



**FINAL**

---

**REPORT**

*GROUP 5*

## TABLE OF CONTENTS

TOPIC	PAGE NUMBER
Project Summary	3
Introduction	4
Problem Statement	5
Goal of Project	5
EDA	6
Data Preview	6
Shape of data	6
Data Information	6-7
Columns in the Dataset	7
Data Description	7
Inferences	8-9
Null Values	9
Inferences	9
Null value Treatment	9-10
Data Description after null values	10
Inferences	10-11
Univariate Analysis	11-13
Inferences	13-14
Bivariate Analysis	14-16
Inferences	16-17
Notes for project team	25

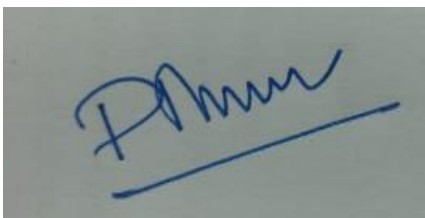


TOPIC	PAGE NUMBER
Label Encoding	<b>18</b>
Changing Datatypes	<b>19</b>
Inferences	<b>19</b>
Datatypes after Conversion	<b>20</b>
Process after EDA Performance	<b>20</b>
NLP	<b>21</b>
NLP Preprocessing	<b>21-22</b>
Recommendation System	<b>22</b>
Recommendation Products	<b>23</b>
Inferences	<b>23-24</b>
Chatbot	<b>24</b>

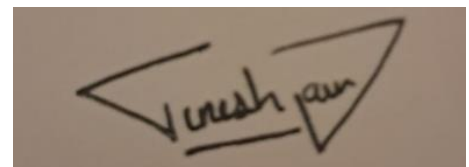
# PROJECT SUMMARY

Batch details	PGP-DSE July'23 Gurgaon
Team members	Dhwani Agrawal, Sachin Bajaj, Jinesh Jain, Parikshit Pounikar, Saumaya Singh Raghuvanshi, Ayush Kumar Raikwar, Perna Sinha
Domain of Project	E-Commerce
Proposed project title	Customised Product Recommendations: Your needs Our Solution
Group Number	Group-5
Team Leader	Jinesh Jain
Mentor Name	Dr. Pankaj Agrawal

Date: 15th March 2024

A blue ink signature on a grey background, appearing to be 'Dr. Pankaj Agrawal'.

Signature of the Mentor

A black ink signature on a brown background, appearing to be 'Jinesh Jain'.

Signature of the Team Leader

# 1. INTRODUCTION

Customers are frequently overpowered by the sheer number of things accessible in the wide world of e-commerce, where choice is king. Decision fatigue, postponed purchases, and even unhappiness with the chosen option might result from this paradox of choice. Product recommendation systems have become a potent tool to help buyers find products that suit their unique needs and preferences in response to this difficulty.

One of the biggest online retailers in the nation, Amazon India, has an enormous selection of goods in many different categories. But for consumers, knowing how to navigate this large catalogue successfully might be intimidating. When it comes to providing a genuinely customised and user-friendly buying experience, traditional search and filtering methods frequently need to be revised.

**This is where chatbot technology's potential shows itself, providing a more dynamic and conversational method of product discovery.**

## 2. PROBLEM STATEMENT

---

Web-scraping an e-commerce website (Amazon India) to create a dataset where the product's details are checked and then a chatbot is coded to provide the solutions based on user input, e.g., the user asks to recommend a phone under 40k. Following are the parameters based on which recommendation will be provided:

- 1) Recommendation based on Price
- 2) Recommendation based on Brand
- 3) Recommendation based on Memory
- 4) Recommendation based on Rating

## 3. GOAL OF THE PROJECT

The goal of this project is to create a chatbot that uses web-scraped data from Amazon India to close the gap between the amount of product information and the demand for individualised recommendations. Price, brand, memory, and rating are the four main product factors that are most important in influencing decisions to buy. The chatbot aims to improve customer happiness by streamlining the purchasing experience by allowing consumers to filter recommendations based on these parameters. Project involves comprehensive data preprocessing, exploratory data analysis, and feature engineering to enhance the accuracy of the predictive model

# 4. EXPLORATORY DATA ANALYSIS

## DATA PREVIEW

[3]:

df=

pd.read\_csv('merged\_file.csv')

df.head(5)

	Product_Name	Selling Price	MRP	Items Bought Last Month	Ratings	Numeric_Ratings	Total Ratings	Brand	Offer%
0	1.5 Ton 3 Star AI Flexicool Inverter Split AC ...	33990.0	67790.0	600	4.0 out of 5 stars 2,085	4.0	2085	Whirlpool	49.86
1	1.5 Ton 3 Star Inverter Split AC (Copper, PM 2...	36990.0	58400.0	600	4.0 out of 5 stars 1,454	4.0	1454	Voltas	36.66
2	1.5 Ton 3 Star Inverter Split AC (5 in 1 Conve...	32990.0	58990.0	500	4.2 out of 5 stars 5,561	4.2	5561	LG	44.08
3	1.5 Ton 5 Star Wi-Fi Inverter Smart Split AC (...)	42990.0	63400.0	500	4.2 out of 5 stars 4,881	4.2	4881	Whirlpool	32.19
4	1.5 Ton 5 Star AI Flexicool Inverter Split AC ...	40990.0	76090.0	300	4.0 out of 5 stars 1,531	4.0	1531	Haier	46.13

## SHAPE OF DATA

In [127]:

df.shape

Out[127]:

(81915, 9)

## DATA INFORMATION

```
In [7]: df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 81915 entries, 0 to 81914
Data columns (total 9 columns):
#   Column                      Non-Null Count  Dtype
---  ---
0   Product_Name                81915 non-null  object
1   Selling Price               81915 non-null  object
2   MRP                         81915 non-null  object
3   Items Bought Last Month     81915 non-null  object
4   Ratings                     81915 non-null  object
5   Numeric_Ratings             81915 non-null  object
6   Total Ratings               81915 non-null  object
7   Brand                       81907 non-null  object
8   Offer%                      81915 non-null  object
dtypes: object(9)
memory usage: 5.6+ MB
```

## COLUMNS IN THE DATASET

```
In [8]: df.columns

Out[8]: Index(['Product_Name', 'Selling Price', 'MRP', 'Items Bought Last Month',
              'Ratings', 'Numeric_Ratings', 'Total Ratings', 'Brand', 'Offer%'],
              dtype='object')
```

## DATA DESCRIPTION

```
In [11]: df.describe().T

Out[11]:
```

	count	unique	top	freq
Product_Name	81915	74415	Women's Sneakers	36
Selling Price	81915.0	17278.0	399.0	3608.0
MRP	81915.0	8670.0	999.0	5021.0
Items Bought Last Month	81915	24	50	23402
Ratings	81915	11230	4.5 out of 5 stars	58
Numeric_Ratings	81915.0	79.0	4.5	8839.0
Total Ratings	81915	4611	58	8301
Brand	81907	3476	Skechers	4253
Offer%	81915.0	10088.0	60.06	3536.0



## INFERENCES

### 1. **Price Analysis:**

- The Selling Price ranges from 399.0 to 81915.0, indicating a wide range of pricing options.
- The MRP (Maximum Retail Price) also varies significantly from 999.0 to 81915.0.
- The Offer% ranges from 0% to 100.06%, suggesting there are various discounts offered on different occasions or for different products within the same category.

### 2. **Sales Performance:**

- The number of items bought last month varies, with a maximum of 23402 units sold, suggesting a high demand for this product.
- The high number of units sold indicates a strong market demand for women's sneakers.
- This could be influenced by factors such as brand reputation, style, quality, and marketing efforts.

### 3. **Customer Satisfaction:**

- The average numeric rating is 4.5 out of 5 stars, based on 8839 ratings.
- This indicates a high level of customer satisfaction with the product.
- Positive ratings can contribute to brand loyalty and repeat purchases.

### 4. **Brand Analysis:**

- Skechers appears to be the dominant brand in this category, with 4253 mentions out of 81907 records.
- Skechers seems to have a strong presence and customer preference in the women's sneaker market.
- The brand's reputation might influence purchasing decisions and contribute to its market share.

### 5. **Price vs. Demand:**

- There seems to be an inverse relationship between price and the number of items bought, with higher-priced items selling fewer units.
- The presence of discounts (Offer%) might influence consumer behavior, leading to increased sales for discounted items.

### 6. **Price vs. Ratings:**

- Higher-priced items may have higher ratings, suggesting that customers perceive them as higher quality or value for money.
- However, this relationship would need further analysis to determine causality accurately.

In conclusion, "Women's Sneakers" seem to be a popular product, especially those offered by the Skechers brand, with varying price points and discounts influencing consumer behavior. The high sales volume and positive ratings indicate a strong market position for this product.

## NULL VALUES

```
In [128]: df.isnull().sum()
Out[128]: Product_Name      0
Selling Price      0
MRP      0
Items Bought Last Month      0
Ratings      0
Numeric_Ratings      0
Total Ratings      0
Brand      165
Offer%      0
dtype: int64
```

## INFERENCES

- The completeness of data for most columns suggests that thorough analysis can be conducted regarding pricing, sales performance, brand popularity, and customer satisfaction.
- The presence of missing values in the "Brand" column may require further investigation or handling, such as imputation or exclusion, depending on the analysis goals.
- Overall, the dataset seems suitable for various analyses related to product performance, pricing strategies, and customer preferences. However, consideration should be given to handling missing brand information appropriately.

## NULL VALUE TREATMENT

```
In [129]: df=df.dropna()
```

```
In [130]: df.isnull().sum()
```

```
Out[130]: Product_Name      0
Selling Price      0
MRP      0
Items Bought Last Month  0
Ratings      0
Numeric_Ratings      0
Total Ratings      0
Brand      0
Offer%      0
dtype: int64
```

```
In [131]: df.shape
```

```
Out[131]: (81750, 9)
```

## DATA DESCRIPTION AFTER NULL VALUE TREATMENT

```
In [12]: df.describe().T
```

```
Out[12]:
```

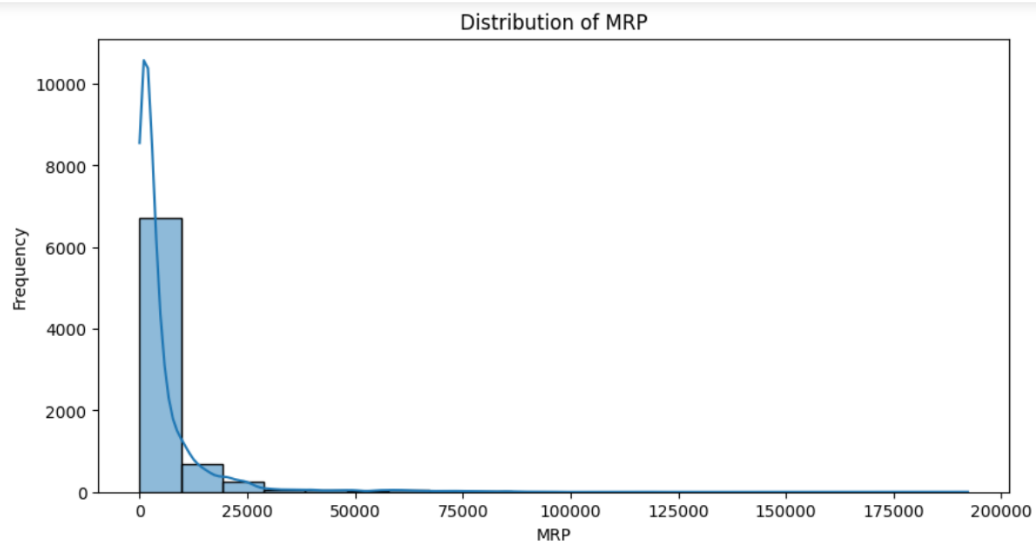
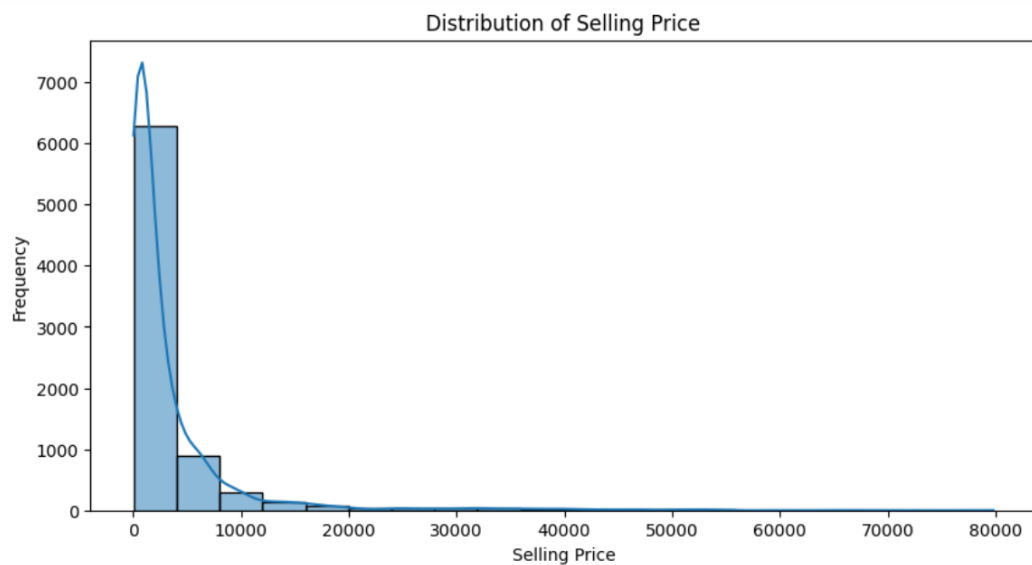
	count	unique	top	freq
Product_Name	81915	74415	Women's Sneakers	36
Selling Price	81915.0	17278.0	399.0	3608.0
MRP	81915.0	8670.0	999.0	5021.0
Items Bought Last Month	81915	24	50	23402
Ratings	81915	11230	4.5 out of 5 stars	58
Numeric_Ratings	81915.0	79.0	4.5	8839.0
Total Ratings	81915	4611	58	8301
Brand	81907	3476	Skechers	4253
Offer%	81915.0	10088.0	60.06	3536.0

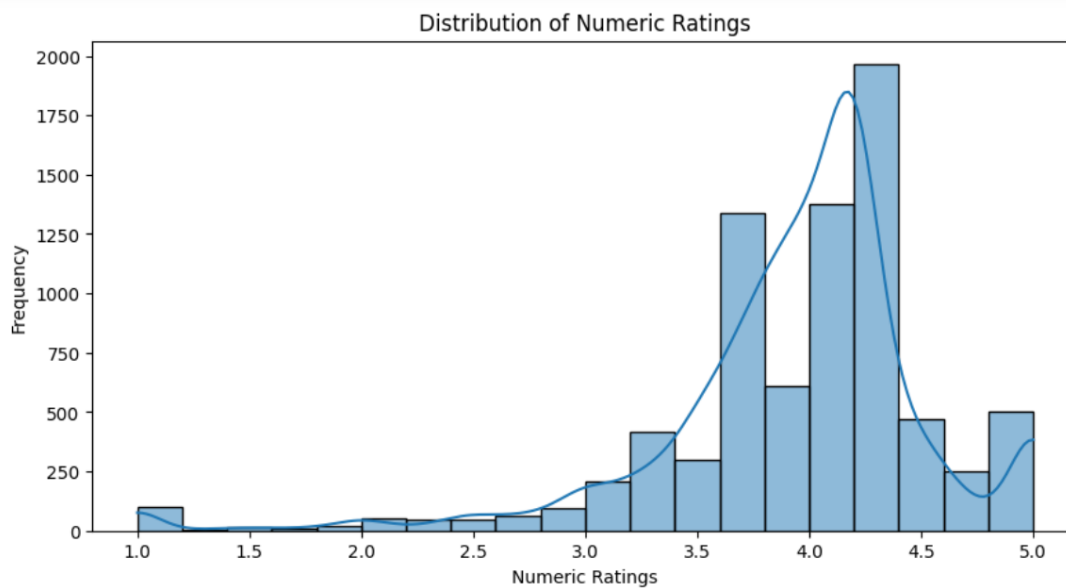
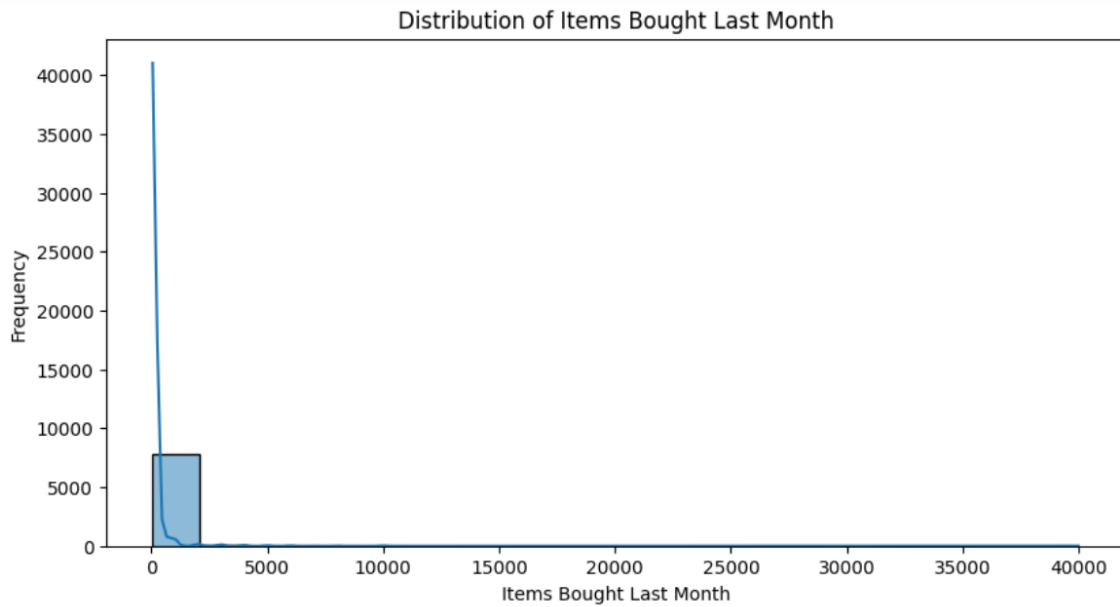
## INFERENCES

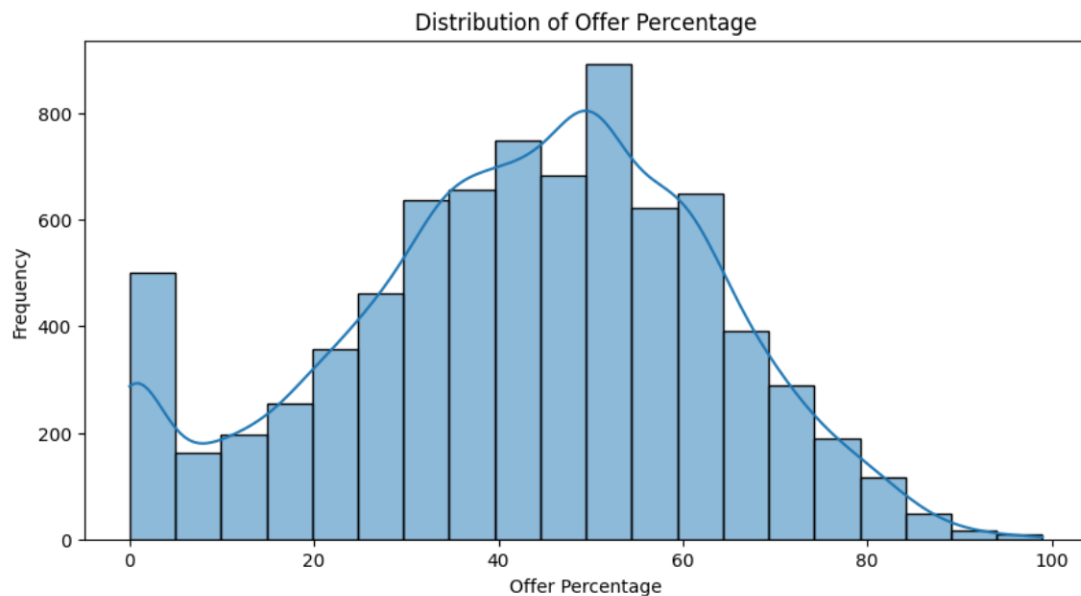
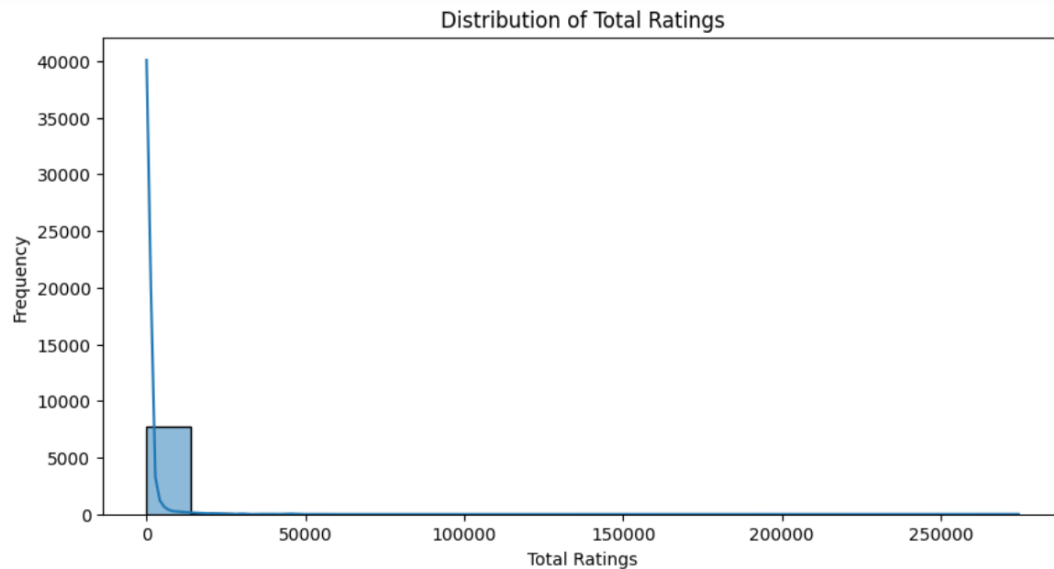
- The provided data suggests that "Women's Sneakers" is a popular product, particularly the variant with 23402 units sold last month, possibly due to its pricing, features, or marketing.
- Skechers, being the associated brand, likely contributes to the product's popularity and sales performance.
- Customer ratings and feedback are generally positive, indicating good product quality and customer satisfaction.
- Variation in pricing and offer percentages suggests dynamic pricing strategies aimed at maximizing sales and revenue.

Overall, the data provides insights into the sales performance, customer feedback, and brand presence of "Women's Sneakers," indicating a successful product line with strong market demand.

## UNIVARIATE ANALYSIS





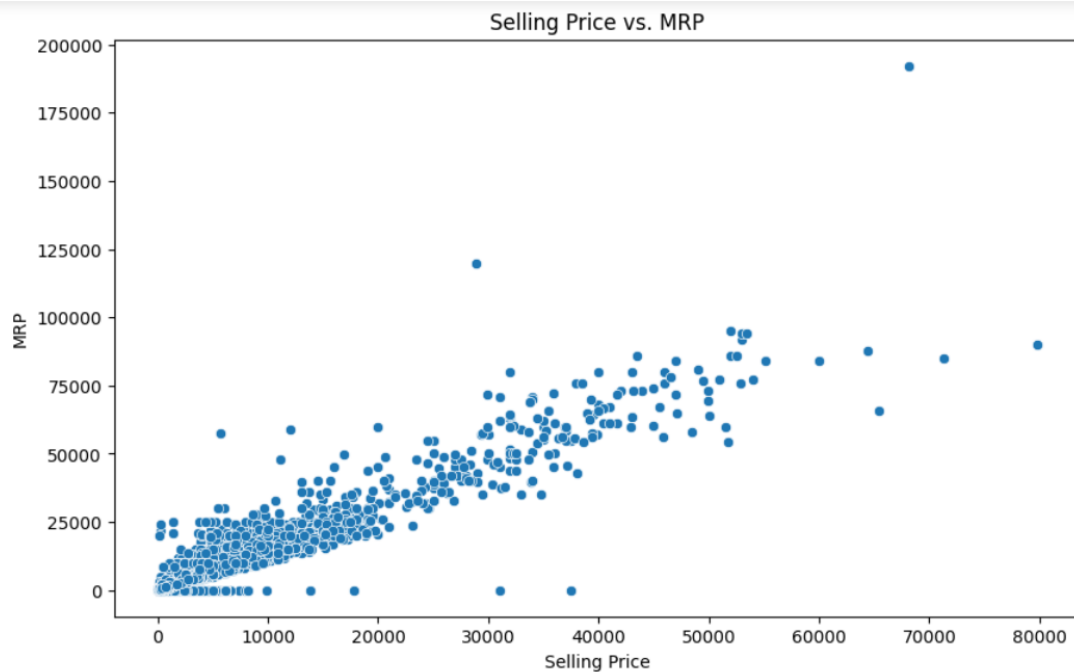


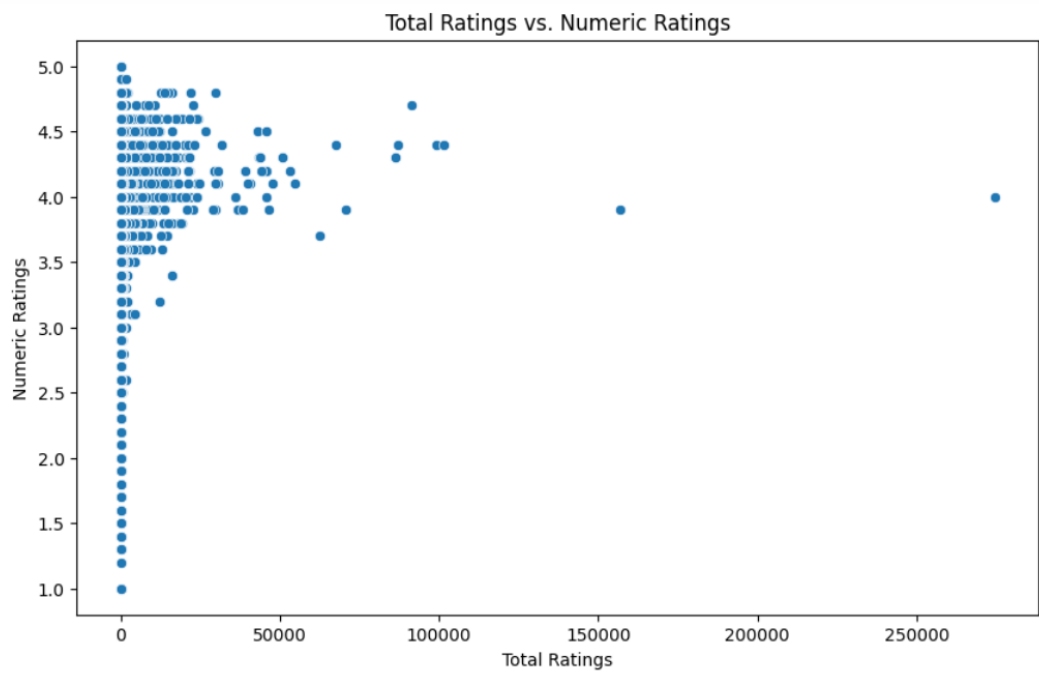
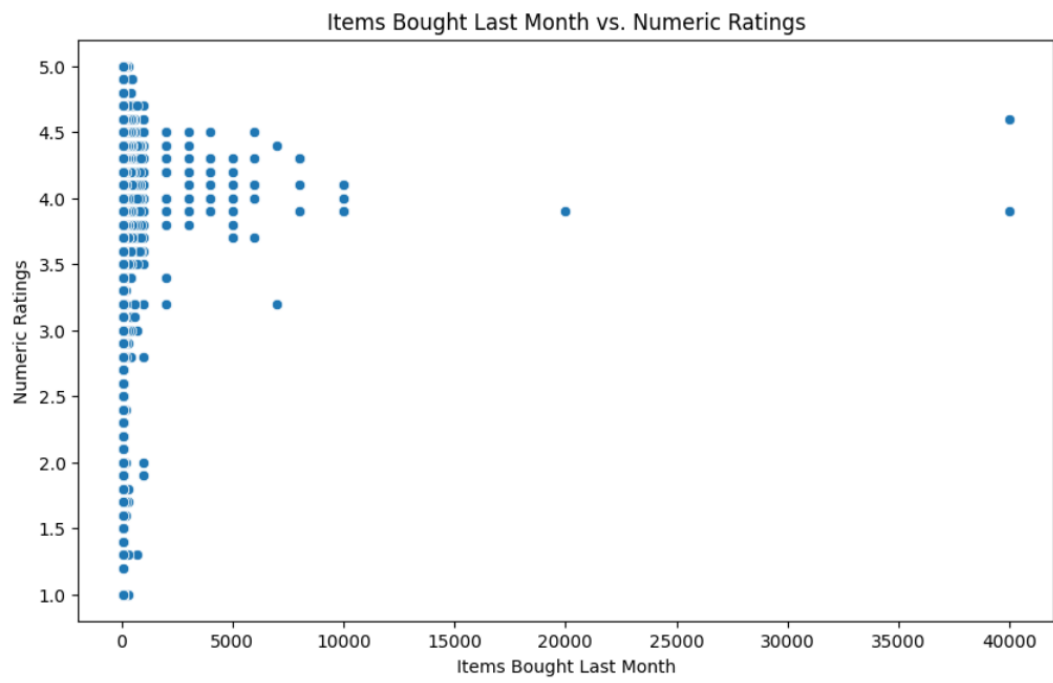
## INFERENCES

- **Selling Price:** The distribution of selling prices is diverse, with a concentration around lower values. There are products with a broad range of selling prices, including some high-priced items.
- **MRP:** The distribution of Maximum Retail Prices (MRP) is right-skewed. Most products have MRPs in the lower to mid-range, with a few products having higher MRPs.
- **Items Bought Last Month:** The distribution of items bought last month indicates variability in product popularity. Some products have lower demand (50 items), while others have higher demand (300 to 40,000 items).
- **Numeric Ratings:** The distribution of numeric ratings is right-skewed, with a concentration around 4.0. Most products have positive ratings, but there are a few with lower ratings.

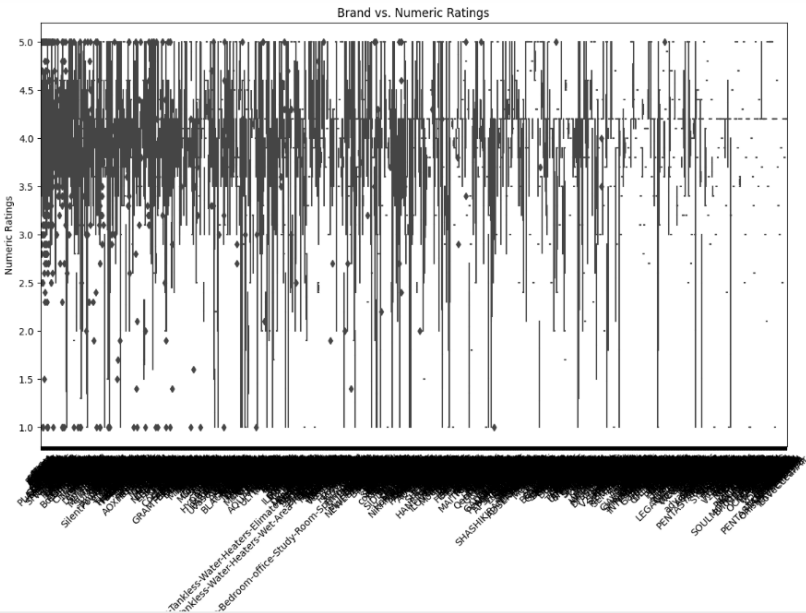
- **Total Ratings:** The distribution of total ratings is right-skewed, with a few products receiving a significantly higher number of ratings. Most products have a moderate number of total ratings.
- **Offer Percentage:** The distribution of offer percentages is right-skewed. Many products have relatively high offer percentages, indicating the presence of discounts.
- **General Inferences:** The dataset includes a mix of products with varied selling prices, MRPs, and popularity. Numeric ratings are generally positive, indicating overall customer satisfaction. Total ratings suggest that most products have a moderate number of ratings, but some products have attracted a large number of reviews. Offer percentages vary, with some products having notable discounts.

## BIVARIATE ANALYSIS









## INFERENCES

- Selling Price vs. MRP: Selling prices and MRPs show a strong positive linear relationship. The increase in selling price corresponds to a proportional increase in MRP, indicating consistent pricing strategies.
- Items Bought Last Month vs. Numeric Ratings: There is no clear linear relationship between the number of items bought last month and numeric ratings. The popularity of a product, as measured by items bought, does not necessarily correlate strongly with higher ratings.
- Total Ratings vs. Numeric Ratings: There is a positive linear relationship between total ratings and numeric ratings. Products with higher total ratings tend to have higher numeric ratings, suggesting that well-received products attract more ratings.
- Offer Percentage vs. Selling Price: There is no clear linear relationship between offer percentage and selling price. Higher-priced products do not consistently have higher or lower offer percentages.
- Offer Percentage vs. Numeric Ratings: There is no clear linear relationship between offer percentage and numeric ratings. The presence of discounts (offer percentage) does not strongly correlate with higher or lower numeric ratings.
- Brand vs. Numeric Ratings: Different brands have varying distributions of numeric ratings. Some brands have a higher median rating, indicating better overall customer satisfaction, while others have a wider range of ratings.
- General Inferences: The strong positive correlation between selling price and MRP suggests consistent pricing strategies across products. Popularity, as measured by the number of items bought last month, does not strongly correlate with higher numeric ratings. Higher total ratings tend to be associated with higher numeric ratings, indicating a positive relationship between overall popularity and customer satisfaction. The presence of discounts (offer percentage) does not consistently correlate with higher or lower ratings. Brands play a role in determining customer satisfaction, with some brands consistently receiving better ratings than others.

## LABEL ENCODING

Label encoding is a technique used in machine learning to convert categorical data, which is in the form of text labels, into numerical values.

### Label Encoding

```
In [9]: # Encode brand names
from sklearn.preprocessing import LabelEncoder

encoder = LabelEncoder()
df['Brand_Encoded'] = encoder.fit_transform(df['Brand'])
```

## CHANGING DATATYPES

### Changing DataTypes

```
In [11]: # Convert columns to the required data types
df['Selling Price'] = pd.to_numeric(df['Selling Price'], errors='coerce')
df['MRP'] = pd.to_numeric(df['MRP'], errors='coerce')
df['Items Bought Last Month'] = pd.to_numeric(df['Items Bought Last Month'], errors='coerce')
df['Total Ratings'] = pd.to_numeric(df['Total Ratings'], errors='coerce')
df['Offer%'] = pd.to_numeric(df['Offer%'], errors='coerce')
df['Numeric_Ratings'] = pd.to_numeric(df['Numeric_Ratings'], errors='coerce')
df['Brand'] = df['Brand'].astype(str)
```

## INFERENCES

1. **Conversion to Numeric Types:** Columns such as 'Selling Price', 'MRP', 'Items Bought Last Month', 'Total Ratings', 'Offer%', and 'Numeric\_Ratings' are being converted to numeric data types using `pd.to_numeric()`. This suggests that these columns likely contain numerical data that needs to be processed as such.
2. **Handling Errors:** The parameter `errors='coerce'` is used with `pd.to_numeric()`. This means that any non-numeric values encountered during the conversion will be replaced with NaN (Not a Number). This is a common practice to handle cases where the data may contain non-numeric values or errors.
3. **Data Cleaning:** By converting these columns to numeric data types and handling errors, we're essentially cleaning the data, making it ready for further analysis or modeling. This is an important step in data preprocessing to ensure the quality and consistency of the data.
4. **Categorical Data:** The 'Brand' column is being converted to string data type explicitly using `.astype(str)`. This suggests that 'Brand' might contain categorical data represented as strings. Converting it to string type ensures that it retains its categorical nature and is not inadvertently treated as numeric data.

## DATATYPE AFTER CONVERSION

```
In [14]: df.dtypes
Out[14]: Product_Name      object
        Selling Price    float64
        MRP              float64
        Items Bought Last Month float64
        Ratings          object
        Numeric_Ratings   float64
        Total Ratings     float64
        Brand            object
        Offer%           float64
        Brand_Encoded     int32
        dtype: object
```

## PROCEDURES AFTER EDA PERFORMANCE

### NLP( NATURAL LANGUAGE PROCESSING)

NLP stands for Natural Language Processing. It is a field of artificial intelligence (AI) that focuses on the interaction between computers and humans through natural language. NLP enables computers to understand, interpret, and generate human language in a way that is both meaningful and contextually relevant.

NLP has numerous applications across various domains, including:

- Information retrieval and search engines
- Text summarization and content extraction
- Sentiment analysis and opinion mining
- Question answering systems
- Speech recognition and synthesis
- Virtual assistants and chatbots
- Machine translation
- Text classification and topic modeling

## NLP

```
In [18]: from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import linear_kernel
import re
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
from nltk.stem import WordNetLemmatizer
```

```
In [19]: # Ensure NLTK resources are downloaded
import nltk
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('wordnet')
```

## NLP PREPROCESSING

In Natural Language Processing (NLP), preprocessing refers to the steps involved in cleaning and preparing textual data for further analysis or modeling. It is an essential step in NLP tasks because raw text data often contains noise, irrelevant information, and inconsistencies that can negatively impact the performance of NLP algorithms. Preprocessing aims to standardize and structure the text data in a way that makes it suitable for the specific NLP task at hand.

### NLP PreProcessing

```
[20]: # Define preprocess_text function
def preprocess_text(text):
    text = text.lower() # Convert text to lowercase
    text = re.sub(r'[^\w\s]', '', text) # Remove special characters and punctuation
    tokens = word_tokenize(text) # Tokenize the text
    stop_words = set(stopwords.words('english')) # Remove stopwords
    tokens = [word for word in tokens if word not in stop_words]
    lemmatizer = WordNetLemmatizer() # Lemmatize the tokens
    tokens = [lemmatizer.lemmatize(word) for word in tokens]

    processed_text = ' '.join(tokens) # Join tokens back into a string
    return processed_text
```

## RECOMMENDATION SYSTEM

A recommendation system in the context of Natural Language Processing (NLP) involves leveraging NLP techniques to provide personalized recommendations to users based on their text inputs, preferences, and behaviors. While recommendation systems are often associated

with other domains such as e-commerce or content streaming services, in NLP, recommendation systems can be applied to tasks such as recommending relevant documents, articles, or even textual responses in conversational interfaces.

## Recommendation System

```
[23]: # Function to recommend products based on brand and ratings
def recommend_products_(brand_name, min_rating=2.0, top_n=10000):
    filtered_df = df[(df['Brand'] == brand_name) & (df['Numeric_Ratings'] >= min_rating)]
    if filtered_df.empty:
        return "No products found for the given brand and rating criteria."
    else:
        recommended_products = filtered_df.sort_values(by='Numeric_Ratings', ascending=False).head(top_n)[['Product_Name', 'Numeric_Ratings']]
        return recommended_products

[24]: # User interaction
print("Welcome to the Product Recommender!")
print("Please enter the brand name:")
brand_name = input("Brand: ")
print("Please enter the minimum rating (e.g., 4.0):")
min_rating = float(input("Minimum Rating: "))

Welcome to the Product Recommender!
Please enter the brand name:
Brand: Puma
Please enter the minimum rating (e.g., 4.0):
Minimum Rating: 2.0
```

## RECOMMENDED PRODUCTS

### Recommended Products

```
[25]: # Recommend products
recommended_products = recommend_products_(brand_name, min_rating)
if isinstance(recommended_products, str):
    print(recommended_products)
else:
    print("\nRecommended Products:")
    print(recommended_products.to_string(index=False))

# Display all recommended products
print("\nAll Recommended Products:")
print(recommended_products)
```

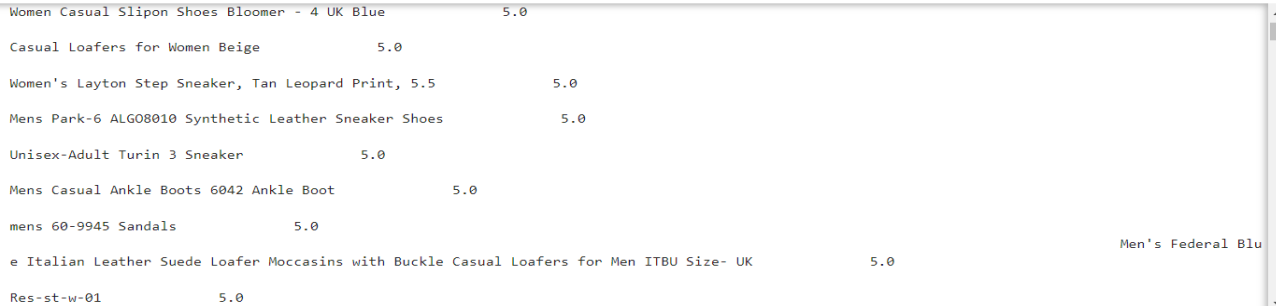
Recommended Products:

Product_Name	Numeric_Ratings
Womens Solid Slip-on Mules with Metal Accent Mule	5.0
Mens Adidas 5.0 M TRAROV/MYSBLU/SILVMT Sneaker - 12 UK (CK9529)	5.0
Premium Men's Comfortable Clogs Sandals with Adjustable Durable Back Strap for Men	5.0
tch Down Desert Boots Chukka Style Handmade Ankle Boot for Men Biking Hiking (SDSUCKBGJB)	5.0
Womens Frida Loafer Aubergine Pat Loafers	5.0
Men's Sneakers Fashion Casual Stylish Comfort Walking Gym Dark Grey Shoes	5.0
Girls Mules	5.0

## Recommended Products

```
[25]: # Recommend products
recommended_products = recommend_products_(brand_name, min_rating)
if isinstance(recommended_products, str):
    print(recommended_products)
else:
    print("\nRecommended Products:")
    print(recommended_products.to_string(index=False))

# Display all recommended products
print("\nAll Recommended Products:")
print(recommended_products)
```



Women Casual Slipon Shoes Bloomer - 4 UK Blue	5.0
Casual Loafers for Women Beige	5.0
Women's Layton Step Sneaker, Tan Leopard Print, 5.5	5.0
Mens Park-6 ALG08010 Synthetic Leather Sneaker Shoes	5.0
Unisex-Adult Turin 3 Sneaker	5.0
Mens Casual Ankle Boots 6042 Ankle Boot	5.0
mens 60-9945 Sandals	5.0
e Italian Leather Suede Loafer Moccasins with Buckle Casual Loafers for Men ITBU Size- UK	5.0
Res-st-w-01	5.0

## INFERENCES

1. **Recommendation Function:** There is a function `recommend_products_by_brand_and_ratings()` being called with parameters `brand_name` and `min_rating`. This function likely takes a brand name and a minimum rating as input and returns recommended products based on these criteria.
2. **Conditional Check:** The code includes a conditional check using `isinstance()` to determine if the result of the recommendation function is a string. This suggests that the function may return a string message in case of errors or if there are no recommended products meeting the specified criteria.
3. **Printing Recommended Products:** If the result of the recommendation function is a string (likely an error message or a notification of no recommendations), it is printed directly using `print()`. Otherwise, if the result is a DataFrame containing recommended products, it is printed as a formatted table using `to_string(index=False)`.
4. **Displaying All Recommended Products:** Additionally, all recommended products are printed without any conditional check. This part of the code is intended to display the recommended products regardless of whether they are in DataFrame format or a string message.
5. **Formatting:** The code seems to be focused on providing a user-friendly display of recommended products, first checking for any error messages, then printing the

recommended products as a table without indices, and finally displaying all recommended products irrespective of their format.

6. **Output:** The output consists of displaying all recommended products, including those that might have been filtered out based on the criteria.

## CHATBOT

Chatbots offer a range of benefits across various industries and applications. One of the primary advantages is their ability to provide round-the-clock customer support, enhancing accessibility and responsiveness for users. Unlike human agents, chatbots can handle multiple inquiries simultaneously, ensuring prompt assistance and reducing wait times. This 24/7 availability translates to improved customer satisfaction and loyalty as users can receive help whenever they need it.

Another benefit of chatbots is their scalability and cost-effectiveness. By automating routine tasks and inquiries, businesses can streamline operations and reduce the need for human resources. This scalability allows organizations to handle large volumes of interactions without significantly increasing operational costs, making chatbots a cost-effective solution for customer service and support.

Furthermore, chatbots contribute to improved efficiency by handling repetitive tasks and frequently asked questions, freeing up human agents to focus on more complex or value-added activities. This not only increases productivity but also ensures consistent and standardized responses across all user interactions, leading to a better overall customer experience.



## NOTES FOR PROJECT TEAM

<b>Original owner of data</b>	<b>Amazon</b>
<b>Data set information</b>	Web scraped from amazon website.
<b>Previous relevant journals used the data set</b>	It is a fresh dataset as we have web scraped it and then form a dataset
<b>Citation</b>	-
<b>Link to web page</b>	-