Package: VoronoiBiomedPlot (via r-universe)

May 25, 2026

Type Package

Title Tesselation Visualization Plots for 2D Data

Version 0.3.2

Description Creates visualization plots for 2D data including ellipse plots, Voronoi tesselation plots, and combined ellipse-Voronoi plots. Designed to visualize class separation in 2D data, raw of from projection techniques like principal component analysis (PCA), partial least squares discriminant analysis (PLS-DA) or others. For more details see Lotsch and Kringel (2026) and Lotsch, J., and Kringel, D. (2026) <doi:10.1371/journal.pone.0333653>.

License GPL (>= 3)

Encoding UTF-8

URL `https://github.com/JornLotsch/voronoi_tesselation_plot`

BugReports `https://github.com/JornLotsch/voronoi_tesselation_plot/issues`

Depends R (>= 3.5.0)

Imports ggplot2, ggrepel, deldir, MASS

NeedsCompilation no

Author Jorn Lotsch [aut, cre] (ORCID: <`https://orcid.org/0000-0002-5818-6958`>)

Maintainer Jorn Lotsch <`j.lotsch@em.uni-frankfurt.de`>

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

Date/Publication 2026-04-25 07:48:26 UTC

RemoteUrl https://github.com/cran/VoronoiBiomedPlot

RemoteRef HEAD

RemoteSha ca2da7340ca9ba213ef50b5843c8f0ca98825be7

Contents

VoronoiBiomedPlot-package 2
create_tesselation_plots 3
create_voronoi_plot 7

Index **10**

`VoronoiBiomedPlot-package`

Tesselation Visualization Plots for 2D Data

Description

Creates visualization plots for 2D data including ellipse plots, Voronoi tesselation plots, and combined ellipse-Voronoi plots. Designed to visualize class separation in 2D data, raw of from projection techniques like principal component analysis (PCA), partial least squares discriminant analysis (PLS-DA) or others. For more details see Lotsch and Kringel (2026) and Lotsch, J., and Kringel, D. (2026) <doi:10.1371/journal.pone.0333653>.

Details

The VoronoiBiomedPlot package provides functions for creating visualization plots of 2D data, particularly useful for biomedical data analysis and dimensionality reduction results. The package includes two main functions:

- `create_tesselation_plots`: Creates three types of plots (ellipse, Voronoi, and combined)
- `create_voronoi_plot`: Creates standalone Voronoi tessellation plots

These functions are designed to visualize class separation in two-dimensional raw or projected data, such as dimensionally reduced data from techniques like PCA, PLS-DA, UMAP, or other projection methods, commonly used in biomedical research. Voronoi tessellation divides the plot space into regions based on proximity to data points, providing an intuitive visualization of class boundaries and decision regions. Confidence ellipses show the distribution spread and correlation structure within each class. Both functions support an optional Voronoi island count, a visualization-intrinsic metric that quantifies class structure disruption by identifying data points whose Voronoi cells are entirely surrounded by cells of a different class. This metric has no equivalent in confidence ellipse visualizations and can be displayed as a plot subtitle via the `show_island_count` parameter.

Author(s)

Jorn Lotsch <j.lotsch@em.uni-frankfurt.de>

References

Lötsch, J. and A. Ultsch (2024). Comparative assessment of projection and clustering method combinations in the analysis of biomedical data. Informatics in Medicine Unlocked 50: 101573. https://www.sciencedirect.com/science/article/pii/S2352914824001291

See Also

- `https://github.com/JornLotsch/voronoi_tesselation_plot`
- Report bugs at `https://github.com/JornLotsch/voronoi_tesselation_plot/issues`

Examples

```
# Load the iris dataset
data <- iris[, c("Sepal.Length", "Petal.Length", "Species")]

# Create comprehensive tesselation plots
plots <- create_tesselation_plots(
  data = data,
  class_column = "Species",
  legend_position = "bottom",
  add_grid_lines = FALSE
)

# Access individual plots
# plots$ellipse_plot
# plots$voronoi_plot
# plots$voronoi_plot_plus_ellipse

# Create standalone Voronoi plot
voronoi_plot <- create_voronoi_plot(
  data = data,
  class_column = "Species",
  legend_position = "bottom",
  add_grid_lines = FALSE
)

# Create standalone Voronoi plot with island count displayed as subtitle
voronoi_plot_islands <- create_voronoi_plot(
  data = data,
  class_column = "Species",
  legend_position = "bottom",
  add_grid_lines = FALSE,
  show_island_count = TRUE
)
```

`create_tesselation_plots` *Create Tesselation Visualization Plots*

Description

Creates three types of visualization plots for 2D data: ellipse plots, Voronoi diagram plots, and combined ellipse-Voronoi plots. The function is designed to visualize class separation in raw or dimensionally reduced data.

Usage

```
create_tesselation_plots(
  data,
  class_column = NULL,
  alternative_class_column = NULL,
  coordinate_columns = NULL,
  case_labels = NULL,
  coord_names = c("Dim1", "Dim2"),
  title = NULL,
  show_labels = FALSE,
  ellipse_alpha = 0.1,
  voronoi_alpha = 0.3,
  point_size = 2,
  legend_position = "bottom",
  color_palette = NULL,
  add_grid_lines = FALSE,
  color_points = "primary",
  fill_voronoi = "primary",
  point_shape = "none",
  label_fontface = "plain",
  label_size = 3.88,
  show_island_count = FALSE,
  label_islands_only = FALSE
)
```

Arguments

`data`	A data frame containing projected data. Must have at least 2 numeric columns. If more than 2 columns are provided, the first 2 are used as coordinates.
`class_column`	Character string specifying the column name containing class labels, or a vector of class labels. If NULL, all observations are treated as a single class. Default: NULL.
`alternative_class_column`	Character string specifying the column name containing alternative class labels, or a vector of alternative class labels. If NULL, uses class_column. Default: NULL.
`coordinate_columns`	Character vector of length 2 specifying the column names to use as coordinates. If NULL, uses the first two numeric columns. Default: NULL.
`case_labels`	Character vector of case labels for individual observations. If NULL, row numbers are used. Default: NULL.
`coord_names`	Character vector of length 2 specifying names for the coordinate axes. Default: c("Dim1", "Dim2").
`title`	Character string for plot title. If NULL, no title is added. Default: NULL.
`show_labels`	Logical indicating whether to show case labels on plots. Default: FALSE.
`ellipse_alpha`	Numeric value (0-1) for ellipse transparency. Default: 0.1.

`voronoi_alpha`	Numeric value (0-1) for Voronoi polygon transparency. Default: 0.3.
`point_size`	Numeric value for point size. Default: 2.
`legend_position`	Character string or numeric vector specifying legend position. Default: "bottom".
`color_palette`	Function or character vector for color palette. If NULL, uses ggplot2 default colors. Default: NULL.
`add_grid_lines`	Logical indicating whether to add dashed grid lines at origin. Default: FALSE.
`color_points`	Character string specifying which classification to use for point colors. Either "primary" (uses class_column) or "alternative" (uses alternative_class_column). Default: "primary".
`fill_voronoi`	Character string specifying which classification to use for Voronoi fill. Either "primary" (uses class_column) or "alternative" (uses alternative_class_column). Default: "primary".
`point_shape`	Character string specifying which classification to use for point shapes. Either "primary" (uses class_column), "alternative" (uses alternative_class_column), or "none" (no shape differentiation). Default: "none".
`label_fontface`	Character string specifying the font face for text labels. Options include "plain", "bold", "italic", "bold.italic". Default: "plain".
`label_size`	Numeric value specifying the size of text labels. Default: 3.88.
`show_island_count`	Logical indicating whether to display the Voronoi island count as a subtitle on the Voronoi-based plots (`voronoi_plot` and `voronoi_plot_plus_ellipse`). A Voronoi island is a data point whose cell is entirely surrounded by cells belonging to a different class. The count and rate (proportion of all cells that are islands) are shown as a subtitle when TRUE. The `ellipse_plot` is unaffected as it contains no tessellation. The island count is always computed using the primary class labels (class_column), regardless of the fill_voronoi setting. Default: FALSE.
`label_islands_only`	Logical indicating whether to show case labels on plots only for Voronoi islands. Default: FALSE.

Details

The function creates visualizations for 2D (raw or projected) data, particularly useful for displaying results from dimensionality reduction techniques like PCA, PLS-DA, or UMAP.

Voronoi tessellation divides the plot space into regions based on proximity to data points, providing an intuitive visualization of class boundaries and decision regions.

Confidence ellipses show the distribution spread and correlation structure within each class.

The Voronoi island count is a visualization-intrinsic metric with no equivalent in confidence ellipse approaches. Neighbor relationships are determined from the Delaunay triangulation (the `dirsgs` component of the `deldir` output), which is the geometric dual of the Voronoi tessellation: two cells share an edge if and only if their corresponding points are connected by a Delaunay edge. A cell is classified as an island if all its Voronoi neighbors belong to a different class. Island detection requires no additional computation beyond the tessellation itself. The Delaunay triangulation is computed once and reused for both the tessellation plot and the island count, so enabling `show_island_count` incurs no additional computational cost.

Value

A list containing three ggplot objects:

`ellipse_plot`	Plot with confidence ellipses for each class. Unaffected by `show_island_count`.
`voronoi_plot`	Plot with Voronoi tessellation regions. If `show_island_count` = `TRUE`, the subtitle displays the number of Voronoi islands and the island rate (proportion of all cells that are islands).
`voronoi_plot_plus_ellipse`	Combined plot with both Voronoi regions and confidence ellipses. If `show_island_count` = `TRUE`, the subtitle displays the same island count and rate as `voronoi_plot`.

Examples

```
# Basic usage with iris dataset
data <- iris[, c("Sepal.Length", "Petal.Length", "Species")]
plots <- create_tesselation_plots(
  data = data,
  class_column = "Species",
  legend_position = "bottom",
  add_grid_lines = FALSE
)

# With Voronoi island count displayed as subtitle on Voronoi-based plots
plots <- create_tesselation_plots(
  data = data,
  class_column = "Species",
  legend_position = "bottom",
  add_grid_lines = FALSE,
  show_island_count = TRUE
)

# With Voronoi island count displayed as subtitle on Voronoi-based plots
# Label only islands
plots <- create_tesselation_plots(
  data = data,
  class_column = "Species",
  legend_position = "bottom",
  add_grid_lines = FALSE,
  show_island_count = TRUE,
  label_islands_only = FALSE
)
```

`create_voronoi_plot` *Create Voronoi Tesselation Visualization Plot*

Description

Creates a Voronoi tessellation visualization for 2D projected data, showing class separation through proximity-based regions.

Usage

```
create_voronoi_plot(
  data,
  class_column = NULL,
  alternative_class_column = NULL,
  coordinate_columns = NULL,
  case_labels = NULL,
  coord_names = c("Dim1", "Dim2"),
  title = NULL,
  show_labels = FALSE,
  voronoi_alpha = 0.3,
  point_size = 2,
  legend_position = "bottom",
  color_palette = NULL,
  add_grid_lines = FALSE,
  color_points = "primary",
  fill_voronoi = "primary",
  point_shape = "none",
  label_fontface = "plain",
  label_size = 3.88,
  show_island_count = FALSE,
  label_islands_only = FALSE
)
```

Arguments

`data`	A data frame containing projected data. Must have at least 2 numeric columns. If more than 2 columns are provided, the first 2 are used as coordinates.
`class_column`	Character string specifying the column name containing class labels, or a vector of class labels. If NULL, all observations are treated as a single class. Default: NULL.
`alternative_class_column`	Character string specifying the column name containing alternative class labels, or a vector of alternative class labels. If NULL, uses class_column. Default: NULL.
`coordinate_columns`	Character vector of length 2 specifying the column names to use as coordinates. If NULL, uses the first two numeric columns. Default: NULL.

`case_labels`	Character vector of case labels for individual observations. If NULL, row numbers are used. Default: NULL.
`coord_names`	Character vector of length 2 specifying names for the coordinate axes. Default: c("Dim1", "Dim2").
`title`	Character string for plot title. If NULL, no title is added. Default: NULL.
`show_labels`	Logical indicating whether to show case labels on plots. Default: FALSE.
`voronoi_alpha`	Numeric value (0-1) for Voronoi polygon transparency. Default: 0.3.
`point_size`	Numeric value for point size. Default: 2.
`legend_position`	Character string or numeric vector specifying legend position. Default: "bottom".
`color_palette`	Function or character vector for color palette. If NULL, uses ggplot2 default colors. Default: NULL.
`add_grid_lines`	Logical indicating whether to add dashed grid lines at origin. Default: FALSE.
`color_points`	Character string specifying which classification to use for point colors. Either "primary" (uses class_column) or "alternative" (uses alternative_class_column). Default: "primary".
`fill_voronoi`	Character string specifying which classification to use for Voronoi fill. Either "primary" (uses class_column) or "alternative" (uses alternative_class_column). Default: "primary".
`point_shape`	Character string specifying which classification to use for point shapes. Either "primary" (uses class_column), "alternative" (uses alternative_class_column), or "none" (no shape differentiation). Default: "none".
`label_fontface`	Character string specifying the font face for text labels. Options include "plain", "bold", "italic", "bold.italic". Default: "plain".
`label_size`	Numeric value specifying the size of text labels. Default: 3.88.
`show_island_count`	Logical indicating whether to display the Voronoi island count as a plot subtitle. A Voronoi island is a data point whose cell is entirely surrounded by cells belonging to a different class. The count and rate (proportion of all cells that are islands) are shown as a subtitle when TRUE. The island count is always computed using the primary class labels (class_column), regardless of the fill_voronoi setting. Default: FALSE.
`label_islands_only`	Logical indicating whether to show case labels on plots only for Voronoi islands. Default: FALSE.

Details

The function creates a Voronoi tessellation visualization for 2D data, particularly useful for displaying results from dimensionality reduction techniques. Voronoi tessellation divides the plot space into regions based on proximity to data points, providing an intuitive visualization of class boundaries and decision regions.

The Voronoi island count is a visualization-intrinsic metric with no equivalent in confidence ellipse approaches. Neighbor relationships are determined from the Delaunay triangulation (the `dirsgs` component of the `deldir` output), which is the geometric dual of the Voronoi tessellation: two cells share an edge if and only if their corresponding points are connected by a Delaunay edge. A cell is classified as an island if all its Voronoi neighbors belong to a different class. Island detection requires no additional computation beyond the tessellation itself.

Value

A ggplot object showing the Voronoi tessellation plot. If `show_island_count = TRUE`, the plot subtitle displays the number of Voronoi islands and the island rate (proportion of all cells that are islands).

Examples

```
# Basic usage with iris dataset
data <- iris[, c("Sepal.Length", "Petal.Length", "Species")]
plot <- create_voronoi_plot(
  data = data,
  class_column = "Species",
  legend_position = "bottom",
  add_grid_lines = FALSE
)

# With Voronoi island count displayed as subtitle
plot <- create_voronoi_plot(
  data = data,
  class_column = "Species",
  legend_position = "bottom",
  add_grid_lines = FALSE,
  show_island_count = TRUE
)
```

Index

∗ **biomedical**
    `VoronoiBiomedPlot-package`, 2
∗ **multivariate**
    `VoronoiBiomedPlot-package`, 2
∗ **package**
    `VoronoiBiomedPlot-package`, 2
∗ **visualization**
    `VoronoiBiomedPlot-package`, 2

`create_tesselation_plots`, *2*, 3
`create_voronoi_plot`, *2*, 7

`VoronoiBiomedPlot`
        (`VoronoiBiomedPlot-package`*)*, 2
`VoronoiBiomedPlot-package`, 2