Blue text on a black background

Description automatically generated

**Manual: Identifying**

**Key Forage Species**

December 2024

Marjoleine M.H. Roos

Quantitative Marine Ecologist

**Table of Contents**

[Introductory summary 3](#_Toc155023741)

[1. Glossary 3](#_Toc155023742)

[2. Background information and rationale 3](#_Toc155023743)

[2.1. The Connectance Index 4](#_Toc155023744)

[2.2. The Mass SURF Index 4](#_Toc155023745)

[2.3. The Energy SURF Index 4](#_Toc155023746)

[3. Methods 5](#_Toc155023747)

[3.1. Step 1: Data sourcing 5](#_Toc155023748)

[3.2. Step 2: Data processing 6](#_Toc155023749)

[3.3. Step 3: Deciding on functional group aggregation 7](#_Toc155023750)

[3.4. Step 4: Running automated index calculations 8](#_Toc155023751)

[Literature Cited 12](#_Toc155023754)

[Appendices 13](#_Toc155023755)

[Appendix 1. R code used to run automated index calculations 13](#_Toc155023756)

[Appendix 2. Overview Ecosystem Information 23](#_Toc155023757)

[Appendix 3. Overview Ecosystem Contact Details 26](#_Toc155023758)

## Introductory summary

This manual is a guide for the identification of key forage species concerning their trophic linkages within an ecosystem’s food web. This is done by calculating three published indices, including the connectance index and the SUpportive Role to Fishery ecosystems (SURF) index calculated from mass-balanced models, and the SURF index calculated from energy-balanced models. Hereafter, these three indices are referred to as ‘the three indices’. All use food web data from existing static ecosystem food web models. The index calculations are automated using R®; steps explained in this manual are linked to the R code by the indication of the R code line number where relevant.

The first section of this manual is a glossary of key terms used throughout. The second section explains the background and rationale behind the index calculations. The third section contains the methods in four main steps.

All datasets and code referenced here are publicly available in a Github repository at <https://github.com/MontereyBayAquarium/SFW_foragefish>. The appendices of this manual also include the complete R code (including annotations), an overview of the information on the ecosystems that have been analyzed (as of December 2024) and contact information for included ecosystems where relevant.

The full report detailing findings of the project associated with this manual can be found at [URL].

## Glossary

**Aggregation**: Merging two or more functional groups within a food web model into one new functional group.

**Connectance index**: An index considering the number of trophic linkages of a species/group in a mass-balanced food web model relative to the total number of trophic linkages existing in the food web model (Smith et al., 2011).

**Diet composition data**: The composition (in proportions) of the diet of each functional group in a food web model presented as a diet matrix.

**Diet matrix**: A matrix including the diet composition (in either mass or energy) of all functional groups existing in a food web model.

**Ecopath model**: Model used to predict and simulate the dynamics (e.g., spatial and temporal) of marine and other aquatic ecosystems (Walters et al., 1997; Christensen et al., 2014).

**Energy SURF index**: An index considering the number of trophic linkages of a species/group in an energy-balanced food web model and the importance of these linkages relative to the total number of trophic linkages existing in the food web model (Surma et al., 2022).

**Food web model**: A model representing the trophic interactions and energy flow between species within an ecosystem.

**Functional group**: A single species or various trophically similar species representing a food web compartment.

**Key forage species:** A forage species that has a vital role in an ecosystem’s food web structure (species trophic interactions) and ultimately the ecosystem’s functioning .

**Mass SURF index**: An index considering the number of trophic linkages of a species/group in a mass-balanced food web model and the importance of these linkages relative to the total number of trophic linkages existing in the food web model (Plagányi and Essington, 2014).

## Background information and rationale

Whilst all forage species are important both due to their high economic value to fisheries and as the bulk of prey within an ecosystem, some forage species are more important than others and are key to sustaining an ecosystem’s food web structure and, eventually, ecosystem functioning. Identifying such ‘key’ forage species is thus vital for establishing effective ecosystem-based fisheries management and re-evaluating existing harvest strategies to prevent stock collapse and cascading ecosystem effects.

There exist three methods to calculate keyness indices to identify key forage species, which are shortly introduced and explained here.

### 2.1. The Connectance Index

Smith et al. (2011) introduced the connectance index, which represents the connectance of a species/group in a food web model relative to the complexity of the food web model. This index is calculated as the total number of links involving a species/group as both prey and predator divided by the total number of links in a food web model (Smith et al., 2011). Smith et al. (2011) suggested that a connectance above 0.04 represents a key species. The connectance index treats all trophic links within a food web the same, ignoring the importance of the trophic link between predators and prey and is sensitive to the aggregation level of species in a food web model (Plagányi and Essington, 2014).

### 2.2. The Mass SURF Index

Plagányi and Essington (2014) developed the SUpportive Role to Fishery ecosystems (SURF) index for mass-balanced food web models, hereafter referred to as the ‘mass SURF index’. In addition to the number of predator species dependent on a forage species, this index considers the level of dependency of a predator species on a forage species (Plagányi and Essington, 2014). The mass SURF index for prey species/group *i* with *S* predators is calculated as:

Where, *i* is the prey species/group, *j* is the predator of the prey species, *S* is the total number of predators that a prey species/group is predated by, *p*ij is the diet fraction for which predator *j* relies on prey species/group *i*, and *L* is the total number of linkages (with *p*ij>0) in the ecosystem food web.

Relying on empirical diet data, a high diet fraction regarding mass indicates a high dependence by a predator on a forage species (Plagányi and Essington, 2014). Furthermore, the weight of dependence increases non-linearly to distinguish between predators that are highly dependent on a forage species (diet fraction close to 1) and predators that rely minimally on a forage species (diet fraction close to 0). In contrast to the connectance index, the mass SURF index is robust to predator species’ aggregation.

The mass SURF index is a similar calculation as for the connectance index, yet the connectance calculation merely considered *pij* to be either 0 (the prey species is not consumed by a predator) or 1 (the prey species is consumed by a predator), whilst the SURF index considered the importance of a forage species to a predator by representing a proportion of the predator's diet in mass represented by a forage species as a value between 0 and 1. Plagányi and Essington (2014) determined that prey species with a SURF above 0.001 are deemed as key in an ecosystem’s food web. The mass SURF index is affected by the aggregation of forage species (Plagányi and Essington, 2014).

### 2.3. The Energy SURF Index

The calculation of the energy SURF index is like that of the mass SURF index, only it considers predators’ diet composition according to energy rather than mass (Surma et al., 2022). This method considers that the importance of the role of forage fish species in an ecosystem can be reinforced by their potentially elevated energy contents compared to other prey species in an ecosystem. The actual consumption of each predator species/group *j* on a prey species/group *i* is multiplied by the energy density of prey species/group *i*. Otherwise, it follows the same method as the mass SURF index and the same threshold to identify key species (Surma et al., 2022). For more details, see Surma et al. (2022). Energy scaling methods are explained by Surma et al., (2018). Given the nature of the calculation of the energy SURF index, it is expected that, like the mass SURF index, this index is influenced by the aggregation of forage species. However, this has not been tested yet.

Goal of analyses

The connectance index and two SURF indices are calculated to score and determine the keyness of (forage) species within an ecosystem. When a species’ index exceeds the threshold set for the three indices (particularly the two SURF indices), the species can be considered to be key in the ecosystem’s food web, suggesting that it is of importance for the ecosystem’s general structure and functioning.

Whether a forage species holds a key position in its ecosystem’s food web (either through mass or energy) can be used to re-evaluate the species’ existing harvest strategies and establish more conservative reference points to prevent significant impacts on the ecosystem through fishing. Furthermore, the results can be applied to adjust the scoring of sustainable consumption regarding forage species.

## Methods

This section provides a step-by-step guide for calculating the three existing indices introduced in section 1 of this manual. The process can be divided into 4 main steps, including:

1. Data sourcing
2. Determining functional group aggregation
3. Data processing
4. Data analyses

The R code (Appendix 1) automatically calculates the three indices. When referring to a step in the R code, the relevant code lines are indicated in orange font. The R code itself includes a detailed description of the purpose of each code line.

### Step 1: Data sourcing

The first step for key forage species identification according to the three indices concerns sourcing the required data that are needed to calculate the indices from existing literature.

The following data are required for all species/guilds in an ecosystem’s food web:

1. **Diet composition data**:

Units: proportion of the complete diet of each functional group by means of mass balance.

These data are presented as a diet matrix with all species or guilds occurring in an ecosystem’s food web (i.e., functional groups) shown as prey in rows and as predators in columns. The sum of each column in the diet matrix should be 1. The diet matrix can also include the functional groups “Import” (migration of species/guilds into the ecosystems), “Discards” (regarding fisheries), and “Bycatch” (regarding fisheries). If the published diet matrix includes diet composition data resolved by balancing the food web model, these should be used.

1. **Biomass data**:

Units: tonnes per squared kilometers (t/km2)

These data are used in Ecopath models as model input for all species/guilds in an ecosystem’s food web model and are generally presented in a table as one of the model parameters. If the published biomass data includes data resolved by balancing the food web model, these should be used.

1. **Consumption per biomass**:

Units: consumption to biomass ratio per year

These data are also used in Ecopath models as model input for all species/guilds in an ecosystem’s food web and are generally presented as one of the model parameters in the same table as the biomass data.

1. **Energy density** (ED):

Units: kilo joules per gram (kJ/g) wet mass

These data are generally not used in Ecopath models. Ideally, a published species ED value in the relevant ecosystem (or ocean/sea) is applied. The ED data for all species for the different ecosystems/waters that have been assessed for the Forage Fish project so far have been collected in an **Excel database** ( NAMEXX). If they do not yet exist in this database, they must be sourced from existing literature. If such information is not available, the ED value of the same species in a different ecosystem can be used. If those data are neither available, the ED value of comparable species, family, or order can be used.

When it concerns a prey group including multiple species, an average ED value of those different species should be applied when these species and their ED values are known (or for the species’ family or order). For calculating the Energy SURF index, constant ED values of adult specimens are considered (i.e., not fluctuating with season or age).

Note, that data suitable for the calculation of the three indices must be taken from Ecopath models meeting the following requirements:

* Consider a complete ecosystem food web, including species/guilds of the lowest and highest trophic levels.
* Include forage species as separate species, not as guilds or aggregated groups (see explanation in section 3.3).
* Ideally represent a food web during which (different) forage species were abundant (see explanation in section 4)

The sources of the data used for the different ecosystems assessed for the Forage Fish Project are shown in the table in Appendix 2.

### Step 2: Determining functional group aggregation

All food web models are unique in how and at what level they organize functional groups in the diet matrix. Most models consider a rather general level of species/groups for the lowest (e.g., microzooplankton, mesozooplankton, microzooplankton) and highest trophic level species/groups (e.g., seabirds, cetaceans, pinnipeds, apex predatory fish). However, depending on their purpose, some can consider a rather detailed level of species/groups regarding these low and high trophic levels (e.g., considering different species/groups within mesozooplankton or species/groups of seabirds). For the identification of key *forage* species according to the indices, such level of grouping is not necessary, plus the index value for models that include such detailed level of grouping for low and high trophic levels cannot be compared to those that consider a more general level grouping. Additionally, different age/size classes considered for the same species/group, which generally only exist for one or some species/groups of interest within a model, should be aggregated for the identification of keyness per species.

Here, aggregation refers to merging two or more functional groups within the food web model into one new functional group. This requires recalculating the new group’s diet composition and the proportion of the diet that it represents as prey to predators using methods by Gaichas et al. (2009). The consideration of a somewhat uniform organization of functional groups for the so-called ‘aggregated model’ among the different ecosystems makes their index results more comparable. Still, if relevant, it is recommended to calculate the three indices for both the original, or so-called ‘disaggregated’, model and the aggregated model to consider how the aggregation of groups may affect the three indices.

1. Decide whether model aggregation is necessary by inspecting the level of functional groups for the low and high trophic levels:
   * Appropriate level (see explanation above) 🡪 model aggregation not necessary, continue to step 4.
   * Too detailed (see explanation above) 🡪 model aggregation recommended
2. If model aggregation is relevant, decide on which low and high trophic functional groups or size/age class groups should be aggregated and the names of the new aggregated functional group. New names of the aggregated groups need to be included in the parameter table as described in step 3:
   * Although the functional groups included in the final diet matrix will always slightly differ between ecosystems, the following low and high trophic functional groups are often relevant:
     + - * Phytoplankton
         * Microzooplankton
         * Mesozooplankton
         * Macrozooplankton
         * Macrobenthos
         * Crustaceans
         * Demersal piscivorous fish
         * Other demersal fish
         * Apex predatory fish
         * Cetaceans
         * Pinnipeds
         * Seabirds

Note: forage species should not be aggregated as this would affect their SURF index (Plagányi and Essington, 2014) and likely their other indices as well.

### Step 3: Data processing

Step 3 concerns processing the data that has been acquired in step one to be used for the automated analyses. Because the analyses are automated, it is of great importance to be consistent in the use of names, terms, order of groups, spelling, capital use, etc.

1. **Create a folder** with the name of the ecosystem (e.g., California Current). Navigate to the following URL to clone the Github repository that contains data and code described in this manual: <https://github.com/MontereyBayAquarium/SFW_foragefish/blob/main/index_calculations_2024.R>. The repository contains a folder titled data/raw where new Excel files can be added as described in the next steps (2-5)

1. **Create an Excel file** with the diet composition data in the form of a matrix (i.e., diet matrix) with
   * functional groups as prey in column 1 with each group starting a separate row, and
   * the same functional groups as predator in row 1 with each group starting a separate column.
   * cell [1,1] empty.
   * each group name starting with a capital letter,
   * and no NA’s (i.e., if a prey is not consumed by a predator, the relevant cell should contain a “0” and should not be left empty).

Note 1: The diet matrix can be imported into Excel from an PDF-file by: Data 🡪 Get Data 🡪 From File 🡪 From PDF. Then select the PDF-file (paper containing the data) and select the page that contains the diet matrix. Always check that each column totals 1.

Note 2: Make sure that the exact same groups are included as both prey (rows) and predators (columns); in some published diet matrices, groups that are not consumers are not included as predators, if so, these need to be added.

Note 3: Make sure that numbers are entered as numbers and not text.

1. **Save this Excel file** as “diet\_geography.xlsx” (e.g., diet\_CaliforniaCurrent.xlsx) in the data/raw folder.
2. **Create a new Excel file** with a table of the group parameters with
   * the first column named “group.no” including the group numbers in order
   * the second column named “group.name” including all group names,
   * the third column called “B” including biomass data,
   * the fourth column called “QB” with consumption per biomass per group, and
   * the fifth column called “ED” with the energy density for each group. It is recommended to add one last column “ED.source” with reference to the energy contents source.

In case of aggregation:

* + the sixth column called “agg.group.name” including the original group names for groups that are not aggregated and the new group names for groups that are aggregated,
  + the seventh column called “group.code” including the original group code (number) for each group.

Note 1: Make sure that the group parameters for all functional groups are in the same order as the diet matrix. If “Import”, “Discards”, or “Bycatch” are included in the diet matrix make sure these groups are also included in this table with group parameters.

Note 2: If a functional group (other than “Import”, “Discards”, or “Bycatch”) has an assigned biomass of “0”, remove this group from the diet matrix and parameter table as this will produce errors in the indices’ calculations

Note 3: Again, make sure that numbers are entered as numbers and not text.

1. **Save this Excel file** as “groupdata\_geography.xlsx” (e.g., groupdata\_CaliforniaCurrent.xlsx) in the data/raw folder.

### Step 4: Running automated index calculations

Here, the general steps to calculate the three indices using the R-code (see Appendix 2) are explained with references to the relevant R-code lines (visible in R only).

1. Load packages that are needed to run the functions used in the R-code.

Note: if a package is not installed, this needs to be done first.

**CODE LINES: 7-11**

1. Set the working directory as the path to the folder from which data will be imported (data/raw) and where output files will be saved (data/output).

**CODE LINES: 17-27**

1. Open a loop that allows the code to run for all geographies represented in the data/raw folder. This outermost for-loop is closed in line 439.

**CODE LINE: 34**

1. Import data, including the ‘diet\*’- and ‘groupdata\*’-Excel files and use these data to define the number of functional groups in the disaggregated model and geography.  
   **CODE LINES: 35-38**

##### Step 4.1: Running automated index calculations for the disaggregated model

Create the parameters and diet matrices needed to calculate the three indices.  
**CODE LINES: 44-73**

1. Count trophic linkages for both mass- and energy- diet matrix.

**CODE LINES: 79-101**

Note: For calculating the three indices, food web linkages of Import, Discards, Bycatch, etc. are not considered as they are not part of the (natural) closed ecosystem and are therefore not included in the sum of linkages (L) to calculate indices.

1. Calculate the three indices and determine keyness for each functional group.

**CODE LINES: 103-148**

1. Process results and save them as an Excel-file in the data/output folder.

**CODE LINES: 150-170**

Note: If a forage species of interest in the disaggregated model was sub-classed by age (or other factor) it was defined as key if one of the sub-classes was identified as key by the index.

##### Step 4.2: Running automated index calculations for the aggregated model (if required)

1. Create a new vector for the energy consumption per biomass (QBE) for each functional group in the disaggregated model as its total energy consumption (calculated in lines 47-51) divided by its biomass.

**CODE LINES: 180-182**

1. Create a function (‘aggregate.groups’) for aggregating groups in both the mass- and energy-balanced diet matrix to calculate the parameters for the newly create groups, including biomass (B), consumption per biomass (QB), consumption per biomass in energy (QBE), group names, mass-balanced diet matrix (p), and energy-balance diet matrix (pE). These new parameters are created by the function and presented as a list called ‘Agg.model’.

**CODE LINES: 185-256**

1. Define the names of the new aggregated groups and original groups that need to be aggregated into those new groups (as decided in step 2).

**CODE LINES: 258-295**

1. Aggregate groups and run the function to create parameters and diet matrices regarding newly created groups needed to calculate the three indices regarding both a mass-balanced and energy-balanced model.

**CODE LINES: 298-337**

1. Count trophic linkages for both mass- and energy- diet matrix

**CODE LINES: 343-365**

1. Calculate the three indices and determine keyness for each functional group in the aggregated model.

**CODE LINES: 368-413**

1. Process results and save them as an Excel-file in the data/output folder.

**CODE LINES: 415-437**

## Literature Cited

Christensen, V., M. Coll, J. Steenbeek, J. Buszowski, D. Chagaris, and C. J. Walters. 2014. Representing variable habitat quality in a spatial food web model. Ecosystems 17(8):1397-1412.

Gaichas, S., G. Skaret, J. Falk-Petersen, J.S. Link, W. Overholtz, B.A. Megrey, H. Gjøsæter, W.T. Stockhausen, A. Dommasnes, K.D. Friedland, K. Aydin. 2009. A comparison of community and trophic structure in five marine ecosystemsbased on energy budgets and system metrics. Progress in Oceanography 81 (1-4): 47–62.

Koehn, L. E., T. E. Essington, K. N. Marshall, I. C. Kaplan, W. J. Sydeman, A. I. Szoboszlai, and J. A. Thayer. 2016. Developing a high taxonomic resolution food web model to assess the functional role of forage fish in the California Current ecosystem. Ecological Modelling 335:87-100. doi: 10.1016/j.ecolmodel.2016.05.010

Plagányi, É. E., and T. E. Essington. 2014. When the SURFs up, forage fish are key. Fisheries Research 159:68-74. doi: 10.1016/j.fishres.2014.05.011

Smith, A. D., C. J. Brown, C. M. Bulman, E. A. Fulton, P. Johnson, I. C. Kaplan, H. Lozano-Montes, S. Mackinson, M. Marzloff, and L. J. Shannon. 2011. Impacts of fishing low–trophic level species on marine ecosystems. Science 333(6046):1147-1150.

Surma, S., Pakhomov, E. A., Pitcher, T. J., 2018. Energy-based ecosystem modelling illuminates the ecological role of Northeast Pacific herring. Marine Ecology-Progress Series 588, 147–161. <https://doi.org/10.3354/meps12430>.

Surma, S., E. A. Pakhomov, and T. J. Pitcher. 2022. Pacific herring (*Clupea pallasii*) as a key forage fish in the southeastern Gulf of Alaska. Deep Sea Research Part II: Topical Studies in Oceanography 196:105001. doi: 10.1016/j.dsr2.2021.105001

Walters, C., V. Christensen, and D. Pauly. 1997. Structuring dynamic models of exploited ecosystems from trophic mass-balance assessments. Reviews in Fish Biology and Fisheries 7:139–172.

## Appendices

### Appendix 1. R code used to run automated index calculations

#==============================================================================

# CALCULATING THE CONNECTANCE, MASS SURF, AND ENERGY SURF INDEX

#==============================================================================

# LOAD PACKAGES, PACKAGES MAY NEED TO BE INSTALLED FIRST

library(openxlsx)

library(readxl)

library(plyr)

library(here)

library(stringr)

options(scipen=999) # PREVENT R FROM USING SCIENTIFIC NOTATION

#======= IMPORT DATA ==========================================================

# SET DIRECTORIES AND LOAD DATA

directory <- here::here("data","raw")

output <- here::here("data","output")

dietfiles = list.files(path=directory, pattern="diet\*", full.names=TRUE)

diet.list = lapply(dietfiles, read\_excel)

names(diet.list) = str\_match(dietfiles, "diet\_\\s\*(.\*?)\\s\*.xlsx")[,2]

groupdatafiles = list.files(path=directory, pattern="groupdata\*", full.names=TRUE)

groupdata.list = lapply(groupdatafiles, read\_excel)

names(groupdata.list) = str\_match(groupdatafiles, "groupdata\_\\s\*(.\*?)\\s\*.xlsx")[,2]

rm(dietfiles, groupdatafiles)

#=============SET UP FOR LOOP TO RUN ALL GEOGRAPHIES===========================

for (geog in 1:length(diet.list)) {

diet.mat <- as.data.frame(diet.list[[geog]], col.names=names(x)) # IMPORT AND CREATE DATA FRAME FROM DIET MATRIX

group.data <- as.data.frame(groupdata.list[[geog]], col.names=names(x)) # IMPORT AND CREATE TABLE FROM GROUP PARAMETERS AS USED IN THE ECOPATH MODEL (FUNCTIONAL GROUP NAMES, BIOMASS (B), AND CONSUMPTION/BIOMASS (QB)) IN COMBINATION WITH DIET DATA

n.groups=dim(group.data)[1] # SET NUMBER OF GROUPS

geography=names(diet.list[geog]) # SET GEOGRAPHY NAME FOR NAMING OUTPUT FILES

#==============================================================================

# STEP 4.1. DISAGGREGATED MODEL

#==============================================================================

#======= STEP 4.1.1. CREATE PARAMETERS AND DIET MATRICES ======================

B=as.matrix(as.numeric(group.data$B)) # CREATE VECTOR FOR BIOMASS (B) VALUES PER GROUP IN T/KM2

QB=as.matrix(as.numeric(group.data$QB)) # CREATE VECTOR FOR CONSUMPTION (Q) PER BIOMASS (B) PER GROUP IN T/YEAR

group.names=as.vector(unlist(group.data$group.name)) # CREATE VECTOR OF GROUP NAMES

ED=as.matrix(group.data$ED) # CREATE VECTOR FOR ENERGY DENSITY (ED) VALUES PER GROUP IN KCAL/G

p=as.matrix(diet.mat[,2:(n.groups+1)]) # CREATE DATA FRAME OF ONLY DIET COMPOSITION (pij) VALUES EXTRACTED FROM DIET MATRIX (IGNORING FIRST COLUMN WITH GROUP NAMES)

CTOT=c(B\*QB) # CREATE VECTOR FOR ACTUAL TOTAL CONSUMPTION IN Tonnes PER GROUP AS PRODUCT OF BIOMASS AND CONSUMPTION/BIOMASS

na.index=which(is.na(CTOT)==TRUE) # CHECK FOR GROUPS WITHOUT CONSUMPTION (E.G., PHYTOPLANKTON AND DETRITUS)

CTOT[na.index]=rep(0,length(na.index)) # FOR THESE GROUPS, SET CONSUMPTION TO BE "0"

CTOT<-t(CTOT) # TRANSPOSE VECTOR FROM COLUMN TO ROW

CTOT <- as.data.frame(lapply(CTOT, rep, length(group.names))) # CREATE DATA FRAME OF TOTAL CONSUMPTION PER GROUP BY REPLICATING THIS ROW AS MANY TIMES AS THERE ARE GROUPS

C=p\*CTOT # CREATE DATA FRAME OF ACTUAL CONSUMPTION OF EACH PREY (ROW) BY EACH PREDATOR (COLUMN) AS PRODUCT OF TOTAL CONSUMPTION AND PIJ

colnames(C)=colnames(p) # GIVE THE DATA FRAME THE COLUMN HEADERS OF GROUP NAMES

EDmat <- t(sapply(ED, rep, length(group.names))) # CREATE DATA FRAME OF ENERGY DENSITY PER GROUP BY REPLICATING THIS COLUMN AS MANY TIMES AS THERE ARE GROUPS

EC= C\*EDmat # CREATE DATA FRAME OF ACTUAL ENERGY CONSUMPTION OF EACH PREY (ROW) BY EACH PREDATOR (COLUMN) AS PRODUCT OF CONSUMPTION AND ENERGY DENSITY

ECTOT=colSums(EC) # GIVES SUM OF ENERGY CONSUMPTION FOR EVERY COLUMN (PREDATOR)

ECTOT<-t(ECTOT) # CONVERT COLUMN TO ROW

ECTOT<- as.data.frame(lapply(ECTOT, rep, length(group.names))) # REPLICATE ROW FOR TOTAL NUMBER OF GROUPS

colnames(ECTOT) <- c(1:length(group.names)) # change column names to group numbers

pE<-EC/ECTOT # CREATE P (DIET PROPORTION) FOR ENERGY

pE <- replace(pE, is.na(pE), 0) # FOR GROUPS OF WHICH EC/ECpredtot = NA (I.E. NON-CONSUMERS SUCH AS PHYTOPLANKTON), SET CONSUMPTION TO BE "0"

pE<-as.matrix(pE) # convert energy diet matrix from data frame to a matrix for further calculations

SUMpE<-colSums(pE) # CHECK WHETHER ALL COLUMNS SUM UP TO 1

#==============================================================================

# CALCULATE THE 3 INDICES FOR DISAGGREGATED MODEL

#==============================================================================

#======= STEP 4.1.2. COUNT TROPHIC LINKAGES FOR ALL GROUPS ====================

S<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR S (# OF TROPHIC LINKAGES IN MASS DIET MATRIX FOR A GROUP AS PREY WITH OTHER GROUPS AS PREDATORS)

S2<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR S2 (TOTAL TROPHIC LINKAGES AS S + # OF TROPHIC LINKAGES IN MASS DIET MATRIX FOR A GROUP AS PREDATOR WITH OTHER GROUP AS PREY)

ES<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR ES (# OF TROPHIC LINKAGES IN ENERGY DIET MATRIX FOR A GROUP AS PREY WITH OTHER GROUP AS PREDATORS)

for (i in 1:n.groups) # RUN A LOOP WITH NUMBER OF RUNS (i) EQUAL TO NUMBER OF GROUPS

{ # OPEN LOOP

S=rbind(S,length(which(p[i,]>0))) # POPULATE VECTOR S WITH VALUE BEING # OF TROPHIC LINKAGES IN MASS DIET MATRIX (p) FOR GROUP i AS PREY WITH OTHER GROUP AS PREDATORS GREATER THAN 0

ES=rbind(ES,length(which(pE[i,]>0))) # POPULATE VECTOR ES WITH VALUE BEING # OF TROPHIC LINKAGES IN ENERGY DIET MATRIX (pE) FOR GROUP i AS PREY WITH OTHER GROUP AS PREDATORS GREATER THAN 0

if (group.names[i]=='Import'|group.names[i]=='Discards'

|group.names[i]=='By-catch') # CONDITION: IF THE GROUP NAME OF GROUP i IS 'IMPORT', 'BY-CATCH', OR 'DISCARDS' (OUTSIDE (NATURAL) ECOSYSTEM) THEN

{

S[i]=0 # # OF TROPHIC LINKAGES IN MASS DIET MATRIX (p) FOR GROUP i AS PREY WITH OTHER GROUP AS PREDATORS (S) IS SET AT 0

ES[i]=0 # # OF TROPHIC LINKAGES IN ENERGY DIET MATRIX (p) FOR GROUP i AS PREY WITH OTHER GROUP AS PREDATORS (ES) IS SET AT 0

}

S2=rbind(S2,(S[i]+length(which(p[,i]>0)))) # POPULATE VECTOR S2 FOR GROUP i WITH VALUE BEING Si + # OF TROPHIC LINKAGES IN MASS DIET MATRIX FOR GROUP i AS PREDATOR WITH OTHER GROUP AS PREY

} # CLOSE LOOP

#======= STEP 4.1.3. CALCULATE INDICES AND DETERMINE KEYNESS ==================

connectance<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR CONNECTANCE INDEX VALUE

KEYconnectance<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR POTENTIAL DETERMINATION OF A KEY GROUP BY CONNECTANCE INDEX

sumpij<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR VARIABLE sumpij TO CALCULATE MASS SURF INDEX

SURF<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR SURF INDEX VALUE

KEYSURF<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR POTENTIAL DETERMINATION OF A KEY GROUP BY MASS SURF INDEX

esumpij<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR VARIABLE esumpij TO CALCULATE ENERGY SURF INDEX

energy<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR ENERGY SURF INDEX VALUE

KEYenergy<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR POTENTIAL DETERMINATION OF A KEY GROUP BY ENERGY SURF INDEX

for (j in 1:n.groups) # RUN A LOOP WITH NUMBER OF RUNS (i) EQUAL TO NUMBER OF GROUPS

{

connectance=rbind(connectance,(S2[j]/sum(S))) # POPULATE 'CONNECTANCE' INDEX FOR GROUP i WITH TOTAL TROPHIC LINKAGES (S2) OF GROUP i DIVIDED BY TOTAL TROPHIC LINKAGES IN FOOD WEB

if (connectance[j]>0.04) # CONDITION: IF 'CONNECTANCE' INDEX FOR GROUP i IS HIGHER THAN 0.04, THEN

{KEYconnectance=rbind(KEYconnectance,KEYconnectance[j]<-'KEY')} # POPULATE 'KEYCONNECTANCE' FOR GROUP i WITH 'KEY' AS THIS GROUP IS IDENTIFIED AS BEING KEY BY THE CONNECTANCE INDEX

else # IF 'CONNECTANCE' INDEX FOR GROUP i IS NOT HIGHER THAN 0.04 (Smith et al., 2011), THEN

{KEYconnectance=rbind(KEYconnectance,KEYconnectance[j]<-'')} # LEAVE 'KEYCONNECTANCE' FOR GROUP i EMPTY AS THIS GROUP IS NOT IDENTIFIED AS BEING KEY BY THE CONNECTANCE INDEX

sumpij<- rbind(sumpij,sum((p[j,])^2)) # POPULATE 'sumpij' FOR GROUP i WITH VALUE CALCULATED AS THE SUM OF ALL SQUARED DIET PROPORTION GROUP i REPRESENTS FOR EACH GROUP AS PREDATOR

SURF<- rbind(SURF,sumpij[j]/sum(S)) # POPULATE 'SURF' INDEX FOR GROUP i AS 'sumpij' OF GROUP i DIVIDED BY TOTAL TROPHIC LINKAGES IN FOOD WEB

if (SURF[j]>0.001) # CONDITION: IF 'SURF' INDEX OF GROUP i IS HIGHER THAN 0.001 (Plaganyi and Essington, 2014), THEN

{KEYSURF=rbind(KEYSURF,KEYSURF[j]<-'KEY')} # POPULATE 'KEYSURF' FOR GROUP i WITH 'KEY' AS THIS GROUP IS IDENTIFIED AS BEING KEY BY THE MASS SURF INDEX

else # IF 'SURF' INDEX FOR GROUP i IS NOT HIGHER THAN 0.001, THEN

{KEYSURF=rbind(KEYSURF,KEYSURF[j]<-'')} # LEAVE 'KEYSURF' FOR GROUP i EMPTY AS THIS GROUP IS NOT IDENTIFIED AS BEING KEY BY THE MASS SURF INDEX

esumpij<- rbind(esumpij,sum((pE[j,])^2)) # POPULATE 'esumpij' FOR GROUP i WITH VALUE CALCULATED AS THE SUM OF ALL SQUARED DIET PROPORTION GROUP i REPRESENTS FOR EACH GROUP AS PREDATOR

energy<- rbind(energy,esumpij[j]/sum(ES)) # POPULATE 'energy' SURF INDEX FOR GROUP i AS 'esumpij' OF GROUP i DIVIDED BY TOTAL TROPHIC LINKAGES IN FOOD WEB

if (energy[j]>0.001) # CONDITION: IF 'SURF' INDEX OF GROUP i IS HIGHER THAN 0.001 (Surma et al., 2022), THEN

{KEYenergy=rbind(KEYenergy,KEYenergy[j]<-'KEY')} # POPULATE 'KEYSURF' FOR GROUP i WITH 'KEY' AS THIS GROUP IS IDENTIFIED AS BEING KEY BY THE ENERGY SURF INDEX

else # IF 'ENERGY' INDEX FOR GROUP i IS NOT HIGHER THAN 0.001, THEN

{KEYenergy=rbind(KEYenergy,KEYenergy[j]<-'')} # LEAVE 'KEYENERGY' FOR GROUP i EMPTY AS THIS GROUP IS NOT IDENTIFIED AS BEING KEY BY THE ENERGY SURF INDEX

if (group.names[j]=='Import'|group.names[j]=='Discards'|

group.names[j]=='By-catch') # CONDITION: IF GROUP i IS EITHER 'IMPORT', 'DISCARDS' or 'BY-CATCH', THEN

{

KEYconnectance[j]<-'' # LEAVE 'KEYCONNECTANCE' EMPTY

KEYSURF[j]<-'' # LEAVE 'KEYSURF' EMPTY

KEYenergy[j]<-'' # LEAVE 'KEYENERGY' EMPTY

}

}

# NAME COLUMNS

colnames(connectance)<-'connectance'

colnames(KEYconnectance)<-'KEYconnectance'

colnames(SURF)<-'SURF mass'

colnames(KEYSURF)<-'KEYSURF mass'

colnames(energy)<-'SURF energy'

colnames(KEYenergy)<-'KEYSURF energy'

# PASTE ALL COLUMNS WITH DATA INTO ONE DATA FRAME

indices<-data.frame(group.names,round(connectance,10),KEYconnectance,round(SURF,10),KEYSURF, round(energy,10),KEYenergy)

worksheetmass<-data.frame(group.names,S,S2,sumpij,round(connectance,10),KEYconnectance,round(SURF,10),KEYSURF,p)

worksheetenergy<-data.frame(group.names,ES,esumpij,round(energy,10),KEYenergy,pE)

# CREATE LIST OF DATA FOR SEPERATE EXCEL DATASHEETS

list\_of\_datasets <- list("Table A - Model Parameters" = group.data,

"Table B - Diet Mass" = worksheetmass,

"Table C - Diet Energy" = worksheetenergy)

# \*\*\* WRITE OUTPUT FILES \*\*\*

write.xlsx(indices, file.path(output, paste0(geography, "\_indices.xlsx"))) # WRITE EXCEL FILE WITH THE THREE INDICES VALUES FOR ALL FUNCTIONAL GROUPS IN DISAGGREGATED MODEL

write.xlsx(list\_of\_datasets, file.path(output, paste0(geography, ".xlsx"))) # WRITE EXCEL FILE OF THE COMPLETE WORKBOOK INCLUDING THE SEPERATE DATASHEETS

#==============================================================================

# STEP 4.2. INDEX CALCULATIONS FOR AGGREGATED MODEL (IF REQUIRED)

#==============================================================================

#======= STEP 4.2.1. DEFINE QBE ===============================================

group.data$QBE<-(t(ECTOT[1,]/group.data$B)) # ADD A VECTOR TO THE GROUP DATA FOR THE ENERGY CONSUMPTION/BIOMASS (QBE) CALCULATED BY DIVIDING THE TOTAL ENERGY CONSUMPTION PER GROUP (CALCULATE AT LINES 40-45) BY ITS BIOMASS

QBE<-group.data$QBE # EXTRACT THESE DATA AS SEPERATE VECTOR

QBE[is.na(QBE)] <- 0 # CONVERT NA VALUES TO ZEROS

#======= STEP 4.2.2. CREATE FUNCTION FOR AGGREGATING GROUPS IN DIET MATRICES ================================

aggregate.groups<-function(combine.groups,new.groupname,C,EC,B,QB,QBE,p,pE,group.names) { # CREATE FUNTION THAT USES THE ASSIGNED PARAMETERS, SOME ARE BEING GENERATED IN LINES BELOW: COMBINE.GROUPS -> 274 AND NEW.GROUPNAME -> 280

ngroups=length(B) # DETERMINE NUMBER OF GROUPS FROM NUMBER OF BIOMASSES

other.groups=c() # CREATE VECTOR OF GROUPS THAT ARE NOT AGGREGATED

for (i in 1:ngroups){ # RUN A LOOP WITH NUMBER OF RUNS (i) EQUAL TO # ORIGINAL GROUPS

group.index=which(combine.groups==i) # DETERMINE WHETHER GROUP i IS GETTING COMBINED (TRUE) OR NOT (FALSE)

if (length(group.index)==0){ # IF GROUP i IS NOT COMBINED (AND THUS INDEX IS FALSE AND LENGTH INDEX =0), THEN

other.groups=c(other.groups,i) # GROUPS NAME REMAINS UNCHANGED

}

}

#======= CREATE TEMP VECTORS FOR BASIC MODEL PARAMETERS ADDING NEW GROUP =====

B.tmp=as.matrix(c(B,sum(B[combine.groups])),nrow=ngroups+1) # ADD SUMMED BIOMASS OF AGGREGATE GROUP TO THE BOTTOM OF EXISTING LIST OF BIOMASS (B) VALUES

QB.newgroup=sum(QB[combine.groups]\*B[combine.groups]/sum(B[combine.groups])) # CALCULATE CONSUMPTION/BIOMASS (QB) BY CALCULATING ACTUAL CONSUMPTION PER SEPERATE GROUPS DIVIDED BY BIOMASS OF AGGREGATED GROUP AND SUM THESE

QB.tmp=as.matrix(c(QB,QB.newgroup),nrow=ngroups+1) # ADD CONSUMPTION/BIOMASS (QB) OF AGGREGATE GROUP TO THE BOTTOM OF EXISTING LIST OF QB VALUES

QBE.newgroup=sum(QBE[combine.groups]\*B[combine.groups]/sum(B[combine.groups])) # CALCULATE ENERGY CONSUMPTION/BIOMASS (QBE) BY CALCULATING ACTUAL ENERGY CONSUMPTION PER SEPERATE GROUPS DIVIDED BY BIOMASS OF AGGREGATED GROUP AND SUM THESE

QBE.tmp=as.matrix(c(QBE,QBE.newgroup),nrow=ngroups+1) # ADD ENERGY CONSUMPTION/BIOMASS (QBE) OF AGGREGATE GROUP TO THE BOTTOM OF EXISTING LIST OF QBE VALUES

group.names.tmp=as.matrix(c(group.names,new.groupname),nrow=ngroups+1) # ADD GROUP NAME OF AGGREGATE GROUP TO THE BOTTOM OF EXISTING LIST OF GROUP NAMES

#======= MASS DIET MATRIX =============================

C.tmp.pred=rowSums(C[,combine.groups]) # CALCULATE ACTUAL CONSUMPTION IN MASS OF EACH PREY BY AGGREGATED GROUP AS PREDATOR BY SUMMING ACTUAL CONSUMPTION OF PREY IN MASS BY SEPERATE GROUPS

p.tmp.pred=C.tmp.pred/sum(C.tmp.pred) # CALCULATE DIET PROPORTION (p) OF EACH PREY IN DIET AGGREGATED GROUP BY DIVIDING ACTUAL CONSUMPTION OF THAT PREY IN MASS BY TOTAL ACTUAL MASS CONSUMPTION OF AGGREGATE GROUP

p.tmp.new=cbind(p,p.tmp.pred) # ADD DIET COMPOSITION (p) OF AGGREGATE GROUP AS COLUMN TO EXISTING DIET MATRIX

p.tmp.prey=colSums(p.tmp.new[combine.groups,]) # CALCULATE DIET PROPORTION (p) THAT AGGREGATED GROUP REPRESENT AS PREY FOR EACH PREDATOR BY SUMMING THEIR PROPORTION (p) AS SEPERATE GROUPS

p.tmp.new=rbind(p.tmp.new,p.tmp.prey) # ADD ROW OF p-VALUES OF AGGREGATED GROUP AS PREY TO BOTTOM OF DIET MATRIX

#======= ENERGY DIET MATRIX =============================

EC.tmp.pred=rowSums(EC[,combine.groups]) # CALCULATE ACTUAL CONSUMPTION IN ENERGY OF EACH PREY BY AGGREGATED GROUP AS PREDATOR BY SUMMING ACTUAL CONSUMPTION OF PREY IN ENERGY BY SEPERATE GROUPS

pE.tmp.pred=EC.tmp.pred/sum(EC.tmp.pred) # CALCULATE DIET PROPORTION (p) OF EACH PREY IN DIET AGGREGATED GROUP BY DIVIDING ACTUAL CONSUMPTION OF THAT PREY IN ENERGY BY TOTAL ACTUAL ENERGY CONSUMPTION OF AGGREGATE GROUP

pE.tmp.new=cbind(pE,pE.tmp.pred) # ADD DIET COMPOSITION (pE) OF AGGREGATE GROUP AS COLUMN TO EXISTING DIET MATRIX

pE.tmp.prey=colSums(pE.tmp.new[combine.groups,]) # CALCULATE ENERGY DIET PROPORTION (pE) THAT AGGREGATED GROUP REPRESENT AS PREY FOR EACH PREDATOR BY SUMMING THEIR PROPORTION (pE) AS SEPERATE GROUPS

pE.tmp.new=rbind(pE.tmp.new,pE.tmp.prey) # ADD ROW OF pE-VALUES OF AGGREGATED GROUP AS PREY TO BOTTOM OF DIET MATRIX

#======= UPDATE NEW DIET MATRICES WITH NEW GROUP NAMES ==================================

other.groups=c(other.groups,n.groups+1) # CREATE VECTOR WITH EXISTING GROUP NAMES PLUS ONE ROW

p.new=p.tmp.new[other.groups,other.groups] # CREATE NEW DIET MATRIX FOR SURF MASS-BALANCED INCLUDING AGGREGATED GROUP

p.new[is.na(p.new)] <- 0 # REPLACE NA'S WITH 0'S

pE.new=pE.tmp.new[other.groups,other.groups] # CREATE NEW DIET MATRIX FOR SURF ENERGY-BALANCED INCLUDING AGGREGATED GROUP

pE.new[is.na(pE.new)] <- 0 # REPLACE NA'S WITH 0'S

new.colnames=colnames(p.new) # ASSIGN NEW COLUMN NAMES

nr=length(new.colnames) # DETERMINE NUMBER OF GROUPS

new.colnames[nr]=new.groupname # DETERMINE NEW GROUP NAME AS THE LAST ONE

colnames(p.new)=new.colnames # GIVE NEW DIET MATRIX FOR SURF MASS-BALANCED NEW COLUMN NAMES

rownames(p.new)=c() # REMOVE ROW NAMES

colnames(pE.new)=new.colnames # GIVE NEW DIET MATRIX FOR SURF ENERGY-BALANCED NEW COLUMN NAMES

rownames(pE.new)=c() # REMOVE ROW NAMES

B.new=B.tmp[other.groups] # ADJUST BIOMASS (B) AS BASIC PARAMETER

QB.new=QB.tmp[other.groups] # ADJUST CONSUMPTION/BIOMASS (QB) AS BASIC PARAMETER

QBE.new=QBE.tmp[other.groups] # ADJUST ENERGY CONSUMPTION/BIOMASS (QBE) AS BASIC PARAMETER

group.names.new=group.names.tmp[other.groups] # ADJUST GROUP NAMES AS BASIC VECTOR

Agg.model=list(B=B.new,QB=QB.new,QBE=QBE.new,group.names=group.names.new,p=p.new,pE=pE.new) # CREATE A LIST OF THE FUNCTION'S OUTCOMES WITH THE PARAMETERS YOU WANT TO BE PRODUCED

}

#======= STEP 4.2.3. DEFINE GROUPS TO AGGREGATE ===============================

obs1<-length(group.data$agg.group.name) # SET LENGTH FOR FOLLOWING LOOP TO NUMBER OF GROUP NAMES (INCLUDING DOUBLES)

combine.group.names.all<-vector("character") # CREATE NEW VECTOR FOR LIST OF NEW NAMES OF GROUPS THAT NEED TO BE AGGREGATED

for (k in 1:obs1){ # RUN LOOP WITH NUMBER OF RUNS (k) EQUAL TO # OF GROUP NAMES

if(length(which(unique(group.data$agg.group.name[k])==

group.data$agg.group.name))>1) # IF GROUP NAME k OCCURS MORE THAN ONCE (AS FOR THE NEW AGGREGATED GROUPS) IN THE GROUP.DATA TABLE

{

combine.group.names.all<-append(combine.group.names.all,

group.data$agg.group.name[k]) # EXTRACT THOSE AGGREGATED GROUP NAMES FROM GROUP DATA AND ADD TO LIST (THIS WILL CONTAIN NAME OF AGGREAGETED GROUP MULTIPLE TIMES)

}

}

combine.group.names<- unique(combine.group.names.all) # CREATE LIST OF EXISTING GROUPS THAT NEED TO BE AGGREGATED WITH EACH AGGREGATED GROUP NAME APPEARING ONLY ONCE IN THE NEW LIST

obs2<-length(combine.group.names) # SET LENGTH FOR FOLLOWING LOOP TO # OF (NAMES FOR) NEW AGGREGATED GROUPS

combine.group.list<-list() # CREATE NEW VECTOR FOR LIST OF GROUP CODES OF GROUPS THAT ARE INCLUDED IN THE NEW AGGREGATED GROUPS

for (m in 1:obs2){ # RUN LOOP WITH NUMBER OF RUNS (m) EQUAL TO # OF (NAMES FOR) NEW AGGREGATED GROUPS

if(length(which(combine.group.names[m]==group.data$agg.group.name))>1) # IF NEW GROUP NAME m OCCURS MORE THAN ONCE (AS FOR THE NEW AGGREGATED GROUPS) IN THE GROUP.DATA TABLE

{

combine.group.list<-append(combine.group.list,

tryCatch(list(group.data$group.code[which(combine.group.names[m]==group.data$agg.group.name)]))) # LIST THE GROUP CODE OF THOSE GROUPS WITH NEW GROUP NAME m AND ADD THESE CODES PER GROUP NAME TO THE LIST

}

}

n.groups.2.combine<-length(combine.group.list) # ASSIGN NUMBER OF NEW GROUPS CREATED THROUGH AGGREGATION

aggregated.group.list=vector("list",n.groups.2.combine) # CREATE EMPTY LIST OF LENGTH OF GROUP NAMES

if (n.groups.2.combine==0) {

next

} # MOVE TO NEXT ITERATION OF OUTSIDE LOOP IF THERE ARE NO GROUPS TO COMBINE FOR THIS GEOGRAPHY

for (q in 1:n.groups.2.combine){ # RUN A LOOP WITH NUMBER OF RUNS (q) EQUAL TO # OF AGGREGATED GROUPS

aggregated.group.list[[q]]=group.names[combine.group.list[[q]]] # POPULATE LIST WITH NAMES OF AGGREGATED NEW GROUPS

}

#=== STEP 4.2.4. AGGREGATE GROUPS AND CREATE PARAMETERS AND DIET MATRICES =====

for(r in 1:n.groups.2.combine){ # RUN A LOOP WITH NUMBER OF RUNS (r) EQUAL TO # OF AGGREGATED GROUPS

combine.groups.names<-aggregated.group.list[[r]] # DETERMINE THE NAMES OF GROUPS TO BE AGGREGATED TO CREATE AGGREGATED GROUP r

n.combined.groups<-length(combine.groups.names) # DETERMINE NUMBER OF GROUPS INCLUDED TO CREATED AGGREGATED GROUP r BY COUNTING NUMBER OF GROUP NAMES

combine.groups<-c() # CREATE EMPTY VECTOR WITH NAMES OF COMBINED GROUPS

for (s in 1:n.combined.groups){ # RUN A LOOP WITH NUMBER OF RUNS (s) EQUAL TO # OF ORIGINAL GROUPS TO COMBINE TO CREATE AGGREGATED GROUP

combine.groups[s]<-which(group.names==combine.groups.names[[s]]) # LIST THE GROUP CODES OF THE GROUPS NAMES THAT EQUAL THE GROUP NAMES OF THOSE TO BE COMBINED TO CREATE AGGREGATED GROUP K

}

new.groupname<-combine.group.names[r] # ASSIGN NEW GROUPNAME r FROM LIST CREATED IN LINE 223

agg.model=aggregate.groups(combine.groups,

new.groupname,C,EC,B,QB,QBE,p,pE,group.names) # RUN FUNCTION 'AGGREGATE.GROUPS' PER 'group\_aggregating\_fxn.R' TO CREATE DATA FOR AGGREGATED MODEL

B=agg.model$B # EXTRACT BIOMASS (B) VALUES FROM AGGREGATED MODEL

QB=agg.model$QB # EXTRACT CONSUMPTION/BIOMASS (QB) VALUES FROM AGGREGATED MODEL

QBE=agg.model$QBE

group.names=agg.model$group.names # EXTRACT GROUP NAMES FROM AGGREGATED MODEL

p=agg.model$p # EXTRACT DIET MATRIX (SURF MASS-BALANCED) FROM AGGREGATED MODEL

pE=agg.model$pE # EXTRACT DIET MATRIX (SURF ENERGY-BALANCED) FROM AGGREGATED MODEL

n.groups=length(B) # DETERMINE NUMBER OF GROUPS AGGREGATED MODEL AS NUMBER OF NEW GROUP NAMES

CTOT=c(B\*QB) # CREATE VECTOR FOR ACTUAL TOTAL CONSUMPTION IN Tonnes PER GROUP AS PRODUCT OF BIOMASS AND CONSUMPTION/BIOMASS

na.index=which(is.na(CTOT)==TRUE) # CHECK FOR GROUPS WITHOUT CONSUMPTION (E.G., PHYTOPLANKTON AND DETRITUS)

CTOT[na.index]=rep(0,length(na.index)) # FOR THESE GROUPS, SET CONSUMPTION TO BE "0"

CTOT<-t(CTOT) # TRANSPOSE VECTOR FROM COLUMN TO ROW

CTOT <- as.data.frame(lapply(CTOT, rep, length(group.names))) # CREATE DATA FRAME OF TOTAL CONSUMPTION PER GROUP BY REPLICATING THIS ROW AS MANY TIMES AS THERE ARE GROUPS

C=p\*CTOT # CREATE DATA FRAME OF ACTUAL CONSUMPTION OF EACH PREY (ROW) BY EACH PREDATOR (COLUMN) AS PRODUCT OF TOTAL CONSUMPTION AND PIJ

ECTOT=c(B\*QBE) # CREATE VECTOR FOR ACTUAL TOTAL ENERGY CONSUMPTION IN Tonnes PER GROUP AS PRODUCT OF BIOMASS AND ENERGY CONSUMPTION/BIOMASS

na.index=which(is.na(ECTOT)==TRUE) # CHECK FOR GROUPS WITHOUT CONSUMPTION (E.G., PHYTOPLANKTON AND DETRITUS)

ECTOT[na.index]=rep(0,length(na.index)) # FOR THESE GROUPS, SET CONSUMPTION TO BE "0"

ECTOT<-t(ECTOT) # TRANSPOSE VECTOR FROM COLUMN TO ROW

ECTOT <- as.data.frame(lapply(ECTOT, rep, length(group.names))) # CREATE DATA FRAME OF TOTAL CONSUMPTION PER GROUP BY REPLICATING THIS ROW AS MANY TIMES AS THERE ARE GROUPS

EC=pE\*ECTOT # CREATE DATA FRAME OF ACTUAL CONSUMPTION OF EACH PREY (ROW) BY EACH PREDATOR (COLUMN) AS PRODUCT OF TOTAL CONSUMPTION AND PIJ

}

#==============================================================================

# CALCULATE THE 3 INDICES OF AGGREGATED MODEL

#==============================================================================

#======= STEP 4.2.5. COUNT TROPHIC LINKAGES FOR ALL GROUPS ====================

S<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR S (# OF TROPHIC LINKAGES IN MASS DIET MATRIX FOR A GROUP AS PREY WITH OTHER GROUPS AS PREDATORS)

S2<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR S2 (TOTAL TROPHIC LINKAGES AS S + # OF TROPHIC LINKAGES IN MASS DIET MATRIX FOR A GROUP AS PREDATOR WITH OTHER GROUP AS PREY)

ES<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR ES (# OF TROPHIC LINKAGES IN ENERGY DIET MATRIX FOR A GROUP AS PREY WITH OTHER GROUP AS PREDATORS)

n.groups<-dim(p)[1] # SET NUMBER OF GROUPS IN AGGREGATED MODEL

for (u in 1:n.groups) # RUN A LOOP WITH NUMBER OF RUNS (u) EQUAL TO NUMBER OF GROUPS

{ # OPEN LOOP

S=rbind(S,length(which(p[u,]>0))) # POPULATE VECTOR S WITH VALUE BEING # OF TROPHIC LINKAGES IN MASS DIET MATRIX (p) FOR GROUP u AS PREY WITH OTHER GROUP AS PREDATORS GREATER THAN 0

ES=rbind(ES,length(which(pE[u,]>0))) # POPULATE VECTOR ES WITH VALUE BEING # OF TROPHIC LINKAGES IN ENERGY DIET MATRIX (pE) FOR GROUP u AS PREY WITH OTHER GROUP AS PREDATORS GREATER THAN 0

if (group.names[u]=='Import'|group.names[u]=='Discards'|

group.names[u]=='By-catch') # CONDITION: IF THE GROUP NAME OF GROUP u IS 'IMPORT','BY-CATCH' OR 'DISCARDS' (OUTSIDE (NATURAL) ECOSYSTEM) THEN

{

S[u]=0 # # OF TROPHIC LINKAGES IN MASS DIET MATRIX (p) FOR GROUP u AS PREY WITH OTHER GROUP AS PREDATORS (S) IS SET AT 0

ES[u]=0 # # OF TROPHIC LINKAGES IN ENERGY DIET MATRIX (p) FOR GROUP u AS PREY WITH OTHER GROUP AS PREDATORS (ES) IS SET AT 0

}

S2=rbind(S2,(S[u]+length(which(p[,u]>0)))) # POPULATE VECTOR S2 FOR GROUP u WITH VALUE BEING Si + # OF TROPHIC LINKAGES IN MASS DIET MATRIX FOR GROUP u AS PREDATOR WITH OTHER GROUP AS PREY

} # CLOSE LOOP

#======= STEP 4.2.6. CALCULATE INDICES AND DETERMINE KEYNESS ==================

connectance<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR CONNECTANCE INDEX VALUE

KEYconnectance<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR POTENTIAL DETERMINATION OF A KEY GROUP BY CONNECTANCE INDEX

sumpij<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR VARIABLE sumpij TO CALCULATE MASS SURF INDEX

SURF<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR MASS SURF INDEX VALUE

KEYSURF<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR POTENTIAL DETERMINATION OF A KEY GROUP BY MASS SURF INDEX

esumpij<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR VARIABLE esumpij TO CALCULATE ENERGY SURF INDEX

energy<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR ENERGY SURF INDEX VALUE

KEYenergy<-vector(mode='numeric') # CREATE EMPTY VECTOR FOR POTENTIAL DETERMINATION OF A KEY GROUP BY ENERGY SURF INDEX

for (v in 1:n.groups) # RUN A LOOP WITH NUMBER OF RUNS (v) EQUAL TO NUMBER OF GROUPS

{

connectance=rbind(connectance,(S2[v]/sum(S))) # POPULATE 'CONNECTANCE' INDEX FOR GROUP v WITH TOTAL TROPHIC LINKAGES (S2) OF GROUP v DIVIDED BY TOTAL TROPHIC LINKAGES IN FOOD WEB

if (connectance[v]>0.04) # CONDITION: IF 'CONNECTANCE' INDEX FOR GROUP v IS HIGHER THAN 0.04, THEN

{KEYconnectance=rbind(KEYconnectance,KEYconnectance[v]<-'KEY')} # POPULATE 'KEYCONNECTANCE' FOR GROUP v WITH 'KEY' AS THIS GROUP IS IDENTIFIED AS BEING KEY BY THE CONNECTANCE INDEX

else # IF 'CONNECTANCE' INDEX FOR GROUP v IS NOT HIGHER THAN 0.04 (Smith et al., 2011), THEN

{KEYconnectance=rbind(KEYconnectance,KEYconnectance[v]<-'')} # LEAVE 'KEYCONNECTANCE' FOR GROUP v EMPTY AS THIS GROUP IS NOT IDENTIFIED AS BEING KEY BY THE CONNECTANCE INDEX

sumpij<- rbind(sumpij,sum((p[v,])^2)) # POPULATE 'sumpij' FOR GROUP v WITH VALUE CALCULATED AS THE SUM OF ALL SQUARED DIET PROPORTION GROUP v REPRESENTS FOR EACH GROUP AS PREDATOR

SURF<- rbind(SURF,sumpij[v]/sum(S)) # POPULATE 'SURF' INDEX FOR GROUP v AS 'sumpij' OF GROUP v DIVIDED BY TOTAL TROPHIC LINKAGES IN FOOD WEB

if (SURF[v]>0.001) # CONDITION: IF 'SURF' INDEX OF GROUP v IS HIGHER THAN 0.001 (Plaganyi and Essington, 2014), THEN

{KEYSURF=rbind(KEYSURF,KEYSURF[v]<-'KEY')} # POPULATE 'KEYSURF' FOR GROUP v WITH 'KEY' AS THIS GROUP IS IDENTIFIED AS BEING KEY BY THE MASS SURF INDEX

else # IF 'SURF' INDEX FOR GROUP v IS NOT HIGHER THAN 0.001, THEN

{KEYSURF=rbind(KEYSURF,KEYSURF[v]<-'')} # LEAVE 'KEYSURF' FOR GROUP v EMPTY AS THIS GROUP IS NOT IDENTIFIED AS BEING KEY BY THE MASS SURF INDEX

esumpij<- rbind(esumpij,sum((pE[v,])^2)) # POPULATE 'esumpij' FOR GROUP v WITH VALUE CALCULATED AS THE SUM OF ALL SQUARED DIET PROPORTION GROUP v REPRESENTS FOR EACH GROUP AS PREDATOR

energy<- rbind(energy,esumpij[v]/sum(ES)) # POPULATE 'energy' INDEX FOR GROUP v AS 'esumpij' OF GROUP v DIVIDED BY TOTAL TROPHIC LINKAGES IN FOOD WEB

if (energy[v]>0.001) # CONDITION: IF 'SURF' INDEX OF GROUP v IS HIGHER THAN 0.001 (Surma et al., 2022), THEN

{KEYenergy=rbind(KEYenergy,KEYenergy[v]<-'KEY')} # POPULATE 'KEYSURF' FOR GROUP v WITH 'KEY' AS THIS GROUP IS IDENTIFIED AS BEING KEY BY THE ENERGY SURF INDEX

else # IF 'ENERGY' SURF INDEX FOR GROUP v IS NOT HIGHER THAN 0.001, THEN

{KEYenergy=rbind(KEYenergy,KEYenergy[v]<-'')} # LEAVE 'KEYENERGY' FOR GROUP v EMPTY AS THIS GROUP IS NOT IDENTIFIED AS BEING KEY BY THE ENERGY SURF INDEX

if (group.names[v]=='Import'|group.names[v]=='Discards'|

group.names[v]=='By-catch') # CONDITION: IF GROUP v IS EITHER 'IMPORT','BY-CATCH', OR 'DISCARDS', THEN

{

KEYconnectance[v]<-'' # LEAVE 'KEYCONNECTANCE' EMPTY

KEYSURF[v]<-'' # LEAVE 'KEYSURF' EMPTY

KEYenergy[v]<-'' # LEAVE 'KEYENERGY' EMPTY

}

}

# NAME COLUMNS

colnames(connectance)<-'connectance'

colnames(KEYconnectance)<-'KEYconnectance'

colnames(SURF)<-'SURF mass'

colnames(KEYSURF)<-'KEYSURF mass'

colnames(energy)<-'SURF energy'

colnames(KEYenergy)<-'KEYSURF energy'

# PASTE ALL COLUMNS WITH DATA INTO ONE DATA FRAME

indices.agg<-data.frame(group.names,round(connectance,10),KEYconnectance,round(SURF,10),KEYSURF, round(energy,10),KEYenergy)

worksheetmassagg<-data.frame(group.names,S,S2,sumpij,round(connectance,10),KEYconnectance,round(SURF,10),KEYSURF,p)

worksheetenergyagg<-data.frame(group.names,ES,esumpij,round(energy,10),KEYenergy,pE)

list\_of\_datasets <- list("Table A - Model Parameters" = group.data,

"Table B - Diet Mass" = worksheetmass,

"Table C - Diet Energy" = worksheetenergy,

"Table D - Diet Mass Agg" = worksheetmassagg,

"Table E - Diet Energy Agg" = worksheetenergyagg) # CREATE LIST OF DATA FOR SEPERATE EXCEL DATASHEETS

# \*\*\* WRITE OUTPUT FILES \*\*\*

write.xlsx(indices.agg,

file.path(output, paste0(geography, "\_indices\_agg.xlsx"))) # WRITE EXCEL FILE WITH THE THREE INDICES VALUES FOR ALL FUNCTIONAL GROUPS IN AGGREGATED MODEL

write.xlsx(list\_of\_datasets,

file.path(output, paste0(geography, "\_workbook.xlsx"))) # WRITE EXCEL FILE OF THE COMPLETE WORKBOOK INCLUDING THE SEPERATE DATASHEETS

}

### Appendix 2. Overview Ecosystem Information

**Overview of information on all ecosystems for which the three indices have been calculated for related forage species of interest up to December 2024.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ecosystem model** | **Source** | **Diet data sourced (yy)** | **Functional group in food web model** | **Scientific name** | **Comments** |
| California Current | Koehn et al., 2016 | 2022 | Pacific sardine | Sardinops sagax |  |
|  |  |  | Northern anchovy | Engraulis mordax |  |
|  |  |  | Pacific mackerel | Scomber japonicus |  |
|  |  |  | Market squid | Doryteuthis opalescens |  |
| Northern Humboldt Current | Chiaverano et al., 2018 | 2022 | Peruvian anchoveta | Engraulis ringens |  |
|  |  |  | Chilean jack mackerel | Trachurus murphyi |  |
| Morocco Atlantic | Stanford et al., 2001 | 2022 | Sardines | - | Includes European sprat (Sprattus sprattus), European pilchard (Sardina pilchardus), round sardinella (Sardinella aurita), Madeiran sardinella (Sardinella maderensis), and European anchovy (Engraulis encrasicolus) |
|  |  |  | Commercial medium pelagic fish | - | Includes various mackerel species, including Scomber scombrus |
| Gulf of Cadiz | Torres et al., 2013 | 2022 | Sardine | Sardina pilchardus |  |
|  |  |  | Anchovy | Engraulis encrasicolus |  |
|  |  |  | Mackerels | Scomber colias, S. scombrus | Scomber spp. combined |
| Bay of Biscay | Corrales et al., 2022 | 2022 | Sardine | Sardina pilchardus |  |
|  |  |  | Anchovy | Engraulis encrasicolus |  |
|  |  |  | Mackerel | Scomber colias, S. scombrus | Scomber spp. combined |
| Portuguese coast | Veiga-Malta et al., 2019 | 2022 | Sardine | Sardina pilchardus |  |
|  |  |  | Anchovy | Engraulis encrasicolus |  |
|  |  |  | Mackerel | Scomber scombrus |  |
|  |  |  | Chub mackerel | Scomber colias |  |
| Japanese East coast | Watari et al., 2019 | 2022 | Japanese sardine | Sardinops melanostictus |  |
|  |  |  | Japanese anchovy | Engraulis japonicus |  |
|  |  |  | Chub mackerel | Scomber japonicus |  |
|  |  |  | Spotted (blue) mackerel | Scomber australasicus |  |
| South Korean waters | Kim et al., 2022 | 2022 | Japanese anchovy | Engraulis japonicus |  |
|  |  |  | Chub mackerel | Scomber japonicus |  |
| Mauritanian shelf | Guénette et al., 2014 | 2022 | Mackerel | Scomber japonicus |  |
|  |  |  | Sardine | Sardina pilchardus |  |
|  |  |  | Sardinelles | - | Includes round sardinella (Sardinella aurita), Madeiran sardinella (Sardinella maderensis), and European anchovy (Engraulis encrasicolus) |
| Northwest Atlantic - Canada | Tam and Bundy, 2019 | 2022 | Herring | Clupea harengus |  |
|  |  |  | Other planktivorous fish |  | This group includes Atlantic Mackerel (Scomber scombrus) |
| Irish Sea | Bentley et al., 2019 | 2022 | European sprat | Sprattus sprattus |  |
| Gulf Stream – Chesapeake Bay | Dias et al., 2019 | 2022 | US menhaden | Brevoortia tyrannus |  |
| Aegean Sea – Mediterranean | Keramidas et al., 2022 | 2022 | European pilchard | Sardina pilchardus |  |
|  |  |  | European anchovy | Engraulis encrasicolus |  |
|  |  |  | Mackerel | Scomber spp. | Scomber colias, Scomber scombrus |
| Gulf of Maine | Dias et al., 2022 | 2023 | Herring | Clupea harengus |  |
|  |  |  | Atlantic mackerel | Scomber scombrus |  |
| Gulf of St Lawrence (NW Atlantic, CA) | Morisette et al., 2003 | 2023 | Capelin | Mallotus villosus |  |
| Adriatic Sea - Med | Coll et al., 2009 | 2023 | European pilchard | Sardina pilchardus |  |
|  |  |  | European anchovy | Engraulis encrasicolus |  |
| Strait of Sicily - Med | Agnetta et al., 2019 | 2023 | European pilchard | Sardina pilchardus |  |
|  |  |  | European anchovy | Engraulis encrasicolus |  |
| East coast Spain & Gulf of Lion - Med | Corrales et al., 2015 | 2023 | European pilchard | Sardina pilchardus |  |
|  |  |  | European anchovy | Engraulis encrasicolus |  |
| Gulf of Lion - Med | Seyer et al., 2023 | 2023 | European pilchard | Sardina pilchardus |  |
|  |  |  | European anchovy | Engraulis encrasicolus |  |
| Ionian Sea - Med | Moutopoulos et al., 2013 | 2023 | European pilchard | Sardina pilchardus |  |
|  |  |  | European anchovy | Engraulis encrasicolus |  |
| Gulf of Alicante - Med (closest to Alboran Sea) | García-Rodríguez et al., 2021 | 2023 | European pilchard | Sardina pilchardus |  |
|  |  |  | European anchovy | Engraulis encrasicolus |  |
| Lake Superior | Cox and Kitchell, 2004 | 2023 | Lake herring | Coregonus artedi |  |
| West coast India | Mohamed et al., 2008 | 2023 | Cephalopods |  | This group includes Indian squid Uroteuthis duvaucelii |
| Ecuador | Neira et al., 2022 | 2023 | Pacific anchoveta | Cetengraulis mysticetus |  |
| Northern Chile | Barros et al., 2014 | 2023 | Peruvian anchoveta | Engraulis ringens |  |
|  |  |  | Pacific jack mackerel | Trachurus symmetricus |  |
| Central Chile | Neira and Arancibia, 2004 | 2023 | Peruvian anchoveta | Engraulis ringens |  |
|  |  |  | Pacific jack mackerel | Trachurus symmetricus |  |
| Yellow Sea - China | Gao et al., 2021 | 2023 | Beka squid | Loligo beka |  |
| Sea of Japan | Inoue et al., 2023 | 2023 | Flying squid | Todarodes pacificus |  |
|  |  |  |  |  |  |

### Appendix 3. Overview Ecosystem Contact Details

**Overview of contact information of authors of different ecosystem models, where relevant.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Ecosystem** | **Contact person** | **Affiliation** | **Email** | **Comments** |
| Moroccan Atlantic | Dr Richard Stanford | Universitas Andalas, Padang, Indonesia | rzstanford@googlemail.com | Contacted to ask whether there existed diet data with sardine and anchovy as separate species. However, he did not have the data anymore on the Moroccan Ecopath model |
| Moroccan Atlantic | Prof Khalid Elkalay | Ecole supérieure de technologie, Essaouira, Morocco | [elkalay\_khalid@yahoo.fr](mailto:elkalay_khalid@yahoo.fr) | Contacted to ask whether there existed diet data with sardine and anchovy as separate species. However, did eventually not receive data |
| Moroccan Atlantic | Prof Daniel Pauly | Institute for the Oceans and Fisheries, The University of British Columbia | d.pauly@oceans.ubc.ca | Contacted to ask whether there existed diet data with sardine and anchovy as separate species. However, he unfortunately couldn’t help. |
| Mauritania | Dr Beyah Meissa | Institut Mauritanien de Recherches Océanographiques et des Pêches (IMROP) | bmouldhabib@gmail.com | Had been in contact but eventually did not receive diet data as they were publishing at the time. |
| Lake Superior | Prof Sean Cox | School of Resource and Environmental Management, Simon Fraser University | sean\_cox@sfu.ca | Provided data |
| Gulf of Maine | Dr Beatriz dos Santos Dias | College of Fisheries and Ocean Science, University of Alaska | bdossantosdias@alaska.edu | Provided data |
| Gulf of Lion – Mediterranean Sea | Thomas Seyer | Institut de Radioprotection et de Sûreté Nucléaire (IRSN) | thomas.seyer@irsn.fr | Provided data |
| California Current | Dr Laura Koehn | NOAA | laura.koehn216@gmail.com | Provided data |