

# Package ‘inti’

November 25, 2020

**Type** Package

**Version** 0.1.2

**Title** Tools and Statistical Procedures in Plant Science

**Description** The ‘inti’ package is part of the ‘inkaverse’ project for developing different procedures and tools used in plant science and experimental designs. The main aim of the package is to support researchers during the planning of experiments and data collection (`tarpuv()`), data analysis and graphics (`yupana()`), and technical writing (`rtciles()`). Learn more about the ‘inkaverse’ project at <https://inkaverse.com/>.

**Date** 2020-11-24

**URL** <https://inkaverse.com/>, <https://github.com/flavjack/inti>

**BugReports** <https://github.com/flavjack/inti/issues/>

**Depends** shiny, ggplot2, dplyr, tidyr, tibble

**Imports** lme4, agricolae, FactoMineR, purrr, stringr, DT, emmeans

**Suggests** agridat, googlesheets4, knitr, rmarkdown

**VignetteBuilder** knitr

**License** GPL-3 | file LICENSE

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.1

**NeedsCompilation** no

**Author** Flavio Lozano-Isla [aut, cre] (<https://orcid.org/0000-0002-0714-669X>)

**Maintainer** Flavio Lozano-Isla <[flavjack@gmail.com](mailto:flavjack@gmail.com)>

**Repository** CRAN

**Date/Publication** 2020-11-25 07:30:06 UTC

## R topics documented:

create_rarticles	2
fieldbook_design	3
fieldbook_mvr	4
fieldbook_plex	5
fieldbook_report	7
fieldbook_reshape	9
fieldbook_summary	10
fieldbook_varlist	12
footnotes	13
H2cal	13
include_figure	16
include_table	17
info_figure	18
info_table	19
jc_tombola	20
means_lm	21
mean_comparison	22
metamorphosis	24
outliers_remove	25
plot_design	26
plot_diag	28
plot_smr	28
rarticles	30
tarpu	31
web_table	31
yupana	32
<b>Index</b>	<b>33</b>

---

create_rarticles	<i>Rarticles template</i>
------------------	---------------------------

---

### Description

Create a complete example for markdown o bookdown documents

### Usage

```
create_rarticles(
  path = NULL,
  type = c("markdown", "bookdown"),
  name = "manuscript",
  project = TRUE,
  server = c("local", "web")
)
```

**Arguments**

path	Directory for create the files.
type	Document type (default = "markdown").
name	Name of the main document (default = "manuscript").
project	Create a R project (default = TRUE). See details.
server	How create the dependencies (default = "local").

**Details**

Create all the files in the path directory. It is recommended use ".Rproj". After create the project Open the ".Rproj" file. Compile the document and you will see the example. If you already have a folder and select such as "bookdown" is necessary change the main (.rmd) file to index.rmd.

**Value**

folder and files

**Source**

<https://github.com/flavjack/rarticles>

---

fieldbook_design	<i>Field book experimental designs</i>
------------------	--

---

**Description**

Function to deploy experimental designs based in agricolae package

**Usage**

```
fieldbook_design(
  data,
  n_factors = 1,
  type = "crd",
  rep = 2,
  serie = 2,
  seed = 0
)
```

**Arguments**

data	Experimental design data frame with the factors and level. See examples.
n_factors	Number of factor in the experiment(default = 1). See details.
type	Type of experimental arrange (default = "crd"). See details.
rep	Number of replications in the experiment (default = 3).
serie	Digits in the plot id (default = 2).
seed	Replicability of draw results (default = 0) always random. See details.

**Details**

The function allows to include the arguments in the sheet that have the information of the design. You should include 2 columns in the sheet: {arguments} and {values}. See examples. The information will be extracted automatically and deploy the design. `n_factors = 1`: crd, rcbd, lsd, lattice. `n_factors = 2` (factorial): split\_crd, split\_rcbd `n_factors >= 2` (factorial): crd, rcbd, lsd.

**Value**

A list with the fieldbook design

**Examples**

```
library(inti)
library(googleworksheets4)
library(dplyr)

if (gs4_has_token()) {

  url <- paste0("https://docs.google.com/spreadsheets/d/"
    , "1ilw0NHT7mihAM-3U48KzkuMt927xe8ukX6rNuIw2fT0/edit#gid=0")
  # browseURL(url)
  gs <- as_sheets_id(url)

  (data <- gs %>%
    range_read("tarpuy"))

  data %>% inti::fieldbook_design(n_factors = 6)

}
```

---

 fieldbook\_mvr

*Multivariate Analysis*


---

**Description**

Multivariate analysis for PCA and HCPC

**Usage**

```
fieldbook_mvr(data, fb_smr, summary_by, groups)
```

**Arguments**

<code>data</code>	Field book data.
<code>fb_smr</code>	Summary of the variables in the fieldbook.
<code>summary_by</code>	Variables for group the analysis.
<code>groups</code>	Groups for color in PCA.

**Details**

Compute and plot information for multivariate analysis (PCA, HCPC and correlation).

**Value**

result and plots

**Examples**

```
library(inti)
library(googleheets4)
library(FactoMineR)

if (gs4_has_token()) {

url <- paste0("https://docs.google.com/spreadsheets/d/"
             , "15r7ZwcZZHbEg1t1F6gSFvCTFA-CFzVBWwg3mF1RyKPs")
# browseURL(url)
gs <- as_sheets_id(url)

(data <- gs %>%
  range_read("fb"))

(fb_smr <- gs %>%
  range_read("fbsm"))

mvr <- fieldbook_mvr(data, fb_smr
, summary_by = c("genotype", "treat"), groups = "treat")

FactoMineR::plot.PCA(mvr$pca
, choix = "ind"
, habillage = mvr$param$groups_n
, invisible = "quali"
)

FactoMineR::plot.HCPC(mvr$hcpc
, choice = "map"
, draw.tree = F
)

}
```

**Description**

Information for build a plan for an experiment (PLEX)

**Usage**

```
fieldbook_plex(  
  data = NULL,  
  idea = NULL,  
  goal = NULL,  
  hypothesis = NULL,  
  rationale = NULL,  
  objectives = NULL,  
  plan = NULL,  
  institutions = NULL,  
  researchers = NULL,  
  manager = NULL,  
  location = NULL,  
  altitude = NULL,  
  georeferencing = NULL,  
  environment = NULL,  
  start = NULL,  
  end = NULL,  
  about = NULL,  
  fieldbook = NULL,  
  album = NULL,  
  github = NULL,  
  nfactor = 2,  
  design = "rcbd",  
  rep = 3,  
  serie = 2,  
  seed = 0  
)
```

**Arguments**

data	Data with the fieldbook information.
idea	How the idea was born.
goal	The main goal of the project.
hypothesis	What are the expected results.
rationale	Based in which evidence is planned the experiment.
objectives	The objectives of the project.
plan	General description of the project (M & M).
institutions	Institutions involved in the project.
researchers	Persons involved in the project.
manager	Persons responsible of the collection of the data.

location	Location of the project.
altitude	Altitude of the experiment (m.a.s.l).
georeferencing	Georeferencing information.
environment	Environment of the experiment (greenhouse, lab, etc).
start	The date of the start of the experiments.
end	The date of the end of the experiments.
about	Short description of the project.
fieldbook	Name or ID for the fieldbook/project.
album	link with the photos of the project.
github	link with the github repository.
nfactor	Number of factors for the design.
design	Type of design.
rep	Number of replication.
serie	Number of digits in the plots.
seed	Seed for the randomization.

### Details

Provide the information available.

### Value

data frame or list of arguments:

1. info
2. variables
3. design
4. logbook
5. timetable
6. budget

---

fieldbook\_report

*Fieldbook report*

---

### Description

Function to create a complete report of the fieldbook

**Usage**

```
fieldbook_report(
  data,
  fb_smr,
  variable,
  model_facts,
  dotplot_groups = NULL,
  model_diag = FALSE
)
```

**Arguments**

<code>data</code>	Field book data.
<code>fb_smr</code>	Summary of the variables in the fieldbook.
<code>variable</code>	Response variable.
<code>model_facts</code>	Model used for the experimental design.
<code>dotplot_groups</code>	Factors to compare. See details.
<code>model_diag</code>	Diagnostic for model (default = FALSE).

**Details**

For compare the factors you should use ":". For example, to compare treatment1 and treatment2:  
treatment1:treatment2.

**Value**

data frame

**Examples**

```
library(inti)
library(googlesheets4)

if (gs4_has_token()) {
  url <- paste0("https://docs.google.com/spreadsheets/d/"
    , "15r7ZwcZZHbEglt1F6gSFvCTFA-CFzVBWwg3mF1RyKPs/edit#gid=172957346")
  # browseURL(url)
  gs <- as_sheets_id(url)

  (data <- gs %>%
    range_read("fb"))

  (fb_smr <- gs %>%
    range_read("fbsm"))

  report <- fieldbook_report(data
    , fb_smr = fb_smr
```



```
      , variable = "LA"  
      , dotplot_groups = "genotype"  
      , model_diag = TRUE  
    )  
  report  
}
```

---

fieldbook\_reshape      *Fieldbook reshape*

---

## Description

Function to reshape fieldbook according a separation character

## Usage

```
fieldbook_reshape(  
  data,  
  last_factor,  
  sep,  
  new_colname,  
  from_var = NULL,  
  to_var = NULL,  
  exc_factors = NULL  
)
```

## Arguments

data	Field book raw data.
last_factor	The last factor in your field book.
sep	Character that separates the last value.
new_colname	The new name for the column created.
from_var	The first variable in case you want to exclude several. variables.
to_var	The last variable in case you want to exclude several variables.
exc_factors	Factor to exclude during the reshape.

## Details

If you variable name is `variable_evaluation_rep`. The reshape function will help to create the column `rep` and the new variable name will be `variable_evaluation`.

## Value

data frame

## Examples

```
library(inti)
library(googleheets4)

if (gs4_has_token()) {

url <- paste0("https://docs.google.com/spreadsheets/d/"
              , "1lslvNuebfI0YQw3KMz7clN8nIs3x8G9Da63E1em7J5o/")
# browseURL(url)
gs <- as_sheets_id(url)

(data <- gs %>%
  range_read("fb"))

fbrs <- fieldbook_reshape(data = data
                          , last_factor = "imbt"
                          , sep = "_"
                          , new_colname = "rep"
                          , from_var = "sch_0_1"
                          , to_var = "swc_0_1"
                          , exc_factors = "bar_code"
                          )

fbrs

}
```

---

fieldbook\_summary

*Fieldbook summary*

---

## Description

Function to deploy a summary of the variables

## Usage

```
fieldbook_summary(
  data,
  last_factor,
  model_facts,
  comp_facts,
  test_comp = "SNK",
  sig_level = 0.05
)
```

**Arguments**

<code>data</code>	Field book data.
<code>last_factor</code>	The last factor in your fieldbook.
<code>model_facts</code>	Model used for the experimental design.
<code>comp_facts</code>	Factors to compare. See details.
<code>test_comp</code>	Comparison test (default = "SNK"). Others: "TUKEY", "DUNCAN".
<code>sig_level</code>	Significance level for the analysis (default = 0.05).

**Details**

For compare the factors you should use ":". For example, to compare treatment1 and treatment2:  
treatment1:treatment2.

**Value**

data frame

**Examples**

```
library(inti)
library(googleheets4)

if (gs4_has_token()) {

  url <- paste0("https://docs.google.com/spreadsheets/d/"
               , "15uwCgQRtR01B3FJaZBE8t_0b0C_8Bbey9ccwV1ZH0jg")
  # browseURL(url)
  gs <- as_sheets_id(url)

  (data <- gs %>%
    range_read("fb"))

  fbsm <- fieldbook_summary(data
    , last_factor = "dosis"
    , model_facts = "bloque + cultivar*fuelleN*dosis"
    , comp_facts = "cultivar:fuelleN:dosis"
    )

  fbsm

  # fbsm %>% write_sheet(ss = gs, sheet = "fbsm")

}
```

---

fieldbook\_varlist      *Fieldbook variable list*

---

### Description

Function to include the variables to evaluate in the fieldbook design.

### Usage

```
fieldbook_varlist(fieldbook, varlist = NULL)
```

### Arguments

fieldbook	Data frame with the fieldbook.
varlist	Data frame with the variables information. See examples.

### Details

The function allows to include the arguments in the sheet that have the information of the variables. You should include 3 columns in the sheet: {abbreviation}, {evaluation} and {sampling}. See examples. The information will be extracted automatically and deploy the list of variable for the fieldbook design. If you will be evaluate the experimental unit one time you should set 1.

### Value

data frame

### Examples

```
library(inti)
library(googleworksheets4)

if (gs4_has_token()) {

  url <- paste0("https://docs.google.com/spreadsheets/d/"
    , "1a82XIbTeWPC4pwvu9Zj14qrGitGQ_B-mf3-w67VS4-Q/edit#gid=0")
  # browseURL(url)
  gs <- as_sheets_id(url)

  (design <- gs %>%
    range_read("design"))

  (varlist <- gs %>%
    range_read("varlist"))

  fieldbook <- design %>%
    inti::fieldbook_design() %>%
    inti::fieldbook_varlist(varlist)
```

```

fieldbook
}

```

---

footnotes	<i>Table footnote</i>
-----------	-----------------------

---

### Description

Include tables footnotes and symbols for kables in pandoc format

### Usage

```
footnotes(table, notes = NULL, label = "Note:", notation = "alphabet")
```

### Arguments

table	Kable output in pandoc format.
notes	Footnotes for the table.
label	Label for start the footnote.
notation	Notation for the footnotes (default = "alphabet"). See details.

### Details

You should use the pandoc format `kable(format = "pipe")`. You can add the footnote symbol using `{hyphen}` in your table. `notation` could be use: "alphabet", "number", "symbol", "none".

### Value

Table with footnotes for word and html documents

---

H2cal	<i>Heritability in plant breeding</i>
-------	---------------------------------------

---

### Description

Heritability in plant breeding on a genotype difference basis

**Usage**

```
H2cal(
  data,
  trait,
  gen.name,
  rep.n,
  loc.n = 1,
  year.n = 1,
  loc.name = NULL,
  year.name = NULL,
  ran.model,
  fix.model,
  summary = FALSE,
  emmeans = FALSE,
  weights = NULL,
  plot_diag = FALSE,
  plot_dots = NULL,
  outliers.rm = FALSE
)
```

**Arguments**

<code>data</code>	Experimental design data frame with the factors and traits.
<code>trait</code>	Name of the trait.
<code>gen.name</code>	Name of the genotypes.
<code>rep.n</code>	Number of replications in the experiment.
<code>loc.n</code>	Number of locations (default = 1). See details.
<code>year.n</code>	Number of years (default = 1). See details.
<code>loc.name</code>	Name of the location (default = NULL). See details.
<code>year.name</code>	Name of the years (default = NULL). See details.
<code>ran.model</code>	The random effects in the model. See examples.
<code>fix.model</code>	The fixed effects in the model. See examples.
<code>summary</code>	Print summary from random model (default = FALSE).
<code>emmeans</code>	Use emmeans for calculate the BLUEs (default = FALSE).
<code>weights</code>	an optional vector of ‘prior weights’ to be used in the fitting process (default = NULL).
<code>plot_diag</code>	Show diagnostic plots (default = FALSE).
<code>plot_dots</code>	Show dotplot genotypes vs trait (default = NULL). See examples.
<code>outliers.rm</code>	Remove outliers (default = FALSE). See references.

## Details

The function allows to made the calculation for individual or multi-environmental trials (MET) using the fixed and random model.

1. The variance components.
2. Heritability under three approaches: Standard, Cullis and Piepho.
3. Best Linear Unbiased Predictors (BLUPs).
4. Best Linear Unbiased Estimators (BLUEs).
5. Outliers remove.

For individual experiments is necessary provide the `trait`, `gen.name`, `rep.n`.

For MET experiments you should `loc.n` and `loc.name` and/or `year.n` and `year.name` according your experiment.

The blues calculation is based in the pairwise comparison and its could takes time according the number of the genotypes.

You can specify as `blues = FALSE` for calculate the variance components and blups faster.

For more information review the references.

## Value

list

## Author(s)

Maria Belen Kistner

Flavio Lozano Isla

## References

Schmidt, P., J. Hartung, J. Bennewitz, and H.P. Piepho. 2019. Heritability in Plant Breeding on a Genotype Difference Basis. *Genetics* 212(4).

Schmidt, P., J. Hartung, J. Rath, and H.P. Piepho. 2019. Estimating Broad Sense Heritability with Unbalanced Data from Agricultural Cultivar Trials. *Crop Science* 59(2).

Bernal Vasquez, Angela Maria, et al. "Outlier Detection Methods for Generalized Lattices: A Case Study on the Transition from ANOVA to REML." *Theoretical and Applied Genetics*, vol. 129, no. 4, Apr. 2016.

## Examples

```
library(inti)
library(agridat)

dt <- john.alpha
hr <- H2cal(data = dt
            , trait = "yield"
            , gen.name = "gen")
```

```

      , rep.n = 3
      , fix.model = "1 + rep + (1|rep:block) + gen"
      , ran.model = "1 + rep + (1|rep:block) + (1|gen)"
      , plot_diag = TRUE
      , emmeans = TRUE
      , outliers.rm = TRUE
    )

hr$tabsmr
hr$blues
hr$blups

```

---

include\_figure      *Figure with caption and notes*

---

### Description

Include figures with title and notes using a data base

### Usage

```
include_figure(data = NULL, figure, caption = NA, notes = NA, label = NA)
```

### Arguments

data	Data frame with the figures information. See details.
figure	Name or path of the figure.
caption	Manual figure caption (default = NA).
notes	Manual figure notes (default = NA).
label	Label for start the footnote (default = NA).

### Details

The data frame information result from 'info\_figure' output.

### Value

Figure with caption and notes

### Examples

```

library(inti)

fig <- info_figure(caption = "caption test."
                  , notes = "note test."
                  , label = "_Source:_")

```



```

      , url = "https://devblackops.io/images/testing.jpg"
      , path = "man/figures/logo.png"
    )

# use this in r chunk `fig.caption = fig$caption`

fig %>% include_figure()

```

---

include\_table

*Table with footnotes*


---

### Description

Include tables with notes and titles for word and html documents

### Usage

```

include_table(
  data = NULL,
  caption = NA,
  notes = NA,
  label = NA,
  notation = "none"
)

```

### Arguments

data	Data frame.
caption	Table caption (default = NULL). See details.
notes	Footnotes for the table (default = NA). See details.
label	Label for start the footnote (default = NA).
notation	Notation for the symbols and footnotes (default = "none"). See details.

### Details

For caption and notes you can include caption and notes in the last rows of your data frame. Where the information will be extracted automatically for include in the formatted table. You can add the footnote symbol using {hyphen} in your table. notation could be use: "alphabet", "number", "none".

### Value

Table with caption and footnotes

**Examples**

```

library(inti)

tab <- data.frame(
  x = rep_len(1, 5)
  , y = rep_len(3, 5)
  , z = rep_len("c", 5)
)

table <- tab %>%
info_table(
  caption = "Figure caption"
  , notes = "test note"
)

table %>% inti::include_table()

```

---

info\_figure

*Figure information*


---

**Description**

Describe figures information for export or print

**Usage**

```

info_figure(
  caption = NA,
  notes = NA,
  url = NA,
  path = NA,
  label = NA,
  nrows = 50
)

```

**Arguments**

caption	Figure caption.
notes	Figure notes.
url	Figure url or link.
path	Figure local path.
label	Label for notes (default = NA)
nrows	Free rows in the export table.

**Value**

data frame

**Examples**

```
finfo <- info_figure(
  caption = "Figure caption."
  , notes = "Test note"
  , url = "www.image.dir"
  , label = "Source:"
)
```

---

info_table	<i>Table information</i>
------------	--------------------------

---

**Description**

Add table information for include\_table

**Usage**

```
info_table(data, caption = NA, notes = NA, label = NA, notation = "none")
```

**Arguments**

data	Data frame
caption	Table caption.
notes	Table notes.
label	Note label.
notation	Notation for the symbols and footnotes (default = "none"). See details.

**Details**

You can add the footnote symbol using {hyphen} in your table. notation could be use: "alphabet", "number", "none".

**Value**

data frame

## Examples

```
library(inti)

tab <- data.frame(
  x = rep_len(1, 5)
  , y = rep_len(3, 5)
  , z = rep_len("c", 5)
)

table <- tab %>%
  info_table(
    caption = "Figure caption"
    , notes = "test note"
  )

table$info
```

---

jc\_tombola

*Journal Club Tombola*

---

## Description

Function for arrange journal club schedule

## Usage

```
jc_tombola(
  data,
  members,
  group,
  gr_lvl,
  status,
  st_lvl,
  frq,
  date,
  seed = NULL
)
```

## Arguments

data	Data frame with the members and their information.
members	Columns with the members names.
group	Column for arrange the group.
gr_lvl	Levels in the groups for the arrange. See details.
status	Column with the status of the members.

st_lvl	Level to confirm the assistance in the JC. See details.
frq	Number of the day for each session.
date	Date when start the first session of JC.
seed	Number for replicate the results (default = date).

### Details

The function could consider n levels for gr\_lvl. In the case of two level the third level will be both. The suggested levels for st\_lvl are: active or spectator. Only the active members will enter in the schedule.

### Value

data frame with the schedule for the JC

### Examples

---

means_lm	<i>Mean comparison</i>
----------	------------------------

---

### Description

Function to calculate the means using Linear models (lm or aov)

### Usage

```
means_lm(
  data,
  trait,
  model,
  comparison,
  test = "SNK",
  anova = FALSE,
  plot_diag = FALSE,
  plot_treat = NULL,
  plot_groups = NULL,
  tab_vars = NULL,
  sep = NULL,
  digits = 3
)
```

**Arguments**

data	Experimental design data frame with the factors and traits.
trait	Name of the trait.
model	The effects in the model. See examples.
comparison	Factor for the comparisons.
test	Test comparison (default = "SNK"). Others: "TUKEY" & "DUNCAN".
anova	ANOVA table (default = FALSE).
plot_diag	Show diagnostic plots (default = FALSE).
plot_treat	Treat of comparison in dot plot (default = NULL). See example.
plot_groups	Group of comparison in dot plot (default = NULL). See example.
tab_vars	Specific the variables in the summary table (default = NULL).
sep	Separator between the variables when use tab_vars. See details.
digits	Number of digits in the table (default = 3).

**Details**

Using tab\_vars select the variables shown the adjusted mean table with the sep. Mean comparison at  $p < 0.05$ .

**Value**

A list with two objects:

1. Summary statistics.
2. Means with the mean comparisons test.

---

mean_comparison	<i>Mean comparison using data frames</i>
-----------------	--

---

**Description**

Function to compare treatment from lm or aov using data frames

**Usage**

```
mean_comparison(
  data,
  fb_smr,
  variable,
  model_facts,
  comp_facts,
  test_comp = "SNK",
  sig_level = 0.05,
  graph_opts = FALSE,
  digits = 2
)
```

**Arguments**

data	Fieldbook data.
fb_smr	Summary of the variables in the fieldbook.
variable	Model used for the experimental design.
model_facts	Comparison test (default = "SNK"). Others: "TUKEY", "DUNCAN".
comp_facts	Significance level for the analysis (default = 0.05).
test_comp	Significance level for the analysis (default = 0.05).
sig_level	Significance level for the analysis (default = 0.05).
graph_opts	Include option in the table for graphs (default = FALSE).
digits	Number of digits in the table.

**Details**

For compare the factors you should use ":". For example, to compare treatment1 and treatment2:  
treatment1:treatment2.

**Value**

data frame

**Examples**

```
library(inti)
library(googleworksheets4)

if (gs4_has_token()) {

url <- paste0("https://docs.google.com/spreadsheets/d/"
              , "15r7ZwcZZHbEglt1F6gSFvCTFA-CFzVBWwg3mF1RyKPs")
# browseURL(url)
gs <- as_sheets_id(url)

(data <- gs %>%
  range_read("fb"))

fbsm <- fieldbook_summary(data
  , last_factor = "genotype"
  , model_facts = "treat*genotype"
  , comp_facts = "treat:genotype"
  )

mc <- mean_comparison(data
  , fb_smr = fbsm
  , variable = "HI"
  , graph_opts = TRUE
  )
```

```
table <- mc$comparison  
  
# fbsm %>% sheet_write(ss = gs, sheet = "fbsm")  
  
}
```

---

metamorphosis

*Transform data frame based in a dictionary*

---

### **Description**

Transform entire fieldbook according to data a dictionary

### **Usage**

```
metamorphosis(fieldbook, dictionary, from, to, index, colnames)
```

### **Arguments**

fieldbook	Data frame with the original information.
dictionary	Data frame with new names and categories. See details.
from	Column of the dictionary with the original names.
to	Column of the dictionary with the new names.
index	Column of the dictionary with the type and level of the variables.
colnames	Character vector with the name of the columns.

### **Details**

The function require at least three columns.

1. Original names (from).
2. New names (to).
3. Variable type (index).

### **Value**

List with two objects. 1. New data frame. 2. Dictionary.



**Examples**

```

library(googleheets4)
library(dplyr)

if (gs4_has_token()) {

url <- paste0("https://docs.google.com/spreadsheets/d/"
, "1lslvNuebfI0YQw3KMz7c1N8nIs3x8G9Da63E1em7J5o")
gs <- as_sheets_id(url)
# browseURL(url)

fb_old <- gs %>%
  range_read("fb_old") %>%
  select(ID:SNum) %>%
  filter(SS == "S1")

dic <- gs %>% range_read("var_old") %>%
  slice(1:20)

mtm <- metamorphosis(fieldbook = fb_old
                     , dictionary = dic
                     , from = "old_name"
                     , to = "Abbreviation"
                     , index = "Type"
                     , colnames = c("colname")
                     )

fb_new <- mtm$fieldbook

}

```

---

outliers\_remove

*Remove outliers*


---

**Description**

Use the method M4 in Bernal Vasquez (2016). Bonferroni Holm test to judge residuals standardized by the re scaled MAD (BH MADR).

**Usage**

```
outliers_remove(data, trait, model)
```

**Arguments**

data	Experimental design data frame with the factors and traits.
trait	Name of the trait.
model	The fixed or random effects in the model.

**Details**

Function to remove outliers in MET experiments

**Value**

list. 1. Table with date without outliers. 2. The outliers in the dataset.

**References**

Bernal Vasquez, Angela Maria, et al. "Outlier Detection Methods for Generalized Lattices: A Case Study on the Transition from ANOVA to REML." *Theoretical and Applied Genetics*, vol. 129, no. 4, Apr. 2016.

**Examples**

```
library(inti)
library(agridat)

outliers <- outliers_remove(
  data = john.alpha
  , trait = "yield"
  , model = "rep + (1|rep:block) + gen"
)

outliers
```

---

plot\_design

*Fieldbook plot experimental designs*

---

**Description**

Plot fieldbook sketch designs based in experimental design

**Usage**

```
plot_design(
  data,
  factor,
  dim = NULL,
  fill = "plots",
  xlab = NULL,
  ylab = NULL,
  glab = NULL
)
```

**Arguments**

data	Experimental design data frame with the factors and level. See examples.
factor	Vector with the name of the columns with the factors.
dim	Dimension for reshape the design arrangement.
fill	Value for fill the experimental units (default = "plots").
xlab	Title for x axis.
ylab	Title for y axis.
glab	Title for group axis.

**Details**

The function allows to plot the experimental design according the field experiment design.

**Value**

plot

**Examples**

```

library(inti)
library(googleworksheets4)
library(purrr)

if (gs4_has_token()) {

url <- paste0("https://docs.google.com/spreadsheets/d/"
, "1ilw0NHT7mihAM-3U48KzkuMt927xe8ukX6rNuIw2fT0/edit#gid=0")
# browseURL(url)
gs <- as_sheets_id(url)

(fb <- gs %>%
  range_read("tarpuy"))

data <- fb %>% inti::fieldbook_design(n_factors = 2
                                     , type = "rcbd"
                                     , rep = 3) %>% pluck("design")

plot_design(data
            , factor = "temp"
            , dim = "block"
            )

plot_design(data
            , factor = "temp"
            , dim = "9x5"
            )

}

```

---

plot_diag	<i>Diagnostic plots</i>
-----------	-------------------------

---

**Description**

Function to plot the diagnostic of models

**Usage**

```
plot_diag(model)
```

**Arguments**

model            Statistical model.

**Value**

plots

---

plot_smr	<i>Plot summary data</i>
----------	--------------------------

---

**Description**

Graph summary data into bar o line plot

**Usage**

```
plot_smr(  
  data,  
  type = NULL,  
  x = NULL,  
  y = NULL,  
  groups = NULL,  
  xlab = NULL,  
  ylab = NULL,  
  glab = NULL,  
  limits = NULL,  
  brakes = NULL,  
  sig = NULL,  
  error = NULL,  
  legend = NULL  
)
```

**Arguments**

data	Data frame with summary data. See details.
type	Type of plot (default = "bar"). Others: "line".
x	Variable in the x axis.
y	Variable in the y axis.
groups	Groups the x axis (legend).
xlab	Label of x axis.
ylab	Label of y axis.
glab	Label of groups or legend.
limits	Limits max and min.
brakes	Units for break the limits.
sig	Comparison test values (default = "sig").
error	Variable used for error bars.
legend	Position of legend (default = "top"). Others: "left", "right", "bottom", "none".

**Details**

If the table is a out put of `mean_comparison(graph_opts = TRUE)` function. Its contain all the parameter for the plot.

**Value**

plot

**Examples**

```
library(inti)
library(googleworksheets4)

if (gs4_has_token()) {

url <- paste0("https://docs.google.com/spreadsheets/d/"
              , "15r7ZwcZZHbEg1t1F6gSFvCTFA-CFzVBWwg3mF1RyKPs")
# browseURL(url)
gs <- as_sheets_id(url)

(data <- gs %>%
  range_read("LA"))

plot_smr(data)

plot_smr(data
  , type = "bar"
  , limits = c(0, 14000)
  , brakes = 2000
  , sig = "sig")
```

```
      , error = "ste"  
    )  
  }  
}
```

---

rticles

*Build markdown o bookdown template*

---

### **Description**

Invoke RStudio addin to create markdown documents

### **Usage**

```
rticles(dependencies = FALSE)
```

### **Arguments**

dependencies    Install package dependencies for run the app

### **Details**

Create all the files in the present work directory. It is recommended use ".Rproj" for bookdown documents. After create the ".Rproj" file. Open the ".Rprj" and you should have the possibility to compile the book.

### **Value**

Shiny app

### **Examples**

```
if(interactive()){  
  inti::rticles()  
}
```

---

tarpuy	<i>Fieldbook experimental designs</i>
--------	---------------------------------------

---

**Description**

Invoke RStudio addin to create field book designs

**Usage**

```
tarpuy(dependencies = FALSE)
```

**Arguments**

dependencies    Install package dependencies for run the app

**Details**

Tarpuy allow to create experimental designs under an interactive app.

**Value**

Shiny app

**Examples**

```
if(interactive()){  
  inti::tarpuy()  
}
```

---

web_table	<i>HTML tables for markdown documents</i>
-----------	---

---

**Description**

Export tables with download, pasta and copy buttons

**Usage**

```
web_table(data, caption = NULL, digits = 2, rnames = FALSE, buttons = NULL)
```

**Arguments**

data	Dataset.
caption	Title for the table.
digits	Digits number in the table exported.
rnames	Row names.
buttons	Buttons: "excel", "copy" or "none". Default c("excel", "copy")

**Value**

table in markdown format for html documents

---

yupana	<i>Interactive data analysis</i>
--------	----------------------------------

---

**Description**

Invoke RStudio addin to analyze and graph experimental design data

**Usage**

```
yupana(dependencies = FALSE)
```

**Arguments**

dependencies Install package dependencies for run the app

**Details**

Yupana: data analysis and graphics for experimental designs.

**Value**

Shiny app

**Examples**

```
if(interactive()){
  inti::yupana()
}
```



# Index

[create\\_rarticles](#), 2

[fieldbook\\_design](#), 3  
[fieldbook\\_mvr](#), 4  
[fieldbook\\_plex](#), 5  
[fieldbook\\_report](#), 7  
[fieldbook\\_reshape](#), 9  
[fieldbook\\_summary](#), 10  
[fieldbook\\_varlist](#), 12  
[footnotes](#), 13

[H2cal](#), 13

[include\\_figure](#), 16  
[include\\_table](#), 17  
[info\\_figure](#), 18  
[info\\_table](#), 19

[jc\\_tombola](#), 20

[mean\\_comparison](#), 22  
[means\\_lm](#), 21  
[metamorphosis](#), 24

[outliers\\_remove](#), 25

[plot\\_design](#), 26  
[plot\\_diag](#), 28  
[plot\\_smr](#), 28

[rticles](#), 30

[tarpuy](#), 31

[web\\_table](#), 31

[yupana](#), 32