

# Package ‘labeling’

May 8, 2026

**Type** Package

**Title** Axis Labeling

**Version** 0.4.3

**Date** 2023-08-29

**Author** Justin Talbot,

**Maintainer** Nuno Sempere <nuno.sempere1h@gmail.com>

**Description** Functions which provide a range of axis labeling algorithms.

**License** MIT + file LICENSE | Unlimited

**Collate** 'labeling.R'

**NeedsCompilation** no

**Imports** stats, graphics

**Repository** CRAN

**Date/Publication** 2023-08-29 22:20:02 UTC

## Contents

labeling-package . . . . .	2
extended . . . . .	3
extended.figures . . . . .	4
gnuplot . . . . .	4
heckbert . . . . .	5
matplotlib . . . . .	6
nelder . . . . .	6
rpretty . . . . .	7
sparks . . . . .	8
thayer . . . . .	9
wilkinson . . . . .	9

<b>Index</b>	<b>11</b>
--------------	-----------

labeling-package      *Axis labeling*

---

## Description

Functions for positioning tick labels on axes

## Details

Package:    labeling  
Type:        Package  
Version:    0.4.3  
Date:        2023-08-29  
License:    Unlimited  
LazyLoad:   yes

Implements a number of axis labeling schemes, including those compared in An Extension of Wilkinson's Algorithm for Positioning Tick Labels on Axes by Talbot, Lin, and Hanrahan, InfoVis 2010.

## Author(s)

Justin Talbot <justintalbot@gmail.com>

## References

Heckbert, P. S. (1990) Nice numbers for graph labels, Graphics Gems I, Academic Press Professional, Inc. Wilkinson, L. (2005) The Grammar of Graphics, Springer-Verlag New York, Inc. Talbot, J., Lin, S., Hanrahan, P. (2010) An Extension of Wilkinson's Algorithm for Positioning Tick Labels on Axes, InfoVis 2010.

## See Also

[extended](#), [wilkinson](#), [heckbert](#), [rpretty](#), [gnuplot](#), [matplotlib](#), [nelder](#), [sparks](#), [thayer](#), [pretty](#)

## Examples

```
heckbert(8.1, 14.1, 4) # 5 10 15
wilkinson(8.1, 14.1, 4) # 8 9 10 11 12 13 14 15
extended(8.1, 14.1, 4) # 8 10 12 14
# When plotting, extend the plot range to include the labeling
# Should probably have a helper function to make this easier
data(iris)
x <- iris$Sepal.Width
y <- iris$Sepal.Length
```

```
x1 <- extended(min(x), max(x), 6)
y1 <- extended(min(y), max(y), 6)
plot(x, y,
      xlim=c(min(x,x1),max(x,x1)),
      ylim=c(min(y,y1),max(y,y1)),
      axes=FALSE, main="Extended labeling")
axis(1, at=x1)
axis(2, at=y1)
```

---

extended	<i>An Extension of Wilkinson's Algorithm for Position Tick Labels on Axes</i>
----------	---

---

### Description

extended is an enhanced version of Wilkinson's optimization-based axis labeling approach. It is described in detail in our paper. See the references.

### Usage

```
extended(dmin, dmax, m, Q = c(1, 5, 2, 2.5, 4, 3),
         only.loose = FALSE, w = c(0.25, 0.2, 0.5, 0.05))
```

### Arguments

dmin	minimum of the data range
dmax	maximum of the data range
m	number of axis labels
Q	set of nice numbers
only.loose	if true, the extreme labels will be outside the data range
w	weights applied to the four optimization components (simplicity, coverage, density, and legibility)

### Value

vector of axis label locations

### Author(s)

Justin Talbot <justintalbot@gmail.com>

### References

Talbot, J., Lin, S., Hanrahan, P. (2010) An Extension of Wilkinson's Algorithm for Positioning Tick Labels on Axes, InfoVis 2010.

extended.figures      *Generate figures from An Extension of Wilkinson's Algorithm for Position Tick Labels on Axes*

---

**Description**

Generates Figures 2 and 3 from our paper.

**Usage**

```
extended.figures(samples = 100)
```

**Arguments**

samples      number of samples to use (in the paper we used 10000, but that takes awhile to run).

**Value**

produces plots as a side effect

**Author(s)**

Justin Talbot <justintalbot@gmail.com>

**References**

Talbot, J., Lin, S., Hanrahan, P. (2010) An Extension of Wilkinson's Algorithm for Positioning Tick Labels on Axes, InfoVis 2010.

---

gnuplot      *gnuplot's labeling algorithm*

---

**Description**

gnuplot's labeling algorithm

**Usage**

```
gnuplot(dmin, dmax, m)
```

**Arguments**

dmin      minimum of the data range  
dmax      maximum of the data range  
m      number of axis labels

**Value**

vector of axis label locations

**Author(s)**

Justin Talbot <justintalbot@gmail.com>

**References**

<http://www.gnuplot.info/>

---

heckbert

*Heckbert's labeling algorithm*

---

**Description**

Heckbert's labeling algorithm

**Usage**

```
heckbert(dmin, dmax, m)
```

**Arguments**

dmin	minimum of the data range
dmax	maximum of the data range
m	number of axis labels

**Value**

vector of axis label locations

**Author(s)**

Justin Talbot <justintalbot@gmail.com>

**References**

Heckbert, P. S. (1990) Nice numbers for graph labels, Graphics Gems I, Academic Press Professional, Inc.

---

matplotlib                      *Matplotlib's labeling algorithm*

---

**Description**

Matplotlib's labeling algorithm

**Usage**

```
matplotlib(dmin, dmax, m)
```

**Arguments**

dmin	minimum of the data range
dmax	maximum of the data range
m	number of axis labels

**Value**

vector of axis label locations

**Author(s)**

Justin Talbot <justintalbot@gmail.com>

**References**

<https://matplotlib.org/>

---

nelder                              *Nelder's labeling algorithm*

---

**Description**

Nelder's labeling algorithm

**Usage**

```
nelder(dmin, dmax, m,  
      Q = c(1, 1.2, 1.6, 2, 2.5, 3, 4, 5, 6, 8, 10))
```

**Arguments**

dmin	minimum of the data range
dmax	maximum of the data range
m	number of axis labels
Q	set of nice numbers

**Value**

vector of axis label locations

**Author(s)**

Justin Talbot <justintalbot@gmail.com>

**References**

Nelder, J. A. (1976) AS 96. A Simple Algorithm for Scaling Graphs, Journal of the Royal Statistical Society. Series C., pp. 94-96.

---

 rpretty

*R's pretty algorithm implemented in R*


---

**Description**

R's pretty algorithm implemented in R

**Usage**

```
rpretty(dmin, dmax, m = 6, n = floor(m) - 1,
        min.n = n%/3, shrink.sml = 0.75, high.u.bias = 1.5,
        u5.bias = 0.5 + 1.5 * high.u.bias)
```

**Arguments**

dmin	minimum of the data range
dmax	maximum of the data range
m	number of axis labels
n	number of axis intervals (specify one of m or n)
min.n	nonnegative integer giving the <i>minimal</i> number of intervals. If min.n == 0, pretty(.) may return a single value.
shrink.sml	positive numeric by which a default scale is shrunk in the case when range(x) is very small (usually 0).
high.u.bias	non-negative numeric, typically > 1. The interval unit is determined as {1, 2, 5, 10} times b, a power of 10. Larger high.u.bias values favor larger units.
u5.bias	non-negative numeric multiplier favoring factor 5 over 2. Default and 'optimal': u5.bias = .5 + 1.5*high.u.bias.

**Value**

vector of axis label locations

**Author(s)**

Justin Talbot <justintalbot@gmail.com>

**References**

Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) *The New S Language*. Wadsworth & Brooks/Cole.

---

sparks

*Sparks' labeling algorithm*

---

**Description**

Sparks' labeling algorithm

**Usage**

```
sparks(dmin, dmax, m)
```

**Arguments**

dmin	minimum of the data range
dmax	maximum of the data range
m	number of axis labels

**Value**

vector of axis label locations

**Author(s)**

Justin Talbot <justintalbot@gmail.com>

**References**

Sparks, D. N. (1971) AS 44. Scatter Diagram Plotting, *Journal of the Royal Statistical Society. Series C.*, pp. 327-331.

---

thayer	<i>Thayer and Storer's labeling algorithm</i>
--------	---

---

**Description**

Thayer and Storer's labeling algorithm

**Usage**

```
thayer(dmin, dmax, m)
```

**Arguments**

dmin	minimum of the data range
dmax	maximum of the data range
m	number of axis labels

**Value**

vector of axis label locations

**Author(s)**

Justin Talbot <justintalbot@gmail.com>

**References**

Thayer, R. P. and Storer, R. F. (1969) AS 21. Scale Selection for Computer Plots, Journal of the Royal Statistical Society. Series C., pp. 206-208.

---

wilkinson	<i>Wilkinson's labeling algorithm</i>
-----------	---------------------------------------

---

**Description**

Wilkinson's labeling algorithm

**Usage**

```
wilkinson(dmin, dmax, m,  
  Q = c(1, 5, 2, 2.5, 3, 4, 1.5, 7, 6, 8, 9),  
  mincoverage = 0.8,  
  mrange = max(floor(m/2), 2):ceiling(6 * m))
```

**Arguments**

dmin	minimum of the data range
dmax	maximum of the data range
m	number of axis labels
Q	set of nice numbers
mincoverage	minimum ratio between the the data range and the labeling range, controlling the whitespace around the labeling (default = 0.8)
mrange	range of m, the number of tick marks, that should be considered in the optimization search

**Value**

vector of axis label locations

**Note**

Ported from Wilkinson's Java implementation with some changes. Changes: 1) m (the target number of ticks) is hard coded in Wilkinson's implementation as 5. Here we allow it to vary as a parameter. Since m is fixed, Wilkinson only searches over a fixed range 4-13 of possible resulting ticks. We broadened the search range to  $\max(\text{floor}(m/2), 2)$  to  $\text{ceiling}(6*m)$ , which is a larger range than Wilkinson considers for 5 and allows us to vary m, including using non-integer values of m. 2) Wilkinson's implementation assumes that the scores are non-negative. But, his revised granularity function can be extremely negative. We tweaked the code to allow negative scores. We found that this produced better labelings. 3) We added 10 to Q. This seemed to be necessary to get steps of size 1. It is possible for this algorithm to find no solution. In Wilkinson's implementation, instead of failing, he returns the non-nice labels spaced evenly from min to max. We want to detect this case, so we return NULL. If this happens, the search range, mrange, needs to be increased.

**Author(s)**

Justin Talbot <justintalbot@gmail.com>

**References**

Wilkinson, L. (2005) The Grammar of Graphics, Springer-Verlag New York, Inc.

# Index

## \* **dplot**

- labeling-package, [2](#)
- extended, [2, 3](#)
- extended.figures, [4](#)
- gnuplot, [2, 4](#)
- heckbert, [2, 5](#)
- labeling (labeling-package), [2](#)
- labeling-package, [2](#)
- matplotlib, [2, 6](#)
- nelder, [2, 6](#)
- pretty, [2](#)
- rpretty, [2, 7](#)
- sparks, [2, 8](#)
- thayer, [2, 9](#)
- wilkinson, [2, 9](#)