```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from scipy.stats import norm
```

```
!gdown '1q5L_yeBEhoAWHJmYHzoGtLkgfUWyW7fM'
```

```
Downloading...
From: https://drive.google.com/uc?id=1q5L_yeBEhoAWHJmYHzoGtLkgfUWyW7fM
To: /content/walmart_data.txt
100% 23.0M/23.0M [00:00<00:00, 64.2MB/s]
```

```
df = pd.read_csv('/content/walmart_data.txt')
df.head()
```

	User_ID	Product_ID	Gender	Age	Occupation	City_Category	Stay_In_Current_City_
0	1000001	P00069042	F	0-17	10	A	
1	1000001	P00248942	F	0-17	10	A	
2	1000001	P00087842	F	0-17	10	A	

Insight :- There are total 550068 rows accross 10 columns.

```
df.shape
```

```
(550068, 10)
```

Insight :- The number of unique users given in the dataset.

```
df['User_ID'].nunique()
```

```
5891
```

Insight : There is no missing data in the dataset.

```
df.isna().sum()
```

```
User_ID                     0
Product_ID                  0
Gender                      0
Age                         0
Occupation                  0
```

```
City_Category                 0
Stay_In_Current_City_Years    0
Marital_Status                0
Product_Category              0
Purchase                      0
dtype: int64
```

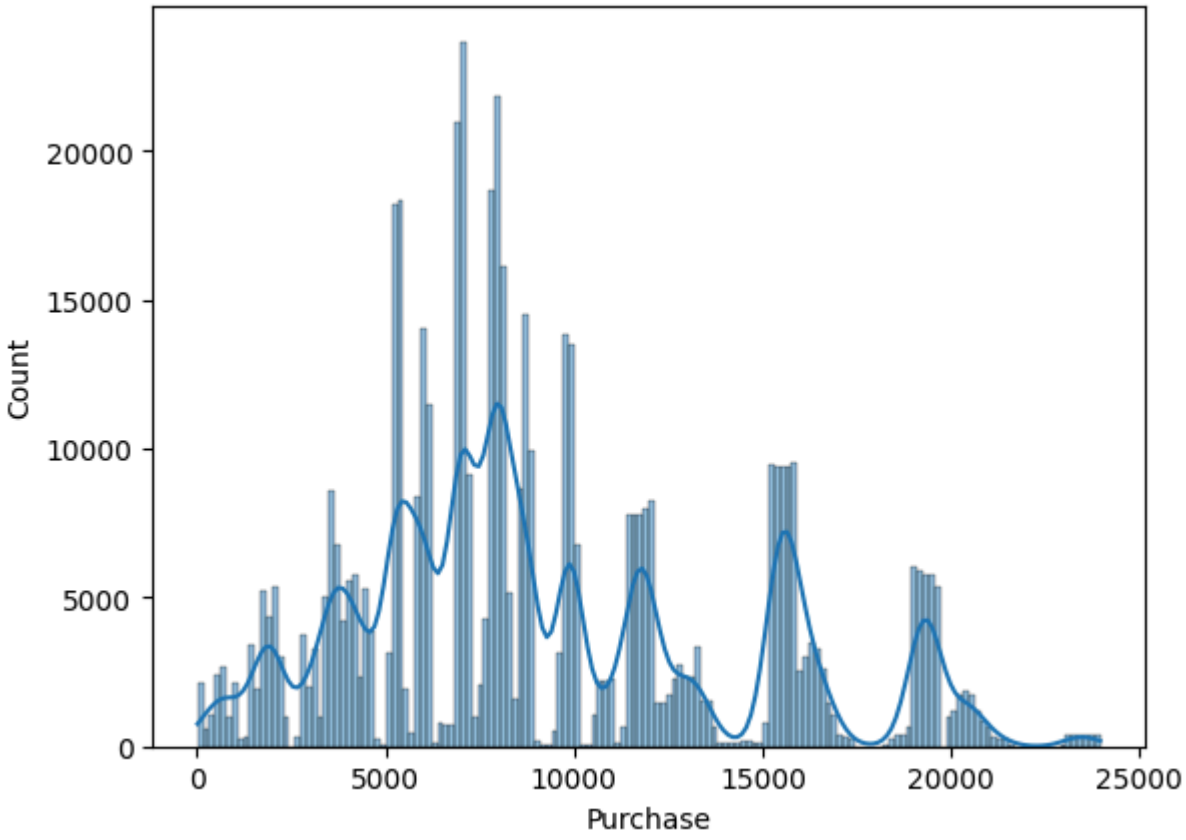
```
df.describe()
```

	User_ID	Occupation	Marital_Status	Product_Category	Purchase
count	5.500680e+05	550068.000000	550068.000000	550068.000000	550068.000000
mean	1.003029e+06	8.076707	0.409653	5.404270	9263.968713
std	1.727592e+03	6.522660	0.491770	3.936211	5023.065394
min	1.000001e+06	0.000000	0.000000	1.000000	12.000000
25%	1.001516e+06	2.000000	0.000000	1.000000	5823.000000
50%	1.003077e+06	7.000000	0.000000	5.000000	8047.000000
75%	1.004478e+06	14.000000	1.000000	8.000000	12054.000000
max	1.006040e+06	20.000000	1.000000	20.000000	23961.000000

Insight :- The data given is not Normal / Gausian.

```
sns.histplot(x='Purchase',data = df,kde =True)
```

```
<Axes: xlabel='Purchase', ylabel='Count'>
```



```
# Individual data for male and female
male = df[df['Gender'] == 'M']['Purchase']
female = df[df['Gender'] == 'F']['Purchase']
```

Test 1 :-

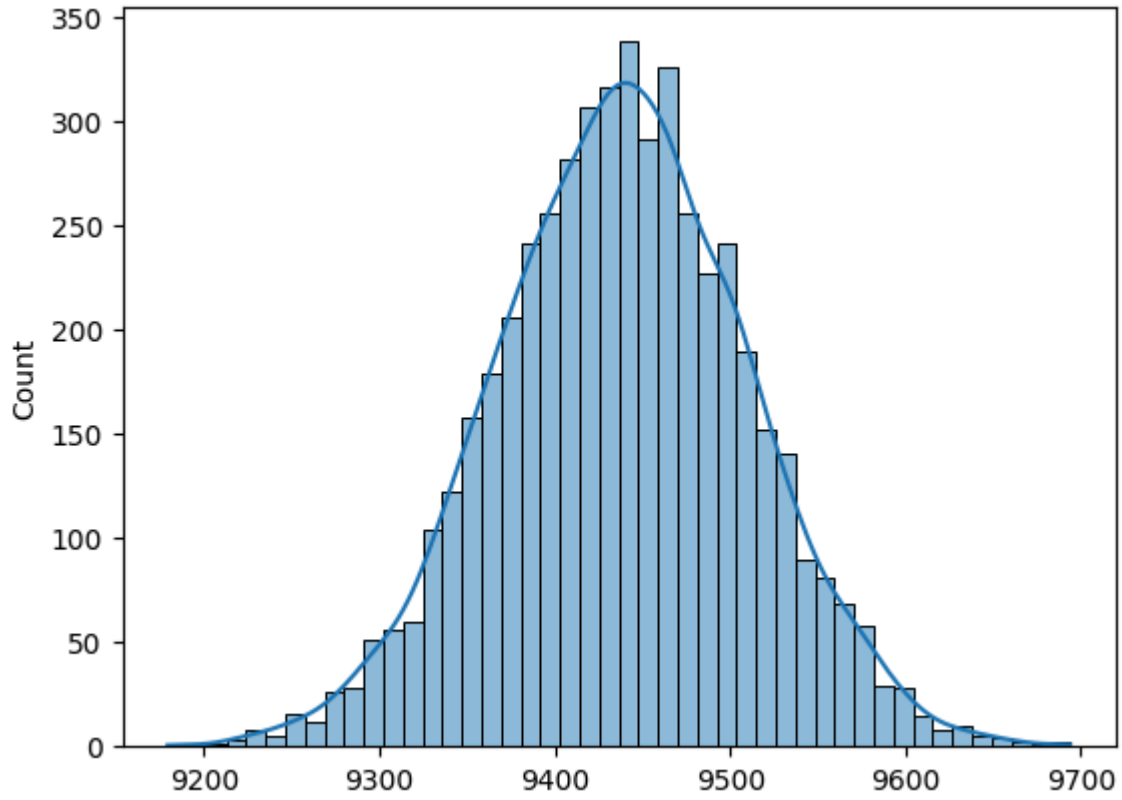
Assumptions :-

1. We are taking sample sizes as 5000.
2. We are taking confidence level as 90 %

```
male_samp_mean = [np.mean(male.sample(5000)) for i in range(5000)]
female_samp_mean = [np.mean(female.sample(5000)) for i in range(5000)]
```

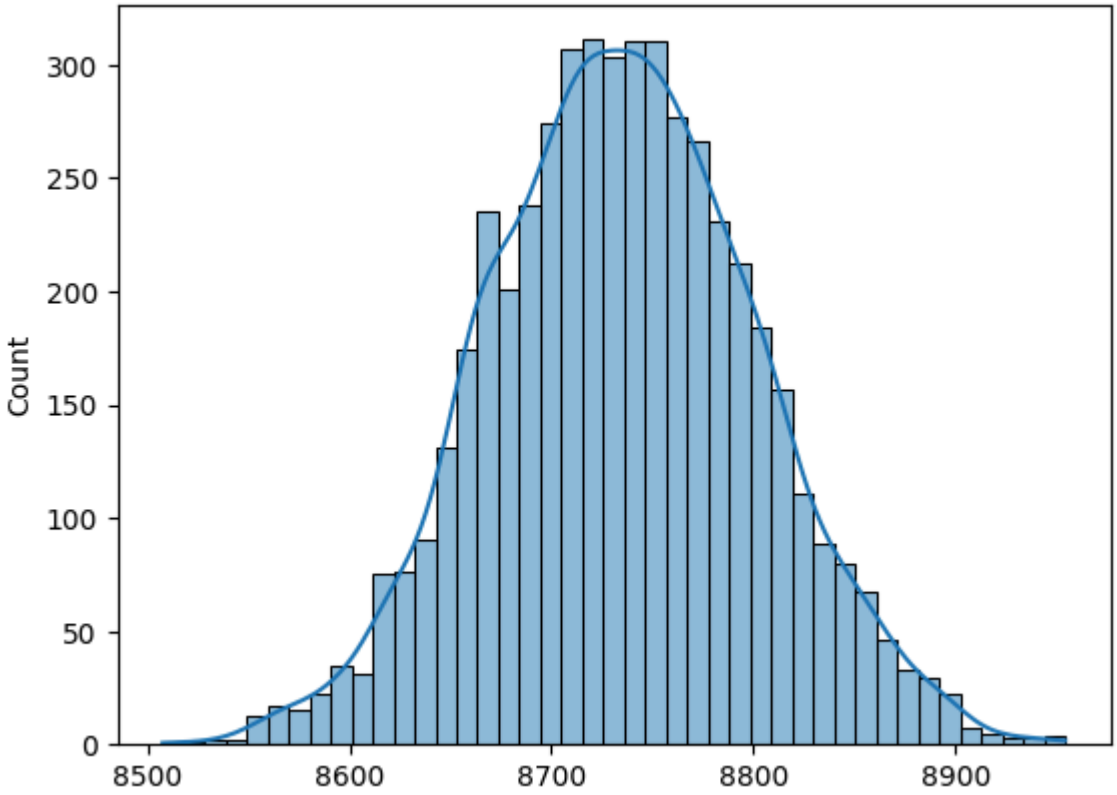
```
# Ploting the histplot of samp. means for male data.
sns.histplot(x= male_samp_mean,kde =True)
```

```
<Axes: ylabel='Count'>
```



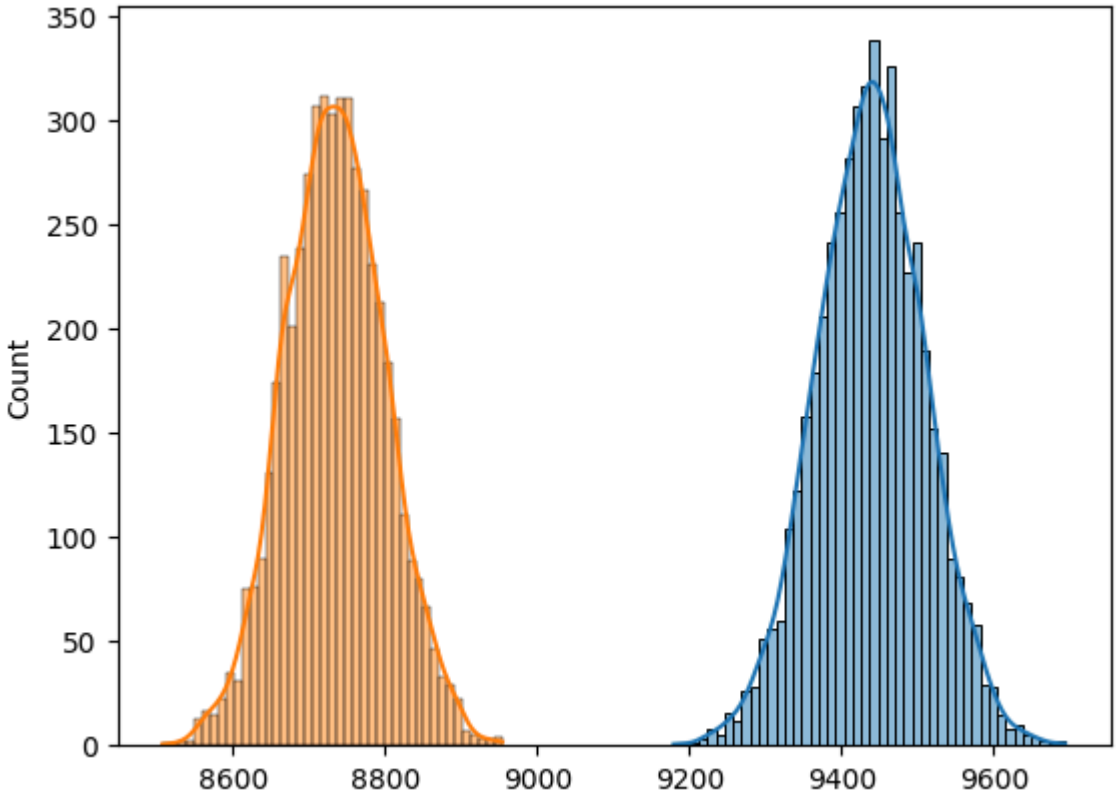

```
# Ploting the histplot of the samp. means for female data.
sns.histplot(x=female_samp_mean,kde= True)
```

```
<Axes: ylabel='Count'>
```




```
# Ploting the histplot of the samp. means for male and female data simultaneously.
sns.histplot(x= male_samp_mean,kde =True)
sns.histplot(x = female_samp_mean,kde =True)
```

```
<Axes: ylabel='Count'>
```



```
#Calcualting the mean of sample means of the male data.
mean_of_samp_mean_male = np.mean(male_samp_mean)

#Calcualting the mean of sample means of the female data.
mean_of_samp_mean_female = np.mean(female_samp_mean)
```

```
# calculating the standard error for building a confidence interval
se_m = np.std(male_samp_mean)/np.sqrt(len(male_samp_mean))
se_f = np.std(female_samp_mean)/np.sqrt(len(female_samp_mean))
```

Insight :- Everytime a male vists the store, he will buy in the range of 9437 to 9440 with 90% confidence level.

```
# critical z-score for 90 % confidence level is 1.64
z = 1.64
# confidence interval for male is :-
print(round(mean_of_samp_mean_male - (z *se_m )), round(mean_of_samp_mean_male + (z *se_m
```

```
9437 9440
```

Insight :- Everytime a female vists the store, he will buy in the range of 8733 to 8736 with 90% confidence level.

```
# confidence interval for female is :-
print(round(mean_of_samp_mean_female - (z *se_f )), round(mean_of_samp_mean_female + (z *
```

```
8733 8736
```

Test 2 :-

Assumptions :-

1. We are taking sample sizes as 5000.
2. We are taking confidence level as 95 %

Insight :- Everytime a female vists the store, he will buy in the range of 8733 to 8736 with 90% confidence level.

```
# critical z-score for 90 % confidence level is 1.64
z = 1.96
# confidence interval for male is :-
print(round(mean_of_samp_mean_male - (z *se_m )), round(mean_of_samp_mean_male + (z *se_m
```

```
9436 9440
```

Insight :- Everytime a female vists the store, he will buy in the range of 8733 to 8736 with 90% confidence level.

```
# confidence interval for female is :-
print(round(mean_of_samp_mean_female - (z *se_f )), round(mean_of_samp_mean_female + (z *s
```

```
8733 8737
```