Package ‘SoilTaxonomy’

August 5, 2021

Title A System of Soil Classification for Making and Interpreting Soil Surveys

Description Taxonomic dictionaries, formative element lists, and functions related to the maintenance, development and application of U.S. Soil Taxonomy. Data and functionality are based on official U.S. Department of Agriculture sources including the latest edition of the Keys to Soil Taxonomy. Descriptions and metadata are obtained from the National Soil Information System or Soil Survey Geographic databases. Other sources are referenced in the data documentation. Provides tools for understanding and interacting with concepts in the U.S. Soil Taxonomic System. Most of the current utilities are for working with taxonomic concepts at the “higher” taxonomic levels: Order, Suborder, Great Group, and Subgroup.

Version 0.1.4

Maintainer Andrew Brown <andrew.g.brown@usda.gov>

Depends R (>= 3.5)

License GPL (>= 3)

Encoding UTF-8

LazyLoad yes

Repository CRAN

URL https://github.com/ncss-tech/SoilTaxonomy

BugReports https://github.com/ncss-tech/SoilTaxonomy/issues

Imports stringr

Suggests testthat, covr, knitr, rmarkdown, markdown

RooxygenNote 7.1.1

VignetteBuilder knitr

NeedsCompilation no

Author Dylan Beaudette [aut], Andrew Brown [aut, cre]

Date/Publication 2021-08-05 21:40:02 UTC
\section*{R topics documented:}

\begin{tabular}{ll}
  code_to_level & .......................................................... 2 \\
  decompose_taxon_code & .................................................. 3 \\
  explainST & .......................................................... 4 \\
  FormativeElements & .................................................. 4 \\
  getChildTaxa & .................................................. 5 \\
  getLastChildTaxon & .................................................. 6 \\
  getParrentTaxa & .................................................. 7 \\
  getTaxonAtLevel & .................................................. 8 \\
  get_ST_features & .................................................. 9 \\
  isValidST & .................................................. 10 \\
  level_to_taxon & .................................................. 11 \\
  parse_family & .................................................. 11 \\
  preceding_taxon_codes & ............................................. 12 \\
  relative_taxon_code_position & ........................................... 13 \\
  ST & .................................................. 14 \\
  ST_family_classes & .................................................. 14 \\
  ST_features & .................................................. 15 \\
  ST_formative_elements & .................................................. 15 \\
  ST_higher_taxa_codes_12th & ........................................... 16 \\
  taxon_code_to_taxon & .................................................. 16 \\
  taxon_to_level & .................................................. 17 \\
  taxon_to_taxon_code & .................................................. 18 \\
\end{tabular}

\section*{Index}

\begin{tabular}{ll}
  code_to_level & 2 \\
\end{tabular}

\begin{tabular}{ll}
  Determine taxonomic level of a taxonomic letter code & code_to_level \\
\end{tabular}

\section*{Description}

Determine taxonomic level of a taxonomic letter code

\section*{Usage}

\texttt{code\_to\_level(code)}

\section*{Arguments}

\begin{itemize}
  \item \texttt{code} A character vector of taxon codes (case sensitive)
\end{itemize}

\section*{Value}

A character vector containing "order", "suborder", "greatgroup" or "subgroup"
decompose_taxon_code

Examples

# order level code (1 character)
code_to_level("B")

# subgroup level code (4 characters)
code_to_level("ABCD")

# subgroup level code (5 characters, 4 uppercase + 1 lowercase)
code_to_level("IFFZh")

--

decompose_taxon_code  Decompose taxon letter codes

Description

Find all codes that logically comprise the specified codes. For instance, code "ABC" ("Anhyturbels") returns "A" ("Gelisols"), "AB" ("Turbels"), "ABC" ("Anhyturbels"). Use in conjunction with a lookup table that maps Order, Suborder, Great Group and Subgroup taxa to their codes (see taxon_code_to_taxon and taxon_to_taxon_code).

Usage

decompose_taxon_code(codes)

Arguments

codes  A character vector of taxon codes to "decompose" – case sensitive

Details

Accounts for Keys that run out of capital letters (more than 26 subgroups) and use lowercase letters for a unique subdivision within the "fourth character position."

Value

A list with equal length to input vector; one character vector per element

See Also

preceding_taxon_codes, taxon_code_to_taxon, taxon_to_taxon_code

Examples

decompose_taxon_code(c("ABC", "ABCDe", "BCDEF"))
explainST

**FormativeElements**

---

**Explanation**

Explain a taxon name using formative elements

**Usage**

```
explainST(x, format = c("text", "html"), viewer = TRUE)
```

**Arguments**

- **x**: a Subgroup, Great Group, Suborder or Order-level taxonomic name; matching is exact and case-insensitive
- **format**: output format: 'text' | 'html'
- **viewer**: show format = 'html' output in browser? default: TRUE

**Value**

a block of text, suitable for display in fixed-width font

**Examples**

```r
cat(explainST("ids"), "\n\n")        # -ids (order suffix)
cat(explainST("aridisols"), "\n\n") # Aridisols (order name)
cat(explainST("argids"), "\n\n")   # Arg- (suborder)
cat(explainST("haplargids"), "\n\n") # Hap- (great group)
cat(explainST("typic haplargids"), "\n\n") # Typic (subgroup)
```

---

**FormativeElements**

Identify formative elements in taxon names at Soil Order, Suborder, Great Group or Subgroup level

---

**Description**

Identify formative elements in taxon names at Soil Order, Suborder, Great Group or Subgroup level
Usage

FormativeElements(x, level = c("order", "suborder", "greatgroup", "subgroup"))

OrderFormativeElements(x)

SubOrderFormativeElements(x)

GreatGroupFormativeElements(x)

SubGroupFormativeElements(x)

Arguments

x
A character vector containing subgroup-level taxonomic names

level
one of c("order", "suborder", "greatgroup", "subgroup")

Value

A list containing $defs: a data.frame containing taxonomic elements, derivations, connotations and links. And $char.index: a numeric denoting the position where the formative element occurs in the search text x

Author(s)

D.E. Beaudette, A.G. Brown

Examples

FormativeElements("acrudoxic plinthic kandiudults", level = "subgroup")
SubGroupFormativeElements("acrudoxic plinthic kandiudults")

FormativeElements("acrudoxic plinthic kandiudults", level = "greatgroup")
GreatGroupFormativeElements("acrudoxic plinthic kandiudults")

FormativeElements("acrudoxic plinthic kandiudults", level = "suborder")
SubOrderFormativeElements("acrudoxic plinthic kandiudults")

FormativeElements("acrudoxic plinthic kandiudults", level = "order")
OrderFormativeElements("acrudoxic plinthic kandiudults")

---

getChildTaxa

Get the lower (child) taxa for a taxon name or code

Description

Get the lower (child) taxa for a taxon name or code
Usage

getchildTaxa(
  taxon = NULL,
  code = NULL,
  convert = TRUE,
  level = c("order", "suborder", "greatgroup", "subgroup")
)

Arguments

taxon A character vector of taxa (case-insensitive)

code A character vector of taxon codes (case sensitive)

convert Convert results from taxon codes to taxon names? Default: TRUE

level Filter results to specific level? Default: "order","suborder","greatgroup","subgroup"

Value

A named list, where names are taxon codes and values are character vectors representing parent taxa

Examples

# suborder children of "Mollisols"
getchildTaxa("Mollisols", level = "suborder")

# get all siblings within a great group, given a subgroup
getchildTaxa(getTaxonAtLevel("Ultic Haploxeralfs", "greatgroup"))

getLastChildTaxon

Get last child taxon in Keys at specified taxonomic level

Description

Get last child taxon in Keys at specified taxonomic level

Usage

getLastChildTaxon(level = c("order", "suborder", "greatgroup"))

Arguments

level Get child taxa from keys at specified level. One of: "order", "suborder", "greatgroup"
Value

A data.frame containing key (parent key), taxon (last taxon name), code (letter code), position (relative taxon position)

Examples

# get last taxa in suborder-level keys
x <- getLastChildTaxon(level = "suborder")

# proportion of keys where last taxon has "Hap" formative element
prop.table(table(grepl("^Hap", x$taxon)))

getParentTaxa

Get the higher (parent) taxa for a taxon name or code

Description

Must specify either taxon or code. taxon is used if both are specified.

Usage

getParentTaxa(
  taxon = NULL,
  code = NULL,
  convert = TRUE,
  level = c("order", "suborder", "greatgroup", "subgroup")
)

Arguments

taxon A character vector of taxa (case-insensitive)

code A character vector of taxon codes (case sensitive)

convert Convert results from taxon codes to taxon names? Default: TRUE

level level Filter results to specific level? Default: "order","suborder","greatgroup","subgroup"

Value

A named list, where names are taxon codes and values are character vectors representing parent taxa
getTaxonAtLevel

Examples

getParentTaxa("ultic haploxeralfs")
getParentTaxa(code = c("ABCD", "DABC"))
getParentTaxa("folists", convert = FALSE)

getTaxonAtLevel(x, level = c("order", "suborder", "greatgroup", "subgroup"))

Arguments

x A character vector containing subgroup-level taxonomic names
level one of c("order", "suborder", "greatgroup", "subgroup")

Value

A named character vector of taxa at specified level, where names are the internal Soil Taxonomy letter codes.

Examples

# default gets the soil order
getTaxonAtLevel(c("typic haplargids", "typic glacistels")) #, level = "order")

# specify alternate levels
getTaxonAtLevel("humic haploxerands", level = "greatgroup")

# can't get subgroup (child) from great group (parent)
getTaxonAtLevel("udifolists", level = "subgroup")

# but can do parents of children
getTaxonAtLevel("udifolists", level = "suborder")
get_ST_features

Get diagnostic horizons, characteristics and features

Description

All parameters to this function are optional (default NULL). If specified, they are used as filters.

Usage

get_ST_features(
  group = NULL,
  chapter = NULL,
  name = NULL,
  page = NULL,
  multiline_sep = "\\n"
)

Arguments

chapter: optional filtering vector; only chapter 3 currently
name: optional filtering vector; these are the "names" of features used in headers
page: optional filtering vector; page number (12th Edition Keys to Soil Taxonomy)
multiline_sep: default "\\n" returns criteria column as a character vector concatenated with "\\n". Use NULL for list

Details

This is a wrapper method around the package data set ST_features.

Value

a data.frame

See Also

ST_features

Examples

# get all features
str(get_ST_features())

# get features in chapter 3
str(get_ST_features(chapter = 3))
isValidST

Check for valid taxonomic level (Order, Suborder, Great Group, Subgroup)

Description

Checks `needle` for matches against a single level of Soil Taxonomy hierarchy: order, suborder, greatgroup, subgroup. Matches are case-insensitive.

Usage

`isValidST(needle, level = c("order", "suborder", "greatgroup", "subgroup"))`

Arguments

- `needle` vector of taxa
- `level` single level of Soil Taxonomy hierarchy; one of: "order", "suborder", "greatgroup", "subgroup"

Value

logical vector, same length as `needle`

Examples

`isValidST('typic haploxeralfs', level = 'subgroup')`
level_to_taxon

level_to_taxon  Get all taxa at specified level

Description
Convenience method for getting taxa from ST_unique_list

Usage
level_to_taxon(level = c("order", "suborder", "greatgroup", "subgroup"))

Arguments
level  character. One or more of "order", "suborder", "greatgroup", "subgroup"

Value
A character vector of taxa at the specified level

Examples

# get all order and suborder level taxa
level_to_taxon(level = c("order", "suborder"))

parse_family

parse_family  Parse components of a family-level taxon name

Description
Parse components of a family-level taxon name

Usage
parse_family(family)

Arguments
family  a character vector containing taxonomic families, e.g. "fine-loamy,mixed,semiactive,mesic ultic haploxeralfs"

Value
a data.frame containing column names: "family" (input), "subgroup" (parsed taxonomic subgroup), "subgroup_code" (letter code for subgroup), "class_string" (comma-separated family classes), "classes_split" (split class_string vector stored as list column)
### Examples

```r
families <- c("fine, kaolinitic, thermic typic kanhapludults", 
               "fine-loamy, mixed, semiactive, mesic ultic haploxeralfs", 
               "euic, thermic typic haplosapris", 
               "coarse-loamy, mixed, active, mesic aquic dystrudepts", 
               "fine, kaolinitic, thermic typic kanhapludults", 
               "fine-loamy, mixed, semiactive, mesic ultic haploxeralfs", 
               "euic, thermic typic haplosapris", 
               "coarse-loamy, mixed, active, mesic aquic dystrudepts")

# inspect parsed list result
str(parse_family(families))
```

### preceding_taxon_codes  Get taxon codes of preceding taxa

#### Description

Find all codes that logically precede the specified codes. For instance, code "ABC" ("Anhyturbels") returns "AA" ("Histels") "ABA" ("Histoturbels") and "ABB" ("Aquiturbels"). Use in conjunction with a lookup table that maps Order, Suborder, Great Group and Subgroup taxa to their codes (see taxon_code_to_taxon and taxon_to_taxon_code).

#### Usage

```r
preceding_taxon_codes(codes)
```

#### Arguments

- **codes**  A character vector of codes to calculate preceding codes for

#### Details

Accounts for Keys that run out of capital letters (more than 26 subgroups) and use lowercase letters for a unique subdivision within the "fourth character position."

#### Value

A list with equal length to input vector; one character vector per element

#### See Also

- decompose_taxon_code
- taxon_code_to_taxon
- taxon_to_taxon_code
**relative_taxon_code_position**

**Examples**

```r
preceding_taxon_codes(c("ABCDe", "BCDEf"))
```

**Description**

The relative position of a taxon is \([\text{number of preceding Key steps}] + 1\), or NA if it does not exist in the lookup table.

**Usage**

```r
relative_taxon_code_position(code)
```

**Arguments**

- **code**
  
  A character vector of taxon codes to determine the relative position of.

**Value**

A numeric vector with the relative position of each code with respect to their individual Keys.

**Examples**

```
# "ABCD" -> "Gypsic Anhyturbels", relative position 7
# "WXYZa" does not exist, theoretical position is 97
# "BAD" -> "Udifolists", relative position is 5

relative_taxon_code_position(c("ABCD", "WXYZa", "BAD"))
```

```
# [1] 7 NA 5
```
ST  

Soil Taxonomy Hierarchy  

Description

The first 4 levels of the US Soil Taxonomy hierarchy (soil order, suborder, greatgroup, subgroup), presented as a data.frame (denormalized) and a list of unique taxa.

Usage

data(ST)

Format

An object of class data.frame with 2665 rows and 4 columns.

References


ST_family_classes  

Family-level Classes for Soil Taxonomy  

Description

A database of family-level class names for Soil Taxonomy.

Usage

data(ST_family_classes)

Format

An object of class data.frame with 198 rows and 7 columns.

References

**ST_features**

*Epipedons, Diagnostic Horizons, Characteristics and Features in Soil Taxonomy*

**Description**


**Usage**

data(ST_features)

**Format**

An object of class data.frame with 84 rows and 6 columns.

**References**


---

**ST_formative_elements**

*Formative Elements used by Soil Taxonomy*

**Description**

A database of formative elements used by the first 4 levels of US Soil Taxonomy hierarchy (soil order, suborder, greatgroup, subgroup).

**Usage**

data(ST_formative_elements)

**Format**

An object of class list of length 4.

**References**

taxon_code_to_taxon

Description
Convert taxon code to taxon name

Usage

taxon_code_to_taxon(code)

Arguments

code A character vector of Taxon Codes
**taxon_to_level**

**Value**
A character vector of matching Taxon Names

**See Also**
decompose_taxon_code, preceding_taxon_codes, taxon_to_taxon_code

**Examples**

taxon_code_to_taxon(c("ABC", "XYZ", "DAB", NA))

taxon_to_level(c("gelisols", NA, "foo", "typic folistels", "folistels"))

---

**Description**
Taxa that resolve to a subgroup level taxon and contain a comma "," are assumed to be "family"-level.

**Usage**
taxon_to_level(taxon)

**Arguments**
taxon character vector of taxon names at Order, Suborder, Great Group or Subgroup level.

**Value**
character of taxonomic hierarchy levels (such as "order", "suborder", "greatgroup", "subgroup", "family") for each element of input vector.

**Examples**

# get the taxonomic levels for various taxa
taxon_to_level(c("gelisols", NA, "foo", "typic folistels", "folistels"))
taxon_to_taxon_code  

Convert taxon name to taxon code

Description

Convert taxon name to taxon code

Usage

\[
\text{taxon_to_taxon_code}(\text{taxon})
\]

Arguments

\[
\text{taxon} \quad \text{A character vector of taxon names, case insensitive}
\]

Value

A character vector of matching taxon codes

See Also

decompose_taxon_code, preceding_taxon_codes, taxon_code_to_taxon

Examples

\[
\text{taxon_to_taxon_code(c("Anhyturbels", "foo", "Cryaquands", NA))}
\]
Index

* datasets
  ST, 14
  ST_family_classes, 14
  ST_features, 15
  ST_formative_elements, 15
  ST_higher_taxa_codes_12th, 16

code_to_level, 2

decompose_taxon_code, 3, 12, 17, 18

explainST, 4

FormativeElements, 4

get_ST_features, 9
getChildTaxa, 5
getLastChildTaxon, 6
getParentTaxa, 7
getTaxonAtLevel, 8
GreatGroupFormativeElements (FormativeElements), 4

isValidST, 10

level_to_taxon, 11

OrderFormativeElements (FormativeElements), 4

parse_family, 11
preceding_taxon_codes, 3, 12, 17, 18

relative_taxon_code_position, 13

ST, 14
ST_family_classes, 14
ST_features, 15
ST_formative_elements, 15
ST_higher_taxa_codes_12th, 16
ST_unique_list (ST), 14

SubGroupFormativeElements (FormativeElements), 4
SubOrderFormativeElements (FormativeElements), 4

taxon_code_to_taxon, 3, 12, 16, 18
taxon_to_level, 17
taxon_to_taxon_code, 3, 12, 17, 18