Two Sample Z-test

Two sample Z-test for example question

In [2]:

```
import numpy as np
import pandas as pd
import math
import scipy.stats as stats
```

In [3]:

```
def check_for(sym, arr):
    i = [i for i in range(len(arr)) if arr[i] == sym]
    if i != []:
        return i[0]
    else:
        return None
```

In [20]:

```
std1 = 100
std2 = 90
x1 = 641
x2 = 613.3
N1 = 20
N2 = 20
alpha = 0.05

two_sample_ztest(x1, x2, std1, std2, N1, N2, alpha)
```

```
Enter Null hypothesis: u1 = u2
u1 - u2 = 0
Null (Ho):-                u1 = u2
Alternate (Ha):-           u1 != u2
Test type :-              Two Tailed Test
Z-Score :-                0.9207791560106305
P-value :-                0.17858286979188148
Critical Value :-         -1.6448536269514729

Null Hypothesis Ho is accepted!
```

Two sample Z-test for Selected Dataset

In [7]:

```
data = pd.read_csv('../Country Wise Gender.csv')[['2014 Male', '2020 Male']]
data.head()
```

Out[7]:

	2014 Male	2020 Male
0	53.7	53.23
1	55.9	55.13
2	46.2	38.91
3	53.7	49.86
4	50.8	47.86

In [8]:

```
sample = data.sample(n = 30)
sample.head()
```

Out[8]:

	2014 Male	2020 Male
36	62.5	56.62
30	52.6	62.01
0	53.7	53.23
14	61.4	55.29
32	53.5	57.79

In [21]:

```
u1 = data['2014 Male'].mean()
u2 = data['2020 Male'].mean()
x1 = sample['2014 Male'].mean()
x2 = sample['2020 Male'].mean()
std1 = sample['2014 Male'].std()
std2 = sample['2020 Male'].std()
N1 = data['2014 Male'].count()
N2 = data['2014 Male'].count()
alpha = 0.01

two_sample_ztest(x1, x2, std1, std2, N1, N2, alpha, u1, u2)
```

```
Enter Null hypothesis: u1 = u2
u1 - u2 = 0
Null (Ho):-              u1 = u2
Alternate (Ha):-         u1 != u2
Test type :-            Two Tailed Test
Z-Score :-              1.295940545486136
P-value :-              0.09749798451318903
Critical Value :-       -2.3263478740408408

Null Hypothesis Ho is accepted!
```

In [4]:

```
def two_sample_ztest(x1, x2, std1, std2, N1, N2, alpha, u1=None, u2=None):
#   asking for the null hypothesis
    null_hypo = input("Enter Null hypothesis: ")
    null_hypo = null_hypo.split(' ')
    alternate_hypo = [i for i in null_hypo]
    test_type = ''
#   based on null hypothesis deducing the alternate hypothesis and the type of test
    if check_for('>=', null_hypo):
        alternate_hypo[check_for('>=', null_hypo)] = '<'
        test_type = 'One Left Tailed Test'
    elif check_for('<=', null_hypo):
        alternate_hypo[check_for('<=', null_hypo)] = '>'
        test_type = 'One Right Tailed Test'
    elif check_for('>', null_hypo):
        alternate_hypo[check_for('>', null_hypo)] = '<='
        test_type = 'One Left Tailed Test'
    elif check_for('<', null_hypo):
        alternate_hypo[check_for('<', null_hypo)] = '>='
        test_type = 'One Right Tailed Test'
    elif check_for('=', null_hypo):
        alternate_hypo[check_for('=', null_hypo)] = '!='
        test_type = 'Two Tailed Test'
    else:
        print("Failed!")
        return

#   if u1 and u2 is not given then ask the user
    d = {'u1 - u2': 0}
    if " ".join(null_hypo) == 'u1 = u2': d['u1 - u2'] = 0
    elif check_for('-', null_hypo): d['u1 - u2'] = float(null_hypo[-1])
    elif not u1 or not u2: d['u1 - u2'] = float(input("Enter u1 - u2: "))
    else: d['u1 - u2'] = u1 - u2
    print('u1 - u2 = ' + str(d['u1 - u2']))

#   Calculate Z-Score, P-value, Critical Value
    z_score = ((x1 - x2) - d['u1 - u2']) / math.sqrt((std1*std1 / N1) + (std2*std2 / N2))
    p_value = stats.norm.sf(z_score)
    if test_type == 'Two Tailed Test': critical_value = stats.norm.ppf(alpha)
    else: critical_value = stats.norm.ppf(alpha)

#   Print the results
    print('Null (Ho):-\t\t', " ".join(null_hypo))
    print('Alternate (Ha):-\t', " ".join(alternate_hypo))
    print('Test type :-\t\t'+test_type)
    print('Z-Score :-\t\t', end='')
    print(z_score)
    print('P-value :-\t\t', end='')
    print(p_value)
    print('Critical Value :-\t', end='')
    print(critical_value)

    if abs(z_score) > abs(critical_value): print("\nNull Hypothesis Ho is rejected!\nAlternate Hypothesis Ho is accepted!"
    else: print("\nNull Hypothesis Ho is accepted!")
```