1.

A F&B manager wants to determine whether there is any significant difference in the diameter of the cutlet between two units. A randomly selected sample of cutlets was collected from both units and measured? Analyze the data and draw inferences at 5% significance level. Please state the assumptions and tests that you carried out to check validity of the assumptions.

In [2]:

```
import pandas as pd
from scipy import stats
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

In [17]:

```
data_1 = pd.read_csv('Cutlets.csv')
data_1
```

Out[17]:

	Unit A	Unit B
0	6.8090	6.7703
1	6.4376	7.5093
2	6.9157	6.7300
3	7.3012	6.7878
4	7.4488	7.1522
5	7.3871	6.8110
6	6.8755	7.2212
7	7.0621	6.6606
8	6.6840	7.2402
9	6.8236	7.0503
10	7.3930	6.8810
11	7.5169	7.4059
12	6.9246	6.7652
13	6.9256	6.0380
14	6.5797	7.1581
15	6.8394	7.0240
16	6.5970	6.6672
17	7.2705	7.4314
18	7.2828	7.3070
19	7.3495	6.7478
20	6.9438	6.8889
21	7.1560	7.4220
22	6.5341	6.5217
23	7.2854	7.1688
24	6.9952	6.7594
25	6.8568	6.9399
26	7.2163	7.0133
27	6.6801	6.9182
28	6.9431	6.3346
29	7.0852	7.5459
30	6.7794	7.0992
31	7.2783	7.1180
32	7.1561	6.6965
33	7.3943	6.5780

	Unit A	Unit B
34	6.9405	7.3875

In [18]:

```
data_1.head()
```

Out[18]:

	Unit A	Unit B
0	6.8090	6.7703
1	6.4376	7.5093
2	6.9157	6.7300
3	7.3012	6.7878
4	7.4488	7.1522

In [19]:

```
data_1.describe()
```

Out[19]:

	Unit A	Unit B
count	35.000000	35.000000
mean	7.019091	6.964297
std	0.288408	0.343401
min	6.437600	6.038000
25%	6.831500	6.753600
50%	6.943800	6.939900
75%	7.280550	7.195000
max	7.516900	7.545900

In [28]:

```
data1.dtypes
```

Out[28]:

```
Unit A    float64
Unit B    float64
dtype: object
```

In [20]:

```
unit_a = data_1['Unit A']
unit_a
```

Out[20]:

```
0     6.8090
1     6.4376
2     6.9157
3     7.3012
4     7.4488
5     7.3871
6     6.8755
7     7.0621
8     6.6840
9     6.8236
10    7.3930
11    7.5169
12    6.9246
13    6.9256
14    6.5797
15    6.8394
16    6.5970
17    7.2705
18    7.2828
19    7.3495
20    6.9438
21    7.1560
22    6.5341
23    7.2854
24    6.9952
25    6.8568
26    7.2163
27    6.6801
28    6.9431
29    7.0852
30    6.7794
31    7.2783
32    7.1561
33    7.3943
34    6.9405
Name: Unit A, dtype: float64
```

In [21]:

```
unit_b = data_1['Unit B']
unit_b
```

Out[21]:

```
0     6.7703
1     7.5093
2     6.7300
3     6.7878
4     7.1522
5     6.8110
6     7.2212
7     6.6606
8     7.2402
9     7.0503
10    6.8810
11    7.4059
12    6.7652
13    6.0380
14    7.1581
15    7.0240
16    6.6672
17    7.4314
18    7.3070
19    6.7478
20    6.8889
21    7.4220
22    6.5217
23    7.1688
24    6.7594
25    6.9399
26    7.0133
27    6.9182
28    6.3346
29    7.5459
30    7.0992
31    7.1180
32    6.6965
33    6.5780
34    7.3875
Name: Unit B, dtype: float64
```

In [15]:

```
stats.ttest_ind(unit_a,unit_b)
```

Out[15]:

```
Ttest_indResult(statistic=0.7228688704678063, pvalue=0.4722394724599501)
```

Inference: Pvalue(0.47223) > 0.05 so we accept null hypothesis

2.

A hospital wants to determine whether there is any difference in the average Turn Around Time (TAT) of reports of the laboratories on their preferred list. They collected a random sample and recorded TAT for reports of 4 laboratories. TAT is defined as sample collected to report dispatch. Analyze the data and determine whether there is any difference in average TAT among the different laboratories at 5% significance level.

In [27]:

```
data_2 = pd.read_csv('LabTAT.csv')
data_2
```

Out[27]:

	Laboratory 1	Laboratory 2	Laboratory 3	Laboratory 4
0	185.35	165.53	176.70	166.13
1	170.49	185.91	198.45	160.79
2	192.77	194.92	201.23	185.18
3	177.33	183.00	199.61	176.42
4	193.41	169.57	204.63	152.60
...	...	...	...	...
115	178.49	170.66	193.80	172.68
116	176.08	183.98	215.25	177.64
117	202.48	174.54	203.99	170.27
118	182.40	197.18	194.52	150.87
119	182.09	215.17	221.49	162.21

120 rows × 4 columns

In [28]:

```
data_2.head()
```

Out[28]:

	Laboratory 1	Laboratory 2	Laboratory 3	Laboratory 4
0	185.35	165.53	176.70	166.13
1	170.49	185.91	198.45	160.79
2	192.77	194.92	201.23	185.18
3	177.33	183.00	199.61	176.42
4	193.41	169.57	204.63	152.60

In [29]:

```
data_2.describe()
```

Out[29]:

	Laboratory 1	Laboratory 2	Laboratory 3	Laboratory 4
count	120.000000	120.000000	120.000000	120.00000
mean	178.361583	178.902917	199.913250	163.68275
std	13.173594	14.957114	16.539033	15.08508
min	138.300000	140.550000	159.690000	124.06000
25%	170.335000	168.025000	188.232500	154.05000
50%	178.530000	178.870000	199.805000	164.42500
75%	186.535000	189.112500	211.332500	172.88250
max	216.390000	217.860000	238.700000	205.18000

In [30]:

```
data_2.dtypes
```

Out[30]:

```
Laboratory 1    float64
Laboratory 2    float64
Laboratory 3    float64
Laboratory 4    float64
dtype: object
```

In [31]:

```
lab_1 = data_2['Laboratory 1'].mean()
lab_2 = data_2['Laboratory 2'].mean()
lab_3 = data_2['Laboratory 3'].mean()
lab_4 = data_2['Laboratory 4'].mean()
print('Laboratory 1=',lab_1)
print('Laboratory 2=',lab_2)
print('Laboratory 3=',lab_3)
print('Laboratory 4=',lab_4)
```

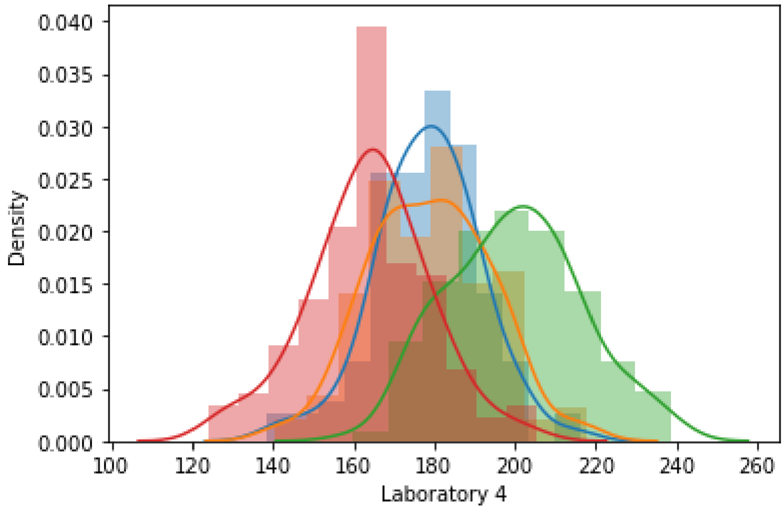
```
Laboratory 1= 178.36158333333339
Laboratory 2= 178.9029166666668
Laboratory 3= 199.91325000000003
Laboratory 4= 163.6827499999999
```

In [32]:

```
sns.distplot(data_2['Laboratory 1'])
sns.distplot(data_2['Laboratory 2'])
sns.distplot(data_2['Laboratory 3'])
sns.distplot(data_2['Laboratory 4'])
```

Out[32]:

```
<AxesSubplot:xlabel='Laboratory 4', ylabel='Density'>
```



In [33]:

```
lab1=pd.DataFrame(data_2["Laboratory 1"])
lab2=pd.DataFrame(data_2["Laboratory 2"])
lab3=pd.DataFrame(data_2["Laboratory 3"])
lab4=pd.DataFrame(data_2['Laboratory 4'])
```

In [34]:

```
stats.f_oneway(lab1,lab2,lab3,lab4)
```

Out[34]:

```
F_onewayResult(statistic=array([118.70421654]), pvalue=array([2.11567089e-5
7]))
```

Inference : Pvalue (2.1156) > 0.05 So we accept null hypothesis

3.

Sales of products in four different regions is tabulated for males and females. Find if male-female buyer rations are similar across regions.

In [26]:

```
data_3 = pd.read_csv('BuyerRatio.csv')
data_3
```

Out[26]:

	Observed Values	East	West	North	South
0	Males	50	142	131	70
1	Females	435	1523	1356	750

In [36]:

```
data_3.dtypes
```

Out[36]:

```
Observed Values    object
East                int64
West                int64
North               int64
South               int64
dtype: object
```

In [83]:

```
alpha = 0.05
Males = [50,142,131,70]
Females = [435,1523,1356,750]
Customers = [Males,Females]
print('Customers:',Males,Females)
```

```
Customers: [50, 142, 131, 70] [435, 1523, 1356, 750]
```

In [98]:

```
chi2_test_1 = stats.chi2_contingency(Customers)
chi2_test_1
```

Out[98]:

```
(1.595945538661058,
 0.6603094907091882,
 3,
 array([[  42.76531299,  146.81287862,  131.11756787,   72.30424052],
        [ 442.23468701, 1518.18712138, 1355.88243213,  747.69575948]]))
```

In [103]:

```
print('chi2_value = %f,Pvalue = %f' % (chi2_test_1[0],chi2_test_1[1]))
print('Interpret by Pvalue')
if chi2_test_1[1] < 0.05:
    print('We reject null hypothesis')
else:
    print('we accept null hypothesis')
```

```
chi2_value = 1.595946,Pvalue = 0.660309
Interpret by Pvalue
we accept null hypothesis
```

Inference : Pvalue (0.6603) < 0.05 So we accept null hypothesis

4.

TeleCall uses 4 centers around the globe to process customer order forms. They audit a certain % of the customer order forms. Any error in order form renders it defective and has to be reworked before processing. The manager wants to check whether the defective % varies by centre. Please analyze the data at 5% significance level and help the manager draw appropriate inferences

In [57]:

```
data_4 = pd.read_csv('Costomer+OrderForm.csv')
data_4
```

Out[57]:

	Phillippines	Indonesia	Malta	India
0	Error Free	Error Free	Defective	Error Free
1	Error Free	Error Free	Error Free	Defective
2	Error Free	Defective	Defective	Error Free
3	Error Free	Error Free	Error Free	Error Free
4	Error Free	Error Free	Defective	Error Free
...	...	...	...	...
295	Error Free	Error Free	Error Free	Error Free
296	Error Free	Error Free	Error Free	Error Free
297	Error Free	Error Free	Defective	Error Free
298	Error Free	Error Free	Error Free	Error Free
299	Error Free	Defective	Defective	Error Free

300 rows × 4 columns

In [58]:

```
data_4.head()
```

Out[58]:

	Phillippines	Indonesia	Malta	India
0	Error Free	Error Free	Defective	Error Free
1	Error Free	Error Free	Error Free	Defective
2	Error Free	Defective	Defective	Error Free
3	Error Free	Error Free	Error Free	Error Free
4	Error Free	Error Free	Defective	Error Free

In [59]:

```
data_4.describe()
```

Out[59]:

	Phillippines	Indonesia	Malta	India
count	300	300	300	300
unique	2	2	2	2
top	Error Free	Error Free	Error Free	Error Free
freq	271	267	269	280

In [60]:

```
data_4.dtypes
```

Out[60]:

```
Phillippines    object
Indonesia       object
Malta           object
India           object
dtype: object
```

In [70]:

```
Phillippines_count = data_4['Phillippines'].value_counts()
print(Phillippines_count)
Indonesia_count = data_4['Indonesia'].value_counts()
print(Indonesia_count)
Malta_count = data_4['Malta'].value_counts()
print(Malta_count)
India_count = data_4['India'].value_counts()
print(India_count)
```

```
Error Free    271
Defective      29
Name: Phillippines, dtype: int64
Error Free    267
Defective      33
Name: Indonesia, dtype: int64
Error Free    269
Defective      31
Name: Malta, dtype: int64
Error Free    280
Defective      20
Name: India, dtype: int64
```

In [101]:

```
chi2_test_2 = stats.chi2_contingency([[271,267,269,280],[29,33,31,20]])
chi2_test_2
```

Out[101]:

```
(3.858960685820355,
 0.2771020991233135,
 3,
 array([[271.75, 271.75, 271.75, 271.75],
        [ 28.25,  28.25,  28.25,  28.25]]))
```

In [102]:

```
print('chi2 = %f,Pvalue = %f' % (chi2_test_2[0],chi2_test_2[1]))
print('Interpret by Pvalue')
if chi2_test_2[1] > 0.05:
    print('We accept null hypothesis')
else:
    print('we reject null hypothesis')
```

```
chi2 = 3.858961,Pvalue = 0.277102
Interpret by Pvalue
We accept null hypothesis
```

Inference : Pvalue (0.2771) > 0.05 So we accept null hypothesis

THE END

In [ ]: