# Package 'hammond'

August 2, 2019

**Type** Package

**Title** Useful analysis utilities

**Version** 0.1.0

**Author** David Hammond

**Maintainer** David Hammond <anotherdavidhammond@gmail.com>

**Description** Useful funtions

**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),
scales (>= 1.0.0)

**LazyData** true

**RoxygenNote** 6.1.1

## R topics documented:

---

`hammond-package` *hammond: some stuff*

---

## Description

Useful funtions

## Installation

devtoolls::install_github("david-hammond/hammond")

---

`haddcountryinfo` *haddcountryinfo*

---

## Description

This function adds country specific information to a dataframe by matching countries to country codes.

## Usage

```
haddcountryinfo(df)
```

## Arguments

| | |
|---|---|
| `countries` | list of countries |
| `countries` | list of countries |

## Examples

```
#need 4 column data frame, geocode, variablename, year, value
hcountryinfo(hcountryexampledata)
hcountry_info

This function calculates correlations between variables

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

hcorr *hcorr*

## 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(hcountryexampledata)
```

hcountrycode *hcountrycode*

## Description

This function replaces country name or code with iso3c country codes. Can also be used in reverse.

## Usage

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

## Arguments

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

## Examples

```
hcountrycode(hcountryexampledata$geocode)
```

## hcountryexampledata *countryinfo*

### Description

countryinfo

### Usage

```
hcountryexampledata
```

### Format

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

## hcountryinfo *countryinfo*

### Description

countryinfo

### Usage

```
hcountryinfo
```

### Format

An object of class `data.frame` with 233 rows and 7 columns.

## hdbkill *hdb_kill*

### Description

This function kills all connections to the database, use as a last resort if you get a db connection error

### Usage

```
hdbkill()
```

### Arguments

| | |
|---|---|
| `countries` | list of countries |

### Examples

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

hdb_backup *hdb_backup*

## Description

This function backsup a database, use only if you know what you are doing

## Usage

```
hdb_backup(host = "192.168.0.98", user = "postgres",
  password = "peace123", port = 5432)
```

## Arguments

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

## Examples

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

hdb_connect *hdb_connect*

## Description

This function calculates correlations between variables

## Usage

```
hdb_connect(port = 5432)
```

## Arguments

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

## Examples

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

hdb_corr *hdb_corr*

## Description

This function calculates correlations between variables

## Usage

```
hdb_corr(df)
```

## Arguments

| | |
|---|---|
| df | name of dataframe to use for correlation, needs to be long format 4 column data frame: geocode, variablename, year, value |

## Examples

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

hdb_create_db *hdb_create_db*

## Description

This function creates a database, use only if you know what you are doing

## Usage

```
hdb_create_db(host, db, user, password)
```

## Arguments

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

## Examples

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

hdb_get *hdb_get*

## Description

This function retrieves and caches data from any source in the database.

## Usage

```
hdb_get(vars)
```

## Arguments

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

## Examples

```
#need 4 column data frame, geocode, variablename, year, value
hdb_login("192.168.0.98", password = "peace123")
db_get("Perceptions of Criminality Raw")
```

hdb_login *hdb_login*

## Description

This function allows access to a database by entering an IP adress and passcode.

## Usage

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

## Arguments

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

## Examples

```
hdb_login("192.168.0.64", db = "countrydata", user = "postgres", password = "peace123")
```

hdb_search *hdb_search*

## Description

This function searches the database and retrieves specified data.

## Usage

```
hdb_search(vars)
```

## Arguments

| | |
|---|---|
| countries | list of countries<br>hdb_login("192.168.0.98", password = "peace123") hdb_search("Criminal) |

hdb_toc *hdb_get_toc*

## Description

This function retrieves the Table of Contents from a specified database.

## Usage

```
hdb_toc()
```

## Arguments

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

## Examples

```
#need 4 column data frame, geocode, variablename, year, value
hdb_login("192.168.0.98", password = "peace123")
hdb_get_toc()
```

## hinterpolate *hinterpolate*

### 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(hcountryexampledata)
```

## hpack_manual *create package manual*

### Description

This function calculates combinations for efficient correlation calculations

### Usage

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

### Arguments

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

## hpc_change *Calculate proportional change*

### 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(hcountryexampledata)

## hpopulation *hpopulation*

### Description

This function appends a column of populations to a dataframe

### Usage

```
hpopulation(df)
```

### Arguments

| | |
|---|---|
| df | |

### Examples

```
x = hpopulation(hammond::hcountryexampledata)
```

# Index