

# Package ‘RImpact’

October 31, 2018

**Type** Package

**Title** Calculates Measures of Scholarly Impact

**Version** 1.0

**Author** John Ruscio

**Maintainer** John Ruscio <ruscio@tcnj.edu>

**Description** The metrics() function calculates measures of scholarly impact. These include conventional measures, such as the number of publications and the total citations to all publications, as well as modern and robust metrics based on the vector of citations associated with each publication, such as the h index and many of its variants or rivals. These methods are described in Ruscio et al. (2012) <DOI: 10.1080/15366367.2012.711147>.

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2018-10-31 12:40:03 UTC

## R topics documented:

GeometricMean . . . . .	2
HarmonicMean . . . . .	2
Metrics . . . . .	3
<b>Index</b>	<b>5</b>

---

GeometricMean

*GeometricMean*

---

**Description**

Calculates the geometric mean (the  $n$ th root of the product of  $n$  values).

**Usage**

GeometricMean(x)

**Arguments**

x                      Vector of  $n$  values whose geometric mean is to be calculated.

**Value**

The geometric mean of  $x$ .

**Author(s)**

John Ruscio

**References**

Ruscio et al. (2012)

**Examples**

GeometricMean(c(1, 3, 9, 27, 81))

---

HarmonicMean

*HarmonicMean*

---

**Description**

Calculates the harmonic mean (the reciprocal of the arithmetic mean of the reciprocals of  $n$  values).

**Usage**

HarmonicMean(x)

**Arguments**

x                      Vector of  $n$  values whose harmonic mean is to be calculated.

**Value**

The harmonic mean of  $x$ .

**Author(s)**

John Ruscio

**References**

Ruscio et al. (2012)

**Examples**

```
HarmonicMean(c(1, 3, 9, 27, 81))
```

---

Metrics

*Metrics*

---

**Description**

Measures scholarly impact using modern citation-based indices.

**Usage**

```
Metrics(citation.counts, publishing.age = 0, display = TRUE)
```

**Arguments**

<code>citation.counts</code>	Number of times each article has been cited. (vector)
<code>publishing.age</code>	Age of the first article author has published. (scalar)
<code>display</code>	Whether to display metrics (if TRUE, the default) or direct output to a file (if FALSE).

**Value**

<code>h.index</code>	$h$ index, the largest number $h$ such that at least $h$ articles are cited $h$ times each (Hirsch, 2005).
<code>tapered.h.index</code>	Tapered $h$ index, credit decreases for citations farther from the origin (Anderson, Hankin, & Killworth, 2008).
<code>f.index</code>	$f$ index, largest value $f$ such that the harmonic mean of the $f$ most highly cited articles is at least $f$ (Tol, 2009).
<code>g.index</code>	$g$ index, largest value $g$ such that the mean citations for the $g$ most highly cited articles is at least $g$ (Egghe, 2006).

hg.index	hg index, geometric mean of h and g (Alonso, Cabrerizo, Herrera-Viedma, & Herrera, 2010).
a.index	a index, mean citations for the papers in Hirsch core (Jin, 2006).
m.index	m index, median citations for papers in Hirsch core (Bornmann, Mutz, Daniel, 2008).
r.index	r index, square root of citations for papers in Hirsch core (Jin, Liang, Rousseau, Egghe, 2007).
weighted.h.index	h index weighted by citation impact (Egghe & Rousseau, 2008).
q2.index	q2 index, geometric mean of h and m indexes (Cabrerizo, Alonso, Herrera-Viedma, & Herrera, 2010).
e.index	e index, excess citations for papers in Hirsch core (Zhang, 2009).
max.product	Maximum product index, maximum product of article's rank and citation count (Kosmulski, 2007).
sqrt.max.product	Rescales maximum product index from an area to a distance measure.
h2.index	h2 index, analogous to h index with more stringent criterion (Kosmulski, 2006).
m.quotient	m quotient, controlling h index for publishing age (Hirsch, 2005).
tapered.m.quotient	Controlling tapered h index for publishing age.

### Author(s)

John Ruscio

### References

Ruscio et al. (2012)

### Examples

```
# Running the Metrics program with illustrative data from Ruscio et al. (2012)
x <- c(24, 18, 12, 8, 6, 5, 5, 4, 4, 3, 2, 2, 1, 1, 1, 0, 0, 0)
Metrics(x)
```

# Index

GeometricMean, [2](#)

HarmonicMean, [2](#)

Metrics, [3](#)