

# tukey\_hsd: An Accurate Implementation of the Tukey Honestly Significant Difference Test in Python

Dominic Chmiel<sup>\*1</sup>, Samuel Wallan<sup>†1</sup>, and Matt Haberland<sup>1¶</sup>

<sup>1</sup> California Polytechnic State University, San Luis Obispo, USA ¶ Corresponding author

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

## Software

- [Review](#) ↗
- [Repository](#) ↗
- [Archive](#) ↗

Editor: [Mikkel Meyer Andersen](#) ↗

Submitted: 13 February 2022

Published: unpublished

## License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

## Summary

In a world awash with data and computers, it is tempting to automate the process of scientific discovery by performing comparisons between many pairs of variables in hope of finding correlations. When frequentist hypothesis tests are performed at a fixed confidence level, increasing the number of tests increases the probability of observing a “statistically significant” result, even when the null hypothesis is actually true. Carefully designed tests, such as Tukey’s honestly significant difference (HSD) test ([Tukey, 1949](#)), protect against this practice of “p-hacking”, producing p-values and confidence intervals that account for the number of comparisons performed. Several such tests rely on the studentized range distribution ([Lund & Lund, 1983](#)), which models the range (i.e. difference between maximum and minimum values) of the means of samples from a normally distributed population. Although there are already implementations of these tests available in the scientific Python ecosystem, all of them rely on approximations of the studentized range distribution, which are not accurate outside the range of inputs for which they are designed. Here we present the implementation of a very accurate and sufficiently fast implementation of the studentized range distribution and a function for performing Tukey’s HSD test. A thorough assessment of the methods, accuracy, and speed of the underlying calculations is available in ([Wallan et al., 2021a](#)), and an extensive test suite included in SciPy guards against regressions. Both of these features are available in SciPy 1.8.0.

## Statement of need

When analysis of variance (ANOVA) indicates that there is a statistically significant difference between at least one pair of groups in an experiment, researchers are often interested in *which* of the differences is statistically significant. Researchers use “post-hoc tests” to study these pairwise differences while controlling the experiment-wise error rate. Until recently, no post-hoc tests were available in SciPy ([Virtanen et al., 2020](#)), the de-facto standard library of fundamental algorithms for scientific computing in Python. To fill this gap, we contributed `scipy.stats.tukey_hsd` ([Wallan et al., 2021b](#)), a function for performing Tukey’s HSD test.

The most computationally-challenging part of implementing Tukey’s HSD test is the evaluation of the cumulative density function (CDF) of the studentized range distribution, which is given by

$$F(q; k, \nu) = \frac{k\nu^{\nu/2}}{\Gamma(\nu/2)2^{\nu/2-1}} \int_0^\infty \int_{-\infty}^\infty s^{\nu-1} e^{-\nu s^2/2} \phi(z) [\Phi(sq + z) - \Phi(z)]^{k-1} dz ds$$

<sup>\*</sup>Co-first author

<sup>†</sup>Co-first author

where  $q$  is the studentized range,  $k$  is the number of groups,  $\nu$  is the number of degrees of freedom used to determine the pooled sample variance, and  $\phi(z)$  and  $\Phi(z)$  represent the normal probability density function and normal cumulative distribution function, respectively. There is no closed-form expression for this integral, and numerical integration requires care, as naive evaluation of the integrand results in overflow even for modest values of the parameters. Consequently, other packages in the open-source scientific Python ecosystem, such as statsmodels (Seabold & Perktold, 2010) and Pingouin (Vallat, 2018), have relied on interpolation between tabulated values. To satisfy the need for a more accurate implementation of this integral, we contributed `scipy.stats.studentized_range` (Chmiel et al., 2021), a class that evaluates the CDF and many other functions of the distribution. Both statsmodels and Pingouin have since adopted this class to perform studentized range distribution calculations.

## Acknowledgements

We gratefully acknowledge the support of Chan Zuckerberg Initiative Essential Open Source Software for Science Grant EOSS-0000000432. Thanks also to reviewers Pamphile Roy, Nicholas McKibben, and Warren Weckesser.

## References

- Chmiel, D., Wallan, S., & Haberland, M. (2021). ENH: stats: Studentized range distribution. In *GitHub Pull Request*. GitHub. <https://github.com/scipy/scipy/pull/13732>
- Lund, R., & Lund, J. (1983). Algorithm AS 190: Probabilities and upper quantiles for the studentized range. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 32(2), 204–210. <https://doi.org/10.2307/2347300>
- Seabold, S., & Perktold, J. (2010). Statsmodels: Econometric and statistical modeling with Python. *Proceedings of the 9th Python in Science Conference*, 57, 61. <https://doi.org/10.25080/majora-92bf1922-011>
- Tukey, J. W. (1949). Comparing individual means in the analysis of variance. *Biometrics*, 99–114. <https://doi.org/10.2307/3001913>
- Vallat, R. (2018). Pingouin: Statistics in Python. *Journal of Open Source Software*, 3(31), 1026. <https://doi.org/10.21105/joss.01026>
- Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., others. (2020). SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods*, 17(3), 261–272. <https://doi.org/10.1038/s41592-019-0686-2>
- Wallan, S., Chmiel, D., & Haberland, M. (2021a). An accurate implementation of the studentized range distribution for Python. *Proceedings of the 9th Python in Science Conference*. <https://doi.org/10.25080/majora-1b6fd038-01e>
- Wallan, S., Chmiel, D., & Haberland, M. (2021b). ENH: stats: Tukey's honestly significant difference test. In *GitHub Pull Request*. GitHub. <https://github.com/scipy/scipy/pull/13002>