

hypothesis-test

November 1, 2024

```
[1]: from scipy.stats import norm # Import the normal distribution functions from
      ↳ SciPy
from math import sqrt # Import the square root function from the math module
def two_sided_hypo(sample_mean, pop_mean, std_dev, sample_size, alpha):
    # Calculate the critical z-value for a two-tailed test
    actual_z = abs(norm.ppf(alpha / 2)) # Get the z-value corresponding to
    ↳ alpha/2
    # Calculate the z-value for the hypothesis test
    hypo_z = (sample_mean - pop_mean) / (std_dev / sqrt(sample_size)) # Z-test
    ↳ statistic
    print('actual z value :', actual_z) # Print the actual critical z-value
    print('hypothesis z value :', hypo_z, '\n') # Print the calculated z-value
    ↳ for the hypothesis
    # Check if the calculated z-value falls into the rejection region
    if hypo_z >= actual_z or hypo_z <= -actual_z:
        return True # Reject the null hypothesis
    else:
        return False # Fail to reject the null hypothesis
# Define parameters for the hypothesis test
alpha = 0.05 # Significance level
sample_mean = 585 # Mean of the sample
pop_mean = 558 # Population mean under the null hypothesis
sample_size = 100 # Sample size
std_dev = 139 # Standard deviation of the population
# Print hypotheses
print('H0 :   =', pop_mean) # Null hypothesis: population mean is equal to
    ↳ pop_mean
print('H1 :   !=', pop_mean) # Alternative hypothesis: population mean is not
    ↳ equal to pop_mean
print('alpha value is :', alpha, '\n') # Print the significance level
# Perform the two-sided hypothesis test
reject = two_sided_hypo(sample_mean, pop_mean, std_dev, sample_size, alpha)
if reject:
    print('Reject NULL hypothesis') # If the test result indicates rejection
else:
    print('Failed to reject NULL hypothesis') # If the test result does not
    ↳ indicate rejection
```

H0 : = 558
H1 : != 558
alpha value is : 0.05

actual z value : 1.9599639845400545
hypothesis z value : 1.9424460431654675

Failed to reject NULL hypothesis

```
[2]: # One-sided hypothesis test (for greater than in the null hypothesis)
def one_sided_hypo(sample_mean, pop_mean, std_dev, sample_size, alpha):
    # Calculate the critical z-value for a one-tailed test
    actual_z = abs(norm.ppf(alpha)) # Get the z-value corresponding to alpha
    # Calculate the z-value for the hypothesis test
    hypo_z = (sample_mean - pop_mean) / (std_dev / sqrt(sample_size)) # Z-test
    ↪ statistic
    print('actual z value :', actual_z) # Print the actual critical z-value
    print('hypothesis z value :', hypo_z, '\n') # Print the calculated z-value
    ↪ for the hypothesis
    # Check if the calculated z-value falls into the rejection region
    if hypo_z >= actual_z:
        return True # Reject the null hypothesis
    else:
        return False # Fail to reject the null hypothesis
# Define parameters for the one-sided hypothesis test
alpha = 0.05 # Significance level
sample_mean = 108 # Mean of the sample
pop_mean = 100 # Population mean under the null hypothesis
sample_size = 36 # Sample size
std_dev = 15 # Standard deviation of the population
# Print hypotheses
print('H0 : <=', pop_mean) # Null hypothesis: population mean is less than or
    ↪ equal to pop_mean
print('H1 : >', pop_mean) # Alternative hypothesis: population mean is
    ↪ greater than pop_mean
print('alpha value is :', alpha, '\n') # Print the significance level
# Perform the one-sided hypothesis test
reject = one_sided_hypo(sample_mean, pop_mean, std_dev, sample_size, alpha)
if reject:
    print('Reject NULL hypothesis') # If the test result indicates rejection
else:
    print('Failed to reject NULL hypothesis') # If the test result does not
    ↪ indicate rejection
```

H0 : <= 100
H1 : > 100
alpha value is : 0.05

actual z value : 1.6448536269514729
hypothesis z value : 3.2

Reject NULL hypothesis