Rachel Nguyen

BUAN 4210

03/15/2023

Part 1: Cleaning the Data

In order to clean data, first we import NumPy as np, pandas as pd (DataFrame), matplotlib.pyplot as plt , spipy.stats, and seaborn from data manipulation, analysis, and visualization. Then, it load a csv file name "MM_Sales.csv" into a pandas DataFrame called MMsales. The code then uses the info() method to display information about the DataFrame, such as the number of rows and columns, column names, and data types. Lastly, the code sets an option to display float values with 3 decimal places instead of the default 6 decimal places.

The method "isna()" checks each element of the Data Frame for missing values and returns a DataFrame of the same shape consisting of Boolean values indicating whether each element is missing or not.

The "Non-Null Count" column shows the number of non-null values for each column, indicating the amount of missing data present. There are two different column that have different data in non-null count those are Order Priority and Item Type. The reason why I put the sum at the end is because that method called on this Boolean DataFrame to sum the number of True value in each column, which represents the number of missing values in that column.

MMsalesClean = MMsales: The code provided creates a clean version of the original DataFrame by making a new copy "MMsalesClean". I put .fillna() to replace missing values in specific columns of the new DataFrame. In the first line, the missing values in the "Country" column are filled with the string "Null". Similarly, the missing values in the "Item Type" column are filled with the string "NULL". The Order Priority column are also filled with the string "NULL" . Lastly, the missing values in the "Order

ID” column are filled with the value 0, integer. The code provided loops through each row in the “MMsalesClean” DataFrame and attempts to convert the value in the “Country” column to a float. If this conversion fails, it is assumed that the value in the “Country” column to a float. If it failed, it is assumed that the value is erroneous data and replaced with the string “NULL” . I did the same thing to Order Priority, Item Type. However, as Order ID attempts to convert to integer, if this conversation fails, it will replace with the integer 0.

By using the “.isnull()” method on the DataFrame, the code generates a Boolean mask that identifies any missing values in the DataFrame. The .sum() method is then used to sum the number of missing values. After all, using .info() method provides useful information about the the DataFrame, including the number of non-null values, data types of each column, and the memory usage of the DataFrame. All count returns to 49971 non null count.

Part 2: Exploratory Data Analysis with report and visualization

In order to generate a rank of the top 10 countries with the highest number of sales in the cleaned MM sales data, I use groupby() method is used to group data by country, and the ‘count() method is applied to each group to count the number of unique Order IDs in each country. The ‘nlargest()’ method is then used to select the top 10 countries with the highest number of Order IDs. Using Matplotlib library is ised to plot the bar chart, with the ‘Country’ column on the x-aix and the ‘ Number of Sales’ column on the y-axis. The chart provides a visual representation of the top 10 sales countries and their relative number of sales.

Write new file “ MM_Ranking.txt: Countries most sale transactions. Use append method to write Country Name and number of sales transactions of that country.

This code groups the cleaned sales data by the ‘Sales Channel’ column and counts the number of orders for each channel. It then prints the resulting Dataframe which shows the count of order for each channel. Then we got Sale Chanel (Offline: 19788 , Online: 30183)

The code above group the cleaned data ‘newMMsales’ by ‘Order Priority’ and counts the number of occurrences of each ‘Order Priority’ using ‘groupby() and count() functions. It returns a DataFrame containing the different ‘ Order Priority’ type and their respective counts in the column “ order id’ . We got C: 5012, H:15094 , L: 9984 , M: 19881

In order to create chart, we first create DataFrame call pie with the number of sales by sales channel(Offline and Online). Then, using Seaborn to visualize this data with a pie chart. The pie chart shoes 60.40 in online and 39.60 in offline. We also create pie2 as Datafram for Order Priority columns by Seaborn library.

Add the results of the sales channel types and the order priorities to the file MM_Ranking.txt by using append method.

I use Seaborn to draw boxplot to visualize the distribution of the total profits by item type. The figure size is 15,10

It seems that the data shows the sum of the total profit for each item type. It could be useful to analyze which item types are generating more profits and which ones could potentially be improve in terms of profit. I also use Bar Graph by Seaborn way.

We then find sum of total profit and rank the top 3 items types we did the most sales. After print, we have cosmetics , household and office supplies show up on the Item Type columns.

Add the result of top 3 to the file MM_Rankings.txt. Be sure to use append data rather than writing over top of the previous data. Include a newline between each append to the file. When writing to the file.

The code in part 4 calculates the total sum, mean and max of various metrics such as units sold, units sold, total revenue, total cost, and total profit for the sale data. Then, create two line plots using Seaborn and save these calculations below to the text file called MM_Calc.txt.

Part 3: Cross-Reference Statistics.

Using the.GROUPBY file type, I was able to divide the country into its myriad of regions in the third and last chapter of this series (). On the other hand, I incorporated.UNIQUE() to verify that there are no countries or regions that are identical to one another and to stop any confusion that may have arisen over this matter. In addition, I have prepared a glossary of terms based on the findings for you to use in the next days, months, and years. The completion of this procedure involved the creation of a pandas data frame that was predicated on the dictionary that came before it. After that, I produced a copy of the data frame in the form of a csv file and titled it Countries By Region.csv. I then proceeded to name each of the countries in the file. I used the option for the index that had the value false, and that was so that the final file would not have a number index that was automatically generated.