Package: granular (via r-universe)

July 5, 2026

Title Resolving starch granule distributions

Version 0.4.2

Description Work out distributions.

License GPL-3 | file LICENSE

LazyData true

Depends R (>= 3.0.1)

Imports mixdist, magrittr, shiny (>= 0.14.2), dplyr (>= 0.4.3), tidyr (>= 0.4.0), purrr (>= 0.2.1), ggplot2 (>= 1.0.0), rmarkdown (>= 1.0.0), shinyjs, lazyeval (>= 0.1.10), tibble

Suggests testthat, shinytest, multidplyr

Remotes rstudio/shinytest, hadley/multidplyr

URL https://github.com/csiro-crop-informatics/granular

BugReports https://github.com/csiro-crop-informatics/granular/issues

RoxygenNote 6.0.0

Config/pak/sysreqs cmake make libicu-dev libuv1-dev zlib1g-dev

Repository https://emitanaka.r-universe.dev

Date/Publication 2020-06-04 02:21:42 UTC

RemoteUrl https://github.com/csiro-crop-informatics/granular

RemoteRef HEAD

RemoteSha 9e2f3a22ead4552cd8be3fcef2d1494314320508

Contents

`check_fit` *A function to test how well the fit matches the data*

Description

A function to test how well the fit matches the data

Usage

```
check_fit(fit_output, dist, ps)
```

Arguments

`fit_output`	A data frame output from mix_dist()
`dist`	A numeric vector defining the distribution
`ps`	A numeric vector describing the granule sizes

Value

a ggplot object

`Dist` *An distribution data set*

Description

A dataset containing the distributions from three samples

Usage

```
Dist
```

Format

A data frame with 100 rows and 4 variables:

size Size of granule, in um

118 - Average Distribution of sample

206 - Average Distribution of sample

85 - Average Distribution of sample

get_heights *Get the heights at peaks*

Description

Get the heights at peaks

Usage

```
get_heights(dist, ps, means)
```

Arguments

dist	A numeric vector defining the distribution
ps	A numeric vector describing the granule sizes
means	A named numeric vector defining the means (center) for each peak

Value

A named vector with heights for each mean

ggfit *A function for plotting a fit output against the true distribution*

Description

A function for plotting a fit output against the true distribution

Usage

```
ggfit(fit_output, dist, ps, title = NULL)
```

Arguments

fit_output	A data frame output from mix_dist()
dist	A numeric vector defining the distribution
ps	A numeric vector describing the granule sizes
title	Logical. Should a title be added to the plot?

Value

a ggplot object

ggfit_grp_tbl *Generate fit plots for a grouped tbl*

Description

Generate fit plots for a grouped tbl

Usage

```
ggfit_grp_tbl(.data, fit_output, proportion, size)
```

Arguments

.data	A tbl output from mix_grp_tbl, with list columns
fit_output	Variable name holding mix_dist_output
proportion	Variable name holding distribution as a list
size	Variable name holding particle size as a list

Value

The original tbl with a list column of ggplot output

make_dist *Make a mixture distribution*

Description

Make a mixture distribution

Usage

```
make_dist(fit_output, ps)
```

Arguments

fit_output	A data frame output from mix_dist()
ps	A numeric vector describing the granule sizes

Value

A numeric vector describing the mixture distribution

`mix_dist` *use mix() to estimate underlying distributions*

Description

use mix() to estimate underlying distributions

Usage

```
mix_dist(dist, ps, mu_vec, pi_vec = NULL, sigma_vec = NULL,
  peak_names = NULL, sample_name = NULL, emnum = 10, log_trans = TRUE)
```

Arguments

`dist`	A numeric vector defining the distribution
`ps`	A numeric vector describing the granule sizes
`mu_vec`	A vector defining distribution means (required)
`pi_vec`	A vector defining distribution proportions (optional)
`sigma_vec`	A vector defining distribution dispersion (optional)
`peak_names`	A vector defining peak names (optional)
`sample_name`	A character string defining the sample name (optional)
`emnum`	passed to mix() - A non-negative integer specifying the number of EM steps to be performed
`log_trans`	Logical. Should values be log-transformed?

Value

A list with the fit parameters for each distribution, and complete output from mixdist::mix()

`mix_grp_tbl` *Title*

Description

Title

Usage

```
mix_grp_tbl(.data, proportion, size, mu_vec, pi_vec = NULL,
  sigma_vec = NULL, peak_names = NULL, emnum = 10, log_trans = TRUE,
  parallel = FALSE)
```

Arguments

.data	A tbl grouped by each distribution
proportion	An unquoted variable name
size	An unquoted variable name
mu_vec	A vector defining distribution means (required)
pi_vec	A vector defining distribution proportions (optional)
sigma_vec	A vector defining distribution dispersion (optional)
peak_names	A vector defining peak names (optional)
emnum	passed to mix() - A non-negative integer specifying the number of EM steps to be performed
log_trans	Logical. Should values be log-transformed?
parallel	Logical. Should multidplyr be used to run in parallel? (EXPERIMENTAL)

Value

A mutated tbl with list column output

---

ms1 *An example mix_dist output*

Description

A dataset containing the output from 118 - Average

Usage

```
ms1
```

Format

A data frame with 3 rows and 8 variables:

sample Name of the sample

peak Peak name

pi Distribution proportion

mu Distribution mean

sigma Distribution sd

pi.se proportion standard error

mu.se mean standard error

sigma.se sd standard error

ms2 *An example mix_dist output*

Description

A dataset containing the output from 85 - Average

Usage

```
ms2
```

Format

A data frame with 3 rows and 8 variables:

sample Name of the sample

peak Peak name

pi Distribution proportion

mu Distribution mean

sigma Distribution sd

pi.se proportion standard error

mu.se mean standard error

sigma.se sd standard error

ms3 *An example mix_dist output*

Description

A dataset containing the output from 206 - Average

Usage

```
ms3
```

Format

A data frame with 3 rows and 8 variables:

sample Name of the sample

peak Peak name

pi Distribution proportion

mu Distribution mean

sigma Distribution sd

pi.se proportion standard error

mu.se mean standard error

sigma.se sd standard error

`run_granular` *Run the granular shiny app locally*

Description

Run the granular shiny app locally

Usage

```
run_granular(port = NULL, launch.browser = TRUE,
  host = getOption("shiny.host", "127.0.0.1"))
```

Arguments

`port`	The TCP port that the application should listen on. Defaults to choosing a random port.
`launch.browser`	If true, the system's default web browser will be launched automatically after the app is started. defaults to true in interactive sessions only.
`host`	The IPv4 address that the application should listen on. Defaults to the `shiny.host` option, if set, or `"127.0.0.1"` if not.

Index