

Package: GFSir (via r-universe)

October 9, 2026

Type Package

Title Global Food Security analysis using IMPACT model and R

Version 5.1.3

Description Global Food Security analysis using IMPACT model and R.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.2

Imports DOORMAT (>= 3.14.1), dplyr (>= 1.1.4), tidyr (>= 1.3.1),
ggplot2 (>= 3.5.1), collapse (>= 2.0.10), data.table (>= 1.14.8), janitor (>= 2.2.0), openxlsx2 (>= 1.9)

Remotes github::eddelbuettel/drat

Additional_repositories <https://ifpri.github.io/drat>

Suggests knitr, rmarkdown

VignetteBuilder knitr

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

Date/Publication 2026-08-10 22:47:54 UTC

RemoteUrl <https://github.com/IFPRI/GFSir>

RemoteRef HEAD

RemoteSha 6951d6dd033ebd2fc9c1b8938a656a18c2fcf0fc

Contents

add_crops	2
add_regions	3
clean_filename	3
create_identifier_columns	4
get_fmc	5
getMapping	6
group1	6
group3	7

group4	8
group5	9
save_ggplot	9

Index 11

add_crops	<i>add_crops</i>
-----------	------------------

Description

add_crops

Usage

```
add_crops(df, mapping = "mapping.xlsx")
```

Arguments

df	dataframe with "j" or "c" column
mapping	Mapping file which contains crops/activities

Value

Cleaned scenario name

Author(s)

Abhijeet Mishra

Examples

```
## Not run:  
add_crops()  
  
## End(Not run)
```

add_regions	<i>add_regions</i>
-------------	--------------------

Description

add_regions

Usage

```
add_regions(df, mapping = "mapping.xlsx")
```

Arguments

df	dataframe with "cty" column
mapping	Mapping file for regions

Value

Cleaned scenario name

Author(s)

Abhijeet Mishra

Examples

```
## Not run:  
add_regions()  
  
## End(Not run)
```

clean_filename	<i>clean_filename</i>
----------------	-----------------------

Description

clean_filename

Usage

```
clean_filename(gdx)
```

Arguments

gdx	GDX file for IMPACT run
-----	-------------------------

Value

Cleaned scenario name

Author(s)

Abhijeet Mishra

Examples

```
## Not run:
clean_filename()

## End(Not run)
```

```
create_identifier_columns
      create_identifier_columns
```

Description

create_identifier_columns

Usage

```
create_identifier_columns(
  df,
  col = "scenario",
  into = c("SSP", "GCM", "RCP", "CO2"),
  sep = "-",
  remove = TRUE
)
```

Arguments

df	Dataframe where identifier columns need to be created
col	Columns which are "ID" and need to be separated
into	String vector with names of new columns
sep	Separator where "col" columns should be snipped for new columns.
remove	If "col" vector should be kicked out when making new columns.

Value

Cleaned scenario name

Author(s)

Abhijeet Mishra

Examples

```
## Not run:  
create_identifier_columns()  
  
## End(Not run)
```

get_fmc

Build FAO Master Codes Table

Description

Reads j-code commodity mappings from the bundled `Sets.xlsx` and `FAOSTAT_data_3-5-2026.csv` files shipped with the package, resolves duplicates, and returns a clean item-to-j-code crosswalk.

Usage

```
get_fmc()
```

Details

The function reads three column ranges from the Commodities sheet of `Sets.xlsx` to build the j-code lookup, appends several hardcoded corrections, then joins against the FAO master code list. Duplicate item names that map to a single unambiguous j-code are resolved automatically; those that cannot be resolved are dropped with a warning.

Duplicate checks are performed at three stages using [get_dupes](#). A warning is issued if any are found.

Value

A data.frame with columns:

itemcode Integer FAO item code.

item Lowercase FAO item name.

j IMPACT j-code string (e.g. "jwhea").

Examples

```
## Not run:  
fmc <- get_fmc()  
head(fmc)  
  
## End(Not run)
```

getMapping

getMapping

Description

getMapping

Usage

```
getMapping(type = "region", file = "mapping.xlsx")
```

Arguments

type	"region" or "crops". Region returns six CGIAR regions + developed or developing countries + world. Crops returns IMPACT mapping for crops in their more user friendly name.
file	mapping file name

Value

mapping according to the type selected

Author(s)

Abhijeet Mishra

Examples

```
## Not run:
getMapping()

## End(Not run)
```

group1

Results for population and GDP

Description

Results for population and GDP

Usage

```
group1(gdx, mapping = "mapping.xlsx")
```

Arguments

gdx	GDX of an IMPACT run
mapping	mapping file name

Value

dataframe results for population and GDP

Author(s)

Abhijeet Mishra

Examples

```
## Not run:
group1()

## End(Not run)
```

group3	<i>Group 3 data processing (Area, yields, production, prices, net trade, demand)</i>
--------	--

Description

Group 3 data processing (Area, yields, production, prices, net trade, demand)

Usage

```
group3(gdx, indicator = "population", mapping = "mapping.xlsx")
```

Arguments

gdx	GDX of an IMPACT run
indicator	Which indicator to return in aggregated format. Defaults to "population". Available settings are "area", "production", "yield", "prices", "trade", "demand", "population", "perCapDemand", "bluewater" and "greenwater"
mapping	mapping file name

Value

dataframe results for Area, yields, production, prices, net trade, demand, demand_all (demand from all components)

Author(s)

Abhijeet Mishra

Examples

```
## Not run:  
group3()  
  
## End(Not run)
```

group4	<i>Results for per capital kcal availability</i>
--------	--

Description

Results for per capital kcal availability

Usage

```
group4(gdx, mapping = "mapping.xlsx")
```

Arguments

gdx	GDX of an IMPACT run
mapping	mapping file name

Value

dataframe results for per capital kcal availability

Author(s)

Abhijeet Mishra

Examples

```
## Not run:  
group4()  
  
## End(Not run)
```

group5	<i>Results for population at risk of hunger</i>
--------	---

Description

Results for population at risk of hunger

Usage

```
group5(gdx, mapping = "mapping.xlsx")
```

Arguments

gdx	GDX of an IMPACT run
mapping	Mapping file for regions

Value

Results for population at risk of hunger

Author(s)

Abhijeet Mishra

Examples

```
## Not run:  
group5()  
  
## End(Not run)
```

save_ggplot	<i>save_ggplot</i>
-------------	--------------------

Description

save_ggplot

Usage

```
save_ggplot(  
  p,  
  dest = tempdir(),  
  plot_name = "dummy",  
  ext = "png",  
  units = "in",  
  width = 7,  
  height = 7,  
  dpi = 300  
)
```

Arguments

<code>p</code>	"plot" to be saved. Has to be a ggplot object
<code>dest</code>	Destination directory of the plot
<code>plot_name</code>	Name of the plot to be exported
<code>ext</code>	Extension for saving the plot. e.g "png" or "eps" etc.
<code>units,width,height</code>	Plot size in units ("in", "cm", "mm", or "px"). If not supplied, uses the size of current graphics device.
<code>dpi</code>	Plot resolution. Also accepts a string input: "retina" (320), "print" (300), or "screen" (72). Applies only to raster output types

Value

NULL. The function only saves a ggplot object

Author(s)

Abhijeet Mishra

Examples

```
## Not run:  
save_ggplot()  
  
## End(Not run)
```

Index

`add_crops`, [2](#)

`add_regions`, [3](#)

`clean_filename`, [3](#)

`create_identifier_columns`, [4](#)

`get_dupes`, [5](#)

`get_fmc`, [5](#)

`getMapping`, [6](#)

`group1`, [6](#)

`group3`, [7](#)

`group4`, [8](#)

`group5`, [9](#)

`save_ggplot`, [9](#)