DSTI Data ScienceTech Institute

Applied MSc in Data Analytics
Applied MSc in Data Science & Artificial Intelligence
Applied MSc in Data Engineering & Artificial Intelligence

Course: Python Machine Learning Labs

Project: Developing a Book Rating Prediction Model

Instructor: ***Hanna Abi Akl***

Group members

Sarvanan Sivakumaran
Claudine Uwitije
Yulia Chernova
Jasser Ismail

September 20th, 2024

TABLE OF CONTENT

CONTRIBUTORS TO THE PROJECT

This project was the result of the a teamwork whose members are the following team members and each contribute to different aspects of the work. Group members GitHub profiles are listed below for reference.

- Sarvanan Sivakumaran
- Claudine Uwitije
- Yulia Chernova
- Jasser Ismail

1. Introduction

In today's world, with an ever-increasing number of books being published, it has become a challenge for readers to discover the ones most aligned with their tastes. As Ernest Hemingway once said, ***"There is no friend as loyal as a book."*** However, in an age where countless books are available, finding that loyal friend has become overwhelming.

Platforms like Goodreads have transformed how readers interact with books, allowing millions of users to share their ratings and reviews. These user-generated ratings provide valuable insights into the quality and appeal of a book. Predicting these ratings using machine learning presents an exciting opportunity to help users make more informed reading choices. In this regard, this project leverages a curated dataset from Goodreads, including various features like title, authors, average rating, language code, number of pages, ratings etc. to train machine learning model that predicts book rating based on its characteristics.

This project is also a component of the machine learning coursework within our master's program, emphasizing practical application in data analysis, feature engineering and model training. It offers a unique opportunity to explore how machine learning techniques can be used in real-world scenarios to tackle everyday challenges, such as book rating

1.2 Project Objective:

The main objective of this project is to develop a machine learning model capable of predicting a book's rating by analyzing various features associated with books, such as title, author, number of pages, publication year, review counts etc.

Additionally, this project ensures that students can comfortably develop an end-to-end pipeline for solving a given problem or use case, enhancing their practical experience in machine learning.

2. Dataset Analysis

2.1 Dataset Overview

The dataset used for this project, provided by our Machine Learning professor, is a curated collection of books from Goodreads. It consists of several key variables that provide valuable insights into each book's characteristics. These variables include:

1. **bookID:** A unique identification number for each book.
2. **title:** The name under which the book was published.
3. **authors:** The names of the authors of the book. Multiple authors are delimited by "/".
4. **average_rating:** The average rating of the book received in total.
5. **isbn:** Another unique number to identify the book, known as the International Standard Book Number.
6. **isbn13:** A 13-digit ISBN to identify the book, instead of the standard 11-digit ISBN.
7. **language_code:** Indicates the primary language of the book.
8. **num_pages:** The number of pages the book contains.
9. **ratings_count:** The total number of ratings the book received.
10. **text_reviews_count:** The total number of written text reviews the book received.
11. **publication_date:** The date the book was published.
12. **publisher:** The name of the book publisher.

The dataset contains **11,127 rows** and **12 columns**, providing enough data for analysis. Upon importing the dataset into Pandas, an additional column was observed due to a data entry error. This column, which contained only one value (corresponding to line 3350 with bookID: 12224), was removed as it was unnecessary. Notably, the dataset does not have any missing values, which simplifies the data cleaning process. However, a preliminary analysis revealed the following key insights which were explored further in the subsequent steps of the project.

- **Zero-rated books**: Some books have zero ratings, which explains why the minimum value for the ratings column is zero.
- **High variability**: In certain variables, the standard deviation is greater than the mean, indicating significant variability in the dataset. This could be due to the presence of outliers or extreme values.
- **Publication dates**: 75% of the books in the dataset were published before 2005, indicating that the dataset is skewed towards older books and may lack more recent publications.

2.2 Data Cleaning

Data cleaning is crucial in any machine learning project to ensure that the dataset is free from inconsistencies, errors, and noise that could negatively impact model performance. For this project, the following preprocessing steps were conducted:

- **Title**: Despite no apparent duplicates in the dataset, 1,412 book titles were found to be duplicated. We addressed this by retaining the title with the highest average_rating.
- **Authors**: We decided to keep only the first author for each book. Analysis revealed that including additional authors resulted in significant missing data: only 5% of the records had complete information for second, third, and further authors.
- **Language**: We retained only the top four languages with a count greater than 120 and grouped all others into a category labeled Other language because most of them represent just one or 2 books. The final categories are: English, French, Spanish, Japan, and Other language .
- **Publisher**: Besides general cleaning, publishers listed in Chinese were translated into English.
- **Handling Missing Values:** The dataset was inspected for missing values and found to be complete, allowing us to proceed without the need for imputation or removal of incomplete records.
- **Removing Duplicates:** Although no duplicate records were found, 1,412 book titles were identified as duplicates. We addressed this issue by keeping the book title with the highest average rating.
- **Correcting Errors:** An erroneous 13th column was detected, containing a single non-null value for bookID: 12224 on line 3350, with all other values null. This column was removed as it did not provide meaningful information.
- **Outlier Detection and Handling**
    - **Books with Zero Ratings**: Some books had zero user ratings, which was unusual for popular titles on Goodreads. These entries were retained to provide insights into lesser-known books.
    - **Extremely Long Books**: A few books had unusually high or low page counts. These were manually checked and found to be legitimate (e.g., multi-volume collections). No further action was taken for these entries.
    - **Extremely Short Books**: even though they are not in outliers' range, books with fewer than 10 pages, often audiobooks, were removed from the dataset.
- **Data Type Corrections:** we ensured all variables had the correct data types. For instance, publication date was initially read as a string and was converted to a datetime format for easier manipulation and analysis.

2.3 Exploratory Data Analysis (EDA)

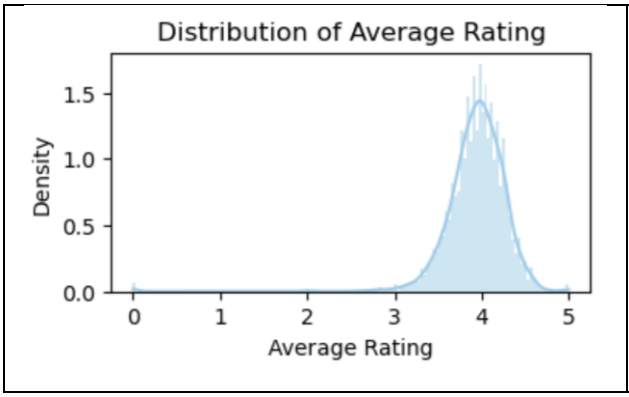
Exploratory Data Analysis (EDA) was conducted to gain a better understanding of the dataset and identify key patterns and relationships between variables. Various visualizations, including histograms, bar plot, boxplots, pie chart, scatter plot and correlation heatmaps, were utilized to uncover trends and guide further analysis. A quick overview of statistical measures

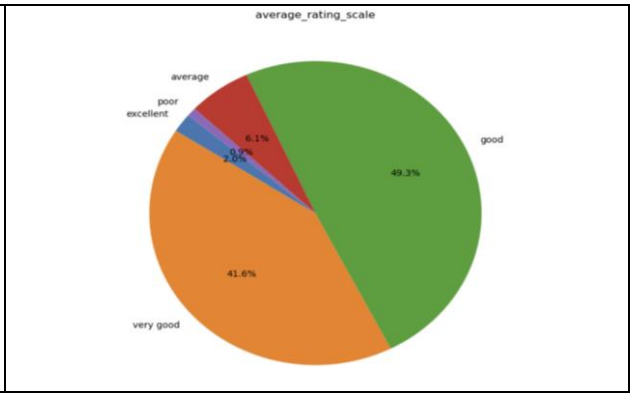
	average_rating	isbn13	num_pages	ratings_count	text_reviews_count	publication_year	ti
count	10287.000	1.028700e+04	10287.000	10287.000	10287.000	10287.000	
mean	3.940	9.758245e+12	335.739	17154.958	528.309	2000.234	
std	0.355	4.605875e+11	235.460	107967.693	2510.063	8.107	
min	0.000	8.987060e+09	11.000	0.000	0.000	1900.000	
25%	3.770	9.780350e+12	192.000	104.000	8.000	1998.000	
50%	3.960	9.780620e+12	294.000	800.000	47.000	2003.000	
75%	4.150	9.780890e+12	408.000	5276.500	244.000	2005.000	
max	5.000	9.790010e+12	6576.000	4597666.000	94265.000	2020.000	

2.3.1 Histograms

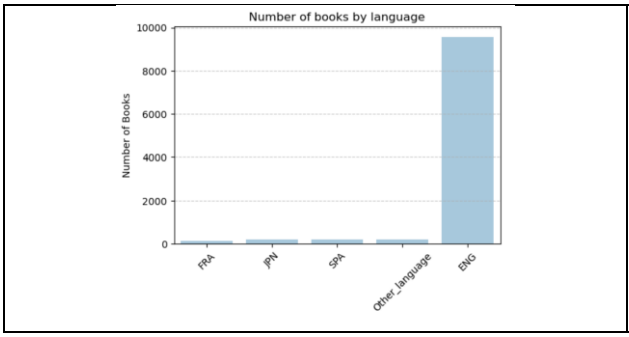
Histograms were created to visualize the distribution of key numerical features such as average rating, ratings count, text reviews count, num pages, and publication year. These distributions provided insights into the nature of the dataset:

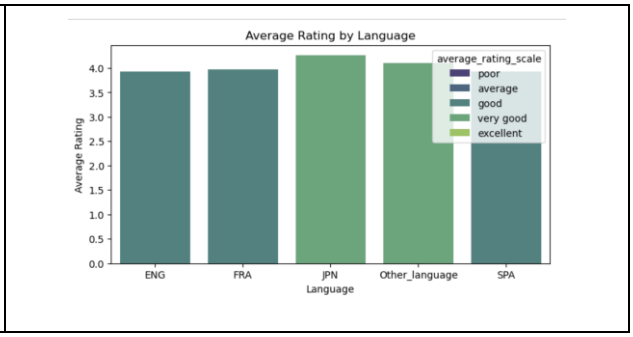
- **Average Rating**: Most books have a rating between 3 and 4.5, with a slight right-skew indicating a higher concentration of well-rated books. This explains why average rating is normally distributed from 3 to 5 and only a small number of books had ratings below 3, suggesting a generally positive reception. For a better visualization we group the average rating into five classes: **Poor (0-3)**, **Average (3-3.5), Good (3.5-4), Very good (4-4.5), Excellent (>4.5)**.



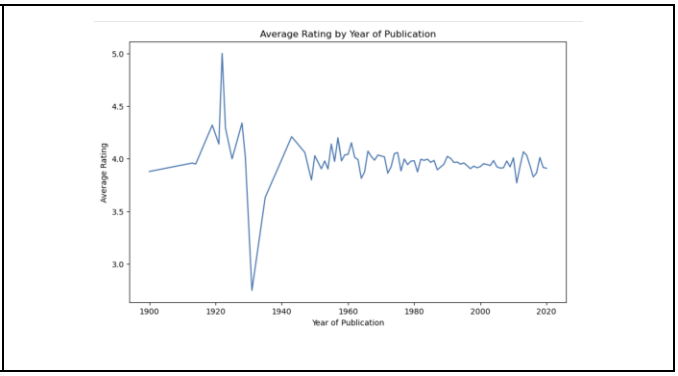


- **Ratings Count and text reviews count:** The distribution of these 2 variables were heavily right-skewed which suggests that while some books are highly popular, the majority are lesser known.
- **Language:** The dataset books are written in several languages, with the most prominent being English, French, Spanish, and Japanese, while all other languages were grouped under the category Other_language. Most books were in **English** (more than 90%), followed by This distribution reflects the global dominance of English in the publishing industry, particularly on platforms like Goodreads. The average rating across different languages did not show drastic variations. However **Japanese books** tended to have slightly higher average ratings compared to **English** , **French** and **Spanish** books, which could indicate that books written in these languages resonate more deeply with their respective readers.





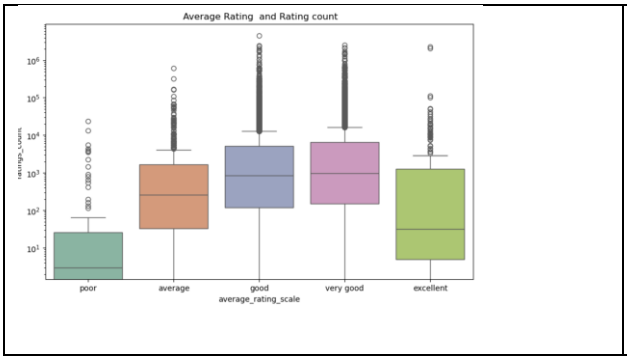
- **Number of Pages**: The number of pages ranged widely, though the bulk of books fell between 200 and 400 pages. Books with fewer than 10 pages were removed during the data cleaning phase.
- **Publication Year**: the dataset shows the publication years ranged from the early 1900s to 2020. We notice an i**ncrease in the number of books published** after the year 2000. This may be explained by the rise of digital publishing and self-publishing platforms, making it easier for authors to release their work. However, we notice that **: older books tend to have higher average ratings.**

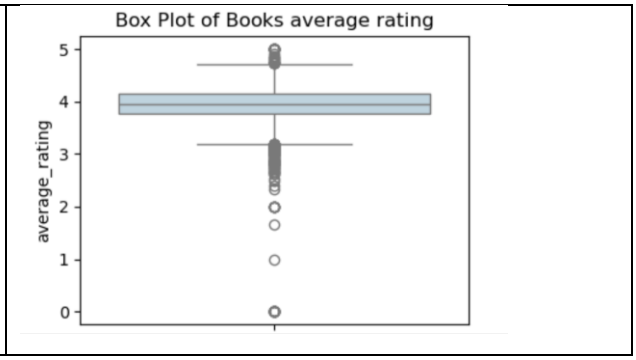


2.3.2 Boxplots

Boxplots were employed to detect outliers and understand the variability in key variables such as ratings count and average rating:

- **Ratings Count**: The boxplot revealed several outliers, particularly books with exceptionally high ratings counts. These outliers represented popular works.
- **Average Rating**: Most books clustered around the median rating of 4, with only a few outliers having unusually low or high ratings. The presence of such outliers was expected, as personal tastes and niche genres can influence ratings.

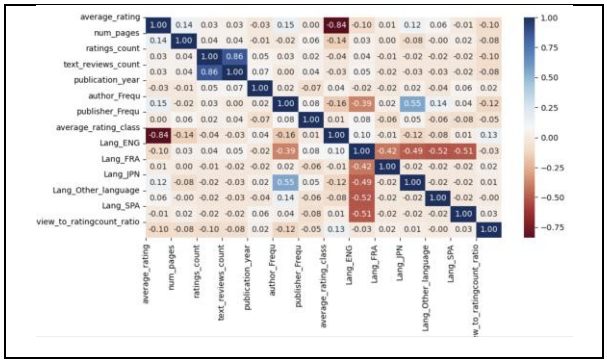




2.3.3 Correlation Heatmap

A correlation heatmap was generated to visualize the relationships among variables in the dataset, including average rating, ratings count, num pages, and publication year:

- **Average Rating vs each numerical variable**: In general, a very low correlation was observed
- **Ratings Count and text reviews count**: A strong positive correlation (approximately 0.86) was observed, indicating that books which receive more ratings tend to also have more written reviews. This is expected, as more popular books are likely to generate a higher volume of feedback, both in terms of ratings and reviews.



2.3.4 Key Trends and Findings

- **Relationship average rating and rating count:** A positive but very weak correlation between the number of ratings and the average rating was observed, suggesting that more popular books generally receive higher ratings. However, some high-rated books received relatively few ratings, indicating that niche or less well-known titles can also be highly regarded.
- **Outliers in ratings count**: Several books were identified as significant outliers in terms of the number of ratings, representing bestsellers or highly popular works.
- **Language Preferences**: English-language books dominated the dataset, followed by French, Spanish, and Japanese. The correlation heatmap showed no significant relationship between language and book ratings, suggesting that language preferences did not impact the overall ratings.
- **Publication year: Older books tend to have higher average ratings with pick straight after 1920** . This may be due to the reason that only the most popular or critically acclaimed older books are still widely read and reviewed today. We also notice an increase and decrease of average rating between 1920 – 1940 which can be explain by socioeconomic situation of that time.

2.4 Feature Engineering

Several transformations were applied to both categorical and numerical variables, and new features were generated to enhance the performance of the model. Below are the details of the feature engineering steps performed:

2.4.1 Encoding Categorical Variables

Many machine learning models require numerical input, so categorical variables were transformed into numerical representations using encoding techniques.

- **Language** feature, which included categorical values was encoded using **one-hot encoding.** This transformation created binary columns for each language, with a value of 1 indicating the presence of that language for a given book and 0 otherwise.
    - **Justification:** One-hot encoding was chosen because the language is a nominal categorical variable with no inherent ordering. By creating binary columns, the model could handle the languages without imposing any artificial ranking, which helped it learn the relationships between different languages and book ratings more effectively
- **Author and Publisher**: Frequency Encoding was used to handle these 2 features. This method involves replacing each unique category with the frequency of its occurrence in the dataset
    - **Justification:** Frequency presents the popularity of an author or a publisher without implying any order or ranking beyond their occurrence in the dataset. This can help the model better understand the potential impact of different authors/ publisher on book ratings.
- **Title: TfidfVectorizer** was utilized to transform the text data into numerical format. In the case of this dataset , the TF-IDF score reflects higher scores to words that are frequent in a specific title but rare across all titles
    - **Justification:** TfidfVectorizer was employed to encode the title feature due to its effectiveness in capturing the significance of words within the dataset

2.4.2 Normalizing Numerical Features

Numerical features with different scales can impact the performance of machine learning algorithms. To standardize the scales, normalization techniques were applied:

- **Number of Pages (num_pages), Ratings Count, text reviews count and publication year**: had a wide range of values. To bring this feature to a consistent scale, MinMaxScaler was used to normalize the values between 0 and 1.
    - **Justification**: Normalizing these features ensured that no single feature dominates the prediction of the model due to differing scales. Scaling these features helped balance its contribution to the model.

2.4.3 Generating New Features

Creating new features from existing data helped improve the richness of the dataset, capturing additional information that may be useful for predictions.

- **Year from Publication Date (publication_date):** originally stored as a full date string, was transformed to extract just the publication year. This new feature was treated as a numerical variable, representing how recent or old the book was.
    - **Justification:** Books published in different years often show varying reader preferences. By extracting the year, the model could learn whether more recent or older books tend to receive higher or lower ratings, contributing to better predictions.
- **Text reviews count to Rating count Ratio:** This is a new feature which was generated to capture the ratio of reviews (text_reviews_count) to ratings (ratings_count). This ratio help to assess how many users left a detailed text review compared to those who only provided a numerical rating.
    - **Justification:** This ratio offers insight into how actively readers engage with a book beyond simply providing a rating. This feature added valuable information about user interaction and engagement levels.

3. Model Training Process

3.1 Model Training Process:

- **Train-test split strategy:** Train-test split: 80% training, 20% testing.
- **Cross-validation:** k-fold cross-validation was employed during the training phase. In this approach, the training set was divided into k smaller subsets (or folds). The model was trained on k-1 folds and validated on the remaining fold, repeating this process k times.
- **Hyperparameter tuning methods:** Different models require different parameters, and every parameter needs to be set with the best possible values. To do so, Grid search.
- **Scaling features** was explained in feature engineering section

3.2 Model Selection:

The dataset includes the column **"average_rating"**, which contains continuous numerical data representing the average rating of books. Our objective is to predict the values in this column, making this a regression problem. Regression models are specifically designed to predict continuous values rather than categorical outcomes (which are handled by classification models). We will explore several regression algorithms to determine which one best fits the data and provides the most accurate predictions.

This table compares models, highlighting their descriptions, strengths, weaknesses, and use cases.

Model	Description	Strengths	Weaknesses	Best Use Case
Linear Regression	Assumes a linear relationship between input features and the target variable.	Simple, easy to interpret, and fast.	Assumes linearity, sensitive to outliers	Best for simple, small datasets with linear relationships
Ridge Regression (L2)	Linear regression with L2 regularization to prevent overfitting.	Reduces overfitting, handles multicollinearity	Struggles with non-linear relationships.	Suitable when features are correlated or multicollinearity is present.
Lasso Regression (L1)	Linear regression with L1 regularization, performing feature selection.	Automatically selects important features, reduces overfitting.	Can discard important features if regularization is too strong.	Best for high-dimensional datasets with irrelevant features
Decision Tree Regressor	Non-parametric model using feature-based splits to predict.	Models non-linear relationships, easy to visualize.	Prone to overfitting, sensitive to data changes.	Works well for non-linear datasets and when interpretability is needed.
Random Forest Regressor	Ensemble of decision trees, averaging their predictions.	Reduces overfitting, handles non-linear relationships well.	Computationally expensive, less interpretable.	Suitable for complex datasets with many features and non-linear patterns
Extra Trees Regressor	Ensemble of decision trees with additional randomness in splitting.	Reduces overfitting, faster than Random Forest, good generalization.	Can be computationally expensive and harder to interpret.	Similar to Random Forest but faster; great for high-dimensional data.
XGB Regressor	Gradient boosting algorithm optimized for speed and accuracy.	Excellent for large datasets, handles non-linear relationships, feature importance.	Requires more tuning, can overfit if not controlled.	Best for large, complex datasets with non-linear relationships and a need for high accuracy.
MLP Regressor	Neural network model for regression tasks.	Captures complex, non-linear relationships, flexible architecture.	Requires tuning, computationally intensive, can overfit.	Best for complex, non-linear problems where other models fail to capture relationships

- **Justification of Final Model Selection:** was based on preliminary tests, which evaluated the performance of each model using metrics such **as Mean Absolute Error (MAE**) and **R-squared.** Among the tested models, **Random Forest Regression** and **Gradient Boosting Regression** demonstrated superior performance in terms of accuracy and generalization to unseen data. Therefore, they were chosen as the final models for this project.

4. Model Evaluation

4.1 Evaluation Metrics

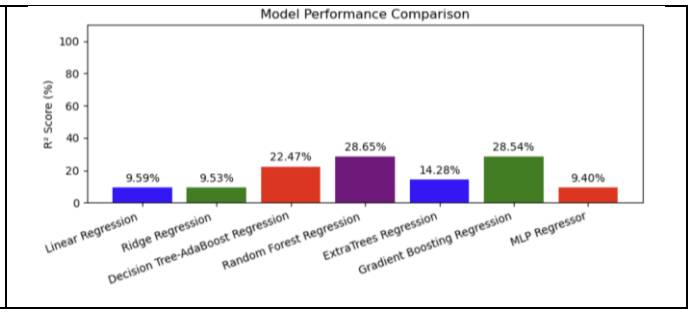
In this project, three metrics (**R², Mean Squared Error (MSE) and Root Mean Squared Error (RMSE)** were used to assess the performance of the regression models and offer a comprehensive view of model performance. These metrics guide the selection and optimization of models to ensure robust and reliable predictions of book ratings.

- **R² (Coefficient of Determination):** measures the proportion of variance in the dependent variable (average rating) that can be explained by the independent variables (features). R² values range from 0 to 1, where a higher R² informs that the model predicts fits well the data and provides a better understanding of the underlying relationships.
- **Mean Squared Error (MSE):** quantifies the average squared difference between the actual and predicted ratings. Lower MSE values indicate better model performance, as they reflect smaller discrepancies between predicted and actual ratings.
- **Root Mean Squared Error (RMSE):** offers insight into the average magnitude of the errors in predictions, making it easier to understand the model's accuracy. Like MSE, lower RMSE values signify better model performance.

4.2 Results Discussion

The performance of the various regression models was evaluated as shown in the following figure.

Model	R2_Score_percent	MSE	RMSE
Linear Regression	9.589600	0.109500	0.330900
Ridge Regression	9.532986	0.109612	0.331077
Decision Tree-AdaBoost Regression	22.473110	0.093933	0.306486
Random Forest Regression	28.649512	0.086450	0.294024
ExtraTrees Regression	14.277317	0.103864	0.322279
Gradient Boosting Regression	28.541672	0.086581	0.294246
MLP Regressor	9.403491	0.109769	0.331314

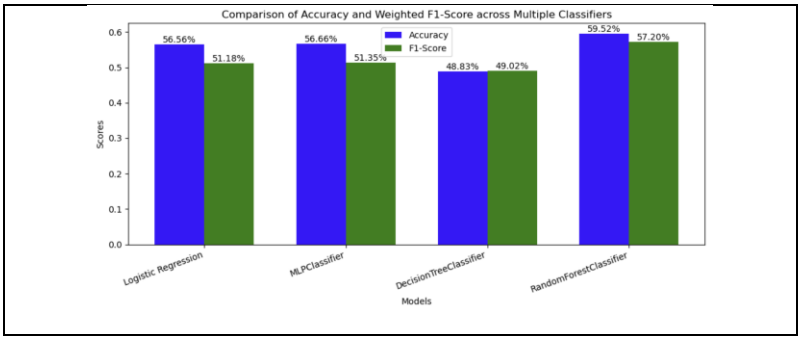


Among the models tested, **Random Forest Regression** emerged as the best performer, achieving an R² score of 28.65%, a MSE of 0.0864, and an RMSE of 0.2940. Following closely was **Gradient Boosting Regression**, which yielded an R² of 28.54%, a MSE of 0.0865, and an RMSE of 0.2942. Both models demonstrated superior predictive capability compared to other tested models, such as Linear Regression and MLP Regressor, which had R² scores below 10%.

The results indicate that the **Random Forest Regression** model not only explained a significant proportion of the variance in book ratings but also provided precise predictions, as evidenced by its lower MSE and RMSE values. This model effectively captures complex relationships within the dataset, allowing it to perform better than simpler models like Linear Regression.

5. Classification

In addition to the regression analysis for predicting continuous book ratings, I also trained several classification models to categorize books into five distinct rating groups: 'poor' (0-3), 'average' (3-3.5), 'good' (3.5-4), 'very good' (4-4.5), and 'excellent' (>4.5). The following result were obtained



6. Conclusion

6.1 Summary of Findings

This project aimed to predict book ratings using a comprehensive approach that included data analysis, feature engineering, and model evaluation. Key insights emerged from the exploratory data analysis, highlighting trends related to language, rating counts, and publication years. Various regression models were tested, with Random **Forest Regression** demonstrating the best performance, achieving an R^2 score of 28.65%, and low RMSE values. Despite its moderate predictive accuracy, the model revealed valuable relationships within the dataset, underscoring the importance of sophisticated modeling techniques for better outcomes.

On the other hand, the classification models provided slightly more encouraging results. **Random Forest Classifier** outperformed the other classifiers, achieving an accuracy of **59.52%** and an F1-score of **57.20%** The classification models showed a better ability to group books into rating categories, potentially because they simplified the prediction task by focusing on broad rating bands rather than precise rating values.

However, both approaches highlight the complexity of predicting book ratings, indicating opportunities for further model tuning or additional feature engineering.

6.2 Future Work

Future enhancements could focus on several areas to improve predictive performance. Expanding the dataset to include a wider range of books and user reviews could lead to richer insights. Additionally, incorporating more features, such as genre, or user demographics, could provide a more nuanced understanding of book ratings. Exploring advanced models, including deep learning techniques, may also yield improvements in accuracy.

6.3 Real-world Application

This project holds significant potential for real-world applications, particularly in the development of book recommendation engines. By accurately predicting book ratings, such systems can enhance user experiences by suggesting titles aligned with individual preferences. This could lead to increased reader engagement and satisfaction. Furthermore, publishers and authors could leverage these insights to understand market trends and reader preferences, ultimately guiding marketing strategies and publication decisions.

7. Reproducibility & Deployment

7.1 Reproducibility

- To ensure the reproducibility of this project, a **requirements.txt file** has been included, detailing all the necessary packages and their versions required to run the code effectively.
- Additionally, **a README file** provides comprehensive instructions on setting up the environment, executing the code, and understanding the project's structure.

7.2 Deployment

- The project has been deployed on GitHub, where all code, documentation, and associated files are accessible for public use.
- The group member GitHub profile are the following:
    - Sarvanan Sivakumaran
    - Claudine Uwitije
    - Yulia Chernova
    - Jasser Ismail

References

1. Python for Data Analysis, 3rd Edition ((McKinney, 2002)
2. Scikit-learn: https://scikit-learn.org/stable/
3. Seaborn: https://seaborn.pydata.org/
4. Youtube: Books Rating Prediction using Machine Learning Algorithm
5. TfidfVectorizer: Feature Engineering for Machine Learning in Python
6. Project resources: Recommending Goodreads Books using Data Mining

ANNEX