

Package: dietary (via r-universe)

September 21, 2026

Type Package

Title Utilities for Calculating Dietary Intake Indicators for Food Security Assessments

Version 0.0.0.9001

Description Food security assessments utilise several dietary intake indicators as proxy measures for diet quality, diet sufficiency, and food availability either at individual or household level. Utilities for recoding and calculating these indicators support in establishing consistent and reliable results.

License GPL (>= 3)

Depends R (>= 2.10)

Suggests knitr, rmarkdown, spelling, testthat (>= 3.0.0)

Encoding UTF-8

Language en-GB

LazyData true

RoxygenNote 7.3.3

Roxygen list(markdown = TRUE)

URL <https://nutriverse.io/dietry/>,
<https://github.com/nutriverse/dietry>

BugReports <https://github.com/nutriverse/dietry/issues>

Config/testthat/edition 3

Config/Needs/website rmarkdown

VignetteBuilder knitr

Repository <https://nutriverse.r-universe.dev>

Date/Publication 2026-01-28 06:38:42 UTC

RemoteUrl <https://github.com/nutriverse/dietry>

RemoteRef HEAD

RemoteSha 89e384bd1f1ba3cf37f9e32bd1f650d38e948d45

Contents

fcs_calculate	2
fcs_classify	3
fcs_get_colours	4
fcs_recode	5
fcs01	5
map_variables	6
Index	8

fcs_calculate	<i>Calculate Food Consumption Score (FCS)</i>
---------------	---

Description

Calculate Food Consumption Score (FCS)

Usage

```
fcs_calculate(df, var_map, weights = NULL, add = TRUE)
```

Arguments

df	A data.frame with FCS data.
var_map	A named list of FCS food groups mapped to corresponding variable names in df. This can be produced using fcs_fg_map_variables() .
weights	A numeric vector of FCS weights applied to corresponding food groups. The weights should be ordered as that for staples, pulses, vegetables, fruits, meat and fish, dairy, sugar, oil, and condiments. Default to NULL which uses the weights based on current FCS recommendations. Only change this if new recommendations have been provided or for testing/studying new/experimental FCS weighting systems.
add	Logical. Should the resulting FCS scores be added to df? Default to TRUE.

Value

If add = TRUE, a data.frame based on df with a new variable named fcs for the calculated food consumption scores. Otherwise, a numeric vector of the calculated food consumption scores.

Author(s)

Ernest Guevarra

Examples

```
var_map <- fcs_fg_map_variables(
  staples = "FCSStap",
  pulses = "FCSPulse",
  vegetables = "FCSVeg",
  fruits = "FCSFruit",
  meatfish = "FCSPr",
  milk = "FCSDairy",
  sugar = "FCSSugar",
  oil = "FCSFat",
  condiment = "FCSCond"
)

fcs_calculate(df = fcs01, var_map = var_map)
```

fcs_classify	<i>Classify Food Consumption Score (FCS)</i>
--------------	--

Description

Classify Food Consumption Score (FCS)

Usage

```
fcs_classify(fcs, cutoff = NULL, add = FALSE, spread = FALSE)
```

Arguments

fcs	A vector of food consumption scores.
cutoff	A numeric vector of length 2 for the cut-offs to use for classifying FCS. Default to NULL in which case standard recommended cut-off values for FCS are used.
add	Logical. Should classification be column bound to fcs? Default to FALSE.
spread	Logical. Should classification be spread into columns? Default to FALSE.

Value

If spread = TRUE, a data.frame with number of rows equal to the length of fcs and number of columns equal to length of fill plus an initial column named fcs containing the FCS values provided by fcs argument if add = TRUE. Otherwise, a vector of class factor containing FCS classifications. If add = TRUE, this vector is concatenated with the fcs values in a data.frame.

Author(s)

Ernest Guevarra

Examples

```
var_map <- fcs_fg_map_variables(  
  staples = "FCStap",  
  pulses = "FCSPulse",  
  vegetables = "FCSVeg",  
  fruits = "FCSFruit",  
  meatfish = "FCSPR",  
  milk = "FCSDairy",  
  sugar = "FCSSugar",  
  oil = "FCSFat",  
  condiment = "FCSCond"  
)  
  
fcs <- fcs_calculate(df = fcs01, var_map = var_map)  
  
fcs_classify(fcs$fcs)
```

fcs_get_colours

Get Food Consumption Score (FCS) colours

Description

Get Food Consumption Score (FCS) colours

Usage

```
fcs_get_colours()
```

Value

A named character vector of recommended FCS classification colours

Author(s)

Ernest Guevarra

Examples

```
fcs_get_colours()
```

fcs_recode	<i>Recode Food Consumption Score (FCS) data</i>
------------	---

Description

Recode Food Consumption Score (FCS) data

Usage

```
fcs_recode(x, na_values = NULL)
```

Arguments

x	A vector of numeric values that can range from 0 to 7 for the number of days in a week that a food group is eaten by a household as per FCS guidelines.
na_values	A value or a vector of values that are to be considered as NA. Default to NA.

Value

An integer vector with possible values ranging from 0 to 7.

Author(s)

Ernest Guevarra

Examples

```
fcs_recode(fcs01$FCSStap)
```

fcs01	<i>Sample Food Consumption Score (FCS) data from World Food Programme (WFP) VAM Resource Centre</i>
-------	---

Description

Sample Food Consumption Score (FCS) data from World Food Programme (WFP) VAM Resource Centre

Usage

```
fcs01
```

Format

A data frame with 18 columns and 26 rows:

Variable	Description
<i>FCSStap</i>	Integer value from 0-7 for consumption frequency of staples
<i>FCSStap_SRf</i>	Staples food source
<i>FCSVeg</i>	Integer value from 0-7 for consumption frequency of vegetables
<i>FCSVeg_SRf</i>	Vegetables food source
<i>FCSFruit</i>	Integer value from 0-7 for consumption frequency of fruits
<i>FCSFruit_SRf</i>	Fruits food source
<i>FCSPr</i>	Integer value from 0-7 for consumption frequency of protein-rich foods
<i>FCSPr_SRf</i>	Protein-rich food source
<i>FCSpulse</i>	Integer value from 0-7 for consumption frequency of pulses
<i>FCSpulse_SRf</i>	Pulses food source
<i>FCSdairy</i>	Integer value from 0-7 for consumption frequency of dairy
<i>FCSdairy_SRf</i>	Dairy food source
<i>FCSSugar</i>	Integer value from 0-7 for consumption frequency of sugary foods
<i>FCSSugar_SRf</i>	Sugary food source
<i>FCSFat</i>	Integer value from 0-7 for consumption frequency of fats
<i>FCSFat_SRf</i>	Fats food source
<i>FCSCond</i>	Integer value from 0-7 for consumption frequency of condiments
<i>FCSCond_SRf</i>	Condiments food source

Source

https://github.com/WFP-VAM/RAMResourcesScripts/blob/main/Static/FCS_Sample_Survey.csv

Examples

```
fcs01
```

map_variables	<i>Map data variables to corresponding indicators</i>
---------------	---

Description

Map data variables to corresponding indicators

Usage

```
map_variables(...)

fcs_fg_map_variables(
  ...,
  foodgroups = c("staples", "pulses", "vegetables", "fruits", "meatfish", "milk",
```

```

    "sugar", "oil", "condiment")
)

hdds_fg_map_variables(
  ...,
  foodgroups = c("cereals", "roots_tubers", "vegetables", "fruits", "meat", "eggs",
    "fish", "pulses", "milk", "oil", "sugar", "condiments")
)

mddw_fg_map_variables(
  ...,
  foodgroups = c("staples", "pulses", "nuts", "milk", "meat_fish", "eggs", "green_leafy",
    "other_vita", "other_vegetables", "other_fruits")
)

```

Arguments

... Name-value pairs. Name gives the labels for indicators. The value should be the corresponding variable name in a dataset used for that indicator.

foodgroups A character vector of food group labels for a specific dietary intake indicator set.

Value

A named list of variable name/s for corresponding food groups.

Examples

```

## Variable names in fcs01 mapped to corresponding food group labels
map_variables(
  staples = "FCSStap",
  pulses = "FCSPulse",
  vegetables = "FCSVeg",
  fruits = "FCSFruit",
  meatfish = "FCSPR",
  milk = "FCSDairy",
  sugar = "FCSSugar",
  oil = "FCSFat",
  condiment = "FCSCond"
)

```

Index

* **datasets**

fcs01, [5](#)

fcs01, [5](#)

fcs_calculate, [2](#)

fcs_classify, [3](#)

fcs_fg_map_variables (map_variables), [6](#)

fcs_fg_map_variables(), [2](#)

fcs_get_colours, [4](#)

fcs_recode, [5](#)

hdds_fg_map_variables (map_variables), [6](#)

map_variables, [6](#)

mddw_fg_map_variables (map_variables), [6](#)