Package: histogram (via r-universe)

October 24, 2024

Type Package

Title Construction of Regular and Irregular Histograms with Different Options for Automatic Choice of Bins

Version 0.0-25

Author Thoralf Mildenberger [aut, cre], Yves Rozenholc [aut], David Zasada [aut]

Maintainer Thoralf Mildenberger <mild@zhaw.ch>

Description Automatic construction of regular and irregular histograms as described in Rozenholc/Mildenberger/Gather (2010).

License GPL (>= 2)

LazyLoad yes

ByteCompile yes

Repository https://thmild.r-universe.dev

RemoteUrl https://github.com/thmild/histogram

RemoteRef HEAD

RemoteSha 8c390edf9bfc3596551f56ea1a9d39ec4a5edf97

Contents

`histogram` *histogram with automatic choice of bins*

Description

Construction of regular and irregular histograms with different options for choosing the number and widths of the bins. By default, both a regular and an irregular histogram using a data-dependent penalty as described in detail in Rozenholc/Mildenberger/Gather (2009) are constructed. The final estimate is the one with the larger penalized likelihood.

Usage

```
histogram(y, type = "combined", grid = "data",
breaks = NULL, penalty = "default",
greedy = TRUE, right=TRUE, freq=FALSE, control = list(),
verbose = TRUE, plot = TRUE, ...)
```

Arguments

`y` a vector of values for which the histogram is desired.

`type` use "irregular" for an irregular and "regular" for a regular histogram. If type="combined" (default value) both a regular and an irregular histogram are computed and the one with the larger penalized likelihood is chosen, see details below.

`grid` if type="irregular", grid chooses the set of possible partitions of the data range. The default value "data" gives a set of partitions constructed from the data points, "regular" uses a fine regular grid of points as possible break points. A regular quantile grid can be chosen using "quantiles". Has no effect for regular histograms.

`breaks` controls the maximum number of bins allowed in a regular histogram, or the size of the finest grid in an irregular histogram when grid is set to "regular" or "quantiles". Usually not needed since the maximum bin number and the size of the finest grid are calculated by a formula depending on the sample size n; the defaults for this can be changed using the parameters g1, g2 and g3 in the control argument. Also see maxbin in the control argument which gives an absolute upper bound bound on the number of bins if type="regular".

`penalty` controls which penalty is used. See description of penalties below.

`greedy` logical; if TRUE and type="irregular", a subgrid of the finest grid is constructed by a greedy step to make the search procedure feasible. Has no effect for regular histograms.

`right` logical; if TRUE, the histograms cells are right-closed (left open) intervals.

`freq` logical; if TRUE, the y-axis gives counts in case of a regular histogram, otherwise density values are given. For irregular histograms, the y-axis always shows the density. Unlike hist(), defaults to FALSE.

`control` list of additional control parameters. Meaning and default values depend on settings of type, penalty and grid. See below.

`verbose` logical; if TRUE (default), some information is given during histogram construction and the resulting histogram object is printed.

`plot` logical. If TRUE (default), the histogram is plotted.

`...` further arguments and graphical parameters passed to hist().

Details

The histogram procedure produces a histogram, i.e. a piecewise constant density estimate from a univariate real-valued sample stored in a vector y. Let n denote the length of y. The range of the data is partitioned into D intervals - called bins - and the density estimate on the i-th bin is given by $N_i/(n * w_i)$ where N_i is the number of observations in the i-th bin and w_i is its width. The histogram thus defined is the maximum likelihood estimate among all densities that are piecewise constant w.r.t. this partition. The arguments of histogram given above determine the way the partition is chosen. In a regular histogram, the partition consists of D bins of the same widths, and the histogram is determined by the choice of D. Strategies based on different criteria can be chosen using the penalty option. The maximum number of bins can be controlled by either the breaks argument or the entries g1, g2 and g3 in the control argument.

An irregular histogram allows for bins of different widths. In this case, not only the number D of bins but also the breakpoints between the bins must be chosen. The set of allowed breakpoints is given by the finest partition selected using the grid argument. At the moment a finest regular grid is supported (grid="regular") as well as grids with possible breakpoints either equal to the observations or between the observations (grid="data" and between in the control argument set to FALSE or TRUE, respectively). Setting grid="quantiles" gives a grid based on regular sample quantiles. If the breaks argument is NULL,

$$G(n) = g1 * n^{g2} * (\log(n))^{g3}$$

controls the grid in the following way: the smallest allowed bin width in a "data" grid is $1/G(n)$ times the sample range, while for grid="regular" and grid="quantiles" the finest grid has floor(G(n)) bins. The parameters g1, g2 and g3 can be changed by modifying the corresponding components in the control argument. If breaks is a positive number, its integer part is used instead of G(n). Different strategies for selection of D and the bin boundaries can be chosen using the penalty option.

To reduce calculation time for irregular histograms, a subset of the breakpoints of the finest grid can be chosen by starting from a one-bin histogram and then subsequently finding the split of an existing bin that leads to the largest increase in the loglikelihood. The full optimization is then performed only over all partitions with endpoints from the subset thus constructed. This is achieved by setting greedy=TRUE. To reduce calculation time for regular histograms, the maxbin parameter in the control argument gives an upper bound for the number of bins. The default value is 1000.

Using type="combined" (the default value), both a regular and an irregular histogram are constructed using a penalized likelihood approach and the one with the larger penalized likelihood is chosen. In this case, the regular histogram is always constructed using the br penalty. The penalty parameter and all other options control the construction of the irregular histogram. penalty must be equal to "penA", "penB" or "penR", since otherwise comparison of penalized likelihood values would not be meaningful.

Value

an object of class "histogram" which is a list with the same components as in the hist command.

Penalties

Most settings of penalty lead to a penalized maximum likelihood histogram. For a sample of size n and a partition J that divides the sample range into D bins, define N_i as the number of observations in the i-th bin, $i = 1, ..., D$ and w_i as the width of the the i-th bin, $i = 1, ..., D$. In this section, the index in sums and products is always $i = 1, \ldots, D$. For any partition J, and a fixed sample, the penalized loglikelihood is defined as

$$\sum N_i * \log(N_i/(n * w_i)) - pen(J).$$

The possible penalties are:

penA Penalty given in formula (5) in in Rozenholc, Mildenberger and Gather (2009):

$$pen(J) = c \log \binom{n-1}{D-1} + \alpha(D-1) + ck\log(D) + 2\sqrt{c\alpha(D-1)(\log \binom{n-1}{D-1} + k\log D)},$$

where the default values are $c = 1$, $\alpha = 0.5$ and $k = 2$. These can be changed using the c, alpha and k components of control.

penB Simplified version of formula (5) in Rozenholc, Mildenberger and Gather (2009):

$$pen(J) = c \log \binom{n-1}{D-1} + \alpha(D-1) + \log^{2.5} D,$$

where the default values are $c = 1$ and $\alpha = 1$. These can be changed using the c and alpha components of control. Default penalty for irregular and combined histograms.

penR Data-dependent penalty as given in formula (6) in Rozenholc, Mildenberger and Gather (2009):

$$pen(J) = c \log \binom{n-1}{D-1} + (\alpha/n)\sum N_i/w_i + \log^{2.5} D,$$

where the default values are $c = 1$ and $\alpha = 0.5$. These can be changed using the c and alpha components of control.

aic Akaike's Information Criterion (AIC). Defined by $pen(J) = \alpha * D$, where α is 1 by default and may be changed using the alpha parameter in the control argument.

bic Bayesian Information Criterion (BIC). Defined by $pen(J) = \alpha * \log(n) * D$, where α is 0.5 by default and may be changed using the alpha parameter in the control argument.

nml Normalized Maximum Likelihood. Formula is given in Davies, Gather, Nordman, Weinert (2009). Only available for regular histograms.

br Improved version of AIC for regular histograms as given in Birge and Rozenholc (2006). Defined as $pen(J) = D + \log^{2.5}(D)$. Default penalty for regular histograms, not available for irregular histograms.

Some settings of penalty do not lead to maximization of a penalized likelihood but optimzation of different measures. These are:

cv Leave-p-out crossvalidation. Different variants can be chosen by setting the cvformula and p components in the control argument. cvformula=1 and cvformula=2 are available both for regular and irregular histograms. These are different versions of leave-p-out L2-crossvalidation, where choice of a partition is achieved by minimizing

$$2\sum N_i/w_i - (n+1)\sum N_i^2/(n * w_i),$$

or

$$2(n-p)\sum N_i/w_i - (n-p+1)\sum N_i^2/w_i$$

respectively, see formulas (11) and (12) in Celisse and Robin (2008). Since formula 1 does not depend on p, if the control parameter p is set to a value greater than 1 cvformula is set to

2. Kullback-Leibler crossvalidation can be performed by setting `cvformula=3`. This is only available if $p = 1$ and `type="regular"`. The number of bins chosen is the maximizer of

$$\sum N_i \log(N_i - 1) + n \log(D),$$

see remark 2.3 in Hall and Hannan (1988).

`sc` Stochastic Complexity criterion, only available for regular histograms. Number of bins is chosen by maximizing

$$\prod N_i! D^n (D-1)!/(D+n-1)!,$$

see formula (2.3) in Hall and Hannan (1988).

`mdl` Minimum Description Length criterion, only available for regular histograms. Number of bins is chosen by maximizing

$$\sum (N_i - 0.5)\log(N_i - 0.5) - (n - 0.5D)\log(n - 0.5D) + n \log D - 0.5D \log n,$$

see formula (2.5) in Hall and Hannan (1988).

Control

The control parameter is a list with different components that affect the construction of the histogram. Meaning and default values depend on setting of the other parameters.

`alpha` Coefficient of the number of bins in penalties `penA`, `penB`, `aic`, `bic`. Coefficient of the data-driven part in the `penR` penalty.

`between` logical; if `TRUE` and `grid="data"`, possible bin ends are put between the observations, if `FALSE` (default) they are placed at the observations

`c` Controls the weight of the penalty component that corrects for the multiplicity of partitions with the same number of bins in irregular histograms; only used in penalties `penA`, `penB` and `penR`.

`cvformula` determines the type of crossvalidation to be performed. Can take the values 1, 2 and 3. 1 and 2 correspond to different versions of L2 crossvalidation, while `cvformula=3` performs Kullback-Leibler crossvalidation, which is at the moment only available for regular histograms. Note that `cvformula=3` automatically forces every bin to include at least 2 observations. If `p` is set to a value greater than 1, `cvformula=2` is used automatically.

`g1` The parameters `g1`, `g2` and `g3` control the maximum number of bins in a regular histogram as well as the bin width and/or number for irregular histograms. Define

$$G(n) = g1 * n^{g2} * (\log(n))^{g3}.$$

The maximum number of bins allowed in a regular histogram is given by `floor(G(n))`, the finest grid in an irregular histogram with `grid="regular"` is obtained by dividing the sample range into `floor(G(n))` equisized bins, and if `grid="quantiles"`, the finest grid is obtained by dividing the interval $[0, 1]$ into equisized intervals and using the sample quantiles corresponding to the boundary points. For an irregular histogram with `grid="data"`, a mimimum allowed bin size of $1/G(n)$ is enforced. This can be disabled by setting `g3` to `Inf`, causing $1/G(n)$ to be zero. Default settings are `g1=1` and `g2=1` for all grids. Default values for `g3` are `-1` for `grid="regular"` and `grid="quantiles"` and `Inf` for `grid="data"`. Also see `maxbin`.

`g2` see `g1`.

g3 see g1.

k Tuning parameter that only has an effect if penalty="penA". Default value is 2.

maxbin Gives an absolute upper bound on the number of bins in order to keep the calculations feasible for large data sets. If the number of bins chosen via breaks or g1, g2 and g3 exceeds maxbin, maxbin in used as the maximum number of bins. Only has an effect for regular histograms. Defaults to 1000.

p Controls the number p of data points left out in the crossvalidation. Can take integer values between 1 (default) and n-1. If a value greater than 1 is chosen, cvformula is automatically set to 2 since crossvalidation formula 1 does not depend on p and Kullback-Leibler crossvalidation is only supported for p=1.

quanttype Determines the way the quantiles are calculated if grid="quantiles". Corresponds to the type argument in quantile, whose default 7 is also the default here.

Author(s)

Thoralf Mildenberger, Yves Rozenholc, David Zasada.

References

Birg?, L. and Rozenholc, Y. (2006). How many bins should be put in a regular histogram? ESAIM: Probability and Statistics, 10, 24-45.

Celisse, A. and Robin, S. (2008). Nonparametric density estimation by exact leave-p-out cross-validation. Computational Statistics and Data Analysis 52, 2350-2368.

Davies, P. L., Gather, U., Nordman, D. J., and Weinert, H. (2009): A comparison of automatic histogram constructions. ESAIM: Probability and Statistics, 13, 181-196.

Hall, P. and Hannan, E. J. (1988). On stochastic complexity and nonparametric density estimation. Biometrika 75, 705-714.

Rozenholc, Y, Mildenberger, T. and Gather, U. (2009). Combining regular and irregular histograms by penalized likelihood. Discussion Paper 31/2009, SFB 823, TU Dortmund. https://eldorado.tu-dortmund.de/handle/2003/26529

Rozenholc, Y., Mildenberger, T., Gather, U. (2010). Combining regular and irregular histograms by penalized likelihood. Computational Statistics and Data Analysis 54, 3313-3323.

See Also

hist

Examples

```
## draw a histogram from a standard normal sample

y<-rnorm(100)
histogram(y)

## draw a histogram from a standard exponential sample

y<-rexp(1500)
```

```
histogram(y)

## draw a histogram from a normal mixture

n<-sum(sample(c(0,1),1000,replace=TRUE))
y<-c(rnorm(n,mean=5,sd=0.1),rnorm(1000-n))
histogram(y)

## the same using a regular histogram with Kullback-Leibler CV

n<-sum(sample(c(0,1),1000,replace=TRUE))
y<-c(rnorm(n,mean=5,sd=0.1),rnorm(1000-n))
histogram(y,type="regular",penalty="cv",control=list(cvformula=3))
```

Index