Goodreads Book Clustering Project

1.Project Statement

With thousands of books varying in genre, format, language, and popularity, identifying similar titles based solely on content characteristics can be challenging. A recommendation model makes it easier to discover books, find similar reads, and keep readers engaged.

2.Data Description

Source: The dataset was **self-collected** from **Goodreads** using **Selenium** in Python. It includes book details such as titles, authors, genres, formats, average ratings, and more, scraped from multiple pages on the site.

Size: The dataset contains **8,654 rows** and **12 columns**.

Columns:

- **Book title**: The name of the book
- **Author:** Author of the book
- **Genre:** One or multiple genres associated with the book
- **Format**: Includes both book format (e.g., Paperback, eBook) and number of pages
- **Language**: Languages the book is published in
- **ISBN:** A 10- or 13-digit International Standard Book Number, used to uniquely identify printed books
- **ASIN:** Amazon Standard Identification Number, primarily used to identify eBooks or books published exclusively through Amazon.
- **Rating:** The average rating given by users
- **Total Ratings**: Total number of users who rated the book
- **Reviews:** Number of written reviews on Goodreads

Data Issues:

- **Missing Values:** The dataset contains **3,703 missing values out of 8,654 rows**, with the majority found in the **ISBN (1,679) and ASIN (1,596) columns,** which are optional identifiers.
- **Formatting Issues:** Genres appear as comma-separated strings and required processing to be usable for analysis and modeling. Some fields also had inconsistent casing and extra whitespace.
- **No Duplicates:** Duplicate entries were avoided during data collection by using a **set()** to filter out repeated links during scraping.

Tools & Libraries:

- **Python** (pandas, numpy, matplotlib, seaborn, scikit-learn, etc.)
- **DuckDB** (used to run SQL queries on dataframes in Python)
- **Jupyter Notebook / VS Code**

3.Data Cleaning:

- **Dropped unnecessary column**:
    - Removed the **Unnamed: 0** column as it was just an index from the CSV file
    - Removed **ISBN** and **ASIN** as they are optional identifiers and not useful for modeling or analysis.
    - Removed **Published** after extracting **Release Date** and **Publisher** from it its original multi-valued format violated First Normal Form (1NF) and was no longer needed post-split.
- **Dropped rows with entirely missing information:**
    - Rows with missing **Book Title (0.30% missing)** were dropped since all other columns in those rows were also null.
- **Handled missing values**:
    - **Published (0.43% missing):** Dropped rows with missing values in the Published column as it was a small fraction of the data.
    - **Language (3% missing):** Filled missing values with **"English",** most frequent language in the dataset.
    - **Genres (3% missing):** Dropped rows with empty genre lists ([]) since they were few and genre is essential for recommendation modeling.
    - **Format (0.96% missing):** Instead of dropping these rows, missing values were filled evenly with the two most frequent values: 'Paperback' and 'Hardcover', to retain balance in the distribution.
    - **no_of_pages: (1.63% missing):** This column was extracted from **Format** during a split operation. In some cases, Format values were incorrectly assigned to **no_of_pages**. After resolving this, missing values in **no_of_pages** were filled with the column's mean (converted to float), then cast back to int for consistency.
    - **Release Date and Publisher (4.66% missing):** Dropped rows with missing values in either column, as these fields were originally extracted from the Published column and could not be reliably reconstructed**.**

- **Converted data types and extracted features:**
  - **Total Ratings and Reviews**: were originally stored as strings with commas (e.g., "1,234"). These were cleaned by removing commas and converting the values to integers.
  - **Format:** representing the format type (e.g., Paperback, Hardcover)
    This split was necessary because the original column violated the First Normal Form (1NF) by containing multi-valued attributes. Separating the values enabled cleaner structure and easier handling of missing data in each part.
  - **Release Date and Publisher:** were extracted from the Published column, which violated the First Normal Form (**1NF**) by containing multi-valued attributes (e.g., Penguin, 2014). The column was split into two distinct fields to normalize the data and enhance usability for analysis

4.Data Structuring

To prepare the dataset for modeling, a series of normalization and transformation steps were applied:

- A **Book_ID** was generated by grouping on **BookTitle and Author using ngroup()**. This provided a unique key for each book and supported clean joins with mapping tables.
- **Duplicate Book_IDs** were created due to independent multi-valued attributes in Format, Language, and Publisher, violating **4NF**. These columns were separated into their own mapping tables:
  - format_map
  - language_map
  - publisher_map
- The **Language** column violated **1NF** by storing multiple languages in a single cell. This was resolved by splitting the values and expanding them into separate rows during the mapping process.
- **Rereleased editions** introduced duplicate entries. However, since these did not materially affect clustering, the **Release_Date** was not separated. The same book across reissues was treated as a single entity.

- The **Genres** column also violated **1NF** by storing multiple genres per book. After expanding, this introduced another **4NF** violation. A new mapping table genre_map was created to resolve this.
- Each mapping table was kept minimal storing only key-value pairs (['Book_ID', 'Language']).
  A total of **311 duplicate rows** were removed across the mapping tables.
- Four new count-based features were engineered and added to the main dataset:
  - no_of_genres
  - no_of_publishers
  - no_of_formats
  - no_of_language
- The original multi-valued columns (**Genres**, **Publisher**, **Format**, **Language**) were dropped from the main dataset to eliminate redundancy.
- After all cleaning and normalization steps, the main dataset was reduced from **8,654** to **7,640** rows.

4a. Before and After Cleaning and Structuring (Snapshots)

- **Before Cleaning Snapshot**
  [Before.png]
- **After Cleaning Snapshot**
  [After.png]
- **ER Diagram**
  [Diagram.png]
- **Mapping Table Snapshots**
  - [Format_map.png]
  - [Genre_map.png]
  - [Language_map.png]
  - [Publisher_map.png]

5.Exploratory Data Analysis (EDA)

- **Engagement & Popularity**
  - Most books in the dataset are rated 4.0 or higher. However, ratings alone don't distinguish books well several titles with high ratings have low review counts. To filter noise, median thresholds for **Reviews** and **Total_Ratings** were used when highlighting "popular" books
- **Outliers Detected**
  - **no_of_pages, Total_Ratings, Reviews, and Rating** contain noticeable outliers, which may influence clustering results. Some form of scaling or adjustment might be needed.
- **Multicollinearity Found**
  - Features like **Total_Ratings**, **Rating**, **no_of_formats**, and **no_of_publishers** show high correlation, which could affect clustering results if not addressed through normalization or reduction
- **Low-Variance Features**
  - Although **Author and Publisher** may seem useful, they suffer from high cardinality and low frequency even the top authors have written fewer than 35 books.
  - Even after normalization, English remains the dominant language in the Language column, showing low variability.

6.Feature Selection

- **Initial Model Frame**: A copy of the structured dataset was made (**model_df**) specifically for modeling purposes.
- **Dropped Features:**
  - **BookTitle, Author, Publisher:** These columns had high cardinality and low frequency thousands of unique values, most appearing only once or twice. While these features might influence a book's success in theory, the sparse and uneven distribution made it difficult for the model to learn from them meaningfully**.**
  - **no_of_formats, no_of_languages, no_of_publishers, no_of_genres, Languages** These were dropped due to **extreme class imbalance** most books had only one format, language, etc., offering little variation for clustering. Additionally, these were aggregated features based on mapping tables, so their dimensionality added more cost than benefit.

- **Cleaned & Transformed Features**
  - **Total_Ratings, Reviews, No_of_Pages:** These numeric features showed strong positive skew. To correct this and reduce the influence of extreme outliers, log transformation was applied. This helps the model treat these features more evenly.
  - **no_of_pages, Total_Ratings, Reviews, Rating, Year**: These features were scaled using StandardScaler to ensure they're on the same scale before clustering. This was necessary because features like **Total_Ratings** and **Rating** have very different value ranges. Without scaling, K-Means would prioritize larger numerical values, leading to biased clustering. Standardization helped the model treat all features equally and improved clustering accuracy.
  - **Book_ID** was made the index so it doesn't interfere with the model while still being safely stored. This way, it won't affect clustering but can still be used later to join back important columns like **Author**, **BookTitle**, or **Publisher**
- **Encoded Features:**
  - **Genre and Format:** high-cardinality categorical variables. Instead of dropping them, **low-frequency categories (those appearing in less than 0.5% of the dataset)** were grouped into an "Other" class, and the remaining values were one-hot encoded. This helped preserve the most informative categories while reducing dimensionality.
    - **Genres:** 547 unique genres originally 421 grouped into "Other" (6% of rows)
    - **Formats:** Top 4 formats kept, remaining 21 grouped into "Other" (10% of rows)

6a. Statistical Testing

Chi-Square Test of Independence was conducted:

- **Tested Variables:**
  - **Year vs Genre**
  - **Month vs Genre**

6b. Hypotheses:

- **Null Hypothesis (H0):** No significant association between the time variable and book genre.
- **Alternative Hypothesis (H1):** Significant association exists.
- **Result:**
  - **Year:** 76% of genre combinations were statistically significant → **Reject H0**.
  - **Month:** Only 39% significance → Weak or no association.

7.Clustering and Evaluation

K-Means clustering was used, and performance was evaluated across different values of **k (2 to 10)** using the following metrics:

- **Silhouette Score:**
  - Highest score: **0.115 at k = 2**
  - Overall range: **0.07 to 0.11**
- **Inertia Curve:**
  - Declined steadily with no clear elbow point.

Despite thorough preprocessing and well-structured feature selection, the clustering results were not particularly strong.

8.Interpretation and Limitations

The results point to a key limitation in the dataset: it lacks deeper behavioral or contextual features that could help uncover meaningful groupings. Genre and Format provide some insight, but they only scratch the surface. More nuanced signals like how popular an author is, the reputation of a publisher, or patterns in how readers engage with certain types of books could've added valuable structure for clustering.

For example, two books from different genres might still appeal to a similar audience if they share a popular author, writing style, or publisher.
But features like **Author** were left out due to **high cardinality and low frequency**; most authors appeared only once making them more noisy than helpful. Including such sparse features would likely have hurt model performance.

Still, with **more data on author popularity or reader engagement patterns**, even these complex variables could support better clustering. Without them, the dataset, while clean and structured, lacks the depth needed for strong unsupervised learning.