

Package ‘steemr’

January 16, 2019

Version 0.1.3

Date 2019-01-15

Title A Tool for Processing Steem Data

Author Peng Zhao <pzhao@pzhao.net>

Maintainer Peng Zhao <pzhao@pzhao.net>

Depends R (>= 3.1.0)

Imports RCurl, XML, rlist, wordcloud, tm, zoo, openair, lattice, latticeExtra, RODBC, VennDiagram, beginr, RColorBrewer, blogdown, ggplot2, httr, knitr, mongolite, plyr, shiny, lubridate, rjson, stringi, stringr, purrr, data.table, htmltab

Suggests

Description Steem is a blockchain-based social media platform (see <<https://en.wikipedia.org/wiki/Steemit>>). The Steem social activity data are saved in the Steem blockchain, the SteemDB database, the SteemSQL database, and so on. 'steemr' is an R package that downloads the Steem data from the SteemDB and SteemSQL servers, re-organizes the data in a user-friendly way, and visualizes the data for further analysis.

License GPL-3 | file LICENSE

URL <https://github.com/pzhaonet/steemr>

BugReports <https://github.com/pzhaonet/steemr/issues>

RoxygenNote 6.1.1

NeedsCompilation no

LazyData true

Repository CRAN

Date/Publication 2019-01-16 09:20:03 UTC

R topics documented:

accountCount	3
acsub	4
adailyf	4
avotenot	5
avotep	5
blog	6
bmd	7
charsplit	8
clearferfing	8
clear_title	9
gaccounts	9
gblog	10
gcner	11
gcomments	11
gdelegation	12
getAccount	13
getAccountVotes	13
getBlog	14
getComments	15
getDelegation	15
getNodes	16
getPost	16
getPostsByTag	17
getReplies	18
getSteemProperties	18
getTransactions	19
getTrending	20
getWitnesses	20
gfollow	21
gfollower	22
gfollowing	22
gid	23
gidpostl	24
gidposts	24
gpost	25
gposts	26
gspmv	27
gur	27
gvotep	28
gvoter	28
idlink	29
pcner	29
pdate	30
phist	30
phour	31
repcalc	33

<i>accountCount</i>	3
sner	34
sner_ui	34
sfollow	34
sfollow_server	35
sfollow_ui	35
skewness	36
sposts	36
sposts_server	37
sposts_ui	37
ssql	38
tag_of_post	38
unitconvert	39
whale	39
xatf	40
Index	41

<i>accountCount</i>	<i>Number of Accounts on Steem Blockchain</i>
---------------------	---

Description

Get the count of the number of accounts on the Steem Blockchain

Usage

`accountCount(node)`

Arguments

node Optional Argument (Steem Node to Query)

Value

number of accounts

Examples

```
## Not run:
accountCount()

## End(Not run)
```

acnsub	<i>Analysis of the CN sub categories.</i>
--------	---

Description

Analysis of the CN sub categories.

Usage

```
acnsub(from = Sys.Date() - 7, to = Sys.Date(), sql_con, if_plot = FALSE, top = 10)
```

Arguments

from	A Date object or character in '2017-10-24' format of the beginning of the period
to	A Date object or character in '2017-10-24' format of the end of the period
sql_con	A SQL connection
if_plot	A logic value of whether plot the time series
top	A numeric value of the Top tags for plotting

Value

A figure showing the active cn sub tag

adailyf	<i>Calculate the the daily frequency. adailyf means 'analysis of the daily frequency'.</i>
---------	--

Description

Calculate the the daily frequency. adailyf means 'analysis of the daily frequency'.

Usage

```
adailyf(mydata, datecol, if_plot = FALSE, col = "steelblue3", ylab = "Daily posts")
```

Arguments

mydata	A data frame with a date column.
datecol	A chracter string of the date column name. The date column could either be Date or Character (in '%Y-%m-%d' format).
if_plot	A logic value of whether plot the time series
col	A color for plotting
ylab	A character string of the y label

Value

A dataframe of the daily frequency

avotenot	<i>Find which followers have not voted a post yet. avotenot means 'analysis who has not voted yet'.</i>
----------	---

Description

Find which followers have not voted a post yet. avotenot means 'analysis who has not voted yet'.

Usage

```
avotenot(postlink = NA)
```

Arguments

postlink A character string of the link to a target post.

Value

A character vector of the name list of the followers who have not voted the target post yet.

Examples

```
## Not run:
avotenot("cn/@dapeng/steemit-markdown")

## End(Not run)
```

avotep	<i>Summary of the voters of a series of posts. avotep means 'analysis of the votes on the given posts retrieved with the appbase_api method.'</i>
--------	---

Description

Summary of the voters of a series of posts. avotep means 'analysis of the votes on the given posts retrieved with the appbase_api method.'

Usage

```
avotep(mypost = NA, if_plot = TRUE, top = 10)
```

Arguments

mypost	A data frame of a series of posts retrieved with the appbase_api method.
if_plot	A logic value of whether plot the pie diagram.
top	A numeric value of the Top voters plotted in the pie diagram.

Value

A data frame of the voters

Examples

```
## Not run:
avotep()

## End(Not run)
```

bblog

Build a hugo blog site for a steem author

Description

Build a hugo blog site for a steem author

Usage

```
bblog(author = NA, post_df, dest_path = "blog", initial = FALSE,
       template = "xmin", post_df_source = c("appbase_api", "steemsql.com"),
       orginal_link = c("steemit.com", "cnsteem.com", "busy.org", "steemdb.com",
                        "steemd.com"), my_github = "your_name/your_repo")
```

Arguments

author	author name without @
post_df	A dataframe with the posts contents retrieved from SteemSQL
dest_path	A character string
initial	if initialize a site
template	the hugo template
post_df_source	A character string of the data frame source.
orginal_link	A character string
my_github	A character string of a github repo

Value

a blogdown-hugo web site

Examples

```
## Not run:  
bblog()  
  
## End(Not run)
```

bmd

build markdown files from the posts

Description

build markdown files from the posts

Usage

```
bmd(post_df = NA, dest_path = "blog", post_df_source = c("appbase_api", "steemsql.com"),  
    if_yaml = FALSE, original_link = c("steemit.com", "cnsteem.com", "busy.org",  
    "steemdb.com", "steemd.com"))
```

Arguments

`post_df` A data frame of the posts.

`dest_path` A character string of the destination path for the markdown files.

`post_df_source` A character string of the data frame source.

`if_yaml` A logical value of whether the markdown files contain yaml headers.

`original_link` A character string vector of the original links to the posts.

Value

markdown files.

Examples

```
## Not run:  
bmd()  
  
## End(Not run)
```

charsplit*Split the characters in a data frame*

Description

Split the characters in a data frame

Usage

```
charsplit(dataframe = NA)
```

Arguments

dataframe A column of a dataframe with characters to split

Value

A character vector

clearferfing*Clear the format of the 'follow' information from steemdb.com.*

Description

Clear the format of the 'follow' information from steemdb.com.

Usage

```
clearferfing(x)
```

Arguments

x A character string of the follower information from steemdb.com.

Value

A character string of the formatted follower information

clear_title	<i>Clear the characters in the titles</i>
-------------	---

Description

Clear the characters in the titles

Usage

```
clear_title(title)
```

Arguments

title	The post title
-------	----------------

Value

A clear title

gaccounts	<i>Get the Steem account information within a period from SteemSQL gaccounts means 'get account information'.</i>
-----------	---

Description

Get the Steem account information within a period from SteemSQL gaccounts means 'get account information'.

Usage

```
gaccounts(from = Sys.Date() - 7, to = Sys.Date(), select = c("name", "created",
  "post_count", "last_post"), sql_con = NA, ylab = "Daily New Accounts",
  if_plot = FALSE)
```

Arguments

from	A Date object or character in '2017-10-24' format of the beginning of the period
to	A Date object or character in '2017-10-24' format of the end of the period
select	A character string vector of the column names
sql_con	A SQL connection
ylab	Label on the y-axis
if_plot	A logic value of whether plot the time series

Value

A data frame of the account information with a figure

Examples

```
## Not run:  
gaccounts()  
  
## End(Not run)
```

gblog

Rename the getBlog() function from the steemRdata package

Description

Rename the getBlog() function from the steemRdata package

Usage

```
gblog(id = NA)
```

Arguments

id A Steem ID

Value

A dataframe with the ID's posts

Examples

```
## Not run:  
gblog()  
  
## End(Not run)
```

gcner	<i>Get the CNer name list</i>
-------	-------------------------------

Description

Get the CNer name list

Usage

```
gcner(mydate = Sys.Date())
```

Arguments

mydate A Date or character string in 'Y-m-d' format

Value

a data frame of the CNer

Examples

```
## Not run:  
gcner()  
  
## End(Not run)
```

gcomments	<i>Get the comment records of an Steem ID within a period from SteemSQL gcomments means 'get comment information'.</i>
-----------	--

Description

Get the comment records of an Steem ID within a period from SteemSQL gcomments means 'get comment information'.

Usage

```
gcomments(id = NA, from = Sys.Date() - 7, to = Sys.Date(), select = c("root_title",  
  "root_comment", "created", "body"), sql_con, ylab = "Daily Comments",  
  if_plot = FALSE)
```

Arguments

id	A character string of a Steem ID
from	A Date object or character in '2017-10-24' format of the beginning of the period
to	A Date object or character in '2017-10-24' format of the end of the period
select	A character string vector of the column names
sql_con	A SQL connection
ylab	Label on the y-axis
if_plot	A logic value of whether plot the time series

Value

A data frame of the comment information with a figure

Examples

```
## Not run:
gcomments()

## End(Not run)
```

gdelegation	<i>Get the delegation information of a Steem ID from SteemSQL gdelegation means 'get delegation information'.</i>
-------------	---

Description

Get the delegation information of a Steem ID from SteemSQL gdelegation means 'get delegation information'.

Usage

```
gdelegation(id = NA, sql_con, if_plot = FALSE)
```

Arguments

id	A character string of a Steem ID
sql_con	A SQL connection
if_plot	A logic value of whether plot the time series

Value

A data frame of the delegation information with a figure

Examples

```
## Not run:  
gdelegation()  
  
## End(Not run)
```

getAccount	<i>Account Details</i>
------------	------------------------

Description

Get the details of a user account

Usage

```
getAccount(username, node)
```

Arguments

- username user name
- node Optional Argument (Steem Node to Query)

Value

Account Details

Examples

```
## Not run:  
getAccount("eroche")  
  
## End(Not run)
```

getAccountVotes	<i>Votes made by an Account</i>
-----------------	---------------------------------

Description

Get a list of all the votes made by an account

Usage

```
getAccountVotes(user = "eroche", node)
```

Arguments

user	Account to query
node	Optional Argument (Steem Node to Query)

Value

Data Table with list of votes and what they voted on

Examples

```
## Not run:
getAccountVotes("eroche")
getAccountVotes("eroche", node = "https://api.steemit.com")

## End(Not run)
```

getBlog	<i>Blog History</i>
---------	---------------------

Description

Get all main posts from a users blog (excluding Resteems). This function may take some time to process depending on the size of the blog history on an account

Usage

```
getBlog(username, node)
```

Arguments

username	Username of Blog Author.
node	Optional Argument (Steem Node to Query)

Value

Data.Table with Details of Blog Posts

Examples

```
## Not run:
getBlog("eroche")

## End(Not run)
```

`getComments`*Comment List*

Description

Get a list of all the comments from a users blog. This function may take some time to process depending on the size of the transaction history on an account

Usage

```
getComments(username, node)
```

Arguments

username	Username of Comment Author.
node	Optional Argument (Steem Node to Query)

Value

Data.Table with Details of Comments Posts

Examples

```
## Not run:  
getComments("eroche")  
  
## End(Not run)
```

`getDelegation`*Delegations Made by an Account*

Description

Delegations Made by an Account

Usage

```
getDelegation(user = "eroche", node)
```

Arguments

user	Account To Query
node	Optional Argument (Steem Node to Query)

Value

Data Table with list of delegations

Examples

```
## Not run:
getDelegation("eroche")

## End(Not run)
```

getNode	<i>List of Steem RPC nodes</i>
---------	--------------------------------

Description

The functions in steemRdata use public RPC nodes. This function returns a list of possible nodes which can be specified in the function calls.

Usage

```
getNode()
```

Value

List of nodes

Examples

```
## Not run:
getNode()

## End(Not run)
```

getPost	<i>Details of a Steem Post</i>
---------	--------------------------------

Description

Get details of a post, specified by the unique link (username and permalink)

Usage

```
getPost(username, permalink, node)
```


Arguments

username	Username of Post Author.
permlink	Permlink of Post
node	Optional Argument (Steem Node to Query)

Value

List with Details of Post

Examples

```
## Not run:
getPost("eroche", "time-series-with-r")
getPost("eroche", "time-series-with-r", "https://api.steemit.com")

## End(Not run)
```

getPostsByTag *Recent Posts containing a particular tag.*

Description

Get posts using a specific tag in Chronological order

Usage

```
getPostsByTag(tag = "steem", limit = 1, node)
```

Arguments

tag	tag to search.
limit	number of items to return
node	Optional Argument (Steem Node to Query)

Value

Data.Table with Details of Posts

Examples

```
## Not run:
getPostsByTag("letseat", 1)

## End(Not run)
```

`getReplies` *Replies Made to a Post*

Description

Replies Made to a Post

Usage

```
getReplies(user, permalink, node)
```

Arguments

<code>user</code>	Account To Query
<code>permalink</code>	Permlink of post to query
<code>node</code>	Optional Argument (Steem Node to Query)

Value

Data Table with list of replies made to specified post

Examples

```
## Not run:  
getReplies("eroche", "data-wrangling-with-r")  
  
## End(Not run)
```

`getSteemProperties` *Steem Properties*

Description

Get details about the latest state of the Steem Blockchain.

Usage

```
getSteemProperties(node)
```

Arguments

<code>node</code>	Optional Argument (Steem Node to Query)
-------------------	---

Value

steem properties

Examples

```
## Not run:  
getSteemProperties()  
  
## End(Not run)
```

getTransactions	<i>Transactions made on an account</i>
-----------------	--

Description

This function may take some time to process depending on the size of the transaction history on an account

Usage

```
getTransactions(user, n, node)
```

Arguments

user	Account To Query
n	Number of Transactions since beginning of account history
node	node

Value

The First n of transactions on an account

Examples

```
## Not run:  
getTransactions("eroche", 100)  
  
## End(Not run)
```

getTrending *Current Trending Posts*

Description

Current Trending Posts

Usage

```
getTrending(tag = "", limit = 100, node)
```

Arguments

tag	Tag to Filter
limit	Number of Posts to Return
node	Optional Argument (Steem Node to Query)
	Get details of the Trending Posts

Value

List with Details of Posts

Examples

```
## Not run:
getTrending() # Get first 100 Trending Posts
getTrending(20) # Get Top 10 Trending Posts
getTrending(20, node = "https://api.steemit.com")

## End(Not run)
```

getWitnesses *Current Steem Witness and Rank*

Description

Not all witnesses returned with this function will be active.

Usage

```
getWitnesses(limit = 1000, node)
```

Arguments

limit	Number of Witnesses to Return
node	Optional Argument (Steem Node to Query)
	Get details of the witnesses

Value

List with Details of Witnesses

Examples

```
## Not run:
getWitnesses() # Get first 1000 Witnesses
getWitnesses(20) # Get Top 20 Witnesses
getWitnesses(20, node = "https://api.steemit.com")

## End(Not run)
```

gfollow

Get a name list of an ID's followers and following. gfollow means 'get an ID's follower information'.

Description

Get a name list of an ID's followers and following. gfollow means 'get an ID's follower information'.

Usage

```
gfollow(id = NA, method = c("steemdb.com", "steemsql.com", "steemdata.com"), sql_con)
```

Arguments

`id` A character string of a Steem ID without '@'.

`method` A character string of the Steem data server to connect.

`sql_con` A connection to the SteemSQL server.

Value

A name list of an ID's followers and following

Examples

```
## Not run:
gfollow("dapeng")

## End(Not run)
```

gfollower	<i>A list of an id's followers from steemdb.com. gfollower means 'get the follower information'.</i>
-----------	--

Description

A list of an id's followers from steemdb.com. gfollower means 'get the follower information'.

Usage

```
gfollower(id = NA)
```

Arguments

id A character string of a Steem ID without '@'.

Value

character a dataframe of an ID's followers' info

Examples

```
## Not run:
gfollower("dapeng")

## End(Not run)
```

gfollowing	<i>The id list of an id's following from steemdb.com. gfollowing means 'get the following information'.</i>
------------	---

Description

The id list of an id's following from steemdb.com. gfollowing means 'get the following information'.

Usage

```
gfollowing(id = NA)
```

Arguments

id A character string of a Steem ID without '@'.

Value

A dataframe of an ID's following info

Examples

```
## Not run:  
gfollowing("dapeng")  
  
## End(Not run)
```

gid	<i>Get an ID's detailed info. gid means 'get an ID's information'.</i>
-----	--

Description

Get an ID's detailed info. gid means 'get an ID's information'.

Usage

```
gid(id = NA, method = c("appbase_api", "steemdb.com", "steemsql.com", "steemdata.com"),  
    sql_con)
```

Arguments

id	A character string of a Steem ID without '@'.
method	A character string of the Steem data server to connect.
sql_con	A connection to the SteemSQL server.

Value

A list or a dataframe of an ID's detailed info

Examples

```
## Not run:  
gid()  
  
## End(Not run)
```

gidpostl	<i>Obtain an ID's post hyperlinks. gidpostl means 'get an ID's post links'.</i>
----------	---

Description

Obtain an ID's post hyperlinks. gidpostl means 'get an ID's post links'.

Usage

```
gidpostl(id = NA, method = c("steemdb.com", "steemsql.com", "steemdata.com"), sql_con,
         post_number = 3, site = "steemit.com")
```

Arguments

id	A character string of a Steem ID without '@'.
method	A character string of the Steem data server to connect.
sql_con	A connection to the SteemSQL server.
post_number	A numeric value or NA. The number of the latest posts to be obtained. If NA, the 100 latest posts will be processed.
site	A character string of the site of the steem web UI

Value

A character string vector of an ID's post hyperlinks.

Examples

```
## Not run:
gidpostl()

## End(Not run)
```

gidposts	<i>Obtain an ID's post detailed info from steemdb.com. 'gidposts' means 'get an ID's posts'.</i>
----------	--

Description

Obtain an ID's post detailed info from steemdb.com. 'gidposts' means 'get an ID's posts'.

Usage

```
gidposts(id = NA, method = c("steemdb.com", "steemsql.com", "appbase_api",
                             "steemdata.com"), sql_con, post_number = NA)
```


Arguments

id	A character string of a Steem ID without '@'.
method	A character string of the Steem data server to connect.
sql_con	A connection to the SteemSQL server.
post_number	A numeric value or NA. The number of the latest posts to be obtained. If NA, all the posts will be processed.

Value

A data frame of an ID's post detailed info.

Examples

```
## Not run:
gidposts()

## End(Not run)
```

gpost	<i>Get the complete info of a single given post. gpost means 'get a post'.</i>
-------	--

Description

Get the complete info of a single given post. gpost means 'get a post'.

Usage

```
gpost(postlink = NA, method = c("steemdb.com", "steemsql.com"), sql_con,
      selected = FALSE, newline = FALSE, oldcolname)
```

Arguments

postlink	A character of the link to a post
method	A character string of the Steem data server to connect.
sql_con	A connection to the SteemSQL server.
selected	A logic value of whether return only selected info
newline	A logic value of whether rbind the returned dataframe with an existing one
oldcolname	A character string. If newline == TRUE, the returned dataframe is ordered according to oldcolname

Value

a dataframe

Examples

```
## Not run:  
gpost("cn/@dapeng/steemit-markdown")  
  
## End(Not run)
```

gposts

Get the detailed information of given posts. gposts means 'get posts'.

Description

Get the detailed information of given posts. gposts means 'get posts'.

Usage

```
gposts(postlinks = NA, method = c("steemdb.com", "steemsql.com", "steemdata.com"),  
       sql_con)
```

Arguments

postlinks	A character string of hyperlinks to target posts
method	A character string of the Steem data server to connect.
sql_con	A connection to the SteemSQL server.

Value

a dataframe of the detailed information of the given posts

Examples

```
## Not run:  
gposts(c("cn/@dapeng/xuer-sale", "utopian-io/@dapeng/steemg-four-more"))  
  
## End(Not run)
```

gspmv	<i>Get the value of Steem per Mvest</i>
-------	---

Description

Get the value of Steem per Mvest

Usage

```
gspmv()
```

Value

A numeric value of the Steem per Mvest

Examples

```
## Not run:  
gspmv()  
  
## End(Not run)
```

gur	<i>Get the utopian review data from utopian.rocks api</i>
-----	---

Description

Get the utopian review data from utopian.rocks api

Usage

```
gur(id = NA)
```

Arguments

id the steecian's ID to query

Value

a dataframe

Examples

```
## Not run:  
gur(NULL)  
  
## End(Not run)
```

gvotep	<i>A post's vote report based on steemdb.com. gvote means 'get the vote information of a post.'</i>
--------	---

Description

A post's vote report based on steemdb.com. gvote means 'get the vote information of a post.'

Usage

```
gvotep(postlink = NA)
```

Arguments

postlink A character string of the link of a post

Value

A dataframe of a post's voter information

Examples

```
## Not run:
gvotep()

## End(Not run)
```

gvoter	<i>Get the vote information of given IDs from SteemSQL. gvoter means 'get the voter activities.'</i>
--------	--

Description

Get the vote information of given IDs from SteemSQL. gvoter means 'get the voter activities.'

Usage

```
gvoter(voters = NA, from = Sys.Date() - 7, to = Sys.Date(), select = "*", sql_con = NA,
       if_plot = FALSE)
```

Arguments

voters A character vector of given Steem IDs.
 from A character string of the starting date. in 'Y-m-d' format.
 to A character string of the ending date. in 'Y-m-d' format.
 select A character vector of the selected columns in the SteemSQL query.
 sql_con A connection to the SteemSQL server.
 if_plot A logic value of whether plot the daily votes.

Value

A list (and a diagram) of the voter report.

Examples

```
## Not run:
gvoter(voters = NA)

## End(Not run)
```

idlink	<i>Convert an id from a character to html hyperlink</i>
--------	---

Description

Convert an id from a character to html hyperlink

Usage

```
idlink(id = NA)
```

Arguments

`id` A character string of a Steem ID without '@'.

Value

A character string of the hyperlink to the Steem id.

pcner	<i>Plot function for the Shiny app scner</i>
-------	--

Description

Plot function for the Shiny app scner

Usage

```
pcner(i, sliderplot, whwechat, zh)
```

Arguments

`i` A numeric indicator
`sliderplot` The ID of the sliderplot
`whwechat` The data from of the cner
`zh` The Chinese dictionary

Value

A diagram

pdate	<i>Plot a time series with x as Date.</i>
-------	---

Description

Plot a time series with x as Date.

Usage

```
pdate(x, y, myylab = "", mylegend = "", mycol = "darkgreen", myxlim = NULL,
      myylim = NULL)
```

Arguments

x	A Date vector.
y	A numeric vector.
myylab	A character string of the y label.
mylegend	A character string of the legend.
mycol	A color.
myxlim	A Date vector of the date range of x.
myylim	A Date vector of the date range of y.

Value

A time series diagram.

phist	<i>Plot a histogram of a vector.</i>
-------	--------------------------------------

Description

Plot a histogram of a vector.

Usage

```
phist(data = rnorm(1000), mybreaks = "Sturges", myxlim = NULL, myylim = NULL,
      eightlines = TRUE, eightdigit = 0, eightcex = 0.8, eightcolors = c("red",
      "darkgreen", "blue", "black", "purple", "gold")[c(1, 2, 3, 2, 1, 6, 6,
      5, 4, 5)], mylegend = "", myylab = "", return_df = FALSE, myfreq = FALSE,
      show_n = TRUE, show_skewness = TRUE, show_density = FALSE, show_normline = FALSE,
      show_mean = FALSE)
```

Arguments

<code>data</code>	A numeric vector to plot.
<code>mybreaks</code>	Breaks of the plotted bars.
<code>myxlim</code>	x limit.
<code>myylim</code>	y limit.
<code>eightlines</code>	Whether to draw the eight lines.
<code>eightdigit</code>	The digit number of the figures displayed on the diagram.
<code>eightcex</code>	The character size of the figures.
<code>eightcolors</code>	The colors of the eight lines.
<code>mylegend</code>	The legend.
<code>myxlab</code>	The x label.
<code>return_df</code>	Whether return the data frame of summary.
<code>myfreq</code>	Whether display the frequency,
<code>show_n</code>	Whether show the sample number.
<code>show_skewness</code>	Whether show the skewness test.
<code>show_density</code>	Whether show the density.
<code>show_normline</code>	Whether show the line of the normal distribution.
<code>show_mean</code>	Whether show the mean value.

Value

A histogram diagram.

<code>phour</code>	<i>Hour rose plot. Copied and modified from the openair package. <code>phour()</code> plotted a 24-hour clock, indicating the active hours on the basis of a time column in a data frame. It is highly customized on the basis of the <code>openair::windrose()</code> function.</i>
--------------------	--

Description

Hour rose plot. Copied and modified from the openair package. `phour()` plotted a 24-hour clock, indicating the active hours on the basis of a time column in a data frame. It is highly customized on the basis of the `openair::windrose()` function.

Usage

```
phour(my_df = NA, col_time = "created", ws = "ws", wd = "hour360", ws2 = NA, wd2 = NA,
      ws.int = 30, angle = 1, type = "default", cols = "default", grid.line = NULL,
      width = 1, seg = 0.9, auto.text = TRUE, breaks = 4, offset = 10, paddle = FALSE,
      key.header = NULL, key.footer = "(SBD)", key.position = "right", key = FALSE,
      dig.lab = 5, statistic = "prop.count", pollutant = NULL, cust_labels = c(0, 6,
      12, 18), annotate = FALSE, border = NA, quantile_line = TRUE, ...)
```

Arguments

<code>my_df</code>	A data frame containing the fields <code>ws</code> and <code>wd</code> .
<code>col_time</code>	the column name for the time stamp
<code>ws</code>	A character string of the name of the column representing the radius of the circular sectors in the diagram. It can be SBD payout, votes, comments, etc..
<code>wd</code>	A character string of the name of the column representing the hour of the day.
<code>ws2</code>	The user can supply a second set of data with which the first can be compared.
<code>wd2</code>	see <code>ws2</code> .
<code>ws.int</code>	A numeric vector of the <code>ws</code> interval. Default is 2.
<code>angle</code>	The hour spokes. Other potentially useful angle is 3 (hours).
<code>type</code>	<code>type</code> determines how the data are split i.e. conditioned, and then plotted. It can be 'season', 'year', 'weekday'... The default is will produce a single plot using the entire data. It can also be a numeric or factor vector.
<code>cols</code>	Colours for plotting. 'default', 'increment', 'heat', 'jet', 'hue' and user defined, such as <code>c("yellow", "green", "blue", "black")</code> .
<code>grid.line</code>	Grid line interval. NULL in default. It can also be a numeric value like 10, or a list like <code>list(value = 10, lty = 5, col = "purple")</code> .
<code>width</code>	The adjustment factor for width of payout intervals. For example, <code>width = 1.5</code> will make the paddle width 1.5 times wider. For <code>paddle = TRUE</code> .
<code>seg</code>	The width of the segments. 0.5 will produce segments $0.5 * \text{angle}$.
<code>auto.text</code>	A logical value of whether formatting the names and units automatically in the titles and axis labels
<code>breaks</code>	A numeric vector of the number of break points for payouts. 4 by default, which generates the break points 2, 4, 6, 8 SBD for <code>ws.int</code> default of 2 SBD. It can also be <code>c(0, 1, 10, 100)</code> , which breaks the data into segments $<1, 1-10, 10-100, >100$.
<code>offset</code>	A numeric value (default 10) of the size of the 'hole' in the middle of the plot, expressed as a percentage of the polar axis scale.
<code>paddle</code>	A logic value. TRUE means the 'paddle' style spokes, and FALSE means the 'wedge' style spokes.
<code>key.header</code>	A character string of additional text above the scale key.
<code>key.footer</code>	A character string of additional text below the scale key.
<code>key.position</code>	A character string of the location of the scale key. 'top', 'right', 'bottom' and 'left'.
<code>key</code>	Fine control of the scale key
<code>dig.lab</code>	A numeric value of the significant digits at which scientific number formatting is used in break point.
<code>statistic</code>	A character string of the statistic to be applied. <ul style="list-style-type: none"> 'prop.count' (default) sizes bins according to the proportion of the frequency of the records, 'prop.mean' sizes bins according to their relative contribution to the mean,

	<ul style="list-style-type: none"> • 'abs.count' provides the absolute count of records in each bin.
pollutant	Alternative data series to be sampled.
cust_labels	A numeric vector displayed as the customized labels
annotate	A logic value or a character string. <ul style="list-style-type: none"> • TRUE: the percentage calm and mean values are printed in each panel together with a description of the statistic below the plot. • " ": only the stastic is below the plot. • Custom annotations may be added by setting value to c("annotation 1", "annotation 2").
border	A character string of the border colour for shaded areas.
quantile_line	whether to display the quantile line
...	other parameters

Value

A figure with the active hour rose

Examples

```
## Not run:
phour()

## End(Not run)
```

repcalc	<i>Calculate the reputation of an ID</i>
---------	--

Description

Calculate the reputation of an ID

Usage

```
repcalc(rep)
```

Arguments

rep A numeric value of the raw reputation.

Value

A numeric value of the real reputation

scner	<i>A shiny app to display and analyze the CNers. scner means shiny app for CNers.</i>
-------	---

Description

A shiny app to display and analyze the CNers. scner means shiny app for CNers.

Usage

```
scner()
```

Value

a shinyapp which can be displayed in a web browser.

scner_ui	<i>UI for the Shiny app scner display and analysis</i>
----------	--

Description

UI for the Shiny app scner display and analysis

Usage

```
scner_ui()
```

Value

A UI function

sfollow	<i>A shiny app to display and analyze the followers a given ID. sfollow means shiny app for followers.</i>
---------	--

Description

A shiny app to display and analyze the followers a given ID. sfollow means shiny app for followers.

Usage

```
sfollow()
```

Value

a shinyapp which can be displayed in a web browser.

Examples

```
## Not run:  
sfollow()  
  
## End(Not run)
```

sfollow_server	<i>Server for the Shiny app sfollow display and analysis</i>
----------------	--

Description

Server for the Shiny app sfollow display and analysis

Usage

```
sfollow_server(input, output, session)
```

Arguments

input	The input of the server.
output	The output of the server.
session	The session of the server.

Value

A server function

sfollow_ui	<i>UI for the Shiny app sfollow()</i>
------------	---------------------------------------

Description

UI for the Shiny app sfollow()

Usage

```
sfollow_ui()
```

Value

A UI function

skewness	<i>Calculate the skewness</i>
----------	-------------------------------

Description

Calculate the skewness

Usage

```
skewness(x)
```

Arguments

x	The data for calculation.
---	---------------------------

Value

The skewness

sposts	<i>A shiny app to display and analyze the posts of a given ID. sposts means shiny app for posts.</i>
--------	--

Description

A shiny app to display and analyze the posts of a given ID. sposts means shiny app for posts.

Usage

```
sposts()
```

Value

a shinyapp which can be displayed in a web browser.

sposts_server	<i>Server for the Shiny app sposts display and analysis</i>
---------------	---

Description

Server for the Shiny app sposts display and analysis

Usage

```
sposts_server(input, output, session)
```

Arguments

input	The input of the server.
output	The output of the server.
session	The session of the server.

sposts_ui	<i>UI for the Shiny app sposts display and analysis</i>
-----------	---

Description

UI for the Shiny app sposts display and analysis

Usage

```
sposts_ui()
```

Value

A UI function

ssql	<i>Open Connections to the SteemSQL server. ssql means 'send to sql'.</i>
------	---

Description

Open Connections to the SteemSQL server. ssql means 'send to sql'.

Usage

```
ssql(uid = NA, pwd = NA)
```

Arguments

uid	A character string of the SteemSQL ID.
pwd	A character string of the SteemSQL password.

Value

A connection to the SeemSQL server.

Examples

```
## Not run:
ssql()

## End(Not run)
```

tag_of_post	<i>Retrieve tags from json_str of a post</i>
-------------	--

Description

Retrieve tags from json_str of a post

Usage

```
tag_of_post(post_json_str)
```

Arguments

post_json_str	json string with tag information
---------------	----------------------------------

Value

tags

unitconvert	<i>Concert unit from GV, MV, kV to numeric values.</i>
-------------	--

Description

Concert unit from GV, MV, kV to numeric values.

Usage

unitconvert(x)

Arguments

x the unit GV, MV, kV to convert

Value

the numeric multiplier.

whale	<i>Calculate the level of and ID</i>
-------	--------------------------------------

Description

Calculate the level of and ID

Usage

whale(x)

Arguments

x the Raw SP value of an ID

Value

the level of the ID

`xatf`*Adapt the x axis to the time range*

Description

Adapt the x axis to the time range

Usage

`xatf(x)`

Arguments

`x` the time range

Value

the optimized label

Index

accountCount, 3
acsub, 4
adailyf, 4
avotenot, 5
avotep, 5

bblog, 6
bmd, 7

charsplit, 8
clear_title, 9
clearferfing, 8

gaccounts, 9
gblog, 10
gcner, 11
gcomments, 11
gdelegation, 12
getAccount, 13
getAccountVotes, 13
getBlog, 14
getComments, 15
getDelegation, 15
getNodes, 16
getPost, 16
getPostsByTag, 17
getReplies, 18
getSteemProperties, 18
getTransactions, 19
getTrending, 20
getWitnesses, 20
gfollow, 21
gfollower, 22
gfollowing, 22
gid, 23
gidpost1, 24
gidposts, 24
gpost, 25
gposts, 26
gspmv, 27

gur, 27
gvotep, 28
gvoter, 28

idlink, 29

pcner, 29
pdate, 30
phist, 30
phour, 31

repcalc, 33

scner, 34
scner_ui, 34
sfollow, 34
sfollow_server, 35
sfollow_ui, 35
skewness, 36
sposts, 36
sposts_server, 37
sposts_ui, 37
ssql, 38

tag_of_post, 38

unitconvert, 39

whale, 39

xatf, 40