



bayesint: A Python Package for Calculating Bayesian Credible Intervals of Ratios of Beta Distributions

SOFTWARE
METAPAPER

M. BEKKER-NIELSEN DUNBAR 

THOMAS J. R. FINNIE 

**Author affiliations can be found in the back matter of this article*

][ubiquity press

ABSTRACT

We present *bayesint*, a Python package for calculating Bayesian credible intervals for ratios of two independent beta distributions such as are used when considering binomial data. Such data could be found in counts of events in medical diagnostic tests, case-control studies, and field epidemiological investigations, among other applications. The package contains general functions providing the expression of the density and distribution of the ratio and two functions for calculating the equal-tailed and highest posterior density credible intervals. The package is intended for use with 2×2 contingency tables. We introduce *bayesint* by comparing two groups in a contingency table through calculating the relative risk of cholera, showcasing its use in a novel context.

CORRESPONDING AUTHOR:

M. Bekker-Nielsen Dunbar

Emergency Response
Department, Public Health
England

maria.dunbar@phe.gov.uk

KEYWORDS:

beta distributions; cholera;
contingency tables; credible
interval; relative risk; Python

TO CITE THIS ARTICLE:

Bekker-Nielsen Dunbar M,
Finnie TJR 2021 bayesint: A
Python Package for Calculating
Bayesian Credible Intervals of
Ratios of Beta Distributions.
*Journal of Open Research
Software*, 9: 35. DOI: [https://
doi.org/10.5334/jors.283](https://doi.org/10.5334/jors.283)

(1) OVERVIEW

INTRODUCTION

The *bayesint* package for Python [20] allows users to calculate credible intervals for ratios found in 2x2 contingency tables. These include relative risks and odds ratios, frequently used in cohort and case control studies. The calculations behind *bayesint* are based on the cumulative distribution of two independent, beta-distributed random variables $Z = Y/X$, where $X \sim B(\alpha, \beta)$ and $Y \sim B(\theta, \phi)$. This work has previously been used to examine deaths during the second plague pandemic [see 1, for details].

STATISTICAL BACKGROUND

Deaths due to plague during the second plague pandemic were examined using a 2x2 contingency table framework. We considered those with counts given in notation of [Table 1](#).

We outline the considerations we made for the relative risk briefly. The relative risk for a dichotomous variable is given by the proportion of the probability of an outcome of interest and the probability of the other outcome. The relative risk for [Table 1](#) is

$$RR = \frac{PN}{MC} \tag{1}$$

which is our variable of interest.

Interval calculations

We consider the ratio of two independent beta distributed random variables $Z = Y/X$, where $X \sim B(\alpha, \beta)$ and $Y \sim B(\theta, \phi)$. Z has density [17]

$$f(z) = \begin{cases} \frac{B(\alpha + \theta, \beta)}{B(\alpha, \beta)B(\theta, \phi)} z^{\theta-1} {}_2F_1(\alpha + \theta, 1 - \phi; \alpha + \theta + \beta; z), & 0 < z \leq 1 \\ \frac{B(\alpha + \theta, \phi)}{B(\alpha, \beta)B(\theta, \phi)} z^{-(1+\alpha)} {}_2F_1\left(\alpha + \theta, 1 - \beta; \alpha + \theta + \phi; \frac{1}{z}\right), & z > 1 \end{cases} \tag{2}$$

and cumulative distribution function [see the appendix of 1, for full derivation]

$$F(z) = \begin{cases} \frac{B(\alpha + \theta, \beta)}{B(\alpha, \beta)B(\theta, \phi)} \frac{z^\theta}{\theta} {}_3F_2(1 - \phi, \alpha + \theta, \theta; \alpha + \theta + \beta, \theta + 1; z), & 0 < z \leq 1 \\ \frac{B(\theta + \alpha, \phi)}{B(\theta, \phi)B(\alpha, \beta)} \frac{z^{-\alpha}}{\alpha} {}_3F_2\left(\theta + \alpha, 1 - \beta, \alpha; \theta + \phi + \alpha, \alpha + 1; -\frac{1}{z}\right), & z > 1 \end{cases} \tag{3}$$

We add our data from [Table 1](#) with priors for the observed values π_1 and π_2 for positive classification and π_3 and π_4 for negative classification by inserting $\alpha = C + \pi_1$, $\beta = N - C + \pi_2$, $\theta = P + \pi_3$, and $\phi = M - P + \pi_4$ into Equation 3.

To obtain an equal-tailed, quantile interval (`eqt_int_frac`) we solve

$$F(u) = \delta / 2, \quad F(l) = 1 - \delta / 2 \tag{4}$$

To obtain a highest posterior density (`hpd_int_frac`), we minimise [3]

$$\min_{l < u} \left(|f(u) - f(l)| + |F(u) - F(l) - (1 - \delta)| \right) \tag{5}$$

to obtain the lower and upper limit of the 100(1-δ)% interval, where δ denotes the desired significance level. These intervals fulfil

$$P(l < z < u) = 1 - \delta \tag{6}$$

Contents

bayesint contains the functions listed in [Table 2](#). Our use of the word *frac* in naming these functions is to emphasise that

INCIDENCE			
CLASSIFICATION	+	-	TOTAL AT RISK
+	P	M - P	M
-	C	N - C	N
Total	C + P	N - C + M - P	N + M

Table 1 Example of 2x2 contingency table.

FUNCTION GROUPING	FUNCTION	DESCRIPTION
table_measures	rel_risk	Relative risk, a comparative measure for 2x2 contingency table
	odds_rat	Odds ratio, a comparative measure for 2x2 contingency table seen in Equation 1
	ratios	A wrapper function returning both the relative risk and odds ratio
table_tests	chi_sq_stat	χ² statistic for testing the hypothesis of no association between the two variables in a 2x2 contingency table
	chi_sq_test	χ² test for the statistic
random_variables	densi_frac	Prior density of a ratio of two independent beta distributions
	distri_frac	Posterior distribution of a ratio of two independent beta distributions
intervals	eqt_int_frac	The equal-tailed credible interval of a ratio of two independent beta distributions
	hpd_int_frac	The highest posterior density credible interval of a ratio of two independent beta distributions
	frac_int	A wrapper function returning the equal-tailed credible interval and the highest posterior density credible interval or both (chosen by the user)

Table 2 Functions in *bayesint*.

the functions depend on fractions, be they used for relative risks or for odds ratios. The functions from the subgroupings listed in [Table 2](#) are all loaded as part of *bayesint*. We have packaged the code to assist users in achieving immediate utility of the interval calculations as the underlying code is rather complicated. The package is released such that users can see the code as well as contribute to future versions of the package. To start using *bayesint*, all of the functions listed in [Table 2](#) can be loaded simultaneously by

```
>>> from bayesint import *
```

or

```
>>> import bayesint
```

or individually by

```
>>> from bayesint import FUNCTION
```

EXAMPLE: CHOLERA

Cholera is a disease caused by the bacterium *Vibrio cholerae*, first identified by Filippo Pacini in 1854 [2], brought to world-wide attention in 1883 by Robert Koch [7] and the subject of John Snow’s famous epidemiological study in 1855 [18]. Inoculation is a process where a smaller infection is induced in order that acquired immunity be educated. It carries a lower risk of mortality than that for those experiencing the full infection [18].

Data

Waldemar Mordecai Haffkine studied the effects of inoculation against cholera in India in 1893–1896 [5].

Our data source for Haffkine’s cholera inoculation study [5] is Greenwood and Yule, who collated data on cholera [4]. From their manuscript, we obtain the data listed in [Table 3](#). For each table from Greenwood and Yule we have data on whether subjects are inoculated against cholera (I) or not (NI), attacked by cholera (A) or not (NA), and the total counts (T). The numbers in [Table 3](#) refer to the collated tables [see 4 for further details]. These numbers are used in [Table 4](#) to identify the 2×2 contingency table for which the odds ratio and intervals are calculated.

Use of package

We illustrate how to use *bayesint* using the numbers from the top left [Table 3](#), that is the 2×2 table with $P = 3$, $C = 66$, $M = 279$, and $N = 539$. To calculate the odds ratio of such a 2×2 contingency table, we use the `rel_risk` function and receive the *RR* value in [Table 4](#). We assume the package has been loaded as listed at the end of the Contents section. An example of how to use the function `rel_risk` to calculate a relative risk can be seen below

```
>>> rel_risk(p_val = 27, c_val = 198, m_val = 5778, n_val = 6549)
2183/14124
```

In order to obtain the intervals (4, 5) for the *RR* above with prior beliefs $\pi_1 = \pi_2 = \pi_3 = \pi_4 = 2.5$ we run

```
>>> eqt_int_frac(p_val = 27, c_val = 198, m_val = 5778, n_val = 6549, pri_val = (2.5, 2.5,
```

	3			4			5			6		
	A	NA	T	A	NA	T	A	NA	T	A	NA	T
I	3	276	279	18	115	133	0	75	75	0	193	193
NI	66	473	539	120	520	640	19	778	797	6	723	729
T	69	749	818	138	635	773	19	853	872	6	916	922
	7			8			9			10		
	A	NA	T	A	NA	T	A	NA	T	A	NA	T
I	8	200	208	5	105	110	4	192	196	27	5751	5778
NI	20	182	202	11	88	99	34	113	147	198	6351	6549
T	28	382	410	16	193	209	38	305	343	225	12102	12327

Table 3 Haffkine’s cholera data from Greenwood and Yule [4] with numbers denoting the original tables they are from.

HAFFKINE	RR	INTERVAL	
		EQUAL-TAILED	HIGHEST POSTERIOR DENSITY
3	0.0878	[0.0513, 0.3352]	[0.0374, 0.3025] (HPD starting values: [0.1, 0.4])
4	0.7218	[0.5393, 1.0333]	[0.5218, 1.4436]
7	0.3885	[0.2205, 0.7060]	[0.1881, 0.7753]
8	0.4091	[0.2179, 0.7972]	[0.1476, 0.9875] (HPD starting values: [0.2, 0.5])
9	0.0882	[0.0353, 0.3319]	[0.0284, 0.3061]
10	0.1546	[0.0180, 0.3946]	[0.0421, 0.4364]

Table 4 Relative risks and credible intervals for data from Table 3.

```
2.5, 2.5), frac_type = "risk", signif =
0.05, ans = "estim")
(2183/14124, 0.0180001733621202,
0.39464791108754)
```

and

```
>>> hpd_int_frac(p_val = 27, c_val = 198, m_val
= 5778, n_val = 6549, pri_val = (2.5, 2.5,
2.5, 2.5), frac_type = "risk", signif =
0.05, minimisation_start = None)
(2183/14124, 0.0421057055996, 0.436364556526)
```

for this example we could have also run `frac_ints` as this assumes an argument of "estim" for the `ans` parameter in `eqt_int_frac` and no provided starting values for the `minimisation_start` in `hpd_int_frac`.

```
>>> frac_ints(p_val = 27, c_val = 198, m_val =
5778, n_val = 6549, pri_val = (2.5, 2.5, 2.5,
2.5), frac_type = "risk", signif = 0.05)
((2183/14124, 0.0180001733621202,
0.39464791108754), (2183/14124,
0.0421057055996, 0.436364556526))
```

If the user does not want those defaults, we recommend that they run the two interval functions separately. As we use *SymPy*, we have the option to get the exact equal-tailed interval. This is retrieved by

```
>>> eqt_int_frac(p_val = 27, c_val = 198, m_val
= 5778, n_val = 6549, pri_val = (2.5, 2.5,
2.5, 2.5), frac_type = "risk", signif =
0.05, ans = "exact")
```

We can engage pretty printing using the `init_printing` function from *SymPy*, which allows us to obtain the result in symbolic mathematic notation, i.e.

$$\left\{ \frac{2183}{14124}, \left\{ \begin{aligned} & z | z \in (-\infty, 1] \wedge z^{29.5} {}_3F_2 \left(\begin{matrix} -5752.5, 230.0, 29.5 \\ 6583.5, 30.5 \end{matrix} \middle| z \right) \\ & -9.21205115784183 \cdot 10^{-38} = 0 \end{aligned} \right\} \cup \right. \\ \left. \left\{ \begin{aligned} & z | z \in (1, \infty) \wedge \frac{1}{z^{200.5}} {}_3F_2 \left(\begin{matrix} 230.0, -6352.5, 200.5 \\ 5983.5, 201.5 \end{matrix} \middle| -\frac{1}{z} \right) \\ & -1.17798248329339 \cdot 10^{-46} = 0 \end{aligned} \right\}, \right. \\ \left\{ \begin{aligned} & z | z \in (-\infty, 1] \wedge z^{29.5} {}_3F_2 \left(\begin{matrix} -5752.5, 230.0, 29.5 \\ 6583.5, 30.5 \end{matrix} \middle| z \right) \\ & -3.59269995155831 \cdot 10^{-36} = 0 \end{aligned} \right\} \cup \\ \left. \left\{ \begin{aligned} & z | z \in (1, \infty) \wedge \frac{1}{z^{200.5}} {}_3F_2 \left(\begin{matrix} 230.0, -6352.5, 200.5 \\ 5983.5, 201.5 \end{matrix} \middle| -\frac{1}{z} \right) \\ & -4.59413168484424 \cdot 10^{-45} = 0 \end{aligned} \right\} \right\}$$

which is the expression we evaluated to obtain numerical values when the "estim" option was used. We can also print the density and distribution functions, 2 and 3), e.g.

```
>>> distri_frac(p_val = 27, c_val = 198, m_val
= 5778, n_val = 6549, pri_val = (2.5, 2.5,
2.5, 2.5), frac_type = "risk")
```

$$\left\{ \begin{aligned} & \frac{z^{P+\pi_2} \beta(C+P+\pi_0+\pi_2, -C+N+\pi_1)}{(P+\pi_2)\beta(C+\pi_0, -C+N+\pi_1)\beta(P+\pi_2, M-P+\pi_3)} \\ & {}_3F_2 \left(\begin{matrix} -M+P-\pi_3+1, C+P+\pi_0+\pi_2, P+\pi_2 \\ N+P+\pi_0+\pi_1+\pi_2, P+\pi_2+1 \end{matrix} \middle| z \right) \text{ for } z \leq 1 \end{aligned} \right\} \\ \left\{ \begin{aligned} & \frac{z^{-C-\pi_0} \beta(C+P+\pi_0+\pi_2, M-P+\pi_3)}{(C+\pi_0)\beta(C+\pi_0, -C+N+\pi_1)\beta(P+\pi_2, M-P+\pi_3)} \\ & {}_3F_2 \left(\begin{matrix} C+P+\pi_0+\pi_2, C-N-\pi_1+1, C+\pi_0 \\ C+M+\pi_0+\pi_2+\pi_3, C+\pi_0+1 \end{matrix} \middle| -\frac{1}{z} \right) \text{ for } z \leq 1 \end{aligned} \right\}$$

SymPy also allows for outputs in LaTeX format. By default *bayesint* does not pretty print.

Intervals for odds ratios

We use the `eqt_int_frac` and `hpd_int_frac` commands from the previous section to calculate the 95% credible intervals for the data from [Table 3](#). In the cases where $RR=0$ due to zero counts, we are not able to calculate credible intervals due to the assumption that $P > \pi_3$ which arises from (3). This means we cannot calculate intervals for Haffkine Tables 5 and 6 in [Table 3](#). We use the priors $\pi_i = 2.5, i = 1, \dots, 4$.

As with all numerical optimisation problems, different starting values can lead to disparate solutions, and as there may not be a closed-form expression of the solution to the highest posterior density minimisation, users are encouraged to investigate their own starting parameters.

COMPARISONS

We compare our *bayesint* Python solution with the two following calculation options for calculating the uncertainty of a ratio of beta distributions. Matsen IV et al. [13] uses the same Pham-Gia [17] density that we consider, but does not explicitly calculate the distribution on the basis of this density and Sverdlov et al. [19] consider Nurminen and Mutanen [15] which is subsumed in Bekker-Nielsen Dunbar et al. [1]. As such, we expect them to contain at least an implementation of the equal-tailed credible interval, as it is the easier of the two to program. At the time of writing, we are unaware of other options besides those listed in [Table 5](#).

The strength of our approach compared with these implementations is that we have considered use of priors for the proportions p_1 and p_2 that need not be the same. Sverdlov et al. [19] mention the use of informative priors

NAME	LANGUAGE	AUTHOR(S)
betarat version 1.0.0	Python	Matsen IV et al. [13]
bayesian2beta	R	supplement to Sverdlov et al. [19]

Table 5 Available options for calculating uncertainty of ratios using credible intervals.

in their conclusion but do not seem to have used them in their paper. Furthermore their code allows the user to provide different priors for the equal-tailed interval, whereby the symmetry assumption behind the equal-tailed interval might be violated. The implementation from Matsen IV et al. [13] considers the situation where both Beta distributions have the same priors, i.e. $\alpha = \theta$ and $\beta = \varphi$. They also note issues with hypergeometrics in R, which we also found, which was our motivation for using Python. Finally, our implementation is the only one of the three that can provide the symbolic functions, as showcased in the Use of package section. We are not aware of other software which allows for calculation of both the HPD and the equal-tailed interval.

DISCUSSION

Our implementation has been demonstrated with examples from epidemiology but can be used in any science where 2×2 contingency table-like data arise. For example, within the field of ecology, the response ratio is an index used in meta-analysis of ecological experiments. It is implied in Hedges et al. [6] and Lajeunesse [11] that only confidence intervals are considered for these, thus our method will allow the calculation of credible intervals for this measure. Another metric mentioned in Lajeunesse [12] is the odds ratio, which is similar to the relative risk in certain situations. Hedges et al. [6] and Lajeunesse [12] consider the distribution of this ratio to which our method is applicable. Another biological ratio we could consider examining is mutation frequencies. Significance tests in this area rely on ratios of mutation rates [10]. Indeed, Matsen IV et al. [13]'s work seems to have arisen from work on hypermutations. We have named *bayesint* with the intention that it will be expanded to contain other intervals with Bayesian origins.

CONCLUSION

We have created a Python package *bayesint* which calculates two types of credible interval for the ratio of beta distributions. We used the package to calculate the credible intervals of the relative risk of subjects inoculated or not being attacked by cholera and highlighted the strengths of our package compared with other available options. We believe our implementation to be a useful tool for analysis of 2×2 contingency tables.

Implementation and architecture

The *bayesint* package is freely available at GitHub (<https://github.com/PublicHealthEngland/bayesint>) and through the PyPI index of Python packages (pypi.org/project/bayesint). This manuscript concerns *bayesint* version 1.0.3, the package's documentation will be updated if future versions have different requirements.

Quality control

bayesint is extensively tested in Python 2.7.14 and also functions in Python 3.5.5 and Python 3.6.6.

(2) AVAILABILITY OPERATING SYSTEM

Any operating system where Python 2.7.14 or better or 3.5.5 or better with the dependencies enumerated below are installed. In practice this includes all three major operating systems (Linux/Unix, Windows 7 or better and Mac OS/OS X 10.6 or newer) together with a number of smaller platforms.

PROGRAMMING LANGUAGE

Python 2.7.14, 3.5.5 and 3.6.6.

ADDITIONAL SYSTEM REQUIREMENTS

This software has no particularly unusual system requirements and should operate on any typical desktop machine produced in the last five years. The calculations are CPU bound so the greater the single thread performance of the machine the more swiftly the computations will complete.

DEPENDENCIES

bayesint uses the following Python packages which are installed when *bayesint* is installed:

- *SymPy* ≥ 1.1.1 [14] A library for symbolic mathematics used to calculate the densities and distributions used in our credible intervals.
- *SciPy* ≥ 0.19.1 [9]. A library used for scientific computing used to obtain a χ^2 distribution for testing and to minimise the functions considered in the credible intervals.
- *NumPy* ≥ 1.13.3 [16]. A library with high-level mathematical functions used for vectorising our *SymPy* function prior to minimisation.
- *mpmath* ≥ 0.19 [8]. A library for arbitrary floating point arithmetic used in the minimisation. It is loaded through *SymPy*.

SOFTWARE LOCATION

Archive

Name: PyPI

Persistent identifier: pypi.org/project/bayesint

Licence: Open Government Licence 3.0

Publisher: Public Health England

Version published: 1.0.3

Date published: 11/04/2019

Code repository

Name: GitHub

Persistent identifier: github.com/PublicHealthEngland/bayesint

Licence: Open Government Licence 3.0

Date published: 11/04/2019

LANGUAGE

English

(3) REUSE POTENTIAL

We broadly outline the reuse potential of this software in the discussion section of this work but as a recapitulation this package permits the calculation of credible intervals wherever 2×2 type contingency table data appear in science. With the additional benefit that prior knowledge may be incorporated into this estimate in a truly Bayesian manner. Issues and feedback about the software can be provided by users through the GitHub issue logging system.

FUNDING STATEMENT

The research leading to these results has received UK government grant-in-aid funding. MBND and TJRF were funded by the National Institute for Health Research—Health Protection Research Unit for Modelling Methodology (HPRU MM). The funding bodies (UK government and HPRU MM) did not play any role in the formulation of this manuscript. The views expressed in this publication are those of the authors and not necessarily those of Public Health England.

COMPETING INTERESTS

The authors have no competing interests to declare.

AUTHOR AFFILIATIONS

M. Bekker-Nielsen Dunbar  orcid.org/0000-0002-7249-3524
Emergency Response Department, Public Health England, GB

Thomas J. R. Finnie  orcid.org/0000-0001-5962-4211
Emergency Response Department, Public Health England, GB

REFERENCES

- Bekker-Nielsen Dunbar M, Finnie TJR, Sloane B, Hall IM.** Methods for calculating credible intervals for ratios of beta distributions with application to relative risks of premature death from the second plague pandemic. *PLOS ONE*. 2019; 14(2): e0211633. DOI: <https://doi.org/10.1371/journal.pone.0211633>
- Bentivoglio M, Pacini P.** Filippo Pacini: A determined observer. *Brain Res. Bull.* 1995; 38(2): 161–65. DOI: [https://doi.org/10.1016/0361-9230\(95\)00083-Q](https://doi.org/10.1016/0361-9230(95)00083-Q)
- Chen M-H, Shao Q-M.** Monte Carlo estimation of Bayesian credible and HPD intervals. *J. Comput. Graph. Stat.* 1999; 8(1): 69–92. DOI: <https://doi.org/10.1080/10618600.1999.10474802>
- Greenwood M, Yule GU.** The statistics of anti-typhoid and anti-cholera inoculations, and the interpretation of such statistics in genera. *Proc. R. Soc. Med.* 1915; 8(Sect. Epidemiol. State Med.): 113–194. DOI: <https://doi.org/10.1177/003591571500801433>
- Haffkine WM.** *Protective Inoculation Against Cholera.* Calcutta: Thacker, Spink & Co; 1913.
- Hedges LV, Gurevitch J, Curtis PS.** The meta-analysis of response ratios in experimental ecology. *Ecology*. 1999; 80(4): 1150–1156. DOI: [https://doi.org/10.1890/0012-9658\(1999\)080\[1150:TMAORR\]2.0.CO;2](https://doi.org/10.1890/0012-9658(1999)080[1150:TMAORR]2.0.CO;2)
- Howard-Jones N.** Robert Koch and the cholera vibrio: A centenary. *BMJ (Clin. Res. Ed.)*. 1984; 288(6414): 379–81. DOI: <https://doi.org/10.1136/bmj.288.6414.379>
- Johansson F, et al.** *mpmath: A Python Library For Arbitrary-Precision Floating-Point Arithmetic*; 2013. URL <http://mpmath.org>.
- Jones E, Oliphant T, Peterson P, et al.** *SciPy: Open source scientific tools for Python*; 2001. URL <http://www.scipy.org>.
- Kastenbaum MA, Bowman KO.** Tables for determining the statistical significance of mutation frequencies. *Mutat. Res.* 1970; 9(5): 527–549. DOI: [https://doi.org/10.1016/0027-5107\(70\)90038-2](https://doi.org/10.1016/0027-5107(70)90038-2)
- Lajeunesse MJ.** On the meta-analysis of response ratios for studies with correlated and multi-group designs. *Ecology*. 2011; 92(11): 2049–2055. DOI: <https://doi.org/10.1890/11-0423.1>
- Lajeunesse MJ.** Bias and correction for the log response ratio in ecological meta-analysis. *Ecology*. 2015; 96(8): 2056–2063. DOI: <https://doi.org/10.1890/14-2402.1>
- Matsen IV FA, Small CT, Soliven K, Engel GA, Feeroz MM, Wang X, Craig KL, Hasan MK, Emerman M, Linial ML, Jones-Engel L.** A novel Bayesian method for detection of apobec3-mediated hypermutation and its application to zoonotic transmission of simian foamy viruses. *PLOS Comput. Biol.* 2014; 10(2): 1–14. DOI: <https://doi.org/10.1371/journal.pcbi.1003493>
- Meurer A, Smith CP, Paprocki M, Čertik O, Kirpichev SB, Rocklin M, Kumar A, Ivanov S, Moore JK, Singh S, Rathnayake T, Vig S, Granger BE, Muller RP, Bonazzi F, Gupta H, Vats S, Johansson F, Pedregosa F, Curry MJ, Terrel AR, Roučka Š, Saboo A, Fernando I, Kulal S, Cimrman R, Scopatz A.** *SymPy: Symbolic computing in Python.* *PeerJ Comput. Sci.* 2017; 3: e103. DOI: <https://doi.org/10.7717/peerj-cs.103>
- Nurminen M, Mutanen P.** Exact Bayesian analysis of two proportions. *Scand. J. Stat.* 1987; 14(1): 67–77.
- Oliphant TE.** *Guide To NumPy.* CreateSpace Independent Publishing Platform, 2nd edition; 2015.
- Pham-Gia T.** Distributions of the ratios of independent beta variables and applications. *Commun. Stat. – Theory Methods.* 2000; 29(12): 2693–2715. DOI: <https://doi.org/10.1080/03610920008832632>
- Sherman IW.** *The Power Of Plagues.* ASM Press, 2nd edition; 2017. DOI: <https://doi.org/10.1128/9781683670018>
- Sverdlov O, Ryznik Y, Wu S.** Exact Bayesian inference comparing binomial proportions, with application to proof-of-concept clinical trials. *Ther. Innov. Regul. Sci.* 2015; 49(1): 163–174. DOI: <https://doi.org/10.1177/2168479014547420>
- Python Software Foundation.** *Python Software, Version 2.7*, 2010. URL <http://www.python.org>.

TO CITE THIS ARTICLE:

Bekker-Nielsen Dunbar M, Finnie TJR 2021 bayesint: A Python Package for Calculating Bayesian Credible Intervals of Ratios of Beta Distributions. *Journal of Open Research Software*, 9: 35. DOI: <https://doi.org/10.5334/jors.283>

Submitted: 19 June 2019 Accepted: 11 November 2021 Published: 21 December 2021

COPYRIGHT:

© 2021 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC-BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. See <http://creativecommons.org/licenses/by/4.0/>.

Journal of Open Research Software is a peer-reviewed open access journal published by Ubiquity Press.

