Package ‘inti’

February 26, 2025

Type Package

Version 0.6.7

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 mean aim of the package is to support researchers during the planning of experiments and data collection (tarpuy()), data analysis and graphics (yupana()) , and technical writing.
Learn more about the 'inkaverse' project at <https://inkaverse.com/>.

Date 2025-02-26

URL https://inkaverse.com/, https://github.com/flavjack/inti

BugReports https://github.com/flavjack/inti/issues/

Depends shiny, ggplot2, dplyr, tidyr, tibble, R (>= 4.1.0)

Imports lme4, agricolae, FactoMineR, emmeans, purrr, stringr, googlesheets4, DT

Suggests gsheet, cowplot, knitr, rmarkdown, bookdown

VignetteBuilder knitr

License GPL-3 | file LICENSE

LazyData true

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

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

Maintainer Flavio Lozano-Isla <`flozanoisla@gmail.com`>

Repository CRAN

Date/Publication 2025-02-26 11:30:06 UTC

Contents

`colortext` *Colourise text for display in the terminal*

Description

If R is not currently running in a system that supports terminal colours the text will be returned unchanged.

Usage

```
colortext(text, fg = "red", bg = NULL)
```

Arguments

`text`	character vector
`fg`	foreground colour, defaults to white
`bg`	background colour, defaults to transparent

Details

Allowed colours are: black, blue, brown, cyan, dark gray, green, light blue, light cyan, light gray, light green, light purple, light red, purple, red, white, yellow

Author(s)

testthat package

Examples

```
print(colortext("Red", "red"))
cat(colortext("Red", "red"), "\n")
cat(colortext("White on red", "white", "red"), "\n")
```

`design_noreps` *Experimental design without replications*

Description

Function to deploy field-book experiment without replications

Usage

```
design_noreps(
  factors,
  type = "sorted",
  zigzag = FALSE,
  nrows = NA,
  serie = 100,
  seed = NULL,
  fbname = "inkaverse",
  qrcode = "{fbname}{plots}{factors}"
)
```

Arguments

`factors`	Lists with names and factor vector [list].
`type`	Randomization in the list [string: sorted, unsorted]
`zigzag`	Experiment layout in zigzag [logic: FALSE].
`nrows`	Experimental design dimension by rows [numeric: value]
`serie`	Number to start the plot id [numeric: 1000].
`seed`	Replicability from randomization [numeric: NULL].
`fbname`	Bar code prefix for data collection [string: "inkaverse"].
`qrcode`	[string: "{fbname}{plots}{factors}"] String to concatenate the qr code.

Value

A list with the field-book design and parameters

Examples

```
## Not run:

library(inti)

factores <- list("geno" = c(1:99))

fb <- design_noreps(factors = factores
                    , type = "sorted"
                    , zigzag = F
                    , nrows = 10
                    )

dsg <- fb$fieldbook

fb %>%
  tarpuy_plotdesign(fill = "plots")

fb$parameters
```

```
## End(Not run)
```

`design_repblock` *Experimental design in CRD and RCBD*

Description

Function to deploy field-book experiment for CRD and RCBD

Usage

```
design_repblock(
  nfactors = 1,
  factors,
  type = "crd",
  rep = 3,
  zigzag = FALSE,
  nrows = NA,
  serie = 100,
  seed = NULL,
  fbname = "inkaverse",
  qrcode = "{fbname}{plots}{factors}"
)
```

Arguments

`nfactors`	Number of factor in the experiment [numeric: 1].
`factors`	Lists with names and factor vector [list].
`type`	Type of experimental arrange [string: "crd" "rcbd" "lsd"]
`rep`	Number of replications in the experiment [numeric: 3].
`zigzag`	Experiment layout in zigzag [logic: F].
`nrows`	Experimental design dimension by rows [numeric: value]
`serie`	Number to start the plot id [numeric: 100].
`seed`	Replicability from randomization [numeric: NULL].
`fbname`	Bar code prefix for data collection [string: "inkaverse"].
`qrcode`	[string: "{fbname}{plots}{factors}"] String to concatenate the qr code.

Value

A list with the field-book design and parameters

Examples

```
## Not run:

library(inti)

factores <- list("geno" = c("A", "B", "C", "D", "D", 1, NA, NA, NULL, "NA")
                 , "salt stress" = c(0, 50, 200, 200, "T0", NA, NULL, "NULL")
                 , time = c(30, 60, 90)
                 )

fb <-design_repblock(nfactors = 2
                     , factors = factores
                     , type = "rcbd"
                     , rep = 5
                     , zigzag = T
                     , seed = 0
                     , nrows = 20
                     , qrcode = "{fbname}{plots}{factors}"
                     )

dsg <- fb$fieldbook

fb %>%
  tarpuy_plotdesign(fill = "plots")

fb$parameters


## End(Not run)
```

---

figure2qmd *Figure to Quarto format*

Description

Use Articul8 Add-ons from Google docs to build Rticles

Usage

```
figure2qmd(text, path = ".", opts = NA)
```

Arguments

`text`	Markdown text with figure information [string]
`path`	Image path for figures [path: "." (base directory)]
`opts`	chunk options in brackets [string: NA]

Details

Quarto option can be included in the title using "{{}}" separated by commas

Value

string mutated

`figure2rmd` *Figure to Rmarkdown format*

Description

Use Articul8 Add-ons from Google docs to build Rticles

Usage

```
figure2rmd(text, path = ".", opts = NA)
```

Arguments

`text`	String with the table information
`path`	Path of the image for the figure
`opts`	chunk options in brackets.

Value

Mutated string

`footnotes` *Footnotes in tables*

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 `{hypen}` in your table. `notation` could be use: "alphabet", "number", "symbol", "none".

Value

Table with footnotes for word and html documents

`gdoc2qmd` *Google docs to Rmarkdown*

Description

Use Articul8 Add-ons from Google docs to build Rticles

Usage

```
gdoc2qmd(file, export = NA, format = "qmd", type = "asis")
```

Arguments

`file`	Zip file path from Articul8 exported in md format [path]
`export`	Path to export the files [path: NA (file directory)]
`format`	Output format [string: "qmd" "rmd"]
`type`	output file type [strig: "asis" "list", "listfull", "full"]

Details

Document rendering until certain point: "#| end" Include for next page: "#| newpage" You can include the cover page params using "#|" in a Google docs table

Value

path

`H2cal` *Broad-sense heritability in plant breeding*

Description

Heritability in plant breeding on a genotype difference basis

Usage

```
H2cal(
  data,
  trait,
  gen.name,
  rep.n,
  env.n = 1,
  year.n = 1,
  env.name = NULL,
  year.name = NULL,
  fixed.model,
  random.model,
  summary = FALSE,
  emmeans = FALSE,
  weights = NULL,
  plot_diag = FALSE,
  outliers.rm = FALSE,
  trial = NULL
)
```

Arguments

`data`	Experimental design data frame with the factors and traits.
`trait`	Name of the trait.
`gen.name`	Name of the genotypes.
`rep.n`	Number of replications in the experiment.
`env.n`	Number of environments (default = 1). See details.
`year.n`	Number of years (default = 1). See details.
`env.name`	Name of the environments (default = NULL). See details.
`year.name`	Name of the years (default = NULL). See details.
`fixed.model`	The fixed effects in the model (BLUEs). See examples.
`random.model`	The random effects in the model (BLUPs). See examples.
`summary`	Print summary from random model (default = FALSE).
`emmeans`	Use emmeans for calculate the BLUEs (default = FALSE).
`weights`	an optional vector of ‘prior weights’ to be used in the fitting process (default = NULL).

`plot_diag`	Show diagnostic plots for fixed and random effects (default = FALSE). Options: "base", "ggplot". .
`outliers.rm`	Remove outliers (default = FALSE). See references.
`trial`	Column with the name of the trial in the results (default = NULL).

Details

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

1. The variance components based in the random model and the population summary information based in the fixed model (BLUEs).

2. Heritability under three approaches: Standard (ANOVA), Cullis (BLUPs) and Piepho (BLUEs).

3. Best Linear Unbiased Estimators (BLUEs), fixed effect.

4. Best Linear Unbiased Predictors (BLUPs), random effect.

5. Table with the outliers removed for each model.

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

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

The BLUEs calculation based in the pairwise comparison could be time consuming with the increase of the number of the genotypes. You can specify `emmeans = FALSE` and the calculate of the BLUEs will be faster.

If `emmeans = FALSE` you should change 1 by 0 in the fixed model for exclude the intersect in the analysis and get all the genotypes BLUEs.

For more information review the references.

Value

list

Author(s)

Maria Belen Kistner

Flavio Lozano Isla

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.

Buntaran, H., Piepho, H., Schmidt, P., Ryden, J., Halling, M., and Forkman, J. (2020). Cross validation of stagewise mixed model analysis of Swedish variety trials with winter wheat and spring barley. Crop Science, 60(5).

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).

Tanaka, E., and Hui, F. K. C. (2019). Symbolic Formulae for Linear Mixed Models. In H. Nguyen (Ed.), Statistics and Data Science. Springer.

Zystro, J., Colley, M., and Dawson, J. (2018). Alternative Experimental Designs for Plant Breeding. In Plant Breeding Reviews. John Wiley and Sons, Ltd.

Examples

```
library(inti)

dt <- potato

hr <- H2cal(data = dt
            , trait = "stemdw"
            , gen.name = "geno"
            , rep.n = 5
            , fixed.model = "0 + (1|bloque) + geno"
            , random.model = "1 + (1|bloque) + (1|geno)"
            , emmeans = TRUE
            , plot_diag = FALSE
            , outliers.rm = TRUE
            )

 hr$tabsmr
 hr$blues
 hr$blups
 hr$outliers
```

`include_pdf` *Include PDF in markdown documents*

Description

Insert PDF files in markdown documents

Usage

```
include_pdf(file, width = "100%", height = "600")
```

Arguments

`file`	file path from pdf file.
`width`	width preview file.
`height`	height preview file.

Value

html code for markdown

`include_table` *Table with footnotes*

Description

Include tables with title and footnotes for word and html documents

Usage

```
include_table(table, caption = NA, notes = NA, label = NA, notation = "none")
```

Arguments

`table`	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") Others: "alphabet", "number", "symbol".

Value

Table with caption and footnotes

Examples

```
library(inti)

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

table %>% inti::include_table(
  caption = "Title caption b) line 0
  a) line 1
  b) line 2"
  , notes = "Footnote"
  , label = "Where:"
  )
```

`jc_tombola` *Journal Club Tombola*

Description

Function for arrange journal club schedule

Usage

```
jc_tombola(
  data,
  members,
  papers = 1,
  group = NA,
  gr_lvl = NA,
  status = NA,
  st_lvl = "active",
  frq = 7,
  date = NA,
  seed = NA
)
```

Arguments

`data`	Data frame withe members and their information.
`members`	Columns with the members names.
`papers`	Number of paper by meeting
`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 more levels using "both" or "all" will be the combination. 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

`mean_comparison` *Mean comparison test*

Description

Function to compare treatment from lm or aov using data frames

Usage

```
mean_comparison(
  data,
  response,
  model_factors,
  comparison,
  test_comp = "SNK",
  sig_level = 0.05
)
```

Arguments

`data`	Fieldbook data.
`response`	Model used for the experimental design.
`model_factors`	Factor in the model.
`comparison`	Significance level for the analysis (default = 0.05).
`test_comp`	Comparison test (default = "SNK"). Others: "TUKEY", "DUNCAN".
`sig_level`	Significance level for the analysis (default = 0.05).

Value

list

Examples

```
## Not run:

library(inti)
library(gsheet)

url <- paste0("https://docs.google.com/spreadsheets/d/"
              , "15r7ZwcZZHbEgltlF6gSFvCTFA-CFzVBWwg3mFlRyKPs/"
              , "edit#gid=172957346")
# browseURL(url)

fb <- gsheet2tbl(url)

mc <- mean_comparison(data = fb
                      , response = "spad_29"
```

```
                            , model_factors = "bloque* geno*treat"
                            , comparison = c("geno", "treat")
                            , test_comp = "SNK"
                            )
mc$comparison
mc$stat


## End(Not run)
```

`met` *Swedish cultivar trial data*

Description

The datasets were obtained from official Swedish cultivar tests. Dry matter yield was analyzed. All trials were laid out as alpha-designs with two replicates. Within each replicate, there were five to seven incomplete blocks.

Usage

```
met
```

Format

A data frame with 1069 rows and 8 variables:

zone Sweden is divided into three different agricultural zones: South, Middle, and North

location Locations: 18 location in the Zones

rep Replications (4): number of replication in the experiment

alpha Incomplete blocks (8) in the alpha-designs

cultivar Cultivars (30): genotypes evaluated

yield Yield in kg/ha

year Year (1): 2016

env enviroment (18): combination zone + location + year

Source

doi:10.1002/csc2.20177

`metamorphosis` *Transform fieldbooks 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.

`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, drop_na = TRUE)
```

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.
`drop_na`	drop NA values from the data.frame

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)

rmout <- potato %>% outliers_remove(
  data = .
  , trait ="stemdw"
  , model = "0 + treat*geno + (1|bloque)"
  , drop_na = FALSE
  )

rmout
```

`plot_diag` *Diagnostic plots*

Description

Function to plot the diagnostic of models

Usage

```
plot_diag(model, title = NA)
```

Arguments

`model`	Statistical model
`title`	Plot title

Value

plots

Examples

```
## Not run:

library(inti)

lm <- aov(stemdw ~ bloque + geno*treat, data = potato)

# lm <- potato %>% lme4::lmer(stemdw ~ (1|bloque) + geno*treat, data = .)

plot(lm, which = 1)
plot_diag(lm)[3]

plot(lm, which = 2)
plot_diag(lm)[2]

plot(lm, which = 3)
plot_diag(lm)[4]

plot(lm, which = 4)
plot_diag(lm)[1]


## End(Not run)
```

`plot_diagnostic` *Diagnostic plots*

Description

Function to plot the diagnostic of models

Usage

```
plot_diagnostic(data, formula, title = NA)
```

Arguments

`data`	Experimental design data frame with the factors and traits.
`formula`	Mixed model formula
`title`	Plot title

Value

plots

Examples

```
## Not run:

library(inti)

plot_diagnostic(data = potato
               , formula = stemdw ~ (1|bloque) + geno*treat)


## End(Not run)
```

`plot_raw` *Plot raw data*

Description

Function use the raw data for made a boxplot graphic

Usage

```
plot_raw(
  data,
  type = "boxplot",
  x,
  y,
  group = NULL,
  xlab = NULL,
  ylab = NULL,
  glab = NULL,
  ylimits = NULL,
  xlimits = NULL,
  xrotation = NULL,
  legend = "top",
  xtext = NULL,
  gtext = NULL,
  color = TRUE,
  linetype = 1,
  opt = NULL
)
```

Arguments

`data`	raw data
`type`	Type of graphic. "boxplot" or "scatterplot"
`x`	Axis x variable
`y`	Axis y variable
`group`	Group variable
`xlab`	Title for the axis x
`ylab`	Title for the axis y
`glab`	Title for the legend
`ylimits`	Limits and break of the y axis c(initial, end, brakes)
`xlimits`	For scatter plot. Limits and break of the x axis c(initial, end, brakes)
`xrotation`	Rotation in x axis c(angle, h, v)
`legend`	the position of legends ("none", "left", "right", "bottom", "top", or two-element numeric vector)
`xtext`	Text labels in x axis using a vector
`gtext`	Text labels in groups using a vector
`color`	Colored figure (TRUE), black & white (FALSE) or color vector
`linetype`	Line type for regression. Default = 0
`opt`	Add new layers to the plot

Details

You could add additional layer to the plot using "+" with ggplot2 options

Value

plot

Examples

```
## Not run:

library(inti)

fb <- potato

fb %>%
  plot_raw(type = "box"
           , x = "geno"
           , y = "twue"
           , group = NULL
           , ylab = NULL
           , xlab = NULL
           , glab = ""
           )
```

```
fb %>%
  plot_raw(type = "sca"
           , x = "geno"
           , y = "twue"
           , group = "treat"
           , color = c("red", "blue")
           )


## End(Not run)
```

`plot_smr` *Plot summary data*

Description

Graph summary data into bar o line plot

Usage

```
plot_smr(
  data,
  type = NULL,
  x = NULL,
  y = NULL,
  group = NULL,
  xlab = NULL,
  ylab = NULL,
  glab = NULL,
  ylimits = NULL,
  xrotation = c(0, 0.5, 0.5),
  xtext = NULL,
  gtext = NULL,
  legend = "top",
  sig = NULL,
  sigsize = 3,
  error = NULL,
  color = TRUE,
  opt = NULL
)
```

Arguments

`data`	Output from summary data
`type`	Type of graphic. "bar" or "line"
`x`	Axis x variable

`y`	Axis y variable
`group`	Group variable
`xlab`	Title for the axis x
`ylab`	Title for the axis y
`glab`	Title for the legend
`ylimits`	limits of the y axis c(initial, end, brakes)
`xrotation`	Rotation in x axis c(angle, h, v)
`xtext`	Text labels in x axis using a vector
`gtext`	Text labels in group using a vector
`legend`	the position of legends ("none", "left", "right", "bottom", "top", or two-element numeric vector)
`sig`	Column with the significance
`sigsize`	Font size in significance letters
`error`	Show the error bar ("ste" or "std")
`color`	colored figure (TRUE), black & white (FALSE) or color vector
`opt`	Add news layer to the plot

Details

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

You could add additional layer to the plot using "+" with ggplot2 options

Value

plot

Examples

```
## Not run:

library(inti)

fb <- potato

yrs <- yupana_analysis(data = fb
                       , response = "lfa"
                       , model_factors = "geno*treat"
                       , comparison = c("geno", "treat")
                       )

yrs$meancomp %>%
  plot_smr(type = "bar"
           , x = "geno"
           , y = "lfa"
           , xlab = ""
```

```
            , group = "treat"
            , glab = "Tratamientos"
            , error = "ste"
            , sig = "sig"
            #, ylimits = c(0, 1, 0.2)
            , color = c("red", "black")
            , gtext = c("Irrigado", "Sequia")
            )


## End(Not run)
```

`potato` *Water use efficiency in 15 potato genotypes*

Description

Experiment to evaluate the physiological response from 15 potatos genotypes under water deficit condition. The experiment had a randomized complete block design with five replications. The stress started at 30 day after planting.

Usage

```
potato
```

Format

A data frame with 150 rows and 17 variables:

treat Water deficit treatments: sequia, irrigado

geno 15 potato genotypes

bloque blocks for the experimentl design

spad_29 Relative chlorophyll content (SPAD) at 29 day after planting

spad_83 Relative chlorophyll content (SPAD) at 84 day after planting

rwc_84 Relative water content (percentage) at 84 day after planting

op_84 Osmotic potential (Mpa) at 84 day after planting

leafdw leaf dry weight (g)

stemdw stem dry weight (g)

rootdw root dry weight (g)

tubdw tuber dry weight (g)

biomdw total biomass dry weight (g)

hi harvest index

ttrans total transpiration (l)

wue water use effiency (g/l)

twue tuber water use effiency (g/l)

lfa leaf area (cm2)

`remove_outliers` *Remove outliers using mixed models*

Description

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

Usage

```
remove_outliers(data, formula, drop_na = FALSE, plot_diag = FALSE)
```

Arguments

`data`	Experimental design data frame with the factors and traits.
`formula`	mixed model formula.
`drop_na`	drop NA values from the data.frame
`plot_diag`	Diagnostic plot based in the raw and clean data

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)

rmout <- potato %>%
  remove_outliers(data = .
  , formula = stemdw ~ 0 + (1|bloque) + treat*geno
  , plot_diag = FALSE
  , drop_na = FALSE
  )
```

```
rmout
```

`split_folder` *Split folder*

Description

Function to split folder by size or number of elements

Usage

```
split_folder(
  folder,
  export,
  units = "megas",
  size = 500,
  zip = TRUE,
  remove = FALSE
)
```

Arguments

`folder`	Path of folder to split (path).
`export`	Path to export the split folders (path).
`units`	Units to split folder (string: "megas", "number").
`size`	Folder size by the units selected (numeric).
`zip`	Zip split folders (logical).
`remove`	Remove the split folder after zip (logical).

Value

zip files

Examples

```
## Not run:

split_folder("pictures/QUINOA 2018-2019 SC SEEDS EDWIN - CAMACANI/"
   , "pictures/split_num", remove = T, size = 400, units = "number")


## End(Not run)
```

`table2qmd` *Table to Quarto format*

Description

Use Articul8 Add-ons from Google docs to build Rticles

Usage

```
table2qmd(text, type = "asis")
```

Arguments

`text`	Markdown text with table information (string)
`type`	output file type [strig: "asis" "list", "listfull", "full"]

Value

string mutated

`table2rmd` *Table to Rmarkdown format*

Description

Use Articul8 Add-ons from Google docs to build Rticles

Usage

```
table2rmd(text, opts = NA)
```

Arguments

`text`	String with the table information
`opts`	chunk options in brackets.

Value

Mutated string

`tarpuy` *Interactive fieldbook designs*

Description

Invoke RStudio addin to create fieldbook 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()

}
```

`tarpuy_design` *Fieldbook experimental designs*

Description

Function to deploy experimental designs

Usage

```
tarpuy_design(
  data,
  nfactors = 1,
  type = "crd",
  rep = 2,
  zigzag = FALSE,
  nrows = NA,
  serie = 100,
  seed = NULL,
  fbname = NA,
  qrcode = "{fbname}{plots}{factors}"
)
```

Arguments

`data`	Experimental design data frame with the factors and level. See examples.
`nfactors`	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).
`zigzag`	Experiment layout in zigzag [logic: FALSE].
`nrows`	Experimental design dimension by rows [numeric: value]
`serie`	Number to start the plot id [numeric: 100].
`seed`	Replicability of draw results (default = 0) always random. See details.
`fbname`	Barcode prefix for data collection.
`qrcode`	[string: "{fbname}{plots}{factors}"] String to concatenate the qr code.

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. `nfactors` = 1: crd, rcbd, lsd, lattice. `nfactors` = 2 (factorial): split-crd, split-rcbd split-lsd `nfactors` >= 2 (factorial): crd, rcbd, lsd.

Value

A list with the fieldbook design

Examples

```
## Not run:

library(inti)
library(gsheet)
```

```
url <- paste0("https://docs.google.com/spreadsheets/d/"
         , "1510fOKj0g4CDEAFkrpFbr-zNMnle_Hou9O_wuf7Vdo4/edit?gid=1479851579#gid=1479851579")
# browseURL(url)

fb <- gsheet2tbl(url)

dsg <- fb %>% tarpuy_design()

dsg %>%
  tarpuy_plotdesign()


## End(Not run)
```

`tarpuy_plex` *Fieldbook plan information*

Description

Information for build a plan for an experiment (PLEX)

Usage

```
tarpuy_plex(
  data = NULL,
  title = NULL,
  objectives = NULL,
  hypothesis = NULL,
  rationale = NULL,
  references = NULL,
  plan = NULL,
  institutions = NULL,
  researchers = NULL,
  manager = NULL,
  location = NULL,
  altitude = NULL,
  georeferencing = NULL,
  environment = NULL,
  start = NA,
  end = NA,
  about = NULL,
  fieldbook = NULL,
  project = NULL,
  repository = NULL,
  manuscript = NULL,
  album = NULL,
  nfactor = 2,
  design = "rcbd",
```

```
  rep = 3,
  zigzag = FALSE,
  nrows = NA,
  serie = 100,
  seed = 0,
  qrcode = "{fbname}{plots}{factors}"
)
```

Arguments

`data`	Data with the fieldbook information.
`title`	Project title.
`objectives`	The objectives of the project.
`hypothesis`	What are the expected results.
`rationale`	Based in which evidence is planned the experiment.
`references`	References.
`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.
`project`	link for project.
`repository`	link to the repository.
`manuscript`	link for manuscript.
`album`	link with the photos of the project.
`nfactor`	Number of factors for the design.
`design`	Type of design.
`rep`	Number of replication.
`zigzag`	Experiment layout in zigzag [logic: F]
`nrows`	Experimental design dimension by rows [numeric: value]
`serie`	Number of digits in the plots.
`seed`	Seed for the randomization.
`qrcode`	[string: "{fbname}{plots}{factors}"] String to concatenate the qr code.

Details

Provide the information available.

Value

data frame or list of arguments:

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

`tarpuy_plotdesign` *Fieldbook plot experimental designs*

Description

Plot fieldbook sketch designs based in experimental design

Usage

```
tarpuy_plotdesign(
  data,
  factor = NA,
  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.
`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

```
## Not run:

library(inti)
library(gsheet)

url <- paste0("https://docs.google.com/spreadsheets/d/"
              , "1_BVzChX_-lzXhB7HAm6FeSrwq9iKfZ39_Sl8NFC6k7U/edit#gid=1834109539")
# browseURL(url)

fb <- gsheet2tbl(url)

dsg <- fb %>% tarpuy_design()

dsg

dsg %>% str()

dsg %>%
  tarpuy_plotdesign()


## End(Not run)
```

`tarpuy_traits` *Field book traits*

Description

Function to export field book and traits for be used in field book app.

Usage

```
tarpuy_traits(fieldbook = NULL, last_factor = NULL, traits = NULL)
```

Arguments

`fieldbook`	Experiment field book [dataframe].
`last_factor`	Last factor in the field book [string: colnames]
`traits`	Traits information [dataframe or list].

Details

For the traits parameters you can used shown in the Field Book app

Value

list

Examples

```
library(inti)

fieldbook <- inti::potato

traits <- list(
  list(variable = "altura de planta"
       , trait = "altp"
       , format = "numeric"
       , when = "30, 40, 50"
       , samples = 3
       , units = "cm"
       , details = NA
       , minimum = 0
       , maximum = 100
       )
  , list(variable = "severidad"
         , trait = "svr"
         , format = "scategorical"
         , when = "30, 40, 50"
         , samples = 1
         , units = "scale"
         , details = NA
         , categories = "1, 3, 5, 7, 9"
  )
  ,  list(variable = "foto"
          , trait = "foto"
          , format = "photo"
          , when = "hrv, pshrv"
          , samples = 1
          , units = "image"
          , details = NA
  )
  ,  list(variable = "germinacion"
          , trait = "ger"
          , format = "boolean"
          , when = "30, 40, 50"
          , samples = 1
          , units = "logical"
          , details = NA
  )
)

fbapp <- tarpuy_traits(fieldbook, last_factor = "bloque", traits)

## Not run:

library(inti)
```

```
library(gsheet)

url_fb <- paste0("https://docs.google.com/spreadsheets/d/"
     , "1510fOKj0g4CDEAFkrpFbr-zNMnle_Hou9O_wuf7Vdo4/edit?gid=1607116093#gid=1607116093")

fb <- gsheet2tbl(url_fb)

url_ds <- paste0("https://docs.google.com/spreadsheets/d/"
     , "1510fOKj0g4CDEAFkrpFbr-zNMnle_Hou9O_wuf7Vdo4/edit?gid=1278145622#gid=1278145622")

ds <- gsheet2tbl(url_ds)

fb <- ds %>% tarpuy_design()

url_trt <- paste0("https://docs.google.com/spreadsheets/d/"
     , "1510fOKj0g4CDEAFkrpFbr-zNMnle_Hou9O_wuf7Vdo4/edit?gid=1665653985#gid=1665653985")

traits <- gsheet2tbl(url_trt)

fbapp <- tarpuy_traits(fb, last_factor = "cols", traits)

dsg <- fbapp[[1]]


## End(Not run)
```

`web_table` *HTML tables for markdown documents*

Description

Export tables with download, pasta and copy buttons

Usage

```
web_table(
  data,
  caption = NULL,
  digits = 2,
  rnames = FALSE,
  buttons = NULL,
  file_name = "file",
  scrolly = NULL,
  columnwidth = "200px",
  width = "100%"
)
```

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")
`file_name`	Excel file name
`scrolly`	Windows height to show the table. Default "45vh"
`columnwidth`	Column width. Default '200px'
`width`	Width in pixels or percentage (Defaults to automatic sizing)

Value

table in markdown format for html documents

Examples

```
## Not run:

library(inti)

met %>%
  web_table(caption = "Web table")


## End(Not run)
```

`yupana` *Interactive data analysis*

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()

}
```

`yupana_analysis` *Fieldbook analysis report*

Description

Function to create a complete report of the fieldbook

Usage

```
yupana_analysis(
  data,
  last_factor = NULL,
  response,
  model_factors,
  comparison,
  test_comp = "SNK",
  sig_level = 0.05,
  plot_dist = "boxplot",
  plot_diag = FALSE,
  digits = 2
)
```

Arguments

`data`	Field book data.
`last_factor`	The last factor in your fieldbook.
`response`	Response variable.
`model_factors`	Model used for the experimental design.
`comparison`	Factors to compare
`test_comp`	Comprasison test c("SNK", "TUKEY", "DUNCAN")
`sig_level`	Significal test (default: p = 0.005)
`plot_dist`	Plot data distribution (default = "boxplot")
`plot_diag`	Diagnostic plots for model (default = FALSE).
`digits`	Digits number in the table exported.

Value

list

Examples

```
## Not run:

library(inti)

fb <- potato

rsl <- yupana_analysis(data = fb
                       , last_factor = "bloque"
                       , response = "spad_83"
                       , model_factors = "geno * treat"
                       , comparison = c("geno", "treat")
                       )


## End(Not run)
```

`yupana_export` *Graph options to export*

Description

Function to export the graph options and model parameters

Usage

```
yupana_export(
  data,
  type = NA,
  xlab = NA,
  ylab = NA,
  glab = NA,
  ylimits = NA,
  xrotation = c(0, 0.5, 0.5),
  xtext = NA,
  gtext = NA,
  legend = "top",
  sig = NA,
  error = NA,
  color = TRUE,
  opt = NA,
  dimension = c(20, 10, 100)
)
```

Arguments

`data`	Result from yupana_analysis or yupana_import.
`type`	Plot type
`xlab`	Title for the axis x
`ylab`	Title for the axis y
`glab`	Title for the legend
`ylimits`	limits of the y axis
`xrotation`	Rotation in x axis c(angle, h, v)
`xtext`	Text labels in x axis
`gtext`	Text labels in group
`legend`	the position of legends ("none", "left", "right", "bottom", "top", or two-element numeric vector)
`sig`	Column with the significance
`error`	Show the error bar ("ste" or "std").
`color`	colored figure (TRUE), otherwise black & white (FALSE)
`opt`	Add news layer to the plot
`dimension`	Dimension of graphs

Value

data frame

Examples

```
## Not run:

library(inti)
library(gsheet)

url <- paste0("https://docs.google.com/spreadsheets/d/"
              , "15r7ZwcZZHbEgltlF6gSFvCTFA-CFzVBWwg3mFlRyKPs/edit#gid=172957346")
# browseURL(url)

fb <- gsheet2tbl(url)

smr <- yupana_analysis(data = fb
                       , last_factor = "bloque"
                       , response = "spad_83"
                       , model_factors = "block + geno*riego"
                       , comparison = c("geno", "riego")
                       )

gtab <- yupana_export(smr, type = "line", ylimits = c(0, 100, 2))

#> import
```

```
url <- paste0("https://docs.google.com/spreadsheets/d/"
              , "15r7ZwcZZHbEgltlF6gSFvCTFA-CFzVBWwg3mFlRyKPs/edit#gid=1202800640")
# browseURL(url)

fb <- gsheet2tbl(url)

info <- yupana_import(fb)

etab <- yupana_export(info)

info2 <- yupana_import(etab)

etab2 <- yupana_export(info2)


## End(Not run)
```

`yupana_import` *Import information from data summary*

Description

Graph summary data

Usage

```
yupana_import(data)
```

Arguments

`data`	Summary information with options

Value

list

Examples

```
## Not run:

library(inti)
library(gsheet)

url <- paste0("https://docs.google.com/spreadsheets/d/"
         , "15r7ZwcZZHbEgltlF6gSFvCTFA-CFzVBWwg3mFlRyKPs/edit?gid=2137596914#gid=2137596914")
# browseURL(url)

fb <- gsheet2tbl(url)
```

```
info <- yupana_import(fb)


## End(Not run)
```

`yupana_mvr` *Multivariate Analysis*

Description

Multivariate analysis for PCA and HCPC

Usage

```
yupana_mvr(
  data,
  last_factor = NULL,
  summary_by = NULL,
  groups = NULL,
  variables = NULL
)
```

Arguments

`data`	Field book data.
`last_factor`	The last factor in your fieldbook [string: NULL].
`summary_by`	Variables for group the analysis.
`groups`	Groups for color in PCA.
`variables`	Variables to be use in the analysis [string: NULL].

Details

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

Value

result and plots

Examples

```
## Not run:

library(inti)

fb <- inti::potato
```

```
mv <- yupana_mvr(data = fb
                 , last_factor = "geno"
                 , summary_by = c("geno", "treat")
                 , groups = "treat"
                 , variables = c("all")
                 #, variables = c("wue", "twue")
                 )

mv$plot[1]

mv$data



## End(Not run)
```

`yupana_reshape` *Fieldbook reshape*

Description

Function to reshape fieldbook according a separation character

Usage

```
yupana_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

Index