

# Package ‘eddington’

March 24, 2020

**Title** Compute a Cyclist's Eddington Number

**Version** 2.1.2

**Description** Compute a cyclist's Eddington number, including efficiently computing cumulative E over a vector. A cyclist's Eddington number <[https://en.wikipedia.org/wiki/Arthur\\_Eddington#Eddington\\_number\\_for\\_cycling](https://en.wikipedia.org/wiki/Arthur_Eddington#Eddington_number_for_cycling)> is the maximum number satisfying the condition such that a cyclist has ridden E miles or greater on E distinct days. The algorithm in this package is an improvement over the conventional approach because both summary statistics and cumulative statistics can be computed in linear time, since it does not require initial sorting of the data. These functions may also be used for computing h-indices for authors, a metric described by Hirsch (2005) <[doi:10.1073/pnas.0507655102](https://doi.org/10.1073/pnas.0507655102)>. Both are specific applications of computing the side length of a Durfee square <[https://en.wikipedia.org/wiki/Durfee\\_square](https://en.wikipedia.org/wiki/Durfee_square)>.

**License** GPL (>=2)

**Encoding** UTF-8

**LazyData** true

**Depends** R (>= 3.4.0)

**LinkingTo** Rcpp

**Imports** Rcpp

**Suggests** testthat,  
knitr,  
rmarkdown,  
dplyr

**VignetteBuilder** knitr

**RoxygenNote** 7.1.0

**URL** <https://github.com/pegeler/eddington2>

**BugReports** <https://github.com/pegeler/eddington2/issues>

## R topics documented:

E\_cum . . . . . 2



**Value**

A named list with the current Eddington number (E) and the number of rides required to increment by one (req).

**See Also**

[E\\_cum](#), [E\\_num](#), [E\\_req](#), [E\\_sat](#)

---

E\_num

*Get the Eddington number for cycling*

---

**Description**

Gets the **Eddington number for cycling**. The Eddington Number for cycling,  $E$ , is the maximum number where a cyclist has ridden  $E$  miles on  $E$  distinct days.

**Usage**

```
E_num(rides)
```

**Arguments**

rides                      A vector of mileage, where each element represents a single day.

**Details**

The Eddington Number for cycling is related to computing the rank of an integer partition, which is the same as computing the side length of its **Durfee square**. Another relevant application of this metric is computing the **Hirsch index** for publications.

This is not to be confused with the **Eddington Number in astrophysics**,  $N_{Edd}$ , which represents the number of protons in the observable universe.

**Value**

An integer which is the Eddington cycling number for the data provided.

**See Also**

[E\\_cum](#), [E\\_next](#), [E\\_req](#), [E\\_sat](#)

**Examples**

```
# Randomly generate a set of 15 rides
rides <- rgamma(15, shape = 2, scale = 10)

# View the rides sorted in decreasing order
setNames(sort(rides, decreasing = TRUE), seq_along(rides))

# Get the Eddington number
E_num(rides)
```

---

E_req	<i>Determine the number of additional rides required to achieve a specified Eddington number</i>
-------	--

---

**Description**

Determine the number of additional rides required to achieve a specified Eddington number.

**Usage**

```
E_req(rides, candidate)
```

**Arguments**

rides	A vector of mileage, where each element represents a single day.
candidate	The Eddington number to test for.

**Value**

An integer vector of length 1. Returns 0L if  $E$  is already achieved.

**See Also**

[E\\_cum](#), [E\\_next](#), [E\\_num](#), [E\\_sat](#)

---

E_sat	<i>Determine if a dataset satisfies a specified Eddington number</i>
-------	--

---

**Description**

Indicates whether a certain Eddington number is satisfied, given the data.

**Usage**

```
E_sat(rides, candidate)
```

**Arguments**

`rides` A vector of mileage, where each element represents a single day.  
`candidate` The Eddington number to test for.

**Value**

A logical vector of length 1.

**See Also**

[E\\_cum](#), [E\\_next](#), [E\\_num](#), [E\\_req](#)

---

<code>rides</code>	<i>A year of simulated bicycle ride mileages</i>
--------------------	--

---

**Description**

Simulated dates and distances of rides occurring in 2009.

**Usage**

`rides`

**Format**

A data frame with 250 rows and 2 variables:

**ride\_date** date the ride occurred

**ride\_length** the length in miles

**Details**

The dataset contains a total of 3,419 miles spread across 178 unique days. The Eddington number for the year was 29.

# Index

## \*Topic **datasets**

rides, [5](#)

E\_cum, [2](#), [3–5](#)

E\_next, [2](#), [2](#), [3–5](#)

E\_num, [2](#), [3](#), [3](#), [4](#), [5](#)

E\_req, [2](#), [3](#), [4](#), [5](#)

E\_sat, [2–4](#), [4](#)

rides, [5](#)