Package 'LandUseQuantifieR'

November 29, 2018

Title This package is an (adapted) R implementation of the land use quantification approach from Hughes et al. 2018

Version 1.0

Description This package is an (adapted) R implementation of the circle diagram approach from Hughes et al. 2018; The package calculates the required area of land for a given number of people, mainly based on their diet. Further information how to use the packge are found in the vignettes. Various nutritional information are collected as data objects within the package. The list of full references are supplied in references.txt and respectively references.bib.

Depends R (>= 3.4)

Imports magrittr,
dplyr,
tibble,
tidyr,
utils

License GPL-3 + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Suggests knitr,
rmarkdown,
ggplot2,
cowplot,
treemapify,
RColorBrewer,
grDevices

VignetteBuilder knitr

R topics documented:

---

animals   *Contains informations of the animal data*

Description

Contains informations of the animal data

Usage

```
animals
```

Format

A data.frame with 14 rows and 10 variables:

animal The type of animal

type The subtype of the animal

calory_requirement The caloric requirements in kcal/kg

meat_yield_factor The factor of the calculation of the animals weight that remains as meat

meat_yield The meat yield in kg; calculated as weight * meat_yield_factor

reference The references for the animal data

herd_composition The composition of the herd of each animal

reference_composition The references of the herd composition

slaughter_factor The slaughter factor

reference_sf The reference for the slaughter factor

animals <- read.table("./animals.csv", header = TRUE, sep = ";", stringsAsFactors = FALSE) usethis::use_data(animals, overwrite = TRUE)

---

animal_products	*Contains informations of the animal products data*

Description

Contains informations of the animal products data

Usage

```
animal_products
```

Format

A data.frame with 13 rows and 4 variables:

animal The type of animal

product The specific animal product

calories The caloric content of the animal product in kcal/kg

animal_products <- read.table("./animal_products.csv", header = TRUE, sep = ";", stringsAsFactors = FALSE) usethis::use_data(animal_products, overwrite = TRUE)

---

arbori_selector   *abori_selector*

Description

Combining, filtering and selecting data sets

Usage

```
arbori_selector(food_df)
```

Arguments

food_df   The consumed food and the proportion in % of the consumed food

Details

This small function called abori_selector adds new variables to one dataset from another dataset, filters the data input and selects the references for the users data input from the packages data

Value

Parameter food_df

---

area_village   *area_village*

Description

Calculates the area of a village

Usage

```
area_village(person_per_hectar = 200, population = 3800)
```

Arguments

person_per_hectar
                     The amount of people per hectar or population density; default 200, based on Jameson et al. 1994

population   The amount of people living in the village; default 3800, based on Jameson et al. 1994

Details

Calculates the area of the village in hectar based on the amount of inhabitants and a measure of their density

Value

A numeric, i.e. the size of a village in hector

---

bronze_production   *Contains informations of the wood requirements for bronze production*

Description

Contains informations of the wood requirements for bronze production

Usage

```
bronze_production
```

Format

A data.frame with 1 row and 4 variables:

energy_req_melt_copper The amount of energy required to melt copper in KJ; default 60000 KJ

energy_req_melt_tin The amount of energy required to melt tin in KJ; default 59000 KJ

energy_req_melt_metals The amount of energy required to melt metal in KJ; default 59000 KJ

weight_melted_bronze_person The amount of weighted metal per person (kg/capita); default 2

bronze_production <- read.table("./bronze_production.csv", header = TRUE, sep = ";", stringsAs-Factors = FALSE) usethis::use_data(bronze_production, overwrite = TRUE)

---

ceramics_production   *Contains informations of the ceramics production*

Description

Contains informations of the ceramics production

Usage

```
ceramics_production
```

Format

A data.frame with 1 row and 5 variables:

vessels_person The number of vessels per person

mean_weight_vessel_kg The mean weight of a vessel in kg

mean_weight_wine_amphora_kg The mean weight of a wine amphora in kg

mean_weight_oil_amphora_kg The mean weight of a oil amphora in kg

ratio_wood_ceramics The ratio of wood to ceramics

ceramics_production <- read.table("./ceramics_production.csv", header = TRUE, sep = ";", stringsAs-Factors = FALSE) usethis::use_data(ceramics_production, overwrite = TRUE)

---

crop_selector   *crop_selector*

Description

Combining, filtering and selecting data sets

Usage

```
crop_selector(food_df)
```

Arguments

food_df   The consumed food and the proportion in % of the consumed food

Details

This small function called abori_selector adds new variables to one dataset from another dataset, filters the data input and selects the references for the users data input from the packages data

Value

Parameter food_df

---

data_checker   *data_checker*

Description

Convert a data frame to a lower case

Usage

```
data_checker(food_df)
```

Arguments

food_df   The consumed food and the proportion in % of the consumed food

Details

This is a small helper function that is called data_checker and it converts the data given in the food_df into lower case

Value

Parameter food_df

---

flax_production   *Contains informations of the flax production*

Description

Contains informations of the flax production

Usage

```
flax_production
```

Format

A data.frame with 1 row and 4 variables:

Productivity The productivity of the flax in kg/ha; default 804

Linen_need_kg The amount of linen needed (kg); default .1

Flax_kg_in_100g_linen The amount of flax (kg) per 100g linen; default 1

PropPop The proportion of the population that uses flax; default .65

flax_production <- read.table("./flax_production.csv", header = TRUE, sep = ",", stringsAsFactors = FALSE) usethis::use_data(flax_production, overwrite = TRUE)

---

game   *Contains informations of the game data*

Description

Contains informations of the game data

Usage

```
game
```

Format

A data.frame with 11 observations and 8 variables:

animal The general category of the game animal: deer, wild boar, birds, fish

type The type of the animal

weight The weight of one individual in kg

meat_yield_factor The factor of the calculation of the animals weight that remains as meat

meat_yield The meat yield in kg; calculated as weight * meat_yield_factor

calories The caloric content in kcal/kg

reference_weight The references of the weight data

reference_calories The references of the caloric data

game <- read.table("./game.csv", header = TRUE, sep = ";", stringsAsFactors = FALSE) usethis::use_data(game, overwrite = TRUE)

---

game_selector   *game_selector*

Description

Combining and selecting data sets

Usage

```
game_selector(food_df)
```

Arguments

food_df   The consumed food and the proportion in % of the consumed food

Details

This small function called abori_selector adds new variables to one dataset from another dataset
and selects the references for the users data input from the packages data

Value

Parameter food_df

---

iron_production   *Contains informations of iron production*

Description

Contains informations of iron production

Usage

```
iron_production
```

Format

A data.frame with 1 row and 3 variables:

iron_person_kg The amount of iron needed per person in kg; default 2; source: Weiberg et al. (in
    prep.)

ratio_charcoal_iron The ratio of charcoal to iron; default 82

ratio_charcoal_wood The ratio of charcoal to wood; default 4

iron_production <- read.table("./iron_production.csv", header = TRUE, sep = ";", stringsAsFactors
= FALSE) usethis::use_data(iron_production, overwrite = TRUE)

---

kcal_req_age   *Kcal requirements of people for different activity levels and sex*

Description

Kcal requirements of people for different activity levels and sex

Usage

```
kcal_req_age
```

Format

A data.frame with 74 rows and 7 variables. # the following code is used to create the data set library(rJava) library(tabulizer) report <- "https://www.cnpp.usda.gov/sites/default/files/usda_food_patterns/EstimatedCal kcal_req <- extract_tables(report, encoding="UTF-8") kcal_req <- kcal_req[[1]] kcal_req <- kcal_req[6:35,] kcal_req <- gsub(pattern = ",", replacement = "", x = kcal_req) kcal_req <- as.data.frame(kcal_req, stringsAsFactors = FALSE) names(kcal_req) <- c("age", "male_sedentary","male_moderately_active","male_active", "female_sedentary","female_moderately_active","female_active") kcal_req[,2:7] <- sapply(kcal_req[,2:7], as.numeric) kcal_req_age <- kcal_req[18:29,] dplyr::mutate(from = lapply(strsplit(kcal_req[18:29,1], split = "−"), function(x) x[1]), to = lapply(strsplit(kcal_req[18:29,1], split = "−"), function(x) x[2])) dplyr::rowwise() dplyr::do(data.frame(age = as.character(seq(.$from, .$to, by = 1)), male_sedentary = .$male_sedentary, male_moderately_active = .$male_moderately_active, male_active = .$male_active, female_sedentary = .$female_sedentary, female_moderately_active = .$female_moderately_active, female_active = .$female_active)) dplyr::bind_rows(kcal_req[1:17,], .) usethis::use_data(kcal_req_age, overwrite = TRUE)

Source

https://www.cnpp.usda.gov/sites/default/files/usda_food_patterns/EstimatedCalorieNeedsPerDayTabl pdf

---

kcal_req_human   *Contains informations of the calory requirements for humans*

Description

Contains informations of the calory requirements for humans

Usage

```
kcal_req_human
```

Format

A data.frame with 80 rows and 6 variables:

pal The Total Energy Expenditure for 24 hours expressed as a multiple of Basal Metabolic Rate. Calculated as TEE/BMR for 24 hours (cf.FAO2001).

age_category The age categorized as a human younger than 19 years or older than 19 years

sex The gender

age The age expressed as category

kcal_day The caloric requirement as kcal/day

reference The sources of literature for the required kcal data

kcal_req_human <- read.table("./calory_requirement_human.csv", header = TRUE, sep = ";", stringsAs-Factors = FALSE) usethis::use_data(kcal_req_human, overwrite = TRUE)

---

nd_herd_medium   *nd_herd_medium*

Description

Calculate herd characteristics when nothing specific is known

Usage

```
nd_herd_medium(input_animals, kcal_req_village_year, proportion)
```

Arguments

input_animals   The animals input of the data.frame

kcal_req_village_year
                 The kcal requirements of the village per year

proportion   The proportion of the consumed foodtype in %

Details

This is a small helper function that is called when there are no specific information or empirical evidences to calculate the village meat requirements. To generate more realistic results, the number of required animals is rounded up to the nearest whole number. The function imports the data on animals and animal_products and takes the mean of those values; besides, fodder-plants are imported and all those plants that are also crops are removed from the dataset

Value

A data.frame

---

nd_herd_simple   *nd_herd_simple*

Description

Calculate herd characteristics when nothing specific is known

Usage

```
nd_herd_simple(kcal_req_village_year, proportion)
```

Arguments

kcal_req_village_year
                The kcal requirements of the village per year

proportion   The proportion of the diet

Details

This is a small helper function that is called when there are no specific information or empirical evidences to calculate the village meat requirements. The function imports the data on animals and animal_products and takes the mean of those values; besides, fodder-plants are imported and all those plants that are also crops are removed from the dataset.

Value

A data_frame

---

pasture_selector   *pasture_selector*

Description

Combining, filtering, selecting and renaming data sets

Usage

```
pasture_selector(food_df)
```

Arguments

food_df   The consumed food and the proportion in % of the consumed food

Details

This small function called abori_selector adds new variables to one dataset from another dataset, filters the data input and selects the references for the users data input from the packages data

Value

Parameter food_df

plants	*Contains informations of the plant data*

Description

Contains informations of the plant data

Usage

```
plants
```

Format

A data.frame with 35 rows and 9 variables:

Type The type of plant

Species The specific plant species

productivity The productivity of the plant in kg/ha

produc_units The units of the productivity

calories The caloric content of the plant in kcal/kg

calories_units The units of the calories

yield_units The units of the yield

reference_calories The references of calories data

reference_productivity The references of productivity data

plants <- read.table("./plants.csv", header = TRUE, sep = ";", stringsAsFactors = FALSE) usethis::use_data(plants, overwrite = TRUE)

setup_bronze_complex	*setup_bronze_complex*

Description

Setup of the bronze production

Usage

```
setup_bronze_complex(population, forest_type, ...)
```

Arguments

population	The number of people living in the village
forest_type	The type of the forest; choose between "oak_mixed_forest" and "olive_mixed_forest"; default "oak_mixed_forest"
...	Further arguments passed to other functions
energy_req_melt_copper	
	The amount of energy required to melt copper; default 60000 KJ
energy_req_melt_tin	
	The amount of energy required to melt tin; default 59000 KJ
energy_req_melt_metals	
	The amount of energy required to melt metal; default 59000 KJ
weight_melted_bronze_person	
	The amount of weighted metal per person (kg/capita); default 2

Details

This function sets up the amount of wood required for bronze production

Value

A tibble

---

setup_ceramics_complex

setup_ceramics_complex

Description

Setup of the ceramics production

Usage

```
setup_ceramics_complex(population, forest_type, food_df, households,
  olive_oil_yield, ...)
```

Arguments

population	The number of people living in the village
forest_type	The type of the forest; choose between "oak_mixed_forest" and "olive_mixed_forest"; default "oak_mixed_forest"
food_df	The product and the proportion of the consumed food
households	the of households
...	Further arguments passed to other functions
vessels_person	The number of vessels per person; default 20;
mean_weight_vessel_kg	
	The mean weight of a vessel in kg; default 1
mean_weight_wine_amphora_kg	
	The mean weight of wine amphora in kg; default 10

mean_weight_oil_amphora_kg

> The mean weight of olive oil amphora in kg; default 10

ratio_wood_ceramics

> The ratio of wood to ceramics; default .907184; source: Kahn 2008

Details

This function sets up the required wood for ceramic production

Value

A tibble

---

setup_diet_arbori_gather_medium

setup_diet_arbori_gather_medium

Description

Setup the general arboriculture and gathered plants diet characteristics

Usage

```
setup_diet_arbori_gather_medium(population, food_df, kcal_req_human_year,
  loss = 0.5)
```

Arguments

population   The number of people living in a village

food_df   The consumed food and the proportion in % of the consumed food

kcal_req_human_year

> The per capita kcal requirement of the year; if nothing is supplied, it is calculated based on setup_population function

loss   The loss due to storage, waste loss or saving for next years planting; default .5

Details

This function creates a data.frame of the diet characteristics of a village that is used in subsequent calculations;

Value

A list with a numeric (required hectar to grow plants to sustain the population) and a data.frame that supplies the individual values

setup_diet_crops_medium

setup_diet_crops_medium

Description

Setup the crop and legume diet characteristics

Usage

```
setup_diet_crops_medium(population, food_df, kcal_req_human_year,
  loss = 0.5)
```

Arguments

population	The number of people living in a village
food_df	The consumed food and the proportion in % of the consumed food
kcal_req_human_year	
	The per capita kcal requirement of the year; if nothing is supplied, it is calculated based on setup_population function
loss	The loss due to storage, waste loss or saving for next years planting; default .5

Details

This function creates a data.frame of the diet characteristics of a village that is used in subsequent calculations;

Value

A list with a numeric (required hectar to grow plants to sustain the population) and a data.frame that supplies the individual values

setup_diet_game_medium

setup_diet_game_medium

Description

Setup the general game diet characteristics

Usage

```
setup_diet_game_medium(population, food_df, kcal_req_human_year)
```

Arguments

population	The number of people living in a village
food_df	The consumed food and the proportion in % of the consumed food
kcal_req_human_year	
	The per capita kcal requirement of the year; if nothing is supplied, it is calculated based on setup_population function

Details

This function creates a data.frame of the diet characteristics of a village that is used in subsequent calculations. To generate more realistic results, the number of required animals is rounded up to the nearest whole number;

Value

A list with a numeric (required hectar to grow plants to sustain the population) and a data.frame that supplies the individual values

---

setup_diet_meat   *setup_diet_meat*

Description

Setup the general meat diet characteristics

Usage

```
setup_diet_meat(population = 3800, propdiet = "nd_mixed",
  proportion = 0.1, kcal_req = "simplified")
```

Arguments

population	The number of people living in a village; default 3800
propdiet	The proportions of the consumed food; currently only "nd_mixed" is implemented what is representative for no data and a mixed diet that consists of crops and fruits in equal amount (1/2); it takes the values supplied in the datasets "plants" and uses the mean values to return productivity (kcal/ha) and calory values
proportion	The proportion of meat of the population's diet; default: .1
kcal_req	The per capita kcal requirement; if is nothing is supplied, it is calculated based on setup_population function; default: "simplified"

Details

This function creates a data.frame of the diet characteristics of a village that is used in subsequent calculations;

Value

A list with a numeric (required hectar for pasture to sustain the animal herd) and a data_frame that supplies the individual values

---

```
setup_diet_pasture_medium
```
setup_diet_pasture_medium

Description

Setup the general meat diet characteristics

Usage

```
setup_diet_pasture_medium(population, food_df, kcal_req_human_year)
```

Arguments

population	The number of people living in a village
food_df	The consumed food and the proportion in % of the consumed food
kcal_req_human_year	
	The per capita kcal requirement of the year; if nothing is supplied, it is calculated based on setup_population function

Details

This function creates a data.frame of the diet characteristics of a village that is used in subsequent calculations;

Value

A list with a numeric (required hectar for pasture to sustain the animal herd) and a data.frame that supplies the individual values

---

```
setup_diet_veg
```
setup_diet_veg

Description

Setup the general vegetarian diet characteristics

Usage

```
setup_diet_veg(population, consumed_plants = NULL, proportion = NULL,
  kcal_req_human_year = NULL, loss = 0.5)
```

Arguments

population	The number of people living in a village; default 3800
consumed_plants	
	The kind of consumed plants
proportion	The proportion of diet in numbers [only used when propdiet is "nd_mixed"]; default is c(.5, .4) assuming that the diet of the population is a mix of crops, fruits, and meat (calculated using setup_diet_meat)
kcal_req_human_year	
	The kcal a human requires per year; default NULL
loss	The loss due to storage, waste loss or saving for next years planting; default: .5

Details

This function creates a data.frame of the diet characteristics of a village that is used in subsequent calculations;

Value

A list with a numeric (required hectar to grow plants to sustain the population) and a data_frame that supplies the individual values

---

setup_flax_complex	*setup_flax_complex*

Description

Setup the kcal etc. values for flax

Usage

```
setup_flax_complex(population)
```

Arguments

population	The number of people living in the village
productivity	The productivity of the flax in kg/ha; default 804
Linen_need_kg	The amount of linen needed (kg); default .1
Flax_kg_in_100g_linen	
	The amount of flax (kg) per 100g linen; default 1
PropPop	The proportion of the population that uses flax; defaul .65
clothes_req	Whether clothes are required; default NA

Details

This function sets up the kcal etc. values for flax in form of a tibble

Value

A tibble

---

setup_iron_complex *setup_iron_complex*

Description

Setup of the iron production

Usage

```
setup_iron_complex(population, forest_type, ...)
```

Arguments

population   The number of people living in the village

forest_type   The type of the forest; choose between "oak_mixed_forest" and "olive_mixed_forest";
                 default "oak_mixed_forest"

...   Further arguments passed to other functions

iron_person_kg   The amount of iron per person as numeric; default 2; source: Weiberg et al. (in
                 prep.)

ratio_charcoal_iron
                 The ratio of charcoal to iron as numeric; default 82

ratio_charcoal_wood
                 The ration of charcoal to wood as numeric; default 4

Details

This function sets up the amount of wood required for iron production

Value

A tibble with the characteristics of Iron

---

setup_olive_oil_complex
setup_olive_oil_complex

Description

Setup of the kcal etc. values for olive_oil

Usage

```
setup_olive_oil_complex(population, households, olive_oil_storage = 0.25,
  rich_people_factor = 0.66, ...)
```

Arguments

population	The number of people living in a village
households	The number of households; default: population / 5, where 5 is the number of people living in a household
...	Further arguments passed to other functions

Details

This function sets up the kcal etc. values for olive_oil in form of a tibble

Value

A tibble

---

setup_population_simple

setup_population_simple

Description

Calculate the energy requirements of a village

Usage

```
setup_population_simple(population, mode = "simplified",
  kcal_requirement = NULL)
```

Arguments

population	The number of people living in a village
mode	The calculation mode; currently only "simplified" (the default) and "medium" is implemented; further modes will be added soon and require further input data simplified approach takes the mean of empirical values of daily kcal requirements for active females and males in the age between 10 and 50 (source for daily requirements: http://www.cnpp.usda.gov/sites/default/files/usda_food_patterns/EstimatedCalorieNeedsPerDayTable.pdf)
kcal_requirement	
	optional; default NULL

Details

Calculate the energy requirements of a village based on a given population size

This functions is a wrapper for more comprehensive functions calculating the nutritional (kcal) requirements of a village; These are not implemented yet; Note that more comprehensive approaches, e.g. considering a death-table of a village, require a lot more data that is often not available

Value

A list with the elements: VillagePeople, KcalHuman, KcalHumanYear, KcalVillageYear

---

setup_wine_complex *setup_wine_complex*

Description

Setup of the kcal etc. values for wine

Usage

```
setup_wine_complex(population, food_df, ...)
```

Arguments

population	The number of people living in a village
food_df	The consumed food and the proportion in % of the consumed food
...	Further arguments passed to other functions

Details

This function sets up the kcal etc. values for wine in form of a tibble

Value

A tibble

---

setup_wood_consumption

wood_consumption

Description

Setup of the wood consumption

Usage

```
setup_wood_consumption(population, forest_type, ...)
```

Arguments

population	The number of people living in the village; default: 3800;
forest_type	The type of the forest; choose between "oak_mixed_forest" and "olive_mixed_forest"; default "oak_mixed_forest"
...	Further arguments passed to other functions
consumption_energy_person_a_GJ	
	The consumption of energy per person in GJ; default 8 (FAO 2013)
consumption_fuelwood_person_a_kg	
	The fuelwood consumption per person; default 551
productivity	The productivity of the mentioned forest_type in t/ha. default: 12.9 (source average wood productivity in agroforestry; (FAO 2013))
calorific_value	
	The calorific value of wood in MJ/kg; default: 18.62 (Olive wood (FAO 2013))

Details

This function calculates the wood consumption of the village

Value

A tibble

---

spelling_checker   *spelling_checker*

Description

checking the spelling of the animal data input

Usage

```
spelling_checker(animal_name)
```

Arguments

animal_name   The name of the animal

Details

This is a small helper function that is called spelling_checker and it allows the user to use plural or singular for data input

Value

Parameter animal_name

---

summarize_areas_complex

summarize_areas_complex

Description

Summarize_areas_complex

Usage

```
summarize_areas_complex(population = 3800, kcal_requirement = NULL,
  household_people = 5, food_df, forest_type = "oak_mixed_forest",
  bronze = FALSE, iron = FALSE, ceramics = FALSE,
  olive_oil = FALSE, flax = FALSE, ...)
```

Arguments

`population`	The number of people living in a village; default 3800
`kcal_requirement`	
	The kcal a human requires per day; default NULL
`household_people`	
	The number of people per household; default 5
`food_df`	The consumed food and the proportion in % of the consumed food
`forest_type`	The type of the forest; choose between "oak_mixed_forest" and "olive_mixed_forest"; default "oak_mixed_forest"
`bronze`	Whether the population melted/used bronze; default FALSE
`iron`	Whether the population melted/used iron; default FALSE
`ceramics`	Whether the population produced/used ceramic; default FALSE
`olive_oil`	Whether the population produced olive oil; default FALSE
`flax`	Whether the population needs flax; default FALSE
`...`	Further arguments passed to other functions

Details

Bring all function calculations together

Value

A tibble

`summarize_areas_medium`

summarize_areas_medium

Description

Summarize_areas

Usage

```
summarize_areas_medium(population = 3800, kcal_requirement = NULL,
  household_people = 5, food_df, forest_type = "oak_mixed_forest",
  wood = setup_wood_consumption(population = population, forest_type,
  ...), ...)
```

Arguments

`population`	The number of people living in a village; default 3800
`kcal_requirement`	
	The kcal a human requires per day; default NULL
`household_people`	
	The number of people per household; default 5
`food_df`	The consumed food and the proportion in % of the consumed food
`forest_type`	The type of the forest; choose between "oak_mixed_forest" and "olive_mixed_forest"; default "oak_mixed_forest"
`...`	Further arguments passed to other functions

Details

Bring all function calculations together

Value

A tibble

---

summarize_areas_simple

summarize_areas_simple

Description

Summarize_areas

Usage

```
summarize_areas_simple(population = 3800, kcal_requirement = NULL,
  household_people = 5, consumed_plants = NULL, proportion = NULL,
  forest_type = "oak_mixed_forest",
  wood = setup_wood_consumption(population = population, forest_type,
  ...), pasture = setup_diet_meat(population = population), ...)
```

Arguments

population	The number of people living in a village; default 3800
kcal_requirement	
	The kcal a human requires per day; default NULL
household_people	
	The number of people per household; default 5
consumed_plants	
	The kind of consumed plants
proportion	The percentage amount of each consumed_plants; default NULL
forest_type	The type of the forest; choose between "oak_mixed_forest" and "olive_mixed_forest"; default "oak_mixed_forest"
wood	The required area of woodland to produce ceramics, iron etc.
pasture	The required area for pasture

Details

Bring all function calculations together

Value

A tibble

---

wood_cons *Contains informations of the wood consumption*

Description

Contains informations of the wood consumption

Usage

```
wood_cons
```

Format

A data.frame with 1 row and 2 variables:

consumption_energy_person_a_GJ The amount of energy one person consumes per year by burning wood in GJ

consumption_fuelwood_person_a_kg The amount of fuelwood a person uses in kg

wood_cons <- read.table("./wood_cons.csv", header = TRUE, sep = ";", stringsAsFactors = FALSE)
usethis::use_data(wood_cons, overwrite = TRUE)

---

wood_prod *Contains informations of the wood productivity*

Description

Contains informations of the wood productivity

Usage

```
wood_prod
```

Format

A data.frame with 2 row and 4 variables:

forest_type The Type of wood that is used by the people/villagers

productivity_in_cbm_per_ha_per_a The productivity of the forest/woodland in cbm per ha and year

calorific_value_in_MJ_per_kg The calorific value of the wood in MJ per kg

references The references of the data

wood_prod <- read.table("./wood_prod.csv", header = TRUE, sep = ";", stringsAsFactors = FALSE)
usethis::use_data(wood_prod, overwrite = TRUE)

Index