CSP 571 – Data Preparation and Analysis

Summer 2024

Amazon Product Review Analysis & Recommendation System

Satwika Sriram (A20563950)

Naga Sunith Appasani (A20552681)

TABLE OF CONTENTS

- Project Introduction
- Dataset Description
- Project Methodology
- Data Transformation
- Exploratory Data Analysis
- Sentiment Distribution
- Sentiment Analysis
- Predictive Modelling
- Recommendation System
- Conclusion

PROJECT INTRODUCTION

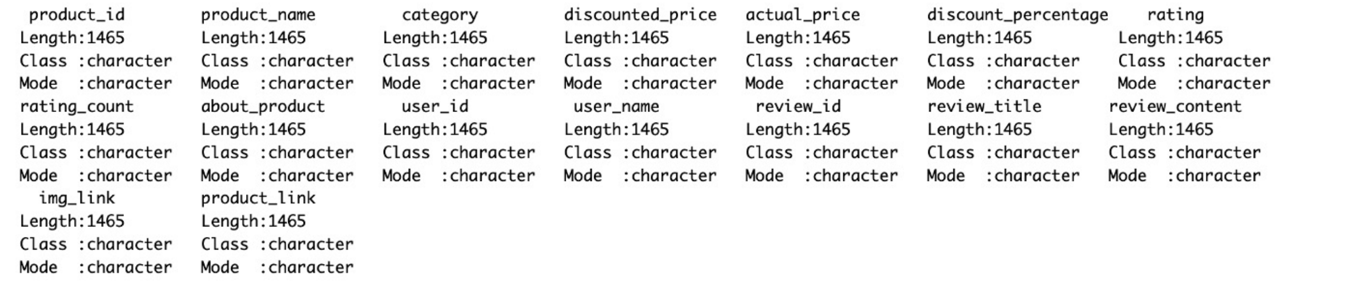
Project Objective: Develop and implement a robust analytical framework to parse and interpret extensive Amazon product review data. Enhance the shopping experience by providing insightful analysis and tailored recommendations.

Analytical Framework Development: Parse and interpret Amazon product reviews focusing on sentiment and key features. Identify key performance indicators such as sentiment trends and prevalent themes. Categorize reviews into positive, negative, or neutral sentiment classes. Utilize exploratory data analysis (EDA) and visualization methods to derive insights.

Predictive Modeling: Utilize machine learning techniques to classify sentiment and predict user preferences. Develop a recommendation engine to suggest products based on user reviews and ratings. Enhance decision-making for consumers by providing personalized product recommendations.

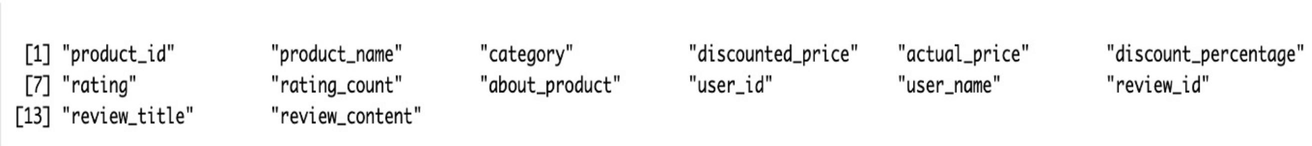
DATASET DESCRIPTION

- The dataset, "amazon.csv," used in this project, contains 1465 rows and 16 columns of Amazon product review data sourced from Kaggle. It includes various attributes such as product name, category, discounted_price, actual_price, rating, user_id, user_name, review_content, and other product characteristics. These features are leveraged to predict and recommend products to users based on their interests and past purchases.



```
  product_id        product_name         category         discounted_price   actual_price        discount_percentage    rating
 Length:1465        Length:1465        Length:1465        Length:1465        Length:1465        Length:1465         Length:1465
 Class :character   Class :character   Class :character   Class :character   Class :character   Class :character    Class :character
 Mode  :character   Mode  :character   Mode  :character   Mode  :character   Mode  :character   Mode  :character    Mode  :character
 rating_count       about_product        user_id           user_name           review_id        review_title       review_content
 Length:1465        Length:1465        Length:1465        Length:1465        Length:1465        Length:1465        Length:1465
 Class :character   Class :character   Class :character   Class :character   Class :character   Class :character   Class :character
 Mode  :character   Mode  :character   Mode  :character   Mode  :character   Mode  :character   Mode  :character   Mode  :character
   img_link         product_link
 Length:1465        Length:1465
 Class :character   Class :character
 Mode  :character   Mode  :character
```

Fig: Above depicts the summary of the Dataset



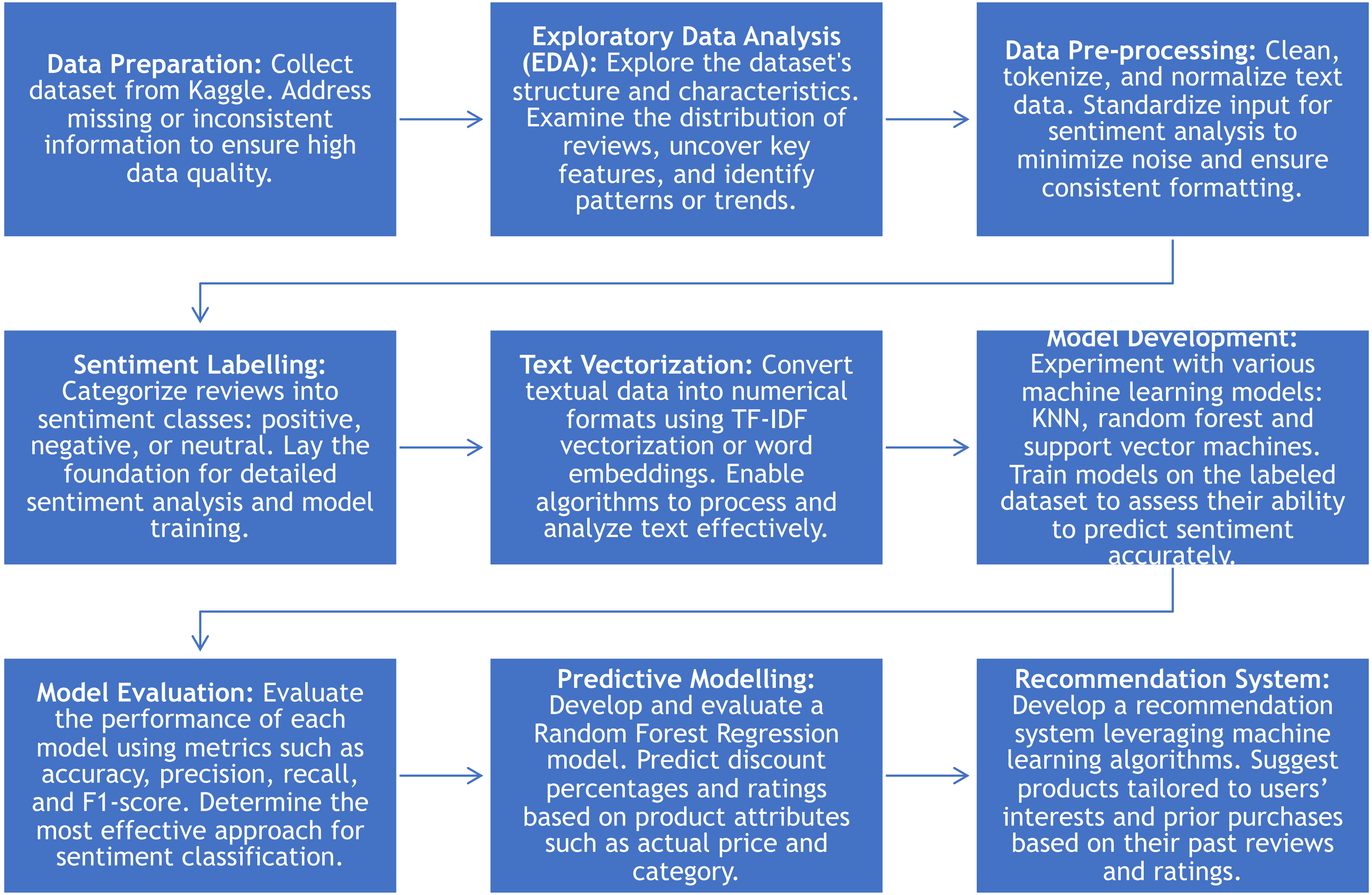
```
 [1] "product_id"        "product_name"     "category"         "discounted_price"  "actual_price"      "discount_percentage"
 [7] "rating"            "rating_count"     "about_product"    "user_id"           "user_name"         "review_id"
[13] "review_title"      "review_content"
```

Fig: Above depicts the columns in the Dataset.

- **Data Source:**

https://www.kaggle.com/datasets/karkavelrajaj/amazon- sales-dataset

PROJECT METHODOLOGY



Data Preparation: Collect dataset from Kaggle. Address missing or inconsistent information to ensure high data quality.

Exploratory Data Analysis (EDA): Explore the dataset's structure and characteristics. Examine the distribution of reviews, uncover key features, and identify patterns or trends.

Data Pre-processing: Clean, tokenize, and normalize text data. Standardize input for sentiment analysis to minimize noise and ensure consistent formatting.

Sentiment Labelling: Categorize reviews into sentiment classes: positive, negative, or neutral. Lay the foundation for detailed sentiment analysis and model training.

Text Vectorization: Convert textual data into numerical formats using TF-IDF vectorization or word embeddings. Enable algorithms to process and analyze text effectively.

Model Development: Experiment with various machine learning models: KNN, random forest and support vector machines. Train models on the labeled dataset to assess their ability to predict sentiment accurately.

Model Evaluation: Evaluate the performance of each model using metrics such as accuracy, precision, recall, and F1-score. Determine the most effective approach for sentiment classification.

Predictive Modelling: Develop and evaluate a Random Forest Regression model. Predict discount percentages and ratings based on product attributes such as actual price and category.

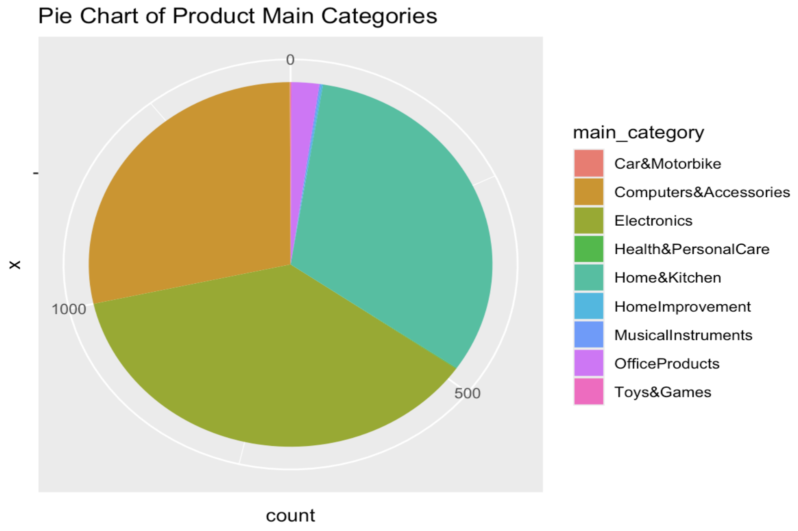
Recommendation System: Develop a recommendation system leveraging machine learning algorithms. Suggest products tailored to users’ interests and prior purchases based on their past reviews and ratings.

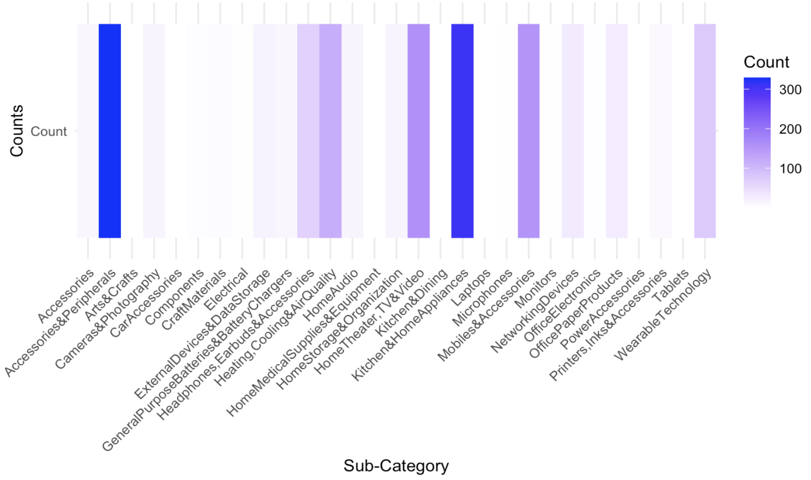
DATA TRANSFORMATION AND STANDARDIZATION

- **Handling Null Values:** Identified and deleted rows with missing values to maintain data integrity. Removed duplicate rows to ensure the accuracy of the dataset before proceeding with further analysis.
- **Standardization and Conversion:** Converted 'price' and 'rating' columns from character to numeric. Addressed inconsistent values in the 'rating' column by replacing them with the highest rating. Checked for missing values after data type conversions.
- **Category Refinement:** Split the 'category' column into 'main category' and 'sub-category' using gsub and substr functions to enhance data organization and granularity.
- **Column Splitting:** Processed 'review_id', 'review_title', 'review_content', and 'user_id' by splitting them based on specific delimiters using sapply and strsplit functions. This was done to facilitate better analysis by breaking down complex data.
- **Descriptive Ratings:** Introduced a new column 'rating_score' to transform numerical ratings into qualitative categories. Provided a more descriptive understanding of ratings.
- **Data Re-integration:** Used cbind to reintegrate newly created columns back into the original dataframe. Ensured that the dataframe remains complete and ready for further analysis, enhancing overall usability.

EXPLORATORY DATA ANALYSIS

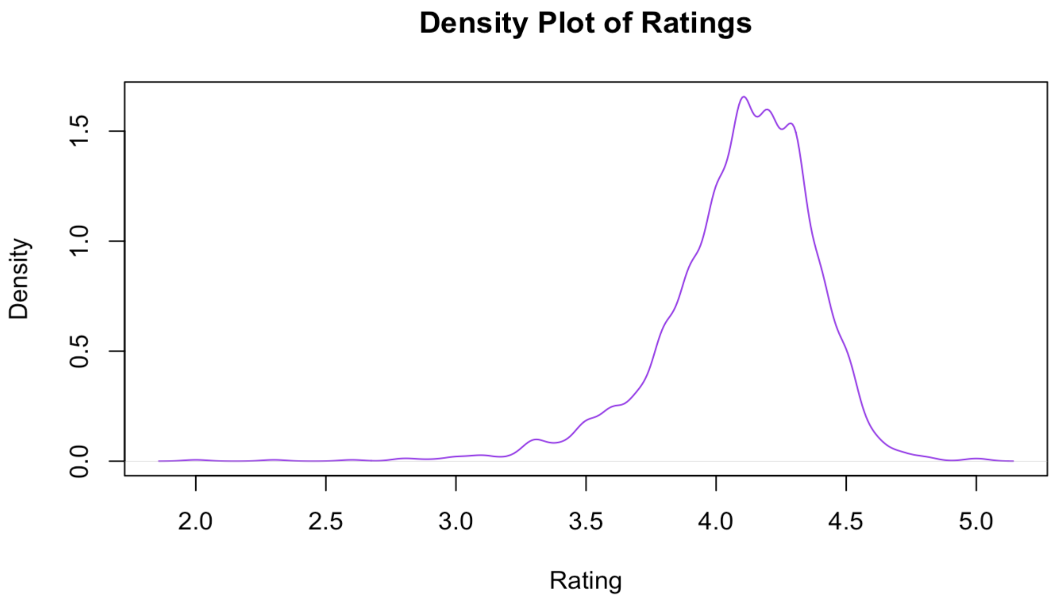
DISTRIBUTION OF PRODUCTS BY CATEGORY AND SUBCATEGORY





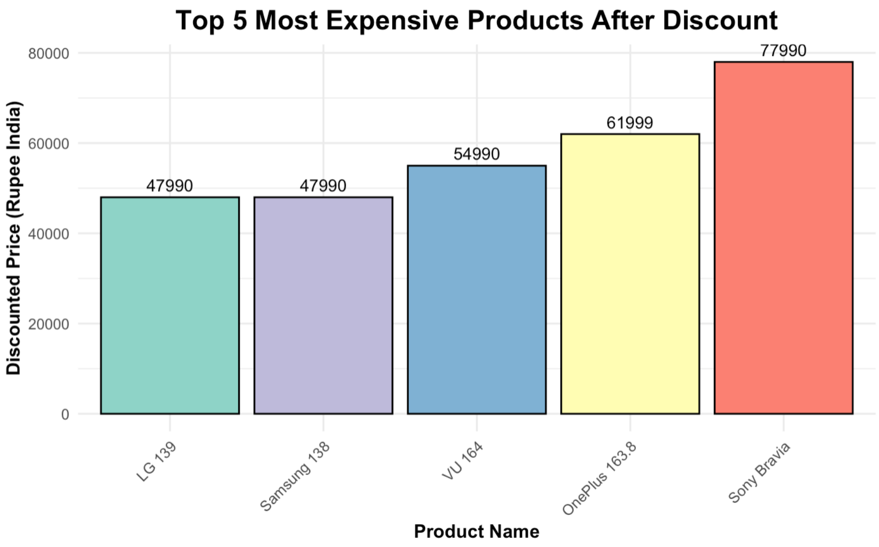
- The pie chart illustrates the distribution of product main categories, highlighting Electronics and Home & Kitchen as the most prominent categories, while Car & Motorbike, Health & Personal Care, and Toys & Games are minimally represented, indicating a focus on popular product areas with potential for further exploration in niche markets.
- The bar chart shows the count of product reviews across various sub-categories, with Accessories, Laptops, and Mobile Phones having the highest counts, indicating their popularity among users, while other sub-categories have relatively lower review counts, suggesting less user engagement or smaller market presence.

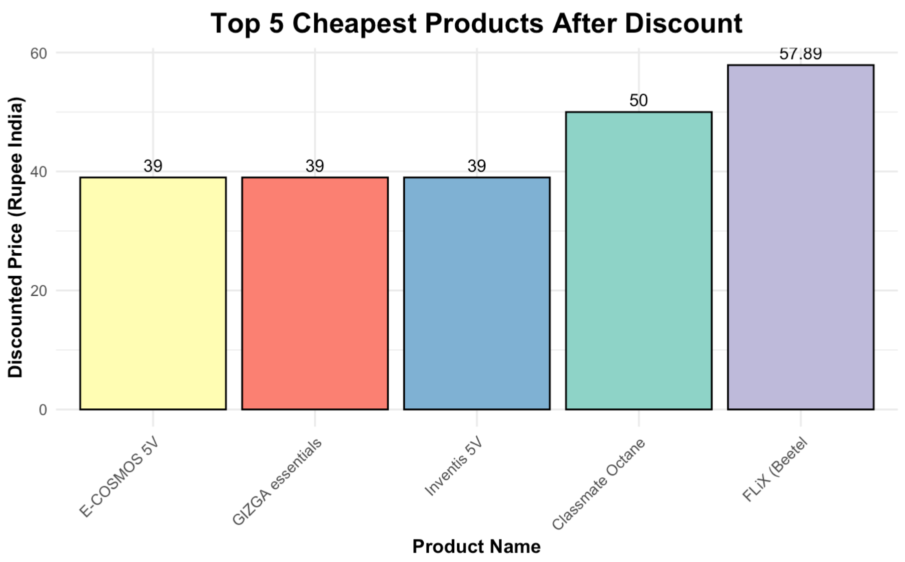
DISTRIBUTION OF PRODUCTS BY RATING



- It is observed that the density plot indicates the distribution of ratings, highlighting that most items receive high ratings with some variability. Majority of ratings are concentrated around a central value, specifically near 4.1. This suggests that most items are rated relatively highly, with 4.1 being a common rating.

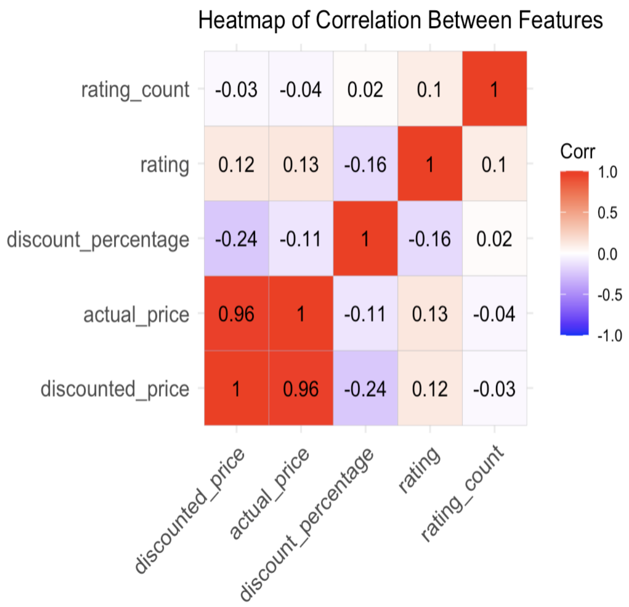
DISTRIBUTION OF PRODUCTS AFTER DISCOUNT



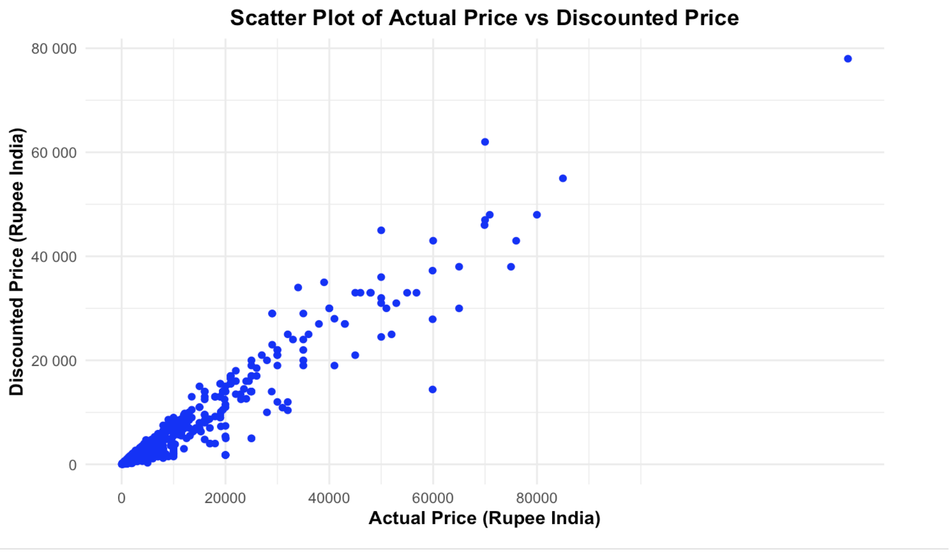


- The left bar chart displays the top 5 most expensive products after discount, with Sony Bravia being the most expensive at 77,990 Rupees, followed by OnePlus 163.8 at 61,999 Rupees, indicating a significant price range among high-end discounted products.
- The right bar chart shows the top 5 cheapest products after discount, with E-COSMOS 5V, GIZGA essentials, and Invertis 5V all priced at 39 Rupees, while FLX (Beetel) is the most expensive among the cheapest products at 57.89 Rupees, highlighting the affordable options available.

CORRELATION BETWEEN FEATURES

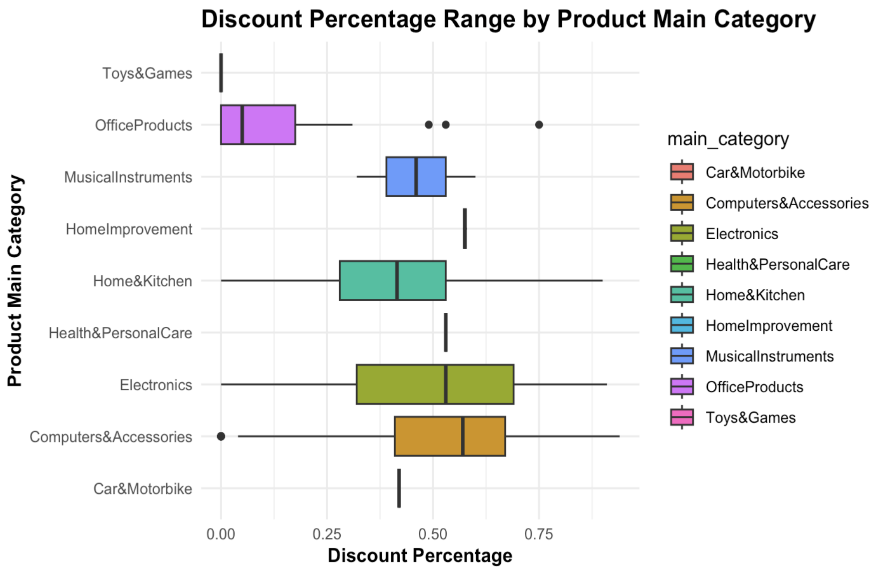

Heatmap of Correlation Between Features

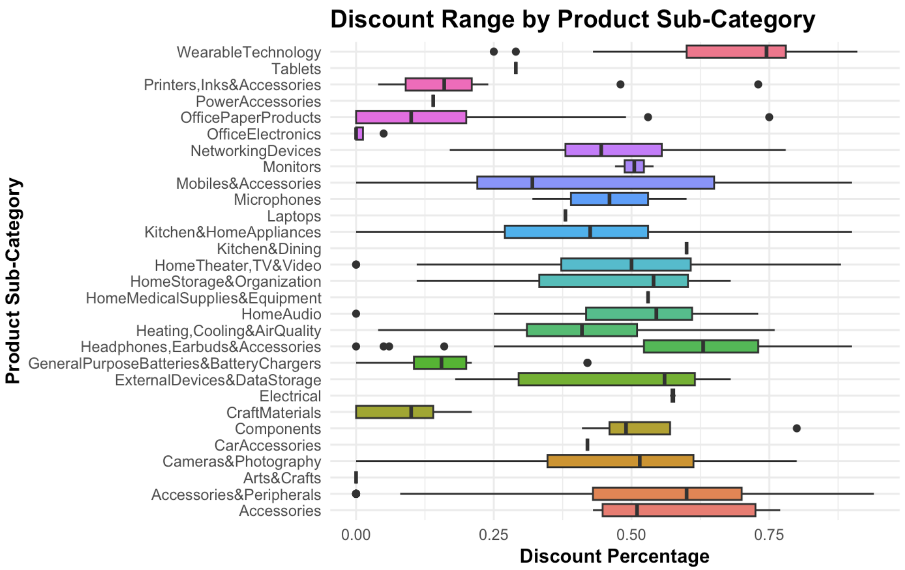
	discounted_price	actual_price	discount_percentage	rating	rating_count
rating_count	-0.03	-0.04	0.02	0.1	1
rating	0.12	0.13	-0.16	1	0.1
discount_percentage	-0.24	-0.11	1	-0.16	0.02
actual_price	0.96	1	-0.11	0.13	-0.04
discounted_price	1	0.96	-0.24	0.12	-0.03



- The correlation matrix reveals that the discounted price and actual price have a strong positive relationship, indicating they tend to increase together. The other variables show mostly weak correlations.
- In the scatterplot, we can observe that the higher the price of the product, the greater the discount.

DISTRIBUTION OF DISCOUNT PERCENTAGE RANGE

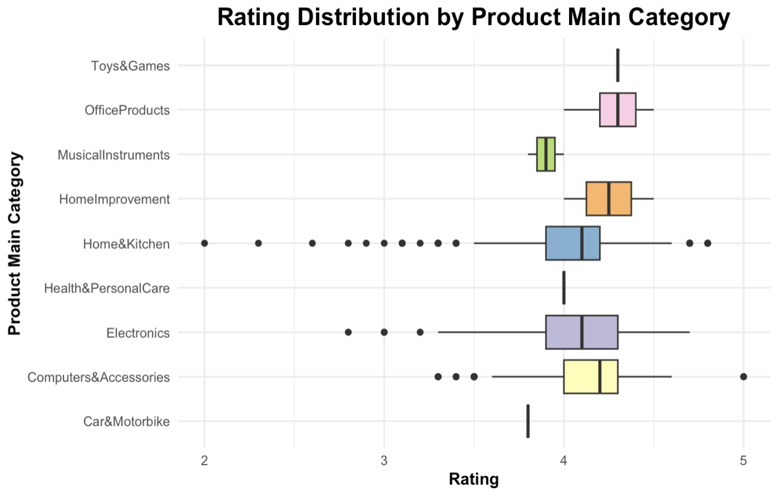


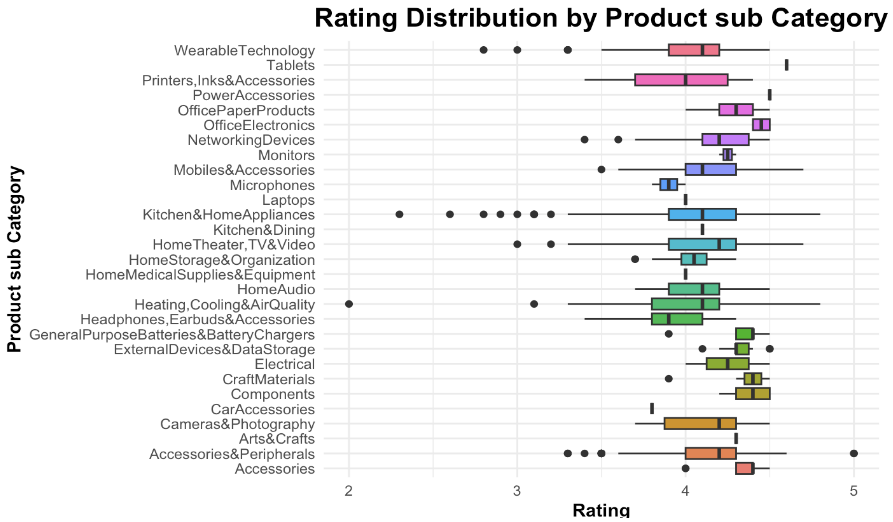


- The left box plot illustrates the discount percentage range by product main category, showing significant variability within categories like Office Products and Electronics, while categories like Car & Motorbike and Home Improvement have more consistent and narrower discount ranges. This indicates differing discount strategies across product categories.

- The right box plot illustrates the discount percentage range by product sub-category, highlighting significant variability in discount strategies within sub-categories such as Tablets and Wearable Technology, while sub-categories like Accessories and Office Paper Products have more consistent discount ranges. This variability indicates different discount practices and potential opportunities for deeper discounts in certain sub-categories.

DISTRIBUTION OF RATING FOR ALL PRODUCTS

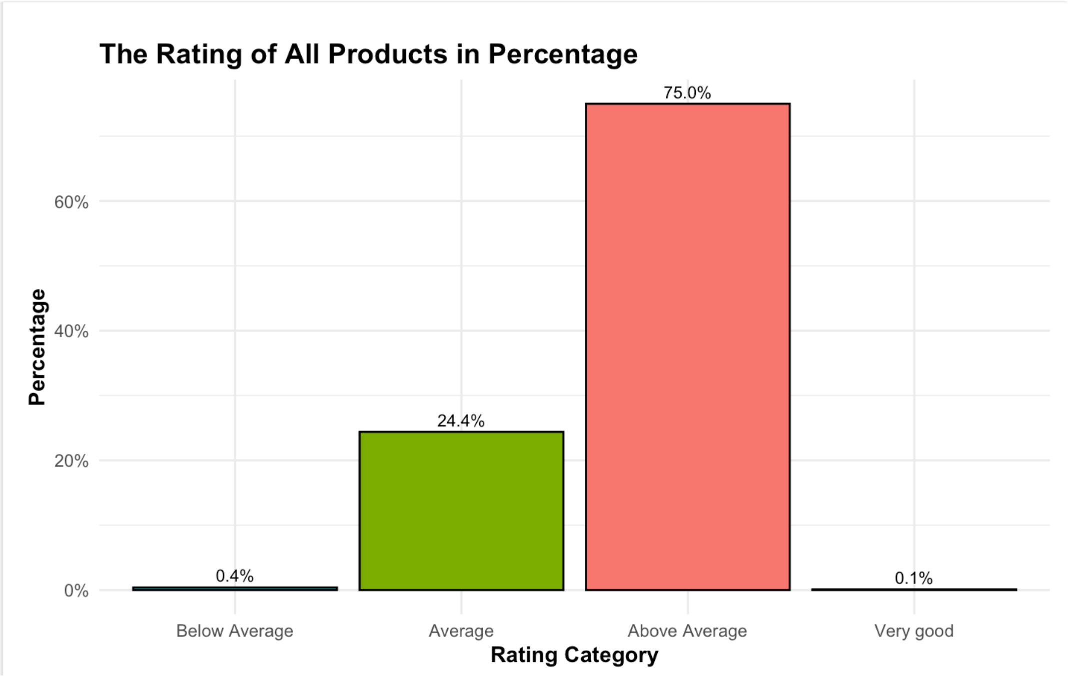




- The left box plot shows the rating distribution by product main category, with categories like Musical Instruments, Office Products, and Home & Kitchen having generally higher and more consistent ratings, while categories like Car & Motorbike and Health & Personal Care show wider variability in ratings. This indicates varying levels of customer satisfaction across different product categories.

- The right box plot illustrates the rating distribution by product sub-category, showing that sub-categories like Printers, Inks & Accessories and Power Accessories have high and consistent ratings, while categories like Car Accessories and External Devices & Data Storage exhibit greater variability. This highlights differences in customer satisfaction across various product sub-categories.

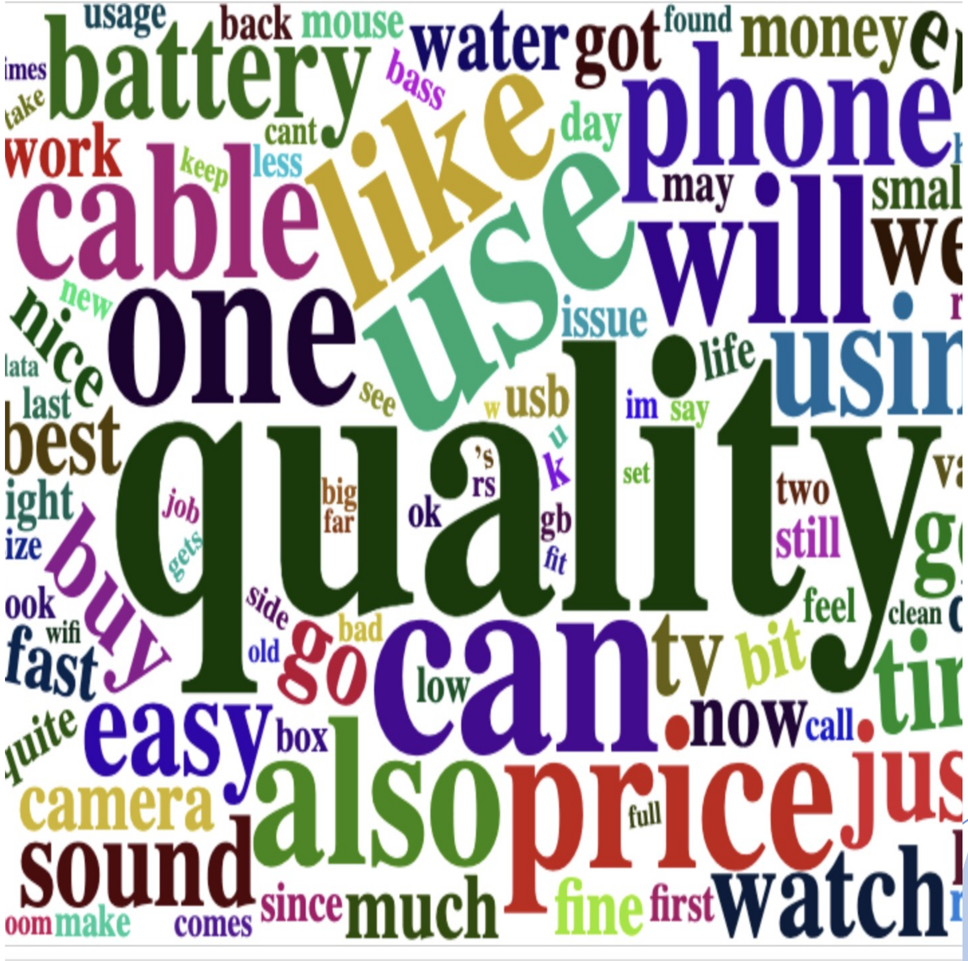
DISTRIBUTION OF RATING SCORE VALUES

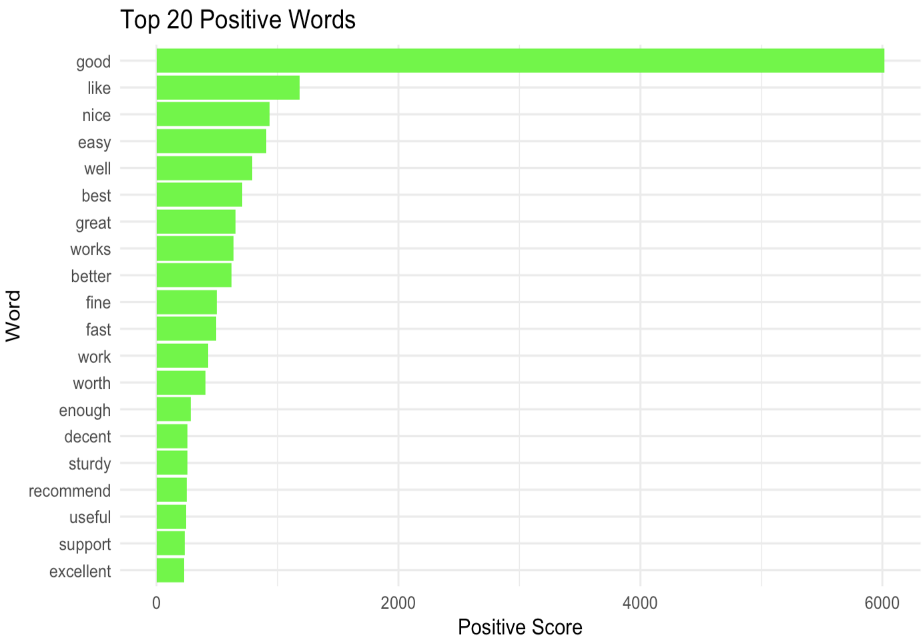


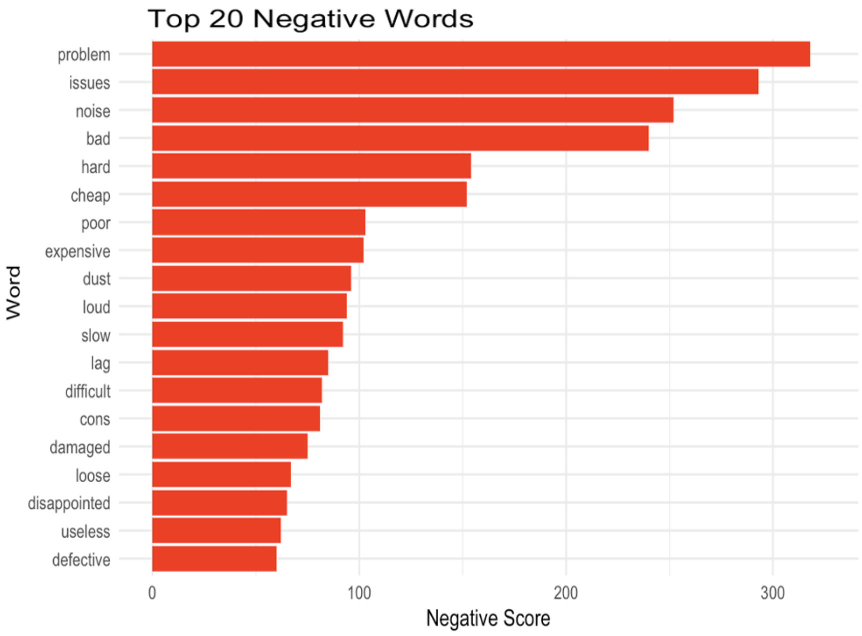
▶ The bar chart illustrates the distribution of product ratings, showing that 75% of all products fall into the "Above Average" category, indicating a generally high level of customer satisfaction. In contrast, only 0.4% and 0.1% of products are rated as "Below Average" and "Very Good," respectively, while 24.4% of products are rated as "Average." This distribution suggests that most products meet or exceed customer expectations, with a significant majority being rated favourably.

DISTRIBUTION OF SENTIMENTS

- Before diving into sentiment analysis, we conducted a word frequency analysis to identify the most commonly used words and phrases in the reviews. This step helped us understand the language customers frequently use in their feedback. To visualize these findings, we created word clouds, which offer an intuitive representation of the prominent terms found in the reviews. This approach provided valuable insights into the recurring themes and expressions used by customers, setting the stage for a more in-depth sentiment analysis.

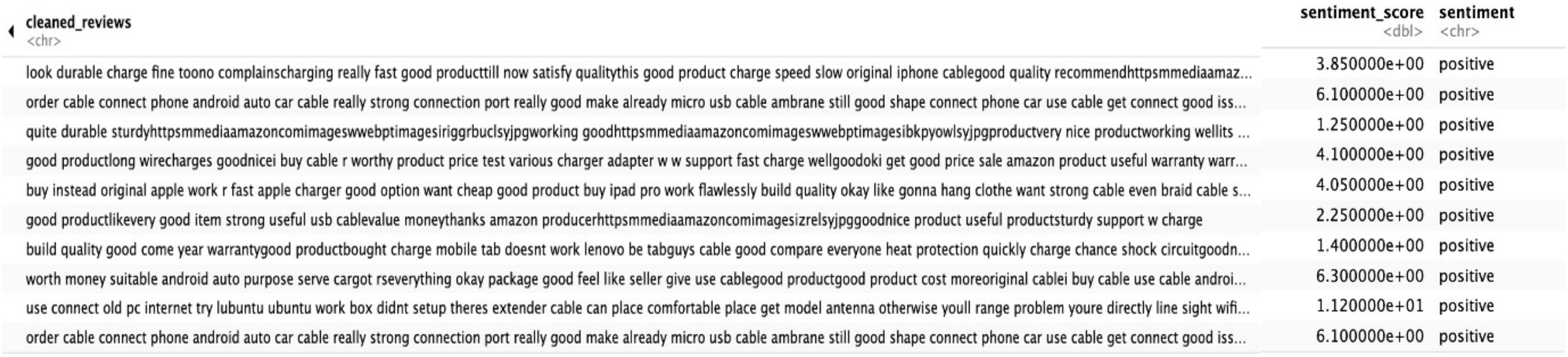






- To gain insights into the sentiment characteristics essential for our prediction engine, we visualized the distribution of both positive and negative sentiments within our dataset. This visualization aids in understanding overall sentiment trends and identifying potential imbalances across various product categories. By examining the proportion of each sentiment type, we can determine if one sentiment predominates and evaluate any skewness in the dataset. This information is crucial for ensuring our analysis is balanced and accurately represents the underlying data.

SENTIMENT ANALYSIS



cleaned_reviews <chr>	sentiment_score <dbl>	sentiment <chr>
look durable charge fine toono complainscharging really fast good producttill now satisfy qualitythis good product charge speed slow original iphone cablegood quality recommendhttpsmmediaamaz...	3.850000e+00	positive
order cable connect phone android auto car cable really strong connection port really good make already micro usb cable ambrane still good shape connect phone car use cable get connect good iss...	6.100000e+00	positive
quite durable sturdyhttpsmmediaamazoncomimageswwebptimagesiriggrbuclsyjpgworking goodhttpsmmediaamazoncomimageswwebptimagesibkpyowlsyjpgproductvery nice productworking wellits ...	1.250000e+00	positive
good productlong wirecharges goodnicei buy cable r worthy product price test various charger adapter w w support fast charge wellgoodoki get good price sale amazon product useful warranty warr...	4.100000e+00	positive
buy instead original apple work r fast apple charger good option want cheap good product buy ipad pro work flawlessly build quality okay like gonna hang clothe want strong cable even braid cable s...	4.050000e+00	positive
good productlikevery good item strong useful usb cablevalue moneythanks amazon producerhttpsmmediaamazoncomimagesizrelsyjpggoodnice product useful productsturdy support w charge	2.250000e+00	positive
build quality good come year warrantygood productbought charge mobile tab doesnt work lenovo be tabguys cable good compare everyone heat protection quickly charge chance shock circuitgoodn...	1.400000e+00	positive
worth money suitable android auto purpose serve cargot rseverything okay package good feel like seller give use cablegood productgood product cost moreoriginal cablei buy cable use cable androi...	6.300000e+00	positive
use connect old pc internet try lubuntu ubuntu work box didnt setup theres extender cable can place comfortable place get model antenna otherwise youll range problem youre directly line sight wifi...	1.120000e+01	positive
order cable connect phone android auto car cable really strong connection port really good make already micro usb cable ambrane still good shape connect phone car use cable get connect good iss...	6.100000e+00	positive

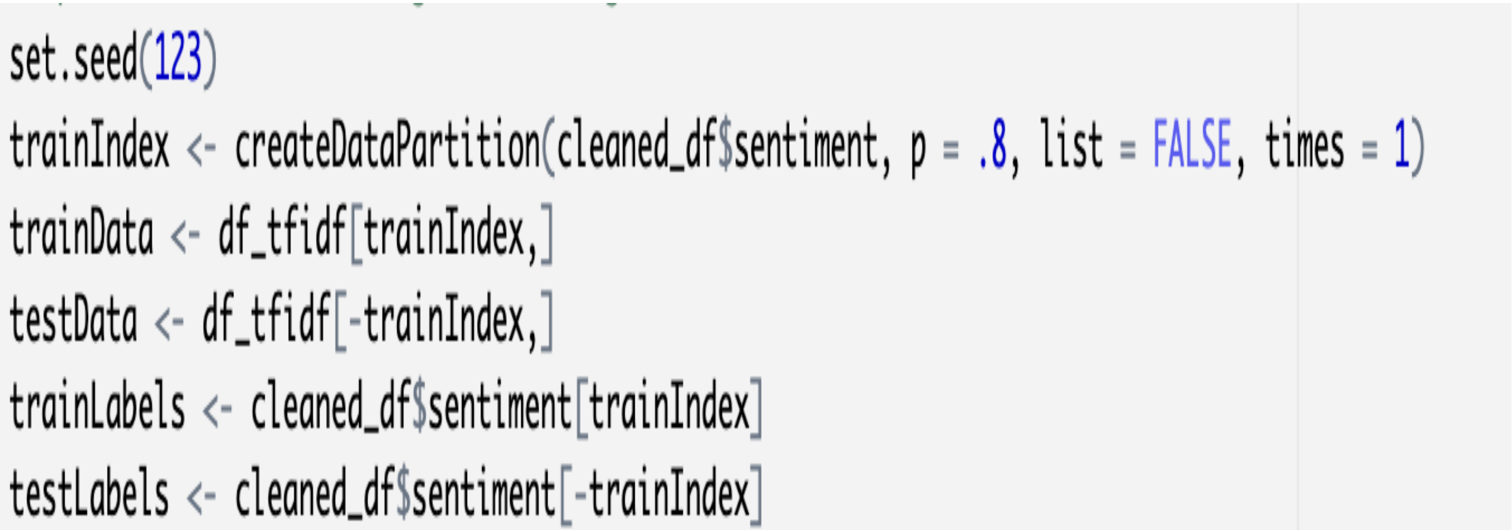
Fig: Cleaned reviews and respective sentiment score labels.

- Our final goal is to conduct sentiment analysis to gain insights into ratings and reviews across products and categories, and to build a predictive model. After performing Exploratory Data Analysis (EDA), we preprocess the dataset by cleaning and transforming the text data. This involves text cleaning (removing punctuation, digits, special characters, and converting text to lowercase), stopword removal, and lemmatization or stemming to standardize words. We also conduct feature engineering by creating additional variables like word count, average word length, and sentiment scores using libraries such as syuzhet. Next, we label each review's sentiment as positive, negative, or neutral, either manually or using machine learning models. Post-labelling, we vectorize the text data using techniques like Bag-of-Words, TF-IDF, and word embeddings to convert text into numerical representations. Finally, we select appropriate machine learning based on the complexity of the text features and the desired prediction accuracy, to build the sentiment analysis model.

MACHINE LEARNING MODELS

MODEL TRAINING

- After selecting the model, we train it using the vectorized text data and corresponding sentiment labels. During this phase, the model learns to recognize patterns in text features that correspond to positive, negative, and neutral sentiments. The training algorithm fine-tunes the model parameters to minimize prediction errors and improve accuracy. We split the data into an 80:20 ratio for training and testing, ensuring robust evaluation of the model's performance.



```
set.seed(123)
trainIndex <- createDataPartition(cleaned_df$sentiment, p = .8, list = FALSE, times = 1)
trainData <- df_tfidf[trainIndex,]
testData <- df_tfidf[-trainIndex,]
trainLabels <- cleaned_df$sentiment[trainIndex]
testLabels <- cleaned_df$sentiment[-trainIndex]
```

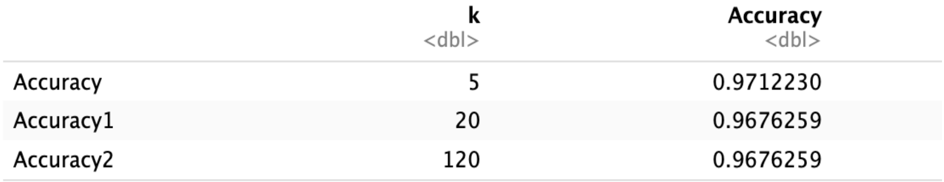
K-NEAREST NEIGHBORS

- The KNN algorithm operates based on similarity or distance metrics, such as Euclidean, Manhattan, or Minkowski distance.

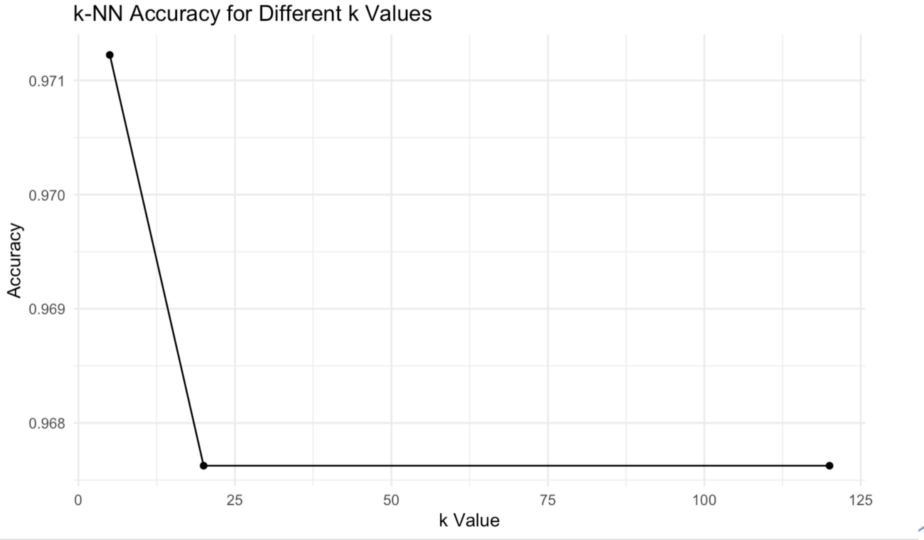
- To classify a new data point, the algorithm calculates the distances from this point to all others in the dataset. It then selects the K closest data points, assigning the new data point to the most common class among these K neighbors.

- The only hyperparameter for this model is K, representing the number of neighbors considered.

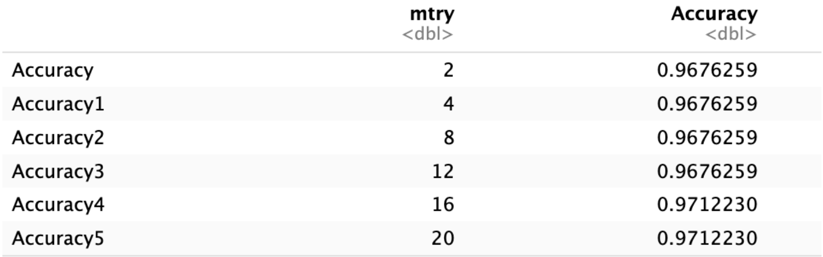
- The highest accuracy was achieved with K=5, as demonstrated by plotting the results.



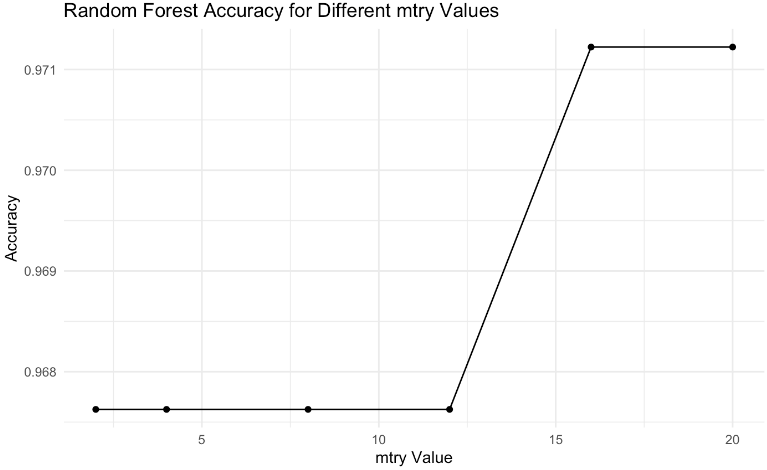
	k <dbl>	Accuracy <dbl>
Accuracy	5	0.9712230
Accuracy1	20	0.9676259
Accuracy2	120	0.9676259



RANDOM FOREST



	mtry	Accuracy
	<dbl>	<dbl>
Accuracy	2	0.9676259
Accuracy1	4	0.9676259
Accuracy2	8	0.9676259
Accuracy3	12	0.9676259
Accuracy4	16	0.9712230
Accuracy5	20	0.9712230

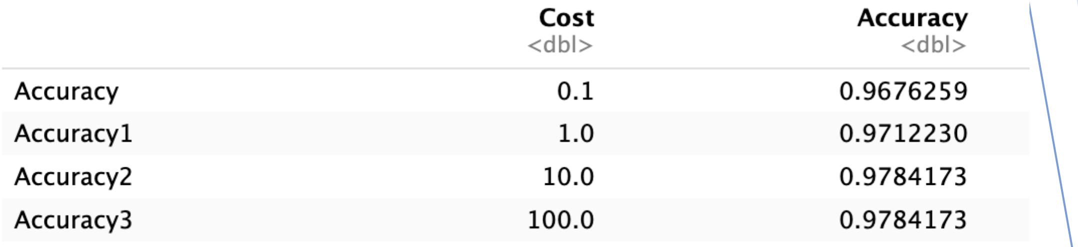


Random Forest Accuracy for Different mtry Values

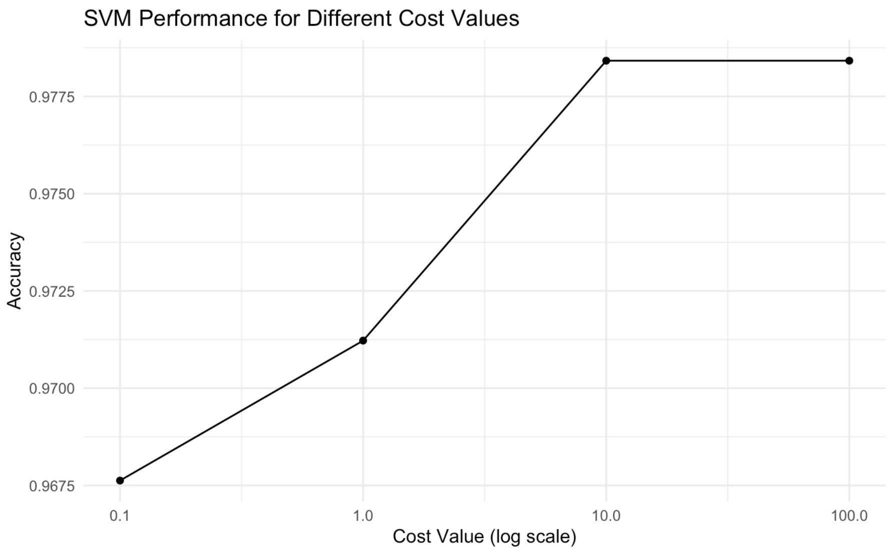
- ▶ This algorithm constructs multiple decision trees and combines them to produce more accurate and stable predictions.
- ▶ Random Forest begins by generating several bootstrap samples from the original dataset, where each sample is created by randomly selecting data points with replacement.
- ▶ At each decision point within the trees, a random subset of features is chosen, introducing randomness that helps to reduce correlation between individual trees.
- ▶ This algorithm has achieved an accuracy of 97.12%.

SUPPORT VECTOR MACHINE (SVM)

- SVM constructs a hyperplane in a high-dimensional space to separate different classes with the maximum margin. The nearest data points to the hyperplane, called support vectors, are essential in defining the hyperplane and the decision boundaries.
- SVM aims to maximize the separation between classes, thereby minimizing classification errors.
- This algorithm has achieved a maximum accuracy of 97.84%.



	Cost	Accuracy
	<dbl>	<dbl>
Accuracy	0.1	0.9676259
Accuracy1	1.0	0.9712230
Accuracy2	10.0	0.9784173
Accuracy3	100.0	0.9784173



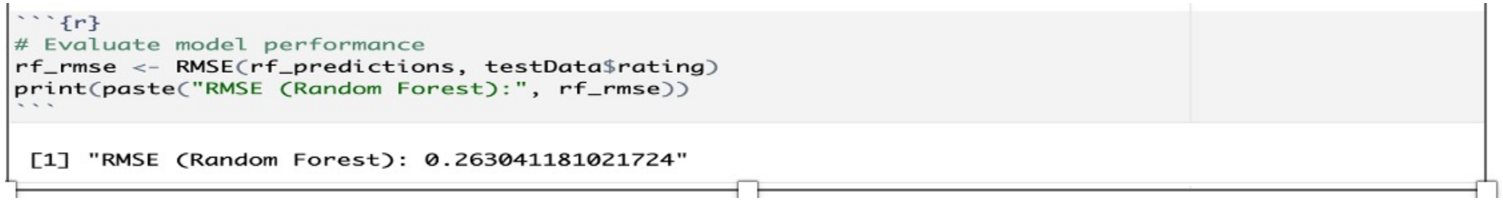
MODEL COMPARISON

- Evaluation of the performance of three models — SVM, Random Forest, and K-NN—was conducted using a dataset with three classes: negative, neutral, and positive. The comparison included overall accuracy, confidence intervals, and class-specific statistics to identify each model's strengths and weaknesses.
- **SVM Model:** Demonstrated high overall accuracy. It excelled in correctly identifying positive classes with perfect sensitivity and negative predictive value. However, it had lower sensitivity for the negative class.
- **Random Forest Model:** Showed slightly lower overall accuracy compared to SVM. It achieved perfect specificity and positive predictive value for the negative class but struggled with sensitivity for the same class, indicating a tendency to correctly identify positives but miss some negative instances.
- **k-NN Model:** Exhibited the lowest overall accuracy among the three models. While it showed perfect specificity, it failed to identify any negative instances (sensitivity = 0.0000), resulting in a kappa value of 0, indicating no agreement between the predicted and actual classes beyond chance.

PREDICTIVE MODELLING

PREDICTING RATING

- We conducted a prediction analysis of product ratings using a Random Forest model. The process began by selecting relevant features from our dataset, specifically focusing on discounted price, actual price, rating count, and rating.
- The Random Forest model was trained to predict product ratings based on the selected features. After training, the model's predictions were tested on the test set, and its performance was evaluated using Root Mean Squared Error (RMSE) to determine the accuracy of the predictions.



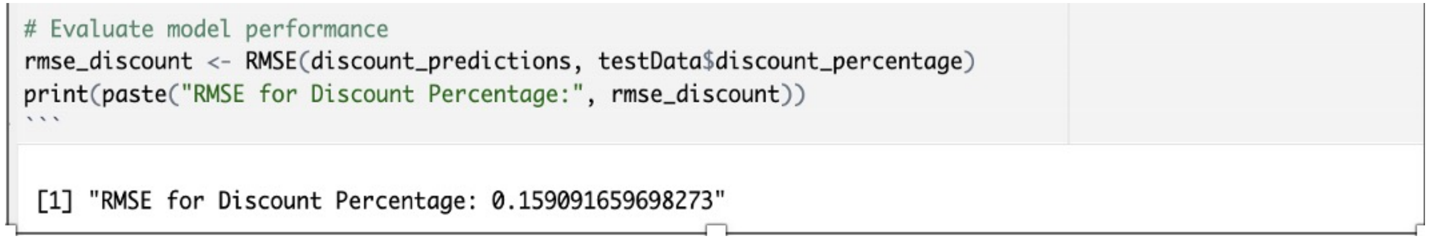
~~~
```{r}
# Evaluate model performance
rf_rmse <- RMSE(rf_predictions, testData$rating)
print(paste("RMSE (Random Forest):", rf_rmse))
```
~~~

```
[1] "RMSE (Random Forest): 0.263041181021724"
```

- The Random Forest model achieved a Root Mean Squared Error (RMSE) of 0.26 when predicting product ratings. This metric indicates the average deviation of the predicted ratings from the actual ratings in the test set. A lower RMSE value suggests better predictive accuracy. The achieved RMSE demonstrates that the model is relatively effective at predicting ratings, though further tuning and validation could potentially improve its performance.

PREDICTING DISCOUNT PERCENTAGE

- We carried out a prediction analysis of discount percentages using a Random Forest model. The process started by selecting pertinent features from our dataset, specifically focusing on actual price, discount percentage, and category.
- The Random Forest model was then trained to predict discount percentages based on these selected features. Post-training, we tested the model's predictions on the test set, evaluating its performance using Root Mean Squared Error (RMSE) to measure prediction accuracy. This analysis sheds light on the key factors influencing discount percentages and demonstrates the Random Forest model's effectiveness in making precise predictions.



~~~
# Evaluate model performance
rmse_discount <- RMSE(discount_predictions, testData$discount_percentage)
print(paste("RMSE for Discount Percentage:", rmse_discount))
```

 [1] "RMSE for Discount Percentage: 0.159091659698273"
~~~

- An RMSE of 0.159 suggests that the model’s predictions are relatively accurate. A lower RMSE generally reflects better model performance. Thus the model provides valuable insights for optimising discount strategies and can be integrated into business decision-making processes.

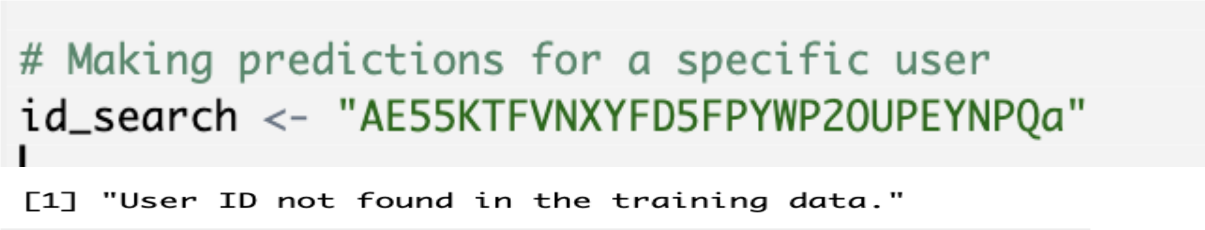
RECOMMENDATION SYSTEM

RECOMMENDATION SYSTEM OVERVIEW

- **Purpose:** Serves as a valuable tool for companies to gain deeper customer insights, elevate user engagement and satisfaction, and drive revenue growth through strategic product recommendations.
- **User-Based Collaborative Filtering (UBCF):**
- **Approach:** A recommendation approach that predicts a user's interest in items based on the preferences of similar users.
- **Assumption:** Operates under the assumption that users who agreed on item preferences in the past will continue to agree in the future.
- **Mechanism:** Involves measuring the similarity between users' ratings and leveraging this information to predict ratings for items not yet rated by a user.
- **Benefit:** Helps in identifying niche products and preferences that may not be immediately apparent, leading to more diverse and comprehensive recommendations.

- **Implementation Details:**
- **Data Preparation:** Begins by preparing the data, including converting a user-product rating dataset into a format suitable for the recommender lab package.
- **Data Splitting:** The dataset is then split into training and testing sets to evaluate model performance.
- **Model Training:** Uses the training data to train a UBCF model, predicting user preferences based on the ratings of similar users.
- **Personalized Recommendations:**
  - For a specific user, the model identifies their rated items.
  - Generates personalized product recommendations tailored to users' interests and prior purchases based on their past reviews and ratings.
  - Retrieves and displays the recommended products along with their details if the user has provided ratings.
  - Continuously updates and refines recommendations as new user data becomes available, ensuring recommendations remain relevant and up-to-date.

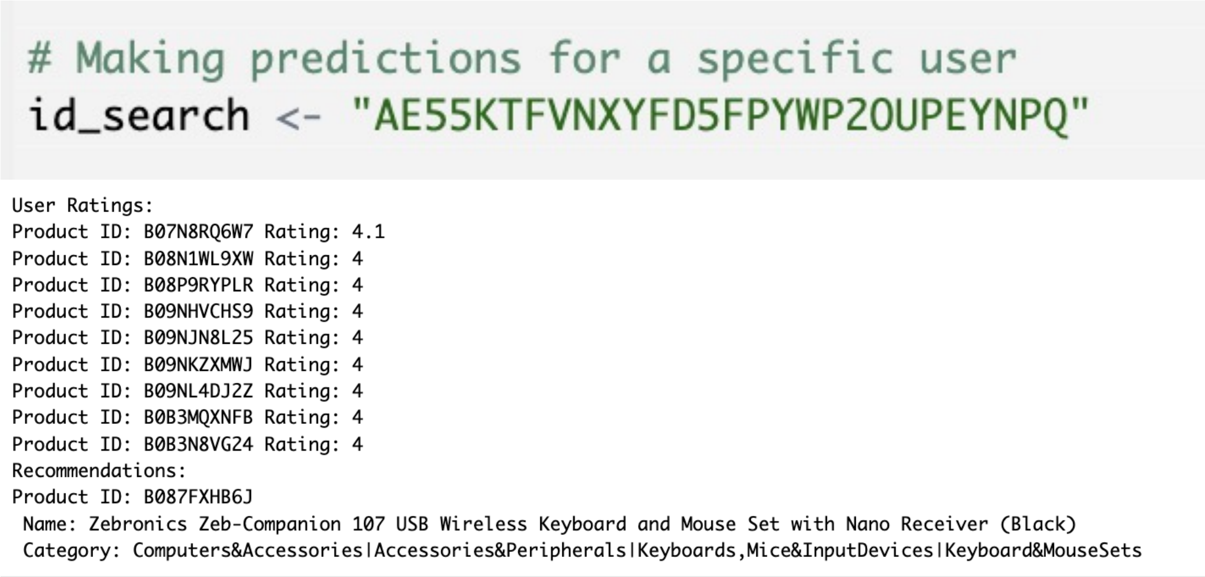
- **Recommendations for invalid user:**



```
# Making predictions for a specific user
id_search <- "AE55KTFVNXYFD5FPYWP2OUPEYNPQa"
```

```
[1] "User ID not found in the training data."
```

- **Recommendations for valid user based on past purchase history:**



```
# Making predictions for a specific user
id_search <- "AE55KTFVNXYFD5FPYWP2OUPEYNPQ"
```

```
User Ratings:
Product ID: B07N8RQ6W7 Rating: 4.1
Product ID: B08N1WL9XW Rating: 4
Product ID: B08P9RYPLR Rating: 4
Product ID: B09NHVCHS9 Rating: 4
Product ID: B09NJN8L25 Rating: 4
Product ID: B09NKZXMWJ Rating: 4
Product ID: B09NL4DJ2Z Rating: 4
Product ID: B0B3MQXNFB Rating: 4
Product ID: B0B3N8VG24 Rating: 4
Recommendations:
Product ID: B087FXHB6J
 Name: Zebronics Zeb-Companion 107 USB Wireless Keyboard and Mouse Set with Nano Receiver (Black)
 Category: Computers&Accessories|Accessories&Peripherals|Keyboards,Mice&InputDevices|Keyboard&MouseSets
```

CONCLUSION

- **Sentiment Analysis (SVM, Random Forest, k-NN):**
  - **SVM:** High accuracy for positive sentiments
  - **Random Forest:** Good accuracy, perfect specificity for negative sentiments
  - **k-NN:** Lower accuracy, struggles with negative instances
- **Rating and Discount Prediction (Random Forest):**
- **Rating Prediction:** Features - discounted price, actual price, rating count (RMSE: 0.2630)
- **Discount Prediction:** Features - actual price, discount percentage, category (RMSE: 0.1591)
- **Recommendation System (UBCF):**
- **Mechanism:** Predicts user interest based on similar users’ ratings.
- **Benefits:** Enhances engagement, increases retention, boosts cross-selling/up-selling.
- **Future Work:**
- Explore advanced ML and ensemble methods for better accuracy.
- Implement hybrid recommendation approaches.

REFERENCES

- Sultana, Najma & Kumar, Pintu & Patra, Monika & Chandra, Sourabh & Alam, Sk. (2019). SENTIMENT ANALYSIS FOR PRODUCT REVIEW. International Journal of Soft Computing. 09. 7. 10.21917/ijsc.2019.0266.

- N. Ul Saaqib, Gunika and H. K. Verma, "Analysis of Sentiment on Amazon Product Reviews," 2023 Third International Conference on Secure Cyber Computing and Communication (ICSCCC), Jalandhar, India, 2023, pp. 697-702, doi: 10.1109/ICSCCC58608.2023.10176787.

- Haque, Tanjim & Saber, Nudrat & Shah, Faisal. (2018). Sentiment analysis on large scale Amazon product reviews.10.1109/ICIRD.2018.8376299.

- Dublin, Griffith & Joseph, Roshan. (2020). Amazon Reviews Sentiment Analysis: A Reinforcement Learning Approach. 10.13140/RG.2.2.31842.35523.

THANK YOU