Contents

Abstract

In this report we examine machine learning feature engineering techniques, with a focus on selecting pertinent features to improve model performance and interpretability and manipulating raw data to extract meaningful features. We demonstrate the usefulness of these techniques in the real-world setting of book pricing. Researchers and practitioners working in the larger fields of data science and machine learning are expected to gain from the new insights.

Introduction

In this report, we explore the field of feature engineering in machine learning. Raw data is manipulated and transformed into useful and instructive features through the process of feature engineering. Our main goal is to learn about the nuances of different feature engineering approaches and investigate ways to choose the most pertinent features. The ultimate goal is to improve interpretability and model performance, which are crucial in the dynamic field of data science. In particular, this report applies these ideas to the book pricing domain, emphasising the role that feature engineering plays in optimising predictive modelling for this use case.

Data Preprocessing: We begin with data preprocessing, which is an essential first step in any machine learning project. In this section, we concentrate on getting the raw data ready for feature engineering. This covers activities like dealing with outliers, handling missing values, and guaranteeing data consistency. We prepare the groundwork for efficient feature extraction and manipulation by carefully cleaning and organising the dataset, which also sets the stage for the other phases of our analysis.

Feature Creation: A key component of our investigation is feature creation, where we examine the process of turning unintelligent data into useful features. This calls for the use of a number of methods, including the construction of composite features, interaction terms, and polynomial expansion. The aim is to incorporate dimensions that more effectively represent the fundamental patterns and connections found in the data. In this section, we will focus on particular techniques used in the book pricing context, demonstrating how feature creation can revolutionise the way machine learning models are predictive.

Implementation: The implementation phase marks the culmination of our journey, where the theoretical understanding of feature engineering translates into practical application. We deploy machine learning models on our feature-engineered dataset to predict book prices. The choice of algorithms, hyperparameter tuning, and model evaluation are integral components of this stage. Through this implementation, we aim to demonstrate the tangible impact of effective feature engineering on model accuracy and interpretability in the specific context of predicting book prices. The results obtained in this phase will not only validate the efficacy of our approach but also provide insights into the broader applicability of feature engineering in diverse machine learning scenarios.

1 Data Information

There are 9 columns and 5699 entries in the dataset. Every row represents a distinct book record, and the columns show various characteristics connected to every book. An overview of the features and organisation of the dataset can be found below.

- **Data Shape:**
  - Number of Rows: 5699
  - Number of Columns: 9
- **Data Columns:**
  1. **Title**
  2. **Author**
  3. **Edition**
  4. **Reviews**
  5. **Ratings**
  6. **Synopsis**
  7. **Genre**
  8. **BookCategory**
  9. **Price**
- **Number of Unique Values in Each Column:**
  1. **Title:** 5130 unique titles
  2. **Author:** 3438 unique authors
  3. **Edition:** 3183 unique editions
  4. **Reviews:** 36 unique review categories
  5. **Ratings:** 333 unique rating values
  6. **Synopsis:** 5114 unique synopses
  7. **Genre:** 335 unique genres
  8. **BookCategory:** 11 unique book categories
  9. **Price:** 1538 unique price values
- **Missing Values:**
  - No missing values are present in any of the columns. The dataset is complete with non-null values for all entries.
- **Data Types:**
  - The dataset consists of a combination of data types:
    * 1 column with a float64 data type (**Price**)
    * 8 columns with object data types representing various categorical and text features.

2 Data Preprocessing

2.1 Ratings and Reviews Conversion

the 'Ratings' column's numerical portion was taken out and converted to integers. The numerical portion was taken out of the 'Reviews' column and converted to floating-point values.

2.2 Edition Date and Type Extraction

The 'Edition' column was used to extract the date and edition type. 'Edition_Type' for the edition type and 'Edition_Date' for the extracted date were created as new columns. I eliminated all unnecessary characters from the 'Edition_Type' column, including ",–". Date extracted was converted to datetime format.

2.3 Outlier Removal

implemented a function that uses the Interquartile Range (IQR) method to eliminate outliers. The 'Ratings' and 'Price' columns have the outlier removal function applied to them. To make later analyses more robust, outliers were found and eliminated.

2.4 Book Category Encoding

The 'BookCategory' column's categories were divided, and new binary columns were made for each category. In this step, the categorical data was expanded into a format that machine learning models could use more easily.

All of these preprocessing steps were taken with the intention of improving the quality of the dataset so that feature engineering and machine learning model training could be done on it more easily. The actions taken take care of things like converting data types, extracting pertinent information, managing outliers, and encoding categorical features to improve model compatibility.

3 Data Analysis

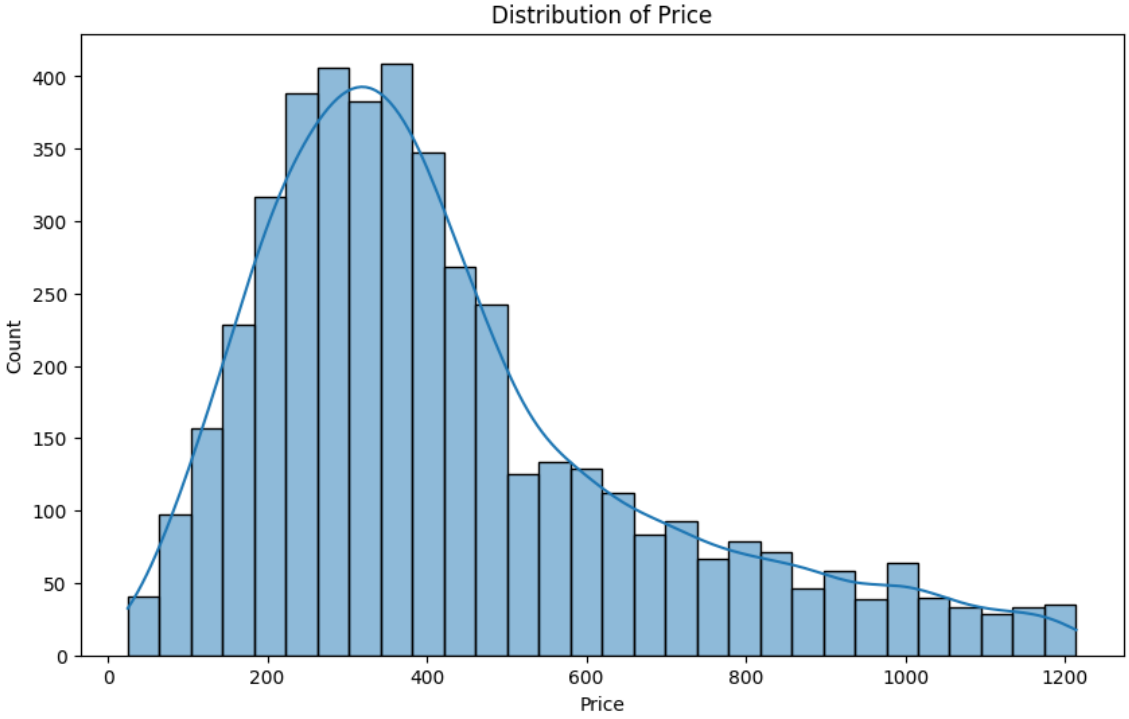
3.1 Univariate Variable Analysis

3.1.1 Distribution of Price

- **Figure Size:** 10 inches in width and 6 inches in height.
- **X-axis:** Prices (`Price`) were represented on the x-axis.
- **Y-axis:** Frequency of prices was represented on the y-axis.
- **Bins:** 30 bins were used for the histogram.
- **Title:** 'Distribution of Price' was assigned to the plot.
- **X-axis Label:** 'Price' was assigned to the x-axis.

Key observations from the histogram include:

- The distribution of book prices shows a wide range, spanning from a little over $50 to $1200.
- The peak of the curve is observed in the range of $250 to $400.
- As prices deviate from this range, the frequency decreases, indicating a tapering distribution.



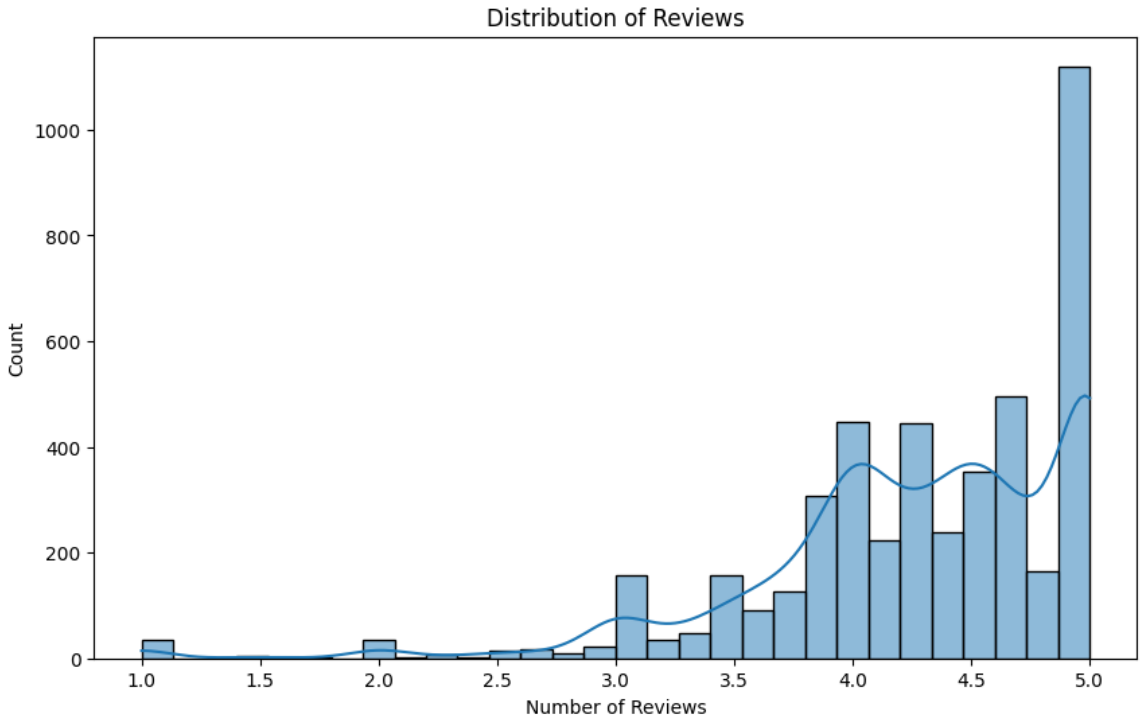
3.1.2 Distribution of Reviews

To see how the distribution of book reviews varied, a kernel density estimation (KDE) histogram was created. Seaborn was used to create the plot and had the following features:

- **Figure Size:** 10 inches in width and 6 inches in height.
- **X-axis:** Number of reviews (`Reviews`) were represented on the x-axis.
- **Y-axis:** Frequency of reviews was represented on the y-axis.
- **Bins:** 30 bins were used for the histogram.
- **Title:** 'Distribution of Reviews' was assigned to the plot.
- **X-axis Label:** 'Number of Reviews' was assigned to the x-axis.

Key observations from the histogram include:

- The distribution of the number of reviews shows an increasing trend around 5.0.
- As the number of reviews tends towards 5.0, the frequency of books with that number of reviews increases.



3.1.3 Distribution of Ratings

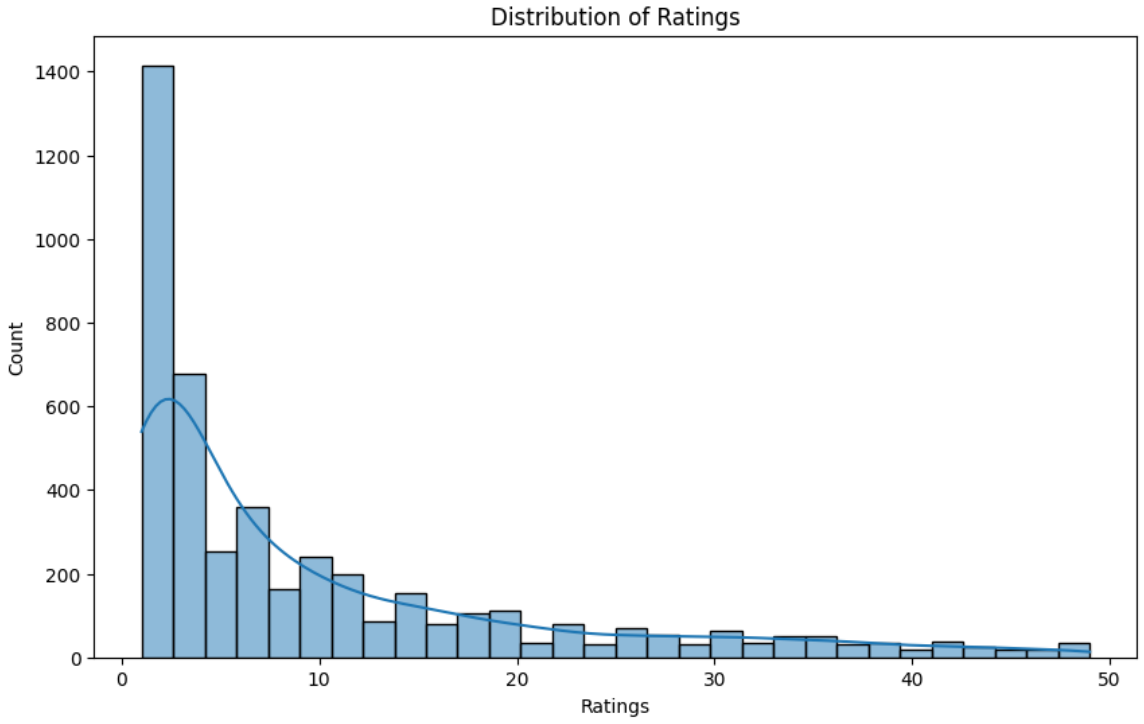
A histogram with kernel density estimation (KDE) was created to visualize the distribution of ratings for books. The plot was generated using seaborn with the following characteristics:

- **Figure Size:** 10 inches in width and 6 inches in height.

- **X-axis:** Ratings (`Ratings`) were represented on the x-axis.
- **Y-axis:** Frequency of ratings was represented on the y-axis.
- **Bins:** 30 bins were used for the histogram.
- **Title:** 'Distribution of Ratings' was assigned to the plot.
- **X-axis Label:** 'Ratings' was assigned to the x-axis.

Key observations from the histogram include:

- The distribution of ratings shows a peak at the lower end (slightly more than 0) with approximately 1400 occurrences.
- As ratings increase towards 50, the distribution gradually decreases.



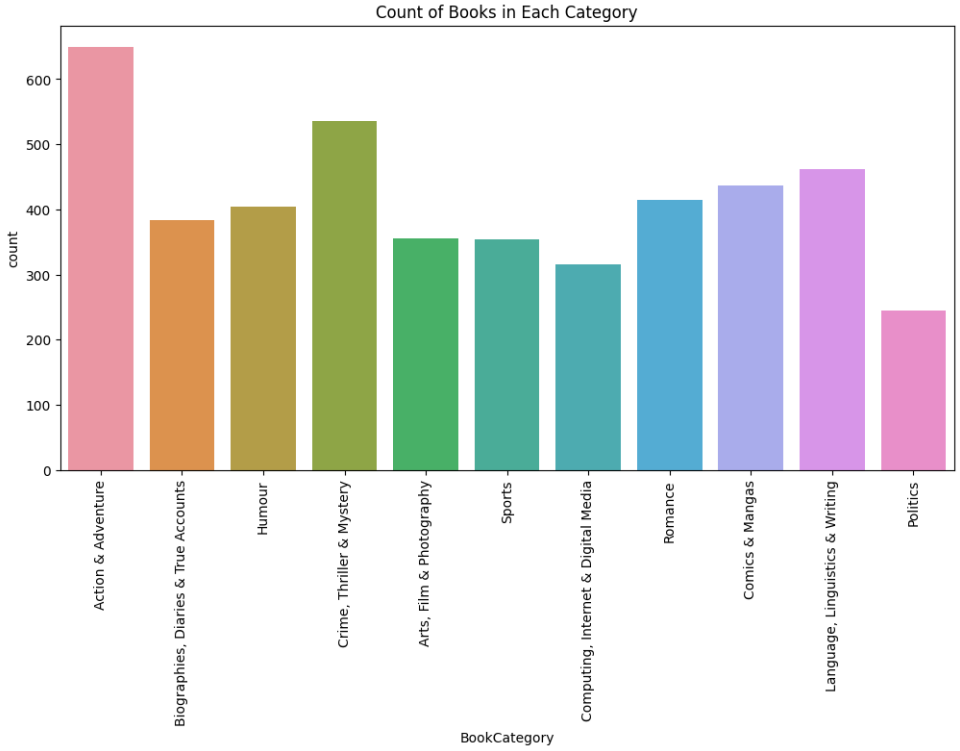
3.1.4 Count of Books in Each Category

To illustrate how books are distributed among various categories, a count plot was made. Seaborn was used to create the plot and had the following features:

- **Figure Size:** 12 inches in width and 6 inches in height.
- **X-axis:** Book categories (`BookCategory`) were represented on the x-axis.
- **Y-axis:** Count of books in each category was represented on the y-axis.
- **Title:** 'Count of Books in Each Category' was assigned to the plot.
- **X-axis Rotation:** Category names were rotated by 90 degrees for better readability.

Key observations from the histogram include:

- Different categories have varying counts of books.
- ”Action and Adventure” appears to have the highest count, while ”Politics” has the lowest count.

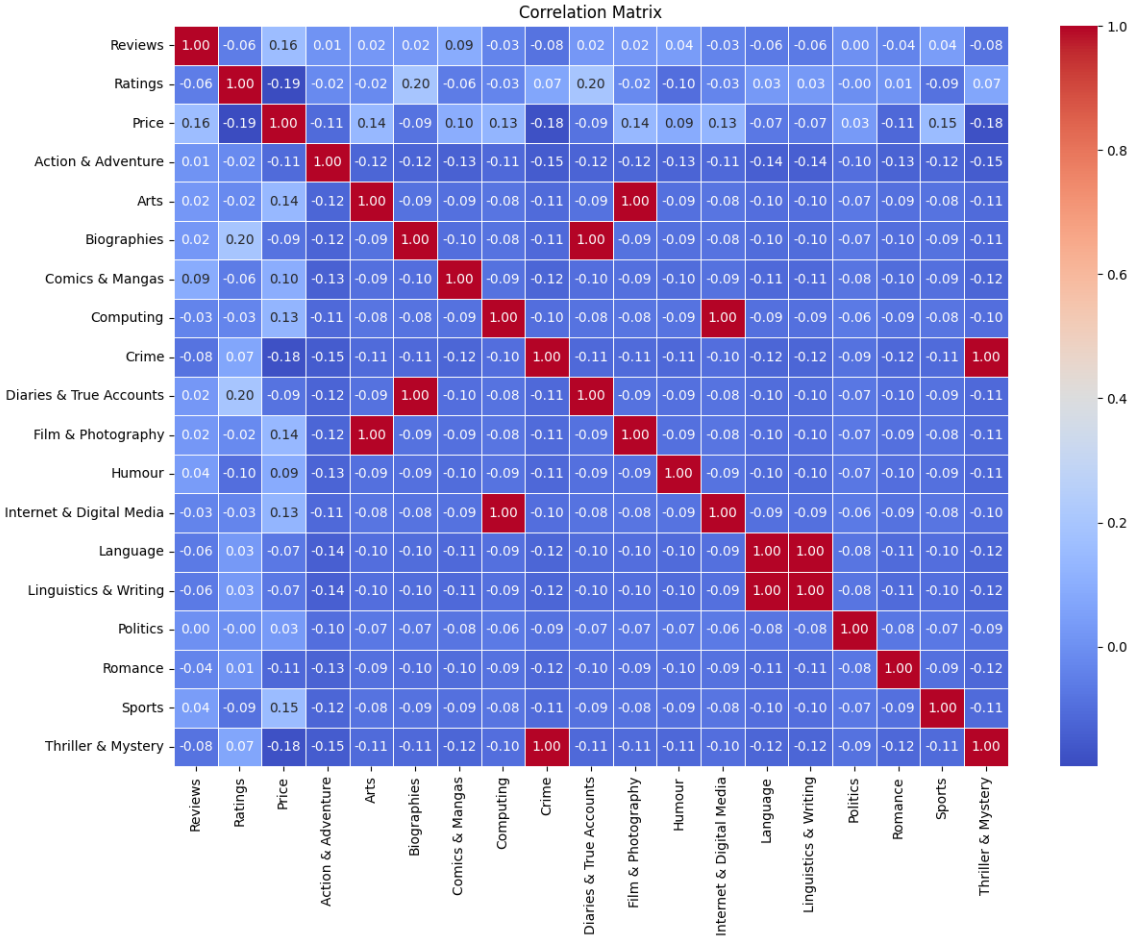


3.2 Correlation Matrix Heatmap

A correlation matrix heatmap was generated to visualize the relationships between different variables in the dataset. The matrix (`correlation_matrix`) was computed based on Pearson correlation coefficients. The heatmap was created using the seaborn library, with the following characteristics:

- **Figure Size:** 14 inches in width and 10 inches in height.
- **Colormap:** 'coolwarm' for better visualization of positive and negative correlations.
- **Annotations:** Numerical values were annotated on the heatmap.
- **Format:** Numerical values were formatted to two decimal places.
- **Linewidths:** Set to 0.5 for better visual separation between cells.
- **Title:** 'Correlation Matrix' was assigned to the plot.

The heatmap provides a visual summary of the correlation strength and direction between different features in the dataset. Notably, it indicates that only some genres exhibit significant correlation, while other columns have relatively low correlation.



Univariate Variable Analysis: Distribution of Reviews

To see how the distribution of book reviews varied, a kernel density estimation (KDE) histogram was created. Seaborn was used to create the plot and had the following features:

- **Figure Size:** 10 inches in width and 6 inches in height.
- **X-axis:** Number of reviews (`Reviews`) were represented on the x-axis.
- **Y-axis:** Frequency of reviews was represented on the y-axis.
- **Bins:** 30 bins were used for the histogram.
- **Title:** 'Distribution of Reviews' was assigned to the plot.
- **X-axis Label:** 'Number of Reviews' was assigned to the x-axis.

Key observations from the histogram include:
- There is a growing trend in the distribution of reviews, with a mean of 5.0. - Books with that amount of reviews become more common as the number of reviews tends towards 5.0.

3.3 Box plot

3.3.1 Price Distribution Across Book Categories

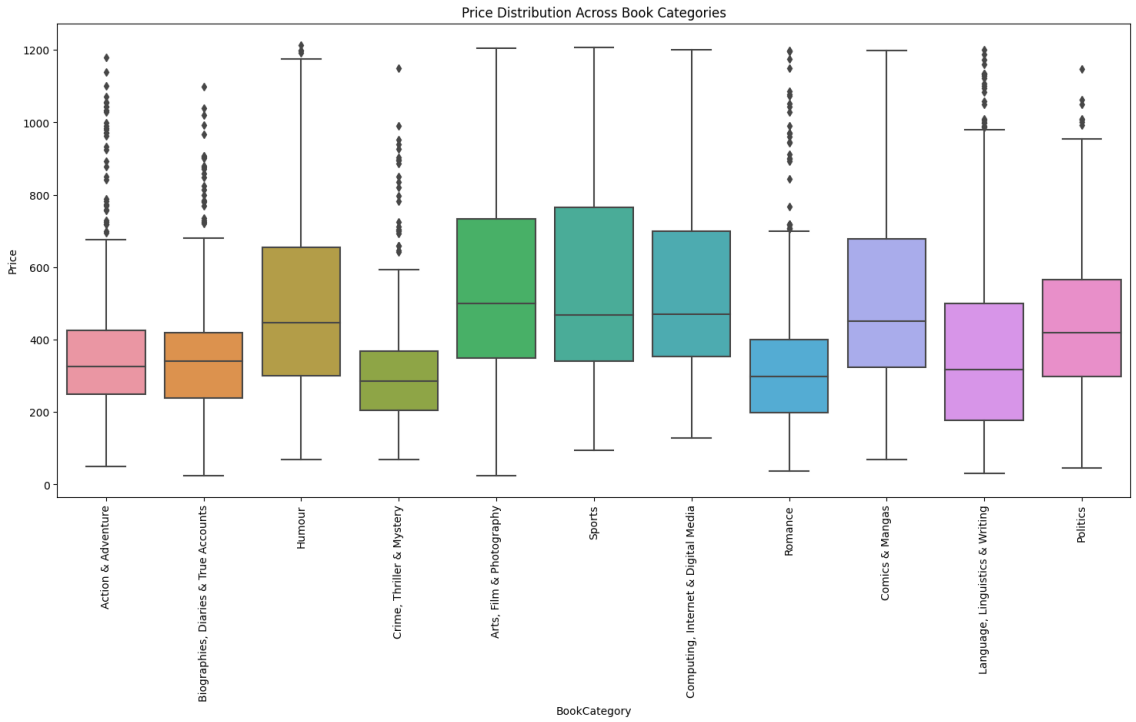
A boxplot was created to visualize the distribution of book prices across different categories. The plot was generated using seaborn, with the following characteristics:

- **Figure Size:** 18 inches in width and 8 inches in height.
- **X-axis:** Book categories were represented on the x-axis.
- **Y-axis:** Book prices were represented on the y-axis.
- **Title:** 'Price Distribution Across Book Categories' was assigned to the plot.
- **X-axis Rotation:** Category names were rotated by 90 degrees for better readability.

The boxplot provides insights into the distribution of prices across different book categories. Key observations include:

1. Various genres exhibit different price ranges.
2. The "Arts, Films & Photography" category tends to have the highest prices, with a wide interquartile range.
3. Categories such as "Crime, Thriller & Mystery" and "Romance" generally have lower prices.
4. The majority of book prices fall within the range of 200 to 650.

This visualization effectively conveys the variability in book prices among different categories, helping to identify trends and potential outliers.



3.3.2 Price Across Genres

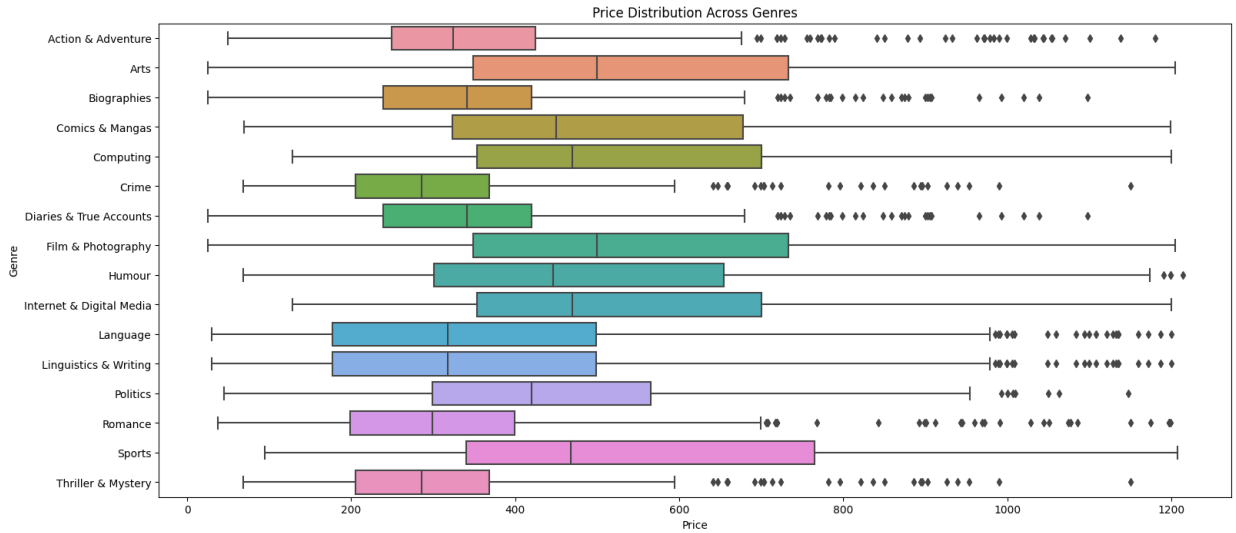
A boxplot was created to visualize the distribution of book prices across different genres. The plot provides a more detailed view of the price distribution within each genre. The key characteristics of the plot are as follows:

- **Figure Size:** 18 inches in width and 8 inches in height.
- **X-axis:** Book prices were represented on the x-axis.
- **Y-axis:** Genres were represented on the y-axis.
- **Title:** 'Price Distribution Across Genres' was assigned to the plot.
- **Boxplot Details:** Each genre has its boxplot, allowing for a detailed comparison of price distributions.

Key observations include:

1. "Arts" and "Film & Photography" genres exhibit similar boxplots, suggesting comparable price distributions.
2. "Language" and "Linguistics & Writing" genres have identical boxplots, indicating similar price patterns.
3. "Arts" and "Film & Photography" genres tend to have higher average prices compared to other genres.
4. "Thriller and Mystery" appears to be among the least expensive genres based on the boxplot.

This detailed genre-wise exploration provides a nuanced understanding of the distribution of book prices within each genre.



3.4 Scatter Plot

3.4.1 Reviews vs Ratings

To see how the quantity of reviews and book ratings related to each other, a scatter plot was made. Seaborn was used to create the plot, which has the following features:

- **Figure Size:** 10 inches in width and 6 inches in height.
- **X-axis:** Number of reviews (`Reviews`) was represented on the x-axis.
- **Y-axis:** Ratings (`Ratings`) were represented on the y-axis.
- **Title:** 'Relationship Between Reviews and Ratings' was assigned to the plot.
- **X-axis Label:** 'Number of Reviews' was assigned to the x-axis.
- **Y-axis Label:** 'Ratings' was assigned to the y-axis.

Key observations from the scatter plot include:

1. The majority of reviews are above 3.5.
2. Generally, an increase in the number of reviews correlates with higher ratings.

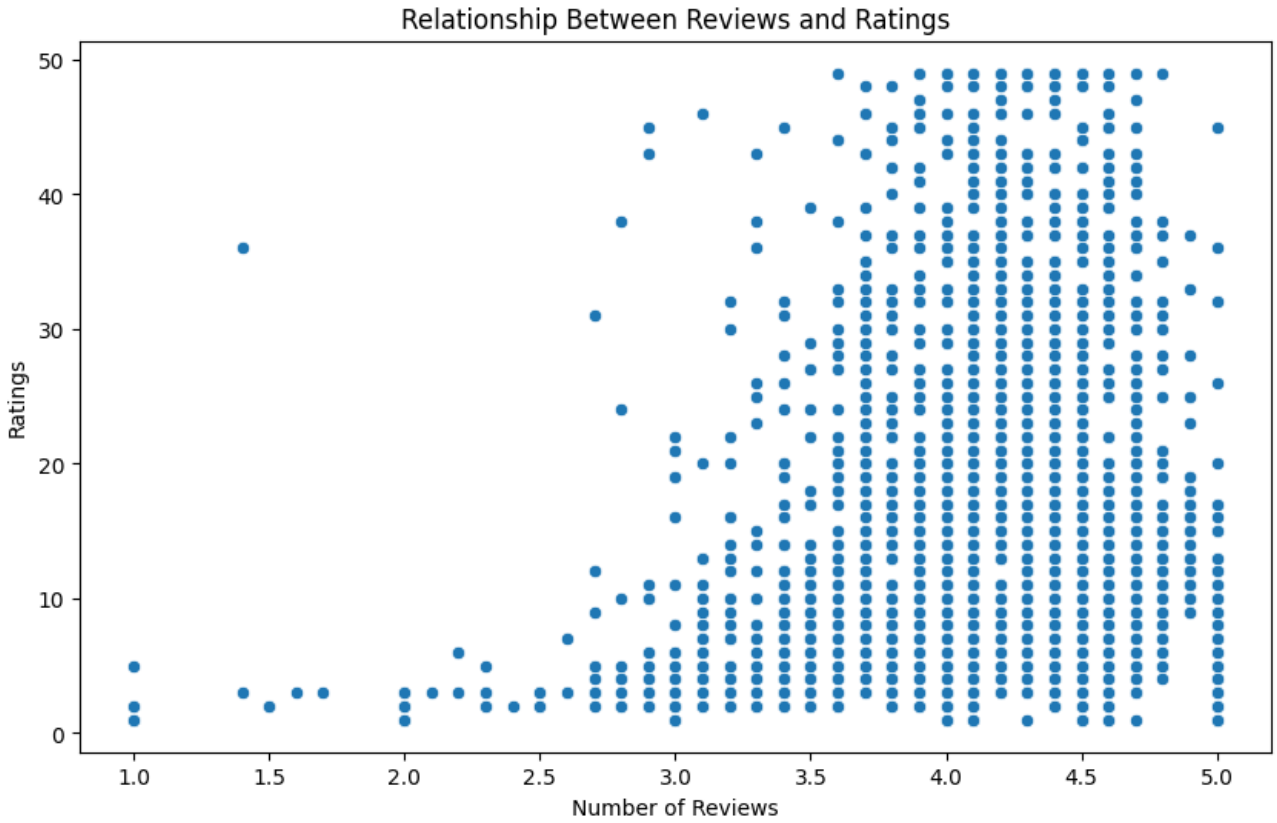
This scatter plot provides a visual representation of the relationship between reviews and ratings, offering insights into potential patterns or trends in the dataset.

3.5 lineplot

3.5.1 Distribution of Prices Over Time

To see how book prices have changed over time according to edition dates, a line plot was made. Seaborn was used to create the plot, which has the following features:

- **Figure Size:** 14 inches in width and 8 inches in height.
- **X-axis:** Edition dates (`Edition_Date`) were represented on the x-axis.
- **Y-axis:** Prices (`Price`) were represented on the y-axis.

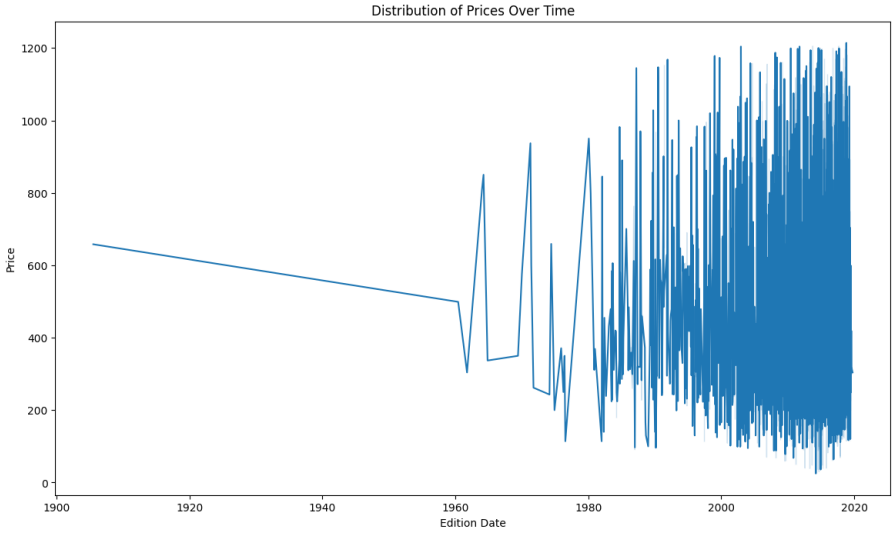


- **Title:** 'Distribution of Prices Over Time' was assigned to the plot.
- **X-axis Label:** 'Edition Date' was assigned to the x-axis.
- **Y-axis Label:** 'Price' was assigned to the y-axis.

Key observations from the line plot include:

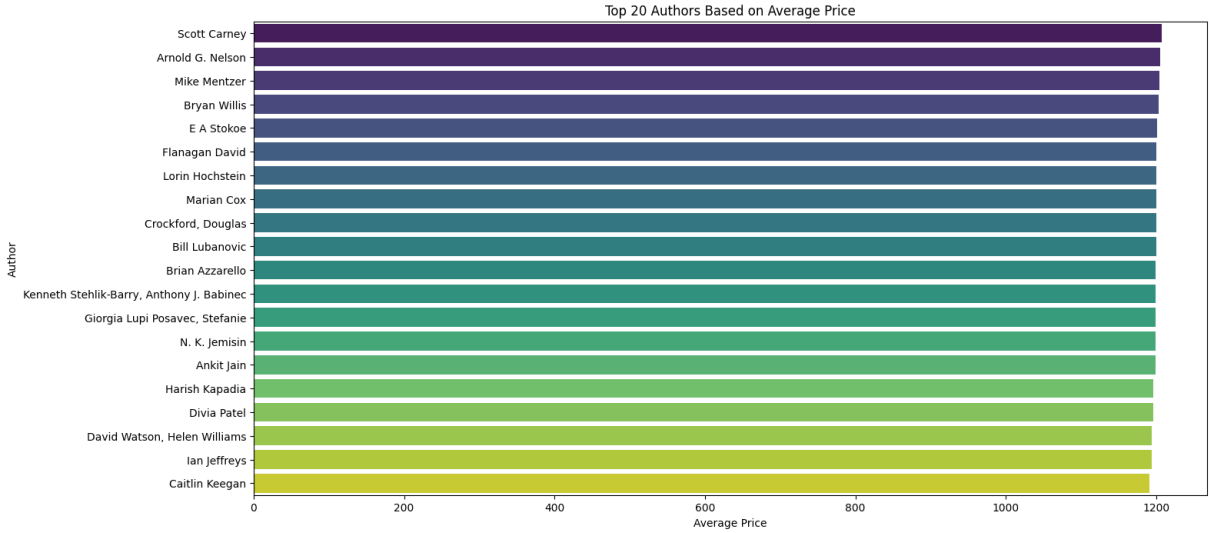
1. There doesn't appear to be a significant trend or pattern in the distribution of book prices over time.
2. Prices seem to vary without a clear upward or downward trend based on edition dates.

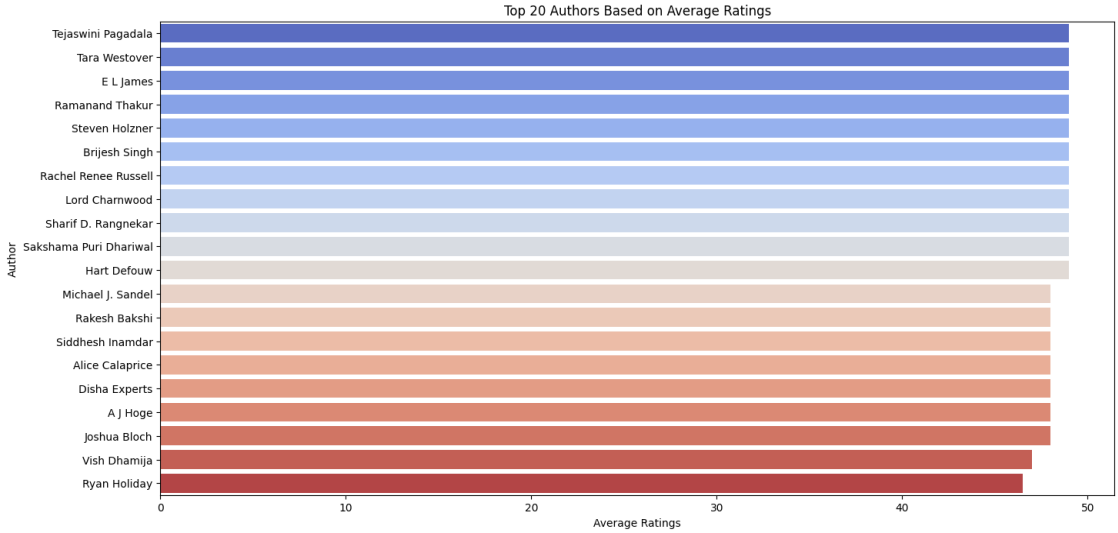
This line plot provides a visual representation of how book prices are distributed across different edition dates, indicating that edition date alone may not be a strong predictor of book prices.

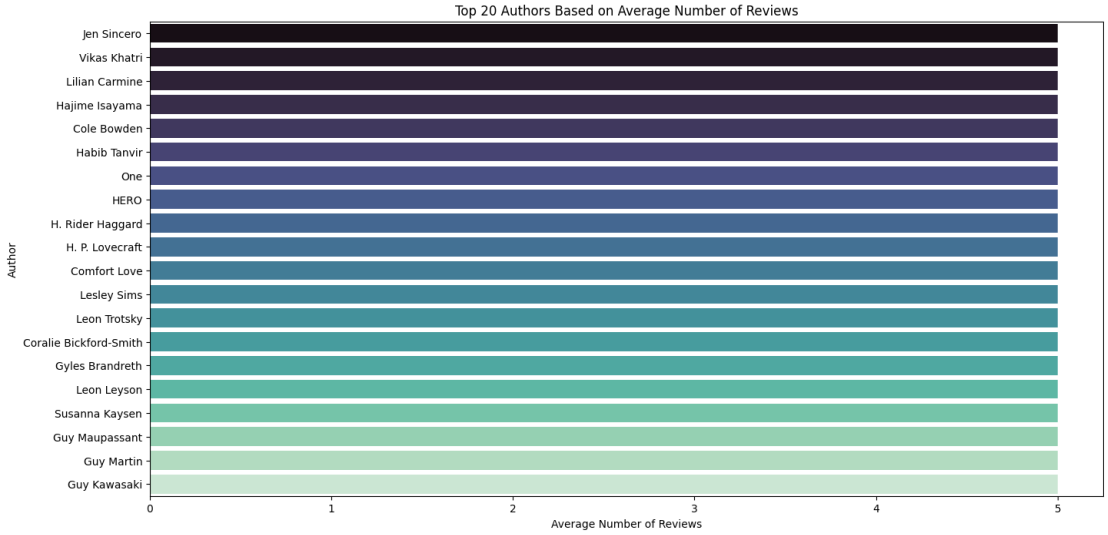


3.6 Barplot

Here are some Bar plots about top 20 Authors in different views







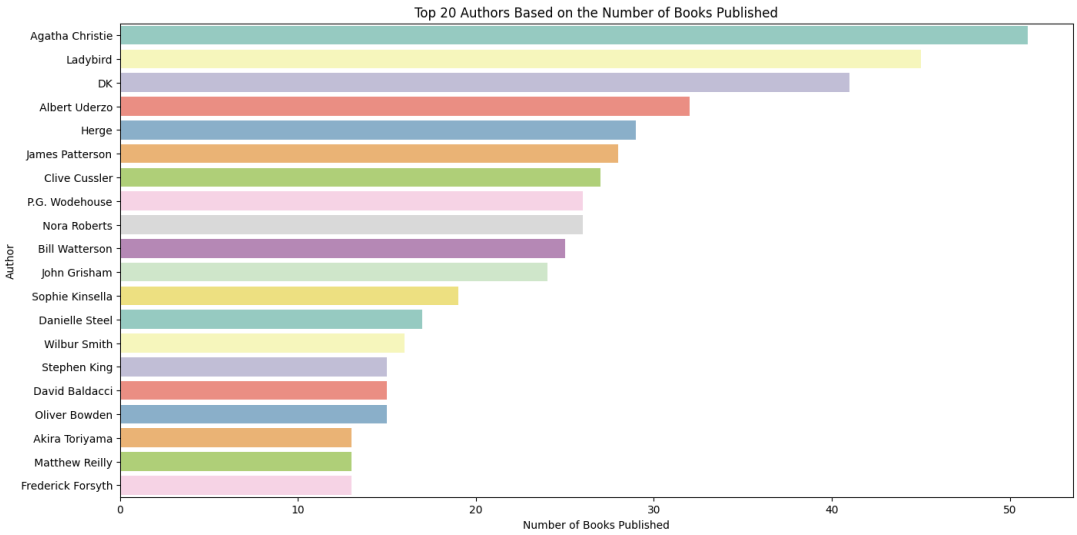
4 Feature Engineering

4.1 Title and Author Features

- Created a new feature 'Title_Length' representing the length of the book titles.
- Derived 'Author_Experience' by calculating the difference between the edition year and the first edition year of each author.
- Established 'Author_Avg_Rating' by computing the average rating for each author based on their books.

4.2 Temporal Features

- Introduced 'Edition_Year' to capture the publication year from the 'Edition_Date' column.
- Incorporated 'Edition_Month' and 'Edition_Day' to extract the month and day from the 'Edition_Date'.



4.3 Genre Features

- Created 'Genre_Count' by summing up the binary columns representing different genres for each book.

4.4 Synopsis Features

- Determined 'Synopsis_Length' by counting the number of words in the book synopsis.
- Conducted sentiment analysis on the synopses, resulting in 'Synopsis_Sentiment'.
- Utilized text analysis to generate a Bag of Words representation ('Synopsis_BOW') and TF-IDF representation ('Synopsis_TFIDF').

4.5 Author Popularity

- Aggregated total ratings for each author to create 'Author_Popularity'.

These engineered features are intended to improve the information content of the dataset by capturing different aspects, including textual information from book synopses, author characteristics, and temporal patterns. The purpose of the extracted features is to improve machine learning models' comprehension and forecasting of book prices by giving them useful input.

5 Feature Transformation

5.1 Polynomial Features

To capture nonlinear relationships, polynomial features were created for selected numerical features. The following steps were undertaken:

- Selected numerical features for polynomial transformation: ['Reviews', 'Ratings', 'Title_Length', 'Total_Ratings', 'Synopsis_Length'].
- Specified the degree of the polynomial features, allowing for flexibility in capturing the complexity of relationships.
- Utilized the PolynomialFeatures class from scikit-learn to instantiate the polynomial feature transformer.
- Applied the polynomial transformation to the selected features, creating a new set of features.
- Concatenated the new polynomial features with the original dataset.

5.2 Handling Non-Numeric Columns

To facilitate further analysis and modeling, non-numeric columns were separated from the main dataset. The process involved:

- Extracting non-numeric columns from the dataset.
- Creating a new DataFrame (`data_non_numeric`) containing only non-numeric columns.
- Creating a new DataFrame (`data_numeric`) containing only numeric columns by dropping non-numeric columns.

5.3 Drop Unnecessary Columns

To streamline the dataset and focus on relevant information, certain columns were dropped from the numeric dataset. The specified columns to drop included:

- 'Edition_Date', 'Edition_Year', 'Author_Experience', 'Edition_Month', 'Edition_Day'.
- The specified columns were removed from `data_numeric`, with potential errors ignored.

6 Modeling

6.1 Data Splitting

The dataset was split into training and testing sets using the `train_test_split` function from scikit-learn. The features (`X`) comprised all columns except the target variable ('Price'), and the target variable (`y`) was set as the 'Price' column. The split ratio was 80:20 for training and testing sets, respectively.

6.2 Random Forest Regression Model

A Random Forest Regressor model was chosen for its ability to handle complex relationships and provide robust predictions. The following steps were taken:

- Instantiated a Random Forest Regressor with the criterion set to 'friedman_mse'.
- Trained the model on the training data (`X_train, y_train`).
- Evaluated the model's performance on the training data using the mean squared error (MSE).
- Generated predictions on the test data (`X_test`) and evaluated the model's performance on the test data using the mean squared error (MSE).

6.3 Model Evaluation

The mean squared error (MSE) was used as the evaluation metric for the model. The MSE quantifies the average squared difference between the predicted and actual values. The results were as follows:

- Train MSE: `mse_train` :0.8984007557894284
- Test MSE: `mse_test` :45964.31063208713

These metrics provide insights into the model's performance on both the training and testing datasets, helping assess its generalization capabilities.

7 Conclusion

We explored most aspects of data transformation, feature engineering, exploratory data analysis, and preprocessing in this in-depth examination of the dataset that was centred on book details and prices. The main conclusions from our analysis are as follows:

Data Preprocessing

The first step was to prepare and understand the dataset. The dataset contained 5699 entries organised into 9 columns, including book titles, authors, editions, reviews, ratings, synopses, genres, book categories, and prices. The data type for every column was appropriate, and no missing values were discovered.

Exploratory Data Analysis

Various data visualisations were utilised to obtain a more profound comprehension of the dataset:

Different features showed varying degrees of correlation, according to a correlation matrix heatmap, with some genres showing particularly strong relationships. The distribution of book prices across genres and book categories was displayed using boxplots, which also highlighted potential outliers and varied price ranges. The relationship between the number of reviews and ratings was depicted using scatter plots, which generally showed a trend towards higher ratings with an increase in reviews. The distribution of book prices over time based on edition dates was examined using line plots, which did not reveal any discernible price trends.

Author-Centric Analysis

In order to compare average prices, ratings, and reviews among various authors, we lastly carried out an author-centric analysis and made bar plots. These visuals provided information about the differences in these metrics between authors.

Feature Engineering

Feature engineering played a crucial role in enhancing the dataset's information content. We introduced novel features, such as 'Title_Length' to capture book title lengths, 'Author_Experience' to measure the experience of each author, and 'Genre_Count' to aggregate the number of genres associated with each book. These features aimed to provide additional insights for predictive modeling.

Feature Transformation

We applied feature transformations to capture complex relationships within numerical features. Polynomial features were introduced for selected attributes, offering a more nuanced representation of their impact on book prices. Additionally, we separated non-numeric columns for a more focused analysis and dropped unnecessary columns to streamline the dataset.

Modeling

The dataset was split into training and testing sets, and a Random Forest Regressor model was trained to predict book prices. The model's performance was evaluated using the mean squared error (MSE), providing insights into its ability to generalize to new data.

In conclusion, our analysis provides a multifaceted view of the dataset, encompassing feature engineering, transformation, modeling, and exploratory data analysis. While no exhaustive model evaluation was performed in this report, the groundwork has been laid for further refinement and optimization of predictive models. Future work could involve fine-tuning the model, exploring advanced feature engineering techniques, and leveraging additional external data sources to enhance predictive accuracy.