

# Package ‘EcoTroph’

February 16, 2012

**Type** Package

**Title** EcoTroph modelling support

**Version** 1.2

**Date** 2012-02-02

**Author** J. Guitton and M. Colleter, D. Gascuel

**Maintainer** Jerome Guitton <jerome.guitton@agrocampus-ouest.fr>

**URL** <http://sirs.agrocampus-ouest.fr/EcoTroph/>

**Description** EcoTroph is an approach and software for modelling marine and freshwater ecosystems. It is articulated entirely around trophic levels. EcoTroph's key displays are bivariate plots, with trophic level as the abscissa, and biomass flows or related quantities as ordinate. Thus, trophic ecosystem functioning can be modelled as a continuous flow of biomass surging up the food web, from lower to higher trophic levels, because of predation and ontogenic processes. Such an approach, wherein species as such disappear, may be regarded as constituting the ultimate stage in the use of the trophic level metric for ecosystem modelling, providing a simplified but potentially useful caricature of ecosystem functioning and impact of fishing.

**License** GPL

**LazyLoad** yes

**Depends** XML

## R topics documented:

check.table . . . . .	2
create.ETdiagnosis . . . . .	3
create.ETmain . . . . .	4
create.smooth . . . . .	5
ecopath_guinee . . . . .	7
plot_ETdiagnosis . . . . .	8
plot_ETmain . . . . .	9
plot_smooth . . . . .	10
plot_Transpose . . . . .	11
read.ecopath.model . . . . .	12
saturation . . . . .	12
Transpose . . . . .	13

<b>Index</b>	<b>15</b>
--------------	-----------

---

check.table	<i>check Ecopath Table Function</i>
-------------	-------------------------------------

---

### Description

This function enables the verification of the entry table for all the EcoTroph routine. A model is available in the provided example : `data(ecopath_guinee)`.

### Usage

```
check.table(ecopath)
```

### Arguments

`ecopath` is the table returned by the Ecopath routine. The different variables are the group name, its trophic level, biomass, production and consumption ratio, catches, omnivory index and accessibility (fraction of the group that can be catch supposing an infinite fishing effort).

### Details

The user have the possibility to check its entry table with the headings and the values (no NA)

### Value

The `check.table` function returns notifications related to the `ecopath` entry table and its viability in the EcoTroph routine (problems in the headings or values)

### Author(s)

Colleter Mathieu and Guitton Jerome.

### References

Gascuel et al. (2009) EcoTroph (ET): a trophic level based software for assessing the impacts of fishing on aquatic ecosystems.

### Examples

```
data(ecopath_guinee)
ecopath_guinee
check.table(ecopath_guinee)
```

---

`create.ETdiagnosis` *ET Diagnosis*

---

## Description

ET-Transpose allows to obtain a picture of an ecosystem under a given fishing mortality. ET-Diagnosis is a routine that can simulate how this baseline ecosystem would be impacted by increasing or decreasing fishing effort. Ecosystem-wide effects of altering fishing effort include potential changes of biomass, accessible biomass, production, kinetics and catch trophic spectra as well as impacts on the mean trophic level of the catch and biomass. Additionally, ET-Diagnosis can be a useful exploratory tool for ecosystem-based management by simulating how reducing or increasing fishing effort and/or preferentially targeting different trophic levels could improve yield at the ecosystem scale. Lastly, ET-Diagnosis allows to view how different assumptions about ecosystem functioning (extent recycling, top-down effect) can have effects on both trophic level specific and ecosystem-wide properties in relation to fishing.

## Usage

```
create.ETdiagnosis(ET_Main, Mul_eff = NULL, Beta = NULL,  
TopD = NULL, FormD = NULL)
```

## Arguments

ET_Main	is the ET_Main table returned by the create_ET_Main function in its list of table.
Mul_eff	is a vector of the different effort multiplier you want to test.
Beta	is a coefficient expressing the extent of biomass input-control. Beta=0 refers to an ecosystem where all secondary production originate from grazing on primary producers and Beta=1 to an ecosystem where detritus and/or recruitment contribute a major part of the biomass input.
TopD	is a coefficient expressing the top-down control, the fraction of the natural mortality which depends on predator abundance. It may vary between 0 and 1.
FormD	is a shape parameter, varying between 0 and 1, which defines the functional relationship between prey and predators. The value 1 relates to a situation where predators abundance has a linear effect on the speed of the flow of their preys.

## Value

The function provides two types of results for each different fishing effort simulated: summary statistics for the whole ecosystem, including absolute and relative values compared to the baseline scenario; catches, total and accessible biomass and production, kinetics for all trophic levels.

## Author(s)

Colleter Mathieu and Guitton Jerome.

## References

Gascuel et al. (2009) EcoTroph (ET): a trophic level based software for assessing the impacts of fishing on aquatic ecosystems.

**See Also**

plot\_ETdiagnosis to plot the principle graphics resulting from the create.ETdiagnosis function, create.smooth function to create the Smooth, plot\_smooth to plot the smooth function, Transpose to calculate the data transposition into trophic spectra, create.ETmain to create a list of table including the ET\_Main table.

**Examples**

```
data(ecopath_guinee)
create.ETdiagnosis(create.ETmain(ecopath_guinee)$ET_Main)
create.ETdiagnosis(create.ETmain(ecopath_guinee)$ET_Main, TopD=0.6)
```

---

<code>create.ETmain</code>	<i>Create ET Main</i>
----------------------------	-----------------------

---

**Description**

This function enables the creation of the ET Main table (summarizing the principal results in function of the TL).

**Usage**

```
create.ETmain(ecopath, smooth_type=NULL, sigmaLN_cst=NULL, pas=NULL,
shift=NULL, smooth_param=NULL)
```

**Arguments**

<code>ecopath</code>	is the table returned by the Ecopath routine. The different variables are the group name, its trophic level, biomass, production and consumption ratio, catches, omnivory index and accessibility (fraction of the group that can be catch supposing an infinite fishing effort).
<code>smooth_type</code>	is a parameter of the create.smooth function. It defines the type of sigma calculation for the lognormal distribution. The value for this parameter is 1, 2 or 3. By default <code>smooth_type=1</code> , this defines a constant sigma. By putting <code>smooth_type=2</code> , the user has the possibility to put a $\sigma_{LN} = \text{smooth\_param} * \ln(TL - 0.05)$ , with the parameter <code>smooth_param=0.07</code> and <code>shift=0.95</code> by default. <code>Smooth_type=3</code> corresponds to the use of the calculated Omnivory Index (OI) as the $\sigma_{LN}$ .
<code>sigmaLN_cst</code>	is a parameter of the create.smooth function. It defines the value of the constant sigma of the lognormal distribution in case of <code>smooth_type=1</code> . By default, <code>sigmaLN_cst=0.12</code> .
<code>pas</code>	is a parameter of the create.smooth function. It defines the splitting of the TL classes.
<code>shift</code>	is a parameter of the create.smooth function. It defines the beginning of the smooth function and allows the subtraction of 0.05 in the sigma calculation accounting for the half interval range of the trophic class.
<code>smooth_param</code>	is a parameter of the create.smooth function. It defines the slope of the log-linearly increase of the TL variability with the mean trophic level of the group for <code>smooth_type=2</code> . $\sigma_{LN}(TL)$ is thus defined as $\sigma_{LN}(TL) = \text{smooth\_param} * \ln(TL - 0.05)$ .

**Value**

This function returns the ET Main table and some intermediate matrix : biomass, accessible biomass, flowP and a list of matrix corresponding to the catch of the different fisheries.

**Author(s)**

Colleter Mathieu and Guitton Jerome.

**References**

Gascuel et al. (2009) EcoTroph (ET): a trophic level based software for assessing the impacts of fishing on aquatic ecosystems.

**See Also**

plot\_ETmain to create the principle graphics resulting from the create.ETmain function, create.smooth function to create the Smooth, plot\_smooth to plot the smooth function, Transpose to calculate the data transposition into trophic spectra, plot\_Transpose to plot the trophic spectra.

**Examples**

```
data(ecopath_guinee)
create.ETmain(ecopath_guinee)
create.ETmain(ecopath_guinee, smooth_type=2)
create.ETmain(ecopath_guinee, smooth_type=3)
```

---

create.smooth

*Create Smooth Function*

---

**Description**

create.smooth is used to create a smooth function. This function enables the conversion of data pertaining to specific taxa or functional groupings into data by trophic classes. The main assumption in this Smooth function is that the distribution of the biomass or catch of a trophic group around its mean trophic level follows a lognormal curve. The curve is defined by a mean (the mean TL of a trophic group) and a standart deviation (sigma), which is a measure of the trophic level variability within the group. The distribution are then defined by the lognormal function LN(mean TL, sigma).

**Usage**

```
create.smooth(ecopath, smooth_type=NULL, sigmaLN_cst=NULL, pas=NULL,
shift=NULL, smooth_param=NULL)
```

**Arguments**

ecopath is the table returned by the Ecopath routine. The different variables are the group name, its trophic level, biomass, production and consumption ratio, catches, omnivory index and accessibility (fraction of the group that can be catch supposing an infinite fishing effort).

<code>smooth_type</code>	defines the type of sigma calculation for the lognormal distribution. The value for this parameter is 1, 2 or 3. By default <code>smooth_type=1</code> , this defines a constant sigma. By putting <code>smooth_type=2</code> , the user has the possibility to put a $\text{sigmaLN} = \text{smooth\_param} * \ln(\text{TL} - 0.05)$ , with the parameter <code>smooth_param=0.07</code> and <code>shift=0.95</code> by default. <code>Smooth_type=3</code> corresponds to the use of the calculated Omnivory Index (OI) as the <code>sigmaLN</code> .
<code>sigmaLN_cst</code>	defines the value of the constant sigma of the lognormal distribution in case of <code>smooth_type=1</code> . By default, <code>sigmaLN_cst=0.12</code> .
<code>pas</code>	defines the splitting of the TL classes. By default, <code>pas=0.1</code>
<code>shift</code>	defines the beginning of the smooth function and allows the subtraction of 0.05 in the sigma calculation accounting for the half interval range of the trophic class. By default, with a constant <code>sigmaLN</code> ( <code>smooth_type=1</code> ), <code>shift=1.8</code> ; with a function defined <code>sigmaLN</code> ( <code>smooth_type=2</code> ), <code>shift=0.95</code> ; with <code>sigmaLN=OI</code> ( <code>smooth_type=3</code> ), <code>shift=0</code> .
<code>smooth_param</code>	defines the slope of the log-linearly increase of the TL variability with the mean trophic level of the group. $\text{SigmaLN}(\text{TL})$ is thus defined as $\text{sigmaLN}(\text{TL}) = \text{smooth\_param} * \ln(\text{TL} - 0.05)$ .

### Details

The user have the possibility to define sigma for each trophic group and also adjust the LN distribution with the `smooth_type`, `sigmaLN_cst`, `smooth_param`, `shift` and `pas` parameters. Different choices are available : a constant sigma, a function defined sigma , or the sigma is equal to the omnivory index.

### Value

`create.smooth` returns a table of the TL distribution within a trophic class. This table enables the calculation of Trophic Spectra.

### Author(s)

Colleter Mathieu and Guitton Jerome.

### References

Gascuel et al. (2009) EcoTroph (ET): a trophic level based software for assessing the impacts of fishing on aquatic ecosystems.

### See Also

`plot_smooth` for plotting the smooth function, `Transpose` to calculate the data transposition into trophic spectra, `plot_Transpose` to plot the trophic spectra.

### Examples

```
data(ecopath_guinee)
create.smooth(ecopath_guinee)
create.smooth(ecopath_guinee, sigmaLN_cst=0.11)
create.smooth(ecopath_guinee, smooth_type=2)
create.smooth(ecopath_guinee, smooth_type=2, pas=0.2)
create.smooth(ecopath_guinee, smooth_type=3)
```

---

ecopath\_guinee      *EcoTroph guinean data*

---

### Description

This dataset is extract from the 2004 Guinean Ecopath model (Gascuel et al., 2009). It gives a model for the variables wanted names and table format, and the different possibilities of this package (used in function's examples).

### Usage

```
data(ecopath_guinee)
```

### Format

A data frame with 35 observations on the following 8 variables.

`group_name` name of the trophic groups used in Ecopath model. Has obligatory to be written `group_name`.

`TL` a numeric vector corresponding to the trophic level of the associated trophic groups. Has obligatory to be written `TL`.

`biomass` a numeric vector corresponding to the biomass of the associated of the trophic groups. Has obligatory to be written `biomass`.

`prod` a numeric vector corresponding to the production/biomass rate. For the Detritus groups (no P/B value needed in Ecopath), put 0 as a value. Has obligatory to be written `prod`.

`catch.1` a numeric vector corresponding to the catch of the traditional fleet. For all the groups a value must be entered, 0 if no catch are maid

`catch.2` a numeric vector corresponding to the catch of the industrial fleet. For all the groups a value must be entered, 0 if no catch are maid

`accessibility` a numeric vector corresponding to the fraction of the trophic group that can be catch supposing an infinite fishing effort. Has obligatory to be written `accessibility`.

`OI` a numeric vector corresponding to the omnivory index calculated by Ecopath for each trophic groups. Has obligatory to be written `OI`.

### Details

No NA are possible in the dataset (0 for the P/B of the detritus groups, 0 for the catch...). Follow the instructions developed in the variables descriptions. Different fleets can be entered in the model using the following system: `catch.1`, `catch.2`... If there is only one, you just have to put `catch` as a variable name.

### Source

Gascuel et al. (2009) Impact de la peche sur l'ecosysteme marin de Guinee - Modelisation EwE 1985/2005 -

### References

Gascuel et al. (2009) Impact de la peche sur l'ecosysteme marin de Guinee - Modelisation EwE 1985/2005 - Gascuel et al. (2009) EcoTroph (ET): a trophic level based software for assessing the impacts of fishing on aquatic ecosystems.

**Examples**

```
data(ecopath_guinee)
ecopath_guinee
names(ecopath_guinee)
```

---

plot\_ETdiagnosis    *Plot Diagnosis Function*

---

**Description**

This function enables the creation of the principle graphics resulting from the create.ETdiagnosis function.

**Usage**

```
plot_ETdiagnosis(m, scale1, scale2, maxrange)
```

**Arguments**

m	is the list of tables returned by the create.ETdiagnosis function.
scale1	is the scale parameter of the Biomass Trophic Spectra, can be log or by default the standart scale of the results.
scale2	is the scale parameter of the Flow Trophic Spectra, can be log or by default the standart scale of the results.
maxrange	is the max Trophic Level for the x-axis of the plots.

**Details**

The scale1 parameter controls the scale of the Yaxis for the BTS, it's by default the scale of the result but it's normally more practical to use a log scale. It's the same thing for the scale2 parameter controlling the Yaxis of the FTS.

**Value**

The function returns the principal graphics of the global ET-Diagnosis routine: the graphics of the biomass, accessible biomass...rates for the different effort multipliers, the Biomass Trophic Spectra (BTS)for the different effort multipliers, the B/Bref(mE=1) graph for the main TL classes, the CTS and FTS.

**Author(s)**

Colleter Mathieu and Guitton Jerome.

**References**

Gascuel et al. (2009) EcoTroph (ET): a trophic level based software for assessing the impacts of fishing on aquatic ecosystems.

**Examples**

```
data(ecopath_guinee)
plot_ETdiagnosis(create.ETdiagnosis(create.ETmain(ecopath_guinee)$ET_Main))
```

---

plot_ETmain	<i>Plot ET Main</i>
-------------	---------------------

---

### Description

This function enables the creation of the principle graphics resulting from the create.ETmain function.

### Usage

```
plot_ETmain(m)
```

### Arguments

`m` is the list of tables returned by the create\_ET\_Main function.

### Value

The function returns the principal graphics of the global ET-Transpose routine: the Biomass Trophic Spectra, the Accessible Biomass Trophic Spectra and other graphics notably the Catch Trophic Spectra for the fisheries in input.

### Author(s)

Colleter Mathieu and Guitton Jerome.

### References

Gascuel et al. (2009) EcoTroph (ET): a trophic level based software for assessing the impacts of fishing on aquatic ecosystems.

### See Also

create.smooth function to create the Smooth, plot\_smooth to plot the smooth function, Transpose to calculate the data transposition into trophic spectra, create.ETmain to create a list of table including the ET\_Main table.

### Examples

```
data(ecopath_guinee)
plot_ETmain(create.ETmain(ecopath_guinee))
```

---

plot_smooth	<i>Plot Smooth Function</i>
-------------	-----------------------------

---

### Description

plot\_smooth is used to plot a smooth function. This function enables the user to see the TL distribution around its mean trophic level.

### Usage

```
plot_smooth(tab_smooth)
```

### Arguments

tab\_smooth is the table returned by the create.smooth function.

### Value

The function returns a plot of the Smooth with the TL classes in abscisse.

### Author(s)

Colleter Mathieu and Guitton Jerome.

### References

Gascuel et al. (2009) EcoTroph (ET): a trophic level based software for assessing the impacts of fishing on aquatic ecosystems.

### See Also

create.smooth function to create the Smooth, Transpose to calculate the data transposition into trophic spectra, plot\_Transpose to plot the trophic spectra.

### Examples

```
data(ecopath_guinee)
plot_smooth(create.smooth(ecopath_guinee))
plot_smooth(create.smooth(ecopath_guinee, smooth_type=2))
```

---

plot\_Transpose      *Plot Transpose Function Results*

---

### Description

plot\_Transpose returns the principal graphics of the Transpose function : the different Trophic Spectra (CTS, BTS...) and other graphic representations.

### Usage

```
plot_Transpose(tab_Trans, scale = NULL, title = NULL)
```

### Arguments

tab\_Trans      is the table returned by the Transpose function.  
scale            defines the scale of the graphic y axis : can be log or not.  
title            defines the title of the graph.

### Value

The function returns the principal graphics of the Transpose function : the different Trophic Spectra (CTS, BTS...) and other graphic representations.

### Author(s)

Colleter Mathieu and Guitton Jerome.

### References

Gascuel et al. (2009) EcoTroph (ET): a trophic level based software for assessing the impacts of fishing on aquatic ecosystems.

### See Also

create.smooth function to create the Smooth, plot\_smooth to plot the smooth function, Transpose to calculate the data transposition into trophic spectra.

### Examples

```
data(ecopath_guinee)
plot_Transpose(Transpose(create.smooth(ecopath_guinee), ecopath_guinee, "biomass"), log)
plot_Transpose(Transpose(create.smooth(ecopath_guinee), ecopath_guinee, "catch.1")
, title="Small Scale Fishery Catch")
```

---

```
read.ecopath.model read.ecopath.model Function
```

---

### Description

This function read inputs datas from an xml file exported from Ecopath with the Ecotroph plugin or from a web service providing inputs for EcopTroph coming from a database populate with parameters of several Ecopath's models.

### Usage

```
read.ecopath.model(filename)
```

### Arguments

`filename` is the name of the file exported by ecopath.

### Value

This function return a data frame containing all the column needed by Ecotroph as input.

### Author(s)

Colleter Mathieu and Guitton Jerome.

### References

Gascuel et al. (2009) EcoTroph (ET): a trophic level based software for assessing the impacts of fishing on aquatic ecosystems.

### See Also

`create.smooth` function to create the Smooth, `plot_smooth` to plot the smooth function, `Transpose` to calculate the data transposition into trophic spectra, `plot_Transpose` to plot the trophic spectra.

---

```
saturation Sigma Saturation Function
```

---

### Description

This function enables an other calculation for the sigma of the `create.smooth` function. Sigma is calculated on the base of a saturation function reflecting a biological reasoning about the variability of the TL within the trophic classes: The variability increases with the TL and reaches a plateau after a certain TL.

### Usage

```
saturation(sigma_inf = NULL, coeff = NULL, pas = NULL)
```

**Arguments**

`sigma_inf` defines the value of the curve's plateau.  
`coeff` defines the value of the slope.  
`pas` defines the splitting of the TL classes.

**Details**

By default sigma is calculated using a lognormal function. This function enables an other user choice reflecting a different reasoning.

**Value**

saturation returns a vector of values for the sigma used in the create.smooth function.

**Author(s)**

Colleter Mathieu and Guitton Jerome.

**References**

Gascuel et al. (2009) EcoTroph (ET): a trophic level based software for assessing the impacts of fishing on aquatic ecosystems.

**See Also**

create.smooth function to create the Smooth, plot\_smooth to plot the smooth function, Transpose to calculate the data transposition into trophic spectra, plot\_Transpose to plot the trophic spectra.

**Examples**

```
plot(saturation())
lines(saturation(0.2))
text(48,0.18,"sigma_inf=0.2")
lines(saturation(coeff=0.5))
text(48,0.35,"coeff=0.5")
```

---

Transpose

*ET Transpose Function*

---

**Description**

Transpose enables the conversion of data pertaining to specific taxa or fonctionnal groupings into data by trophic classes. Data can be catches, biomasses or production in order to produce continuous distributions of those variables over trophic levels.

**Usage**

```
Transpose(tab_smooth, ecopath, column)
```

**Arguments**

<code>tab_smooth</code>	is the table returned by the <code>create.smooth</code> function.
<code>ecopath</code>	is the table returned by the <code>Ecopath</code> routine. The different variables are the group name, its trophic level, biomass, production and consumption ratio, catches, omnivory index and accessibility (fraction of the group that can be catch supposing an infinite fishing effort).
<code>column</code>	is the table <code>ecopath</code> column name of the variable you want to transpose (for example "biomass" or "catch").

**Value**

The function returns a table of the choosen variable split into the defined trophic classes per `ecopath` group.

**Author(s)**

Colleter Mathieu and Guitton Jerome.

**References**

Gascuel et al. (2009) `EcoTroph (ET)`: a trophic level based software for assessing the impacts of fishing on aquatic ecosystems.

**See Also**

`create.smooth` function to create the `Smooth`, `plot_smooth` to plot the smooth function, `plot_Transpose` to plot the trophic spectra.

**Examples**

```
data(ecopath_guinee)
Transpose(create.smooth(ecopath_guinee), ecopath_guinee, "biomass")
Transpose(create.smooth(ecopath_guinee), ecopath_guinee, "catch.1")
```

# Index

- \*Topic **\textasciitildeDiagnosis**
  - create.ETdiagnosis, 3
  - plot\_ETdiagnosis, 8
- \*Topic **\textasciitildeEcoTroph**
  - create.ETdiagnosis, 3
  - create.ETmain, 4
  - create.smooth, 5
  - plot\_ETdiagnosis, 8
  - plot\_ETmain, 9
  - plot\_smooth, 10
  - plot\_Transpose, 11
  - read.ecopath.model, 12
  - saturation, 12
  - Transpose, 13
- \*Topic **\textasciitildeSmooth**
  - create.smooth, 5
  - plot\_smooth, 10
  - saturation, 12
- \*Topic **\textasciitildeTranspose**
  - create.ETmain, 4
  - plot\_ETmain, 9
- \*Topic **datasets**
  - ecopath\_guinee, 7

check.table, 2

create.ETdiagnosis, 3

create.ETmain, 4

create.smooth, 5

ecopath\_guinee, 7

plot\_ETdiagnosis, 8

plot\_ETmain, 9

plot\_smooth, 10

plot\_Transpose, 11

read.ecopath.model, 12

saturation, 12

Transpose, 13