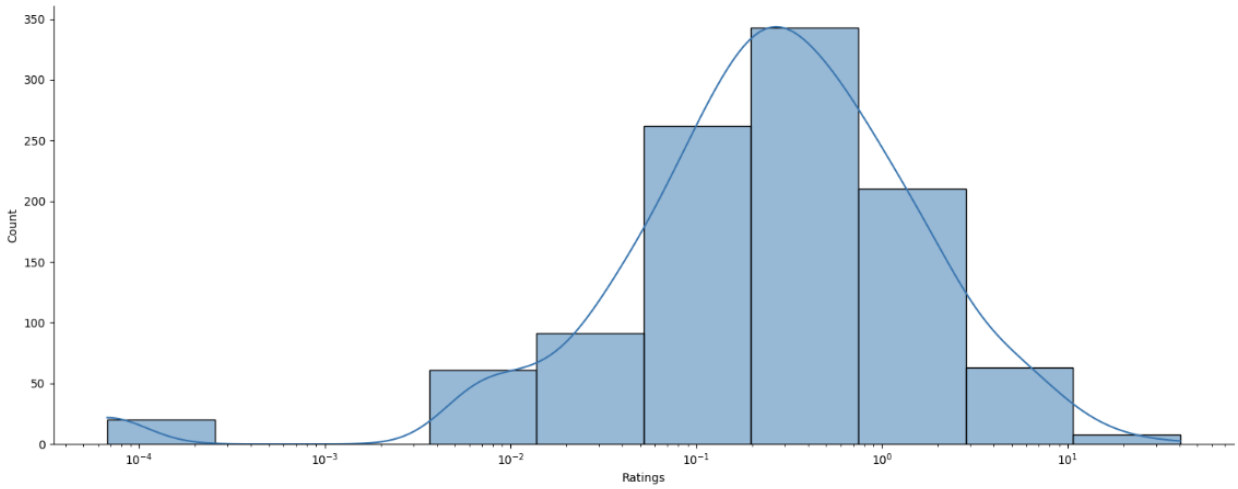
6. Normalization

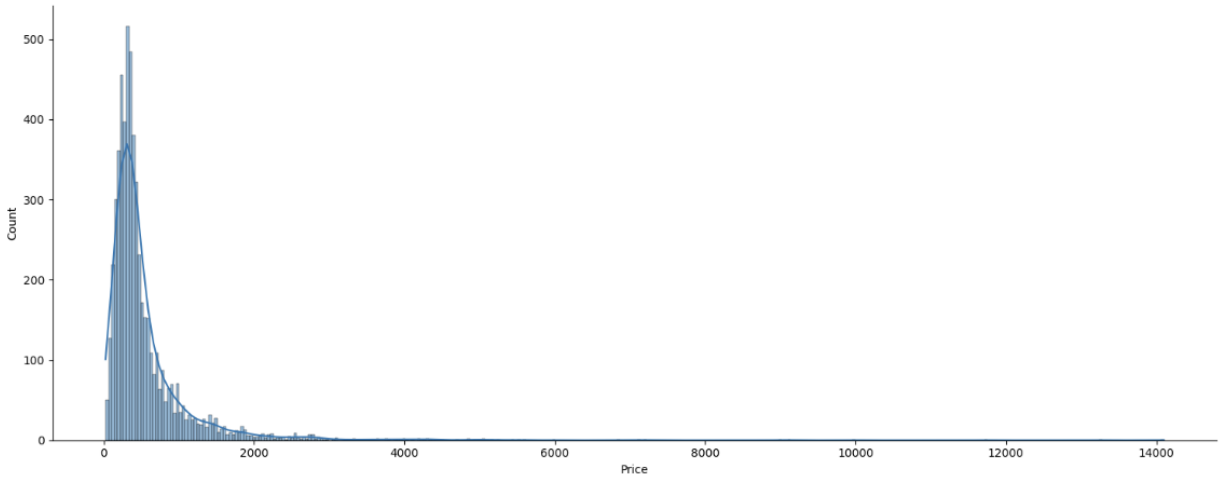
Before we start looking at the graphs, we have to make sure that some numbers, like 'Reviews,' 'Ratings,' and 'Popularity,' are on the same scale. We do this to compare them fairly and avoid any confusion because they might have different scales. This way, the graphs will give us a clearer and more accurate picture of what's going on with these numbers. It just helps us see things better.

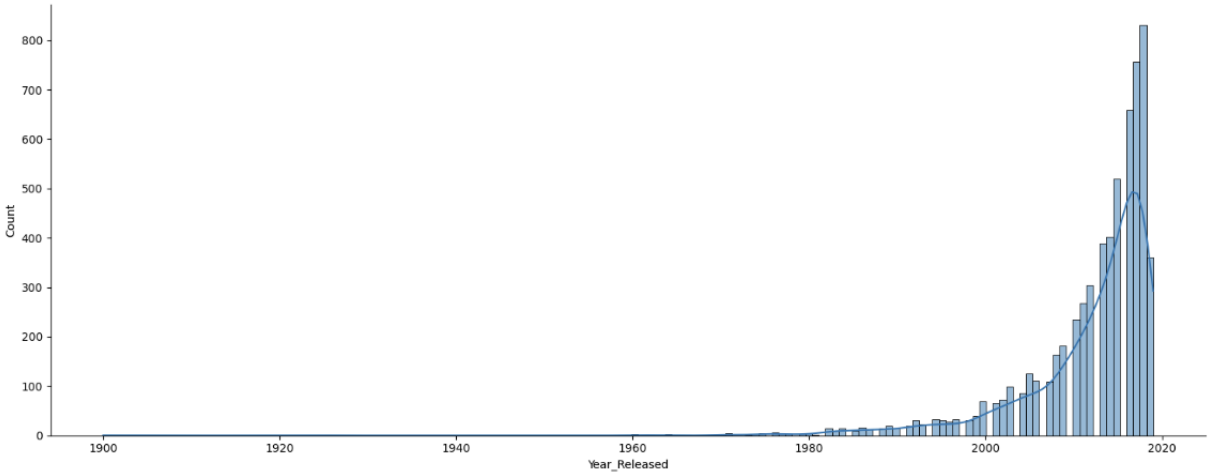
7. Visualization

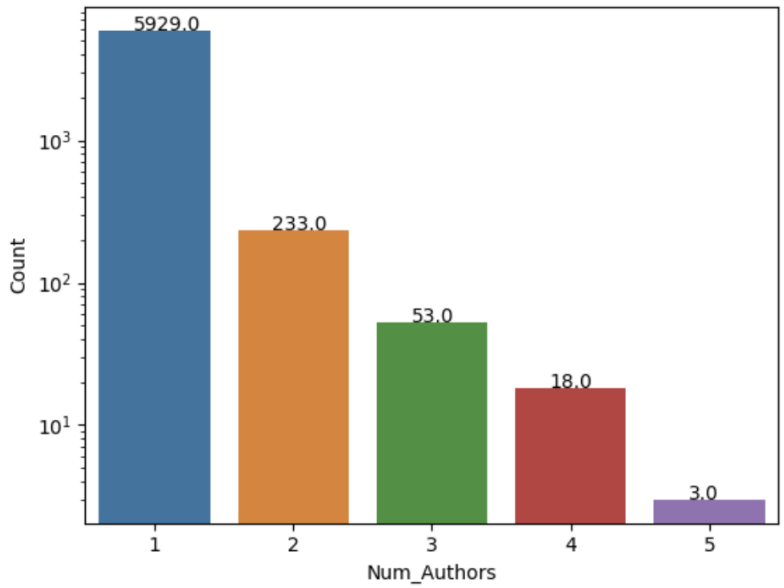
Barplots for numerical features:



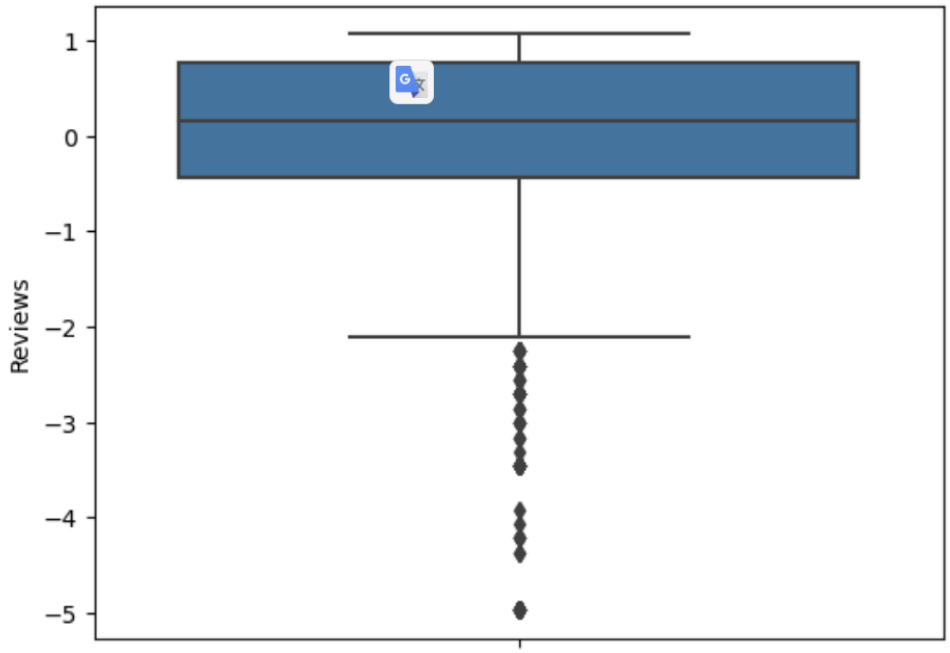


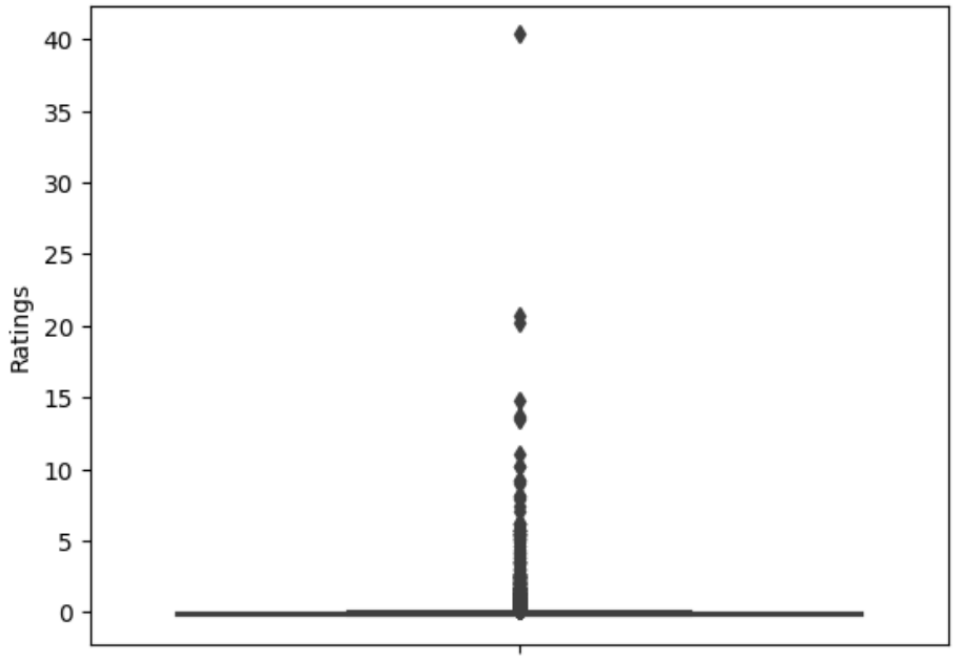


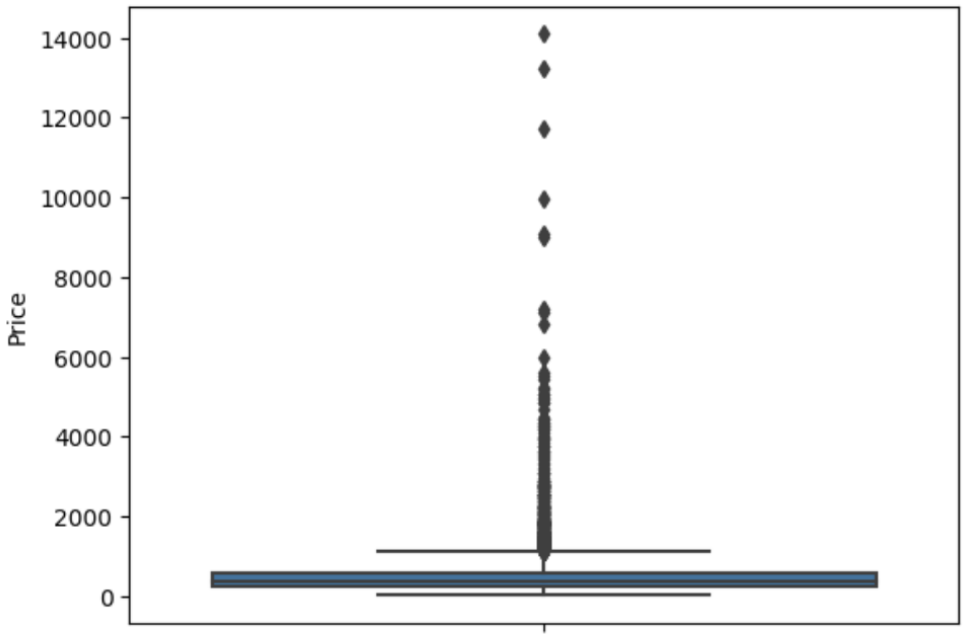

Count
Year_Released
0
100
200
300
400
500
600
700
800
1900
1920
1940
1960
1980
2000
2020

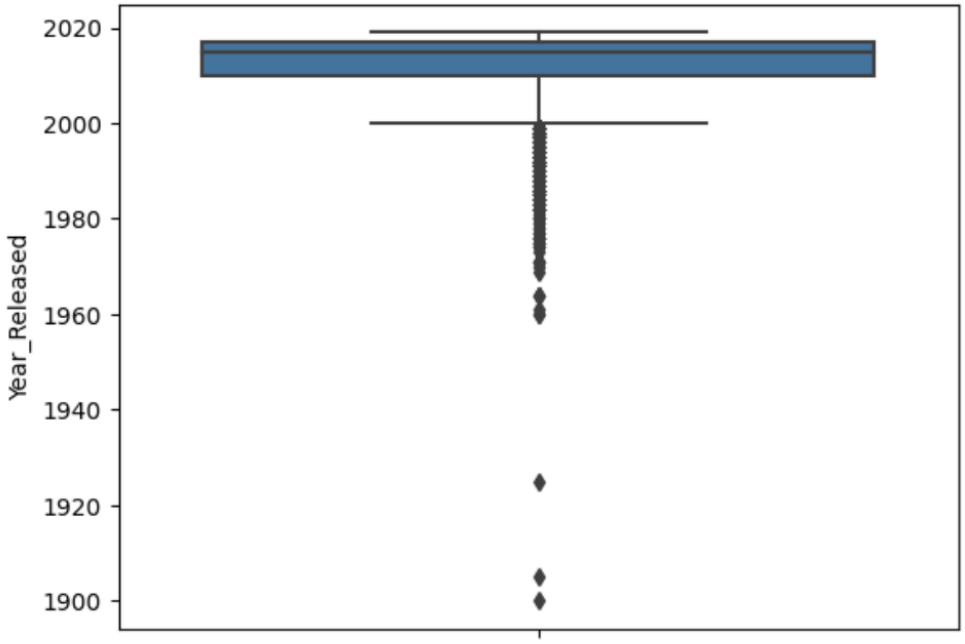

5929.0
233.0
53.0
18.0
3.0
Count
Num_Authors
1
2
3
4
5

Boxplot for numerical features:

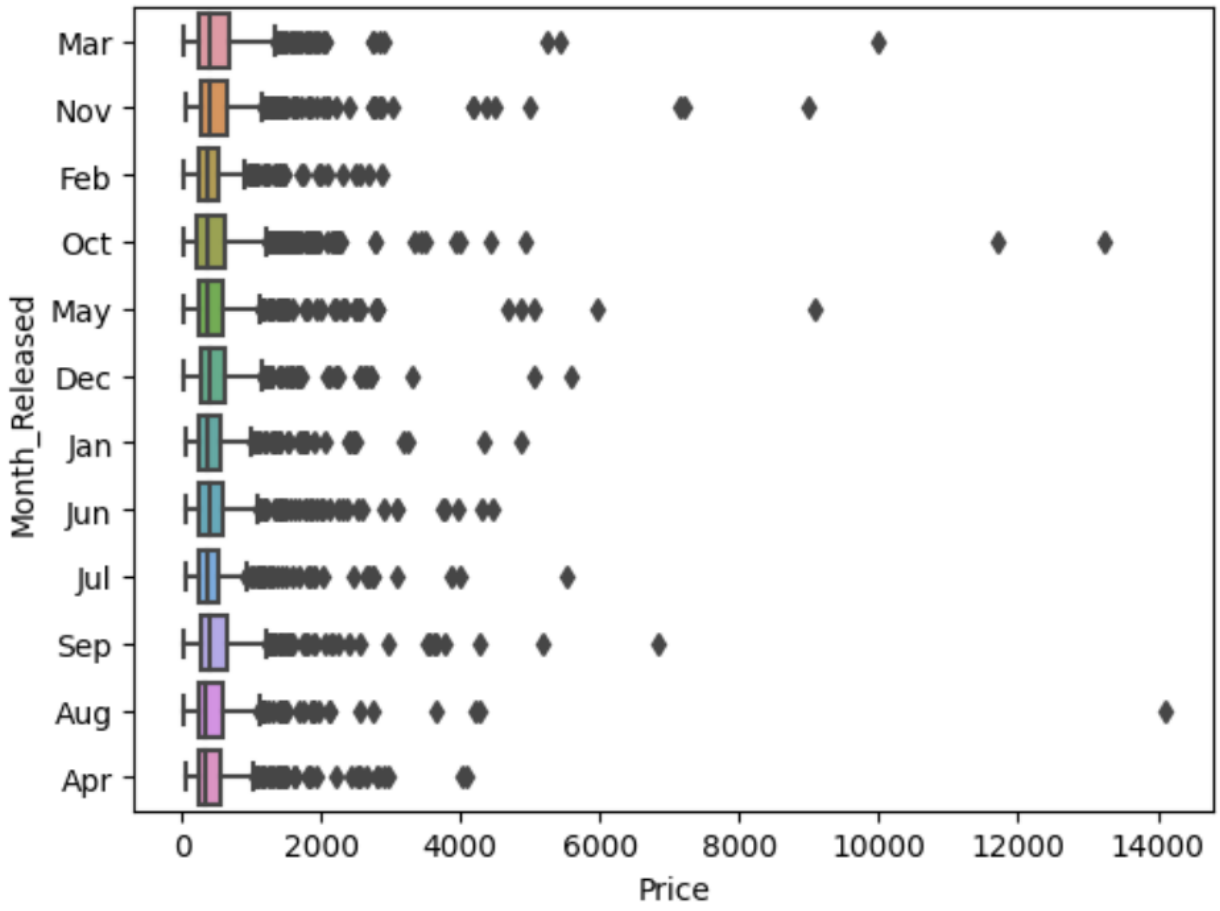




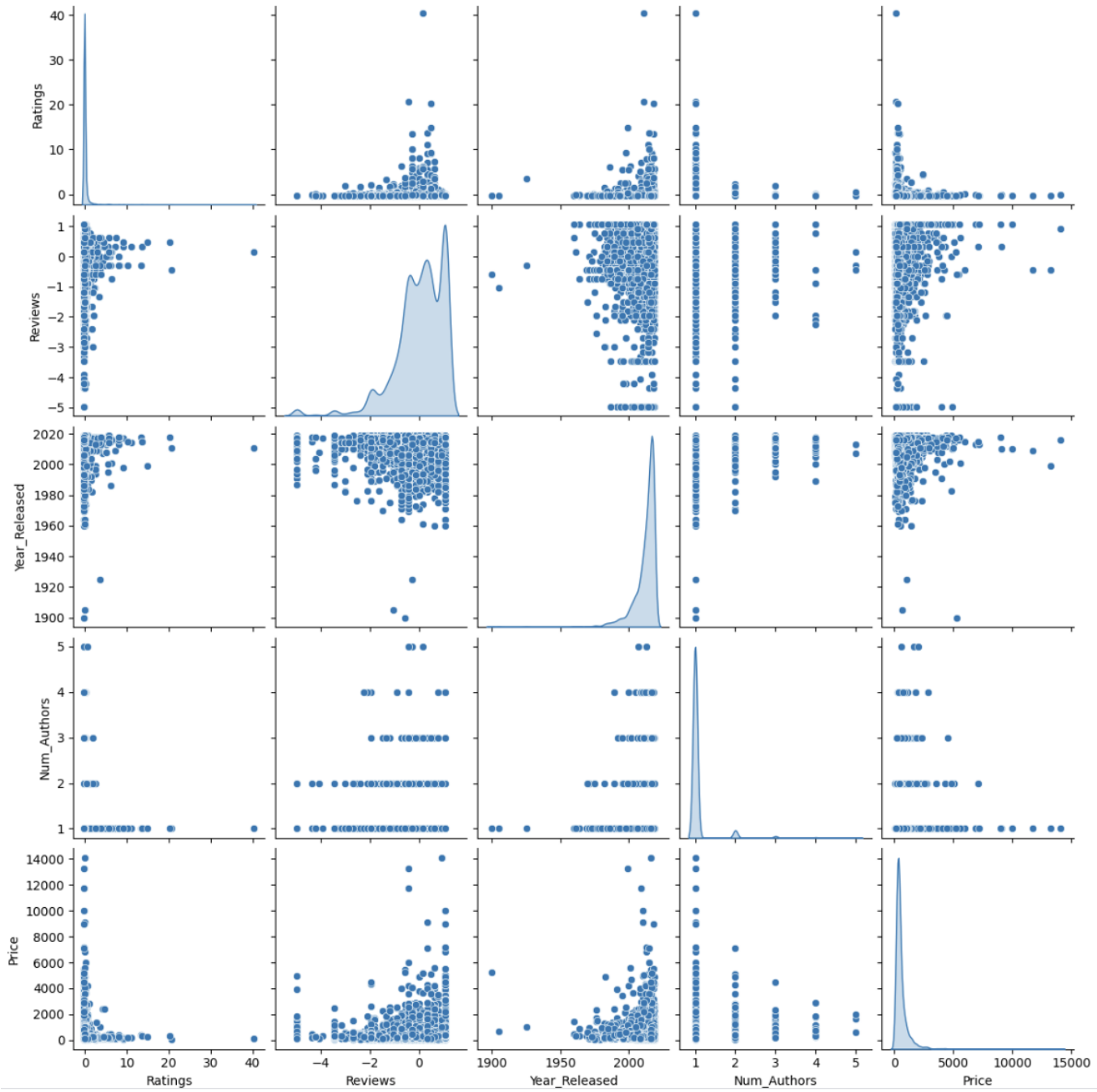

Price
14000
12000
10000
8000
6000
4000
2000
0


Year_Released
2020
2000
1980
1960
1940
1920
1900

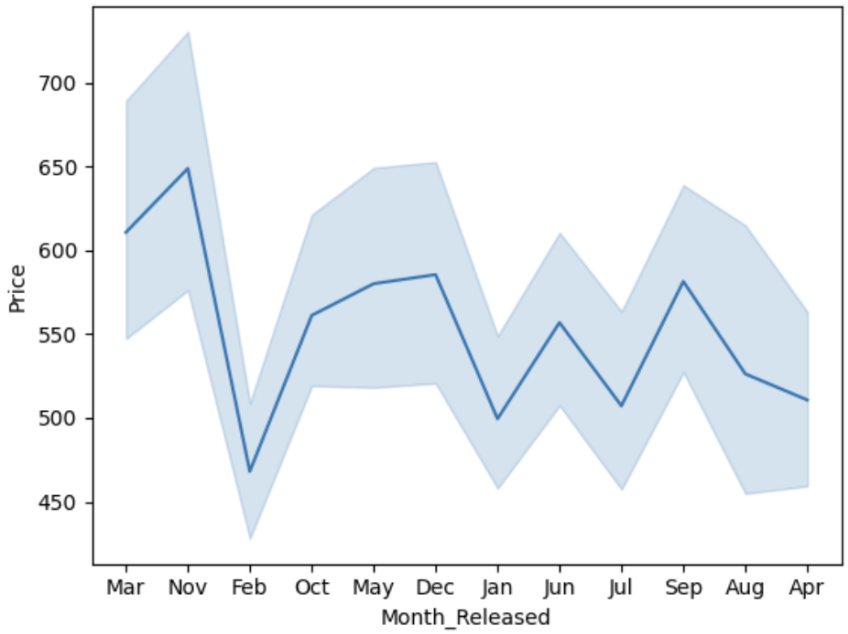
Price and month:



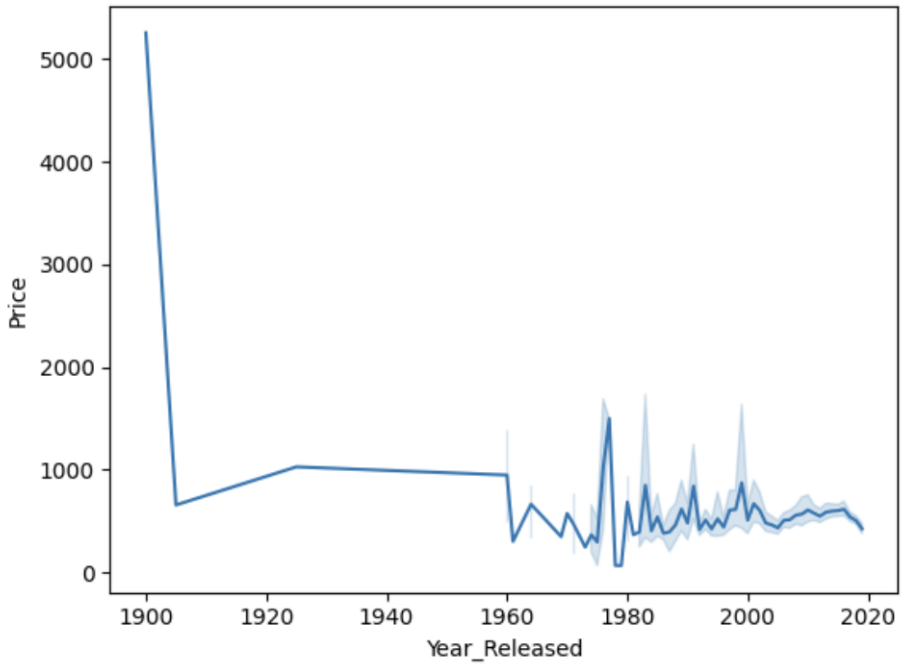
Ratings , Reviews, Year, Number of Authors & Price:



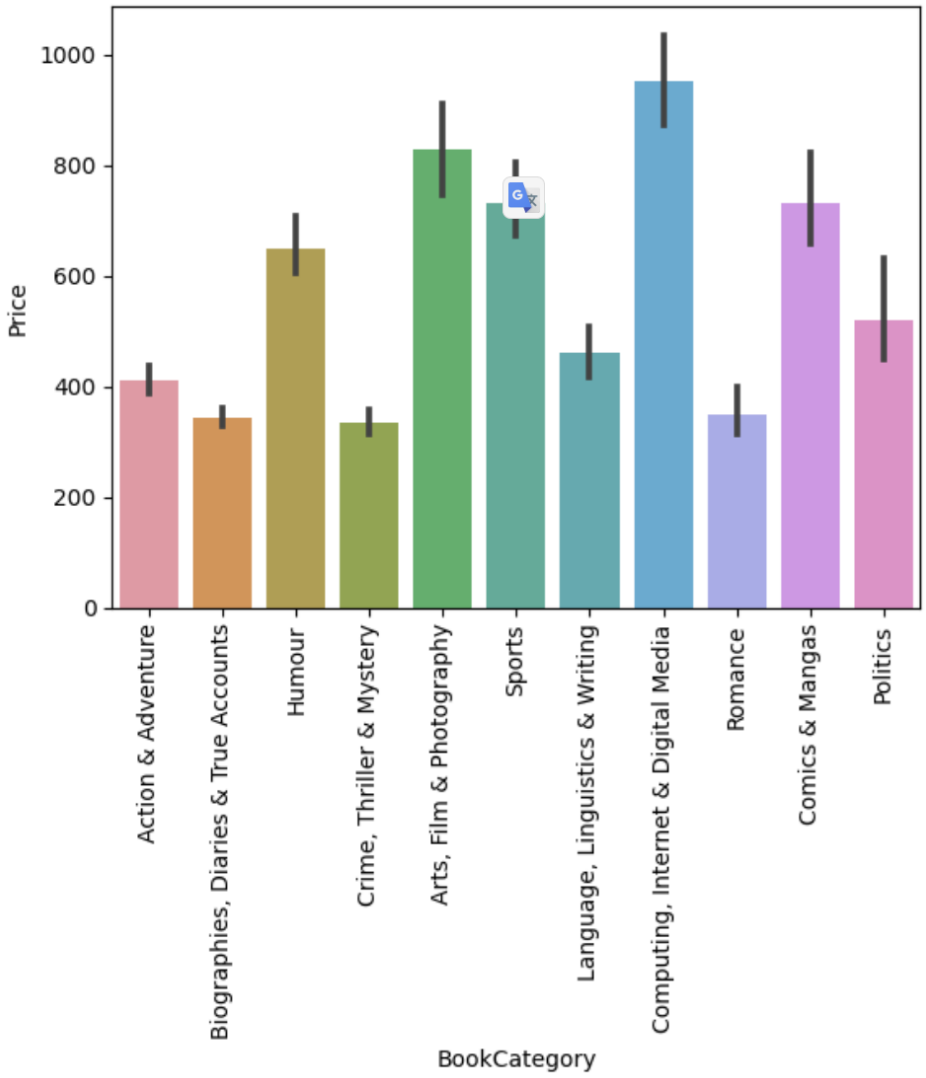
Month & Price:

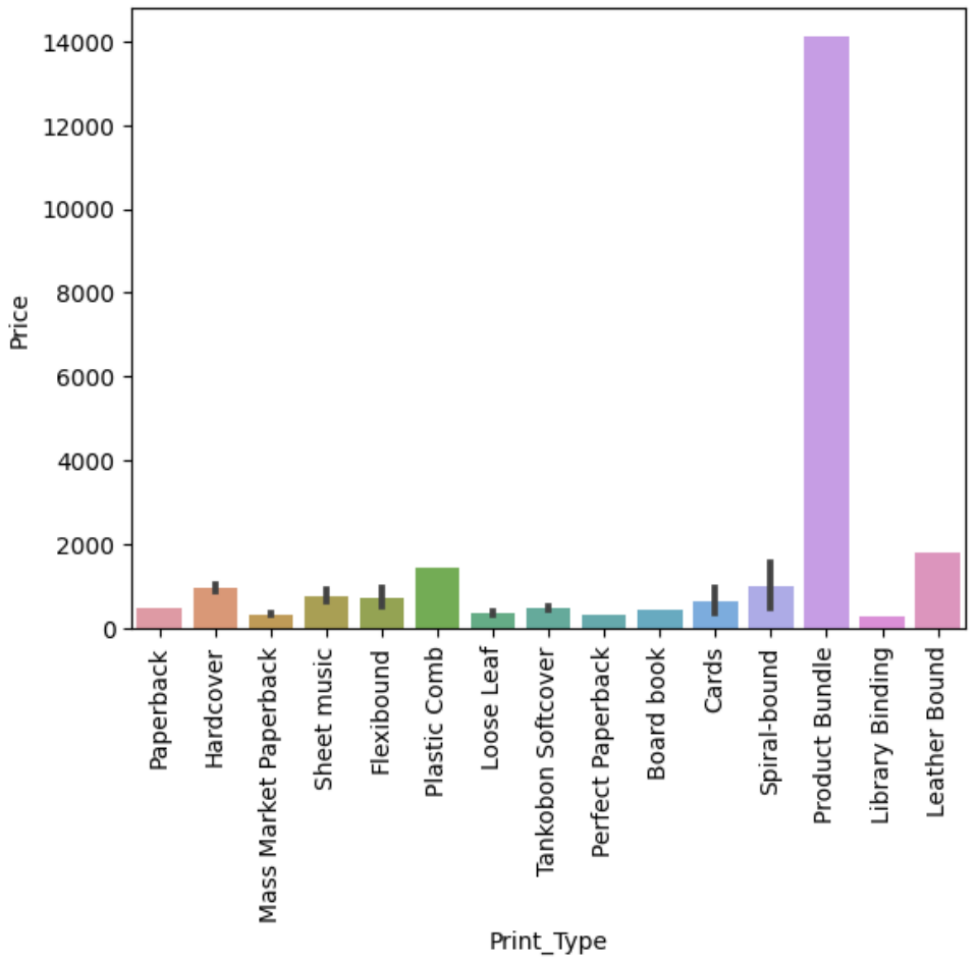


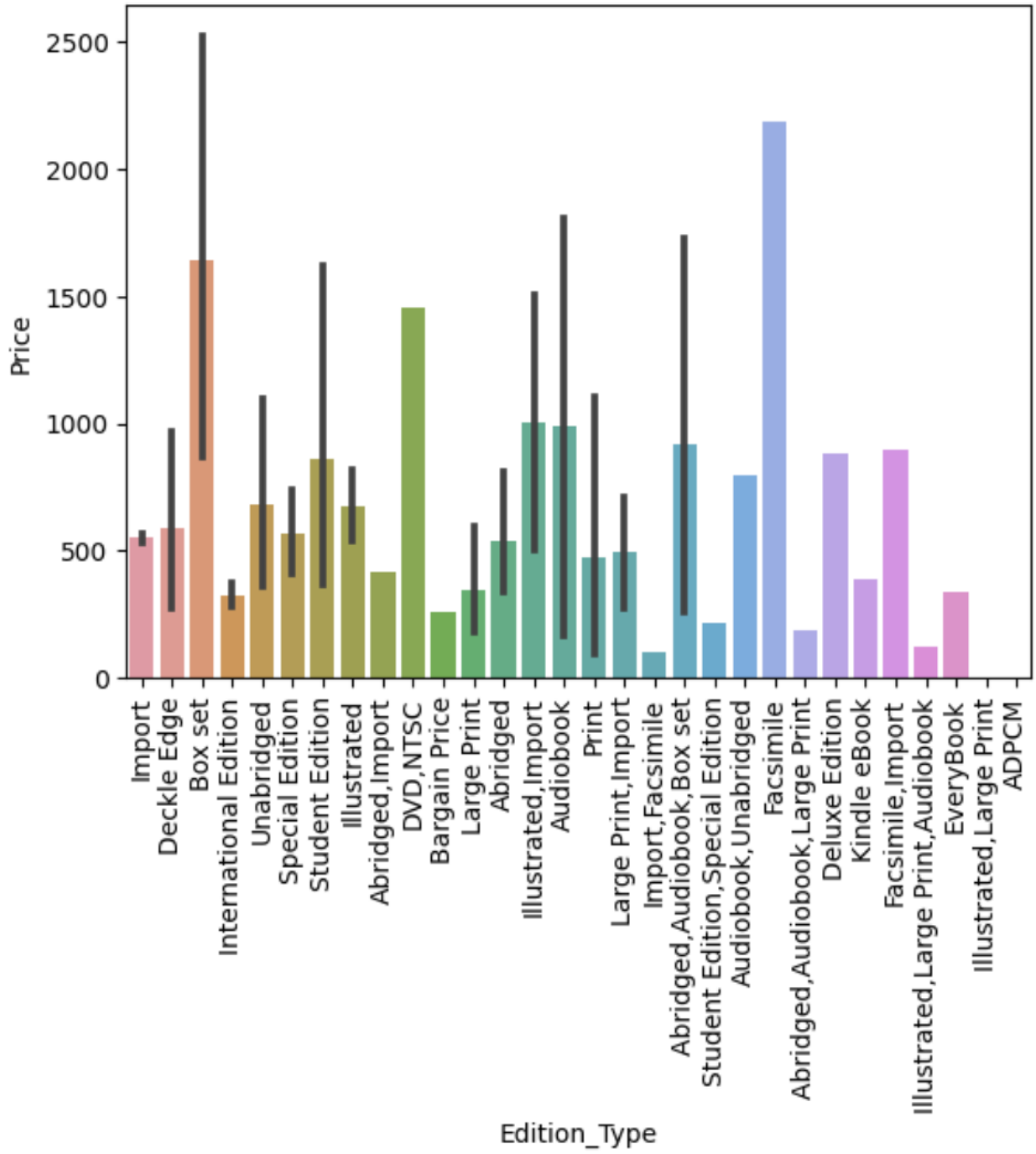
Year & Price:

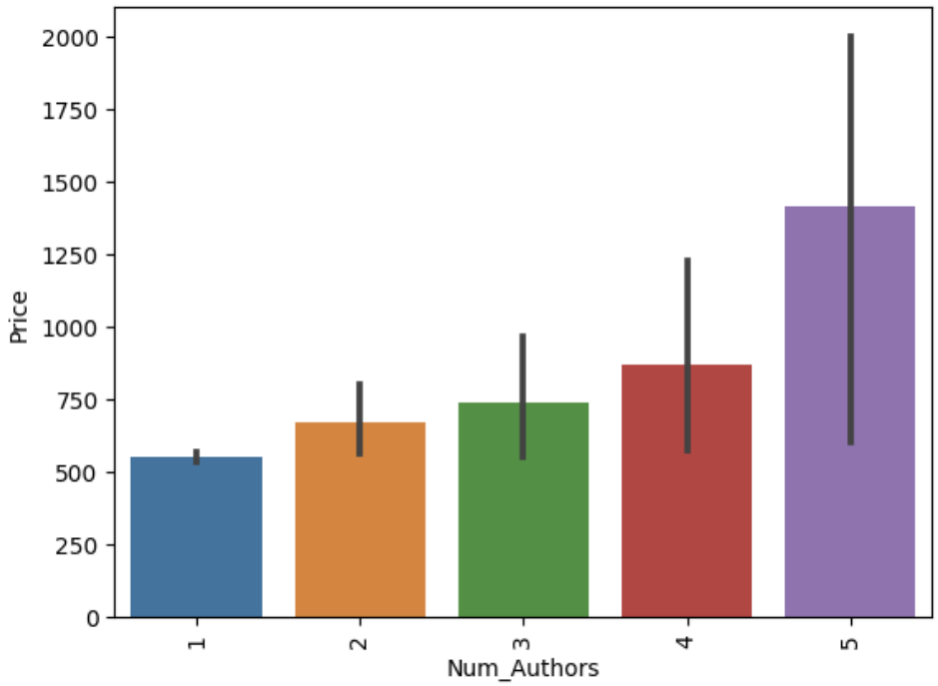


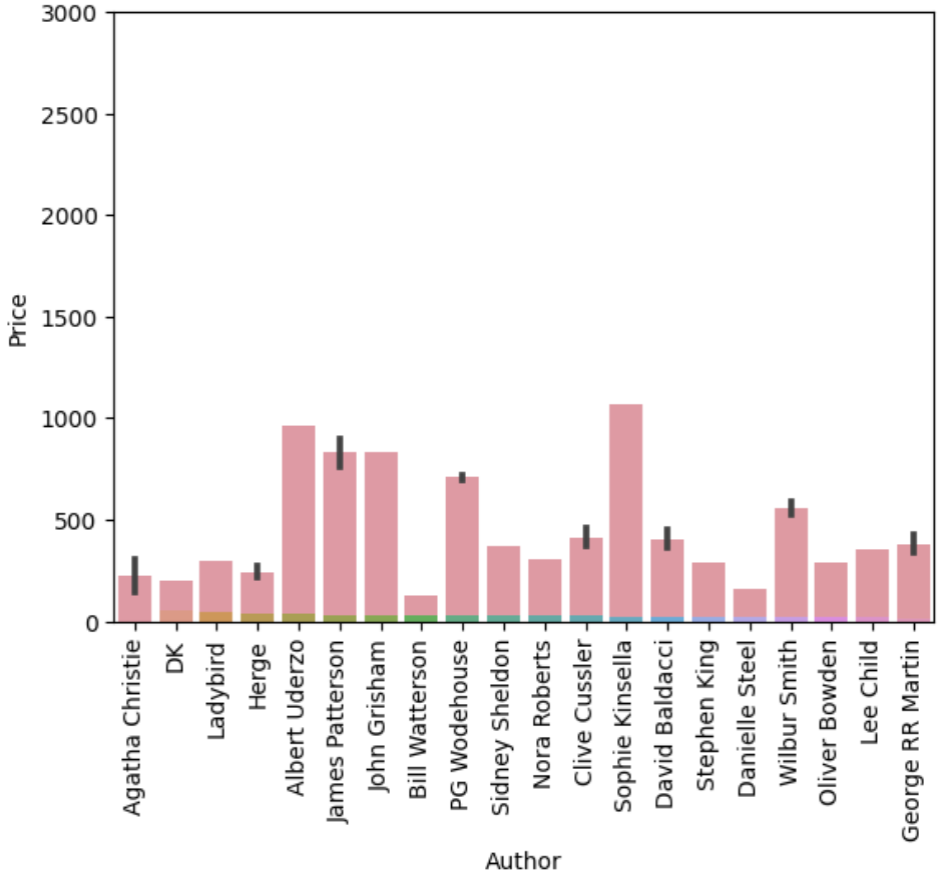
Barplots for Price and Categorical Features:



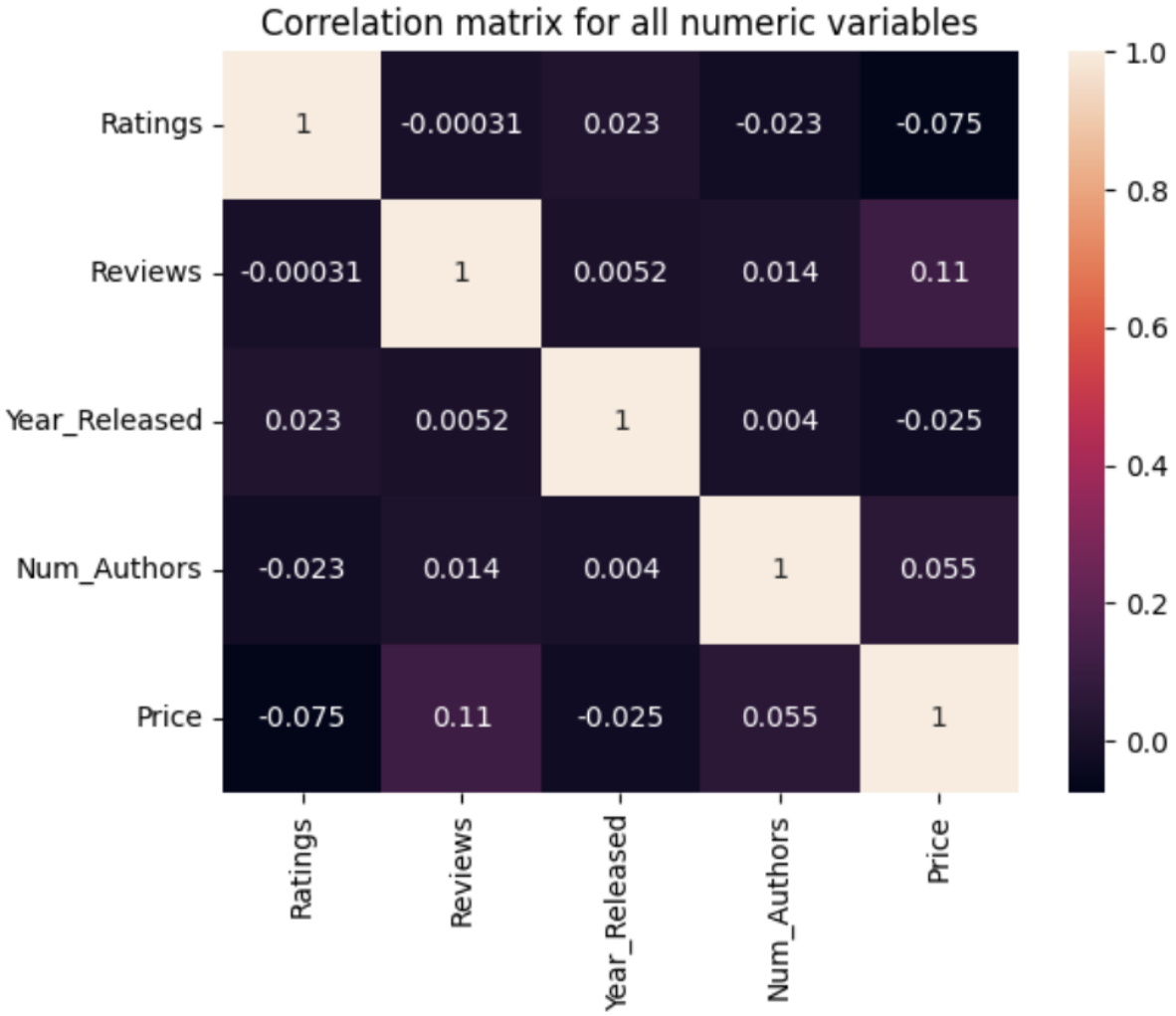

14000
12000
10000
8000
6000
4000
2000
0
Price
Paperback
Hardcover
Mass Market Paperback
Sheet music
Flexibound
Plastic Comb
Loose Leaf
Tankobon Softcover
Perfect Paperback
Board book
Cards
Spiral-bound
Product Bundle
Library Binding
Leather Bound
Print_Type


Price
2500
2000
1500
1000
500
0
Import
Deckle Edge
Box set
International Edition
Unabridged
Special Edition
Student Edition
Illustrated
Abridged,Import
DVD,NTSC
Bargain Price
Large Print
Abridged
Illustrated,Import
Audiobook
Print
Large Print,Import
Import,Facsimile
Abridged,Audiobook,Box set
Student Edition,Special Edition
Audiobook,Unabridged
Facsimile
Abridged,Audiobook,Large Print
Deluxe Edition
Kindle eBook
Facsimile,Import
Illustrated,Large Print,Audiobook
EveryBook
Illustrated,Large Print
ADPCM
Edition_Type


Price
2000
1750
1500
1250
1000
750
500
250
0
1
2
3
4
5
Num_Authors


Price
3000
2500
2000
1500
1000
500
0
Agatha Christie
DK
Ladybird
Herge
Albert Uderzo
James Patterson
John Grisham
Bill Watterson
PG Wodehouse
Sidney Sheldon
Nora Roberts
Clive Cussler
Sophie Kinsella
David Baldacci
Stephen King
Danielle Steel
Wilbur Smith
Oliver Bowden
Lee Child
George RR Martin
Author

Correlation Matrix:



8. Encoding

We chose to use a simple method called one-hot encoding to deal with categorical columns. This approach is effective and easy to use when working with categorical data. With one-hot encoding, we turn categorical variables into binary vectors, making a new binary column for each unique category. This way, we ensure that the categorical information is presented in a way that works well for machine learning algorithms. We decided on one-hot encoding because it's easy to use, and it allows us to capture the categorical differences without making things too complicated.

9. Training

We used the training set to train the RandomForestRegressor, and then we used the trained model to make predictions on the specified test set. To measure how well the model predicts, we calculated the Mean Squared Error (MSE) values for both the training and test sets.

```
Train mse is: 55217.80914648059 // Test mse is: 202404.8534612078
```

10. Post processing

R-Squered Value: This will give us an idea of how well ourregression model is performing on both the training and test sets. If R-squared is close to 1, it indicates a good fit, and if it's close to 0, the model is not explaining much of the variance.

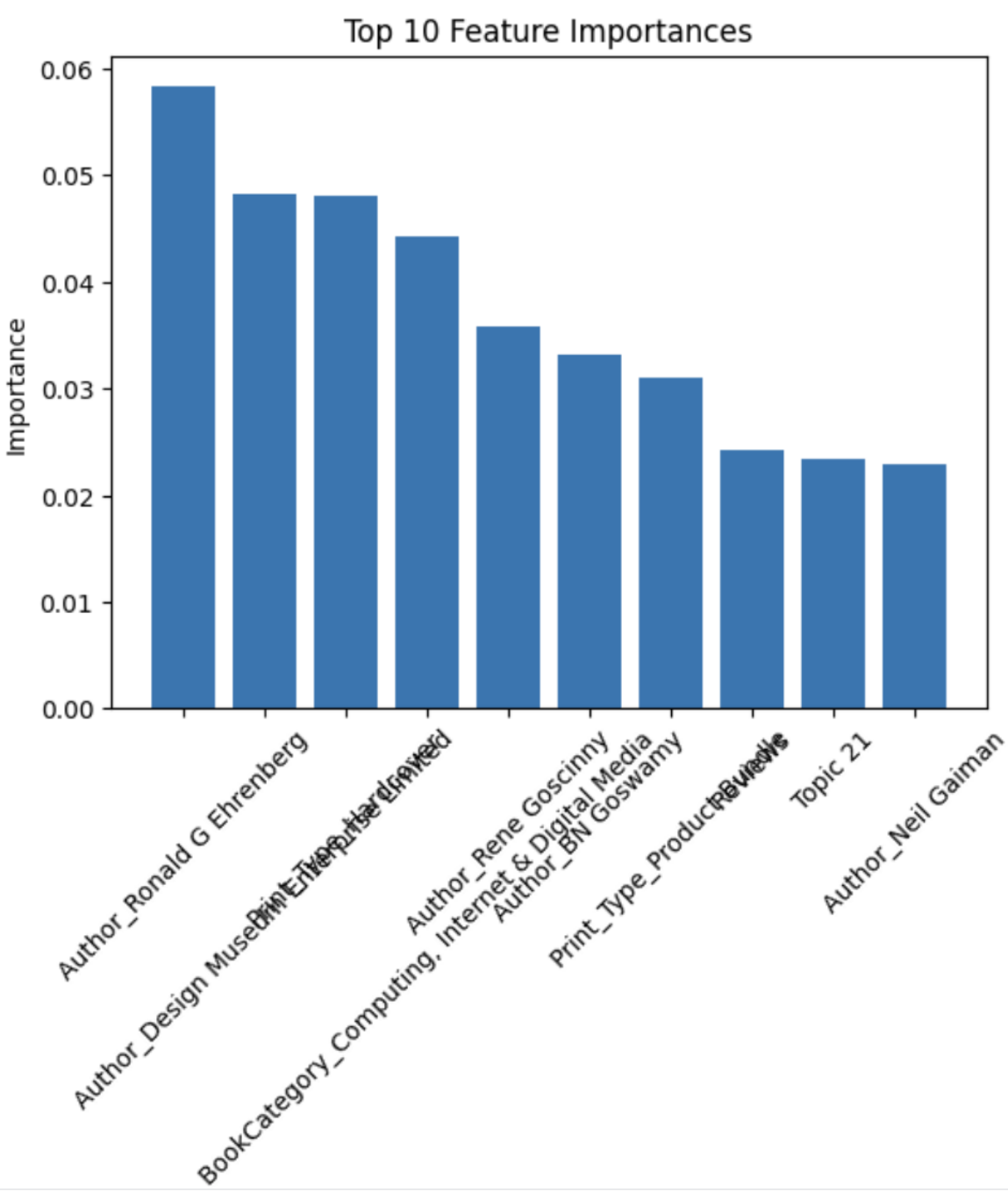
```
R-squared for training set: 0.8838
R-squared for test set: 0.2433
```

Feature Importance: Random Forest models provide feature importances. Analyze these importances to understand which features have the most influence on predictions. This can help us identify whether certain features are causing errors or need further investigation.

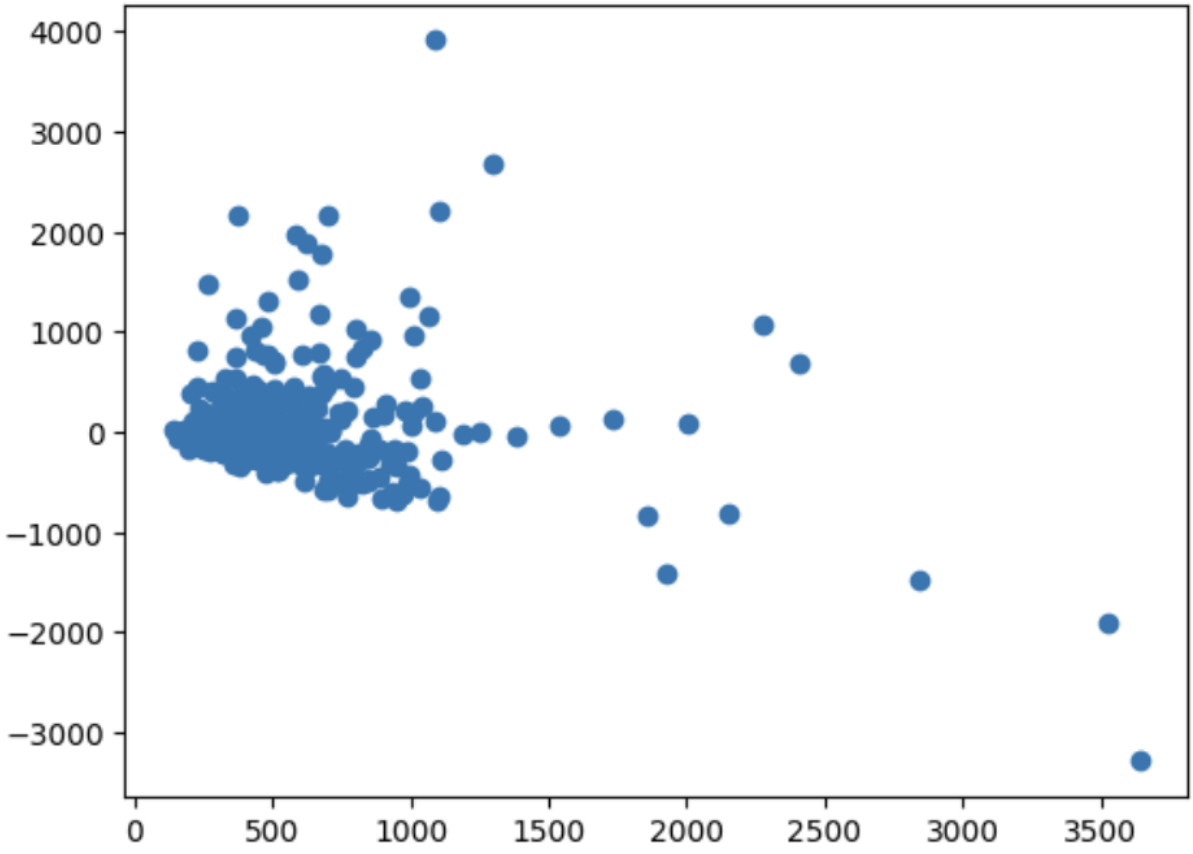
```
array([0.02416742, 0.0156163 , 0.00175423, ..., 0.00111288, 0.00069808,
       0.00112668])
```

And 10 of the most important features are:

```
Index(['Author_Ronald G Ehrenberg', 'Author_Design Museum Enterprise Limited',
       'Print_Type_Hardcover',
       'BookCategory_Computing, Internet & Digital Media',
       'Author_Rene Goscinny', 'Author_BN Goswamy',
       'Print_Type_Product Bundle', 'Reviews', 'Topic 21',
       'Author_Neil Gaiman'],
      dtype='object')
```


Top 10 Feature Importances
Importance
0.06
0.05
0.04
0.03
0.02
0.01
0.00
Author_Ronald G Ehrenberg
Author_Design Museum
Author_Rene Goscinny
Author_BN Goswamy
Topic 21
Author_Neil Gaiman

Residual Analysis:



11. Conclusion

This assignment on Feature Engineering helped us understand important machine learning techniques. We learned how to turn raw data into useful features, choose the right ones, and understand how they affect the model. The assignment focused on data preprocessing, including cleaning, handling missing values, and dealing with outliers. It stressed the importance of scaling, normalization, and standardization to make the model results easier to understand. We also saw how to handle categorical variables using methods like one-hot encoding, making our approach more flexible. We explored creating new features, like polynomial features and domain-specific engineering, to make the dataset more complex. Additionally, we looked into extracting features from text, aiming to learn valuable insights from different data sources. The assignment encourages using various methods in data preprocessing without limitations. While the implementation allows a specified model, the evaluation prioritizes the effectiveness of the preprocessing pipeline over model accuracy, showing the assignment's focus on strong feature engineering practices.