Amazon case study answer

### Step-by-Step Instructions for Data Analysis Project

1. Read the Data for Analysis:
   * Load the dataset into a pandas DataFrame.

python

Copy code

import pandas as pd

# Load the dataset

reviews\_df = pd.read\_csv('/mnt/data/Reviews.csv')

# Display the first few rows of the dataframe to understand its structure

print(reviews\_df.head())

1. Perform Data Pre-processing / Data Cleaning:
   * Handle missing values, remove duplicates, and correct data types.

python

Copy code

# Check for missing values

print(reviews\_df.isnull().sum())

# Drop rows with missing values

reviews\_df = reviews\_df.dropna()

# Remove duplicates

reviews\_df = reviews\_df.drop\_duplicates()

# Convert columns to appropriate data types if necessary

reviews\_df['reviewTime'] = pd.to\_datetime(reviews\_df['reviewTime'])

# Display the cleaned data

print(reviews\_df.head())

1. How Amazon Recommends Products:
   * Discuss the basics of recommendation systems (collaborative filtering, content-based filtering, hybrid methods).

markdown

Copy code

# Recommendation Systems Overview

Amazon recommends products using a combination of:

- \*\*Collaborative Filtering\*\*: Based on user behavior and preferences, such as previous purchases and ratings.

- \*\*Content-Based Filtering\*\*: Based on product attributes and similarities.

- \*\*Hybrid Methods\*\*: Combining collaborative and content-based filtering for more accurate recommendations.

1. Analyzing Which Product Has a Good Number of Reviews:
   * Identify products with the highest number of reviews.

python

Copy code

product\_reviews = reviews\_df.groupby('product\_id').size().reset\_index(name='review\_count')

top\_products = product\_reviews.sort\_values(by='review\_count', ascending=False).head(10)

# Display top products by review count

print(top\_products)

1. Understanding Behaviors of Amazon Users:
   * Analyze user activity patterns, such as frequency of reviews and ratings given.

python

Copy code

user\_activity = reviews\_df.groupby('user\_id').size().reset\_index(name='review\_count')

frequent\_users = user\_activity.sort\_values(by='review\_count', ascending=False).head(10)

# Display most active users

print(frequent\_users)

1. Analysis of Frequent Users:
   * Dive deeper into the behaviors and patterns of frequent users.

python

Copy code

frequent\_users\_reviews = reviews\_df[reviews\_df['user\_id'].isin(frequent\_users['user\_id'])]

# Analyze review patterns, average ratings, etc.

frequent\_users\_summary = frequent\_users\_reviews.groupby('user\_id').agg({

'rating': ['mean', 'count'],

'reviewTime': ['min', 'max']

}).reset\_index()

# Display summary of frequent users

print(frequent\_users\_summary)

1. Perform Sentiment Analysis of Data:
   * Use NLP techniques to perform sentiment analysis on the reviews.

python

Copy code

from textblob import TextBlob

# Function to calculate sentiment

def get\_sentiment(text):

analysis = TextBlob(text)

return analysis.sentiment.polarity

# Apply sentiment analysis to review text

reviews\_df['sentiment'] = reviews\_df['reviewText'].apply(get\_sentiment)

# Display the sentiment analysis results

print(reviews\_df[['reviewText', 'sentiment']].head())

### Additional Tips

* Encourage the student to explore additional analyses and visualizations.
* They can use libraries like matplotlib and seaborn for visualizations.
* Emphasize the importance of documenting their process and findings in a report or presentation.

This structured approach will help the intern perform a comprehensive analysis of the Amazon Reviews dataset.