Pandas Analysis of Anime and K-Pop Merchandise Sales

December 31, 2024

1 Dataset

The following dataset contains information about anime and K-pop merchandise sales.

Table 1: Anime and K-Pop Merchandise Sales Data

Order ID	Product Name	Category	Quantity Sold	Price pe
1	Demon Slayer: Kimetsu no Yaiba - Tanjiro Figure	Figurines	5	
2	Attack on Titan Season 1 DVD Box Set	Anime	3	
3	My Hero Academia: All Might Pop Figure	Figurines	2	
4	Blackpink: The Album	Music	1	
5	One Piece: Luffy Figure	Figurines	4	
6	Naruto Shippuden: Kakashi Figure	Figurines	10	
7	Jujutsu Kaisen: Yuuta Okkotsu Figure	Figurines	2	
8	Your Name Blu-ray	Anime	3	
9	Cowboy Bebop: Complete Series DVD	Anime	1	
10	Studio Ghibli: The Art of Spirited Away	Book	4	
11	Demon Slayer: Nezuko Figure	Figurines	6	
12	Attack on Titan Season 2 DVD Box Set	Anime	5	
13	One Punch Man: Saitama Figure	Figurines	2	
14	Spirited Away: Chihiro Figure	Figurines	3	
15	Cowboy Bebop Soundtrack	Music	1	
16	My Neighbor Totoro Plush	Figurines	8	
17	Howl's Moving Castle DVD	Anime	3	
18	Naruto: Shippuden DVD Box Set	Anime	4	
19	Dragon Ball Z: Goku Figure	Figurines	2	
20	Fullmetal Alchemist: Brotherhood DVD Box Set	Anime	5	

2 Questions

Here are some analytical questions based on the dataset:

1. How many missing values are present in each column?
2. What strategies can you use to fill the missing values in the 'Quantity Sold' column?
3. How will you handle missing values in the 'Total Sales' column?
4. For the 'Rating' column, what would be a reasonable value to fill in for missing data?
5. After filling the missing values, how can you verify that there are no remaining missing values in the DataFrame?
6. What impact might filling in missing values have on your analysis?
7. What are the top three products in terms of total sales revenue? Provide a breakdown of their sales figures.
8. Can you identify which city has the highest number of orders? What does this tell us about our customer base in that location?
9. How do the average ratings differ across product categories (e.g., Figurines, Anime, Music)? Are there any categories that consistently receive higher ratings?
10. Is there any correlation between delivery time and customer ratings? How does delivery time affect customer satisfaction?

3 Explanations and Python Pandas Code

Here are the explanations and Python Pandas code for each question:

3.1 1. Count Missing Values

To count the number of missing values in each column:

```
# Count missing values in each column
missing_values = df.isnull().sum()
print("Missing␣values␣in␣each␣column:")
print(missing_values)
```

3.2 2. Strategies for Filling Missing Values

For the 'Quantity Sold' column, you can fill missing values using the mean:

```
# Fill missing values in 'Quantity Sold' with the mean
df['Quantity␣Sold'].fillna(df['Quantity␣Sold'].mean(),
    inplace=True)
```

3.3 3. Handling Missing Values in Total Sales

You can calculate the total sales based on quantity sold and price per unit:

```
# Fill missing values in 'Total Sales' with calculated
    values
df['Total␣Sales'].fillna(df['Quantity␣Sold'] * df['
   Price␣per␣Unit'], inplace=True)
```

3.4 4. Filling Missing Values in Ratings

For the 'Rating' column, fill missing values with the mean rating:

```
# Fill missing values in 'Rating' with the mean
df['Rating'].fillna(df['Rating'].mean(), inplace=True)
```

3.5 5. Verifying No Remaining Missing Values

Check for any remaining missing values after filling:

```
# Check for remaining missing values
remaining_missing = df.isnull().sum()
print("\nRemaining␣missing␣values␣after␣filling:")
print(remaining_missing)
```

3.6 6. Impact of Filling Missing Values

Consider how filling missing values can affect your analysis:

- Filling missing values can lead to more accurate statistical measures but may also introduce bias if not done carefully.

3.7 7. Top Three Products by Total Sales

To find the top three products in terms of total sales revenue:

```
# Top three products by total sales
top_products = df.groupby('Product␣Name')['Total␣Sales
   '].sum().nlargest(3)
print("Top␣Three␣Products␣by␣Total␣Sales␣Revenue:")
print(top_products)
```

3.8 8. City with Highest Number of Orders

To identify the city with the highest number of orders:

```
# City with the highest number of orders
top_city = df['City'].value_counts().idxmax()
print("City␣with␣the␣highest␣number␣of␣orders:",
    top_city)
```

3.9 9. Average Ratings by Product Category

To compare average ratings across product categories:

```
# Average ratings by category
average_ratings = df.groupby('Category')['Rating'].
    mean()
print("Average␣Ratings␣by␣Category:")
print(average_ratings)
```

3.10 10. Correlation Between Delivery Time and Customer Ratings

To analyze the correlation between delivery time and ratings:

```
# Correlation between delivery time and ratings
correlation = df['Delivery␣Time␣(Days)'].corr(df['
    Rating'])
print("Correlation␣between␣delivery␣time␣and␣ratings:"
    , correlation)
```