Hypothesis testing with Scipy Cheat Sheet

by sasha2411 (sasha2411) via cheatography.com/65860/cs/16458/

1 Sample T-Testing

For **numerical** data.

Compares a sample mean to a hypothetical population mean.

from scipy.stats import ttest_1samp

ttest_1samp requires two inputs, a distribution of values and an expected mean.

tstat, pval = ttest_1samp(example_distribution, expected_mean)

2 Sample T-Test

For **numerical** data.

Compares two sets of data, which are both approximately normally distributed.

The *null hypothesis*, in this case, is that the two distributions have the same mean.

from scipy.stats import ttest_ind

It takes the two distributions as inputs and returns the t-statistic and a p-value.

t, pval = ttest_ind(dataset1, dataset2)

ANOVA

For **numerical** data.

Compares more than two numerical datasets.

ANOVA (Analysis of Variance) tests the null hypothesis that all of the datasets have the same mean.

from scipy.stats import f_oneway

It takes in each dataset as a different input and returns the t-statistic and the p-value.

t, pval = f_oneway(a, b, c)

Tukey's Range Test

For **numerical** data.

We can perform a **Tukey's Range Test** to determine the difference between datasets.

from statsmodels.stats.multicomp import pairwise_tukeyhsd

We have to provide the function with one *list of all of the data* and a *list of labels* that tell the function which elements of the list are from which set. We also provide the *significance level* we want, which is usually 0.05.

values = np.concatenate([a, b, c])

labels = ['a'] * len(a) + ['b'] * len(b) + ['c'] * len(c)

tukey_results = pairwise_tukeyhsd(values, labels, 0.05)

Binomial Test

For **categorical** data.

To analyze a dataset with two different possibilities for entries.

The **null hypothesis**, in this case, would be that there is no difference between the observed behavior and the expected behavior.

from scipy.stats import binom_test

binom_test requires three inputs, the number of observed successes, the number of total trials, and an expected probability of success.

pval = binom_test(525, n=1000, p=0.5)

Chi Square Test

For **categorical** data.

To compare two or more categorical datasets.

from scipy.stats import chi2_contingency

The input to chi2_contingency is a **contingency table** where:

- **The columns** represent different outcomes, like "Survey Response A" vs. "Survey Response B" or "Clicked a Link" vs. "Didn't Click"
- **The rows** are each a different condition, such as men vs. women or Interface A vs. Interface B

X = [[30, 10],

[35, 5],

[28, 12],

[20, 20]]

_, pval, _, _ = chi2_contingency(X)

Sponsored by **CrosswordCheats.com**

Learn to solve cryptic crosswords!

http://crosswordcheats.com