Rscript\_Diagram

Karolina

12/10/2020

# R script

The script developed to assign metiers based on several variables is called script\_metiers\_test.R.

## Prerequisites

The packages required to run the script are,

* stringr
* data.table
* openxlsx
* purr

as well as auxiliary information described in sections XXX and a set of functions developed to facilitate the readability of the script. These functions are descripted below in detail.

* loadInputData.R : reads the Input file provided it’s in csv format
* loadAreaList.R : reads the [RCG area file](https://github.com/ices-eg/RCGs/raw/master/Metiers/Reference_lists/AreaRegionLookup.csv) from the Github repository
* loadSpeciesList.R : reads the [RCG species file](https://github.com/ices-eg/RCGs/raw/master/Metiers/Reference_lists/Metier%20Subgroup%20Species%202020.xlsx) from the Github repository
* loadMetierList.R : reads the [RCG metier file](https://github.com/ices-eg/RCGs/raw/master/Metiers/Reference_lists/RDB_ISSG_Metier_list.csv) from the Github repository
* loadGearList.R : reads the Gear file from ?? (why not repo)
* getMeasure.R : Determines the if the sequence factor is weight or value.
* getMetier.R : Assigns the metier level 6.
* getMetierLvl5FromPattern.R : Assignes metier level 5

## Data

The data used as an input should be a csv file format as described in detail in section 2 of the report (Input format for transversal data). The first X rows of the example data set are shown below for clarity.

**Remove if not in pdf output or add as annex**

| Country | year | vessel\_id | vessel\_length | trip\_id | haul\_id | fishing\_day | area | ices\_rectangle | gear | mesh | selection | registered\_target\_assemblage | FAO\_species | metier\_level\_6 | measure | KG | EUR |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| POL | 2018 | AAA-1 | 15 | POLAAA1201806100325 |  | 10-06-2018 | 27.2.a |  | OTB | 100 |  |  | HXC |  | weight | 400 | 2,500 |
| POL | 2018 | AAA-1 | 15 | POLAAA1201806100325 |  | 10-06-2018 | 27.2.a |  | OTB | 100 |  |  | SHO |  | weight | 200 | 400 |
| POL | 2018 | AAA-1 | 15 | POLAAA1201806100325 |  | 10-06-2018 | 27.2.a |  | OTB | 100 |  |  | HER |  | weight | 500 | 475 |
| POL | 2018 | AAA-1 | 15 | POLAAA1201806100325 |  | 10-06-2018 | 27.2.a |  | OTB | 100 |  |  | SPR |  | weight | 1,800 | 1,710 |
| POL | 2018 | AAA-1 | 15 | POLAAA1201807011230 |  | 02-07-2018 | 27.3.d.25 | 37G5 | OTB | 110 |  |  | COD |  | weight | 500 |  |
| POL | 2018 | AAA-1 | 15 | POLAAA1201807011230 |  | 02-07-2018 | 27.3.d.25 | 37G5 | OTB | 110 |  |  | FLE |  | weight | 300 |  |

## Code

First the selection column is split between the selectivity device code, if any, and the selectivity device mesh size.

input.data[, `:=`(c("selection\_type", "selection\_mesh"), data.table(str\_split\_fixed(selection,   
 "\_", 2)))]  
input.data[, `:=`(selection\_type, ifelse(selection\_type == "",   
 NA, selection\_type))]  
input.data[, `:=`(selection\_mesh, ifelse(selection\_mesh == "",   
 NA, selection\_mesh))]

| selection | selection\_type | selection\_mesh |
| --- | --- | --- |
| 1\_120 | 1 | 120 |
| 1\_120 | 1 | 120 |
| 1\_120 | 1 | 120 |

The RCG area is assigned based on the area (Subarea, Division, SubDivision) that was provided in the input data.

input.data <- merge(input.data, area.list, all.x = T, by = "area")  
input.data[is.na(RCG) & substr(area, 1, 2) %in% c("31", "34",   
 "41", "47", "51", "57", "58", "87"), `:=`(RCG, "LDF")]  
input.data[is.na(RCG) & substr(area, 1, 2) == "37", `:=`(RCG,   
 "MED")]

| area | RCG | Description |
| --- | --- | --- |
| 27.2.a | NSEA | North Sea and Eastern Arctic |
| 27.3.d.24 | BALT | Baltic Sea |
| 27.3.d.25 | BALT | Baltic Sea |
| 27.3.d.26 | BALT | Baltic Sea |
| 87.3.3 | LDF |  |

Next the target assemblage is assigned, depending on the species, and it’s indicated if they belong to the Deep Water Species (DWS) group.

input.data <- merge(input.data, species.list, all.x = T, by = "FAO\_species")

| FAO\_species | species\_group | dws\_group |
| --- | --- | --- |
| CJM | SPF |  |
| COD | DEF |  |
| FLE | DEF |  |
| FPE | FWS |  |
| HER | SPF |  |
| HXC | LPF | DWS |
| MAS | SPF |  |
| PLE | DEF |  |
| SHO | DEF | DWS |
| SPR | SPF |  |
| TUR | DEF |  |
| YYY |  |  |

A set of variables is necessary to define the level the metier is assigned to and calculate the total weight and value of this group. The default sequence is shown below, however it can be changed to reflect national groupings.

sequence.def <- c("Country", "year", "vessel\_id", "vessel\_length",   
 "trip\_id", "haul\_id", "fishing\_day", "area", "ices\_rectangle",   
 "gear", "mesh", "selection", "registered\_target\_assemblage")  
  
# Calculate group totals for each sequence  
input.data[, `:=`(seq\_group\_KG = sum(KG, na.rm = T), seq\_group\_EUR = sum(EUR,   
 na.rm = T)), by = c(sequence.def, "species\_group")]

Below you can see the calculated totals in the example dataset with the default sequence.

| Country | year | vessel\_id | vessel\_length | trip\_id | haul\_id | fishing\_day | area | ices\_rectangle | gear | mesh | selection | registered\_target\_assemblage | species\_group | seq\_group\_KG | seq\_group\_EUR |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| POL | 2017 | GDY-555 | 40 | POLGDY555201701011230 |  | 02-01-2017 | 87.3.3 |  | OTM | 50 |  |  | SPF | 18,000 | 0 |
| POL | 2018 | BBB-3 | 12 | POLBB3123456 |  | 21-06-2018 | 27.3.d.24 | 37G4 | GNS | 220 |  |  | DEF | 396 | 376 |
| POL | 2018 | AAA-1 | 15 | POLAAA1201807011230 |  | 02-07-2018 | 27.3.d.25 | 37G5 | OTB | 110 |  |  | DEF | 800 | 0 |
| POL | 2018 | ZZZ-2 | 12 | POLZZZ2201801250940 |  | 25-01-2018 | 27.3.d.25 | 38G5 | OTB | 105 | 1\_120 |  | DEF | 1,696 | 1,611 |
| POL | 2010 | ZZZ-2 | 12 | POLZZZ2201001250940 |  | 17-03-2010 | 27.3.d.25 | 38G5 | OTB | 120 |  |  | DEF | 180 | 0 |
| POL | 2020 | DDD-4 | 28 | POLDD9898 |  | 14-04-2020 | 27.3.d.26 | 39G9 | GNS | 110 |  | FWS | DEF | 150 | 300 |
| POL | 2020 | DDD-4 | 28 | POLDD9898 |  | 15-04-2020 | 27.3.d.26 | 39G9 | GNS | 110 |  | DEF | DEF | 280 | 560 |
| POL | 2020 | DDD-4 | 28 | POLDD9898 |  | 02-05-2020 | 27.3.d.26 | 39G9 | OTB | 115 |  |  | DEF | 120 | 400 |
| POL | 2020 | DDD-4 | 28 | POLDD9898 |  | 28-04-2020 | 27.3.d.26 | 39G9 | OTB | 115 |  |  | DEF | 40 | 120 |
| POL | 2020 | DDD-4 | 28 | POLDD9898 |  | 29-04-2020 | 27.3.d.26 | 39G9 | OTB | 115 |  |  | DEF | 80 | 240 |
| POL | 2020 | DDD-4 | 28 | POLDD9898 |  | 30-04-2020 | 27.3.d.26 | 39G9 | OTB | 115 |  |  | DEF | 60 | 180 |
| POL | 2020 | DDD-4 | 28 | POLDD9898 |  | 01-05-2020 | 27.3.d.26 | 39G9 | OTB | 115 |  |  | DEF | 70 | 210 |
| POL | 2020 | DDD-4 | 28 | POLDD9898 |  | 14-04-2020 | 27.3.d.26 | 39G9 | GNS | 110 |  | FWS | FWS | 50 | 100 |
| POL | 2020 | DDD-4 | 28 | POLDD9898 |  | 15-04-2020 | 27.3.d.26 | 39G9 | GN | 1 |  | DEF | FWS | 300 | 500 |
| POL | 2018 | AAA-1 | 15 | POLAAA1201806100325 |  | 10-06-2018 | 27.2.a |  | OTB | 100 |  |  | SPF | 2,300 | 2,185 |
| POL | 2018 | AAA-1 | 15 | POLAAA1201806100325 |  | 10-06-2018 | 27.2.a |  | OTB | 100 |  |  | LPF | 400 | 2,500 |
| POL | 2018 | AAA-1 | 15 | POLAAA1201807011230 |  | 03-07-2018 | 27.3.d.25 | 37G5 | OTB | 110 |  |  | DEF | 100 | 0 |
| POL | 2018 | AAA-1 | 15 | POLAAA1201806100325 |  | 10-06-2018 | 27.2.a |  | OTB | 100 |  |  | DEF | 200 | 400 |
| POL | 2020 | DDD-4 | 28 | POLDD9898 |  | 08-05-2020 | 27.3.d.26 | 39G9 | XYZ | 115 |  |  |  | 20 | 100 |

The function getMeasure.R is used to determine the dominant factor (weight or value) for each sequence as stated in the measure column. If at least one species in the sequence has value, the reference measure becomes the value.

input.data[, `:=`(seq\_measure = getMeasure(measure)), by = sequence.def]

| FAO\_species | area | measure | RCG | species\_group | seq\_group\_KG | seq\_group\_EUR |
| --- | --- | --- | --- | --- | --- | --- |
| CJM | 87.3.3 | weight | LDF | SPF | 18,000 | 0 |
| COD | 27.3.d.24 | value | BALT | DEF | 396 | 376 |
| COD | 27.3.d.25 | weight | BALT | DEF | 800 | 0 |
| COD | 27.3.d.25 | value | BALT | DEF | 1,696 | 1,611 |
| COD | 27.3.d.25 | value | BALT | DEF | 180 | 0 |
| COD | 27.3.d.26 | weight | BALT | DEF | 150 | 300 |

Next, the target assemblage is assigned based on the highest weight or value of each sequence.

input.data[seq\_measure == "weight", `:=`(seq\_dom\_group = species\_group[which.max(seq\_group\_KG)]),   
 by = sequence.def]  
input.data[seq\_measure == "value", `:=`(seq\_dom\_group = species\_group[which.max(seq\_group\_EUR)]),   
 by = sequence.def]

| FAO\_species | area | measure | RCG | **species\_group** | seq\_group\_KG | seq\_group\_EUR | **seq\_dom\_group** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| CJM | 87.3.3 | weight | LDF | **SPF** | 18,000 | 0 | **SPF** |
| COD | 27.3.d.24 | value | BALT | **DEF** | 396 | 376 | **DEF** |
| COD | 27.3.d.25 | weight | BALT | **DEF** | 800 | 0 | **DEF** |
| COD | 27.3.d.25 | value | BALT | **DEF** | 1,696 | 1,611 | **DEF** |
| COD | 27.3.d.25 | value | BALT | **DEF** | 180 | 0 | **DEF** |
| COD | 27.3.d.26 | weight | BALT | **DEF** | 150 | 300 | **DEF** |

The DWS rule is applied; when more than 8% of the species in the sequence are deep water, it is assigned in the DWS group.

input.data[dws\_group == "DWS", `:=`(seq\_DWS\_kg, sum(KG, na.rm = T)),   
 by = c(sequence.def, "dws\_group")]  
input.data[, `:=`(seq\_total\_kg, sum(KG, na.rm = T)), by = sequence.def]  
input.data[, `:=`(seq\_DWS\_perc, ifelse(is.na(seq\_DWS\_kg), 0,   
 seq\_DWS\_kg/seq\_total\_kg) \* 100)]  
input.data[, `:=`(seq\_DWS\_perc, max(seq\_DWS\_perc)), by = sequence.def]  
input.data[seq\_DWS\_perc > 8, `:=`(seq\_dom\_group, "DWS")]

| FAO\_species | area | measure | RCG | species\_group | seq\_group\_KG | seq\_DWS\_perc | seq\_dom\_group |
| --- | --- | --- | --- | --- | --- | --- | --- |
| HER | 27.2.a | weight | NSEA | SPF | 2,300 | 21 | DWS |
| HXC | 27.2.a | weight | NSEA | LPF | 400 | 21 | DWS |
| SHO | 27.2.a | weight | NSEA | DEF | 200 | 21 | DWS |
| SPR | 27.2.a | weight | NSEA | SPF | 2,300 | 21 | DWS |

The function getMetier.R first checks if the user has provided a target assemblage and gives it priority over the calculated dominant assemblage. Then it conditionally assigns the **metier level 6** in a stepwise procedure:

* The metier is assigned based on the RCG, year, gear, target assemblage, dominant target assemblage, mesh size, selectivity device and the mesh size of the selectivity device. The metiers with 0 or >0 mesh size are disregarded.
* If the metier is not assigned, the next step is to disregard the selectivity device code and mesh size and try to assign a metier again. The metiers with 0 or >0 mesh size are also disregarded.
* If the metier is not assigned at this step, the function examines the 0 and >0 mesh sizes and assigns the metier.
* If the metier at this step is still NA, then the MIS\_MIS\_0\_0\_0 metier is assigned.

input.data$metier\_level\_6 <- NA  
input.data[, `:=`(metier\_level\_6, as.character(pmap(list(RCG,   
 year, gear, registered\_target\_assemblage, seq\_dom\_group,   
 mesh, selection\_type, selection\_mesh), function(r, y, g,   
 t, d, m, st, sm) getMetier(r, y, g, t, d, m, st, sm))))]

| RCG | year | gear |  | seq\_dom\_group | mesh | selection\_type | selection\_mesh | metier\_level\_6 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| LDF | 2017 | OTM |  | SPF | 50 |  |  | OTM\_SPF\_32-69\_0\_0 |
| BALT | 2018 | GNS |  | DEF | 220 |  |  | GNS\_DEF\_>=157\_0\_0 |
| BALT | 2018 | OTB |  | DEF | 110 |  |  | OTB\_DEF\_105-115\_1\_120 |
| BALT | 2018 | OTB |  | DEF | 105 | 1 | 120 | OTB\_DEF\_105-115\_1\_120 |
| BALT | 2010 | OTB |  | DEF | 120 |  |  | OTB\_DEF\_>=120\_3\_120 |
| BALT | 2020 | GNS |  | DEF | 110 |  |  | GNS\_FWS\_>0\_0\_0 |

The metier vessel pattern is used to assign **metier level 5**. The code below identifies the main metiers used by a vessel by taking into account the combination of gear and target assemblage. The column seq\_no\_lvl5 shows the number of times the metier level 5 was used by the respective vessel in the same year. The percentage (seq\_perc\_lvl5 column) is used to assign the dominant metier level 5 for each vessel.

input.data[, `:=`(metier\_level\_5, paste(gear, ifelse(is.na(registered\_target\_assemblage),   
 seq\_dom\_group, registered\_target\_assemblage), sep = "\_"))]  
pattern <- unique(input.data[, .SD, .SDcols = c(sequence.def,   
 "metier\_level\_5")])  
pattern <- pattern[, .(seq\_no\_lvl5 = .N), by = .(year, vessel\_id,   
 metier\_level\_5)]  
pattern[, `:=`(seq\_perc\_lvl5, seq\_no\_lvl5/sum(seq\_no\_lvl5, na.rm = T) \*   
 100), by = .(year, vessel\_id)]  
pattern <- pattern[!is.na(metier\_level\_5)]  
input.data <- merge(input.data, pattern, all.x = T, by = c("year",   
 "vessel\_id", "metier\_level\_5"))

| year | vessel\_id | metier\_level\_5 | seq\_no\_lvl5 | seq\_perc\_lvl5 |
| --- | --- | --- | --- | --- |
| 2017 | GDY-555 | OTM\_SPF | 1 | 100 |
| 2018 | BBB-3 | GNS\_DEF | 1 | 100 |
| 2018 | AAA-1 | OTB\_DEF | 2 | 67 |
| 2018 | ZZZ-2 | OTB\_DEF | 1 | 100 |
| 2010 | ZZZ-2 | OTB\_DEF | 1 | 100 |
| 2020 | DDD-4 | GNS\_FWS | 1 | 11 |
| 2020 | DDD-4 | GNS\_DEF | 1 | 11 |
| 2020 | DDD-4 | OTB\_DEF | 5 | 56 |
| 2020 | DDD-4 | GN\_DEF | 1 | 11 |
| 2018 | AAA-1 | OTB\_DWS | 1 | 33 |
| 2020 | DDD-4 | XYZ\_NA | 1 | 11 |

If a “rare” metier is assigned based on one different catch composition or a mis-assigned métier code, it will be set to one of the main métiers of the vessel belonging to this pattern. The percentage threshold for defining rare metiers can be set in the script. The default is 13, meaning that if there is less than 13 percent of the sequences that has a level 5 allocated, it is considered rare and will get the suffix “rare\_” in the column “metier\_level\_5”. If possible, the script will suggest another major metier used by the vessel to be used instead. If it is different from the méetier code assigned it will have the suffix “pattern\_”.

rare.threshold <- 13  
input.data[seq\_perc\_lvl5 < rare.threshold, `:=`(metier\_level\_5,   
 NA)]  
pattern <- pattern[seq\_perc\_lvl5 >= rare.threshold]  
pattern[, `:=`(c("gear", "target\_assemblage"), data.table(str\_split\_fixed(metier\_level\_5,   
 "\_", 2)))]  
pattern <- merge(pattern, gear.list, all.x = T, by.x = "gear",   
 by.y = "gear\_code")  
input.data <- merge(input.data, gear.list, all.x = T, by.x = "gear",   
 by.y = "gear\_code")  
input.data[is.na(metier\_level\_5), `:=`(metier\_level\_5, as.character(pmap(list(vessel\_id,   
 year, gear, gear\_group, registered\_target\_assemblage, seq\_dom\_group),   
 function(v, y, g, gg, rt, d) getMetierLvl5FromPattern(v,   
 y, g, gg, rt, d))))]

| vessel\_id | year | gear |  | seq\_dom\_group | metier\_level\_5 |
| --- | --- | --- | --- | --- | --- |
| DDD-4 | 2020 | GN |  | FWS | pattern\_OTB\_DEF |
| BBB-3 | 2018 | GNS |  | DEF | GNS\_DEF |
| BBB-3 | 2018 | GNS |  | DEF | GNS\_DEF |
| BBB-3 | 2018 | GNS |  | DEF | GNS\_DEF |
| BBB-3 | 2018 | GNS |  | DEF | GNS\_DEF |
| DDD-4 | 2020 | GNS |  | DEF | pattern\_OTB\_DEF |

## Workflow diagramm

