

```

import math
import numpy as np
from statsmodels.stats.weightstats import ztest
from scipy.stats import norm

sample_marks =
[650,730,510,670,480,800,690,530,590,620,710,670,640,780,650,490,800,6
00,510,700]

# Method 1 : Using Z-score

sample_mean = np.mean(sample_marks)
sample_size = np.count_nonzero(sample_marks)
population_mean = 600
population_std = 100
alpha = 0.05
z_score = (sample_mean-
population_mean)/(population_std/math.sqrt(sample_size))
critical_value = 1.645 # from z table
if(z_score<critical_value):
    print('Null hypothesis is accepted!')
else:
    print('Null hypothesis is rejected. \nAlternate hypothesis is
accepted!')

# Method 2: Using built in function of ztest

ztest_score, pval =
ztest(sample_marks,value=population_mean,alternative='larger')
print('Z-test Score:',ztest_score,'\nP-value:',pval)
if(pval>alpha):
    print('Null hypothesis is accepted!')
else:
    print('Null hypothesis is rejected. \nAlternate hypothesis is
accepted!')

# Method 3: Creating a function

def ztest(x,mu,sigma,n):
    deno = sigma/math.sqrt(n)
    z = (x-mu)/deno
    p = 2*(1-norm.cdf(abs(z)))
    return z,p

s_mean = np.mean(sample_marks)
p_mean = 600
p_std = 100
s_size = np.count_nonzero(sample_marks)

```

```
ztest(s_mean,p_mean,p_std,s_size)
```

```
ztest(641,600,100,20)
```

```
Null hypothesis is rejected.
```

```
Alternate hypothesis is accepted!
```

```
Z-test Score: 1.831744911595958
```

```
P-value: 0.03349471703839336
```

```
Null hypothesis is rejected.
```

```
Alternate hypothesis is accepted!
```

```
(1.8335757415498277, 0.06671699590108493)
```