

Package ‘ohicore’

December 6, 2013

Version 0.1

Date 2013-09-25

Title Ocean Health Index calculation package

Author Steve Hastings

Maintainer Steve Hastings <hastings@nceas.ucsb.edu>

Depends R (>= 2.14.0),plyr,reshape2,RJSONIO

Description A collection of functions for generically calculating the Ocean Health Index scores as well as individual goals and sub-goals.

License MIT

Collate

‘CalculatePressuresComponent.R’ ‘CalculateResilienceComponent.R’ ‘CalculateStatusComponent.R’ ‘CalculateSubgoal

LazyData TRUE

R topics documented:

CalculatePressuresComponent	2
CalculateResilienceComponent	3
CalculateStatusComponent	3
CalculateSubgoal	4
Halpern2012.	4
Halpern2012.AO	5
Halpern2012.BD.HAB	5
Halpern2012.BD.SPP	6
Halpern2012.CP	6
Halpern2012.CS	7
Halpern2012.CW	7
Halpern2012.FP	8
Halpern2012.FP.FIS	8
Halpern2012.FP.MAR	9
Halpern2012.ICO	9
Halpern2012.LE	10
Halpern2012.LE.ECO	10
Halpern2012.LE.LIV	11

Halpern2012.LSP	11
Halpern2012.NP	12
Halpern2012.SP	12
Halpern2012.TR	13
Layers	13
layers.Global2012.Nature2012ftp	14
layers.Global2012.www2013	14
layers.Global2013.www2013	15
Scores	15
scores.Global2012.www2013	16
scores.Global2013.www2013	16
SelectLayers	16
SpatialSchemes	17
TransformSpatialScheme	17
Index	19

CalculatePressuresComponent

Calculate the pressures component of each (sub)goal.

Description

Calculate the pressures component of each (sub)goal.

Usage

```
CalculatePressuresComponent(echo.pressures,
  social.pressures, c.name = "category",
  s.name = "region", gamma = 0.5)
```

Arguments

echo.pressures data.frame containing columns 'region', 'category', 'weight', and 'value'

social.pressures data.frame containing columns 'region', and 'value'

gamma (optional) if not specified defaults to 0.5

Value

data.frame containing columns 'region', 'p_E', 'p_S', and 'p_x'

`CalculateResilienceComponent`*Calculate the Resilience component of each (sub)goal.*

Description

Calculate the Resilience component of each (sub)goal.

Usage

```
CalculateResilienceComponent(goal.specific.regulations,  
  ecological.integrity, social.integrity,  
  c.name = "category", s.name = "region", gamma = 0.5)
```

Arguments

<code>goal.specific.regulations</code>	(data.frame) contains columns 'region', 'weight', and 'value'
<code>gamma</code>	(numeric) represents the weighting between ecological and social aspects of resilience, defaults to 0.5 (equal weights)

Value

(data.frame)

`CalculateStatusComponent`*Compute a single subgoal.*

Description

Compute a single subgoal.

Usage

```
CalculateStatusComponent(DATA, fun, trend.Years = 5,  
  c.name = "year", s.name = "region")
```

Arguments

<code>DATA</code>	data.frame containing columns 'region', 'value', and (optionally) 'w'
<code>fun</code>	(optional) function for calculating the subgoal value, if not specified it will default to a weighted average
<code>w</code>	(optional) numeric vector describing the

Value

stuff

CalculateSubgoal	<i>Compute a single subgoal.</i>
------------------	----------------------------------

Description

Compute a single subgoal.

Usage

```
CalculateSubgoal(current.data, eco.pressures,
  social.pressures, gs.regulations, social.integrity,
  eco.integrity, fun = stats::weighted.mean,
  trend.Years = 5)
```

Arguments

DATA	data.frame containing columns 'region', 'value', and (optionally) 'w'
fun	(optional) function for calculating the subgoal value, if not specified it will default to a weighted average
w	(optional) numeric vector describing the

Value

stuff

Halpern2012.	<i>Calculate Biodiversity.</i>
--------------	--------------------------------

Description

Calculate Biodiversity.

Usage

```
Halpern2012.(A, G, w, Cc, Cr, ...)
```

Arguments

placeholder	placeholder
-------------	-------------

Value

1

Halpern2012.AO	<i>Calculate Artisanal Fishing Opportunities.</i>
----------------	---

Description

Calculate Artisanal Fishing Opportunities.

Usage

Halpern2012.AO(Sao, Oao, PPPpcGDP, ...)

Arguments

placeholder	placeholder Sao
placeholder	placeholder Oao
placeholder	placeholder PPPpcGDP

Value

1

Halpern2012.BD.HAB	<i>Calculate Habitats subgoal of Biodiversity.</i>
--------------------	--

Description

Calculate Habitats subgoal of Biodiversity.

Usage

Halpern2012.BD.HAB(Cc, Cr, ...)

Arguments

placeholder	placeholder
-------------	-------------

Value

1

Halpern2012.BD.SPP	<i>Calculate Species subgoal of Biodiversity.</i>
--------------------	---

Description

Calculate Species subgoal of Biodiversity.

Usage

Halpern2012.BD.SPP(A, G, w, ...)

Arguments

placeholder	placeholder
-------------	-------------

Value

1

Halpern2012.CP	<i>Calculate Coastal Protection</i>
----------------	-------------------------------------

Description

Calculate Coastal Protection

Usage

Halpern2012.CP(Cc, Cr, w, A, ...)

Arguments

placeholder	placeholder Cc current 'condition' of habitat k
placeholder	placeholder Cr reference 'condition' of habitat k
placeholder	placeholder A amount of area covered by habitat k
placeholder	placeholder w rank weight of habitat protective ability

Value

1

Halpern2012.CS	<i>Calculate Carbon Storage</i>
----------------	---------------------------------

Description

Calculate Carbon Storage

Usage

Halpern2012.CS(Cc, Cr, A, ...)

Arguments

placeholder	placeholder Cc current 'condition' of habitat k
placeholder	placeholder Cr reference 'condition' of habitat k
placeholder	placeholder A amount of area covered by habitat k

Value

1

Halpern2012.CW	<i>Calculate Clean Waters.</i>
----------------	--------------------------------

Description

Calculate Clean Waters.

Usage

Halpern2012.CW(a, u, l, d, ...)

Arguments

placeholder	placeholder a number of coastal people without access to sanitation rescaled to global maximum
placeholder	placeholder u 1 - (nutrient input)
placeholder	placeholder l 1 - (chemical input)
placeholder	placeholder d 1 - (marine debris input)

Value

1

Halpern2012.FP	<i>Calculate Food Provision.</i>
----------------	----------------------------------

Description

Calculate Food Provision.

Usage

Halpern2012.FP(w, dBt, mMSY, Bt, Tc, k, Smk, Ac, Yk, ...)

Arguments

placeholder	placeholder k each mariculture species
placeholder	placeholder Smk sustainability score for each species k
placeholder	placeholder Ac area of coastal waters (3nm strip)
placeholder	placeholder Yl yield of each species k

Value

1

Halpern2012.FP.FIS	<i>Calculate Fisheries subgoal of Food Provision.</i>
--------------------	---

Description

Calculate Fisheries subgoal of Food Provision.

Usage

Halpern2012.FP.FIS(mMSY, Bt, Tc, ...)

Arguments

placeholder	placeholder dBt absolute difference between landed biomass and mMSY
placeholder	placeholder mMSY multi-species maximum sustainable yield
placeholder	placeholder Tc taxonomic report quiality correction factor
placeholder	placeholder Bt wild-caught fishing yield

Value

1

Halpern2012.FP.MAR	<i>Calculate Mariculture subgoal of Food Provision.</i>
--------------------	---

Description

Calculate Mariculture subgoal of Food Provision.

Usage

Halpern2012.FP.MAR(k, Smk, Ac, Yk, ...)

Arguments

placeholder	placeholder k each mariculture species
placeholder	placeholder Smk sustainability score for each species k
placeholder	placeholder Ac area of coastal waters (3nm strip)
placeholder	placeholder Yl yield of each species k

Value

1

Halpern2012.ICO	<i>Calculate Iconic Species subgoal of Sense of Place.</i>
-----------------	--

Description

Calculate Iconic Species subgoal of Sense of Place.

Usage

Halpern2012.ICO(S, w, ...)

Arguments

placeholder	placeholder S number of assessed species in each category
placeholder	placeholder w status weight assigned per threat category

Value

1

Halpern2012.LE	<i>Calculate Coastal Livelihoods and Economies.</i>
----------------	---

Description

Calculate Coastal Livelihoods and Economies.

Usage

Halpern2012.LE(jc, jr, gc, gr, ec, er, ...)

Arguments

placeholder	placeholder jc total adjusted jobs per sector at current time
placeholder	placeholder jr total adjusted jobs per sector at reference time
placeholder	placeholder gc average PPP-adjusted per-capita annual wages per sector in current region
placeholder	placeholder gr average PPP-adjusted per-capita annual wages per sector in reference region
placeholder	placeholder ec total adjusted revenue generated per sector at current time
placeholder	placeholder er total adjusted revenue generated per sector at reference time

Value

1

Halpern2012.LE.ECO	<i>Calculate Economies subgoal of Coastal Livelihoods and Economies.</i>
--------------------	--

Description

Calculate Economies subgoal of Coastal Livelihoods and Economies.

Usage

Halpern2012.LE.ECO(ec, er, ...)

Arguments

placeholder	placeholder ec total adjusted revenue generated per sector at current time
placeholder	placeholder er total adjusted revenue generated per sector at reference time

Value

1

Halpern2012.LE.LIV	<i>Calculate Livelihoods subgoal of Coastal Livelihoods and Economies.</i>
--------------------	--

Description

Calculate Livelihoods subgoal of Coastal Livelihoods and Economies.

Usage

Halpern2012.LE.LIV(jc, jr, gc, gr, ...)

Arguments

placeholder	placeholder jc total adjusted jobs per sector at current time
placeholder	placeholder jr total adjusted jobs per sector at reference time
placeholder	placeholder gc average PPP-adjusted per-capita annual wages per sector in current region
placeholder	placeholder gr average PPP-adjusted per-capita annual wages per sector in reference region

Value

1

Halpern2012.LSP	<i>Calculate Lasting Special Places subgoal of Sense of Place.</i>
-----------------	--

Description

Calculate Lasting Special Places subgoal of Sense of Place.

Usage

Halpern2012.LSP(CMPA, tCMPA, CP, tCP, ...)

Arguments

placeholder	placeholder CMPA coastal marine protected area
placeholder	placeholder tCMPA total coastal marine area
placeholder	placeholder CP coastline protected
placeholder	placeholder tCP total coastline

Value

1

Halpern2012.NP

*Calculate Natural Products. (Needs work)***Description**

Calculate Natural Products. (Needs work)

Usage

Halpern2012.NP(N, wp, Hp, E, R, Nv, Nk, w, ...)

Arguments

placeholder	placeholder N number of products that have ever been harvested
placeholder	placeholder wp proportional peak dollar value of each product relative to the total peak dollar value of all products
placeholder	placeholder Hp harvest of a product relative to its buffered peak reference point
placeholder	placeholder E exposure term
placeholder	placeholder R risk term
placeholder	placeholder Nv 1 or 2, depending on whether or not a viability term is used
placeholder	placeholder Nk number of species in each k category of exploitation
placeholder	placeholder w weight assigned to each k category of exploitation status

Value

1

Halpern2012.SP

*Calculate Sense of Place.***Description**

Calculate Sense of Place.

Usage

Halpern2012.SP(S, w, CMPA, tCMPA, CP, tCP, ...)

Arguments

placeholder	placeholder S number of assessed species in each category
placeholder	placeholder w status weight assigned per threat category
placeholder	placeholder CMPA coastal marine protected area
placeholder	placeholder tCMPA total coastal marine area
placeholder	placeholder CP coastline protected
placeholder	placeholder tCP total coastline

Value

1

Halpern2012.TR

*Calculate Tourism and Recreation.***Description**

Calculate Tourism and Recreation.

Usage

Halpern2012.TR(D, t, V, S, ...)

Arguments

placeholder	placeholder D number of tourist-days
placeholder	placeholder t most recent year
placeholder	placeholder V total region population size
placeholder	placeholder S sustainability factor

Value

1

Layers

*Layers reference class.***Description**

Layers reference class.

Usage

Layers(...)

Arguments

layers.csv	path to comma-seperated value file with row of metadata per layer
layers.dir	path of directory containing individual layer files

Details

To instantiate this object, `Layers(layers.csv, layers.dir)` is used. The `layers.csv` is expected to have the following columns:

- *layer* - unique identifier (no spaces or special characters)
- *targets* - the pipe and space (' | ') delimited list of targets (goal name, 'Pressures' or 'Resilience') to feed this data layer
- *title* - full title of the variable
- *description* detailed description

- *citation* - reference for documentation
- *units* - indicating units and required column name in the layer csv file
- *filename* - the csv data file for the layer

The layers.dir directory should contain all the csv filenames listed in the layers.csv file.

Value

object (non-instantiated) reference class of Layers containing

- *meta* - metadata data frame of original layers.csv
- *data* - named list of data frames, one per layer
- *targets* - named list of character vector indicating a layer's targets, goal (status, trend) or dimension (pressures, resilience)

layers.Global2012.Nature2012ftp

Layers accompanying Nature 2012 publication on the FTP site for Global 2012 analysis.

Description

These layers get used to calculate the Ocean Health Index.

Format

a [Layers](#) object

References

<http://ohi-science.org>

layers.Global2012.www2013

Layers used for the 2013 web launch applied to Global 2012 analysis.

Description

These layers get used to calculate the Ocean Health Index.

Format

a [Layers](#) object

References

<http://ohi-science.org>

layers.Global2013.www2013

Layers used for the 2013 web launch applied to Global 2013 analysis.

Description

These layers get used to calculate the Ocean Health Index.

Format

a [Layers](#) object

References

<http://ohi-science.org>

Scores

Scores reference class.

Description

Scores reference class.

Usage

```
Scores(...)
```

Arguments

scores.csv path to comma-seperated results file, long style

Details

To instantiate this object, Scores(results.csv) is used. The results.csv is expected to have the following columns:

- *region_id* - unique numeric region identifier, reserving 0 as the region_id for the area-weighted average of the entire study area
- *goal* - the goal code or Index
- *dimension* - the dimension code, one of: status, trend, pressures, resilience, future, score
- *score* - the numeric score: 0-100 for all dimensions, except trend (-1 to 1)

Value

object (non-instantiated) reference class of Layers containing

- *long* - long view (many rows) of score results with columns: region, goal, dimension, score
- *wide* - wide view (many columns) with one row per region and columns having combination of goal and dimension

scores.Global2012.www2013

Scores resulting from the 2013 web launch applied to Global 2012 analysis.

Description

These scores are the results of the Ocean Health Index.

Format

a [Scores](#) object

References

<http://ohi-science.org>

scores.Global2013.www2013

Scores resulting from the 2013 web launch applied to Global 2013 analysis.

Description

These scores are the results of the Ocean Health Index.

Format

a [Scores](#) object

References

<http://ohi-science.org>

SelectLayers

Select a set of layers.

Description

Select a set of layers.

Usage

```
SelectLayers(object, mode = "all", cast = T,
  target = NULL, layers = NULL,
  expand.time.invariant = F,
  alternate.layer.names = NULL)
```


Arguments

object	instance of Layers class
mode	all target layers defines how to select layers
target	only needed if mode='target', specifies the target (from layers.navigation) which should be selected
layers	only needed if mode='layers', specifies the layers which should be selected. if given as a named character vector, then layers get renamed with new names as values, and old names as names per plyr::rename
alternate.layer.names	aliases for layer names
expand.time.invariant	for layers without a year column, populate the same value throughout all years where available in other layer(s)
cast	TIF whether to cast the resulting dataset, or leave it melted, defaults to TRUE

Value

data.frame with data from selected layers

SpatialSchemes	<i>SpatialSchemes reference class.</i>
----------------	--

Description

SpatialSchemes reference class.

Usage

```
SpatialSchemes(...)
```

Value

object (non-instantiated) reference class of SpatialSchemes

TransformSpatialScheme	<i>Transform data</i>
------------------------	-----------------------

Description

Transform data

Usage

```
TransformSpatialScheme(object, data, target, origin,
  categories)
```

Arguments

object	instance of SpatialSchemes class
data	data.frame such as returned from 'SelectLayers' function
target	single spatial scheme to which data should be transformed
origin	spatial schemes from which to transform, can be vector
categories	layers for which transformation should be done (to be safe, for now this should be all the layers in param data)

Value

data.frame transformed data

Index

*Topic **datasets**

layers.Global2012.Nature2012ftp,
[14](#)

layers.Global2012.www2013, [14](#)

layers.Global2013.www2013, [15](#)

scores.Global2012.www2013, [16](#)

scores.Global2013.www2013, [16](#)

CalculatePressuresComponent, [2](#)

CalculateResilienceComponent, [3](#)

CalculateStatusComponent, [3](#)

CalculateSubgoal, [4](#)

Halpern2012., [4](#)

Halpern2012.AO, [5](#)

Halpern2012.BD.HAB, [5](#)

Halpern2012.BD.SPP, [6](#)

Halpern2012.CP, [6](#)

Halpern2012.CS, [7](#)

Halpern2012.CW, [7](#)

Halpern2012.FP, [8](#)

Halpern2012.FP.FIS, [8](#)

Halpern2012.FP.MAR, [9](#)

Halpern2012.ICO, [9](#)

Halpern2012.LE, [10](#)

Halpern2012.LE.ECO, [10](#)

Halpern2012.LE.LIV, [11](#)

Halpern2012.LSP, [11](#)

Halpern2012.NP, [12](#)

Halpern2012.SP, [12](#)

Halpern2012.TR, [13](#)

Layers, [13](#), [14](#), [15](#)

layers.Global2012.Nature2012ftp, [14](#)

layers.Global2012.www2013, [14](#)

layers.Global2013.www2013, [15](#)

plyr::rename, [17](#)

Scores, [15](#), [16](#)

scores.Global2012.www2013, [16](#)

scores.Global2013.www2013, [16](#)

SelectLayers, [16](#)

SpatialSchemes, [17](#)

TransformSpatialScheme, [17](#)