Data Visualization of Online Retail Dataset

Shannon School of Business, Cape Breton University

MGSC-5127-10.2023F – Data Visualization

Instructor: Asma ul Husna

Submission Date: 23 November 2023

Table of Contents

Data Visualization for Online Retail Dataset

Project Description

The dataset contains all the transactions occurring between December 2010 and December 2011 of online retail. The analysis of this data will provide an insight and understanding of the retail online market trends over the period shown. Detailed this report highlights key phases of data collection, data pre-processing, exploring data analysis, and storytelling by using Python. Visualization such as different types of graphs will be used during the storytelling phase for easy comprehension of key findings. From the visualization of the data, we can get information about sales trends, the performance of the products along countries, and customers' performance. This information will be valuable for a business in making decisions, forecasting, and planning their business.

1. Data Collection

The dataset can be accessed at: https://archive.ics.uci.edu/dataset/352/online+retail

The data contains the following attributes:

InvoiceNo: Invoice number. Nominal, is a 6-digit integral number uniquely assigned to each transaction. If this code starts with the letter "c", it illustrates a cancellation.

StockCode: Product code. Nominal, is a 5-digit integral number uniquely assigned to each distinct product.

Description: Product name. Nominal

Quantity: The quantities of each product per transaction. Numeric

InvoiceDate: Invoice date and time, indicates the date and time when each transaction was generated. Numeric

UnitPrice: Product price per unit in sterling. Numeric

CustomerID: Customer number. Numeric, a 5-digit integral number uniquely assigned to each customer

Country: The name of the country where each transaction resides. Nominal

2. Data Pre-processing

- Importing some necessary libraries and the dataset into Python:

```
# Import

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

```
# Importing data

sales = pd.read_csv('Online Retail.csv')
```

- Exploring the dataset by understanding datatype, central tendency, and data shape:

```
## Exploring dataset

sales.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 8 columns):
 #   Column       Non-Null Count   Dtype
---  ------       --------------   -----
 0   InvoiceNo    541909 non-null  object
 1   StockCode    541909 non-null  object
 2   Description  540455 non-null  object
 3   Quantity     541909 non-null  int64
 4   InvoiceDate  541909 non-null  object
 5   UnitPrice    541909 non-null  float64
 6   CustomerID   406829 non-null  float64
 7   Country      541909 non-null  object
dtypes: float64(2), int64(1), object(5)
memory usage: 33.1+ MB
```

```
sales.describe()
```

	Quantity	UnitPrice	CustomerID
count	541909.000000	541909.000000	406829.000000
mean	9.552250	4.611114	15287.690570
std	218.081158	96.759853	1713.600303
min	-80995.000000	-11062.060000	12346.000000
25%	1.000000	1.250000	13953.000000
50%	3.000000	2.080000	15152.000000
75%	10.000000	4.130000	16791.000000
max	80995.000000	38970.000000	18287.000000

```
sales.shape
```

```
(541909, 8)
```

- Checking missing values and filling in these missing values:

```
# Checking missing value
sales.isna().sum()
```

```
InvoiceNo          0
StockCode          0
Description     1454
Quantity           0
InvoiceDate        0
UnitPrice          0
CustomerID    135080
Country            0
dtype: int64
```

```
import random

# Handle missing values

# fill missing descriptions with "Unknown":
sales['Description'].fillna('Unknown', inplace=True)

# Filling missing customer IDs with random values
sales['CustomerID'].fillna(random.randint(10000, 99999), inplace=True)

# Checking after filling missing values:
sales.isna().sum()
```

- Checking duplicate values and removing:

```
# Remove duplicated values
sales = sales.drop_duplicates()

#Checking after removing duplicate values:

duplicate_count = sales.duplicated().sum()
duplicate_rows = sales[sales.duplicated(keep=False)]
print("Number of duplicate rows:", duplicate_count)
sales.shape
```

```
Number of duplicate rows: 0
(536641, 8)
```

```
import random

# Handle missing values

# fill missing descriptions with "Unknown":
sales['Description'].fillna('Unknown', inplace=True)

# Filling missing customer IDs with random values
sales['CustomerID'].fillna(random.randint(10000, 99999), inplace=True)

# Checking after filling missing values:
sales.isna().sum()
```

```
InvoiceNo      0
StockCode      0
Description    0
Quantity       0
InvoiceDate    0
UnitPrice      0
CustomerID     0
Country        0
dtype: int64
```

- Checking negative and 0 values in Quantity and Unitprice and removing:

```
# Checking negative and 0 values of Quantity and UnitPrice:


negative_unit_price = sales[sales['UnitPrice'] <= 0]

negative_quantity = sales[sales['Quantity'] <= 0]

# Print the count of negative and 0 values:

print("\nCount of negative and 0 UnitPrice values:", len(negative_unit_price))
print("Count of negative and 0 Quantity values:", len(negative_quantity))
```

```
Count of negative and 0 UnitPrice values: 2512
Count of negative and 0 Quantity values: 10587
```

```
# Removing negative values from Quantities and UnitPrice column

sales=sales[(sales['Quantity']>0) & (sales['UnitPrice']>0)]

# Checking cancelled invoice:

sales['InvoiceNo'].apply(str).str.startswith('C').sum()
```

```
0
```

```
sales.shape
```

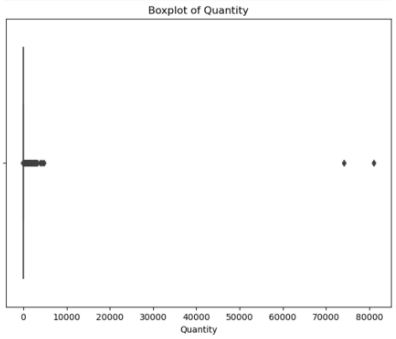
```
(524878, 8)
```

- Detecting outliers of Quantity attribute and remove all of these outliers:

```
# Detecing outliers of Quantity:

Q1 = sales['Quantity'].quantile(0.25)
Q3 = sales['Quantity'].quantile(0.75)
IQR = Q3 - Q1

lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR

outliers = sales[(sales['Quantity'] < lower_bound) | (sales['Quantity'] > upper_bound)]

total_data_count = len(sales)
total_outliers_percentage = len(outliers) / len(sales)
print("Number of data count:",len(sales))
print("Number of outliers:", len(outliers))
print("Outliers percentage (%):", (len(outliers) / len(sales)*100))
```

```
Number of data count: 524878
Number of outliers: 27111
Outliers percentage (%): 5.165200294163596
```

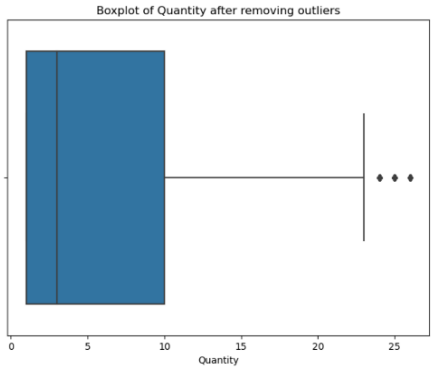
```
# Visualizing outliers of Quantity:

plt.figure(figsize=(8, 6))
sns.boxplot(x=sales["Quantity"])
plt.title("Boxplot of Quantity")
plt.show()
```



```
# Removing outliers of Quantity and UnitPrice

# Calculate percentiles of Quantity:
q_low = sales["Quantity"].quantile(0.25)
q_hi  = sales["Quantity"].quantile(0.75)

IQR=q_hi-q_low

# Filter out outliers of Quantity:
sales=sales[~((sales['Quantity']<(q_low-1.5*IQR)) | (sales['Quantity']>(q_hi+1.5*IQR)))]

print('Dataset shape after removing outliers: ',sales.shape)
```

```
Dataset shape after removing outliers:  (497767, 8)
```

```
# boxplot to visualize after removing the outliers:

plt.figure(figsize=(8, 6))
sns.boxplot(x=sales["Quantity"])
plt.title("Boxplot of Quantity after removing outliers")
plt.show()
```



- Creating new feature TotalSales:

```
# Creating new feature sales from Quantity and Unitprice:

sales['TotalSales'] = sales['Quantity'] * sales['UnitPrice']

print(sales.head())
```

```
  InvoiceNo StockCode                          Description  Quantity  \
0    536365    85123A   WHITE HANGING HEART T-LIGHT HOLDER         6
1    536365     71053                  WHITE METAL LANTERN         6
2    536365    84406B       CREAM CUPID HEARTS COAT HANGER         8
3    536365    84029G  KNITTED UNION FLAG HOT WATER BOTTLE         6
4    536365    84029E       RED WOOLLY HOTTIE WHITE HEART.         6

      InvoiceDate  UnitPrice  CustomerID         Country  TotalSales
0  12/1/2010 8:26       2.55     17850.0  United Kingdom       15.30
1  12/1/2010 8:26       3.39     17850.0  United Kingdom       20.34
2  12/1/2010 8:26       2.75     17850.0  United Kingdom       22.00
3  12/1/2010 8:26       3.39     17850.0  United Kingdom       20.34
4  12/1/2010 8:26       3.39     17850.0  United Kingdom       20.34
```

```
# Print new data with TotalSales feature

total_sales.to_csv('total_sales.csv', index=False)
```

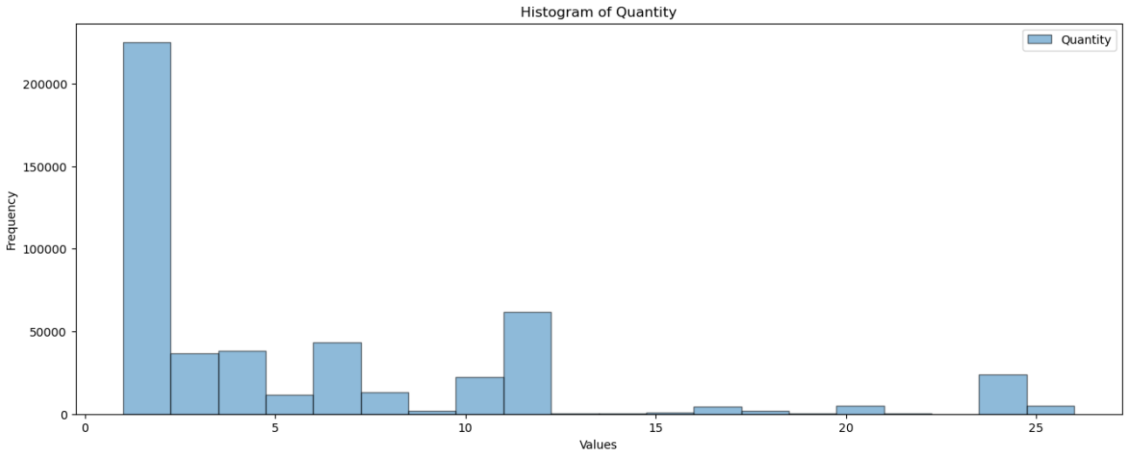
- Understanding datatype, central tendency, and data shape of the new data:

```
# Central Tendency:
total_sales.describe()
```

	Quantity	UnitPrice	CustomerID	TotalSales
count	497767.000000	497767.000000	497767.000000	497767.000000
mean	5.883108	4.049788	29218.522317	13.593931
std	6.281662	37.045316	23362.374963	40.210337
min	1.000000	0.001000	12347.000000	0.001000
25%	1.000000	1.250000	14422.000000	3.750000
50%	3.000000	2.100000	16364.000000	8.850000
75%	10.000000	4.130000	68285.000000	16.630000
max	26.000000	13541.330000	68285.000000	13541.330000

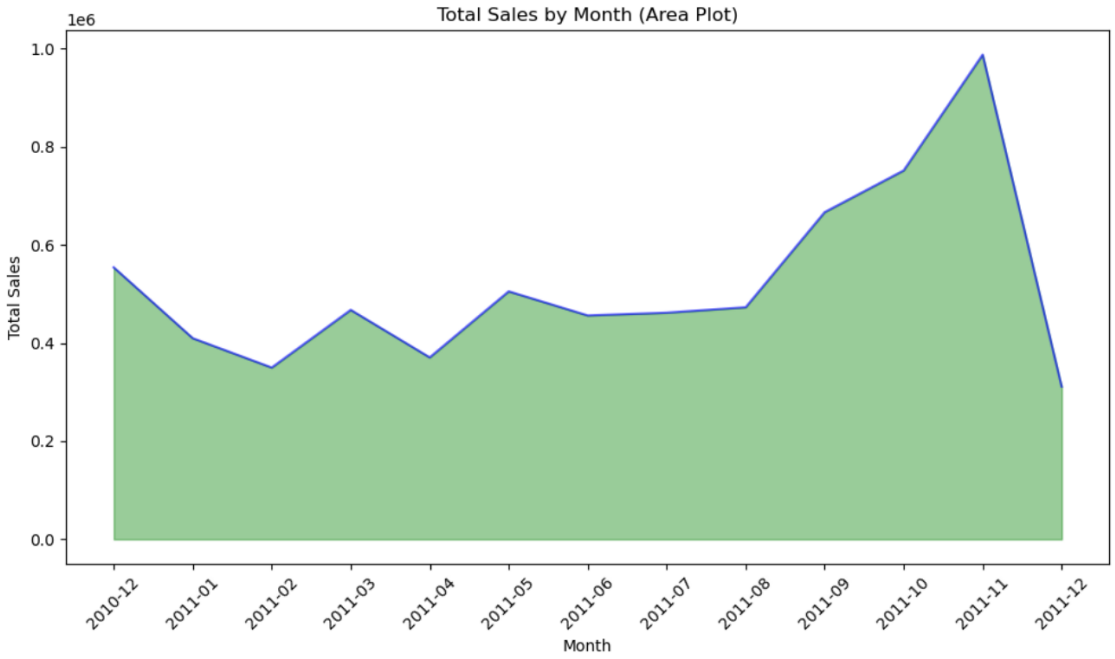
3. Exploring Data Analysis and Data Storytelling

3.1. The distribution of Quantity



The Histogram indicates the distribution of the Quantity attribute after removing outliers, the figure shows that the value of 2 has the most frequency, which is more than 200000. The highest frequency value is extremely far from the other values under 50000.

3.2. Total Sales by Month



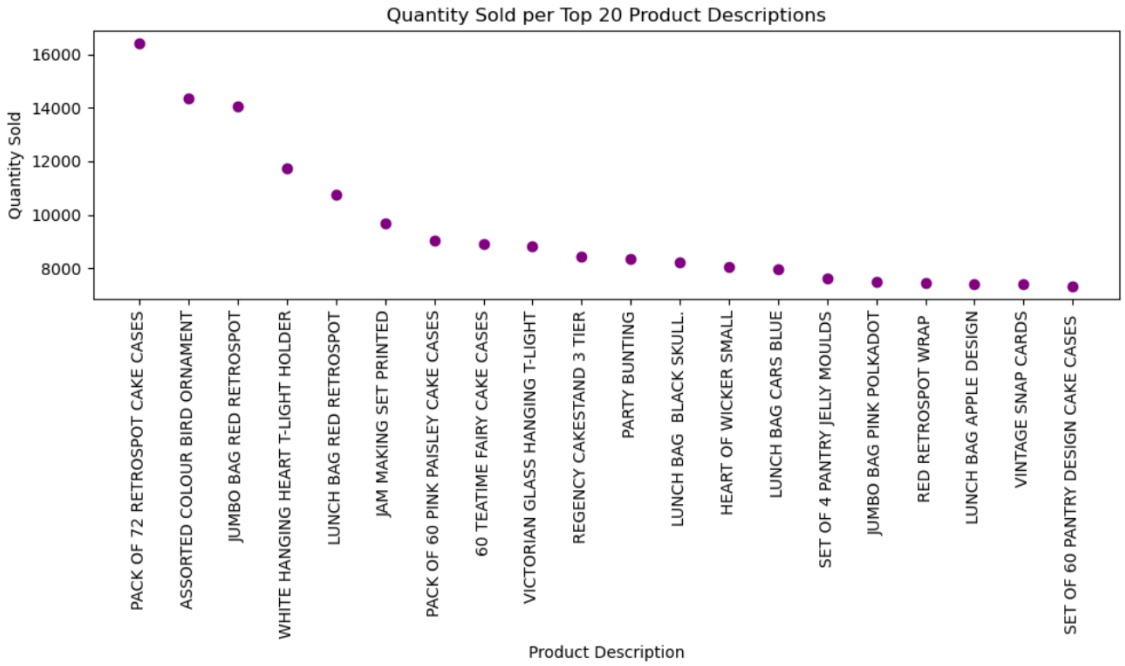
The Area Plot shows the Total Sales between December 2010 and December 2011. As can be seen, the Total Sales fluctuated at the beginning of the chart until August 2011. Then, the Total Sales peaked in November 2011, which was nearly 1 million, before dropping significantly in December 2021.

3.3. Word Cloud of Product Descriptions

Number of Unique product descriptions: 4007

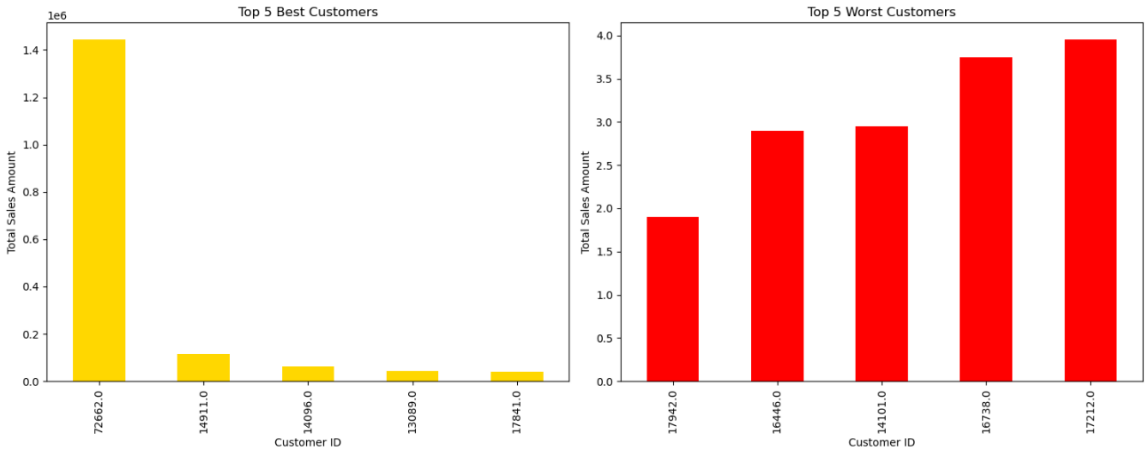
A Word Cloud generated from the Descriptions attribute visualizes the frequency of words in those descriptions, with the size of each word indicating its frequency. As can be seen from Word Cloud, words that appear more frequently in the descriptions will be displayed with larger font sizes which are Regency, Cakestand, 3, Tire, White, Hanging, Heart, T-Light, and Holder compared with a total of over 4000 unit Descriptions.

3.4. The Quantity sold per top 20 Product Description



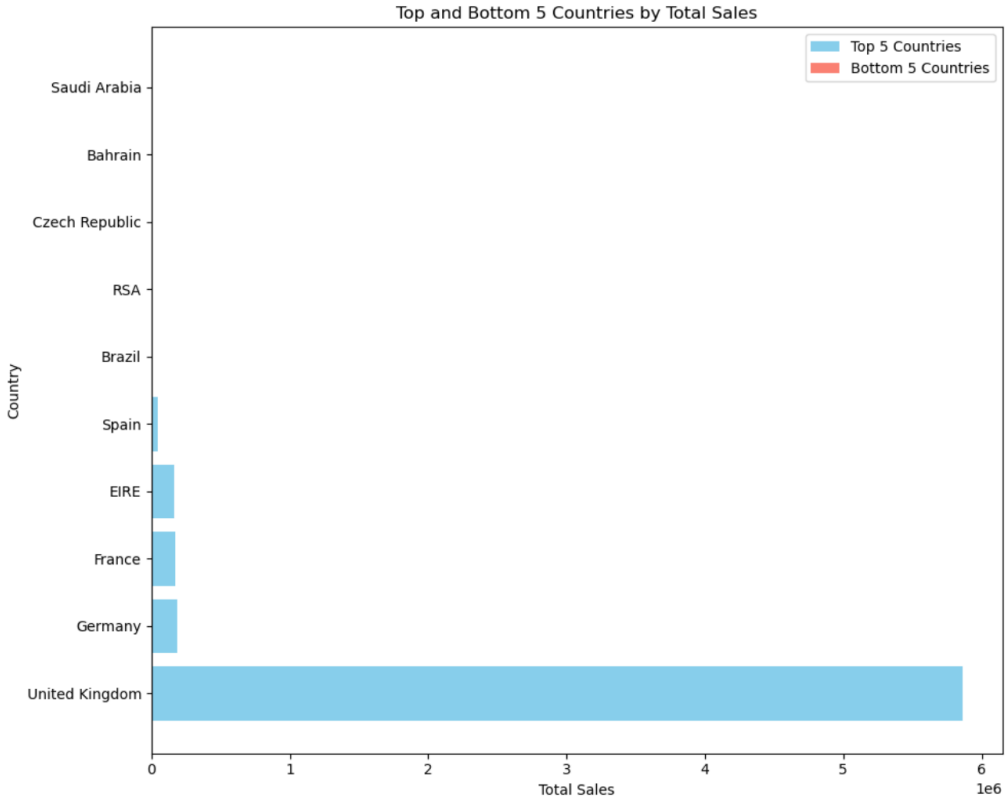
The Scatter Plot illustrates the Quantity sold per Top 20 Product Descriptions. As can be seen, the top 5 Product Descriptions have over 10000 Quantities sold, while 15 other Product Descriptions have the quantity sold between 8000 to 10000.

3.5. Total sales of the Top and Bottom 5 customers



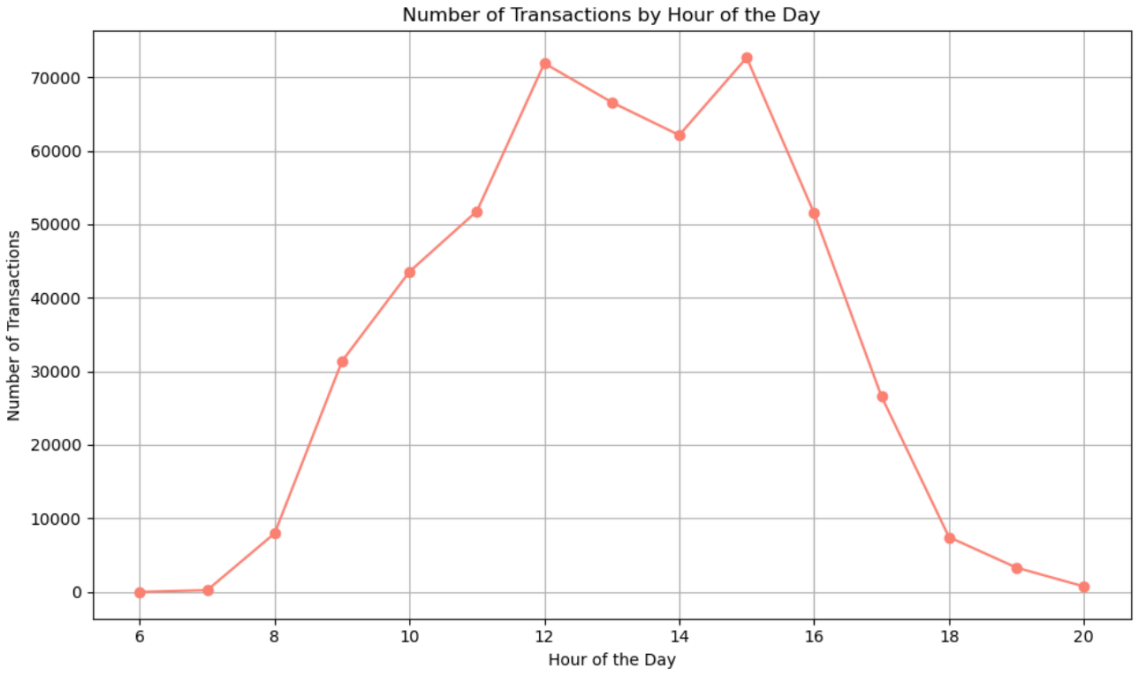
The top and bottom five Customers are depicted by the bar chart. Customer ID 72662.0 has the highest Total Sales Amount which is over 1.4 million, while the other of the top five has under 0.2 million. On the other hand, Customer ID 17942.0 has the lowest Total Sales Amount which is 2, and the orders of the bottom have sales between 2.5 and 4.

3.6. Total sales of the Top and Bottom 5 countries



The Horizontal Bar Chart is used to visualize the Top and Bottom five countries by Total Sales. As the chart shows, the United Kingdom has the highest Total Sales, which is nearly 6 million, while Spain, EIRE, France, and Germany's Total Sales are around 0.5 million. On the other hand, Saudi Arabia, Bahrain, the Czech Republic, and RSA are the countries that have the lowest Total Sales, which is almost 0.

3.7. Number of Transactions by Hour of the Day



The Line Plot is used to depict the number of Transactions by Hour of the Day. As can be seen, all of the transactions are done between 6 o'clock to 20 o'clock. The number of transactions starts to increase at 8 o'clock from nearly 10000 to reach a peak of over 70000. After slightly decreasing until 14 o'clock, its transactions reach a peak again at around 15 o'clock. Then, its figure fell until 20 o'clock.

Conclusion

During this project, we learned a lot of the work and best practices that go into studying a dataset, along with the process of using Python from the Data Pre-processing, and then going

through the Explore Data Analysis along with doing Data Storytelling. Other than this, we were able to learn the concepts of storytelling and presentation to reach the audience with the best information and explanation. Moreover, the outcome visualization of this dataset helped us to get the best understanding of sales trends such as November having the highest sales of the year, and the time that most of the transactions were done. Besides, these visualizations also give insights into every product, customer, and country. This information can be useful in decision-making to improve the business overall.

Appendix

Appendix 1: Group Meeting Log

Date	Topic	PIC	Deadline	Tracking
12-Oct	Selecting data and submit for getting approval	Thi Minh Ngoc Bui	13-Oct	On time
20-Oct	Allocate tasks and divided into 2 sub-groups:			On time
	- Data collection, Data Pre-processing, combination group work	Thi Minh Ngoc Bui Shijie Liu	30-Oct	On time
	- Exploring Data Analysis and Data Storytelling	Simarpreet Kaur Krina Chiragkumar Patel Rutvi Dixitkumar Patel	20-Nov	On time
23-Oct	Working on describe variable, loading data into Python and doing some basis statistics	Thi Minh Ngoc Bui Shijie Liu		On time
27-Oct	Working on cleaning dataset			On time
29-Oct	Informing a confusion between outliers at Quantitive and UnitPrice, discussing then aligning solution			On time
3-Nov	Done with data collection and Data Pre-processing parts, sending working file for next phase			On time
8-Nov	Done with the code of Distribution of Quantity, Total Sales by Month, Word Cloud	Krina Chiragkumar Patel Rutvi Dixitkumar Patel	8-Nov	On time
15-Nov	Done with the code of The Quantity sold per top 20 product descriptions, Top and Bottom 5 of customers by Total Sales, Top and Bottom 5 of countries by Total Sales, and the Number of Transactions by Hour of the Day		14-Nov	On time
19-Nov	Done with storytelling by using visualization from Krina and Rutvi work	Simarpreet Kaur	20-Nov	On time
20-Nov	Done with combination of Python file, Word file, and PPT	Thi Minh Ngoc Bui Shijie Liu	22-Nov	On time

Appendix 2: Table of Contribution

First Name	Last Name	Email	Student ID	Contribution
Thi Minh Ngoc	Bui	CBU22CPWQ@cbu.ca	0280860	20%
Kepler	Lau	CBU22BTWD@cbu.ca	0276875	20%
Simarpreet	Kaur	CBU22CLBN@cbu.ca	0285168	20%
Krina	Chiragkumar Patel	CBU22CGXL@cbu.ca	0283599	20%
Rutvi	Dixitkumar Patel	CBU22CKJW@cbu.ca	0284699	20%