

Package ‘ggallin’

July 22, 2025

Type Package

Maintainer Steven E. Pav <shabbychef@gmail.com>

Version 0.1.1

Date 2017-10-01

License LGPL-3

Title Grab Bag of 'ggplot2' Functions

BugReports <https://github.com/shabbychef/ggallin/issues>

Description Extra geoms and scales for 'ggplot2', including `geom_cloud()`, a Normal density cloud replacement for errorbars; transforms `ssqrt_trans` and `pseudolog10_trans`, which are loglike but appropriate for negative data; `interp_trans()` and `warp_trans()` which provide scale transforms based on interpolation; and an infix compose operator for scale transforms.

Depends ggplot2 (>= 2.2.1)

Suggests knitr, testthat

Imports scales, grid

RoxygenNote 6.0.1

URL <https://github.com/shabbychef/ggallin>

Collate 'geom_cloud.R' 'ggallin.R' 'transforms.R'

NeedsCompilation no

Author Steven E. Pav [aut, cre]

Repository CRAN

Date/Publication 2017-10-02 23:24:58 UTC

Contents

ggallin-package	2
ggallin-NEWS	2
interp_trans	2
ssqrt_trans	4
%of%	5

Index**6**

ggallin-package	<i>Grab Bag of GGplot2 Functions.</i>
-----------------	---------------------------------------

Description

This package consists of some helper functions for working with ggplot2: geoms, transforms, *etc.*, with no real unifying theme among them.

Legal Mumbo Jumbo

ggallin is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

Author(s)

Steven E. Pav <shabbychef@gmail.com>

ggallin-NEWS	<i>News for package 'ggallin':</i>
--------------	------------------------------------

Description

News for package 'ggallin'

Version 0.1.1 (2017-10-01)

- submit to CRAN

interp_trans	<i>Interpolation based scale transforms.</i>
--------------	--

Description

Interpolation based scale transformations. The user supplies x and y (which should be monotonic increasing or decreasing in x) to create a scale transformation based on linear interpolation.

A 'warp' transformation is also supported wherein the user supplies x and w where, after sorting on x , the cumulative sum of w are used as the y in an interpolation transformation. Here w are the rate of increase, or 'weights'.

Usage

```
interp_trans(x=NULL,y=NULL,data=NULL,na.rm=TRUE,breaks=NULL,format=NULL)
```

```
warp_trans(x=NULL,w=NULL,data=NULL,na.rm=TRUE,breaks=NULL,format=NULL)
```

Arguments

<code>x</code>	the x coordinates for linear interpolation.
<code>y</code>	the y coordinates for linear interpolation.
<code>data</code>	A data.frame with columns of x and y for <code>interp_trans</code> or x and w for <code>warp_trans</code> . If data is given, it takes precedence over the given x , y , w .
<code>na.rm</code>	If TRUE, then missing x or y will be removed.
<code>breaks</code>	default breaks function for this transformation. The breaks function is applied to the raw data.
<code>format</code>	default format for this transformation. The format is applied to breaks generated to the raw data.
<code>w</code>	the w coordinates for the ‘warp’ interpolation. The cumulative sum of w are computed then fed to the interpolation transform.

Value

A scale transformation object.

Author(s)

Steven E. Pav <shabbychef@gmail.com>

See Also

[trans_new](#).

Examples

```
set.seed(1234)
ggplot(data.frame(x=rnorm(100),y=runif(100)),aes(x=x,y=y)) +
  geom_point() +
  scale_x_continuous(trans=interp_trans(x=seq(-10,10,by=1),y=cumsum(runif(21))))

set.seed(1234)
ggplot(data.frame(x=rnorm(100),y=runif(100)),aes(x=x,y=y)) +
  geom_point() +
  scale_x_continuous(trans=warp_trans(x=seq(-10,10,by=1),w=runif(21)))

# equivalently:
set.seed(1234)
ggplot(data.frame(x=rnorm(100),y=runif(100)),aes(x=x,y=y)) +
  geom_point() +
  scale_x_continuous(trans=warp_trans(data=data.frame(x=seq(-10,10,by=1),w=runif(21))))
```

```
# this is like trans_sqrt:
set.seed(1234)
myx <- seq(0,5,by=0.01)
ggplot(data.frame(x=rnorm(100),y=runif(100)),aes(x=x,y=y)) +
  geom_point() +
  scale_y_continuous(trans=interp_trans(x=myx,y=sqrt(myx)))
```

ssqrt_trans	<i>Various scale transforms.</i>
-------------	----------------------------------

Description

Various scale transformations.

Usage

ssqrt_trans

pseudolog10_trans

Format

An object of class `trans` of length 7.

Details

The available transforms:

- `ssqrt_trans` a signed square root transform appropriate for negative or positive numbers.
- `pseudolog10_trans` an asinh transformation, which is like a logarithm, but appropriate for negative or positive numbers. This transformation was taken from the Win Vector blog, <http://www.win-vector.com/blog/2012/03/modeling-trick-the-signed-pseudo-logarithm/>.

Value

A scale transformation object.

Author(s)

Steven E. Pav <shabbychef@gmail.com>

See Also

[trans_new](#).

<http://www.win-vector.com/blog/2012/03/modeling-trick-the-signed-pseudo-logarithm/>

Examples

```
set.seed(1234)
ggplot(data.frame(x=rnorm(100),y=runif(100)),aes(x=x,y=y)) +
  geom_point() +
  scale_x_continuous(trans=ssqrt_trans)

set.seed(1234)
ggplot(data.frame(x=rnorm(100),y=runif(100)),aes(x=x,y=y)) +
  geom_point() +
  scale_x_continuous(trans=pseudolog10_trans)
```

%of%*Composition of scale transforms.*

Description

A binary infix operator that allows one to compose together two scale transformations. We should have that the transformation `a %of% b` first applies `b`, then applies `a` to the results. This is useful for reversing scales, for example, along with other transformations.

Usage

```
a %of% b
```

Arguments

<code>a</code>	a transformation object.
<code>b</code>	a transformation object.

Value

a transformation object that performs `a` on the output of `b`.

Author(s)

Steven E. Pav <shabbychef@gmail.com>

See Also

[trans_new](#).

Examples

```
set.seed(1234)
# compose transformations with %of%:
ggplot(data.frame(x=rnorm(100),y=exp(rnorm(100,mean=-2,sd=4))),aes(x=x,y=y)) +
  geom_point() +
  scale_y_continuous(trans=scales::reverse_trans() %of% scales::log10_trans())
```

Index

- * **datasets**
 - ssqrt_trans, [4](#)
- * **package**
 - ggallin-package, [2](#)
- * **plotting**
 - %of%, [5](#)
 - interp_trans, [2](#)
 - ssqrt_trans, [4](#)
- %of%, [5](#)
- ggallin-NEWS, [2](#)
- ggallin-package, [2](#)
- interp_trans, [2](#)
- pseudolog10_trans (ssqrt_trans), [4](#)
- ssqrt_trans, [4](#)
- trans_new, [3–5](#)
- warp_trans (interp_trans), [2](#)