Package 'rocnp'

October 14, 2022

Title Work with Romanian Personal Numeric Codes PNC / CNP

Version 0.1.0

Description A set of tools for working with Romanian personal numeric codes. The core is a validation function which applies several verification criteria to assess the validity of numeric codes. This is accompanied by functionality for extracting the different components of a personal numeric code. A personal numeric code is issued to all Romanian residents either at birth or when they obtain a residence permit.

License MIT + file LICENSE

URL https://github.com/dragosmg/rocnp

BugReports https://github.com/dragosmg/rocnp/issues

Depends R (>= 3.5.0)

Imports dplyr, glue, magrittr, purrr, rlang, stringr, tibble

Suggests covr, testthat

Encoding UTF-8

RoxygenNote 7.1.2

NeedsCompilation no

Author Dragoș Moldovan-Grünfeld [aut, cre, cph]

Maintainer Dragoș Moldovan-Grünfeld <dragos.mold@gmail.com>

Repository CRAN

Date/Publication 2021-11-05 08:10:05 UTC

R topics documented:

check_cnp_is_valid 2
decompose_cnp 2
get_birth_month 3
get_birth_year 4
get_county 4
get_sex 5
get_status 6

Index 7

`check_cnp_is_valid` *Check if a CNP is valid*

Description

Check if a CNP is valid

Usage

```
check_cnp_is_valid(cnp)
```

Arguments

`cnp`	a 13-digit number (or a vector of 13-digit numbers) representing the Personal Numeric Code - CNP

Value

a logical vector (`TRUE`, `FALSE` or `NA`)

Examples

```
check_cnp_is_valid(1940616346114)
check_cnp_is_valid(7041218318525)
check_cnp_is_valid(62012060180783)
check_cnp_is_valid(NA)
```

`decompose_cnp` *Split the CNP into its components*

Description

A CNP is made up of 13 digits each with a specific meaning: * `S` - the first digit is indicative of gender * `AA` - digits 2-3 represent the last 2 digits of the year of birth * `LL` - digits 4-5 represent the month of birth * `ZZ` - digits 6-7 represent the day of birth * `JJ` - digits 8-9 represent the county of birth (for native Romanians) or the county of issue (for residents) * `NNN`- digits 10-12 number between 001 and 999 used to differentiate between people born on the same day in the same county * `C` - digit 13 checksum

Usage

```
decompose_cnp(cnp)
```

Arguments

cnp	a 13-digit number (or a vector of 13-digit numbers) representing the Personal Numeric Code - CNP

Value

a named character vector splitting the CNP into its components.

Examples

```
decompose_cnp(6201206018078)
decompose_cnp(5201206346491)
```

---

get_birth_month *Extract the Month of Birth from the Personal Numeric Code*

Description

This function extracts the month of birth starting from the "LL" component.

Usage

```
get_birth_month(cnp)
```

Arguments

cnp	a 13-digit number (or a vector of 13-digit numbers) representing the Personal Numeric Code - CNP

Value

a numeric vector representing the month of birth

Examples

```
get_birth_month(1940616346114)
get_birth_month(7041218318525)
cnps <- c(5201206346491, 1940616346114, 7041218318525, 6201206018078)
get_birth_month(cnps)
get_birth_month(c(5201206346491, 1940616346114, 7041218318525, NA))
```

get_birth_year *Extract Year of Birth from the Personal Numeric Code*

Description

This function infers the year of birth based on the code for sex - which is also indicative of the century of birth - and the AA component denoting the final 2 digits of the birth year. The year is returned as string, especially since for residents not born in Romania, there might be two possible valid values for the birth year - in this case the year is returned as "__yy"

Usage

```
get_birth_year(cnp)
```

Arguments

cnp	a 13-digit number (or a vector of 13-digit numbers) representing the Personal Numeric Code - CNP

Value

a character vector representing the year of birth (the century is unknown for non-natives)

Examples

```
get_birth_year(1940616346114)
get_birth_year(7041218318525)
```

get_county *Extract County of Issue from the Personal Numeric Code*

Description

Extract County of Issue from the Personal Numeric Code

Usage

```
get_county(cnp)
```

Arguments

cnp	a 13-digit number (or a vector of 13-digit numbers) representing the Personal Numeric Code - CNP

Value

a string representing the name of the county where the CNP was issued

Examples

```
get_county(6201206018078)
get_county(5201206346491)
get_county(1940616346114)
get_county(7041218318525)
```

`get_sex` *Extract the Sex from the Personal Numeric Code*

Description

This function extracts the sex based on the sex component `S` of the CNP. It is worth noting that, at the moment of the implementation, Romanian authorities define sex as binary. In the event of a sex change a new CNP could be issued. The function returns an error if there is at least one invalid CNP in the input vector, forcing you to confront the issue early. The easiest way to get around this is to use `check_cnp_is_valid()`.

Usage

```
get_sex(cnp)
```

Arguments

`cnp`	a 13-digit number (or a vector of 13-digit numbers) representing the Personal Numeric Code - CNP

Value

a character vector of the recorded sex: `M`, `F` (if the CNP is valid) or `NA_character` if the CNP is missing

Examples

```
get_sex(1940616346114)
get_sex(7041218318525)
get_sex(6201206018078)
get_sex(5201206346491)
get_sex(c(5201206346491, 1940616346114, 7041218318525, 6201206018078))
get_sex(c(5201206346491, 1940616346114, 7041218318525, NA))
```

`get_status` *Extract Residence Status from the Personal Numeric Code*

Description

Extract Residence Status from the Personal Numeric Code

Usage

```
get_status(cnp, lang = c("RO", "EN"))
```

Arguments

`cnp`	a 13-digit number (or a vector of 13-digit numbers) representing the Personal Numeric Code - CNP
`lang`	a string denoting the language for the status - either `RO` for Romanian (the default) or `EN` for English.

Value

a string denoting residence status in the language of choice

Examples

```
get_status(6201206018078)
get_status(5201206346491)
get_status(1940616346114)
get_status(1940616346114, lang = "EN")
get_status(7041218318525)
get_status(7041218318525, lang = "EN")
```

Index

check_cnp_is_valid, 2

decompose_cnp, 2

get_birth_month, 3
get_birth_year, 4
get_county, 4
get_sex, 5
get_status, 6