Loading\_HMRC-data-trial\_DELETE-LATER

B. Scheliga, A. Lofstedt

2022-10-26

## Part ???: Cleaning HMRC Trade data to map UK seafood supply chains

This R Markdown document outlines how the HRMC trade data was complied and cleaned. The HRMC trade data ranges from 2009 - 2019.

The justification for the data included in this data set can be found in the supporting excel document.

## Preparation

# It is good practice to load all needed libraries in the beginning of the scripted  
#library(here)  
library(tidyr)  
library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──  
## ✔ ggplot2 3.3.6 ✔ dplyr 1.0.9  
## ✔ tibble 3.1.8 ✔ stringr 1.4.0  
## ✔ readr 2.1.2 ✔ forcats 0.5.1  
## ✔ purrr 0.3.4   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

library(vroom) #for loading and transforming data  
library(data.table) # for fread()-function

##   
## Attaching package: 'data.table'  
##   
## The following objects are masked from 'package:dplyr':  
##   
## between, first, last  
##   
## The following object is masked from 'package:purrr':  
##   
## transpose

library(mice) # md.pattern to show missing data

##   
## Attaching package: 'mice'  
##   
## The following object is masked from 'package:stats':  
##   
## filter  
##   
## The following objects are masked from 'package:base':  
##   
## cbind, rbind

#Loading file path to project folder  
source("Data\_filepath.R")# Data\_filepath.R is listed in .gitignore-file. So, you will need to create that file yourself and provide your respected filepath using "data\_dir <- [enter your here]"

## Loading the HMRC trade data

########### HMRC Trade Data  
  
# Filepath to RawData   
filepath <- paste(data\_dir,"RawData/csv-files", sep="")   
# reading the HRMC trade data dataset (.csv)  
#df\_HRMC\_2009 <- vroom(file=paste(filepath,"HMRC\_UK\_trade-2009.csv", sep="/"))  
vec\_filenames\_HMRC <- list.files(filepath, pattern = "HMRC\_UK\_trade", full.names = TRUE)  
  
  
  
print("The following files were loaded:")

## [1] "The following files were loaded:"

i = 1  
# Loading the cleaned dataset  
for(f in vec\_filenames\_HMRC){  
  
 temp <- fread(f) # storing the data in a temporary   
 assign(paste("df\_HMRC\_",i,sep=""),temp) # assigning the df a name based on input dataset  
 print(paste(i,". ",sub(paste(".\*",filepath,sep=""),"",f),sep="")) # print out which dataset files had been used  
 i = i+1  
 rm(temp)# removing temp object  
}

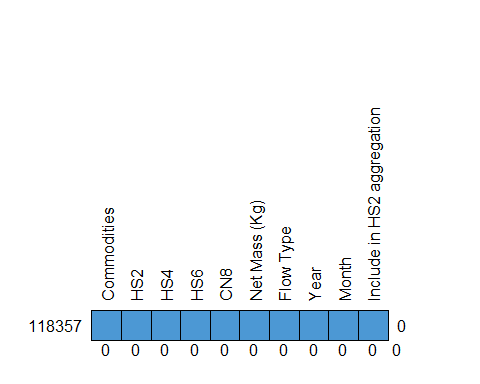
## [1] "1. /HMRC\_UK\_trade-2009.csv"  
## [1] "2. /HMRC\_UK\_trade-201011.csv"  
## [1] "3. /HMRC\_UK\_trade-201213.csv"  
## [1] "4. /HMRC\_UK\_trade-201415.csv"  
## [1] "5. /HMRC\_UK\_trade-201617.csv"  
## [1] "6. /HMRC\_UK\_trade-201819.csv"

rm(i)  
  
# Next step is to combine this all into big HMRC data.frame  
df\_HRMC\_2009\_to\_2019 <- bind\_rows(df\_HMRC\_1, df\_HMRC\_2, df\_HMRC\_3, df\_HMRC\_4, df\_HMRC\_5, df\_HMRC\_6)  
  
rm(df\_HMRC\_1, df\_HMRC\_2, df\_HMRC\_3, df\_HMRC\_4, df\_HMRC\_5, df\_HMRC\_6) # removing the unneeded df

#### Checking for missing values

md.pattern(df\_HRMC\_2009\_to\_2019, rotate.names = TRUE)

## /\ /\  
## { `---' }  
## { O O }  
## ==> V <== No need for mice. This data set is completely observed.  
## \ \|/ /  
## `-----'



## Commodities HS2 HS4 HS6 CN8 Net Mass (Kg) Flow Type Year Month  
## 118357 1 1 1 1 1 1 1 1 1  
## 0 0 0 0 0 0 0 0 0  
## Include in HS2 aggregation   
## 118357 1 0  
## 0 0

### Loading the EUMOFA data

We use the EUMOFA data as a help to facilitate the classification and translation of the HRMC CN-8 codes in to our desired species and species type. The EUMOFA has a classification mapped to each CN-8 code. Using this, instead of crawling through the HMRC “Product name”-column and searching for specific key words its text, will mitigate a lot of potential miss-classification. As the HMRC “Product name”, does not seem to use controlled vocabulary and has various spelling version of the same word (plural & single) included.

[MAYBE MORE DETAILS ON EUMOFA DATA HERE]

############### EUMOFA-file  
  
# Filepath to EUMOFA-file with complete CN-8 codes  
filepath <- paste(data\_dir,"Methods/SpeciesTypeClassification", sep="")  
# reading the EUMOFA-file  
df\_EUMOFA\_CN8 <- vroom(file=paste(filepath,"EUMOFA\_CN-8-values.csv", sep="/"))

## Rows: 9386 Columns: 7  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (5): Year of Reg, CN-8, Comment, CN-8 product name, Explanation  
## dbl (2): Year, CF  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

## Note: the CN8 codes are related to the respective year, and have change over time.  
  
# We are also loading the Annex 4 from the Metadata 2 - Data management EUFOMA https://www.eumofa.eu/supply-balance-and-other-methodologies  
# ANNEX 4 Correlation between Main commercial species(MCS)/Commodity Groups (CG) and CN-8 from 2001 to 2022 Revision 1.2  
  
df\_EUMOFA\_CN8\_MCS\_CG <- vroom(file=paste(filepath,"EUMOFA\_Annex4.csv", sep="/"))

## Rows: 9917 Columns: 9  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (8): CN8 code, Description, PS, PR, MCS\_code, MCS\_descr, CG code, CG  
## dbl (1): Year  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

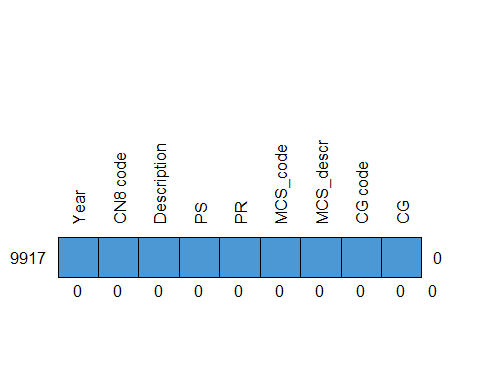
# loading Species classification - translation from EUMOFA MCS & CG to the classification we want to use.  
df\_Species\_Class <- vroom(file=paste(filepath,"SpeciesTypeClassificationCode.csv", sep="/"))

## Rows: 103 Columns: 4  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (4): EUMOFA\_MCS, EUMOFA\_CG, EUMOFA\_MCS\_AL, SpeciesType\_AL  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

#### Checking for missing values

md.pattern(df\_EUMOFA\_CN8\_MCS\_CG, rotate.names = TRUE)

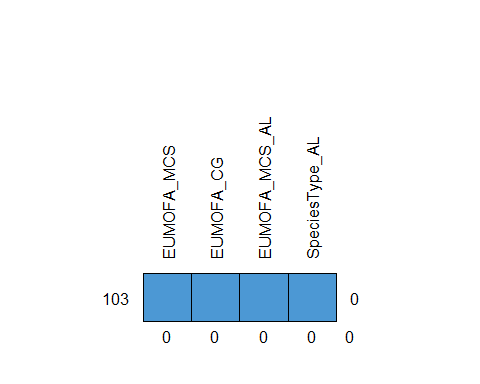
## /\ /\  
## { `---' }  
## { O O }  
## ==> V <== No need for mice. This data set is completely observed.  
## \ \|/ /  
## `-----'



## Year CN8 code Description PS PR MCS\_code MCS\_descr CG code CG   
## 9917 1 1 1 1 1 1 1 1 1 0  
## 0 0 0 0 0 0 0 0 0 0

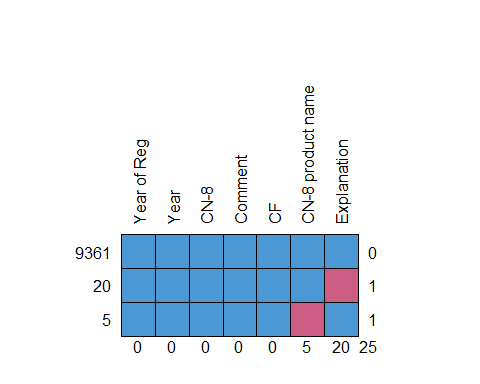
md.pattern(df\_Species\_Class, rotate.names = TRUE)

## /\ /\  
## { `---' }  
## { O O }  
## ==> V <== No need for mice. This data set is completely observed.  
## \ \|/ /  
## `-----'



## EUMOFA\_MCS EUMOFA\_CG EUMOFA\_MCS\_AL SpeciesType\_AL   
## 103 1 1 1 1 0  
## 0 0 0 0 0

md.pattern(df\_EUMOFA\_CN8, rotate.names = TRUE)



## Year of Reg Year CN-8 Comment CF CN-8 product name Explanation   
## 9361 1 1 1 1 1 1 1 0  
## 20 1 1 1 1 1 1 0 1  
## 5 1 1 1 1 1 0 1 1  
## 0 0 0 0 0 5 20 25

We have 25 missing (NA) values in the EUMOFA dataset, 20 in the “Explanation”-column and 5 in the “CN-8 product name”-column. As annoying as NA-values are, in this instance we can ignore them. As these values are irrelevant for us and the next steps, since the EUMOFA Annex 4 dataset has no NA-values and the main commercial species (MCS) and commodity groups (CG) for the respective CN-8 code.

## Processing the data

# need to remove the space from the 'CN-8'-column. But leave it as character, so we don't loose the "0" at the beginning of the CN8 code  
# https://stackoverflow.com/questions/20309876/r-how-to-replace-in-a-string  
  
df\_EUMOFA\_CN8$`CN-8` <- gsub("\\ ","", df\_EUMOFA\_CN8$`CN-8`)  
  
#df\_EUMOFA\_CN8\_2009 <- df\_EUMOFA\_CN8 %>% filter(Year %in% "2009" )  
  
print("FYI: In the EUMOFA CN-8 product name-Column are large - symbols, which R does not recognize replaces them with ? in a square")

## [1] "FYI: In the EUMOFA CN-8 product name-Column are large - symbols, which R does not recognize replaces them with ? in a square"

# Need to separate the CN8 code from the description in the CN8-column of the HMRC data  
df\_HRMC\_2009\_to\_2019$Subset\_aid\_CN8 <- gsub(" .\*$", "", df\_HRMC\_2009\_to\_2019$CN8) # subset string before white space  
  
# Thanks to our psychic abilities, we know that one of the CN8-codes in the HRMC Uk trade is missing its lead 0. And we need to deal with that.   
  
# Check what will not be joined  
df\_HMRC\_anti <- df\_HRMC\_2009\_to\_2019 %>% select('Net Mass (Kg)', 'Flow Type', 'Year','Month', 'Subset\_aid\_CN8', 'CN8') %>% anti\_join(.,df\_EUMOFA\_CN8, by = c('Subset\_aid\_CN8' = 'CN-8'))  
#The following CN8 code items were not joined  
unique(df\_HMRC\_anti$CN8)

## [1] "03 HS2 Below Threshold Trade"   
## [2] "3074959"   
## [3] "03076000 Snails, live, fresh, chilled, frozen, salted, dried or in brine, even smoked, with or without shell (excl. sea snails)"   
## [4] "05119910 Sinews or tendons of animal origin, parings and similar waste of raw hides or skins"   
## [5] "05119931 Raw natural sponges of animal origin"   
## [6] "05119939 Natural sponges of animal origin (excl. raw)"   
## [7] "05119985 Animal products, n.e.s.; dead animals, unfit for human consumption (excl. fish, crustaceans, molluscs and other aquatic invertebrates)"  
## [8] "23011000 Flours, meals and pellets, of meat or offal, unfit for human consumption; greaves"   
## [9] "03076010 Snails, smoked, even in shell, even cooked but not otherwise prepared (excl. sea snails)"   
## [10] "03076090 Snails, live, fresh, chilled, frozen, salted, dried or in brine, even in shell (excl. smoked and sea snails)"   
## [11] "16055800 Snails, prepared or preserved (excl. smoked and sea snails)"

It is item [2] “3074959”, were we need to add a lead zero.

# adding the lead 0 to "3074959" in the HRMC UK trade data  
df\_HRMC\_2009\_to\_2019$Subset\_aid\_CN8 <- str\_replace(df\_HRMC\_2009\_to\_2019$Subset\_aid\_CN8,"3074959","03074959")  
  
# the "Flow Type" is also separate into "EU" and "Non-EU" Imports and Exports  
df\_HRMC\_2009\_to\_2019$`Flow Type` <- str\_replace(df\_HRMC\_2009\_to\_2019$`Flow Type`,"Non EU - ","") # needs to be first. Or else line 83 will remove "EU" from "Non EU -"  
df\_HRMC\_2009\_to\_2019$`Flow Type` <- str\_replace(df\_HRMC\_2009\_to\_2019$`Flow Type`,"EU - ","")  
  
  
# Now, we can join the HMRC trade data and the EUMOFA data  
df\_HMRC\_inner <- df\_HRMC\_2009\_to\_2019 %>% select('Net Mass (Kg)', 'Flow Type', 'Year','Month', 'Subset\_aid\_CN8') %>% inner\_join(.,df\_EUMOFA\_CN8, by = c('Subset\_aid\_CN8' = 'CN-8', 'Year' ))

# calculating annual sum, selecting needed columns and converting to Net Mass from kg to 1000 tonnes  
df\_HRMC.sum <- df\_HMRC\_inner %>% group\_by(CN8 = Subset\_aid\_CN8,`Product name` =`CN-8 product name`, Year = Year, Commodity =`Flow Type`, CF) %>% summarise(`Net Mass (1000 t)`= sum(`Net Mass (Kg)`)/1000000,`Net Mass (kg)`= sum(`Net Mass (Kg)`))

## `summarise()` has grouped output by 'CN8', 'Product name', 'Year', 'Commodity'.  
## You can override using the `.groups` argument.

rm(df\_HMRC\_inner)  
  
df\_HMRC\_4DB <-df\_HRMC.sum

# mapping EUMOFA Main Commercial Species (MCS) and Commodity Group (CG) classification  
df\_HMRC\_4DB <- df\_HMRC\_4DB %>% inner\_join(., df\_EUMOFA\_CN8\_MCS\_CG, by = c('CN8' = 'CN8 code', 'Year'))  
  
# Mapping our desired Species Species Type classification  
df\_HMRC\_4DB <- df\_HMRC\_4DB %>% inner\_join(., df\_Species\_Class, by = c('MCS\_descr' = 'EUMOFA\_MCS'))  
  
# We will now need to aggregated the weight values for some species again. As we have aggregate some some of the species and species type e.g. atlantic & pacific salmon as just salmon  
  
df\_HRMC.sum <- df\_HMRC\_4DB %>% group\_by(Species = EUMOFA\_MCS\_AL, SpeciesType = SpeciesType\_AL, Year, Commodity) %>% summarise(`Net Mass (1000 t)`= sum(`Net Mass (kg)`)/1000000,`Net Mass (kg)`= sum(`Net Mass (kg)`))

## `summarise()` has grouped output by 'Species', 'SpeciesType', 'Year'. You can  
## override using the `.groups` argument.

df\_HMRC\_4DB <- df\_HRMC.sum

# Adding DataSupplier information  
df\_HMRC\_4DB$DataSupplier <- "HMRC"  
df\_HMRC\_4DB$DataSet <- "HMRC Overseas Trade data table - UK Trade Info"  
#determining if the Fish is for Human consumption or not based on the CF from the EUMOFA-file  
df\_HMRC\_4DB <- rename(df\_HMRC\_4DB, Value =`Net Mass (1000 t)`)  
df\_HMRC\_4DB$Units <- "1000 tonnes"  
df\_HMRC\_4DB$TemporalResolution <- "Annual"  
df\_HMRC\_4DB$Flag <- "EXAMPLES: UC, SSTAlig"  
df\_HMRC\_4DB$FlagDescription <- "EXAMPLES:Units changed, Species & Species Type aligned"

# Selecting Columns  
df\_HMRC\_4DB <- df\_HMRC\_4DB %>% ungroup %>% select(DataSupplier, DataSet, Commodity, Species, SpeciesType, Value, Units, Year, TemporalResolution, Flag, FlagDescription)  
  
# Removing Non-food uses im- & exports  
df\_HMRC\_4DB <- df\_HMRC\_4DB %>% filter(!SpeciesType %in% "OtherNFU")  
  
filepath <- paste(data\_dir,"ProcessedData", sep="")  
  
write.csv(df\_HMRC\_4DB, paste(filepath,"TradeData\_HMRC\_Preliminary-Cleaned.csv", sep="/"), row.names = FALSE)