Package ‘isopleuros’

May 16, 2023

**Title**  Ternary Plots

**Version**  1.0.0

**Maintainer**  Nicolas Frerebeau <nicolas.frerebeau@u-bordeaux-montaigne.fr>

**Description**  Ternary plots made simple. This package allows to create ternary plots using 'graphics'. It provides functions to display the data in the ternary space, to add or tune graphical elements and to display statistical summaries. It also includes common ternary diagrams which are useful for the archaeologist (e.g. soil texture charts, ceramic phase diagram).

**License**  GPL (>= 3)

**URL**  https://packages.tesselle.org/isopleuros/, https://github.com/tesselle/isopleuros

**BugReports**  https://github.com/tesselle/isopleuros/issues

**Depends**  R (>= 2.10)

**Imports**  graphics, grDevices, methods, utils

**Suggests**  akima, testthat (>= 3.0.0), vdiff (>= 1.0.0)

**Config/testthat/edition**  3

**Encoding**  UTF-8

**LazyData**  true

**RoxygenNote**  7.2.3

**Collate**  'AllGenerics.R' 'coordinates.R' 'data.R'

  'isopleuros-package.R' 'ternary_arrows.R' 'ternary_axes.R'
  'ternary_box.R' 'ternary_contour.R' 'ternary_crosshairs.R'
  'ternary_density.R' 'ternary_ellipse.R' 'ternary_grid.R'
  'ternary_hull.R' 'ternary_lines.R' 'ternary_mean.R'
  'ternary_pairs.R' 'ternary_pca.R' 'ternary_plot.R'
  'ternary_points.R' 'ternary_polygon.R' 'ternary_segments.R'
  'ternary_text.R' 'ternary_title.R' 'triangle_phase.R'
  'triangle_soil.R' 'utilities.R' 'zzz.R'

**NeedsCompilation**  no
Description

Sand, silt, clay compositions of 39 sediment samples at different water depths in an Arctic lake.

Usage

arctic
**boxite**

**Format**

A `data.frame` with 4 variables:

- `sand`
- `silt`
- `clay`
- `depth` Water depth (m).

**Source**


**See Also**

Other datasets: `boxite`, `lava`

---

**boxite**

*Boxite Compositions*

**Description**

Compositions of 25 specimens of boxite.

**Usage**

`boxite`

**Format**

A `data.frame` with 5 variables:

- `A` albite.
- `B` blandite.
- `C` cornite.
- `D` daubite.
- `E` endite.

**Source**


**See Also**

Other datasets: `arctic`, `lava`
Skye Lavas Compositions

Description

AFM compositions of 23 aphyric Skye lavas.

Usage

lava

Format

A data.frame with 3 variables:

A  Na2O + K2O (percent).
F  Fe2O3 (percent).
M  MgO (percent).

Source


See Also

Other datasets: arctic, boxite

Add Arrows to a Ternary Plot

Description

Draw arrows between pairs of points.

Usage

ternary_arrows(x0, y0, z0, ...)

## S4 method for signature 'numeric,numeric,numeric'
ternary_arrows(x0, y0, z0, x1 = x0, y1 = y0, z1 = z0, ...)
ternary_axis

Arguments

x0, y0, z0  A numeric vector giving the x, y and z ternary coordinates of points from which to draw.
...
Further arguments to be passed to graphics::arrows().
x1, y1, z1  A numeric vector giving the x, y and z ternary coordinates of points to which to draw.

Value

ternary_arrows() is called it for its side-effects.

Author(s)

N. Frerebeau

See Also

graphics::arrows()

Other geometries: ternary_crosshairs(), ternary_lines(), ternary_points(), ternary_polygon(), ternary_segments(), ternary_text()

Examples

## Add arrows
ternary_plot(NULL, panel.first = ternary_grid())
ternary_arrows(x0 = 40, y0 = 20, z0 = 40,  
x1 = 20, y1 = 40, z1 = 40)

description

Add an axis to the current plot.

Usage

ternary_axis(  
side,  
at = NULL,  
labels = TRUE,  
tick = TRUE,  
font = NA,  
lty = "solid",  
lwd = 1,  
lwd.ticks = lwd,  
col = NULL,
Arguments

side
An integer specifying which side of the plot the axis is to be drawn on. The axis is placed as follows: 1=below, 2=right and 3=left.

at
A numeric vector giving the points at which tick-marks are to be drawn.

labels
A logical scalar specifying whether (numerical) annotations are to be made at the tickmarks, or a character vector of labels to be placed at the tickpoints. If this is not logical, at should also be supplied and of the same length.

tick
A logical scalar: should tickmarks and an axis line be drawn?

font
font for text. Defaults to par("font.axis").

lty
A character string or numeric value specifying the line type for both the axis line and the tick marks.

lwd, lwd.ticks
A non-negative numeric value specifying the line widths for the axis line and the tick marks.

col, col.ticks
Colors for the axis line and the tick marks respectively. Defaults to par("col.axis").

... Other graphical parameters may also be passed as arguments to this function, particularly, cex.axis, col.axis and font.axis for axis annotation.

Value
ternary_axis() is called it for its side-effects.

Author(s)
N. Frerebeau

See Also
Other graphical elements: ternary_box(), ternary_grid(), ternary_pairs(), ternary_plot(), ternary_title()

Examples

```r
## Add axis
ternary_plot(NULL, axes = FALSE)
ternary_axis(side = 1, col = "red")
ternary_axis(side = 2, col = "blue")
ternary_axis(side = 3, col = "green")

## Add box and grid
ternary_plot(NULL, axes = FALSE)
ternary_box(lty = "dashed", col = "red")
ternary_grid(lty.primary = "dotted")
```
Description

Draw a Box around a Ternary Plot

Usage

ternary_box(lty = "solid", ...)

Arguments

lty A character string or numeric value specifying the line type of the box.

... Other graphical parameters may also be passed as arguments to this function, particularly, col or lwd.

Value

ternary_box() is called it for its side-effects.

Author(s)

N. Frerebeau

See Also

Other graphical elements: ternary_axis(), ternary_grid(), ternary_pairs(), ternary_plot(), ternary_title()

Examples

## Add axis
ternary_plot(NULL, axes = FALSE)
ternary_axis(side = 1, col = "red")
ternary_axis(side = 2, col = "blue")
ternary_axis(side = 3, col = "green")

## Add box and grid
ternary_plot(NULL, axes = FALSE)
ternary_box(lty = "dashed", col = "red")
ternary_grid(lty.primary = "dotted")
ternary_contour  Contour Lines

Description
Computes and draws contour lines.

Usage
ternary_contour(x, y, z, ...)

## S4 method for signature 'numeric,numeric,numeric'
ternary_contour(
  x,
  y,
  z,
  value,
  n = 50,
  nlevels = 10,
  levels = pretty(range(value, na.rm = TRUE), nlevels),
  ilr = TRUE,
  linear = TRUE,
  extrapolate = FALSE,
  palette = function(i) grDevices::hcl.colors(i, "YlOrRd", rev = TRUE),
  ...
)

## S4 method for signature 'ANY,missing,missing'
ternary_contour(
  x,
  value,
  n = 50,
  nlevels = 10,
  levels = pretty(range(value, na.rm = TRUE), nlevels),
  ilr = TRUE,
  linear = TRUE,
  extrapolate = FALSE,
  palette = function(i) grDevices::hcl.colors(i, "YlOrRd", rev = TRUE),
  ...
)

Arguments

x, y, z  A numeric vector giving the x, y and z ternary coordinates of a set of points. If y and z are missing, an attempt is made to interpret x in a suitable way (see grDevices::xyz.coords()).

...  Further arguments to be passed to ternary_lines().
### Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>value</code></td>
<td>A numeric vector giving the values to be plotted.</td>
</tr>
<tr>
<td><code>n</code></td>
<td>A length-one numeric specifying the number of grid points.</td>
</tr>
<tr>
<td><code>nlevels</code></td>
<td>A length-one numeric vector specifying the number of contour levels desired. Only used if <code>levels</code> is NULL.</td>
</tr>
<tr>
<td><code>levels</code></td>
<td>A numeric vector of levels at which to draw contour lines.</td>
</tr>
<tr>
<td><code>ilr</code></td>
<td>A logical scalar: should interpolation be computed in ILR space? If FALSE, interpolation is computed in Cartesian space.</td>
</tr>
<tr>
<td><code>linear</code></td>
<td>A logical scalar: should linear interpolation be used? If FALSE, spline interpolation is used (see <code>akima::interp()</code>).</td>
</tr>
<tr>
<td><code>extrapolate</code></td>
<td>A logical scalar: should extrapolation be used outside of the convex hull determined by the data points (see <code>akima::interp()</code>)?</td>
</tr>
<tr>
<td><code>palette</code></td>
<td>A color palette function that takes a single integer argument (the number of levels) and returns a vector of colors.</td>
</tr>
</tbody>
</table>

### Details

Contour are computed from a bivariate interpolation onto a grid, after an isometric log ratio transformation of the original data.

### Value

ternary_contour() is called it for its side-effects. Invisibly returns a list with elements `levels` (the contour levels) and `colors` (the contour colors) that can be used for a legend.

### Note

The akima package needs to be installed on your machine.

### Author(s)

N. Frerebeau

### See Also

- `akima::interp()`, `grDevices::contourLines()`
- Other statistics: `ternary_density()`, `ternary_ellipse()`, `ternary_hull()`, `ternary_mean()`, `ternary_pca()`

### Examples

```r
## Add density
## Data from Aitchison 1986
ternary_plot(arctic, panel.first = ternary_grid())
levels <- ternary_contour(arctic, value = arctic$depth, n = 100, nlevels = 10)
## Add a legend
legend_image <- grDevices::as.raster(rev(levels$colors))
```
ternary_crosshairs

Add Cross-Hairs to a Ternary Plot

Description

Draw lines that intersect at a point.

Usage

ternary_crosshairs(x, y, z, ...)

## S4 method for signature 'numeric,numeric,numeric'
ternary_crosshairs(x, y, z, x_mark = TRUE, y_mark = TRUE, z_mark = TRUE, ...)

## S4 method for signature 'ANY,missing,missing'
ternary_crosshairs(x, x_mark = TRUE, y_mark = TRUE, z_mark = TRUE, ...)

Arguments

x, y, z  
A numeric vector giving the x, y and z ternary coordinates of a set of points. If y and z are missing, an attempt is made to interpret x in a suitable way (see grDevices::xyz.coords()).

...  
Further graphical parameters (see graphics::par()) may also be supplied as arguments, particularly, line type, lty, line width, lwd and color, col. Also the line characteristics lend, ljoin and lmitre.

x_mark, y_mark, z_mark  
A logical scalar: should the x, y or z axis component be drawn?

Value

ternary_crosshairs() is called it for its side-effects.

Author(s)

N. Frerebeau

See Also

Other geometries: ternary_arrows(), ternary_lines(), ternary_points(), ternary_polygon(), ternary_segments(), ternary_text()
ternary_density

Examples

```r
## Add cross-hairs
## Data from Aitchison 1986
ternary_plot(lava, panel.first = ternary_grid())
ternary_crosshairs(lava)

ternary_plot(lava, panel.first = ternary_grid())
ternary_crosshairs(lava, y_mark = FALSE, z_mark = FALSE, col = "red")

ternary_plot(lava, panel.first = ternary_grid())
ternary_crosshairs(lava, x_mark = FALSE, z_mark = FALSE, col = "green")

ternary_plot(lava, panel.first = ternary_grid())
ternary_crosshairs(lava, x_mark = FALSE, y_mark = FALSE, col = "blue")
```

density

Density Contour Lines

Description

Computes and draws density contour lines.

Usage

```r
ternary_density(x, y, z, ...)
```

## S4 method for signature 'numeric,numeric,numeric'
```r
ternary_density(
  x,
  y,
  z,
  h = NULL,
  n = 25,
  nlevels = 10,
  levels = NULL,
  palette = function(i) grDevices::hcl.colors(i, "YlOrRd", rev = TRUE),
  ...
)
```

## S4 method for signature 'ANY,missing,missing'
```r
ternary_density(
  x,
  h = NULL,
  n = 25,
  nlevels = 10,
  levels = NULL,
  palette = function(i) grDevices::hcl.colors(i, "YlOrRd", rev = TRUE),
  ...
)
```
ternary_density

Arguments

x, y, z  A numeric vector giving the x, y and z ternary coordinates of a set of points. If y and z are missing, an attempt is made to interpret x in a suitable way (see grDevices::xyz.coords()).

...  Further arguments to be passed to ternary_lines().

h  A length-one numeric vector giving the bandwidth.

n  A length-one numeric specifying the number of grid points.

nlevels  A length-one numeric vector specifying the number of contour levels desired. Only used if levels is NULL.

levels  A numeric vector of levels at which to draw contour lines.

palette  A color palette function that takes a single integer argument (the number of levels) and returns a vector of colors.

Details

Two-dimensional kernel density estimation with an axis-aligned bivariate normal kernel. Normal kernel is evaluated on a square grid, after an isometric log ratio transformation of the original data.

Value

ternary_density() is called it for its side-effects.

Invisibly returns a list with elements levels (the contour levels) and colors (the contour colors) that can be used for a legend.

Note

Two-dimensional kernel density estimation is adapted from MASS::kde2d().

This must be considered as experimental and subject to major changes in a future release.

Author(s)

N. Frerebeau

See Also

grDevices::contourLines()

Other statistics: ternary_contour(), ternary_ellipse(), ternary_hull(), ternary_mean(), ternary_pca()

Examples

## Add density
## Data from Aitchison 1986
ternary_plot(lava, panel.first = ternary_grid())
levels <- ternary_density(lava, n = 500, nlevels = 10)

## Add a legend
ternary_ellipse

Add an Ellipse to a Ternary Plot

Description

Computes and draws a confidence/tolerance ellipse.

Usage

ternary_ellipse(x, y, z, ...)
ternary_confidence(x, y, z, ...)
ternary_tolerance(x, y, z, ...)

## S4 method for signature 'numeric,numeric,numeric'
ternary_ellipse(x, y, z, radius = 1, ...)

## S4 method for signature 'ANY,missing,missing'
ternary_ellipse(x, radius = 1, ...)

## S4 method for signature 'numeric,numeric,numeric'
ternary_confidence(x, y, z, level = 0.95, ...)

## S4 method for signature 'ANY,missing,missing'
ternary_confidence(x, level = 0.95, ...)

## S4 method for signature 'numeric,numeric,numeric'
ternary_tolerance(x, y, z, level = 0.95, ...)

## S4 method for signature 'ANY,missing,missing'
ternary_tolerance(x, level = 0.95, ...)

Arguments

x, y, z  
A numeric vector giving the x, y and z ternary coordinates of a set of points. If y and z are missing, an attempt is made to interpret x in a suitable way (see grDevices::xyz.coords()).

...  
Further arguments to be passed to graphics::polygon().

radius  
A numeric vector specifying the scaling of the half-diameters.

level  
A numeric vector specifying the confidence/tolerance level.
Details

Ellipse coordinates are computed after an isometric log ratio transformation of the original data.

Value

ternary_ellipse() is called it for its side-effects.

Author(s)

N. Frerebeau

See Also

graphics::polygon()

Other statistics: ternary_contour(), ternary_density(), ternary_hull(), ternary_mean(), ternary_pca()

Examples

## Ellipses
## Data from Aitchison 1986
ternary_plot(lava, panel.first = ternary_grid(5, 10))
ternary_tolerance(lava, level = 0.95, border = "blue", lty = 2)
ternary_confidence(lava, level = 0.95, border = "red", lty = 3)

ternary_grid

Add Grid to a Ternary Plot

Description

Adds a triangular grid to an existing plot.

Usage

ternary_grid(
  primary = NULL,
  secondary = NULL,
  col.primary = "darkgray",
  col.secondary = "lightgray",
  lty.primary = "dashed",
  lty.secondary = "dotted",
  lwd.primary = 1,
  lwd.secondary = lwd.primary
)
ternary_grid

Arguments

primary      An integer specifying the number of cells of the primary grid in x, y and z direction.
secondary    An integer specifying the number of cells of the secondary grid in x, y and z direction.
col.primary, col.secondary
             A character string specifying the color of the grid lines.
lty.primary, lty.secondary
             A character string or numeric value specifying the line type of the grid lines.
lwd.primary, lwd.secondary
             A non-negative numeric value specifying the line width of the grid lines.

Value

ternary_grid() is called it for its side-effects.

Author(s)

N. Frerebeau

See Also

Other graphical elements: ternary_axis(), ternary_box(), ternary_pairs(), ternary_plot(), ternary_title()

Examples

## Blank plot
ternary_plot(NULL)

## Compositional data
coda <- data.frame(
    X = c(20, 60, 20, 1/3),
    Y = c(20, 20, 60, 1/3),
    Z = c(60, 20, 20, 1/3)
)

## Ternary plot
ternary_plot(coda, pch = 16, col = "red")

## Add a grid
ternary_plot(coda, panel.first = ternary_grid(5, 10))

## Zoom
ternary_plot(coda, xlim = c(0.5, 1), panel.first = ternary_grid())
ternary_plot(coda, ylim = c(0.5, 1), panel.first = ternary_grid())
ternary_plot(coda, zlim = c(0.5, 1), panel.first = ternary_grid())

## Color according to a supplementary variable
## Data from Aitchison 1986
ternary_hull

Convex Hull of a Set of Points

Description

Computes and draws the convex hull of the set of points specified.

Usage

ternary_hull(x, y, z, ...)

## S4 method for signature 'numeric,numeric,numeric'
ternary_hull(x, y, z, ...)

## S4 method for signature 'ANY,missing,missing'
ternary_hull(x, y, z, ...)

Arguments

x, y, z

A numeric vector giving the x, y and z ternary coordinates of a set of points. If y and z are missing, an attempt is made to interpret x in a suitable way (see grDevices::xyz.coords()).

... Further arguments to be passed to graphics::polygon().

Value

ternary_hull() is called it for its side-effects.

Author(s)

N. Frerebeau

See Also

grDevices::chull(), graphics::polygon()

Other statistics: ternary_contour(), ternary_density(), ternary_ellipse(), ternary_mean(), ternary_pca()

Examples

## Convex hull
## Data from Aitchison 1986
ternary_plot(lava, panel.first = ternary_grid(5, 10))
ternary_hull(lava, border = "red"
Add Connected Line Segments to a Ternary Plot

Description

Add Connected Line Segments to a Ternary Plot

Usage

ternary_lines(x, y, z, ...)  

## S4 method for signature 'numeric,numeric,numeric'
ternary_lines(x, y, z, type = "l", ...)

## S4 method for signature 'ANY,missing,missing'
ternary_lines(x, type = "l", ...)

Arguments

x, y, z  
A numeric vector giving the x, y and z ternary coordinates of a set of points. If y and z are missing, an attempt is made to interpret x in a suitable way (see grDevices::xyz.coords()).

...  
Further graphical parameters (see graphics::par()) may also be supplied as arguments, particularly, line type, lty, line width, lwd, color, col and for type = "b", pch. Also the line characteristics lend, ljoin and lmitre.

type  
A character string indicating the type of plotting; actually any of the types as in graphics::plot.default().

Value

ternary_lines() is called it for its side-effects.

Author(s)

N. Frerebeau

See Also

graphics::lines()

Other geometries: ternary_arrows(), ternary_crosshairs(), ternary_points(), ternary_polygon(), ternary_segments(), ternary_text()
Examples

```r
## Compositional data
coda <- data.frame(
  X = c(20, 60, 20, 20),
  Y = c(20, 20, 60, 40),
  Z = c(60, 20, 20, 40)
)

## Add lines
ternary_plot(NULL, panel.first = ternary_grid())
ternary_lines(coda, col = "red", lwd = 2)
```

---

### ternary_mean

#### Compositional Mean

**Description**

Computes and draws the closed geometric mean of the set of points specified.

**Usage**

```r
ternary_mean(x, y, z, ...)
```

**Arguments**

- `x, y, z` are numeric vectors giving the x, y and z ternary coordinates of a set of points. If y and z are missing, an attempt is made to interpret x in a suitable way (see `grDevices::xyz.coords()`).
- `...` are further arguments to be passed to `graphics::points()`.

**Value**

`ternary_mean()` is called it for its side-effects.

**Author(s)**

N. Frerebeau

**See Also**

Other statistics: `ternary_contour()`, `ternary_density()`, `ternary_ellipse()`, `ternary_hull()`, `ternary_pca()`
### Examples

```r
## Mean
## Data from Aitchison 1986
ternary_plot(lava, panel.first = ternary_grid())
ternary_mean(lava, pch = 16, col = "red")
ternary_confidence(lava, level = 0.95, border = "red", lty = 1)
```

---

### Description

Produces a matrix of ternary plots.

### Usage

```r
ternary_pairs(x, ...)
```

## S4 method for signature 'matrix'
```r
ternary_pairs(x, margin = NULL, ...)
```

## S4 method for signature 'data.frame'
```r
ternary_pairs(x, margin = NULL, ...)
```

### Arguments

- `x`: A matrix or a data.frame. Columns are converted to numeric in the same way that `data.matrix()` does.
- `...`: Further arguments to be passed to `graphics::arrows()`.
- `margin`: A character string or an integer giving the index of the column to be used as the third part of the ternary plots. If NULL (the default), marginal compositions will be used (i.e. the geometric mean of the non-selected parts).

### Value

`ternary_pairs()` is called it for its side-effects.

### Author(s)

N. Frerebeau

### See Also

- `graphics::arrows()`

Other graphical elements: `ternary_axis()`, `ternary_box()`, `ternary_grid()`, `ternary_plot()`, `ternary_title()`
## ternary_pca

### Description

Computes and draws principal component.

### Usage

```r
ternary_pca(x, y, z, ...)  
## S4 method for signature 'numeric,numeric,numeric'
ternary_pca(x, y, z, axis = 1, ...)  
## S4 method for signature 'ANY,missing,missing'
ternary_pca(x, axis = 1, ...)  
```

### Arguments

- `x, y, z` A numeric vector giving the x, y and z ternary coordinates of a set of points. If y and z are missing, an attempt is made to interpret x in a suitable way (see `grDevices::xyz.coords()`).
- `...` Further arguments to be passed to `graphics::lines()`.
- `axis` An integer specifying the dimension to be plotted.

### Value

`ternary_pca()` is called it for its side-effects.

### Author(s)

N. Frerebeau

### See Also

Other statistics: `ternary_contour()`, `ternary_density()`, `ternary_ellipse()`, `ternary_hull()`, `ternary_mean()`
**Examples**

```r
## PCA
## Data from Aitchison 1986
ternary_plot(lava, panel.first = ternary_grid())
ternary_pca(lava, axis = 1, col = "red", lty = 2)
```

---

**ternary_plot**  
*Ternary Plot*

---

**Description**

Produces a ternary plot.

**Usage**

```r
ternary_plot(x, y, z, ...)
```

```r
## S4 method for signature 'numeric,numeric,numeric'
ternary_plot(
  x,
  y,
  z,
  xlim = NULL,
  ylim = NULL,
  zlim = NULL,
  xlab = NULL,
  ylab = NULL,
  zlab = NULL,
  main = NULL,
  sub = NULL,
  ann = graphics::par("ann"),
  axes = TRUE,
  frame.plot = axes,
  panel.first = NULL,
  panel.last = NULL,
  ...
)
```

```r
## S4 method for signature 'ANY,missing,missing'
ternary_plot(
  x,
  xlim = NULL,
  ylim = NULL,
  zlim = NULL,
  xlab = NULL,
  ylab = NULL,
  zlab = NULL,
```
ternary_plot()

Arguments

x, y, z
A numeric vector giving the x, y and z ternary coordinates of a set of points. If y and z are missing, an attempt is made to interpret x in a suitable way (see grDevices::xyz.coords()).

... Other graphical parameters may also be passed as arguments to this function.

xlim A length-two numeric vector giving the x limits in the range [0, 1].

ylim A length-two numeric vector giving the y limits in the range [0, 1].

zlim A length-two numeric vector giving the z limits in the range [0, 1].

xlab, ylab, zlab A character string giving a label for the x, y and z axes.

main A character string giving a main title for the plot.

sub A character string giving a subtitle for the plot.

ann A logical scalar: should the default annotation (title and x, y and z axis labels) appear on the plot?

axes A logical scalar: should axes be drawn on the plot?

frame.plot A logical scalar: should a box be drawn around the plot?

panel.first An an expression to be evaluated after the plot axes are set up but before any plotting takes place. This can be useful for drawing background grids.

panel.last An expression to be evaluated after plotting has taken place but before the axes, title and box are added.

Value
ternary_plot() is called it for its side-effects: it results in a graphic being displayed.

Author(s)
N. Frerebeau

See Also
Other graphical elements: ternary_axis(), ternary_box(), ternary_grid(), ternary_pairs(), ternary_title()
ternary_points

Examples

```r
## Blank plot
ternary_plot(NULL)

## Compositional data
coda <- data.frame(
  X = c(20, 60, 20, 1/3),
  Y = c(20, 20, 60, 1/3),
  Z = c(60, 20, 20, 1/3)
)

## Ternary plot
ternary_plot(coda, pch = 16, col = "red")

## Add a grid
ternary_plot(coda, panel.first = ternary_grid(5, 10))

## Zoom
ternary_plot(coda, xlim = c(0.5, 1), panel.first = ternary_grid())
ternary_plot(coda, ylim = c(0.5, 1), panel.first = ternary_grid())
ternary_plot(coda, zlim = c(0.5, 1), panel.first = ternary_grid())

## Color according to a supplementary variable
## Data from Aitchison 1986
col <- grDevices::colorRampPalette(c("red", "blue"))(nrow(arctic))
ternary_plot(arctic, panel.first = ternary_grid(), pch = 16, col = col)
```

### ternary_points

Add Points to a Ternary Plot

#### Description

Add Points to a Ternary Plot

#### Usage

```r
ternary_points(x, y, z, ...)
```

#### Arguments

- `x`, `y`, `z`:
  
  A `numeric` vector giving the x, y and z ternary coordinates of a set of points.
  If `y` and `z` are missing, an attempt is made to interpret `x` in a suitable way (see `grDevices::xyz.coords()`).
Further graphical parameters (see `graphics::par()`) may also be supplied as arguments, particularly, plotting character, `pch`, character expansion, `cex` and color, `col`.

**type**
A character string indicating the type of plotting; actually any of the types as in `graphics::plot.default()`.

**Value**
ternary_points() is called it for its side-effects.

**Author(s)**
N. Frerebeau

**See Also**

`graphics::points()`

Other geometries: `ternary_arrows()`, `ternary_crosshairs()`, `ternary_lines()`, `ternary_polygon()`, `ternary_segments()`, `ternary_text()`

**Examples**

```r
## Add points
## Data from Aitchison 1986
ternary_plot(NULL, panel.first = ternary_grid())
ternary_points(lava, col = "red", pch = 16)
```

---

**ternary_polygon** **Polygon Drawing**

**Description**

Draws the polygons whose vertices are given in x, y and z.

**Usage**

```r
ternary_polygon(x, y, z, ...)
```

```r
## S4 method for signature 'numeric,numeric,numeric'
ternary_polygon(x, y, z, ...)
```

```r
## S4 method for signature 'ANY,missing,missing'
ternary_polygon(x, y, z, ...)
```
ternary_segments

Arguments

- **x**, **y**, **z**  
  A numeric vector giving the x, y and z ternary coordinates of a set of points. If y and z are missing, an attempt is made to interpret x in a suitable way (see \texttt{grDevices::xyz.coords()}).

  ... Further arguments to be passed to \texttt{graphics::polygon()}.

Value

- \texttt{ternary_polygon()} is called it for its side-effects.

Author(s)

- N. Frerebeau

See Also

- \texttt{graphics::polygon()}
- Other geometries: \texttt{ternary_arrows()}, \texttt{ternary_crosshairs()}, \texttt{ternary_lines()}, \texttt{ternary_points()}, \texttt{ternary_segments()}, \texttt{ternary_text()}

Examples

```r
## Compositional data
coda <- data.frame(
  X = c(20, 60, 20),
  Y = c(20, 20, 60),
  Z = c(60, 20, 20)
)

## Add a polygon
ternary_plot(NULL, panel.first = ternary_grid())
ternary_polygon(coda, density = 5, border = "red")
```

Description

- Draw line segments between pairs of points.

Usage

- \texttt{ternary_segments(x0, y0, z0, ...)}

  ## S4 method for signature 'numeric,numeric,numeric'
  \texttt{ternary_segments(x0, y0, z0, x1 = x0, y1 = y0, z1 = z0, ...)}
Arguments

- \(x_0, y_0, z_0\)
  
  A numeric vector giving the x, y and z ternary coordinates of points from which to draw.

- \(x_1, y_1, z_1\)
  
  A numeric vector giving the x, y and z ternary coordinates of points to which to draw.

Value

\(\text{ternary\_segments()}\) is called it for its side-effects.

Author(s)

N. Frerebeau

See Also

\(\text{graphics\::\:segments()}\)

Other geometries: \(\text{ternary\_arrows()}, \text{ternary\_crosshairs()}, \text{ternary\_lines()}, \text{ternary\_points()}, \text{ternary\_polygon()}, \text{ternary\_text()}\)

Examples

```r
## Add segments
ternary_plot(NULL, panel.first = ternary_grid())
ternary_segments(x0 = 40, y0 = 20, z0 = 40,
                 x1 = 20, y1 = 40, z1 = 40)
```

---

**ternary\_text**

### Add Text to a Ternary Plot

Description

Draws the strings given in the vector labels at the coordinates given by x, y and z.

Usage

\[
\text{ternary\_text}(x, y, z, \ldots)
\]

\[
\text{## S4 method for signature 'numeric,numeric,numeric'}
\]

\[
\text{ternary\_text}(x, y, z, \text{labels = seq\_along}(x), \ldots)
\]

\[
\text{## S4 method for signature 'ANY,missing,missing'}
\]

\[
\text{ternary\_text}(x, \text{labels = seq\_along}(x$x), \ldots)
\]
Arguments

\(x, y, z\)  
\(x\), \(y\), and \(z\) are numeric vectors giving the ternary coordinates of a set of points. If \(y\) and \(z\) are missing, an attempt is made to interpret \(x\) in a suitable way (see \texttt{grDevices::xyz.coords()}).

...  
Further arguments to be passed to \texttt{graphics::text()}.

labels  
A character vector or expression specifying the text to be written.

Value

ternary_text() is called it for its side-effects.

Author(s)

N. Frerebeau

See Also

\texttt{graphics::text()}

Other geometries: \texttt{ternary_arrows()}, \texttt{ternary_crosshairs()}, \texttt{ternary_lines()}, \texttt{ternary_points()}, \texttt{ternary_polygon()}, \texttt{ternary_segments()}

Examples

## Compositional data
coda <- data.frame(  
  X = c(20, 60, 20),  
  Y = c(20, 20, 60),  
  Z = c(60, 20, 20)
)

## Add text
ternary_plot(NULL, panel.first = ternary_grid())
ternary_text(coda, labels = c("A", "B", "C"), col = "red", cex = 2)
ternary_title

Usage

ternary_title(
  main = NULL,
  sub = NULL,
  xlab = NULL,
  ylab = NULL,
  zlab = NULL,
  line = NA,
  outer = FALSE,
  ...
)

Arguments

main       A character string specifying the main title (on top).
sub        A character string specifying the sub-title (at bottom).
xlab, ylab, zlab
           A character string giving a label for the x, y and z axes.
line       Specifying a value for line overrides the default placement of labels, and places
           them this many lines outwards from the plot edge.
onner      A logical scalar: should the titles be placed in the outer margins of the plot?
...        Other graphical parameters may also be passed as arguments to this function,
           particularly, font.main, cex.main, col.main and font.sub, cex.sub, col.sub
           for title annotation; font.lab, cex.lab and col.lab for axis label.

Value

ternary_title() is called it for its side-effects.

Author(s)

N. Frerebeau

See Also

Other graphical elements: ternary_axis(), ternary_box(), ternary_grid(), ternary_pairs(),
ternary_plot()

Examples

## Add title
ternary_plot(NULL, main = "Main title", sub = "Subtitle",
             xlab = "A", ylab = "B", zlab = "C")

ternary_plot(NULL, ann = FALSE)
ternary_title(main = "Main title", sub = "Subtitle",
              xlab = "A", ylab = "B", zlab = "C")
Description

Ceramic Phase Diagram

Usage

```r
triangle_phase_cas(labels = TRUE, symbol = FALSE, mol = FALSE, ...)
triangle_phase_ceramic(labels = TRUE, symbol = FALSE, mol = FALSE, ...)
```

Arguments

- **labels**: A logical scalar: should labels be displayed?
- **symbol**: A logical scalar: should symbol be used instead of full labels? Only used if labels is TRUE.
- **mol**: A logical scalar: should molarity be used instead of molar mass?
- **...**: Further arguments to be passed to `graphics::polygon()`.

Author(s)

N. Frerebeau

See Also

Other charts: `triangle_soil`

Examples

```r
## Ceramic phase diagram
ternary_plot(NULL, xlab = "CaO", ylab = "Al2O3", zlab = "SiO2")
triangle_phase_ceramic(symbol = TRUE, mol = TRUE, pch = 16)

ternary_plot(NULL, axes = FALSE, ann = FALSE, frame.plot = TRUE)
triangle_phase_ceramic(mol = FALSE, pch = 16)
```

## CAS diagram
```r
ternary_plot(NULL, axes = FALSE, ann = FALSE, frame.plot = TRUE)
triangle_phase_cas(mol = FALSE, pch = 16)
```
triangle Soil Texture Triangle

Description
Soil Texture Triangle

Usage
triangle_soil_hypres(labels = TRUE, symbol = FALSE, ...)
triangle_soil_folk(labels = TRUE, symbol = FALSE, ...)
triangle_soil_shepard(labels = TRUE, symbol = FALSE, ...)
triangle_soil_usda(labels = TRUE, symbol = FALSE, ...)

Arguments
labels A logical scalar: should labels be displayed?
symbol A logical scalar: should symbol be used instead of full labels? Only used if labels is TRUE.
... Further arguments to be passed to graphics::polygon().

Author(s)
N. Frerebeau

See Also
Other charts: triangle_phase_cas()

Examples
## HYPRES soil texture
ternary_plot(NULL, xlab = "sand", ylab = "silt", zlab = "clay")
triangle_soil_hypres()

## USDA (1951) soil texture
ternary_plot(NULL, xlab = "sand", ylab = "silt", zlab = "clay")
triangle_soil_usda(symbol = TRUE)

## Folk (1954) soil texture
ternary_plot(NULL, xlab = "sand", ylab = "silt", zlab = "clay")
triangle_soil_folk(symbol = TRUE)

## Shepard (1954) soil texture
ternary_plot(NULL, xlab = "sand", ylab = "silt", zlab = "clay")
triangle_soil_shepard()
Index

* charts
  triangle_phase_cas, 29
  triangle_soil, 30
* datasets
  arctic, 2
  boxite, 3
  lava, 4
* geometries
  ternary_arrows, 4
  ternary_crosshairs, 10
  ternary_lines, 17
  ternary_points, 23
  ternary_polygon, 24
  ternary_segments, 25
  ternary_text, 26
* graphical elements
  ternary_axis, 5
  ternary_box, 7
  ternary_grid, 14
  ternary_pairs, 19
  ternary_plot, 21
  ternary_title, 27
* statistics
  ternary_contour, 8
  ternary_density, 11
  ternary_ellipse, 13
  ternary_hull, 16
  ternary_mean, 18
  ternary_pca, 20
akima::interp(), 9
arctic, 2, 3, 4
boxite, 3, 4
character, 6, 7, 15, 17, 19, 22, 24, 27, 28
data.frame, 3, 4, 19
data.matrix(), 19
expression, 27
function, 9, 12

graphical parameters, 6, 7, 22, 28
graphics::arrows(), 5, 19
graphics::lines(), 17, 20
graphics::par(), 10, 17, 24, 26
graphics::plot.default(), 17, 24
graphics::points(), 18, 24
graphics::polygon(), 13, 14, 16, 25, 29, 30
graphics::segments(), 26
graphics::text(), 27
grDevices::chull(), 16
grDevices::contourLines(), 9, 12
grDevices::xyz.coords(), 8, 10, 12, 13,
  16–18, 20, 22, 23, 25, 27
integer, 6, 15, 19, 20
lava, 3, 4
list, 9, 12
logical, 6, 9, 10, 22, 28–30
MASS::kde2d(), 12
matrix, 19
numeric, 5–10, 12, 13, 15–18, 20, 22, 23,
  25–27
ternary_arrows, 4, 10, 17, 24–27
ternary_arrows,numeric,numeric,numeric-method
  (ternary_arrows), 4
ternary_arrows-method(ternary_arrows),
  4
ternary_axis, 5, 7, 15, 19, 22, 28
ternary_box, 6, 7, 15, 19, 22, 28
ternary_confidence (ternary_ellipse), 13
ternary_confidence,ANY,missing,missing-method
  (ternary_ellipse), 13
ternary_confidence,numeric,numeric,numeric-method
  (ternary_ellipse), 13
ternary_contour, 8, 12, 14, 16, 18, 20
ternary_contour, ANY, missing, missing-method
(ternary_contour), 8
ternary_contour, numeric, numeric, numeric-method
(ternary_contour), 8
ternary_contour-method
(ternary_contour), 8
ternary_crosshairs, 5, 10, 17, 24–27
ternary_contour-method (ternary_contour), 8
ternary_crosshairs, ANY, missing, missing-method
(ternary_crosshairs), 10
ternary_crosshairs-method
(ternary_crosshairs), 10
ternary_density, ANY, missing, missing-method
(ternary_density), 11
ternary_density, numeric, numeric, numeric-method
(ternary_density), 11
ternary_density-method
(ternary_density), 11
ternary_ellipse, 9, 12, 13, 16, 18, 20
ternary_ellipse-method
(ternary_ellipse), 13
ternary_ellipse, numeric, numeric, numeric-method
(ternary_ellipse), 13
ternary_ellipse-method
(ternary_ellipse), 13
ternary_grid
ternary_hull, 9, 12, 14, 16, 18, 20
ternary_hull-method
(ternary_hull), 16
ternary_hull, numeric, numeric, numeric-method
(ternary_hull), 16
ternary_hull-method (ternary_hull), 16
ternary_hull-method
(ternary_hull), 16
ternary_lines, 5, 10, 17, 24–27
ternary_lines-method
(ternary_lines), 17
ternary_lines, numeric, numeric, numeric-method
(ternary_lines), 17
ternary_lines-method (ternary_lines), 17
ternary_mean, 9, 12, 14, 16, 18, 20
ternary_mean-method
(ternary_mean), 18
ternary_mean, numeric, numeric, numeric-method
(ternary_mean), 18
ternary_mean-method
(ternary_mean), 18
ternary_pairs, 6, 7, 13, 19, 22, 28
ternary_pairs-method
(ternary_pairs), 19
ternary_pairs, data.frame, missing, missing-method
(ternary_pairs), 19
ternary_pairs-method (ternary_pairs), 19
ternary_pairs, matrix.frame-method
(ternary_pairs), 19
ternary_pairs-method (ternary_pairs), 19
ternary_plot
ternary_pca, 9, 12, 14, 16, 18, 20
ternary_pca-method
(ternary_pca), 20
ternary_pca, numeric, numeric, numeric-method
(ternary_pca), 20
ternary_pca-method (ternary_pca), 20
ternary_plot, ANY, missing, missing-method
(ternary_plot), 21
ternary_plot-method
(ternary_plot), 21
ternary_points, 5, 10, 17, 23, 25–27
ternary_points-method
(ternary_points), 23
ternary_points, numeric, numeric, numeric-method
(ternary_points), 23
ternary_points-method
(ternary_points), 23
ternary_points-method (ternary_points), 23
ternary_polygon, 5, 10, 17, 24, 26, 27
ternary_polygon-method
(ternary_polygon), 24
ternary_polygon, numeric, numeric, numeric-method
(ternary_polygon), 24
ternary_polygon-method (ternary_polygon), 24
ternary_segments, 5, 10, 17, 24, 25, 25, 27
ternary_segments-method
(ternary_segments), 25
ternary_segments-method (ternary_segments), 25
ternary_segments
ternary_text, 5, 10, 17, 24–26, 26
ternary_text-method
(ternary_text), 26
ternary_text, ANY, missing, missing-method
(ternary_text), 26
ternary_text-method (ternary_text), 26
ternary_title, 6, 7, 15, 19, 22, 27
ternary_tolerance (ternary_ellipse), 13
ternary_tolerance, ANY, missing, missing-method
(ternary_ellipse), 13
ternary_tolerance, numeric, numeric, numeric-method
  (ternary_ellipse), 13
triangle_phase_cas, 29, 30
triangle_phase_ceramic
  (triangle_phase_cas), 29
triangle_soil, 29, 30
triangle_soil_folk (triangle_soil), 30
triangle_soil_hypres (triangle_soil), 30
triangle_soil_shepard (triangle_soil), 30
triangle_soil_usda (triangle_soil), 30