

Package ‘hammond’

June 14, 2019

Type Package

Title Useful analysis utilities

Version 0.1.0

Author David Hammond

Maintainer David Hammond <anotherdavidhammond@gmail.com>

Description Just some useful stuff for me

License GPL-3

Encoding UTF-8

Depends R (>= 3.5.0)

Imports tidyverse (>= 1.2.1),
countrycode (>= 1.1.0),
RPostgreSQL (>= 0.6.2),
whereami (>= 0.1.8.1),
digest (>= 0.6.19),
processx (>= 3.3.1),
fs (>= 1.3.1),
devtools (>= 2.0.2),
pbapply (>= 1.4-0),
Hmisc (>= 4.2-0),
padr (>= 0.4.2)

LazyData true

RoxygenNote 6.1.1

R topics documented:

hammond-package	2
hcorr	2
hcountrycode	3
hcountryexampledata	3
hcountryinfo	3
hdbkill	4
hdb_connect	4
hdb_create_db	5
hdb_get	5
hdb_get_toc	6
hdb_login	6

hdb_search	7
hdb_update_master	7
hinterpolate	8
hpack_manual	8
hpc_change	9
Index	10

hammond-package	<i>hammond: some stuff</i>
-----------------	----------------------------

Description

Just some useful stuff for me

Installation

```
devtools::install_github("david-hammond/hammond")
```

hcorr	<i>hcorr</i>
-------	--------------

Description

This function calculates correlations between variables

Usage

```
hcorr(df, min.pairs = 20, verbose = TRUE, filter.by.p = FALSE)
```

Arguments

df	name of dataframe to use for correlation, needs to be long format 4 column data frame: geocode, variablename, year, value
min.pairs	minimum number of pairs to correlate
verbose	enable n and p values reporting, TRUE or FALSE
filter.by.p	Do you want to filter for significant p values?

Examples

```
#need 4 column data frame, geocode, variablename, year, value
library(hammond)
corr = hcorr(hcountryexempladata)
```

hcountrycode	<i>hcountrycode</i>
--------------	---------------------

Description

This function calculates correlations between variables

Usage

```
hcountrycode(x, source_file = whereami::thisfile())
```

Arguments

countries list of countries

Examples

```
hcountrycode(hcountryexempladata$geocode)
```

hcountryexempladata	<i>countryinfo</i>
---------------------	--------------------

Description

countryinfo

Usage

```
hcountryexempladata
```

Format

An object of class data.frame with 11934 rows and 5 columns.

hcountryinfo	<i>countryinfo</i>
--------------	--------------------

Description

countryinfo

This function calculates correlations between variables

Usage

```
hcountryinfo(df)
```

```
hcountryinfo(df)
```

Arguments

countries	list of countries
countries	list of countries

Examples

```
#need 4 column data frame, geocode, variablename, year, value  
  
hcountry_info  
  
This function calculates correlations between variables  
  
#need 4 column data frame, geocode, variablename, year, value
```

hdbkill	<i>hdb_kill</i>
---------	-----------------

Description

This function calculates correlations between variables

Usage

```
hdbkill()
```

Arguments

countries	list of countries
-----------	-------------------

Examples

```
#need 4 column data frame, geocode, variablename, year, value
```

hdb_connect	<i>hdb_connect</i>
-------------	--------------------

Description

This function calculates correlations between variables

Usage

```
hdb_connect(db = "postgres", port = 5432, user = "postgres")
```

Arguments

countries	list of countries
-----------	-------------------

Examples

```
#need 4 column data frame, geocode, variablename, year, value
```

hdb_create_db	<i>hdb_create_db</i>
---------------	----------------------

Description

This function calculates correlations between variables

Usage

```
hdb_create_db(db)
```

Arguments

countries list of countries

Examples

```
#need 4 column data frame, geocode, variablename, year, value
```

hdb_get	<i>hdb_get</i>
---------	----------------

Description

This function calculates correlations between variables

Usage

```
hdb_get(vars)
```

Arguments

countries list of countries

Examples

```
#need 4 column data frame, geocode, variablename, year, value
```

hdb_get_toc	<i>hdb_get_toc</i>
-------------	--------------------

Description

This function calculates correlations between variables

Usage

```
hdb_get_toc(db = "master")
```

Arguments

countries list of countries

Examples

```
#need 4 column data frame, geocode, variablename, year, value
```

hdb_login	<i>hdb_login</i>
-----------	------------------

Description

This function calculates correlations between variables

Usage

```
hdb_login(host = NULL, password = NULL)
```

Arguments

countries list of countries

Examples

```
hdb_login("192.168.0.98", password = "peace123")
```

hdb_search	<i>hdb_search</i>
------------	-------------------

Description

This function calculates correlations between variables

Usage

```
hdb_search(vars, db = "master")
```

Arguments

countries list of countries

Examples

```
#need 4 column data frame, geocode, variablename, year, value
```

hdb_update_master	<i>hdb_update_master</i>
-------------------	--------------------------

Description

This function calculates correlations between variables

Usage

```
hdb_update_master()
```

Arguments

countries list of countries

Examples

```
#need 4 column data frame, geocode, variablename, year, value
```

hinterpolate	<i>hinterpolate</i>
--------------	---------------------

Description

This is a wrapper function takes a data frame and fills in interpolated and extrapolated data for the whole time series

Usage

```
hinterpolate(df)
```

Arguments

df	dataframe in iep format
----	-------------------------

Value

Returns list with filled in time series, column yhat is the interpolated value. Please check original value with yhat column to make sure you are happy with the results

Author(s)

Dave

Examples

```
hinterpolate(hcountryexempladata)
```

hpack_manual	<i>create package manual</i>
--------------	------------------------------

Description

This function calculates combinations for efficient correlation calculations

Usage

```
hpack_manual(pack = "hammond")
```

Arguments

pack	name of package
------	-----------------

hpc_change	<i>Calculate proportional change</i>
------------	--------------------------------------

Description

This function calculates proportional change in GPI for a country from one year to another.

Usage

```
hpc_change(all)
```

Arguments

`all` the dataframe to be processed

Value

Returns a dataframe containing the raw and annual growths in GPI for each country `hpc_change(hcountryexempladata)`

Index

*Topic **analysis-utils**

hpc_change, [9](#)

*Topic **datasets**

hcountryexampledata, [3](#)

*Topic **imputation**

hinterpolate, [8](#)

*Topic **utilities**

hpc_change, [9](#)

hammond (hammond-package), [2](#)

hammond-package, [2](#)

hcorr, [2](#)

hcountrycode, [3](#)

hcountryexampledata, [3](#)

hcountryinfo, [3](#)

hdb_connect, [4](#)

hdb_create_db, [5](#)

hdb_get, [5](#)

hdb_get_toc, [6](#)

hdb_login, [6](#)

hdb_search, [7](#)

hdb_update_master, [7](#)

hdbkill, [4](#)

hinterpolate, [8](#)

hpack_manual, [8](#)

hpc_change, [9](#)