Northeastern University

Boston, Massachusetts, USA – 02115

Mechanical and Industrial Engineering Department
Master of Science in Engineering Management
IE 6200: Engineering Probability & Statistics

Project Report on

"A Statistical Analysis of Amazon Sales Data"

Submitted by

Group 9

NAME	NUID
Kush Patel.	002808574
Ashutosh Kulkarni	002878121
Manav Shah	002822746
Pushkar Kukadolkar	002874935

Prof. Paul Pei
Fall 2023

Abstract

We are performing an observational study on the sales data from Amazon India website. The dataset consists of 1000+ product listings and their attributes like original price, discount, rating, review, review count, etc. Our study will draw inferences based on the above-mentioned attributes. We will use visualisation tools to identify trends, correlation between attributes and comment on customer sentiment.

This project proposal outlines an observational study conducted by Group 9 from Northeastern University's Mechanical and Industrial Engineering Department, as part of the IE 6200: Engineering Probability & Statistics course, under the guidance of Prof. Paul Pei during the Fall 2023 semester. The focus of the study is a statistical analysis of Amazon sales data sourced from Kaggle, comprising over 1000 product listings with attributes such as original price, discount, rating, review count, and more.

The project aims to draw meaningful inferences from the dataset, utilizing visualization tools to identify trends, correlations between attributes, and provide insights into customer sentiment. The study poses questions such as the mean rating for a given product category, the proportion of electronics with low ratings, and explores relationships between variables like mean discount and mean rating for a given category.

The observational study employs a sample size of 30 to 50, randomly selected from the larger population, and relies on web scraping techniques using Python and libraries like BeautifulSoup and requests. The team acknowledges potential biases in the dataset, such as missing attributes in certain listings, and addresses them by cleaning up the database. Additionally, the project recognizes the inherent biases in user-generated ratings and reviews on public platforms like Amazon and outlines measures to mitigate their impact on the study's conclusions.

This project proposal establishes the groundwork for a comprehensive statistical analysis of Amazon sales data, providing a structured approach to address research questions and potential challenges, while ensuring ethical and legal data collection practices.

Table of Contents

Chapter 1: Introduction

In the rapidly evolving landscape of e-commerce, understanding consumer behaviour and product performance is integral to shaping strategic decisions for both businesses and academic inquiry. This project proposal presents an initiative by Group 9 from Northeastern University's Mechanical and Industrial Engineering Department, undertaken as part of the IE 6200: Engineering Probability & Statistics course during the Fall 2023 semester. Under the guidance of Prof. Paul Pei, the project delves into a statistical analysis of Amazon sales data, with the objective of extracting valuable insights into product dynamics, customer sentiment, and market trends.

The widespread availability of online retail platforms has changed the traditional retail scenario, providing consumers unmatched access to a variety of items and services. Amazon, as a global e-commerce giant, is at the center of this transition, providing an extensive range of products to a diversified customer base. Analysing the massive dataset accessible on Amazon's sales platform allows for the discovery of patterns, correlations, and trends that can inform not only corporate strategy but also contribute to a broader understanding of customer preferences and market dynamics.

Our study focuses on a dataset comprising over 1000 product listings from Amazon India, encompassing key attributes such as original price, discount, rating, review count, and more. Leveraging the principles of engineering probability and statistics, we aim to employ robust analytical methodologies to extract meaningful information from this dataset. Through visualizations, statistical tests, and exploratory data analysis, our goal is to address pertinent questions, ranging from the mean rating of specific product categories to the proportion of products with unfavourable ratings.

The contemporary importance of data-driven decision-making underscores the significance of projects like ours, where engineering students apply statistical techniques to real-world datasets. As we embark on this journey of exploration and analysis, we are not only contributing to the academic discourse within our department but also gaining practical insights into the intersection of engineering and e-commerce analytics.

1.1 Questions for Study

In the dynamic landscape of e-commerce, a critical question emerges: **"Does Amazon require improvement in product listings for the 'Computers & Accessories' category?"**

This shift in focus directs our statistical analysis towards a targeted evaluation of a specific product category, aiming to unearth insights that can influence the platform's user experience. Our project delves into this question by meticulously examining key attributes such as original price, discount, and review metrics within the dataset of over 1000 product listings.

Leveraging statistical methods and visualization tools, we seek to identify potential areas for enhancement in the presentation and quality of product listings, contributing valuable insights to optimize the customer journey and elevate the overall performance of the 'Computers & Accessories' category on Amazon.

In the expansive realm of online retail, a pertinent inquiry arises: **"Does Amazon provide different discounts for different categories?"**

This question forms the crux of our statistical analysis project, where we aim to explore the potential variations in discount offerings across diverse product categories on the Amazon platform. Leveraging the powerful tool of hypothesis testing, we seek to rigorously examine whether there are statistically significant differences in the discount percentages offered.

Our investigation delves into a dataset containing over 1000 product listings, meticulously scrutinizing attributes such as original price, discount, and category distinctions.

By employing hypothesis testing, our objective is to discern meaningful patterns, providing valuable insights into Amazon's pricing strategies and potentially informing both consumers and sellers about the nuances of discount structures across various product categories.

This analytical journey holds the promise of unravelling essential dynamics within Amazon's pricing framework, contributing to a nuanced understanding of e-commerce strategies in the digital marketplace.

In pursuit of a deeper understanding of Amazon's product landscape, we turn our analytical lens toward a pivotal question: **which category boasts a higher prevalence of products on heavy discount – Electronics or Home & Kitchen?**

The strategic implications of discounting strategies in the e-commerce realm are profound, shaping consumer choices and influencing market dynamics. Through meticulous data sampling and statistical analysis, we aim to discern the proportion of products with substantial discounts in these two distinct categories.

By employing random sampling and proportion estimation techniques, our investigation seeks to unveil whether Electronics consistently surpasses Home & Kitchen in offering compelling discounts. This exploration not only contributes to the nuanced understanding of Amazon's pricing tactics but also sheds light on the intricate interplay between discounting strategies and consumer engagement in the ever-evolving e-commerce landscape.

In our pursuit of comprehending the underlying dynamics of a dataset, we delve into a fundamental question: **"What is the correlation between the variables—discounted price, actual price, discount percentage, rating, and rating count?"**

This inquiry propels us into a systematic analytical process where we navigate through the identification of pertinent variables, prepare the data, and calculate correlations using statistical metrics, particularly the Pearson correlation coefficient. The essence of our exploration lies in unraveling the intricate relationships within this dataset and visually representing them through a heatmap.

This visual tool not only quantifies the strengths of connections but also reveals patterns and dependencies, paving the way for nuanced interpretations. Join us on this analytical journey as we decode the complex interplay of variables and extract meaningful insights from their correlations

Chapter 2: Data Collection

2.1 Method of sample collection

The given dataset consists of Amazon product sales data. It has been compiled from the official Amazon website using a “web scraper”. Web Scraping is the programming-based technique for extracting relevant information from websites and storing it in the local system for further use. This can be done by using Python (*or other programming languages*) and its libraries like “BeautifulSoup” and “requests”.

Extracting/getting data from sources like Amazon or any other institution for non-commercial uses requires certain permissions and licenses. This dataset is completely legal to use for academic projects like ours.

Check the following link for the same:
CC BY-NC-SA 4.0

A simple web scraper example:
The following code scrapes one Amazon product listing and extracts the brand, title, price, rating and number of reviews.

URL:
https://www.amazon.com/fire-tv-stick-with-3rd-gen-alexa-voice-remote/dp/B08C1W5N87/ref=zg_bs_g_amazon-devices_sccl_2/145-1117512-8074652?psc=1#tech

Code:

```
import requests
from bs4 import BeautifulSoup

url = ("https://www.amazon.com/fire-tv-stick-with-3rd-gen-alexa-
voice-remote/dp/B08C1W5N87/ref=zg_bs_g_amazon"
       "-devices_sccl_2/145-1117512-8074652?psc=1")

HEADERS = {
    'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36',
     'Accept-Language': 'en-US, en;q=0.5'
}
```

```
html = requests.get(url, headers=HEADERS)
soup = BeautifulSoup(html.text, features="html.parser")
title = soup.find('span', {'id': "productTitle"}).text.strip()
price = soup.find('span', {'class': "a-offscreen"}).text.strip()
rating = soup.find('span', {'class': "a-icon-alt"}).text.strip()
ratingn = soup.find('span', {'id':
"acrCustomerReviewText"}).text.strip()
brand = soup.find('a', {'id': "bylineInfo"}).text.strip()
print(brand)
print(title)
print(f'price = {price}')
print(f'rating = {rating}')
print(f'number of reviews = {ratingn}')
```

Output:

```
Brand: Amazon
Amazon Fire TV Stick, fast HD streaming, free & live TV, quick app starts, Alexa
Voice Remote with TV controls price = $19.99
rating = 4.7 out of 5 stars
number of reviews = 429,541 ratings

Process finished with exit code 0.
```

Chapter 3: Data Preparation

3.1 Loading the Dataset:

The code starts by importing necessary libraries such as NumPy, pandas, and matplotlib for data manipulation, analysis, and visualization.
The dataset is loaded into a pandas DataFrame ('df') using the read_csv function.

```
import numpy as np
import pandas as pd
from tabulate import tabulate
import random
import matplotlib.pyplot as plt

# Giving file path to python
dataset = '/Users/ashu_k/Documents/MEM_PS/PS
Project/Datasets/amazon.csv'

# Storing dataset as "df" for manipulation by python
df = pd.read_csv(dataset)
```

3.2 Handling Missing Values:

The function `check_missing_values` is defined to identify and count missing values in each column of the DataFrame.
Rows with missing values in the 'rating_count' column are removed from the DataFrame.

```
# Function to check missing values column-wise

def check_missing_values(dataframe):
    return dataframe.isnull().sum()

# Remove rows with missing values from df
df.dropna(subset=['rating_count'], inplace=True)
```

3.3 Checking for Duplicate Entries:

The function `check_duplicates` is created to count the number of duplicated rows in the DataFrame.

```
# Check for duplicate entries
def check_duplicates(dataframe):
    return dataframe.duplicated().sum()
```

3.4 Checking Data Types:

The function `check_data_types` is defined to inspect the data types of each column in the DataFrame.

```
# Check data type in csv file
def check_data_types(dataframe):
    return dataframe.dtypes
```

3.5 Converting Numerical Values:

Columns with numerical values must be converted from object to float data type to avoid errors.
Columns like 'discounted_price', 'actual_price', 'discount_percentage', 'rating', and 'rating_count' are converted to float.

```
#  Converting numerical for required categories values to float
for calculation
df['discounted_price'] =
df['discounted_price'].astype(str).str.replace('₹',
'').str.replace(',', '').astype(float)
df['actual_price'] =
df['actual_price'].astype(str).str.replace('₹',
'').str.replace(',', '').astype(float)
df['discount_percentage'] =
df['discount_percentage'].astype(str).str.replace('%',
'').astype(float)
df['rating'] = df['rating'].astype(float)
df['rating_count'] =
df['rating_count'].astype(str).str.replace(',',
'').astype(float)
```

3.6 Splitting Main and Sub-Categories:

The 'category' column is split into three separate columns: 'main_category', 'sub_category-1', and 'sub_category-2'.
This enables a more granular analysis based on different levels of product categorization.

```
#  splitting of main and sub categories
df['sub_category-2'] =
df['category'].astype(str).str.split('|').str[-1]
df['main_category'] =
df['category'].astype(str).str.split('|').str[0]
df['sub_category-1'] =
df['category'].astype(str).str.split('|').str[1]
```

These data preparation steps ensure that the dataset is cleaned, missing values are handled appropriately, and the data types are suitable for subsequent analysis. The splitting of categories makes it easy to search and get data as required by our questions.

Chapter 4: Data Analysis and Visualization

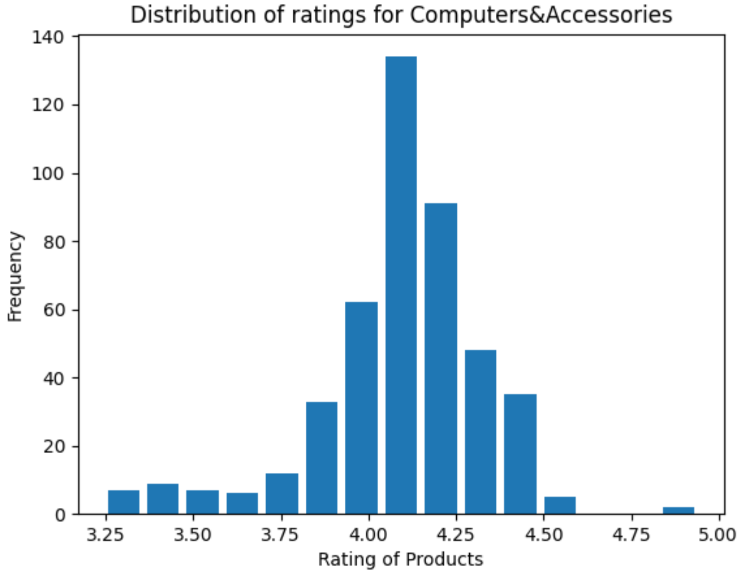
4.1 Type 1 – Mean of a Variable

Question:
Does Amazon require improvement in product listings for the “Computers & Accessories” category?

Process:
The analysis focuses on assessing the customer ratings for the "Computers & Accessories" category on Amazon. Due to the large number of listings (451), an estimate of the average product rating is sought. The population, in this case, is the list of ratings for the main category "Computers & Accessories." The distribution of the data is examined to determine an appropriate sample size, followed by obtaining a sample mean rating.

Let’s check the customer ratings for the products. But we have 451 listings! It is not possible to check the rating for all. An estimate of the average product rating will be helpful to decide whether an improvement is required.

1. Get the list of ratings for main category -> “Computers & Accessories”. This will be our population.
2. Check the distribution of data to decide sample size.



3. Now, generate a random sample of 45.
4. Calculate the sample mean and sample standard deviation.
   Sample average x-bar = 4.12
   Sample Std dev = 0.24364653981461828
5. For 95% confidence level, get Z-value.
   Z value -> 1.959963984540054
6. Calculate the confidence interval bounds.
   95% CI -> (4.048812771972999,4.191187228027001)

Analysis Details

Sample Size Determination:
As we can observe, the data distribution is approximately normal with a slight negative skew. Hence, we will take a good enough sample size of 45.

Sample Statistics Calculation:
Sample mean (x-bar) is calculated as 4.12.
Sample standard deviation (s) is calculated as 0.2436.

Confidence Interval Calculation:
For a 95% confidence level, the Z value is calculated as 1.9599.
The confidence interval bounds are computed as (4.0488, 4.1912).

Choice of Analysis:
This is a suitable choice when the objective is to obtain an approximation of the population mean.
A Confidence Interval (CI) for the mean is chosen. This analysis provides a range estimate for the true average rating based on a sample. It allows for a more nuanced understanding of the likely location of the population mean.

Choice of Confidence Level:
A confidence level of 95% is selected. This is a standard level providing a balance between precision and reliability. It means that we are 95% confident that the true population mean rating will fall within the calculated interval.

Python Output:
Sample mean rating for Computers & Accessories -> 4.175555555555555

4.2 Type 2 - Comparison of 2 means

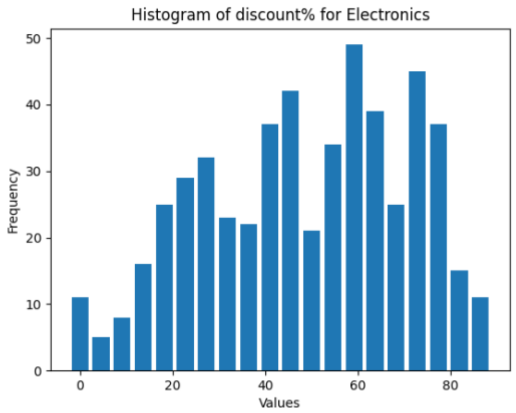
Question:

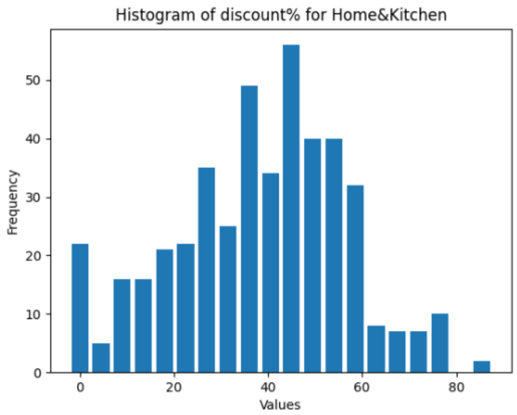
Does Amazon provide different discounts according to categories? (Electronics and Home & Kitchen)?

Process:

Let’s check the difference between the mean ratings of the two categories. It is not ideal to check all the listings. So, we will perform a Hypothesis test to check if there is a difference between discount offered on Electronics and Home & Kitchen.

1. Let’s check how the population values are distributed for both categories.





The distribution for both is non-normal. Hence, we will perform a large sample Z-test with a good enough sample size of 75.

2. Hypothesis test:
   a. μ_1 = mean discount on Electronics, μ_2 = same for Home & Kitchen
   b. H_0: $\mu_1 - \mu_2 = 0$ (There is no difference in mean discount percentages)
      H_a: $\mu_1 - \mu_2 \neq 0$ (There is an enough difference in mean discount percentages)
   c. Test at significance level (α) = 0.05

3. Calculate Z statistic:
   Python output -> Z value is: 2.5381320186476963

4. Calculate P-value:
   Python output -> Two-tailed p-value: 0.011144593494670563

5. Conclusion of test:
   Python output -> Reject Null Hypothesis
   We can say that Amazon has provided different discounts depending on the category and electronics have generally more discount.

Choice of Analysis:
The chosen analysis for comparing mean discounts on Electronics and Home & Kitchen categories is a two-sample Z-test. Here's the rationale behind this choice:

Nature of the Data:
The dataset indicates that the distribution of discount percentages for both categories is non-normal. In such cases, the Z-test is a robust choice, especially when dealing with large sample sizes.

Large Sample Size:
The sample size for each category is deemed to be sufficiently large (75 for each category). The Z-test is particularly effective with large samples, providing reliable results.

Objective of the Analysis:
The goal is to determine whether there is a significant difference in mean discount percentages between the two categories. The two-sample Z-test is well-suited for this comparison, allowing us to draw conclusions about population means.

Type of Hypothesis:
The hypothesis involves comparing means between two independent samples (Electronics and Home & Kitchen). The two-sample Z-test is designed for precisely this scenario.

Choice of Significance Level (α):
A significance level (α) of 0.05 is chosen. This common alpha level provides a balance between Type I and Type II errors. It is a standard practice in statistical analysis, widely accepted in many disciplines.
A significance level of 0.05 means that there is a 5% chance of rejecting the null hypothesis when it is true (Type I error). This level is considered standard and strikes a balance between being stringent enough to control false positives and allowing for reasonable sensitivity to detect true differences.

4.3 Type 3: Comparison of 2 proportions

Question:
Which category has more products on heavy discount? Electronics or Home & Kitchen?

Process:
Let’s calculate and compare the proportion of products with heavy discount.

1. Get the list of discounts for Electronics and Home & Kitchen.
2. Generate a random sample of 40 for each.
3. Calculate sample proportion $\hat{p}$ for both.
4. Compare the two proportion estimates.

The chosen analysis involves the comparison of proportions between two categories, Electronics and Home & Kitchen, specifically focusing on the proportion of products with heavy discounts (greater than or equal to 65%).

Code:

```
import numpy as np
import pandas as pd
from tabulate import tabulate
import random
import matplotlib.pyplot as plt
from scipy.stats import norm
import seaborn as sns


# Giving file path to python
dataset = '/Users/ashu_k/Documents/MEM_PS/PS
Project/Datasets/amazon.csv'


# Storing dataset as "df" for manipulation by python
df = pd.read_csv(dataset)


# Function to check missing values column-wise
def check_missing_values(dataframe):
    return dataframe.isnull().sum()


# Remove rows with missing values from df
df.dropna(subset=['rating_count'], inplace=True)
```

```
# Check for duplicate entries
def check_duplicates(dataframe):
    return dataframe.duplicated().sum()

# Check data type in csv file
def check_data_types(dataframe):
    return dataframe.dtypes


#  Converting numerical for required categories values to float
for calculation
df['discounted_price'] =
df['discounted_price'].astype(str).str.replace('₹',
'').str.replace(',', '').astype(float)
df['actual_price'] =
df['actual_price'].astype(str).str.replace('₹',
'').str.replace(',', '').astype(float)
df['discount_percentage'] =
df['discount_percentage'].astype(str).str.replace('%',
'').astype(float)
df['rating'] = df['rating'].astype(float)
df['rating_count'] =
df['rating_count'].astype(str).str.replace(',',
'').astype(float)

#  splitting of main and sub categories
df['sub_category-2'] =
df['category'].astype(str).str.split('|').str[-1]
df['main_category'] =
df['category'].astype(str).str.split('|').str[0]
df['sub_category-1'] =
df['category'].astype(str).str.split('|').str[1]
#  List of discount% for main_category -> Electronics
mean_disc1 = (df[(df['main_category'] ==
'Electronics')]['discount_percentage'].tolist())
#  List of discount% for main_category -> Home&Kitchen
mean_disc2 = (df[(df['main_category'] ==
'Home&Kitchen')]['discount_percentage'].tolist())


#  Sample calculations for Electronics
disc1_sample = random.sample(mean_disc1, 44)  # generate a
random sample of n=100
sum_Xi = [value for value in disc1_sample if value >= 65]
prop_1 = len(sum_Xi)/len(disc1_sample)
print(f'Summation Xi = {sum_Xi}')
print(f'p1 = {prop_1}')
```

```
#  Sample calculations for Home&Kitchen
disc2_sample = random.sample(mean_disc2, 44)  # generate a
random sample of n=100
sum_Yi = [value2 for value2 in disc2_sample if value2 >= 65]
prop_2 = len(sum_Yi)/len(disc2_sample)
print(f'Summation Yi = {sum_Yi}')
print(f'p2 = {prop_2}')

#  As we can see, consistently the Electronics category has more
proportion of products with a discount >= 65%
```

Choice of Analysis:

The chosen analysis involves the comparison of proportions between two categories, Electronics and Home & Kitchen, specifically focusing on the proportion of products with heavy discounts (greater than or equal to 65%).

Type 4: Correlation

Question:
What is the correlation between the variables—discounted price, actual price, discount percentage, rating, and rating count?

Process:
Certainly, let's break down the process of identifying and visualizing the correlation between discount price, discount percentage, rating, actual price, and rating count using the correlation function and a heatmap:

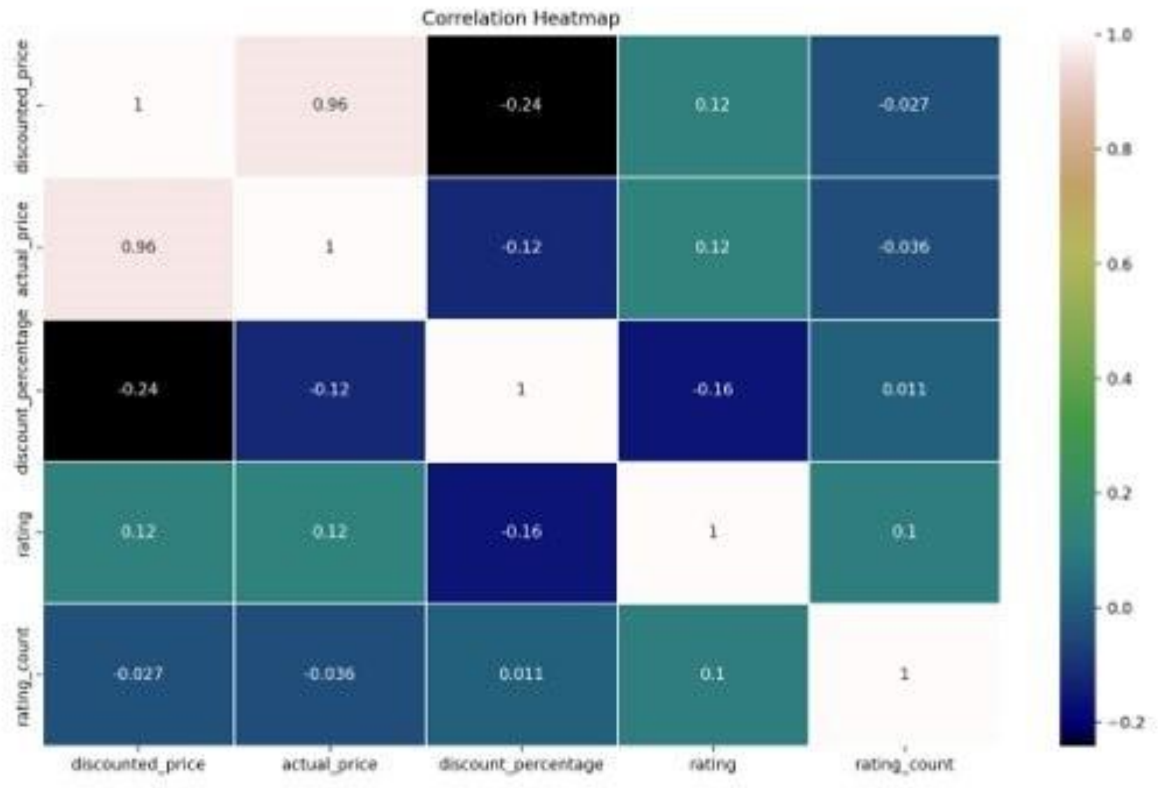
1. Identification of Variables:
Identify the variables for which correlation needs to be assessed: discount price, discount percentage, rating, actual price, and rating count.

2. Data Preparation:
Ensure the dataset includes the relevant variables needed for correlation analysis.

3. Correlation Calculation:
Utilize the correlation function (e.g., Pearson correlation coefficient) to calculate the pairwise correlations between the identified variables. The correlation coefficient ranges from -1 (perfect negative correlation) to 1 (perfect positive correlation), with 0 indicating no correlation.

The optimum ratio to classify a positive association as weak, moderate, or strongly relies on the research issue and the field of investigation. But as a general guideline, remember this: There is just a weak positive association (r = 0.1–0.3) when it comes to correlation coefficients. An intermediate degree of positive connection is indicated by a correlation coefficient (r) between 0.3 and 0.5. A significant positive association is indicated by a correlation coefficient (r) of more than 0.5. Remember that this is only a general guideline and that when analyzing correlation coefficients, other factors like sample size, measurement errors, and outliers should also be taken into consideration.

4. Heatmap Generation:
Visualize the correlation matrix using a heatmap. Heatmaps provide an intuitive way to understand the strength and direction of correlations between variables.



5. Code Used for Heatmap Plotting:

```
import numpy as np
import pandas as pd
from tabulate import tabulate
import random
import matplotlib.pyplot as plt
import seaborn as sns


# Giving file path to python
dataset = '/Users/ashu_k/Documents/MEM_PS/PS
Project/Datasets/amazon.csv'


# Storing dataset as "df" for manipulation by python
df = pd.read_csv(dataset)
# Function to check missing values column-wise
def check_missing_values(dataframe):
    return dataframe.isnull().sum()
# Remove rows with missing values from df
df.dropna(subset=['rating_count'], inplace=True)
```

```
# Check for duplicate entries
def check_duplicates(dataframe):
    return dataframe.duplicated().sum()
# Check data type in csv file
def check_data_types(dataframe):
    return dataframe.dtypes

#  Converting numerical for required categories values to float
for calculation
df['discounted_price'] =
df['discounted_price'].astype(str).str.replace('₹',
'').str.replace(',', '').astype(float)
df['actual_price'] =
df['actual_price'].astype(str).str.replace('₹',
'').str.replace(',', '').astype(float)
df['discount_percentage'] =
df['discount_percentage'].astype(str).str.replace('%',
'').astype(float)
df['rating'] = df['rating'].astype(float)
df['rating_count'] =
df['rating_count'].astype(str).str.replace(',',
'').astype(float)

#  splitting of main and sub categories
df['sub_category-2'] =
df['category'].astype(str).str.split('|').str[-1]
df['main_category'] =
df['category'].astype(str).str.split('|').str[0]
df['sub_category-1'] =
df['category'].astype(str).str.split('|').str[1]
selected_columns = ['discounted_price', 'actual_price',
'discount_percentage', 'rating', 'rating_count']
selected_df = df[selected_columns]
correlation_matrix = selected_df.corr()
# Displaying the correlation matrix
print("\nCorrelation Matrix:")
print(correlation_matrix)

# Plotting the heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='gist_earth',
linewidths=.5)
plt.title("Correlation Heatmap")
plt.show()
```

Chapter 5: Conclusion

The sample mean rating of 4.18 for the "Computers & Accessories" category serves as a preliminary estimate of the average customer rating. Looks like the average rating of products is around 4.1. The customers seem satisfied and there is no immediate requirement to review the product listings.

The chosen two-sample Z-test with a significance level of 0.05 is appropriate for addressing the research question and aligns with the characteristics of the dataset, sample sizes, and the specific hypothesis being tested. There exists a good enough difference between the discounts offered on Electronics and Home & Kitchen. Generally, Electronics have more discount.

The analysis directly addresses the question of which category has more products on heavy discount, providing a focused comparison. Proportion comparison is suitable for addressing the question as it measures the relative frequency of products with heavy discounts in each category. The use of random sampling ensures that the selected subsets are representative, reducing bias and allowing for generalization to the entire population. The conclusion is straightforward and easy to interpret – Electronics consistently has a higher proportion of products with heavy discounts compared to Home & Kitchen. The analysis opens the possibility for further statistical testing to determine if the observed difference in proportions is statistically significant.

We have a heatmap of correlations between a few of the variables in our dataset here. A marginally favorable association can be observed between the total rating and the weighted rating as well as the rating count. This implies that more reviews and weighted ratings are typically found for products with higher ratings. The "rating" and "discounted price" variables have a somewhat positive association (0.121), suggesting that customers are likely to rate a product higher if it is discounted. Although correlation does not always imply causation, it is crucial to remember that these insights might aid in our understanding of the links between various elements in our data.

Chapter 6: References

Probability & Statistics for Engineering and the Sciences, 9th ed., by Jay L. Devore, Cengage Learning (January 1, 2015).

https://stats.stackexchange.com/

https://statisticsbyjim.com/