Institute of Technology of Cambodia

Department of Information and Communication Engineering

Subject: Research Methodology

Report Assignment-3

Professor: SOK Kimheng

Student: YORN Vanda

ID: e20181287

Group: I5-C

2022-2023

Report

- **About Dataset**

  Our dataset describes an international e-commerce company based wants to discover key insights from their customer database. The owner of this dataset wants to use some of the most advanced machine-learning techniques to study their customers. The company sells electronic products.

  The data points in this dataset contained 10999 observations of 12 variables. The data contains the following information:

  - **ID:** ID Number of Customers.
  - **Warehouse block:** The Company have a big Warehouse which is divided into block such as A, B, C, D, E.
  - **Mode of shipment:** The Company Ships the products in a multiple way such as Ship, Flight and Road.
  - **Customer care calls:** The number of calls made from the enquiry for enquiry of the shipment.
  - **Customer rating:** The company has rated from every customer. 1 is the lowest (Worst), and 5 is the highest (Best).
  - **Cost of the product:** Cost of the Product in US Dollars.
  - **Prior purchases:** The Number of Prior Purchases.
  - **Product importance:** The company has categorized the product in the various parameter such as low, medium, high.
  - **Gender:** Male and Female.
  - **Discount offered:** Discount offered on that specific product.
  - **Weight in gms:** It is the weight in grams.
  - **Reached on time:** It is the target variable, where 1 Indicates that the product has NOT reached on time and 0 indicates it has reached on time.

The below figure is showing about the code and libraries that import and show the first 10 data points which are contained in the dataset

```
In [52]: # import Panda Library to read CSV file
         import pandas as pd
         import scipy
         import numpy as np
         import seaborn as sns
         import matplotlib.pyplot as plt
         from statistics import mean, median, mode, stdev, variance


         # define variable data to represent the whole data set
         data = pd.read_csv('shipment.csv')

         #Display number of row and column of data set
         data.shape

         # Display the last 10 values of the data set
         #data.tail(10)
         data.head(10)
         #data.describe()
```

ID	Warehouse_block	Mode_of_Shipment	Customer_care_calls	Customer_rating	Cost_of_the_Product	Prior_purchases	Product_importance	Gender	Discount_offered	Weight_in_gms	Reached.on.Time_Y.N
1	D	Flight	4	2	177	3	low	F	44	1233	1
2	F	Flight	4	5	216	2	low	M	59	3088	1
3	A	Flight	2	2	183	4	low	M	48	3374	1
4	B	Flight	3	3	176	4	medium	M	10	1177	1
5	C	Flight	2	2	184	3	medium	F	46	2484	1
6	F	Flight	3	1	162	3	medium	F	12	1417	1
7	D	Flight	3	4	250	3	low	F	3	2371	1
8	F	Flight	4	1	233	2	low	F	48	2804	1
9	A	Flight	3	4	150	3	low	F	11	1861	1
10	B	Flight	3	2	164	3	medium	F	29	1187	1

- **Implementation on Statistic Terminology**

Before go deeper into the dataset, we should know the purpose of the implementation of the dataset.

In the dataset implementation we are going to work on the shipment type, warehouses, and cost of product of the E-Commerce's shipment.

The below descriptive will explain more detail about the implementation:

⇒ **First, I split shipment type into 3 differences**:

~ For Shipment type = Ship

```
# Spliting Mode_of_Shipment into difference part. (Flight)
flight_mode = data[data.Mode_of_Shipment=='Flight']
flight_mode.head(10)
#flight_mode.shape
```

	ID	Warehouse_block	Mode_of_Shipment	Customer_care_calls	Customer_rating	Cost_of_the_Product	Prior_purchases	Product_importance	Gender
0	1	D	Flight	4	2	177	3	low	F
1	2	F	Flight	4	5	216	2	low	M
2	3	A	Flight	2	2	183	4	low	M
3	4	B	Flight	3	3	176	4	medium	M
4	5	C	Flight	2	2	184	3	medium	F
5	6	F	Flight	3	1	162	3	medium	F
6	7	D	Flight	3	4	250	3	low	F
7	8	F	Flight	4	1	233	2	low	F
8	9	A	Flight	3	4	150	3	low	F
9	10	B	Flight	3	2	164	3	medium	F

After splitting, we initialized into a new dataset which contained 7462 observations of 12 variables:

In [3]:
```
# Spliting Mode_of_Shipment into difference part. (Ship)

shiping_mode = data[data.Mode_of_Shipment=='Ship']
shiping_mode.tail()
shiping_mode.shape
```

Out[3]: (7462, 12)

~ For Shipment type = Flight

In [58]:
```
# Spliting Mode_of_Shipment into difference part. (Flight)
flight_mode = data[data.Mode_of_Shipment=='Flight']
#flight_mode.shape
flight_mode.head(10)
```

Out[58]:

	ID	Warehouse_block	Mode_of_Shipment	Customer_care_calls	Customer_rating	Cost_of_the_Product	Prior_purchases	Product_importance	Gender	Discoun
0	1	D	Flight	4	2	177	3	low	F	
1	2	F	Flight	4	5	216	2	low	M	
2	3	A	Flight	2	2	183	4	low	M	
3	4	B	Flight	3	3	176	4	medium	M	
4	5	C	Flight	2	2	184	3	medium	F	
5	6	F	Flight	3	1	162	3	medium	F	
6	7	D	Flight	3	4	250	3	low	F	
7	8	F	Flight	4	1	233	2	low	F	
8	9	A	Flight	3	4	150	3	low	F	
9	10	B	Flight	3	2	164	3	medium	F	

After splitting, we initialized into a new dataset which contained 1777 observations of 12 variables:

```
In [59]: # Spliting Mode_of_Shipment into difference part. (Flight)
         flight_mode = data[data.Mode_of_Shipment=='Flight']
         flight_mode.shape
         #flight_mode.head(10)

Out[59]: (1777, 12)
```

~ For Shipment type = Road

```
In [60]: # Spliting Mode_of_Shipment into difference part. (Road)
         road_mode = data[data.Mode_of_Shipment=='Road']
         road_mode.head(10)
         #road_mode.shape
```

Out[60]:

	ID	Warehouse_block	Mode_of_Shipment	Customer_care_calls	Customer_rating	Cost_of_the_Product	Prior_purchases	Product_importance	Gender	Disc
110	111	A	Road	3	1	158	2	low	M	
111	112	B	Road	4	2	239	3	low	F	
112	113	C	Road	4	3	175	2	low	M	
113	114	F	Road	3	2	263	3	low	M	
114	115	D	Road	3	5	168	2	high	M	
115	116	F	Road	3	4	176	3	high	M	
116	117	A	Road	4	1	150	4	high	M	
117	118	B	Road	4	3	265	2	medium	F	
118	119	C	Road	2	5	263	3	low	F	
119	120	F	Road	5	1	145	3	low	M	

After splitting, we initialized into a new dataset which contained 1760 observations of 12 variables:

```
In [61]: # Spliting Mode_of_Shipment into difference part. (Road)
         road_mode = data[data.Mode_of_Shipment=='Road']
         #road_mode.head(10)
         road_mode.shape

Out[61]: (1760, 12)
```

After finished splitting and initialized it into a new dataset, we will use it to do the operation of statistic, it would be easier to access and manage the data than using the whole dataset to do the operation, it could be a bit complicated.

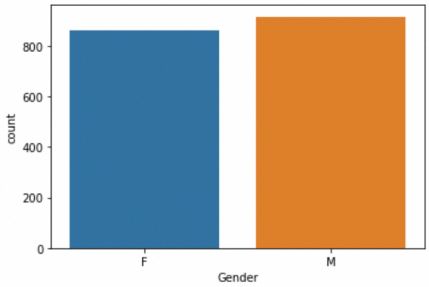
⇒ **Count the number of male and female customers based on Variable "Gender" in each dataset and plot in into bar chart.**

~ For Shipment type = Flight

In the new dataset "Flight_Mode" number of both male and female are1777. In this number we see that number of male customers are more than female 915 and 862, respectively

```
In [6]: # Ploting Gender from Flight mode of Shipment
        sns.countplot(x='Gender', data=flight_mode)

Out[6]: <AxesSubplot:xlabel='Gender', ylabel='count'>
```



Male Customers:

```
In [67]: # Ploting Gender from Flight mode of Shipment
         #sns.countplot(x='Gender', data=flight_mode)
         flight_mode[flight_mode.Gender=="M"].shape

Out[67]: (915, 12)
```

Female Customers:

```
In [66]: # Ploting Gender from Flight mode of Shipment
         #sns.countplot(x='Gender', data=flight_mode)
         flight_mode[flight_mode.Gender=="F"].shape

Out[66]: (862, 12)
```

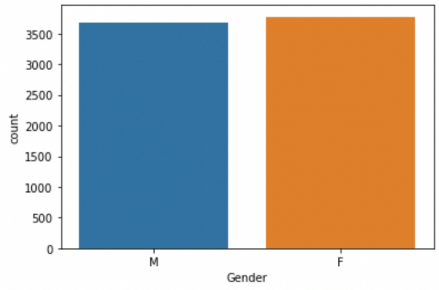
~ For Shipment type = Ship

In the new dataset “Shipping_Mode” number of both male and female are7462. In this number we see that number of female customers are more than male 3775 and 3687, respectively

```
In [75]: # Ploting Gender from Shiping mode of Shipment

         sns.countplot(x='Gender', data=shiping_mode)
         #shiping_mode[shiping_mode.Gender=="M"].shape
```

```
Out[75]: <AxesSubplot:xlabel='Gender', ylabel='count'>
```



Male Customers:

```
In [76]: # Ploting Gender from Shiping mode of Shipment

         #sns.countplot(x='Gender', data=shiping_mode)
         shiping_mode[shiping_mode.Gender=="M"].shape
```

```
Out[76]: (3687, 12)
```

Female Customers:

```
In [77]: # Ploting Gender from Shiping mode of Shipment

         #sns.countplot(x='Gender', data=shiping_mode)
         shiping_mode[shiping_mode.Gender=="F"].shape
```

```
Out[77]: (3775, 12)
```

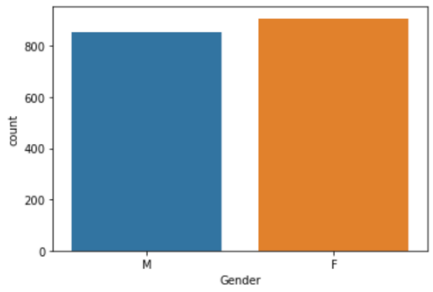
~ For Shipment type = Road

In the new dataset “Road_Mode” number of both male and female are1760. In this number we see that number of male customers are less than female 852 and 902, respectively

```
In [80]: # Ploting Gender from Road mode of Shipment

         sns.countplot(x='Gender', data=road_mode)
         #road_mode[road_mode.Gender=='M'].shape

Out[80]: (852, 12)
```



Male Customers:

```
In [81]: # Ploting Gender from Road mode of Shipment

         #sns.countplot(x='Gender', data=road_mode)
         road_mode[road_mode.Gender=='M'].shape

Out[81]: (852, 12)
```

Female Customers:

```
In [82]: # Ploting Gender from Road mode of Shipment

         #sns.countplot(x='Gender', data=road_mode)
         road_mode[road_mode.Gender=='F'].shape

Out[82]: (908, 12)
```

⇒ Measure of Central tendency (Mean, Median, Mode) of Variable "Cost_of_the_Product"

Due to the E-Commerce busines concept, the Company have big Warehouse which is divided in to block such as A, B, C, D, E.

Since the main purpose of this topic is only to study about the Statistic Terminologies. Since then we choose only one warehouse to do the operation which contained 1833 data points.

*** **NOTE:** Only warehouse A which is studied.

~ For all Shipment type of Warehouse A

```
In [87]: # Define Wharehouse block "A" and find the mean value of the product cost in wharehouse A from the whole data set

         wharehouse_block_A = data[data.Warehouse_block=='A']
         wharehouse_block_A.shape
         #mean(wharehouse_block_A['Cost_of_the_Product'])

Out[87]: (1833, 12)
```

~ For Shipment type = Flight

- Mean Value = 210.037

```
In [89]: # Define Wharehouse block "A" and find the mean value of the product cost in wharehouse A from the flight_mode data set

         wharehouse_block_A1 = flight_mode[flight_mode.Warehouse_block=='A']
         #wharehouse_block_A1.shape
         mean(wharehouse_block_A1['Cost_of_the_Product'])

Out[89]: 210.03703703703704
```

- Median Value = 216

```
In [90]: #Find the median value of the product cost in wharehouse A from the flight_mode data set

         median(wharehouse_block_A1['Cost_of_the_Product'])

Out[90]: 216
```

- Mode Value = 255

```
In [17]: #Find the mode value of the product cost in wharehouse A from the flight_mode data set
         mode(wharehouse_block_A1['Cost_of_the_Product'])

Out[17]: 255
```

~ For Shipment type = Ship

- Mean Value = 208.818

```
In [12]: # Define Wharehouse block "A" and find the mean value of the product cost in wharehouse A from the shiping data set

         wharehouse_block_A2 = shiping_mode[shiping_mode.Warehouse_block=='A']
         mean(wharehouse_block_A2['Cost_of_the_Product'])

Out[12]: 208.81884057971016
```

- Median Value = 212.0

```
In [15]: #Find the median value of the product cost in wharehouse A from the shiping_mode data set

         median(wharehouse_block_A2['Cost_of_the_Product'])

Out[15]: 212.0
```

- Mode Value = 271

```
In [18]: #Find the mode value of the product cost in wharehouse A from the shiping_mode data set

         mode(wharehouse_block_A2['Cost_of_the_Product'])
Out[18]: 271
```

~ For Shipment type = Ship

- Mean Value = 207.268

```
In [13]: ## Define Wharehouse block "A" and find the mean value of the product cost in wharehouse A from the Road data set

         wharehouse_block_A3 = road_mode[road_mode.Warehouse_block=='A']
         mean(wharehouse_block_A3['Cost_of_the_Product'])
Out[13]: 207.2687074829932
```

- Median Value = 207.5

```
In [16]: #Find the median value of the product cost in wharehouse A from the road_mode data set

         median(wharehouse_block_A3['Cost_of_the_Product'])
Out[16]: 207.5
```

- Mode Value = 204

```
In [19]: #Find the mode value of the product cost in wharehouse A from the road_mode data set

         mode(wharehouse_block_A3['Cost_of_the_Product'])
Out[19]: 204
```

⇒ **Measure of spread (variance, standard deviation) of Variable "Cost_of_the_Product"**

~ For Shipment type = Flight

- Variance Value = 2371.002

```
In [20]: #Find the variance value of the product cost in wharehouse A from the flight_mode data set

         variance(wharehouse_block_A1['Cost_of_the_Product'])
Out[20]: 2317.002002002002
```

- Standard Deviation Value = 48.135

```
In [23]: #Find the standard deviation value of the product cost in wharehouse A from the flight_mode data set
         stdev(wharehouse_block_A1['Cost_of_the_Product'])
Out[23]: 48.13524698183237
```

~ For Shipment type = Ship

- Variance Value = 2397.615

```
In [21]: #Find the variance value of the product cost in wharehouse A from the shiping_mode data set
         variance(wharehouse_block_A2['Cost_of_the_Product'])
Out[21]: 2397.615019444347
```

- Standard Deviation Value = 48.965

```
In [24]: #Find the standard deviation value of the product cost in wharehouse A from the shiping_mode data set
         stdev(wharehouse_block_A2['Cost_of_the_Product'])
Out[24]: 48.965447199472685
```

~ For Shipment type = Road

- Variance Value = 2282.579

```
In [22]: #Find the variance value of the product cost in wharehouse A from the road_mode data set
         variance(wharehouse_block_A3['Cost_of_the_Product'])
Out[22]: 2282.57942699264
```

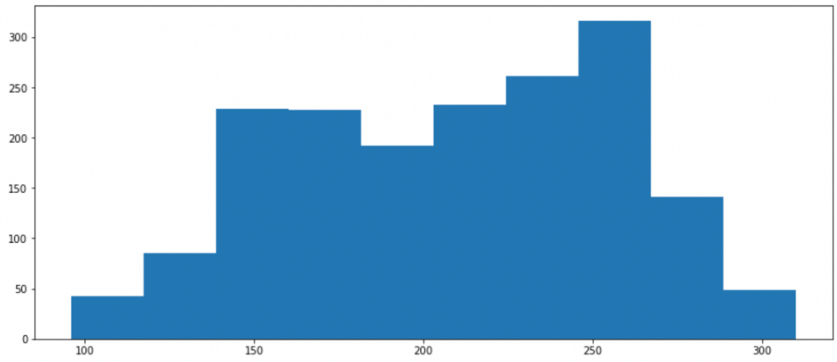
- Standard Deviation Value = 47.776

```
In [25]: #Find the standard deviation value of the product cost in wharehouse A from the road_mode data set
         stdev(wharehouse_block_A3['Cost_of_the_Product'])
Out[25]: 47.77634798718546
```

⇒ Show the distribution of the "Cost_of_the_Product" in histogram

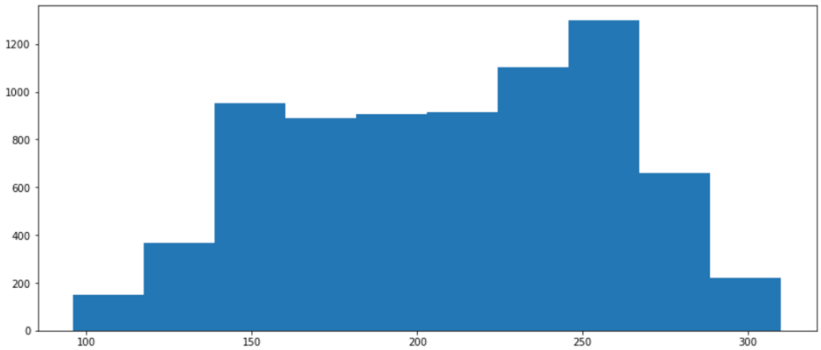
~ For Shipment = Flight

```
In [27]: plt.figure(figsize=(14,6))
         plt.hist(flight_mode['Cost_of_the_Product'])
         plt.show()
```



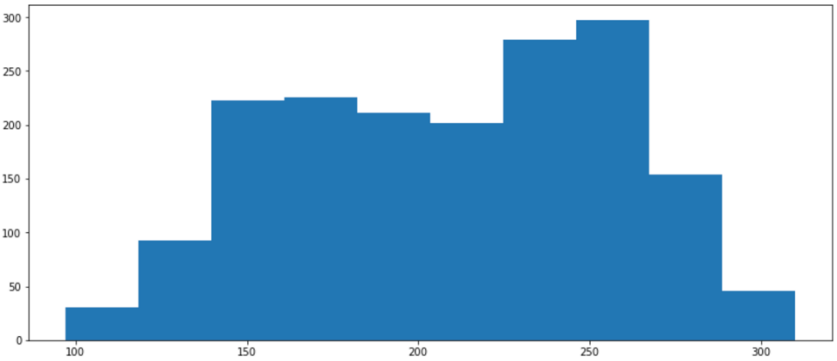
~ For Shipment = Ship

```
In [28]: plt.figure(figsize=(14,6))
         plt.hist(shiping_mode['Cost_of_the_Product'])
         plt.show()
```



~For Shipment = Road

```
In [29]: plt.figure(figsize=(14,6))
         plt.hist(road_mode['Cost_of_the_Product'])
         plt.show()
```



⇒ Hypothesis testing (P-Value, T-Test, Chi-Square Test, ANOVA Test, Pearson Correlation, spearman rank correlation)

~ For Shipment = Flight

- Statistic = 0.97, P-Value = 0.000000000000000007409235417650

  Not a normal distribution.

```
In [33]: from scipy.stats import shapiro
         data_to_test = flight_mode['Cost_of_the_Product']
         stat,p = shapiro(data_to_test)
         print('stat=%.2f, p=%.30f' % (stat,p))

         if p > 0.05:
             print('Normal distribution')
         else:
             print('Not a normal distribution')

         stat=0.97, p=0.000000000000000007409235417650
         Not a normal distribution
```

~ For Shipment = Ship

- Statistic = 0.97, P-Value = 0.00000000000000000

  Not a normal distribution.

```
In [34]: from scipy.stats import shapiro
         data_to_test = shiping_mode['Cost_of_the_Product']
         stat,p = shapiro(data_to_test)
         print('stat=%.2f, p=%.30f' % (stat,p))

         if p > 0.05:
             print('Normal distribution')
         else:
             print('Not a normal distribution')

         stat=0.97, p=0.000000000000000000000000000000
         Not a normal distribution
```

~ For Shipment = Road

- Statistic = 0.97, P-Value = 0.000000000000000001365729371655
  Not a normal distribution.

```
In [35]: from scipy.stats import shapiro
         data_to_test = road_mode['Cost_of_the_Product']
         stat,p = shapiro(data_to_test)
         print('stat=%.2f, p=%.30f' % (stat,p))

         if p > 0.05:
             print('Normal distribution')
         else:
             print('Not a normal distribution')

         stat=0.97, p=0.000000000000000001365729371655
         Not a normal distribution
```

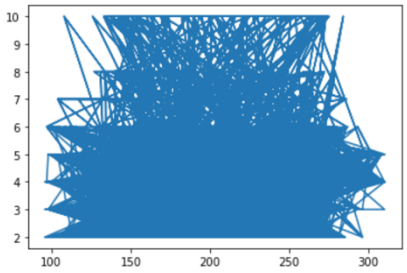
. Pearson Correlation

In [39]:
```
data.corr(method='pearson')
```

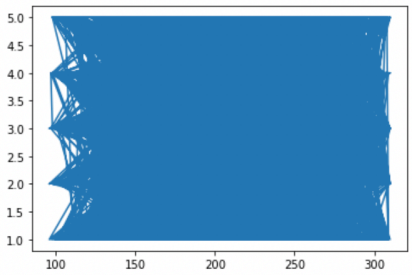
Out[39]:

	ID	Customer_care_calls	Customer_rating	Cost_of_the_Product	Prior_purchases	Discount_offered	Weight_in_gms	Reached.on.Time
ID	1.000000	0.188998	-0.005722	0.196791	0.145369	-0.598278	0.278312	-0.41
Customer_care_calls	0.188998	1.000000	0.012209	0.323182	0.180771	-0.130750	-0.276615	-0.06
Customer_rating	-0.005722	0.012209	1.000000	0.009270	0.013179	-0.003124	-0.001897	0.01
Cost_of_the_Product	0.196791	0.323182	0.009270	1.000000	0.123676	-0.138312	-0.132604	-0.07
Prior_purchases	0.145369	0.180771	0.013179	0.123676	1.000000	-0.082769	-0.168213	-0.05
Discount_offered	-0.598278	-0.130750	-0.003124	-0.138312	-0.082769	1.000000	-0.376067	0.39
Weight_in_gms	0.278312	-0.276615	-0.001897	-0.132604	-0.168213	-0.376067	1.000000	-0.26
Reached.on.Time_Y.N	-0.411822	-0.067126	0.013119	-0.073587	-0.055515	0.397108	-0.268793	1.00

In [36]:
```
# Testing correlation by using "Pearson and Spearman's rank correlation"
first_sample = data['Cost_of_the_Product']
second_sample = data['Prior_purchases']
# The result show that "score" and "study hour" are not correlate to one another
plt.plot(first_sample,second_sample)
plt.show()
```



In [40]:
```
first_sample = data['Cost_of_the_Product']
second_sample = data['Customer_rating']
#pearson show that "score" and "read_book" are correlated to one another, you can see the chart
plt.plot(first_sample,second_sample)
plt.show()
```



In [41]:
```
from scipy.stats import pearsonr
stat,p = pearsonr(first_sample,second_sample)
print('stat=%.3f,p=%5f' % (stat,p))
if p > 0.05:
    print('Independent samples')
else:
    print('Dependent samples')
```

```
stat=0.009,p=0.331020
Independent samples
```

⇒ T-Test

```
In [42]: #T-TEST
         print('All Shipment mean score',np.mean(data['Cost_of_the_Product']))
         print('flight_mode mean score',np.mean(flight_mode['Cost_of_the_Product']))
         print('flight_mode mean score',np.mean(shiping_mode['Cost_of_the_Product']))
         print('flight_mode mean score',np.mean(road_mode['Cost_of_the_Product']))

         All Shipment mean score 210.19683607600692
         flight_mode mean score 209.3066966797974
         flight_mode mean score 210.34307156258376
         flight_mode mean score 210.47556818181818
```

⇒ Chi-Square

```
In [49]: from scipy.stats import chi2_contingency
         stat,p,dof,expected = chi2_contingency(data_contingency)
         print('stat=%.3f,p=%.3f'%(stat,p))
         if p > 0.05:
             print('Independent category')
         else:
             print('Dependent category')

         stat=191.021,p=0.795
         Independent category

In [50]: #Chi-square test for numerical variable
         data_contingency=[[25,125],[1200,240]]
         stat,p,dof,expected = chi2_contingency(data_contingency)
         print('stat=%.3f,p=%.3f'%(stat,p))
         if p > 0.05:
             print('Independent value')
         else:
             print('Dependent value')

         stat=337.622,p=0.000
         Dependent value
```

- **Conclusion**

After doing some operation of statistic and go deeper into this dataset, we can see that mostly of the E-Commerce system used shipping type of shipment for delivery. Most customers are female who always order or buy product from the other countries.