Darshan UNIVERSITY
योग: कर्मसु कौशलम्

Data Mining

Project Preprocessing (Part-1)

137 | Vishal Baraiya | 23010101014

1. Import the pandas Libraries and Read csv or excel File

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

In [2]:
```
# Try reading the CSV with ISO-8859-1 encodingo
OnlineRetail = pd.read_csv("D:\\VS_CODES\\DataMining\\ProjectDataMining\\Dataset\\O
OnlineRetail
```

Out[2]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	12/1/2010 8:26	2.55	17850.0
1	536365	71053	WHITE METAL LANTERN	6	12/1/2010 8:26	3.39	17850.0
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	12/1/2010 8:26	2.75	17850.0
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	12/1/2010 8:26	3.39	17850.0
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	12/1/2010 8:26	3.39	17850.0
...	...	...	...	...	...	...	...
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	12/9/2011 12:50	0.85	12680.0
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	12/9/2011 12:50	2.10	12680.0
541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	12/9/2011 12:50	4.15	12680.0
541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	12/9/2011 12:50	4.15	12680.0
541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	12/9/2011 12:50	4.95	12680.0

541909 rows × 8 columns

2. Print All columns.

In [3]:
```
OnlineRetail.columns
```

Out[3]:
```
Index(['InvoiceNo', 'StockCode', 'Description', 'Quantity', 'InvoiceDate',
       'UnitPrice', 'CustomerID', 'Country'],
      dtype='object')
```

3. Print number of rows and columns.

In [4]:
```
OnlineRetail.shape
```

Out[4]:
```
(541909, 8)
```

4. See the First 10 Rows

In [5]:
```
OnlineRetail.head(10)
```

Out[5]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Cou
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	12/1/2010 8:26	2.55	17850.0	Un Kingd
1	536365	71053	WHITE METAL LANTERN	6	12/1/2010 8:26	3.39	17850.0	Un Kingd
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	12/1/2010 8:26	2.75	17850.0	Un Kingd
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	12/1/2010 8:26	3.39	17850.0	Un Kingd
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	12/1/2010 8:26	3.39	17850.0	Un Kingd
5	536365	22752	SET 7 BABUSHKA NESTING BOXES	2	12/1/2010 8:26	7.65	17850.0	Un Kingd
6	536365	21730	GLASS STAR FROSTED T-LIGHT HOLDER	6	12/1/2010 8:26	4.25	17850.0	Un Kingd
7	536366	22633	HAND WARMER UNION JACK	6	12/1/2010 8:28	1.85	17850.0	Un Kingd
8	536366	22632	HAND WARMER RED POLKA DOT	6	12/1/2010 8:28	1.85	17850.0	Un Kingd
9	536367	84879	ASSORTED COLOUR BIRD ORNAMENT	32	12/1/2010 8:34	1.69	13047.0	Un Kingd

5. See the Last 10 Rows.

In [6]:
```
OnlineRetail.tail(10)
```

Out[6]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID
541899	581587	22726	ALARM CLOCK BAKELIKE GREEN	4	12/9/2011 12:50	3.75	12680.0
541900	581587	22730	ALARM CLOCK BAKELIKE IVORY	4	12/9/2011 12:50	3.75	12680.0
541901	581587	22367	CHILDRENS APRON SPACEBOY DESIGN	8	12/9/2011 12:50	1.95	12680.0
541902	581587	22629	SPACEBOY LUNCH BOX	12	12/9/2011 12:50	1.95	12680.0
541903	581587	23256	CHILDRENS CUTLERY SPACEBOY	4	12/9/2011 12:50	4.15	12680.0
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	12/9/2011 12:50	0.85	12680.0
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	12/9/2011 12:50	2.10	12680.0
541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	12/9/2011 12:50	4.15	12680.0
541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	12/9/2011 12:50	4.15	12680.0
541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	12/9/2011 12:50	4.95	12680.0

6. Data type of each columns.

In [7]:
```
OnlineRetail.dtypes
```

Out[7]:
```
InvoiceNo       object
StockCode       object
Description     object
Quantity         int64
InvoiceDate     object
UnitPrice      float64
CustomerID     float64
Country         object
dtype: object
```

7. Display Summary Information

In [8]:
```
OnlineRetail.describe()
```

Out[8]:

	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

8. Access a specific column

In [9]:
```
OnlineRetail["Description"]
```

Out[9]:
```
0          WHITE HANGING HEART T-LIGHT HOLDER
1                         WHITE METAL LANTERN
2              CREAM CUPID HEARTS COAT HANGER
3         KNITTED UNION FLAG HOT WATER BOTTLE
4              RED WOOLLY HOTTIE WHITE HEART.
                          ...
541904            PACK OF 20 SPACEBOY NAPKINS
541905          CHILDREN'S APRON DOLLY GIRL
541906          CHILDRENS CUTLERY DOLLY GIRL
541907        CHILDRENS CUTLERY CIRCUS PARADE
541908          BAKING SET 9 PIECE RETROSPOT
Name: Description, Length: 541909, dtype: object
```

9. Access a Multiple column

In [10]:
```
OnlineRetail[["Description","StockCode"]]
```

Out[10]:

	Description	StockCode
0	WHITE HANGING HEART T-LIGHT HOLDER	85123A
1	WHITE METAL LANTERN	71053
2	CREAM CUPID HEARTS COAT HANGER	84406B
3	KNITTED UNION FLAG HOT WATER BOTTLE	84029G
4	RED WOOLLY HOTTIE WHITE HEART.	84029E
...	...	...
541904	PACK OF 20 SPACEBOY NAPKINS	22613
541905	CHILDREN'S APRON DOLLY GIRL	22899
541906	CHILDRENS CUTLERY DOLLY GIRL	23254
541907	CHILDRENS CUTLERY CIRCUS PARADE	23255
541908	BAKING SET 9 PIECE RETROSPOT	22138

541909 rows × 2 columns

10. Access rows by their integer location.

In [10]:
```
OnlineRetail.iloc[0]
```

Out[10]:
```
InvoiceNo                                   536365
StockCode                                   85123A
Description    WHITE HANGING HEART T-LIGHT HOLDER
Quantity                                        6
InvoiceDate                       12/1/2010 8:26
UnitPrice                                    2.55
CustomerID                                17850.0
Country                           United Kingdom
Name: 0, dtype: object
```

In [11]:
```
OnlineRetail.iloc[[0, 2]]
```

Out[11]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Cou
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	12/1/2010 8:26	2.55	17850.0	Un Kingd
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	12/1/2010 8:26	2.75	17850.0	Un Kingd

11. Perform Condition Selection on DataFrame

In [12]:
```
OnlineRetail[OnlineRetail["Quantity"] > 10]
```

Out[12]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID
9	536367	84879	ASSORTED COLOUR BIRD ORNAMENT	32	12/1/2010 8:34	1.69	13047.(
26	536370	22728	ALARM CLOCK BAKELIKE PINK	24	12/1/2010 8:45	3.75	12583.(
27	536370	22727	ALARM CLOCK BAKELIKE RED	24	12/1/2010 8:45	3.75	12583.(
28	536370	22726	ALARM CLOCK BAKELIKE GREEN	12	12/1/2010 8:45	3.75	12583.(
29	536370	21724	PANDA AND BUNNIES STICKER SHEET	12	12/1/2010 8:45	0.85	12583.(
...	...	...	...	...	...	...	.
541894	581587	22631	CIRCUS PARADE LUNCH BOX	12	12/9/2011 12:50	1.95	12680.(
541895	581587	22556	PLASTERS IN TIN CIRCUS PARADE	12	12/9/2011 12:50	1.65	12680.(
541896	581587	22555	PLASTERS IN TIN STRONGMAN	12	12/9/2011 12:50	1.65	12680.(
541902	581587	22629	SPACEBOY LUNCH BOX	12	12/9/2011 12:50	1.95	12680.(
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	12/9/2011 12:50	0.85	12680.(

132631 rows × 8 columns

12. Compute the sum of value

In [13]:
```
OnlineRetail["UnitPrice"].mean()
```

Out[13]: 4.611113626088513

13. Compute the non-null values

In [14]:
```
(~OnlineRetail.isnull()).sum()
```

Out[14]:
```
InvoiceNo      541909
StockCode      541909
Description    540455
Quantity       541909
InvoiceDate    541909
UnitPrice      541909
CustomerID     406829
Country        541909
dtype: int64
```

14. Compute the Minimun or Maximum values

In [15]:
```
OnlineRetail["UnitPrice"].min()
```

Out[15]: -11062.06

In [16]:
```
OnlineRetail["UnitPrice"].max()
```

Out[16]: 38970.0

In [17]:
```
OnlineRetail["UnitPrice"].median()
```

Out[17]: 2.08

In [18]:
```
OnlineRetail["UnitPrice"].mode()
```

Out[18]:
```
0    1.25
Name: UnitPrice, dtype: float64
```

In [19]:
```
OnlineRetail["UnitPrice"].std()
```

Out[19]: 96.75985306119716

15. Convert To Excel File

In [21]:
```
# genrate a excel file
OnlineRetail.to_excel("output.xlsx")
```

16. How many different Country are in this dataset?

In [22]:
```
OnlineRetail.Country.nunique()
```

Out[22]: 38

17. Convert all values in the 'InvoiceDate' column to string (str) data type.

In [23]:
```
OnlineRetail['InvoiceDate'].astype(str)
OnlineRetail
```

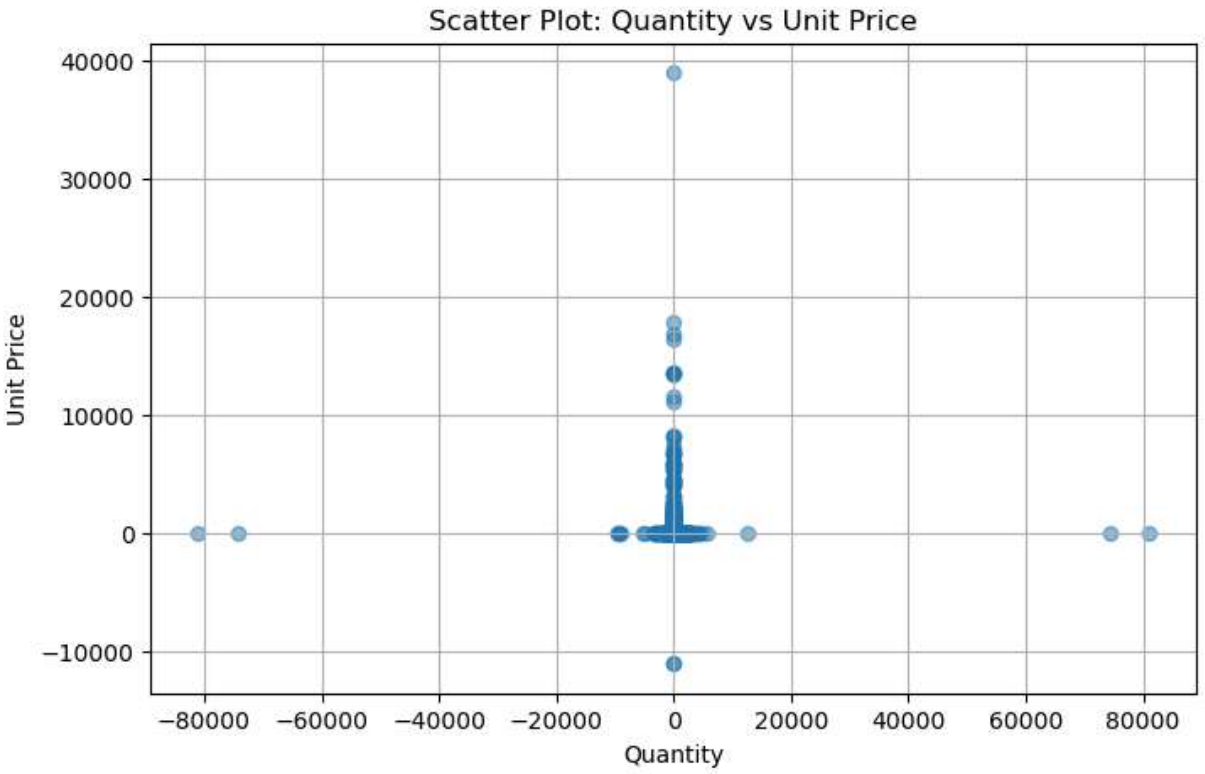
Out[23]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	12/1/2010 8:26	2.55	17850.0
1	536365	71053	WHITE METAL LANTERN	6	12/1/2010 8:26	3.39	17850.0
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	12/1/2010 8:26	2.75	17850.0
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	12/1/2010 8:26	3.39	17850.0
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	12/1/2010 8:26	3.39	17850.0
...	...	...	...	...	...	...	...
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	12/9/2011 12:50	0.85	12680.0
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	12/9/2011 12:50	2.10	12680.0
541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	12/9/2011 12:50	4.15	12680.0
541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	12/9/2011 12:50	4.15	12680.0
541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	12/9/2011 12:50	4.95	12680.0

541909 rows × 8 columns

18. Scatter Plot of Unit Price vs Quantity

In [24]:
```
plt.figure(figsize=(8, 5))
plt.scatter(OnlineRetail['Quantity'], OnlineRetail['UnitPrice'], alpha=0.5)
plt.xlabel('Quantity')
plt.ylabel('Unit Price')
plt.title('Scatter Plot: Quantity vs Unit Price')
plt.grid(True)
plt.show()
```

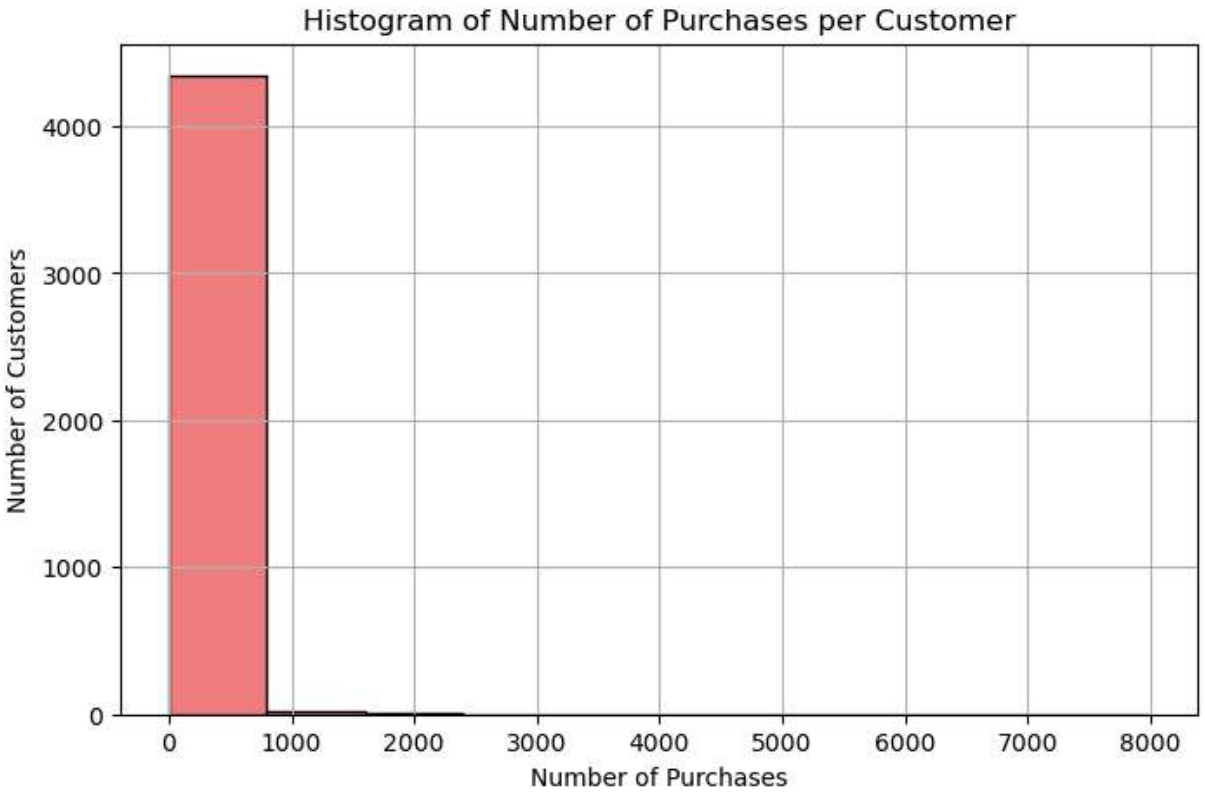


19. Box Plot of UnitPrice

In [25]:
```
OnlineRetail_filtered = OnlineRetail[OnlineRetail['UnitPrice'] > 0]

# Create box plot
plt.figure(figsize=(6, 5))
plt.boxplot(OnlineRetail_filtered['UnitPrice'])

plt.title('Box Plot of Unit Price')
plt.ylabel('Unit Price')
plt.grid(True)
plt.show()
```

20. Histogram of Number of Purchases per Customer

In [26]:
```
customer_counts = OnlineRetail['CustomerID'].value_counts()

# Plot histogram of purchase frequency
plt.figure(figsize=(8, 5))
plt.hist(customer_counts, bins=10, color='lightcoral', edgecolor='black')
plt.title('Histogram of Number of Purchases per Customer')
plt.xlabel('Number of Purchases')
plt.ylabel('Number of Customers')
plt.grid(True)
plt.show()
```



21. compute the covariance Matrix

In [27]:
```
print("Covariance Matrix : ")
OnlineRetail.cov(numeric_only=True)
```

```
Covariance Matrix :
```

Out[27]:

	Quantity	UnitPrice	CustomerID
Quantity	47559.391409	-26.058761	-1.534050e+03
UnitPrice	-26.058761	9362.469164	-5.415793e+02
CustomerID	-1534.050176	-541.579276	2.936426e+06

22. compute the correlation Matrix

In [28]:
```
print("Correlation Matrix : ")
OnlineRetail.corr(numeric_only=True)
```

```
Correlation Matrix :
```

Out[28]:

	Quantity	UnitPrice	CustomerID
Quantity	1.000000	-0.001235	-0.00360
UnitPrice	-0.001235	1.000000	-0.00456
CustomerID	-0.003600	-0.004560	1.00000

23. How many products were sold in each country?

In [29]:
```
OnlineRetail.groupby('Country')['Quantity'].sum().sort_values(ascending=False)
```

Out[29]:
```
Country
United Kingdom          4263829
Netherlands              200128
EIRE                     142637
Germany                  117448
France                   110480
Australia                 83653
Sweden                    35637
Switzerland               30325
Spain                     26824
Japan                     25218
Belgium                   23152
Norway                    19247
Portugal                  16180
Finland                   10666
Channel Islands            9479
Denmark                    8188
Italy                      7999
Cyprus                     6317
Singapore                  5234
Austria                    4827
Hong Kong                  4769
Israel                     4353
Poland                     3653
Unspecified                3300
Canada                     2763
Iceland                    2458
Greece                     1556
USA                        1034
United Arab Emirates        982
Malta                       944
Lithuania                   652
Czech Republic              592
European Community          497
Lebanon                     386
Brazil                      356
RSA                         352
Bahrain                     260
Saudi Arabia                 75
Name: Quantity, dtype: int64
```

24. What is the average price of each product?

In [30]:
```
OnlineRetail.groupby('StockCode')['UnitPrice'].mean().sort_values(ascending=False)
```

Out[30]:
```
StockCode
AMAZONFEE       7324.784706
CRUK             495.839375
M                375.566392
DOT              290.495859
BANK CHARGES     202.855162
                    ...
84526              0.000000
72732              0.000000
85018B             0.000000
35824B             0.000000
B              -3687.353333
Name: UnitPrice, Length: 4070, dtype: float64
```

25. How many purchases has each customer made?

In [31]:
```
OnlineRetail.groupby('CustomerID')['InvoiceNo'].nunique().sort_values(ascending=Fal
```

Out[31]:
```
CustomerID
14911.0    248
12748.0    224
17841.0    169
14606.0    128
13089.0    118
           ...
13877.0      1
16400.0      1
13878.0      1
13886.0      1
13670.0      1
Name: InvoiceNo, Length: 4372, dtype: int64
```

In [ ]: