Package ‘klassR’

October 18, 2021

Type Package
Title Classifications and Codelists for Statistics Norway
Version 0.1.7
Author Susie Jentoft, Lisa Li, Diana-Cristina Iancu
Maintainer Susie Jentoft <susie.jentoft@ssb.no>
Description Functions to search, retrieve and apply classifications and codelists using Statistics Norway's API <https://www.ssb.no/klass> from the system 'KLASS'. Retrieves classifications by date with options to choose language, hierarchical level and formatting.

Imports tm, httr, jsonlite

BugReports https://github.com/statisticsnorway/klassR/issues
License Apache License 2.0
Encoding UTF-8
LazyData true
RoxygenNote 7.1.1
Suggests testthat (>= 3.0.0)
Config/testthat/edition 3
NeedsCompilation no
Repository CRAN
Date/Publication 2021-10-18 09:10:02 UTC

R topics documented:

ApplyKlass ................................................................. 2
CheckDate ............................................................... 3
check_connect ............................................................ 3
ConvertTable ............................................................. 4
CorrespondList .......................................................... 5
formatting ................................................................. 5
GetFamily ................................................................. 6
ApplyKlass

Match and convert a classification

Description

Match and convert a classification

Usage

ApplyKlass(
  x,
  klass,
  date = NULL,
  variant = NULL,
  correspond = NULL,
  language = "nb",
  output_level = NULL,
  output = "name",
  format = TRUE
)

Arguments

x Input vector
klass Classification number
date Date for classification (format = "YYYY-mm-dd"). Default is current date
variant The classification variant to fetch (if a variant is wanted).
correspond: ID number for target in correspondence table. For correspondence between two dates within the same classification, use correspond = TRUE.

language: Default "nb" for Norwegian (Bokmål). Also "nn" (Nynorsk) and "en" (English available for some classifications)

output_level: Desired output level

output: String describing output. May be "name" (default), "code" or "both".

format: Logical for whether to run formatting av input vector x (Default = TRUE), important to check if formatting is in one level.

Value
A vector or data frame is returned with names and/or code of the desired output level.

Examples
```
data(klassdata)
kommune_names <- ApplyKlass(x = klassdata$kommune, klass = 131, language = "en", format=FALSE)
```

CheckDate

*Internal function to check date*

Description
Internal function to check date

Usage
```
CheckDate(date)
```

Arguments

```- date: Date```

check_connect

*Check connection Function to check that a connection to data.ssb.no is able to be established*

Description
Check connection Function to check that a connection to data.ssb.no is able to be established

Usage
```
check_connect(url)
```
Arguments

url String url address for connection to check

Value

Nothing is returned but a error or warning message is return if no connection is available

ConvertTable

Code level convert table (internal function)

Description

Code level convert table (internal function)

Usage

ConvertTable(
    klass_data,
    code = "code",
    parentcode = "parentCode",
    level = "level"
)

Arguments

klass_data Klass data frame to convert
code Name of variable containing code (Default = "code")
parentcode Name of variable of parent code (Default = "parentCode")
level Name of variable containing level (Default = "level")

Value

A dataframe
**CorrespondList**

Correspondence list
Print a list of correspondence tables for a given klass with source and target IDs

**Description**

Correspondence list Print a list of correspondence tables for a given klass with source and target IDs

**Usage**

CorrespondList(klass, date = NULL)

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>klass</td>
<td>Classification number</td>
</tr>
<tr>
<td>date</td>
<td>Date for classification (format = &quot;YYYY-mm-dd&quot;). Default is current date</td>
</tr>
</tbody>
</table>

**Value**

Data frame with list of correspondence tables, source ID and target ID.

**Examples**

CorrespondList("7")

---

**formattering**

Convert vector to the right format

**Description**

Convert vector to the right format

**Usage**

formattering(x, input_level, klass, klass_data)

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>- vector of character</td>
</tr>
<tr>
<td>input_level</td>
<td>- which classification level</td>
</tr>
<tr>
<td>klass</td>
<td>- classification number</td>
</tr>
<tr>
<td>klass_data</td>
<td>- the right formatting to the classification levels</td>
</tr>
</tbody>
</table>

---
GetKlass

Value

vector of character

Examples

```r
klass_data <- GetKlass(klass = "6", date = "2007-01-01")
input_level <- levelCheck(x = klassdata$nace5, klass_data = klass_data)
formattering(x = klassdata$nace5, input_level = input_level, klass = 6, klass_data=klass_data)
```

GetFamily

Identify corresponding family from a classification number

Description

Identify corresponding family from a classification number

Usage

```r
GetFamily(klass)
```

Arguments

- **klass**: Classification number

Value

Family number

Examples

```r
GetFamily(klass = 7)
```

GetKlass

Fetch classification data Fetch Statistics Norway classification data using API

Description

Fetch classification data Fetch Statistics Norway classification data using API
GetKlass

Usage

GetKlass(
  klass,
  date = NULL,
  correspond = NULL,
  variant = NULL,
  output_level = NULL,
  language = "nb",
  output_style = "normal"
)

Arguments

klass  Number/string of the classification ID/number. (use Klass_list() to find this)
date   String for the required date of the classification. Format must be "yyyy-mm-dd". For an interval, provide two dates as a vector. If blank, will default to today’s date.
correspond  Number/string of the target correspondence (if a correspondence table is requested).
variant  The classification variant to fetch (if a variant is wanted).
output_level  Number/string specifying the requested hierarchy level (optional).
language   Two letter string for the requested language output. Default is bokmål ("nb"). Nynorsk ("nn") and English ("en") also available for some classification.
output_style  String variable for the output type. Default is "normal". Specify "wide" for a wide formatted table output.

Value

The function returns a data frame of the specified classification/correspondence table. Output variables include: code, parentCode, level, and name for standard lists. For correspondence tables variables include: sourceCode, sourceName, targetCode and targetName. For time correspondence tables variables include: oldCode, oldName, newCode and newName. For "wide" output, code and name with level suffixes is specified.

Examples

# Get classification for occupation classifications
head(GetKlass(klass = "7"))
# Get classification for occupation classifications in English
head(GetKlass(klass = "7", language = "en"))
**GetName**

*Get the name of a classification version*

**Description**

Get the name of a classification version

**Usage**

`GetName(version)`

**Arguments**

- **version**
  - Version number

**Value**

string or vector of strings with name of version

**Examples**

`GetName("33")`

---

**GetNums**

*Get target ID numbers from Url*

**Description**

Get target ID numbers from Url

**Usage**

`GetNums(x)`

**Arguments**

- **x**
  - Url address

**Value**

Number
**GetUrl**

Get json file from Url

**Usage**

GetUrl(url)

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>String url address</td>
</tr>
</tbody>
</table>

**Value**

text in json format

**GetUrl2**

Get json file from Url - alternative version

**Usage**

GetUrl2(url, check = TRUE)

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>String url address</td>
</tr>
<tr>
<td>check</td>
<td>Logical parameter on whether to check if the url exists</td>
</tr>
</tbody>
</table>

**Value**

text in json format
**GetVersion**  
*Get version number of a class given a date*

**Description**
Get version number of a class given a date

**Usage**
GetVersion(klass = NULL, date = NULL, family = NULL, klassNr = FALSE)

**Arguments**
- **klass**: Classification number
- **date**: Date for version to be valid
- **family**: Family ID number if a list of version number for all classes is desired
- **klassNr**: True/False for whether to output classification numbers. Default = FALSE

**Value**
Number, vector or data frame with version numbers and classification numbers if specified.

**Examples**
GetVersion(7)

---

**get_variant_name**  
*Get variant name Internal function for fetching the variant name based on the number*

**Description**
Get variant name Internal function for fetching the variant name based on the number

**Usage**
get_variant_name(variant)

**Arguments**
- **variant**: The variant number
klassdata

Description
A dataset containing variables for testing of Statistics Norway's classification API with the klassR package. Some observations are missing or incorrect for testing and demonstrations.

Usage
klassdata

Format
A data frame containing 100 rows and 7 variables:

- **ID**: Identification number
- **sex**: 1/2 variable for sex
- **education**: 4-digit number for education standard ISCED97 (level and subject area) NUS (klass = 66) 2015.01.01
- **kommune**: 4-digit code for Norwegian municipality (klass = 131). Based on 2015.01.01
- **kommune2**: Numeric variable for Norwegian municipality with dropped leading zero's for testing (klass = 131). Based on 2015.01.01
- **nace5**: 5-digit code for industry (NACE). Based on 01.01.2015 standard industry codes (klass = 7)
- **occupation**: 4-digit occupation codes using standard for STYRK-08 (klass = 7) 2015.01.01

levelCheck

Description
The input level is calculated as the mode (most common) hierarchical level in the input vector.

Usage
levelCheck(x, klass_data)

Arguments
- **x**: vector of character
- **klass_data**: - the right formatting to the classification levels

Value
The hierarchical level of the input data is returned.
Examples

```r
data(klassdata)

sn <- GetKlass(klass = "6", date = "2007-01-01")
levelCheck(x = klassdata$nace5, klass_data = sn)
```

<table>
<thead>
<tr>
<th>Levels</th>
<th>Title</th>
</tr>
</thead>
</table>

Description

Title

Usage

```r
Levels(input_level, output_level, klass_data)
```

Arguments

- `input_level`: Classification level from the original dataset.
- `output_level`: Classification level for which the codes and names are desired by the user.
- `klass_data`: Classification file retrieved from KLASS.

Value

Data frame with the input and extra desired classification levels. Includes codes and names for each level.

Examples

```r
klass_data <- GetKlass(klass = "6")
Levels(input_level = 5, output_level = 2, klass_data = klass_data)[1:10, ]
Levels(input_level = 5, output_level = c(2, 3), klass_data = klass_data)[1:10, ]
```

ListFamily

Classification family list Print a list of all families and the number of classifications in each

Description

Classification family list Print a list of all families and the number of classifications in each

Usage

```r
ListFamily(family = NULL, codelists = FALSE, language = "nn")
```
Arguments

family  Input family ID number to get a list of classifications in that family

codelists  True/False for whether to include codelists. Default = FALSE

language  Two letter string for the requested language output. Default is Bokmål ("nb"). Nynorsk ("nn") and English ("en").

Value

dataset containing a list of families

Examples

ListFamily(family = 1)

---

ListKlass  Classification list Get a full list of all classifications and codelists

Description

Classification list Get a full list of all classifications and codelists

Usage

ListKlass(codelists = FALSE, language = "nb")

Arguments

codelists  True/False for whether to include codelists. Default = FALSE

language  Two letter string for the requested language output. Default is Bokmål ("nb"). Nynorsk ("nn") and English ("en").

Value

A data frame containing a full list of classifications. The data frame includes the classification name, number, family and type.

Examples

head(ListKlass(codelists = TRUE))
MakeChar

Conversion to character

Description
Conversion to character

Usage
MakeChar(x)

Arguments
x
a number or vector of numbers

Value
x converted to a string or vector of strings.

MakeUrl

Internal function to create URL address

Description
Internal function to create URL address

Usage
MakeUrl(
    klass,
    correspond = NULL,
    variant_name = NULL,
    type = "vanlig",
    fratil = FALSE,
    date = NULL,
    output_level_coding = NULL,
    language_coding = NULL
)
SearchKlass

**Arguments**

- **klass**: Classification number
- **correspond**: Target number for correspondence table
- **variant_name**: The name of the variant of the classification
- **type**: String describing type. "vanlig" for normal classification and "kor" for correspondence. Default = "vanlig"
- **fratil**: True/False for whether a date interval is to be used. Default = False
- **date**: Date(s) for classification
- **output_level_coding**: Coding for output level
- **language_coding**: Coding for language

**Value**

- String url adress

---

**Description**

Search Klass

**Usage**

```r
SearchKlass(query, codelists = FALSE, size = 20)
```

**Arguments**

- **query**: String with key word to search for
- **codelists**: True/False for whether to include codelists. Default = FALSE
- **size**: The number of results to show. Default = 20.

**Value**

- Data frame of possible classifications that match the query

**Examples**

```r
SearchKlass("yrke")
```
splitChar

*insert missing dots to the right place in a string*

**Description**

insert missing dots to the right place in a string

**Usage**

`splitChar(x, dot)`

**Arguments**

- `x` - character with missing dots
- `dot` - the place of missing dots

**Value**

a string is returned with the insertion of a formatted period (.) in the specified location.

---

**stop_quietly**

*Stop quietly function Stop from a function without an error. Used for stopping when no internet*

**Description**

Stop quietly function Stop from a function without an error. Used for stopping when no internet

**Usage**

`stop_quietly()`
Index

* datasets
  klassdata, 11

ApplyKlass, 2
check_connect, 3
CheckDate, 3
ConvertTable, 4
CorrespondList, 5

formattering, 5
get_variant_name, 10
GetFamily, 6
GetKlass, 6
GetName, 8
GetNums, 8
GetUrl, 9
GetUrl2, 9
GetVersion, 10

klassdata, 11
levelCheck, 11
Levels, 12
ListFamily, 12
ListKlass, 13

MakeChar, 14
MakeUrl, 14

SearchKlass, 15
splitChar, 16
stop_quietly, 16