Hypothesis Testing

Why do we have to test hypothesis? In normal studies we study on the population: since we cannot get all the data information about the population, usually we will sample the data from the population and after we have sampled and performed the analytics for the sample and we will infer the population.

However, these inferences need to be tested. Statistical Hypothesis is an assumption about a population parameter that we do not know, which needs to be tested.

Statistical hypothesis testing is the process of steps by which we can reject or retain a given hypothesis.

In [1]:

```
import numpy as np
import pandas as pd
from scipy import stats
import seaborn as sns
import scikit_posthocs as sp
pd.options.display.float_format = '{:,.4f}'.format   # to display all data
```

In [ ]:

```
STEP 1: State the hypotheses. (Population)
STEP 2: Set the level of Significance: α (Criterion)
STEP 3: Compute test Statistics (Sample)
STEP 4: Make a decision based on p value
```

Creat function

Check whether or not a sample comes from a normal distribution

The Shapiro-Wilk test is a test of normality. It is used to determine whether or not a sample comes from a normal distribution.

In [ ]:

```
H0: The data is normally distributed.
H1: The data is not normally distributed.
```

In [2]:

```
def check_normality(data):
    test_stat_normality, p_value_normality=stats.shapiro(data)
    print("p value:%.4f" % p_value_normality)
    if p_value_normality <0.05:
        print("Reject null hypothesis (H0) => The data is not normally distributed")
    else:
        print("Fail to reject null hypothesis (H0) => The data is normally distributed"
```

Levene's Test is used to determine whether two or more groups have equal variances.

In [ ]:

```
H0: The variances of the samples are same.
H1: The variances of the samples are different.
```

In [3]:

```
def check_variance_homogeneity(group1, group2):
    test_stat_var, p_value_var= stats.levene(group1, group2)
    print("p value:%.4f" % p_value_var)
    if p_value_var <0.05:
        print("Reject null hypothesis (H0) => The variances of the samples are differen
    else:
        print("Fail to reject null hypothesis (H0) => The variances of the samples are 
```

CASE

An e-commerce company makes advertising on 3 platforms YouTube, Instagram, Facebook. However, does the director care about the average number of users obtained through different channels? So, the number of users interacting through the channels recorded over 15 days is as follows:

Based on the data collected, determine whether there is a difference in customers averages across advertising channels using hypothesis testing? with significance alpha=5%

To determine if there is a difference in average customers across advertising channels using a hypothesis test, follow these steps:

- Validate the normality assumption: Conduct a Shapiro-Wilk test to evaluate whether the data for each ad channel follows normal delivery. If the data is not normally distributed, consider applying a transformation or using a non-parametric test instead.
- True equality of variances: Use Levene's Test to determine if the variance of the average customer across the advertising channels is equal. If the assumption of equal variances is violated, consider using alternative tests such as Welch's ANOVA or the Kruskal-Wallis test.
- Perform ANOVA: Apply the F-test within the ANOVA framework to assess whether the average customer medium across advertising channels is significantly different. If the p-value associated with the F-test is lower than the predefined level of significance, conclude that there is a significant difference in the average of the customers across the advertising channels.
- Analyze which group is really different with posthoc_mannwhitney

In [4]:

```
youtube=np.array([1913, 1879, 1939, 2146, 2040, 2127, 2122, 2156, 2036, 1974, 1956,
                  2146, 2151, 1943, 2125])

instagram = np.array([2305., 2355., 2203., 2231., 2185., 2420., 2386., 2410., 2340.,
                      2349., 2241., 2396., 2244., 2267., 2281.])

facebook = np.array([2133., 2522., 2124., 2551., 2293., 2367., 2460., 2311., 2178.,
                     2113., 2048., 2443., 2265., 2095., 2528.])
```

Check whether or not a sample comes from a normal distribution

In [ ]:

```
H0: The data is normally distributed.
H1: The data is not normally distributed.
```

In [5]:

```
check_normality(youtube)
check_normality(instagram)
check_normality(facebook)
```

```
p value:0.0285
Reject null hypothesis (H0) => The data is not normally distributed
p value:0.4156
Fail to reject null hypothesis (H0) => The data is normally distributed
p value:0.1716
Fail to reject null hypothesis (H0) => The data is normally distributed
```

Check whether or not the variances of two or more groups are the same

```
H0: The variances of the samples are the same.
H1: The variances of the samples are different.
```

In [6]:

```
stat, pvalue_levene= stats.levene(youtube, instagram, facebook)

print("p value:%.4f" % pvalue_levene)
if pvalue_levene <0.05:
    print("Reject null hypothesis >> The variances of the samples are different.")
else:
    print("Fail to reject null hypothesis >> The variances of the samples are same.")
```

```
p value:0.0012
Reject null hypothesis >> The variances of the samples are different.
```

Check whether or not the means of three channels are equal?

```
H0: Mean.youtube = Mean.facekook = Mean.instagram or The mean of the samples are the
same.
H1: At least one of them is different.
```

In [7]:

```
F, p_value = stats.kruskal(youtube, instagram, facebook) #more than 2 variables, use kr
print("p value:%.6f" % p_value)
if p_value <0.05:
    print("Reject null hypothesis, at least one of means is different")
else:
    print("Fail to reject null hypothesis")
```

```
p value:0.000015
Reject null hypothesis, at least one of means is different
```

At this significance level, at least one of the average customer acquisition numbers is different. Note: Since, the data is not normal, nonparametric version of posthoc test is used.

Analyze which customers average is really different from others using posthoc_mannwhitney test

In [8]:

```
posthoc_df = sp.posthoc_mannwhitney([youtube,instagram, facebook], p_adjust = 'bonferro

group_names= ["youtube", "instagram","facebook"]
posthoc_df.columns= group_names
posthoc_df.index= group_names
posthoc_df.style.applymap(lambda x: "background-color:violet" if x<0.05 else "backgroun
```

Out[8]:

	youtube	instagram	facebook
youtube	1.000000	0.000010	0.002337
instagram	0.000010	1.000000	1.000000
facebook	0.002337	1.000000	1.000000

The average number of customers coming from YouTube is different than the other (actually smaller than the others).