Package 'iwaqr'

July 22, 2025

Type Package

Title Irrigation Water Quality Assessment and Visualizations

Version 1.8.4

Description Calculates irrigation water quality ratios and has functions that could be used to plot several popular diagrams for irrigation water quality classification.

Imports ggplot2, ggthemes, ggrepel, scales

Suggests knitr, rmarkdown

VignetteBuilder knitr

RoxygenNote 7.2.3

License GPL (>= 3)

NeedsCompilation no

Author Wajid Ali [aut, cre, cph]

Maintainer Wajid Ali <wajeedaali@gmail.com>

Repository CRAN

Date/Publication 2024-02-23 18:50:02 UTC

Contents

`calculate_Napercent` *Calculate Na percent*

Description

This function calculates the percentage of sodium (Na

Usage

```
calculate_Napercent(df, convert_units = FALSE)
```

Arguments

`df`	A dataframe containing the necessary columns.
`convert_units`	Logical indicating whether to convert values from mg/l to meq/l.

Value

A numeric vector containing Na percent values.

Examples

```
df <- data.frame(Ca = c(10, 20, 30),
Mg = c(5, 10, 15),
Na = c(15, 25, 35), K = c(3, 5, 6))

calculate_Napercent(df, convert_units = TRUE)
```

calculate_PI *Calculate PI*

Description

This function calculates the PI for water quality.

Usage

```
calculate_PI(df, convert_units = FALSE)
```

Arguments

df	dataframe containing the necessary columns,
convert_units	Logical, indicating whether to convert units from mg/l to meq/l.

Value

A numeric vector representing the permeability index (PI) for each row in the dataframe, @examples df <- data.frame(Ca = c(10, 20, 30), Mg = c(5, 10, 15), Na = c(8, 16, 24), Na = c(15, 25, 10), K = c(2, 6, 4), HCO3 = c(15, 30, 45), SO4 = c(110, 115, 88), CO3 = c(0, 0, 0), Cl = c(42, 25, 16)), calculate_PI <- function(df, convert_units = FALSE)

calculate_sar *Calculate SAR*

Description

This function calculates the Sodium Adsorption Ratio (SAR) for water quality.

Usage

```
calculate_sar(df, convert_units = FALSE)
```

Arguments

df	dataframe
convert_units	logical, for conversion to meq/l

Value

SAR values

calculate_tc *Calculate Total Concentration (tc)*

Description

This function calculates the total concentration (tc) based on the provided dataframe.

Usage

```
calculate_tc(df, convert_units = FALSE)
```

Arguments

df	Data frame containing the necessary columns.
convert_units	Logical, indicating whether to convert units from mg/l to meq/l.

Value

A numeric vector representing the total concentration (tc) for each row in the dataframe.

Examples

```
df <- data.frame(Ca = c(10, 20, 30),
Mg = c(5, 10, 15), Na = c(8, 16, 24),
Na = c(15, 25, 10),
K = c(2, 6, 4),
HCO3 = c(15, 30, 45),
SO4 = c(110, 115, 88),
CO3 = c(0, 0, 0),
Cl = c(42, 25, 16))
calculate_tc(df, convert_units = TRUE)
```

CC *Color palette for Wilcox diagram*

Description

This vector defines the color palette used in the Wilcox diagram. It contains a sequence of color names.

Usage

```
CC
```

Format

An object of class `character` of length 12.

irrigationALL *Irrigation Water Quality Index Calculations*

Description

This function calculates multiple water quality indices for irrigation.

Usage

```
irrigationALL(df, convert_to_meq = TRUE)
```

Arguments

df	Dataframe containing necessary variables.
convert_to_meq	Logical, indicating whether to convert concentrations to meq/L (default: TRUE).

Value

Dataframe containing calculated indices: SAR, MAR, SCAR, RSC, RSBC, PI, KR, NaPercentage, and PS.

KR *Kelly Ratio (KR) Calculation*

Description

Kelly Ratio (KR) Calculation

Usage

```
KR(df, convert_to_meq = TRUE)
```

Arguments

df	Dataframe containing necessary variables (Na, Ca).
convert_to_meq	Logical, indicating whether to convert concentrations to meq/L (default: TRUE).

Value

KR value.

MAR *Magnesium Adsorption Ratio (MAR) Calculation*

Description

Magnesium Adsorption Ratio (MAR) Calculation

Usage

```
MAR(df, convert_to_meq = TRUE)
```

Arguments

df	Dataframe containing necessary variables (Mg, Ca).
convert_to_meq	Logical, indicating whether to convert concentrations to meq/L (default: TRUE).

Value

MAR value.

NaPercentage *Sodium Percentage (NaPercentage) Calculation*

Description

Sodium Percentage (NaPercentage) Calculation

Usage

```
NaPercentage(df, convert_to_meq = TRUE)
```

Arguments

df	Dataframe containing necessary variables (Na, Ca, Mg, K).
convert_to_meq	Logical, indicating whether to convert concentrations to meq/L (default: TRUE).

Value

Sodium Percentage value.

PI *Permeability Index (PI) Calculation*

Description

Permeability Index (PI) Calculation

Usage

```
PI(df, convert_to_meq = TRUE)
```

Arguments

df	Dataframe containing necessary variables (Na, HCO3, Ca, Mg).
convert_to_meq	Logical, indicating whether to convert concentrations to meq/L (default: TRUE).

Value

PI value.

plot_DoneenH *Plot Doneen diagram (High permeability) for all rows*

Description

Plot Doneen diagram (High permeability) for all rows

Usage

```
plot_DoneenH(
  df,
  tc_column,
  PI_column,
  label_column = NULL,
  grp_column = NULL,
  convert_units = FALSE
)
```

Arguments

df	Data frame containing the necessary columns.
tc_column	Column name for total concentration (tc).
PI_column	Column name for PI (optional).
label_column	Column name for labels (optional).
grp_column	Column name for grouping (optional).
convert_units	Logical, whether to convert units.

Value

A ggplot object representing the USSL diagram. #' @examples df <- data.frame(tc = c(80, 65, 70), PI = c(30, 65, 150), Color = c("red", "green", "blue")) plot_DoneenH(df, tc_column = "tc", PI_column = "PI", label_column = NULL, grp_column = NULL, convert_units = FALSE)

`plot_DoneenL` *Plot Doneen diagram (Low permeability) for all rows*

Description

This function plots the USSL diagram for the given data frame.

Usage

```
plot_DoneenL(
  df,
  tc_column,
  PI_column,
  label_column = NULL,
  grp_column = NULL,
  convert_units = FALSE
)
```

Arguments

`df`	Data frame containing the necessary columns.
`tc_column`	Column name for total concentration (tc).
`PI_column`	Column name for PI (optional).
`label_column`	Column name for labels (optional).
`grp_column`	Column name for grouping (optional).
`convert_units`	Logical, whether to convert units.

Value

A ggplot object representing the USSL diagram. #'@examples df <- data.frame(tc = c(80, 65, 70), PI = c(30, 65, 150), Color = c("red", "green", "blue")) plot_DoneenL(df, tc_column = "tc", PI_column = "PI", label_column = NULL, grp_column = NULL, convert_units = FALSE)

`plot_DoneenM` *Plot Doneen diagram (Low permeability) for all rows*

Description

This function plots the USSL diagram for the given data frame.

Usage

```
plot_DoneenM(
  df,
  tc_column,
  PI_column,
  label_column = NULL,
  grp_column = NULL,
  convert_units = FALSE
)
```

Arguments

`df`	Data frame containing the necessary columns.
`tc_column`	Column name for total concentration (tc).
`PI_column`	Column name for PI (optional).
`label_column`	Column name for labels (optional).
`grp_column`	Column name for grouping (optional).
`convert_units`	Logical, whether to convert units.

Value

A ggplot object representing the USSL diagram. #' @examples df <- data.frame(tc = c(80, 65, 70), PI = c(30, 65, 150), Color = c("red", "green", "blue")) plot_DoneenM(df, tc_column = "tc", PI_column = "PI", label_column = NULL, grp_column = NULL, convert_units = FALSE)

`plot_Riverside` *Plot Riverside diagram for all rows*

Description

This function plots the USSL diagram for the given data frame.

Usage

```
plot_Riverside(
  df,
  ec_column,
  sar_column,
  label_column = NULL,
  grp_column = NULL,
  convert_units = FALSE
)
```

Arguments

df	containing values in relevant columns
ec_column	Column name for electrical conductivity (EC).
sar_column	Column name for SAR (optional).
label_column	Column name for labels (optional).
grp_column	Column name for grouping (optional).
convert_units	logical wether to convert from mg/l to meq/l

Value

A ggplot object representing the USSL diagram.

a numeric vector containing SAR values

Examples

```
df <- data.frame(EC = c(1000, 2000, 3000),
SAR = c(20, 30, 40),
Color = c("red", "green", "blue"))
plot_Riverside(df, ec_column = "EC" , sar_column = "SAR",grp_column = "Color",
convert_units= FALSE)
```

---

plot_USSL *Plot USSL diagram for all rows*

Description

This function plots the USSL diagram for the given data frame.

Usage

```
plot_USSL(
  df,
  ec_column,
  sar_column,
  label_column = NULL,
  grp_column = NULL,
  convert_units = FALSE
)
```

Arguments

df	Data frame containing the necessary columns.
ec_column	Column name for electrical conductivity (EC).
sar_column	Column name for SAR (optional).
label_column	Column name for labels (optional).
grp_column	Column name for grouping (optional).
convert_units	Logical, whether to convert units.

Value

A ggplot object representing the USSL diagram. #' @examples df <- data.frame(EC = c(1000, 2000, 3000), Na_percent = c(20, 30, 40), Group = c("red", "green", "blue")) plot_USSL(df, ec_column = "EC", sar_column = "SAR", label_column = NULL, grp_column = "Group", convert_units = FALSE)

---

plot_Wilcox *Plot Wilcox diagram for all rows*

Description

This function plots the USSL diagram for the given data frame.

Usage

```
plot_Wilcox(
  df,
  ec_column,
  Napercent_column,
  label_column = NULL,
  grp_column = NULL,
  convert_units = FALSE
)
```

Arguments

df	containig relevant columns with values
ec_column	Column name for electrical conductivity (EC).
Napercent_column	Column name for Na percent (optional).
label_column	Column name for labels (optional).
grp_column	Column name for grouping (optional).
convert_units	logical wether to convert values from mg/l to meq/l

Value

A ggplot object representing the Wilcox diagram.

#' @examples f <- data.frame(EC = c(1000, 2000, 3000), Na_percent = c(20, 30, 40), Color = c("red", "green", "blue")) plot_Wilcox(f, ec_column = "EC", Napercent_column = "Na_percent", label_column = NULL, grp_column = "Color", convert_units = FALSE)

a numeric vector containing Na percent values

PS *Potential Salinity (PS) Calculation*

Description

Potential Salinity (PS) Calculation

Usage

```
PS(df, convert_to_meq = TRUE)
```

Arguments

df	Dataframe containing necessary variables (Cl, SO4).
convert_to_meq	Logical, indicating whether to convert concentrations to meq/L (default: TRUE).

Value

PS value.

RSBC *Residual Sodium Bicarbonate (RSBC) Calculation*

Description

Residual Sodium Bicarbonate (RSBC) Calculation

Usage

```
RSBC(df, convert_to_meq = TRUE)
```

Arguments

df	Dataframe containing necessary variables (HCO3, Ca).
convert_to_meq	Logical, indicating whether to convert concentrations to meq/L (default: TRUE).

Value

RSBC value.

RSC *Residual Sodium Carbonate (RSC) Calculation*

Description

Residual Sodium Carbonate (RSC) Calculation

Usage

```
RSC(df, convert_to_meq = TRUE)
```

Arguments

df	Dataframe containing necessary variables (HCO3, CO3, Ca, Mg).
convert_to_meq	Logical, indicating whether to convert concentrations to meq/L (default: TRUE).

Value

RSC value.

SAR *Sodium Adsorption Ratio (SAR) Calculation*

Description

Sodium Adsorption Ratio (SAR) Calculation

Usage

```
SAR(df, convert_to_meq = TRUE)
```

Arguments

df	Dataframe containing necessary variables (Na, Ca, Mg, K).
convert_to_meq	Logical, indicating whether to convert concentrations to meq/L (default: TRUE).

Value

SAR value.

SCAR *Sodium Adsorption Ratio (SCAR) Calculation*

Description

Sodium Adsorption Ratio (SCAR) Calculation

Usage

```
SCAR(df, convert_to_meq = TRUE)
```

Arguments

df	Dataframe containing necessary variables (Na, Ca).
convert_to_meq	Logical, indicating whether to convert concentrations to meq/L (default: TRUE).

Value

SCAR value.

Index