The Journal of Open Source Software

regioncode: Convert Region Names and Division Codes of China Over Years

Yue Hu[1]**, Yufei Sun**[1]**, and Wenquan Wu**[1]

1 Department of Political Science, Tsinghua University

DOI: DOIunavailable

Software

- Review
- Repository
- Archive

Editor: Pending Editor

Reviewers:

- @Pending Reviewers

Submitted: N/A
Published: N/A

License
Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License (CC BY 4.0).

Summary

The Chinese government gives unique geocodes for each county, city (prefecture), and provincial-level administrative unit. The so-called "administrative division codes" were consistently adjusted to matched national and regional plans of development. Geocode adjustments disturb researchers when they merge data with different versions of geocodes or region names. Especially, when researchers render statistical data on a Chinese map, different geocodes between map data and statistical data may cause mess-up data output or visualization.

The package is developed to conquer such difficulties to match regional data across years more conveniently and correctly. Inspired by Vincent Arel-Bundock's well-known `countrycode` (Arel-Bundock et al., 2018), we created `regioncode` to achieve similar functions specifically for China studies. `regioncode` enables seamlessly converting formal names, common-used names, and division codes of Chinese prefecture regions between each other and across thirty-four years from 1986 to 2019.

Basic Usage

We uses a randomly drawn sample of Yuhua Wang's `China's Corruption Investigations Dataset` to illustrate how the package works.

In `regioncode` package, we named administrative division codes as `code`, regions' formal names as `name`, and their commonly used abbreviation as `sname`. In the current version, the function can produce three types of output at both the prefectural and provincial levels: codes (`code`), names (`name`) and area (`area`).One just needs to specify which type of the output they want in the argument `convert_to` and corresponding years of the input and output.

For example, the following codes convert the 2019 geocodes in the `corruption` data to their 1989 version:

```r
library(regioncode)

data("corruption")

# Original 2019 version
corruption$prefecture_id
```

```
##  [1] 370100 321200 310117 420500 451300 431200 350300 511500 469021 420600
```

```r
# 1999 version
regioncode(data_input = corruption$prefecture_id,
           convert_to = "code", # default set
           year_from = 2019,
           year_to = 1989)
```

```
##  [1] 370100 329001 310227 420500 422700 452200 433000 350300 512500 460025
## [11] 420600
```

Note that if a region was initially geocoded in e.g., 1989 and included in a new region, in 2019, the new region geocode will be used hereafter.If a big place was broken into several regions, the later-year codes will be aligned with the first region according to the ascendant order of the regions' numeric geocodes.

By altering the output format to `name`, one can easily convert codes or region names of a given year to region names in another year.`regioncode` automatically detects the input format, so users need to specify the *output* format only (together with the input and output years) to gain what they want.

```
# The original name
corruption$prefecture
```

```
##  [1] "济南市" "泰州市" "松江区" "宜昌市" "来宾市" "怀化市" "莆田市" "宜宾市"
##  [9] "定安县" "襄阳市"
```

```
# Codes to name

regioncode(data_input = corruption$prefecture_id,
           convert_to = "name",
           year_from = 2019,
           year_to = 1989)
```

```
##  [1] "济南市"   "泰州市"   "松江县"   "宜昌市"   "宜昌地区" "柳州地区"
##  [7] "怀化地区" "莆田市"   "宜宾地区" "定安县"   "襄樊市"
```

```
# Name to codes of the same year

regioncode(data_input = corruption$prefecture,
           convert_to = "code",
           year_from = 2019,
           year_to = 2019)
```

```
##  [1] 370100 321200 310117 420500 451300 431200 350300 511500 469021 420600
```

Advanced Applications

To further help uses with "messier" data and diverse demands, `regioncode` provides five special conversions: conversion from data with incomplete data, specification of municipalities, conversion sociopolitical areas and linguistic areas, and pinyin output.

Incomplete naming prefectures.

More than often, data codes may omit the administrative level when recording geo-information. To accomplish conversions of such data, one needs to specify the `incomplete_name` argument. If the input data is incomplete, users should set the argument as "from." Optional values also include "to" and ' "both".'

```
# Full, official names
corruption$prefecture
```

```
##  [1] "济南市" "泰州市" "松江区" "宜昌市" "来宾市" "怀化市" "莆田市" "宜宾市"
##  [9] "定安县" "襄阳市"
```

```
regioncode(data_input = corruption$prefecture,
           convert_to = "name",
           year_from = 2019,
           year_to = 1989,
           incomplete_name = "to")
```

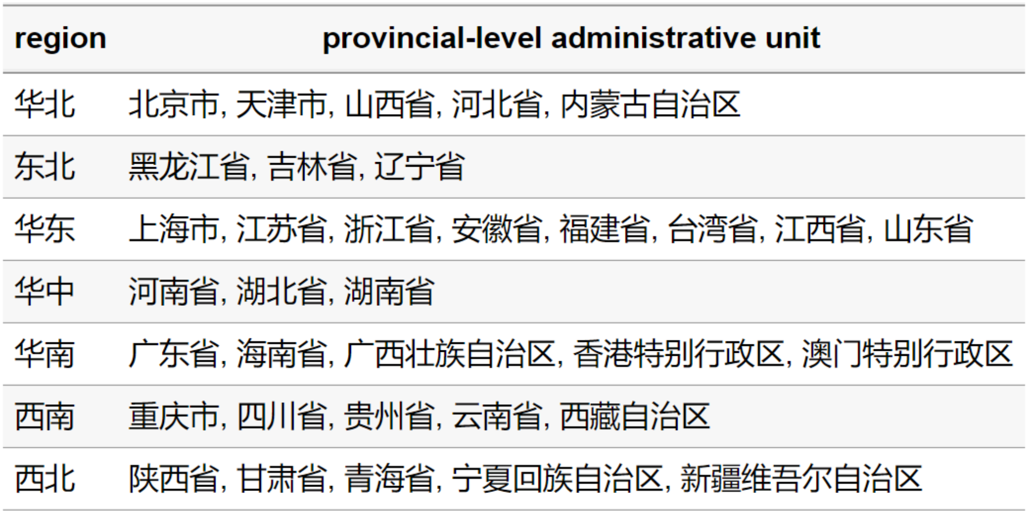
```
##  [1] "济南" "泰州" "松江" "宜昌" "柳州" "来宾" "怀化" "莆田" "宜宾" "定安"
## [11] "襄樊" "襄阳"
```

Municipalities

Municipalities (named "zhixiashi" in Chinese Pinyin) are geographically cities but administratively provincial. `regioncode` sets an argument `zhixiashi`. When the argument is set `TRUE`, the municipalities are treated as whole prefectures, and their provincial codes are used as the geocodes.

Sociopolitical and Linguistic Areas

Due to social, political, and martial reasons, Chinese regions are divided into eight regions:



region	provincial-level administrative unit
华北	北京市, 天津市, 山西省, 河北省, 内蒙古自治区
东北	黑龙江省, 吉林省, 辽宁省
华东	上海市, 江苏省, 浙江省, 安徽省, 福建省, 台湾省, 江西省, 山东省
华中	河南省, 湖北省, 湖南省
华南	广东省, 海南省, 广西壮族自治区, 香港特别行政区, 澳门特别行政区
西南	重庆市, 四川省, 贵州省, 云南省, 西藏自治区
西北	陕西省, 甘肃省, 青海省, 宁夏回族自治区, 新疆维吾尔自治区

`regioncode` also offers a method "area" to convert codes and names of the region into such areas.

```
regioncode(data_input = corruption$prefecture,
           year_from = 2019,
           year_to = 1989,
           convert_to="area")
```

```
##  [1] "华东" "华东" "华东" "华中" "华南" "华中" "华东" "西南" "华南" "华中"
```

China is a multilingual country with a variety of dialects. These dialects may be used by several prefectures in a province or province. Prefectures from different provinces may also share the same dialect. `regioncode` allows users to gain linguistic zones the prefectures belong as an output. Users can gain two levels of linguistic zones, dialect groups and dialect sub-groups by setting the argument `to_pinyin` to `dia_group` or `dia_sub_group`.

```r
regioncode(data_input = corruption$prefecture,
           year_from = 2019,
           year_to = 1989,
           to_dialect = "dia_group")
```

```
## [1] "冀鲁官话" "江淮官话" "吴语"     "西南官话" "西南官话" "湘语"
## [7] "莆仙区"   "西南官话" "琼文区"   "西南官话"
```

Pinyin

Pinyin is a Chinese phonetic romanization. Some data stores the region names with pinyin instead of Chinese characters. The default name output of `regioncode` uses Chinese characters, but one can gain pinyin output by setting the argument `to_pinyin = TRUE`.

```r
regioncode(data_input = corruption$prefecture,
           year_from = 2019,
           year_to = 1989,
           convert_to="name",
           to_pinyin=TRUE
           )
```

```
##           济南市           泰州市            松江县         宜昌市
##     "ji_nan_shi"   "tai_zhou_shi" "song_jiang_xian"  "yi_chang_shi"
##         宜昌地区         柳州地区          怀化地区         莆田市
## "yi_chang_di_qu" "liu_zhou_di_qu"  "huai_hua_di_qu"   "pu_tian_shi"
##         宜宾地区           定安县            襄樊市
##   "yi_bin_di_qu"   "ding_an_xian"   "xiang_fan_shi"
```

Provinces

`regioncode` allows conversions at not only the prefectural but provincial level. By setting the argument `province = TRUE`, users can accomplish all the code, name, and area conversions at the provincial level. When the inputs are abbreviations, users can set the `convert_to` argument to `abbreTocode`, `abbreToname`, or `abbreToarea`.

```r
regioncode(data_input = corruption$province_id,
           convert_to = "codeToabbre",
           year_from = 2019,
           year_to = 1989,
           province = TRUE)
```

```
## [1] "鲁" "苏" "沪" "鄂" "桂" "湘" "闽" "蜀" "琼" "鄂"
```

Acknowledgements

We acknowledge contributions from Meng Zhu, Xueyan Liu, Yuyang Shi, Yujia Xu, and Yuxin Pan, Haiting Tian, Weihang Shao, and Yuanqian Chen.

References

Arel-Bundock, V., Enevoldsen, N., & Yetman, C. (2018). Countrycode: An r package to convert country names and country codes. *Journal of Open Source Software*, *3*(28), 848. https://doi.org/10.21105/joss.00848