

Package: HistDat (via r-universe)

September 5, 2024

Type Package

Title Summary Statistics for Histogram/Count Data

Version 0.2.0

Author Michael Milton

Maintainer Michael Milton <michael.r.milton@gmail.com>

Description In some cases you will have data in a histogram format, where you have a vector of all possible observations, and a vector of how many times each observation appeared. You could expand this into a single 1D vector, but this may not be advisable if the counts are extremely large. 'HistDat' allows for the calculation of summary statistics without the need for expanding your data.

License GPL (>=3)

Encoding UTF-8

Suggests knitr, rmarkdown, testthat (>= 3.0.0),

Config/testthat/edition 3

RoxygenNote 7.1.1

Roxygen list(markdown = TRUE)

VignetteBuilder knitr

Imports methods, stats

Repository <https://multimeric.r-universe.dev>

RemoteUrl <https://github.com/multimeric/histdat>

RemoteRef HEAD

RemoteSha d33e1ca8071c5b4ea777fab8485921201b056e4f

Contents

| | |
|----------------------------------|---|
| HistDat-package | 2 |
| as.ecdf | 3 |
| as.ecdf,HistDat-method | 4 |

| | |
|------------------------------------|----|
| as.vector,HistDat-method | 4 |
| c,HistDat-method | 5 |
| HistDat | 6 |
| HistDat-class | 6 |
| length,HistDat-method | 7 |
| max,HistDat-method | 7 |
| mean,HistDat-method | 8 |
| median,HistDat-method | 9 |
| min,HistDat-method | 9 |
| quantile,HistDat-method | 10 |
| range,HistDat-method | 11 |
| sd,HistDat-method | 11 |
| sort,HistDat-method | 12 |
| sum,HistDat-method | 13 |
| var,HistDat-method | 13 |
| [,HistDat-method | 14 |

| | |
|--------------|-----------|
| Index | 15 |
|--------------|-----------|

| | |
|-----------------|---|
| HistDat-package | <i>'HistDat': Summary statistics for histogram/count data</i> |
|-----------------|---|

Description

In some cases you will have data in a "histogram" format, where you have a vector of all possible observations, and a vector of how many times each observation appeared. You could expand this into a single 1D vector, but this may not be advisable if the counts are extremely large. 'HistDat' allows for the calculation of summary statistics without the need for expanding your data.

Details

Note that all the methods described for HistDat instances have been transformed into generic methods in this package where they are not already, with default implementations for general numeric vectors. This allows you to equally apply these same functions to any type of data.

Class Definition and Constructor Function

- [HistDat](#)
- [HistDat\(\)](#)

HistDat Statistics

- [length,HistDat-method](#)
- [max,HistDat-method](#)
- [min,HistDat-method](#)
- [range,HistDat-method,](#)
- [median,HistDat-method](#)

- [mean,HistDat-method](#)
- [var,HistDat-method](#)
- [sd,HistDat-method](#)
- [sum,HistDat-method](#)
- [quantile,HistDat-method](#)
- [as.ecdf,HistDat-method](#)

HistDat Utilities

- [as.vector,HistDat-method](#)
- [\[,HistDat-method](#)
- [c,HistDat-method](#)

Misc Functions

- [as.ecdf\(\)](#)

| | |
|---------|--|
| as.ecdf | <i>Converts an object to an empirical cumulative density function. This is a generic function.</i> |
|---------|--|

Description

Converts an object to an empirical cumulative density function. This is a generic function.

Usage

```
as.ecdf(x)
```

Arguments

x The object to coerce to a eCDF

Value

An instance of the "ecdf" class

See Also

[ecdf\(\)](#)

Examples

```
cdf <- as.ecdf(1:4)
cdf(2) # returns 0.5
```

as.ecdf,HistDat-method

Converts this histogram to an instance of the "ecdf" class, allowing the calculation of cumulative densities, and quantiles

Description

Converts this histogram to an instance of the "ecdf" class, allowing the calculation of cumulative densities, and quantiles

Usage

```
## S4 method for signature 'HistDat'  
as.ecdf(x)
```

Arguments

x An instance of the class HistDat

Value

An instance of the ecdf class. It can be invoked as a function to return the cumulative proportion of the count data less than or equal to x.

Examples

```
hd <- HistDat(vals = 1:3, counts = c(1, 2, 1))  
cdf <- as.ecdf(hd)  
cdf(2) # returns 0.75
```

as.vector,HistDat-method

Converts this histogram to a vector. Not recommended if there are many counts as this would result in an incredibly long vector

Description

Converts this histogram to a vector. Not recommended if there are many counts as this would result in an incredibly long vector

Usage

```
## S4 method for signature 'HistDat'  
as.vector(x)
```

Arguments

x An instance of the class HistDat

Value

A vector with the same length as x, but as a 1-D vector with an element for each count in the counts vector. In other words, all length(x) observations will be represented as a single element instead of being just counted as in the original HistDat object.

Examples

```
hd <- HistDat(vals = 1:3, counts = c(1, 2, 1))
as.vector(hd) # returns 1 2 2 3
```

c,HistDat-method *Concatenate observations into this instance*

Description

Concatenate observations into this instance

Usage

```
## S4 method for signature 'HistDatCompatible'
c(x, ...)
```

Arguments

x The first value to concatenate
 ... The remaining values to concatenate

Value

A new HistDat object, with the other numeric values integrated into it

Examples

```
hd <- HistDat(vals = 1:3, counts = c(1, 2, 1))
hd_2 <- c(1, 1, hd)
hd@counts # returns 1 2 1
hd_2@counts # returns 3 2 1, as the first value now has 2 more counts
hd_2@vals # returns 1 2 3 (this is unchanged)
```

HistDat *The constructor function for the HistDat class. This is the only official way to create an instance of this class.*

Description

The constructor function for the HistDat class. This is the only official way to create an instance of this class.

Usage

```
HistDat(vals, counts)
```

Arguments

vals A vector of observation values, ie all the possible values that could be observed
counts A vector of counts, each of which corresponds to the same index in the vals parameter

Examples

```
hd <- HistDat::HistDat(vals = 1:3, counts = c(1, 2, 1))  
length(hd) # returns 4
```

HistDat-class *S4 class for histogram data*

Description

S4 class for histogram data

Slots

vals A vector of observations
counts A vector of counts, each of which corresponds to the same index in the vals parameter

length,HistDat-method *Calculates the total number of observations in a histogram dataset*

Description

Calculates the total number of observations in a histogram dataset

Usage

```
## S4 method for signature 'HistDat'  
length(x)
```

Arguments

x An instance of the class HistDat

Value

A numeric of length 1, holding the number of observations in the dataset

Examples

```
hd <- HistDat(vals = 1:3, counts = c(1, 2, 1))  
length(hd) # returns 4
```

max,HistDat-method *Calculates the largest observation in the histogram dataset*

Description

Calculates the largest observation in the histogram dataset

Usage

```
## S4 method for signature 'HistDat'  
max(x, ..., na.rm = FALSE)
```

Arguments

x An instance of the class HistDat
... Passed verbatim to `base::max()`
na.rm Passed verbatim to `base::max()`

Value

A numeric of length 1, holding the largest observation in the dataset

Examples

```
hd <- HistDat(vals = 1:3, counts = c(1, 2, 1))
max(hd) # returns 3
```

mean,HistDat-method *Calculates the mean value of all observations in the histogram dataset*

Description

Calculates the mean value of all observations in the histogram dataset

Usage

```
## S4 method for signature 'HistDat'
mean(x, ...)
```

Arguments

| | |
|-----|---|
| x | An instance of the class HistDat |
| ... | Additional arguments that will be ignored |

Details

An S3 and S4 generic is defined for this method, allowing compatibility with existing code that calls `base::mean()` instead of `[mean()]`, which is defined as an S4 generic in this package

Value

A numeric of length 1, holding the mean of the observations in the dataset

Examples

```
hd <- HistDat(vals = 1:3, counts = c(1, 2, 1))
mean(hd) # returns 2
```

median,HistDat-method *Calculates the median value of the observations in the histogram dataset*

Description

Calculates the median value of the observations in the histogram dataset

Usage

```
## S4 method for signature 'HistDat'  
median(x, na.rm = FALSE, ...)
```

Arguments

| | |
|-------|--|
| x | An instance of the class HistDat |
| na.rm | Provided for compatibility with <code>stats::median()</code> , but ignored |
| ... | Additional arguments that will be ignored |

Details

An S3 and and S4 generic is defined for this method, allowing compatibility with existing code that calls `stats::median()` instead of `median`, which is defined as an S4 generic in this package

Value

A numeric of length 1, holding the median value of the observations in the histogram dataset

Examples

```
hd <- HistDat(vals = 1:3, counts = c(1, 2, 1))  
median(hd) # returns 2
```

min,HistDat-method *Calculates the smallest observation in the histogram dataset*

Description

Calculates the smallest observation in the histogram dataset

Usage

```
## S4 method for signature 'HistDat'  
min(x, ..., na.rm = FALSE)
```

Arguments

| | |
|-------|---|
| x | An instance of the class HistDat |
| ... | Passed verbatim to <code>base::min()</code> |
| na.rm | Passed verbatim to <code>base::min()</code> |

Value

A numeric of length 1, holding the smallest observation in the dataset

Examples

```
hd <- HistDat(vals = 1:3, counts = c(1, 2, 1))
min(hd) # returns 1
```

quantile,HistDat-method

Returns the empirical quantiles of the observations represented by this class

Description

Returns the empirical quantiles of the observations represented by this class

Usage

```
## S4 method for signature 'HistDat'
quantile(x, ...)
```

Arguments

| | |
|-----|---|
| x | An instance of the class HistDat |
| ... | Remaining arguments to pass to <code>stats::quantile()</code> |

Examples

```
hd <- HistDat(vals = 1:3, counts = c(1, 2, 1))
quantile(hd, 0.1) # returns 1.3
```

range,HistDat-method *Calculates the range of values of the observations in the histogram dataset*

Description

Calculates the range of values of the observations in the histogram dataset

Usage

```
## S4 method for signature 'HistDat'  
range(x, ..., na.rm = FALSE)
```

Arguments

| | |
|-------|---|
| x | An instance of the class HistDat |
| ... | Additional arguments to pass to range() |
| na.rm | Passed verbatim to <code>base::range()</code> |

Value

A numeric of length 2, indicating the minimum and maximum value of the observations

Examples

```
hd <- HistDat(vals = 1:3, counts = c(1, 2, 1))  
range(hd) # returns 1 3
```

sd,HistDat-method *Calculates the standard deviation of the observations in the histogram dataset*

Description

Calculates the standard deviation of the observations in the histogram dataset

Usage

```
## S4 method for signature 'HistDat'  
sd(x)
```

Arguments

| | |
|---|----------------------------------|
| x | An instance of the class HistDat |
|---|----------------------------------|

Value

A numeric of length 1, holding the standard deviation of all observations in the dataset

Examples

```
hd <- HistDat(vals = 1:3, counts = c(1, 2, 1))
sd(hd) # returns 0.8164966
```

sort,HistDat-method *This is a dummy method so that sort can be applied to HistDat entries. However it does nothing, because the values in a HistDat are sorted at the time of creation.*

Description

This is a dummy method so that sort can be applied to HistDat entries. However it does nothing, because the values in a HistDat are sorted at the time of creation.

Usage

```
## S4 method for signature 'HistDat'
sort(x, decreasing = F, ...)
```

Arguments

| | |
|------------|---|
| x | HistDat A HistDat instance |
| decreasing | If TRUE, this function will fail, as the observations are sorted in ascending order by default and this cannot be changed |
| ... | Additional arguments allowed for compatibility that will be ignored |

Details

An S3 and S4 generic is defined for this method, allowing compatibility with existing code that calls `base::sort()` instead of `[sort()]`, which is defined as an S4 generic in this package

Value

The same HistDat instance, completely unchanged

Examples

```
hd <- HistDat(vals = 1:3, counts = c(1, 2, 1))
sort(hd) # returns `hd` verbatim
```

sum,HistDat-method *Calculates the sum of all observations in the histogram dataset*

Description

Calculates the sum of all observations in the histogram dataset

Usage

```
## S4 method for signature 'HistDat'  
sum(x, ..., na.rm = FALSE)
```

Arguments

| | |
|-------|--|
| x | An instance of the class HistDat |
| ... | Additional arguments to pass to sum() |
| na.rm | Passed verbatim to base::sum() |

Value

A numeric of length 1, holding the sum of all values in the dataset

Functions

- sum,HistDat-method: The S4 version

Examples

```
hd <- HistDat(vals = 1:3, counts = c(1, 2, 1))  
sum(hd) # returns 8
```

var,HistDat-method *Calculates the variance of observations in the histogram dataset*

Description

Calculates the variance of observations in the histogram dataset

Usage

```
## S4 method for signature 'HistDat'  
var(x, y = NULL, na.rm = FALSE, use)
```

Arguments

| | |
|-------|---|
| x | An instance of the class HistDat |
| y | Provided for compatibility with <code>stats::var()</code> , but ignored |
| na.rm | Provided for compatibility with <code>stats::var()</code> , but ignored |
| use | Provided for compatibility with <code>stats::var()</code> , but ignored |

Value

A numeric of length 1, holding the variance of all observations in the dataset

Examples

```
hd <- HistDat(vals = 1:3, counts = c(1, 2, 1))
var(hd) # returns 0.6666667
```

[,HistDat-method *Index the histogram data*

Description

Index the histogram data

Usage

```
## S4 method for signature 'HistDat'
x[i, j, ..., drop = TRUE]
```

Arguments

| | |
|--------------|---|
| x | An instance of the class HistDat |
| i | A vector of indices to find in the sorted array of observations |
| j, drop, ... | Included for compatibility, but ignored |

Value

The observations that would be returned if you flattened the array and then indexed it

Examples

```
hd <- HistDat(vals = 1:3, counts = c(1, 2, 1))
hd[1] # returns 1
hd[2] # returns 2
hd[3] # returns 2
```

Index

[,HistDat-method, [3](#), [14](#)

as.ecdf, [3](#)
as.ecdf(), [3](#)
as.ecdf,HistDat-method, [3](#), [4](#)
as.vector,HistDat-method, [3](#), [4](#)

base::max(), [7](#)
base::mean(), [8](#)
base::min(), [10](#)
base::range(), [11](#)
base::sort(), [12](#)
base::sum(), [13](#)

c,HistDat-method, [3](#), [5](#)
c,HistDatCompatible-method
 (c,HistDat-method), [5](#)

ecdf(), [3](#)

HistDat, [2](#), [6](#)
HistDat(), [2](#)
HistDat-class, [6](#)
HistDat-package, [2](#)

length,HistDat-method, [2](#), [7](#)

max,HistDat-method, [2](#), [7](#)
mean,HistDat-method, [3](#), [8](#)
median,HistDat-method, [2](#), [9](#)
min,HistDat-method, [2](#), [9](#)

quantile,HistDat-method, [3](#), [10](#)

range,HistDat-method, [2](#), [11](#)

sd,HistDat-method, [3](#), [11](#)
sort,HistDat-method, [12](#)
stats::median(), [9](#)
stats::quantile(), [10](#)
stats::var(), [14](#)

sub,HistDat-method ([,HistDat-method),
 [14](#)
sum,HistDat-method, [3](#), [13](#)
var,HistDat-method, [3](#), [13](#)