

SPONSORED BY

two samples z test in Python [closed]

Asked 5 years, 3 months ago Active 5 years, 2 months ago Viewed 24k times



Closed. This question is off-topic. It is not currently accepting answers.





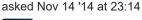
Want to improve this question? <u>Update the question</u> so it's <u>on-topic</u> for Cross Validated. Closed 5 years ago.



Does Scipy have a z test to compare the mean of two samples? I searched this page but couldn't find one.









I haven't used Python for Stats in a little while but, as I recall, it's best to use Statsmodels instead of Scipy. Stats. – Steve S Nov 15 '14 at 11:13

This question appears to be off-topic because it is about available functions in Python. – gung - Reinstate Monica ♦ Nov 15 '14 at 19:20

2 Answers



Statsmodels has a ztest function that allows you to compare two means, assuming they are independent and have the same standard deviation. See the documentation here



If you need to compare means from distributions with different standard deviation, you should use <code>CompareMeans.ztest_ind</code> . See documentation here.



There might be other functions I'm missing so search through the documentation!



answered Nov 15 '14 at 19:14





No, but this wouldn't be that hard to write a function for:





1

def twoSampZ(X1, X2, mudiff, sd1, sd2, n1, n2):
 from numpy import sqrt, abs, round
 from scipy.stats import norm
 pooledSE = sqrt(sd1**2/n1 + sd2**2/n2)
 z = ((X1 - X2) - mudiff)/pooledSE
 pval = 2*(1 - norm.cdf(abs(z)))
 return round(z, 3), round(pval, 4)

where X1 = $\bar{X1}$, X2 = $\bar{X2}$, mudiff = null = $\mu_1 - \mu_2$, sd1 = σ_1 , sd2 = σ_2 , n1 = n_1 and n2 = n_2 . So, going off of this example:

```
z, p = twoSampZ(28, 33, 0, 14.1, 9.5, 75, 50)
print z, p
```

answered Nov 15 '14 at 13:56



Just a little tip: 1 - norm.cdf can be replaced in numpy using norm.sf, which gives better precision. - AkiRoss Oct 15 '15 at 9:07