Package ‘SoilManageR’

January 20, 2025

Type Package

Title Calculate Soil Management Indicators for Agricultural Practice Assessment

Version 1.0.1

Description Calculate numerical agricultural soil management indicators from on a management timeline of an arable field. Currently, indicators for carbon (C) input into the soil system, soil tillage intensity rating (STIR), number of soil cover and living plant cover days, N fertilization and livestock intensity, and plant diversity are implemented.
The functions can also be used independently of the management timeline to calculate some indicators. The package contains tables with reference information for the functions, as well as a '*.xlsx' template to collect the management data.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.3.1

Imports magrittr (>= 2.0.3), dplyr (>= 1.1.4), lubridate (>= 1.9.2), ggplot2 (>= 3.4.4), ggthemes (>= 5.0.0), tidyr (>= 1.3.0), tibble (>= 3.2.1), readxl (>= 1.4.3), Rdpack (>= 2.6)

Depends R (>= 2.10)

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

RdMacros Rdpack

URL https://gitlab.com/SoilManageR/

BugReports https://gitlab.com/SoilManageR/SoilManageR/-/issues

NeedsCompilation no

Author Olivier Heller [cre, aut] (<https://orcid.org/0000-0002-5918-4161>), Raphaël Wittwer [ctb] (<https://orcid.org/0000-0002-2129-7195>)

Maintainer Olivier Heller <olivier.heller@agroscope.admin.ch>

Repository CRAN

Date/Publication 2024-10-01 10:00:07 UTC

Contents

`arrange_management_df` *arrange management_df by date, category*

Description

the funciton arranges the management_df by date and by category of operations. The order of the operations is harvest, fertilizer_application,crop_protection, tillage, sowing, irrigation, other

Usage

```
arrange_management_df(var_MGMT_data, include.combination = TRUE)
```

Arguments

`var_MGMT_data` management_df to be arranged

`include.combination`
logical, if the combinations should considered Default is TRUE

Value

a rearranged management_df

Examples

```
#rearrange EXAMPLE data
arrange_management_df(EXAMPLE_data)
```

`calculate_indicators` *Calculate all soil management indicators*

Description

Checks the `management_df` for consitency with the `check_management_df()` function. Then it calculates the `C_input()`, `tillage_intensity()`, `soil_cover()`,`plant_diversity()` `N_input()` and `productivity_indicator()`.

Usage

```
calculate_indicators(var_MGMT_data)
```

Arguments

`var_MGMT_data` a `management_df` with a management history

Value

data frame with indices per year

See Also

- `check_management_df()`
- `C_input()`
- `tillage_intensity()`
- `soil_cover()`
- `N_input()`
- `plant_diversity()`
- `productivity_indicator()`

Examples

```
#example input
calculate_indicators(EXAMPLE_data)
```

check_management_df *Check management_df for consitency*

Description

The function checkes objects of the class management_df() for internal consistency. It formally checks the class and the column names. Additionally, the function checks if dates are consistently increasing and if all organic amendments, tillage and sowing devices and crops are in the relevant look-up-tables. Furthermore, the amount of organic amendments (<100t/ha) and N fertilizer (<100kgN/ha) application rates per event are checked. The depth of tillage operations are compared with the min and max depth from the `STIR_value_LUT`. Finally, the order of tillage, sowing and harvest operations are checked for plausibility (see details for more information).

Usage

```
check_management_df(var_MGMT_data)
```

Arguments

`var_MGMT_data`	an object of the management_df() class

Details

The order of tillage, sowing and harvest operations are checked with the following assumptions:

- after "stubble_cultivation" the following operations are allowed: "stubble_cultivation", "primary_tillage", "seedbed_preparation", "sowing_main_crop", "sowing_cover_crop"
- after "primary_tillage" the following operations are allowed: "seedbed_preparation", "sowing_main_crop", "sowing_cover_crop"
- after "seedbed_preparation" the following operations are allowed: "stubble_cultivation", "seedbed_preparation", "sowing_main_crop", "sowing_cover_crop"
- after "sowing_main_crop" the following operations are allowed: "sowing_main_crop", "harvest_main_crop"
- after "sowing_cover_crop" the following operations are allowed: "stubble_cultivation", "primary_tillage", "seedbed_preparation", "sowing_main_crop", "sowing_cover_crop"
- after "harvest_main_crop" the following operations are allowed: "harvest_main_crop", "straw_removal", "stubble_cultivation", "primary_tillage", "seedbed_preparation", "sowing_main_crop", "sowing_cover_crop"
- after "straw_removal" the following operations are allowed: "harvest_main_crop", "straw_removal", "stubble_cultivation", "primary_tillage", "seedbed_preparation", "sowing_main_crop", "sowing_cover_crop"

Additionally, there are exceptions for potato crops: "bedder" can be used after a "potato_planter" and "mulching" can be applied before a "potato_harvester"

Value

a test_report list (only returned if some tests failed)

See Also

- `management_df()` for creating an management_df
- `management_df_from_excel()` for importing a management_df from an excel template

Examples

```
#example input
check_management_df(EXAMPLE_data)
```

CN_input_amendments *Estimate C and N inputs of organic amendments*

Description

Estimates the carbon (C) and total nitrogen (N) input into the soil system by organic amendments.

Usage

```
CN_input_amendments(
  amount,
  amd_type = NA,
  DMC = NA,
  C_content = NA,
  N_content = NA,
  return.comment = FALSE,
  concentration_liquids = 0.5
)
```

Arguments

`amount`	Amount of organic amendment in fresh weight (t/ha)
`amd_type`	Type of organic manure, there are default values available for "Slurry_dairy_cow", "Manure_dairy_cow", "Slurry_cattle", "Manure_cattle", "Slurry_pig", "Manure_pig", "Compost", "Biorga_Quick_12N" and others (see `CN_input_amendments_LUT`)
`DMC`	dry matter content of the organic amendment (gDM/kgFM)
`C_content`	C content of the dry matter (gC/kgDM)
`N_content`	N content of the dry matter (gN/kgDM)
`return.comment`	(optional): logical value if comments are returned or not. Default = FALSE
`concentration_liquids`	concentration factor for liquid amendments. Default value is .5.

Details

The C and N inputs by organic amendments is calculated based on the dry matter content (DMC) and the C and N content of the dry matter of the amendment. If the contents are not specified, default values from the Swiss fertilizer recommendations (Sinaj et al. 2017) are assumed. The default values are available the look-up-table `CN_input_amendments_LUT` For all slurries and liquid amendments (DMC < 150 g/kg), a dilution of 50% is assumed if default DMC values from the `CN_input_amendments_LUT` are used.

Value

a tibble with the following parameters:

- C_input_org: C input by organic ammendment (kgC/ha)
- N_input_org: N input by organic ammendment (kgN/ha)
- comment (optional): Source of information on properties of organic amendment

References

Sinaj S, Charles R, Baux A, Dupuis B, Hiltbrunner J, Levy Häner L, Pellet D, Blanchet G, Jeangros B (2017). "Grundlagen für die Düngung landwirtschaftlicher Kulturen in der Schweiz (GRUD): Düngung von Ackerkulturen." *Agrarforschung Schweiz, Spezialpublikation*, **Chapter 8**(6), 1–46. https://ira.agroscope.ch/en-US/Page/Publikation/Index/36799.

See Also

- `C_input()` to calculate C inputs for a management_df
- `N_input()` to calculate N inputs for a management_df
- `C_input_crops()` to calculate C inputs by crops
- `C_input_cover_crops()` to calculate C input for cover crops
- `CN_input_amendments_LUT()` for the look-up-table for organic amendments reference values

Examples

```
#example where amount, dry matter content, C and N content are known.
CN_input_amendments(40, DMC = 300, C_content = 300, N_content = 30)

#example where only amount and type of amendment are known
CN_input_amendments(20, "Manure_pig")

#example of a diluted slurry
CN_input_amendments(20, amd_type = "Slurry_dairy_cow", DMC = 50)
```

`CN_input_amendments_LUT`
Look-up-table with default values to calculate C and N inputs by organic amendments

---

Description

The dataset is a look-up-table that is used to calculate the carbon (C) and nitrogen (N) inputs by organic amendments with the function 'SoilManageR::CN_input_amendments()'. The data set is produced from the excel table 'CN_input_amendments_LUT.xlsx' file under '/inst/extdata/'.

Usage

```
CN_input_amendments_LUT
```

Format

A tibble with 27 rows and 7 columns:

Amendment Name of the amendment

DMC Dry matter content of the amendment [gDM/kgFM]

OM Organic matter content of the amendment, relative to its fresh weight [gDM/kgFM]

C_content Carbon content of the amendment, relative to its dry matter [gC/kgDM]

N_tot Total N conent of the amendment, relative to it's dry fresh weight [gN/kgFM]

N_content Nitrogen content of the amendment, relative to its dry matter [gN/kgDM]

Comment Comment, e.g. source of the information. These lines are shown as part of the function output

References

Compilation of Values from the SoilX project. Please check the 'CN_input_amendments_LUT.xlsx' file under '/inst/extdata/' for more information

Sinaj S, Charles R, Baux A, Dupuis B, Hiltbrunner J, Levy Häner L, Pellet D, Blanchet G, Jeangros B (2017). "Grundlagen für die Düngung landwirtschaftlicher Kulturen in der Schweiz (GRUD): Düngung von Ackerkulturen." *Agrarforschung Schweiz, Spezialpublikation*, **Chapter 8**(6), 1–46. https://ira.agroscope.ch/en-US/Page/Publikation/Index/36799.

Richner W, Flisch R, Mayer J, Schlegel P, Zähner M, Menzi H (2017). "Grundlagen für die Düngung landwirtschaftlicher Kulturen in der Schweiz (GRUD): Eigenschaften und Anwendung von Düngern." *Agrarforschung Schweiz, Spezialpublikation*, **Chapter 4**(6), 1–24. https://ira.agroscope.ch/en-US/publication/41476.

C_input *Estimate carbon input*

Description

C_input() estimates the carbon (C) input into the soil system per year of a management_df.

Usage

```
C_input(var_MGMT_data, extended.output = FALSE)
```

Arguments

var_MGMT_data a management_df that contains the management information

extended.output
an optional logical value.

- If FALSE, C input values are aggregated by year.
- If TRUE, a tibble with daily resolution is returned.
- Default value is FALSE

Details

The function takes a management_df as input and returns a C input values per year in the management_df.

Alternatively, it can return a tibble with all managememnt operations and their respective C input values.

The functions calculates the C input with the C_input_crops(), C_input_cover_crops() and CN_input_amendments() functions.

Value

- By default, a tibble with C input values (total and by category) by year is returned.
- If extended.output = TRUE, a tibble with all management operations and their respecitve C inputs is returned.

See Also

- calculate_indicators() to calculate all management indicators for a management_df
- C_input_crops() to calculate C input for crops
- C_input_cover_crops() to calculate C input for cover crops
- CN_input_amendments() to calculate C (and N) inputs for organic amendments

Examples

```
#example that returns annual C input values
C_input(EXAMPLE_data)

#example that returns a tibble with all operations that have a C input
C_input(EXAMPLE_data, extended.output = TRUE)
```

C_input_cover_crops *Estimate C inputs by cover crops*

Description

This function estimates the Carbon (C) input into the soil system by cover crops based on the duration of the cover crop stand.

Usage

```
C_input_cover_crops(
  abvg_biomass = NA,
  days = 180,
  min_C_abvg = 1253,
  min_days = 180,
  max_C_abvg = 1916,
  max_days = 240,
  Cc_biomass = 450
)
```

Arguments

abvg_biomass	(optional): Dry weight of aboveground biomass of the cover crop (tDM/ha)
days	(optional): Number of days that the cover crop was established. If no value is provided, mind_days is assumed (days)
min_C_abvg	(optional): Minimal above ground C that the cover crop produces, given it is established for the min_days number of days. Default value is 1.253 (kgC/ha)
min_days	(optional): Number of days where the interpolation of the biomass starts. The default value is 180 (days)
max_C_abvg	(optional): Maximum biomass that the cover crop can produce. Default value is 1.916 (kgC/ha)
max_days	(optional): Number of days when the maximum biomass of the cover crop is reached. The default value is 240 (days)
Cc_biomass	(optional): Assumed C content of the cover crop biomass. Default value is 450 (kgC/tDM)

Details

C_input_cover_crops() internally calls C_input_crops() to calculate the different C fractions. The C in the above ground biomass ($C_{Product}$) is a function of the time a cover crop is established. A minimum and a maximum cover crop biomass are assumed at 180 and 240 days respectively, and linearly interpolated for the period in between.

$$C_{Product} = \begin{cases} 1253\ kgC/ha\ ,\ duration < 180\ days \\ 1253\ kgC/ha\ +\ (duration - 180\ days) * \frac{663\ kgC/ha}{60\ days}\ ,\ 180\ days \leq duration \leq 240\ days \\ 1916\ kgC/ha\ ,\ duration > 240\ days \end{cases}$$

Assumptions on the C inputs at day 180 and 240 are based on values extracted from Seitz et al. (2022).

The remaining parameters to calculate the C input by cover crops are `HI = 1`, `SER = 3.67`, and `REF = 0.31`, all derived from Seitz et al. (2022).

Note, that with these assumptions the C input of short term cover crops (e.g. few weeks) is overestimated.

The function C_input_cover_crops() estimates the C input by applying the assumptions mentioned above. Alternatively, the user can supply an above ground biomass and a CC of the biomass, or other parameters to estimate the C input by cover crops.

Value

a tibble with the following parameters:

- C_input_product: Estimated soil carbon input from product (i.e., the cover corp aboveground biomass) (`kgC/ha`),
- C_input_straw: Estimated soil carbon input by straw or other residues (typically 0 for cover crops) (`kgC/ha`),
- C_input_root: Estimated soil carbon input by roots (`kgC/ha`),
- C_input_exudate: Estimated soil carbon input by roots (`kgC/ha`),
- C_input_total: Total estimated Soil carbon input, sum of C_input_straw, C_input_root, C_input_exudate (`kgC/ha`),

References

Seitz D, Fischer LM, Dechow R, Wiesmeier M, Don A (2022). "The potential of cover crops to increase soil organic carbon storage in German croplands." *Plant and Soil*, **488**(1-2), 157–173. doi:10.1007/s1110402205438w.

See Also

- C_input() to calculate C inputs for a `management_df`
- C_input_crops() to calculate C input for crops
- CN_input_amendments() to calculate C (and N) inputs for organic amendments
- C_input_crops_LUT() for the look-up-table for crop reference values

Examples

```
#example when only the duration is known
C_input_cover_crops(days = 205)

#example if the cover crop biomass is known
C_input_cover_crops(abvg_biomass = 2.5, Cc_biomass = 450)

#example with custom assumptions on the above ground biomass developnent over time
C_input_cover_crops(days = 60, min_C_abvg = 600 , min_days = 50, max_C_abvg = 1916, max_days = 240)
```

C_input_crops *Estimate C inputs by crops*

Description

Calculates the estimated carbon (C) input into the soil system by harvested main crops.

Usage

```
C_input_crops(
  crop,
  crop_product = NA,
  crop_residue = NA,
  harvest_index = NA,
  variable_harvest_index = NA,
  HI_intercept = NA,
  HI_slope = NA,
  shoot_root_ratio = NA,
  root_exudation_factor = NA,
  Cc_product = 450,
  Cc_residue = 450,
  Cc_root = 450,
  straw.removal = NA,
  fixed_belowground_input = NA,
  fixed_C_input_root = NA,
  return.comment = FALSE
)
```

Arguments

`crop`	Crop type. Must match predefined list (see `C_input_crops_LUT`)
`crop_product`	(optional) Dry weight of the exported product, i.e. yield. Default value is taken from table (tDM/ha)
`crop_residue`	(optional) Dry weight of the residues of the main crop (e.g., straw, sugar beet leaves) (tDM/ha)
`harvest_index`	(optional) Ratio of the product to the total above ground biomass

variable_harvest_index
: (optional) Logical value that is TRUE if the variable harvest index assumptions of Fan et al. (2017) are to be applied (TRUE / FALSE)

HI_intercept
: (optional) Intercept of the variable harvest index. Values provided by Fan et al. (2017)

HI_slope
: (optional) Slope of the variable harvest index. Values provided by Fan et al. (2017) (ha/tDM)

shoot_root_ratio
: (optional) Ratio of the total above ground biomass to the root biomass

root_exudation_factor
: (optional) Ratio of the root exudated C to the C in the root biomass

Cc_product
: (optional) C concentration in the exported product. Default value is 450 (gC/kgDM)

Cc_residue
: (optional) C concentration in the residues of the main crop. Default value is 450 (gC/kgDM)

Cc_root
: (optional) C concentration in the roots. Default value is 450 (gC/kgDM)

straw.removal
: (optional) Logical value that is TRUE if straw is removed at harvest.

- When NA then SR value from table is take
- When TRUE SS is 1
- When FALSE SS is 0.
- Default value is NA

fixed_belowground_input
: (optional) Logical value that is TRUE if fixed below ground Carbon inputs are to be assumed (e.g. for temporary leys) (TRUE / FALSE)

fixed_C_input_root
: (optional) amount of root C that is assumed if fixed_belowground_input is TRUE (kgC/ha)

return.comment
: (optional) logical value if comment are returned or not. Default = FALSE (TRUE/FALSE)

Details

The annual C input by crops were estimated based on crop type and crop yield with the allometric functions of Bolinder et al. (2007):

$$C_{Product} = Product * CC_{Product}$$
$$C_{Straw} = Product * \frac{1 - HI}{HI} * CC_{Straw}$$
$$C_{Root} = \frac{Product}{SRR * HI} * CC_{Root}$$
$$C_{Exudates} = C_{Root} * REF$$

Where C is the C per fraction (in kgC/ha) and CC is the C content of given fraction (kgC/tDM). $Prodcuct$ is the dry matter yield of a crop in tDM/ha, HI is the harvest index (ratio of product total of product and straw), SRR is the ratio of the shoot biomass (product and straw) to the root biomass, and REF is the root exudation factor (i.e., the ratio of the C exudated by the roots to the C in the root biomass). All fractions are multiplied with a crop and fraction specific S-factor that determines the share of the fraction that is returned to the soil.

If not mentioned otherwise parameters were taken from the publications of Bolinder et al. (2007), Keel et al. (2017) or Wüst-Galley et al. (2020). Parameters for potatoes and sugar beets were derived from Bolinder et al. (2015) For temporary leys we assumed yield-independent annual C_{Root} of 1.5 MgC/ha and a REF of 0.5 (Taghizadeh-Toosi et al. 2020). Furthermore, like Wüst-Galley et al. (2020), the belowground C input ($C_{Root} + C_{Exudates}$) of corn maize, silage maize and cereals were fixed to 0.46 MgC/ha, 1.1 MgC/ha and 0.6 MgC/ha respectively, based on the values from Hirte et al. (2018). Additionally, we applied the yield dependent harvest index ($HI = Intercept + Product * Slope$) proposed by Fan et al. (2017) for cereals, faba beans, peas, corn, rapeseed, and soybeans.

Reference yields were derived from the Swiss fertilizer recommendations Sinaj et al. (2017).

All default values can be found in the look-up-table `C_input_crops_LUT`.

Value

a tibble with the following parameters:

- C_input_product: Estimated soil carbon input from product (e.g. damaged patatos) (kgC/ha),
- C_input_straw: Estimated soil carbon input by straw or other residues (kgC/ha),
- C_input_root: Estimated soil carbon input by roots (kgC/ha),
- C_input_exudate: Estimated soil carbon input by roots (kgC/ha),
- C_input_total: Total estimated Soil carbon input, sum of C_input_straw, C_input_root, C_input_exudate (kgC/ha),
- comment (optional): comment on the derived values

References

Bolinder MA, Janzen HH, Gregorich EG, Angers DA, VandenBygaart AJ (2007). "An approach for estimating net primary productivity and annual carbon inputs to soil for common agricultural crops in Canada." *Agriculture, Ecosystems & Environment*, **118**(1-4), 29–42. doi:10.1016/j.agee.2006.05.013.

Bolinder MA, Kätterer T, Poeplau C, Börjesson G, Parent LE (2015). "Net primary productivity and below-ground crop residue inputs for root crops: Potato (*Solanum tuberosum* L.) and sugar beet (*Beta vulgaris* L.)." *Canadian Journal of Soil Science*, **95**(2), 87–93. doi:10.4141/cjss2014091.

Fan J, McConkey B, Janzen H, Townley-Smith L, Wang H (2017). "Harvest index - yield relationship for estimating crop residue in cold continental climates." *Field Crops Research*, **204**, 153–157. doi:10.1016/j.fcr.2017.01.014.

Hirte J, Leifeld J, Abiven S, Oberholzer H, Mayer J (2018). "Below ground carbon inputs to soil via root biomass and rhizodeposition of field-grown maize and wheat at harvest are independent of net primary productivity." *Agriculture, Ecosystems & Environment*, **265**, 556–566. doi:10.1016/j.agee.2018.07.010.

Keel SG, Leifeld J, Mayer J, Taghizadeh-Toosi A, Olesen JE (2017). "Large uncertainty in soil carbon modelling related to method of calculation of plant carbon input in agricultural systems." *European Journal of Soil Science*, **68**(6), 953–963. doi:10.1111/ejss.12454.

Sinaj S, Charles R, Baux A, Dupuis B, Hiltbrunner J, Levy Häner L, Pellet D, Blanchet G, Jeangros B (2017). "Grundlagen für die Düngung landwirtschaftlicher Kulturen in der Schweiz (GRUD): Düngung von Ackerkulturen." *Agrarforschung Schweiz, Spezialpublikation*, **Chapter 8**(6), 1–46. `https://ira.agroscope.ch/en-US/Page/Publikation/Index/36799`.

Taghizadeh-Toosi A, Cong W, Eriksen J, Mayer J, Olesen J, Keel SG, Glendining M, Kätterer T, Christensen BT (2020). "Visiting dark sides of model simulation of carbon stocks in European temperate agricultural soils: allometric function and model initialization." *Plant and Soil*, **450**(1-2), 255–272. doi:10.1007/s11104020045009.

Wüst-Galley C, Keel SG, Leifeld J (2020). "A model-based carbon inventory for Switzerland's mineral agricultural soils using RothC." *Agroscope Science*, 1–110. doi:10.34776/as105e.

See Also

- `C_input()` to calculate C inputs for a `management_df`
- `C_input_cover_crops()` to calculate C input for cover crops
- `CN_input_amendments()` to calculate C (and N) inputs for organic amendments
- `C_input_crops_LUT()` for the look-up-table for crop reference values

Examples

```
#example without yield information, default yield is assumed
C_input_crops("wheat, winter")

#example with yield information and straw retention
C_input_crops("barley, spring", crop_product = 4.5, straw.removal = FALSE)

#example with more information
C_input_crops("barley, spring", crop_product = 4.5, harvest_index = 0.4,
              shoot_root_ratio = 2.4, root_exudation_factor = 0.5)

#example with variable harvest index
C_input_crops("barley, spring", crop_product = 4.5, variable_harvest_index = TRUE,
              HI_intercept = 0.35, HI_slope = 0.015, shoot_root_ratio = 2.4,
              root_exudation_factor = 0.5)

#example with fixed below ground input
C_input_crops("maize, silage", crop_product = 18.5,
              fixed_belowground_input = TRUE, fixed_C_input_root = 1500,
              root_exudation_factor = 0.3)
```

`C_input_crops_LUT` *Look-up-table with default values to calculate carbon (C) inputs by crops*

Description

The data set is a look-up-table that is used to calculate the C inputs by crops with the Bolinder formula, that is implemented in the function ‘SoilManageR::C_input_crops()‘. The data set is produced from the excel table ‘C_input_crops_LUT.xlsx‘ file under ‘/inst/extdata/‘.

Usage

```
C_input_crops_LUT
```

Format

A tibble with 28 rows and 19 columns:

Crop Name of the crop

RP Ratio of the C in the product to the total carbon that is allocated by the plant (in a year)

RS Ratio of the C in the above ground residues (e.g. straw) to the total carbon that is allocated by the plant (in a year)

RR Ratio of the C in the plant roots to the total carbon that is allocated by the plant (in a year)

RE Ratio of the C in the root exudates to the total carbon that is allocated by the plant (in a year)

SP Proportion of the C in the Product that is transfered to the soil

SS Proportion of the C in the above ground residues (e.g. straw) that is transfered to the soil

SR Proportion of the C in the roots that is transfered to the soil

SE Proportion of the C in root exudates that is transfered to the soil

crop_product Reference yield, derived from the Swiss fertilizer recommendations (GRUD, 2017, Chapters 8 and 9) [tDM/ha]

harvest_index Ratio of the product to the total of the product and the above ground residues. Calculated by RP/(RP+RS) (assuming all biomass has 45% C)

varible_harvest_index Logical value, if the variable harvest index assumption of Fan et al. (2017) are to be aplied or not.

HI_intercept Intercept of the variable harvest index assumption of Fan et al. (2017) are to be aplied.

HI_slope Slope of the variable harvest index assumption of Fan et al. (2017) are to be aplied. [ha/tDM]

shoot_root_ratio Ratio of the product and the above ground residues to the root biomass. Calculated by (RP+RS)/RR (assuming all biomass has 45% C)

root_exudation_factor Ratio of the root exudates to the root biomass. Calculated by RE/RR (assuming all biomass has 45% C)

fixed_belowground_input Logical value if the fixed below ground C allocation assumption of Taghizadeh-Toosi et al. (2020) is to be applied or not.

C_input_root Fixed value of root carbon input that is to be assumed. [kgC/ha]

Source Source where the information was derived.

References

Compilation of values from the SoilX project. Please check the 'C_input_crops_LUT.xlsx' file under '/inst/extdata/' for more information.

Bolinder MA, Janzen HH, Gregorich EG, Angers DA, VandenBygaart AJ (2007). "An approach for estimating net primary productivity and annual carbon inputs to soil for common agricultural crops in Canada." *Agriculture, Ecosystems & Environment*, **118**(1-4), 29–42. doi:10.1016/j.agee.2006.05.013.

Bolinder MA, Kätterer T, Poeplau C, Börjesson G, Parent LE (2015). "Net primary productivity and below-ground crop residue inputs for root crops: Potato (*Solanum tuberosum* L.) and sugar beet (*Beta vulgaris* L.)." *Canadian Journal of Soil Science*, **95**(2), 87–93. doi:10.4141/cjss2014091.

Fan J, McConkey B, Janzen H, Townley-Smith L, Wang H (2017). "Harvest index - yield relationship for estimating crop residue in cold continental climates." *Field Crops Research*, **204**, 153–157. doi:10.1016/j.fcr.2017.01.014.

Hirte J, Leifeld J, Abiven S, Oberholzer H, Mayer J (2018). "Below ground carbon inputs to soil via root biomass and rhizodeposition of field-grown maize and wheat at harvest are independent of net primary productivity." *Agriculture, Ecosystems & Environment*, **265**, 556–566. doi:10.1016/j.agee.2018.07.010.

Keel SG, Leifeld J, Mayer J, Taghizadeh-Toosi A, Olesen JE (2017). "Large uncertainty in soil carbon modelling related to method of calculation of plant carbon input in agricultural systems." *European Journal of Soil Science*, **68**(6), 953–963. doi:10.1111/ejss.12454.

Seitz D, Fischer LM, Dechow R, Wiesmeier M, Don A (2022). "The potential of cover crops to increase soil organic carbon storage in German croplands." *Plant and Soil*, **488**(1-2), 157–173. doi:10.1007/s1110402205438w.

Sinaj S, Charles R, Baux A, Dupuis B, Hiltbrunner J, Levy Häner L, Pellet D, Blanchet G, Jeangros B (2017). "Grundlagen für die Düngung landwirtschaftlicher Kulturen in der Schweiz (GRUD): Düngung von Ackerkulturen." *Agrarforschung Schweiz, Spezialpublikation*, **Chapter 8**(6), 1–46. `https://ira.agroscope.ch/en-US/Page/Publikation/Index/36799`.

Taghizadeh-Toosi A, Cong W, Eriksen J, Mayer J, Olesen J, Keel SG, Glendining M, Kätterer T, Christensen BT (2020). "Visiting dark sides of model simulation of carbon stocks in European temperate agricultural soils: allometric function and model initialization." *Plant and Soil*, **450**(1-2), 255–272. doi:10.1007/s11104020045009.

Wüst-Galley C, Keel SG, Leifeld J (2020). "A model-based carbon inventory for Switzerland's mineral agricultural soils using RothC." *Agroscope Science*, 1–110. doi:10.34776/as105e.

EXAMPLE_data *Example of a management_df*

Description

The dataset is derived from a Swiss long term agricultural field experiment. It is intended for demonstration purposes only

Usage

```
EXAMPLE_data
```

Format

A tibble with 130 rows and 13 columns:

crop Name of the main crop. Cover crop related operations are linked to the next main crop in the rotation [String from list]

date Date of the management operation [Date]

year Year of the management operation [Integer]

category Categorization of the managment operation [1 level] [String from list]

operation Categorization of the managment operation [2 level] [String from list]

device Categorization of the managment operation [3 level] [String from list]

value Numerical value linked to managment operation (e.g., depth of tillage operation, mass of organic amendment) [Integer]

unit Unit of the numerical value (e.g. cm, t/ha) [String from list]

machine Further information on the machine used (e.g., type, manufacturer, tool) [String [UTF-8]]

product Further information on the applied product (e.g., name, manufacturer, C content) [String [UTF-8]]

combination Indicate if a operation was done in combination with others. Use consequtive integer numbers if combinded operations occur. Leave empty if not combined

comments Comments related to the management operation [String [UTF-8]]

DMC Dry matter content of organic amendments [gDM/kgFM]

C_content Carbon content of the amendments, relative to its dry matter [gC/kgDM]

N_content Nitrogen content of organic amendments, relative to its dry matter [gN/kgDM]

crop_product Crop product yield [tDM/ha]

crop_residue Crop residue mass [tDM/ha]

Cc_product Carbon content of the crop product [gC/kgDM$]

Cc_residue Carbon content of the crop residue [gC/kgDM] filter_management_df *Filter management_df for pattern in comments*

Description

Excludes operations (lines) from management_df based on a character pattern in the comments column.

Usage

```
filter_management_df(var_MGMT_data, filter_pattern)
```

Arguments

var_MGMT_data	management_df
filter_pattern	string based on which the lines should be excluded

Value

a filtered management_df

Examples

```
# filter EXAMPLE_data and exclude all lines that contain "UFA 330" in comments
filter_management_df(EXAMPLE_data,"UFA 330")
```

management_df *Constructor for management_df*

Description

This function is a constructor for empty objects of the management_df() class, the core of the SoilManageR package.

Usage

```
management_df(
  crop = NA,
  year = NA,
  date = NA,
  category = NA,
  operation = NA,
  device = NA,
  value = NA,
  unit = NA,
```

```
    machine = NA,
    product = NA,
    combination = NA,
    comments = NA,
    DMC = NA,
    C_content = NA,
    N_content = NA,
    crop_product = NA,
    crop_residue = NA,
    Cc_product = NA,
    Cc_residue = NA
  )
```

Arguments

`crop`	Name of the main crop. Must match a pre-existing list
`year`	Year of the management operation ("YYYY")
`date`	Date of the management operation ("YYYY-MM-DD")
`category`	Categorization of the managment operation (1 level). Must match a pre-existing list.
`operation`	Categorization of the managment operation (2 level). Must match a pre-existing list.
`device`	Categorization of the managment operation (3 level). Must match a pre-existing list.
`value`	Numerical value linked to managment operation (e.g., depth of tillage operation, mass of organic amendment)
`unit`	Unit of the numerical value (e.g. cm, t/ha)
`machine`	Further information on the machine used (e.g., type, manufacturer, tool)
`product`	Further information on the applied product (e.g., name, manufacturer). Must match pre-existing list for organic amendments.
`combination`	Indicate if a operation was done in combination with others. Use consequtive integer numbers if combinded operations occur. Leave empty if not combined.
`comments`	Comments related to the management operation
`DMC`	Dry matter content of organic amendments (gDM/kgFM)
`C_content`	Carbon content of the amendments, relative to its dry matter (gC/kgDM)
`N_content`	Nitrogen content of organic amendments, relative to its dry matter (gN/kgDM)
`crop_product`	Crop product yield (tDM/ha)
`crop_residue`	Crop residue mass (tDM/ha)
`Cc_product`	Carbon content of the crop product (gC/kgDM)
`Cc_residue`	Carbon content of the crop residue (gC/kgDM)

Value

a `management_df()`

See Also

- `management_df_from_excel()` for importing a `management_df` from an excel template
- `check_management_df()` to check the integrity of a `management_df`
- `EXAMPLE_data()` for an example of a `management_df`

Examples

```
#creation of an empty management_df
management_df()
```

---

management_df_from_excel
Import management_df from excel file

Description

This function imports management data from an excel template and transforms it into a management_df. Additionaly, it checks if all columns that are expected for a managment_df are available. The excel template can be found in the SoilManageR Package under inst/extdata/SoilManageR_mgmt_data_template. Optionaly, the parameter *year* can be overwritten by the year extracted from *date*.

Usage

```
management_df_from_excel(
  path_to_xlsx,
  var_sheet = "Management_template",
  overwrite_year = TRUE
)
```

Arguments

`path_to_xlsx`	path to the excel file with the management data
`var_sheet`	name of the sheet with the management data template in the excel sheet, default is "Management_template"
`overwrite_year`	logical: if TRUE (default), the *year* will be set to the year extracted from the *date*

Value

a management_df

See Also

- `management_df()` for creating an management_df
- `check_management_df()` to check the integrity of a management_df

Examples

```
#create path
path_to_xlsx_template <- system.file(
"/extdata/SoilManageR_mgmt_data_template_V2.5.xlsx", package = "SoilManageR")

#load management_df
management_df_from_excel(path_to_xlsx_template)
```

N_input *Estimate nitrogen input*

Description

This function estimates the nitrogen (N) input by mineral and organic fertilization into the soil system per year.

Usage

```
N_input(var_MGMT_data, extended.output = FALSE)
```

Arguments

var_MGMT_data a management_df that contains the management information

extended.output
an optional logical value:
- If FALSE, N input values are aggregated by year
- If TRUE, a tibble with all management operations is returned
- Default value is FALSE

Details

The function takes a management_df as input and returns N input values per year in the management_df.

Alternatively, it can return a extensive tibble with all management operations and their N input values.

The functions calculates the N input by organic fertilization with the CN_input_amendments() function. Furthermore, it calculates the livestock intensity (LSU/ha) by deviding the animal derived N by 105kgN/LSU.

Be aware that the function currently neglects N that is fixated by plants (e.g. legumes)

Value

by default, a tibble with N input values (organic N, mineral N and total N) by year is returned. If extended.output = TRUE, a tibble with daily resolution is returned.

See Also

- `calculate_indicators()` to calculate all management indicators for a `management_df`
- `CN_input_amendments()` for the calculation of N inputs from organic amendments

Examples

```
#example that returns annual N input values
N_input(EXAMPLE_data)

#example that returns a tibble with all management operations and their N input
N_input(EXAMPLE_data, extended.output = TRUE)
```

`plant_cover` *Estimate soil cover percentage by plants*

Description

This function estimates the percentage of soil cover based on the number of days since sowing. The parameters used are derived from Mosimann and Rüttimann (2006).

Usage

```
plant_cover(varCrop, varDays = 0)
```

Arguments

`varCrop`	Crop type, must match with crop name in `plant_cover_LUT`
`varDays`	Number of days since sowing of the crop

Details

The function assumes that plant cover unfolds in four phases with different soil cover rates:

- 0 to 10 % of soil cover
- 10 to 50 % of soil cover
- 50 to 75 % of soil cover
- 75 to 100 % of soil cover

Value

percentage of soil cover by plants, value of 0 to 100 %.

References

Mosimann T, Rüttimann M (2006). "Berechnungsgrundlagen zum Fruchtfolgefaktor zentrales Mittelland 2005 im Modell Erosion CH (V2.02)." Terragon, Bubendorf. https://uwe.lu.ch/-/media/UWE/Dokumente/Themen/Bodenschutz/Bodenschutz_Landwirtschaft/dokumentationbodenerosionsschlues terragon2006.pdf.

See Also

- `soil_cover()` to calculate soil coverage by plants and residue for a management_df
- `plant_cover_LUT()` for the data used by the `plant_cover()` function

Examples

```
plant_cover("wheat, winter", 140)
```

---

plant_cover_LUT   *Look-up-table with default values to estimate soil cover by plants*

Description

The dataset is a look-up-table that is used to estimate the soil cover percentage by plant with the function 'SoilManageR::plant_cover()'. The data set is produced from the excel table 'plant_cover_LUT.xlsx' file under '/inst/extdata/'.

Usage

```
plant_cover_LUT
```

Format

A tibble with 28 rows and 7 columns:

Crop Name of the crop

Slope_0_10 Increase of crop cover per day between 0 and 10% soil cover [%/day]

Slope_10_50 Increase of crop cover per day between 10 and 50% soil cover [%/day]

Slope_50_75 Increase of crop cover per day between 50 and 75% soil cover [%/day]

Slope_75_100 Increase of crop cover per day between 75 and 100% soil cover [%/day]

days_30 Number of days it takes to reach 30% soil cover [day]

Comments Source where the information was derived

References

Mosimann T, Rüttimann M (2006). "Berechnungsgrundlagen zum Fruchtfolgefaktor zentrales Mittelland 2005 im Modell Erosion CH (V2.02)." Terragon, Bubendorf. https://uwe.lu.ch/-/media/UWE/Dokumente/Themen/Bodenschutz/Bodenschutz_Landwirtschaft/dokumentationbodenerosionsschlues terragon2006.pdf.

`plant_diversity` *Calculate plant diversity indicators*

Description

Derives three indicators for plant diversity of a crop rotation based on management information (mainly sowing events).

Usage

```
plant_diversity(var_MGMT_data, start_year, end_year)
```

Arguments

`var_MGMT_data`	a 'management_df' with a management history
`start_year`	start year of the cropping sequence of interest
`end_year`	end year of the cropping sequence of interest

Details

For the function to work properly the species (or variety) must be mentioned in the "product" column, and all sown species of mixtures must be represented in a single row of the `management_df` each.

The function calculates the plant diversity index (PDI), the total number of different species, and the Shannon index for all sown species.

The PDI is calculated in the following way inspired by Tiemann et al. (2015):

$$PDI = \overline{S_{year}} * S_{rotation}$$

where $\overline{S_{year}}$ is the average number of sown species per year and $S_{rotation}$ is the total number of different species sown in the full crop rotation or cropping sequence.

The Shannon Index is calculated with the `shannon_index()` function.

Value

a tibble with three indicators for plant diversity

References

Tiemann LK, Grandy AS, Atkinson EE, Marin-Spiotta E, McDaniel MD (2015). "Crop rotational diversity enhances belowground communities and functions in an agroecosystem." *Ecology Letters*, **18**(8), 761-771. doi:10.1111/ele.12453.

See Also

- `calculate_indicators()` to calculate all management indicators for a `management_df`
- `shannon_index()` for more detail on the Shannon index

Examples

```
plant_diversity(EXAMPLE_data,2013,2020)
```

---

plot.soil_cover_tibble *Plotting soil_cover_tibbles*

Description

This function plots objects of the class soil_cover_tibble in a custom format

Usage

```
## S3 method for class 'soil_cover_tibble'
plot(x, ...)
```

Arguments

x	an soil_cover_tibble object
...	arguments to be passed to methods

Value

none, it plots the soil_cover_tibble

Examples

```
data <- soil_cover(EXAMPLE_data, extended.output = TRUE)
plot(data)
```

---

plot_management_df *Plot a management dataframe*

Description

Visual representation of a management_df.

Usage

```
plot_management_df(management_df, title = "Management Timeline")
```

Arguments

`management_df`	a management data frame
`title`	the title of the plot

Details

The colors in the background represent the soil cover by plants (green), and crop residues (brown). "T" indicates tillage event with a STIR value >= 0. "P" indicates a crop protection event. Dotted vertical lines represent sowing events, the sown crop is noted in the lowest quarter at the right side of the line. The values in the middle indicate C inputs into the soil system, either by crops, cover crops or organic amendments. Negative values are withdrawls by residue removal. Note that large management_df can be heavy to compute and the plots may be messy.

Value

a plot

Examples

```
# creates a visual representation of the EXAMPLE data
plot_management_df(EXAMPLE_data)
```

---

productivity_indicator

Calculate average productivity

Description

Estimates estimates the relative yield of a cropping sequence per year. The function takes a `management_df` as input and returns a `relative_yield` value per year in the `management_df`. Alternatively, it can return a tibble with additional information on each crop. The `productivity_indicator()` calculates the relative yields with the `relative_yield()` function.

Usage

```
productivity_indicator(var_MGMT_data, extended.output = FALSE)
```

Arguments

`var_MGMT_data` a management_df that contains the management information

`extended.output`
an optional logical value.

- If FALSE, relative yields are aggregated by year.
- If TRUE, a tibble with relative yield of each crop is returned.
- Default value is FALSE

Value

- By default, a tibble with relative yields by year is returned.
- If extended.output = TRUE, a tibble with additional information is returned.

See Also

- `relative_yield()` for caluclating relative yields
- `calculate_indicators()` for calculating all soil management indicators

Examples

```
#example that returns annual relative yield values
productivity_indicator(EXAMPLE_data)

#example that returns a tibble with additional information
productivity_indicator(EXAMPLE_data, extended.output = TRUE)
```

---

relative_yield *Calculate relative yield*

Description

This function calculates the relative yield of an observed dry matter yield compared to the reference dry matter yield in the Swiss fertilizing guidelines (Sinaj et al. 2017).

Usage

```
relative_yield(var_crop, var_yield)
```

Arguments

var_crop	string with name of crop type, must match with the Crops in the C_input_crops_LUT
var_yield	observed dry matter yield (tDM/ha)

Value

the numeric value for the relative yield

References

Sinaj S, Charles R, Baux A, Dupuis B, Hiltbrunner J, Levy Häner L, Pellet D, Blanchet G, Jeangros B (2017). "Grundlagen für die Düngung landwirtschaftlicher Kulturen in der Schweiz (GRUD): Düngung von Ackerkulturen." *Agrarforschung Schweiz, Spezialpublikation*, **Chapter 8**(6), 1–46. https://ira.agroscope.ch/en-US/Page/Publikation/Index/36799.

See Also

- `productivity_indicator()` to calculate relative yields for a `management_df`
- `C_input_crops_LUT()` for reference yield used in the `relative_yield()` function

Examples

```
relative_yield("wheat, winter", 4.8)
```

---

shannon_index — *Calculate Shannon Index for diversity*

Description

This function calculates the Index of Shannon (1948) for a tibble with different species.

The formula that is used is

$$SI = -\sum_{S=1}^{i}(p_i * ln(p_i))$$

where p_i is the relative abundance of each species (S).

Usage

```
shannon_index(var_tibble)
```

Arguments

`var_tibble`	a tibble with two columns (name of species, count per species), the second column must be called "count"

Value

double of the Shannon Index

References

Shannon CE (1948). "A Mathematical Theory of Communication." *Bell System Technical Journal*, **27**(3), 379-423.

Spellerberg IF, Fedor PJ (2003). "A tribute to Claude Shannon (1916–2001) and a plea for more rigorous use of species richness, species diversity and the 'Shannon–Wiener' Index." *Global ecology and biogeography*, **12**(3), 177–179. doi:10.1046/j.1466822X.2003.00015.x.

See Also

`plant_diversity()` to calculate the `shannon_index()` (and other diversity indices) for a management_df

Examples

```
#create tibble
tibble_example <- tibble::tibble(Plant = c("A","B","C","D","E"), count = c(10,5,8,20,10))

#calculate Shannon Index
shannon_index(tibble_example)  # = 1.505...
```

---

soil_cover — *Estimate soil cover by plants and residues*

Description

Derives the days where soil cover by living plants or resiudes is `>=30%`.

Usage

```
soil_cover(var_MGMT_data, extended.output = FALSE)
```

Arguments

`var_MGMT_data` a `management_df` that contains the management information

`extended.output`
an optional logical value.

- If FALSE, soil cover days are aggregated by year.
- If TRUE, a soil_cover_tibble with daily resolution is returned.
- Defaul value is FALSE

Details

The function takes a `management_df` as input and returns soil cover days per year in the `management_df`. Alternatively, it can return a `soil_cover_tibble` with daily resolution of the estimated soil cover.

The function calculates plant soil cover with the `plant_cover()` function and the soil coverage by residues.

The residue mass is dependent on the residue supply by crops, its decay and its incorporation by tillage operations (Büchi et al. 2016).

Residue supply is estimated with the yield dependent residue C `C_input_straw` provided by the function `C_input_crops()` and a C content of 450 $[mgC/gDM]$. If residues are removed, the removed residue mass is subtracted.

Residue decay is calculated with the formula of Steiner et al. (1999):

$$M_t = M_{t-1} * (1 - k_{decay})$$

Where M_{t-1} is the residue mass of the prior day $[g/m^2]$ and k_{decay} is the daily decay rate that was assumed to be `0.028` g/g, the average decomposition rate of winter wheat straw (Steiner et al. 1999).

Residue incorporation by tillage was estimated with the operation-specific burial coefficient extracted from the RUSLE2 database (USDA-NRCS 2023) that are provided in the look-up-table `STIR_values_LUT`.

Residue mass is translated into percentage of soil cover by the formula of Steiner et al. (2000):

$$cover_{residues} = (1 - e^{-k(M)}) * 100\%$$

Where M is the residue mass $[g/m^2]$ and k is a cover coefficient $[m^2/g]$. k was assumed to be `0.0175` (Steiner et al. 2000).

Value

- By default, a tibble with soil cover days by year is returned.
- If extended.output = TRUE, an object of the class `soil_cover_tibble` with daily resolution is returned.

References

Büchi L, Valsangiacomo A, Burel E, Charles R (2016). "Integrating simulation data from a crop model in the development of an agri-environmental indicator for soil cover in Switzerland." *European Journal of Agronomy*, **76**, 149–159. doi:10.1016/j.eja.2015.11.004.

Steiner JL, Schomberg HH, Unger PW, Cresap J (1999). "Crop residue decomposition in no-tillage small-grain fields." *Soil Science Society of America Journal*, **63**(6), 1817–1824. doi:10.2136/sssaj1999.6361817x.

Steiner JL, Schomberg HH, Unger PW, Cresap J (2000). "Biomass and residue cover relationships of fresh and decomposing small grain residue." *Soil Science Society of America Journal*, **64**(6), 2109–2114. doi:10.2136/sssaj2000.6462109x.

USDA-NRCS (2023). "Revised Universal Soil Loss Equation, Version 2 (RUSLE2), Official NRCS RUSLE2 Program and Database (V 2023-02-24)." USDA-NCRS. `https://fargo.nserl.purdue.edu/rusle2_dataweb/RUSLE2_Index.htm`.

See Also

- `calculate_indicators()` to calculate all management indicators for a `management_df`
- `plant_cover()` for more detail on the plant cover function
- `plot.soil_cover_tibble()` for plotting the `soil_cover_tibble`
- `STIR_values_LUT` for tillage operation specific burial coefficients

Examples

```
#example that returns annual soil cover days by plants and residues
soil_cover(EXAMPLE_data)

#example that returns a soil_cover_tibble
soil_cover(EXAMPLE_data, extended.output = TRUE)
```

STIR *Calculate STIR value*

Description

The function calculates the soil tillage intensity rating (STIR) value of tillage and sowing operations. By default, the STIR() function of SoilManageR operates on SI units (cm and km/h) and not in imperial units (inch, mph) as the original STIR equation (USDA-NRCS 2023). However, the user can specifiy that the input is in imperial units. The function can process custom input, if no such input is provided it assumes default values from the STIR_value_LUT.

Usage

```
STIR(
  device = NA,
  speed = NA,
  speed_unit = "km/h",
  type_modifier = NA,
  depth = NA,
  depth_unit = "cm",
  area_disturbed = NA,
  original.STIR.value = FALSE
)
```

Arguments

device	tillage or sowing implement or operation, must match predefined list, or all necessary inputs (speed, type modidifier, depth, area_disturbed) must be provided
speed	speed of the operation (km/h or mph)
speed_unit	unit of the speed input, must be either NA, "km/h" or "mph". Default value is "km/h"
type_modifier	tillage operation type modifier, must be a value between 0 and 1
depth	depth of the soil that is affected by the operation (cm or inch)
depth_unit	unit of the depth input, must be either NA, "cm" or "inch". Default value is "cm"
area_disturbed	share of the surfave that is disturbed by the operation (0 to 1)
original.STIR.value	logical value, if TRUE, the original STIR value of the operation in the STIR_value_LUT (instead of the calculated STIR value) is returned. Default value is FALSE

Details

The concept of the STIR value was developed within the RUSLE2 framework by the USDA-NRCS (2023). The STIR equation is defined as

$$STIR = (0.5 * speed) * (3.25 * type_{modifier}) * depth * area_{disturbed}$$

where $speed$ and $depth$ are provided in mph and inches(!). For the purpose of this function we assume that `1 inch = 2.54 cm` and `1 mph = 1.609 km/h`. The tillage $type_{modifier}$ is defined to be:

- `1.00` for inversion operation
- `0.80` for mixing and some inversion operations
- `0.70` for mixing operations
- `0.40` for lifting and fracturing operations
- `0.15` for compression operations

In the `STIR_value_LUT` there are more than 400 operations, incl. the original operations from the RUSLE2 software (as of 2023-02-24) and operations defined by the SoilX project.

For further details on the STIR please consider the RUSLE2 website (https://fargo.nserl.purdue.edu/rusle2_dataweb/RUSLE2 or the description of the SoilManageR package

Value

STIR value of the operation

References

USDA-NRCS (2023). "Revised Universal Soil Loss Equation, Version 2 (RUSLE2), Official NRCS RUSLE2 Program and Database (V 2023-02-24)." USDA-NCRS. https://fargo.nserl.purdue.edu/rusle2_dataweb/RUSLE2_Index.htm.

See Also

- `tillage_intensity()` to calculate STIR values for management_df
- `STIR_values_LUT()` for the reference data used by the `STIR()` function

Examples

```
#example without additional information
STIR("plough")

#example with additional information
STIR("rotary_harrow", depth = 15)

#custom example
STIR(speed = 10, type_modifier = 0.8, depth = 15, area_disturbed = 0.45)

#example that returns orginial STIR value
STIR("plough", original.STIR.value = TRUE)

#example that uses imperial units
STIR("plough", depth = 5, depth_unit = "inch", speed = 8, speed_unit = "mph")
```

STIR_values_LUT *Look-up-table with default values for tillage operations*

Description

The dataset is a look-up-table that is used to derive STIR values with the function ‘SoilManageR::STIR()‘ and residue incorperation by tillage operations with the function ‘SoilManageR::soil_cover()‘. The data set is produced from the excel table ‘STIR_value_LUT.xlsx‘ file under ‘/inst/extdata/‘ and was mostly derived from the official RUSLE2 database (USDA-NRCS 2023).

Usage

```
STIR_values_LUT
```

Format

A tibble with 50 rows and 15 columns:

Operation Name of the operation

Speed Average speed of the operation [km/h]

Speed_MIN Min speed of the operation [km/h]

Speed_Max Max speed of the operation [km/h]

Surf_Disturbance Share of the disturbed soil surface [%]

Depth Average depth of the operation [cm]

Depth_MIN Min depth of the operation [cm]

Depth_MAX Max depth of the operation [cm]

TILLAGE_TYPE Type of tillage operation

TILLAGE_TYPE_Modifier Numerical value of the tillage type modifier [0-1]

STIR Soil tillage intensity rating value, based on default values

Diesel_use iesel use of the operation per area [l/ha]

Burial_Coefficient Burial of plant residues on the soil surface [0-1]

Source Source of the values in the table

Description Description of the operation

References

USDA-NRCS (2023). “Revised Universal Soil Loss Equation, Version 2 (RUSLE2), Official NRCS RUSLE2 Program and Database (V 2023-02-24).” USDA-NCRS. https://fargo.nserl.purdue.edu/rusle2_dataweb/RUSLE2_Index.htm.

tillage_intensity *Estimate tillage intensity*

Description

Calculates the soil tillage intensity per year. The function takes a management_df as input and returns a STIR value per year in the management_df. Alternatively, it can return a extensive tibble with each operation and their respective STIR values.

Usage

```
tillage_intensity(var_MGMT_data, extended.output = FALSE)
```

Arguments

var_MGMT_data a management_df that contains the management information

extended.output
an optional logical value. Default value is FALSE

- If FALSE, STIR values are aggregated by year.
- If TRUE, a tibble with all management operations is returned.

Value

by default, a tibble with STIR values by year is returned. If extended.output = TRUE, a tibble with daily resolution is returned.

See Also

- STIR() for the calculation of a STIR value for operations
- STIR_values_LUT() for the reference data used for tillage operations
- calculate_indicators() to calculate all management indicators for a management_df

Examples

```
#example that returns annual STIR values
tillage_intensity(EXAMPLE_data)

#example that returns a tibble with all operations that have a STIR value
tillage_intensity(EXAMPLE_data, extended.output = TRUE)
```

Index