Package 'binsmooth'

October 12, 2022

Type Package

Title Generate PDFs and CDFs from Binned Data

Version 0.2.2

Author David J. Hunter and McKalie Drown

Maintainer Dave Hunter <dhunter@westmont.edu>

Description Provides several methods for generating density functions based on binned data. Methods include step function, recursive subdivision, and optimized spline. Data are assumed to be nonnegative, the top bin is assumed to have no upper bound, but the bin widths need be equal. All PDF smoothing methods maintain the areas specified by the binned data. (Equivalently, all CDF smoothing methods interpolate the points specified by the binned data.) In practice, an estimate for the mean of the distribution should be supplied as an optional argument. Doing so greatly improves the reliability of statistics computed from the smoothed density functions. Includes methods for estimating the Gini coefficient, the Theil index, percentiles, and random deviates from a smoothed distribution. Among the three methods, the optimized spline (splinebins) is recommended for most purposes. The percentile and random-draw methods should be regarded as experimental, and these methods only support splinebins.

License MIT + file LICENSE

Imports stats, pracma, ineq, triangle

LazyData TRUE

NeedsCompilation no

RoxygenNote 6.1.1

Repository CRAN

Date/Publication 2020-03-11 21:40:03 UTC

R topics documented:

`county_bins` *ACS County Income Data, 2006-2010*

Description

Binned income data from 3,221 counties in the U.S. and Puerto Rico.

Usage

```
data("county_bins")
```

Format

A data frame with 51536 observations on the following 6 variables.

`fips` Number identifying the county

`households` Bin counts

`bin_min` Left endpoints of bins (US Dollars)

`bin_max` Right endpoints of bins

`county` County name

`state` State name

Source

U.S. Census Bureau, American Community Survey: https://www.census.gov/programs-surveys/acs/

See Also

`county_true`

Examples

```
data(county_bins)
data(county_true)
binedges <- county_bins$bin_max[county_bins$fips=="6083"]+0.5 # continuity correction
bincounts <- county_bins$households[county_bins$fips=="6083"]
smean <- county_true$mean_true[county_true$fips=="6083"]
plot(splinebins(binedges, bincounts, smean)$splinePDF, 0, 300000,
     n=500, main="Santa Barbara County")
plot(stepbins(binedges, bincounts, smean)$stepPDF, do.points=FALSE, col="red", add=TRUE)
```

`county_true` *ACS County Income Statistics, 2006-2010*

Description

Statistics computed from raw data on 3,221 counties in the U.S. and Puerto Rico.

Usage

```
data("county_true")
```

Format

A data frame with 3221 observations on the following 4 variables.

`fips` Number identifying the county

`mean_true` Sample mean

`median_true` Sample median

`gini_true` Gini coefficient

Source

U.S. Census Bureau, American Community Survey: https://www.census.gov/programs-surveys/acs/

See Also

county_bins

Examples

```
data(county_bins)
data(county_true)
binedges <- county_bins$bin_max[county_bins$fips=="6083"]+0.5 # continuity correction
bincounts <- county_bins$households[county_bins$fips=="6083"]
smean <- county_true$mean_true[county_true$fips=="6083"]
plot(stepbins(binedges, bincounts, smean)$stepPDF, do.points=FALSE,
     main="Santa Barbara County")
```

gini *Estimate the Gini coefficient*

Description

Estimates the Gini coefficient from a smoothed distribution.

Usage

```
gini(binFit)
```

Arguments

binFit	A list as returned by splinebins, stepbins, or rsubbins. (Alternatively, a list containing a PDF of non-negative support, its CDF, and an upper bound for the support of the PDF.)

Details

For distributions of non-negative support, the Gini coefficient can be computed from a cumulative distribution function $F(x)$ by the integral

$$G = 1 - \frac{1}{\mu}\int_0^\infty (1 - F(x))^2\, dx$$

where μ is the mean of the distribution.

Value

Returns the Gini coefficient G.

Author(s)

David J. Hunter and McKalie Drown

References

Paul T. von Hippel, David J. Hunter, McKalie Drown. *Better Estimates from Binned Income Data: Interpolated CDFs and Mean-Matching*, Sociological Science, November 15, 2017. https://www.sociologicalscience.com/articles-v4-26-641/

Examples

```
# 2005 ACS data from Cook County, Illinois
binedges <- c(10000,15000,20000,25000,30000,35000,40000,45000,
              50000,60000,75000,100000,125000,150000,200000,NA)
bincounts <- c(157532,97369,102673,100888,90835,94191,87688,90481,
               79816,153581,195430,240948,155139,94527,92166,103217)
stepfit <- stepbins(binedges, bincounts, 76091)
```

```
splinefit <- splinebins(binedges, bincounts, 76091)
gini(stepfit)
gini(splinefit) # More accurate
```

---

rsubbins *Recursive subdivision PDF and CDF fitted to binned data*

Description

Creates a PDF and CDF based on a set of binned data, using recursive subdivision on a step function.

Usage

```
rsubbins(bEdges, bCounts, m=NULL, eps1 = 0.25, eps2 = 0.75, depth = 3,
        tailShape = c("onebin", "pareto", "exponential"),
        nTail=16, numIterations=20, pIndex=1.160964, tbRatio=0.8)
```

Arguments

bEdges	A vector $e_1, e_2, \ldots, e_n$ giving the right endpoints of each bin. The value in e_n is ignored and assumed to be Inf or NA, indicating that the top bin is unbounded. The edges determine n bins on the intervals $e_{i-1} \leq x \leq e_i$, where e_0 is assumed to be 0.
bCounts	A vector $c_1, c_2, \ldots, c_n$ giving the counts for each bin (i.e., the number of data elements in each bin). Assumed to be nonnegative.
m	An estimate for the mean of the distribution. If no value is supplied, the mean will be estimated by (temporarily) setting e_n equal to $2e_{n-1}$, and a warning message will be generated.
eps1	Parameter controlling how far the edges of the subdivided bins are shifted. Must be between 0 and 0.5.
eps2	Parameter controlling how wide the middle subdivsion of each bin should be. Must be between 0 and 1.
depth	Number of times to subdivide the bins.
tailShape	Must be one of "onebin", "pareto", or "exponential".
nTail	The number of bins to use to form the initial tail, before recursive subdivision. Ignored if tailShape equals "onebin".
numIterations	The number of iterations to optimize the tail to fit the mean. Ignored if tailShape equals "onebin".
pIndex	The Pareto index for the shape of the tail. Defaults to $\ln(5)/\ln(4)$. Ignored unless tailShape equals "pareto".
tbRatio	The decay ratio for the tail bins. Ignored unless tailShape equals "exponential".

Details

First, a step function PDF is created, as described in stepbins. The bins of the resulting PDF are then recursively subdivided and shifted in a manner that preserves the area of the original bins, resulting in a step function with finer bins.

The methods stepbins and rsubbins are included in this package mainly for the purpose of comparison. For most use cases, splinebins will produce more accurate smoothing results.

Value

Returns a list with the following components.

rsubPDF	A stepfun function giving the fitted PDF.
rsubCDF	A piecewise-linear approxfun function giving the CDF.
E	The right-hand endpoint of the support of the PDF.
shrinkFactor	If the supplied estimate for the mean is too small to be fitted with a step function, the bins edges will be scaled by shrinkFactor, which will be chosen less than (and close to) 1.

Author(s)

David J. Hunter and McKalie Drown

References

Paul T. von Hippel, David J. Hunter, McKalie Drown. *Better Estimates from Binned Income Data: Interpolated CDFs and Mean-Matching*, Sociological Science, November 15, 2017. https://www.sociologicalscience.com/articles-v4-26-641/

See Also

stepbins

Examples

```
# 2005 ACS data from Cook County, Illinois
binedges <- c(10000,15000,20000,25000,30000,35000,40000,45000,
              50000,60000,75000,100000,125000,150000,200000,NA)
bincounts <- c(157532,97369,102673,100888,90835,94191,87688,90481,
               79816,153581,195430,240948,155139,94527,92166,103217)
rsb <- rsubbins(binedges, bincounts, 76091, tailShape="pareto")

plot(rsb$rsubPDF, do.points=FALSE)
plot(rsb$rsubCDF, 0, rsb$E)

library(pracma)
integral(rsb$rsubPDF, 0, rsb$E)
integral(function(x){1-rsb$rsubCDF(x)}, 0, rsb$E) #mean is approximated
```

sb_percentiles *Estimate percentiles from splinebins*

Description

Estimates percentiles of a smoothed distribution obtained using splinebins.

Usage

```
sb_percentiles(splinebinFit, p = seq(0,100,25))
```

Arguments

splinebinFit	A list as returned by splinebins.
p	A vector of percentages in the range $0 \leq p \leq 100$.

Details

The approximate inverse of the CDF calculated by splinebins is used to approximate percentiles of the smoothed distribution.

Value

A vector of percentiles. Returns NA if an inaccurate fit is detected, as indicated by fitWarn.

Author(s)

David J. Hunter and McKalie Drown

References

Paul T. von Hippel, David J. Hunter, McKalie Drown. *Better Estimates from Binned Income Data: Interpolated CDFs and Mean-Matching*, Sociological Science, November 15, 2017. https://www.sociologicalscience.com/articles-v4-26-641/

Examples

```
# 2005 ACS data from Cook County, Illinois
binedges <- c(10000,15000,20000,25000,30000,35000,40000,45000,
              50000,60000,75000,100000,125000,150000,200000,NA)
bincounts <- c(157532,97369,102673,100888,90835,94191,87688,90481,
               79816,153581,195430,240948,155139,94527,92166,103217)
splinefit <- splinebins(binedges, bincounts, 76091)
sb_percentiles(splinefit)
sb_percentiles(splinefit, c(27, 32, 93))
```

sb_sample *Random sample from splinebins distribution*

Description

Draw a random sample of points from a smoothed distribution obtained using splinebins.

Usage

```
sb_sample(splinebinFit, n = 1)
```

Arguments

splinebinFit	A list as returned by splinebins.
n	A positive integer giving the sample size.

Details

The approximate inverse of the CDF calculated by splinebins is used to generate random values of the smoothed distribution.

Value

A vector of random deviates. Returns NA if an inaccurate fit is detected, as indicated by fitWarn.

Author(s)

David J. Hunter and McKalie Drown

References

Paul T. von Hippel, David J. Hunter, McKalie Drown. *Better Estimates from Binned Income Data: Interpolated CDFs and Mean-Matching*, Sociological Science, November 15, 2017. https://www.sociologicalscience.com/articles-v4-26-641/

Examples

```
# 2005 ACS data from Cook County, Illinois
binedges <- c(10000,15000,20000,25000,30000,35000,40000,45000,
              50000,60000,75000,100000,125000,150000,200000,NA)
bincounts <- c(157532,97369,102673,100888,90835,94191,87688,90481,
               79816,153581,195430,240948,155139,94527,92166,103217)
splinefit <- splinebins(binedges, bincounts, 76091)
sb_sample(splinefit, 5)
hist(sb_sample(splinefit, 3000))
```

simcounty *Simulate data to mimic* county_bins *and* county_true

Description

Samples from a selection of distributions (Gamma, Lognormal, Weibull, Triangle) to simulate income data in the format used in the American Community Survey data (county_bins and county_true).

Usage

```
simcounty(numCounties, minPop = 1000, maxPop = 100000,
       bin_minimums = c(0, 10000, 15000, 20000, 25000, 30000, 35000, 40000, 45000,
                        50000, 60000, 75000, 100000, 125000, 150000, 200000))
```

Arguments

numCounties	The number of counties to simulate data for
minPop	Minimum population to sample (default = 1000)
maxPop	Maximum population to sample (default = 100000)
bin_minimums	Bin edges. Defaults to the edges used in the Census data.

Details

The county names will tell which distributions were sampled to simulate each county.

Value

Returns a list of two data frames:

county_bins	Simulated binned income data
county_true	Statistics computed from the raw data

Author(s)

David J. Hunter and McKalie Drown

References

Paul T. von Hippel, David J. Hunter, McKalie Drown. *Better Estimates from Binned Income Data: Interpolated CDFs and Mean-Matching*, Sociological Science, November 15, 2017. https://www.sociologicalscience.com/articles-v4-26-641/

See Also

county_bins, county_true

Examples

```
l1 <- simcounty(5)
cb <- l1$county_bins
ct <- l1$county_true
sbl <- splinebins(cb$bin_max[cb$fips==103], cb$households[cb$fips==103],
                  ct$mean_true[ct$fips==103])
stl <- stepbins(cb$bin_max[cb$fips==105], cb$households[cb$fips==105],
                ct$mean_true[ct$fips==105])
plot(sbl$splinePDF, 0, 300000, n=500)
plot(stl$stepPDF, do.points=FALSE, main=cb$county[cb$fips==105][1])

## Simulate one county and estimate gini and theil from binned data
l2 <- simcounty(1)
binedges <- l2$county_bins$bin_max + 0.5 # continuity correction
bincounts <- l2$county_bins$households
splinefit <- splinebins(binedges, bincounts, l2$county_true$mean_true)
gini(splinefit)
theil(splinefit)
l2$county_true
```

`splinebins` *Optimized spline PDF and CDF fitted to binned data*

Description

Creates a smooth cubic spline CDF and piecewise-quadratic PDF based on a set of binned data (edges and counts).

Usage

```
splinebins(bEdges, bCounts, m = NULL,
           numIterations = 16, monoMethod = c("hyman", "monoH.FC"))
```

Arguments

`bEdges`	A vector $e_1, e_2, \ldots, e_n$ giving the right endpoints of each bin. The value in e_n is ignored and assumed to be `Inf` or `NA`, indicating that the top bin is unbounded. The edges determine n bins on the intervals $e_{i-1} \leq x \leq e_i$, where e_0 is assumed to be 0.
`bCounts`	A vector $c_1, c_2, \ldots, c_n$ giving the counts for each bin (i.e., the number of data elements in each bin). Assumed to be nonnegative.
`m`	An estimate for the mean of the distribution. If no value is supplied, the mean will be estimated by (temporarily) setting e_n equal to $2e_{n-1}$, and a warning message will be generated.
`numIterations`	The number of iterations performed by a binary search that optimizes the CDF to fit the mean.
`monoMethod`	The method for constructing a monotone spline. Must be one of `"hyman"` or `"monoH.FC"`. The former choice tends to integrate faster and produce smoother density functions. See `splinefun` for more details.

Details

Fits a monotone cubic spline to the points specified by the binned data to produce a smooth cumulative distribution function. The PDF is then obtained by differentiating, so it will be piecewise quadratic and preserve the area of each bin.

Value

Returns a list with the following components.

`splinePDF`	A piecewise-quadratic function giving the fitted PDF.
`splineCDF`	A piecewise-cubic function giving the CDF.
`E`	The right-hand endpoint of the support of the PDF.
`shrinkFactor`	If the supplied estimate for the mean is too small to be fitted with our method, the bins edges will be scaled by `shrinkFactor`, which will be chosen less than (and close to) 1.
`splineInvCDF`	An approximate inverse of `splineCDF`.
`fitWarn`	Flag set to `TRUE` if the fitted median falls in the wrong bin.

Author(s)

David J. Hunter and McKalie Drown

References

Paul T. von Hippel, David J. Hunter, McKalie Drown. *Better Estimates from Binned Income Data: Interpolated CDFs and Mean-Matching*, Sociological Science, November 15, 2017. https://www.sociologicalscience.com/articles-v4-26-641/

Examples

```
# 2005 ACS data from Cook County, Illinois
binedges <- c(10000,15000,20000,25000,30000,35000,40000,45000,
              50000,60000,75000,100000,125000,150000,200000,NA)
bincounts <- c(157532,97369,102673,100888,90835,94191,87688,90481,
               79816,153581,195430,240948,155139,94527,92166,103217)
sb <- stepbins(binedges, bincounts, 76091)
splb <- splinebins(binedges, bincounts, 76091)

plot(splb$splinePDF, 0, 300000, n=500)
plot(sb$stepPDF, do.points=FALSE, col="gray", add=TRUE)
# notice that the curve preserves bin area

library(pracma)
integral(splb$splinePDF, 0, splb$E)
integral(function(x){1-splb$splineCDF(x)}, 0, splb$E) # should be the mean
splb <- splinebins(binedges, bincounts, 76091, numIterations=20)
integral(function(x){1-splb$splineCDF(x)}, 0, splb$E) # closer to given mean
```

`stats_from_distribution` *Estimate various statistics*

Description

Estimates the mean, variance, standard deviation, Gini coefficient, and Theil index from a smoothed distribution.

Usage

```
stats_from_distribution(binFit)
```

Arguments

`binFit`	A list as returned by `splinebins`, `stepbins`, or `rsubbins`. (Alternatively, a list containing a PDF of non-negative support, its CDF, and an upper bound for the support of the PDF.)

Details

The mean and variance are calculated from the CDF. For details on the other statistics, see `gini` and `theil`.

Value

A vector of five statistics.

Author(s)

David J. Hunter and McKalie Drown

References

Paul T. von Hippel, David J. Hunter, McKalie Drown. *Better Estimates from Binned Income Data: Interpolated CDFs and Mean-Matching*, Sociological Science, November 15, 2017. https://www.sociologicalscience.com/articles-v4-26-641/

Examples

```
# 2005 ACS data from Cook County, Illinois
binedges <- c(10000,15000,20000,25000,30000,35000,40000,45000,
              50000,60000,75000,100000,125000,150000,200000,NA)
bincounts <- c(157532,97369,102673,100888,90835,94191,87688,90481,
               79816,153581,195430,240948,155139,94527,92166,103217)
stepfit <- stepbins(binedges, bincounts, 76091)
splinefit <- splinebins(binedges, bincounts, 76091)
stats_from_distribution(stepfit)
stats_from_distribution(splinefit) # More accurate
```

stepbins *Step function PDF and CDF fitted to binned data*

Description

Creates a step function PDF and CDF based on a set of binned data (edges and counts).

Usage

```
stepbins(bEdges, bCounts, m = NULL,
         tailShape = c("onebin", "pareto", "exponential"),
         nTail = 16, numIterations = 20, pIndex = 1.160964, tbRatio = 0.8)
```

Arguments

bEdges	A vector $e_1, e_2, \ldots, e_n$ giving the right endpoints of each bin. The value in e_n is ignored and assumed to be Inf or NA, indicating that the top bin is unbounded. The edges determine n bins on the intervals $e_{i-1} \leq x \leq e_i$, where e_0 is assumed to be 0.
bCounts	A vector $c_1, c_2, \ldots, c_n$ giving the counts for each bin (i.e., the number of data elements in each bin). Assumed to be nonnegative.
m	An estimate for the mean of the distribution. If no value is supplied, the mean will be estimated by (temporarily) setting e_n equal to $2e_{n-1}$, and a warning message will be generated.
tailShape	Must be one of "onebin", "pareto", or "exponential".
nTail	The number of bins to use to form the tail. Ignored if tailShape equals "onebin".
numIterations	The number of iterations to optimize the tail to fit the mean. Ignored if tailShape equals "onebin".
pIndex	The Pareto index for the shape of the tail. Defaults to $\ln(5)/\ln(4)$. Ignored unless tailShape equals "pareto".
tbRatio	The decay ratio for the tail bins. Ignored unless tailShape equals "exponential".

Details

We assume that the left endpoint of the first bin is 0 and that the top bin is unbounded. Options exist to replace the top bin with a single bin or a sequence of bins in the shape of a Pareto or exponential tail. The density functions will fit a supplied estimate for the population mean, if supplied.

The methods stepbins and rsubbins are included in this package mainly for the purpose of comparison. For most use cases, splinebins will produce more accurate smoothing results.

Value

Returns a list with the following components.

stepPDF	A stepfun function giving the fitted PDF.
stepCDF	A piecewise-linear approxfun function giving the CDF.
E	The right-hand endpoint of the support of the PDF.
shrinkFactor	If the supplied estimate for the mean is too small to be fitted with a step function, the bins edges will be scaled by shrinkFactor, which will be chosen less than (and close to) 1.

Author(s)

David J. Hunter and McKalie Drown

References

Paul T. von Hippel, David J. Hunter, McKalie Drown. *Better Estimates from Binned Income Data: Interpolated CDFs and Mean-Matching*, Sociological Science, November 15, 2017. https://www.sociologicalscience.com/articles-v4-26-641/

Examples

```
# 2005 ACS data from Cook County, Illinois
binedges <- c(10000,15000,20000,25000,30000,35000,40000,45000,
              50000,60000,75000,100000,125000,150000,200000,NA)
bincounts <- c(157532,97369,102673,100888,90835,94191,87688,90481,
               79816,153581,195430,240948,155139,94527,92166,103217)
sb <- stepbins(binedges, bincounts, 76091)
sbpt <- stepbins(binedges, bincounts, 76091, tailShape="pareto")

plot(sb$stepPDF)
plot(sbpt$stepPDF, do.points=FALSE)
plot(sb$stepCDF, 0, sb$E+100000)

library(pracma)
integral(sb$stepPDF, 0, sb$E) # should be approximately 1
integral(function(x){1-sb$stepCDF(x)}, 0, sb$E) # should be the mean
```

---

theil *Estimate the Theil index*

Description

Estimates the Theil index from a smoothed distribution.

Usage

```
theil(binFit)
```

Arguments

binFit	A list as returned by splinebins, stepbins, or rsubbins. (Alternatively, a list containing a PDF of non-negative support, its CDF, and an upper bound for the support of the PDF.)

Details

For distributions of non-negative support, the Theil index can be computed from a probability density function $f(x)$ by the integral

$$T = \int_0^\infty f(x) \frac{x}{\mu} \ln\left(\frac{x}{\mu}\right) dx$$

where μ is the mean of the distribution.

Value

Returns the Theil index T.

Author(s)

David J. Hunter and McKalie Drown

References

Paul T. von Hippel, David J. Hunter, McKalie Drown. *Better Estimates from Binned Income Data: Interpolated CDFs and Mean-Matching*, Sociological Science, November 15, 2017. https://www.sociologicalscience.com/articles-v4-26-641/

Examples

```
# 2005 ACS data from Cook County, Illinois
binedges <- c(10000,15000,20000,25000,30000,35000,40000,45000,
              50000,60000,75000,100000,125000,150000,200000,NA)
bincounts <- c(157532,97369,102673,100888,90835,94191,87688,90481,
               79816,153581,195430,240948,155139,94527,92166,103217)
stepfit <- stepbins(binedges, bincounts, 76091)
splinefit <- splinebins(binedges, bincounts, 76091)
theil(stepfit)
theil(splinefit) # More accurate
```

Index